Brunel University London

Department of Mathematics

College of Engineering, Design and Physical Sciences

LOCATING THE SOURCE OF AN ACOUSTIC WAVE EQUATION USING LIKELIHOOD ESTIMATES FROM THE KALMAN FILTER APPLIED TO SURFACE READINGS

A thesis submitted for the degree of Doctor of Philosophy in the field of Mathematics

April 2020

Author:

Matthew Peter Francis Elliott-Sands

Supervisory Team:

Dr Simon Shaw and Dr Paresh Date

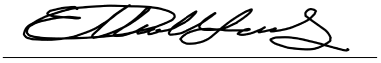
Declaration of Authorship

I, Matthew Elliott-Sands, declare that this thesis titled

Locating the source of an acoustic wave equation using likelihood estimates from the Kalman filter applied to surface readings

is my work. On occasions where work other than my own has been included, it has been referenced accordingly. Furthermore, I can declare that this thesis has not previously been submitted at another degree-granting institution or Brunel University London, for any other degree. I confirm that the contents of this thesis will not be submitted to another degree-granting institution until the outcome of this submission is known.

DATE: Wednesday 15 April 2020

SIGNATURE OF AUTHOR: 

Abstract

Cardiovascular disease (CVD) was the second-largest cause of death in the United Kingdom in 2014 [1], accounting for 32% of all deaths in 2009 [2]. CVD encompasses many diseases, one of which is coronary artery disease (CAD), otherwise known as atherosclerosis. Atherosclerosis is the build-up of fatty material, called plaque, inside the wall of the artery. Over time, this plaque will grow too large or break off, causing a blockage resulting in a heart attack. Currently, mortality from CAD has decreased by 72% between 1979 and 2013 [3]. However, predictions show that if the increasing trend of Body Mass Index (BMI) continues, then mortality from CAD could start increasing again [4]. There are several different methods currently available to the National Health Service (NHS) to diagnose CAD. However, there are long waiting lists and expensive costs associated with current diagnosis methods.

Our aim is to look at a non-invasive approach of diagnosing CAD. We have limited our investigation to simple model problems. Therefore, further work would be required to consider more complex cases which align with the real-world application.

In this thesis, we consider both 1-dimensional (1D) and 2-dimensional (2D) problems modelled by an acoustic wave equation with a forcing function which attempts to emulate a localised disturbance caused by CAD. We use an explicit finite difference method (FDM) to approximate the solution in our partial differential equation (PDE) and discard the disturbance location used. Having added noise to these approximations in an attempt to mimic noise from real readings, we record these approximations at specific locations on the surface of our domains to imitate data collected from actual sensors. Using this data in the Kalman filter (KF), where guesses for the disturbance location are made, we can estimate the approximation of u throughout our domain. Using data generated by the KF, we compute likelihood estimates for each guess made and obtain the most probable disturbance location used to generate our sensor readings.

Acknowledgements

First and foremost, I would like to express my gratitude towards my principal supervisor, Simon Shaw. Had it not been for his patience, professional guidance and intellectual integrity, this thesis would not have been possible. In addition to this, I would like to take this opportunity to thank him for letting me complete this thesis under his supervision - it has been an experience I will never forget.

I would also like to extend my gratitude to my second supervisor, Paresh Date, for his expert knowledge of filtering techniques, without which this thesis would not have been possible.

Furthermore, I would like to thank Martin Greenhow, who encouraged me to pursue a PhD and Julie Bradshaw for her invaluable advice and professionalism during my time spent as a doctoral researcher.

I want to thank the Engineering and Physical Science Research Council for funding my studies for 3.5 years, without which I would not have been able to undertake this research.

Finally, but most certainly not least, I would like to thank my family and friends for their constant support throughout my studies. In particular, I would like to thank Lauren Strathearn for her persistence and drive to widen my horizons towards research which has undoubtedly improved the quality of this thesis. Also, I would like to thank some fellow PhD candidates: Philip Woodhead, Arianna Salili-James and Farukh Mukhamedov, for on-going encouragement and the availability to meet for a chat.

Table of Contents

List of Figures

List of Tables

List of Mathematical Symbols

Symbol	Description
u	The exact solution of our PDE (acoustic wave equation) used in all model problems.
u_x, u_{xx}, u_{xxx}	First, second and third partial derivatives of u with respect to x.
u_y, u_{yy}, u_{yyy}	First, second and third partial derivatives of u with respect to y.
u_t, u_{tt}	The first and second partial derivatives of u with respect to time.
f	The forcing function used in our model problems.
p	The variable used to denote the generic dimension of our 1D and 2D model problems, that is $\mathbb{R}^p$ where $p \in \{1, 2\}$.
Ω	The variable used to denote our generic domain.
Γ_{D}	The variable used to denote our generic Dirichlet boundary.
Γ_{N}	The variable used to denote our generic Neumann boundary.
T	The total simulation duration, for all model problems.
N	The number of intervals in the x-direction which arise from discretising our 1D and 2D domains.
L	The number of intervals, in time, arising from discretising different model problems.
H_x	Size of the spatial intervals, in the x-direction, for our 1D and 2D model problems.
Δt	Size of the intervals in time for our 1D and 2D model problems.
t_n	Represents a specific discrete-time step used in our 1D and 2D model problems denoted by $t_n = n\Delta t$ where $n \in \{0, 1, 2, ..., L\}$.
x_i	Represents a specific spatial node in our mesh for our 1D model problem denoted by $x_i = (i-1)H_x$ where $i \in \{1, 2, ..., N+1\}$.

Symbol	Description
$u(x_i, t_n)$	The exact solution of our 1D acoustic wave equation at the spatial node x_i, and the discrete-time step denoted by t_n where $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, 2, ..., L\}$.
$f(x_i, t_n)$	The forcing function used in our 1D model problem at the spatial node x_i and discrete-time step denoted by t_n where $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, 2, ..., L\}$.
A	The amplitude of our forcing function, f.
ϵ	Denotes a small number, used in our forcing function, denoted by f, which controls the Gaussian spread.
F	The frequency of our disturbance, in the forcing function.
x_0, $\underline{\mathbf{x}}_0$	Denote the disturbance location in our 1D and 2D model problems, respectively. This is known when generating the explicit FDM approximations of u, whereas in the KF this is unknown.
V_i^n	Denotes the explicit FDM approximation of $u(x_i, t_n)$ where $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, 2, ..., L\}$.
M	The number of intervals, in the y-direction, which arises from discretising our 2D domain.
H_y	Size of the spatial intervals, in the y-direction, for our 2D model problems.
(x_i, y_j)	Represents a spatial node for our 2D model problems denoted by $x_i = (i-1)H_x$ where $i \in \{1, 2, ..., N+1\}$ and $y_j = (j-1)H_y$ where $j \in \{1, 2, ..., M+1\}$.
$u(x_i, y_j, t_n)$	The exact solution of our 2D acoustic wave equation at the spatial node (x_i, y_j) and discrete-time step denoted by t_n where $i \in \{1, 2, ..., N+1\}$, $j \in \{1, 2, ..., M+1\}$ and $n \in \{0, 1, 2, ..., L\}$.
$f(x_i, y_j, t_n)$	The forcing function used in our 2D model problems at the spatial node (x_i, y_j) and the discrete-time step denoted by t_n where $i \in \{1, 2, ..., N+1\}$, $j \in \{1, 2, ..., M+1\}$ and $n \in \{0, 1, 2, ..., L\}$.
$V_{i,j}^n$	Denotes the explicit FDM approximation of $u(x_i, y_j, t_n)$ where $i \in \{1, 2, ..., N+1\}$, $j \in \{1, 2, ..., M+1\}$ and $n \in \{0, 1, 2, ..., L\}$.

Symbol	Description
Ω_{1D}	The domain used in our 1D model problem.
$\partial\Omega_{1D}$	The entire boundary used in our 1D model problem.
Γ_{N_1}	The Neumann boundary, a subset of $\partial\Omega_{1D}$.
Γ_{D_1}	The Dirichlet boundary, a subset of $\partial\Omega_{1D}$.
Ω_{2D}	The domain used in our 2D model problems.
$\partial\Omega_{2D_1}$	The entire boundary used in our first 2D model problem.
$\partial\Omega_{2D_2}$	The entire boundary used in our second 2D model problem.
Γ_{D_2}	The Dirichlet boundary, a subset of $\partial\Omega_{2D_1}$.
Γ_{N_2}	A Neumann boundary, a subset of $\partial\Omega_{2D_1}$ or $\partial\Omega_{2D_2}$.
Γ_{N_3}	A Neumann boundary, a subset of $\partial\Omega_{2D_2}$.
Γ_{N_4}	A Neumann boundary, a subset of $\partial\Omega_{2D_2}$.
Γ_{D_3}	A Dirichlet boundary, a subset of $\partial\Omega_{2D_2}$.
$\mathbf{V}^n$	A vector containing V_i^n or $V_{i,j}^n$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$ at the discrete-time step denoted by t_n where $n \in \{0, 1, 2, ..., L\}$ for our 1D and 2D model problems, respectively.
$\boldsymbol{\mathcal{F}}^n$	A vector containing $f(x_i, t_n)$ or $f(x_i, y_j, t_n)$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$ at the discrete-time step denoted by t_n where $n \in \{0, 1, 2, ..., L\}$ for our 1D and 2D models, respectively.
$\mathbf{B}$	A matrix containing the coefficients of the variables which arise from our explicit FDM approximation of u in our 1D and 2D model problems. These include V_i^n or $V_{i,j}^n$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$ at the discrete-time step denoted by t_n where $n \in \{0, 1, 2, ..., L\}$ for our 1D and 2D models, respectively..
$\tilde{\mathbf{x}}^{n+1}$	A vector containing $\mathbf{V}^n$ which is predicted using the Kalman filter (KF).
$\mathbf{G}$, $\mathbf{g}^n$	A matrix and vector used in the standard KF, which together map the previous state, $\tilde{\mathbf{x}}^n$, to the next state, denoted by $\tilde{\mathbf{x}}^{n+1}$.
$\mathbf{y}^n$	A vector containing the explicit FDM approximation of u at specific nodes in our meshes on the surface of both domains considered in this thesis.

Symbol	Description
$\mathbf{w}^n$	A vector containing uncorrelated Gaussian noise with a zero mean which we add to our explicit FDM approximations of u. We do this to account for the numerical error which arises due to the explicit FDM approximation of u.
$\mathbf{z}^n$	A vector containing uncorrelated Gaussian noise with a zero mean which is added to our sensor traces, denoted by $\mathbf{y}^n$, which correspond to inaccuracies arising from the recording of observations in real-life scenarios.
$\mathbf{C}$	A rectangular matrix, containing a single one on each row to extract specific explicit FDM approximations of u from $\hat{\mathbf{x}}^{n\|n-1}$ in the KF which correspond to $\mathbf{y}^n$.
μ	Used to symbolise the mean for the noise added, denoted by $\mathbf{w}^n$ and $\mathbf{z}^n$.
$\mathbf{S}^2$	Used to symbolise the variance for the noise added, denoted by $\mathbf{w}^n$ and $\mathbf{z}^n$.
$\mathbf{Q}$	Covariance matrix corresponding to the noise added, denoted by $\mathbf{w}^n$.
$\mathbf{R}$	Covariance matrix corresponding to the noise added, denoted by $\mathbf{z}^n$.
$\hat{\mathbf{x}}^{n+1\|n}$	A vector containing the estimate deduced in the KF of $\tilde{\mathbf{x}}^{n+1}$, otherwise known as the predicted mean in KF literature.
$\mathbf{P}^{n+1\|n}$	A matrix, known in KF literature as the covariance of the predicted mean, contains the covariance between the predicted variable, denoted by $\hat{\mathbf{x}}^{n+1\|n}$, and the variable we are predicting, denoted by $\tilde{\mathbf{x}}^{n+1}$.
$\mathbf{v}^n$	A vector, known as the innovations in KF literature, contains the difference between the sensor traces, denoted by $\mathbf{y}^n$, and the KFs prediction of these, denoted by $\mathbf{C}\hat{\mathbf{x}}^{n\|n-1}$.
$\mathbf{\Sigma}^n$	A matrix, containing what is known as the covariance of the innovations in KF literature.
$\mathbf{K}^n$	Denotes a matrix, which contains what is known as the Kalman Gain in KF literature.

Symbol	Description
$\hat{\mathbf{x}}^{n\|n}$	Denotes a vector, which contains what is known as the updated mean in KF literature.
$\underline{\tilde{\mathbf{x}}}^{n}$	A block vector, containing $\mathbf{V}^{n}$ and $\mathbf{V}^{n-1}$ for the given model problem.
$\mathbf{A}$	A block matrix, containing matrices denoted by $\mathbf{B}$, $-\mathbf{I}$, $\mathbf{I}$ and $\mathbf{0}$ in the top left, top right, bottom left and bottom right respectively.
$\tilde{\boldsymbol{\mathcal{F}}}^{n}$	A block vector, containing $\boldsymbol{\mathcal{F}}^{n}$ for the given model problem, and $\mathbf{0}$ with the same dimensions as $\boldsymbol{\mathcal{F}}^{n}$.
$\tilde{\mathbf{w}}^{n}$	Denotes a block vector, which contains $\mathbf{w}^{n}$ and $\mathbf{0}$, with the same dimensions as $\mathbf{w}^{n}$.
$\tilde{\mathbf{z}}^{n}$	Denotes a block vector, which contains $\mathbf{z}^{n}$ and $\mathbf{0}$, with the same dimensions as $\mathbf{z}^{n}$.
$\tilde{\mathbf{y}}^{n}$	Denotes a block vector, which contains $\mathbf{y}^{n}$ and $\mathbf{y}^{n-1}$, for our model problem.
$\tilde{\mathbf{C}}$	Denotes a block matrix, which contains $\mathbf{C}$ in the top left and bottom right, and $\mathbf{0}$ elsewhere.
$\tilde{\mathbf{Q}}$	Denotes a block matrix, which contains $\mathbf{Q}$ and $\mathbf{0}$, with the same dimensions as $\mathbf{Q}$. This block matrix is the covariance matrix for the noise denoted by $\tilde{\mathbf{w}}^{n}$.
$\tilde{\mathbf{R}}$	Denotes a block matrix, which contains $\mathbf{R}$ and $\mathbf{0}$, with the same dimensions as $\mathbf{R}$. This block matrix is the covariance matrix for the noise denoted by $\tilde{\mathbf{z}}^{n}$.
$\underline{\mathbf{v}}^{n}$	A block vector containing the difference between the sensor traces, denoted by $\tilde{\mathbf{y}}^{n}$, and the KFs prediction of these, denoted by $\tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{n\|n-1}$.
$\underline{\boldsymbol{\Sigma}}^{n}$	Denotes a block matrix, which contains what is known as the covariance of the innovations in KF literature.
$\underline{\mathbf{K}}^{n}$	Denotes a matrix, which contains what is known as the Kalman Gain in KF literature.
$\underline{\hat{\mathbf{x}}}^{n\|n}$	Denotes a block vector, which contains what is known as the updated mean in KF literature.
$\underline{\hat{\mathbf{x}}}^{n+1\|n}$	A block vector containing the estimate deduced in the KF of $\underline{\tilde{\mathbf{x}}}^{n+1}$, otherwise known as the predicted mean in KF literature.

Symbol	Description
$\underline{\mathbf{P}}^{n+1\|n}$	A block matrix containing the covariance between the predicted variable, denoted by $\hat{\underline{\mathbf{x}}}^{n+1\|n}$, and the variable we predict, denoted by $\tilde{\underline{\mathbf{x}}}^{n+1}$. This is otherwise known as the covariance of the predicted mean in KF literature.
$\mathcal{G}^{i-1}$	The family of sensor traces up to and including the discrete-time step t_{i-1}. That is $\mathcal{G}^{i-1} := \{\tilde{\mathbf{y}}^{i-1}, \tilde{\mathbf{y}}^{i-2}, ..., \tilde{\mathbf{y}}^{1}, \tilde{\mathbf{y}}^{0}\}$ where $i \in \{0, 1, ..., L-1\}$.
$L(x_0)$, $L(\underline{\mathbf{x}}_0)$	The likelihood function for a disturbance location guess in our 1D and 2D model problems denoted by x_0 and $\underline{\mathbf{x}}_0$, respectively.
$h(\tilde{\mathbf{y}}^i; x_0)$, $h(\tilde{\mathbf{y}}^i; \underline{\mathbf{x}}_0)$	The probability density function for $\mathbb{P}\left(\tilde{\mathbf{y}}^i \| \mathcal{G}^{i-1}\right)$ given the unknown parameter x_0 or $\underline{\mathbf{x}}_0$ in our 1D and 2D model problems, respectively.
$\hat{\mathbf{M}}$	Denotes a rectangular matrix, which we use for illustration in the outline of the singular value decomposition (SVD).
T_s	The length of time, in seconds, we use the explicit FDM approximations of u to form the SVD.
r	The number of rows in $\hat{\mathbf{M}}$. For our model problems, this corresponds to the number of nodes in our mesh. In 1D this is $(N+1)$, whereas in 2D this is $(N+1)(M+1)$.
c	The number of columns in $\hat{\mathbf{M}}$, which is equivalent to the amount of discrete-time steps used, according to T_s, from the FDM approximation of u. For our model problems, $n \in \left\{ (2-T_s)\frac{L}{T}, (2-T_s)\frac{L}{T}+1, ..., \frac{2L}{T} \right\}$.
$\hat{\mathbf{U}}$	A matrix containing the left singular vectors of $\hat{\mathbf{M}}$ in the SVD.
$\hat{\mathbf{\Sigma}}$	A matrix containing the singular values of $\hat{\mathbf{M}}$ across the main diagonal.
$\hat{\mathbf{V}}$	A matrix containing the right singular vectors of $\hat{\mathbf{M}}$ in the SVD.
d	Denotes how many principal components we take from our SVD. That is, we take d columns from $\hat{\mathbf{U}}$ which correspond to the d largest singular values.
$\boldsymbol{\phi}_i$	A vector containing the i-th left singular vector taken from $\hat{\mathbf{U}}$, where $i \in \{1, 2, ..., d\}$.

Symbol	Description
$\boldsymbol{\Phi}$	A matrix with dimensions of $(r \times d)$ which contains $\boldsymbol{\phi}_i$ for all $i \in \{1, 2, ..., d\}$. This matrix contains our principal components from the SVD formed.
σ_{d+1}	Represents the $(d+1)^{\text{th}}$ largest singular value of matrix $\hat{\mathbf{M}}$.
$\hat{\mathbf{M}}^j$	A vector corresponding to the $j-$th column of $\hat{\mathbf{M}}$.
$\boldsymbol{\beta}^j$	A vector corresponding to the $j-$th column of $\boldsymbol{\Phi}^{\mathrm{T}}\hat{\mathbf{M}}$.
$\mathbf{B}_{\text{New}}$	A matrix equivalent to $\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{B}\boldsymbol{\Phi}$ which is used only to simplify notation in this thesis.
$\hat{\boldsymbol{\beta}}^{n\|n-1}$	A block vector containing both $\boldsymbol{\beta}^{n\|n-1}$ and $\boldsymbol{\beta}^{n-1\|n-1}$.
$\tilde{\boldsymbol{\Phi}}$	Denotes a block matrix, which contains $\boldsymbol{\Phi}$ in the top left and bottom right, and $\mathbf{0}$ with the same dimensions as $\boldsymbol{\Phi}$ elsewhere.
$\tilde{\mathbf{C}}_{\text{New}}$	A block matrix with $\tilde{\mathbf{C}}\boldsymbol{\Phi}$ in the top left and bottom right, and $\mathbf{0}$ with the same dimensions as $\tilde{\mathbf{C}}\boldsymbol{\Phi}$ elsewhere.
$\underline{\tilde{\mathbf{P}}}^{n\|n-1}$	A matrix, equivalent to $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n\|n-1}\tilde{\boldsymbol{\Phi}}$ which is used only to simplify notation in this thesis.
$\underline{\tilde{\mathbf{K}}}^{n}$	A matrix, equivalent to $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{K}}^{n}$ which is used only to simplify notation in this thesis.
$\tilde{\mathbf{A}}$	A matrix, equivalent to $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}$ which is used only to simplify notation in this thesis.
$\tilde{\mathbf{Q}}_{\text{New}}$	A matrix, equivalent to $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{Q}}\tilde{\boldsymbol{\Phi}}$ which is used only to simplify notation in this thesis.
k	Used as a scale for the meshes in our explicit FDM approximation of u. The larger k, the finer our mesh, and likewise the smaller k, the coarser our mesh.
$\tilde{V}_i^n$	Contains the approximation of $\partial u(x_i, t_n)/\partial t$ at node x_i and discrete-time step t_n where $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, 2, ..., L\}$.
$\tilde{\mathbf{V}}^n$	A vector containing $\tilde{V}_i^n$ for all $i \in \{1, 2, ..., N+1\}$ at the discrete-time step denoted by t_n where $n \in \{0, 1, 2, ..., L\}$.

List of Abbreviations

Abbreviation	**Meaning**
1D, 2D	1-dimensional, 2-dimensional
1NBC, 3NBCs	1 Neumann Boundary Condition, 3 Neumann Boundary Conditions
BMI	Body Mass Index
BPM	Beats Per Minute
CAD	Coronary Artery Disease
CVD	Cardiovascular Disease
ECG	Electrocardiogram
FDM	Finite Difference Method
GB	Gigabyte
KF	Kalman Filter
LHS, RHS	Left-Hand Side, Right-Hand Side
MRI	Magnetic Resonance Imaging
NHS	National Health Service
PDE	Partial Differential Equation
RAM	Random Access Memory
SVD	Singular Value Decomposition

Chapter 1

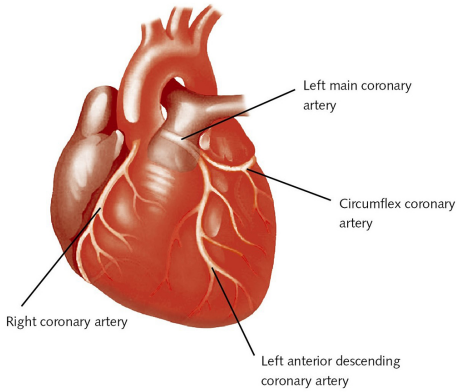
Introduction

In this chapter, we first look at the motivation behind conducting this research. We follow this with a brief literature review linking our model problems to their real-world application while detailing how much more research would be required to make this concept a reality. We finish this chapter by outlining the aims for this thesis, the contents chapter-by-chapter and our achievements and contributions to knowledge.

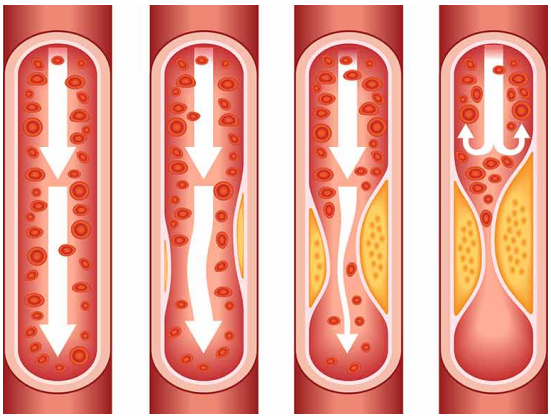
1.1 Motivation

Cardiovascular disease (CVD) was the second-largest cause of death in the United Kingdom in 2014 [1]. CVD is responsible for almost 32% of all deaths in 2009 and a significant cause of morbidity in England [2]. CVD encompasses diseases of the heart and venous system, including coronary artery disease (CAD), cerebrovascular disease and peripheral arterial disease. Coronary arteries supply blood which carries oxygen and essential nutrients to the heart muscle, enabling it to function, see Figure (1.1)(a). In CAD, arteries become blocked by a process known as atherosclerosis. Atherosclerosis is the build-up of fatty material inside the wall of the artery, called plaque, see Figure (1.1)(b). This plaque has the potential of becoming too large or breaking off and getting lodged in a narrow coronary artery, resulting in the loss of blood supply to the heart muscle, causing a heart attack and permanent damage to the heart tissue.

Mortality from CAD in England has reduced by 72% between 1979 and 2013. However, the proportion of the population suffering from CAD has remained constant at 3% [3]. CAD remains a substantial burden to society, and it was reported in 2017 to



(a) Human heart diagram depicting the location of the coronary arteries [5].



(b) Different stages of atherosclerosis [6]. Left-right: no obstruction - severe obstruction caused by a build-up of plaque.

Figure 1.1: Graphical interpretations of a human heart (illustrating the location of coronary arteries) and the different stages of atherosclerosis.

be amongst a set of diseases which result in the most years of life lost [7]. In 2002 it was reported that CAD had a large economic impact, estimated to cost the UK economy £7.06 billion annually resulting from health care costs, informal care and productivity loss [8]. Not only does CAD affect the economy, but it also destroys the lives of the individual involved and their family.

The World Health Organisation estimates that over 75% of premature CVD is preventable. By targeting risk factors, the burden of CAD on both individuals and healthcare providers could be reduced [9].

Risk factors for the development of CAD can stem from lifestyle choices such as smoking, physical inactivity and obesity but also family history, ethnicity, sex and age [10]. It is suggested by Lassale et al. (2017) that whether or not someone is obese should be considered as an independent risk factor [11], due to an increased risk of CAD, by 28%, when compared to those in the healthy weight range.

Amongst the UK population, obesity is rising. In 2017, approximately 64% of adults were either overweight or obese. Moreover, 29% of adults in 2019 were classed as obese which is an increase of 26% since 2016 where 23% of adults were classed as obese [12]. It is predicted that if Body Mass Index (BMI), the score used to define if someone is overweight or obese, continues to increase then mortality from CAD could increase [4].

1.2 Context

There are several different methods currently available to the National Health Service (NHS) to diagnose CAD. These include, for example, an electrocardiogram (ECG) and magnetic resonance imaging (MRI) [13]. Access to this specialised equipment and the requirement for a trained professional to analyse the results can cause delays in diagnosis - with a median waiting time of 2 weeks for an MRI scan, and as much as 2% of patients waiting longer than 6 weeks for diagnosis [14]. Additionally, the cost of these procedures is expensive. Within the NHS, an ECG costs £160, and an MRI scan costs £130 [15].

There is extensive literature available that makes known the importance for early detection of CAD; [16], [17] and [18]. There have been numerous studies that look into the possibility of diagnosing CAD using acoustic detection of coronary turbulence; [19], [20], [21], [22], [23], [24] and [25]. These studies look at a wide range of techniques and include some clinical trials, with a varying level of success. However, these papers mainly focus on the medical aspect of acoustic detection of coronary turbulence. Two elementary systems for diagnosing CAD using acoustic detection of coronary turbulence are discussed in detail, see [24]. Both systems make use of a commercially available digital electronic stethoscope to record noises, with the aim of determining if someone is at high risk of suffering from CAD. Small clinical trials were conducted in both cases, resulting in a success rate of 89% and 76%.

Given the associated costs and the duration of time taken to get diagnosed with CAD currently, we will be looking at a cheaper and non-invasive approach for localising CAD using acoustic detection of coronary turbulence in this thesis. This work is purely theoretical, so would need support from clinical trials in the future. In addition to this, we have limited our investigation to model problems in simple cases. Therefore, further work in this field would be required in the future where more complex cases are considered such that this theory could be applied to the real-world application of diagnosing CAD.

Acoustic detection of coronary turbulence revolves around the idea that a auditory signature is created around the location of CAD due to blood flow disturbance. Studies conducted show that when a coronary artery is partially blocked due to plaque build-up, blood flow is disturbed causing a sound to be emitted with a frequency range of $300 - 800$Hz; [26], [27], [28], [29], [30] and [31]. However, due to the presence of other sounds generated in the human thorax, with a frequency range of 10-400Hz, it can be

challenging to identify the disturbance caused by the partially blocked coronary artery using non-invasive techniques, [24] and [20]. Therefore if our concept is to be applicable in this real-world application, specialised sensors capable of picking up the sound generated from blood flow disturbance in coronary arteries would need to be engineered.

1.2.1 Outlining our approach in this thesis

In this thesis, we construct simple model problems which attempt to mimic an audible signal caused by CAD. In the real-world application being considered for this work, the domain would be a human thorax. However, such a domain is too complicated for this initial exploratory work, and so, in this thesis we opt for simpler 1-dimensional (1D) and 2-dimensional (2D) domains illustrated in Figure (1.2) and Figure (1.3), respectively. The reason for choosing these domain dimensions stemmed from an interdisciplinary project we have been involved with. The aim of this project was to collect surface readings from a 3-dimensional (3D) block, with a fabricated disturbance within the domain which would attempt to mimic atherosclerosis. Therefore, the domains we have chosen represent 1D and 2D cross sections of this 3D domain.

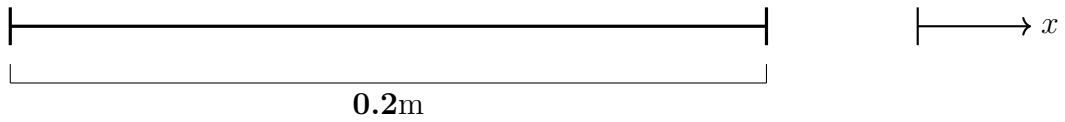


Figure 1.2: Detailed outline for our 1D domain, where all measurements are in meters.

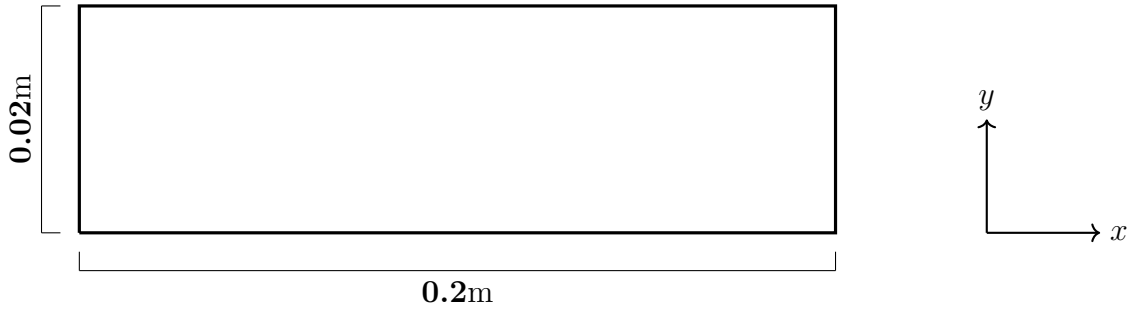


Figure 1.3: Detailed outline for our 2D domain, where all measurements are in meters.

In this thesis our aim is to construct a series of model problems that will simulate the propagation of a localised disturbance within a domain. Given the real-world application

being considered for the motivation behind this research, an appropriate governing equation for our model problems would be a viscoelastic wave equation with a forcing function that generates a localised disturbance within the domain. However, due to the infancy of this research and the added complexity of using a viscoelastic wave equation as our governing equation, we instead choose our governing equation to be the acoustic wave equation. Therefore, the general form for the model problem considered in this thesis is given as

$$u_{tt} - \nabla^2 u = f \quad \text{in} \quad (0,T] \times \Omega, \tag{1.1}$$

$$u = 0 \quad \text{on} \quad [0,T] \times \Gamma_{\mathrm{D}_1}, \tag{1.2}$$

$$\frac{\partial u}{\partial n} = 0 \quad \text{on} \quad [0,T] \times \Gamma_{\mathrm{N}_1}, \tag{1.3}$$

$$u = 0 \quad \text{on} \quad \{0\} \times \Omega, \tag{1.4}$$

$$u_t = 0 \quad \text{on} \quad \{0\} \times \Omega \tag{1.5}$$

where Ω denotes our domain in $\mathbb{R}^p$ for $p \in \{1,2\}$, Γ_{D_1} and Γ_{N_1} represent Dirichlet and Neumann boundaries, respectively, $[0,T]$ is our time domain, u_t and u_{tt} represent the first and second order partial derivatives of u with respect to time, respectively, and $\nabla^2 u$ represents the Laplacian of u in $\mathbb{R}^p$.

There is no readily available literature that outlines what our initial conditions should be. As a result of this, we set both initial conditions to zero. Doing this does limit the findings in this thesis, but it would be simple to change these initial conditions if experimental data became available that would enable us to choose accurate initial conditions.

The reason for choosing our boundary conditions stem from the domains we have chosen. Recall that the domains in this thesis are taken to be 1D and 2D cross sections of a 3D domain used in another interdisciplinary project that we were involved with. This 3D domain represents a container with solid boundaries on each side, with an open side on the top. Therefore we choose Dirichlet boundary conditions equal to zero for our 2D model problem, with a Neumann boundary condition equal to zero on the top. We extended this 2D problem to consider Neumann boundary conditions that equal zero on all sides of our 2D domain, apart from one, as this results in a more complex model problem that we wanted to explore. A similar scenario is applied for our 1D model problem in chapter 3.

Recall that the real-world application for this work is to locate a disturbance caused by atherosclerosis, using sensors placed on the surface of the human chest. Therefore, the domain in this scenario would be a human thorax that contains bones, organs and muscle tissue amongst other materials. These different materials will interact with an acoustic signal differently since they all have a different acoustic impedance, see [32]. This research goes further by illustrating that at a boundary between media with different acoustic impedance, for example, bone and muscle tissue, some wave energy is reflected. Therefore, it is possible that a Dirichlet or Neumann boundary condition could be applicable. However, more in-depth experiments in the future, closely aligned with our application, would need to be conducted so we could choose more accurate boundary conditions in our model problems.

Before choosing a forcing function for our model problems, denoted by f in (1.1), we first looked for relevant literature that could inform our decision. However, there was very little available literature that could have helped guide our choice for the forcing function used in our model problems for our primary application. Therefore, we used our intuition to construct a realistic forcing function that will be used in our model problems. If experiments are conducted in the future that could help improve the choice of our forcing function, then this could be changed without difficulty.

Therefore, the forcing function that will be used in our model problems is given by

$$f(\mathbf{x}, t; \mathbf{x}_0) = A \sin^2(\pi t) \exp^{-||\mathbf{x}-\mathbf{x}_0||^2/\epsilon} \sin(2\pi F t) \tag{1.6}$$

where $\mathbf{x}$ and t denote the continuous spatial and time domains, $\mathbf{x}_0$ denotes the location of our disturbance, A represents the amplitude of the disturbance, F is the carrier frequency of our disturbance, and the magnitude of ϵ controls the Gaussian spread of the disturbance.

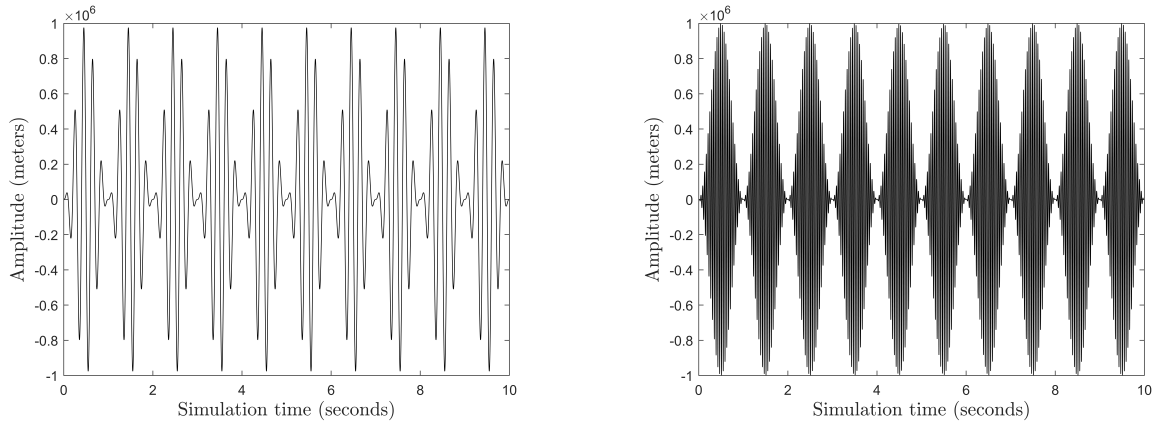
Having now defined our forcing function, we need to explain why we chose it, and what values we take for the unknown parameters in (1.6). The first *sine* function in (1.6) has a modulation frequency of 1Hz, and is attempting to mimic a heartbeat with 60 beats per minute (BPM). We include this in our forcing function because the real-world application would involve placing sensors on a human thorax to record all sound signatures generated, which would include a heartbeat. Recall that we have taken the heartbeat to have 60 BPM, however, it would be easy to modify this to reflect a different BPM.

The second *sine* function in (1.6) attempts to mimic the disturbance caused by CAD. We have already discussed that the frequency of this disturbance in our real-world

application would be between $300 - 800$Hz. In this thesis, we consider disturbance frequencies of $F = 25$Hz, $F = 150$Hz and $F = 300$Hz for simplicity.

Lastly, the exponential function in (1.6) localises the entire forcing function in our domains, resulting in a Gaussian spread which decays further away from the disturbance location. The value chosen for ϵ is problem-dependent, and so we discuss this in the result chapters.

Figure (1.4) shows our forcing function in (1.6), when $\mathbf{x} = \underline{\mathbf{x}}_0$, with disturbance frequencies of $F = 5$Hz and $F = 25$Hz. The reasoning for our choice of the amplitude will be discussed in the result chapters.



(a) Disturbance frequency $F = 5$Hz. **(b)** Disturbance frequency $F = 25$Hz.

Figure 1.4: An illustration of the amplitude for the spatial exponential term in our forcing function, see (1.6), for a simulation duration of $T = 10$ seconds and with $A = 10^6$.

Having outlined our domains and their governing equations, we are now interested in giving a broad outline to the stages taken in our model problems to locate the source of our acoustic wave equation, see Figure (1.5). We first chose a location for the source of our acoustic wave equation within our domain and generated explicit finite difference method (FDM) approximations of u across our domain. The mathematics used at this stage is common practice when approximating a PDE within simple domains. The reason we chose to use explicit FDM approximations of u rather than implicit FDM approximations of u was due to the requirement to inverse a matrix after each time step with the implicit method. We experimented with both methods, and similar results were obtained. The mathematics used to approximate u in (1.1) for this stage of our model problem is outlined in chapter 2.

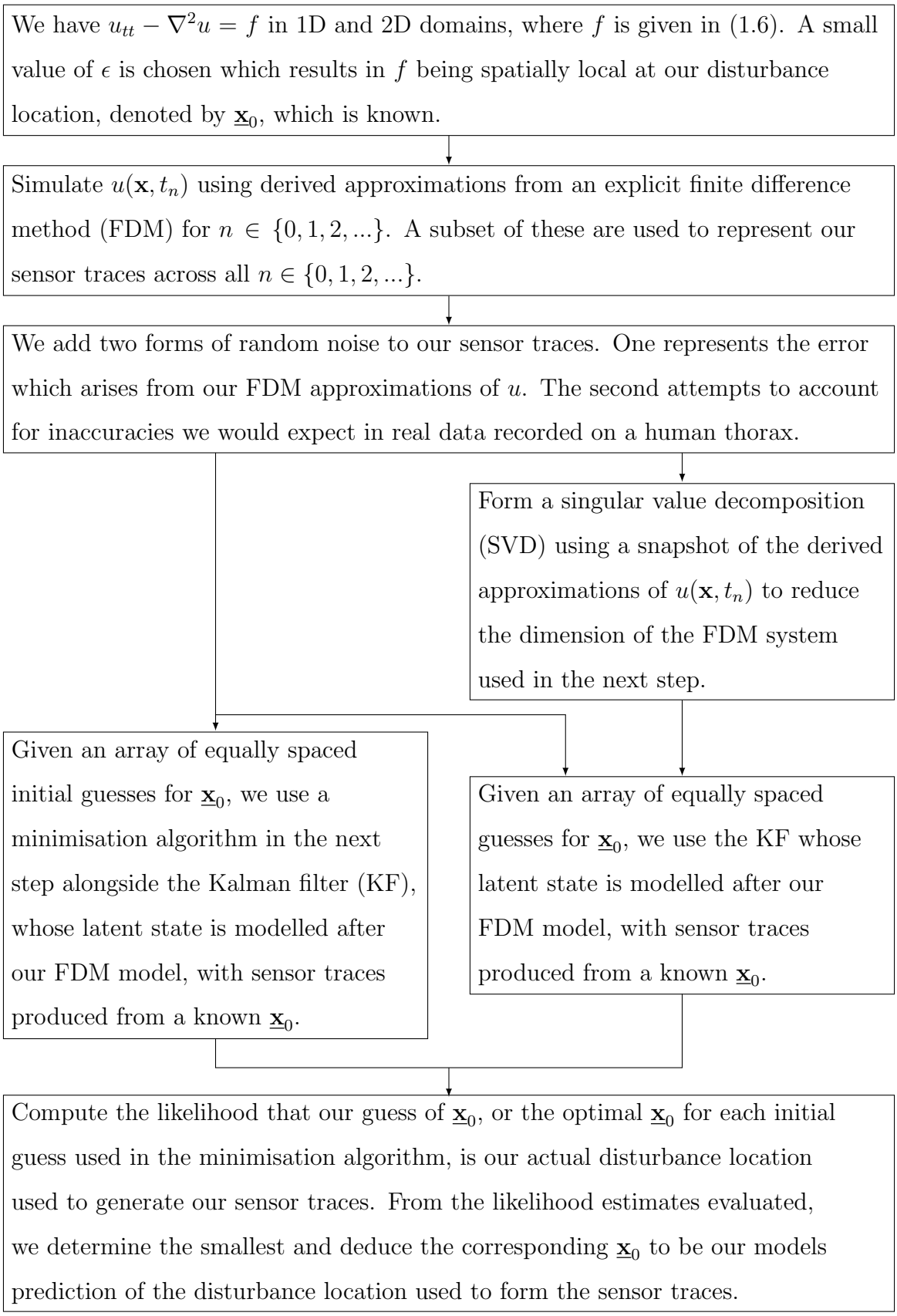


Figure 1.5: A schematic showing the broad steps required to locate the source of our acoustic wave equation.

We then make use of the Kalman filter (KF) to estimate our latent state for an array of guesses for the location of our disturbance, which uses sensor traces that correspond to the explicit FDM approximations of u at specific nodes in our mesh. This stage of our model differs throughout the thesis, as different strategies are used to reduce the run-time and RAM requirements for our model problems, which are discussed in the next two sections of this chapter.

The final stage of our model is to use a maximum likelihood estimator, which requires data generated from the KF, to predict which of the array of guesses for the location of our disturbance is the most likely to be the disturbance used to generate the sensor traces. Again, this part of the model differs throughout this thesis and several alternatives were tried as using a minimisation algorithm with an array of initial guesses required a significant amount of run-time. For more detail, see the schematics outlined in chapters 3-7.

1.3 Thesis outline

Having defined our aim for the work in this thesis, we break down the work presented chapter-by-chapter in the following set of bullet points

- *Chapter 2:* This chapter outlines the methodology used in this thesis. We start by introducing the acoustic wave equation in both 1D and 2D, which will govern our model problems. We derive explicit FDM solutions to approximate u in the acoustic wave equations. We test the convergence of these approximations against an exact solution. We outline the KF, then derive a likelihood estimator. Lastly, we set out the singular value decomposition (SVD), and subsequently, modify the KF equations to reduce the matrix dimensions.

- *Chapter 3:* This chapter contains results corresponding to our 1D model problem, with a disturbance frequency of $F = 25$Hz in our forcing function, denoted by f.

- *Chapter 4:* This chapter contains results corresponding to our 1D model problem, with disturbance frequencies of $F = 150$Hz and $F = 300$Hz in our forcing function, denoted by f.

- *Chapter 5:* This chapter contains results corresponding to our 2D model problems, with a disturbance frequency of $F = 25$Hz in our forcing function, denoted by f.

- *Chapter 6:* This chapter contains results corresponding to our 2D model problems, with disturbance frequencies of $F = 150$Hz and $F = 300$Hz in our forcing function, denoted by f.

- *Chapter 7:* This chapter contains results corresponding to our 1D model problem where we consider both the pressure, denoted by u, and partial derivatives of u in time to predict the location of our disturbance.

- *Chapter 8:* This chapter contains the conclusion, where we discuss the work presented in this thesis and highlight the significance and weaknesses of the results. In addition to this, several recommendations for future work are discussed which outline multiple directions to extend the work presented in this thesis.

1.4 Achievements and contributions to knowledge

In this section, we outline our achievements and contributions to knowledge in the following set of bullet points

1. We solved 1D and 2D model problems which attempt to mimic a localised source representing CAD. We first do this in chapter 3, based on a 1D model with a disturbance frequency of $F = 25$Hz. In chapter 4, we remain spatially in 1D, but consider disturbance frequencies of $F = 150$Hz and $F = 300$Hz. In chapter 5, we outline our 2D model problems and consider a disturbance frequency of $F = 25$Hz. In chapter 6, we remain spatially in 2D and instead look at disturbance frequencies of $F = 150$Hz and $F = 300$Hz. Lastly, in chapter 7 we consider our 1D model, but instead of having only explicit FDM approximations of u as our sensor traces, we consider $\partial u/\partial t$ only, and both u and $\partial u/\partial t$. There is some literature within the medical field involving different methods of diagnosing CAD using sensors and the knowledge of blood flow disturbance caused by plaque build-up at the site of CAD; [24] and [20]. However, these publications do not explain the mathematical model behind the method used to diagnose CAD. The mathematics used to approximate the solution of our PDE in (1.1) are commonly used for similar purposes.

2. We discovered that we could use explicit FDM approximations of u from a coarse mesh for our sensor traces used in the KF. As a result of this, we incorporate the error created by these coarse explicit FDM approximations into the noise elements of the KF, resulting in substantially smaller matrices in the KF, which for limitations in system random access memory (RAM) and run-time, made our more complex model problems viable. This is a significant discovery, and something that is not documented in available literature. Therefore, this discovery forms part of the novel research presented in this thesis. In chapter 3, we consider both fine and coarse explicit FDM approximations of u in (1.1). We were able to do this since in 1D and with a disturbance frequency of $F = 25$Hz, the required density of the mesh to form a good FDM approximation was not beyond our computational reach. In the remaining chapters, we only consider coarse explicit FDM approximations of u and subsequently partial derivatives of u in time in chapter 7.

3. We formed an SVD from our explicit FDM approximations of u, and by taking a subset of the principal components, we were able to reduce the size of several matrices within the KF. It is widely known that by constructing an SVD for a matrix and taking a subset of the principal components with the largest singular values, you can reduce the dimensions of that matrix. In addition to this, there are several cases where an SVD has been used in conjunction with the KF to reduce the dimensions of the matrices in the KF; [33], [34] and [35]. In this thesis, we repeatedly used this technique to reduce the run-time and RAM constraints of the KF. Results that use this technique can be found in chapters 3-7.

4. The remaining limitation in our model problems arises from the number of discrete-time steps required when running the KF. We discovered that the KF could be terminated after 40% of the total discrete-time steps and still yield good results. This is another significant discovery, and something that is not documented in readily available literature within this area of mathematics. Therefore, this discovery forms part of the novel research presented in this thesis. This is shown in chapter 5 for our 2D model problem with $F = 25$Hz.

5. In chapter 7, we formed sensor traces that corresponded to approximations of $\partial u/\partial t$ which yielded similar results to using explicit FDM approximations of u as our sensor

traces. The mathematics used to form these sensor traces using the explicit FDM approximations of u is outlined in chapter 7. The mathematics used here is common within this field and so does not warrant a novel piece of research within its own right. However, what is interesting with these results is our ability to use a range of different sensors with little effect on the success of our model problems. If future experiments were to be carried out, this finding could prove to be vital when deciding what sensors to use when recording data.

In this chapter we have outlined the motivation behind this research, written a short literature review linking our model problems to their real-world application, given a broad outline of the model problems considered, a breakdown of the work presented in each chapter, and portrayed our achievements and contributions to knowledge. We now move on to our next chapter which outlines the mathematical theory and more details for the different model problems used in this thesis.

Chapter 2

Methodology

In this chapter, we outline our model problems in 1D and 2D. After which, we discretise the corresponding domains and produce FDM approximations, which at specific locations are used as our sensor traces. The errors associated with these FDM approximations are calculated, and convergence of the approximate solutions are checked against exact solutions which are both dependent on time and space.

We outline the Kalman filter and illustrate why noise is added. In addition to this, we make modifications to the KF enabling it to work with our discrete model problems. Lastly, we outline the likelihood estimator - allowing us to predict the location of our disturbance.

In cases where we require fine meshes from the discretisation of our model problems, requiring large amounts of system RAM and time for the KF to run, we approximate the FDM approximations using an SVD, resulting in considerably smaller matrix dimensions in the KF. In this chapter, we outline the SVD and the associated modifications required to the KF.

2.1 Discretisation of the 1D wave equation

In this section, we look at discretising our 1D wave equation. Recall the general form for the model problem considered in this thesis, see (1.1)-(1.5). For our 1D model problem, $u_{tt} \equiv \partial^2 u/\partial t^2$, $\nabla^2 u \equiv \partial^2 u/\partial x^2$ and f is our forcing function in (1.6) with $\mathbf{x} \equiv (x)$, and $\mathbf{x}_0 \equiv x_0$ which represents the location of our disturbance.

Now we have defined our PDE in (1.1) for our 1D model problem, we look at outlining

the domain and the boundary. From the general form in (1.1)-(1.5) to our 1D model problem, we alter the notation used for both the domain and boundaries. We defined our 1D domain as $\Omega_{1D} := (0, 0.2)$, see Figure (1.2). The corresponding boundary, denoted by $\partial\Omega_{1D}$, is made up from

1. $\frac{\partial u}{\partial x} := 0$ on $\Gamma_{N_1} := \{x \in \overline{\Omega}_{1D} : x = 0.2\}$,
2. $u := 0$ on $\Gamma_{D_1} := \{x \in \overline{\Omega}_{1D} : x = 0\}$.

Now we have defined our 1D model problem, we approximate the solution, u, at equally spaced nodes across the domain. We achieve this by discretising our model problem, see Figures (2.1) and (2.2), into a mesh with $N + 1$ nodes in space and $L + 1$ in time. The interval between nodes in space and time are denoted by $H_x := 0.2/N$ and $t_n := n\Delta t$ respectively, where $n \in \{0, 1, ..., L\}$, $\Delta t := T/L$ and T is the duration of our simulation.

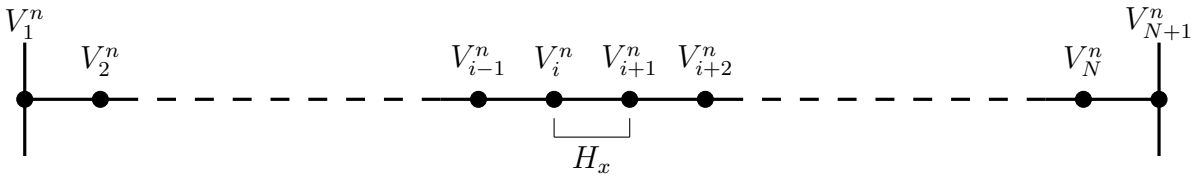


Figure 2.1: The discretisation of our 1D domain where H_x denotes the spacing between the nodes in our mesh, V_i^n denotes the explicit FDM approximation of u at node x_i and discrete-time step t_n for $i \in \{1, 2, ..., N + 1\}$ and $n \in \{0, 1, 2, ..., L\}$.

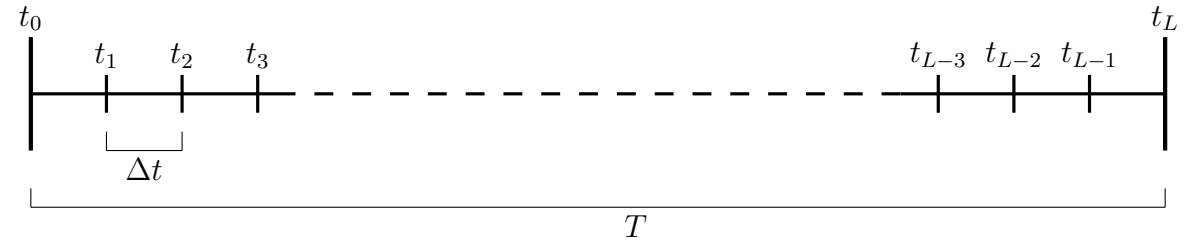


Figure 2.2: The discretisation in time for all model problems considered in this thesis, where Δt denotes the spacing between our discrete-time steps, t_n represents the discrete-time steps for $n \in \{0, 1, 2, ..., L\}$, and T is the duration of our simulation.

Let us first denote $u(x_i, t_n)$ as the solution of our PDE, denoted by u, at node $x_i = (i - 1)H_x$ and time $t_n = n\Delta t$, where $i \in \{1, 2, ..., N + 1\}$ and $n \in \{0, 1, 2, ..., L\}$. Now we have defined all the relevant notation, we start by discretising at the interior nodes

in space. That is, x_i for $i \in \{2, 3, ..., N\}$. To achieve this, we need to approximate all partial derivatives in (1.1), in this case, that is both $\nabla^2 u \equiv \partial^2 u/\partial x^2$ and u_{tt} using Taylor series. Starting with $\partial^2 u/\partial x^2$, we formulate the solution, u, at nodes denoted by $(x_{i+1}, t_n) \in \Omega_{1\mathrm{D}} \times [0, T]$ and $(x_{i-1}, t_n) \in \Omega_{1\mathrm{D}} \times [0, T]$ giving us

$$u(x_{i+1}, t_n) = u(x_i, t_n) + H_x u_x(x_i, t_n) + \frac{H_x^2}{2} u_{xx}(x_i, t_n) + \frac{H^3}{3!} u_{xxx}(x_i, t_n) + \mathcal{O}(H_x^4), \quad (2.1)$$

$$u(x_{i-1}, t_n) = u(x_i, t_n) - H_x u_x(x_i, t_n) + \frac{H_x^2}{2} u_{xx}(x_i, t_n) - \frac{H^3}{3!} u_{xxx}(x_i, t_n) + \mathcal{O}(H_x^4) \quad (2.2)$$

respectively where $u_x \equiv \partial u/\partial x$ and $H_x := 0.2/N$. We now add both (2.1) and (2.2) together, resulting in several terms cancelling to get

$$u(x_{i+1}, t_n) + u(x_{i-1}, t_n) = 2u(x_i, t_n) + H_x^2 u_{xx}(x_i, t_n) + \mathcal{O}(H_x^4). \quad (2.3)$$

Recall that we want to determine an approximation to our partial derivative, $\partial^2 u/\partial x^2 \equiv u_{xx}$, at all interior nodes. Hence, we make $u_{xx}(x_i, t_n)$ the subject of (2.3) to get

$$u_{xx}(x_i, t_n) = \frac{u(x_{i+1}, t_n) + u(x_{i-1}, t_n) - 2u(x_i, t_n)}{H_x^2} + \mathcal{O}(H_x^2). \quad (2.4)$$

To approximate $\nabla^2 u$, we need to neglect the higher-order terms in (2.4). We do this later as we are interested in the combined higher-order terms being neglected for the entire expression. First, we must approximate u_{tt}.

We now look at approximating the second partial derivative in (1.1), that is u_{tt}. Again, similar to above, we use the Taylor series and get

$$u_{tt}(x_i, t_n) = \frac{u(x_i, t_{n-1}) + u(x_i, t_{n+1}) - 2u(x_i, t_n)}{\Delta t^2} + \mathcal{O}(\Delta t^2). \quad (2.5)$$

Substituting both (2.4) and (2.5) into (1.1), we eliminate the original partial derivatives in the strong form to get

$$\frac{u(x_i, t_{n-1}) - 2u(x_i, t_n) + u(x_i, t_{n+1})}{\Delta t^2} + \frac{2u(x_i, t_n) - u(x_{i-1}, t_n) - u(x_{i+1}, t_n)}{H_x^2} = f(x_i, t_n) + \mathcal{O}(\Delta t^2 + H_x^2) \quad (2.6)$$

where $f(x_i, t_n)$ is f at the node defined by x_i and t_n in time. We are interested in making $u(x_i, t_{n+1})$ the subject of (2.6), enabling us to approximate u at all interior nodes at the next time step with initial conditions. Therefore, we get

$$u(x_i, t_{n+1}) = \Delta t^2 f(x_i, t_n) - u(x_i, t_{n-1}) + \left(2 - \frac{2\Delta t^2}{H_x^2}\right) u(x_i, t_n)$$

$$+ \frac{\Delta t^2}{H_x^2}\bigg(u(x_{i-1}, t_n) + u(x_{i+1}, t_n)\bigg) + \Delta t^2 \mathcal{O}(\Delta t^2 + H_x^2). \tag{2.7}$$

We now want to make (2.7) an approximation of (1.1). Therefore, we denote $V_i^n \approx u(x_i, t_n)$. Recall that our approximate to (1.1) became approximate by neglecting the higher-order terms in (2.7), and so, we arrive at an explicit FDM approximation to (1.1) at interior nodes, given by

$$V_i^{n+1} = \Delta t^2 f(x_i, t_n) - V_i^{n-1} + \left(2 - \frac{2\Delta t^2}{H_x^2}\right) V_i^n + \frac{\Delta t^2}{H_x^2}\bigg(V_{i-1}^n + V_{i+1}^n\bigg) \tag{2.8}$$

for $i \in \{2, 3, ..., N\}$.

Having discretised the interior nodes, we now discretise the nodes which lie on the boundary of our domain. We start to discretise the Neumann boundary, denoted by Γ_{N_1} at x_{N+1}. Therefore we use Taylor series like before, but start at the node denoted by $(x_N, t_n) \in \Omega_{1\mathrm{D}} \times [0, T]$ giving

$$u(x_N, t_n) = u(x_{N+1}, t_n) - H_x u_x(x_{N+1}, t_n) + \frac{H_x^2}{2} u_{xx}(x_{N+1}, t_n) + \mathcal{O}(H_x^3). \tag{2.9}$$

Recall that the boundary condition defined on Γ_{N_1} is $\partial u/\partial x = 0$, at the node x_{N+1} and t_n where $n \in \{0, 1, ..., L\}$. Another way to write this using notation already used is $u_x(x_{N+1}, t_n) = 0$. Therefore, by substituting this into (2.9), we get

$$u(x_N, t_n) = u(x_{N+1}, t_n) + \frac{H_x^2}{2} u_{xx}(x_{N+1}, t_n) + \mathcal{O}(H_x^3). \tag{2.10}$$

We still have a partial derivative in (2.10) which needs approximating. By rearranging (1.1), we can eliminate the partial derivative, denoted by $u_{xx}(x_{N+1}, t_n)$, to get

$$u(x_N, t_n) = u(x_{N+1}, t_n) + \frac{H_x^2}{2}\bigg(u_{tt}(x_{N+1}, t_n) - f(x_{N+1}, t_n)\bigg) + \mathcal{O}(H_x^3). \tag{2.11}$$

Using (2.5) with $i := N+1$, we can eliminate the last partial derivative in (2.11), denoted by $u_{tt}(x_{N+1}, t_n)$, to get

$$\begin{aligned} u(x_N, t_n) = u(x_{N+1}, t_n) &+ \frac{H_x^2}{2\Delta t^2}\bigg(u(x_{N+1}, t_{n-1}) - 2u(x_{N+1}, t_n) + u(x_{N+1}, t_{n+1})\bigg) \\ &- \frac{H_x^2}{2} f(x_{N+1}, t_n) + \mathcal{O}(H_x^3 + \Delta t^2). \end{aligned} \tag{2.12}$$

Recall that we want to approximate u on Γ_{N_1}, that is we want to find an approximation at the node denoted by x_{N+1} across all time intervals using initial conditions. Since our

problem is recursive, we want to make $u(x_{N+1}, t_{n+1})$ the subject of (2.12). Therefore, we get

$$u(x_{N+1}, t_{n+1}) = \Delta t^2 f(x_{N+1}, t_n) - u(x_{N+1}, t_{n-1}) + \left(2 - \frac{2\Delta t^2}{H_x^2}\right) u(x_{N+1}, t_n) + \frac{2\Delta t^2}{H_x^2} u(x_N, t_n) + \frac{2\Delta t^2}{H_x^2} \mathcal{O}(H_x^3 + \Delta t^2). \tag{2.13}$$

Currently, in (2.13), the higher-order terms imply that in space, we would be neglecting terms of order H_x and higher, which is not sufficient. We are looking at neglecting terms of H_x^2 and higher. To do this, we look at the condition required for our approximations of u in our PDE to be stable. The stability of approximate solutions for an array of partial differential equations (PDE) are discussed in [36]. On page 187 of this book, the condition required for a stable approximate solution of a hyperbolic PDE in the same form as our PDE in (1.1) is identified to be

$$\frac{\Delta t}{H_x} \leq C$$

where C is a positive constant. Using this knowledge, we replace the $\Delta t^2/H_x^2$ multiplier before the higher-order term in (2.13) with a squared constant, C^2. Hence, we get

$$u(x_{N+1}, t_{n+1}) = \Delta t^2 f(x_{N+1}, t_n) - u(x_{N+1}, t_{n-1}) + \left(2 - \frac{2\Delta t^2}{H_x^2}\right) u(x_{N+1}, t_n) + \frac{2\Delta t^2}{H_x^2} u(x_N, t_n) + \mathcal{O}(H_x^3 + \Delta t^2). \tag{2.14}$$

Now we have the desired higher-order terms in (2.14), we can neglect the higher-order terms to arrive at our approximation of u at the node denoted by x_{N+1} across all time steps. Again, using $V_i^n \approx u(x_i, t_n)$, we get

$$V_{N+1}^{n+1} = \Delta t^2 f(x_{N+1}, t_n) - V_{N+1}^{n-1} + \left(2 - \frac{2\Delta t^2}{H_x^2}\right) V_{N+1}^n + \frac{2\Delta t^2}{H_x^2} V_N^n. \tag{2.15}$$

Recall that part of the boundary on our domain is a Dirichlet boundary, whose boundary condition is denoted by $u(x_1, t_n) = 0$ for $n \in \{0, 1, 2, ..., L\}$. Since this has no partial derivatives, this does not need discretising. Hence, to keep in-line with the notation used we rewrite it as $V_1^n = 0$ for $n \in \{0, 1, 2, ..., L\}$.

Together (2.8) and (2.15) approximate u at every node in our mesh, across all time steps. Using these, we now construct a matrix system to approximate (1.1). Given that

our domain, denoted by $\Omega_{1\mathrm{D}}$, has $N+1$ nodes we can define $\mathbf{V}^n$ and $\boldsymbol{\mathcal{F}}^n$ as

$$\mathbf{V}^n = \begin{pmatrix} V_1^n \\ V_2^n \\ \vdots \\ V_{N+1}^n \end{pmatrix} \text{ and } \boldsymbol{\mathcal{F}}^n = \begin{pmatrix} f(x_1, t_n) \\ f(x_2, t_n) \\ \vdots \\ f(x_{N+1}, t_n) \end{pmatrix}$$

respectively. In addition to this, we define a matrix $\mathbf{B}$, which contains all the coefficients of V_i^n for $i \in \{1, 2, \cdots, N+1\}$ in (2.8), (2.15) and the boundary condition defined on the Dirichlet boundary. Having defined our notation, we can deduce the matrix system to be

$$\mathbf{V}^{n+1} = \mathbf{B}\mathbf{V}^n - \mathbf{V}^{n-1} + \Delta t^2 \boldsymbol{\mathcal{F}}^n \tag{2.16}$$

for $n \in \{1, 2, ..., L-1\}$.

2.1.1 MATLAB implementation

In this section, we look at verifying whether the MATLAB code which approximates u in (1.1) using (2.16) works as intended. We achieve this by computing the maximum absolute error between an exact solution and the approximated solution across all nodes at $t_L := T$, where for these convergence tests, $T = 1$. First, we must construct an exact solution for u such that all boundary conditions are satisfied.

Manufactured space and time-dependent solution

We choose an exact solution for u which contains trigonometric functions since this is similar to the sensor traces we would expect to observe using our forcing function in (1.6) within our acoustic wave equation. Hence, a reasonable choice for u such that the boundary conditions are satisfied is

$$u(x_i, t_n) = \sin\left(\frac{2\pi x_i}{0.8}\right)\left(1 + \sin(t_n)\right) \tag{2.17}$$

where $i := \{1, 2, ..., N+1\}$ and $n \in \{0, 1, ..., L\}$. Subsequently, by substituting (2.17) into (1.1) we get

$$f(x_i, t_n) = \sin\left(\frac{2\pi x_i}{0.8}\right)\left(6.25\pi^2\big(1 + \sin(t_n)\big) - \sin(t_n)\right) \tag{2.18}$$

which we will use in this convergence test.

Recall the matrix system in (2.16) which approximates u in (1.1) at every node across

all time steps. This matrix system is recursive and so initially requires both $\mathbf{V}^0$ and $\mathbf{V}^1$ to be known. That is, we need the approximation of u when $t_0 := 0$ and $t_1 := \Delta t$ respectively. We have two initial conditions given by $u(x_i, t_0)$ and $u_t(x_i, t_0)$ for $i \in \{1, 2, ..., N+1\}$. Therefore, $\mathbf{V}^0$ is known. However, to get $\mathbf{V}^1$, we need to use Taylor series and both initial conditions.

In this convergence test, we know the exact solution to u, and so by using (2.17), the initial conditions across all nodes, defined for all $i \in \{1, 2, \cdots, N+1\}$, are both

$$u(x_i, t_0) = u_t(x_i, t_0) = \sin\left(\frac{2\pi x_i}{0.8}\right). \tag{2.19}$$

Therefore, directly from (2.19), $\mathbf{V}^0 = \sin\left(\frac{2\pi x_i}{0.8}\right)$ for all $i \in \{1, 2, ..., N+1\}$. Using Taylor series in conjunction with (2.19), we get

$$\begin{aligned} u(x_i, t_1) &= u(x_i, t_0) + \Delta t u_t(x_i, t_0) + \mathcal{O}(\Delta t^2), \\ &= \sin\left(\frac{2\pi x_i}{0.8}\right)(1 + \Delta t) + \mathcal{O}(\Delta t^2). \end{aligned}$$

By neglecting the higher-order terms, we get $\mathbf{V}^1 \approx \sin\left(\frac{2\pi x_i}{0.8}\right)(1 + \Delta t)$ for all $i \in \{1, 2, ..., N+1\}$.

Error convergence

Recall that our approximation to (1.1) became approximate because we disregarded the higher-order terms in (2.7) and (2.14). These higher-order terms directly correlate to the magnitude of error we expect to get between an exact solution and our approximation of u. Recall that the higher-order terms were $\mathcal{O}(\Delta t^2 + H_x^2)$ at the interior nodes and $\mathcal{O}(H_x^3 + \Delta t^2)$ along the Neumann boundary. It is trivial that the maximum error recorded would correlate to the higher-order terms resulting in the larger error. Hence, to find out what these are we combined both higher-order terms to get

$$\mathcal{O}\left(\Delta t^2 + H_x^2 + H_x^3 + \Delta t^2\right) = \mathcal{O}(H_x^2 + \Delta t^2).$$

Therefore, we would expect the maximum absolute error to decrease by a factor of four as both H_x and Δt half. Table (2.1) displays the maximum absolute error across all nodes in our mesh at $T = 1$ between an exact solution in (2.17) and our approximations of u for increasing values of N, which subsequently decrease H_x, and decreasing Δt. We can deduce that as N doubles, that is H_x halves, and Δt halves the error convergence approaches four which is what we expected.

N	Δt	Maximum |Error|	Error Percentage†	Error Convergence	Computation Time†† (s)
4	250^{-1}	0.021495477	1.153830611	N/A	0.004961
8	500^{-1}	0.005489642	0.297225719	3.915642769	0.003367
16	1000^{-1}	0.001381220	0.074950142	3.974487772	0.003710
32	2000^{-1}	0.000346053	0.018788657	3.991353926	0.009028
64	4000^{-1}	0.000086585	0.004701703	3.996685338	0.030778
128	8000^{-1}	0.000021654	0.001175883	3.998568394	0.101443

† Error Percentage = [maximum error at node (i, j)]/[approximation at node (i, j)] $\times$ 100.

†† Computed using an i5 − 4590 @ 3.30GHz, 16GB RAM @ 1333 MHz and Intel HD Graphics 4600.

Table 2.1: The error convergence of the explicit FDM approximations of u for our acoustic 1D wave equation.

The Lax equivalent theorem states that if a finite difference scheme is both stable and consistent, then it is convergent [37]. We already know that our finite difference scheme in (2.16) is stable, see page 187 in [36]. For our finite difference scheme to be consistent, the maximum absolute error in Table (2.1) must approach zero as both $H_x = 0.2/N$ and Δt approach zero. Upon inspection of Table (2.1), we can see that this is the case. Therefore, we can conclude that our finite difference scheme is convergent.

2.2 Discretisation of the 2D wave equation

In this section, we look at discretising our 2D wave equation. Recall the general form for the model problem considered in this thesis, see (1.1)-(1.5). For our 2D model problem, $u_{tt} \equiv \partial^2 u/\partial t^2$, $\nabla^2 u \equiv \partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$ and f is our forcing function in (1.6) with $\mathbf{x} \equiv (x, y)$, and $\mathbf{x}_0 \equiv (x, y)$ which represents the location of our disturbance.

Now we have defined our PDE in (1.1) for our 2D model problem, we look at outlining the domain and the boundary. From the general form in (1.1)-(1.5) to our 2D model problem, we alter the notation used for both the domain and boundaries. We defined our 2D domain as $\Omega_{2\mathrm{D}} := (0, 0.2) \times (0, 0.02)$, see Figure (1.3). We consider two model problems in 2D, each with different boundary conditions. For our first model problem, the boundary, denoted by $\partial\Omega_{2\mathrm{D}_1}$, is made up of

1. $\frac{\partial u}{\partial y} := 0$ on $\Gamma_{\mathrm{N}_2} := \big\{(x,y) \in \overline{\Omega}_{\mathrm{2D}} : 0 < x < 0.2,\ y = 0.02\big\}$,

2. $u := 0$ on $\Gamma_{\mathrm{D}_2} := \partial\Omega_{\mathrm{2D}_1} \setminus \overline{\Gamma}_{\mathrm{N}_1}$.

The boundary for our second problem, denoted by $\partial\Omega_{\mathrm{2D}_2}$, is made up from

1. $\frac{\partial u}{\partial y} := 0$ on $\Gamma_{\mathrm{N}_2} := \big\{(x,y) \in \overline{\Omega}_{\mathrm{2D}} : 0 < x < 0.2,\ y = 0.02\big\}$,

2. $\frac{\partial u}{\partial x} := 0$ on $\Gamma_{\mathrm{N}_3} := \big\{(x,y) \in \overline{\Omega}_{\mathrm{2D}} : x = 0,\ 0 < y < 0.02\big\}$,

3. $\frac{\partial u}{\partial x} := 0$ on $\Gamma_{\mathrm{N}_4} := \big\{(x,y) \in \overline{\Omega}_{\mathrm{2D}} : x = 0.2,\ 0 < y < 0.02\big\}$,

4. $u := 0$ on $\Gamma_{\mathrm{D}_3} := \partial\Omega_{\mathrm{2D}_2} \setminus \big\{\overline{\Gamma_{\mathrm{N}_2} \cup \Gamma_{\mathrm{N}_3} \cup \Gamma_{\mathrm{N}_4}}\big\}$.

Having defined both 2D model problems, we are interested in approximating the solution, u, across the domain. At interior nodes, the method for approximating u for both model problems is the same. However, on the boundary, they are mostly different and so will be computed separately where necessary. To approximate u we discretise our domain, see Figures (2.3) and (2.2), into a mesh with $(N+1) \times (M+1)$ nodes and $L+1$ intervals in time. We define our intervals in space by $H_x := 0.2/N$ and $H_y := 0.02/M$ in the x and y-directions respectively. Our time step variable is the same as in 1D. Let us denote the solution of our PDE, denoted by u, at the node (x_i, y_j) for $x_i = (i-1)H_x$ where $i \in \{1, 2, ..., N+1\}$, and $y_j = (j-1)H_y$ where $j \in \{1, 2, ..., M+1\}$ as $u(x_i, y_j, t_n)$ at the time step $t_n = n\Delta t$ for $n \in \{0, 1, 2, ..., L\}$.

Having all relevant notation defined, we can discretise the domain to get the approximation of u at the interior nodes. That is, nodes denoted by (x_i, y_j) for $i \in \{2, 3, ..., N\}$ and $j \in \{2, 3, ..., M\}$. We need to approximate all partial derivatives in (1.1), that is $\partial^2 u/\partial x^2, \partial^2 u/\partial y^2$ and $\partial^2 u/\partial t^2$ from $\nabla^2 u$, and u_{tt}. We achieve this using Taylor series as demonstrated in 1D already.

We start with $\partial^2 u/\partial x^2$ and evaluate u at nodes $(x_{i+1}, y_j, t_n) \in \Omega_{\mathrm{2D}} \times [0, T]$ and $(x_{i-1}, y_j, t_n) \in \Omega_{\mathrm{2D}} \times [0, T]$ to get

$$u(x_{i+1}, y_j, t_n) = u(x_i, y_j, t_n) + H_x u_x(x_i, y_j, t_n) + \frac{H_x^2}{2} u_{xx}(x_i, y_j, t_n) + \frac{H_x^3}{3!} u_{xxx}(x_i, y_j, t_n) + \mathcal{O}(H_x^4), \tag{2.20}$$

$$u(x_{i-1}, y_j, t_n) = u(x_i, y_j, t_n) - H_x u_x(x_i, y_j, t_n) + \frac{H_x^2}{2} u_{xx}(x_i, y_j, t_n)$$

$$- \frac{H_x^3}{3!} u_{xxx}(x_i, y_j, t_n) + \mathcal{O}(H_x^4) \tag{2.21}$$

respectively where $u_x \equiv \partial u / \partial x$. By combining both (2.20) and (2.21), we can cancel like terms resulting in

$$u(x_{i+1}, y_j, t_n) + u(x_{i-1}, y_j, t_n) = 2u(x_i, y_j, t_n) + H_x^2 u_{xx}(x_i, y_j, t_n) + \mathcal{O}(H_x^4). \tag{2.22}$$

Recall that we aim to approximate our partial derivative, denoted by $\partial^2 u / \partial x^2$, at all interior nodes. We know that $u_{xx} \equiv \partial^2 u / \partial x^2$, and so we make $u_{xx}(x_i, y_j, t_n)$ the subject of (2.22) to get

$$u_{xx}(x_i, y_j, t_n) = \frac{u(x_{i+1}, y_j, t_n) + u(x_{i-1}, y_j, t_n) - 2u(x_i, y_j, t_n)}{H_x^2} + \mathcal{O}(H_x^2). \tag{2.23}$$

By repeating steps (2.20)-(2.23), we can approximate $\partial^2 u / \partial y^2$ by changing the nodes u is evaluated at in $\Omega_{2\mathrm{D}}$ to $(x_i, y_{j+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ and $(x_i, y_{j-1}, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ to give

$$u_{yy}(x_i, y_j, t_n) = \frac{u(x_i, y_{j+1}, t_n) + u(x_i, y_{j-1}, t_n) - 2u(x_i, y_j, t_n)}{H_y^2} + \mathcal{O}(H_y^2). \tag{2.24}$$

By combining both (2.23) and (2.24), we have evaluated the first two derivatives in (1.1) to get

$$\begin{aligned} -\nabla^2 u(x_i, y_j, t_n) = & \frac{2u(x_i, y_j, t_n) - u(x_{i+1}, y_j, t_n) - u(x_{i-1}, y_j, t_n)}{H_x^2} \\ & + \frac{2u(x_i, y_j, t_n) - u(x_i, y_{j+1}, t_n) - u(x_i, y_{j-1}, t_n)}{H_y^2} \\ & + \mathcal{O}(H_x^2 + H_y^2). \end{aligned} \tag{2.25}$$

We now evaluate the final partial derivative in (1.1), denoted by u_{tt}, at the interior nodes. We achieve this by following the steps outlined above but instead change the nodes u is evaluated at in $\Omega_{2\mathrm{D}}$ to $(x_i, y_j, t_{n-1}) \in \Omega_{2\mathrm{D}} \times [0, T]$ and $(x_i, y_j, t_{n+1}) \in \Omega_{2\mathrm{D}} \times [0, T]$ to get

$$u_{tt}(x_i, y_j, t_n) = \frac{u(x_i, y_j, t_{n+1}) - 2u(x_i, y_j, t_n) + u(x_i, y_j, t_{n-1})}{\Delta t^2} + \mathcal{O}(\Delta t^2). \tag{2.26}$$

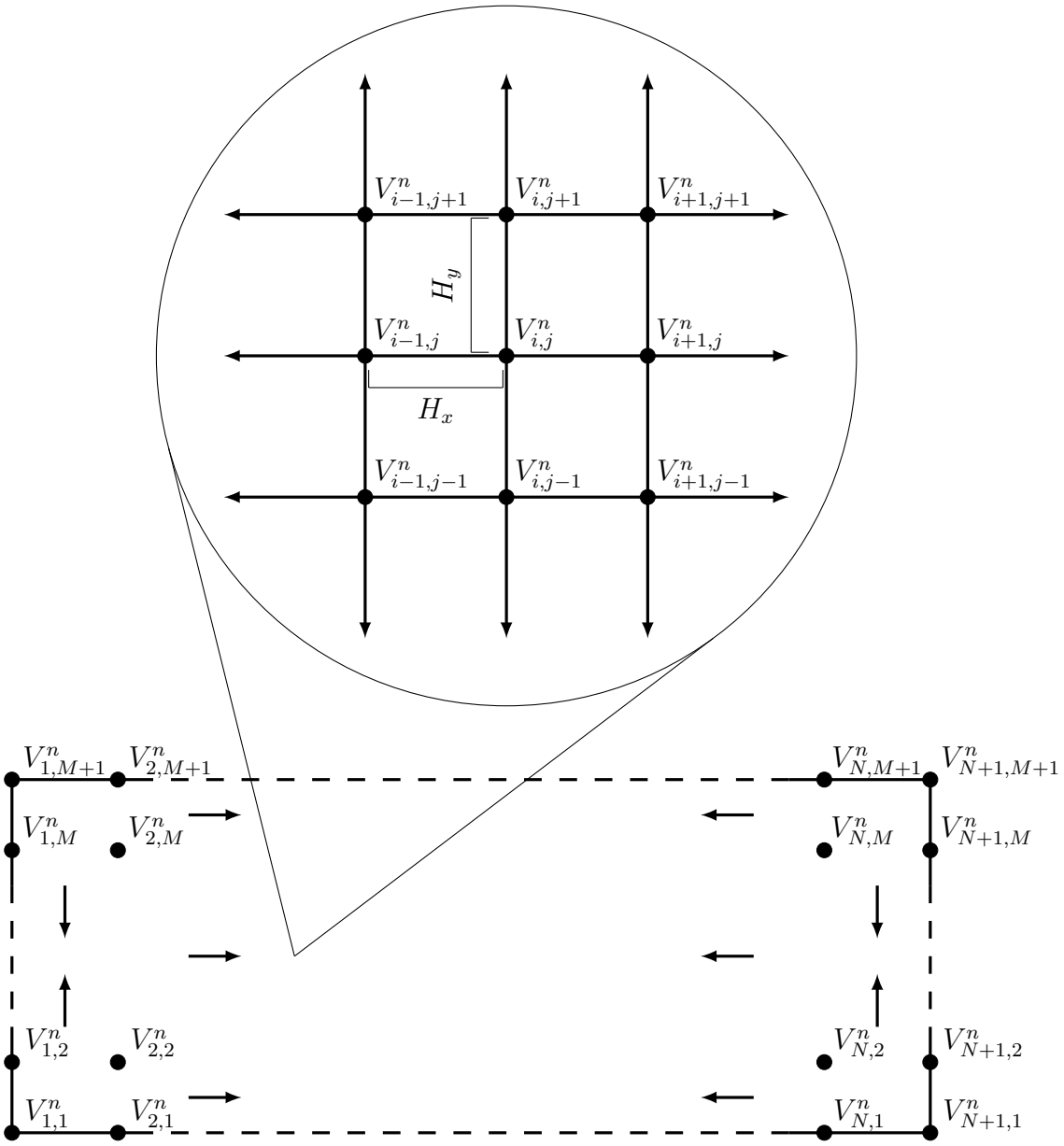


Figure 2.3: The discretisation of our 2D domain where H_x and H_y denote the spacing between nodes in our mesh for the x and y-directions respectively. Moreover, $V^n_{i,j}$ represents the explicit FDM approximation of u at the node denoted by (x_i, y_j) and discrete-time step t_n for $i \in \{1, 2, ..., N+1\}$, $j \in \{1, 2, ..., M+1\}$ and $n \in \{0, 1, 2, ..., L\}$.

Now we have evaluated all partial derivatives in (1.1), we substitute (2.25) and (2.26) into (1.1) to get

$$\frac{u(x_i, y_j, t_{n+1}) - 2u(x_i, y_j, t_n) + u(x_i, y_j, t_{n-1})}{\Delta t^2}$$
$$+ \frac{2u(x_i, y_j, t_n) - u(x_{i-1}, y_j, t_n) - u(x_{i+1}, y_j, t_n)}{H_x^2}$$

$$+\frac{2u(x_i,y_j,t_n)-u(x_i,y_{j-1},t_n)-u(x_i,y_{j+1},t_n)}{H_y^2}=f(x_i,y_j,t_n)+\mathcal{O}(H_x^2+H_y^2+\Delta t^2) \tag{2.27}$$

where $f(x_i,y_j,t_n)$ represents the forcing function in (1.1) at the node denoted by (x_i,y_j) and time step denoted by t_n. We are interested in evaluating u across the mesh, given initial conditions. We achieve this by making $u(x_i,y_j,t_{n+1})$ the subject of (2.27) resulting in

$$\begin{aligned}u(x_i,y_j,t_{n+1}) &= \Delta t^2 f(x_i,y_j,t_n)-u(x_i,y_j,t_{n-1})\\ &+\left(2-\frac{2\Delta t^2}{H_x^2}-\frac{2\Delta t^2}{H_y^2}\right)u(x_i,y_j,t_n)+\frac{\Delta t^2}{H_x^2}\Big(u(x_{i-1},y_j,t_n)+u(x_{i+1},y_j,t_n)\Big)\\ &+\frac{\Delta t^2}{H_y^2}\Big(u(x_i,y_{j-1},t_n)+u(x_i,y_{j+1},t_n)\Big)+\mathcal{O}(H_x^2+H_y^2+\Delta t^2).\end{aligned} \tag{2.28}$$

Lastly, we neglect the higher-order terms to arrive at an explicit FDM approximation of u at all interior nodes. Therefore, let us denote $V_{i,j}^n\approx u(x_i,y_j,t_n)$, and so we have

$$\begin{aligned}V_{i,j}^{n+1} &= \Delta t^2 f(x_i,y_j,t_n)-V_{i,j}^{n-1}+\left(2-\frac{2\Delta t^2}{H_x^2}-\frac{2\Delta t^2}{H_y^2}\right)V_{i,j}^n\\ &+\frac{\Delta t^2}{H_x^2}\Big(V_{i-1,j}^n+V_{i+1,j}^n\Big)+\frac{\Delta t^2}{H_y^2}\Big(V_{i,j-1}^n+V_{i,j+1}^n\Big)\end{aligned} \tag{2.29}$$

where $n\in\{1,2,...,L-1\}$.

In our two model problems, the boundary conditions defined on $\partial\Omega_{2D_1}$ and $\partial\Omega_{2D_2}$ are different, and so, need discretising separately. We note that in both model problems, Γ_{N_2} is a common boundary. Therefore, we start by discretising Γ_{N_2} which is defined at nodes denoted by $(x_i,y_{M+1},t_n)\in\Omega_{2D}\times[0,T]$ for $i\in\{2,3,...,N\}$. This is achieved by representing u at the nodes denoted by $(x_i,y_M,t_n)\in\Omega_{2D}\times[0,T]$ for $i\in\{2,3,...,N\}$ using Taylor series to get

$$u(x_i,y_M,t_n)=u(x_i,y_{M+1},t_n)-H_yu_y(x_i,y_{M+1},t_n)+\frac{H_y^2}{2}u_{yy}(x_i,y_{M+1},t_n)+\mathcal{O}(H_y^3). \tag{2.30}$$

We can represent both partial derivatives in (2.30) using the boundary condition and the strong form given in (1.1) at specific nodes. We achieve this by rearranging the latter such that $u_{yy}(x_i,y_{M+1},t_n)$ is the subject. Therefore, by substituting both into (2.30), we get

$$u(x_i,y_M,t_n)=u(x_i,y_{M+1},t_n)$$

$$+\frac{H_y^2}{2}\left[u_{tt}(x_i,y_{M+1},t_n)-u_{xx}(x_i,y_{M+1},t_n)-f(x_i,y_{M+1},t_n)\right]+\mathcal{O}(H_y^3) \tag{2.31}$$

where $f(x_i, y_{M+1}, t_n)$ is the forcing function in (1.1) at the nodes defined by (x_i, y_{M+1}) for $i \in \{2, 3, ..., N\}$ and time step t_n. However, this attempt to eliminate the partial derivatives in (2.30) has caused two more partial derivatives to arise. To evaluate the partial derivatives in (2.31), we use the central difference scheme at specific nodes in space and time given by

$$-u_{xx}(x_i,y_{M+1},t_n)=\frac{-u(x_{i-1},y_{M+1},t_n)+2u(x_i,y_{M+1},t_n)-u(x_{i+1},y_{M+1},t_n)}{H_x^2}+\mathcal{O}(H_x^2),$$
$$u_{tt}(x_i,y_{M+1},t_n)=\frac{u(x_i,y_{M+1},t_{n-1})-2u(x_i,y_{M+1},t_n)+u(x_i,y_{M+1},t_{n+1})}{\Delta t^2}+\mathcal{O}(\Delta t^2)$$

respectively. Hence, once we substitute both into (2.31), we get

$$\begin{aligned}u(x_i,y_M,t_n)&=u(x_i,y_{M+1},t_n)\\&+\frac{H_y^2}{2}\left[\frac{u(x_i,y_{M+1},t_{n-1})-2u(x_i,y_{M+1},t_n)+u(x_i,y_{M+1},t_{n+1})}{\Delta t^2}\right]\\&-\frac{H_y^2}{2}\left[\frac{u(x_{i-1},y_{M+1},t_n)-2u(x_i,y_{M+1},t_n)+u(x_{i+1},y_{M+1},t_n)}{H_x^2}\right]\\&-\frac{H_y^2}{2}f(x_i,y_{M+1},t_n)+\mathcal{O}(\Delta t^2+H_y^3+H_x^2).\end{aligned} \tag{2.32}$$

We now rearrange (2.32) to obtain an expression which represents a discrete solution of our PDE, denoted by u, at nodes along Γ_{N_2} and the $(n+1)^{\text{th}}$ time step. Therefore, we make $u(x_i, y_{M+1}, t_{n+1})$ the subject of (2.32) to get

$$\begin{aligned}u(x_i,y_{M+1},t_{n+1})&=\Delta t^2 f(x_i,y_{M+1},t_n)-u(x_i,y_{M+1},t_{n-1})+\frac{2\Delta t^2}{H_y^2}u(x_i,y_M,t_n)\\&+\left[2-2\Delta t^2\left(\frac{1}{H_x^2}+\frac{1}{H_y^2}\right)\right]u(x_i,y_{M+1},t_n)+\frac{\Delta t^2}{H_x^2}u(x_{i-1},y_{M+1},t_n)\\&+\frac{\Delta t^2}{H_x^2}u(x_{i+1},y_{M+1},t_n)+\frac{2\Delta t^2}{H_y^2}\mathcal{O}(H_x^2+H_y^3+\Delta t^2).\end{aligned} \tag{2.33}$$

Currently, the higher-order terms in (2.33) result in an error term in y of H_y. However, when we neglect these higher-order terms, we want the error term in y to be H_y^2. Therefore, noting that from the translated version of the original paper, see [38], where the convergence of a solution to a PDE when solved numerically was first discussed. Given our PDE in (1.1), the following condition is required to ensure our approximation of u in (2.33) is stable

$$\frac{\Delta t}{H_x}+\frac{\Delta t}{H_y}\leq C$$

where C is a positive constant. Since in our model the spacing in the x and y-directions, H_x and H_y, are equivalent we get

$$\frac{\Delta t}{H_y} \leq C. \tag{2.34}$$

Using this knowledge, we can replace the $\Delta t^2/H_y^2$ multiplier before the higher-order term in (2.33) with a constant squared, denoted by C^2, resulting in the desired higher-order terms. To obtain an approximation for u at the exterior nodes along Γ_{N_2}, we neglect these higher-order terms. Therefore, by denoting $V_{i,j}^n \approx u(x_i, y_j, t_n)$ we get

$$\begin{aligned} V_{i,M+1}^{n+1} &= \Delta t^2 f(x_i, y_{M+1}, t_n) - V_{i,M+1}^{n-1} + \frac{2\Delta t^2}{H_y^2} V_{i,M}^n \\ &\quad + \left[2 - 2\Delta t^2 \left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] V_{i,M+1}^n + \frac{\Delta t^2}{H_x^2} V_{i-1,M+1}^n + \frac{\Delta t^2}{H_x^2} V_{i+1,M+1}^n \end{aligned} \tag{2.35}$$

for $i \in \{2, 3, ..., N\}$.

We now look at the implementation of the boundary conditions defined on our Dirichlet boundary, denoted by Γ_{D_2}. Since there are no partial derivatives in boundary conditions defined on Dirichlet boundaries, we do not need to approximate anything. Instead, we ensure that $V_{i,j}^{n+1} = 0$ under the following conditions

1. $\forall i \in \{1, 2, ..., N+1\}$ when $j := 1$,

2. $\forall j \in \{1, 2, ..., M+1\}$ when $i := 1$ and $i := N+1$.

We now consider the boundary for our second model problem, denoted by $\partial\Omega_{2\mathrm{D}_2} := \left\{\overline{\Gamma_{\mathrm{N}_2} \cup \Gamma_{\mathrm{N}_3} \cup \Gamma_{\mathrm{N}_4} \cup \Gamma_{\mathrm{D}_3}}\right\}$. Currently, we have only approximated u at nodes along Γ_{N_2}, meaning we still need to approximate u on nodes along Γ_{N_3}, Γ_{N_4} and Γ_{D_3}.

Starting with Γ_{N_3}, we know that $\partial u/\partial x = 0$ is satisfied along the nodes denoted by $(x_1, y_j, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ for $j \in \{2, 3, ..., M+1\}$. Since we have a partial derivative, we need to approximate u at the nodes along Γ_{N_2} using Taylor series. We start by approximating u at the nodes defined by $(x_2, y_j, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ for $j \in \{2, 3, ..., M+1\}$, resulting in

$$u(x_2, y_j, t_n) = u(x_1, y_j, t_n) + H_x u_x(x_1, y_j, t_n) + \frac{H_x^2}{2} u_{xx}(x_1, y_j, t_n) + \mathcal{O}(H_x^3). \tag{2.36}$$

We can eliminate the partial derivatives using the boundary condition and a rearrangement of (1.1) to get

$$u(x_2, y_j, t_n) = u(x_1, y_j, t_n) + \frac{H_x^2}{2}\Big[u_{tt}(x_1, y_j, t_n) - u_{yy}(x_1, y_j, t_n) - f(x_1, y_j, t_n)\Big] + \mathcal{O}(H_x^3). \tag{2.37}$$

Despite attempting to remove the partial derivatives in (2.36), two new partial derivatives have arisen in (2.37). We can eliminate these using the central difference scheme, denoted by

$$-u_{yy}(x_1, y_j, t_n) = \frac{-u(x_1, y_{j-1}, t_n) + 2u(x_1, y_j, t_n) - u(x_1, y_{j+1}, t_n)}{H_y^2} + \mathcal{O}(H_y^2) \text{ and}$$

$$u_{tt}(x_1, y_j, t_n) = \frac{u(x_1, y_j, t_{n-1}) - 2u(x_1, y_j, t_n) + u(x_1, y_j, t_{n+1})}{\Delta t^2} + \mathcal{O}(\Delta t^2).$$

By substituting these both into (2.37) we get

$$\begin{aligned} u(x_2, y_j, t_n) = u(x_1, y_j, t_n) + \frac{H_x^2}{2}\Bigg[&\frac{u(x_1, y_j, t_{n-1}) - 2u(x_1, y_j, t_n) + u(x_1, y_j, t_{n+1})}{\Delta t^2} \\ &+ \frac{2u(x_1, y_j, t_n) - u(x_1, y_{j-1}, t_n) - u(x_1, y_{j+1}, t_n)}{H_y^2} - f(x_1, y_j, t_n)\Bigg] \\ &+ \mathcal{O}(H_x^3 + H_y^2 + \Delta t^2). \end{aligned} \tag{2.38}$$

Recall that we are interested in evaluating u at the nodes along Γ_{N_3} denoted by $(x_1, y_j, t_n) \in \Omega_{\mathrm{2D}} \times [0, T]$ for $j \in \{2, 3, ..., M+1\}$. Therefore, we rearrange (2.38) to make $u(x_1, y_j, t_{n+1})$ the subject to get

$$\begin{aligned} u(x_1, y_j, t_{n+1}) = \frac{2\Delta t^2}{H_x^2} u(x_2, y_j, t_n) + u(x_1, y_j, t_n)&\left[2 - \frac{2\Delta t^2}{H_x^2} - \frac{2\Delta t^2}{H_y^2}\right] \\ &- u(x_1, y_j, t_{n-1}) + \frac{\Delta t^2}{H_y^2}\Big[u(x_1, y_{j-1}, t_n) + u(x_1, y_{j+1}, t_n)\Big] \\ &+ \Delta t^2 f(x_1, y_j, t_n) + \frac{2\Delta t^2}{H_x^2}\mathcal{O}(H_x^3 + H_y^2 + \Delta t^2). \end{aligned} \tag{2.39}$$

The higher-order terms in (2.39) mean we only have an accuracy of H_x in the x-direction, rather than the desired accuracy of H_x^2. We can rectify this by recalling that for (1.1) to be stable (2.34) holds. Therefore, substituting this into the $\Delta t^2/H_x^2$ multiplier before the higher-order terms in (2.39) results in the desired level of accuracy in the x-direction. We now want to approximate u along Γ_{N_3}. We achieve this by neglecting the higher-order terms in (2.39), and by denoting $V_{i,j}^n \approx u(x_i, y_j, t_n)$, we get

$$\begin{aligned} V_{1,j}^{n+1} = \frac{2\Delta t^2}{H_x^2} V_{2,j}^n + \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] V_{1,j}^n - V_{1,j}^{n-1} \\ + \frac{\Delta t^2}{H_y^2}\left(V_{1,j-1}^n + V_{1,j+1}^n\right) + \Delta t^2 f(x_1, y_j, t_n) \end{aligned} \tag{2.40}$$

for $j \in \{2, 3, ..., M\}$. Since $V_{1,j+1}^n$ is required to approximate $V_{1,j}^{n+1}$, (2.40) cannot approximate u at all the nodes along Γ_{N_3} because when $j := M + 1$ we would require

$V^n_{1,M+2}$ to compute $V^{n+1}_{1,M+1}$, which does not exist.

Instead, to obtain the approximation of u at the node defined by $(x_1, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0,T]$, we must take a different approach. Using Taylor series, we evaluate u at the nodes defined by $(x_2, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0,T]$ and $(x_1, y_M, t_n) \in \Omega_{2\mathrm{D}} \times [0,T]$ to get

$$u(x_2, y_{M+1}, t_n) = u(x_1, y_{M+1}, t_n) + H_x u_x(x_1, y_{M+1}, t_n) + \frac{H_x^2}{2} u_{xx}(x_1, y_{M+1}, t_n) + \mathcal{O}(H_x^3), \tag{2.41}$$

$$u(x_1, y_M, t_n) = u(x_1, y_{M+1}, t_n) - H_y u_y(x_1, y_{M+1}, t_n) + \frac{H_y^2}{2} u_{yy}(x_1, y_{M+1}, t_n) + \mathcal{O}(H_y^3) \tag{2.42}$$

respectively. At the node defined by (x_1, y_{M+1}, t_n), both Γ_{N_2} and Γ_{N_3} hold, and so, the boundary conditions along these two boundaries are also valid. Therefore, we know that $u_x(x_1, y_{M+1}, t_n) = 0$ and $u_y(x_1, y_{M+1}, t_n) = 0$. Substituting these into both (2.41) and (2.42), we get

$$u(x_2, y_{M+1}, t_n) = u(x_1, y_{M+1}, t_n) + \frac{H_x^2}{2} u_{xx}(x_1, y_{M+1}, t_n) + \mathcal{O}(H_x^3), \tag{2.43}$$

$$u(x_1, y_M, t_n) = u(x_1, y_{M+1}, t_n) + \frac{H_y^2}{2} u_{yy}(x_1, y_{M+1}, t_n) + \mathcal{O}(H_y^3). \tag{2.44}$$

By multiplying (2.43) by H_y^2 and (2.44) by H_x^2, adding the two equations together and substituting in (1.1), we can eliminate the spatial second-order partial derivatives to get

$$H_y^2 u(x_2, y_{M+1}, t_n) + H_x^2 u(x_1, y_M, t_n) = H_y^2 u(x_1, y_{M+1}, t_n) + H_x^2 u(x_1, y_{M+1}, t_n) + \frac{H_x^2 H_y^2}{2} \Big[u_{tt}(x_1, y_{M+1}, t_n) - f(x_1, y_{M+1}, t_n) \Big] + \mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3). \tag{2.45}$$

Using the central difference scheme at the nodes defined as $(x_1, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0,T]$, that is

$$u_{tt}(x_1, y_{M+1}, t_n) = \frac{u(x_1, y_{M+1}, t_{n-1}) - 2u(x_1, y_{M+1}, t_n) + u(x_1, y_{M+1}, t_{n+1})}{\Delta t^2} + \mathcal{O}(\Delta t^2),$$

we can eliminate the final partial derivative in (2.45) to get

$$H_y^2 u(x_2, y_{M+1}, t_n) + H_x^2 u(x_1, y_M, t_n) = H_y^2 u(x_1, y_{M+1}, t_n) + H_x^2 u(x_1, y_{M+1}, t_n) + \frac{H_x^2 H_y^2}{2} \left[\frac{u(x_1, y_{M+1}, t_{n-1}) - 2u(x_1, y_{M+1}, t_n) + u(x_1, y_{M+1}, t_{n+1})}{\Delta t^2} - f(x_1, y_{M+1}, t_n) \right] + \mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3 + H_x^2 H_y^2 \Delta t^2). \tag{2.46}$$

Recall that we want to approximate u at the node defined by (x_1, y_{M+1}, t_n), across all time intervals. Therefore, we make $u(x_1, y_{M+1}, t_{n+1})$ the subject of (2.46) to get

$$\begin{aligned} u(x_1, y_{M+1}, t_{n+1}) &= \Delta t^2 f(x_1, y_{M+1}, t_n) - u(x_1, y_{M+1}, t_{n-1}) \\ &+ \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] u(x_1, y_{M+1}, t_n) \\ &+ \frac{2\Delta t^2}{H_x^2} u(x_2, y_{M+1}, t_n) + \frac{2\Delta t^2}{H_y^2} u(x_1, y_M, t_n) \\ &+ \frac{2\Delta t^2}{H_x^2 H_y^2} \mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3 + H_x^2 H_y^2 \Delta t^2). \end{aligned} \tag{2.47}$$

To approximate u we need to neglect the higher-order terms in (2.47). But first, we must ensure the approximation is sufficiently accurate. We start by recalling that for our PDE in (1.1), the approximation of u in (2.47) is stable if the condition in (2.34) holds. Substituting this into the multiplier before the higher-order terms in (2.47), we get

$$\begin{aligned} &\frac{2C^2}{H_y^2} \mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3 + H_x^2 H_y^2 \Delta t^2), \\ &= \mathcal{O}(H_x^3 + H_x^2 H_y + H_x^2 \Delta t^2). \end{aligned}$$

In our model, the spacing in both the x and y-directions are equivalent. Therefore, using the fact that $H_x \equiv H_y$, we get our higher-order terms to be

$$\mathcal{O}(H_x^3 + H_x^2 \Delta t^2). \tag{2.48}$$

Now we have determined that the higher-order terms in (2.47) result in a sufficiently accurate solution for u, we neglect these higher-order terms and denote $V_{i,j}^n \approx u(x_i, y_j, t_n)$ to get

$$\begin{aligned} V_{1,M+1}^{n+1} &= \Delta t^2 f(x_1, y_{M+1}, t_n) - V_{1,M+1}^{n-1} + \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] V_{1,M+1}^n \\ &+ \frac{2\Delta t^2}{H_x^2} V_{2,M+1}^n + \frac{2\Delta t^2}{H_y^2} V_{1,M}^n \end{aligned} \tag{2.49}$$

for $n \in \{1, 2, ..., L-1\}$.

Having approximated u along Γ_{N_3}, we move on to Γ_{N_4}. The boundary condition defined along this boundary is given by $\partial u/\partial x := 0$, which is satisfied along the nodes denoted by $(x_{N+1}, y_j, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ for $j \in \{2, 3, ..., M+1\}$. We start by representing u using Taylor series at the nodes denoted by $(x_N, y_j, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ for $j \in \{2, 3, ..., M+1\}$, to get

$$u(x_N, y_j, t_n) = u(x_{N+1}, y_j, t_n) - H_x u_x(x_{N+1}, y_j, t_n) + \frac{H_x^2}{2} u_{xx}(x_{N+1}, y_j, t_n) + \mathcal{O}(H_x^3). \tag{2.50}$$

We can eliminate the partial derivatives in (2.50) using the boundary condition and a rearrangement of (1.1) to get

$$u(x_N, y_j, t_n) = u(x_{N+1}, y_j, t_n) + \frac{H_x^2}{2}\Big[u_{tt}(x_{N+1}, y_j, t_n) - u_{yy}(x_{N+1}, y_j, t_n) - f(x_{N+1}, y_j, t_n)\Big] + \mathcal{O}(H_x^3). \tag{2.51}$$

This technique has resulted in two more partial derivatives to arise. To evaluate the partial derivatives in (2.51), we use the central difference scheme at specific nodes in space and time given by

$$u_{tt}(x_{N+1}, y_j, t_n) = \frac{u(x_{N+1}, y_j, t_{n-1}) - 2u(x_{N+1}, y_j, t_n) + u(x_{N+1}, y_j, t_{n+1})}{\Delta t^2} + \mathcal{O}(\Delta t^2),$$

$$-u_{yy}(x_{N+1}, y_j, t_n) = \frac{2u(x_{N+1}, y_j, t_n) - u(x_{N+1}, y_{j-1}, t_n) - u(x_{N+1}, y_{j+1}, t_n)}{H_y^2} + \mathcal{O}(H_y^2).$$

Therefore, after substituting both central difference schemes into (2.51), we get

$$\begin{aligned} u(x_N, y_j, t_n) = u(x_{N+1}, y_j, t_n) &+ \frac{H_x^2}{2}\Bigg[\frac{u(x_{N+1}, y_j, t_{n-1}) - 2u(x_{N+1}, y_j, t_n) + u(x_{N+1}, y_j, t_{n+1})}{\Delta t^2} \\ &+ \frac{2u(x_{N+1}, y_j, t_n) - u(x_{N+1}, y_{j-1}, t_n) - u(x_{N+1}, y_{j+1}, t_n)}{H_y^2} - f(x_{N+1}, y_j, t_n)\Bigg] \\ &+ \mathcal{O}(H_x^3 + H_y^2 + \Delta t^2). \end{aligned} \tag{2.52}$$

Recall that we are interested in evaluating u at the nodes along Γ_{N_4}, that is $(x_{N+1}, y_j, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ for $j \in \{2, 3, ..., M+1\}$. To do this, we rearrange (2.52) to make $u(x_{N+1}, y_j, t_{n+1})$ the subject which results in

$$\begin{aligned} u(x_{N+1}, y_j, t_{n+1}) = \Delta t^2 f(x_{N+1}, y_j, t_n) - u(x_{N+1}, y_j, t_{n-1}) &+ \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right]u(x_{N+1}, y_j, t_n) + \frac{2\Delta t^2}{H_x^2}u(x_N, y_j, t_n) \\ &+ \frac{\Delta t^2}{H_y^2}\Big[u(x_{N+1}, y_{j-1}, t_n) + u(x_{N+1}, y_{j+1}, t_n)\Big] + \frac{2\Delta t^2}{H_x^2}\mathcal{O}(H_x^3 + H_y^2 + \Delta t^2). \end{aligned} \tag{2.53}$$

Recall that for our PDE in (1.1), the approximation of u in (2.53) is stable if the condition in (2.34) holds. Using this, we deduce the higher-order terms in (2.53) to be

$$\mathcal{O}(H_x^3 + H_y^2 + \Delta t^2) \tag{2.54}$$

which illustrates a sufficient level of accuracy for what will be our approximation of u. To find this approximation of u at the nodes along Γ_{N_4}, we denote $V_{i,j}^n \approx u(x_i, y_j, t_n)$, and neglect the higher-order terms in (2.53) to get

$$V_{N+1,j}^{n+1} = \Delta t^2 f(x_{N+1}, y_j, t_n) - V_{N+1,j}^{n-1} + \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] V_{N+1,j}^n + \frac{2\Delta t^2}{H_x^2} V_{N,j}^n + \frac{\Delta t^2}{H_y^2}\left(V_{N+1,j-1}^n + V_{N+1,j+1}^n\right) \tag{2.55}$$

for $j \in \{2, 3, ..., M\}$. Since we require $V_{N+1,j+1}^n$ to compute $V_{N+1,j}^{n+1}$, (2.55) cannot approximate u at all the nodes along Γ_{N_4} because when $j := M + 1$, we would require $V_{N+1,M+2}^n$ to compute $V_{N+1,M+1}^{n+1}$, which does not exist.

Instead, to obtain the approximation of u at the node defined by $(x_{N+1}, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ we must take a different approach. Using Taylor series, we evaluate u at the nodes defined by $(x_N, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ and $(x_{N+1}, y_M, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$ to get

$$u(x_N, y_{M+1}, t_n) = u(x_{N+1}, y_{M+1}, t_n) - H_x u_x(x_{N+1}, y_{M+1}, t_n) + \frac{H_x^2}{2} u_{xx}(x_{N+1}, y_{M+1}, t_n) + \mathcal{O}(H_x^3), \tag{2.56}$$

$$u(x_{N+1}, y_M, t_n) = u(x_{N+1}, y_{M+1}, t_n) - H_y u_y(x_{N+1}, y_{M+1}, t_n) + \frac{H_y^2}{2} u_{yy}(x_{N+1}, y_{M+1}, t_n) + \mathcal{O}(H_y^3). \tag{2.57}$$

At the node defined by (x_{N+1}, y_{M+1}, t_n), both Γ_{N_2} and Γ_{N_4} are valid boundaries. We know that the boundary conditions on these two boundaries are satisfied. Therefore, we know that $u_x(x_{N+1}, y_{M+1}, t_n) = 0$ and $u_y(x_{N+1}, y_{M+1}, t_n) = 0$. Substituting these into both (2.56) and (2.57), we get

$$u(x_N, y_{M+1}, t_n) = u(x_{N+1}, y_{M+1}, t_n) + \frac{H_x^2}{2} u_{xx}(x_{N+1}, y_{M+1}, t_n) + \mathcal{O}(H_x^3), \tag{2.58}$$

$$u(x_{N+1}, y_M, t_n) = u(x_{N+1}, y_{M+1}, t_n) + \frac{H_y^2}{2} u_{yy}(x_{N+1}, y_{M+1}, t_n) + \mathcal{O}(H_y^3). \tag{2.59}$$

By multiplying (2.58) by H_y^2 and (2.59) by H_x^2, adding the two equations together and substituting in (1.1), we can eliminate the spatial second-order partial derivatives to get

$$H_y^2 u(x_N, y_{M+1}, t_n) - H_y^2 u(x_{N+1}, y_{M+1}, t_n) + H_x^2 u(x_{N+1}, y_M, t_n) - H_x^2 u(x_{N+1}, y_{M+1}, t_n) = \frac{H_x^2 H_y^2}{2}\left[u_{tt}(x_{N+1}, y_{M+1}, t_n) - f(x_{N+1}, y_{M+1}, t_n)\right] + \mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3). \tag{2.60}$$

This technique has caused another partial derivative to arise. To evaluate the partial derivative in (2.60), we use the central difference scheme at a specific node in space given by

$$u_{tt}(x_{N+1}, y_{M+1}, t_n) = \frac{u(x_{N+1}, y_{M+1}, t_{n-1}) - 2u(x_{N+1}, y_{M+1}, t_n) + u(x_{N+1}, y_{M+1}, t_{n+1})}{\Delta t^2} + \mathcal{O}(\Delta t^2).$$

Substituting this central difference scheme into (2.60), we eliminate the last partial derivative to get

$$\begin{aligned} &H_y^2 u(x_N, y_{M+1}, t_n) - H_y^2 u(x_{N+1}, y_{M+1}, t_n) + H_x^2 u(x_{N+1}, y_M, t_n) - H_x^2 u(x_{N+1}, y_{M+1}, t_n) \\ &+ \mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3 + H_x^2 H_y^2 \Delta t^2) = \frac{H_x^2 H_y^2}{2\Delta t^2}\Big[u(x_{N+1}, y_{M+1}, t_{n-1}) - 2u(x_{N+1}, y_{M+1}, t_n) \\ &+ u(x_{N+1}, y_{M+1}, t_{n+1}) - \Delta t^2 f(x_{N+1}, y_{M+1}, t_n)\Big]. \end{aligned} \tag{2.61}$$

Recall that we want to approximate u at the node denoted by $(x_{N+1}, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$. Therefore, we make $u(x_{N+1}, y_{M+1}, t_{n+1})$ the subject of (2.61) resulting in

$$\begin{aligned} u(x_{N+1}, y_{M+1}, t_{n+1}) &= \Delta t^2 f(x_{N+1}, y_{M+1}, t_n) - u(x_{N+1}, y_{M+1}, t_{n-1}) \\ &+ \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] u(x_{N+1}, y_{M+1}, t_n) + \frac{2\Delta t^2}{H_x^2} u(x_N, y_{M+1}, t_n) \\ &+ \frac{2\Delta t^2}{H_y^2} u(x_{N+1}, y_M, t_n) + \frac{2\Delta t^2}{H_x^2 H_y^2}\mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3 + H_x^2 H_y^2 \Delta t^2). \end{aligned} \tag{2.62}$$

Recall that for our PDE in (1.1), our approximation of u in (2.62) is stable if the condition in (2.34) holds. Using this, we can deduce the higher-order terms in (2.62) to be

$$\begin{aligned} &\frac{2C^2}{H_y^2}\mathcal{O}(H_x^3 H_y^2 + H_x^2 H_y^3 + H_x^2 H_y^2 \Delta t^2), \\ &= \mathcal{O}(H_x^3 + H_x^2 H_y + H_x^2 \Delta t^2). \end{aligned}$$

Since our model has the same spacing in both the x and y-directions, that is $H_x \equiv H_y$, we get our higher-order terms in (2.62) to be

$$\mathcal{O}(H_x^3 + H_x^2 \Delta t^2) \tag{2.63}$$

which yields a sufficient level of accuracy for what will be our approximation of u. To find this approximation of u at the specific node denoted by $(x_{N+1}, y_{M+1}, t_n) \in \Omega_{2\mathrm{D}} \times [0, T]$,

across all time steps, we denote $V_{i,j}^n \approx u(x_i, y_j, t_n)$ and neglect the higher-order terms in (2.62) to get

$$V_{N+1,M+1}^{n+1} = \Delta t^2 f(x_{N+1}, y_{M+1}, t_n) - V_{N+1,M+1}^{n-1} + \left[2 - 2\Delta t^2\left(\frac{1}{H_x^2} + \frac{1}{H_y^2}\right)\right] V_{N+1,M+1}^n + \frac{2\Delta t^2}{H_x^2} V_{N,M+1}^n + \frac{2\Delta t^2}{H_y^2} V_{N+1,M}^n \tag{2.64}$$

for $n \in \{1, 2, ..., L-1\}$.

Lastly, we now look at the implementation of the boundary conditions defined on the Dirichlet boundary, denoted by Γ_{D_3}. Since there are no partial derivatives in these boundary conditions, we do not need to approximate anything. Instead, we ensure that $V_{i,j}^{n+1} = 0$ for all $i \in \{1, 2, ..., N+1\}$ when $j := 1$.

Now we have equations to approximate u at every node in our mesh across all discrete-time steps in both model problems, we construct a matrix system for each to approximate u in (1.1). Given that our domain, denoted by $\Omega_{2\mathrm{D}}$, has $(N+1) \times (M+1)$ nodes we define $\mathbf{V}^n$ and $\boldsymbol{\mathcal{F}}^n$ below:

$$\mathbf{V}^n = \begin{pmatrix} V_{1,1}^n & V_{2,1}^n & \cdots & V_{N+1,1}^n & V_{1,2}^n & V_{2,2}^n & \cdots\cdots & V_{N,M+1}^n & V_{N+1,M+1}^n \end{pmatrix}^{\mathrm{T}},$$

$$\boldsymbol{\mathcal{F}}^n = \Big(f(x_1, y_1, t_n) \quad f(x_2, y_1, t_n) \quad \cdots \quad f(x_{N+1}, y_1, t_n) \quad f(x_1, y_2, t_n) \quad f(x_2, y_2, t_n) \quad \cdots \quad \cdots \quad f(x_N, y_{M+1}, t_n) \quad f(x_{N+1}, y_{M+1}, t_n) \Big)^{\mathrm{T}}.$$

In addition to this, we define a matrix $\mathbf{B}$ which contains all the coefficients of $V_{i,j}^n$ for $i \in \{1, 2, 3, ..., N+1\}$ and $j \in \{1, 2, 3, ..., M+1\}$ in (2.29), (2.35), (2.40), (2.49), (2.55), (2.64), and the Dirichlet boundaries defined above. Having defined all of our notation, the matrix system is given by

$$\mathbf{V}^{n+1} = \mathbf{B}\mathbf{V}^n - \mathbf{V}^{n-1} + \Delta t^2 \boldsymbol{\mathcal{F}}^n \tag{2.65}$$

for $n \in \{1, 2, ..., L-1\}$.

2.2.1 MATLAB implementation

In this section, we look at verifying whether the MATLAB code which approximates u in (1.1) using (2.65) works correctly for both model problems. We achieve this by computing the maximum absolute error between an exact solution of our PDE and an approximated solution of our PDE across all nodes at $t_L = T$. For both model problems,

we take the total simulation time to be $T = 1$. The two model problems have different boundary conditions, and so require different exact solutions for u which satisfy them accordingly.

First Model Problem

The boundary of our first model problem is defined by $\partial\Omega_{2\mathrm{D}_1}$. In the following sections, we construct an exact solution for u such that all boundary conditions are satisfied and test the convergence of the error which arises from the FDM approximation of u.

Time and space-dependent manufactured solution

We choose an exact solution for u which contains trigonometric functions since this is similar to the sensor traces we would expect to observe using our forcing function in (1.6) within our acoustic wave equation. Therefore, a reasonable choice for u such that all boundary conditions are satisfied is

$$u(x_i, y_j, t_n) = \sin\left(\frac{2\pi x_i}{0.2}\right)\sin\left(\frac{2\pi y_j}{0.08}\right)\Big(1 + \sin\left(t_n\right)\Big) \tag{2.66}$$

where $i \in \{1, 2, ..., N+1\}$, $j \in \{1, 2, ..., M+1\}$ and $n \in \{0, 1, ..., L\}$. Subsequently, by substituting (2.66) into (1.1) we find the forcing function, denoted by f, which will be used in this convergence test. Hence, we get

$$f(x_i, y_j, t_n) = \sin\left(\frac{2\pi x_i}{0.2}\right)\sin\left(\frac{2\pi y_j}{0.08}\right)\Big(725\pi^2\big(1 + \sin(t_n)\big) - \sin(t_n)\Big). \tag{2.67}$$

Recall the matrix system in (2.65) which approximates u in (1.1) at every node across all time steps. The matrix system is recursive and so initially requires both $\mathbf{V}^0$ and $\mathbf{V}^1$ to be known. That is, we need the approximation of u at $t_0 := 0$ and $t_1 := \Delta t$ respectively. We have two initial conditions given by $u(x_i, y_j, t_0)$ and $u_t(x_i, y_j, t_0)$ for $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$. Therefore, $\mathbf{V}^0$ is known. However, to get $\mathbf{V}^1$, we need to use Taylor series and both initial conditions.

In this convergence test, we know the exact solution to u, and so by using (2.66), the initial conditions across all nodes, defined for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$, are both

$$u(x_i, y_j, t_0) = u_t(x_i, y_j, t_0) = \sin\left(\frac{2\pi x_i}{0.2}\right)\sin\left(\frac{2\pi y_j}{0.08}\right). \tag{2.68}$$

Therefore, by definition in (2.68), $\mathbf{V}^0 := \sin\left(\frac{2\pi x_i}{0.2}\right)\sin\left(\frac{2\pi y_j}{0.08}\right)$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$. Using Taylor series and substituting in (2.68), we get

$$\begin{aligned} u(x_i, y_j, t_1) &= u(x_i, y_j, t_0) + \Delta t u_t(x_i, y_j, t_0) + \mathcal{O}(\Delta t^2), \\ &= \sin\left(\frac{2\pi x_i}{0.2}\right)\sin\left(\frac{2\pi y_j}{0.08}\right)(1 + \Delta t) + \mathcal{O}(\Delta t^2). \end{aligned} \tag{2.69}$$

By neglecting the higher-order terms in (2.69), for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$ we get

$$\mathbf{V}^1 := \sin\left(\frac{2\pi x_i}{0.2}\right)\sin\left(\frac{2\pi y_j}{0.08}\right)(1 + \Delta t)\,.$$

Error convergence

Recall that our approximation to (1.1) became approximate because we neglected the higher-order terms in (2.28) and (2.33). These higher-order terms directly correlate to the magnitude of error we would expect to get between an exact solution and our approximation of u. Recall that the higher-order terms were $\mathcal{O}(H_x^2 + H_y^2 + \Delta t^2)$ at the interior nodes and $\mathcal{O}(H_x^2 + H_y^3 + \Delta t^2)$ along the Neumann boundary. It makes sense that the maximum error recorded would correlate to the higher-order terms resulting in the most substantial error. Therefore to find out what these are, we combined both higher-order terms to get

$$\mathcal{O}\left(H_x^2 + H_y^2 + \Delta t^2 + H_x^2 + H_y^3 + \Delta t^2\right) = \mathcal{O}(H_x^2 + H_y^2 + \Delta t^2).$$

Therefore, we would expect the maximum absolute error to decrease by a factor of four as H_x, H_y and Δt half. Table (2.2) displays the maximum absolute error across all nodes in our mesh at $T = 1$ between an exact solution in (2.66) and our approximations of u for increasing values of N and M, which subsequently decreases H_x and H_y, respectively, and decreasing Δt. We can deduce as both N and M double and Δt halves, the error convergence approaches four which is what we expected. Therefore, we can conclude that the mathematics set-out in this section and the code are correct and working.

N	M	Δt	Maximum |Error|	Error Percentage†	Error Convergence	Computation Time†† (s)
30	3	1800^{-1}	0.046423654	2.472227298	N/A	0.038300
60	6	3600^{-1}	0.013846597	0.746319520	3.352712150	0.137714
120	12	7200^{-1}	0.003532304	0.191452475	3.919990182	0.975390
240	24	14400^{-1}	0.000886343	0.048109192	3.985256272	7.748006
480	48	28800^{-1}	0.000221782	0.012042293	3.996460488	63.326669
960	96	57600^{-1}	0.000055459	0.003011588	3.999026308	577.356112

† Error Percentage = [maximum error at node (i, j)]/[approximation at node (i, j)] $\times$ 100.

†† Computed using an i5 − 4590 @ 3.30GHz, 16GB RAM @ 1333 MHz and Intel HD Graphics 4600.

Table 2.2: The error convergence for our explicit FDM approximation of u from the acoustic 2D wave equation, at $T = 1$, with the boundary defined on $\partial\Omega_{2D_1}$.

The Lax equivalent theorem states that if a finite difference scheme is both stable and consistent, then it is convergent [37]. We already know that our finite difference scheme in (2.65) is stable, see [38]. For our finite difference scheme to be consistent, the maximum absolute error in Table (2.2) must approach zero as $H_x = 0.2/N$, $H_y = 0.02/M$ and Δt approach zero. Upon inspection of Table (2.2), we can see that this is the case. Therefore, we can conclude that our finite difference scheme is convergent.

Second Model Problem

The boundary of our second model problem is defined by $\partial\Omega_{2D_2}$. In the following sections, we construct an exact solution for u such that all boundary conditions are satisfied and test the convergence of the error which arises from the FDM approximation of u.

Time and space dependent manufactured solution

We choose an exact solution for u which contains trigonometric functions since this is similar to the sensor traces we would expect to observe using our forcing function in (1.6) within our acoustic wave equation. Therefore, a reasonable choice for u such that all boundary conditions are satisfied is

$$u(x_i, y_j, t_n) = \cos\left(\frac{2\pi x_i}{0.4}\right) \sin\left(\frac{2\pi y_j}{0.08}\right)\Big(1 + \sin(t_n)\Big) \tag{2.70}$$

where $i \in \{1, 2, ..., N+1\}$, $j \in \{1, 2, ..., M+1\}$ and $n \in \{0, 1, ..., L\}$. Subsequently, by substituting (2.70) into (1.1) we find the forcing function, denoted by f, which will be used in this convergence test. Hence, we get

$$f(x_i, y_j, t_n) = \cos\left(\frac{2\pi x_i}{0.4}\right) \sin\left(\frac{2\pi y_j}{0.08}\right) \Big(650\pi^2\big(1 + \sin(t_n)\big) - \sin(t_n)\Big). \tag{2.71}$$

Recall the matrix system in (2.65) which approximates u in (1.1) at every node across all time steps. The matrix system is recursive and so initially requires both $\mathbf{V}^0$ and $\mathbf{V}^1$ to be known. That is, we need the approximation of u at $t_0 := 0$ and $t_1 := \Delta t$ respectively. We have two initial conditions given by $u(x_i, y_j, t_0)$ and $u_t(x_i, y_j, t_0)$ for $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$. Therefore, $\mathbf{V}^0$ is known. However, to get $\mathbf{V}^1$, we need to use Taylor series and both initial conditions.

In this convergence test, we know the exact solution to u, and so by using (2.70), the initial conditions across all nodes, defined for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$, are both

$$u(x_i, y_j, t_0) = u_t(x_i, y_j, t_0) = \cos\left(\frac{2\pi x_i}{0.4}\right) \sin\left(\frac{2\pi y_j}{0.08}\right). \tag{2.72}$$

Therefore, by definition in (2.72), $\mathbf{V}^0 := \cos\left(\frac{2\pi x_i}{0.4}\right) \sin\left(\frac{2\pi y_j}{0.08}\right)$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$. Using Taylor series and substituting in (2.72), we get

$$\begin{aligned} u(x_i, y_j, t_1) &= u(x_i, y_j, t_0) + \Delta t u_t(x_i, y_j, t_0) + \mathcal{O}(\Delta t^2), \\ &= \cos\left(\frac{2\pi x_i}{0.4}\right) \sin\left(\frac{2\pi y_j}{0.08}\right)(1 + \Delta t) + \mathcal{O}(\Delta t^2). \end{aligned} \tag{2.73}$$

By neglecting the higher-order terms in (2.73), for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$ we get

$$\mathbf{V}^1 := \cos\left(\frac{2\pi x_i}{0.4}\right) \sin\left(\frac{2\pi y_j}{0.08}\right)(1 + \Delta t).$$

Error convergence

Recall that our approximation to (1.1) became approximate because we neglected the higher-order terms in (2.28), (2.33), (2.39), (2.48), (2.54) and (2.63). These higher-order terms directly correlate to the magnitude of error we expect to get between an exact solution and our approximation of u. Recall that the higher-order terms were $\mathcal{O}(H_x^2 + H_y^2 + \Delta t^2)$ at the interior nodes, and $\mathcal{O}(H_x^2 + H_y^3 + \Delta t^2)$, $\mathcal{O}(H_x^3 + H_y^2 + \Delta t^2)$, $\mathcal{O}(H_x^3 + H_x^2 \Delta t^2)$, $\mathcal{O}(H_x^3 + H_y^2 + \Delta t^2)$ and $\mathcal{O}(H_x^3 + H_x^2 \Delta t^2)$ along the Neumann boundaries. It makes sense

that the maximum error recorded would correlate to the higher-order terms resulting in the largest error. Hence, to find out what these are we combined all higher-order terms to get

$$\mathcal{O}\bigg(H_x^2 + H_y^2 + \Delta t^2 + H_x^2 \Delta t^2\bigg). \tag{2.74}$$

Recall that for our PDE in (1.1), our approximation of u in (2.65) is stable if the condition in (2.34) holds. Therefore, by multiplying the last term in (2.74) by H_x^2/H_x^2, we can use the stability condition in (2.34) to get

$$\mathcal{O}\bigg(H_x^2 + H_y^2 + \Delta t^2\bigg). \tag{2.75}$$

Looking at the combined higher-order terms in (2.75), we would expect the maximum absolute error to decrease by a factor of four as H_x, H_y and Δt half. Table (2.3) displays the maximum absolute error across all nodes in our mesh at $T = 1$ between an exact solution in (2.66) and our approximations of u for increasing values of N and M, which subsequently decreases H_x and H_y, respectively, and decreasing Δt. We can deduce as both N and M double, and Δt halves the error convergence approaches four which is what we expected. Therefore, we can conclude that the mathematics set-out in this section and the code are correct and working.

N	M	Δt	Maximum |Error|	Error Percentage†	Error Convergence	Computation Time†† (s)
30	3	1800^{-1}	0.058812755	3.094945955	N/A	0.031026
60	6	3600^{-1}	0.011568330	0.624289521	5.083945133	0.140547
120	12	7200^{-1}	0.002655675	0.144007174	4.356078963	1.044969
240	24	14400^{-1}	0.000648499	0.035203959	4.095110401	8.344721
480	48	28800^{-1}	0.000161100	0.008747699	4.025443824	68.401874
960	96	57600^{-1}	0.000040204	0.002183191	4.007063974	619.233829

† Error Percentage = [maximum error at node (i, j)]/[approximation at node (i, j)] $\times$ 100.

†† Computed using an i5 $-$ 4590 @ 3.30GHz, 16GB RAM @ 1333 MHz and Intel HD Graphics 4600.

Table 2.3: The error convergence for our explicit FDM approximation of u from the acoustic 2D wave equation, at $T = 1$, with the boundary defined on $\partial\Omega_{2\mathrm{D}_2}$.

The Lax equivalent theorem states that if a finite difference scheme is both stable and consistent, then it is convergent [37]. We already know that our finite difference scheme in

(2.65) is stable, see [38]. For our finite difference scheme to be consistent, the maximum absolute error in Table (2.3) must approach zero as $H_x = 0.2/N$, $H_y = 0.02/M$ and Δt approach zero. Upon inspection of Table (2.3), we can see that this is the case. Therefore, we can conclude that our finite difference scheme is convergent.

2.3 Standard Kalman filter outline

The aim of this thesis is to solve an inverse problem. That is, given time-series data at specific locations on the surface of our domain, we want to be able to find the location of the disturbance used to generate this data. The KF has the ability to solve inverse problems when it is coupled with a likelihood estimate, see [39]. Therefore, we will be using the KF alongside a likelihood estimate in this thesis to solve our inverse problems.

The KF was first proposed by Rudolf Kalman in 1960, see [40], and was first applied to aid the navigation of Project Apollo in the 1960s [41]. Since the first application of the KF, it has proved to be an extremely valuable tool, which earned the author the National Academy of Engineering Draper prize in 2008. To this day, the KF is used in many applications including but not limited to navigation, GPS, tracking objects and finance [42], [43] and [44].

The KF is a set of equations which estimate a latent state, which we model after our explicit FDM approximations of u, over recursive steps using noisy observations from the same recursive steps [42]. This is achieved by minimising the mean squared error between the estimate and expected solution – that is we attempt to minimise the trace of the covariance matrix for this error.

Considering a discrete-time case in this thesis, we define the state to be $\tilde{\mathbf{x}}^n$ and the observations as $\mathbf{y}^n$. The KF formulation depends on a state space representation of a dynamic system, which we describe first. The state at the discrete-time step $(n+1)$ is formed from the previous state at n as follows

$$\tilde{\mathbf{x}}^{n+1} := \mathbf{G}\tilde{\mathbf{x}}^n + \mathbf{g}^n + \mathbf{w}^n \tag{2.76}$$

where $\mathbf{G}$ and $\mathbf{g}^n$ are a matrix and a vector, respectively, of compatible dimensions. Additionally, $\mathbf{w}^n$ represents random Gaussian noise with zero mean, that is $\mathbf{w}^n \sim N(\mu, \mathbf{S}^2) = N(0, \mathbf{Q})$, where $\mathbf{Q}$ is the covariance matrix of $\mathbf{w}^n$ and $\mathbf{S}$ is any positive definite square root of $\mathbf{Q}$.

At the discrete-time step n, we construct the noisy observations as follows

$$\mathbf{y}^n := \mathbf{C}\tilde{\mathbf{x}}^n + \mathbf{z}^n \tag{2.77}$$

where $\mathbf{C}$ extracts data from $\tilde{\mathbf{x}}^n$ at specific locations, hereafter known as our sensor locations. Additionally, $\mathbf{z}^n$ represents random Gaussian noise with zero mean, that is $\mathbf{z}^n \sim N(0, \mathbf{R})$, where $\mathbf{R}$ is the covariance matrix of $\mathbf{z}^n$.

The purpose of the KF is to compute an estimate for $\tilde{\mathbf{x}}^{n+1}$ in (2.76), given the estimate and observations at the previous discrete-time step. Since this procedure is recursive, we can imply this for $n \in \{1, 2, ..., L-1\}$.

Let us denote $\hat{\mathbf{x}}^{n+1|n}$ as the estimate of $\tilde{\mathbf{x}}^{n+1}$, and note that $\hat{\mathbf{x}}^{1|0}$ is required to be known. In our numerical experiments, we take this to be zero, although its choice may be guided by the application under consideration. The covariance matrix of the estimate is denoted by $\mathbf{P}^{n+1|n}$, where $\mathbf{P}^{1|0}$ is taken to be the identity matrix. As our study in chapters 3-7 result in state estimate values which are much smaller than 1, our choice of $\mathbf{P}^{1|0}$ denotes a large dispersion in our estimate of $\hat{\mathbf{x}}^{1|0}$. Despite this, if these initial conditions are bad it has little effect on the success of the KF, with the only impact being slower convergence [42].

Having already defined both (2.76) and (2.77), we now outline seven KF equations which estimate the mean, and covariance of the state variable in (2.76) using (2.77). The KF has two steps: the *prediction step* and the *update step* as outlined below

Prediction step:

Predicted mean
$$\hat{\mathbf{x}}^{n+1|n} := \mathbf{G}\hat{\mathbf{x}}^{n|n} + \mathbf{g}^n, \tag{2.78}$$

Covariance of predicted mean
$$\mathbf{P}^{n+1|n} := \mathbf{G}\mathbf{P}^{n|n}\mathbf{G}^{\mathrm{T}} + \mathbf{Q}. \tag{2.79}$$

Update step:

Innovations
$$\mathbf{v}^n := \mathbf{y}^n - \mathbf{C}\hat{\mathbf{x}}^{n|n-1}, \tag{2.80}$$

Covariance of innovations
$$\mathbf{\Sigma}^n := \mathbf{C}\mathbf{P}^{n|n-1}\mathbf{C}^{\mathrm{T}} + \mathbf{R}, \tag{2.81}$$

Kalman gain
$$\mathbf{K}^n := \mathbf{P}^{n|n-1}\mathbf{C}^{\mathrm{T}}\left(\mathbf{\Sigma}^n\right)^{-1}, \tag{2.82}$$

Updated mean
$$\hat{\mathbf{x}}^{n|n} := \hat{\mathbf{x}}^{n|n-1} + \mathbf{K}^n\mathbf{v}^n, \tag{2.83}$$

Covariance of updated mean
$$\mathbf{P}^{n|n} := \left(\mathbf{I} - \mathbf{K}^n\mathbf{C}\right)\mathbf{P}^{n|n-1}. \tag{2.84}$$

Both (2.78) and (2.79) estimate the mean and covariance matrix of $\tilde{\mathbf{x}}^{n+1}$ in (2.76) using observations up to and including the discrete-time step n. In (2.80), we calculate the error between the observations and the models prediction of (2.76), and the corresponding covariance matrix is constructed and given in (2.81). The updated mean in (2.83) is formulated, where the KF estimation of (2.76) is improved using the Kalman gain in (2.82). As previously discussed, we compute the Kalman gain by minimising the trace of the covariance matrix in (2.84), which results in an improved estimate of (2.76).

Before we conclude this section, we can reduce the seven KF equations in (2.78)-(2.84) to only six. The reason for this is because (2.79) is independent of any observations, and so we combine it with (2.84), to give us the following equations which outline the standard KF

$$\mathbf{v}^n := \mathbf{y}^n - \mathbf{C}\hat{\mathbf{x}}^{n|n-1}, \tag{2.85}$$

$$\mathbf{\Sigma}^n := \mathbf{C}\mathbf{P}^{n|n-1}\mathbf{C}^{\mathrm{T}} + \mathbf{R}, \tag{2.86}$$

$$\mathbf{K}^n := \mathbf{P}^{n|n-1}\mathbf{C}^{\mathrm{T}}\big(\mathbf{\Sigma}^n\big)^{-1}, \tag{2.87}$$

$$\hat{\mathbf{x}}^{n|n} := \hat{\mathbf{x}}^{n|n-1} + \mathbf{K}^n\mathbf{v}^n, \tag{2.88}$$

$$\hat{\mathbf{x}}^{n+1|n} := \mathbf{G}\hat{\mathbf{x}}^{n|n} + \mathbf{g}^n, \tag{2.89}$$

$$\mathbf{P}^{n+1|n} := \mathbf{G}\mathbf{P}^{n|n-1}\mathbf{G}^{\mathrm{T}} + \mathbf{Q} - \mathbf{G}\mathbf{P}^{n|n-1}\mathbf{C}^{\mathrm{T}}\big(\mathbf{\Sigma}^n\big)^{-1}\mathbf{C}\mathbf{P}^{n|n-1}\mathbf{G}^{\mathrm{T}} \tag{2.90}$$

for $n \in \{1, 2, ..., L-1\}$.

2.4 Our application of the Kalman filter

In this thesis, we use both the KF and likelihood estimation to predict a latent state, using sensor traces obtained from our explicit FDM approximations of u, and predict an unknown parameter in our forcing function. Since the structure of our FDM approximation equations in (2.16) and (2.65) are different from the latent state in (2.76), we have to modify the structure of both (2.76) and (2.77), and subsequently (2.85)-(2.90). Doing this means we can apply the KF to estimate our latent state using our FDM approximations in (2.16) or (2.65) for our 1D and 2D model problems, respectively.

In this section, we look at the modifications required to the KF, derive a likelihood function, and outline the singular value decomposition used to reduce matrix dimensions within the KF.

2.4.1 Alterations made to the standard Kalman filter

To modify the KF equations, we use block matrices which enables the combination of both $\mathbf{V}^n$ and $\mathbf{V}^{n-1}$ in (2.16) or (2.65). Having constructed the required block matrices, the following forms our discrete-time recursive equations, to be used in conjunction with the KF, based on our explicit FDM approximation equations outlined in (2.16) and (2.65)

$$\underline{\tilde{\mathbf{x}}}^{n+1} := \mathbf{A}\underline{\tilde{\mathbf{x}}}^n + \Delta t^2 \tilde{\boldsymbol{\mathcal{F}}}^n + \tilde{\mathbf{w}}^n, \tag{2.91}$$

$$\tilde{\mathbf{y}}^n := \tilde{\mathbf{C}}\underline{\tilde{\mathbf{x}}}^n + \tilde{\mathbf{z}}^n \tag{2.92}$$

$$\text{where } \mathbf{A} := \begin{pmatrix} \mathbf{B} & -\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{pmatrix}, \underline{\tilde{\mathbf{x}}}^n := \begin{pmatrix} \mathbf{V}^n \\ \mathbf{V}^{n-1} \end{pmatrix}, \tilde{\boldsymbol{\mathcal{F}}}^n := \begin{pmatrix} \boldsymbol{\mathcal{F}}^n \\ \mathbf{0} \end{pmatrix}, \tilde{\mathbf{w}}^n := \begin{pmatrix} \mathbf{w}^n \\ \mathbf{0} \end{pmatrix},$$

$$\tilde{\mathbf{y}}^n := \begin{pmatrix} \mathbf{y}^n \\ \mathbf{y}^{n-1} \end{pmatrix}, \tilde{\mathbf{C}} := \begin{pmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{C} \end{pmatrix} \text{ and } \tilde{\mathbf{z}}^n := \begin{pmatrix} \mathbf{z}^n \\ \mathbf{z}^{n-1} \end{pmatrix}.$$

In this modified formulation, matrices $\mathbf{B}$, $\mathbf{V}^n$, $\mathbf{V}^{n-1}$ and $\mathbf{C}$, and vectors $\mathbf{y}^n$ and $\mathbf{y}^{n-1}$ are known. A new block matrix, $\tilde{\boldsymbol{\mathcal{F}}}^n$, contains the forcing function for a given model problem, at the $n-$ th discrete-time step across every node in our mesh, denoted by $\boldsymbol{\mathcal{F}}^n$.

Now we have outlined the changes made to (2.76) and (2.77) in (2.91) and (2.92), respectively, we outline the KF for $n \in \{1, 2, ..., L-1\}$ below:

$$\underline{\mathbf{v}}^n := \tilde{\mathbf{y}}^n - \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{n|n-1}, \tag{2.93}$$

$$\underline{\boldsymbol{\Sigma}}^n := \tilde{\mathbf{C}}\underline{\mathbf{P}}^{n|n-1}\tilde{\mathbf{C}}^{\mathrm{T}} + \tilde{\mathbf{R}}, \tag{2.94}$$

$$\underline{\mathbf{K}}^n := \underline{\mathbf{P}}^{n|n-1}\tilde{\mathbf{C}}^{\mathrm{T}}\left(\underline{\boldsymbol{\Sigma}}^n\right)^{-1}, \tag{2.95}$$

$$\underline{\hat{\mathbf{x}}}^{n|n} := \underline{\hat{\mathbf{x}}}^{n|n-1} + \underline{\mathbf{K}}^n\underline{\mathbf{v}}^n, \tag{2.96}$$

$$\underline{\hat{\mathbf{x}}}^{n+1|n} := \mathbf{A}\underline{\hat{\mathbf{x}}}^{n|n} + \Delta t^2 \tilde{\boldsymbol{\mathcal{F}}}^n, \tag{2.97}$$

$$\underline{\mathbf{P}}^{n+1|n} := \mathbf{A}\underline{\mathbf{P}}^{n|n-1}\mathbf{A}^{\mathrm{T}} + \tilde{\mathbf{Q}} - \mathbf{A}\underline{\mathbf{P}}^{n|n-1}\tilde{\mathbf{C}}^{\mathrm{T}}\left(\underline{\boldsymbol{\Sigma}}^n\right)^{-1}\tilde{\mathbf{C}}\underline{\mathbf{P}}^{n|n-1}\mathbf{A}^{\mathrm{T}} \tag{2.98}$$

where for symmetric, positive definite covariance matrices $\mathbf{Q}$ and $\mathbf{R}$

$$\tilde{\mathbf{Q}} := \begin{pmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} \end{pmatrix} \text{ and } \tilde{\mathbf{R}} := \begin{pmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix}.$$

2.4.2 Derivation of the likelihood function

Now that we have outlined the modified KF equations, we want to estimate the location of a disturbance in our forcing function, which we denote by x_0 or $\underline{\mathbf{x}}_0$ in our 1D and 2D model problems respectively. For more information regarding the forcing function used in our model problems, see our approach outline in the introduction.

Let us first denote $\mathcal{G}^{i-1} := \{\tilde{\mathbf{y}}^{i-1}, \tilde{\mathbf{y}}^{i-2}, ..., \tilde{\mathbf{y}}^1, \tilde{\mathbf{y}}^0\}$ as the family of observations up to and including the discrete-time step $(i-1)$ where $i \in \{1, 2, ..., L-1\}$. We want to construct a likelihood function which can determine how likely a guess for our unknown parameter, denoted by x_0 or $\underline{\mathbf{x}}_0$ in our 1D and 2D model problems respectively, is at being the actual value used to form the observations denoted by $\tilde{\mathbf{y}}^i$ for all $i \in \{1, 2, ..., L-1\}$. To achieve this, we need to evaluate the probability density function for $\mathbb{P}(\tilde{\mathbf{y}}^i|\mathcal{G}^{i-1})$, denoted by $h(\tilde{\mathbf{y}}^i; x_0)$ or $h(\tilde{\mathbf{y}}^i; \underline{\mathbf{x}}_0)$ for our 1D and 2D model problems respectively. In the following derivation, we use the notation associated with our 2D model problems. Assuming that $\mathbb{P}(\tilde{\mathbf{y}}^i|\mathcal{G}^{i-1}) \sim N(\mu, \mathbf{S}^2)$, we can deduce the multivariate Gaussian density function for a particular value of i to be

$$h(\tilde{\mathbf{y}}^i; \underline{\mathbf{x}}_0) := \frac{1}{\sqrt{2\pi \det\left(\mathbf{S}^2\right)}} \exp\left(-\frac{1}{2}\left(\tilde{\mathbf{y}}^i - \mu\right)^{\mathrm{T}} \left(\mathbf{S}^2\right)^{-1} \left(\tilde{\mathbf{y}}^i - \mu\right)\right) \tag{2.99}$$

where $\det(\cdot)$ denotes the determinant.

Before we proceed with deriving the likelihood function, we must first find μ and $\mathbf{S}^2$ in (2.99). Starting with the mean, we have

$$\mu = \mathbb{E}(\tilde{\mathbf{y}}^i|\mathcal{G}^{i-1}). \tag{2.100}$$

Substituting (2.92) into (2.100), noting that the noise is independent of the observations and matrix $\tilde{\mathbf{C}}$ is constant we get

$$\begin{aligned} \mathbb{E}(\tilde{\mathbf{y}}^i|\mathcal{G}^{i-1}) &= \mathbb{E}(\tilde{\mathbf{C}}\underline{\tilde{\mathbf{x}}}^{i|i-1} + \tilde{\mathbf{z}}^i), \\ &= \tilde{\mathbf{C}}\mathbb{E}(\underline{\tilde{\mathbf{x}}}^{i|i-1}) + \mathbb{E}(\tilde{\mathbf{z}}^i), \\ &= \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{i|i-1}. \end{aligned} \tag{2.101}$$

Therefore, $\mu = \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{i|i-1}$ for $i \in \{1, 2, ..., L-1\}$. We are now interested in deriving the variance, denoted by $\mathbf{S}^2$. We know that the variance of a vector, in our case $\tilde{\mathbf{y}}^i$ given $\mathcal{G}^{i-1}$, can be computed using

$$\mathbf{S}^2 = \mathrm{Var}\big(\tilde{\mathbf{y}}^i|\mathcal{G}^{i-1}\big),$$

$$= \mathbb{E}\Big[\big(\tilde{\mathbf{y}}^i - \mathbb{E}[\tilde{\mathbf{y}}^i]\big)\big(\tilde{\mathbf{y}}^i - \mathbb{E}[\tilde{\mathbf{y}}^i]\big)^{\mathrm{T}}\Big|\mathcal{G}^{i-1}\Big]. \tag{2.102}$$

Substituting (2.101) into (2.102), noting that $\tilde{\mathbf{y}}^i$ for all $i \in \{1, 2, ..., L-1\}$ are independent, and making use of (2.93) to simplify further, we get

$$\begin{aligned}\mathbf{S}^2 &= \mathbb{E}\Big[\big(\tilde{\mathbf{y}}^i - \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{i|i-1}\big)\big(\tilde{\mathbf{y}}^i - \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{i|i-1}\big)^{\mathrm{T}}\Big],\\ &= \mathbb{E}\Big[\big(\underline{\mathbf{v}}^i\big)\big(\underline{\mathbf{v}}^i\big)^{\mathrm{T}}\Big],\\ &= \mathrm{Var}\big(\underline{\mathbf{v}}^i\big).\end{aligned} \tag{2.103}$$

Recall that (2.94) represents the covariance matrix for (2.93), and so, using this in (2.103) we can conclude that

$$\mathbf{S}^2 = \underline{\boldsymbol{\Sigma}}^i \tag{2.104}$$

for $i \in \{1, 2, ..., L-1\}$.

Substituting both (2.101) and (2.104) into (2.99), and recalling that our observations are independent, we can compute the likelihood function by taking the product of all multivariate Gaussian density functions to get

$$\begin{aligned}L(\underline{\mathbf{x}}_0) &= \prod_{i=1}^{L-1} h(\tilde{\mathbf{y}}^i; \underline{\mathbf{x}}_0),\\ &= \prod_{i=1}^{L-1} \frac{1}{\sqrt{2\pi \det(\underline{\boldsymbol{\Sigma}}^i)}} \exp\bigg(-\frac{1}{2}\big(\tilde{\mathbf{y}}^i - \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{i|i-1}\big)^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^i\big)^{-1}\big(\tilde{\mathbf{y}}^i - \tilde{\mathbf{C}}\underline{\hat{\mathbf{x}}}^{i|i-1}\big)\bigg),\\ &= \prod_{i=1}^{L-1} \frac{1}{\sqrt{2\pi \det(\underline{\boldsymbol{\Sigma}}^i)}} \exp\bigg(-\frac{1}{2}\big(\underline{\mathbf{v}}^i\big)^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^i\big)^{-1}\underline{\mathbf{v}}^i\bigg).\end{aligned} \tag{2.105}$$

Since (2.105) contains an exponential function, we can simplify further by taking the natural logarithm. The natural logarithm is an increasing function, meaning the value of $\underline{\mathbf{x}}_0$ which maximises (2.105), will also maximise the natural logarithm of (2.105). Therefore, making use of logarithmic and exponential identities, we can further simplify to get

$$\begin{aligned}\log\big(L(\underline{\mathbf{x}}_0)\big) &= \sum_{i=1}^{L-1} \log\left(\frac{1}{\sqrt{2\pi \det(\underline{\boldsymbol{\Sigma}}^i)}} \exp\bigg(-\frac{1}{2}\big(\underline{\mathbf{v}}^i\big)^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^i\big)^{-1}\underline{\mathbf{v}}^i\bigg)\right),\\ &= -\frac{1}{2}\sum_{i=1}^{L-1}\log\big(2\pi\big) - \frac{1}{2}\sum_{i=1}^{L-1}\bigg(\log\big(\det(\underline{\boldsymbol{\Sigma}}^i)\big) + \big(\underline{\mathbf{v}}^i\big)^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^i\big)^{-1}\underline{\mathbf{v}}^i\bigg).\end{aligned} \tag{2.106}$$

Since the first term in (2.106) is parameterless, we neglect it. Therefore, to be able to determine the value of $\underline{\mathbf{x}}_0$ which maximises $\log\big(L(\underline{\mathbf{x}}_0)\big)$, we need to find $\underline{\mathbf{x}}_0$ which minimises the following function:

$$-\log\big(L(\underline{\mathbf{x}}_0)\big) = \frac{1}{2}\sum_{i=1}^{L-1}\left(\log\big(\det(\underline{\mathbf{\Sigma}}^i)\big) + \big(\underline{\mathbf{v}}^i\big)^{\mathrm{T}}\big(\underline{\mathbf{\Sigma}}^i\big)^{-1}\underline{\mathbf{v}}^i\right). \tag{2.107}$$

For our 1D model problems, we use a minimisation algorithm within a MATLAB function called *fminsearch* which uses the Nelder-Mead algorithm. The Nelder-Mead algorithm was first published in 1965, see [45]. Its purpose is to find the minimum of a function in $\mathbb{R}^n$, see [46], which in our case is (2.107). We are interested in obtaining the smallest negative logarithmic likelihood function in (2.107) for an unknown parameter in our forcing function. The initial guess chosen for the unknown parameter in the minimisation algorithm is essential, otherwise, we run the risk of converging to a non-stationary point [47]. As a result of this, we make an array of initial guesses for x_0, requiring the KF to be run in each case, increasing the possibility that we deduce the value of x_0 used to generate our sensor traces. This approach proves to be a problem in our 2D model problems, and when we consider higher frequencies within our forcing function. Therefore, we devise an alternative strategy in chapter 3 under alternative approaches.

In the next section, we outline the SVD and further modify the KF equations, resulting in substantial matrix dimension reductions within the KF, enabling us to solve problems with disturbances with higher frequencies in the forcing function. We investigate this for both our 1D and 2D model problems in chapters 4 and 6, respectively.

2.4.3 Singular value decomposition outline

In this thesis, we aim to solve inverse problems using time-series data along the surface of our domains. This involves using the KF whose matrix dimensions are linked to the mesh density used to generate our approximations of u in our PDE, which at specific nodes in our mesh represent the time-series data along the surface of our domains. The density of this mesh, and subsequently the size of the matrices in the KF, increase significantly when we consider higher-dimensional model problems, and disturbances with higher frequencies in chapters 4, 5 and 6. Therefore we use the SVD to decompose our matrix containing the approximations of u for our PDE, from which we take the principle components which correspond to the largest singular values. By doing this, we can modify the KF to reduce

the system RAM and run-time requirements, while retaining a significant amount of the information from the original data.

The SVD is a technique which decomposes a real rectangular matrix, denoted by $\hat{\mathbf{M}} \in \mathbb{R}^{r\times c}$ into

$$\hat{\mathbf{M}} = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^{\mathrm{T}} \tag{2.108}$$

where $\hat{\mathbf{U}} \in \mathbb{R}^{r\times r}$ and $\hat{\mathbf{V}} \in \mathbb{R}^{c\times c}$ are real orthogonal matrices and $\hat{\mathbf{\Sigma}}$ is a matrix containing the singular values of $\hat{\mathbf{M}}$ along its main diagonal, see [48] and [49]. The singular values of $\hat{\mathbf{M}}$ are positive square roots of the eigenvalues of $\hat{\mathbf{M}}\hat{\mathbf{M}}^{\mathrm{T}}$. Since $\hat{\mathbf{V}}$ is an orthogonal matrix, we know that $\hat{\mathbf{V}}^{\mathrm{T}} \equiv \hat{\mathbf{V}}^{-1}$. Matrices $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ contain the left and right singular vectors for $\hat{\mathbf{M}}$, respectively, which correspond to the eigenvectors for $\hat{\mathbf{M}}\hat{\mathbf{M}}^{\mathrm{T}}$ and $\hat{\mathbf{M}}^{\mathrm{T}}\hat{\mathbf{M}}$ respectively.

The SVD can be used to reduce the dimension of a matrix, in our case $\hat{\mathbf{M}}$, see [50]. Using the SVD is essential for us when the dimension of matrices in the KF become too large due to higher-dimensional models or higher frequencies within our forcing function. From the SVD formed, we are able to extract the principal components which correspond to the largest singular values in matrix $\hat{\mathbf{M}}$. Using these principal components, we are able to dynamically modify the KF equations resulting in matrix dimension reductions.

Before we modify our explicit FDM approximation of u and KF equations, we must first extract the principal components from the SVD. In our case, matrix $\hat{\mathbf{M}}$ contains the FDM approximations of u at every node in our mesh across a range of consecutive discrete-time steps, whose choice is illustrated in Figure (3.14), and so, matrix $\hat{\mathbf{M}}$ is given by

$$\hat{\mathbf{M}} = \begin{pmatrix} \mathbf{V}^{(2-T_s)\frac{L}{T}} & \mathbf{V}^{(2-T_s)\frac{L}{T}+1} & \dots & \mathbf{V}^{\frac{2L}{T}} \end{pmatrix} \tag{2.109}$$

where T_s represents the duration of time, in seconds, we use the FDM approximations of u to form the SVD, L is the total number of discrete-time steps, and T is the duration of our simulation.

Using (2.109) in (2.108), we get the SVD of our matrix $\hat{\mathbf{M}}$, where the singular values in $\hat{\mathbf{\Sigma}}$ are in descending order. We then take the first d columns of $\hat{\mathbf{U}}$, otherwise known as the principal components. Since the singular values in $\hat{\mathbf{\Sigma}}$ are in descending order, the data in the first d columns of $\hat{\mathbf{U}}$ accounts for the largest amount of variance in matrix $\hat{\mathbf{M}}$. We use the MATLAB built-in function, called *svds*, to form the SVD of $\hat{\mathbf{M}}$ for our model problems.

Now we have outlined how to construct an SVD for our matrix $\hat{\mathbf{M}}$, we want to modify our explicit FDM approximation equations and the KF equations. Let us first denote the first d columns in $\hat{\mathbf{U}}$ as

$$\boldsymbol{\Phi} = \begin{pmatrix} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_d \end{pmatrix} \tag{2.110}$$

where $\boldsymbol{\phi}_i$ is the i-th column in $\hat{\mathbf{U}}$ for $i \in \{1, 2, \cdots, d\}$, where $d \leq r$. Since $\hat{\mathbf{U}}$ is an orthogonal matrix, the columns of $\hat{\mathbf{U}}$ form an orthonormal basis of $\mathbb{R}^r$ [51]. The matrix $\boldsymbol{\Phi}$ is a subset of the columns in $\hat{\mathbf{U}}$. Therefore, the columns of $\boldsymbol{\Phi}$ form an orthonormal basis of $\mathbb{R}^d$. As a result of this, we know that $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} = \mathbf{I}$ since the inner product, denoted by $\langle \cdot, \cdot \rangle$, is given by

$$\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise} \end{cases}$$

as illustrated by [52]. To reduce the dimensions of the matrices in the KF, we use $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} = \mathbf{I}$ and assume that $\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}} \approx \mathbf{I}$ when σ_{d+1} is sufficiently small, where $\sigma_{d+1} \geq 0$ denotes the $(d+1)^{\text{th}}$ largest singular value of matrix $\hat{\mathbf{M}}$. As shown in [53], this results in an error term of $||\hat{\mathbf{M}} - \boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\hat{\mathbf{M}}|| = \sigma_{d+1}$. Using this assumption, we can approximate $\hat{\mathbf{M}}$ to get

$$\begin{aligned} \hat{\mathbf{M}} &\approx \boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\hat{\mathbf{M}}, \\ \hat{\mathbf{M}}^j &\approx \boldsymbol{\Phi}\boldsymbol{\beta}^j, \\ \mathbf{V}^j &\approx \boldsymbol{\Phi}\boldsymbol{\beta}^j \end{aligned} \tag{2.111}$$

where $\hat{\mathbf{M}}^j$ and $\boldsymbol{\beta}^j$ are the j^{th} columns of $\hat{\mathbf{M}}$ and $\boldsymbol{\Phi}^{\mathrm{T}}\hat{\mathbf{M}}$ respectively for $j \in \{1, 2, \cdots, c\}$.

With this knowledge, we modify the explicit FDM model equation outlined in (2.16) or (2.65) for our 1D and 2D model problems, respectively. We use (2.111) to deduce $\mathbf{V}^{n+1} \approx \boldsymbol{\Phi}\boldsymbol{\beta}^{n+1}$, $\mathbf{V}^n \approx \boldsymbol{\Phi}\boldsymbol{\beta}^n$ and $\mathbf{V}^{n-1} \approx \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1}$. We then substitute these into (2.16) or (2.65) to get

$$\boldsymbol{\Phi}\boldsymbol{\beta}^{n+1} \approx \mathbf{B}\boldsymbol{\Phi}\boldsymbol{\beta}^n - \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1} + \Delta t^2 \boldsymbol{\mathcal{F}}^n. \tag{2.112}$$

Recall that $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} = \mathbf{I}$. Therefore, by multiplying (2.112) on the left by $\boldsymbol{\Phi}^{\mathrm{T}}$ we get

$$\begin{aligned} \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\beta}^{n+1} &\approx \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{B}\boldsymbol{\Phi}\boldsymbol{\beta}^n - \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\beta}^{n-1} + \Delta t^2 \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\mathcal{F}}^n, \\ \boldsymbol{\beta}^{n+1} &\approx \mathbf{B}_{\text{New}}\boldsymbol{\beta}^n - \boldsymbol{\beta}^{n-1} + \Delta t^2 \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\mathcal{F}}^n \end{aligned} \tag{2.113}$$

where $\mathbf{B}_{\text{New}} := \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{B}\boldsymbol{\Phi}$. Now we have modified our explicit FDM approximation equations, we move on to modifying the KF equations.

Recall that the KF consists of six equations outlined in (2.93)-(2.98). Starting with the *innovations* in (2.93), we use (2.111) to deduce $\mathbf{V}^{n|n-1} \approx \boldsymbol{\Phi}\boldsymbol{\beta}^{n|n-1}$ and $\mathbf{V}^{n-1|n-1} \approx \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1|n-1}$. We substitute these into (2.93) to get

$$\begin{aligned}\underline{\mathbf{v}}^n &\approx \tilde{\mathbf{y}}^n - \tilde{\mathbf{C}}\begin{pmatrix}\boldsymbol{\Phi}\boldsymbol{\beta}^{n|n-1}\\ \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1|n-1}\end{pmatrix},\\ &= \tilde{\mathbf{y}}^n - \begin{pmatrix}\mathbf{C} & \mathbf{0}\\ \mathbf{0} & \mathbf{C}\end{pmatrix}\begin{pmatrix}\boldsymbol{\Phi} & \mathbf{0}\\ \mathbf{0} & \boldsymbol{\Phi}\end{pmatrix}\begin{pmatrix}\boldsymbol{\beta}^{n|n-1}\\ \boldsymbol{\beta}^{n-1|n-1}\end{pmatrix},\\ &= \tilde{\mathbf{y}}^n - \tilde{\mathbf{C}}_{\text{New}}\hat{\boldsymbol{\beta}}^{n|n-1} \qquad (2.114)\end{aligned}$$

where $\tilde{\mathbf{C}}_{\text{New}} := \tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}} = \begin{pmatrix}\mathbf{C} & \mathbf{0}\\ \mathbf{0} & \mathbf{C}\end{pmatrix}\begin{pmatrix}\boldsymbol{\Phi} & \mathbf{0}\\ \mathbf{0} & \boldsymbol{\Phi}\end{pmatrix}$ and $\hat{\boldsymbol{\beta}}^{n|n-1} := \begin{pmatrix}\boldsymbol{\beta}^{n|n-1}\\ \boldsymbol{\beta}^{n-1|n-1}\end{pmatrix}$. We continue with the *covariance of the innovations* in (2.94). From the assumption that $\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}} \approx \mathbf{I}$, we arrive at the assumption that $\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}} \approx \mathbf{I}$. Therefore, by substituting this assumption into (2.94), we get

$$\begin{aligned}\underline{\boldsymbol{\Sigma}}^n &\approx \tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{C}}^{\mathrm{T}} + \tilde{\mathbf{R}},\\ &= \tilde{\mathbf{C}}_{\text{New}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\tilde{\mathbf{C}}_{\text{New}}^{\mathrm{T}} + \tilde{\mathbf{R}} \qquad (2.115)\end{aligned}$$

since $\tilde{\mathbf{C}}_{\text{New}}^{\mathrm{T}} := \big(\tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}}\big)^{\mathrm{T}} = \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{C}}^{\mathrm{T}}$. Let us denote $\underline{\tilde{\mathbf{P}}}^{n|n-1} = \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}$, and so, by substituting this into (2.115) we get

$$\underline{\boldsymbol{\Sigma}}^n \approx \tilde{\mathbf{C}}_{\text{New}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\text{New}}^{\mathrm{T}} + \tilde{\mathbf{R}}. \qquad (2.116)$$

Moving on to the *Kalman Gain* in (2.95), we multiply through on the left side by $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}$ to get

$$\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{K}}^n = \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\mathbf{C}}^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^n\big)^{-1}. \qquad (2.117)$$

Recall the assumption that $\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}} \approx \mathbf{I}$. Using this in (2.117) we can further simplify to get

$$\begin{aligned}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{K}}^n &\approx \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{C}}^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^n\big)^{-1},\\ &= \Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}}\Big)^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^n\big)^{-1},\\ &= \underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\text{New}}^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^n\big)^{-1}. \qquad (2.118)\end{aligned}$$

Let us denote $\underline{\tilde{\mathbf{K}}}^n := \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{K}}^n$, and so, after substituting this into (2.118) we get

$$\underline{\tilde{\mathbf{K}}}^n \approx \underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\text{New}}^{\mathrm{T}}\big(\underline{\boldsymbol{\Sigma}}^n\big)^{-1}. \qquad (2.119)$$

Next, we look at the *updated mean* in (2.96). Substituting (2.111) again, at different conditional time intervals, into (2.96) we get

$$\begin{pmatrix} \boldsymbol{\Phi}\boldsymbol{\beta}^{n|n} \\ \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1|n} \end{pmatrix} \approx \begin{pmatrix} \boldsymbol{\Phi}\boldsymbol{\beta}^{n|n-1} \\ \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1|n-1} \end{pmatrix} + \underline{\mathbf{K}}^n \underline{\mathbf{v}}^n,$$

$$\begin{pmatrix} \boldsymbol{\Phi} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Phi} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}^{n|n} \\ \boldsymbol{\beta}^{n-1|n} \end{pmatrix} \approx \begin{pmatrix} \boldsymbol{\Phi} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Phi} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}^{n|n-1} \\ \boldsymbol{\beta}^{n-1|n-1} \end{pmatrix} + \underline{\mathbf{K}}^n \underline{\mathbf{v}}^n,$$

$$\tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n|n} \approx \tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n|n-1} + \underline{\mathbf{K}}^n \underline{\mathbf{v}}^n. \tag{2.120}$$

Recall that $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}} = \mathbf{I}$. Therefore, we multiply (2.120) on the left by $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}$ to get

$$\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n|n} \approx \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n|n-1} + \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{K}}^n \underline{\mathbf{v}}^n,$$

$$\hat{\boldsymbol{\beta}}^{n|n} \approx \hat{\boldsymbol{\beta}}^{n|n-1} + \underline{\tilde{\mathbf{K}}}^n \underline{\mathbf{v}}^n. \tag{2.121}$$

Next, we look at the *predicted mean* in (2.97). Substituting (2.111) again, at different conditional time intervals, into (2.97) results in

$$\begin{pmatrix} \boldsymbol{\Phi}\boldsymbol{\beta}^{n+1|n} \\ \boldsymbol{\Phi}\boldsymbol{\beta}^{n|n} \end{pmatrix} \approx \mathbf{A} \begin{pmatrix} \boldsymbol{\Phi}\boldsymbol{\beta}^{n|n} \\ \boldsymbol{\Phi}\boldsymbol{\beta}^{n-1|n} \end{pmatrix} + \Delta t^2 \tilde{\boldsymbol{\mathcal{F}}}^n,$$

$$\begin{pmatrix} \boldsymbol{\Phi} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Phi} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}^{n+1|n} \\ \boldsymbol{\beta}^{n|n} \end{pmatrix} \approx \mathbf{A} \begin{pmatrix} \boldsymbol{\Phi} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Phi} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}^{n|n} \\ \boldsymbol{\beta}^{n-1|n} \end{pmatrix} + \Delta t^2 \tilde{\boldsymbol{\mathcal{F}}}^n,$$

$$\tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n+1|n} \approx \mathbf{A}\tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n|n} + \Delta t^2 \tilde{\boldsymbol{\mathcal{F}}}^n. \tag{2.122}$$

Recall that $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}} = \mathbf{I}$. Using this, we multiply (2.122) on the left by $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}$ to get

$$\hat{\boldsymbol{\beta}}^{n+1|n} \approx \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\beta}}^{n|n} + \Delta t^2 \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\mathcal{F}}}^n. \tag{2.123}$$

Let us denote $\tilde{\mathbf{A}} := \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}} = \begin{pmatrix} \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{B}\boldsymbol{\Phi} & -\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{pmatrix}$. Substituting this into (2.123), we can simplify further to get

$$\hat{\boldsymbol{\beta}}^{n+1|n} \approx \tilde{\mathbf{A}}\hat{\boldsymbol{\beta}}^{n|n} + \Delta t^2 \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\mathcal{F}}}^n. \tag{2.124}$$

Finally, we look at the covariance matrix (2.98). We start by multiplying (2.98) by $\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}$ and $\tilde{\boldsymbol{\Phi}}$ on the left and right respectively, to get

$$\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n+1|n}\tilde{\boldsymbol{\Phi}} = \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\underline{\mathbf{P}}^{n|n-1}\mathbf{A}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}} + \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{Q}}\tilde{\boldsymbol{\Phi}} - \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\underline{\mathbf{P}}^{n|n-1}\tilde{\mathbf{C}}^{\mathrm{T}}\left(\underline{\boldsymbol{\Sigma}}^n\right)^{-1}\tilde{\mathbf{C}}\underline{\mathbf{P}}^{n|n-1}\mathbf{A}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}}. \tag{2.125}$$

Recall that we assume $\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}} \approx \mathbf{I}$. Substituting this into (2.125) where required results in

$$\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n+1|n}\tilde{\boldsymbol{\Phi}} \approx \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}} + \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{Q}}\tilde{\boldsymbol{\Phi}}$$

$$- \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{C}}^{\mathrm{T}}\Big(\underline{\boldsymbol{\Sigma}}^{n}\Big)^{-1}\tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\tilde{\boldsymbol{\Phi}},$$

$$\Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n+1|n}\tilde{\boldsymbol{\Phi}}\Big) \approx \Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\Big)^{\mathrm{T}} + \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{Q}}\tilde{\boldsymbol{\Phi}}$$
$$- \Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}}\Big)^{\mathrm{T}}\Big(\underline{\boldsymbol{\Sigma}}^{n}\Big)^{-1}\Big(\tilde{\mathbf{C}}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\underline{\mathbf{P}}^{n|n-1}\tilde{\boldsymbol{\Phi}}\Big)\Big(\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{A}\tilde{\boldsymbol{\Phi}}\Big)^{\mathrm{T}},$$

$$\underline{\tilde{\mathbf{P}}}^{n+1|n} \approx \tilde{\mathbf{A}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{A}}^{\mathrm{T}} + \tilde{\mathbf{Q}}_{\mathrm{New}} - \tilde{\mathbf{A}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\mathrm{New}}^{\mathrm{T}}\Big(\underline{\boldsymbol{\Sigma}}^{n}\Big)^{-1}\tilde{\mathbf{C}}_{\mathrm{New}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{A}}^{\mathrm{T}} \tag{2.126}$$

where $\tilde{\mathbf{Q}}_{\mathrm{New}} := \tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\mathbf{Q}}\tilde{\boldsymbol{\Phi}}$. Therefore in summary, as outlined above in (2.114), (2.116), (2.119), (2.121), (2.124) and (2.126), the modified KF using the SVD is defined by

$$\underline{\mathbf{v}}^{n} \approx \tilde{\mathbf{y}}^{n} - \tilde{\mathbf{C}}_{\mathrm{New}}\hat{\boldsymbol{\beta}}^{n|n-1}, \tag{2.127}$$

$$\underline{\boldsymbol{\Sigma}}^{n} \approx \tilde{\mathbf{C}}_{\mathrm{New}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\mathrm{New}}^{\mathrm{T}} + \tilde{\mathbf{R}}, \tag{2.128}$$

$$\underline{\tilde{\mathbf{K}}}^{n} \approx \underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\mathrm{New}}^{\mathrm{T}}\Big(\underline{\boldsymbol{\Sigma}}^{n}\Big)^{-1}, \tag{2.129}$$

$$\hat{\boldsymbol{\beta}}^{n|n} \approx \hat{\boldsymbol{\beta}}^{n|n-1} + \underline{\tilde{\mathbf{K}}}^{n}\underline{\mathbf{v}}^{n}, \tag{2.130}$$

$$\hat{\boldsymbol{\beta}}^{n+1|n} \approx \tilde{\mathbf{A}}\hat{\boldsymbol{\beta}}^{n|n} + \Delta t^{2}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}\tilde{\boldsymbol{\mathcal{F}}}^{n}, \tag{2.131}$$

$$\underline{\tilde{\mathbf{P}}}^{n+1|n} \approx \tilde{\mathbf{A}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{A}}^{\mathrm{T}} + \tilde{\mathbf{Q}}_{\mathrm{New}} - \tilde{\mathbf{A}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{C}}_{\mathrm{New}}^{\mathrm{T}}\Big(\underline{\boldsymbol{\Sigma}}^{n}\Big)^{-1}\tilde{\mathbf{C}}_{\mathrm{New}}\underline{\tilde{\mathbf{P}}}^{n|n-1}\tilde{\mathbf{A}}^{\mathrm{T}}. \tag{2.132}$$

Having now dynamically modified our explicit FDM model equation in (2.113) and the KF equations in (2.127)-(2.132), we can control the size of our matrices by altering the number of principal components taken from the SVD, denoted by d. To illustrate how significant this is, we look at two examples. The vector dimensions of (2.16) and (2.65) are $2(N+1)$ and $2(N+1)(M+1)$, respectively. After modifying our explicit FDM model equation, the vector dimensions in both (2.16) and (2.65) are reduced to $2d$ in (2.113). The matrix dimensions of (2.98) are $2(N+1)^2$ or $2\big((N+1)(M+1)\big)^2$ for our 1D and 2D model problems, respectively. After modifying our KF equations, the matrix dimension of (2.132) is only $2d^2$.

2.5 Summary

Having outlined the methodology used in this thesis, we want to investigate the effectiveness of the theory in an array of different cases. Over the next five chapters, we look at the possibility of localising a disturbance of varying frequency within both 1D and 2D domains, using explicit FDM approximations of u for (1.1) and (1.1), respectively, at sensor locations. In chapter 7 we extend the model problem considered in chapter 3 to

use partial derivatives of u at sensor locations in addition to, and instead of, using only the pressure, denoted by u. In the next chapter, we look at our 1D model problem with a disturbance frequency of $F = 25$Hz.

Chapter 3

1D Model Problem

In this chapter, we outline our 1D model problem using the forcing function outlined in chapter 1 with a disturbance frequency of $F = 25$Hz, we detail the noise added to our explicit FDM approximations of u, and we outline a schematic for our 1D model problem detailing the stages required to locate x_0 in our forcing function.

We present probabilistic results using a minimisation algorithm to find x_0 which minimises (2.107), using our explicit FDM approximations of u from a fine and coarse mesh. We look at alternative approaches, one without using a minimisation algorithm and another utilising an SVD to reduce matrix dimensions in the KF. We compare the run-time for the different approaches considered in this chapter to illustrate the issues that could arise when we consider higher dimensional model problems, and higher disturbance frequencies in chapters 4, 5 and 6 of this thesis. For our 1D model problems with a disturbance frequency of $F = 25$Hz, these alternative approaches are not essential but are important to illustrate the use of data from a coarse mesh yield good results - enabling us to explore 2D model problems and higher frequencies in our forcing function.

3.1 Model problem outline

3.1.1 Forcing function

In this section we recall the forcing function we outlined in chapter 1, see (1.6), and illustrate our choices for the unknown parameters for the 1D model problem with a disturbance frequency of $F = 25$Hz considered in this chapter. Firstly, we discuss the value chosen for the amplitude in (1.6).

The amplitude, denoted by A, was initially taken to be one for simplicity. However, when $A = 1$ in (1.6) the covariance matrix in (2.94) was close to singular, and subsequently, (2.94) was also close to being singular. We discovered that when $A = 10^6$ in (1.6), the covariance matrix in (2.94) was invertible. Therefore, for our model problems we set $A = 10^6$.

The choice for ϵ in (1.6) alters the Gaussian spread of our disturbance. If we take ϵ to be too small and the location of our disturbance is not near a node in our mesh where u is approximated, f in (1.6) would decay before reaching a node meaning the approximations of u would be unreliable. Additionally, if our choice for ϵ is too large, there is a risk that the Gaussian spread of our disturbance would be so large our model could struggle to pinpoint the exact location of the disturbance in our domain. In the results section of this chapter, we take $\epsilon = 5.77 \times 10^{-6}$ in (1.6) by investigating the level of success based on whether our model problem deduces the actual disturbance location, for an array of different values for epsilon.

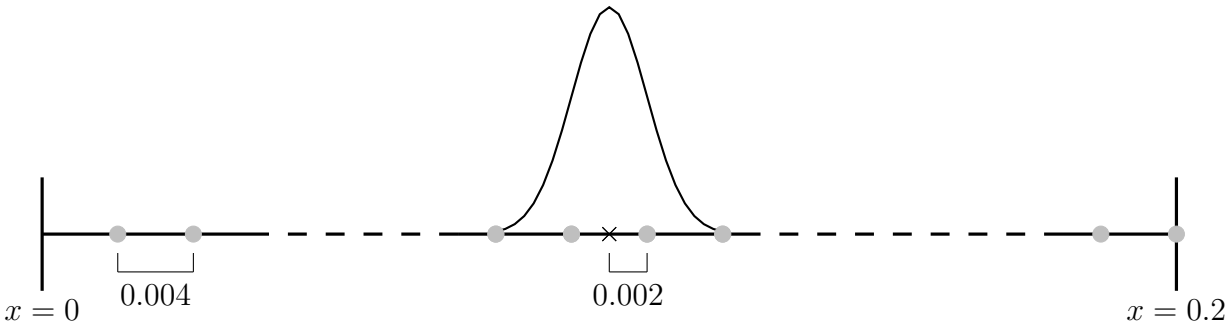


Figure 3.1: A snapshot of the Gaussian spread, given our value of ϵ, around the location of the disturbance denoted by a cross. The disturbance here has a frequency of $F = 25$Hz. The grey dots represent where we evaluate the likelihood function in (2.107) and the position of the nodes in our coarse mesh (50 in total).

Now we have chosen a value for ϵ in our forcing function we look at the Gaussian spread around a disturbance location. Figure (3.1) shows a snapshot of the Gaussian spread caused by our chosen value of ϵ in (1.6) in the worst-case scenario. The worst-case scenario is when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible. The black cross in Figure (3.1) represents the location of our disturbance, denoted by x_0, and the grey dots illustrate the position of the nodes in our coarse mesh (50 in total). At the nearest node

to the disturbance location, in the worst-case scenario, only 50% of the peak is remaining. At the next node along, a distance of 0.006 from the disturbance location representing 3% of the domain size, the signal decays to approximately zero. This is good since we want the disturbance to be localised in our domain, and we have seen that the Gaussian spread decays to zero just 3% of the domain away from the disturbance location.

Having defined our forcing function in (1.6) to be used in (1.1), we want to investigate the convergence of our explicit FDM approximations of u in (2.16). To do this, we need to know both $\mathbf{V}^0$ and $\mathbf{V}^1$ in (2.16). The initial conditions of u for (1.1) are $u(x_i, t_0) := 0$ and $u_t(x_i, t_0) := 0$ for all $i \in \{1, 2, ..., N+1\}$. As a result, we know $\mathbf{V}^0$. However, we do not know $\mathbf{V}^1$. Using both initial conditions of u, and Taylor series, we can approximate $\mathbf{V}^1$ to get

$$u(x_i, t_{n+1}) = u(x_i, t_n) + \Delta t u_t(x_i, t_n) + \mathcal{O}(\Delta t^2)$$

and subsequently, when $n = 0$, we substitute in both initial conditions of u to get

$$u(x_i, t_1) = \mathcal{O}(\Delta t^2).$$

Neglecting higher-order terms, and recalling $\mathbf{V}^n \approx u(x_i, t_n)$ for all $i \in \{1, 2, ..., N+1\}$, we get the second initial condition required in (2.16) to be $\mathbf{V}^1 \approx \mathbf{0}$.

N	Δt	Maximum absolute solution over all nodes and discrete-time steps	% from $N = 800$
50	3000^{-1}	47.096540561132862	0.66237463
100	6000^{-1}	48.124952897427079	1.50678765
150	9000^{-1}	47.714573116812034	0.64120066
200	12000^{-1}	47.574614025259791	0.34599418
250	15000^{-1}	47.511043619546868	0.21190932
300	18000^{-1}	47.476504975870455	0.13905924
350	21000^{-1}	47.456264515091256	0.09636737
400	24000^{-1}	47.443047317756452	0.06848921
800	48000^{-1}	47.410576190783118	0

Table 3.1: The convergence of our explicit FDM approximation of u in (1.1) for a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz and a disturbance location at $x_0 = 0.05$.

Using these initial conditions in (2.16), we can look at the convergence of the maximum absolute explicit FDM approximation of u across all nodes and discrete-time steps for different mesh densities in both space and time. Table (3.1) shows the results gathered at different mesh densities, and we decide to use a mesh density of $N = 50$ and $\Delta t = 3000^{-1}$ for our coarse mesh, whereas for our fine mesh, we will use a mesh with $N = 250$ nodes and $L = 15000$ discrete-time steps.

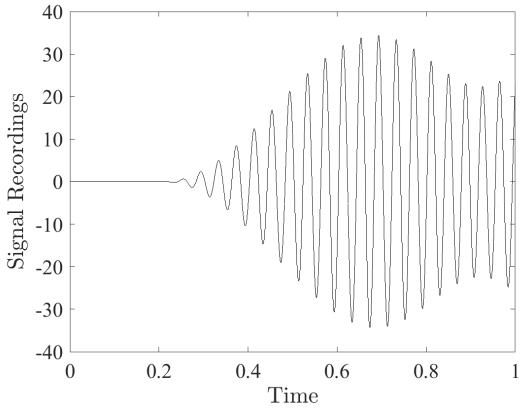
In this chapter we investigate the feasibility of using explicit FDM approximations of u from a coarse mesh for our sensor traces to reduce the matrix dimensions in the KF. Therefore, we compare the results obtained when our sensor traces originated from explicit FDM approximations of u from a coarse mesh to explicit FDM approximations of u from a fine mesh.

3.1.2 Added noise

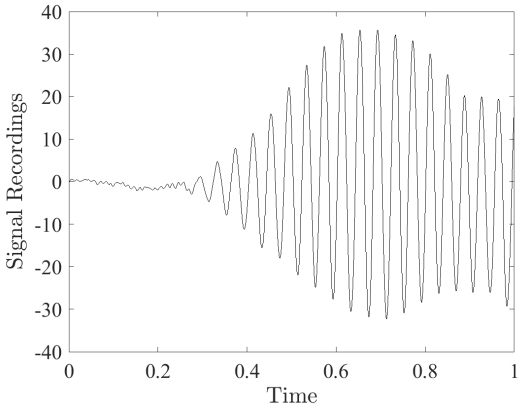
Recall from chapter 1 that we add two different forms of random Gaussian noise to our explicit FDM approximations of u in (2.16), denoted by $\tilde{\mathbf{w}}^n$ and $\tilde{\mathbf{z}}^n$ in the KF. We scale the first by the error associated with the explicit FDM approximations of u in (2.16). For our 1D model problem considered in this chapter, this error is measured by the combination of the higher-order terms neglected at the interior nodes and the Neumann boundary in (2.7) and (2.14), respectively. The second random Gaussian noise added attempts to mimic inaccuracies recorded by sensors within a real-life scenario, be that ambient noise in the background or errors due to manufacturing defects. Without experimental data, we cannot accurately predict the magnitude of this noise. As a result, we model it by the error associated with the explicit FDM approximation of u in (2.16) and scale it ensuring it is the larger of the two random Gaussian noises added since we would expect this noise to be the dominant of the two. We scale the noise added so that the original explicit FDM approximations of u are still recognisable but significantly obscured.

Figure (3.2) illustrates the effect the added noise has on our explicit FDM approximations of u in (2.16) for our 1D model problem. These figures show a sensor trace at $x := 0.2$ formed using a mesh density of $N = 50$ nodes with $L = 3000$ discrete-time steps for a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz, and a disturbance location at $x_0 = 0.004$. We scale the random Gaussian noise added to our explicit FDM approximations of u, see Figure (3.2)(b) by 5×10^4, and by 1×10^7 in

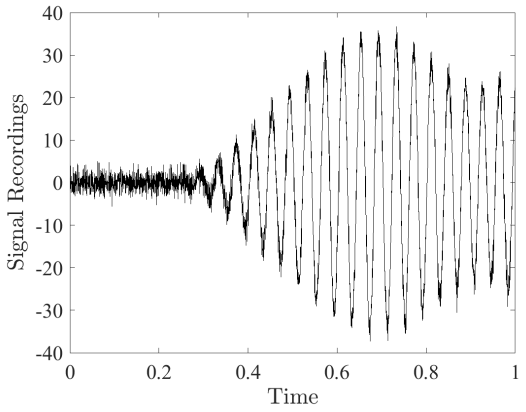
Figure (3.2)(c). In total, the noise added is significant, see Figure (3.2)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (3.2)(a).



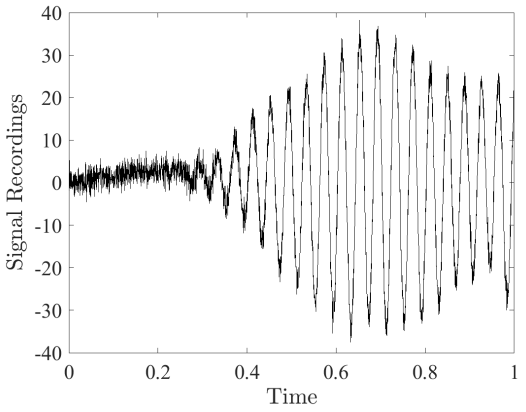
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.



(d) All added noise present.

Figure 3.2: Illustration of the noise added to our explicit FDM approximations of u in (2.16) using a single sensor trace at $x := 0.2$, a mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps, a disturbance frequency of $F = 25$Hz and a disturbance location at $x_0 = 0.004$.

3.1.3 Model problem schematic

In this section we outline the placement of sensors in our domain and detail the stages involved to obtain a prediction for the location of our disturbance. We will start by defining where sensors are placed, that is the nodes in the mesh the explicit FDM

approximations of u are recorded across all discrete-time steps.

In Figure (3.3), we outline the position of our sensors depending on how many are present. We always place a sensor at $x := 0.2$, on the Neumann boundary. On the occasions we have more than a single sensor, the rest are equidistant from one another. Since $u = 0$ at $x := 0$, we never place a sensor here. As illustrated in the results section, see Figure (3.6), having more than six sensors present has negligible impact on our models ability to accurately predict the location of the disturbance, and will result in increased matrix dimensions in the KF. Therefore, for our 1D model problems, we consider having between 1 and 6 sensors present in our domain.

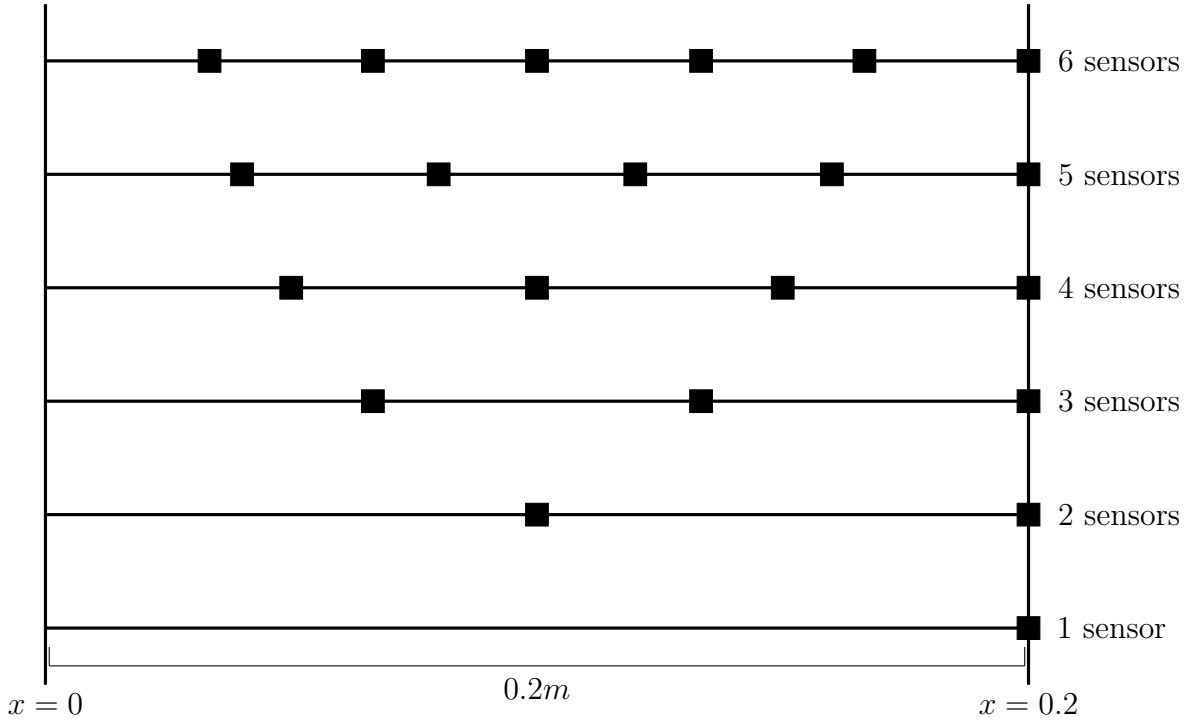


Figure 3.3: The distribution of 1-6 sensors, denoted by black squares, present on our 1D domain.

In chapter 1, we outlined a schematic in Figure (1.5) which gives a broad approach to the steps involved to obtain a prediction for the disturbance location, denoted by x_0. In this section we construct another schematic which goes into more detail for our 1D model problem without using the SVD.

Figure (3.4) shows a schematic illustrating the steps required to predict the location of our disturbance, denoted by x_0, for our 1D model problem without using an SVD. Our first step has two different routes. One route involves creating 50 random disturbance locations in our 1D domain defined on $\Omega_{1D} := (0, 0.2)$. When creating these, if the same location

occurs twice, we remove the duplicate and replace it with another random location until all 50 are unique. These are generated once for our 1D model problems, with the same 50 random disturbance locations used to ensure consistency between our probabilistic results. The second route in our first step involves taking 50 disturbances positioned where we evaluate the likelihood function in (2.107), that is $x_0 := \{0.004, 0.008, ..., 0.2\}$.

Moving onto the second step in our schematic, we solve the acoustic wave equation in (1.1) by approximating u using an explicit FDM approximation, see (2.16), over a total simulation duration of $T = 1$ second. We add two different forms of noise to the approximations of u, which we illustrated in Figure (1.5), and in the previous section of this chapter. Having added noise to our approximations of u, we store these across all discrete-time steps at the nodes in our mesh which correspond to sensor locations, see Figure (3.3).

Having obtained our sensor traces across all discrete-time steps from our explicit FDM approximations of u, we move onto step 3. We use these sensor traces as our observations in (2.92) for the KF in (2.93)-(2.98). We run the KF over a series of initial guesses for x_0, given by $x_0 := \{0.004, 0.008, ..., 0.2\}$, and compute the likelihood function in (2.107) using a minimisation algorithm for each initial guess using time-dependent data derived in the KF. From the 50 likelihood estimates obtained, one for each initial guess of x_0, we determine the smallest and deduce the corresponding x_0 to be the models prediction of the actual disturbance location used to generate the sensor traces. Using only a single initial guess for x_0, rather than 50, yielded bad results due to local minimums in (2.107). We solved this issue using the 50 initial guesses, however, this lengthened the run-time of our 1D model problem.

In chapters 4, 5 and 6 we consider model problems which result in infeasible run-times for our approach with the minimisation algorithm. As a result, we investigate the feasibility of a different method for our 1D model problems so we can compare the two approaches. In this approach, instead of using a minimisation algorithm, we simply take 50 equidistant guesses for x_0 and run the KF and compute the likelihood function in (2.107) for each. From the 50 likelihood estimates obtained, one for each guess of x_0, we determine the smallest and deduce the corresponding x_0 to be the models prediction of the actual disturbance location used to generate the sensor traces.

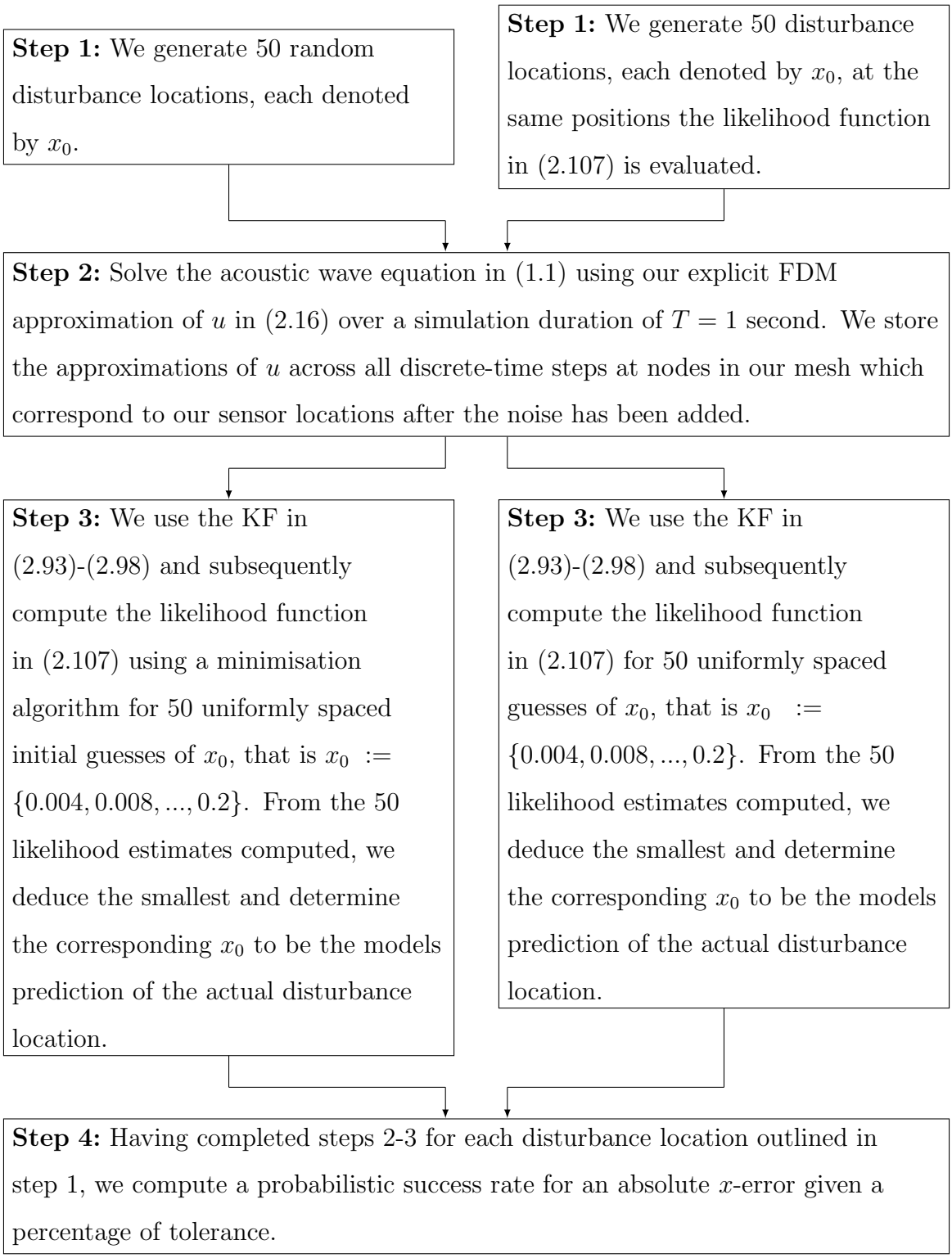


Figure 3.4: A schematic for our 1D model problem without using the SVD.

In step 4, having completed steps 2-3 in Figure (3.4) for each disturbance location in step 1, we compute probabilistic success rates for an absolute x-error given a percentage of tolerance. For our alternative approach without the minimisation algorithm, when

the disturbance locations are random we will not be evaluating the likelihood function in (2.107) where the actual disturbance is positioned. As a result, in the worst-case scenario when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible, we have an unavoidable error of 1%.

3.2 Results

In this section we explain our choice for our value of ϵ in (1.6) and how many sensors is enough. Trivially, insufficient sensors will result in less data being known to the KF resulting in worse results, and too many will result in longer run-times since the matrix dimensions in the KF, see (2.93)-(2.98), are dependent on the number of sensors present. Additionally, we discuss the results obtained from our 1D model problems using a minimisation algorithm. We compare the results from sensor traces collected from a coarse mesh with $N = 50$ nodes and $L = 3000$ discrete-time steps, to results from a fine mesh with $N = 250$ nodes and $L = 15000$ discrete-time steps. This is important because the matrix dimensions in the KF are dependent on the size of our mesh used to get our explicit FDM approximations of u.

In Figure (3.5) we have the success rate of our model for an array of different ϵ values, given three different tolerances using 50 different random disturbance locations for x_0 with a single sensor present at $y := 0.2$. The ϵ values chosen were dependent on the percentage of the Gaussian spreads peak remaining in the worst-case scenario. Recall that the worst-case scenario is when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible, and so, since the results in Figure (3.5) have used explicit FDM approximations of u from a mesh with 50 equidistant nodes, this distance is 0.002.

Upon inspection of Figure (3.5), we see the most significant level of success when the peak disturbance at the nearest node is 50%. In this case, $\epsilon = 5.77 \times 10^{-6}$, and so this is the value we take for ϵ in our forcing function.

Figure (3.6) shows the success rate of our 1D model problem for 3 different levels of relative tolerance, with a differing number of sensors present. Figure (3.6)(a) corresponds to sensor traces from a coarse mesh, whereas Figure (3.6)(b) represents sensor traces from a fine mesh.

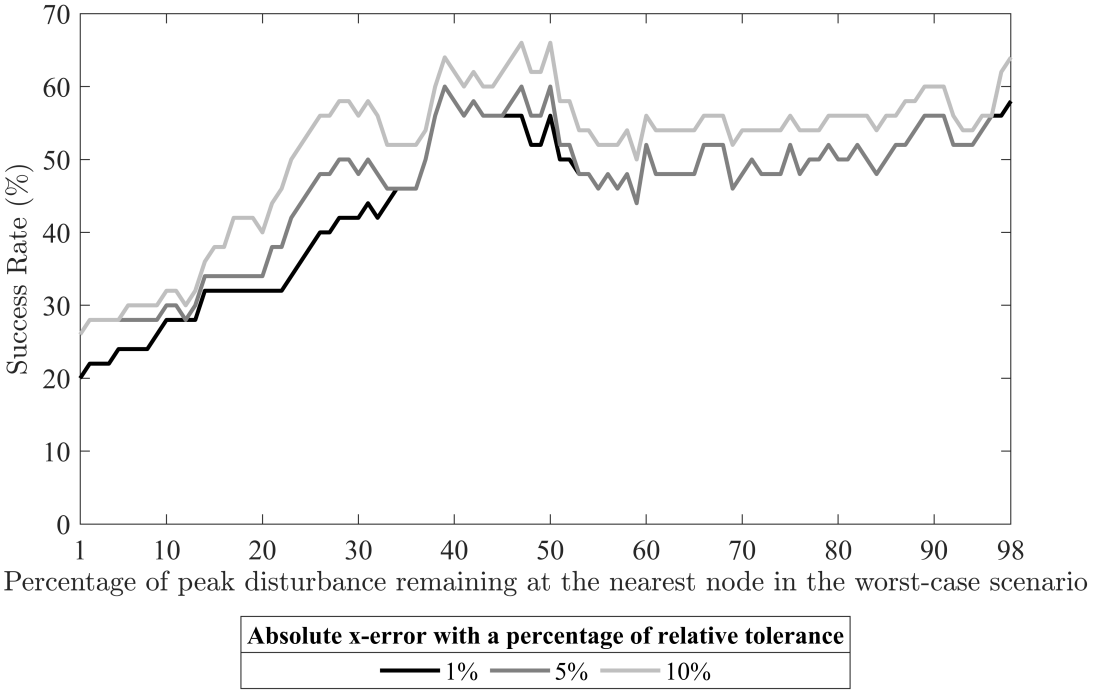
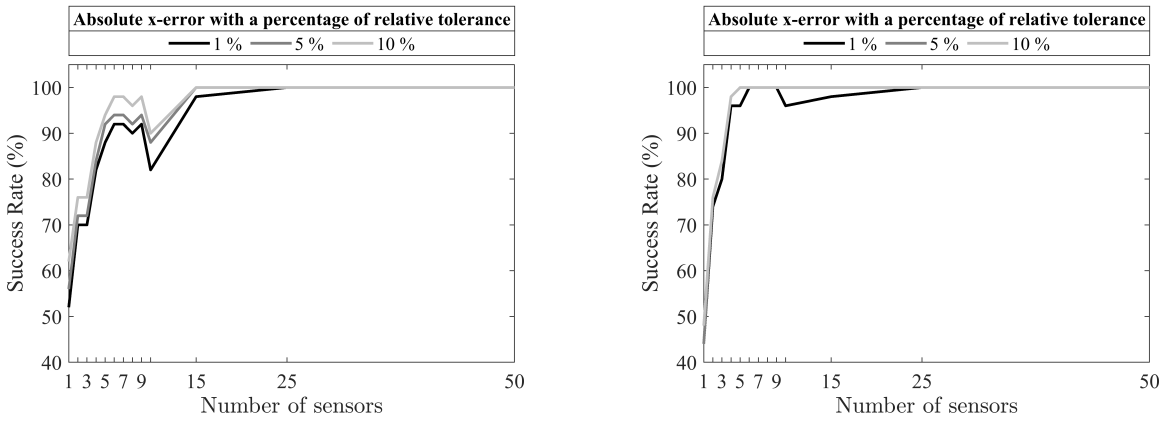


Figure 3.5: The success rate of our 1D model problem for an array of epsilon values in our forcing function, f, a single sensor present at $y := 0.2$, a mesh dimension of $N = 50$ and $L = 3000$, and random disturbance locations.



(a) Mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps.

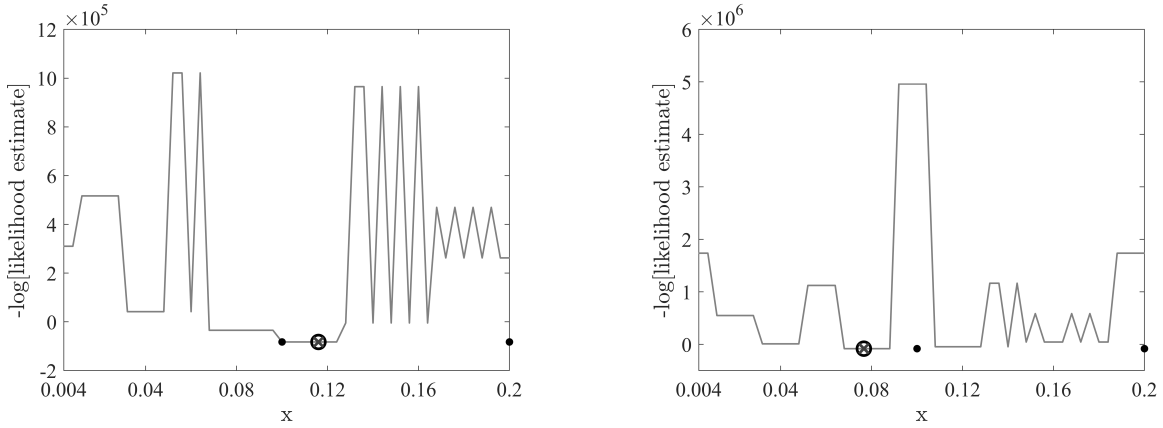
(b) Mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.

Figure 3.6: The success rate of our 1D model problem given an absolute x-error with different relative tolerances for a varying number of sensors present. The probabilistic success rates displayed were obtained using 50 random disturbance locations in our 1D domain.

As mentioned previously, we do not want too many sensors present due to the associated increase in matrix dimensions in the KF. Additionally, in a real-life scenario - a trained professional would not place 50 sensors onto a human thorax. As a result, upon inspecting Figure (3.6), it makes sense to run our 1D model problems for an array of sensors starting with 1 and going up to 6, where we observe good success rates. Furthermore, it is worth noting that the success rates obtained using sensor traces from a coarse mesh look promising in Figure (3.6)(a), and so, we will be investigating this further.

3.2.1 FDM approximation of u from a fine mesh

In this section, we look at results corresponding to sensor traces obtained from an explicit FDM approximation of u on a fine mesh with $N = 250$ nodes and $L = 15000$ discrete-time steps.



(a) Disturbance location at $x_0 = 0.116$.

(b) Disturbance location at $x_0 = 0.076608$.

Figure 3.7: The likelihood estimates across our 1D domain using a minimisation algorithm with a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps to generate our explicit FDM approximations of u which are recorded at two sensors, shown as solid black dots. The actual disturbance location is shown as a hollow black circle, with the models prediction of this location denoted by a grey cross.

Figure (3.7) shows the likelihood function in (2.107) plotted against the corresponding location of x_0 which was obtained using the minimisation algorithm. This algorithm uses 50 equidistant initial guesses for x_0 across our 1D domain. We do this to reduce the possibility of obtaining bad results from using only a single initial guess for x_0. Figure

(3.7)(a) has an actual disturbance location of $x_0 = 0.116$, whereas in Figure (3.7)(b) it is at $x_0 = 0.076608$. The examples shown used 2 sensors, denoted by black dots, and in both cases, in Figure (3.7), our model has accurately predicted the disturbance location.

Having seen two examples where our 1D model has successfully predicted the actual disturbance location, we now look at probabilistic results from 50 different disturbance locations. The full set of results can be found in Appendix A.1, where the results are presented both graphically and in tabular form.

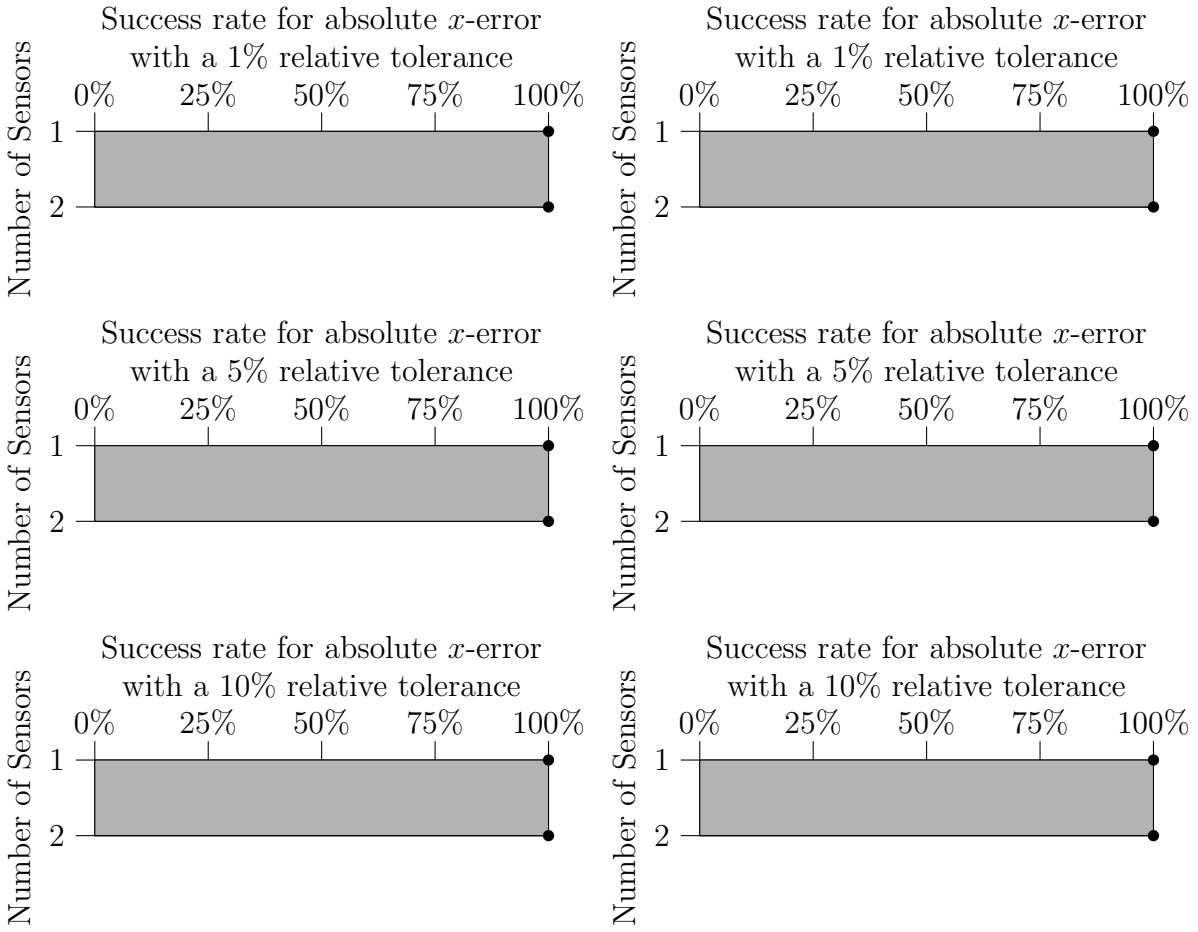


Figure 3.8: The probabilistic success rates for our 1D model problem using a minimisation algorithm with a varying number of sensors, a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps. The results on the left-hand side (LHS) come from 50 disturbance locations where the likelihood function is evaluated, and on the right-hand side (RHS) from 50 random disturbance locations.

Figure (3.8) has been constructed to show the probabilistic success rates for both 50 random disturbance locations and 50 disturbance locations positioned where the likelihood

in (2.107) is evaluated. The number of sensors on our 1D domain when generating the success rates are shown on the left of each figure, in this case one or two, with the probabilistic success rate denoted by the position of the black dot aligned with the number of sensors present on the left of each figure. The figures show that we get a 100% success rate, irrespective of the number of sensors on our domain and different initial guesses made for the minimisation algorithm. This is what we would expect, since we are using a minimisation algorithm where a more optimal location for our disturbance can be predicted as it can move away from the initial guess.

Moreover, we only have results for when there are 1 and 2 sensors present. The reason for this is due to the significant run-time required to run our 1D model problem for 50 different disturbance locations. In each case, it is run through a minimisation algorithm running the KF many times until a preset tolerance is met. In addition to this, we have large matrix dimensions in the KF due to the sensor traces originating from a fine mesh, which slows down our 1D model problem. Despite this, as seen in Figure (3.8), for 1 and 2 sensors, we get a 100% success rate, even at a small 1% relative tolerance. As a result, we do not need to run our 1D model problem for more than 2 sensors.

3.2.2 FDM approximation of u from a coarse mesh

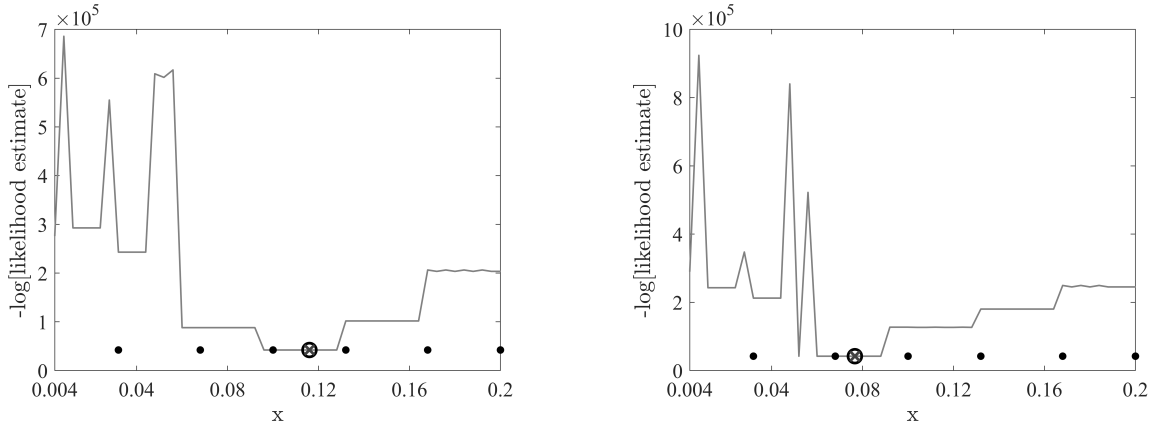
In this section, we look at results corresponding to sensor traces obtained from our explicit FDM approximations of u on a coarse mesh with $N = 50$ nodes and $L = 3000$ discrete-time steps.

Figure (3.9) shows the likelihood function in (2.107) plotted against the corresponding location of x_0 which was obtained using the minimisation algorithm. This algorithm uses 50 equidistant initial guesses for x_0 across our 1D domain. We do this to eliminate the possibility of obtaining bad results from using only a single initial guess for x_0. Figure (3.9)(a) has an actual disturbance location of $x_0 = 0.116$, whereas in Figure (3.9)(b) it is at $x_0 = 0.076608$. The examples shown used 6 sensors, shown as black dots, and in both cases, in Figure (3.9), our model has accurately predicted the disturbance location.

Having seen two examples where our 1D model has successfully predicted the actual disturbance location, we now look at probabilistic results from 50 different disturbance locations. The full set of results can be found in Appendix A.2, where the results are shown both graphically and in tabular form.

Figure (3.10) has been constructed to show the probabilistic success rates for 50 random disturbance locations and 50 disturbances positioned where the likelihood function in (2.107) is evaluated. The number of sensors on our 1D domain when generating the success rates are shown on the left of each figure, in this case one through to six, with the probabilistic success rate denoted by the position of the black dot aligned with the number of sensors present on the left of each figure. Since we are using a minimisation algorithm, an optimal location for our disturbance can be more accurately predicted as it can move away from the initial guess. As a result, we would not expect any substantial difference between the success rate in Figure (3.10) on the left when compared to the success rate on the right, and this is what is observed.

Moreover, in every case across all error tolerances, we see a lower success rate when only a single sensor is present. The success rate when 2 or more sensors are present is 100%, the same as what was observed when we had sensor traces from a fine mesh.



(a) Disturbance location at $x_0 = 0.116$. **(b)** Disturbance location at $x_0 = 0.076608$.

Figure 3.9: The likelihood estimates across our 1D domain using a minimisation algorithm with a mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps to generate our explicit FDM approximations of u which are recorded at six sensors, shown as solid black dots. The actual disturbance location is shown as a hollow black circle, with the models prediction of this location denoted by a grey cross.

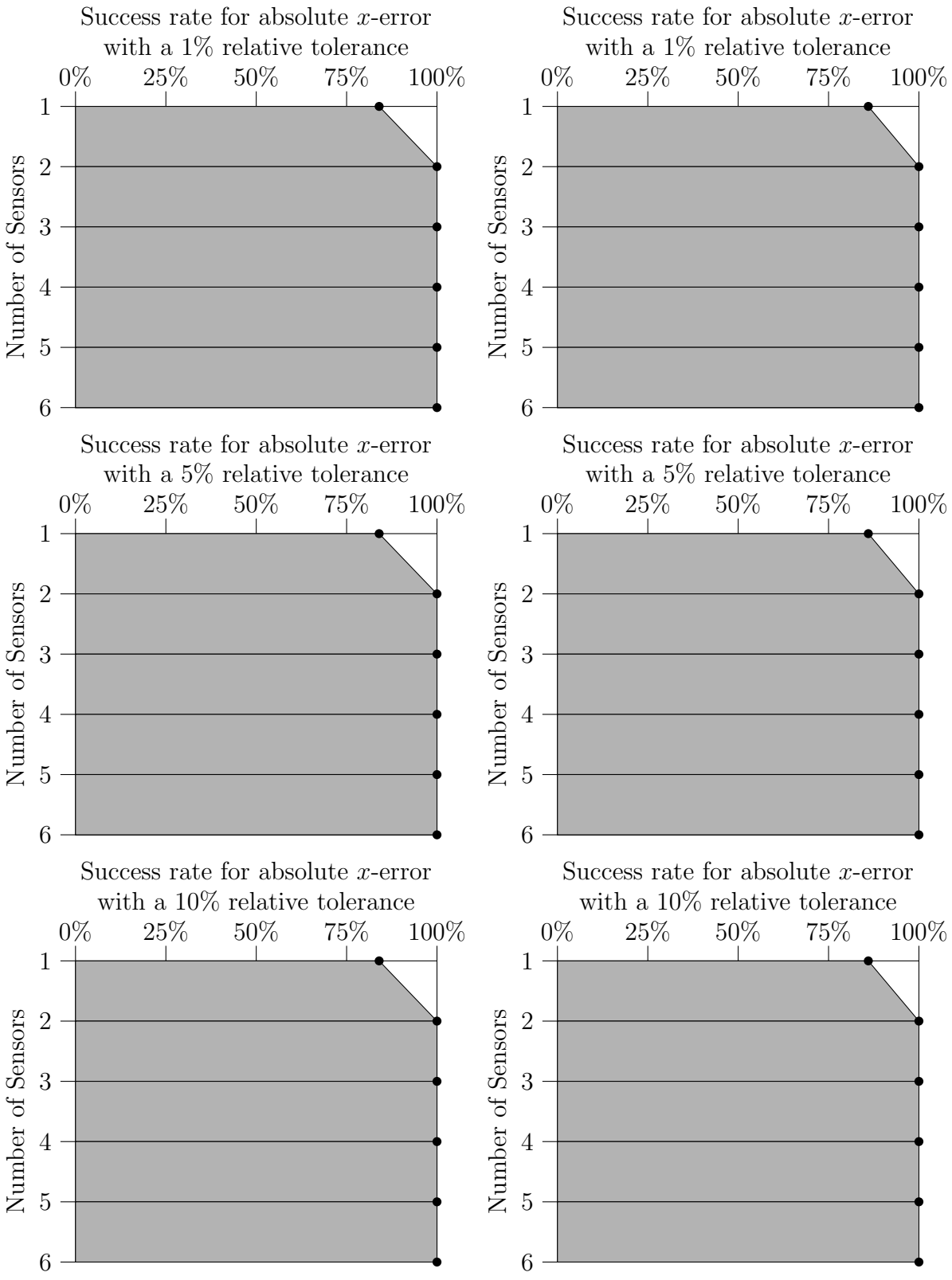


Figure 3.10: The probabilistic success rates for our 1D model problem using a minimisation algorithm with a varying number of sensors, a mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps. The results on the LHS come from 50 disturbance locations where the likelihood function is evaluated, and on the RHS from 50 random disturbance locations.

3.2.3 Summary

Now we have presented the results for our 1D model problems where the sensor traces have come from fine and coarse meshes, we can compare the two approaches.

We first saw results corresponding to sensor traces from a fine mesh, that is one with $N = 250$ nodes and $L = 15000$ discrete-time steps. Of course, this yielded better results which we would expect since the sensor traces are more accurately approximating u in (1.1). However, it was observed that the results were only better when a single sensor was present. For our results corresponding to our coarse mesh with $N = 50$ nodes and $L = 3000$ discrete-time steps, with 2 or more sensors present, the probabilistic success rate was equivalent to our 1D model problem using sensor traces from a fine mesh.

This result is very significant. Not only does it show that the KF in (2.93)-(2.98) can reasonably account for the larger error in the sensor traces from the coarse mesh caused by less accurate approximations of u in (2.16), but it also means that we can use approximate solutions of u from a coarse mesh in (1.1) from (2.16), which has a very significant impact on the matrix dimensions in the KF.

Although this is not a huge problem when only considering 1D model problems, with a disturbance whose frequency is $F = 25$Hz, it is significant when we have higher-dimensional problems or higher frequencies for our disturbance in (1.6). We will return to this point with an explanation later in chapters 4, 5 and 6 where we consider such model problems.

3.3 Alternative approaches

In this section, we consider two different methods to locate our disturbance in (1.6). In the first approach, we do not use a minimisation algorithm as we have done so already. Instead, we evaluate the likelihood in (2.107) at 50 equidistant locations, given by $x_0 := \{0.004, 0.008, ..., 0.2\}$, creating the unavoidable possibility of a bad result since the largest distance possible between the location of our disturbance and where we evaluate the likelihood in (2.107) has a relative error of 1%. The reason we explore this method in 1D is due to the run-time required to use the minimisation algorithm, which for higher-dimensional problems and problems with a higher frequency in the forcing function, would not be feasible.

We also look at using an SVD to reduce the matrix dimensions in the KF, see the modified KF in (2.127)-(2.132) due to the SVD. These results have been obtained without the use of the minimisation algorithm. Additionally, in 1D, it is possible to run our model problems with sensor traces originating from fine and coarse meshes so that we can compare the results between the two.

3.3.1 Without using a minimisation algorithm

In Figure (3.11) we have four examples, that is two different disturbance locations using two different mesh densities to generate the sensor traces, showing the likelihood function in (2.107) plotted against the corresponding x_0 in our 1D domain. Where this approach differs from what we have already done is that we do not use a minimisation algorithm. Meaning what was our initial guesses for x_0 in our minimisation algorithm before, are now our only guesses for the location of our disturbance.

What is of significance in Figure (3.11) is that the disturbance location in Figure (3.11)(b) and Figure (3.11)(d) is not at a position where we evaluate the likelihood function using this approach, and so it is impossible for our model to predict the exact location of our disturbance. However, as can be seen by inspecting both Figure (3.11)(b) and Figure (3.11)(d), our model has determined the location of our disturbance to be at a position we evaluate the likelihood function, which is very close to the actual location, giving us optimism for this approach.

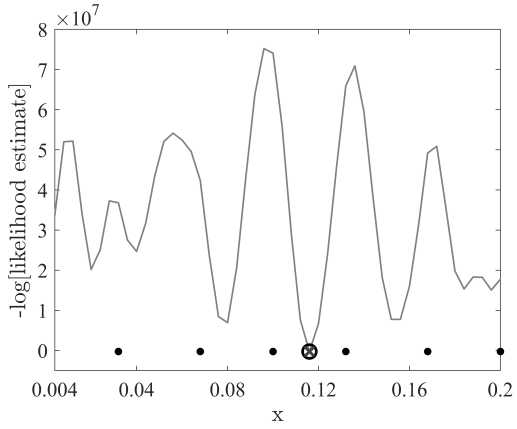
Having seen four examples where our model has successfully predicted the disturbance location, we now look at probabilistic results from 50 different disturbance locations. The full set of results can be found in Appendix A.3, where the results are shown both graphically and in tabular form.

In Figure (3.12) and Figure (3.13), we have the success rate based on three different relative tolerances for a varying number of sensors present, using sensor traces from a fine and coarse mesh, respectively.

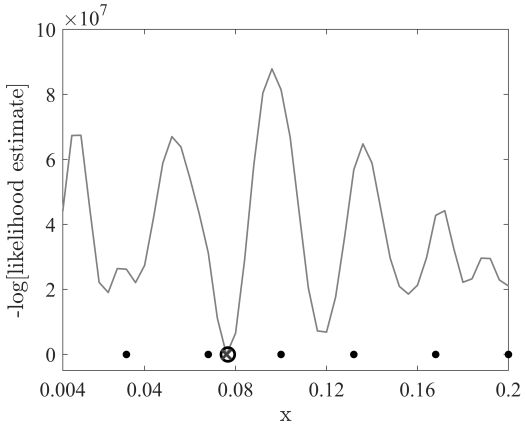
In both cases, when the disturbance location is at a position where we evaluate the likelihood function, and we have more than only 1 sensor present, we have a success rate of 100% which is very promising.

Again, in both cases, when the disturbance locations are random, the model struggles to get a reasonable success rate when there are fewer sensors present. We can attribute

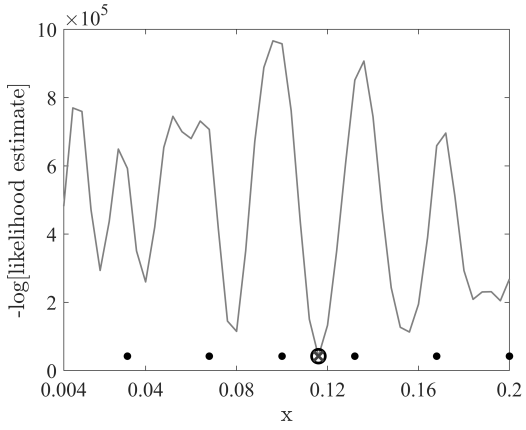
this to the lack of information known to the KF due to the sensor count, and the fact that if the disturbance is not where we evaluate the likelihood function, we can see that this will have an impact on the success rate by comparing Figure (3.8) with Figure (3.12), and Figure (3.10) with Figure (3.13).



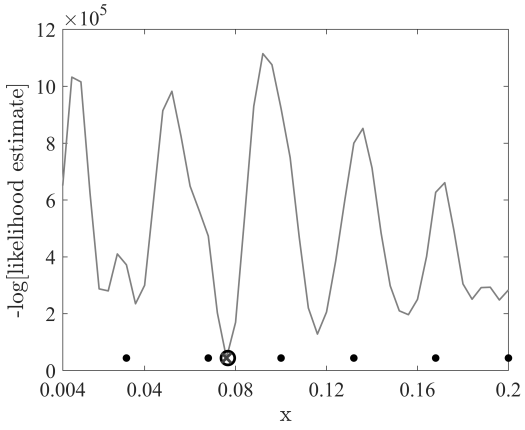
(a) Disturbance location at $x_0 = 0.116$ with a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.



(b) Disturbance location at $x_0 = 0.076608$ with a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.



(c) Disturbance location at $x_0 = 0.116$ with a mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps.



(d) Disturbance location at $x_0 = 0.076608$ with a mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps.

Figure 3.11: The likelihood estimates across our 1D domain without using a minimisation algorithm for two different mesh dimensions to generate our explicit FDM approximations of u which are recorded at six sensors, shown as solid black dots. The actual disturbance location is shown as a hollow black circle, with the models prediction of this location denoted by a grey cross.

When we have 6 sensors present; however, the success rate for both cases do very well. By comparing Figure (3.12) to Figure (3.13), we can see that the success rate is better for the example where the sensor traces were generated from a fine mesh. This means that using sensor traces from a coarse mesh, when the disturbance location is random, has a small impact on the probabilistic success rate, as opposed to using sensor traces generated from a fine mesh. However, the margin is tiny, and for the run-time reduction that we will require in higher-dimensional problems - this approach seems to be a viable option.

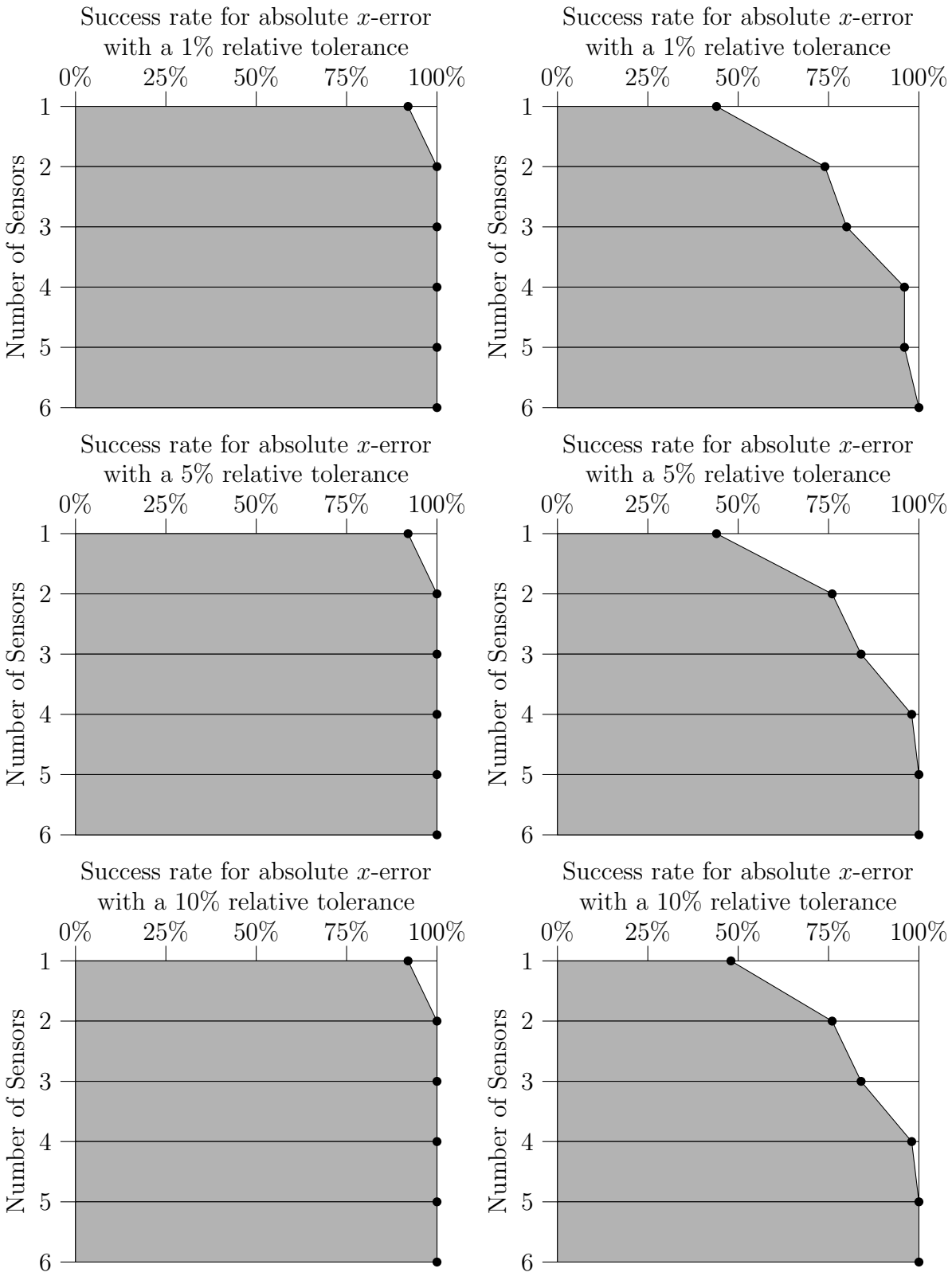


Figure 3.12: The probabilistic success rates for our 1D model problem without using a minimisation algorithm with a varying number of sensors, a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps. The results on the LHS come from 50 disturbance locations where the likelihood function is evaluated, and on the RHS from 50 random disturbance locations.

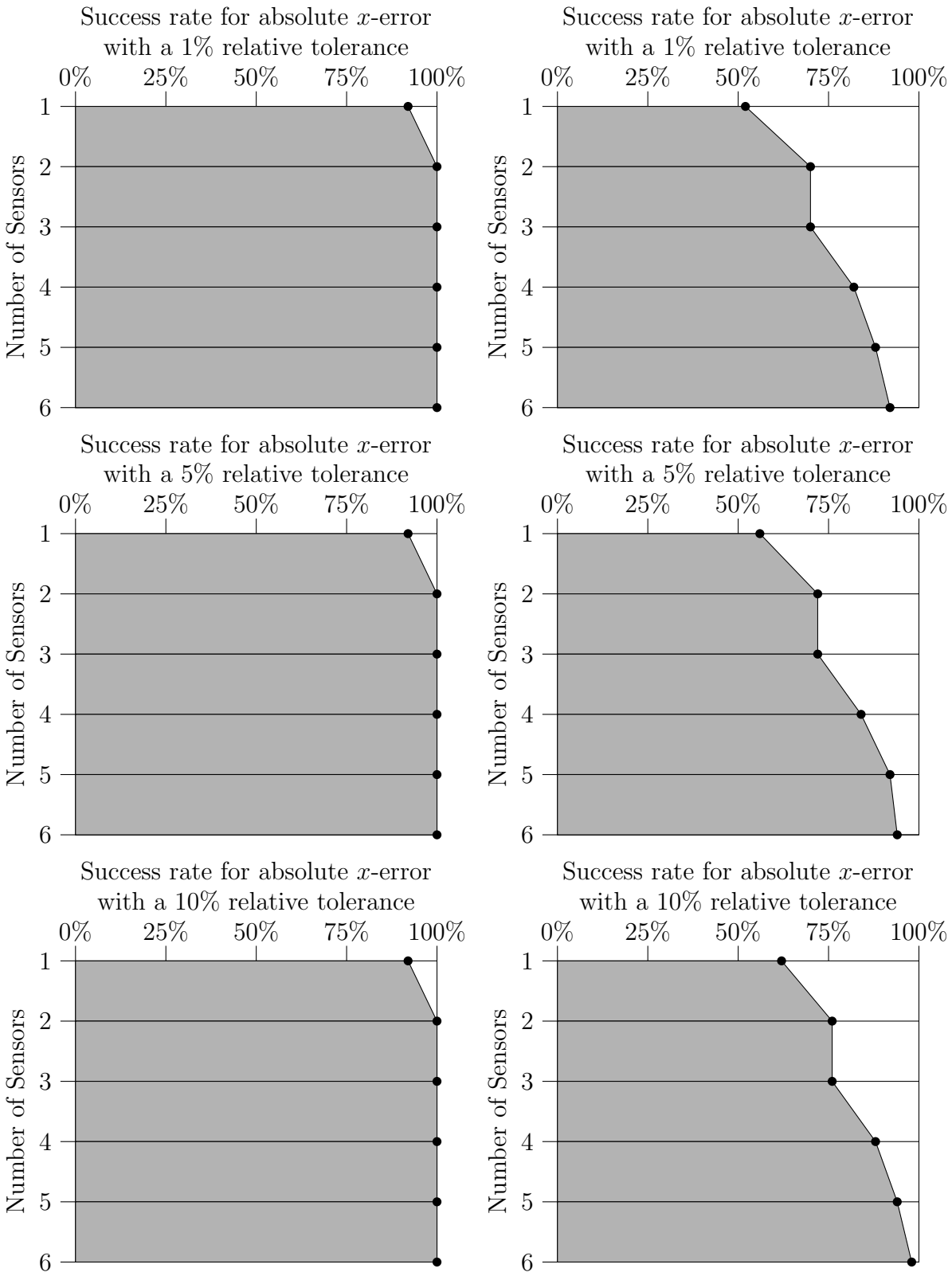


Figure 3.13: The probabilistic success rates for our 1D model problem without using a minimisation algorithm with a varying number of sensors, a mesh dimension of $N = 50$ nodes and $L = 3000$ discrete-time steps. The results on the LHS come from 50 disturbance locations where the likelihood function is evaluated, and on the RHS from 50 random disturbance locations.

3.3.2 Use of an SVD to reduce matrix dimensions in the KF

In this section, we use an SVD to reduce the matrix dimensions in the KF. The largest matrix in the standard KF, see (2.93)-(2.98), is $\underline{\mathbf{P}}^{n+1|n}$ in (2.98) with dimensions of $\big(2(N+1)\big)^2$, where N represents the number of nodes in our mesh used to approximate u in (1.1). In the modified KF, using a subset of the data formed by an SVD, the equivalent matrix in (2.132) has a dimension of $(2d)^2$ where d represents the first d columns in the matrix containing the left singular vectors in (2.108), which have the largest singular values. As a result, the matrix dimensions are significantly reduced, meaning faster run-times and a significant decrease in the amount of system RAM required, see Figure (3.15) for an illustration of this.

The benefits of changing the dynamics of our system using an SVD, rather than using the dynamics of our explicit FDM approximations of u in the KF are evident. We now look at how we form an SVD, using data from the explicit FDM approximations of u in (2.16). Previously, our total simulation duration was $T = 1$ second. We did attempt to keep the simulation length the same, however, it was apparent when looking at sensor traces with a total simulation length of $T = 1$ second, see Figure (3.2), that using data from (2.16) over the first 40% of the simulation to form the SVD would not have produced an SVD that reflects the actual sensor traces produced in (2.16). As a result, we decided to extend the total simulation time to $T = 3$ seconds.

Figure (3.14) shows a schematic of the new simulation duration of $T = 3$ seconds. We chose to disregard the first second of data from (2.16), ensuring any transients had died out. We then select $T_s := \{1, 0.5, 0.25, 0.125\}$, which represents the segment of time we use the explicit FDM approximations of u to form our SVD. As illustrated in Figure (3.14), T_s is taken from $T = 2$ to $T = 1$ second. Doing this ensures the modified KF in (2.127)-(2.132) always runs over the third second of our simulation irrespective of the choice of T_s. Once we have formed an SVD for our explicit FDM approximations of u in (2.16), we take the first d columns from the matrix containing the left singular vectors to form (2.110) - these are the principal components of our SVD with the largest singular values.

Using the principal components in (2.110), we can run the modified KF in (2.127)-(2.132) over the last second of our simulation, using estimated sensor traces generated using our d principal components over the final second of our simulation.

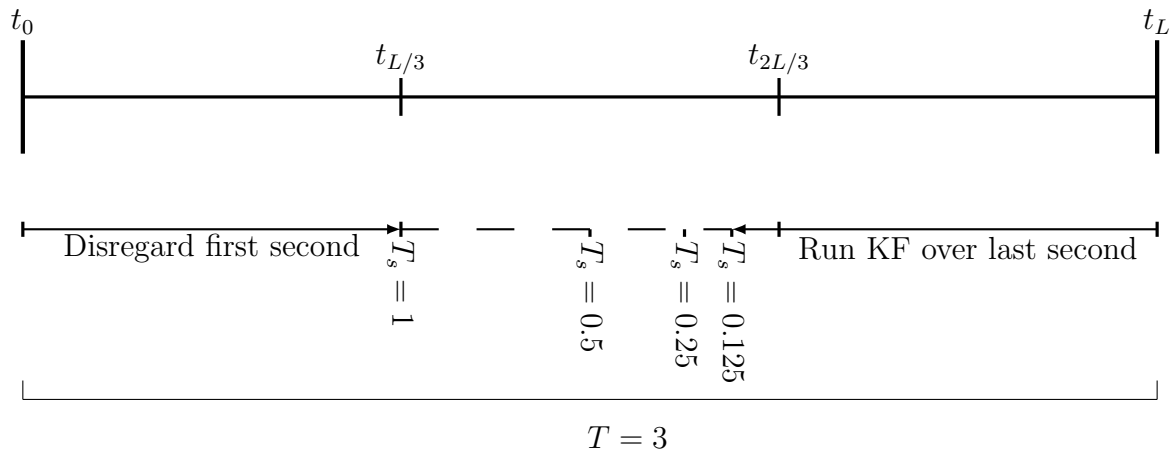


Figure 3.14: An illustration showing the breakdown of our simulation time for the model problems using an SVD, where T_s represents the duration, in time, of data used from the FDM approximations of u to form the SVD, t_n denotes a discrete-time step where $n \in \{0, 1, 2, ..., L\}$ and T is the duration of our simulation.

Recall from chapter 1 the schematic in Figure (1.5) which gives a broad approach to the steps involved to obtain a prediction for the disturbance location, denoted by x_0. Since our simulation duration and the way we obtain our sensor traces has changed, the schematic outlined in Figure (3.4) does not apply to our 1D model problem using an SVD, and so, we construct a new schematic in Figure (3.16).

In Figure (3.16), step 1 is the same as before. In step 2, we generate and store our explicit FDM approximations of u at every node in our mesh across a simulation from $T = 0$ to $T = 2$ seconds.

In step 3, we disregard the first $(2 - T_s)$ seconds of our explicit FDM approximations of u stored in step 2, and use the remaining approximations of u at every node in our mesh, stored in (2.109) to form our SVD. We then take the first d columns in the matrix containing the left singular vectors in (2.108) to establish our principal components, see (2.110). These principal components correspond to the largest singular values, which account for the most significant amount of variance in our data.

In step 4, we use these principal components stored in (2.110) to dynamically reduce our numerical scheme. Having achieved this, we run our reduced wave equation over the third second of our simulation to produce sensor traces on the reduced system. Lastly, we approximate the sensor traces for our original numerical scheme using the sensor traces from our reduced system. Random noise is then added to these approximated sensor

traces.

Moving onto Step 5, we use the approximated sensor traces in our modified KF over the third second of our simulation. We run the KF 50 times for different disturbance location guesses, denoted by $x_0 := \{0.004, 0.008, ..., 0.2\}$, each resulting in a likelihood function using (2.107). From the 50 likelihood estimates computed, we compute the smallest and conclude that the corresponding disturbance location is the models prediction of the true disturbance location used to generate our sensor traces.

Finally, in step 6, we compute probabilistic success rates given a relative percentage of tolerance given an absolute x-error. We are able to do this by following steps 2-5 for each disturbance location in step 1. Using these results, we conclude how well our 1D model problem is at predicting the disturbance location.

We have already discussed the advantages of using an SVD, as opposed to not using one in our 1D model problems. In Figure (3.15) we can see the system RAM requirements in gigabytes (GB) for the KF, with six sensors and 12 principal components, against different mesh refinements used to get our explicit FDM approximations of u in (2.16). We can see, upon inspection of Figure (3.15), that the system RAM requirements for the KF in our 1D model problem using an SVD are significantly less when compared to our 1D model problem without an SVD, as was expected.

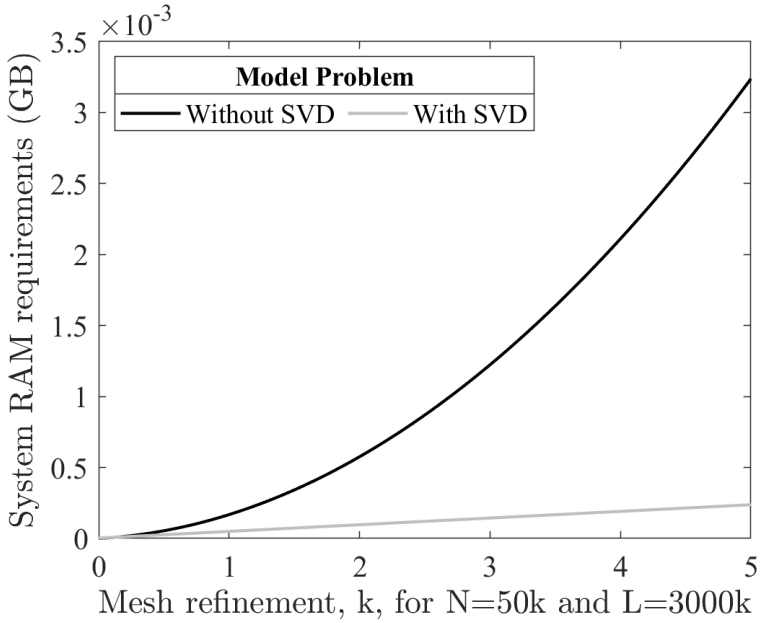


Figure 3.15: The system RAM requirements for the KF used in our 1D model problems with six sensors present and $d = 12$, when an SVD is used in our 1D model problem.

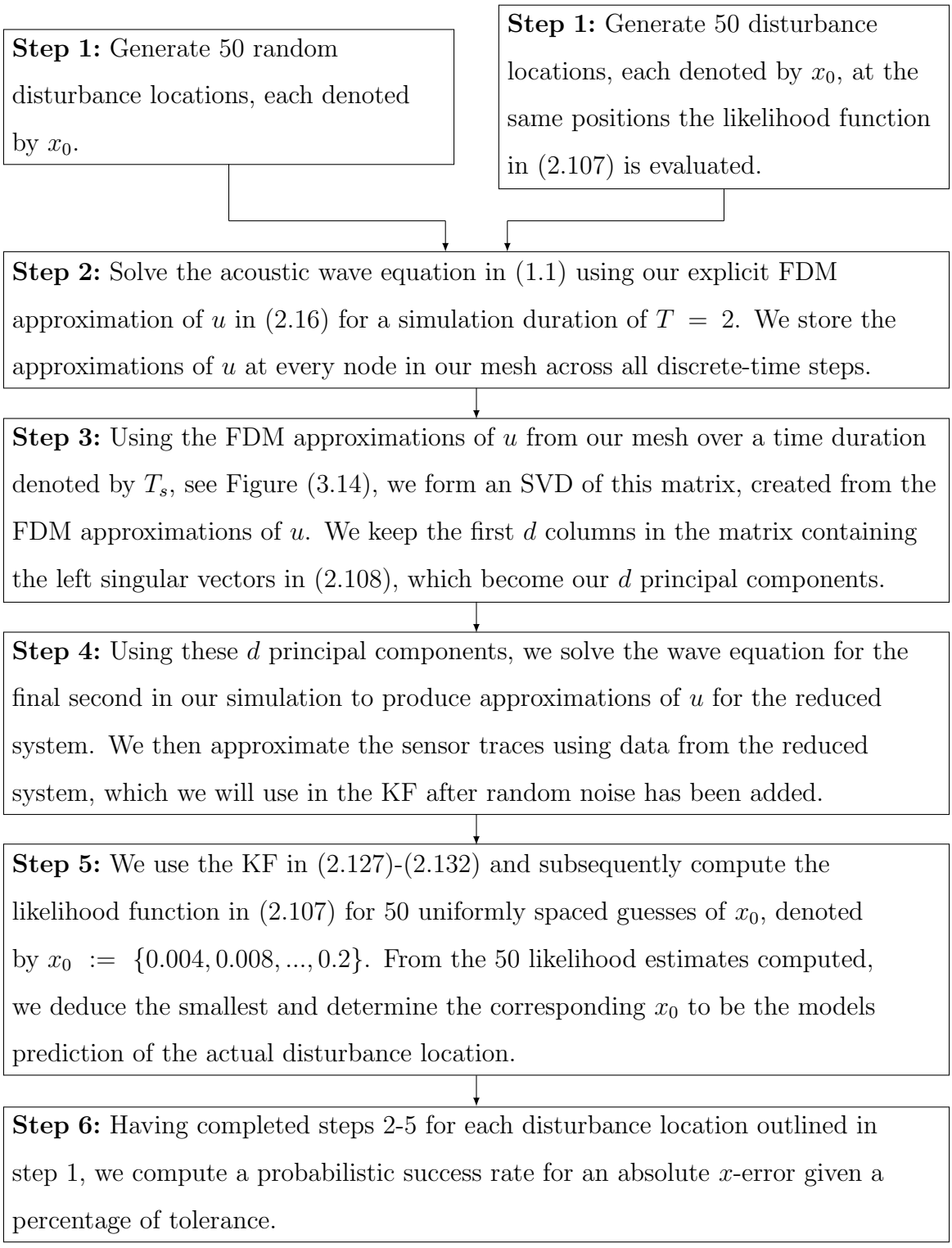


Figure 3.16: A schematic for our 1D model problem which uses an SVD.

Since our simulation duration has changed, we need to redo the convergence testing of our explicit FDM approximations of u. Again, we look for the maximum absolute approximation of u in (2.16) at every node in our mesh and across all discrete-time steps for the new simulation duration of $T = 3$ seconds. Using the same initial conditions in

(2.16) as before, Table (3.2) contains the convergence of our explicit FDM approximation of u computed at different mesh densities. Therefore, we decide to use a mesh with $N = 50$ nodes and $L = 9000$ discrete-time steps for our coarse mesh, whereas for our fine mesh we will use a mesh with $N = 250$ nodes and $L = 45000$ discrete-time steps. The percentage error from a converged solution for these explicit FDM approximations of u is 22% and 1%, respectively.

N	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 1300$
50	9000^{-1}	59.753396	22.11650406
100	18000^{-1}	80.652083	5.122211805
150	27000^{-1}	78.490472	2.304757441
200	36000^{-1}	77.798606	1.402975908
250	45000^{-1}	77.420574	0.9102471477
300	54000^{-1}	77.204065	0.6280482567
350	63000^{-1}	77.071149	0.4548048914
400	72000^{-1}	76.984083	0.3413226405
450	81000^{-1}	76.923656	0.2625617928
500	90000^{-1}	76.880536	0.2063589736
550	99000^{-1}	76.848262	0.1642928839
600	108000^{-1}	76.823909	0.1325510645
650	117000^{-1}	76.804888	0.1077589999
1300	234000^{-1}	76.722213	0

Table 3.2: The convergence of our explicit FDM approximation of u in (1.1) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 25$Hz and a disturbance location at $x_0 = 0.05$.

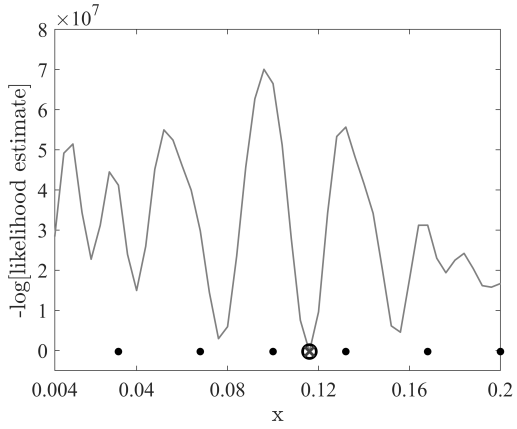
Due to our simulation duration changing, we cannot compare the results in this section to those earlier on in this chapter. As a result, Table (3.3) and Table (3.4) contain probabilistic success rates for an array of sensors, where the KF is run over the final, third second in our simulation. In Table (3.3), the probabilistic results correspond to 50 disturbance locations positioned where we evaluate the likelihood function, whereas the results in Table (3.4) come from 50 random disturbance locations.

	Success Rate		
Number of sensors	1%	5%	10%
1	64	64	66
2	98	98	98
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100

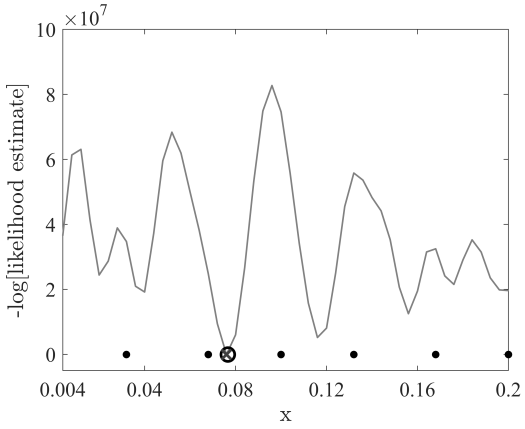
Table 3.3: The probabilistic success rates for our 1D model problem without using a minimisation algorithm with a mesh dimension of $N = 50$ nodes and $L = 9000$ discrete-time steps over a simulation duration of $T = 3$ seconds using 50 disturbance locations where the likelihood function is evaluated. The KF is run over the final second only in an attempt to mimic the scenario considered when using an SVD.

	Success Rate		
Number of sensors	1%	5%	10%
1	44	44	48
2	70	72	76
3	74	76	78
4	82	84	88
5	86	92	94
6	94	96	98

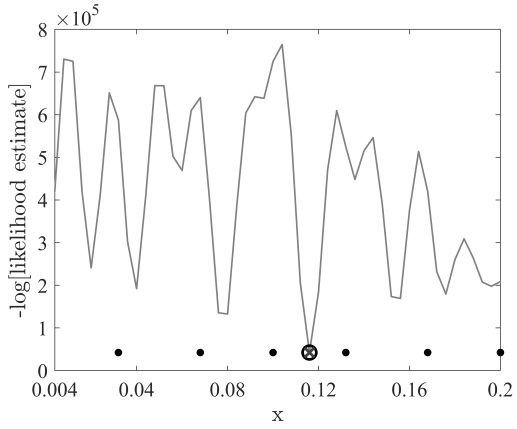
Table 3.4: The probabilistic success rates for our 1D model problem without using a minimisation algorithm with a mesh dimension of $N = 50$ nodes and $L = 9000$ discrete-time steps over a simulation duration of $T = 3$ seconds using 50 random disturbance locations. The KF is run over the final second only in an attempt to mimic the scenario considered when using an SVD.



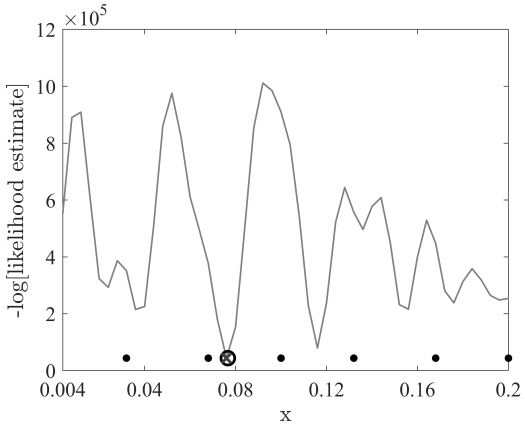
(a) Disturbance location at $x_0 = 0.116$, and a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.



(b) Disturbance location at $x_0 = 0.076608$, and a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.



(c) Disturbance location at $x_0 = 0.116$, and a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.



(d) Disturbance location at $x_0 = 0.076608$, and a mesh dimension of $N = 250$ nodes and $L = 15000$ discrete-time steps.

Figure 3.17: The likelihood estimates across our 1D domain using an SVD with $Ts = 0.5$ and $d = 12$ for two different mesh dimensions to generate our explicit FDM approximations of u which are recorded at six sensors, shown as solid black dots. The actual disturbance location is shown as a hollow black circle, with the models prediction of this location denoted by a grey cross.

Having now defined everything for our 1D model problems using an SVD, we discuss the results obtained. In Figure (3.17) we have four examples, that is two different disturbance locations using two different mesh densities to generate the explicit FDM approximations of u, showing the likelihood function in (2.107) plotted against the

corresponding location in our domain that (2.107) was evaluated.

What is of significance in Figure (3.17) is that the disturbance location in Figure (3.17)(b) and Figure (3.17)(d) are not at a position where we evaluate the likelihood function, and so our model is not able to predict the exact location of our disturbance. However, as can be seen by inspecting both Figure (3.17)(b) and Figure (3.17)(d), our model has determined the location of x_0 to be at a position we evaluate the likelihood function which is the closest to the actual disturbance location. We observed the same in Figure (3.7), Figure (3.9), and Figure (3.11) where an SVD was not used.

Having seen four examples where our model has successfully predicted the disturbance location, we now look at probabilistic results from 50 different disturbance locations. The full set of results can be found in Appendix A.4, where the results are shown both graphically and in tabular form. Amongst these results, we have one through to six sensors on our domain for both a fine and coarse mesh with two different types of disturbance locations used to generate the success rates.

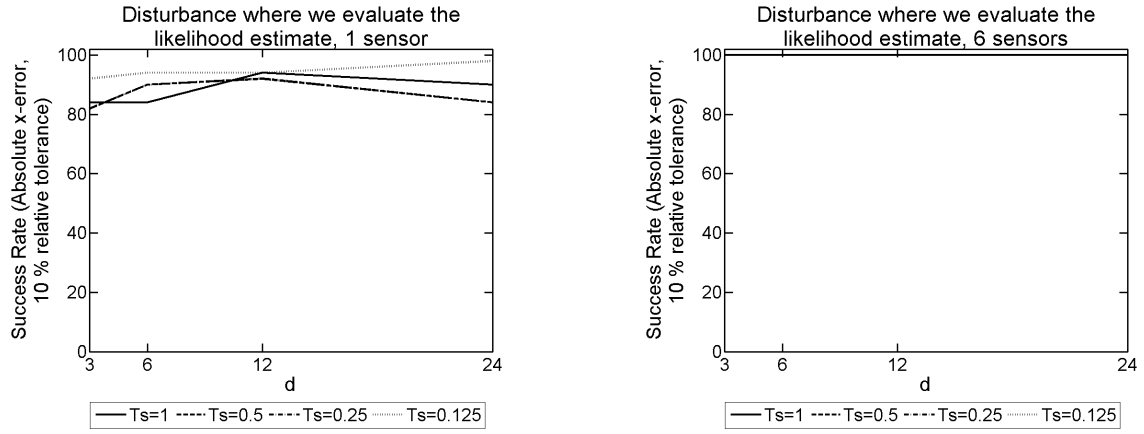
In Figures (3.18), (3.19), (3.20) and (3.21), we have the success rate, given a 10% relative tolerance, of our 1D model problem using an SVD with different values for T_s and d. That is, the length of time the explicit FDM approximations of u in (2.16) are used to construct the SVD, and the number of principal components taken from the SVD formed, respectively.

We start by comparing Figure (3.18) to Figure (3.19), where the disturbance locations used to deduce the probabilistic results are positioned where the likelihood function is evaluated. In both figures, the values chosen for T_s and d have little impact on our success rate. The only difference between the results in both figures is the accuracy of the explicit FDM approximations of u used to form our SVD. Figure (3.18) uses approximations of u from a fine mesh, whereas Figure (3.19) uses them from a coarse mesh. We observe that when there is only a single sensor present in our domain, at $x := 0.2$, the results from the use of a fine mesh are substantially better, see Figures (3.18)(a) and (3.19)(a). However, we also observe that when we have six sensors present in our domain, the results between the two mesh densities used are indistinguishable, see Figures (3.18)(b) and (3.19)(b).

This result is significant, as it means using a coarse mesh to obtain our explicit FDM approximations of u has the potential to yield good results while having smaller matrix dimensions in the KF.

Having compared results for two different problems using an SVD, we are interested in comparing these results to model problems that did not use an SVD. We achieve this by comparing the results in Figure (3.19) with the results in Table (3.3). The results between the two are indistinguishable which means that using an SVD to reduce the matrix dimensions in the KF has little impact on the success rate. This was expected since the singular values in our SVD converge to zero quickly, and so, even when we take d to be small the resultant principle components contain a lot of the original information despite the reduced matrix sizes.

We now compare Figure (3.20) to Figure (3.21), where the disturbance locations used to deduce the probabilistic results are random. In both figures, the values chosen for T_s have little impact on our success rate. However, when we have six sensors present, the larger the value of d, the better the success rate, which makes sense as more principal components taken from the SVD correlates to more variance being captured in the data.



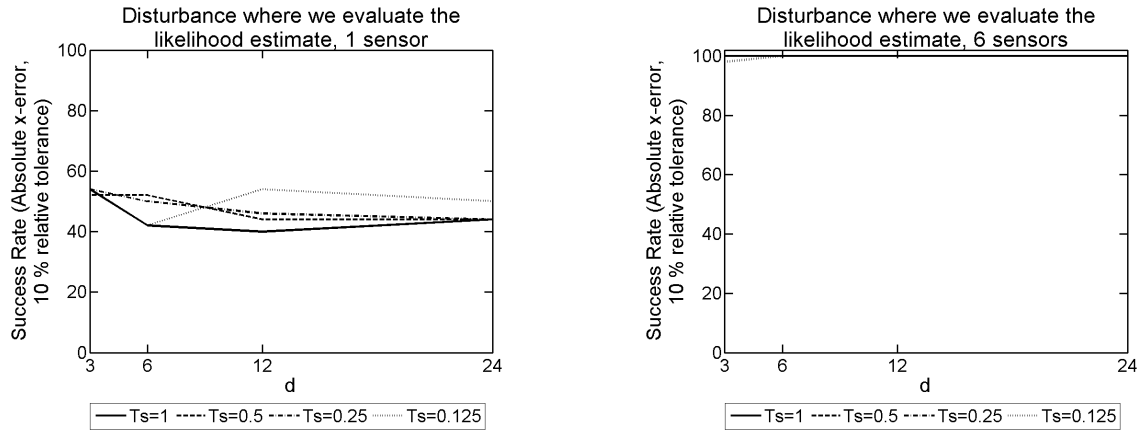
(a) One sensor present.

(b) Six sensors present.

Figure 3.18: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 250$ nodes and $L = 45000$ discrete-time steps. These probabilistic results were produced using 50 disturbance locations where the likelihood function is evaluated.

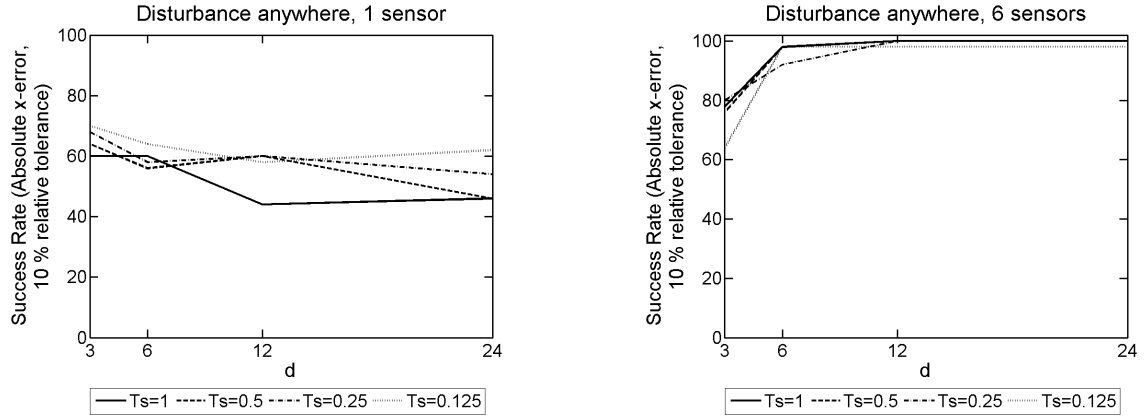
Again, the only difference between the results in both figures is the accuracy of the explicit FDM approximations of u used to form our SVD. Figure (3.20) uses approximations of u from a fine mesh, whereas Figure (3.21) uses them from a coarse mesh. We observe that in both cases, when a single sensor is present, neither have good success rates. However, when we have six sensors present, both have excellent success

rates.



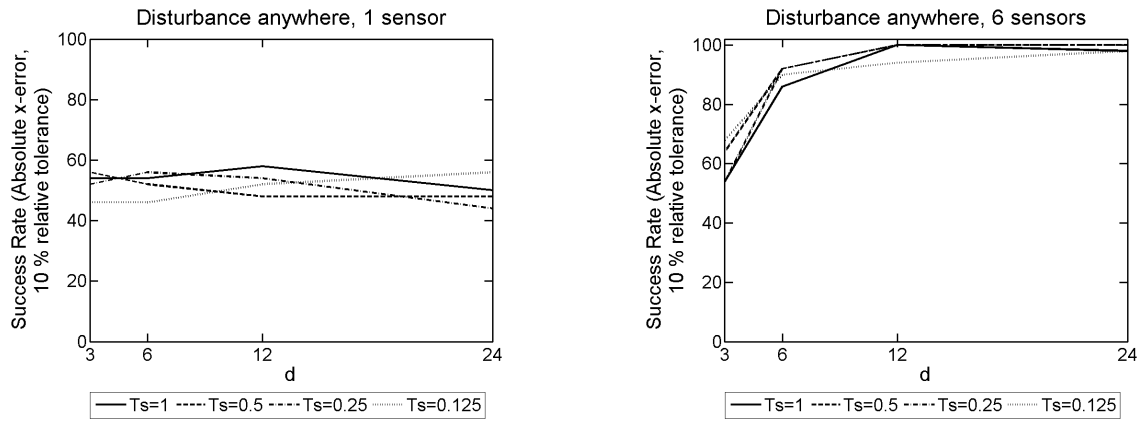
(a) One sensor present. **(b)** Six sensors present.

Figure 3.19: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 50$ nodes and $L = 9000$ discrete-time steps. These probabilistic results were produced using 50 disturbance locations where the likelihood function is evaluated.



(a) One sensor present. **(b)** Six sensors present.

Figure 3.20: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 250$ nodes and $L = 45000$ discrete-time steps. These probabilistic results were produced using 50 random disturbance locations.



(a) One sensor present.

(b) Six sensors present.

Figure 3.21: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 50$ nodes and $L = 9000$ discrete-time steps. These probabilistic results were produced using 50 random disturbance locations.

Unlike what we observed in Figures (3.18) and (3.19) with only a single sensor present, having an SVD formed from explicit FDM approximations of u from a fine mesh does not have a significant effect on the success rate when compared to an SVD created from data originating from a coarse mesh. Meaning, we can use explicit FDM approximations of u from a coarse mesh in (2.16) to construct an SVD and the results when the disturbance location is random are indistinguishable from those obtained using explicit FDM approximations of u from a fine mesh.

We are now interested in comparing results obtained using an SVD, to problems that did not use an SVD. We achieve this by comparing the results in Figure (3.21) with the results in Table (3.4). The results between the two are indistinguishable when $d \geq 6$, meaning that by using an SVD to reduce matrix dimensions in the KF, the success rate has not been significantly affected. As stated previously, this was expected since the singular values in our SVD converge to zero quickly, and so, even when we take d to be small the resultant principle components contain a lot of the original information despite the reduced matrix sizes.

3.4 Run-time optimisations

In this section, we look at how the run-time of our model problem changes when we use the different approaches outlined in this chapter. We are interested in the impact these different approaches have on run-time, as this would be magnified when we consider higher dimensional model problems and higher disturbance frequencies in chapter 4, 5 and 6 of this thesis.

In this chapter, we have used the following three different approaches in our model problems in an attempt to find the location of our disturbance:

1. A minimisation algorithm with an array of initial guesses for the location of our disturbance,

2. An array of initial guesses for the location of our disturbance,

3. An SVD used with an array of initial guesses for the location of our disturbance.

Before we can compare the run-time between these different approaches, we need to consider what effects the run-time of our model problem. After some exploratory investigations, it was apparent that the time consuming part of our model problem is the KF. Therefore we will need to compare the run-time with the same number of sensors present, since this has an impact on the matrix dimensions in the KF. The results presented in Table (3.5) all correspond to our 1D model problem, with 2 sensors present. We chose 2 sensors because we only have results for 1 and 2 sensors for the approach using a minimisation algorithm with a fine mesh to approximate u in our PDE, due to run-time limitations. In addition to this, the mesh density used to approximate u in our PDE will have an impact on the matrix dimensions in the KF, and so, we will consider the run-time for both a fine and coarse mesh for the three different approaches above. Since our guess for the disturbance location has an insignificant impact on the run-time of our model problem, we will compile an average run-time for each approach.

Table (3.5) contains the average run-time for our different approaches. These run-times were computed on the same computer, removing any variance in the run-times between different hardware. For the SVD approach, the run-time average is based over all values of d and T_s since these effect the matrix dimensions in the KF.

Upon inspection of Table (3.5), we can see that using a fine mesh to get the

approximations of u in our PDE results in significantly longer run-times for our model problem. This is expected since a finer mesh results in larger matrices in the KF, and more discrete-time steps for the KF to run through. In addition to this, we observe that using an SVD to reduce the matrix dimensions in the KF has a significant impact on the run-time, irrespective of the mesh density used to approximate u in our PDE. Again, this is expected since we know that the size of the matrices in the KF are the largest contributor to the run-time of our model problem.

	Average run-time (hours)		
	Minimisation algorithm	Array of initial guesses	SVD
Fine Mesh	147.654	0.692	0.001
Coarse Mesh	2.588	0.025	1.74×10^{-5}

Table 3.5: The average run-time, in hours, for our different 1D model problem approaches. In each case, the time presented is an average based on several runs. In all cases, the run-time presented originated from a model problem with two sensors.

In chapters 4, 5 and 6 where we consider higher dimensional model problems, and disturbances with higher frequencies, these run-times will increase because finer meshes will be required. As a result of this, the SVD approach outlined in this chapter will be used due to is effectiveness at predicting the location of the disturbance, and the significantly reduced run-time requirement.

3.5 Summary

In this chapter, we have discovered a lot. We know that we do not need to have sensor traces generated from an explicit FDM approximation of u using a fine mesh. Instead, we can use a coarse mesh which significantly reduces the matrix dimensions in the KF and shortens the run-time of our 1D model problems. In addition to this, when we are using a minimisation algorithm, the number of sensors used is not essential to obtain good success rates.

However, when we do not use a minimisation algorithm, the success rates increase with the number of sensors present for our 1D model problems irrespective of the coarseness of mesh used, when the actual disturbance location is random. That is when we have only

a single sensor, the success rate is poor, whereas when we have six sensors present, the success rate is excellent.

When we use an SVD to further reduce the matrix dimensions in the KF, and subsequently the system RAM and run-time requirements, we get reasonable success rates when a larger quantity of sensors are present. This result is exceptionally significant because when we consider higher-dimensional problems in chapters 5 and 6, and problems with disturbances that have higher frequencies in chapters 4 and 6, this will be a necessity.

In the next chapter, we extend the 1D model problem considered in this chapter to investigate a disturbance with higher frequencies in (1.6). As a result, we use an SVD to reduce the matrix dimensions in the KF.

Chapter 4

1D Model Problem: Higher Frequencies

In this chapter, we look at extending our 1D model problem considered in chapter 3 to consider frequencies of $F = 150$Hz and $F = 300$Hz in (1.6). We outline our model problem, the convergence of the explicit FDM approximations of u in (2.16) for our different forcing functions in (1.6), we detail the noise added to these approximations and outline a schematic for our 1D model problem.

Once everything is outlined, we discuss the results for both frequencies considered which we obtain using the SVD approach explored in the previous chapter.

4.1 Model problem outline

In chapter 1, we outlined the forcing function in (1.6), which attempts to mimic a disturbance caused by CAD. In the same chapter, we noted from research that the real frequency of this disturbance is in the range of $300 - 800$Hz. Previously, we chose a frequency of $F = 25$Hz for simplicity, with the knowledge of a denser mesh requirement to approximate u in (1.1) for higher frequencies.

In the previous chapter, we explored an approach which resulted in significantly smaller matrix dimensions in the KF, and so the run-time for our computations was reduced. We deploy the same approach in this chapter by taking explicit FDM approximations of u in (2.16) from a coarse mesh to form an SVD, whose principal components are used in our modified KF, see (2.127)-(2.132).

4.1.1 Forcing function

In this chapter, we investigate disturbance frequencies of $F = 150$Hz and $F = 300$Hz in (1.6). By changing the disturbance frequency in our forcing function, we had to increase the force of our forcing function. We tried achieving this by increasing the amplitude, A, however, this did not result in good enough results. Instead, we altered the Gaussian spread of our disturbance by altering ϵ in (1.6).

Figure (4.1) shows a snapshot of the Gaussian spread caused by our chosen value of $\epsilon = 2.46 \times 10^{-5}$ in (1.6) when $F = 150$Hz. The black cross represents the location of our disturbance, denoted by x_0, and the grey dots illustrate the 50 equidistant positions we evaluate the likelihood function in (2.107). In Figure (4.1) we consider the worst-case scenario, that is when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible, which in our case is a distance of 0.002. Upon further inspection of (4.1), we observe 85% of the Gaussian spreads peak remains in the worst-case scenario at the nearest position we evaluate the likelihood function in (2.107). The Gaussian spread decays to zero at a distance of 0.01 away from the disturbance location, accounting for 5% of our domain.

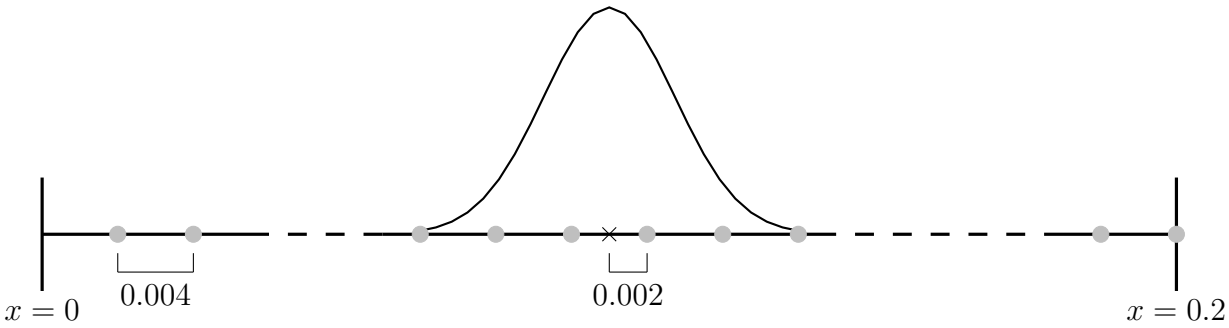


Figure 4.1: A snapshot of the Gaussian spread, given our value of ϵ, around the location of the disturbance denoted by a cross. The disturbance here has a frequency of $F = 150$Hz. The grey dots represent positions at which we evaluate the likelihood function in (2.107).

Figure (4.2) shows a snapshot of the Gaussian spread caused by our chosen value of $\epsilon = 3.98 \times 10^{-4}$ in (1.6) when $F = 300$Hz. The black cross represents the location of our disturbance, denoted by x_0, and the grey dots illustrate the 50 equidistant positions we evaluate the likelihood function in (2.107). In Figure (4.2) we consider the worst-case scenario when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible, which in our case is a

distance of 0.002. Upon further inspection of Figure (4.2), we observe 99% of the Gaussian spreads peak remains in the worst-case scenario at the nearest position we evaluate the likelihood function in (2.107). The Gaussian spread decays to zero at a distance of 0.014 away from the disturbance location, accounting for 7% of our domain.

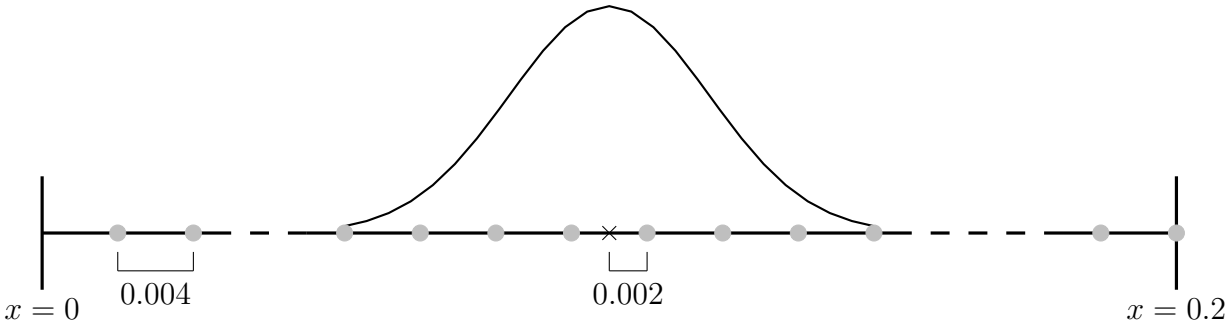


Figure 4.2: A snapshot of the Gaussian spread, given our value of ϵ, around the location of the disturbance denoted by a cross. The disturbance here has a frequency of $F = 300$Hz. The grey dots represent positions at which we evaluate the likelihood function in (2.107).

Since our forcing function in (1.1) has changed, our explicit FDM approximations of u in (2.16) are different. As a result, we re-examine the convergence of the absolute maximum approximation of u across all nodes and discrete-time steps in (2.16) for both disturbance frequencies. To solve (2.16), we need $\mathbf{V}^0$ and $\mathbf{V}^1$, both of which remain the same as in chapter 3.

Table (4.1) illustrates the convergence of the maximum absolute approximation of u across all nodes and discrete-time steps in (2.16) for a disturbance frequency of $F = 150$Hz. The simulation duration is $T = 3$ seconds since we will be using an SVD, see Figure (3.14) for more detail, and the disturbance location is chosen to be $x_0 = 0.05$. Upon inspection of Table (4.1), we select our coarse mesh to have $N = 150$ nodes and $L = 18000$ discrete-time steps which yield a percentage error of 3% from a converged solution. In the previous chapter, when $F = 25$Hz, a percentage error of 22% produced good results. However, when $F = 150$Hz, we did not observe the same rate of success, and so, required a mesh density with a smaller percentage error.

Table (4.2) illustrates the convergence of the maximum absolute approximation of u across all nodes and discrete-time steps in (2.16) for a disturbance frequency of $F = 300$Hz. The simulation duration is $T = 3$ seconds, and the disturbance location is taken to be $x_0 = 0.05$. Upon inspection of Table (4.2), we choose our coarse mesh to have $N = 400$

nodes and $L = 36000$ discrete-time steps which yields a percentage error of 0.28% from a converged solution. Again, when $F = 25$Hz, a percentage error of 22% produced good results. However, when $F = 300$Hz, we did not observe the same rate of success, and so, required a mesh density with a smaller percentage error.

N	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 2000$
50	9000^{-1}	1.021670	20.7467109
100	13500^{-1}	1.258501	2.37518617
150	18000^{-1}	1.244897	3.43047971
200	22500^{-1}	1.282996	0.47505275
250	27000^{-1}	1.264364	1.92037979
300	31500^{-1}	1.261235	2.16310351
350	36000^{-1}	1.272980	1.25201688
400	40500^{-1}	1.301058	0.92605809
450	45000^{-1}	1.274986	1.09640685
500	49500^{-1}	1.278137	0.85197654
550	54000^{-1}	1.279175	0.77145650
600	58500^{-1}	1.285137	0.30897046
650	63000^{-1}	1.286002	0.24187042
700	67500^{-1}	1.291759	0.20471329
750	72000^{-1}	1.294764	0.43781805
800	76500^{-1}	1.293106	0.30920318
850	81000^{-1}	1.288892	0.01768648
900	85500^{-1}	1.285045	0.31610711
950	90000^{-1}	1.283313	0.45046233
1000	94500^{-1}	1.283798	0.41283976
2000	184500^{-1}	1.289120	0

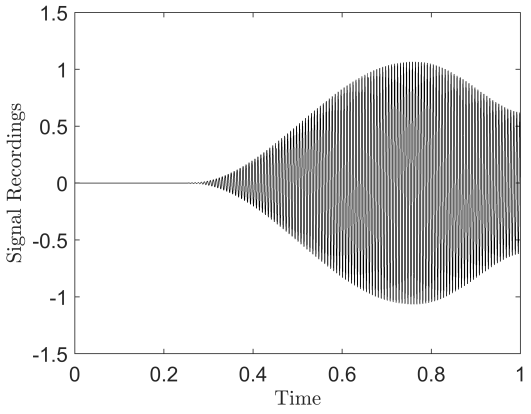
Table 4.1: The convergence of our explicit FDM approximation of u in (1.1) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 150$Hz and a disturbance location at $x_0 = 0.05$.

N	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 4000$
100	9000^{-1}	0.277091	1.851110631
200	18000^{-1}	0.284714	0.849045576
300	27000^{-1}	0.281466	0.301434203
400	36000^{-1}	0.283130	0.287974157
500	45000^{-1}	0.282567	0.088552939
600	54000^{-1}	0.282863	0.193399618
700	63000^{-1}	0.282383	0.023377976
800	72000^{-1}	0.282359	0.014876894
900	81000^{-1}	0.281999	0.112639338
1000	90000^{-1}	0.281899	0.148060514
1100	99000^{-1}	0.283622	0.462246340
1200	108000^{-1}	0.282090	0.080406068
1300	117000^{-1}	0.282957	0.226695523
1400	126000^{-1}	0.282264	0.018773223
1500	135000^{-1}	0.281588	0.258220369
1600	144000^{-1}	0.281850	0.165416890
1700	153000^{-1}	0.282105	0.075092892
1800	162000^{-1}	0.282230	0.030816423
1900	171000^{-1}	0.282340	0.008146870
2000	180000^{-1}	0.282711	0.139559431
4000	360000^{-1}	0.282317	0

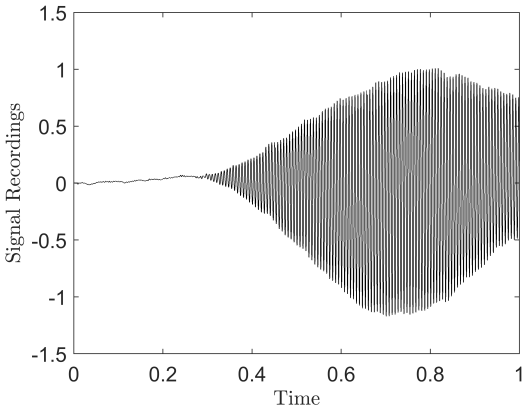
Table 4.2: The convergence of our explicit FDM approximation of u in (1.1) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 300$Hz and a disturbance location at $x_0 = 0.05$.

4.1.2 Added noise

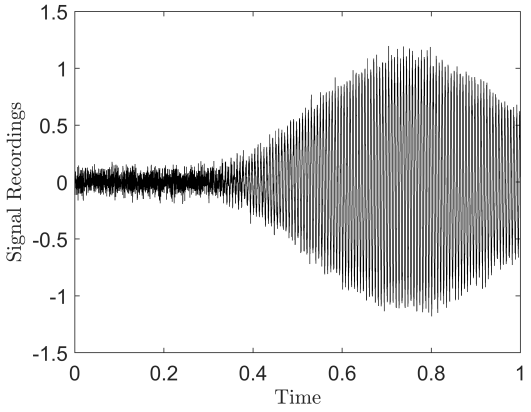
Recall from chapter 1 that we add two different forms of random Gaussian noise to our explicit FDM approximations of u in (2.16), denoted by $\tilde{\mathbf{w}}^n$ and $\tilde{\mathbf{z}}^n$ in the KF. We scale the first by the error associated with the explicit FDM approximations of u in (2.16).



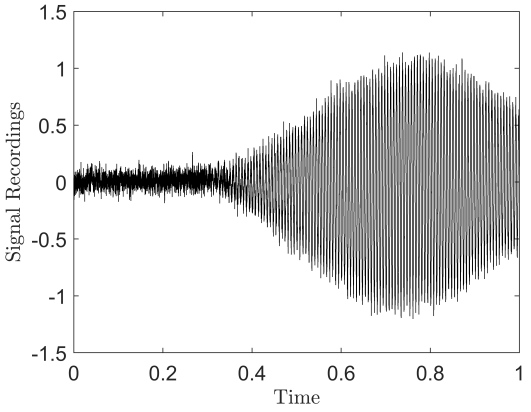
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.

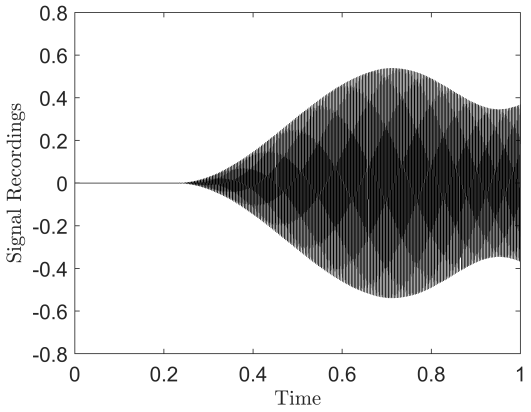


(d) All added noise present.

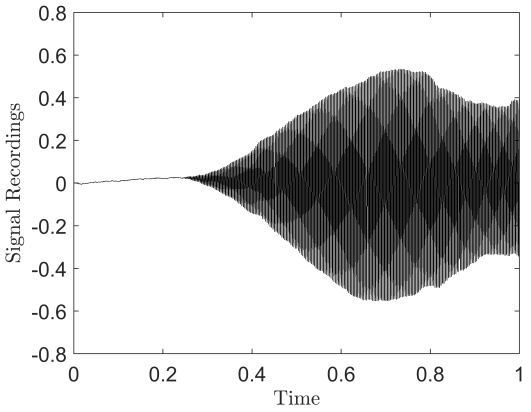
Figure 4.3: Illustration of the noise added to our explicit FDM approximations of u in (2.16) using a single sensor trace at $y := 0.2$, a mesh dimension of $N = 150$ nodes and $L = 18000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 150$Hz and a disturbance location at $x_0 = 0.004$. All figures show only the first second of the sensor traces.

For our 1D model problem considered in this chapter, this error is measured by the combination of the higher-order terms neglected at the interior nodes and the Neumann boundary in (2.7) and (2.14), respectively. The second random Gaussian noise added attempts to mimic inaccuracies recorded by sensors in a real-life scenario, be that ambient noise in the background or errors due to manufacturing defects. Without experimental data, we cannot accurately predict the magnitude of this noise. As a result, we model it by the error associated with the explicit FDM approximation of u in (2.16) but scale it

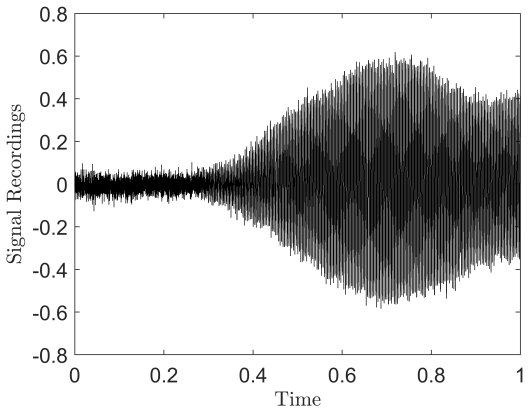
ensuring it is the larger of the two random Gaussian noises added since we would expect this noise to be the dominant of the two. We scale the noise added such that the original explicit FDM approximations of u are still recognisable but significantly obscured.



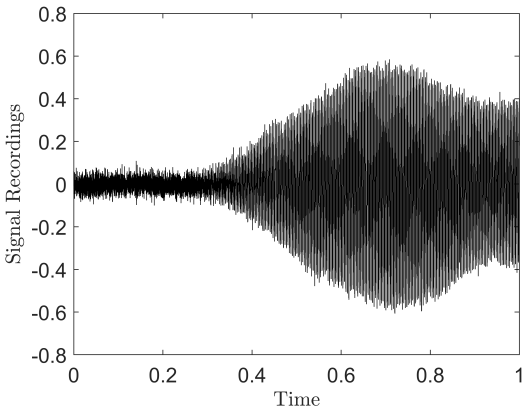
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.



(d) All added noise present.

Figure 4.4: Illustration of the noise added to our explicit FDM approximations of u in (2.16) using a single sensor trace at $y := 0.2$, a mesh dimension of $N = 400$ nodes and $L = 36000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 300$Hz and a disturbance location at $x_0 = 0.004$. All figures show only the first second of the sensor traces.

Figure (4.3) illustrates the effect the added noise has on our explicit FDM approximations of u in (2.16) for our 1D model problem with $F = 150$Hz. These figures show a sensor trace at $x = 0.2$ formed using a mesh density of $N = 150$ nodes with $L = 18000$ discrete-time steps over a simulation duration of $T = 3$ seconds, and a

disturbance location at $x_0 = 0.004$. We scale the random Gaussian noise added to our explicit FDM approximations of u by 6×10^3, see Figure (4.3)(b), and by 2×10^6 in Figure (4.3)(c). In total, the noise added is significant, see Figure (4.3)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (4.3)(a).

Figure (4.4) illustrates the effect the added noise has on our explicit FDM approximations of u in (2.16) for our 1D model problem with $F = 300$Hz. These figures show a sensor trace at $x = 0.2$ formed using a mesh density of $N = 400$ nodes with $L = 36000$ discrete-time steps over a simulation duration of $T = 3$ seconds, and a disturbance location at $x_0 = 0.004$. We scale the random Gaussian noise added to our explicit FDM approximations of u by 1×10^4, see Figure (4.4)(b), and by 4×10^6 in Figure (4.4)(c). In total, the noise added is significant, see Figure (4.4)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (4.4)(a).

4.1.3 Model problem schematic

As discussed previously, we will use the SVD approach seen in chapter 3 to solve our 1D model problems with disturbance frequencies of $F = 150$Hz and $F = 300$Hz in our forcing function to reduce the run-time of our computations. As a result, the sensor locations are the same as seen in Figure (3.3), and the schematic for our 1D model problems considered in this chapter follow the same steps outlined in Figure (3.16).

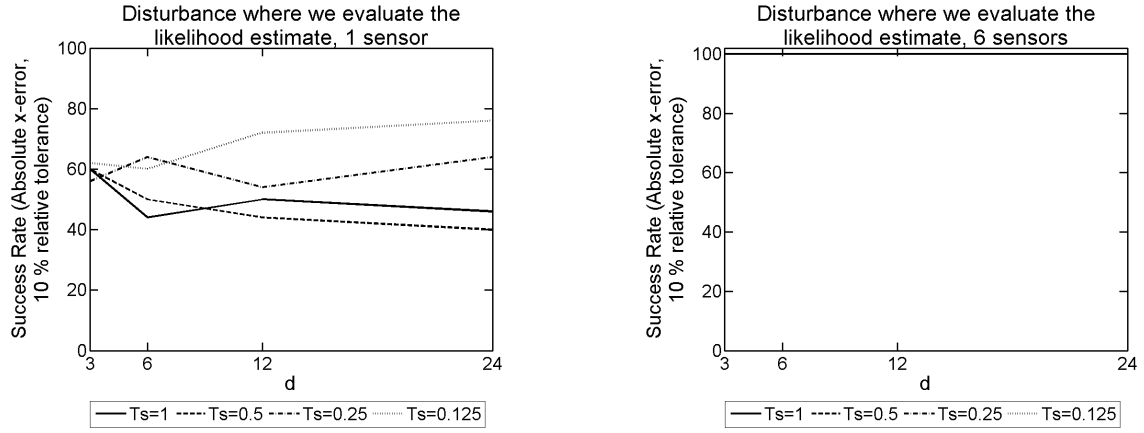
4.2 Results

4.2.1 Disturbance frequency $F = 150$Hz

In this section, we look at the results obtained for our 1D model problem using a disturbance frequency of $F = 150$Hz in (1.6). The full set of results can be found in Appendix B.1, where the results are shown both graphically and in tabular form. Amongst these results, we have one through to six sensors on our domain, with two different types of disturbance locations used to generate the success rates.

Figure (4.5) shows the success rate of our 1D model, given a 10% relative tolerance error, when the disturbance frequency is $F = 150$Hz. The probabilistic results have been deduced by allocating 50 disturbance locations at the same positions we evaluate the

likelihood function in (2.107). By inspecting Figure (4.5), we can see that the values for T_s and d have insufficient positive impact on the success rate. This was also observed in chapter 3, see Figures (3.18) and (3.19), when the disturbance locations are at the same positions, that is where we evaluate the likelihood function. In addition to this, when we compare Figure (4.5)(a) to Figure (4.5)(b), we discover that having six sensors rather than a single sensor result in significantly better success rates.



(a) One sensor present.

(b) Six sensors present.

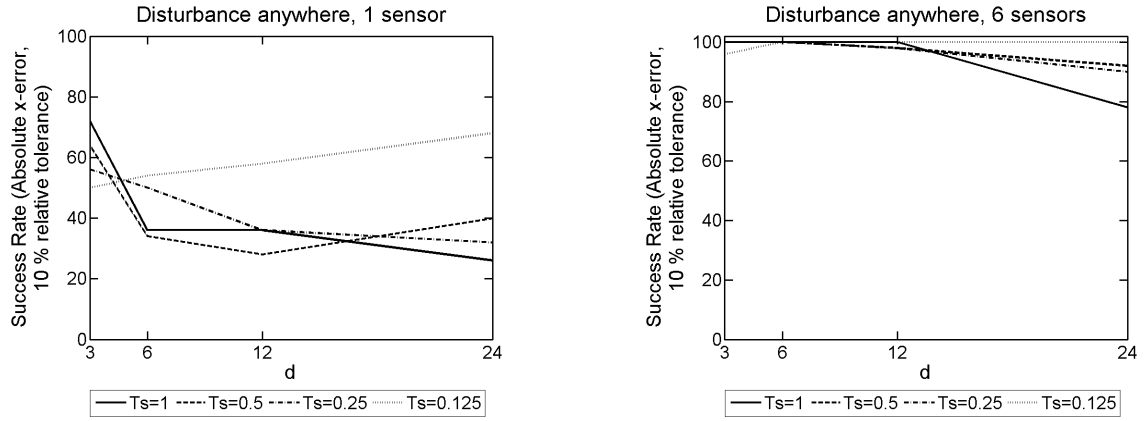
Figure 4.5: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 150$ nodes and $L = 18000$ discrete-time steps for a disturbance frequency of $F = 150$Hz. These probabilistic results were produced using 50 disturbance locations where the likelihood function is evaluated.

Figure (4.6) shows the success rate for our model in a similar manner to Figure (4.5), however, the 50 disturbance locations used to generate our probabilistic results are now random. Comparing Figure (4.5) to Figure (4.6), we can see that the latter achieves a lower success rate, which is expected. The reason for this is due to the likelihood function in (2.107) only being evaluated at certain positions due to the run-time constraints of using a minimisation algorithm, meaning our model cannot access the likelihood our actual disturbance location has of being the disturbance location, which as a result yields some negative results.

Comparing Figure (4.6)(a) to Figure (4.6)(b) we can see that the success rate for a lower value of T_s and d result in better success rates, irrespective of how many sensors are present. A lower value of T_s, resulting in better success rates makes sense because we take

such small values for d. If T_s is large, we have more time steps which if approximating with a small value for d, could result in less variance in the data being maintained. What is harder to explain is the drop in success rate when d is larger. In Figure (4.6)(b), when $d > 12$, a significant reduction in the success rate is observed. A potential reason for this could be due to the columns in $\hat{\mathbf{U}}$ from our SVD, which correspond to our principal components, become linearly dependent.

Despite not knowing precisely why the success rate drops off when using more principal components, the run-time of our computations is shorter when d is lower, meaning the optimal choice when $F = 150$Hz is to take a small number of principal components in our SVD. Again by comparing Figure (4.6)(a) to Figure (4.6)(b), we observe that having more sensors present results in significantly better results as would be expected.



(a) One sensor present.

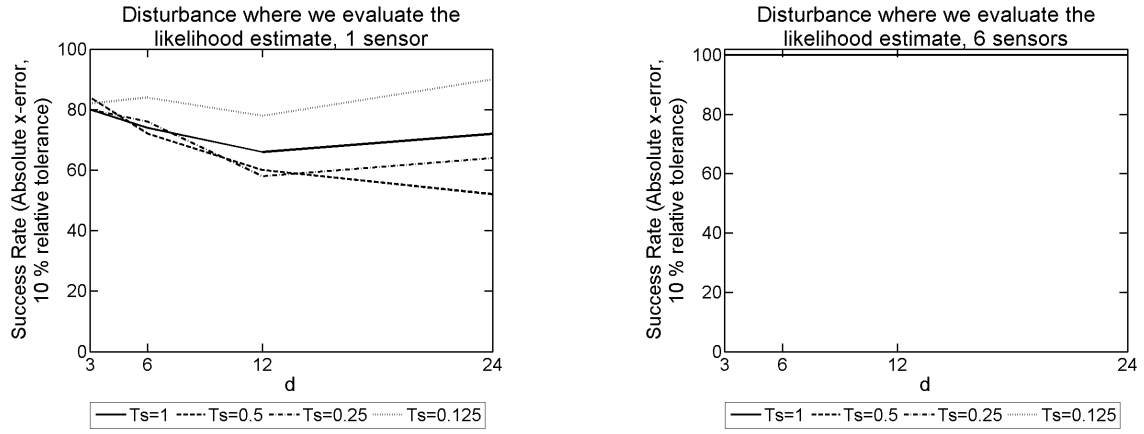
(b) Six sensors present.

Figure 4.6: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 150$ nodes and $L = 18000$ discrete-time steps for a disturbance frequency of $F = 150$Hz. These probabilistic results were produced using 50 random disturbance locations.

4.2.2 Disturbance frequency $F = 300$Hz

In this section, we look at the results obtained for our 1D model problem using a disturbance frequency of $F = 300$Hz in (1.6). The full set of results can be found in Appendix B.2, where the results are shown both graphically and in tabular form. Amongst these results, we have one through to six sensors on our domain, with two different types of disturbance locations used to generate the success rates.

Figure (4.7) shows the success rate of our 1D model, given a 10% relative tolerance error, when the disturbance frequency is $F = 300$Hz.



(a) One sensor present.

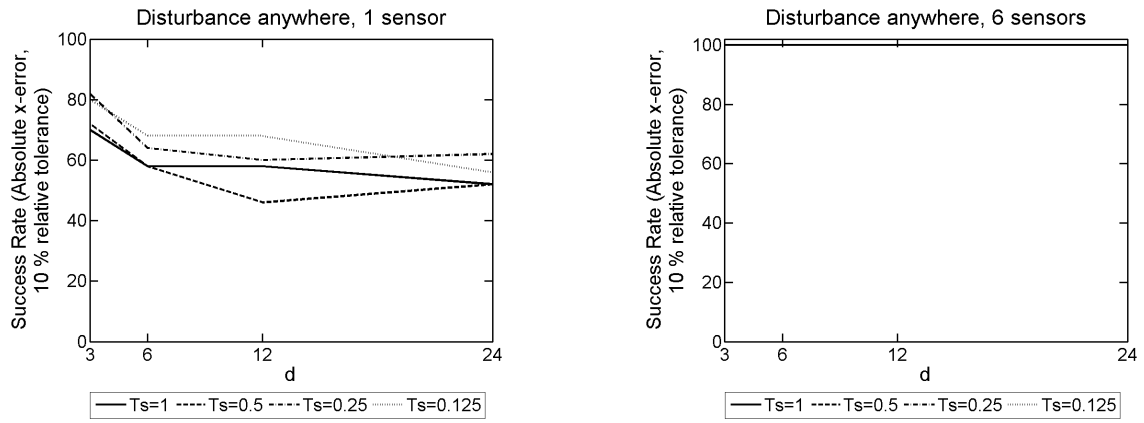
(b) Six sensors present.

Figure 4.7: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 400$ nodes and $L = 36000$ discrete-time steps for a disturbance frequency of $F = 300$Hz. These probabilistic results were produced using 50 disturbance locations where the likelihood function is evaluated.

The probabilistic results have been deduced by allocating 50 disturbance locations at the same positions we evaluate the likelihood function in (2.107). By inspecting Figure (4.7), we can see that the values for T_s and d have little effect on the success rate. We also observed this when $F = 50$Hz and $F = 150$Hz. Moreover, when we compare Figure (4.7)(a) to Figure (4.7)(b), as is expected, having more sensors results in significantly better success rates.

Figure (4.8) shows the success rate of our 1D model, given a 10% relative tolerance error, when the disturbance frequency is $F = 300$Hz. The probabilistic results have been deduced by allocating 50 random disturbance locations.

Comparing Figure (4.8)(a) to Figure (4.8)(b) we can see that the success rate for differing values of T_s and d are insignificant. Again by comparing Figure (4.8)(a) to Figure (4.8)(b), we observe that having more sensors present results in better results as would be expected.



(a) One sensor present.

(b) Six sensors present.

Figure 4.8: The 1D model success rate for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with a dimension of $N = 400$ nodes and $L = 36000$ discrete-time steps for a disturbance frequency of $F = 300$Hz. These probabilistic results were produced using 50 random disturbance locations.

4.3 Summary

In this chapter, we investigated the feasibility of increasing the frequency of our disturbance in the forcing function, see (1.6), to more closely align with the application of this work where the frequency would be in the range of $300 - 800$Hz.

Initially, we had to increase the Gaussian spread of our forcing function in our 1D domain, which was achieved by changing the value of ϵ for both $F = 150$Hz and $F = 300$Hz in (1.6). Having done this, we were able to get reasonable results at these higher frequencies.

Interestingly, we observed that for $F = 150$Hz, the results were better when both T_s and d were smaller. From a run-time perspective, this is great. However, we are not entirely sure why we get worse results here when we have more principal components from our SVD. When we have more sensors present, the success rate of our 1D model with a disturbance frequency of $F = 150$Hz are excellent.

Moving onto the results where we have a disturbance frequency of $F = 300$Hz, the success rate drop off for larger d values was not as prominent. Again, when more sensors were present, the success rate of our 1D model was good.

In the future, it would be a feasible idea to rerun these model problems using a minimisation algorithm to investigate whether the success rate improves when using more principal components from our SVD.

In the next chapter, we look at extending our 1D model problem with a disturbance frequency of $F = 25$Hz to a 2D domain, see Figure (1.3). Different boundary conditions will be considered, the run-time and system RAM limitations will be illustrated and how we attempt to mitigate these drawbacks.

Chapter 5

2D Model Problem

In this chapter, we extend the work in chapter 3 to a 2D domain as illustrated in Figure (1.3). We consider two model problems with different boundary conditions. The first hereby referred to as the model problem with one Neumann boundary condition (1NBC), has the following boundary conditions on the domain defined by $\partial\Omega_{2\mathrm{D}_1}$

1. $\frac{\partial u}{\partial y} := 0$ on $\Gamma_{\mathrm{N}_2} := \big\{(x,y) \in \overline{\Omega}_{2\mathrm{D}} : 0 < x < 0.2,\ y = 0.02\big\}$,
2. $u := 0$ on $\Gamma_{\mathrm{D}_2} := \partial\Omega_{2\mathrm{D}_1} \setminus \overline{\Gamma}_{\mathrm{N}_1}$.

The second problem hereby referred to as the model problem with three Neumann boundary conditions (3NBCs), has the following boundary conditions on the domain defined by $\partial\Omega_{2\mathrm{D}_2}$

1. $\frac{\partial u}{\partial y} := 0$ on $\Gamma_{\mathrm{N}_2} := \big\{(x,y) \in \overline{\Omega}_{2\mathrm{D}} : 0 < x < 0.2,\ y = 0.02\big\}$,
2. $\frac{\partial u}{\partial x} := 0$ on $\Gamma_{\mathrm{N}_3} := \big\{(x,y) \in \overline{\Omega}_{2\mathrm{D}} : x = 0,\ 0 < y < 0.02\big\}$,
3. $\frac{\partial u}{\partial x} := 0$ on $\Gamma_{\mathrm{N}_4} := \big\{(x,y) \in \overline{\Omega}_{2\mathrm{D}} : x = 0.2,\ 0 < y < 0.02\big\}$,
4. $u := 0$ on $\Gamma_{\mathrm{D}_3} := \partial\Omega_{2\mathrm{D}_2} \setminus \big\{\overline{\Gamma_{\mathrm{N}_2} \cup \Gamma_{\mathrm{N}_3} \cup \Gamma_{\mathrm{N}_4}}\big\}$.

For both model problems considered, we outline the forcing function and detail how it differs to what we defined in chapter 3, see (1.6). We determine the initial conditions used to produce our explicit FDM approximations of u in (2.65) for both model problems and demonstrate the convergence of the absolute maximum approximation of u across all nodes and discrete-time steps for both model problems.

We then identify the noise added to these explicit FDM approximations of u and

outline at which nodes in our mesh they will be stored to represent our sensor traces. We outline a schematic for the 2D model problems considered in this chapter, which illustrate the step-by-step process required to obtain a prediction of the disturbance location.

We first considered the approach used in chapter 3 where we use a minimisation algorithm to obtain a more optimal value for x_0 which minimises (2.107), however, due to the system RAM requirements and run-time limitations caused by the matrix dimensions in the KF and the number of discrete-time steps, we devised an alternative approach which we detail more in the results section of this chapter. We discuss the results obtained for both 2D model problems, using our standard KF equations in (2.93)-(2.98) without the use of a minimisation algorithm to find x_0 which minimises (2.107). Additionally, we discuss the results produced from the use of an SVD in both 2D model problems, making use of the modified KF in (2.127)-(2.132).

Lastly, we consider further optimisations to our 2D model problems. As was fairly well documented in the previous two chapters, the run-time and system RAM requirements for our model problems are considerable. Now we are operating in a 2D domain these constraints are significantly more troublesome. As a result, we explore different approaches which could further optimise our 2D model problems by reducing run-times or producing better probabilistic success rates.

5.1 Model outline

5.1.1 Forcing function

We start this section by recalling the forcing function outlined in (1.6), which we will use in (1.1) as our forcing function. The only differences made to this forcing function is that both x and x_0 now contain 2D Cartesian coordinates, and so, we will reference them as $\mathbf{x}$ and $\underline{\mathbf{x}}_0$, respectively.

The amplitude, denoted by A in (1.6) remains the same to ensure (2.94) in the KF is invertible. The frequency of our disturbance for the 2D model problems considered in this chapter will be $F = 25$Hz, which, like in chapter 3, was chosen for simplicity.

We choose a different value for ϵ in our forcing function since the distance in our worst-case scenario has changed. Recall that the worst-case scenario is when the distance between the actual disturbance location and a position we evaluate the likelihood function

in (2.107) is as larger as possible. In 1D, this distance was 0.002, whereas in 2D this distance is 0.0028284.

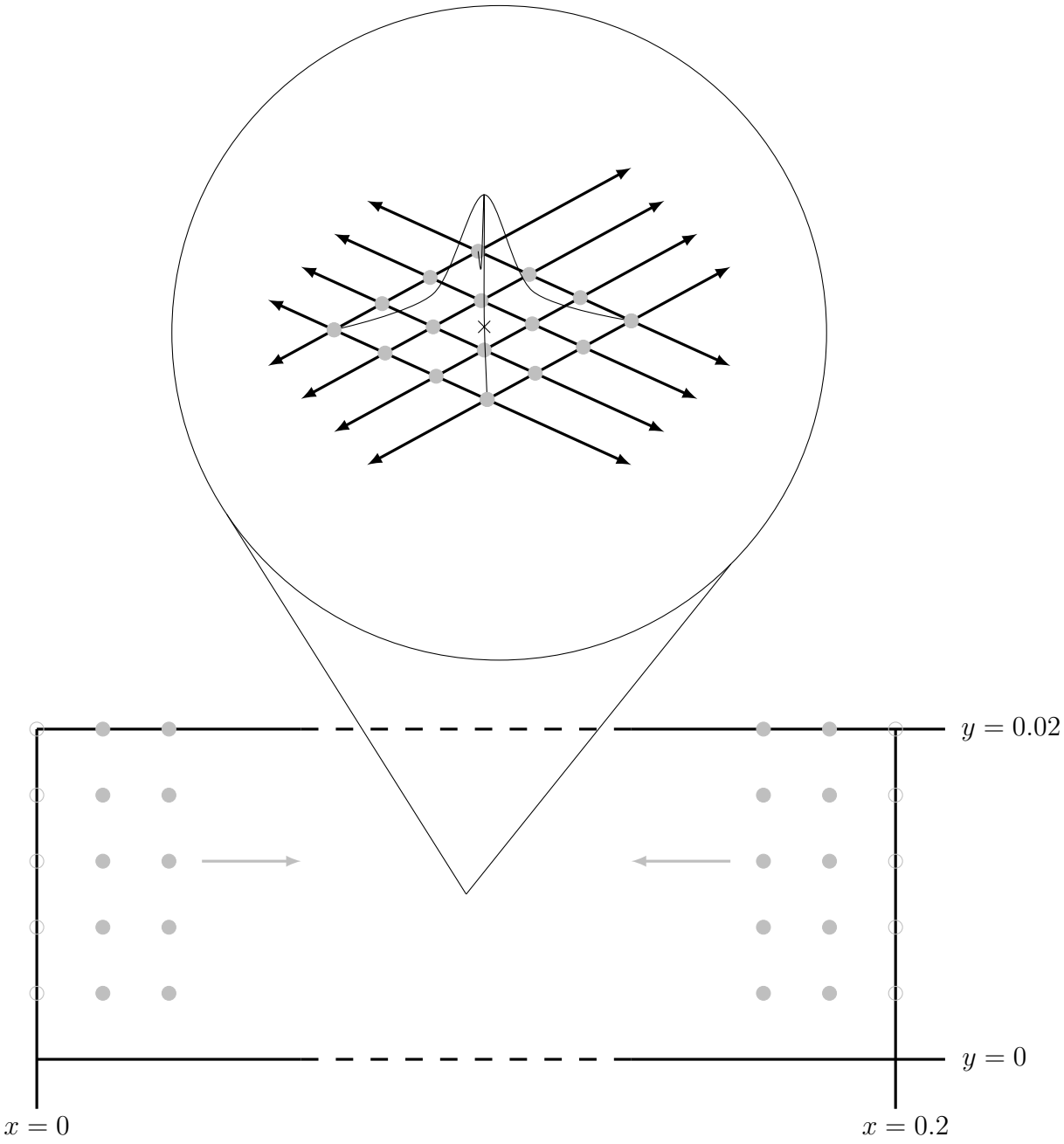


Figure 5.1: A snapshot of the Gaussian spread, given our value of ϵ, around the location of the disturbance in our 2D model problems, denoted by a cross. The disturbance here has a frequency of $F = 25$Hz, the grey dots represent where we evaluate the likelihood function in (2.107) for which there are 245 or 255 in total for our 1NBC and 3NBCs model problems, respectively.

In chapter 3, we chose our value for ϵ in (1.6) by experimenting with its value to get a range of different Gaussian spreads, resulting in a percentage of the peak remaining in the worst-case scenario of between 1% to 98%, see Figure (3.5). However, in 2D, this would have been time consuming due to the run-time limitations of our 2D model problem, and so instead, we investigate the probabilistic success rates for three different values of ϵ for both 2D model problems. In the results section of this chapter, we take $\epsilon = 4.62 \times 10^{-5}$.

Having chosen our value for ϵ in (1.6), we look at the Gaussian spread around a disturbance location in the worst-case scenario. Figure (5.1) shows a snapshot of the Gaussian spread caused by our chosen value of ϵ in (1.6) in the worst-case scenario. The black cross represents the location of our disturbance, denoted by $\underline{\mathbf{x}}_0$, and the grey dots illustrate the positions where we evaluate the likelihood function in (2.107). At the nearest grey dot to the disturbance location in the worst-case scenario, only 84% of the peak is remaining. At the next node along, a distance of 0.0084852 from the disturbance location, there is only 21% of the peak disturbance remaining. This is good as we want the disturbance to be localised in our 2D domain, enabling our 2D model problems to predict the location of our disturbance.

Having defined our forcing function in (1.6), to be used in (1.1), we want to investigate the convergence of our explicit FDM approximations of u in (2.65) for both 2D model problems considered. To do this, we need to know both $\mathbf{V}^0$ and $\mathbf{V}^1$ in (2.65). The initial conditions of u for (1.1) are $u(x_i, y_j, t_0) := 0$ and $u_t(x_i, y_j, t_0) := 0$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$. As a result, we know $\mathbf{V}^0 = \mathbf{0}$. However, we do not know $\mathbf{V}^1$. Using both initial conditions of u, and Taylor series, we can approximate $\mathbf{V}^1$ to get

$$u(x_i, y_j, t_{n+1}) = u(x_i, y_j, t_n) + \Delta t u_t(x_i, y_j, t_n) + \mathcal{O}(\Delta t^2)$$

and subsequently, when $n = 0$, we substitute in both initial conditions of u to get

$$u(x_i, y_j, t_1) = \mathcal{O}(\Delta t^2).$$

Neglecting the higher-order terms, and recalling that $\mathbf{V}^n \approx u(x_i, y_j, t_n)$ for all $i \in \{1, 2, ..., N+1\}$ and $j \in \{1, 2, ..., M+1\}$, we get the second initial condition required in (2.65) to be $\mathbf{V}^1 \approx \mathbf{0}$.

Using these initial conditions in (2.65), we can look at the convergence of the maximum absolute explicit FDM approximations of u across all nodes and discrete-time steps for different mesh densities in both space and time. Tables (5.1) and (5.2) illustrate the results

obtained at different mesh densities for our 2D model problems with 1NBC and 3NBCs, respectively. Upon inspection of both tables, we decide to use a mesh with $N = 50$ by $M = 5$ nodes and $L = 3000$ discrete-time steps to generate our sensor traces. In this chapter, we do not consider a fine mesh as we did in chapter 3. The reason for this is due to the large matrix dimensions in the KF, caused by a fine mesh, and we have obtained results in chapter 3 that show the use of a fine mesh to generate our sensor traces is not a requirement for reasonable success rates.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 800$
50	5	3000^{-1}	92.147693392515436	7.067449891
100	10	6000^{-1}	87.419443007890507	1.573642152
150	15	9000^{-1}	86.470243914577964	0.470756961
200	20	12000^{-1}	86.265155880852731	0.232462849
250	25	15000^{-1}	86.204612373469388	0.162116662
300	30	18000^{-1}	86.172658600827077	0.124989211
350	35	21000^{-1}	86.148657407330290	0.097101952
400	40	24000^{-1}	86.124993465234951	0.069608549
800	80	48000^{-1}	86.065086528491435	0

Table 5.1: 2D model problem with 1NBC: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

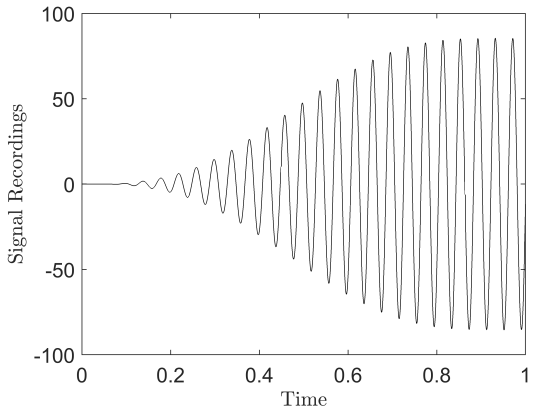
N	M	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 800$
50	5	3000^{-1}	51.770847386036735	6.622289335
100	10	6000^{-1}	54.450557430399805	1.788967073
150	15	9000^{-1}	54.954846754390246	0.879393732
200	20	12000^{-1}	55.156295657551482	0.516045664
250	25	15000^{-1}	55.287654922143666	0.279116426
300	30	18000^{-1}	55.349491365873995	0.167583665
350	35	21000^{-1}	55.381101306400623	0.110569650
400	40	24000^{-1}	55.397960194404696	0.080161719
800	80	48000^{-1}	55.442403778236844	0

Table 5.2: 2D model problem with 3NBCs: The convergence of our explicit FDM approximation of u in (2.65), for a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

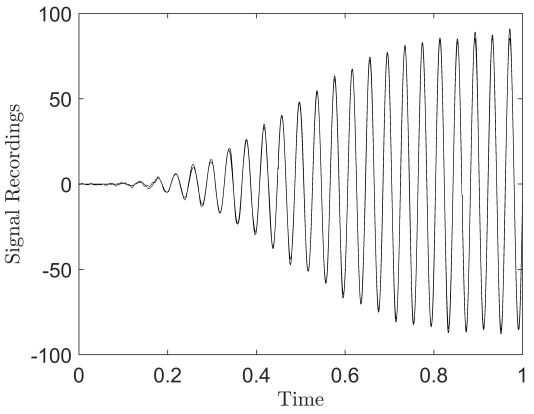
5.1.2 Added noise

Recall from chapter 1 that we add two different forms of random Gaussian noise to our explicit FDM approximations of u in (2.65), denoted by $\tilde{\mathbf{w}}_n$ and $\tilde{\mathbf{z}}_n$ in the KF. We scale the first by the error associated with the explicit FDM approximations of u in (2.65). For our 2D model problem with 1NBC considered in this chapter, this error is measured by the combination of the higher-order terms neglected at the interior nodes and the single Neumann boundary in (2.28) and (2.33), respectively. For our 2D model problem with 3NBCs, this error is measured by the combination of the higher-order terms neglected at the interior nodes in (2.28), and the three Neumann boundaries in (2.33), (2.39), (2.47), (2.53) and (2.62). The second random Gaussian noise added attempts to mimic inaccuracies recorded by sensors in a real-life scenario, be that ambient noise in the background or errors due to manufacturing defects. Without experimental data, we cannot accurately predict the magnitude of this noise. As a result, we model it after the error associated with the explicit FDM approximation of u in (2.65) but scale it ensuring it is the larger of the two random Gaussian noises added since we would expect this noise to be the dominant of the two. We scale the noise added to mimic a scenario where the

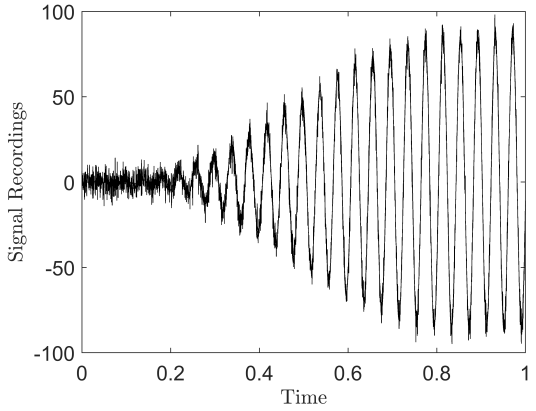
original explicit FDM approximations of u are still recognisable but significantly obscured.



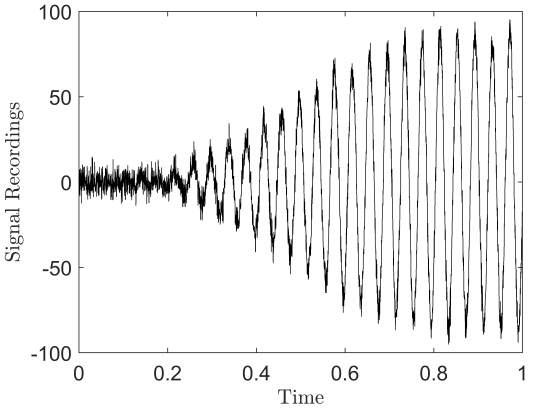
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.

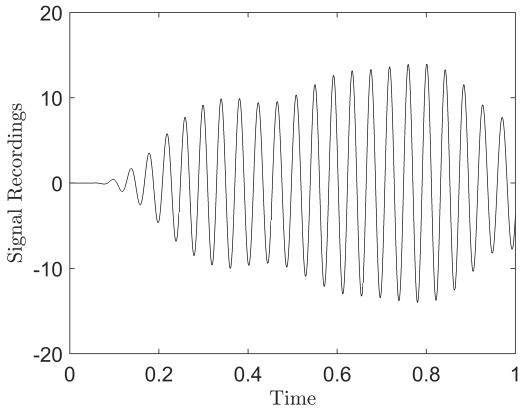


(d) All added noise present.

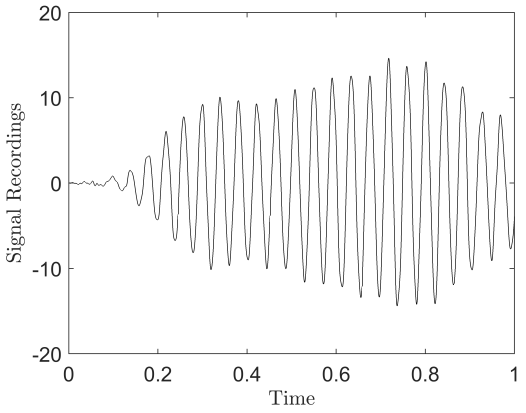
Figure 5.2: 2D model problem with 1NBC: Illustration of the noise added to our explicit FDM approximations of u in (2.65) using a single sensor trace at $(x, y) = (0.1, 0.02)$, a mesh dimension of $N = 50$ by $M = 5$ nodes and $L = 3000$ discrete-time steps, a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

Figure (5.2) illustrates the effect the added noise has on our explicit FDM approximations of u in (2.65) for our 2D model problem with 1NBC. These figures show a sensor trace at $(x, y) = (0.1, 0.02)$ formed using a mesh density of $N = 50$ by $M = 5$ nodes with $L = 3000$ discrete-time steps over a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz, and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$. We scale the random Gaussian noise added to our explicit

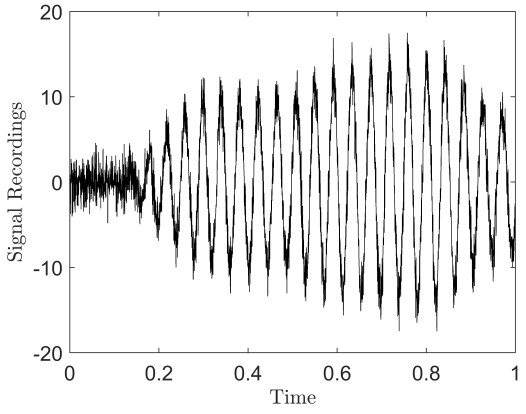
FDM approximations of u, see Figure (5.2)(b) by 8×10^2, and by 3×10^5 in Figure (5.2)(c). In total, the noise added is significant, see Figure (5.2)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (5.2)(a).



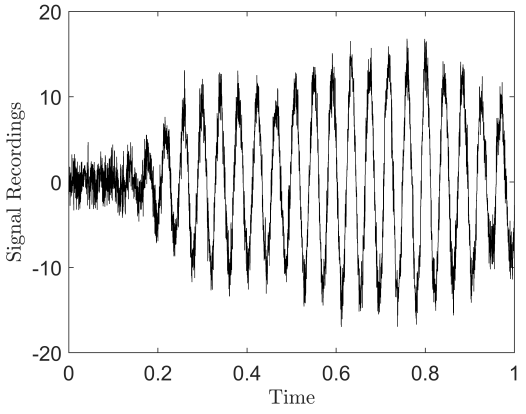
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.



(d) All added noise present.

Figure 5.3: 2D model problem with 3NBCs: Illustration of the noise added to our explicit FDM approximations of u in (2.65) using a single sensor trace at $(x, y) = (0.1, 0.02)$, a mesh dimension of $N = 50$ by $M = 5$ nodes and $L = 3000$ discrete-time steps, a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

Figure (5.3) illustrates the effect the added noise has on our explicit FDM approximations of u in (2.65) for our 2D model problem with 3NBCs. These figures show a sensor trace at $(x, y) = (0.1, 0.02)$ formed using a mesh density of $N = 50$

by $M = 5$ nodes with $L = 3000$ discrete-time steps over a simulation duration of $T = 1$ second, a disturbance frequency of $F = 25$Hz, and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$. We scale the random Gaussian noise added to our explicit FDM approximations of u, see Figure (5.3)(b) by 3×10^2, and by 5×10^4 in Figure (5.3)(c). In total, the noise added is significant, see Figure (5.3)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (5.3)(a).

5.1.3 Model schematic

In this section, we outline the location of the sensors on our 2D domain and construct a schematic which shows the stages required to predict the location of the disturbance for our 2D model problems without using a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107), and without using an SVD to reduce the matrix dimensions in the KF.

Figure (5.4) shows the position of the sensors in our 2D domain depending on the quantity present. This figure attempts to illustrate that the sensors are always along the top of our 2D domain, that is when $y := 0.02$, because the solution of our PDE, denoted by u, for our 2D model problem with 1NBC is zero along the remaining three boundaries. Therefore, we never place a sensor at $y := 0$ or $y := 0.2$ when $y := 0.02$. The sensors are always equidistant from one another, and as observed in Figure (5.6), we do not consider more than 6 sensors due to the small improvement in success rate at the cost of longer run-times and larger system RAM requirements.

In chapter 1, we outlined a schematic in Figure (1.5) which gives a broad approach to the steps involved to obtain a prediction for the disturbance location, denoted by $\underline{\mathbf{x}}_0$. In this section we construct another schematic which goes into more detail for our 2D model problems without using a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107) or an SVD which reduces the matrix dimensions in the KF.

Figure (5.5) shows a schematic portraying the steps involved to obtain a prediction for the disturbance location, denoted by $\underline{\mathbf{x}}_0$, for our 2D model problems. The first step has two different routes. One route involves creating 100 random disturbance locations in our 2D domain. When we initially created these disturbance locations, if the same location was produced twice, we would remove the duplicate and replace with another random disturbance location until we had 100 unique disturbance locations. These random disturbance locations are generated once for all 2D model problems to ensure

the success rates based on probabilistic results are comparable. The second route in our first step involves taking a random subset of 100 positions where the likelihood function in (2.107) is evaluated. For our 2D model problem with 1NBC, there are 245 locations to choose from, whereas for our 2D model problem with 3NBCs, there are 255 locations in total. Again, these disturbance locations are chosen once for each 2D model problem considered, and so the same 100 disturbance locations are used, ensuring the success rates based on probabilistic results are comparable.

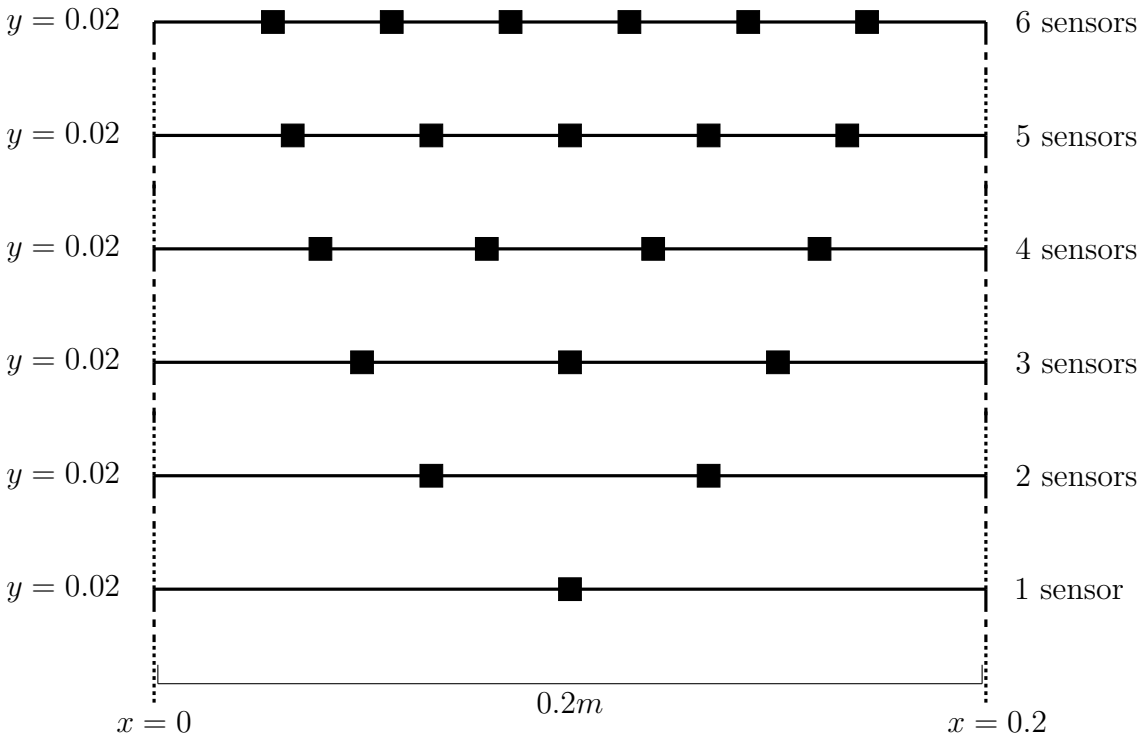


Figure 5.4: Distribution of 1-6 sensors, denoted by black squares, present on our 2D domain.

In step 2, we generate an explicit FDM approximation of u for our acoustic wave equation in (1.1) over a simulation duration of $T = 1$ second. We add two different forms of noise to our approximations of u, which we illustrated in Figure (1.5), and in the previous section of this chapter. Having added noise to our approximations of u, we store these across all discrete-time steps at the nodes in our mesh which correspond to sensor locations, see Figure (5.4).

Moving onto step 3, we run the standard KF in (2.93)-(2.98) over a series of uniformly spaced guesses for $\underline{\mathbf{x}}_0$ and subsequently calculate the likelihood function in (2.107) for each guess. We consider 245 or 255 guesses for $\underline{\mathbf{x}}_0$ in our 2D model problems with 1NBC

and 3NBCs, respectively. Now we have 244 or 255 likelihood estimates, one for each guess of $\underline{\mathbf{x}}_0$, we determine the smallest and deduce the corresponding $\underline{\mathbf{x}}_0$ to be the models prediction of the actual disturbance location used to generate the sensor traces.

In step 4, having completed steps 2-3 in Figure (5.5) for each disturbance location in step 1, we compute probabilistic success rates for an absolute x-error, y-error and Euclidean-error given a percentage of tolerance in each case.

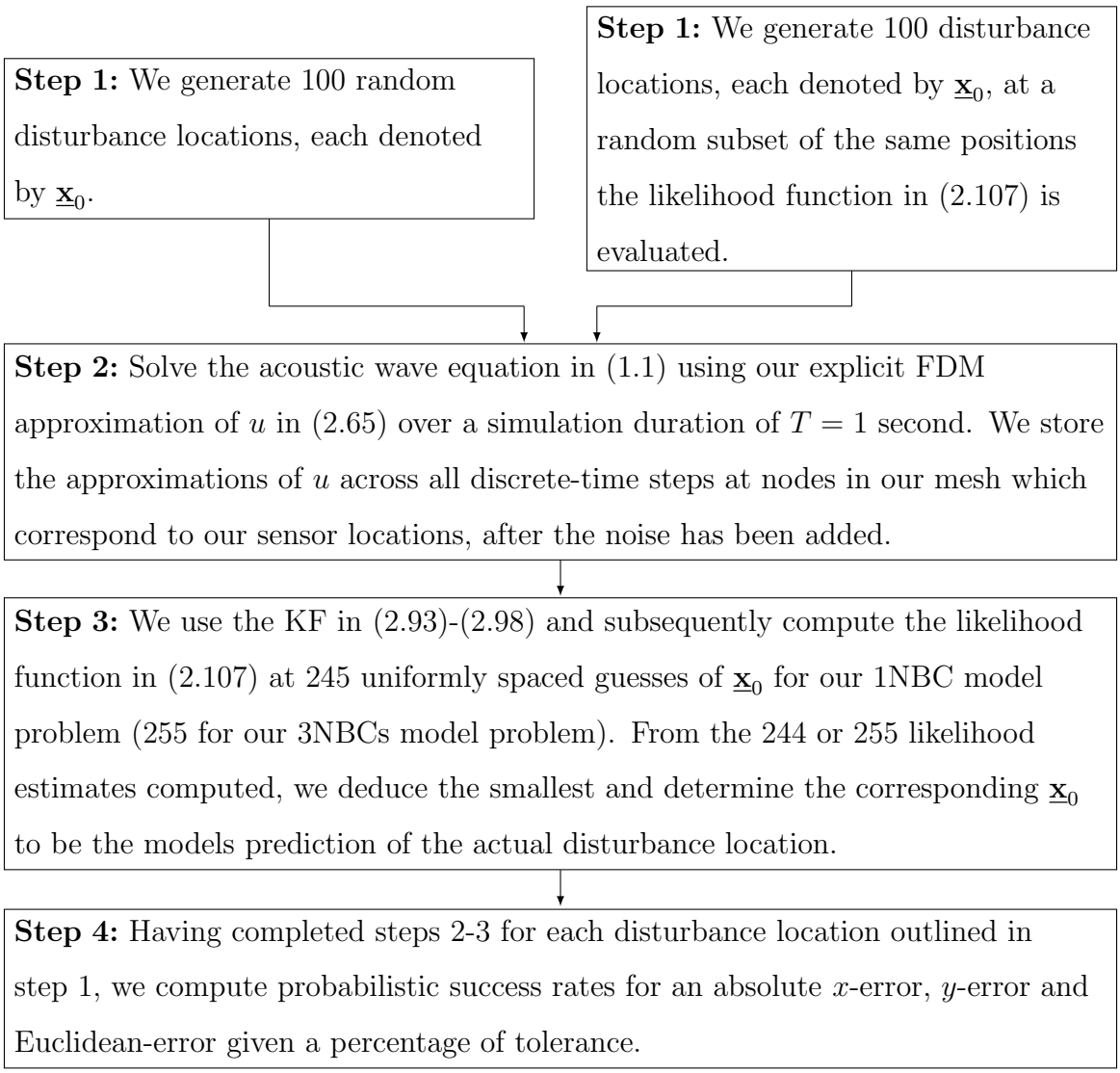


Figure 5.5: A schematic for our 2D model problems which do not use a minimisation algorithm or an SVD.

5.2 Results

In this section we explain our choice for our value of ϵ in (1.6) and how many sensors are enough. We then take a look at the results obtained for our two 2D model problems and discuss why we have not used a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107) for our 2D model problems. We then delve into the results themselves, with the first set corresponding to the use of our standard KF in (2.93)-(2.98), without using a minimisation algorithm. We then look at results produced from the use of an SVD to reduce matrix dimensions in our KF, see the modified KF in (2.127)-(2.132). Finally, we investigate different approaches to further optimise our 2D model problems in an attempt to improve the success rates and decrease the run-time and system RAM requirements.

	Percentage of peak Gaussian spread remaining at a distance illustrated in the worst-case		
Model variations	27%	50%	84%
1NBC, disturbance locations where the likelihood function is evaluated	57	54	55
1NBC, random disturbance locations	20	19	21
3NBCs, disturbance locations where the likelihood function is evaluated	38	53	52
3NBCs, random disturbance locations	13	15	17

Table 5.3: The success rate for an absolute Euclidean-error with a relative tolerance of 10% for our 2D model problems with a sensor placed at $(x, y) = (0.1, 0.02)$. The table shows results for different values of ϵ in our forcing function, resulting in different positive peak percentages remaining at the worst-case scenario distance of 0.0028284. That is where the actual disturbance location is the furthest as is possible from a position we evaluate the likelihood function in (2.107).

Firstly, we investigate our choice for ϵ in our forcing function. Table (5.3) contains the probabilistic success rates given an absolute Euclidean-error with a relative tolerance of 10% with a single sensor present at $(x, y) = (0.1, 0.02)$ for both 2D model problems. In both cases, we consider 100 disturbance locations, either at positions where the likelihood

function is evaluated or at random locations, to obtain probabilistic results.

Upon inspection of Table (5.3), we can see that 84% of the peak Gaussian spread remaining at a distance of 0.0028284 from the actual disturbance location, representing the worst-case, results in the best success rates. Therefore, we take the corresponding value of $\epsilon = 4.62 \times 10^{-5}$ and use this in both 2D model problems considered in this chapter.

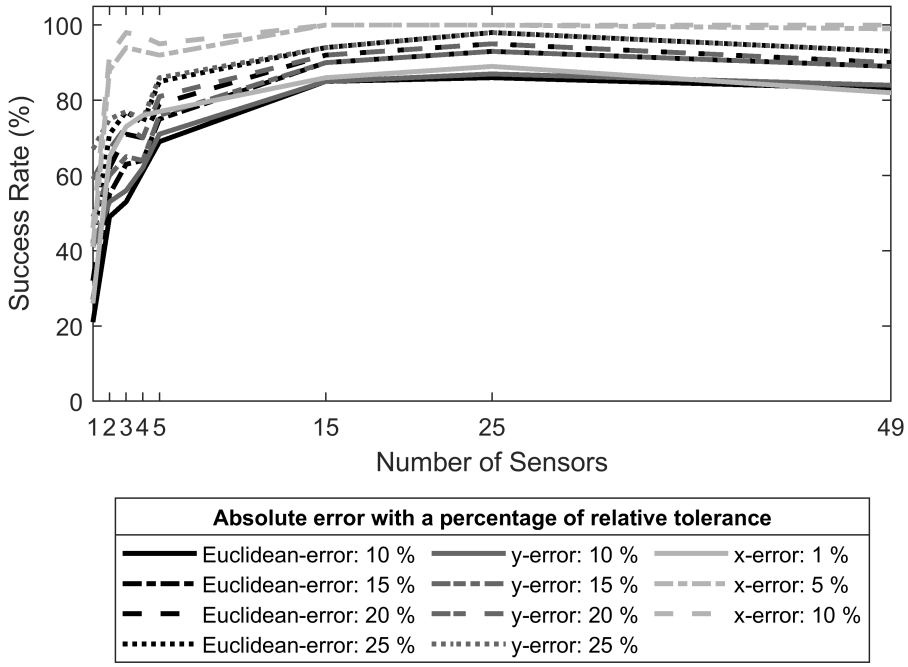


Figure 5.6: The success rates of our 2D model problem with 1NBC for a varying number of sensors on our 2D domain. These probabilistic results were obtained using 100 random disturbance locations.

Figure (5.6) contains the probabilistic success rates of our 2D model problem with 1NBC for an array of absolute errors given different acceptable relative tolerance percentages. Upon inspection of Figure (5.6), we can see that a larger quantity of sensors on our 2D domain results in higher success rates. However, as mentioned previously, more sensors correlate to longer run-times and more system RAM being required. In addition to this, in a real-life scenario, a trained professional would not place 49 sensors onto a human thorax. Therefore, like in chapter 3, we will run our 2D model problems for an array of sensors starting with 1 and going up to 6, where we observe good success rates

in Figure (5.6).

The success rates in this chapter, like in the previous chapters, are produced using an array of disturbance locations and are defined as being successful based on absolute errors given different relative tolerances. Now we are working spatially in 2D we have an absolute y-error and Euclidean-error. As previously mentioned, the results in this chapter have been obtained without the use of a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107). We did attempt to use a minimisation algorithm. However, the run-time requirements were too large to have 245 or 255 initial guesses at our disturbance location for the two different 2D model problems considered in this chapter. Upon inspection of Figures (5.7)(c) and (5.8)(c), we can see that the likelihood function surface from the y-perspective is relatively flat, illustrating why taking only a single initial guess for our disturbance location could result in bad success rates.

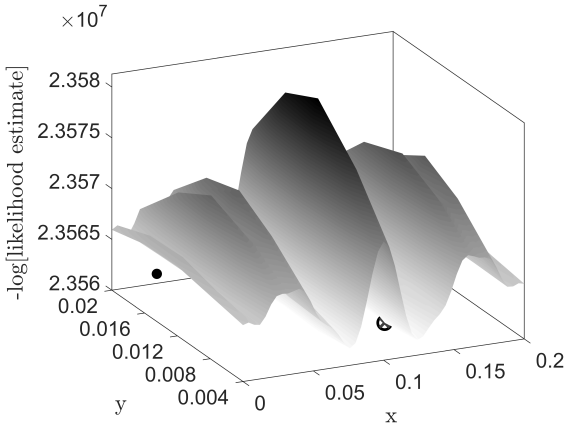
Therefore, none of the results in this chapter were obtained using a minimisation algorithm. Instead, we evaluate the likelihood function in (2.107) at 245 or 255 uniformly spaced positions for our 2D model problems with 1NBC or 3NBCs, respectively. Therefore, in the worst-case scenario, the best outcome our 2D model problems could deduce would be to predict the actual disturbance location to be at the closest position where the likelihood function is evaluated. However, even in this case, we would attain an unavoidable x-error of 1.41% and a y-error and Euclidean-error of 14.1%. As a result, we have used higher acceptable relative tolerance percentages for both our y and Euclidean-errors.

5.2.1 Without using a minimisation algorithm

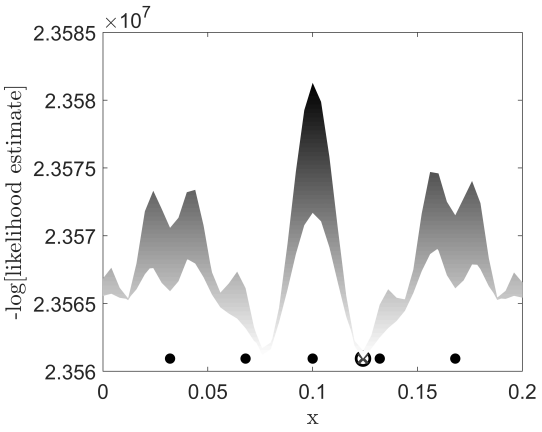
In this section, we look at the results obtained for both 2D model problems without using a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107) or an SVD to reduce the matrix dimensions in the KF.

Figures (5.7) and (5.8) show the likelihood function evaluated using (2.107) plotted against the corresponding disturbance location in our 2D domain, which produced this estimate. Both figures show three viewing angles of the same surface plot enabling the distinction between the models prediction for the x and y coordinates in the disturbance location, $\underline{\mathbf{x}}_0$. Both figures have been produced using our 2D model problem with 3NBCs. Figure (5.7) has a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.124, 0.008)$, which is at a

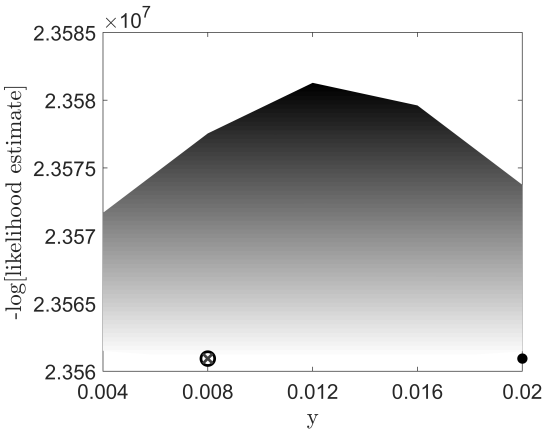
position the likelihood function in (2.107) is evaluated, whereas Figure (5.8) has a random disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.106811, 0.007611)$.



(a) General view of the likelihood surface.

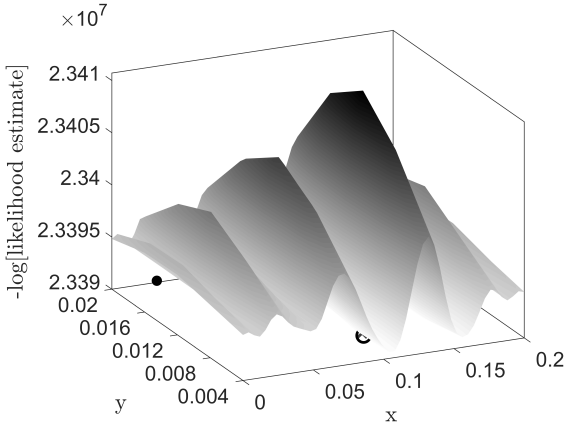


(b) x-direction perspective of the likelihood surface.

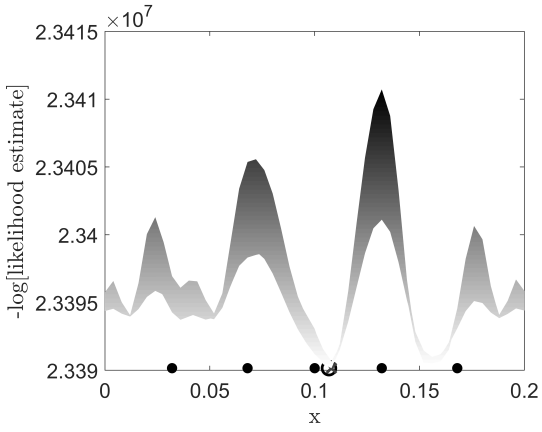


(c) y-direction perspective of the likelihood surface.

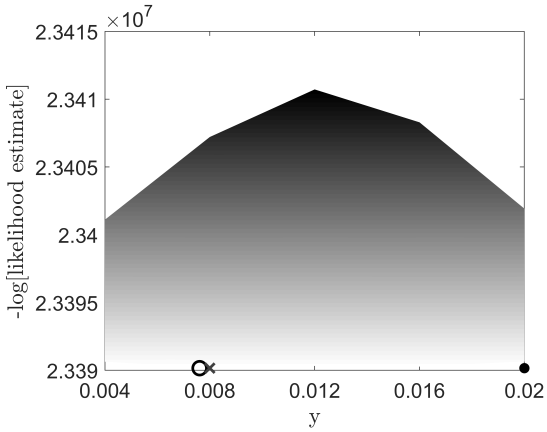
Figure 5.7: The likelihood estimates across our 2D domain for a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.124, 0.008)$ with a frequency of $F = 25$Hz. Data from our explicit FDM approximations of u were collected at five sensors, shown as solid black dots along $y := 0.02$. The actual disturbance location is shown as a hollow black circle, with the models prediction of this location denoted by a grey cross.



(a) General view of the likelihood surface.



(b) x-direction perspective of the likelihood surface.



(c) y-direction perspective of the likelihood surface.

Figure 5.8: The likelihood estimates across our 2D domain for a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.106811, 0.007611)$ with a frequency of $F = 25$Hz. Data from our explicit FDM approximations of u were collected at five sensors, shown as solid black dots along $y = 0.02$. The actual disturbance location is shown as a hollow black circle, with the models prediction of this location denoted by a grey cross.

In both cases, the models prediction of the actual disturbance location is excellent. In the latter case, in Figure (5.8), the prediction is not perfect. However, unless the random disturbance location is at a position the likelihood function in (2.107) is evaluated at then it is impossible to obtain a perfect prediction.

Having seen two examples where our model has successfully predicted the disturbance location, we now look at an array of probabilistic results from 100 different disturbance locations. The full set of results in both graphical and tabular form can be found in

Appendix C.1 and C.2 for our 2D model problems with 1NBC and 3NBCs, respectively. Amongst these results, we have one through to five sensors on the surface of our domain, with two different types of disturbance locations used to generate the success rates. Additionally, we have results for three different absolute errors with a range of acceptable relative tolerances.

Figures (5.9) and (5.10) display the success rate based on three different absolute errors for a varying quantity of sensors present for our 2D model problems with 1NBC and 3NBCs, respectively.

In both cases, when the disturbance location is positioned where we evaluate the likelihood function in (2.107), and when we have more than a single sensor present, we attain near to a 100% success rate. We observed this in chapter 3 for our 1D model problems as well.

When the disturbance location is randomly set to any location in our 2D domain, both model problems struggle to obtain reasonable success rates when there are fewer sensors present. However, in both cases, when there are more sensors present the success rate given an absolute x-error is very good. This was also observed in chapter 3 for our 1D model problems.

However, upon further inspection of Figures (5.9) and (5.10), we can see that when we have random disturbance locations, even with a high quantity of sensors, the success rate given either an absolute y-error or Euclidean-error is not excellent. Since the Euclidean-error is dependent on both the x and y errors, we can deduce that the y-error negatively affects the Euclidean-error.

There could be many reasons why we exhibit a lower success rate for our absolute y-error. The first being that in the worst-case scenario, as discussed previously, induces an unavoidable error of 14.1% when the disturbance locations are random. In addition to this, upon inspecting both Figures (5.7)(c) and (5.8)(c) as previously discussed, we observe that the likelihood surface from the y-perspective is very flat. This makes the ability to accurately predict a random disturbance location without evaluating the likelihood function at the same location is a significant challenge.

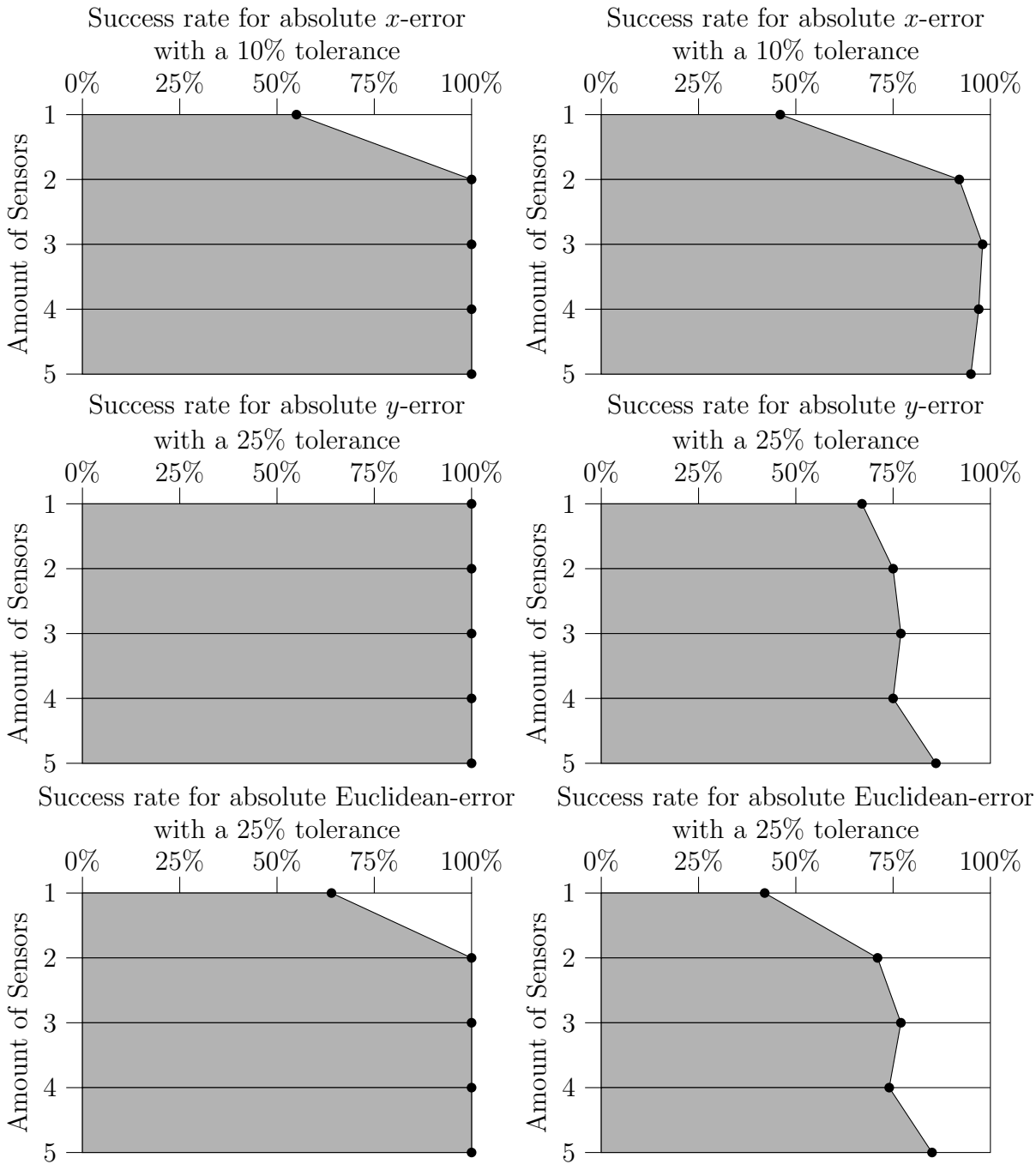


Figure 5.9: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$, $|y\text{-error}|$ and |Euclidean-error| for a varying quantity of sensors present and a disturbance frequency of $F = 25\text{Hz}$. Results on the LHS correspond to 100 random disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

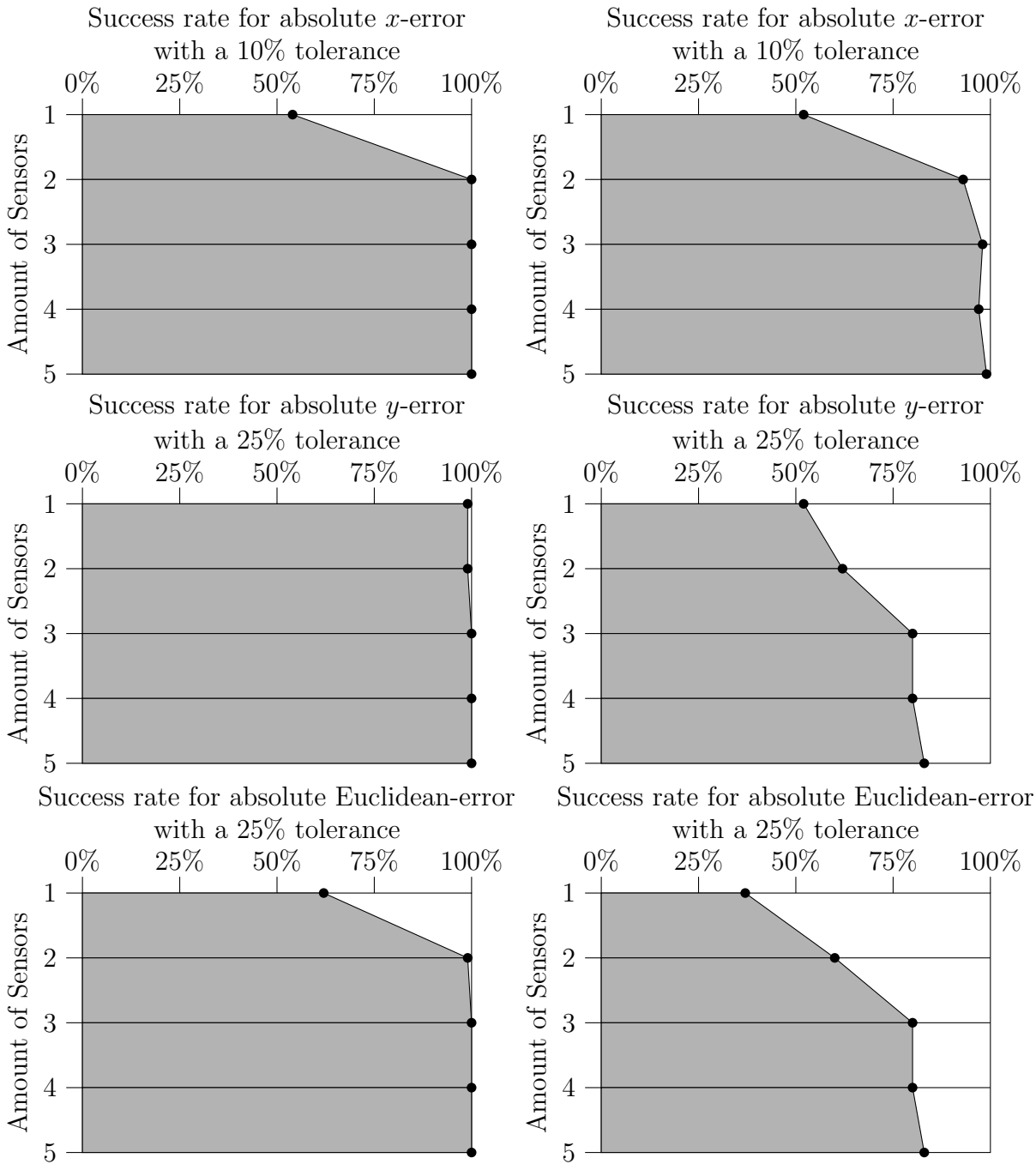


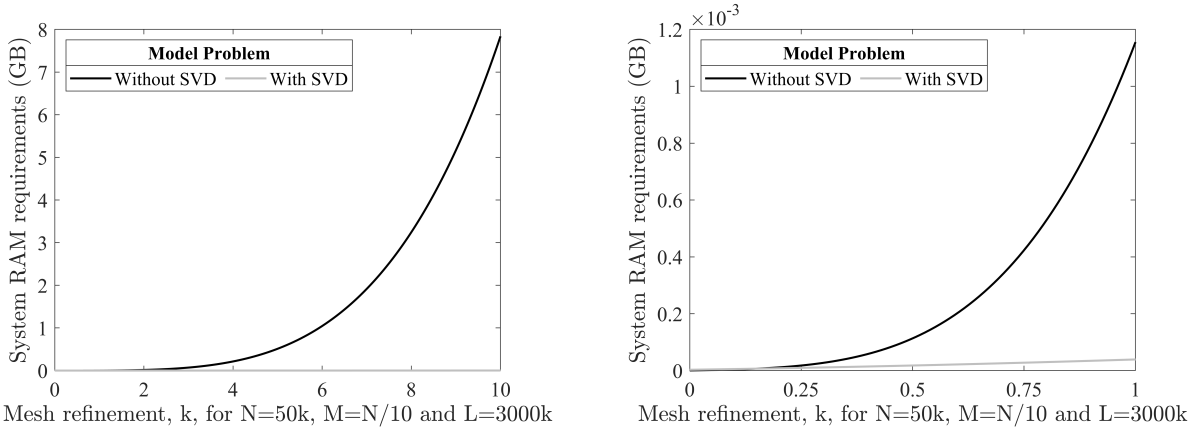
Figure 5.10: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$, $|y\text{-error}|$ and |Euclidean-error| for a varying quantity of sensors present and a disturbance frequency of $F = 25$Hz. Results on the LHS correspond to 100 random disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

5.2.2 Use of an SVD to reduce matrix dimensions in the KF

In this section, we discuss the results attained using d principal components from an SVD formed using our explicit FDM approximations of u in (2.65). As discussed previously, we use an SVD to reduce the size of the matrices in the standard KF, see (2.93)-(2.98), to form our modified KF, see (2.127)-(2.132). The largest matrix in the standard KF is (2.98), with dimensions of $\Big(2(N+1)(M+1)\Big)^2$. The same matrix in the modified KF has dimensions of $(2d)^2$.

We can further illustrate the matrix sizes in both methods by inspecting Figure (5.11). In MATLAB, eight bytes are required for each numerical entry. Figure (5.11) illustrates the total system RAM requirements of the KF for our 2D model problems with, and without an SVD, across an array of different mesh densities.

Upon inspection of Figure (5.11), we can see that the system RAM requirements for the standard KF in 2D are significantly higher than our modified KF, even when the explicit FDM approximations of u originate from a coarse mesh. A smaller requirement of system RAM means we get faster run-times. Therefore, even in this chapter where a coarser mesh is used to form our explicit FDM approximations of u, considerable amounts of time will be saved by using an SVD to reduce the matrix dimensions in the KF.



(a) Full picture for finer mesh refinements.

(b) A subset of mesh refinements.

Figure 5.11: The system RAM requirements for the KF used in our 2D model problems with six sensors present, and $d = 12$ when an SVD is used in our 2D model problems.

Recall from chapter 1 the schematic in Figure (1.5) which gives a broad approach to the steps involved to obtain a prediction for the disturbance location, denoted by $\underline{\mathbf{x}}_0$. We construct another schematic in this chapter, see Figure (5.12), for our 2D model problems

which use an SVD to reduce the matrix dimensions in the KF. In this schematic, we outline a step-by-step process required to predict the location of our disturbance.

In step 1, we have two routes. The first route involves generating 100 random disturbance locations. The second route includes creating 100 disturbance locations from a random subset of the positions we evaluate the likelihood function in (2.107). In both cases, if a duplicate arises, it is replaced until we have 100 unique disturbance locations. We create these 100 disturbance locations once, which get reused every time we want to produce probabilistic success rates for our 2D model problems.

In step 2 we solve the acoustic wave equation in (1.1) using our explicit FDM approximations of u in (2.65) for our 2D model problems with 1NBC and 3NBCs, separately, over a simulation duration of $T = 2$ seconds. Recall from chapter 3, that when we try to form an SVD from explicit FDM approximations of u, we extend the total simulation duration to $T = 3$ seconds, see Figure (3.14) for more detail on the simulation duration used. We store the approximations of u at every node in our mesh across all discrete-time steps for our simulation between $T = 0$ and $T = 2$ seconds.

Step 3 involves using the explicit FDM approximations of u over our entire mesh for a duration of time denoted by T_s, see Figure (3.14), to form our SVD. To reduce the matrix dimensions in the KF, we take the first d columns from the matrix containing the left singular vectors in (2.108). These first d columns represent our principal components, corresponding to the d largest singular values in our SVD.

In step 4, we use these d principal components to run the wave equation for the third second in our simulation to produce approximations of u from the reduced system. We then use these approximations of u from the reduced system to approximate the sensor traces over the third second of our simulation for the original model. Random noise is then added to the approximate sensor traces.

These approximated sensor traces are then used in the KF in step 5. We run the KF in (2.127)-(2.132) for 245 or 255 equally spaced guesses for $\underline{\mathbf{x}}_0$ in our 2D model problems with 1NBC and 3NBCs, respectively. Using time series data generated by the KF, we compute a likelihood function using (2.107) for each guess of $\underline{\mathbf{x}}_0$. Having calculated an array of likelihood estimates for each guess of $\underline{\mathbf{x}}_0$, we compute the smallest and conclude that the corresponding disturbance location is the models prediction of the true disturbance location used to generate our sensor traces.

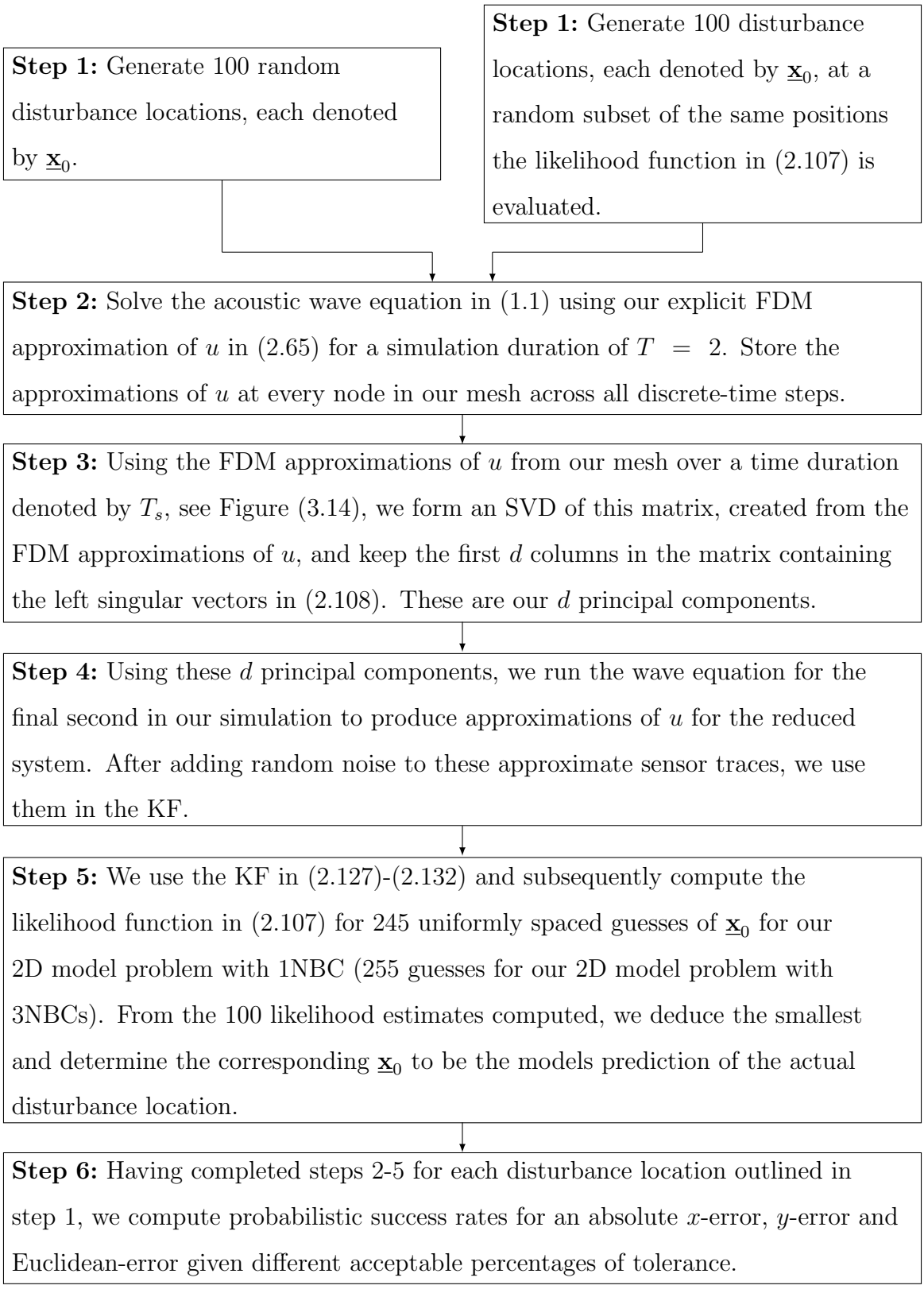


Figure 5.12: A schematic for our 2D model problems that use an SVD.

Finally, in step 6, we compute probabilistic success rates given different acceptable relative percentages of tolerance for an absolute x-error, y-error and Euclidean-error. We

are able to do this by following steps 2-5 for each disturbance location defined in step 1.

Since our simulation duration has changed, we need to redo the convergence of the maximum absolute approximation of u in (2.65) at every node in our mesh across all discrete-time steps for the simulation duration of $T = 3$ seconds. Using the same initial conditions in (2.65) outlined previously in this chapter, we can obtain the convergence of the maximum absolute approximation of u for an array of different mesh refinements for both 2D model problems.

Tables (5.4) and (5.5) contain the convergence of the maximum absolute approximation of u in (2.65) for our 2D model problems with 1NBC and 3NBCs, respectively.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from N = 1600
50	5	9000^{-1}	92.754787850588556	7.93832
100	10	18000^{-1}	87.584011743121181	13.0705
150	15	27000^{-1}	90.233329351909475	10.4409
200	20	36000^{-1}	95.137110665665844	5.57380
250	25	45000^{-1}	97.262336991851129	3.46446
300	30	54000^{-1}	98.359187469466534	2.37580
350	35	63000^{-1}	98.992292324084474	1.74743
400	40	72000^{-1}	99.416643183206560	1.32625
450	45	81000^{-1}	99.742651418312064	1.00268
500	50	90000^{-1}	99.960968671509008	0.78599
550	55	99000^{-1}	100.11152606117142	0.63656
600	60	108000^{-1}	100.21769112026581	0.53119
650	65	117000^{-1}	100.30099191998644	0.44851
700	70	126000^{-1}	100.38683413291274	0.36331
750	75	135000^{-1}	100.44992712496935	0.30068
800	80	144000^{-1}	100.49650211403481	0.25446
1600	160	288000^{-1}	100.75287585092920	0

Table 5.4: 2D model problem with 1NBC: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 25$Hz, and a disturbance location of $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

Upon inspection of both tables, we decide to use a mesh with $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps for both 2D model problems to produce the explicit FDM approximations of u in (2.65). Choosing this mesh density results in an error of 8% and 11% for our 2D model problems with 1NBC and 3NBCs, respectively, between our explicit FDM approximations of u we plan to use in the KF as our sensor traces and a converged solution for our PDE.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from N = 1600
50	5	9000^{-1}	52.295100527921761	10.9319707
100	10	18000^{-1}	54.564567053802584	7.06665811
150	15	27000^{-1}	56.329012926793702	4.06148717
200	20	36000^{-1}	57.330043021226821	2.35655158
250	25	45000^{-1}	57.820186223291351	1.52174892
300	30	54000^{-1}	58.115807361629464	1.01825257
350	35	63000^{-1}	58.267875384384830	0.75925318
400	40	72000^{-1}	58.378017773577476	0.57166074
450	45	81000^{-1}	58.459107819442998	0.43354970
500	50	90000^{-1}	58.504803079766944	0.35572240
550	55	99000^{-1}	58.547014807255323	0.28382818
600	60	108000^{-1}	58.579864831816472	0.22787864
650	65	117000^{-1}	58.598319246850032	0.19644742
700	70	126000^{-1}	58.619409179463496	0.16052745
750	75	135000^{-1}	58.635756667199715	0.13268472
800	80	144000^{-1}	58.644546150466326	0.11771463
1600	160	288000^{-1}	58.713660721093262	0

Table 5.5: 2D model problem with 3NBCs: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 25$Hz, and a disturbance location of $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

Due to the simulation duration change, we cannot compare the results in this section to those already discussed in this chapter. Instead, we rerun our 2D model problems for a simulation duration of $T = 3$ seconds and run the standard KF in (2.93)-(2.98)

over the third and final second in our simulation. Tables (5.6) and (5.8) contain success rates corresponding to 100 random disturbance locations positioned where the likelihood function in (2.107) is evaluated for our 2D model problems with 1NBC and 3NBCs, respectively.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	99	99	99	99	99	100	100	98	98	99	100
2	100	100	100	100	100	100	100	100	100	100	100
3	100	100	100	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100

Table 5.6: The success rate of our 2D model problem with 1NBC for a mesh dimension of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps over a simulation duration of $T = 3$ seconds. These probabilistic results use 100 random disturbance locations where the likelihood function in (2.107) is evaluated. The KF is run over the final second in an attempt to mimic the scenario considered when using an SVD.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	50	67	71	42	51	61	68	29	37	45	60
2	63	87	91	43	53	61	70	39	50	57	67
3	73	96	100	56	62	70	75	55	62	70	75
4	76	95	98	62	68	74	79	60	67	73	79
5	81	96	99	76	80	85	88	74	79	84	88
6	76	95	97	77	85	88	92	75	83	85	91

Table 5.7: The success rate of our 2D model problem with 1NBC for a mesh dimension of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps over a simulation duration of $T = 3$ seconds. These probabilistic results were produced using 100 random disturbance locations, and the KF is run over the final second in an attempt to mimic the scenario considered when using an SVD.

Tables (5.7) and (5.9) contain probabilistic success rates corresponding to 100 random disturbance locations for our 2D model problems with 1NBC and 3NBCs, respectively.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	87	91	91	84	84	95	95	80	80	87	91
2	99	100	100	96	96	99	99	96	96	99	99
3	100	100	100	99	99	100	100	99	99	100	100
4	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100

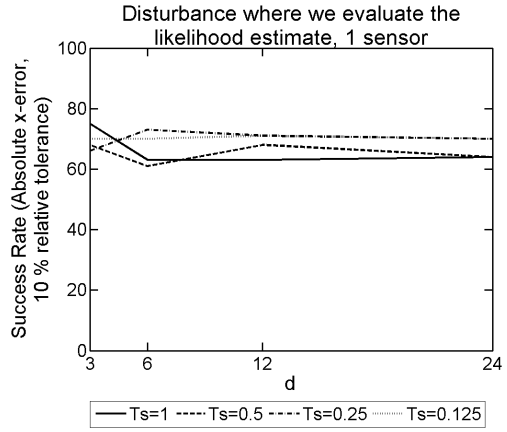
Table 5.8: The success rate of our 2D model problem with 3NBCs for a mesh dimension of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps over a simulation duration of $T = 3$ seconds. These probabilistic results use 100 random disturbance locations where the likelihood function in (2.107) is evaluated. The KF is run over the final second in an attempt to mimic the scenario considered when using an SVD.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	55	77	77	38	45	50	55	33	38	43	51
2	67	88	88	40	51	57	62	37	48	54	61
3	78	96	96	58	71	76	79	55	68	73	78
4	74	98	98	57	69	73	79	57	68	72	77
5	76	97	97	55	68	73	77	54	67	72	77
6	82	99	99	65	79	85	87	65	79	85	87

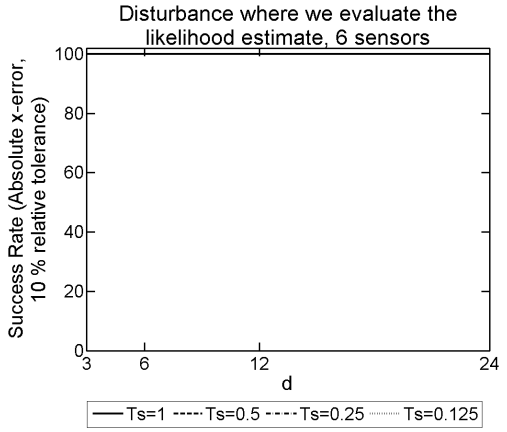
Table 5.9: The success rate of our 2D model problem with 3NBCs for a mesh dimension of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps over a simulation duration of $T = 3$ seconds. These probabilistic results were produced using 100 random disturbance locations, and the KF is run over the final second in an attempt to mimic the scenario considered when using an SVD.

Having computed probabilistic success rates without using an SVD for a simulation duration of $T = 3$ seconds, with the KF running over the third second, we can discuss the results obtained using an SVD and compare to those outlined in Tables (5.6), (5.7), (5.8) and (5.9). The full set of results in both graphical and tabular form can be found in Appendix C.3 and C.4 for our 2D model problems with 1NBC and 3NBCs, respectively. Amongst these results, we have one through to six sensors on the surface of our domain, with two different types of disturbance locations used to generate the success rates. Additionally, we have results for three different absolute errors with a range of acceptable relative tolerances. Since the results in this section involve the SVD, the results correspond to a range of values chosen for d and T_s.

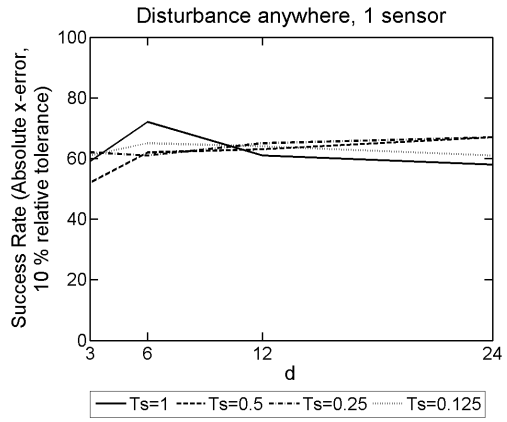
In Figure (5.13), we have the success rate for an absolute x-error, given a relative tolerance of 10%, for our 2D model problem with 1NBC.



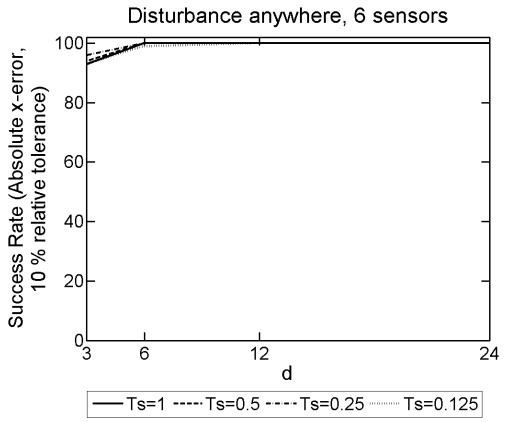
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



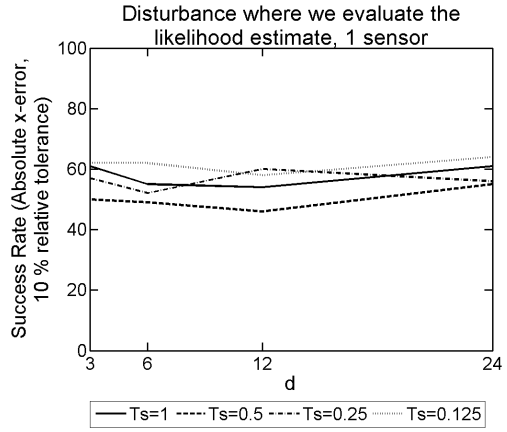
(c) One sensor present with random disturbance locations.



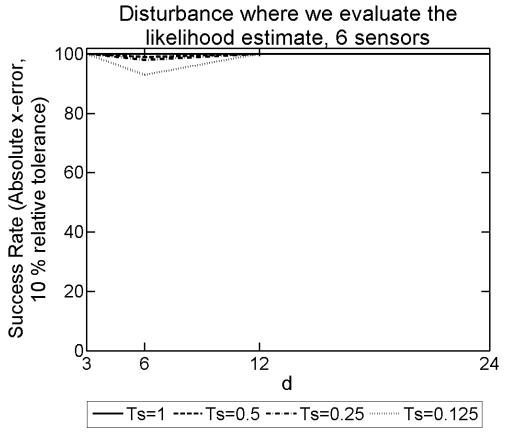
(d) Six sensors present with random disturbance locations.

Figure 5.13: The success rate for our 2D model problem with 1NBC given an absolute x-error with a relative tolerance of 10% for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with dimensions of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps.

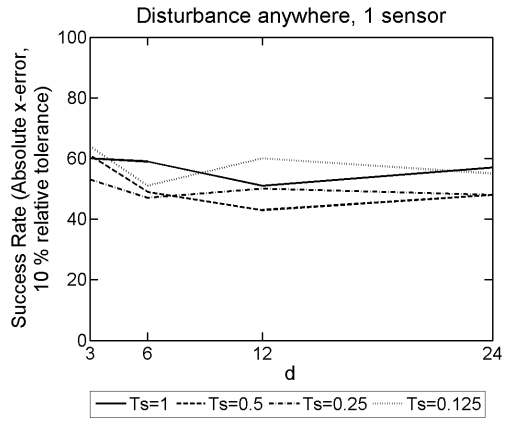
Inspecting this figure, we observe that irrespective of the disturbance locations used to produce the probabilistic results and the number of sensors present, both d and T_s have little impact on the success rate.



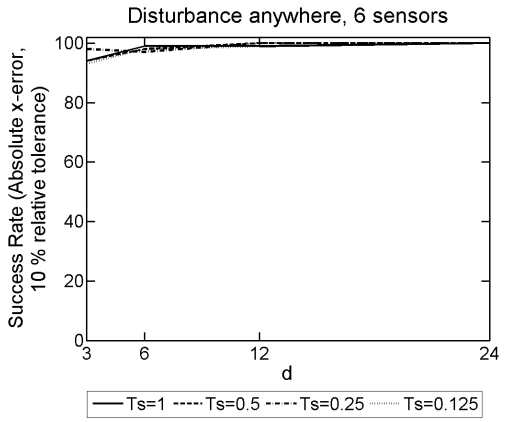
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



(d) Six sensors present with random disturbance locations.

Figure 5.14: The success rate for our 2D model problem with 3NBCs given an absolute x-error with a relative tolerance of 10% for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with dimensions of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps.

Further inspection of Figure (5.13) yields what we would expect, that is better success rates when more sensors are present, which we also observed for our 1D model problems in chapter 3. Comparing the results in Figure (5.13) obtained using an SVD to the results in both Tables (5.6) and (5.7), deduced without using an SVD, we can see that with fewer sensors present, the success rates when using an SVD are worse. However, with six sensors present the success rates given an absolute x-error with a relative tolerance of 10% are indistinguishable. This was expected since the singular values in our SVD converge to zero quickly, and so, even when we take d to be small the resultant principle components

contain a lot of the original information despite the reduced matrix sizes.

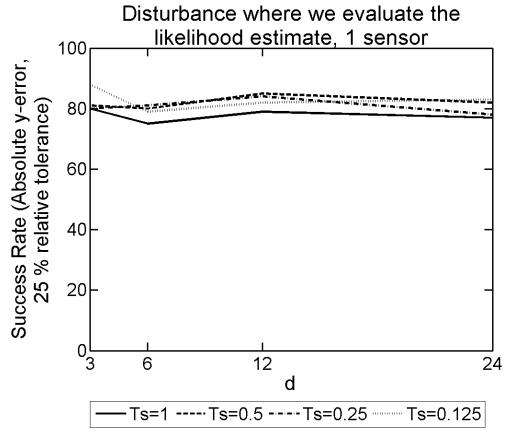
In Figure (5.14), we have the success rate for an absolute x-error, given a relative tolerance of 10%, for our 2D model problem with 3NBCs. Again, irrespective of the number of sensors present, and the type of disturbance locations used, the values we choose for d and T_s do not affect the success rate. Comparing Figure (5.14) to Figure (5.13), we observe that the success rate is not as good when fewer sensors are present. However, with six sensors present, the success rates are indistinguishable.

Comparing the results in Figure (5.14) to those in Tables (5.8) and (5.9), that is we compare the results deduced using an SVD to those without, we observe a lower success rate again with fewer sensors present when using an SVD. However, having six sensors present in our 2D model problems with and without an SVD result in similar success rates given an absolute x-error with a 10% relative tolerance.

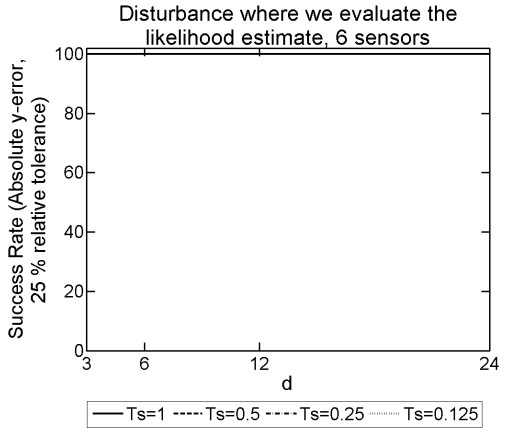
In Figure (5.15), we have the success rate for an absolute y-error, given a relative tolerance of 25%, for our 2D model problem with 1NBC. As discussed previously, we consider higher relative tolerance percentages for our y-error than we did for our x-error. We did this due to an unavoidable y-error of 14.1% in the worst-case scenario when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible.

Upon inspection of Figure (5.15), as observed previously, the values used for d and T_s have little impact on the probabilistic success rates. Again, we observe better success rates with more sensors present which we would expect. Comparing Figure (5.15) to Tables (5.6) and (5.7), we observe the same as we did with our absolute x-error for our 2D model problems with and without using an SVD. That is, there is no significant observable difference.

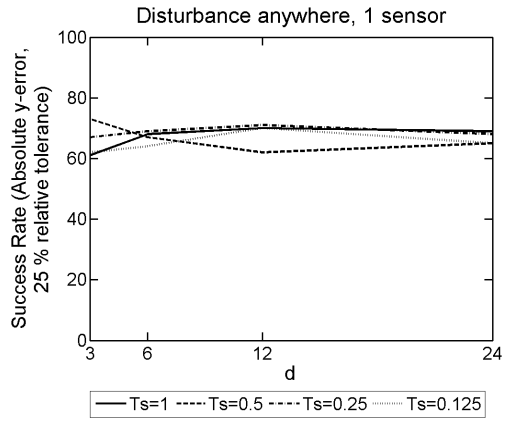
In Figure (5.16), we have the success rate for an absolute y-error, given a relative tolerance of 25%, for our 2D model problem with 3NBCs. Comparing Figure (5.15) to Figure (5.16), we observe similar results to what we observed with an absolute x-error.



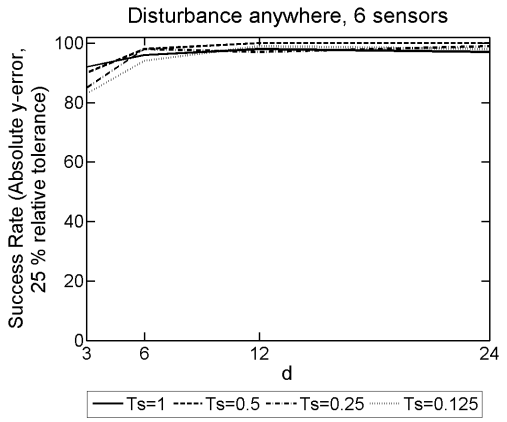
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

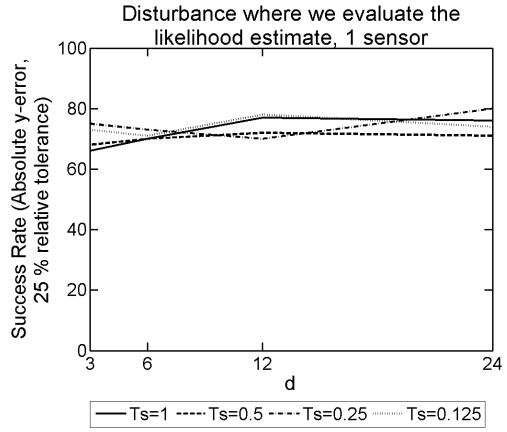


(c) One sensor present with random disturbance locations.

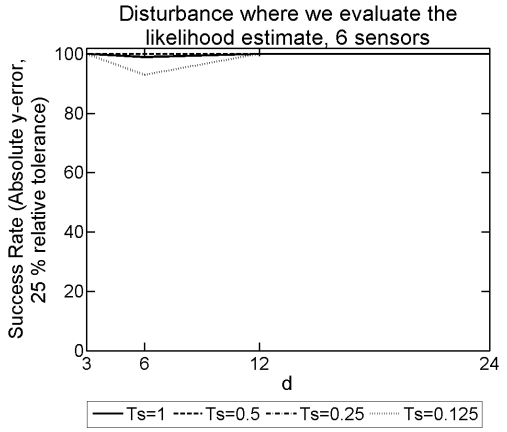


(d) Six sensors present with random disturbance locations.

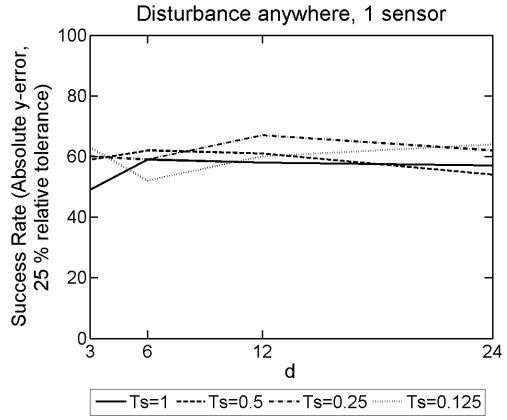
Figure 5.15: The success rate for our 2D model problem with 1NBC given an absolute y-error with a relative tolerance of 25% for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with dimensions of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps.



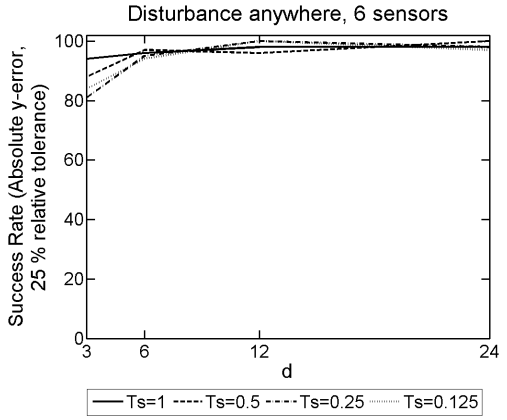
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



(d) Six sensors present with random disturbance locations.

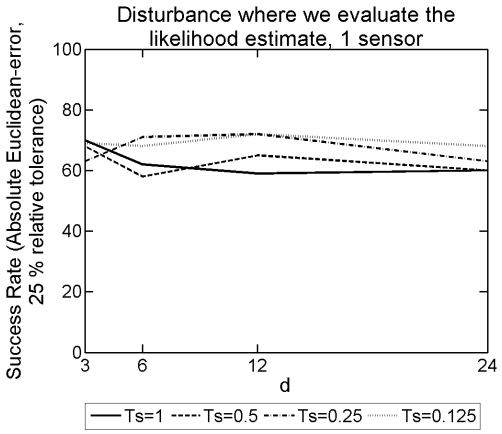
Figure 5.16: The success rate for our 2D model problem with 3NBCs given an absolute y-error with a relative tolerance of 25% for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with dimensions of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps.

Figures (5.17) and (5.18) show probabilistic success rates for an absolute Euclidean-error, given a relative tolerance of 25%, for our 2D model problem with 1NBC and 3NBCs, respectively. The Euclidean-error accounts for both the x and y-errors already discussed. As a result, we would expect the same patterns already discovered between the values of d and T_s, the success rate fluctuations with less and more sensors present, see Figures (5.17) and (5.18).

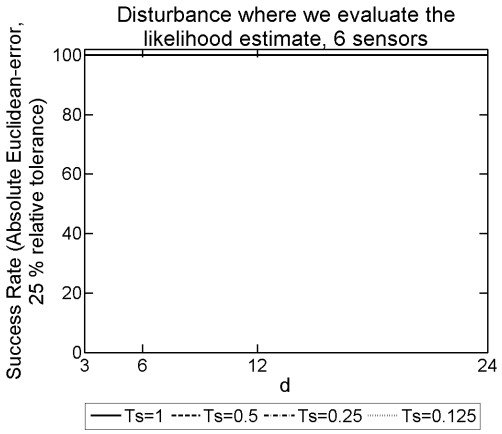
Upon inspection of Figures (5.17) and (5.18), we can confirm that all observations discussed in Figures (5.13), (5.14), (5.15) and (5.16) are observed when we are dealing

with a Euclidean-error.

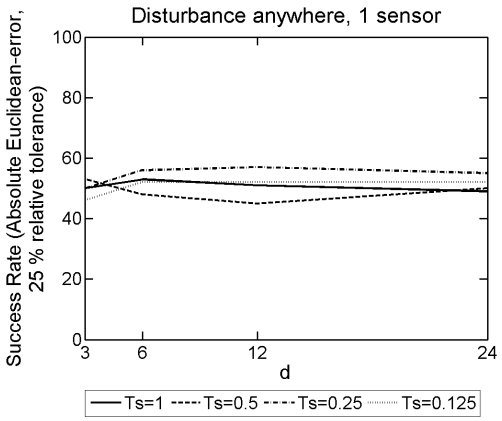
To conclude these results, we observe very good success rates for our 2D model problems with 1NBC and 3NBCs using an SVD when we have six sensors present, irrespective of the values chosen for d and T_s. This is a very significant set of results, as it shows we can reduce the matrix size in our KF and still obtain good results.



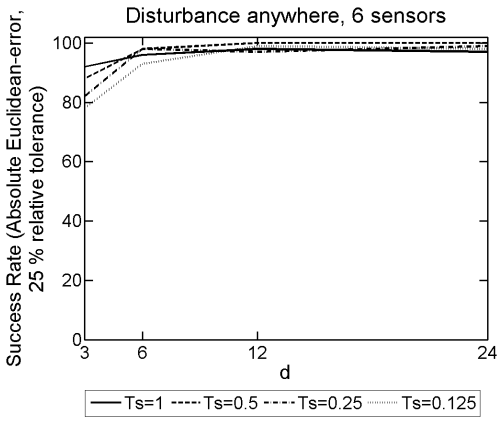
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

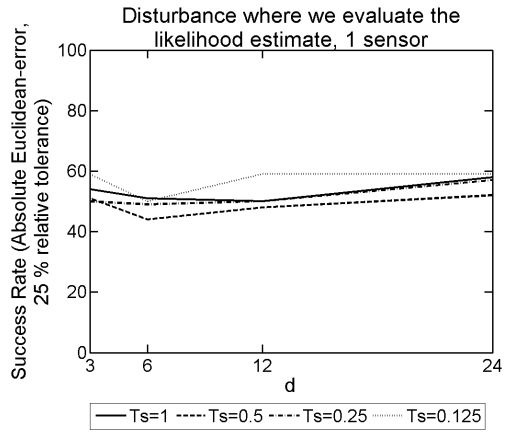


(c) One sensor present with random disturbance locations.

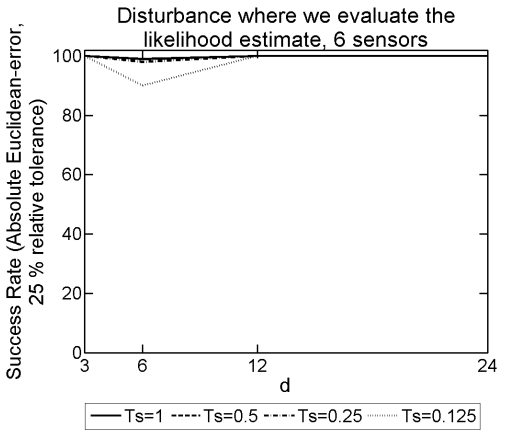


(d) Six sensors present with random disturbance locations.

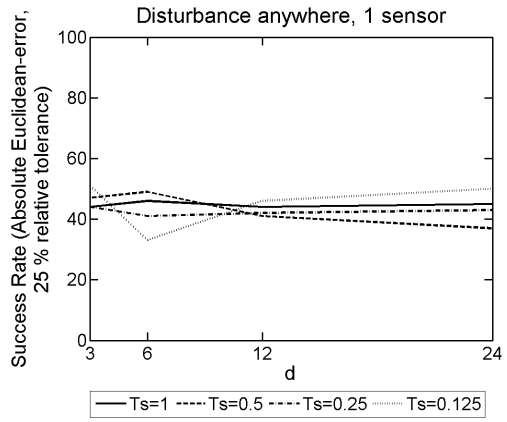
Figure 5.17: The success rate for our 2D model problem with 1NBC given an absolute Euclidean-error with a relative tolerance of 25% for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with dimensions of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps.



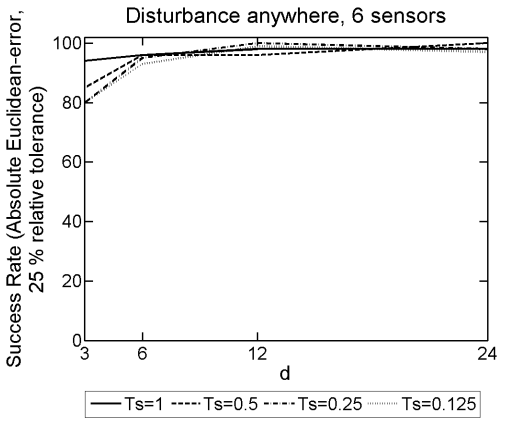
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



(d) Six sensors present with random disturbance locations.

Figure 5.18: The success rate for our 2D model problem with 3NBCs given an absolute Euclidean-error with a relative tolerance of 25% for an array of T_s and principal components, d, used to form an SVD from explicit FDM approximations of u on a mesh with dimensions of $N = 50$ by $M = 5$ nodes and $L = 9000$ discrete-time steps.

5.2.3 Further optimisations

In this section, we look at alternative approaches not yet explored to investigate the possibility of further reducing the run-time of our 2D model problems and improving the probabilistic success rates.

Tables (5.10) and (5.11) show the success rate of our 2D model problem with 3NBCs without using an SVD, with sensors placed along all three Neumann boundaries. By comparing the results in both tables to those in Tables (5.12) and (5.13), we can deduce

whether having sensors along the side of our 2D domain makes any noticeable impact on our success rate when compared to having sensors placed along $y := 0.02$ only.

	Success Rate						
	|x-error|			|y-error|			
Model variations	1%	5%	10%	10%	15%	20%	25%
3NBCs, disturbance locations where we evaluate the likelihood function	100	100	100	100	100	100	100
3NBCs, random disturbance locations	93	97	97	63	72	75	78

Table 5.10: The success rate for our 2D model problem with 3NBCs without using an SVD for absolute x and y-errors given a range of relative tolerances. These probabilistic results used a mesh with a dimension of $N = 50$ by $M = 5$ nodes and $L = 3000$ discrete-time steps to produce explicit FDM approximations of u collected at five sensor locations (three along $y := 0.02$, one along $y := 0$, and one along $y := 0.2$) in our 2D domain.

	Success Rate			
	|Euclidean-error|			
Model variations	10%	15%	20%	25%
3NBCs, disturbance locations where we evaluate the likelihood function	100	100	100	100
3NBCs, random disturbance locations	61	70	73	78

Table 5.11: The success rate for our 2D model problem with 3NBCs without using an SVD given an absolute Euclidean-error for a range of relative tolerance. These probabilistic results used a mesh with a dimension of $N = 50$ by $M = 5$ nodes and $L = 3000$ discrete-time steps to produce explicit FDM approximations of u collected at five sensor locations (three along $y := 0.02$, one along $y := 0$, and one along $y := 0.2$) in our 2D domain.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	52	53	54	95	95	99	99	52	53	59	62
2	98	100	100	96	96	99	99	96	96	98	99
3	99	100	100	99	99	100	100	99	99	99	100
4	100	100	100	99	99	100	100	99	99	100	100
5	100	100	100	100	100	100	100	100	100	100	100

Table 5.12: 2D model problem with 3NBCs: The success rate without the use of a minimisation algorithm for a varying number of sensors, mesh dimensions of $N = 50$, $M = 5$ and $L = 3000$, $F = 25$Hz, and 100 disturbance locations positioned where the likelihood function is evaluated.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	31	48	52	33	41	47	52	17	24	31	37
2	67	93	93	41	49	56	62	39	47	54	60
3	80	98	98	55	70	75	80	53	68	73	80
4	74	97	97	58	72	77	80	57	71	77	80
5	80	99	99	62	75	80	83	62	75	80	83

Table 5.13: 2D model problem with 3NBCs: The success rate without the use of a minimisation algorithm for a varying number of sensors, mesh dimensions of $N = 50$, $M = 5$ and $L = 3000$, $F = 25$Hz, and 100 random disturbance locations.

Upon inspection of both Tables (5.12) and (5.13), we can conclude that when we have only three sensors present along the top of our domain, that is along $y := 0.02$, the probabilistic success rates are indistinguishable from the those presented in Tables (5.10) and (5.11). Therefore, we note there is no improvement in success rate by placing additional sensors along $x := 0$ and $x := 0.2$ on our 2D domain.

In addition to this, Tables (5.10) and (5.11) have in total five sensors present, meaning

the run-time and system RAM requirements are equivalent to having five sensors along $y := 0.02$ only. From Tables (5.12) and (5.13), we can conclude that placing all five sensors along $y := 0.02$ on our 2D domain yields slightly better results than placing sensors along three sides of our 2D domain.

Moving onto another alternative approach, we have already looked at using an SVD to reduce the dimension of matrices in the KF. However, something that has not changed is the amount of discrete-time steps the KF is run over to compute the likelihood function in (2.107). We do this 244 or 255 times, 2D model problem-dependent, for each 100 different disturbance locations.

Tables (5.14) and (5.15) show probabilistic success rates for our 2D model problem with 1NBC using an SVD and having six sensors present for 100 disturbance locations positioned where we evaluate the likelihood function, and 100 random disturbance locations, respectively. The choice of d and T_s here is arbitrary since the values chosen have little impact on the success rate for our 2D model problems with a disturbance frequency of $F = 25$Hz.

Upon inspection of both Tables (5.14) and (5.15), we can see that terminating the KF after 20% of the total discrete-time steps gives similar results to when the KF is run for 100% of the total discrete-time steps. This result is very significant since this approach could save up to 80% of the total run-time for our 2D model problem with 1NBC using an SVD.

Moreover, Tables (5.16) and (5.17) show probabilistic success rates for our 2D model problem with 3NBCs using an SVD and having six sensors present for 100 disturbance locations positioned where we evaluate the likelihood function in (2.107), and 100 random disturbance locations, respectively. In a similar manner to our 2D model problem with 1NBC, we observe in both tables that terminating the KF after 40% of the total discrete-time steps gives similar results to when the KF is run for 100% of the total discrete-time steps. Again, another extremely significant result with the potential of saving up to 60% of the total run-time for our 2D model problem with 3NBCs using an SVD.

	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
KF Length† (%)	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
10	27	68	74	40	40	83	83	37	37	49	72
20	97	100	100	89	89	97	97	89	89	97	97
30	99	100	100	99	99	100	100	99	99	100	100
40	100	100	100	100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100	100	100	100	100

† Simulation duration is $T = 1$ second. If KF length = 10%, we would terminate the KF after 0.1 seconds.

Table 5.14: 2D model problem with 1NBC: The success rate using a random subset of 100 disturbance locations where the likelihood estimates are evaluated while terminating the KF early. These results correspond to using an SVD formed with $d = 6$ and $T_s = 0.5$ and having 6 sensors present.

	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
KF Length† (%)	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
10	20	57	68	40	48	61	69	29	36	47	58
20	80	99	100	81	89	93	97	81	89	93	97
30	87	100	100	88	93	95	97	88	93	95	97
40	87	100	100	88	91	92	95	88	91	92	95
100	87	100	100	90	93	95	98	90	93	95	98

† Simulation duration is $T = 1$ second. If KF length = 10%, we would terminate the KF after 0.1 seconds.

Table 5.15: 2D model problem with 1NBC: The success rate using 100 random disturbance locations, while terminating the KF early. These results correspond to using an SVD formed with $d = 6$ and $T_s = 0.5$ and having 6 sensors present.

	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
KF Length† (%)	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
10	4	26	38	28	28	51	51	13	16	22	36
20	33	60	63	39	39	75	75	31	33	45	59
30	65	91	91	67	67	94	94	66	66	79	89
40	86	96	96	85	85	94	94	84	84	90	93
50	93	98	98	93	93	100	100	92	92	97	99
60	93	99	99	97	97	99	99	97	97	98	99
100	98	99	99	97	97	100	100	97	97	99	99

† Simulation duration is $T = 1$ second. If KF length = 10%, we would terminate the KF after 0.1 seconds.

Table 5.16: 2D model problem with 3NBCs: The success rate using a random subset of 100 disturbance locations where the likelihood estimates are evaluated while terminating the KF early. These results correspond to using an SVD formed with $d = 6$ and $T_s = 0.5$ and having 6 sensors present.

	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
KF Length† (%)	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
10	9	26	41	29	40	51	53	11	16	31	35
20	31	67	73	52	66	74	79	42	50	62	70
30	52	87	89	58	76	85	88	56	72	81	86
40	63	94	95	68	85	90	91	67	83	89	91
50	74	99	100	71	89	91	93	71	89	91	93
60	77	98	99	77	92	94	94	77	92	94	94
100	83	98	98	80	93	97	97	80	93	96	96

† Simulation duration is $T = 1$ second. If KF length = 10%, we would terminate the KF after 0.1 seconds.

Table 5.17: 2D model problem with 3NBCs: The success rate using 100 random disturbance locations, while terminating the KF early. These results correspond to using an SVD formed with $d = 6$ and $T_s = 0.5$ and having 6 sensors present.

5.3 Summary

In this chapter, we have investigated several approaches which have produced some good results for our 2D model problems. Initially, we made the decision not to use a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107). This decision was made due to the run-time requirement of the minimisation algorithm with 245 or 255 initial guesses of $\underline{\mathbf{x}}_0$ for our 2D model problem with 1NBC and 3NBCs, respectively. As a result of this, we apply the alternative approach discussed in chapter 3, where the initial guesses of $\underline{\mathbf{x}}_0$ used in the minimisation algorithm are our only guesses for $\underline{\mathbf{x}}_0$. Using time series data produced in the KF, we calculate the likelihood function for each guess of $\underline{\mathbf{x}}_0$ and determine the value of $\underline{\mathbf{x}}_0$ which minimises (2.107) to be our prediction of the disturbance location used to generate the sensor traces.

The results obtained using the standard KF in (2.93)-(2.98) yielded good probabilistic success rates when more than a single sensor was present for disturbance locations positioned where we evaluate the likelihood function in (2.107). However, the probabilistic success rate for random disturbance locations given y and Euclidean-errors are significantly worse, although with more sensors present the results are reasonable. We can associate this drop in success rate to the unavoidable y-error of 14.1% when the distance between a random disturbance location and where we evaluate the likelihood function is as larger as possible. In our 2D model problems, this distance is 0.0028284.

In addition to these results, we have compiled an array of outputs for both 2D model problems using an SVD to reduce the size of the matrices in the KF. We discovered the same patterns in the results obtained as we did without using an SVD. That is, we observed lower success rates given y and Euclidean-errors for random disturbance locations. Despite this, the fact we were able to obtain similar results using an SVD which significantly reduces the matrix dimensions in the KF, makes this approach an extremely viable option. This was expected since the singular values in our SVD converge to zero quickly, and so, even when we take d to be small the resultant principle components contain a lot of the original information despite the reduced matrix sizes.

At the end of this chapter, we looked at alternative approaches in a bid to further optimise our 2D model problems. We discovered that placing sensors along $x := 0$ and $x := 0.2$ in our 2D domain had little impact on our success rates when compared to only placing sensors along $y := 0.02$. What was more insightful is the ability to terminate the

KF after running for 20% or 40% of the total discrete-time steps in our 2D model problems with 1NBC or 3NBCs, respectively. This means that, over the approach implemented using an SVD, we could reduce the total run-time by up to 80%.

Chapter 6

2D Model Problem: Higher Frequencies

In this chapter we extend the work in chapter 5 where we considered two different 2D model problems, each with a disturbance frequency of $F = 25$Hz in (1.6). We now investigate higher frequencies of $F = 150$Hz and $F = 300$Hz. This was first explored in chapter 4 for our 1D model problem, where we obtained good probabilistic success rates.

We first outline the model problem and test the convergence of our explicit FDM approximations of u in (2.65) for different forcing functions in (1.6). We then illustrate and explain the noise added to these approximations of u, and show a schematic for our model problems.

Once we have outlined the 2D model problems we consider in this chapter, we discuss the results obtained from both 2D model problems for each disturbance frequency considered. Since we require more accurate explicit FDM approximations of u for our sensor traces when compared to our 2D model problems in chapter 5 with $F = 25$Hz, the matrix dimensions in the KF will be larger. Therefore, the results in this chapter were obtained using an SVD to minimise the run-time and system RAM requirements.

6.1 Model problem outline

In chapter 1, we outlined the forcing function in (1.6), which attempts to mimic a disturbance caused by CAD. Recall that from research, see [26], [27], [28], [29], [30] and [31], the real frequency range of this disturbance is $300 - 800$Hz. In chapter 5 we chose

the frequency of this disturbance to be $F = 25$Hz for simplicity, with the knowledge that a finer mesh would be required to approximate u in (2.65) for higher frequencies in f, see (1.6).

In chapters 3, 4 and 5, we explored an approach which results in significantly smaller matrix dimensions in the KF, which in turn reduced the run-time and system RAM requirements for our model problems. Now we are working spatially in 2D, and require finer meshes to approximate u in (2.65) for higher frequencies, this approach is a necessity for these 2D model problems. This alternative approach involves taking the explicit FDM approximations of u in (2.65) from a subset of our simulation duration, denoted by T_s, to form an SVD. From the SVD created, we take the first d principal components which correspond to the largest singular values and modify our KF, see (2.127)-(2.132), which subsequently reduces the size of the matrices in the KF.

6.1.1 Forcing function

In this chapter we investigate disturbance frequencies of $F = 150$Hz and $F = 300$Hz in (1.6). The amplitude, denoted by A, in our forcing function remains the same, ensuring the matrix in (2.128), forming part of the KF, is invertible. As discussed previously in chapter 4, by increasing the disturbance frequency in our forcing function, we increased the Gaussian spread of our forcing function by altering ϵ in (1.6) to get better results.

Figure (5.1) shows the Gaussian spread in our 2D domain for our forcing function in (1.6). We define the value chosen for ϵ by obtaining the percentage of the positive peak remaining where we evaluate the likelihood function in (2.107) for the worst-case scenario. Recall that the worst-case scenario is when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible. For our 2D model problems, this distance is 0.0028284. Using this distance, we choose $\epsilon = 1.23 \times 10^{-5}$ and $\epsilon = 1.99 \times 10^{-4}$ for our 2D model problems with disturbance frequencies of $F = 150$Hz and $F = 300$Hz, respectively.

In our 2D model problem with a disturbance frequency of $F = 150$Hz, the Gaussian spreads peak remaining in the worst-case scenario is 52% at the nearest position (2.107) is evaluated. Our Gaussian spread decays to approximately zero at a distance of 0.012 away from the actual disturbance location, that is 6% and 60% of our domain in the x and y-directions, respectively. For our 2D model problem with a disturbance frequency

of $F = 300$Hz, the Gaussian spreads peak remaining in the worst-case scenario is 96% at the nearest position (2.107) is evaluated. Our Gaussian spread decays to approximately zero at a distance of 0.048 away from the actual disturbance location, that is 24% and 240% of our domain in the x and y-directions, respectively.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 2000$
50	5	9000^{-1}	1.476307850713303	37.9302273
100	10	13500^{-1}	2.362060668466852	0.68970458
150	15	18000^{-1}	1.901351068458216	20.0597433
200	20	22500^{-1}	1.664278368474221	30.0272094
250	25	27000^{-1}	2.106417873434576	11.4379304
300	30	31500^{-1}	2.640073671596904	10.9990520
350	35	36000^{-1}	2.146214630779191	9.76471864
400	40	40500^{-1}	2.595812293482607	9.13813060
450	45	45000^{-1}	2.698998970912966	13.4765033
500	50	49500^{-1}	2.752907678275824	15.7430368
600	60	58500^{-1}	1.907032806113935	19.8208607
700	70	67500^{-1}	2.021893312134235	14.9916745
800	80	76500^{-1}	2.299334076959987	3.32697653
900	90	85500^{-1}	2.436987096704126	2.46049635
1000	100	94500^{-1}	2.528047967955005	6.28905256
1100	110	103500^{-1}	2.565648928902063	7.86994444
1200	120	112500^{-1}	2.560724325751560	7.66289481
1300	130	121500^{-1}	2.537677657433419	6.69392247
1400	140	130500^{-1}	2.508789805595369	5.47936387
1500	150	139500^{-1}	2.476758088380485	4.13262484
2000	200	184500^{-1}	2.378465050853140	0

Table 6.1: 2D model problem with 1NBC: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 150$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

Since the forcing function in (1.6) has changed for our 2D model problems in this

chapter, our explicit FDM approximations of u in (2.65) change. Therefore, we need to redo the convergence of the absolute maximum approximations of u across all nodes and discrete-time steps in (2.65) for both disturbance frequencies. To be able to solve (2.65), we need to know both $\mathbf{V}^0$ and $\mathbf{V}^1$, which we outlined in chapter 5.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from N = 2000
50	5	9000^{-1}	1.476285738616596	31.87156983
100	10	13500^{-1}	2.362072353843534	9.006188435
150	15	18000^{-1}	1.924220296222152	11.20004438
200	20	22500^{-1}	1.790811537284375	17.35666371
250	25	27000^{-1}	1.702925734694870	21.41246511
300	30	31500^{-1}	1.645754953664509	24.05081313
350	35	36000^{-1}	2.236976411982961	3.233179985
400	40	40500^{-1}	2.438993062109521	12.55596675
450	45	45000^{-1}	2.741505784602714	26.51648696
500	50	49500^{-1}	2.939024445632943	35.63168462
600	60	58500^{-1}	2.182944028638430	0.739705600
700	70	67500^{-1}	1.848640824717378	14.68792657
800	80	76500^{-1}	2.034733425484549	6.100024907
900	90	85500^{-1}	2.107002091310943	2.764931555
1000	100	94500^{-1}	2.202162063803347	1.626562159
1100	110	103500^{-1}	2.239722873026367	3.359938633
1200	120	112500^{-1}	2.240631983861499	3.401892770
1300	130	121500^{-1}	2.227541463494679	2.797784378
1400	140	130500^{-1}	2.210038900153278	1.990066649
1500	150	139500^{-1}	2.195485956845127	1.318469575
2000	200	184500^{-1}	2.166915830888715	0

Table 6.2: 2D model problem with 3NBCs: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 150$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

Upon inspection of Table (6.1), we take our mesh to have $N = 150$ by $M = 15$ nodes

with $L = 18000$ discrete-time steps, yielding an error of 20%. We chose this discretised mesh spacing because the coarser mesh spacing did not obtain sufficiently good enough results.

Table (6.2) contains the maximum absolute approximation of u across all nodes and discrete-time steps in (2.65) for our 2D model problem with 3NBCs, with a disturbance frequency of $F = 150$Hz. We take our mesh to have $N = 150$ by $M = 15$ nodes with $L = 18000$ discrete-time steps, resulting in an error of 11%. Again, we chose this because it resulted in better results than when we used coarser meshes.

In Tables (6.3) and (6.4), we observe the same as we did in Tables (6.1) and (6.2), but for a disturbance frequency of $F = 300$Hz. In both cases, we take our mesh to have $N = 500$ by $M = 50$ nodes with $L = 45000$ discrete-time steps, resulting in an error of 18.23% and 18.33%, respectively. As mentioned previously for other model problems, we chose this mesh density because it resulted in better results than when we used coarser meshes.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 2000$
100	10	9000^{-1}	0.278277766906823	43.09738373
200	20	18000^{-1}	2.542590528603840	419.9123695
300	30	27000^{-1}	0.480724136359279	1.700874756
400	40	36000^{-1}	2.675249752050029	447.0387080
500	50	45000^{-1}	0.578180301046536	18.22709435
600	60	54000^{-1}	0.513371582979070	4.974919533
700	70	63000^{-1}	0.495968763310072	1.416367297
800	80	72000^{-1}	0.492032642807167	0.611504023
900	90	81000^{-1}	0.490443578099907	0.286569911
1000	100	90000^{-1}	0.490060014756417	0.208138356
2000	200	180000^{-1}	0.489042130504032	0

Table 6.3: 2D model problem with 1NBC: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 300$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

N	M	Δt	Maximum absolute solution over all nodes and time steps	% from $N = 2000$
100	10	9000^{-1}	0.278273424374160	43.09752665
200	20	18000^{-1}	2.691860222890391	450.4424469
300	30	27000^{-1}	0.480613379726861	1.722235621
400	40	36000^{-1}	2.675036862826007	447.0023383
500	50	45000^{-1}	0.578657321528331	18.32618545
600	60	54000^{-1}	0.513546164323530	5.011993140
700	70	63000^{-1}	0.495839682343571	1.391300215
800	80	72000^{-1}	0.492031202253952	0.612526829
900	90	81000^{-1}	0.490443487311206	0.287864467
1000	100	90000^{-1}	0.490067564895042	0.210994334
2000	200	180000^{-1}	0.489035727220794	0

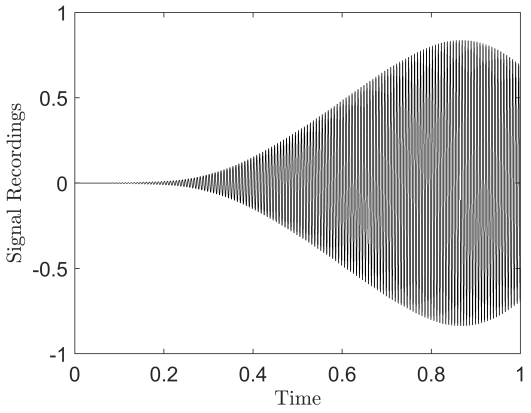
Table 6.4: 2D model problem with 3NBCs: The convergence of our explicit FDM approximation of u in (2.65) for a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 300$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$.

6.1.2 Added noise

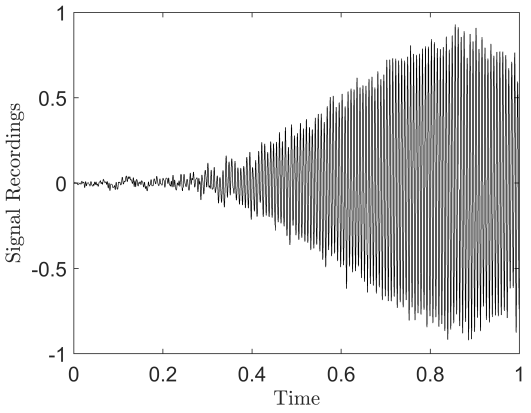
As mentioned already for other model problems considered in this thesis, we add two different forms of random Gaussian noise to our explicit FDM approximations of u in (2.65), denoted by $\tilde{\mathbf{w}}^n$ and $\tilde{\mathbf{z}}^n$ in the KF. We scale the first by the error associated with the explicit FDM approximations of u in (2.65). For our 2D model problem with 1NBC, this error is measured by the combination of the higher-order terms neglected in (2.28) and (2.33). Additionally, for our 2D model problem with 3NBCs this error is measured by the combination of the higher-order terms neglected in (2.28), (2.33), (2.39), (2.47), (2.53) and (2.62). The second random Gaussian noise added attempts to mimic inaccuracies recorded by sensors in a real-life scenario, be that ambient noise in the background or errors due to manufacturing defects. Without experimental data, we cannot accurately predict the magnitude of this noise. As a result, we model it after the error associated with the explicit FDM approximation of u in (2.65) but scale it ensuring it is the larger of the two random Gaussian noises added since we would expect this noise to be the

dominant of the two.

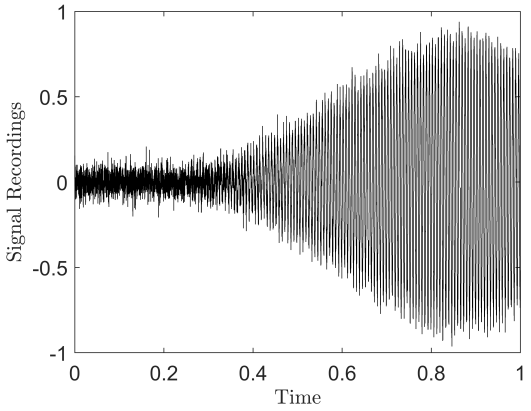
For each 2D model problem considered in this chapter, we scale the noise added to mimic a scenario where the original explicit FDM approximations of u are still recognisable but significantly obscured.



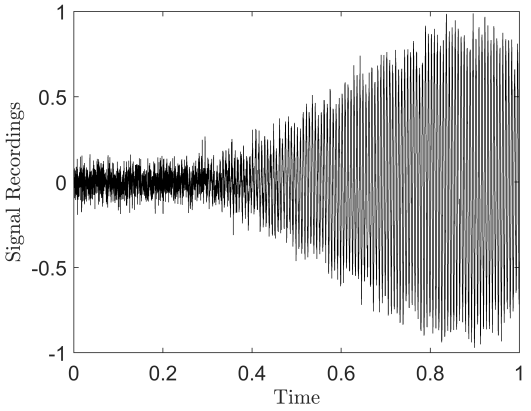
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).

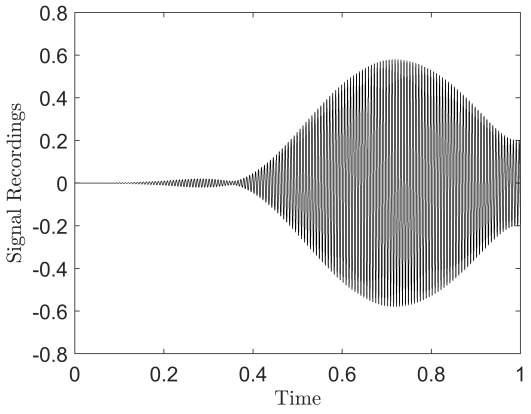


(c) Noise added to mimic errors recorded in real-world scenarios.

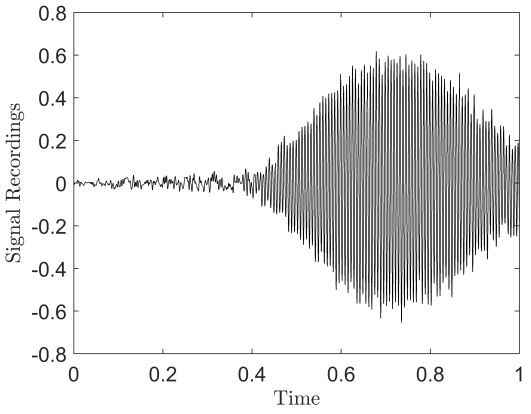


(d) All added noise present.

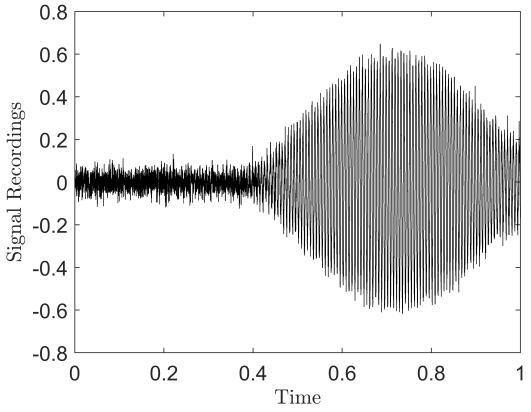
Figure 6.1: 2D model problem with 1NBC: Illustration of the noise added to our explicit FDM approximations of u in (2.65) using a single sensor trace at $(x, y) = (0.1, 0.02)$, a mesh dimension of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 150$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$. All figures show only the first second of our sensor traces.



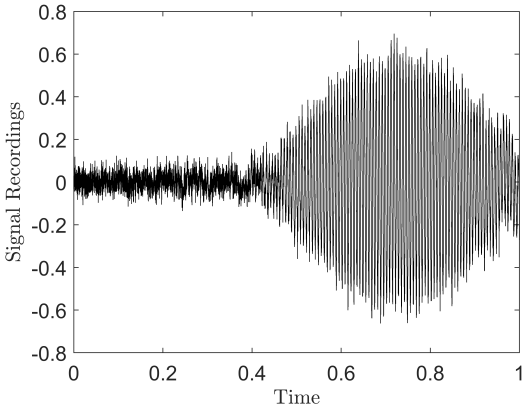
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.



(d) All added noise present.

Figure 6.2: 2D model problem with 3NBCs: Illustration of the noise added to our explicit FDM approximations of u in (2.65) using a single sensor trace at $(x, y) = (0.1, 0.02)$, a mesh dimension of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 150$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$. All figures show only the first second of our sensor traces.

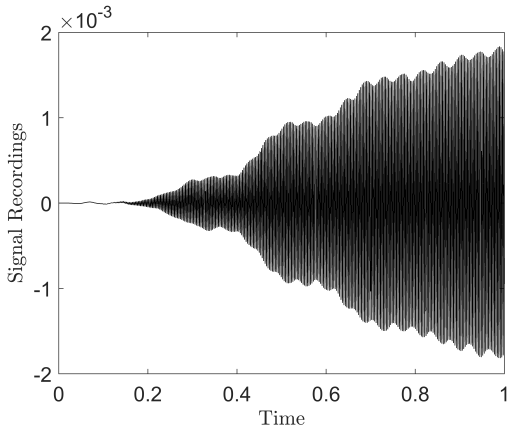
Figure (6.1) illustrates the effect the added noise has on our explicit FDM approximations of u in (2.65) for our 2D model problem with 1NBC. These figures only show the first second of our $T = 3$ second simulation and are formed using a mesh density of $N = 150$ by $M = 15$ nodes with $L = 18000$ discrete-time steps, a disturbance frequency of $F = 150$Hz, and a single sensor trace located at $(x, y) = (0.1, 0.02)$. We scale the random Gaussian noise added to our explicit FDM approximations of u, see

Figure (6.1)(b) by 3×10^2, and by 3×10^4 in Figure (6.1)(c). In total, the noise added is significant, see Figure (6.1)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (6.1)(a).

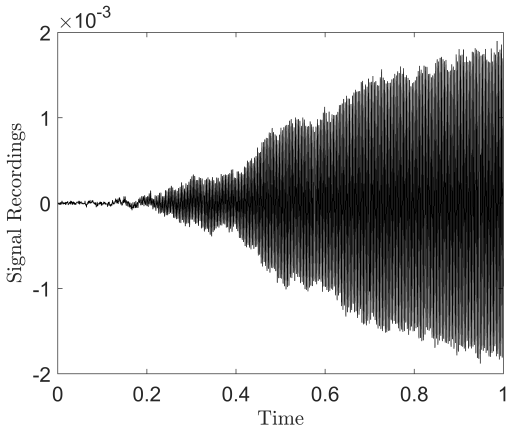
In Figure (6.2), the effect of the noise added to our explicit FDM approximations of u in (2.65) for our 2D model problem with 3NBCs is shown. These figures only show the first second of our $T = 3$ second simulation and are formed using a mesh density of $N = 150$ by $M = 15$ nodes with $L = 18000$ discrete-time steps, a disturbance frequency of $F = 150$Hz, and a single sensor trace located at $(x, y) = (0.1, 0.02)$. We scale the random Gaussian noise added to our explicit FDM approximations of u, see Figure (6.2)(b) by 2×10^2, and by 1×10^4 in Figure (6.2)(c). Again in total, the noise added is significant, see Figure (6.2)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (6.2)(a).

Figure (6.3) contains an illustration of the added noise to our explicit FDM approximations of u in (2.65) for our 2D model problem with 1NBC. These figures only show the first second of our $T = 3$ second simulation and are formed using a mesh density of $N = 500$ by $M = 50$ nodes with $L = 45000$ discrete-time steps, a disturbance frequency of $F = 300$Hz, and a single sensor trace located at $(x, y) = (0.1, 0.02)$. We scale the random Gaussian noise added to our explicit FDM approximations of u, see Figure (6.3)(b) by 5, and by 3×10^2 in Figure (6.3)(c). Again, the noise added is significant, see Figure (6.3)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (6.3)(a).

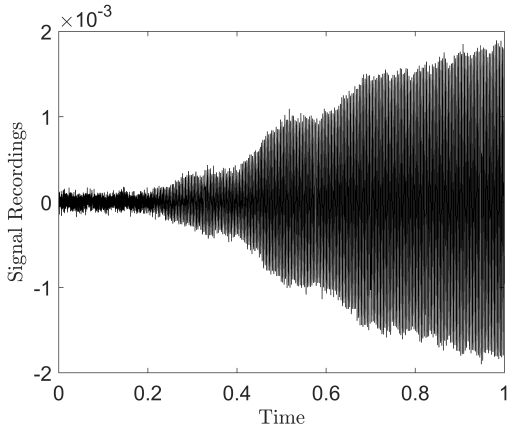
Lastly, Figure (6.4) shows the effect from the added noise on our explicit FDM approximations of u in (2.65) for our 2D model problem with 3NBCs. These figures only show the first second of our $T = 3$ second simulation and are formed using a mesh density of $N = 500$ by $M = 50$ nodes with $L = 45000$ discrete-time steps, a disturbance frequency of $F = 300$Hz, and a single sensor trace located at $(x, y) = (0.1, 0.02)$. We scale the random Gaussian noise added to our explicit FDM approximations of u, see Figure (6.4)(b) by 3, and by 2×10^2 in Figure (6.4)(c). In total, the noise added is significant, see Figure (6.4)(d), but does not overwhelm the explicit FDM approximations of u, see Figure (6.4)(a).



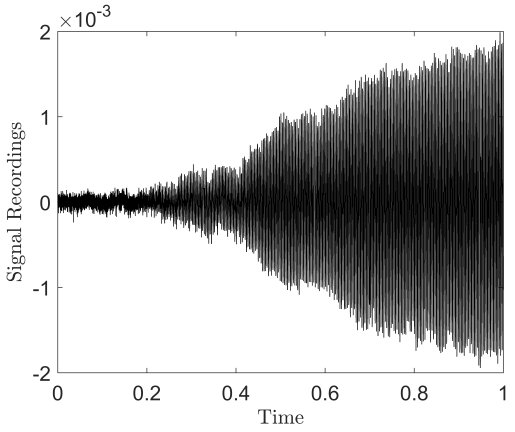
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).

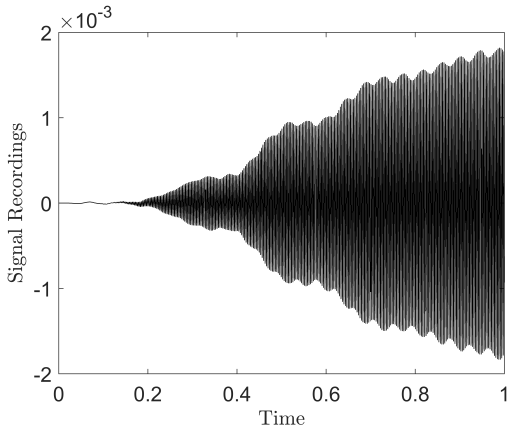


(c) Noise added to mimic errors recorded in real-world scenarios.

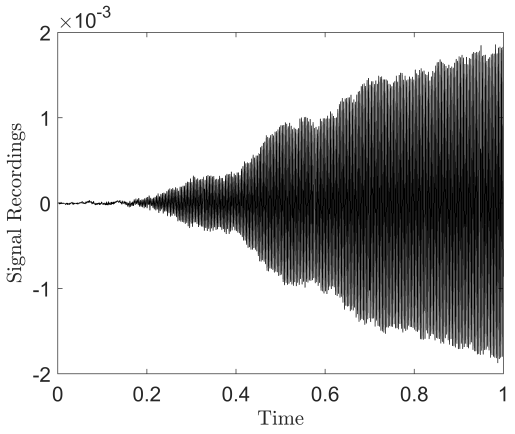


(d) All added noise present.

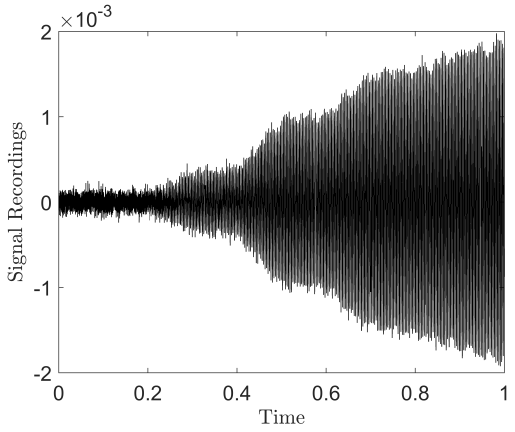
Figure 6.3: 2D model problem with 1NBC: Illustration of the noise added to our explicit FDM approximations of u in (2.65) using a single sensor trace at $(x, y) = (0.1, 0.02)$, a mesh dimension of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 300$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$. All figures show only the first second of our sensor traces.



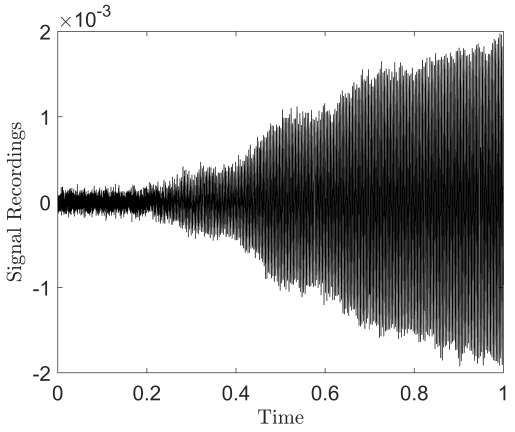
(a) No added noise.



(b) Noise added to mimic the error associated with the FDM approximation of u in (1.1).



(c) Noise added to mimic errors recorded in real-world scenarios.



(d) All added noise present.

Figure 6.4: 2D model problem with 3NBCs: Illustration of the noise added to our explicit FDM approximations of u in (2.65) using a single sensor trace at $(x, y) = (0.1, 0.02)$, a mesh dimension of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 300$Hz and a disturbance location at $\underline{\mathbf{x}}_0 \equiv (x, y) = (0.144, 0.008)$. All figures show only the first second of our sensor traces.

6.1.3 Model problem schematic

As discussed previously in this chapter, we will use an SVD to solve the 2D model problems with higher disturbance frequencies in f, see (1.6) As a result, the sensor locations are the same as is illustrated in Figure (5.4), and the schematic for our 2D model

problems considered in this chapter follow the same steps outlined in Figure (5.12).

6.2 Results

In this section, we discuss the results obtained for our 2D model problems with 1NBC and 3NBCs for disturbance frequencies of $F = 150$Hz and $F = 300$Hz. We produce probabilistic success rates for three different types of error: an absolute x-error, y-error and Euclidean-error. In each case, we classify the success rates based on a relative tolerance percentage. The probabilistic results are obtained using 100 random disturbance locations and a random subset of 100 disturbances positioned where the likelihood in (2.107) is evaluated.

6.2.1 Disturbance frequency of $F = 150$Hz

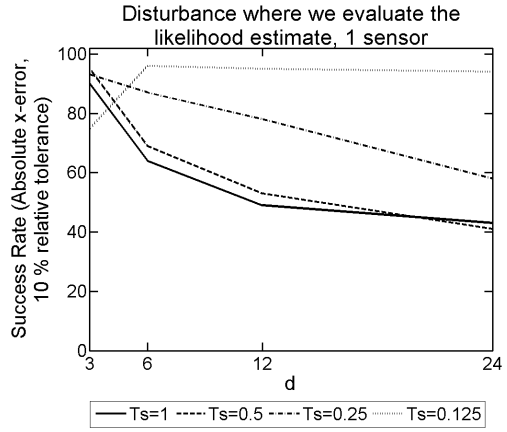
We start by looking at the results obtained for our 2D model problems with 1NBC and 3NBCs for a disturbance frequency of $F = 150$Hz. For the full set of results, see Appendix D.1 and D.2 for our 2D model problems with 1NBC and 3NBCs, respectively.

The results in this section originate from our 2D model problems using an SVD, which is constructed using data from our explicit FDM approximations of u over a duration of T_s seconds, from which we take the first d principle components with the largest singular values. In both figures we use a range of values for T_s and d, and display the corresponding success rate. This success rate is calculated by running our 2D model problem 100 times, with a different disturbance location each time. To be classed as a success, they must be able to predict the disturbance location within an absolute error with a relative tolerance percentage. The results presented in Figure (6.5) and Figure (6.6) have one and six sensors present, whose success rate is determined from an absolute x-error with a relative tolerance percentage of 10%. Additional results corresponding to two, three, four and five sensors present with different absolute errors with a range of tolerance percentages can be found in Appendix D.1 and D.2.

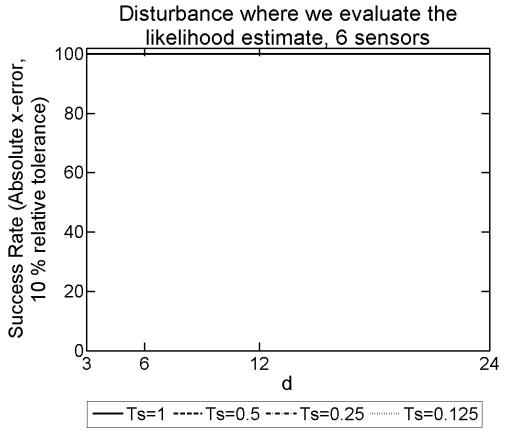
Moving on to inspecting the results in Figure (6.5) and Figure (6.6), we observe that when there is one sensor present, the success rate for higher values of d are poor when $T_s > 0.125$. This was observed in chapter 4 for our 1D model problem with a disturbance frequency of $F = 150$Hz. We are not sure why the success rate drops when we take more

principal components in our SVD. However, it is worth noting that when we take only $d = 3$ principal components, we get a reasonable success rate irrespective of the value taken for T_s, see Figures (6.5)(a) and (6.5)(c).

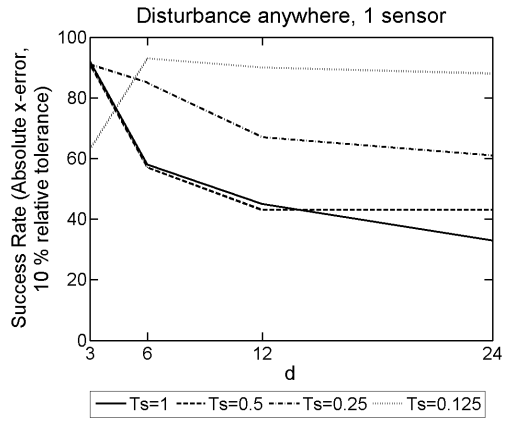
In addition to these observations, we conclude that when we have six sensors present as opposed to only one, we get significantly better results. We would expect this since there is more data input into the KF, and we have also observed this in chapters 3, 4 and 5. Lastly, upon inspection of Figures (6.5) and (6.6), we can see that forming our SVD using less data from our explicit FDM approximations of u in (2.65), that is having a smaller value for T_s which represents how much of the simulation our explicit FDM approximations of u are used to form the SVD, yields better results irrespective of the number of sensors present.



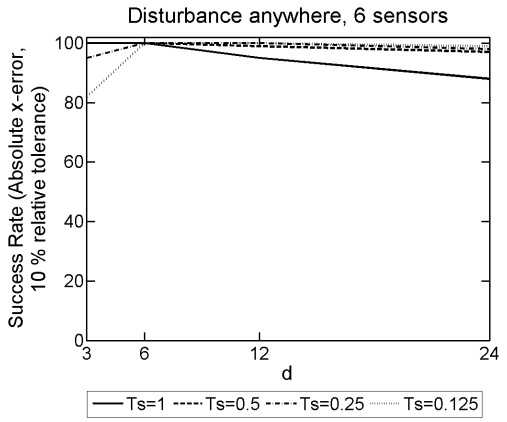
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

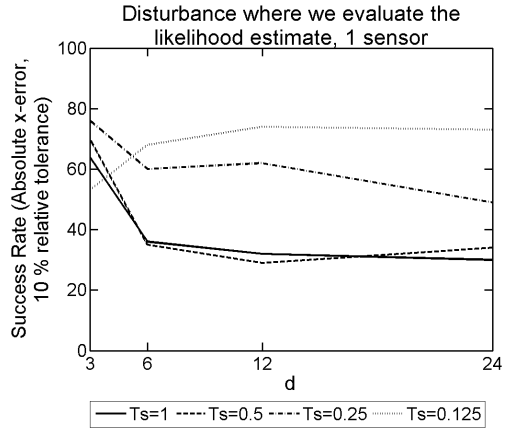


(c) One sensor present with random disturbance locations.

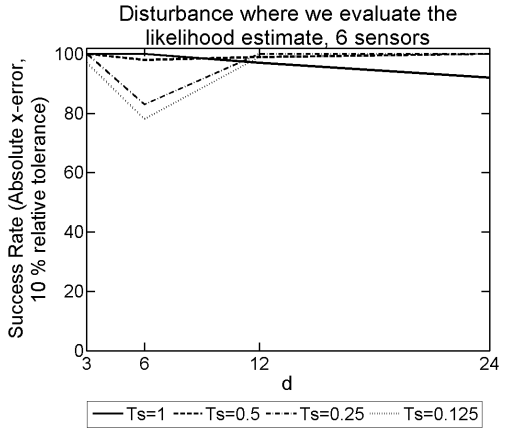


(d) Six sensors present with random disturbance locations.

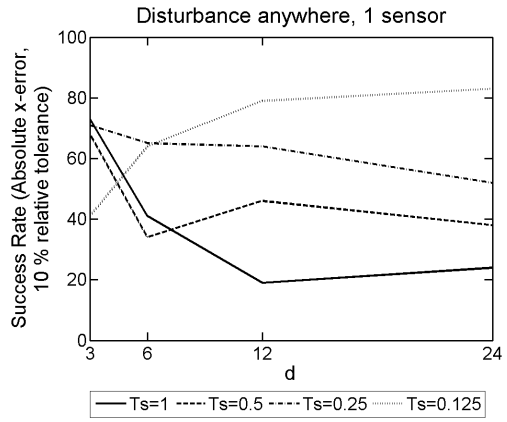
Figure 6.5: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 150$Hz given an absolute x-error with a relative tolerance of 10%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps.



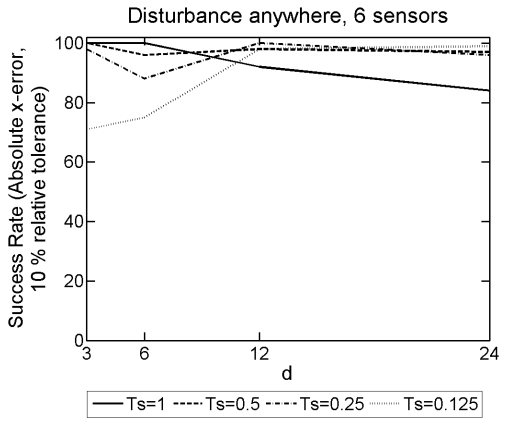
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



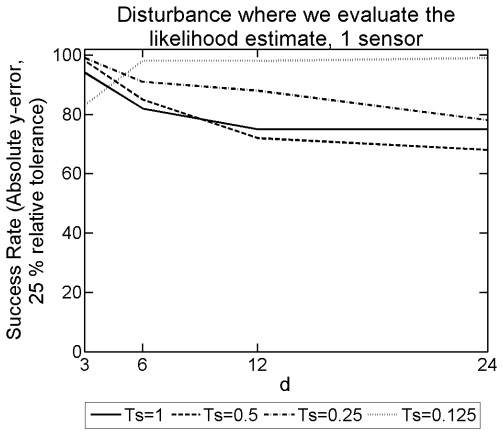
(d) Six sensors present with random disturbance locations.

Figure 6.6: The success rate for our 2D model problem with 3NBCs with a disturbance frequency of $F = 150$Hz given an absolute x-error with a relative tolerance of 10%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps.

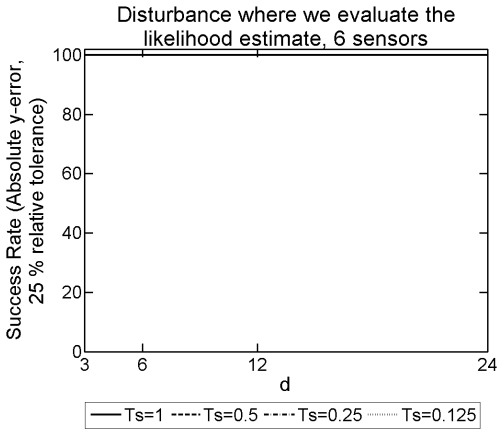
Next, we inspect Figures (6.7) and (6.8) which contain probabilistic success rates for an absolute y-error, given a relative tolerance percentage of 25% for our 2D model problems with 1NBC and 3NBCs, respectively. As discussed in chapter 5, we take a higher relative tolerance percentage for our y-error due to the unavoidable error of 14.1% caused in the worst-case scenario when the distance between the actual disturbance location and a position we evaluate the likelihood function in (2.107) is as larger as possible.

Upon inspection of both Figures (6.7) and (6.8), when we have only one sensor present,

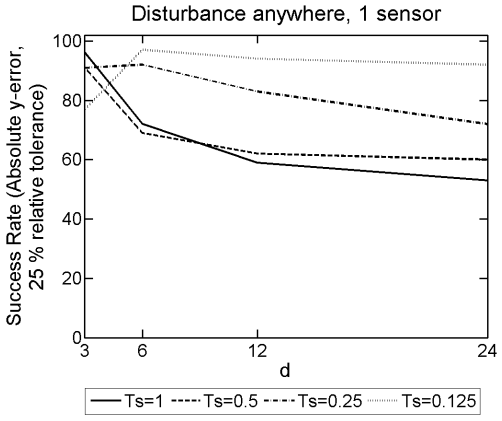
the success rate for higher values of d is weak except for when $T_s = 0.125$. This was also observed when we had an absolute x-error, see Figures (6.5) and (6.6). We are not sure why the success rate drops when we take more principal components in our SVD. Nevertheless, with only a single sensor present, as was observed in chapters 3, 4 and 5, we would not expect reasonable success rates due to the lack of information known to the KF. Therefore, the fact that we get reasonable success rates with only one sensor when $T_s = 0.125$ is remarkable.



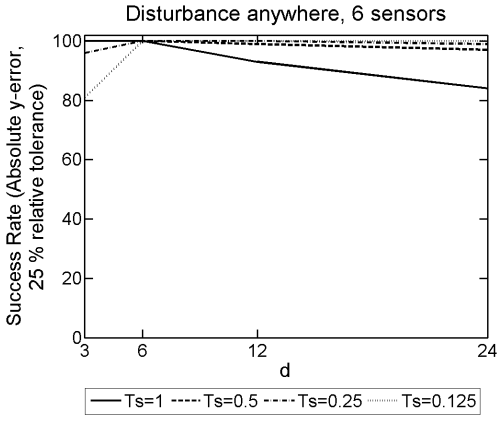
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

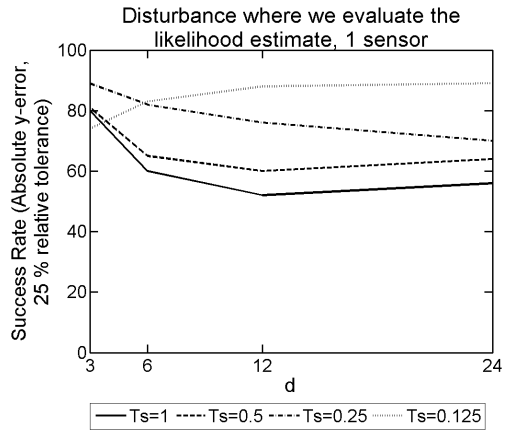


(c) One sensor present with random disturbance locations.

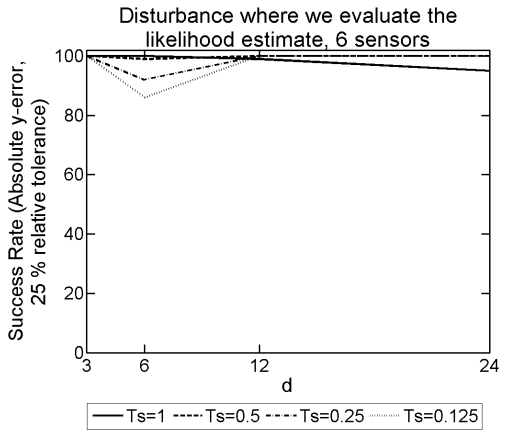


(d) Six sensors present with random disturbance locations.

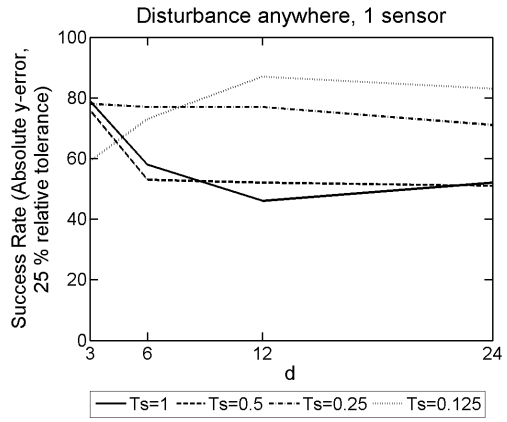
Figure 6.7: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 150$Hz given an absolute y-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps.



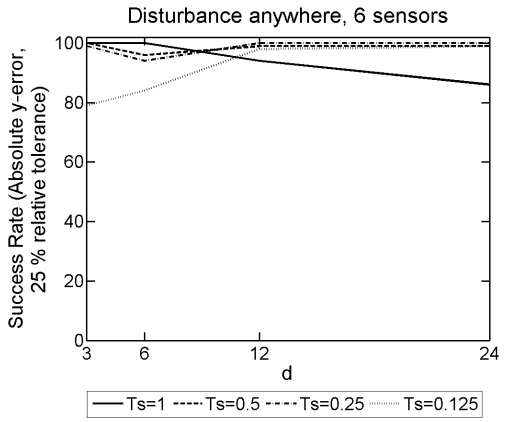
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.

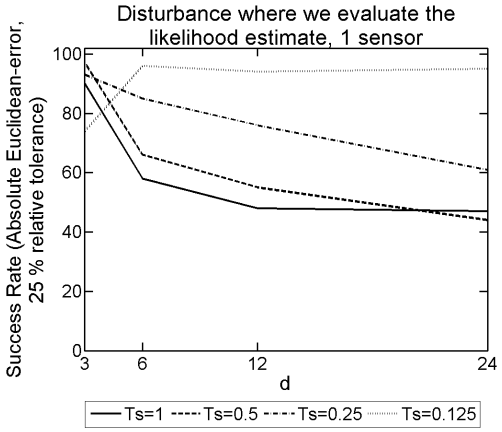


(d) Six sensors present with random disturbance locations.

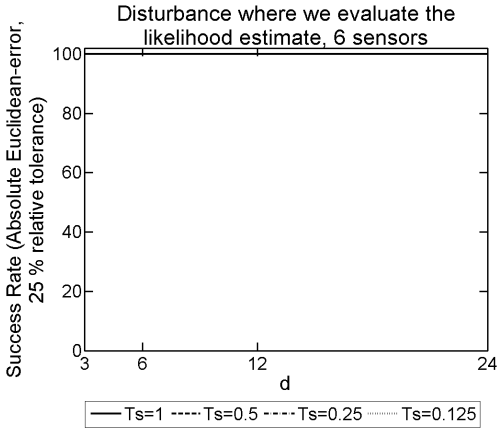
Figure 6.8: The success rate for our 2D model problem with 3NBCs with a disturbance frequency of $F = 150$Hz given an absolute y-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps.

Additionally, inspecting Figures (6.7) and (6.8) further, we observe that when we have six sensors present as opposed to only one, we get significantly better results irrespective of the values chosen for d or T_s. Furthermore, when the 100 disturbance locations used to generate our probabilistic results are random, the success rates are not as good as those observed when the disturbance locations are a random subset of the positions we evaluate the likelihood function in (2.107). We would expect this, and we have seen this before in chapters 3, 4 and 5. Since we are only evaluating the likelihood function in

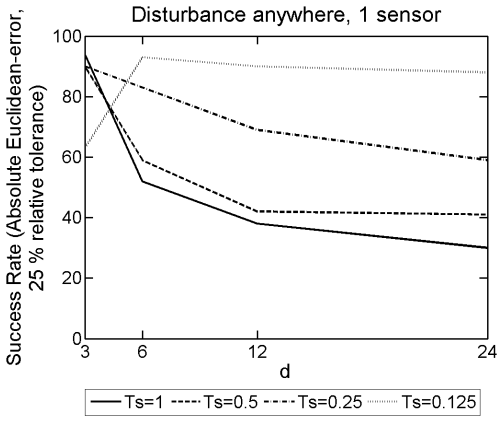
(2.107) at specific locations, if the disturbance location is random, we cannot predict the exact location of our disturbance which induces an unavoidable error using this approach without a minimisation algorithm.



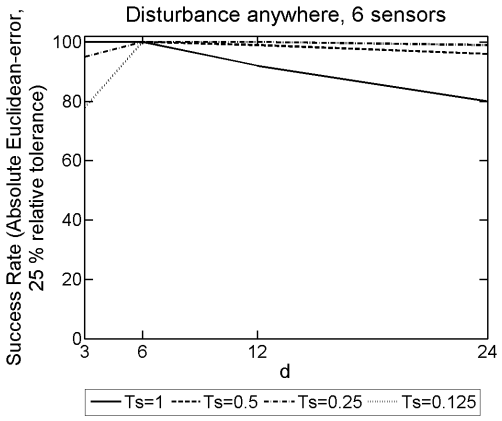
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

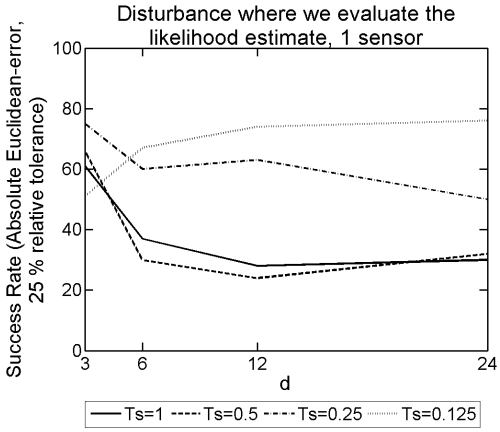


(c) One sensor present with random disturbance locations.

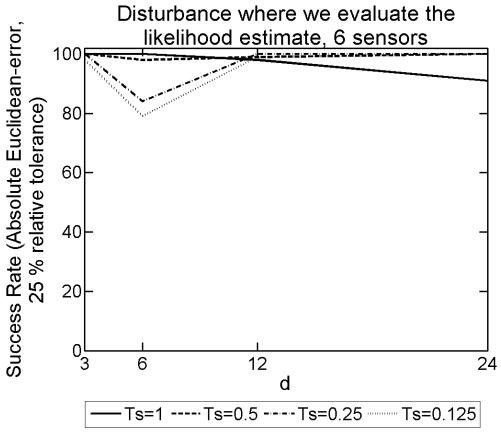


(d) Six sensors present with random disturbance locations.

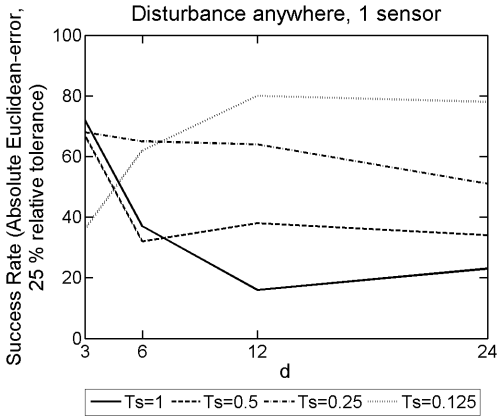
Figure 6.9: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 150$Hz given an absolute Euclidean-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps.



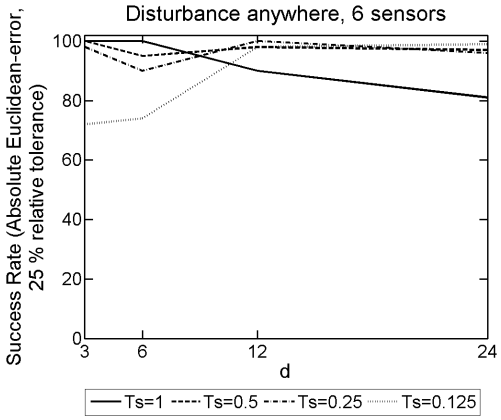
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



(d) Six sensors present with random disturbance locations.

Figure 6.10: The success rate for our 2D model problem with 3NBCs with a disturbance frequency of $F = 150$Hz given an absolute Euclidean-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 150$ by $M = 15$ nodes and $L = 18000$ discrete-time steps.

Lastly, we inspect Figures (6.9) and (6.10) which portray probabilistic success rates for an absolute Euclidean-error for our 2D model problems with 1NBC and 3NBCs, respectively. The Euclidean-error is dependent on both the x-error and y-error discussed already and as a result, we take a relative tolerance percentage of 25%. Additionally, we would expect the same patterns already observed in Figures (6.5), (6.6), (6.7) and (6.8) to be present in Figures (6.9) and (6.10).

We observe that having more sensors present results in better success rates, when

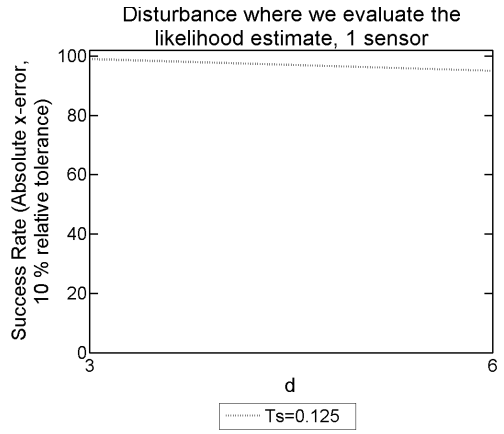
$T_s = 0.125$ with a single sensor present we obtain reasonable results. Furthermore, when the 100 disturbance locations are random, we get slightly lower success rates, as observed previously.

6.2.2 Disturbance frequency of $F = 300$Hz

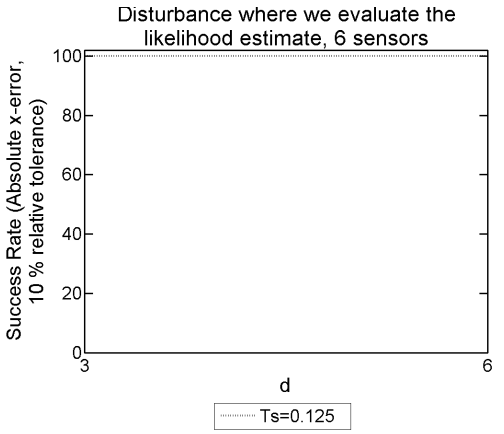
In this section, we look at the results obtained from our 2D model problems with 1NBC and 3NBCs for a disturbance frequency of $F = 300$Hz. For the full set of results, see Appendix D.3 and D.4 for our 2D model problems with 1NBC and 3NBCs, respectively.

The following set of figures display results which originate from our 2D model problems using an SVD, which is constructed using data from our explicit FDM approximations of u over a duration of T_s seconds, from which we take the first d principle components with the largest singular values. In the previous section, that is when our 2D model problems had a disturbance frequency of $F = 150$Hz, we produced results for $d := \{3, 6, 12, 24\}$ and $T_s := \{1, 0.5, 0.25, 0.125\}$. However, due to the amount of discrete-time steps involved when working with a disturbance frequency of $F = 300$Hz in 2D, the run-time requirement meant running for 16 combinations of d and T_s was not feasible. Therefore, having gained insight from the results corresponding to a disturbance frequency of $F = 150$Hz in the previous section, we decide to choose $d := \{3, 6\}$ and $T_s = 0.125$. This means we only have 2 combinations in total, and by choosing the smaller values for d and T_s, we are running our 2D model problems with the shortest run-time. To put the run-time requirements for all 16 combinations into perspective, it took six weeks using high-performance computers to obtain the results in this section which correspond to the two fastest cases.

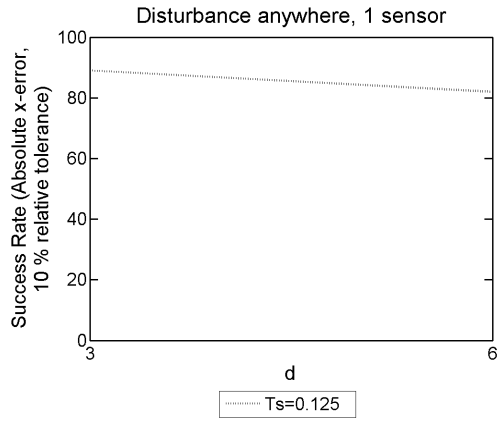
The figures in this section display the success rate of our 2D model problems. This success rate is calculated by running our 2D model problem 100 times, with a different disturbance location each time. To be classed as a success, they must be able to predict the disturbance location within an absolute error with a relative tolerance percentage. The cases presented in Figure (6.11) and (6.12) have one and six sensors present, whose success rate is determined from an absolute x-error with a relative tolerance percentage of 10%. Additional results corresponding to two, three, four and five sensors present with different absolute errors with a range of tolerance percentages can be found in Appendix D.3 and D.4.



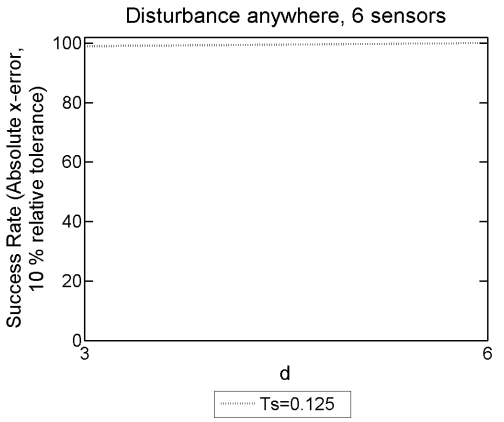
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.

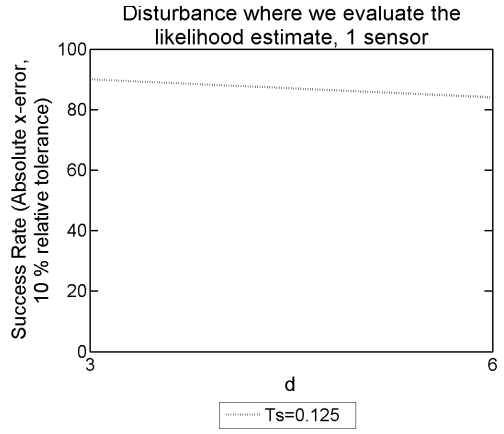


(d) Six sensors present with random disturbance locations.

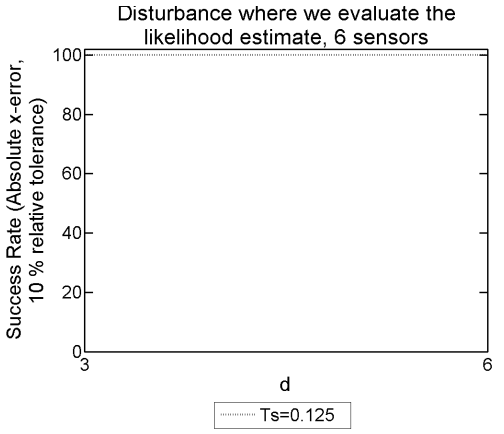
Figure 6.11: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 300$Hz given an absolute x-error with a relative tolerance of 10%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps.

Moving onto a discussion about the results in Figure (6.11) and Figure (6.12), we observe reasonable success rates when only one sensor is present for $T_s = 0.125$ and $d := \{3, 6\}$. We observed similar results in the previous section with a disturbance frequency of $F = 150$Hz. When we have six sensors present along the surface of our 2D domain, we achieve a success rate of 100%, irrespective of the value we chose for d and the type of disturbance locations used to generate our success rates. This is a very significant result, since choosing $T_s = 0.125$ means our SVD is based on a small subset of our explicit

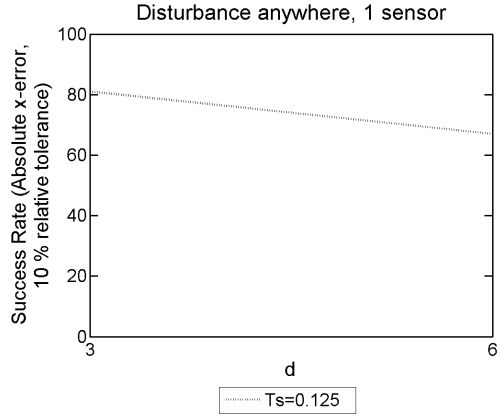
FDM approximations of u in (2.65). Again, similar results were observed in the previous section when $F = 150$Hz in Figure (6.5) and Figure (6.6) for the values of T_s and d used in this section.



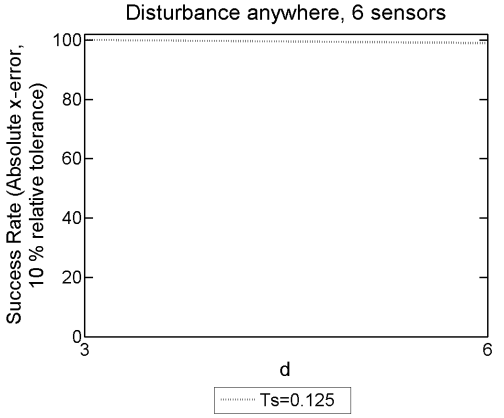
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.

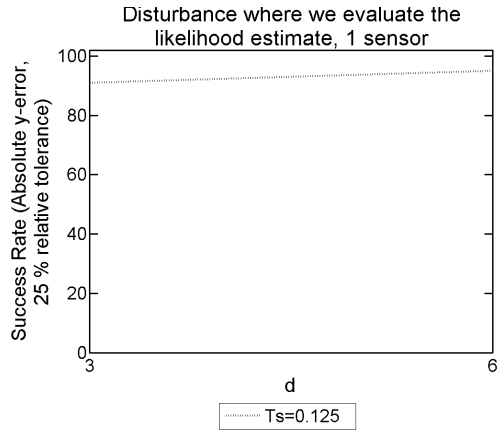


(d) Six sensors present with random disturbance locations.

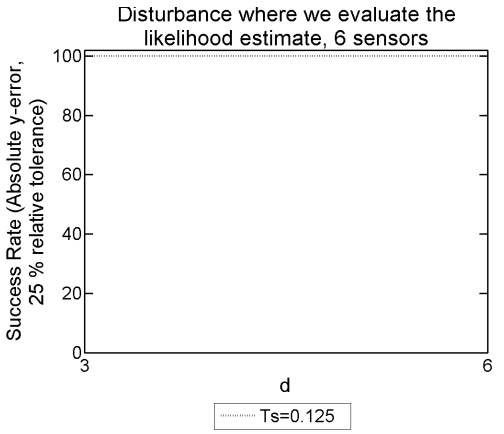
Figure 6.12: The success rate for our 2D model problem with 3NBCs with a disturbance frequency of $F = 300$Hz given an absolute x-error with a relative tolerance of 10%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps.

Figures (6.13) and (6.14) contain the probabilistic success rates for an absolute y-error, given a relative tolerance percentage of 25% for our 2D model problems with 1NBC and 3NBCs, respectively. Upon inspection of both figures, we observe better results when we have more sensors present and $d = 6$ principal components. Moreover, we find better

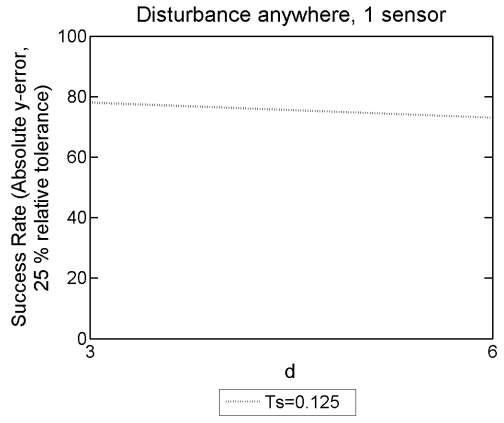
success rates for our 2D model problem with 1NBC, see Figure (6.13), when compared to our 2D model problem with 3NBCs, see Figure (6.14). Lastly, the results obtained using 100 random disturbance locations are not as good when we have only one sensor present or when $d = 3$ with six sensors present, when compared to the probabilistic success rates computed using 100 disturbances at a random subset of the positions we evaluate the likelihood function in (2.107).



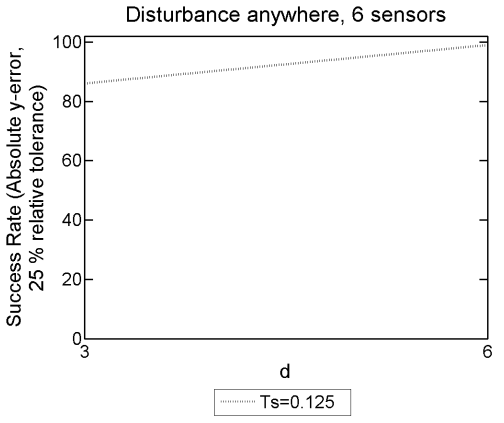
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

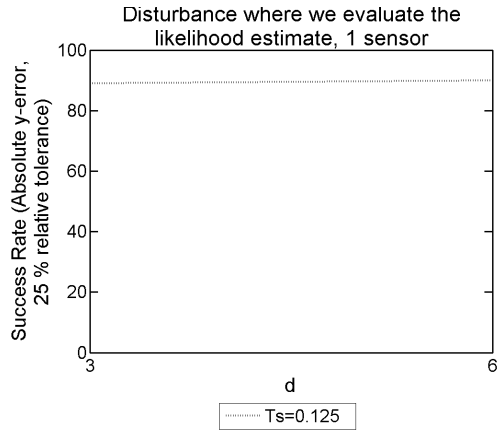


(c) One sensor present with random disturbance locations.

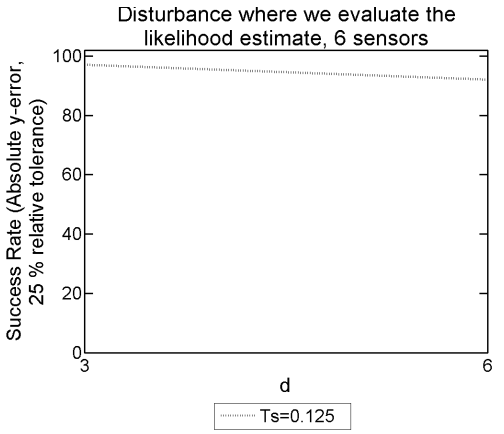


(d) Six sensors present with random disturbance locations.

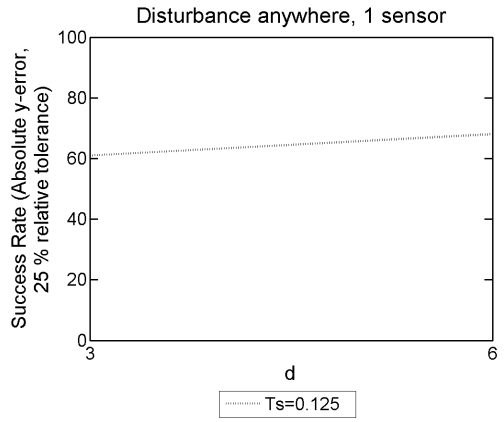
Figure 6.13: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 300$Hz given an absolute y-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps.



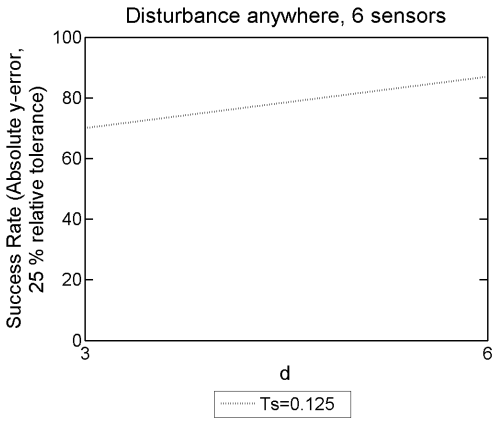
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



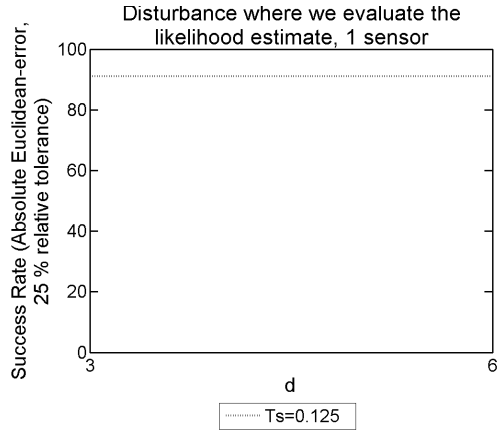
(d) Six sensors present with random disturbance locations.

Figure 6.14: The success rate for our 2D model problem with 3NBCs with a disturbance frequency of $F = 300$Hz given an absolute y-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps.

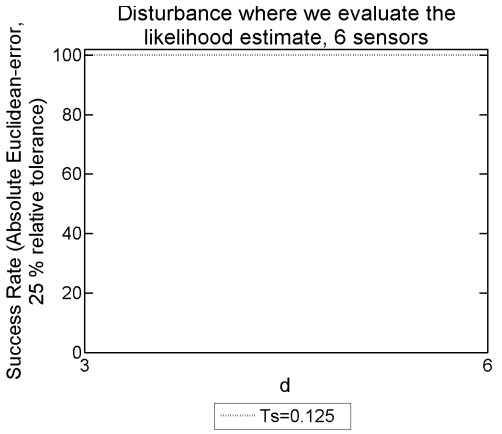
Figures (6.15) and (6.16) contain the probabilistic success rates for an absolute Euclidean-error, given a relative tolerance percentage of 25% for our 2D model problems with 1NBC and 3NBCs, respectively. Since the Euclidean-error is a combination of both the x-error and y-error, we would expect to observe the same in Figures (6.15) and (6.16) as we have already seen in Figures (6.11), (6.12), (6.13) and (6.14).

Upon inspection of Figures (6.15) and (6.16), we see that having a greater number of sensors present results in better probabilistic success rates. Moreover, we observe better

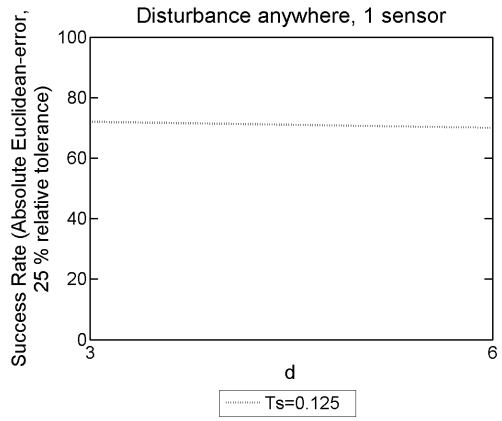
results when our 100 disturbances used to produce the probabilistic success rates are a random subset of the positions we evaluate the likelihood function in (2.107) when compared to having 100 random disturbance locations. And lastly, the probabilistic success rates obtained from our 2D model problem with 1NBC are better than those observed from our 2D model problem with 3NBCs.



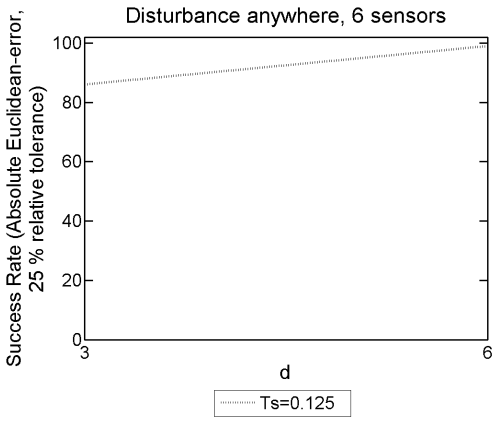
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.

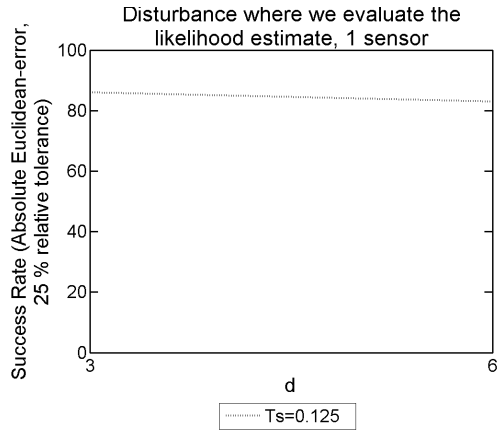


(c) One sensor present with random disturbance locations.

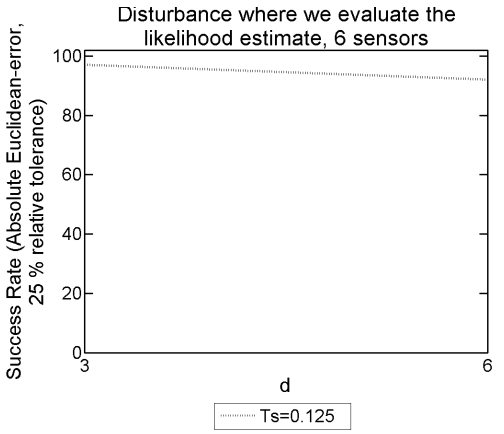


(d) Six sensors present with random disturbance locations.

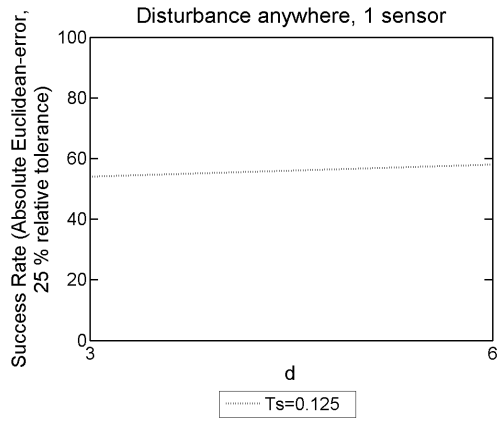
Figure 6.15: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 300$Hz given an absolute Euclidean-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps.



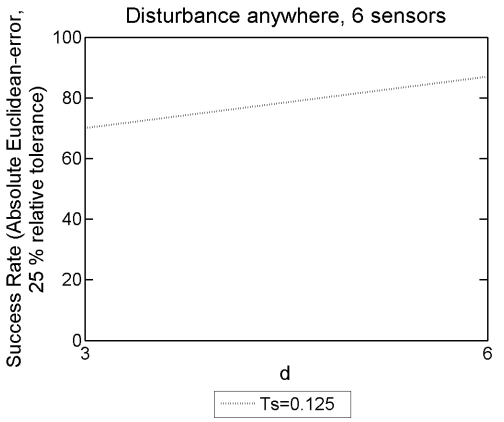
(a) One sensor present with disturbance locations where (2.107) is evaluated.



(b) Six sensors present with disturbance locations where (2.107) is evaluated.



(c) One sensor present with random disturbance locations.



(d) Six sensors present with random disturbance locations.

Figure 6.16: The success rate for our 2D model problem with 1NBC with a disturbance frequency of $F = 300$Hz given an absolute Euclidean-error with a relative tolerance of 25%. The probabilistic results were produced using an array of T_s and principal components, d, to form an SVD from explicit FDM approximations of u with a mesh density of $N = 500$ by $M = 50$ nodes and $L = 45000$ discrete-time steps.

6.3 Summary

In this chapter, we investigated the feasibility of increasing the frequency of our disturbance in the forcing function, see (1.6), to $F = 150$Hz and $F = 300$Hz. As discussed in the introduction, a partially blocked coronary artery emits a signal with a frequency in the range of $300 - 800$Hz.

Initially, we had to increase the Gaussian spread of our forcing function in our 2D domain to obtain better success rates, which was achieved by increasing the size of ϵ in (1.6) for both disturbance frequencies considered in this chapter.

Having done this, we observed good probabilistic success rates with only one sensor present when $T_s = 0.125$, irrespective of d, when we have a disturbance frequency of $F = 150$Hz. We expected that the value taken for d would have little impact on the success rate since the singular values in our SVD converge to zero quickly, and so, even when we take d to be small the resultant principle components contain a lot of the original information despite the reduced matrix sizes.

For both disturbance frequencies considered in this chapter, we observed better success rates when we had six sensors present as opposed to having only one sensor present, which we expected. In addition to this, we observed better results when we used 100 disturbance locations from a random subset of the positions we evaluate the likelihood function in (2.107), as opposed to having 100 random disturbance locations. This result is expected since we are not using a minimisation algorithm, and so in the worst-case scenario discussed previously, an unavoidable y-error of 14.1% arises. Lastly, the probabilistic success rates for our 2D model problem with 1NBC are better than the results obtained for our 2D model problem with 3NBCs. Again, this is expected since the latter 2D model problem is more complex.

The probabilistic success rates produced for our 2D model problems with a disturbance frequency of $F = 300$Hz used $d := \{3, 6\}$ and $T_s = 0.125$ rather than $d := \{3, 6, 12, 24\}$ and $T_s := \{1, 0.5, 0.25, 0.125\}$. The reason for this was due to the run-time requirement for our 2D model problems with a disturbance frequency of $F = 300$Hz, which even though we were using an SVD to reduce the matrix dimensions in our KF, there were still a significant amount of discrete-time steps.

A recommendation for this work in the future would be to terminate the KF early, reducing the run-time requirements, and computing the likelihood function in (2.107) using information from the KF over a subset of the total discrete-time steps. We explored this approach in chapter 5, and with a disturbance frequency of $F = 25$Hz, it is a feasible approach which can reduce the run-time requirement with little impact on the models success rate.

In the next chapter, we return to our 1D model problem and consider approximations of

$\partial u/\partial t$ as our sensor traces in addition to, and instead of the explicit FDM approximations of u.

Chapter 7

1D Model Problem: Different Sensor Traces

In this chapter, we revisit our 1D model problem in chapter 3, with a disturbance frequency of $F = 25$Hz in our forcing function, see (1.6). Previously, we used explicit FDM approximations of u in (1.1) for our sensor traces. In this chapter, we investigate the feasibility of using approximations of $\partial u/\partial t$ in addition to, and instead of, using explicit FDM approximations of u as our sensor traces.

As a result, we first outline the numerical approximations for $\partial u/\partial t$ and check the convergence rate using an exact solution for u. We then detail the required changes to the KF, and how we extract $\partial u/\partial t$ in (2.93).

Having derived an approximation of $\partial u/\partial t$ for every node in our mesh and discrete-time step, we outline our model problem and look at the random Gaussian noise we add to the approximations of $\partial u/\partial t$. Before discussing the results, we show a schematic which illustrates a step-by-step process used to generate our probabilistic success rates in this chapter.

Finally, we discuss the results obtained for our 1D model problem with a disturbance frequency of $F = 25$Hz using sensor traces corresponding to approximations of $\partial u/\partial t$, and sensor traces from approximations of both u and $\partial u/\partial t$. It was initially observed in chapter 3 that using an SVD to reduce the matrix dimensions in the KF still yields good results. Therefore, we deploy the same approach in this chapter to reduce the run-time requirements.

7.1 Approximating $\partial u/\partial t$

Recall that our 1D model problem has a domain defined by $\Omega_{1D} := (0, 0.2)$, see Figure (1.2), and we are interested in approximating the solution, denoted by $\partial u/\partial t$, for this problem.

We achieve this by discretising our domain, see Figures (2.1) and (2.2), into a mesh with $N+1$ nodes, and $L+1$ discrete-time steps. Recall from chapter 2 that the interval between nodes in space and time are denoted by $H_x := 0.2/N$ and $t_n := n\Delta t$ respectively, where $n \in \{0, 1, ..., L\}, \Delta t := T/L$ and T is the duration of our simulation.

We recall from chapter 2 that $u(x_i, t_n)$ denotes the solution of our PDE at node x_i and discrete-time step t_n, where $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, ..., L\}$. Extending this, let $\partial u(x_i, t_n)/\partial t$ denote an approximation of $\partial u/\partial t$ at node x_i and discrete-time step t_n.

Having now defined all relevant notation, we start by obtaining approximations of $\partial u/\partial t$ at every node in our mesh, across the discrete-time steps defined by t_n for $n \in \{1, 2, ..., L-1\}$. Using Taylor series, we expand u at nodes denoted by $u(x_i, t_{n+1}) \in \Omega_{1D} \times [0, T]$ and $u(x_i, t_{n-1}) \in \Omega_{1D} \times [0, T]$, to get

$$u(x_i, t_{n+1}) = u(x_i, t_n) + \Delta t u_t(x_i, t_n) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_n) + \mathcal{O}(\Delta t^3), \tag{7.1}$$

$$u(x_i, t_{n-1}) = u(x_i, t_n) - \Delta t u_t(x_i, t_n) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_n) + \mathcal{O}(\Delta t^3) \tag{7.2}$$

respectively, where $u_t \equiv \partial u/\partial t$. We now subtract (7.2) from (7.1) to simplify further and get

$$u_t(x_i, t_n) = \frac{u(x_i, t_{n+1}) - u(x_i, t_{n-1})}{2\Delta t} + \mathcal{O}(\Delta t^2). \tag{7.3}$$

To obtain an approximation of $\partial u/\partial t$ across all nodes in our mesh at discrete-time steps, defined by t_n for $n \in \{1, 2, ..., L-1\}$, we need to neglect the higher-order terms in (7.3). Therefore, by denoting $\tilde{V}_i^n \approx \partial u(x_i, t_n)/\partial t$, we get

$$\tilde{V}_i^n \approx \frac{u(x_i, t_{n+1}) - u(x_i, t_{n-1})}{2\Delta t}. \tag{7.4}$$

We still need to obtain an approximation of $\partial u/\partial t$ across every node in our mesh at discrete-time steps denoted by t_n for $n \in \{0, L\}$. We know from our initial conditions $u_t(x_i, t_0)$ for $i \in \{1, 2, ..., N+1\}$. Using this, we only need to approximate $\partial u(x_i, t_L)/\partial t$ for $i \in \{1, 2, ..., N+1\}$. Using Taylor series again, we can expand u at nodes denoted by $u(x_i, t_{L-1}) \in \Omega_{1D} \times [0, T]$ and $u(x_i, t_{L-2}) \in \Omega_{1D} \times [0, T]$, to get

$$u(x_i, t_{L-1}) = u(x_i, t_L) - \Delta t u_t(x_i, t_L) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_L) + \mathcal{O}(\Delta t^3), \tag{7.5}$$

$$u(x_i, t_{L-2}) = u(x_i, t_L) - 2\Delta t u_t(x_i, t_L) + \frac{4\Delta t^2}{2} u_{tt}(x_i, t_L) + \mathcal{O}(\Delta t^3). \tag{7.6}$$

We now multiply (7.5) by four, and subtract (7.6) from the resultant equation. After some rearranging, we get

$$u_t(x_i, t_L) = \frac{u(x_i, t_{L-2}) - 4u(x_i, t_{L-1}) + 3u(x_i, t_L)}{2\Delta t} + \mathcal{O}(\Delta t^2). \tag{7.7}$$

Recall that we are trying to obtain an approximation for $\partial u(x_i, t_L)/\partial t$ across every node in our mesh. By neglecting the higher-order terms in (7.7), we achieve this. By denoting $\tilde{V}_i^n \approx \partial u(x_i, t_n)/\partial t$, we get

$$\tilde{V}_i^L \approx \frac{u(x_i, t_{L-2}) - 4u(x_i, t_{L-1}) + 3u(x_i, t_L)}{2\Delta t}. \tag{7.8}$$

Recall, from chapter 2, our explicit FDM approximations of u in (2.16) for our 1D model problem, denoted by $\mathbf{V}^n$, across all nodes in our mesh at the discrete-time step t_n. Using an initial condition and our explicit FDM approximations of u in both (7.4) and (7.8), we are able to obtain approximations for $\partial u(x_i, t_n)/\partial t$ at every node in our mesh across all discrete-time steps.

Therefore, we are now able to construct a matrix system using an initial condition and the approximations of $\partial u(x_i, t_n)/\partial t$ for $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, ..., L\}$. Recall that our 1D domain, denoted by $\Omega_{1\mathrm{D}}$, has $N+1$ nodes and so we can define $\tilde{\mathbf{V}}^n$ as

$$\tilde{\mathbf{V}}^n = \begin{pmatrix} \tilde{V}_1^n \\ \tilde{V}_2^n \\ \vdots \\ \tilde{V}_{N+1}^n \end{pmatrix}.$$

Applying this to our initial condition, denoted by $u_t(x_i, t_0)$ for $i \in \{1, 2, ..., N+1\}$, (7.4) and (7.8), we are able to construct three matrix systems to approximate $\partial u/\partial t$ at every node in our mesh, across all discrete-time steps. Therefore, we have

$$\tilde{\mathbf{V}}^0 = \begin{pmatrix} u_t(x_1, t_0) \\ u_t(x_2, t_0) \\ \vdots \\ u_t(x_{N+1}, t_0) \end{pmatrix}, \tag{7.9}$$

$$\tilde{\mathbf{V}}^n \approx \frac{\mathbf{V}^{n+1} - \mathbf{V}^{n-1}}{2\Delta t}, \tag{7.10}$$

$$\tilde{\mathbf{V}}^L \approx \frac{\mathbf{V}^{L-2} - 4\mathbf{V}^{L-1} + 3\mathbf{V}^L}{2\Delta t} \tag{7.11}$$

for $n \in \{1, 2, ..., L-1\}$.

7.1.1 MATLAB Implementation

In this section, we look at verifying whether our MATLAB code used to evaluate $\partial u/\partial t$ using (7.9), (7.10) and (7.11) works as intended. We achieve this by computing the maximum absolute error between an exact solution of $\partial u/\partial t$, from an exact solution of our PDE, and our approximation of $\partial u/\partial t$ in (7.9), (7.10) and (7.11) across every node in our mesh and all discrete-time steps.

As discussed previously, to approximate $\partial u/\partial t$ using (7.10) and (7.11), we require our explicit FDM approximations of u in (2.16). As a result, we must first construct an exact solution for u that satisfies all boundary conditions in our 1D model problem and subsequently obtain an exact solution for $\partial u/\partial t$. We then need to obtain $\mathbf{V}^0$ and $\mathbf{V}^1$ from our initial conditions to compute our explicit FDM approximations of u in (2.16).

Manufactured space and time-dependent solution

We want to choose an exact solution for u that satisfies all boundary conditions in our 1D model problem, denoted by $\partial\Omega_{\text{1D}}$. Therefore, we reuse the exact solution of our PDE, denoted by u, outlined in chapter 2 for our 1D model problem in (2.17), and subsequently use (2.18) as our forcing function, f, in (1.1) for this convergence test. Using (2.17), we deduce our exact solution of $\partial u/\partial t$, to be

$$\frac{\partial u(x_i, t_n)}{\partial t} = \sin\left(\frac{2\pi x_i}{0.8}\right)\cos(t_n) \tag{7.12}$$

where $i \in \{1, 2, ..., N+1\}$ and $n \in \{0, 1, ..., L\}$.

Recall the matrix system in (2.16) which approximates u in (1.1) at every node in our mesh across all discrete-time steps. This matrix system is recursive and so initially requires both $\mathbf{V}^0$ and $\mathbf{V}^1$ to be known. That is we need a solution of our PDE, denoted by u, at $t_0 := 0$ and $t_1 := \Delta t$, respectively. We have two initial conditions given by $u(x_i, t_0)$ and $u_t(x_i, t_0)$ for $i \in \{1, 2, ..., N+1\}$. Therefore, $\mathbf{V}^0$ is known. However, to get $\mathbf{V}^1$, we need to use Taylor series and both initial conditions.

In this convergence test, we know the exact solution to u and $\partial u/\partial t$, and so by using these we know that both initial conditions across all nodes in our mesh, defined for all $i \in \{1, 2, \cdots, N+1\}$, are both

$$u(x_i, t_0) = u_t(x_i, t_0) = \sin\left(\frac{2\pi x_i}{0.8}\right). \tag{7.13}$$

Therefore, directly from (7.13), $\mathbf{V}^0 = \sin\left(\frac{2\pi x_i}{0.8}\right)$ for all $i \in \{1, 2, ..., N+1\}$. Using Taylor series in conjunction with (7.13), we get

$$\begin{aligned} u(x_i, t_1) &= u(x_i, t_0) + \Delta t u_t(x_i, t_0) + \mathcal{O}(\Delta t^2), \\ &= \sin\left(\frac{2\pi x_i}{0.8}\right)(1 + \Delta t) + \mathcal{O}(\Delta t^2). \end{aligned}$$

By neglecting the higher-order terms, we get $\mathbf{V}^1 \approx \sin\left(\frac{2\pi x_i}{0.8}\right)(1 + \Delta t)$ for all $i \in \{1, 2, ..., N+1\}$.

Error convergence

Recall that we can evaluate $\partial u/\partial t$ using the three matrix systems in (7.9), (7.10) and (7.11). Since the latter two are approximations, we have an error linked to the higher-order terms disregarded when approximating u in (2.16) and when approximating $\partial u/\partial t$ in (7.10) and (7.11). These higher-order terms can be seen in (2.7), (2.14), (7.3) and (7.7). Therefore, the combined error is $\mathcal{O}(H_x^3 + \Delta t^2)$.

N	Δt	Maximum |Error|	Error Percentage†	Error Convergence	Computation Time†† (s)
8	1500^{-1}	0.028672	14.63374	N/A	0.003117
16	3000^{-1}	0.007162	0.748061	4.003351019	0.006903
32	6000^{-1}	0.001790	1.219832	4.001117318	0.021056
64	12000^{-1}	0.000447	0.099515	4.004474273	0.072011
128	24000^{-1}	0.000112	0.019094	3.991071429	0.276281
256	48000^{-1}	0.000028	0.004281	4.0	1.059496

† Error Percentage = [maximum error at node (i, j)]/[approximation at node (i, j)] $\times$ 100.

†† Computed using an i5 − 4590 @ 3.30GHz, 16GB RAM @ 1333 MHz and Intel HD Graphics 4600.

Table 7.1: The error convergence of our FDM approximations of $\partial u/\partial t$ using (7.9), (7.10) and (7.11) for our 1D acoustic wave equation.

Therefore, we would expect the maximum absolute error to decrease by a factor of four as both H_x and Δt half. Table (7.1) displays the maximum absolute error across all nodes in our mesh and discrete-time steps between an exact solution of $\partial u/\partial t$ in (7.12), derived using an exact solution of our PDE, and our approximations of $\partial u/\partial t$ for increasing values of N, which subsequently decrease H_x, and decreasing values for Δt. We can deduce that

as N doubles, and Δt halves the error convergence approaches four which is what we expected. Therefore, we can conclude that the mathematics set-out in this section and the code are correct and working.

7.1.2 Using $\partial u/\partial t$ in the Kalman filter

The KF in (2.93)-(2.98) uses sensor traces, denoted by $\tilde{\mathbf{y}}^n$, to predict the latent state, denoted by $\underline{\tilde{\mathbf{x}}}^{n|n-1}$, which is modelled after our explicit FDM approximations of u in (2.16) for a specific disturbance location, denoted by x_0, in our forcing function. In (2.93), the error between our sensor traces and the KFs prediction of the latent state is calculated at each discrete-time step.

In this chapter, we are looking at the possibility of using our approximations of $\partial u/\partial t$ for our sensor traces. As a result, we must be able to approximate $\partial u/\partial t$ in (2.93) using only the KFs prediction of the latent state, denoted by $\underline{\hat{\mathbf{x}}}^{n|n-1}$. We achieve this by altering the matrices used to set up the KF in (2.91)-(2.92).

Keeping the notation the same in both (2.91) and (2.92) for simplicity, we redefine the matrices as follows

$$\mathbf{A} := \begin{pmatrix} \mathbf{B} & -\mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \end{pmatrix}, \underline{\tilde{\mathbf{x}}}^n := \begin{pmatrix} \mathbf{V}^n \\ \mathbf{V}^{n-1} \\ \mathbf{V}^{n-2} \end{pmatrix}, \tilde{\boldsymbol{\mathcal{F}}}^n := \begin{pmatrix} \boldsymbol{\mathcal{F}}^n \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \tilde{\mathbf{w}}^n := \begin{pmatrix} \mathbf{w}^n \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix},$$

$$\tilde{\mathbf{y}}^n := \begin{pmatrix} \mathbf{y}^n \\ \mathbf{y}^{n-1} \\ \mathbf{y}^{n-2} \end{pmatrix}, \tilde{\mathbf{z}}^n := \begin{pmatrix} \mathbf{z}^n \\ \mathbf{z}^{n-1} \\ \mathbf{z}^{n-2} \end{pmatrix}$$

and $\tilde{\mathbf{C}}$ which is a sparse matrix with dimensions of $\big(3 \times \text{number of sensors}\big)$ by $3(N+1)$. To be able to compute the error in (2.93) between our sensor traces originating from our approximations of $\partial u/\partial t$, and the KFs prediction of these stored in $\underline{\hat{\mathbf{x}}}^{n|n-1}$, we need to find a way of approximating $\partial u(x_i, t_n)/\partial t$, $\partial u(x_i, t_{n-1})/\partial t$ and $\partial u(x_i, t_{n-2})/\partial t$ for $i \in \{1, 2, ..., N+1\}$ using only $\mathbf{V}^n$, $\mathbf{V}^{n-1}$ and $\mathbf{V}^{n-2}$ where $n \in \{2, 3..., L\}$. This involves deriving a different numerical scheme for each approximation of $\partial u/\partial t$.

We start by considering $\partial u(x_i, t_n)/\partial t$ for $i \in \{1, 2, ..., N+1\}$ and $n \in \{2, 3..., L\}$. Using Taylor series we can manipulate the solution of our PDE, denoted by u, at nodes denoted

by $(x_i, t_{n-1}) \in \Omega_{1D} \times [0, T]$ and $(x_i, t_{n-2}) \in \Omega_{1D} \times [0, T]$ to approximate $\partial u(x_i, t_n)/\partial t$. Therefore, initially we get

$$u(x_i, t_{n-1}) = u(x_i, t_n) - \Delta t u_t(x_i, t_n) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_n) + \mathcal{O}(\Delta t^3), \tag{7.14}$$

$$u(x_i, t_{n-2}) = u(x_i, t_n) - 2\Delta t u_t(x_i, t_n) + \frac{4\Delta t^2}{2} u_{tt}(x_i, t_n) + \mathcal{O}(\Delta t^3). \tag{7.15}$$

We now multiply (7.14) by four and subtract (7.15) from the resultant equation. After some simplifying, we get

$$u_t(x_i, t_n) = \frac{u(x_i, t_{n-2}) - 4u(x_i, t_{n-1}) + 3u(x_i, t_n)}{2\Delta t} + \mathcal{O}(\Delta t^2) \tag{7.16}$$

where $\partial u(x_i, t_n)/\partial t \equiv u_t(x_i, t_n)$. By neglecting the higher-order terms in (7.16), and recalling that both $\tilde{\mathbf{V}}^n \approx u_t(x_i, t_n)$ and $\mathbf{V}^n \approx u(x_i, t_n)$ for $i \in \{1, 2, ..., N+1\}$, we get

$$\tilde{\mathbf{V}}^n \approx \frac{\mathbf{V}^{n-2} - 4\mathbf{V}^{n-1} + 3\mathbf{V}^n}{2\Delta t} \tag{7.17}$$

for $n \in \{2, 3, ..., L\}$.

Next, we look at approximating $\partial u(x_i, t_{n-1})/\partial t$ as outlined above. Using Taylor series, we can use the value of u at nodes denoted by $u(x_i, t_n) \in \Omega_{1D} \times [0, T]$ and $u(x_i, t_{n-2}) \in \Omega_{1D} \times [0, T]$ to approximate $\partial u(x_i, t_{n-1})/\partial t$. Therefore, we get

$$u(x_i, t_n) = u(x_i, t_{n-1}) + \Delta t u_t(x_i, t_{n-1}) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_{n-1}) + \mathcal{O}(\Delta t^3), \tag{7.18}$$

$$u(x_i, t_{n-2}) = u(x_i, t_{n-1}) - \Delta t u_t(x_i, t_{n-1}) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_{n-1}) + \mathcal{O}(\Delta t^3) \tag{7.19}$$

for $n \in \{2, 3, ..., L\}$. Subtracting (7.19) from (7.18) and simplifying further, we get

$$u_t(x_i, t_{n-1}) = \frac{u(x_i, t_n) - u(x_i, t_{n-2})}{2\Delta t} + \mathcal{O}(\Delta t^2). \tag{7.20}$$

By neglecting the higher-order terms in (7.20), and recalling that $\tilde{\mathbf{V}}^{n-1} \approx u_t(x_i, t_{n-1})$ and $\mathbf{V}^n \approx u(x_i, t_n)$ for $i \in \{1, 2, ..., N+1\}$, we get

$$\tilde{\mathbf{V}}^{n-1} \approx \frac{\mathbf{V}^n - \mathbf{V}^{n-2}}{2\Delta t} \tag{7.21}$$

for $n \in \{2, 3, ..., L\}$.

Lastly, we look at approximating $\partial u(x_i, t_{n-2})/\partial t$. Using Taylor series again, we can use the value of u at nodes denoted by $u(x_i, t_{n-1}) \in \Omega_{1D} \times [0, T]$ and $u(x_i, t_n) \in \Omega_{1D} \times [0, T]$ to approximate $\partial u(x_i, t_{n-2})/\partial t$. Therefore, initially, we get

$$u(x_i, t_{n-1}) = u(x_i, t_{n-2}) + \Delta t u_t(x_i, t_{n-2}) + \frac{\Delta t^2}{2} u_{tt}(x_i, t_{n-2}) + \mathcal{O}(\Delta t^3), \tag{7.22}$$

$$u(x_i,t_n)=u(x_i,t_{n-2})+2\Delta t u_t(x_i,t_{n-2})+\frac{4\Delta t^2}{2}u_{tt}(x_i,t_{n-2})+\mathcal{O}(\Delta t^3) \tag{7.23}$$

for $i\in\{1,2,...,N+1\}$ and $n\in\{2,3,...,L\}$. We now multiply (7.22) by four and subtract (7.23) from the resultant equation. After some simplifying, we get

$$u_t(x_i,t_{n-2})=\frac{-3u(x_i,t_{n-2})+4u(x_i,t_{n-1})-u(x_i,t_n)}{2\Delta t}+\mathcal{O}(\Delta t^3). \tag{7.24}$$

By neglecting the higher-order terms in (7.24), and recalling that $\tilde{\mathbf{V}}^{n-2}\approx u_t(x_i,t_{n-2})$ and $\mathbf{V}^n\approx u(x_i,t_n)$ for $i\in\{1,2,...,N+1\}$, we get

$$\tilde{\mathbf{V}}^{n-2}\approx\frac{-3\mathbf{V}^{n-2}+4\mathbf{V}^{n-1}-\mathbf{V}^n}{2\Delta t} \tag{7.25}$$

for $n\in\{2,3,...,L\}$.

Having derived numerical schemes to approximate $\partial u(x_i,t_n)/\partial t$, $\partial u(x_i,t_{n-1})/\partial t$ and $\partial u(x_i,t_{n-2})/\partial t$ in (7.17), (7.21) and (7.25), respectively, we are able to compute (2.93) in the KF for sensor traces, denoted by $\tilde{\mathbf{y}}^n$, originating from our approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11).

As outlined previously in this chapter, we plan to use an SVD to reduce the matrix dimensions in the KF. The modified KF in (2.127)-(2.132) is derived from the larger matrices outlined in this section within the standard KF (2.93)-(2.98). This derivation follows the same steps outlined in chapter 2, and so, we will not repeat it here.

7.2 Model problem outline

7.2.1 Forcing function

In chapter 1, we outlined a forcing function in (1.6) which attempts to mimic a disturbance caused by CAD. In this chapter, we use this forcing function in (1.1) with the amplitude, given by $A=10^6$, ensuring that (2.128) is an invertible matrix. In previous chapters, we altered the Gaussian spread of our forcing function by changing ϵ in (1.6) to improve our success rates. We do the same for our 1D model problem considered in this chapter.

Figure (3.1) shows the Gaussian spread of our forcing function in (1.6) used in this chapter when $\epsilon=5.77\times10^{-6}$. In the worst-case scenario, that is when the distance between the actual disturbance location and a position we evaluate the likelihood function

in (2.107) is as larger as possible, 50% of the Gaussian spreads peak remains at the nearest position (2.107) is evaluated. At a distance of 0.00815 away from the disturbance location, that is 4% of our domain, the Gaussian spread of our forcing function decays to approximately zero.

We now want to investigate the convergence of the maximum absolute approximation of $\partial u/\partial t$. To do this, we have to solve (2.16), which requires both $\mathbf{V}^0$ and $\mathbf{V}^1$ to be known. Both were defined in chapter 3 and remain the same for our convergence testing in this chapter.

Table (7.2) shows the convergence of the maximum absolute approximation of $\partial u/\partial t$ for every node in our mesh across all discrete-time steps using (7.9), (7.10) and (7.11) for a disturbance frequency of $F = 25$Hz, simulation duration of $T = 3$ seconds and a disturbance location at $x_0 = 0.05$. Upon inspection of Table (7.2), we decide to use the approximation of $\partial u/\partial t$ from a coarse mesh with $N = 50$ nodes and $L = 9000$ discrete-time steps, resulting in an error of 20% when compared to a converged solution.

N	Δt	Maximum $\lvert\partial u/\partial t\rvert$ over all nodes and discrete-time steps	% from $N = 1000$
50	9000^{-1}	9590.2054890	20.1729975
100	18000^{-1}	12950.243204	7.79530203
150	27000^{-1}	12464.592259	3.75283815
200	36000^{-1}	12272.933639	2.15750913
250	45000^{-1}	12177.195243	1.36060137
300	54000^{-1}	12124.196074	0.91944661
350	63000^{-1}	12092.045100	0.65182816
400	72000^{-1}	12071.124817	0.47769180
450	81000^{-1}	12056.765306	0.35816603
500	90000^{-1}	12046.523110	0.27291199
1000	180000^{-1}	12013.736184	0

Table 7.2: The convergence for our approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11) over a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 25$Hz and a disturbance location at $x_0 = 0.05$.

In this chapter, we use sensor traces in the KF originating from our approximations

of $\partial u/\partial t$. In addition to these, we investigate the scenario where we have sensor traces from the approximations of both u and $\partial u/\partial t$. The convergence testing for our explicit FDM approximations of u can be found in Table (3.2).

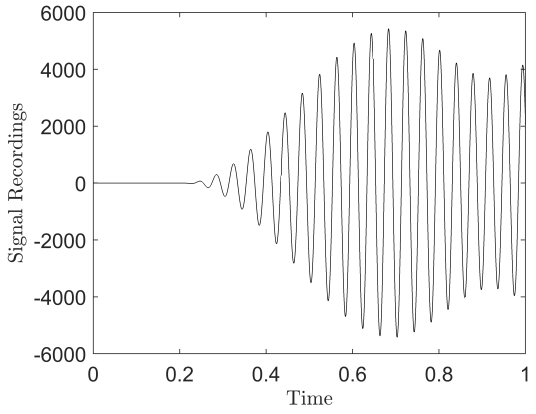
7.2.2 Noise added

As mentioned already for other model problems considered in this thesis, we add two different forms of random Gaussian noise to our approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11), denoted by $\tilde{\mathbf{w}}^n$ and $\tilde{\mathbf{z}}^n$ in the KF. The first represents the error associated with the approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11).
For our 1D model problem considered in this chapter, this error is measured by the combination of the higher-order terms neglected in (2.7), (2.14), (7.3) and (7.7).

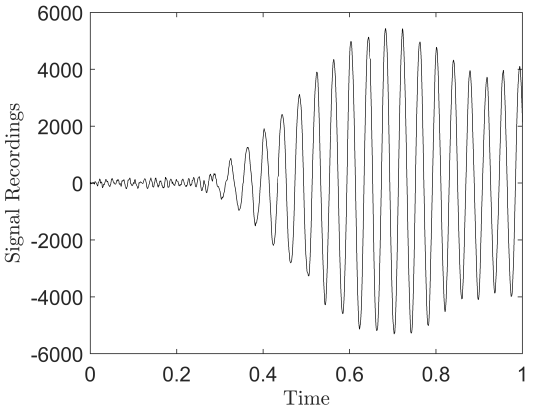
The other random Gaussian noise added represents inaccuracies recorded by sensors in a real-life scenario, be that ambient noise in the background or errors due to manufacturing defects. Without experimental data, we cannot accurately predict the magnitude of this noise. As a result, we model it after the noise corresponding to the error associated with the approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11), but scale it ensuring it is the larger of the two random Gaussian noises added since we would expect this noise to be the dominant of the two.

Figure (7.1) illustrates the effect the added noise has on our approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11). These figures show only the first second of our $T = 3$ second simulation and are formed using a mesh density of $N = 50$ nodes with $L = 9000$ discrete-time steps, a disturbance frequency of $F = 25$Hz, and a single sensor trace located at $x = 0.2$. We scale the random Gaussian noise added to our approximations of $\partial u/\partial t$, see Figure (7.1)(b) by 5×10^4, and by 6×10^5 in Figure (7.1)(c). In total, the noise added is significant, see Figure (7.1)(d), but does not overwhelm our approximations of $\partial u/\partial t$, see Figure (7.1)(a).

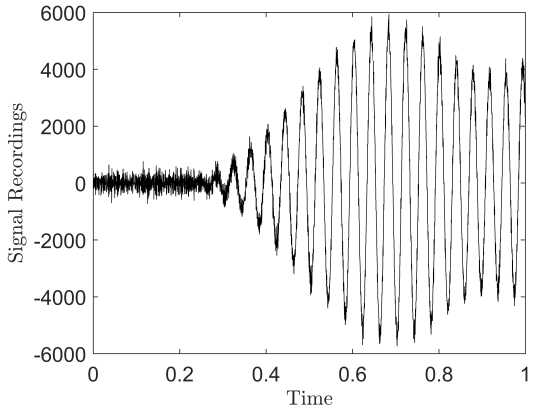
As mentioned in the previous chapter, we consider in addition to sensor traces originating from only $\partial u/\partial t$, sensor traces from our approximations of both u and $\partial u/\partial t$. The noise added to the sensor traces from our explicit FDM approximations of u used in this chapter can be observed in Figure (3.2).



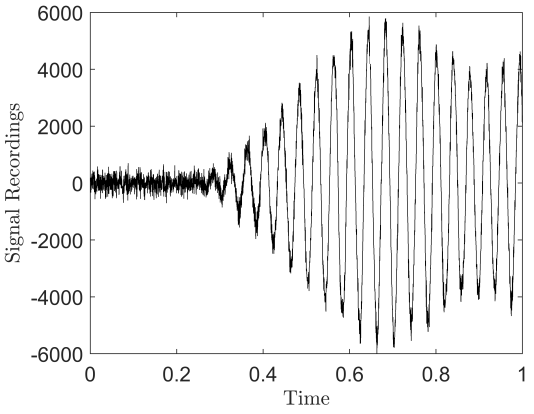
(a) No added noise.



(b) Noise added to mimic the error associated with the approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11).



(c) Noise added to mimic errors recorded in real-world scenarios.



(d) All added noise present.

Figure 7.1: Illustration of the noise added to our approximations of $\partial u/\partial t$ in (7.9), (7.10) and (7.11) using a single sensor trace at $y := 0.2$, a mesh dimension of $N = 50$ nodes and $L = 9000$ discrete-time steps, a simulation duration of $T = 3$ seconds, a disturbance frequency of $F = 25$Hz and a disturbance location at $x_0 = 0.004$. All figures show only the first second of the sensor traces.

7.2.3 Model problem schematic

In this section, we outline the steps taken by our 1D model problem in this chapter to obtain the probabilistic success rates displayed in the next section. We start by defining the locations our sensors will be placed, that is the nodes in our mesh where the explicit

FDM approximations of u and our approximations of $\partial u/\partial t$ are recorded. We will then show a schematic, outlining the steps from start to finish, enabling our model to predict the disturbance location.

In Figure (3.3) we outline the sensor positions on our 1D domain, dependent on the quantity present. We always place a sensor at $x = 0.2$, on the Neumann boundary. When we have more than a single sensor, the rest are equidistant from one another. There was no point in placing a sensor at $x = 0$ as the boundary condition here is equal to zero. As illustrated in previous chapters, having more than six sensors present yields little improvement in our probabilistic success rates given the addition run-time requirements. Therefore, all the results presented in this chapter correspond to having 1-6 sensors present.

Figure (7.2) shows a schematic illustrating the steps from start to finish for our 1D model problem considered in this chapter using an SVD, for sensor traces originating from the approximations of $\partial u/\partial t$ only. The first step has two different routes. One route involves creating 50 random disturbance locations in our 1D domain defined on $\Omega_{1\text{D}} := (0, 0.2)$. When these disturbance locations were created, duplicates were removed and replaced by another random disturbance location until all 50 disturbance locations were different. The second route in our first step involves taking 50 disturbance locations at the same positions we evaluate the likelihood function in (2.107), that is $x_0 := \{0.004, 0.008, ..., 0.2\}$. In both cases, the disturbance locations are generated once and are reused to ensure consistency between our probabilistic results.

In step 2 we solve our acoustic wave equation in (1.1) using the explicit FDM approximations of u in (2.16) over a simulation duration of $T = 2$ seconds. For more information on the simulation duration used in this chapter, see Figure (3.14). We store the approximations of u at every node in our mesh across all discrete-time steps.

In step 3, we use the explicit FDM approximations of u at every node in our mesh over a time duration, denoted by $T_s := \{1, 0.5, 0.25, 0.125\}$ seconds, to form a matrix which we decompose using the SVD procedure outlined in chapter 2. Having decomposed our matrix, we take the first d columns from the matrix containing the left singular vectors in (2.108). These d columns correspond to the d largest singular values, and represent our principal components which are used to reduce the size of our matrix system in the KF.

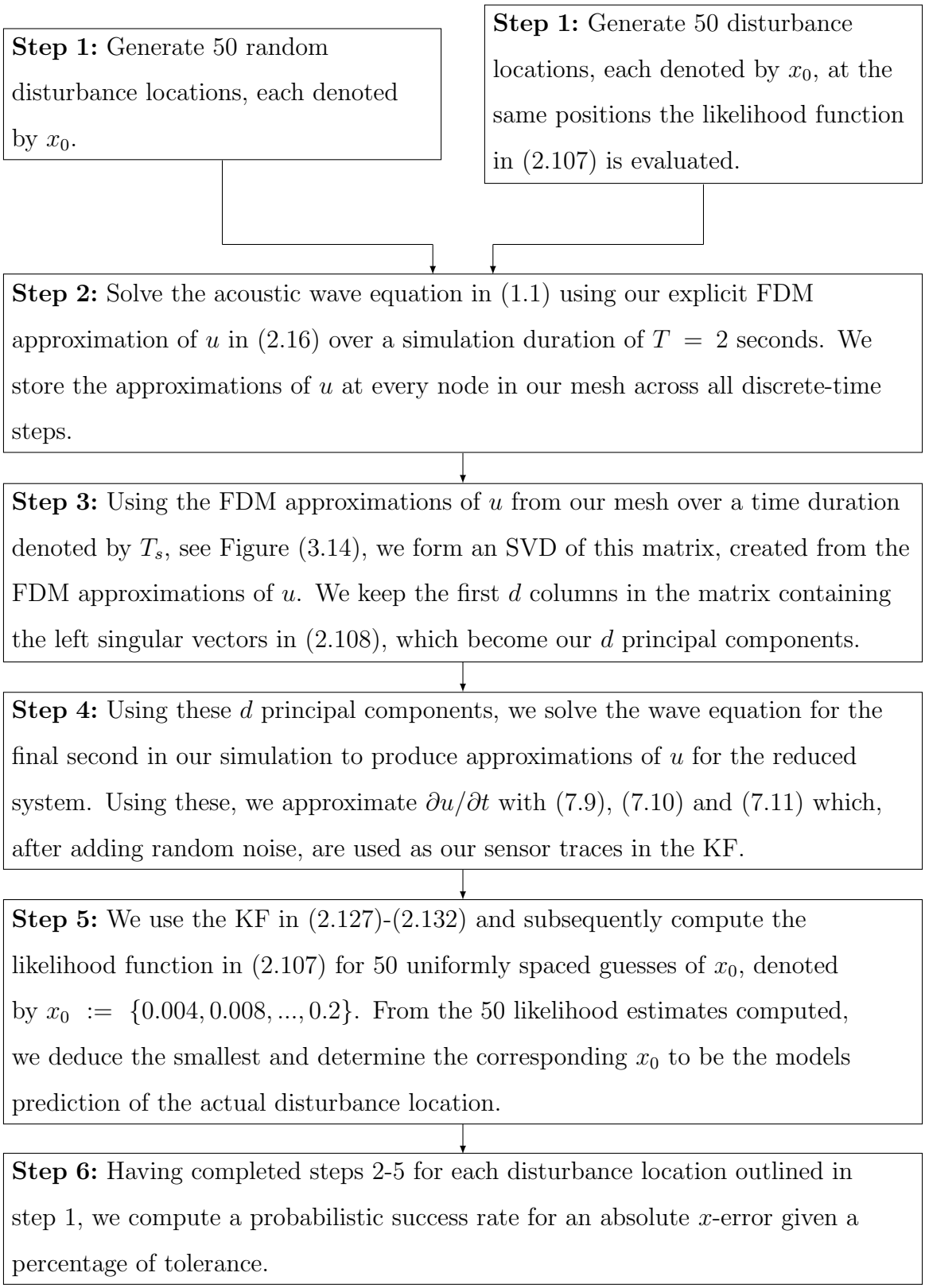


Figure 7.2: A schematic for our 1D model problem using an SVD and approximations of $\partial u/\partial t$ for sensor traces only.

For step 4, we use the d principal components to solve the acoustic wave equation over the final second of our simulation to produce approximations of u for our reduced system. Using these, we are able to obtain approximations of u for our original system. Using these approximations of u deduced for our original system we can approximate $\partial u/\partial t$ with (7.9), (7.10) and (7.11), which once adding random noise, are used in the KF as our sensor traces.

In step 5, we run the KF using approximations of $\partial u/\partial t$ for our sensor traces. In the case where we use approximations of both u and $\partial u/\partial t$ as our sensor traces, both are input here. We run the KF 50 times for different disturbance location guesses, denoted by $x_0 := \{0.004, 0.008, ..., 0.2\}$, each resulting in a likelihood function using (2.107). From the 50 likelihood estimates computed, we compute the smallest and conclude that the corresponding disturbance location is the models prediction of the true disturbance location used to generate our sensor traces.

Lastly, in step 6, we compute probabilistic success rates within a range of relative tolerances given an absolute x-error. We are able to do this by following steps 2-5 for each disturbance location in step 1. Using these results, we conclude how well our 1D model problem is at predicting the disturbance location.

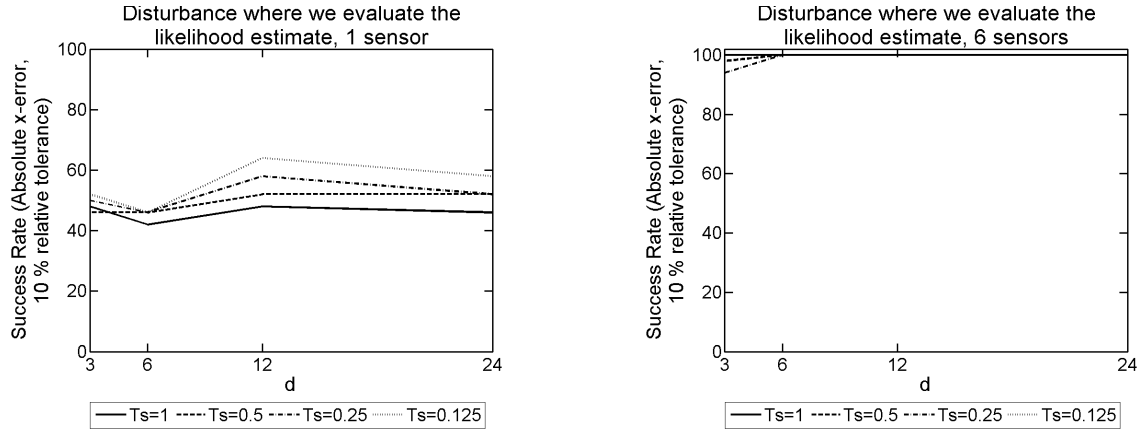
7.3 Results

In this section, we look at the results obtained for our 1D model problem using an SVD for a disturbance frequency of $F = 25$Hz in (1.6). We consider two different scenarios. In the first approach, we take sensor traces originating from the approximations of $\partial u/\partial t$ only. In contrast, in our second scenario, we consider sensor traces from the approximations of both u and $\partial u/\partial t$. The full set of results in both graphical and tabular form can be found in Appendix E.1 and E.2. Amongst these results, we have one through to six sensors on our 1D domain, with two different types of disturbance locations used to generate the success rates. Since the results in this section involve the SVD, the results correspond to a range of values chosen for d and T_s.

7.3.1 Using approximations of $\partial u/\partial t$ for sensor traces

In this section, we discuss results obtained using only approximations of $\partial u/\partial t$ as our sensor traces. Figures (7.3) and (7.4) show the probabilistic success rates for an array of T_s and d values used to form our SVD given an absolute x-error with a 10% relative tolerance for 50 disturbance locations where we evaluate the likelihood function in (2.107) and 50 random disturbance locations, respectively.

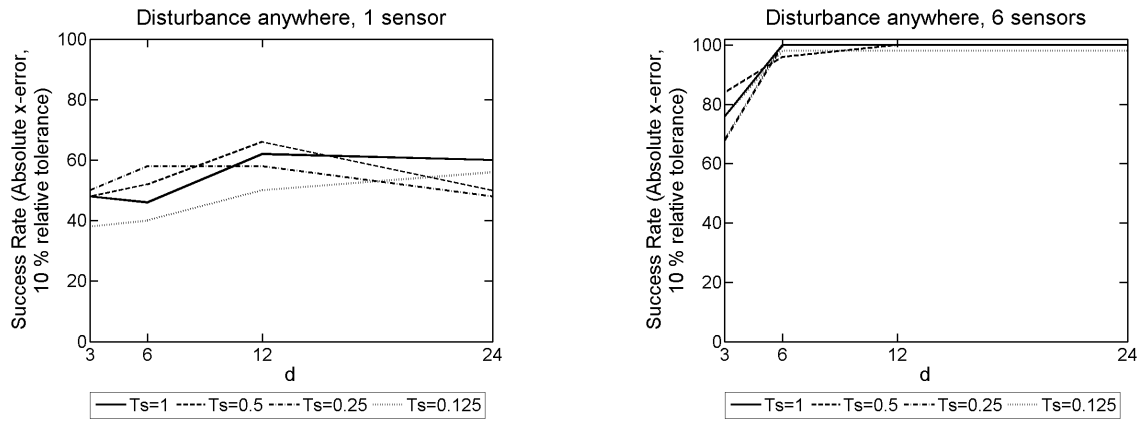
Upon inspection of both Figures (7.3) and (7.4), we conclude that having more sensors present results in better success rates irrespective of the disturbance locations used to produce these results. The value taken for T_s has no noticeable effect on our success rates, whereas the amount of principal components taken from our SVD, denoted by d, affects the success rates when we have six sensors present. That is, when $d > 3$, we get better results, especially when we have random disturbance locations. This was expected since the singular values in our SVD did not converge to zero as quickly as has been observed in other cases, and so, when we take a larger value for d the resultant principle components contain more of the original information.



(a) One sensor present.

(b) Six sensors present.

Figure 7.3: The success rate for our 1D model problem with a disturbance frequency of $F = 25$Hz given an absolute x-error with a 10% relative tolerance. The sensor traces are our approximations of $\partial u/\partial t$ from a mesh with dimensions of $N = 50$ nodes and $L = 9000$ discrete-time steps. These probabilistic results come from 50 disturbance locations where the likelihood function in (2.107) is evaluated.



(a) One sensor present.

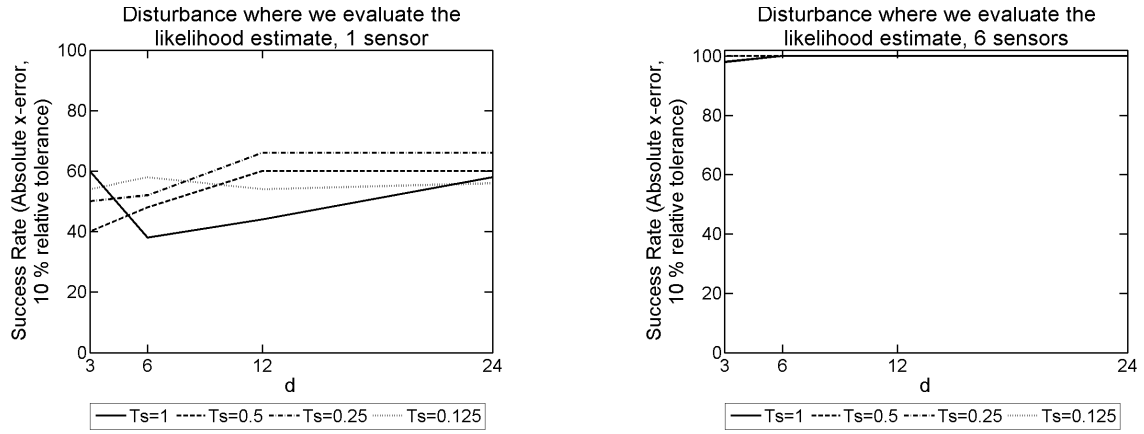
(b) Six sensors present.

Figure 7.4: The success rate for our 1D model problem with a disturbance frequency of $F = 25$Hz given an absolute x-error with a 10% relative tolerance. The sensor traces are our approximations of $\partial u/\partial t$ from a mesh with dimensions of $N = 50$ nodes and $L = 9000$ discrete-time steps. These probabilistic results come from 50 random disturbance locations.

Comparing these results to those corresponding to sensor traces originating from explicit FDM approximations of u only in Figures (3.19) and (3.21), we can conclude that when the mesh dimensions are the same, the results follow the same pattern despite using different sensor traces. A significant difference worth noting is when there are six sensors present, and the disturbance locations are random, see Figures (7.4)(b) and (3.21)(b). When our sensor traces come from approximations of $\partial u/\partial t$, we exhibit better results for a lower value of d, when compared to when our sensor traces come from the explicit FDM approximations of u.

7.3.2 Using approximations of u and $\partial u/\partial t$ for sensor traces

In this section, we investigate the results obtained using approximations of both u and $\partial u/\partial t$ as our sensor traces. Figures (7.5) and (7.6) show the probabilistic success rates for an array of T_s and d values used to form our SVD given an absolute x-error with a 10% relative tolerance for 50 disturbance locations where we evaluate the likelihood function in (2.107) and 50 random disturbance locations, respectively.



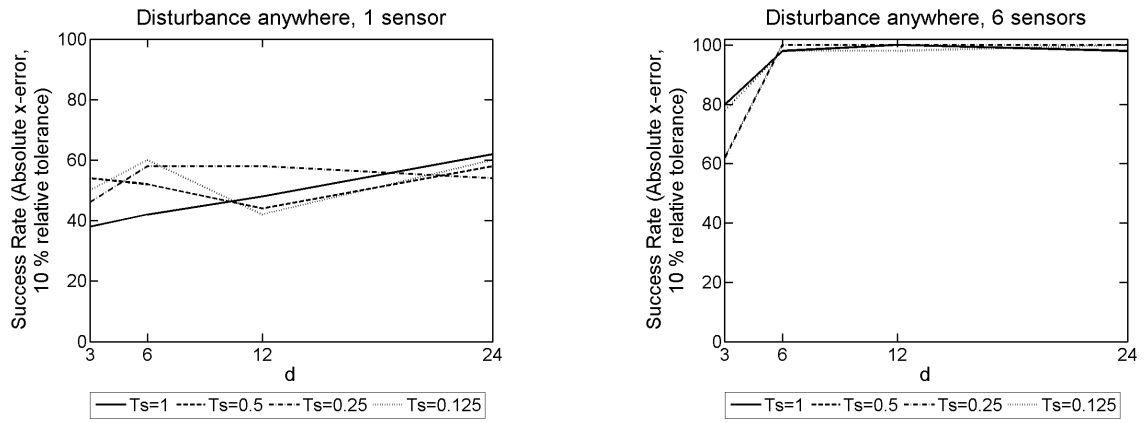
(a) One sensor present.

(b) Six sensors present.

Figure 7.5: The success rate for our 1D model problem with a disturbance frequency of $F = 25$Hz given an absolute x-error with a 10% relative tolerance. The sensor traces correspond to our explicit FDM approximations of u and our approximations of $\partial u/\partial t$ from a mesh with dimensions of $N = 50$ nodes and $L = 9000$ discrete-time steps. These probabilistic results come from 50 disturbance locations where the likelihood function in (2.107) is evaluated.

Upon inspection of both Figures (7.5) and (7.6), we see that having a larger quantity of sensors present results in better success rates irrespective of the disturbance locations used to produce these results. The value taken for T_s has no impact on our success rates, whereas the amount of principal components taken from our SVD, denoted by d, positively affects the success rates when we have six sensors present and our disturbance locations are random. This was expected since the singular values in our SVD did not converge to zero as quickly as has been observed previously, and so, when we take a larger value for d the resultant principle components contain more of the original information.

Comparing these results to those corresponding to sensor traces originating from explicit FDM approximations of u only in Figures (3.19) and (3.21), and the results obtained using sensor traces from the approximations of $\partial u/\partial t$ only in Figures (7.3) and (7.4), we observe that overall the probabilistic success rates are indistinguishable.



(a) One sensor present.

(b) Six sensors present.

Figure 7.6: The success rate for our 1D model problem with a disturbance frequency of $F = 25$Hz given an absolute x-error with a 10% relative tolerance. The sensor traces correspond to our explicit FDM approximations of u and our approximations of $\partial u/\partial t$ from a mesh with dimensions of $N = 50$ nodes and $L = 9000$ discrete-time steps. These probabilistic results come from 50 random disturbance locations.

7.4 Summary

In this chapter, we have investigated whether using different sensors will impact our probabilistic success rates. We considered two different scenarios. The first involved sensor traces from approximations of $\partial u/\partial t$ only, and the second had sensor traces from both our explicit FDM approximations of u and from approximations of $\partial u/\partial t$.

In both cases, we opted to use a coarse mesh to approximate u and $\partial u/\partial t$. We did this due to what we observed in chapter 3, that is when using the SVD on our 1D model problem with a disturbance frequency of $F = 25$Hz, a coarse approximation still yields good results and reduces the run-time requirement.

Having already discussed the results obtained for both different approaches, we found that using different sensor traces had little impact on our probabilistic success rates. This is a good observation, as it means there is room for optimism that in a real-life scenario, different sensors could be used to collect data, and still yield good results.

Chapter 8

Conclusion and Recommendations

8.1 Conclusion

In this thesis, we attempt to locate the source of an acoustic wave equation using likelihood estimates and the Kalman filter. We set up a forcing function in (1.6) in an attempt to mimic a beating heart signal, combined with a disturbance caused by a partially blocked coronary artery. We use this forcing function in a 1D and 2D acoustic wave equation, and approximate the solution of our PDE using an explicit FDM approximation, which at specific nodes in our mesh having had an appropriate amount of noise added, becomes our sensor traces. Using these sensor traces, we use the KF in conjunction with a likelihood function in (2.107) to determine how likely a disturbance location, denoted by x_0 in 1D and $\underline{\mathbf{x}}_0$ in 2D, is at being the true disturbance location that generated the sensor traces.

The work presented in this thesis considers an array of different model problems. In chapter 3, we consider a 1D model problem with a disturbance frequency of $F = 25$Hz. In this chapter, we experiment with using explicit FDM approximations of u from a coarse mesh as opposed to a fine mesh, which works very well as observed by comparing Figure (3.8) to Figure (3.10) and Figure (3.12) to Figure (3.13). In addition to observing good success rates, using a coarse mesh to form our explicit FDM approximations of u significantly reduces the run-time and system RAM requirements. Initially, we used a minimisation algorithm to find x_0 which minimised (2.107) and realised that taking only a single initial guess for x_0 did not yield good results. Therefore, we decided to run our 1D model problem for 50 equidistant initial guesses of x_0 in our domain, and

so, the minimisation algorithm produces an optimal disturbance location for each initial guess of x_0. From these 50 optimal disturbance locations, we choose our prediction of x_0 depending on which minimises our likelihood function in (2.107) the most. This approach worked extremely well, see Figures (3.8) and (3.10). However, due to the tolerances set in the minimisation algorithm, our 1D model problem would run over 3500 times, making this approach infeasible for model problems requiring finer meshes and more discrete time steps

Therefore, we investigated the possibility of only evaluating the likelihood function in (2.107) at these 50 equidistant initial guesses of x_0. As was expected, the results were not as good because, in the scenario where the disturbance was not located at one of these 50 guesses, our 1D model problem could not identify the actual disturbance location. Despite this, the results observed in Figures (3.12) and (3.13) are still good, making this approach a feasible option for model problems requiring finer meshes.

In an attempt to further optimise our 1D model problem, we used an SVD to decompose a matrix containing our explicit FDM approximations to u at every node in our mesh across a subset, denoted by T_s, of the total discrete-time steps. From this decomposition, we extracted the first d columns corresponding to the largest singular values. Using these d columns, otherwise known as our principal components, we were able to reduce the matrix dimensions in the KF. Combining this with explicit FDM approximations of u from a coarse mesh, only evaluating the likelihood function at 50 positions, and choosing d and T_s to be small yielded good probabilistic success rates.

In chapter 4, we extended the work in chapter 3 to consider higher frequencies for our disturbance in (1.6) since the real-life frequency emitted from a partially blocked coronary artery is in the range of $300 - 800$Hz. Due to the finer mesh requirements for our 1D model problems with frequency disturbances of $F = 150$Hz and $F = 300$Hz, we used the insights obtained in chapter 3 and only considered the approach where we use an SVD to reduce the matrix dimensions in the KF.

The results observed in Figures (4.5), (4.6), (4.7) and (4.8) for both frequency disturbances are good when six sensors are presents, even for small values taken for both d and T_s when forming and using our SVD. However, when our disturbance frequency was $F = 150$Hz, we observed lower probabilistic success rates for larger values of d. We are not sure what the root cause of this is, however, it could be linear dependence between

the principal components used from our SVD. The same negative effect was not observed when $F = 300$Hz.

For chapter 5, we introduced our two 2D model problems with a disturbance frequency of $F = 25$Hz. The first had 1NBC and three Dirichlet boundary conditions, whereas the second had 3NBCs and a single Dirichlet boundary condition. We attempted to use a minimisation algorithm to find $\underline{\mathbf{x}}_0$ which minimises (2.107) using 245 initial guesses of $\underline{\mathbf{x}}_0$ in our 2D model problem with 1NBC (255 for our 2D model problem with 3NBCs). However, the run-time requirement made this infeasible.

Therefore, we deployed the approach investigated in chapter 3 where we only evaluate the likelihood function in (2.107) at specific locations in our domain using sensor traces from a coarse mesh. Doing this meant we could actually obtain results for our 2D model problems, although it did induce an unavoidable x-error of 1% and a y-error of 14.1% in the worst-case scenario. As a result of this, we increased the acceptable tolerance for our y-error and Euclidean-error. Having done this, we observed good probabilistic success rates for our 2D model problems with a disturbance frequency of $F = 25$Hz in Figures (5.9) and (5.10).

In addition to this approach, we also used an SVD to reduce the matrix dimensions in the KF for our 2D model problems, making a significant impact on the run-time requirements. The results observed in Figures (5.13), (5.14), (5.15), (5.16), (5.17) and (5.18) are respectable, making this approach a feasible option for our 2D model problems.

To finish chapter 5, we investigated an array of different approaches which would further optimise our 2D model problems. Using the SVD approach with sensor traces from a coarse mesh, we were able to terminate the KF up to 80% early without ruining the probabilistic success rates obtained. This is a hugely significant result which could make running our 2D model problems with a minimisation algorithm feasible. We have not attempted this, but it is a recommendation for future work.

In chapter 6, we extended the work in chapter 5 to consider higher frequencies for our disturbance in (1.6) since the real-life frequency emitted from a partially blocked coronary artery is in the range of $300 - 800$Hz. Due to the significantly finer mesh requirements for our 2D model problems with frequency disturbances of $F = 150$Hz and $F = 300$Hz, we had no choice other than to use the approach which uses the SVD to reduce the dimension of the matrices in the KF. The results observed in this chapter were good, even when we

chose relatively small values for d and T_s.

Lastly, we have chapter 7, where we extend the work in chapter 3 to consider sensor traces from approximations of $\partial u/\partial t$. We considered two cases: the first case has sensor traces originating from approximations of $\partial u/\partial t$ only, whereas the second case has sensor traces from both our explicit FDM approximation of u and the approximations of $\partial u/\partial t$. Since up to this point using the SVD approach to reduce the size of the matrices in the KF has been very successful, we implement this approach to lower the run-time requirements. Having observed the results for each case in Figures (7.3), (7.4), (7.5) and (7.6), we can conclude that using sensor traces from either our approximations of u or $\partial u/\partial t$ makes little difference to the probabilistic success rates. Moreover, when both are present in our sensor traces, this does not particularly improve the success rates for this 1D model problem.

8.2 Recommendations for future work

In this section, we discuss six recommendations for future work based on the content presented in this thesis:

1. Extending the model problems in this thesis to a viscoelastic PDE, rather than an acoustic PDE. This will enable these model problems to become closer to the application of this work.

2. In chapter 5 we considered 2D model problems with a disturbance frequency of $F = 25$Hz. We discovered in that chapter issues dealing with larger dimensional problems and the associated run-time requirements due to the number of nodes in our mesh and quantity of discrete-time steps. We have already shown that using an SVD enables the reduction of matrix dimensions in the KF. However, the KF is still run over all discrete-time steps. We investigated the possibility of terminating the KF early at the end of chapter 5, which turned out to be a feasible approach. As a result, this recommendation is to consider the use of a minimisation algorithm, as we did in chapter 3, but terminating the KF early. Due to the early termination of the KF, using a minimisation algorithm for our 2D model problems with a disturbance frequency of $F = 25$Hz could be feasible, and we would expect the probabilistic

success rates for our relative absolute y-error to be better as the unavoidable error of 14.1% in the worst-case scenario would no longer be present.

3. This recommendation is again aimed at our 2D model problems. Previously, the results in chapters 5 and 6 did not use a minimisation algorithm. Instead, the likelihood function in (2.107) was evaluated at specific locations in our 2D domain which induced a relative absolute x-error of 1% and a relative absolute y-error of 14.1% in the worst-case scenario. Therefore, we recommend that the likelihood function is evaluated at a higher quantity of places in the y-direction, to reduce the scale of the unavoidable y-error. This will lengthen the run-time, but we would expect an improvement in the models ability to predict the location of the disturbance successfully.

4. In a real-life scenario, first discussed in chapter 1, sensors would be placed on a human thorax causing inaccuracies in their placement. As a result, a reasonable consideration would be to generate sensor traces at specific nodes in our mesh, as we have done before, but trick the KF into believing they are within a user-defined tolerance from their true positions. This would almost certainly yield worse results. However, it would be interesting to see how large a tolerance could be achieved before the model struggles to predict the true disturbance location.

5. When someone has CAD, it is extremely likely they have more than a single blockage in their coronary arteries. As a result, it would be interesting to investigate the possibility of setting up the forcing function in (1.6) to have two different disturbance locations and see if our models can locate both.

6. Like the previous recommendation, a good experiment would be to construct a forcing function, similar to that in (1.6), to have two different disturbances with different amplitudes. One would have the same amplitude used to obtain the results in this thesis, whereas the other would have a significantly smaller amplitude. In the KF, we would assume there was only a single disturbance and try to deduce the disturbance location with the larger amplitude. This would look at the feasibility of having sensor traces with conflicting disturbances, and locating the most serious blockage.

Bibliography

[1] Townsend, N., Bhatnagar, P., Wilkins, E., Wickramasinghe, K. and Rayner, M. (2015). Cardiovascular disease statistics, 2015. British Heart Foundation.

[2] Nichols, M., Townsend, N., Luengo-Fernandez, R., Leal, J., Gray, A., Scarborough, P. and Rayner, M. (2012). European Cardiovascular Disease Statistics 2012. European Heart Network AISBL.

[3] Bhatnagar, P., Wickramasinghe, K., Wilkins, E. and Townsend, N. (2016). Trends in the epidemiology of cardiovascular disease in the uk. *Heart*, **102**(24), 1945–1952.

[4] Hardoon, S.L., Morris, R.W., Whincup, P.H., Shipley, M.J., Britton, A.R., Masset, G., Stringhini, S., Sabia, S., Kivimaki, M., Singh-Manoux, A. and Brunner, E.J. (2011). Rising adiposity curbing decline in the incidence of myocardial infarction: 20-year follow-up of british men and women in the whitehall ii cohort. *European heart journal*, **33**(4), 478–485.

[5] PromoCell (2018). Human coronary artery endothelial cells (hcaec). Available: https://www.promocell.com/product/human-coronary-artery-endothelial-cells-hcaec/#tab-data_figures. Accessed: 17/07/2019.

[6] Saga (2017). High blood pressure and atherosclerosis. Available: https://www.saga.co.uk/magazine/health-wellbeing/conditions/cardiovascular/high-blood-pressure-atherosclerosis. Accessed: 17/07/2019.

[7] Public Health England (2017). Health profile for England: 2017. HM Government.

[8] Liu, J., Maniadakis, N., Gray, A. and Rayner, M. (2002). The economic burden of coronary heart disease in the uk. *Heart*, **88**(6), 597–603.

[9] WHO (2019). The challenge of cardiovascular disease - quick statistics. Available: http://www.euro.who.int/en/health-topics/noncommunicable-diseases/cardiovascular-diseases/data-and-statistics. Accessed: 07/07/2019.

[10] British Heart Foundation (2019). Coronary heart disease. Available: https://www.bhf.org.uk/informationsupport/conditions/coronary-heart-disease. Accessed: 26/02/2019.

[11] Lassale, C., Tzoulaki, I., Moons, K., Sweeting, M., Boer, J., Johnson, L., Huerta, J.M., Agnoli, C., Freisling, H., Wennberg, E.W.P., van der A, D.L., Arriola, L., Benetou, V., Boeing, H., Bonnet, F., Colorado-Yohar, S., Engström, G., Eriksen, A., Ferrari, P., Grioni, S., Johansson, M., Kaaks, R., Katsoulis, M., Katzke, V., Key, T., Matullo, G., Melander, O., Molina-Portillo, E., Moreno-Iribas, C., Norberg, M., Overvad, K., Panico, S., Quirós, J.R., Saieva, C., Skeie, G., Steffen, A., Stepien, M., Tjønneland, A., Trichopoulou, A., Tumino, R., Schouw, Y., Verschuren, M., Langenberg, C., Angelantonio, E.D., Riboli, E., Wareham, E., Danesh, J. and Butterworth, A. (2017). Separate and combined associations of obesity and metabolic health with coronary heart disease: a pan-european case-cohort analysis. *European Heart Journal*.

[12] NHS (2019). Statistics on obesity, physical activity and diet. Available: https://digital.nhs.uk/data-and-information/publications/statistical/statistics-on-obesity-physical-activity-and-diet/statistics-on-obesity-physical-activity-and-diet-england-2019. Accessed: 18/07/2019.

[13] NHS (2017). Diagnosis (coronary heart disease). Available: https://www.nhs.uk/conditions/coronary-heart-disease/diagnosis/. Accessed: 18/07/2019.

[14] NHS (2018). Nhs diagnostic waiting times and activity data. Available: https://www.england.nhs.uk/statistics/wp-content/uploads/sites/2/2018/03/DWTA-Report-January-2018.pdf. Accessed: 18/07/2019.

[15] The Guardian (2016). How much have i cost the nhs? Available: https://www.theguardian.com/society/ng-interactive/2016/feb/08/how-much-have-i-cost-the-nhs. Accessed: 18/07/2019.

[16] Adams, A., Bojara, W. and Schunkc, K. (2017). Early diagnosis and treatment of coronary heart disease in symptomatic subjects with advanced vascular atherosclerosis of the carotid artery (type iii and iv b findings using ultrasound). *Cardiology Research*, **8**(1), 7 – 12.

[17] Institute, A.H..V. (2019). The importance of the early detection of cardiovascular disease. Available: https://www.alaskaheart.com/the-importance-of-the-early-detection-of-cardiovascular-disease/. Accessed: 18/07/2019.

[18] British Heart Foundation (2018). Early detection drive could prevent 115000 cases of heart and circulatory disease over next decade. Available: https://www.bhf.org.uk/what-we-do/news-from-the-bhf/news-archive/2018/november/early-detection-drive-could-prevent-115000-cases-of-heart-and-circulatory-disease-over-next-decade. Accessed: 18/07/2019.

[19] Wang, J. Z., Tie, B., Welkowitz, W., Semmlow, J. L. and Kostis, J. B. (1990). Modeling sound generation in stenosed coronary arteries. *Transactions on Biomedical Engineering*, **37**(11), 1087–1094.

[20] Semmlow, J. and Rahalkar, K. (2007). Acoustic detection of coronary artery disease. *Annual Review of Biomedical Engineering*, **9**, 449–469.

[21] Schmidt, S. E., Hansen, J., Zimmermann, H., Hammershøi, D., Toft, E. and Struijk, J. J. (2011). Coronary artery disease and low frequency heart sound signatures. *Computing in Cardiology*, 481–484.

[22] Schmidt, S. E., Holst-Hansen, C., Hansen, J., Toft, E. and Struijk, J. J. (2015). Acoustic features for the identification of coronary artery disease. *Transactions on Biomedical Engineering*.

[23] Winther, S., Schmidt, S. E. and Holm, N. R. (2016). Diagnosing coronary artery disease by sound analysis from coronary stenosis induced turbulent blood flow: diagnostic performance in patients with stable angina pectoris. *International Journal of Cardiovascular Imaging*, **32**, 235–245.

[24] Thomas, J.L., Winther, S., Wilson, R.F. and Bøttcher, M. (2017). A novel approach to diagnosing coronary artery disease: acoustic detection of coronary turbulence. *The International Journal of Cardiovascular Imaging*, **33**(1), 129 – 136.

[25] Winther, S., Nissen, L. and Schmidt, S. E. (2018). Diagnostic performance of an acoustic-based system for coronary artery disease risk stratification. *Heart*, **104**(11), 928–935.

[26] Dock, W. and Zoneraich, Z. (1967). A diastolic murmur arising in a stenosed coronary artery. *AM J Med*, **42**(4), 617–619.

[27] Cheng, T. (1970). Diastolic murmur caused by coronary artery stenosis. *Ann Intern Med*, **72**(4), 543–546.

[28] Akay, M., Welkowitz, W., Semmlow, J.L. and Kostis, J.B. (1991). Applications of the arma method to acoustic detection of coronary artery disease. *Medical & Biological Engineering & Computing*, **29**(4), 365 – 372.

[29] Akay, M., Akay, Y.M., Welkowitz, W., Semmlow, J.L. and Kostis, J.B. (1992). Application of adaptive filters to noninvasive acoustical detection of coronary occlusions before and after angioplasty. *IEEE Transactions on Biomedical Engineering*, **39**(2), 176 – 184.

[30] Akay, M., Welkowitz, W., Semmlow, J.L., Akay, Y.M. and Kostis, J. (1992). Noninvasive acoustical detection of coronary artery disease using the adaptive line enhancer method. *Medical and Biological Engineering and Computing*, **30**(2), 147 – 154.

[31] Akay, M., Akay, Y.M., Welkowitz, W., Semmlow, J.L. and Kostis, J. (1993). Application of adaptive ftf/faest zero tracking filters to noninvasive characterization of the sound pattern caused by coronary artery stenosis before and after angioplasty. *Annals of Biomedical Engineering*, **21**(1), 9 – 17.

[32] Chan, V. (2011). Basics of ultrasound imaging, Atlas of Ultrasound-Guided Procedures in Interventional Pain Management. Springer.

[33] Wang, L., Libert, G. and Manneback, P. (1992). Kalman filter algorithm based on singular value decomposition. *Proceedings of the 31st IEEE Conference on Decision and Control*, **1**, 1224–1229.

[34] Tsyganova, J. V. and Kulikova, M. V. (2017). Svd-based kalman filter derivative computation. *IEEE Transactions on Automatic Control*, **62**(9), 4869–4875.

[35] Ordoñez, J. C. B., Valdés, R. M., Gómez Comendador, V. F. (2020). Engineering applications of adaptive kalman filtering based on singular value decomposition (svd). *Applied Sciences*, **10**(15).

[36] Smith, G. (1965). Numerical Solution of Partial Differential Equations. Oxford University Press.

[37] Lax, P. D. and Richtmyer, R. D. (1956). Survey of the stability of linear finite difference equations. *Communications on Pure and Applied Mathematics*, **9**(2), 267 – 293.

[38] Courant, R., Friedrichs, K. and Lewy, H. (1956). On the partial difference equations of mathematical physics. *New York: Courant Institute of Mathematical Sciences, New York University*, 96. Available: https://archive.org/details/onpartialdiffere00cour. Accessed: 24/01/2018.

[39] Rauch, H. E., Tung, F. and Striebel, C. T. (1965). Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, **3**(8), 1445–1456.

[40] Kalman, R.E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, **82**(Series D), 35–45. Available: https://www.cs.unc.edu/~welch/kalman/media/pdf/Kalman1960.pdf. Accessed: 21/11/2017.

[41] Grewal, M.S. and Andrews, A.P. (2010). Applications of kalman filtering in aerospace 1960 to the present [historical perspectives]. *IEEE Control Systems Magazine*, **30**(3), 69–78. Available: https://ieeexplore.ieee.org/document/5466132. Accessed: 12/08/2019.

[42] Cornell University (2008). Kalman filter applications. Available: https://www.cs.cornell.edu/courses/cs4758/2012sp/materials/MI63slides.pdf. Accessed: 12/08/2019.

[43] Fruhwirth, R. (1987). Application of kalman filtering to track and vertex fitting. *Nuclear Instruments and Methods in Physics*, **262**(2-3), 444–450.

[44] Hun, L.C., Yeng, O.L., Sze, L.T. and Chet, K.V. (2016). Kalman Filtering and Its Real-Time Applications. Intechopen. Available: https://www.intechopen.com/books/real-time-systems/kalman-filtering-and-its-real-time-applications. Accessed: 12/08/2019.

[45] Nelder, J.A. and Mead, R. (1965). A simplex method for function minimization. *The Computer Journal*, **7**(4), 308–313.

[46] Scholarpedia (2011). Nelder–mead algorithm. Available: http://www.scholarpedia.org/article/Nelder-Mead_algorithm. Accessed: 21/11/2017.

[47] McKinnon, K.I. (1998). Convergence of the nelder–mead simplex method to a nonstationary point. *SIAM Journal on Optimization*, **9**(1), 148–158. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.52.3900&rep=rep1&type=pdf. Accessed: 28/11/2017.

[48] Datta, B.N. (1995). Numerical Linear Algebra and Applications. Brooks/Cole Publishing Company.

[49] Golub, G. and Kahan, W. (1965). Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial and Applied Mathematics Series B Numerical Analysis*, **2**(2), 205 – 224. Available: https://epubs.siam.org/doi/10.1137/0702016. Accessed: 27/08/2019.

[50] Tamir, M. (2014). When and where do we use svd? Available: https://www.quora.com/When-and-where-do-we-use-SVD. Accessed: 27/08/2019.

[51] Rowland, T. (2020). Orthogonal Matrix. From MathWorld – A Wolfram Web Resource, created by Eric W. Weisstein. Available: http://mathworld.wolfram.com/OrthogonalMatrix.html. Accessed: 19/02/2020.

[52] Rowland, T. (2019). Orthonormal Basis. From MathWorld – A Wolfram Web Resource, created by Eric W. Weisstein. Available: http://mathworld.wolfram.com/OrthonormalBasis.html. Accessed: 13/11/2019.

[53] Luo, Z. and Jin, S. (2019). A reduced-order extrapolated crank–nicolson collocation spectral method based on proper orthogonal decomposition for the two-dimensional viscoelastic wave equations. *Numerical Methods for Partial Differential Equations*, **36**(1), 49–65. Available: https://onlinelibrary.wiley.com/doi/10.1002/num.22397. Accessed: 26/01/2020.

Appendices

Appendix A: Results for chapter 3

A.1 FDM approximation of u from a fine mesh

	Success Rate		
Number of Sensors	1%	5%	10%
1	100	100	100
2	100	100	100

Table 9.1: The 1D model success rate using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 250$ nodes and $L = 15000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 disturbance locations where the likelihood function in (2.107) is evaluated.

	Success Rate		
Number of Sensors	1%	5%	10%
1	100	100	100
2	100	100	100

Table 9.2: The 1D model success rate using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 250$ nodes and $L = 15000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 random disturbance locations.

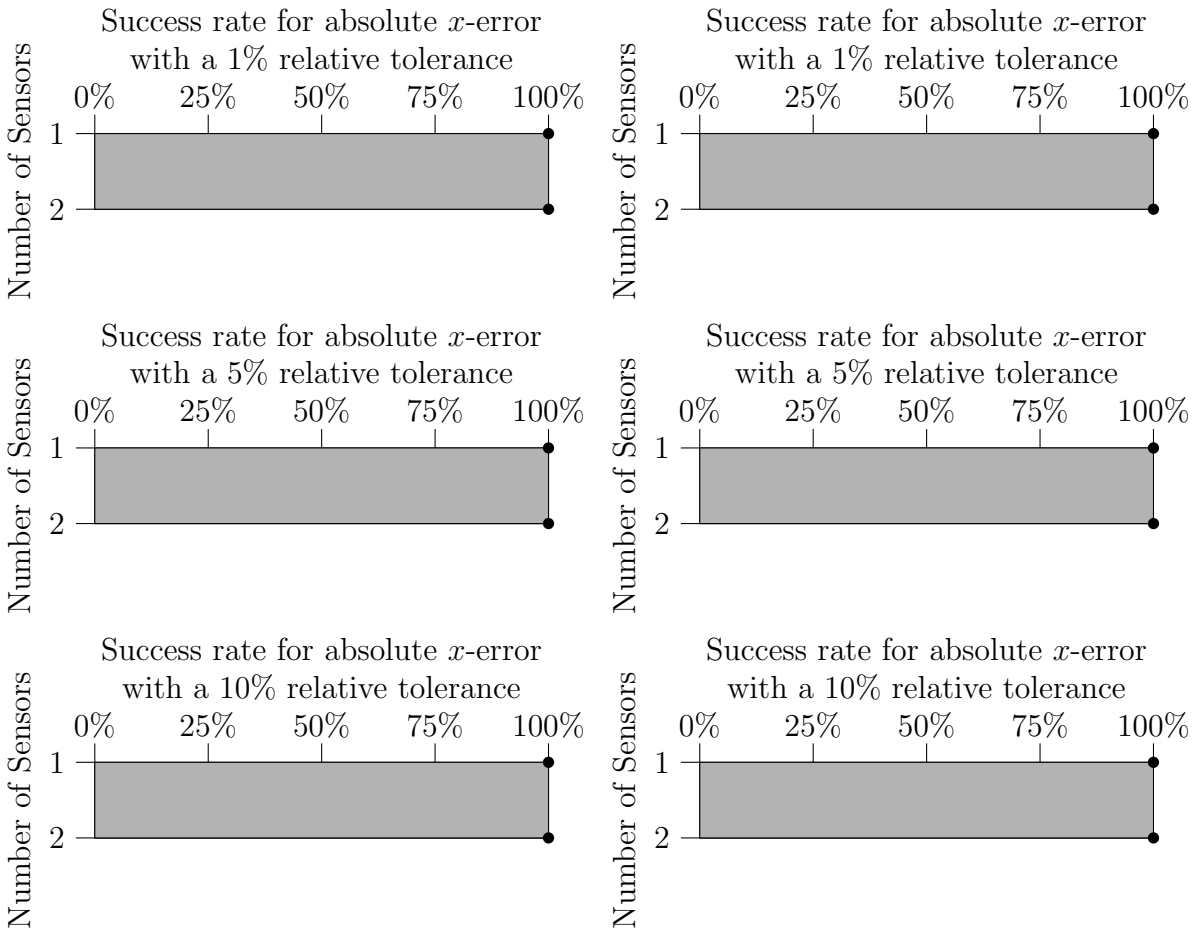


Figure 9.1: The 1D model success rate using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 250$ nodes and $L = 15000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. The probabilistic results on the LHS come from 50 disturbance locations where the likelihood function in (2.107) is evaluated, and the results on RHS used 50 random disturbance locations.

A.2 FDM approximation of u from a coarse mesh

	Success Rate		
Number of Sensors	1%	5%	10%
1	84	84	84
2	100	100	100
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100

Table 9.3: The 1D model success rate using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 50$ nodes and $L = 3000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 disturbance locations where the likelihood function in (2.107) is evaluated.

	Success Rate		
Number of Sensors	1%	5%	10%
1	86	86	86
2	100	100	100
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100

Table 9.4: The 1D model success rate using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 50$ nodes and $L = 3000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 random disturbance locations.

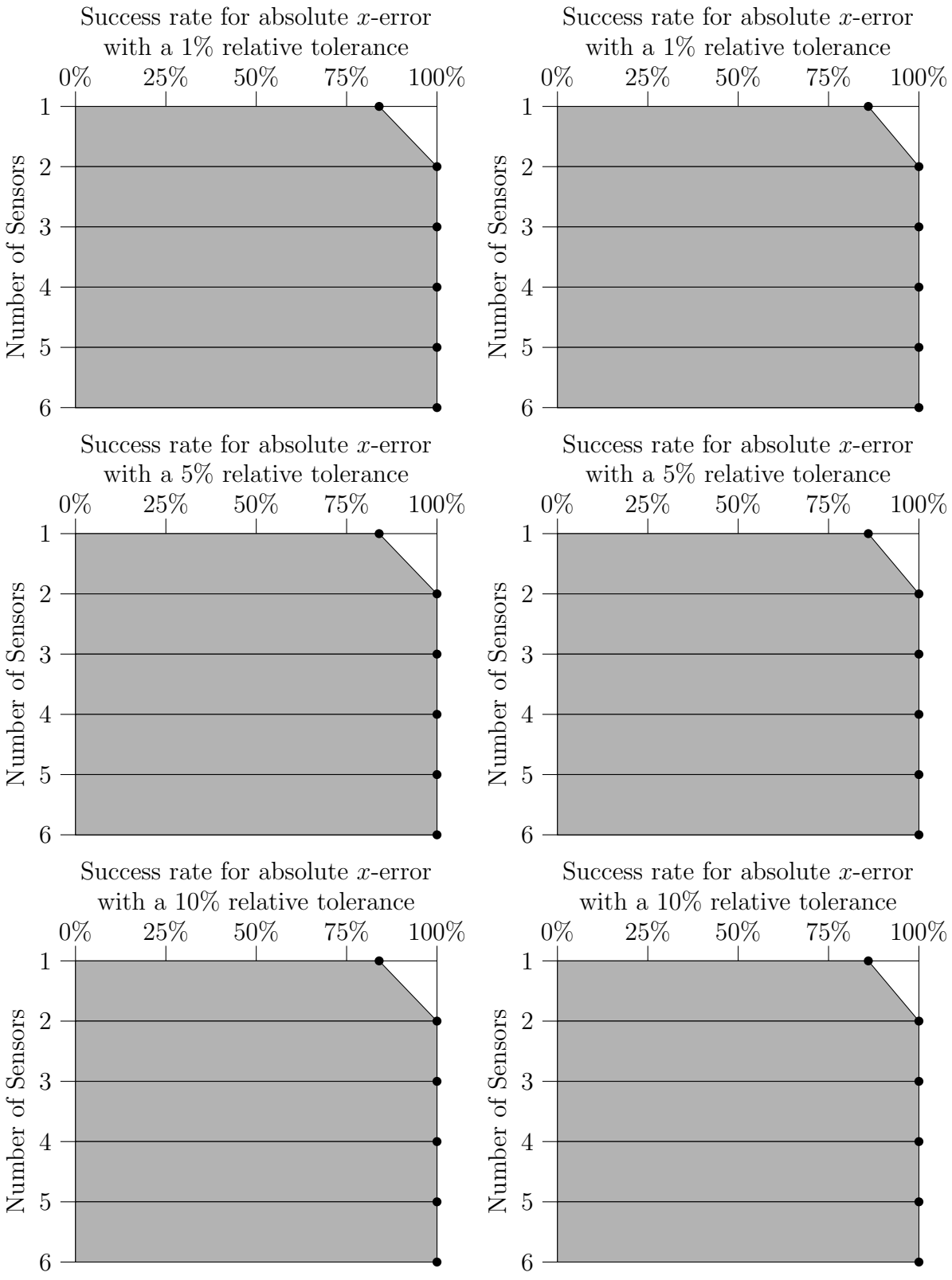


Figure 9.2: The 1D model success rate using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 50$ nodes and $L = 3000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. The probabilistic results on the LHS come from 50 disturbance locations where the likelihood function in (2.107) is evaluated, and the results on RHS used 50 random disturbance locations.

A.3 Without using a minimisation algorithm

	Success Rate		
Number of Sensors	1%	5%	10%
1	92	94	94
2	100	100	100
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100

Table 9.5: The 1D model success rate without using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 250$ nodes and $L = 15000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 disturbance locations where the likelihood function in (2.107) is evaluated.

	Success Rate		
Number of Sensors	1%	5%	10%
1	44	44	48
2	74	76	76
3	80	84	84
4	96	98	98
5	96	100	100
6	100	100	100

Table 9.6: The 1D model success rate without using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 250$ nodes and $L = 15000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 random disturbance locations.

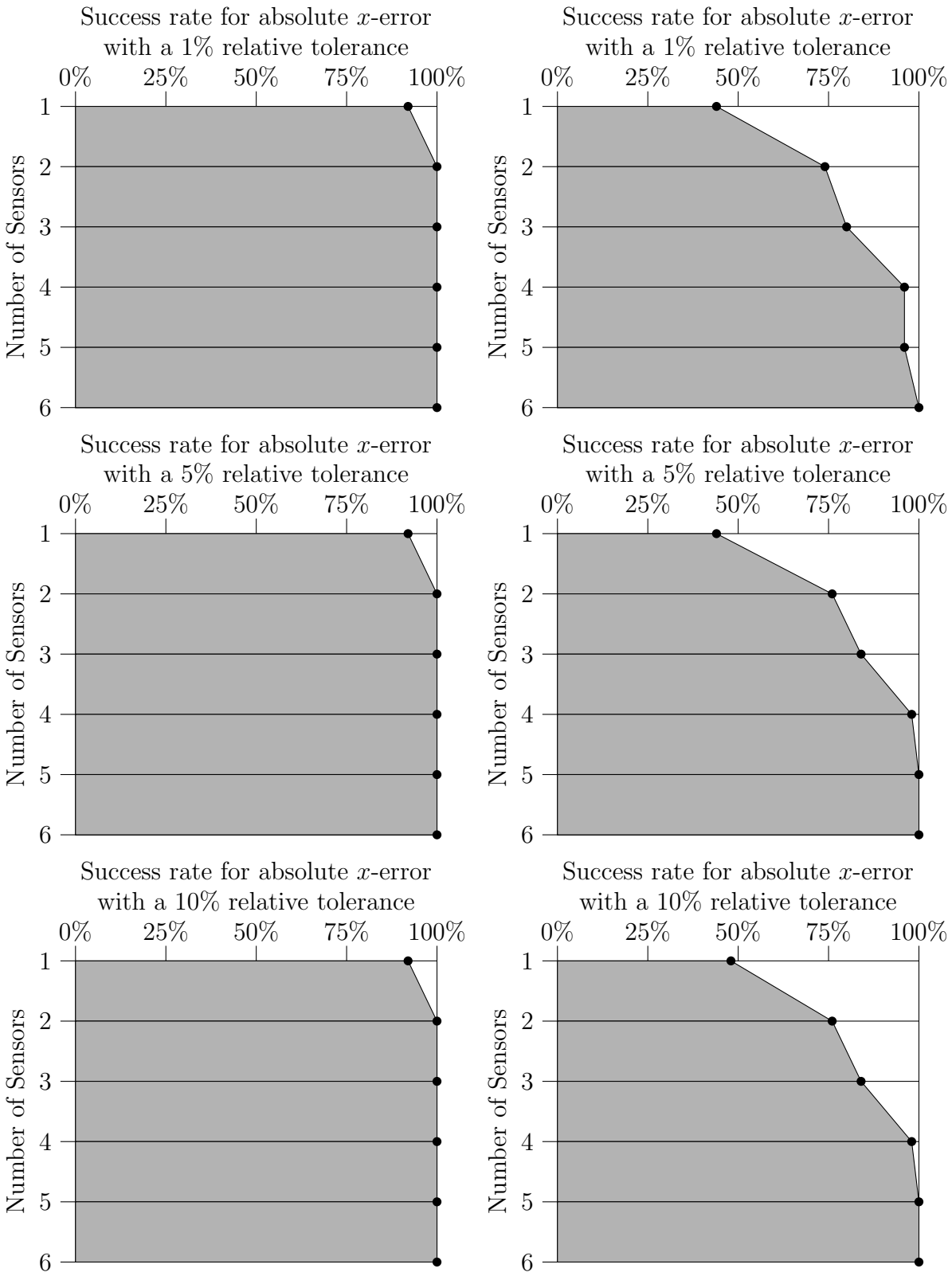


Figure 9.3: The 1D model success rate without using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 250$ nodes and $L = 15000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. The probabilistic results on the LHS come from 50 disturbance locations where the likelihood function in (2.107) is evaluated, and the results on RHS used 50 random disturbance locations.

	Success Rate		
Number of Sensors	1%	5%	10%
1	92	94	94
2	100	100	100
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100

Table 9.7: The 1D model success rate without using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 50$ nodes and $L = 3000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 disturbance locations where the likelihood function in (2.107) is evaluated.

	Success Rate		
Number of Sensors	1%	5%	10%
1	52	56	62
2	70	72	76
3	70	72	76
4	82	84	88
5	88	92	94
6	92	94	98

Table 9.8: The 1D model success rate without using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 50$ nodes and $L = 3000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. These probabilistic results were produced using 50 random disturbance locations.

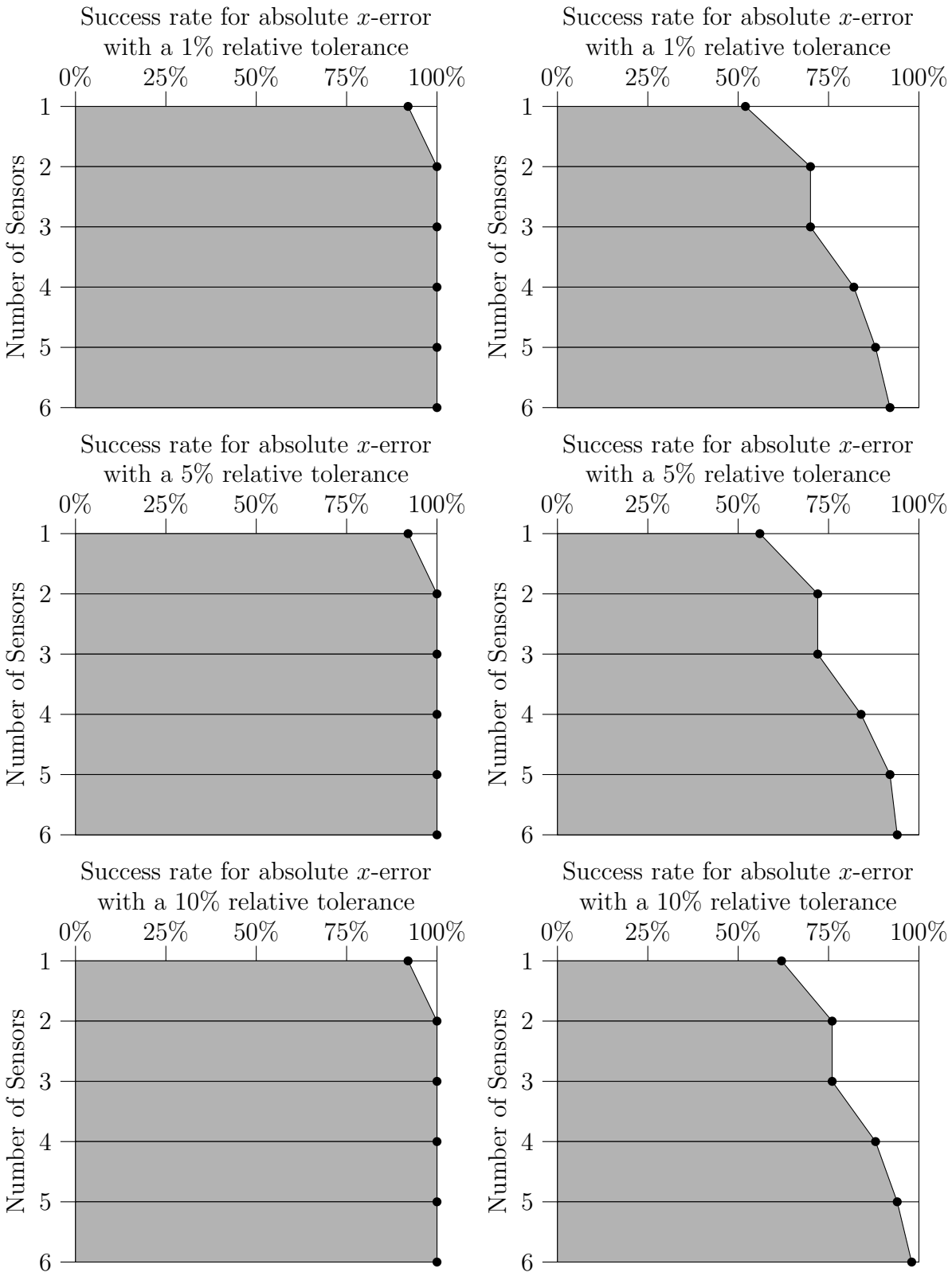


Figure 9.4: The 1D model success rate without using a minimisation algorithm with a varying number of sensors, a mesh with dimensions of $N = 50$ nodes and $L = 3000$ discrete-time steps and a disturbance frequency of $F = 25$Hz. The probabilistic results on the LHS come from 50 disturbance locations where the likelihood function in (2.107) is evaluated, and the results on RHS used 50 random disturbance locations.

A.4 Use of an SVD to reduce matrix dimensions in the KF

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	84	84	84
1.000	6	84	84	84
1.000	12	92	94	94
1.000	24	88	90	90
0.500	3	82	82	82
0.500	6	86	90	90
0.500	12	90	92	92
0.500	24	80	84	84
0.250	3	82	82	82
0.250	6	86	88	90
0.250	12	88	92	92
0.250	24	84	84	84
0.125	3	92	92	92
0.125	6	92	94	94
0.125	12	94	94	94
0.125	24	98	98	98

Table 9.9: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	98	98	98
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	98	98	98
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.10: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 2 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.11: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 3 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.12: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.13: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.14: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	52	58	60
1.000	6	50	56	60
1.000	12	36	44	44
1.000	24	40	44	46
0.500	3	56	64	64
0.500	6	44	52	56
0.500	12	48	54	60
0.500	24	42	46	46
0.250	3	62	68	68
0.250	6	50	54	58
0.250	12	50	60	60
0.250	24	42	48	54
0.125	3	60	70	70
0.125	6	52	60	64
0.125	12	44	54	58
0.125	24	50	62	62

Table 9.15: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	68	78	80
1.000	6	82	88	90
1.000	12	62	66	72
1.000	24	56	58	60
0.500	3	68	80	82
0.500	6	86	94	94
0.500	12	76	82	82
0.500	24	64	68	72
0.250	3	68	80	80
0.250	6	72	78	78
0.250	12	64	76	80
0.250	24	56	62	66
0.125	3	70	76	76
0.125	6	76	88	92
0.125	12	72	84	86
0.125	24	80	86	88

Table 9.16: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	58	68	80
1.000	6	92	92	92
1.000	12	76	78	80
1.000	24	84	88	94
0.500	3	56	64	76
0.500	6	78	86	88
0.500	12	84	86	86
0.500	24	76	80	86
0.250	3	62	70	76
0.250	6	72	80	82
0.250	12	80	84	88
0.250	24	86	88	92
0.125	3	64	72	76
0.125	6	74	82	86
0.125	12	82	90	90
0.125	24	78	86	90

Table 9.17: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	58	68	78
1.000	6	92	98	100
1.000	12	88	92	92
1.000	24	80	84	86
0.500	3	60	70	76
0.500	6	96	96	96
0.500	12	92	94	94
0.500	24	84	86	90
0.250	3	68	76	80
0.250	6	88	92	94
0.250	12	98	100	100
0.250	24	88	92	94
0.125	3	64	70	70
0.125	6	88	92	92
0.125	12	86	94	96
0.125	24	94	98	98

Table 9.18: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	62	72	82
1.000	6	92	100	100
1.000	12	92	98	100
1.000	24	86	90	94
0.500	3	62	72	82
0.500	6	88	96	98
0.500	12	96	100	100
0.500	24	90	96	98
0.250	3	62	76	78
0.250	6	90	94	96
0.250	12	96	100	100
0.250	24	96	100	100
0.125	3	62	70	70
0.125	6	86	94	94
0.125	12	96	100	100
0.125	24	96	100	100

Table 9.19: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	60	70	78
1.000	6	96	98	98
1.000	12	96	100	100
1.000	24	92	96	100
0.500	3	62	72	76
0.500	6	96	98	98
0.500	12	96	100	100
0.500	24	96	100	100
0.250	3	68	76	80
0.250	6	86	92	92
0.250	12	98	100	100
0.250	24	94	98	100
0.125	3	60	64	64
0.125	6	90	98	98
0.125	12	94	98	98
0.125	24	92	98	98

Table 9.20: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

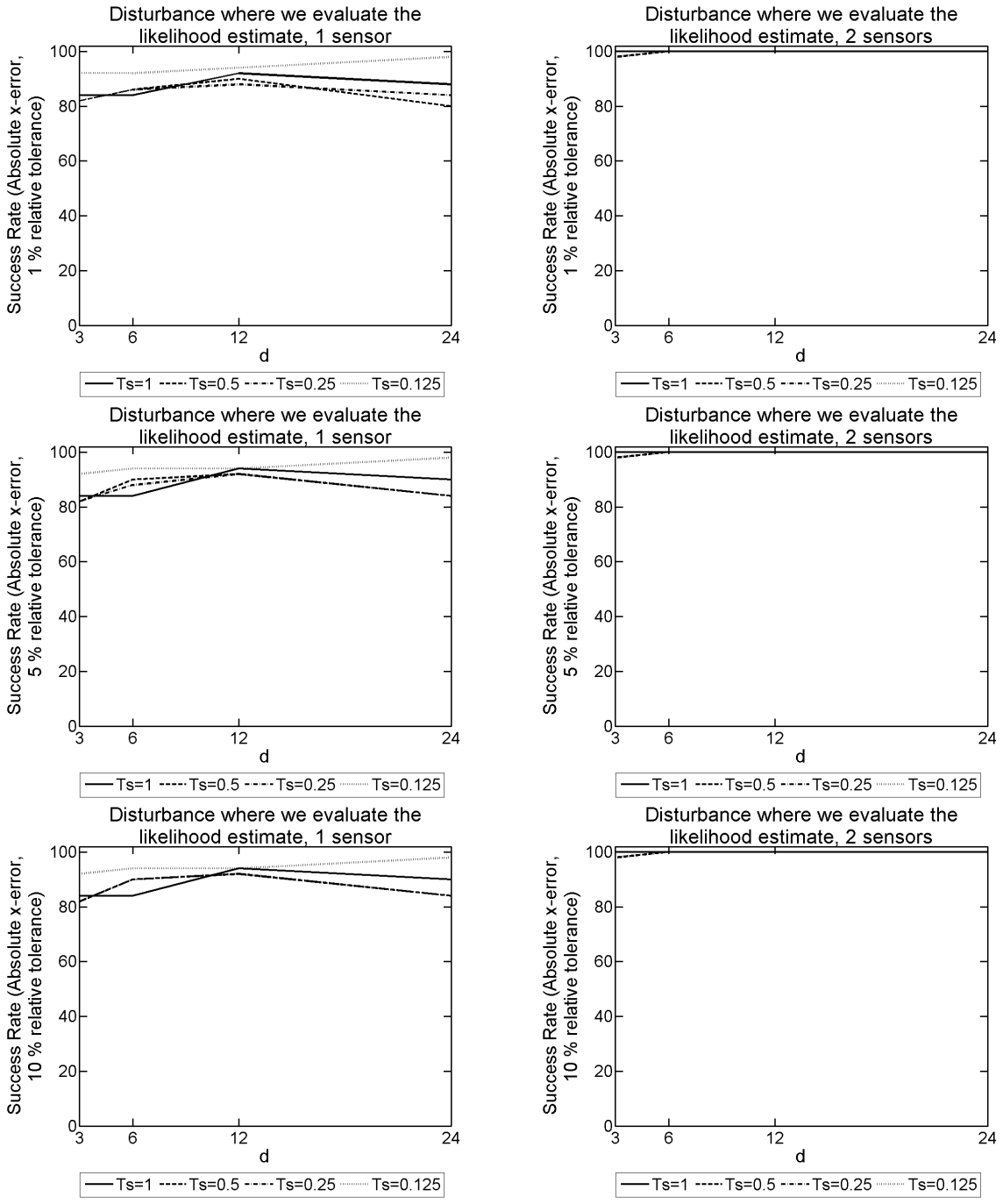


Figure 9.5: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

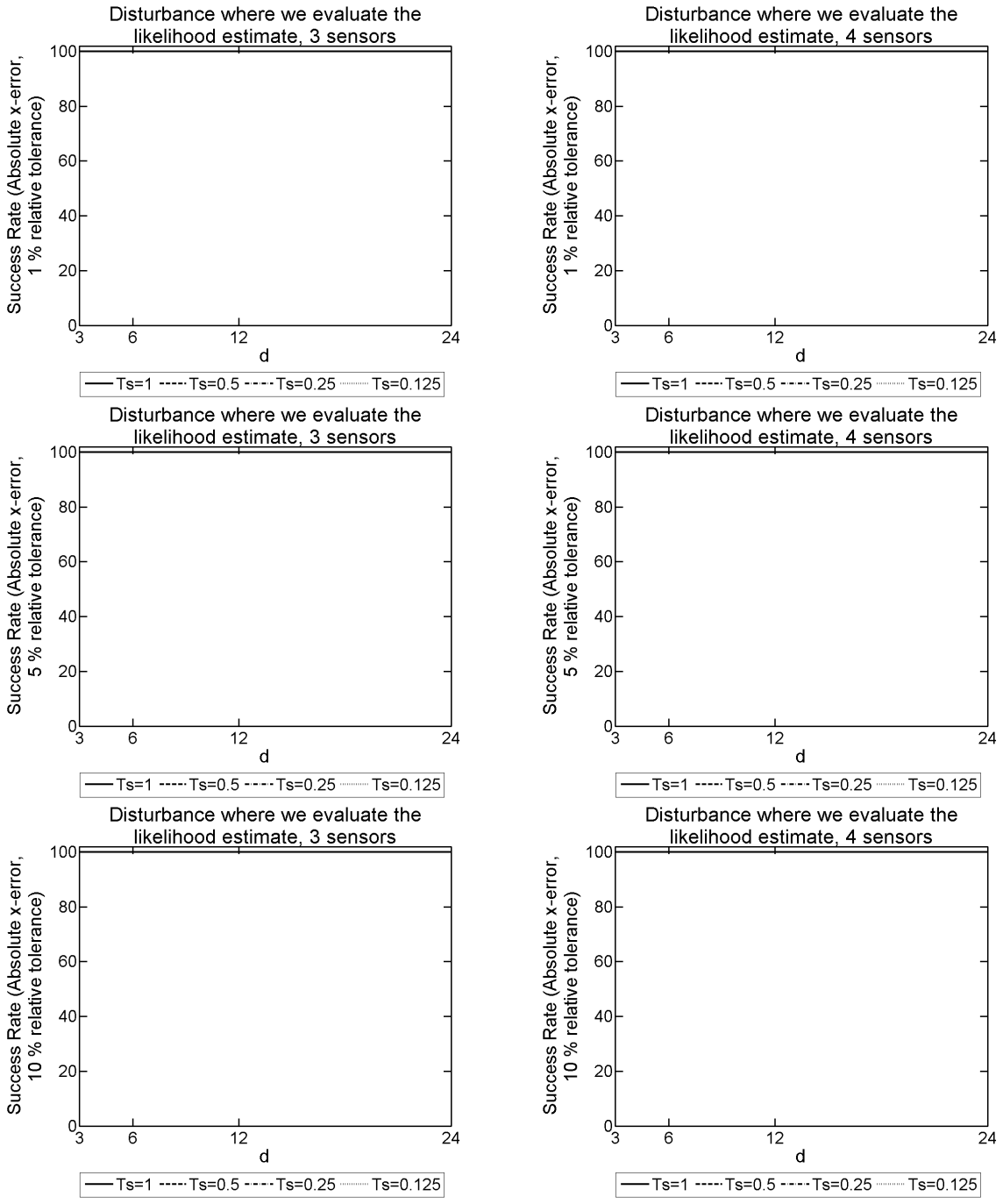


Figure 9.6: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

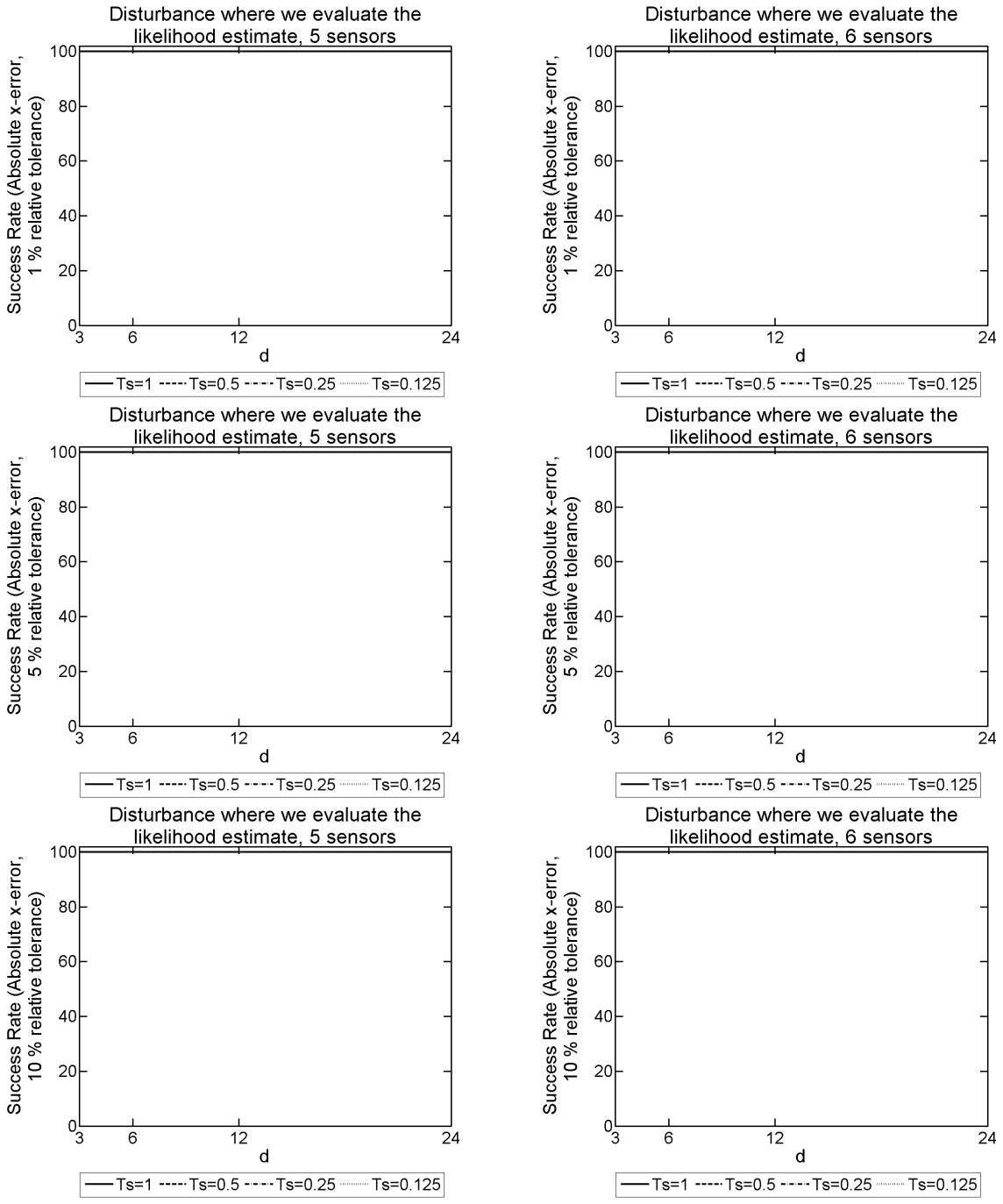


Figure 9.7: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

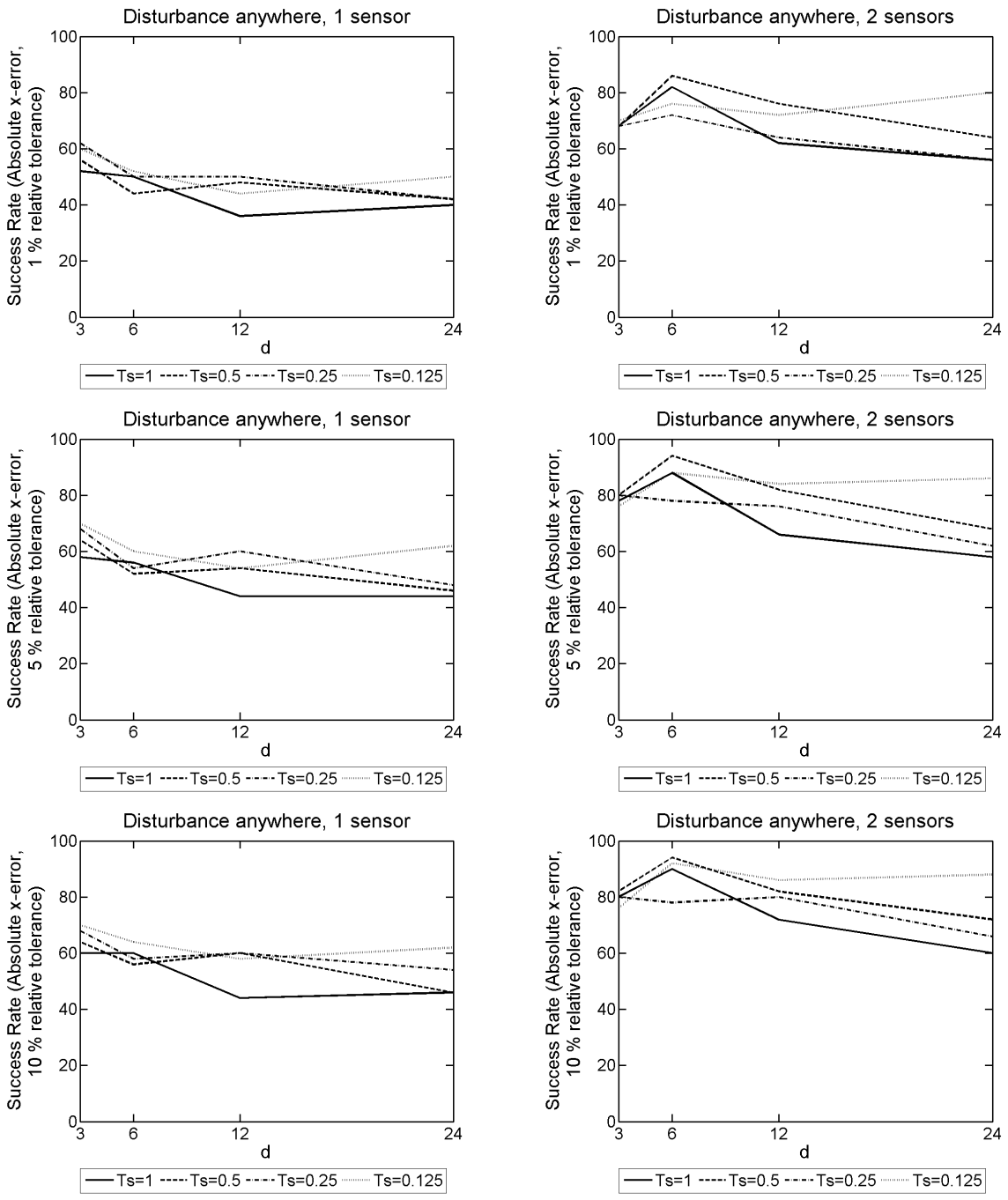


Figure 9.8: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

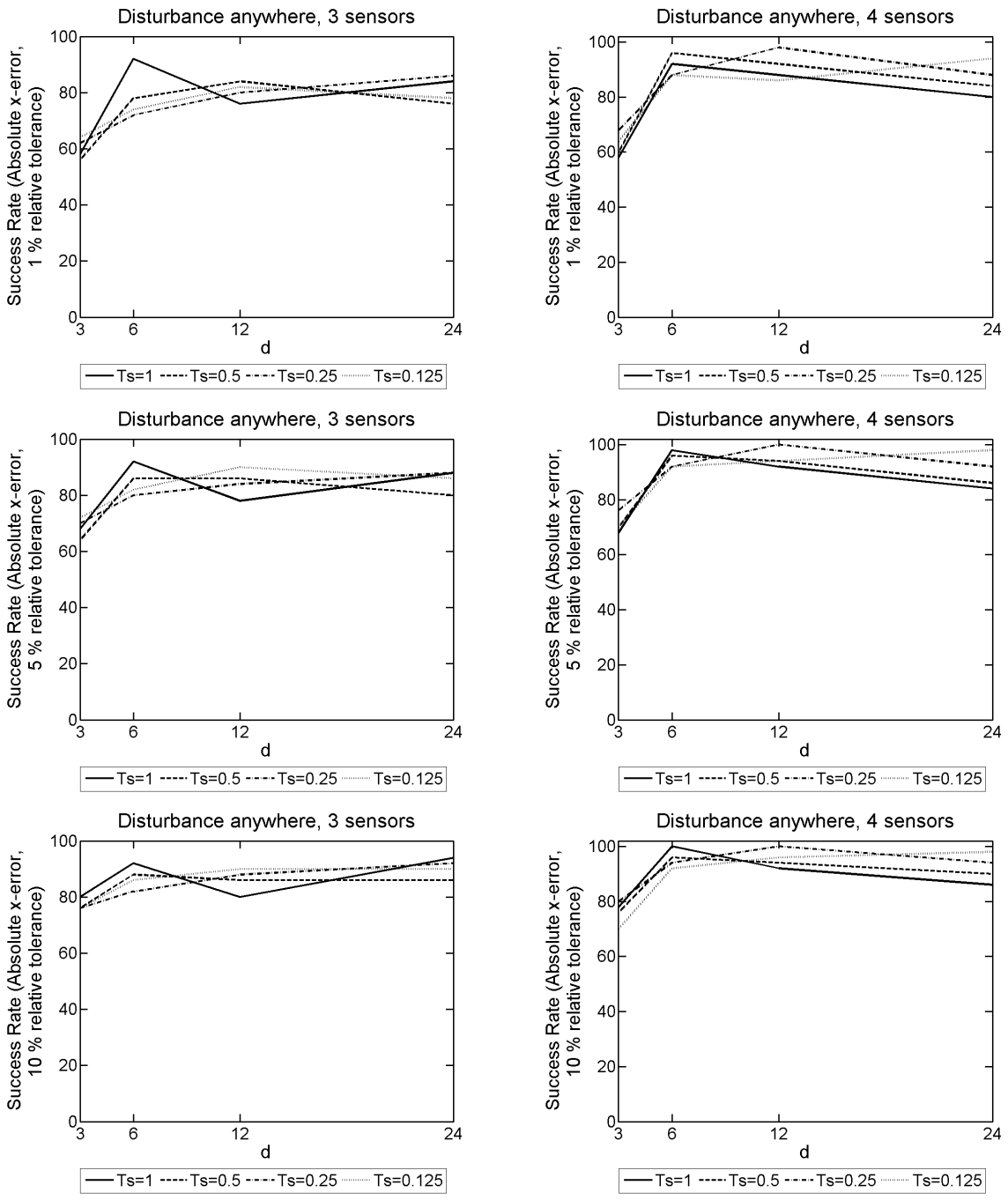


Figure 9.9: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

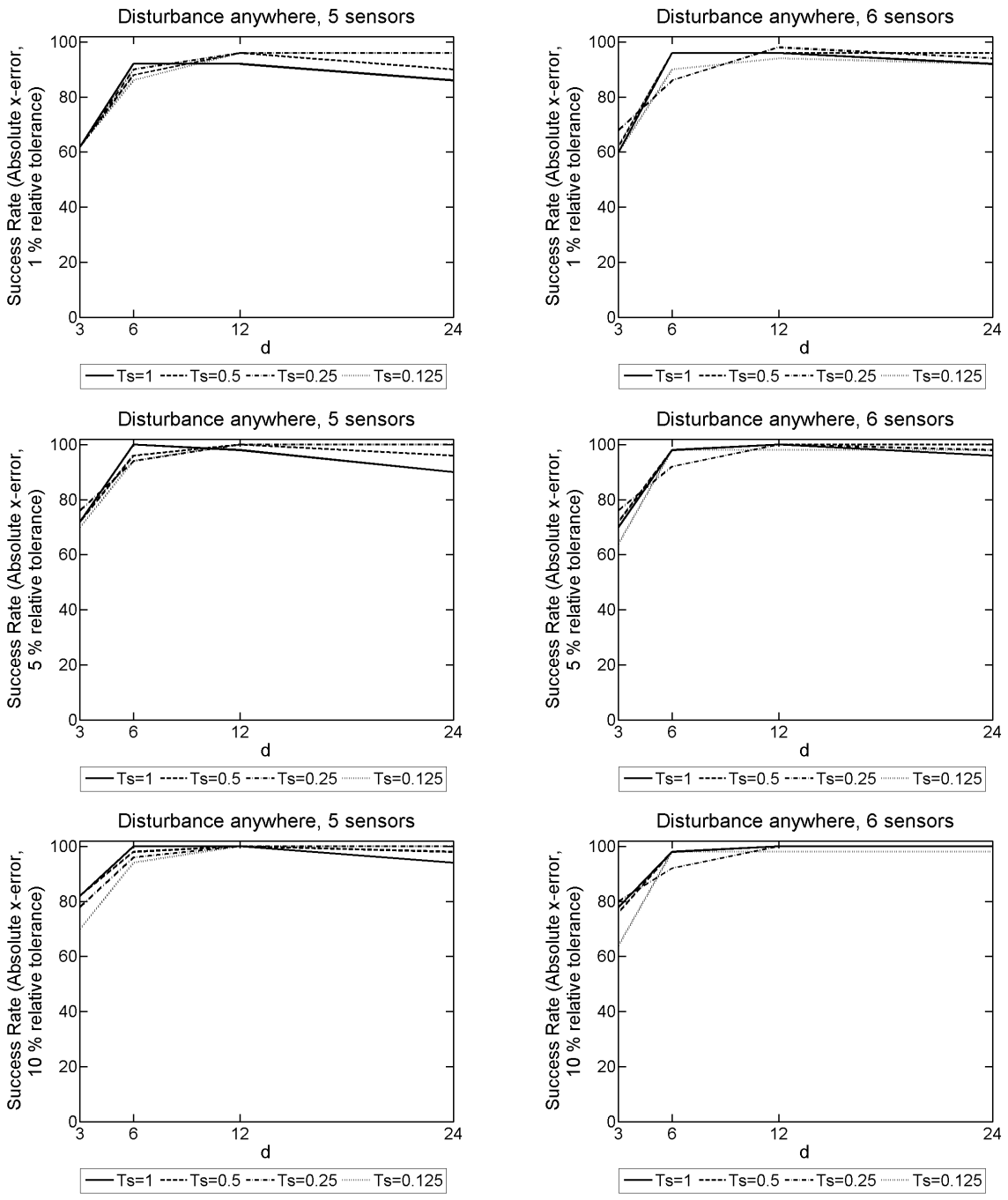


Figure 9.10: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 250$ and $L = 45000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	50	52	54
1.000	6	38	42	42
1.000	12	28	38	40
1.000	24	34	40	44
0.500	3	38	52	52
0.500	6	42	52	52
0.500	12	36	42	44
0.500	24	32	40	44
0.250	3	44	52	54
0.250	6	38	50	50
0.250	12	34	46	46
0.250	24	32	42	44
0.125	3	36	52	54
0.125	6	34	42	42
0.125	12	44	52	54
0.125	24	42	50	50

Table 9.21: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	76	80	84
1.000	6	92	92	92
1.000	12	84	88	88
1.000	24	76	84	86
0.500	3	74	78	82
0.500	6	82	86	88
0.500	12	80	82	82
0.500	24	70	74	74
0.250	3	72	78	78
0.250	6	80	80	80
0.250	12	82	88	88
0.250	24	86	88	88
0.125	3	54	64	66
0.125	6	74	82	82
0.125	12	90	94	94
0.125	24	92	94	94

Table 9.22: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	98	98	98
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	98	98	98
0.500	3	94	94	94
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	94	96	96
0.125	6	98	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.23: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.24: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	84	86	86
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	88	92	92
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	86	88	88
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	56	64	64
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.25: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	98	98	98
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.26: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	40	54	54
1.000	6	42	54	54
1.000	12	36	58	58
1.000	24	30	44	50
0.500	3	34	52	56
0.500	6	36	52	52
0.500	12	32	46	48
0.500	24	34	44	48
0.250	3	34	52	52
0.250	6	32	56	56
0.250	12	38	52	54
0.250	24	30	40	44
0.125	3	32	44	46
0.125	6	32	44	46
0.125	12	38	52	52
0.125	24	42	54	56

Table 9.27: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	42	54	56
1.000	6	64	72	72
1.000	12	54	66	70
1.000	24	58	64	66
0.500	3	40	56	60
0.500	6	68	78	78
0.500	12	56	70	72
0.500	24	62	66	68
0.250	3	42	62	68
0.250	6	64	72	72
0.250	12	78	88	90
0.250	24	60	68	72
0.125	3	40	60	64
0.125	6	52	74	74
0.125	12	72	86	86
0.125	24	66	78	80

Table 9.28: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	48	54	56
1.000	6	62	72	76
1.000	12	80	86	86
1.000	24	72	72	76
0.500	3	48	54	62
0.500	6	66	74	74
0.500	12	72	82	82
0.500	24	80	82	86
0.250	3	52	56	56
0.250	6	70	80	80
0.250	12	86	92	92
0.250	24	88	88	88
0.125	3	54	72	72
0.125	6	70	72	72
0.125	12	78	88	88
0.125	24	88	88	88

Table 9.29: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	42	48	52
1.000	6	76	80	80
1.000	12	92	94	94
1.000	24	84	86	86
0.500	3	44	56	62
0.500	6	74	82	82
0.500	12	94	96	96
0.500	24	92	94	94
0.250	3	50	58	60
0.250	6	76	84	84
0.250	12	92	94	94
0.250	24	90	92	92
0.125	3	52	70	72
0.125	6	82	84	84
0.125	12	92	96	96
0.125	24	92	94	94

Table 9.30: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	34	46	48
1.000	6	82	88	88
1.000	12	92	100	100
1.000	24	94	96	96
0.500	3	40	48	54
0.500	6	74	86	86
0.500	12	90	100	100
0.500	24	90	98	98
0.250	3	44	58	60
0.250	6	82	90	90
0.250	12	92	98	98
0.250	24	98	100	100
0.125	3	46	68	74
0.125	6	78	96	96
0.125	12	96	98	98
0.125	24	88	96	96

Table 9.31: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	46	52	54
1.000	6	78	86	86
1.000	12	98	100	100
1.000	24	96	98	98
0.500	3	48	58	64
0.500	6	84	92	92
0.500	12	98	100	100
0.500	24	98	100	100
0.250	3	48	54	54
0.250	6	82	92	92
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	54	66	68
0.125	6	80	88	90
0.125	12	94	94	94
0.125	24	92	98	98

Table 9.32: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

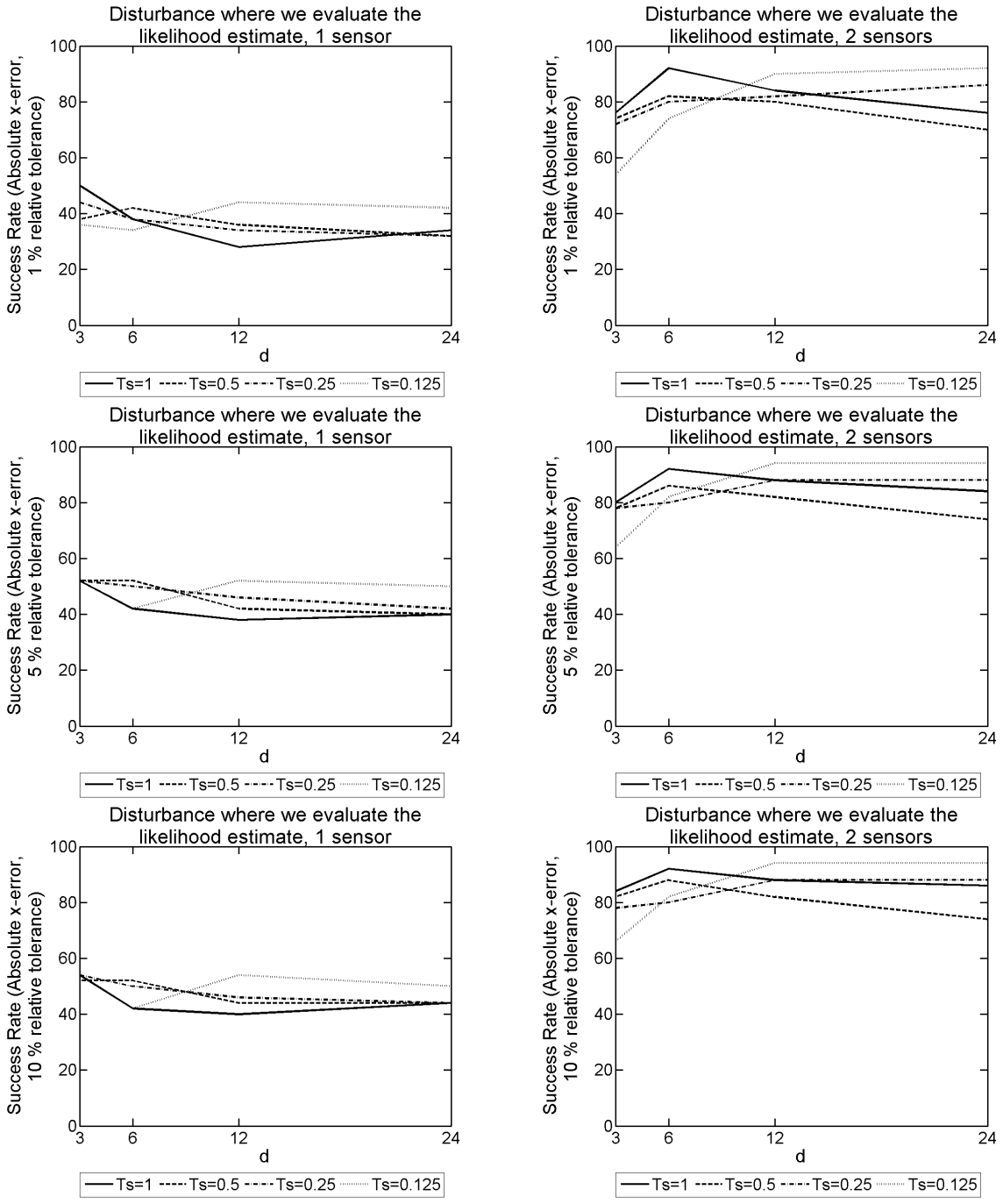


Figure 9.11: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

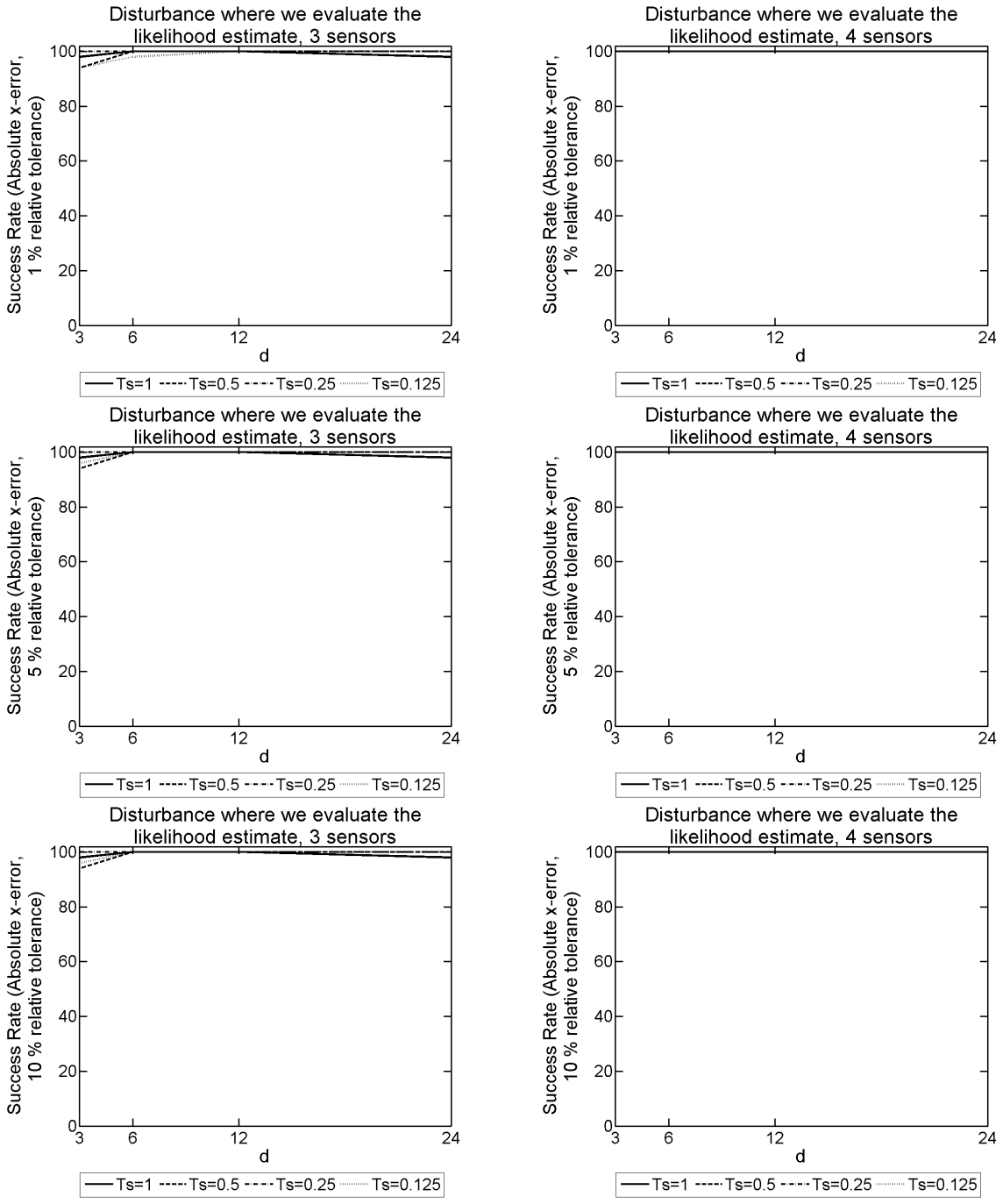


Figure 9.12: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 2 sensors present, whereas on the right-hand side there are 3 sensors present.

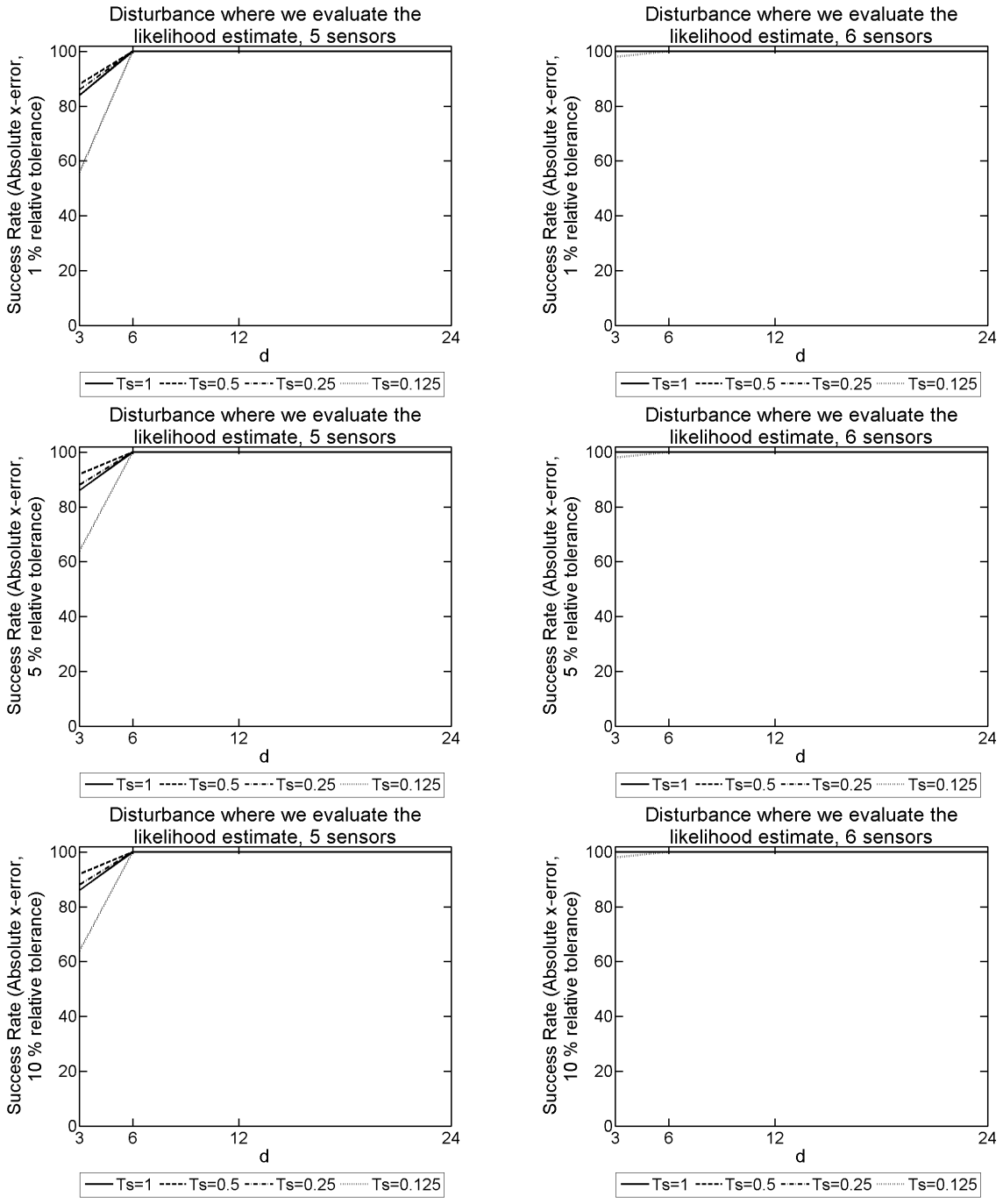


Figure 9.13: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 4 sensors present, whereas on the right-hand side there are 5 sensors present.

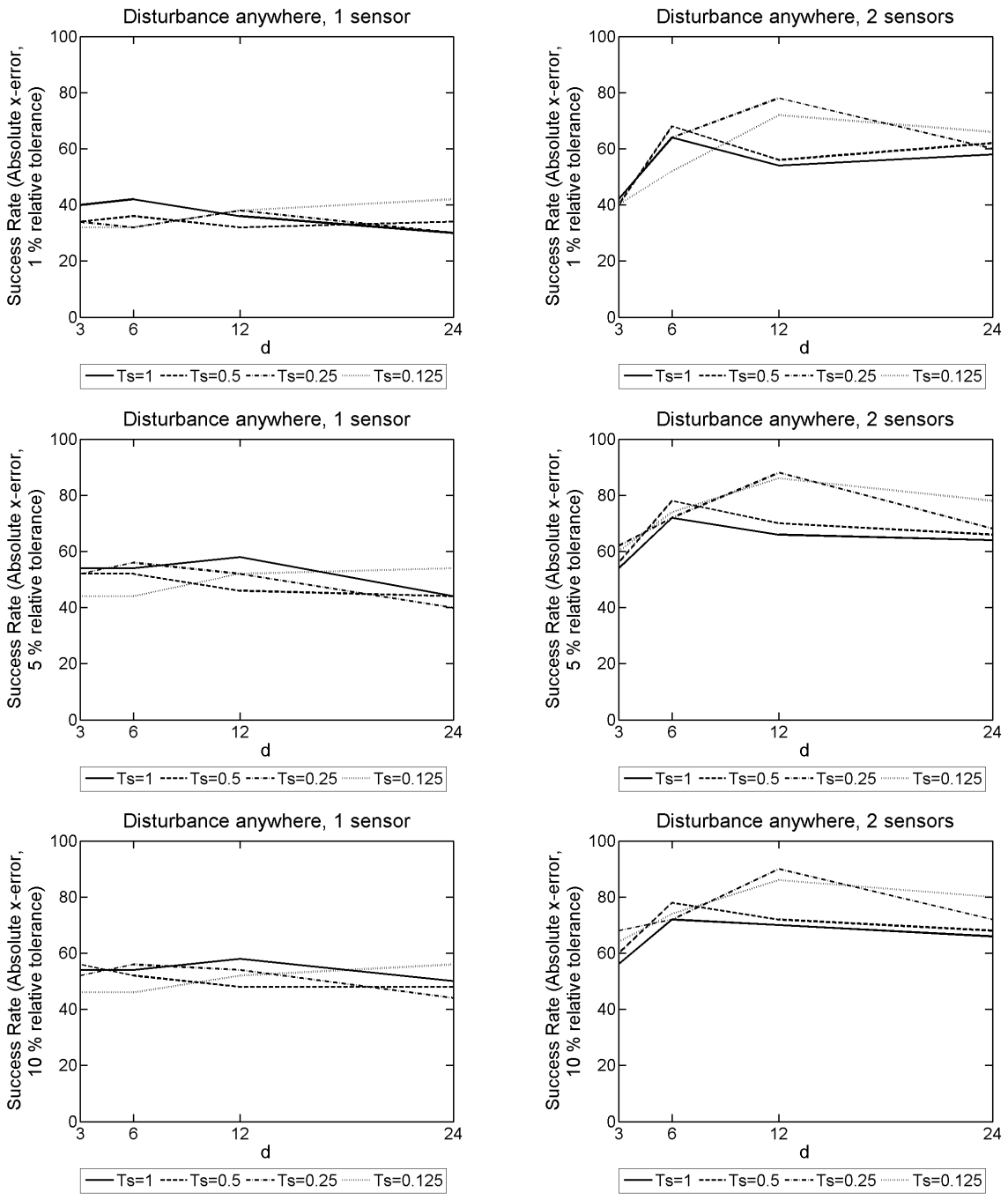


Figure 9.14: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

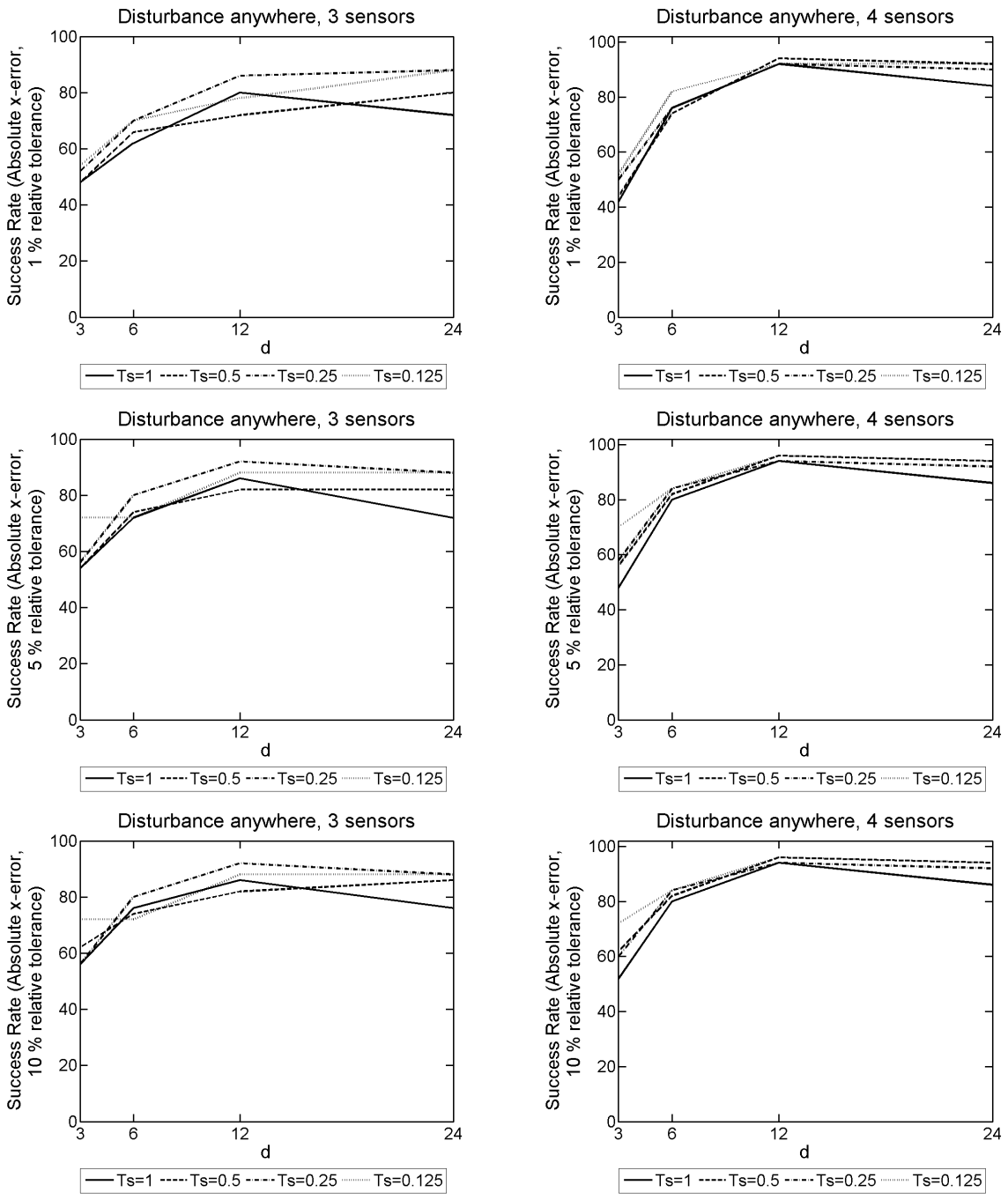


Figure 9.15: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

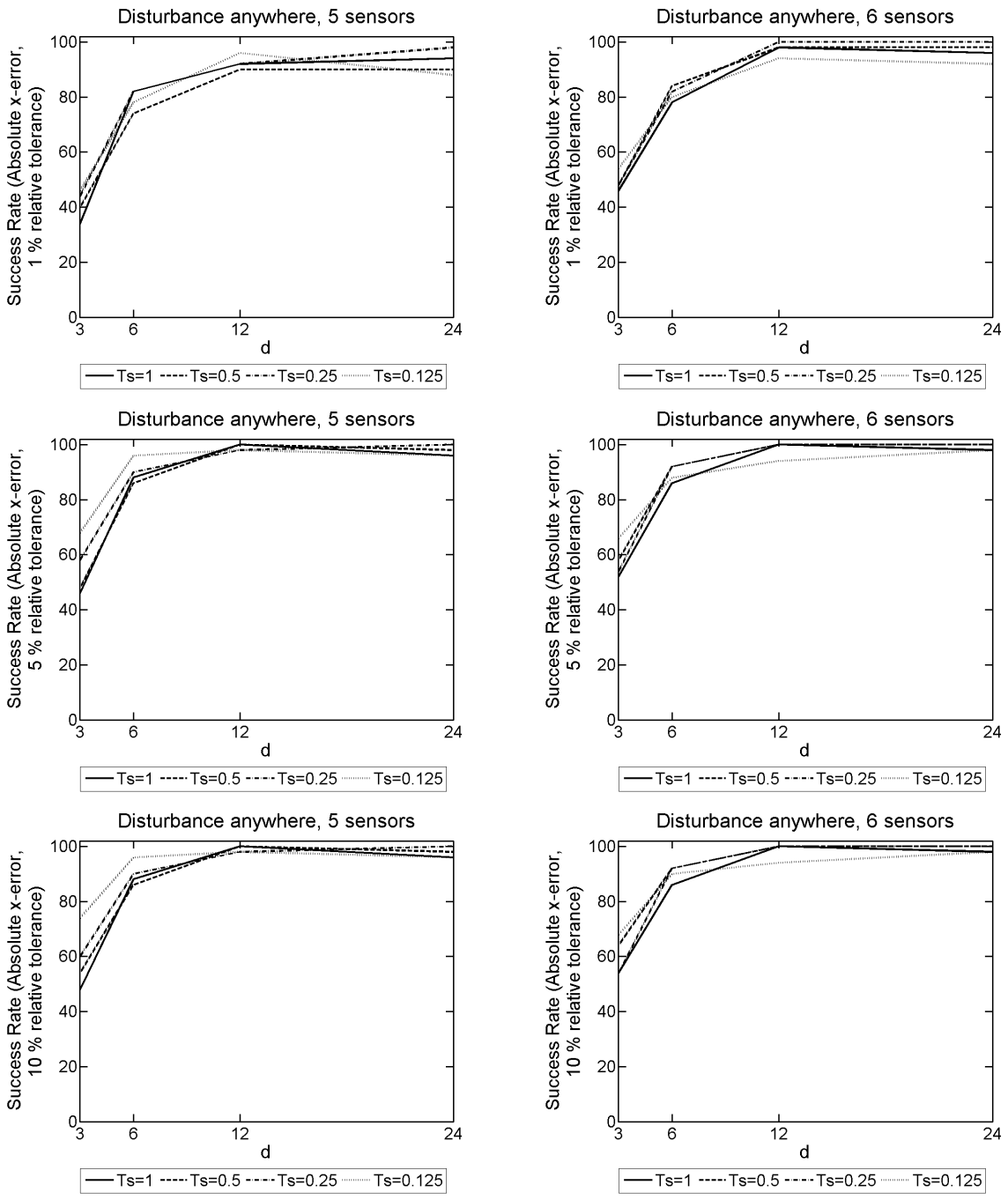


Figure 9.16: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, and $F = 25$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

Appendix B: Results for chapter 4

B.1 Disturbance frequency $F = 150$Hz

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	48	58	60
1.000	6	32	34	44
1.000	12	32	40	50
1.000	24	38	46	46
0.500	3	46	54	60
0.500	6	36	46	50
0.500	12	24	32	44
0.500	24	28	32	40
0.250	3	32	48	56
0.250	6	48	62	64
0.250	12	36	48	54
0.250	24	52	58	64
0.125	3	32	58	62
0.125	6	42	56	60
0.125	12	56	68	72
0.125	24	70	70	76

Table 9.33: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	82	94	94
1.000	6	76	78	78
1.000	12	64	66	68
1.000	24	56	64	68
0.500	3	80	90	92
0.500	6	74	80	82
0.500	12	78	82	82
0.500	24	72	74	80
0.250	3	66	94	94
0.250	6	84	86	86
0.250	12	80	82	84
0.250	24	78	82	86
0.125	3	60	76	82
0.125	6	88	94	98
0.125	12	92	92	92
0.125	24	96	96	98

Table 9.34: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	90	96	98
1.000	6	92	98	98
1.000	12	92	92	96
1.000	24	88	90	90
0.500	3	98	98	98
0.500	6	90	98	98
0.500	12	94	96	96
0.500	24	88	88	90
0.250	3	72	96	100
0.250	6	96	98	98
0.250	12	98	98	98
0.250	24	88	90	90
0.125	3	76	98	100
0.125	6	96	98	98
0.125	12	96	96	96
0.125	24	98	98	98

Table 9.35: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	96	100	100
1.000	6	98	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	96	98	98
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	98	98	98
0.250	3	78	98	98
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	98	98	98
0.125	3	76	92	94
0.125	6	96	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.36: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	98	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	88	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	82	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.37: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	98	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	96	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	80	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	84	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.38: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	50	68	72
1.000	6	18	34	36
1.000	12	12	24	36
1.000	24	12	20	26
0.500	3	42	60	64
0.500	6	20	28	34
0.500	12	16	20	28
0.500	24	22	30	40
0.250	3	26	52	56
0.250	6	36	44	50
0.250	12	16	30	36
0.250	24	18	28	32
0.125	3	24	40	50
0.125	6	30	44	54
0.125	12	32	58	58
0.125	24	44	58	68

Table 9.39: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	82	92	92
1.000	6	58	70	70
1.000	12	40	48	54
1.000	24	24	40	52
0.500	3	80	92	94
0.500	6	66	80	80
0.500	12	36	44	52
0.500	24	20	32	44
0.250	3	60	94	96
0.250	6	72	88	92
0.250	12	36	50	60
0.250	24	24	36	42
0.125	3	50	86	86
0.125	6	56	68	76
0.125	12	48	60	62
0.125	24	62	80	82

Table 9.40: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	88	100	100
1.000	6	82	96	96
1.000	12	70	80	80
1.000	24	32	44	56
0.500	3	86	98	98
0.500	6	78	96	96
0.500	12	64	76	78
0.500	24	44	54	56
0.250	3	70	96	100
0.250	6	78	98	98
0.250	12	72	84	88
0.250	24	56	62	72
0.125	3	64	88	92
0.125	6	82	96	98
0.125	12	80	86	88
0.125	24	80	94	94

Table 9.41: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	80	96	96
1.000	6	80	96	96
1.000	12	82	90	92
1.000	24	60	72	76
0.500	3	90	100	100
0.500	6	80	98	98
0.500	12	68	82	86
0.500	24	62	70	74
0.250	3	60	98	100
0.250	6	86	98	98
0.250	12	76	88	94
0.250	24	56	66	70
0.125	3	70	88	90
0.125	6	90	96	100
0.125	12	92	100	100
0.125	24	84	94	94

Table 9.42: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	92	100	100
1.000	6	88	98	98
1.000	12	80	94	96
1.000	24	62	68	72
0.500	3	88	100	100
0.500	6	90	100	100
0.500	12	72	88	88
0.500	24	68	74	78
0.250	3	72	100	100
0.250	6	92	100	100
0.250	12	82	94	96
0.250	24	70	74	74
0.125	3	70	94	96
0.125	6	88	100	100
0.125	12	90	100	100
0.125	24	94	98	98

Table 9.43: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	90	100	100
1.000	6	90	100	100
1.000	12	86	98	100
1.000	24	60	72	78
0.500	3	92	100	100
0.500	6	90	100	100
0.500	12	80	98	98
0.500	24	74	86	92
0.250	3	72	100	100
0.250	6	92	100	100
0.250	12	88	98	98
0.250	24	72	86	90
0.125	3	58	92	96
0.125	6	94	100	100
0.125	12	92	100	100
0.125	24	92	100	100

Table 9.44: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

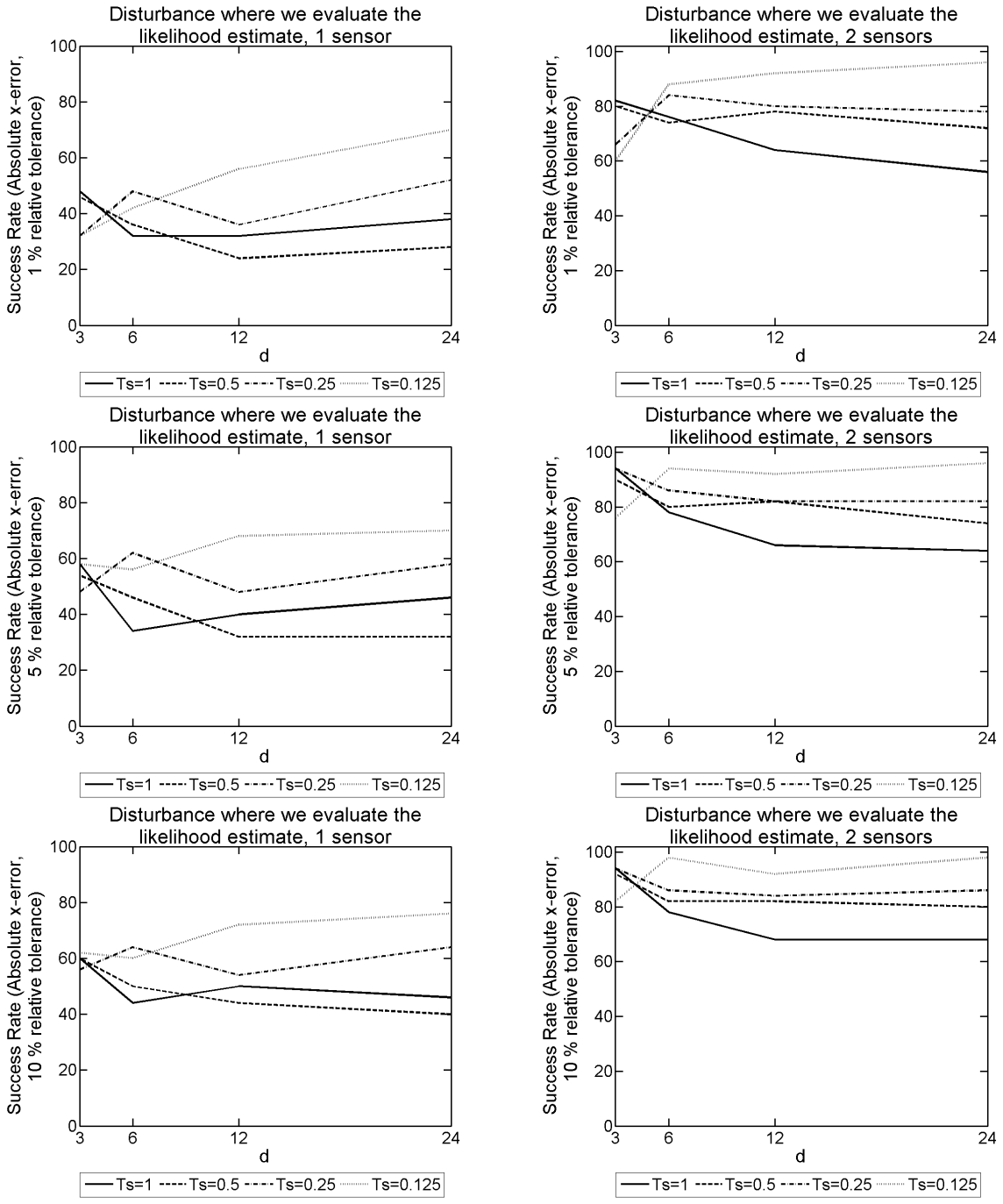


Figure 9.17: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

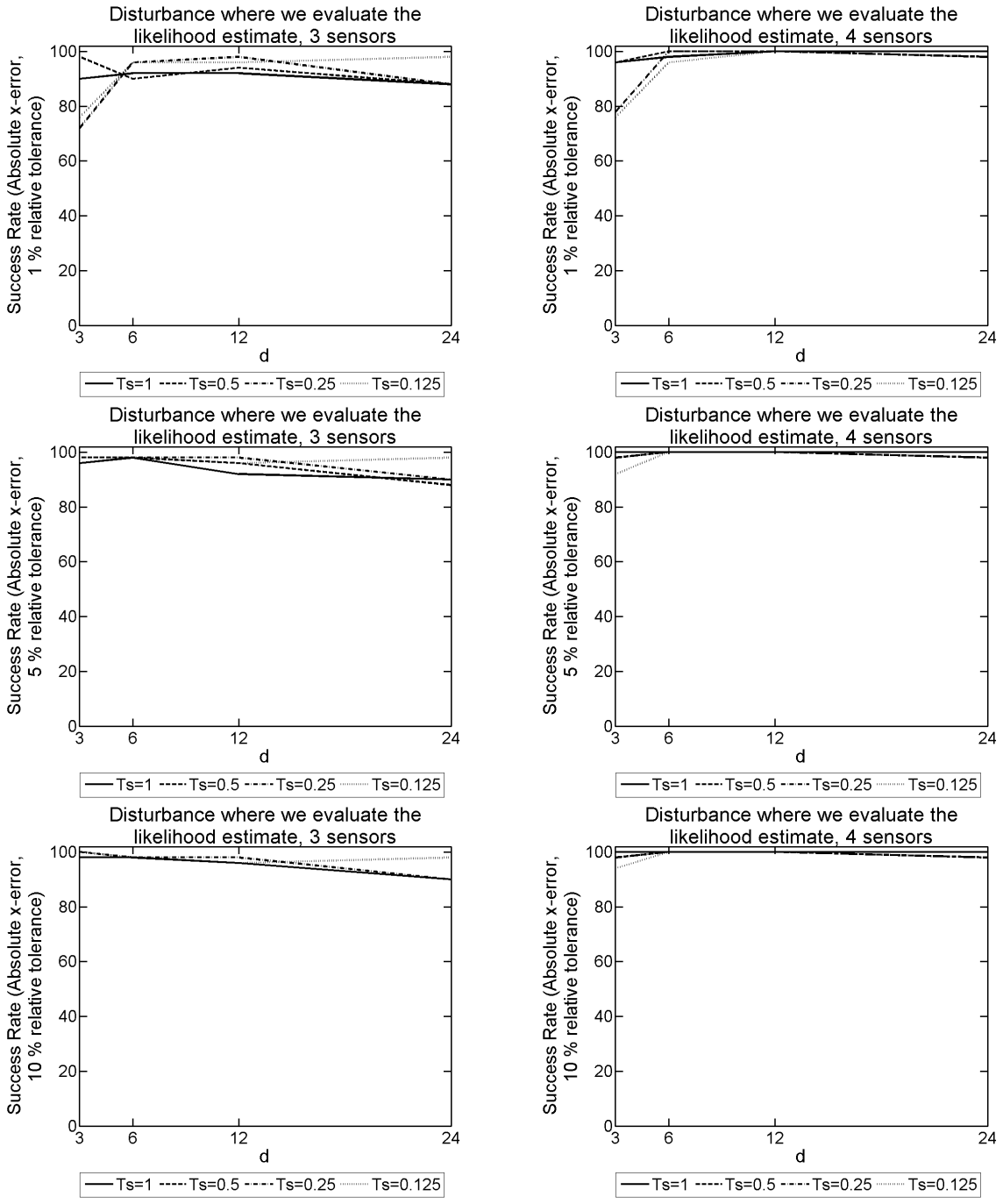


Figure 9.18: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

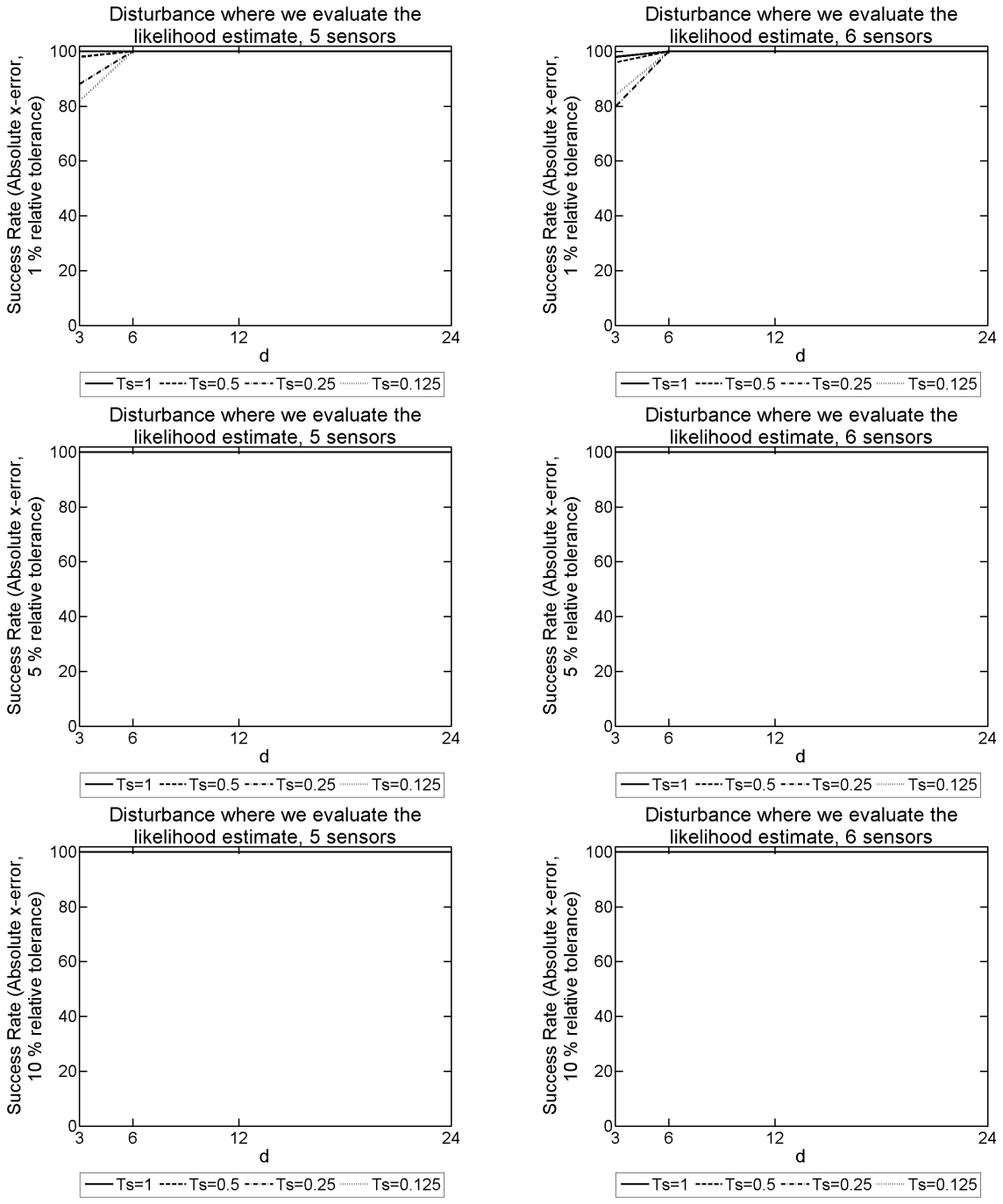


Figure 9.19: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

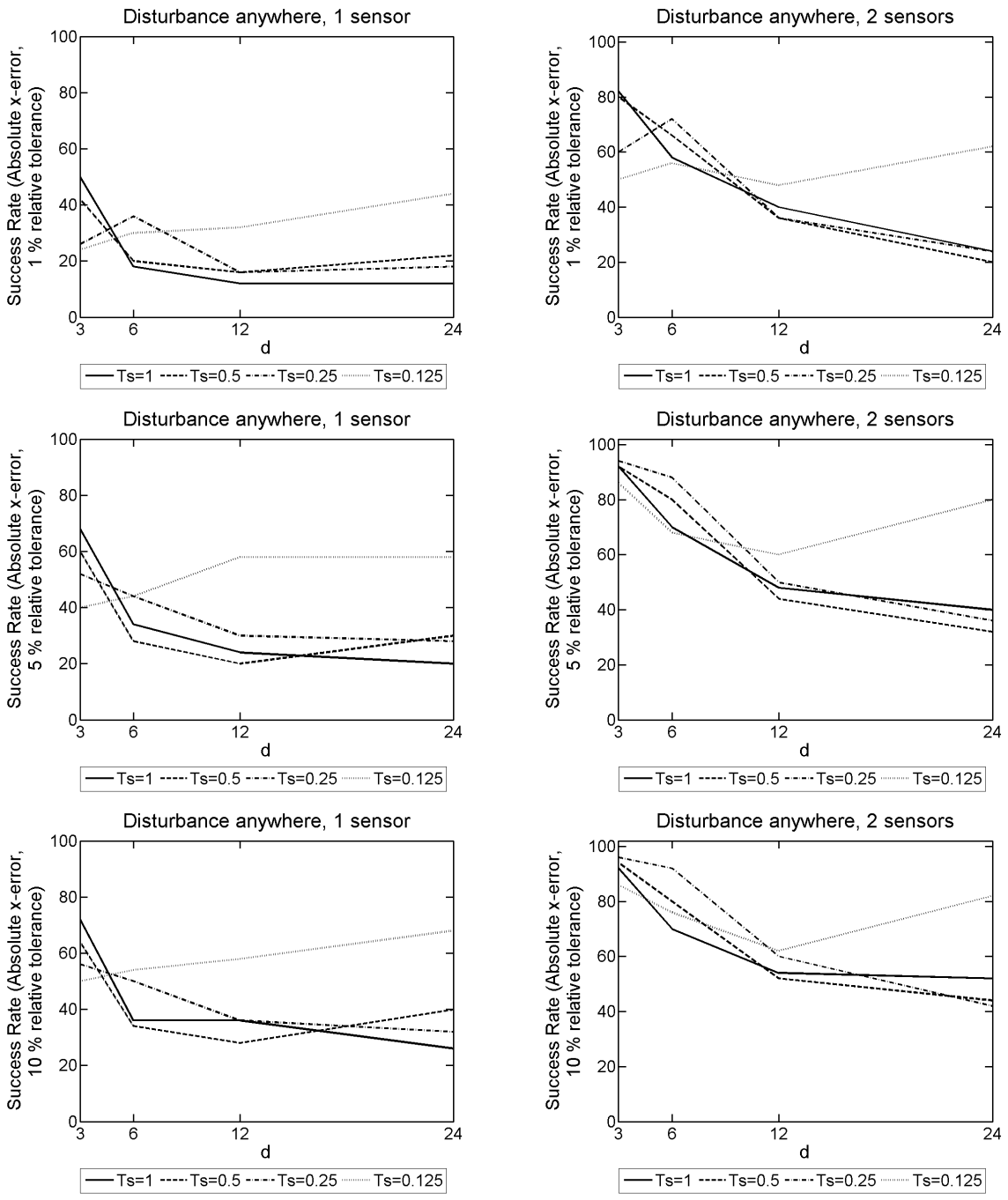


Figure 9.20: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

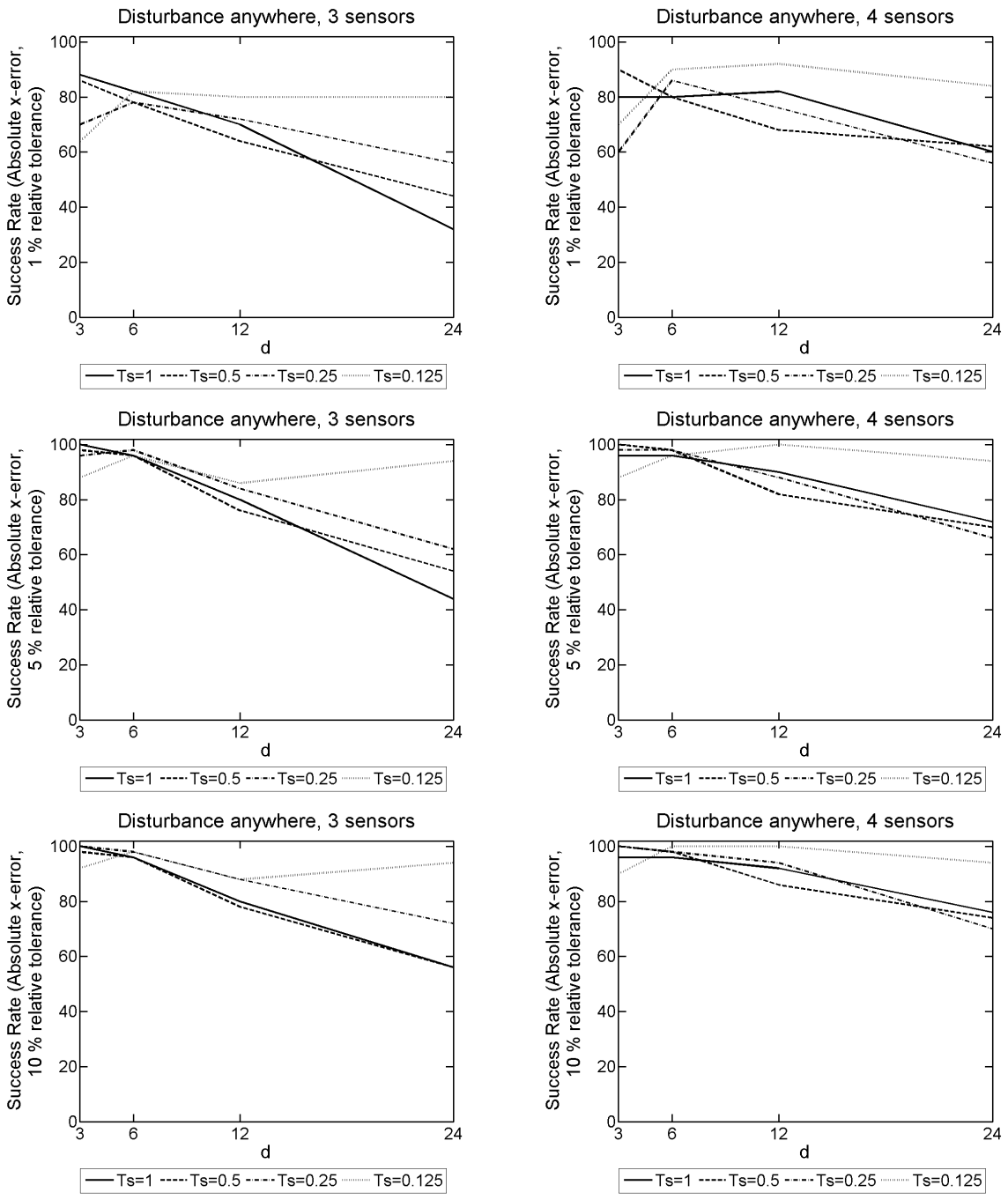


Figure 9.21: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

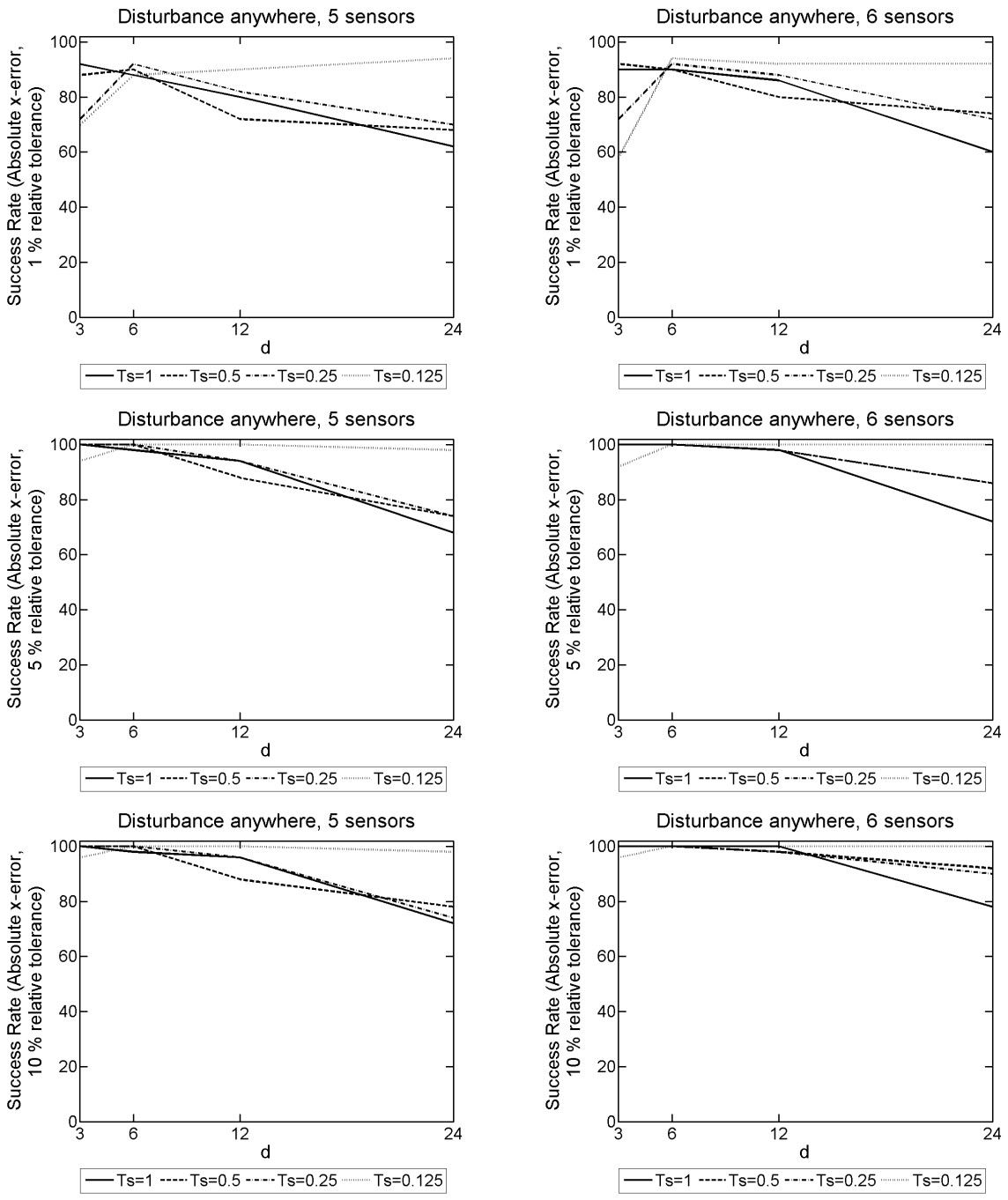


Figure 9.22: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$ and $L = 18000$, and $F = 150$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

B.2 Disturbance frequency $F = 300$Hz

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	58	70	80
1.000	6	62	72	74
1.000	12	56	60	66
1.000	24	48	64	72
0.500	3	58	76	84
0.500	6	60	68	72
0.500	12	50	56	60
0.500	24	40	48	52
0.250	3	68	76	80
0.250	6	64	70	76
0.250	12	52	56	58
0.250	24	60	64	64
0.125	3	70	82	82
0.125	6	72	76	84
0.125	12	68	74	78
0.125	24	70	80	90

Table 9.45: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	94	100	100
1.000	6	90	94	94
1.000	12	92	94	96
1.000	24	78	82	84
0.500	3	90	94	96
0.500	6	100	100	100
0.500	12	92	96	98
0.500	24	80	86	90
0.250	3	92	98	98
0.250	6	96	100	100
0.250	12	96	98	98
0.250	24	82	90	92
0.125	3	84	98	100
0.125	6	98	100	100
0.125	12	98	98	98
0.125	24	98	100	100

Table 9.46: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	96	98	100
1.000	6	98	100	100
1.000	12	100	100	100
1.000	24	92	96	96
0.500	3	94	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	94	94	96
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	96	100	100
0.125	3	98	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.47: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	98	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	98	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	98	100	100
0.125	3	94	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.48: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	98	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	98	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	98	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.49: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	98	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	100	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.50: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	40	50	70
1.000	6	34	40	58
1.000	12	42	56	58
1.000	24	34	44	52
0.500	3	38	58	72
0.500	6	42	54	58
0.500	12	34	40	46
0.500	24	36	46	52
0.250	3	42	64	82
0.250	6	44	58	64
0.250	12	44	50	60
0.250	24	34	52	62
0.125	3	64	76	80
0.125	6	50	62	68
0.125	12	50	62	68
0.125	24	40	48	56

Table 9.51: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	84	98	100
1.000	6	80	88	88
1.000	12	64	76	84
1.000	24	68	78	78
0.500	3	78	94	96
0.500	6	78	88	90
0.500	12	62	70	78
0.500	24	62	78	82
0.250	3	88	94	96
0.250	6	88	88	94
0.250	12	78	86	88
0.250	24	60	72	78
0.125	3	88	96	98
0.125	6	78	94	94
0.125	12	84	88	90
0.125	24	82	96	98

Table 9.52: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	94	100	100
1.000	6	94	98	98
1.000	12	82	94	94
1.000	24	70	78	84
0.500	3	86	100	100
0.500	6	92	96	100
0.500	12	88	96	96
0.500	24	80	90	94
0.250	3	92	100	100
0.250	6	96	100	100
0.250	12	92	98	98
0.250	24	88	96	96
0.125	3	90	96	96
0.125	6	94	98	98
0.125	12	96	100	100
0.125	24	90	96	98

Table 9.53: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	86	100	100
1.000	6	100	100	100
1.000	12	88	98	98
1.000	24	78	88	98
0.500	3	86	100	100
0.500	6	96	100	100
0.500	12	86	96	98
0.500	24	76	86	96
0.250	3	90	100	100
0.250	6	92	100	100
0.250	12	92	98	98
0.250	24	82	94	100
0.125	3	90	100	100
0.125	6	94	100	100
0.125	12	96	100	100
0.125	24	94	100	100

Table 9.54: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	94	100	100
1.000	6	98	100	100
1.000	12	96	100	100
1.000	24	98	100	100
0.500	3	92	100	100
0.500	6	96	100	100
0.500	12	100	100	100
0.500	24	98	100	100
0.250	3	96	100	100
0.250	6	100	100	100
0.250	12	98	100	100
0.250	24	98	100	100
0.125	3	100	100	100
0.125	6	96	100	100
0.125	12	96	100	100
0.125	24	100	100	100

Table 9.55: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	96	100	100
1.000	6	96	100	100
1.000	12	96	100	100
1.000	24	96	100	100
0.500	3	92	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	96	100	100
0.250	3	94	100	100
0.250	6	98	100	100
0.250	12	96	100	100
0.250	24	94	100	100
0.125	3	94	100	100
0.125	6	98	100	100
0.125	12	92	100	100
0.125	24	96	100	100

Table 9.56: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

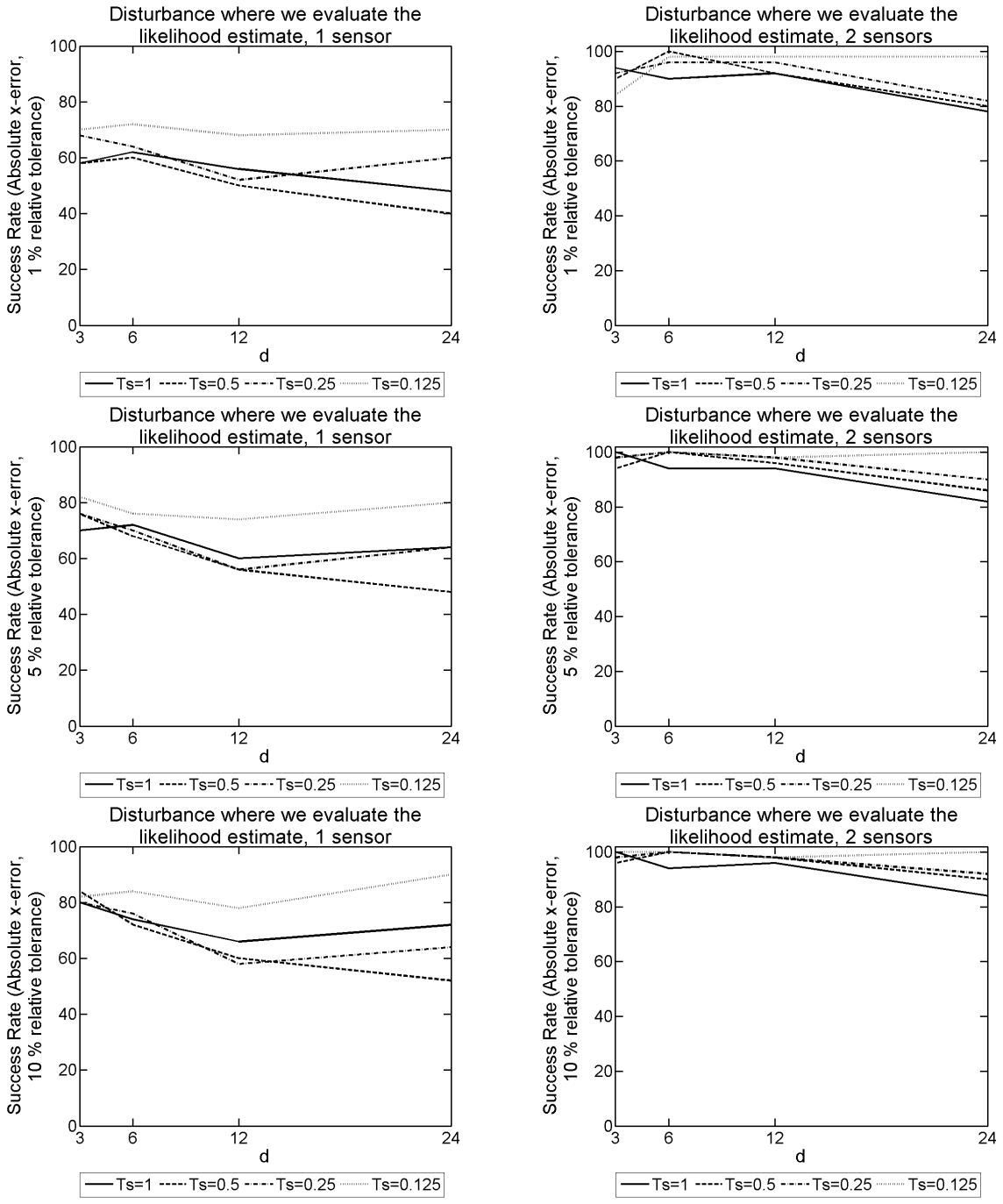


Figure 9.23: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

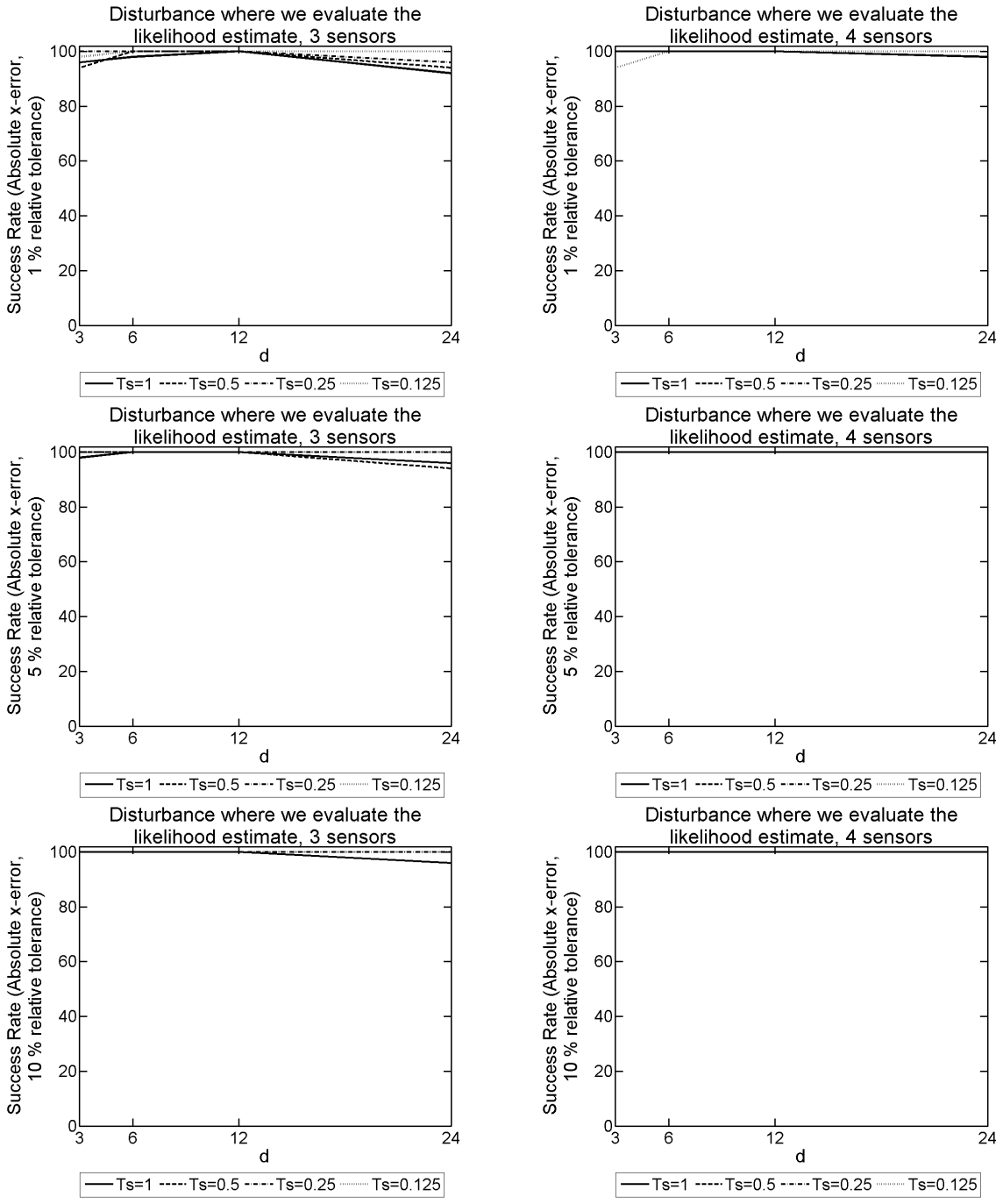


Figure 9.24: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

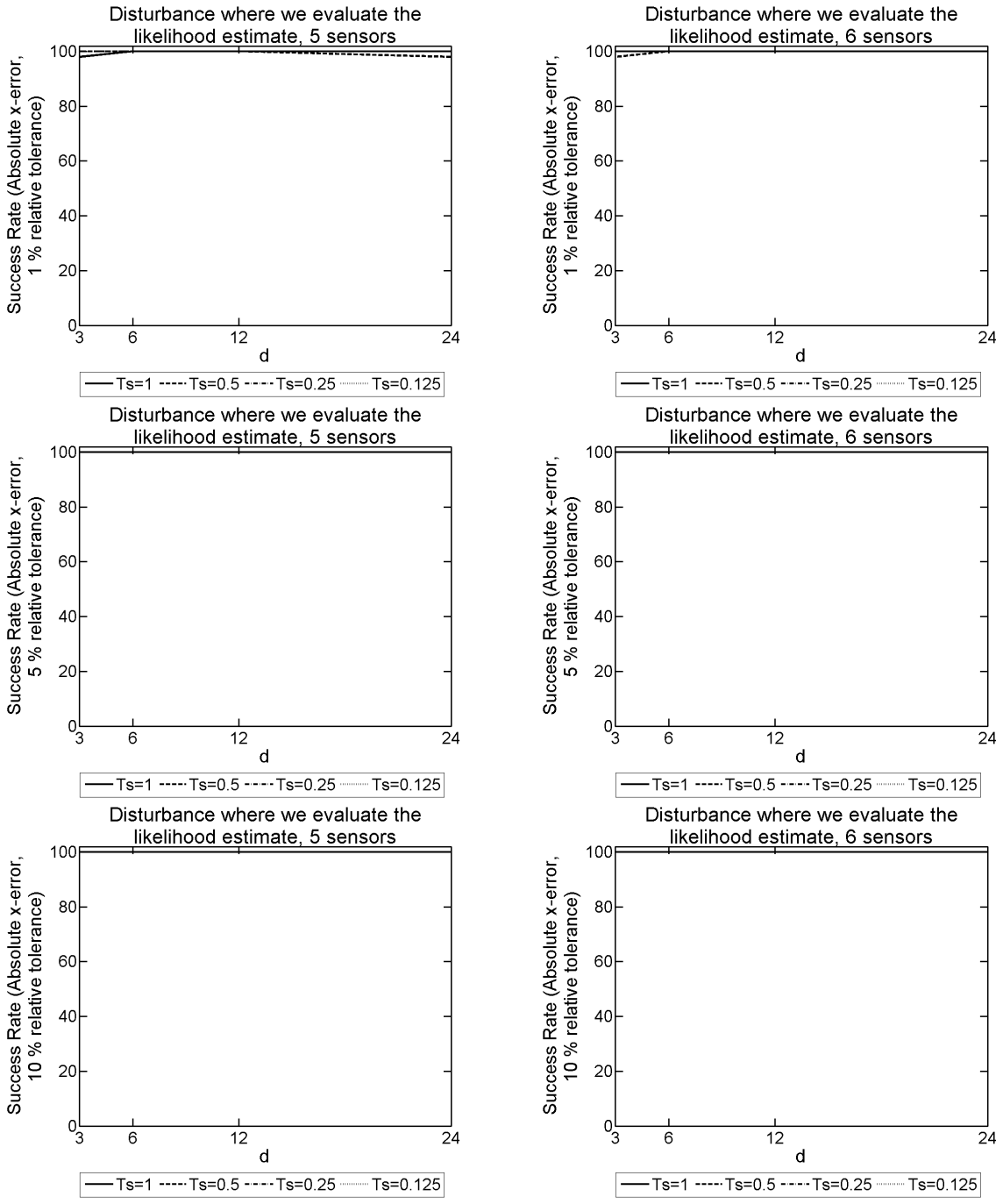


Figure 9.25: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

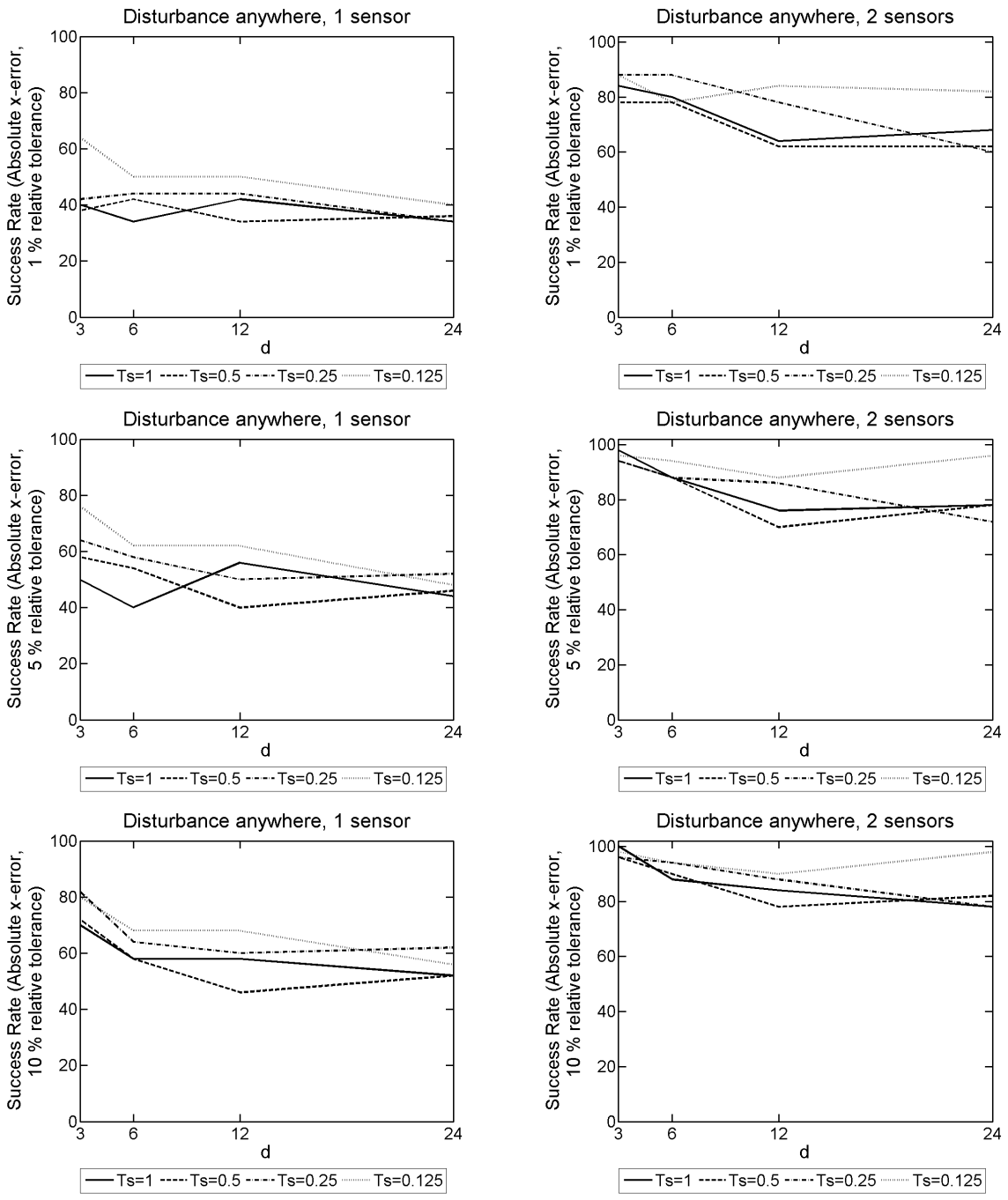


Figure 9.26: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 1 sensor present, whereas on the right-hand side there are 2 sensors present.

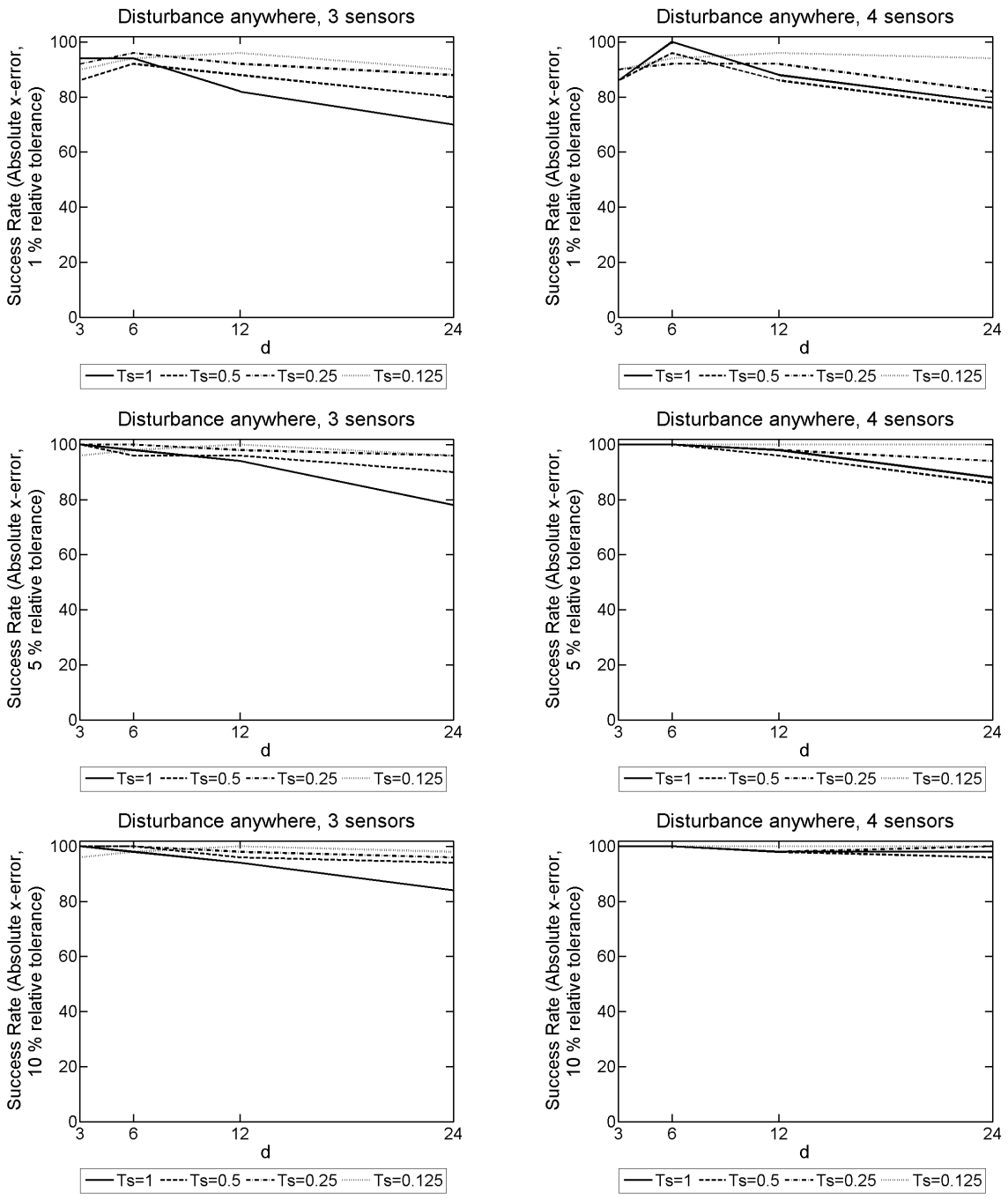


Figure 9.27: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 3 sensors present, whereas on the right-hand side there are 4 sensors present.

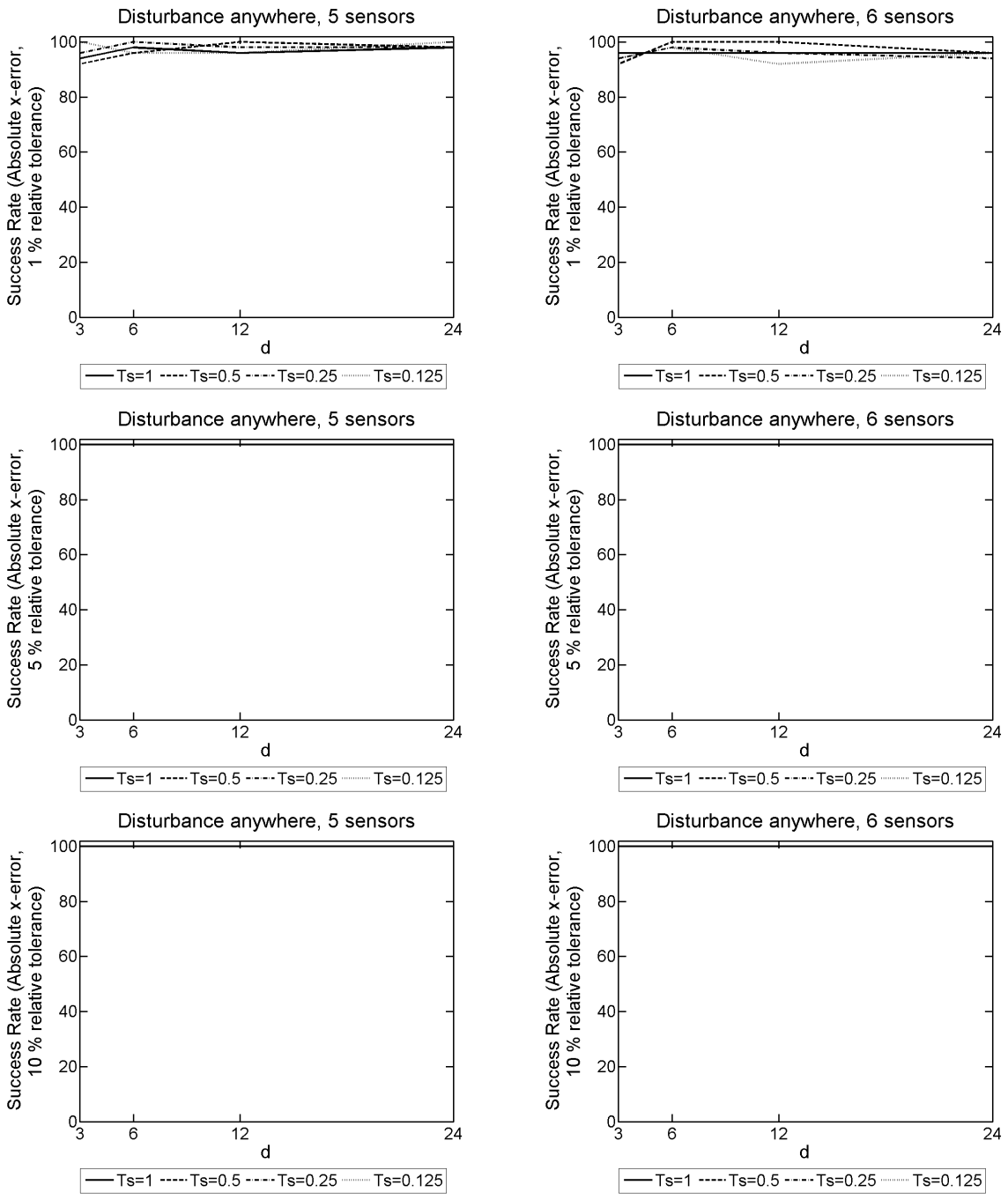


Figure 9.28: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 400$ and $L = 36000$, and $F = 300$Hz. These probabilistic results come from 50 randomly generated disturbance locations. The results on the left-hand side have 5 sensors present, whereas on the right-hand side there are 6 sensors present.

Appendix C: Results for chapter 5

C.1 Without using a minimisation algorithm, 1NBC model problem

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	51	54	55	100	100	100	100	55	57	62	64
2	100	100	100	100	100	100	100	100	100	100	100
3	100	100	100	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100

Table 9.57: 2D model problem with 1NBC: The success rate without the use of a minimisation algorithm for a varying number of sensors, mesh dimensions of $N = 50$, $M = 5$ and $L = 3000$, $F = 25$Hz, and 100 disturbance locations positioned where the likelihood function is evaluated.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	26	41	46	41	49	59	67	21	27	32	42
2	65	88	92	53	60	67	75	49	55	62	71
3	73	94	98	56	65	73	77	53	63	71	77
4	76	93	97	62	64	70	75	61	64	70	74
5	77	92	95	71	76	81	86	69	75	79	85

Table 9.58: 2D model problem with 1NBC: The success rate without the use of a minimisation algorithm for a varying number of sensors, mesh dimensions of $N = 50$, $M = 5$ and $L = 3000$, $F = 25$Hz, and 100 random disturbance locations.

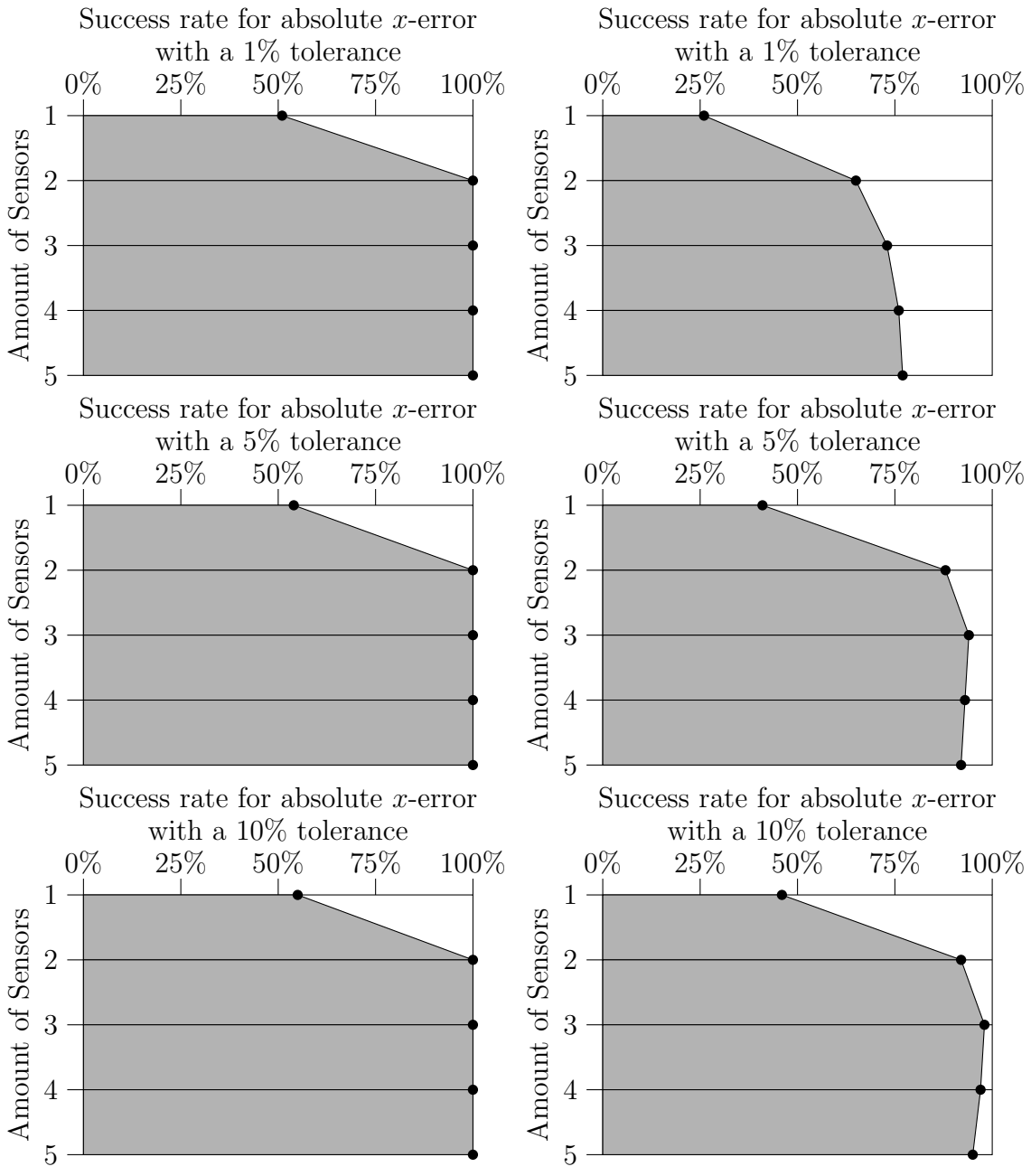


Figure 9.29: The 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for an array of sensors, when $F = 25$Hz. Results on the LHS correspond to 100 disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

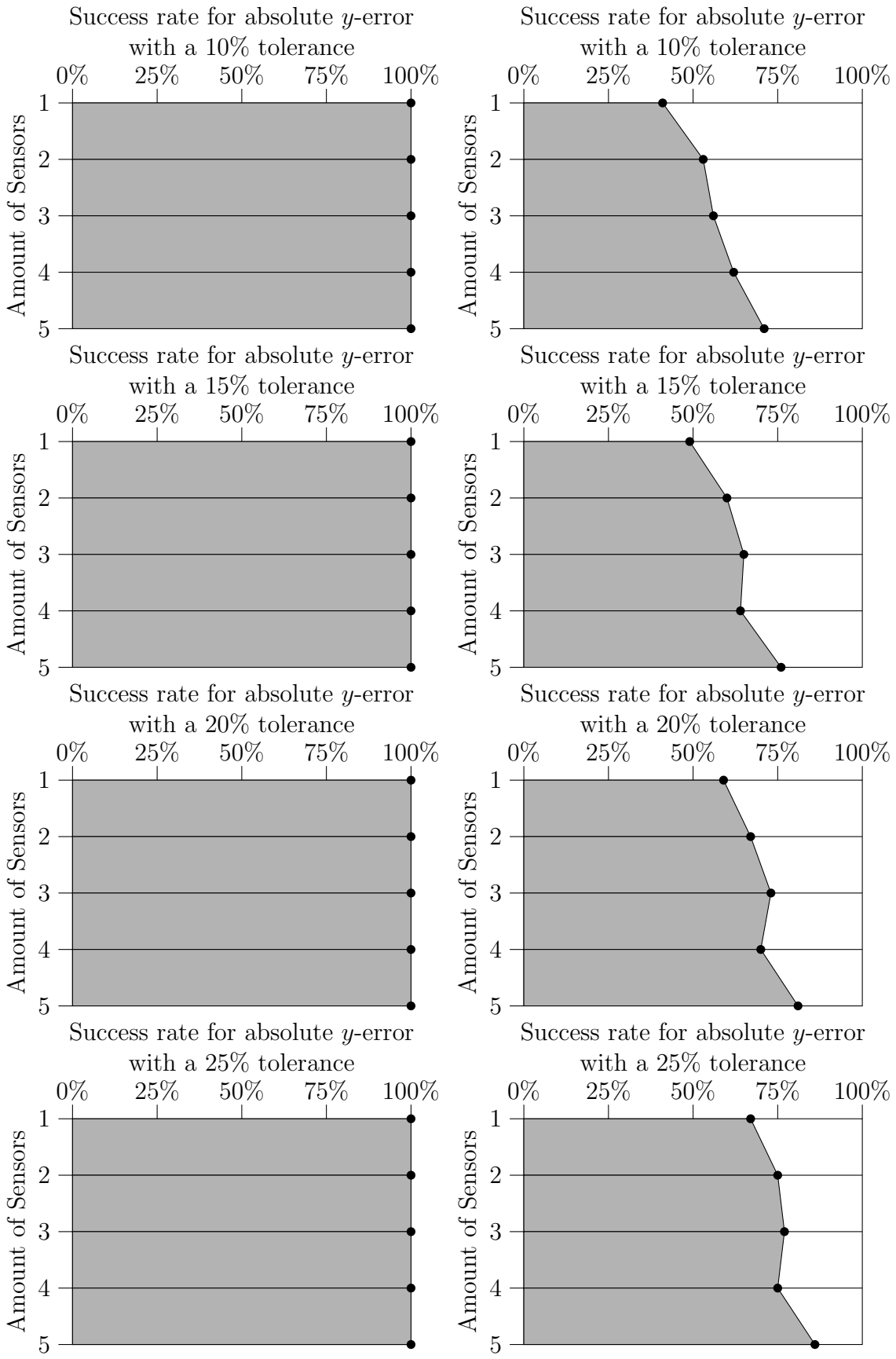


Figure 9.30: The 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ for an array of sensors, when $F = 25$Hz. Results on the LHS correspond to 100 disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

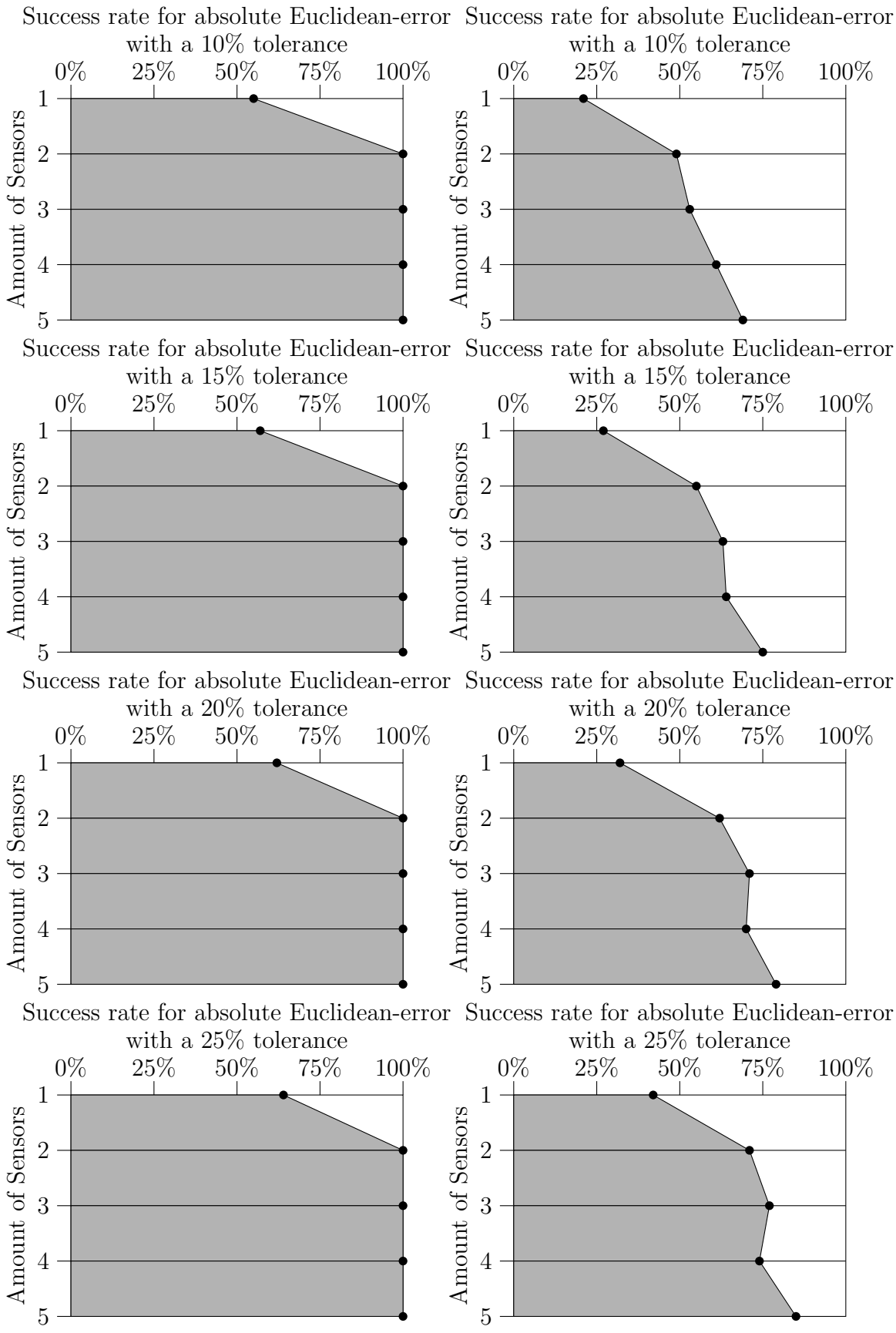


Figure 9.31: The 2D model problem with 1NBC: The success rate given |Euclidean-error| for an array of sensors, when $F = 25$Hz. Results on the LHS correspond to 100 disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

C.2 Without using a minimisation algorithm, 3NBCs model problem

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	52	53	54	95	95	99	99	52	53	59	62
2	98	100	100	96	96	99	99	96	96	98	99
3	99	100	100	99	99	100	100	99	99	99	100
4	100	100	100	99	99	100	100	99	99	100	100
5	100	100	100	100	100	100	100	100	100	100	100

Table 9.59: 2D model problem with 3NBCs: The success rate without the use of a minimisation algorithm for a varying number of sensors, mesh dimensions of $N = 50$, $M = 5$ and $L = 3000$, $F = 25$Hz, and 100 disturbance locations positioned where the likelihood function is evaluated.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	31	48	52	33	41	47	52	17	24	31	37
2	67	93	93	41	49	56	62	39	47	54	60
3	80	98	98	55	70	75	80	53	68	73	80
4	74	97	97	58	72	77	80	57	71	77	80
5	80	99	99	62	75	80	83	62	75	80	83

Table 9.60: 2D model problem with 3NBCs: The success rate without the use of a minimisation algorithm for a varying number of sensors, mesh dimensions of $N = 50$, $M = 5$ and $L = 3000$, $F = 25$Hz, and 100 random disturbance locations.

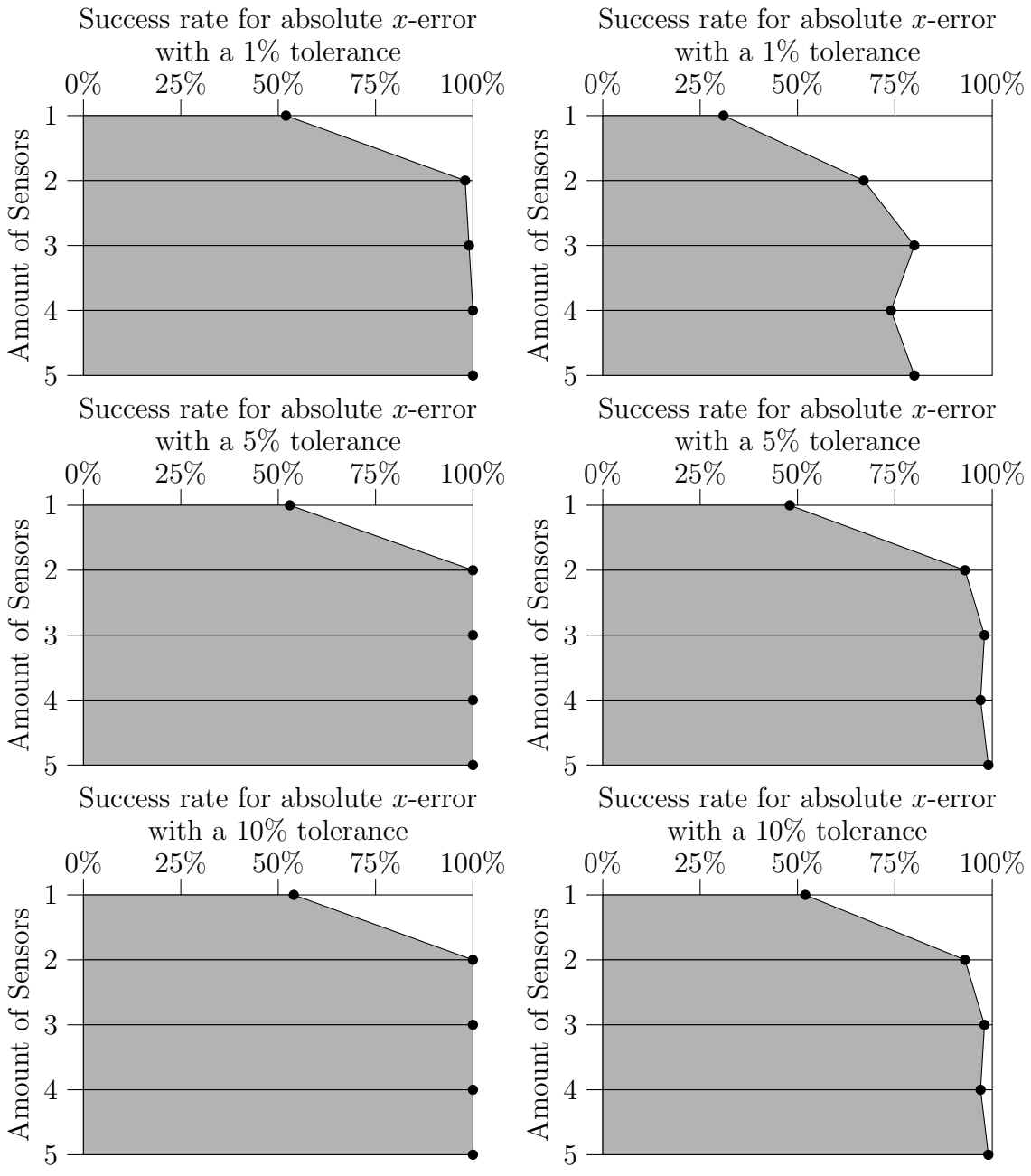


Figure 9.32: The 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for an array of sensors, when $F = 25$Hz. Results on the LHS correspond to 100 disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

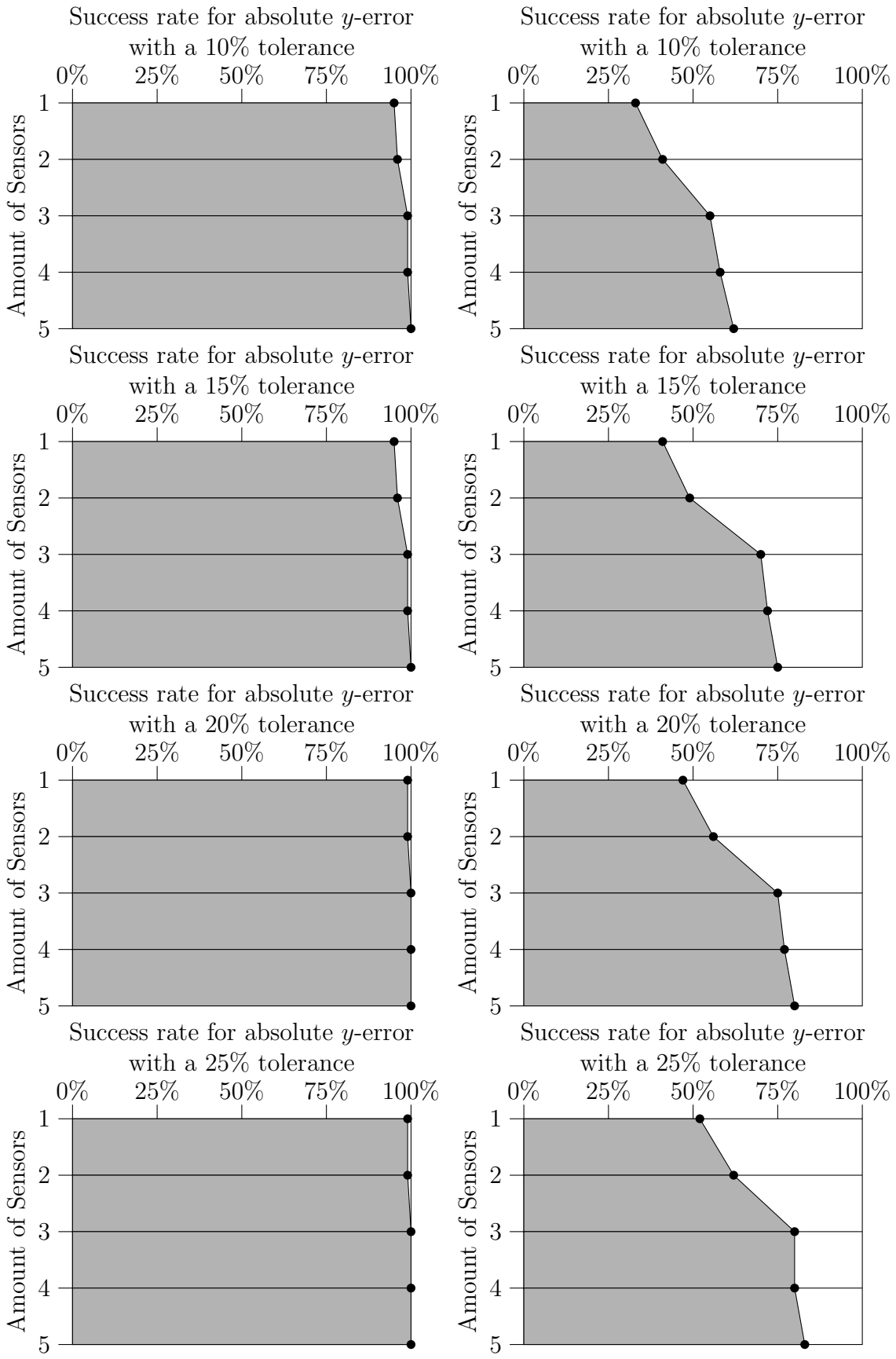


Figure 9.33: The 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ for an array of sensors, when $F = 25$Hz. Results on the LHS correspond to 100 disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

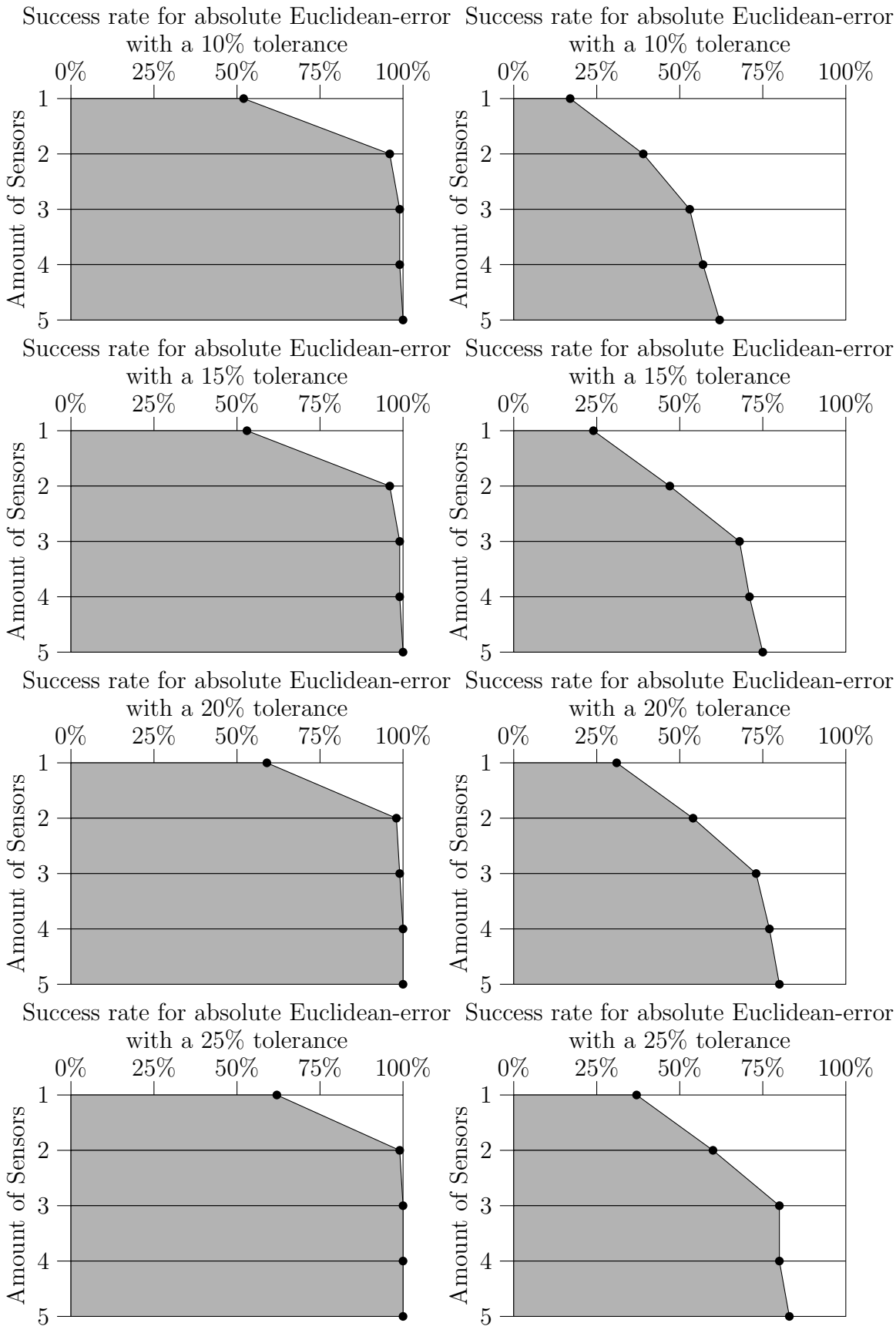


Figure 9.34: The 2D model problem with 3NBCs: The success rate given |Euclidean-error| for a varying sensor quantity, when $F = 25$Hz. Results on the LHS correspond to 100 disturbance locations where (2.107) is evaluated, and the RHS come from 100 random disturbance locations.

C.3 Use of an SVD to reduce matrix dimensions in the KF, 1NBC model problem

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	99	99	99	99	99	100	100	98	98	99	100
2	100	100	100	100	100	100	100	100	100	100	100
3	100	100	100	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100

Table 9.61: The success rate of our 2D model problem with 1NBC. Here we have a mesh dimension of $N = 50$, $M = 5$ and $L = 9000$ over a simulation duration of $T = 3$ seconds, using 100 different disturbance locations positioned where the likelihood function in (2.107) is evaluated. The KF is then run over the final second in an attempt to mimic the scenarios considered when using an SVD.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	50	67	71	42	51	61	68	29	37	45	60
2	63	87	91	43	53	61	70	39	50	57	67
3	73	96	100	56	62	70	75	55	62	70	75
4	76	95	98	62	68	74	79	60	67	73	79
5	81	96	99	76	80	85	88	74	79	84	88
6	76	95	97	77	85	88	92	75	83	85	91

Table 9.62: The success rate of our 2D model problem with 1NBC. Here we have a mesh dimension of $N = 50$, $M = 5$ and $L = 9000$ over a simulation duration of $T = 3$ seconds, using 100 random disturbance locations. The KF is then run over the final second in an attempt to mimic the scenarios considered when using an SVD.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	59	72	75	62	62	80	80	56	56	60	70
1.000	6	46	63	63	59	59	75	75	47	47	53	62
1.000	12	44	61	63	56	56	79	79	46	46	50	59
1.000	24	46	61	64	49	49	77	77	40	40	53	60
0.500	3	57	66	68	59	59	81	81	52	52	60	68
0.500	6	43	59	61	49	49	80	80	40	40	49	58
0.500	12	47	66	68	56	56	85	85	47	47	53	65
0.500	24	45	59	64	54	54	82	82	45	45	53	60
0.250	3	52	64	66	53	53	80	80	48	48	57	63
0.250	6	58	73	73	65	65	81	81	55	55	63	71
0.250	12	52	68	71	61	61	84	84	51	51	61	72
0.250	24	54	68	70	57	57	78	78	46	46	56	63
0.125	3	57	69	70	63	63	88	88	54	54	63	69
0.125	6	59	69	70	62	62	79	79	52	52	61	68
0.125	12	58	69	71	62	62	82	82	54	54	62	72
0.125	24	61	68	70	57	57	83	83	54	54	61	68

Table 9.63: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	97	99	99	95	95	100	100	94	94	98	100
1.000	6	92	97	97	84	84	95	95	84	84	90	93
1.000	12	89	94	94	82	82	93	93	79	79	87	90
1.000	24	84	92	92	80	80	94	94	76	76	85	91
0.500	3	97	98	98	94	94	100	100	94	94	97	98
0.500	6	87	95	96	83	83	95	95	81	81	87	93
0.500	12	77	90	90	75	75	94	94	71	71	79	89
0.500	24	83	92	93	82	82	95	95	80	80	86	91
0.250	3	100	100	100	99	99	100	100	99	99	100	100
0.250	6	96	98	99	87	87	98	98	87	87	95	97
0.250	12	95	99	99	91	91	97	97	91	91	96	96
0.250	24	94	98	99	84	84	96	96	84	84	92	95
0.125	3	100	100	100	99	99	100	100	99	99	100	100
0.125	6	97	99	99	96	96	100	100	96	96	98	99
0.125	12	99	100	100	96	96	99	99	96	96	99	99
0.125	24	93	95	96	89	89	96	96	89	89	94	94

Table 9.64: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	99	99	100	100	99	99	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	99	99	100	100	99	99	100	100
1.000	24	98	100	100	98	98	99	99	98	98	98	99
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	99	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	97	97	100	100	97	97	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	99	99	100	100	99	99	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	99	99	99	99	99	100	100	99	99	99	99
0.125	3	99	99	99	99	99	99	99	99	99	99	99
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.65: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.66: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.67: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.68: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	33	55	59	33	44	50	61	25	34	39	50
1.000	6	42	67	72	44	54	59	68	36	44	49	53
1.000	12	36	57	61	38	47	58	70	25	33	39	51
1.000	24	35	53	58	40	53	61	69	30	38	43	49
0.500	3	27	48	52	38	54	62	73	28	38	44	53
0.500	6	40	60	62	44	51	57	67	34	41	45	48
0.500	12	39	61	63	38	45	55	62	26	33	40	45
0.500	24	39	66	67	36	51	59	65	27	38	44	50
0.250	3	32	57	62	44	56	60	67	33	38	41	50
0.250	6	41	58	61	42	52	61	69	30	39	46	56
0.250	12	42	61	65	41	51	63	71	29	36	46	57
0.250	24	44	64	67	39	52	63	68	28	40	46	55
0.125	3	35	59	61	41	48	53	62	26	33	36	46
0.125	6	43	64	65	41	50	56	64	29	36	40	52
0.125	12	38	62	64	36	53	60	70	25	37	42	52
0.125	24	37	58	61	48	54	59	65	36	41	44	52

Table 9.69: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	63	85	86	61	73	78	83	56	68	73	81
1.000	6	62	89	89	67	74	81	86	66	72	77	83
1.000	12	58	87	89	63	69	78	84	63	68	75	80
1.000	24	58	82	82	56	64	71	79	53	61	66	74
0.500	3	64	87	88	69	81	90	95	61	71	79	85
0.500	6	69	92	93	64	71	81	88	61	68	77	86
0.500	12	60	86	88	66	74	83	90	62	68	76	84
0.500	24	52	84	86	61	69	76	84	55	63	68	79
0.250	3	54	78	80	50	67	72	78	44	58	62	71
0.250	6	71	92	95	66	75	81	85	65	74	80	84
0.250	12	69	92	94	61	71	80	83	58	69	78	81
0.250	24	68	89	91	70	80	87	90	64	73	80	85
0.125	3	61	77	83	52	64	72	78	45	52	59	68
0.125	6	57	83	87	59	70	75	81	54	64	68	75
0.125	12	76	91	93	62	72	79	82	57	68	75	80
0.125	24	68	88	92	70	78	85	90	64	73	80	86

Table 9.70: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	72	93	93	78	86	87	92	75	82	83	92
1.000	6	79	100	100	77	89	91	96	76	89	91	96
1.000	12	72	98	99	75	85	89	96	73	84	88	96
1.000	24	82	98	99	73	83	86	91	71	83	86	90
0.500	3	69	89	90	76	80	83	87	71	75	78	85
0.500	6	77	98	100	82	88	90	95	82	88	90	95
0.500	12	76	99	99	77	88	92	97	75	87	91	96
0.500	24	79	99	99	69	81	85	91	69	81	85	90
0.250	3	73	91	93	65	73	79	83	62	69	75	82
0.250	6	79	95	97	84	91	93	98	83	90	92	96
0.250	12	78	97	100	75	86	89	96	75	86	89	95
0.250	24	76	97	98	72	83	90	93	72	83	90	92
0.125	3	67	88	90	56	65	72	77	52	61	67	72
0.125	6	74	91	91	71	78	83	87	68	72	77	84
0.125	12	80	92	92	73	78	83	89	70	75	79	85
0.125	24	78	95	96	78	85	89	91	76	84	88	91

Table 9.71: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	70	93	93	73	84	87	90	71	81	84	90
1.000	6	84	98	100	80	88	92	95	80	88	92	95
1.000	12	89	97	98	79	88	93	99	78	87	92	97
1.000	24	84	98	99	72	81	86	91	71	80	85	91
0.500	3	77	94	94	77	84	87	90	73	80	83	89
0.500	6	84	95	98	77	86	90	95	77	86	90	94
0.500	12	82	99	99	78	89	94	97	77	88	93	97
0.500	24	85	98	99	85	93	96	98	84	92	95	97
0.250	3	74	91	95	74	79	82	85	72	77	80	83
0.250	6	82	99	100	85	89	91	95	85	89	91	95
0.250	12	82	99	100	79	90	94	97	79	90	94	97
0.250	24	84	100	100	85	92	94	97	84	92	94	97
0.125	3	67	88	93	69	78	81	84	63	73	76	81
0.125	6	79	98	98	83	93	94	96	81	91	92	95
0.125	12	82	94	96	83	90	93	97	81	89	92	94
0.125	24	89	97	98	81	92	95	95	81	91	94	94

Table 9.72: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	73	93	93	75	86	88	91	72	82	84	91
1.000	6	83	100	100	89	94	95	98	89	94	95	98
1.000	12	84	100	100	86	93	93	98	86	93	93	98
1.000	24	83	98	98	84	91	93	97	83	90	92	96
0.500	3	79	94	94	78	84	87	90	75	81	84	87
0.500	6	85	100	100	86	90	93	96	86	90	93	96
0.500	12	87	100	100	81	90	93	97	81	90	93	97
0.500	24	77	99	99	83	89	90	95	82	89	90	95
0.250	3	74	89	91	68	77	79	83	64	72	74	80
0.250	6	90	99	100	89	94	97	99	89	94	97	99
0.250	12	89	100	100	87	92	94	96	87	92	94	96
0.250	24	82	100	100	88	92	95	97	88	92	95	97
0.125	3	67	87	92	67	72	77	79	63	69	73	76
0.125	6	87	99	99	88	95	97	98	86	94	96	97
0.125	12	88	100	100	88	93	96	98	88	93	96	98
0.125	24	86	100	100	89	95	97	98	89	95	97	98

Table 9.73: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	73	93	93	76	87	89	92	72	82	84	92
1.000	6	80	100	100	87	91	93	96	87	91	93	96
1.000	12	85	100	100	90	96	97	98	90	96	97	98
1.000	24	85	100	100	87	92	94	97	87	92	94	97
0.500	3	75	93	94	76	84	87	90	72	80	83	88
0.500	6	87	100	100	90	93	95	98	90	93	95	98
0.500	12	83	100	100	93	97	98	100	93	97	98	100
0.500	24	87	100	100	87	94	96	100	87	94	96	100
0.250	3	76	94	96	73	79	81	85	69	75	77	82
0.250	6	86	99	100	89	94	95	98	89	94	95	98
0.250	12	87	100	100	90	94	95	97	90	94	95	97
0.250	24	91	100	100	87	92	96	99	86	92	96	99
0.125	3	71	88	93	65	73	78	83	61	68	73	78
0.125	6	83	98	99	80	89	92	94	79	88	91	93
0.125	12	89	99	100	91	98	98	99	91	98	98	99
0.125	24	83	100	100	87	94	96	98	86	94	96	98

Table 9.74: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

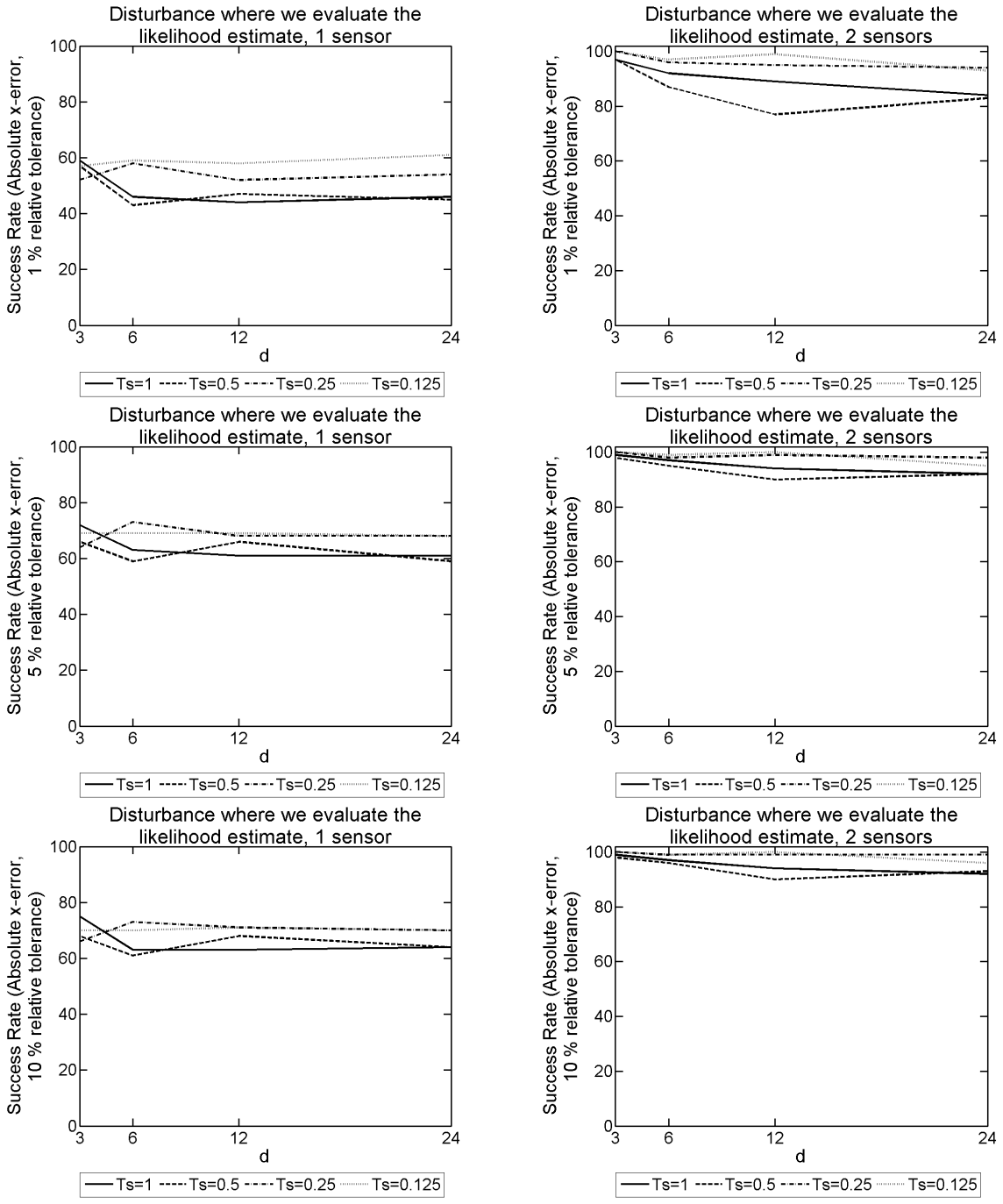


Figure 9.35: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

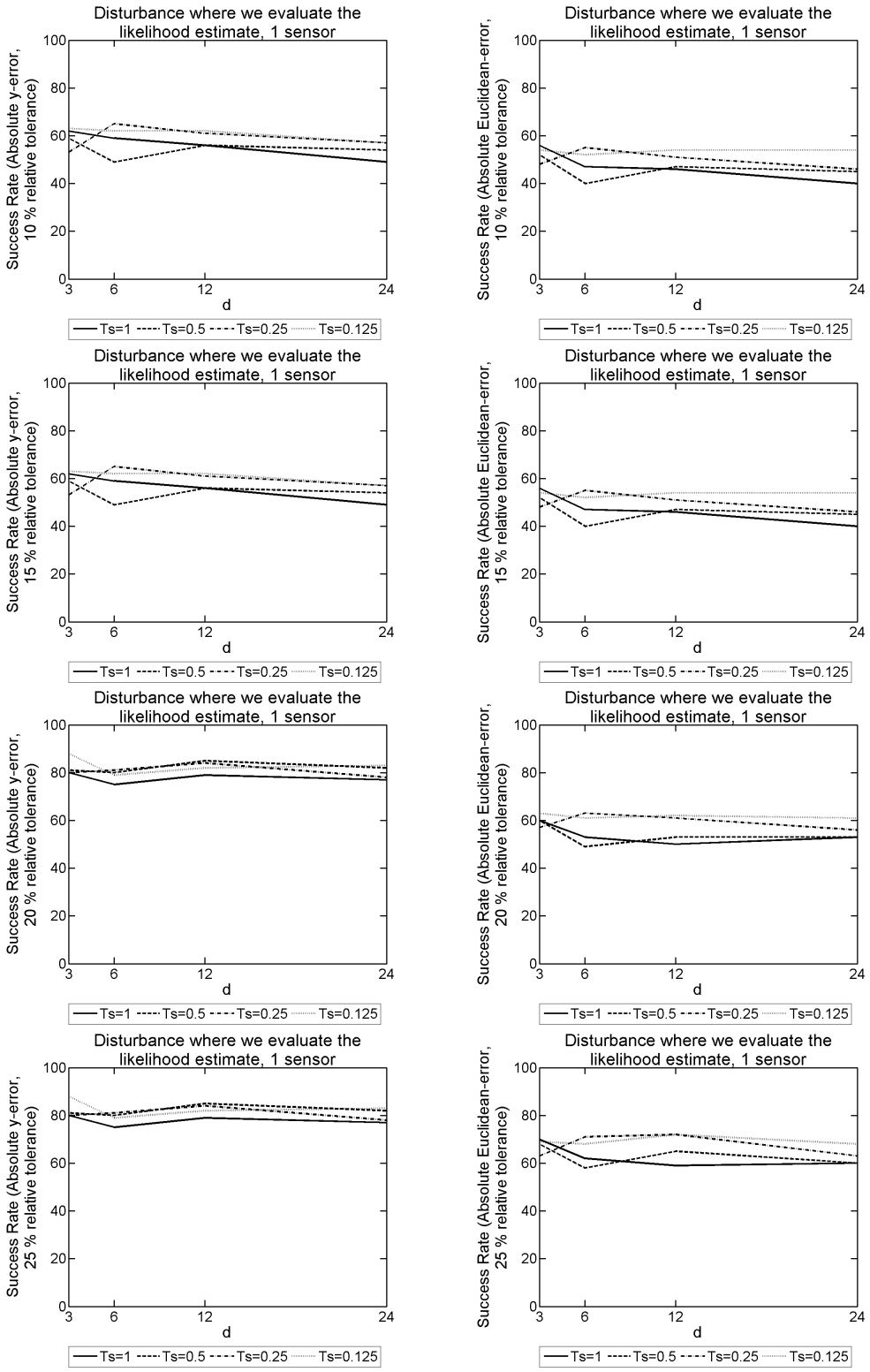


Figure 9.36: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 1 sensor present.

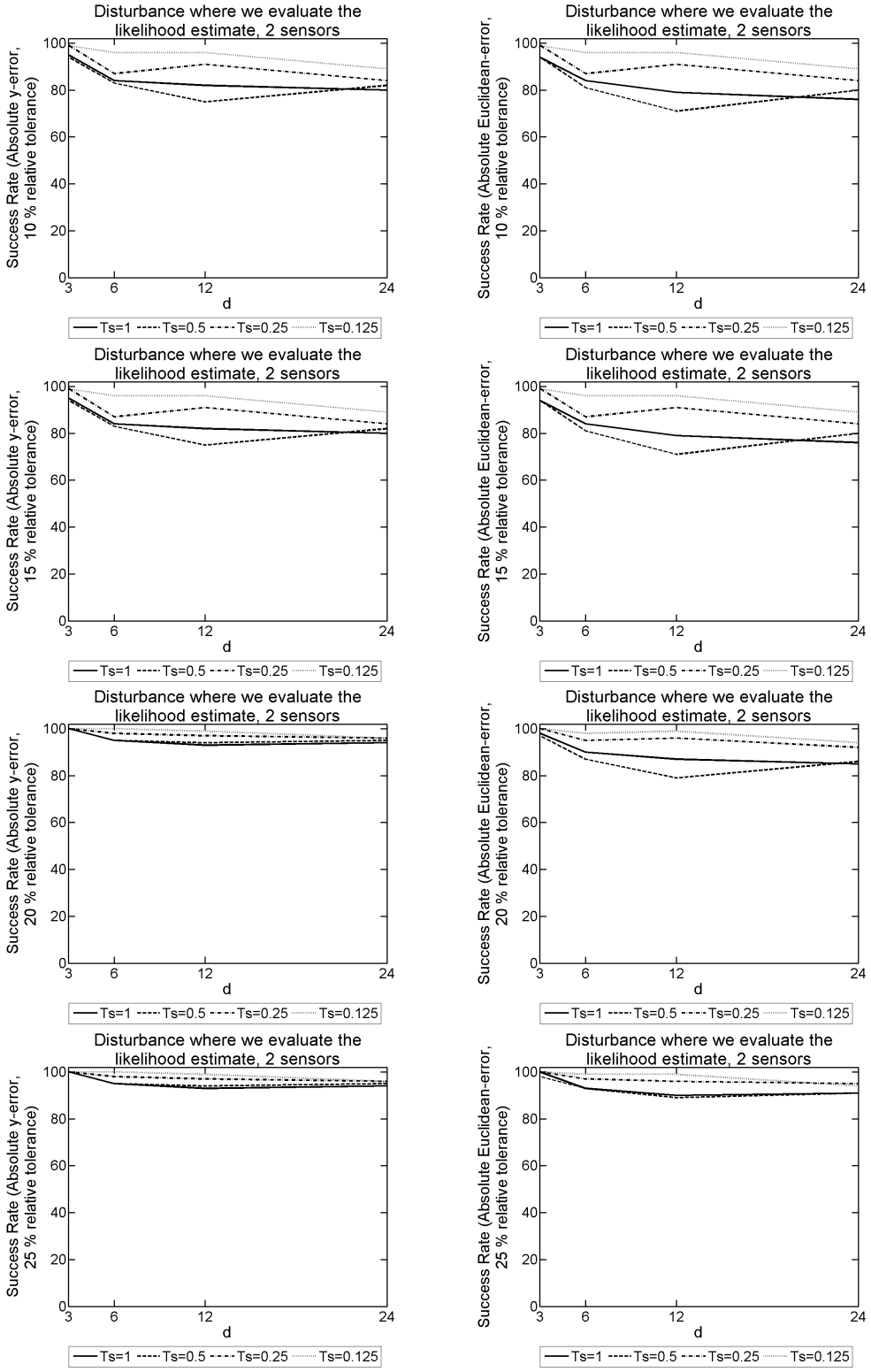


Figure 9.37: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 2 sensors present.

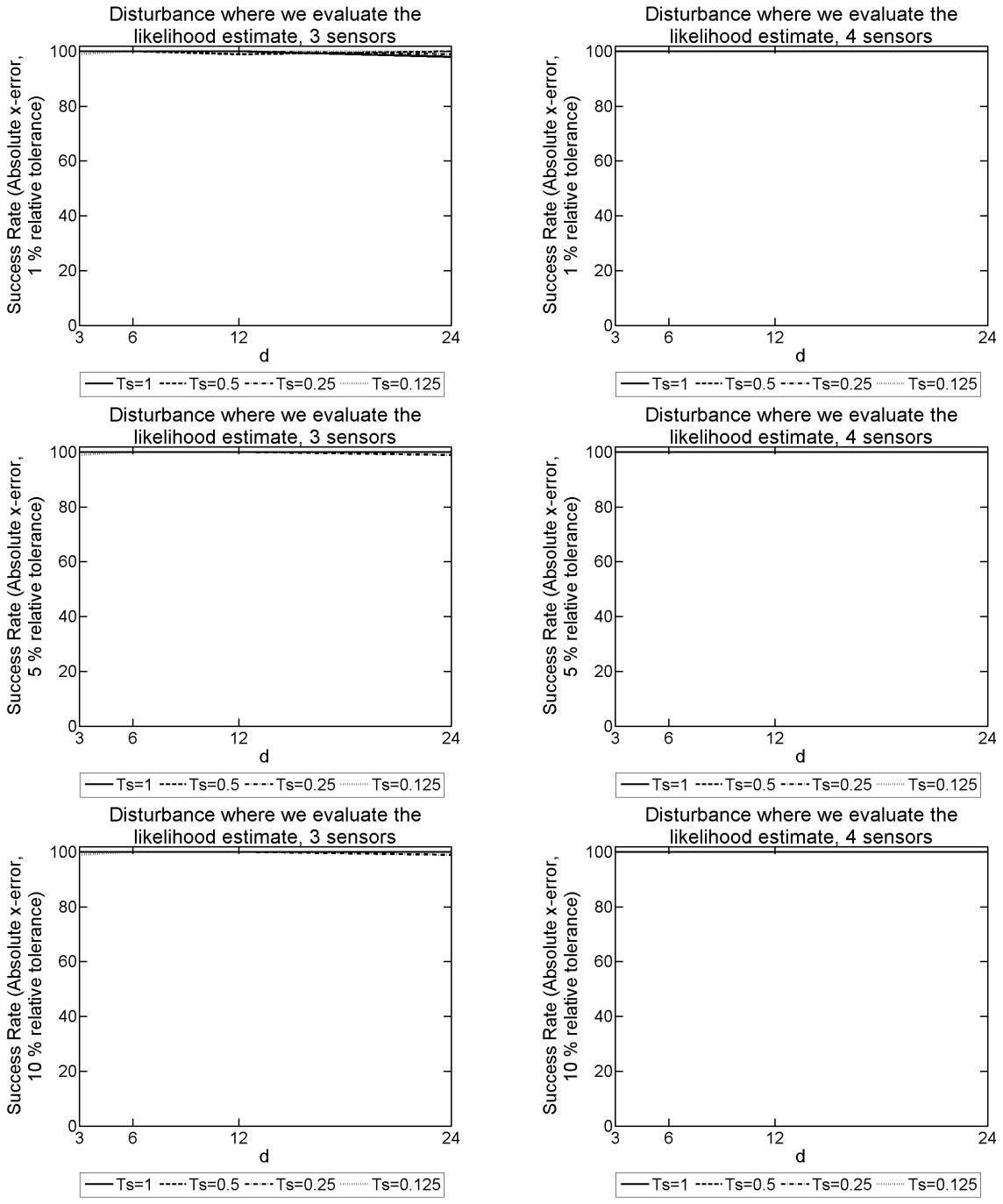


Figure 9.38: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

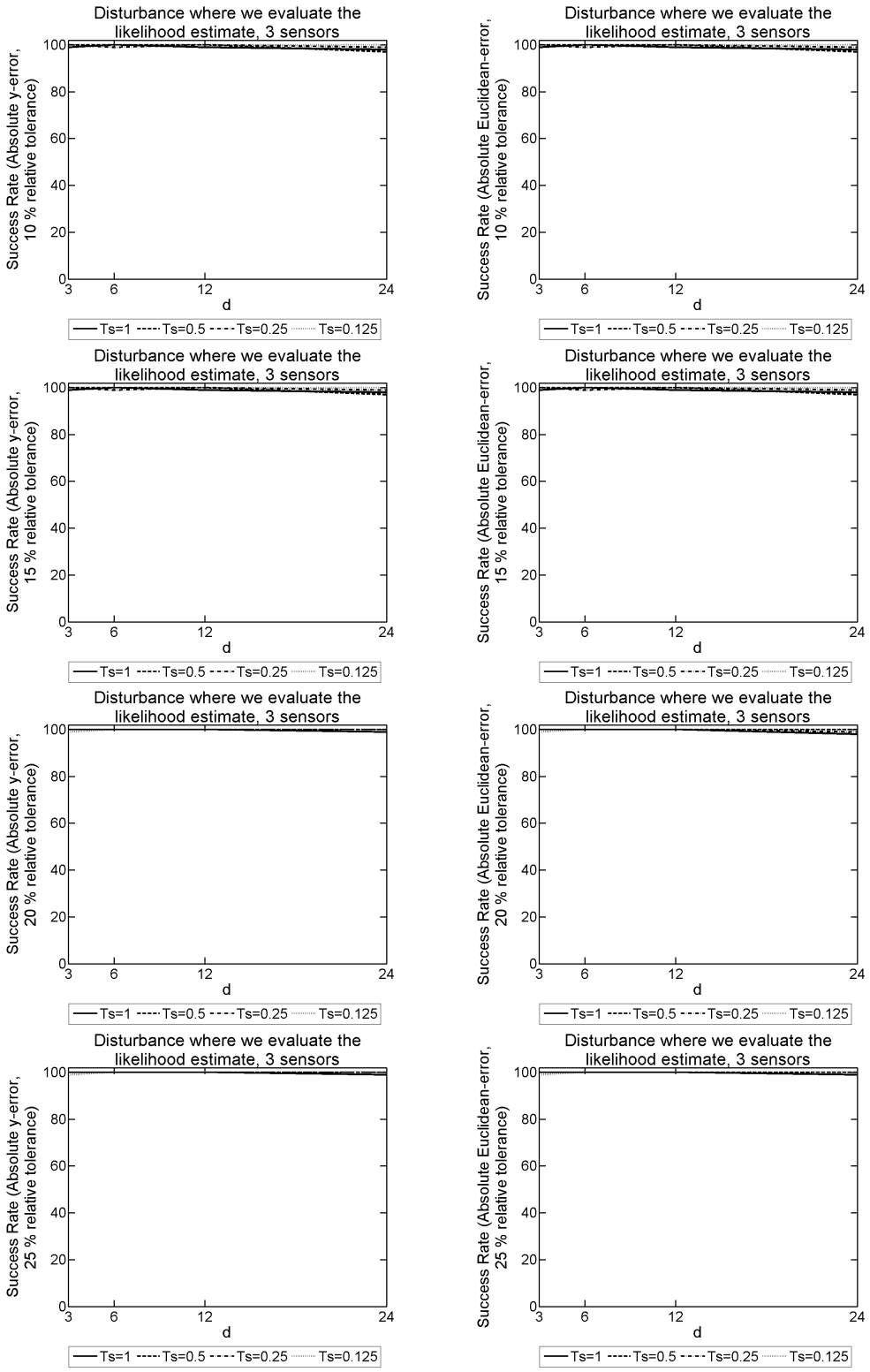


Figure 9.39: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 3 sensors present.

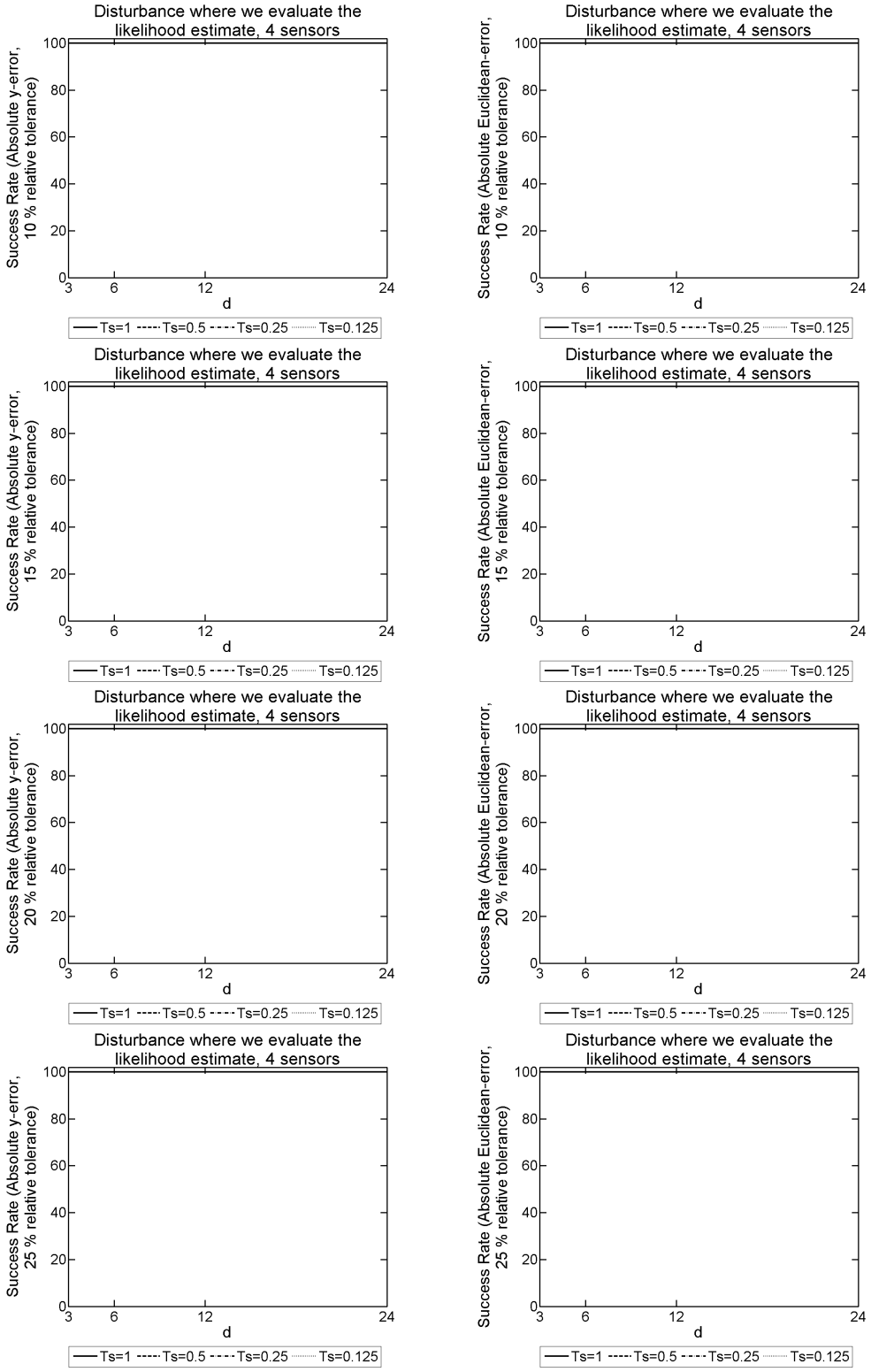


Figure 9.40: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 4 sensors present.

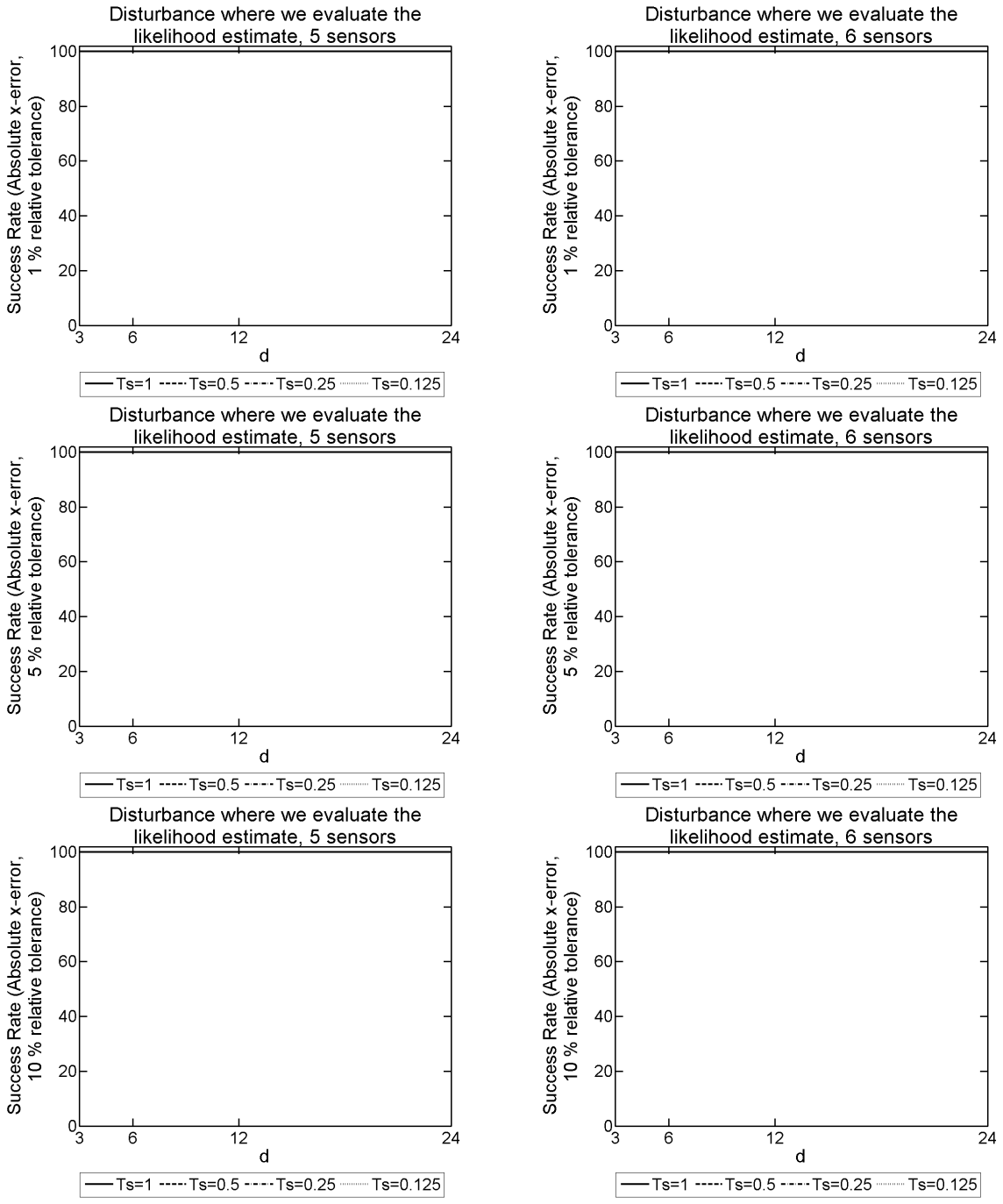


Figure 9.41: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

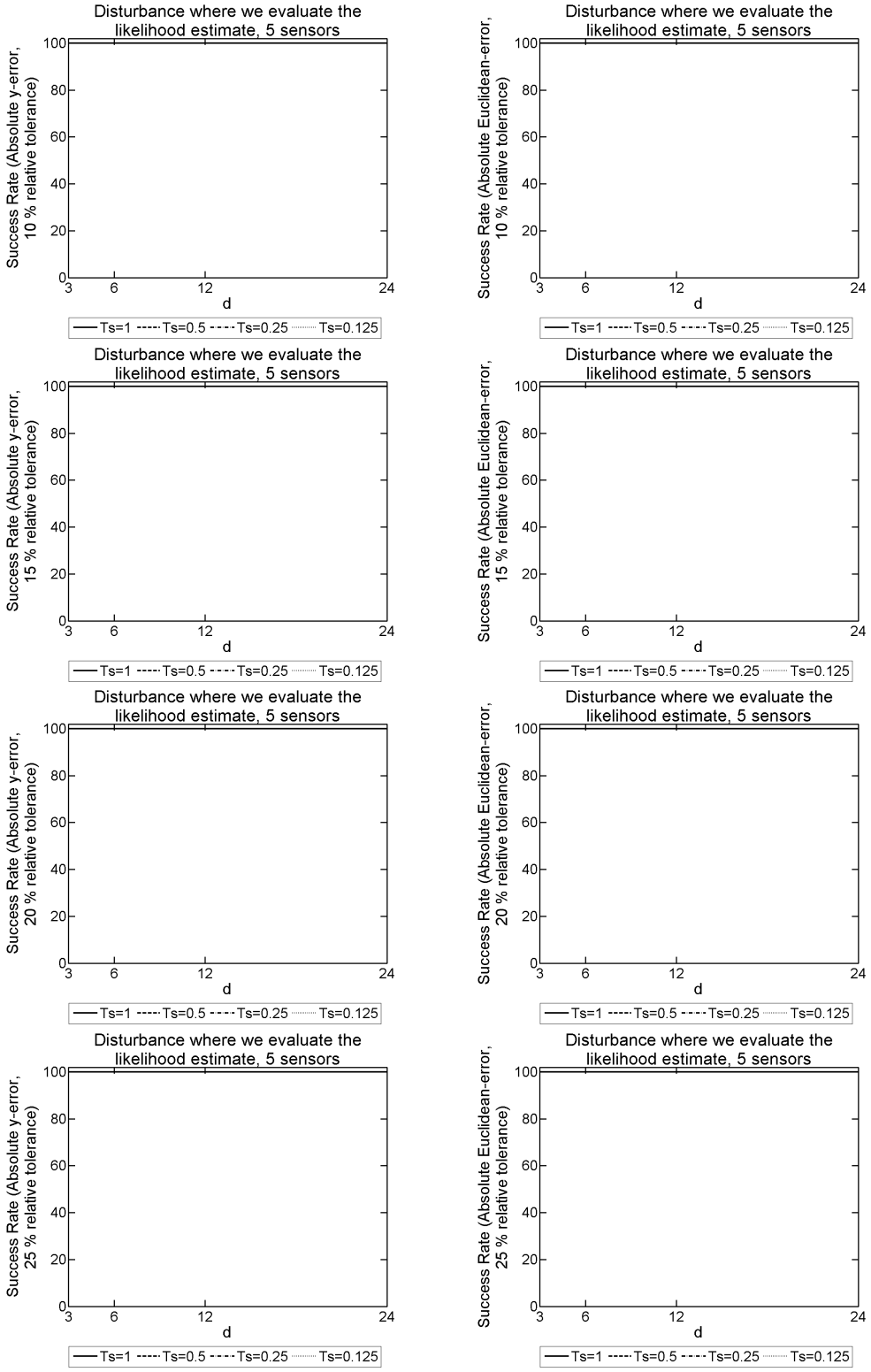


Figure 9.42: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 5 sensors present.

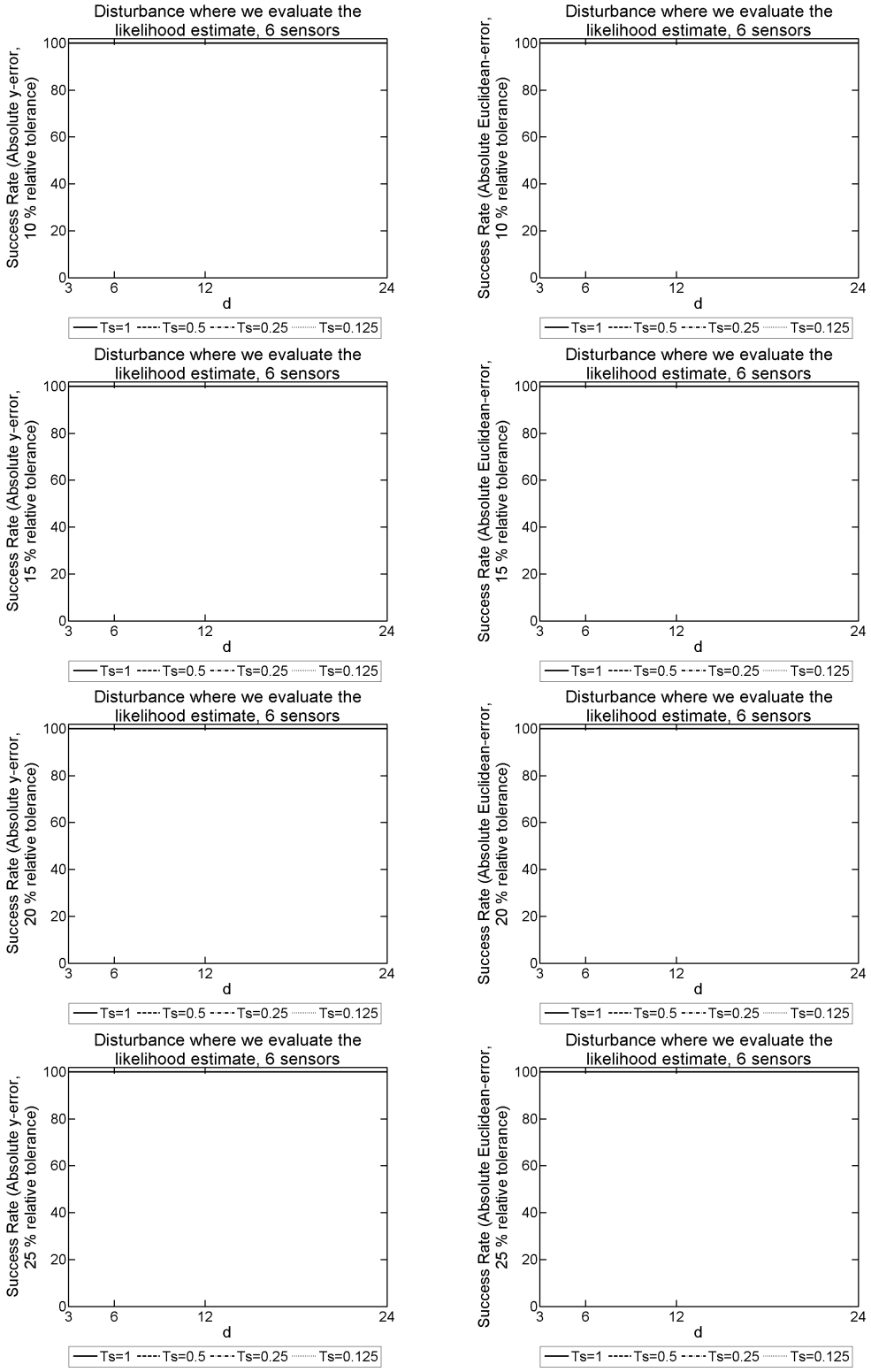


Figure 9.43: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 6 sensors present.

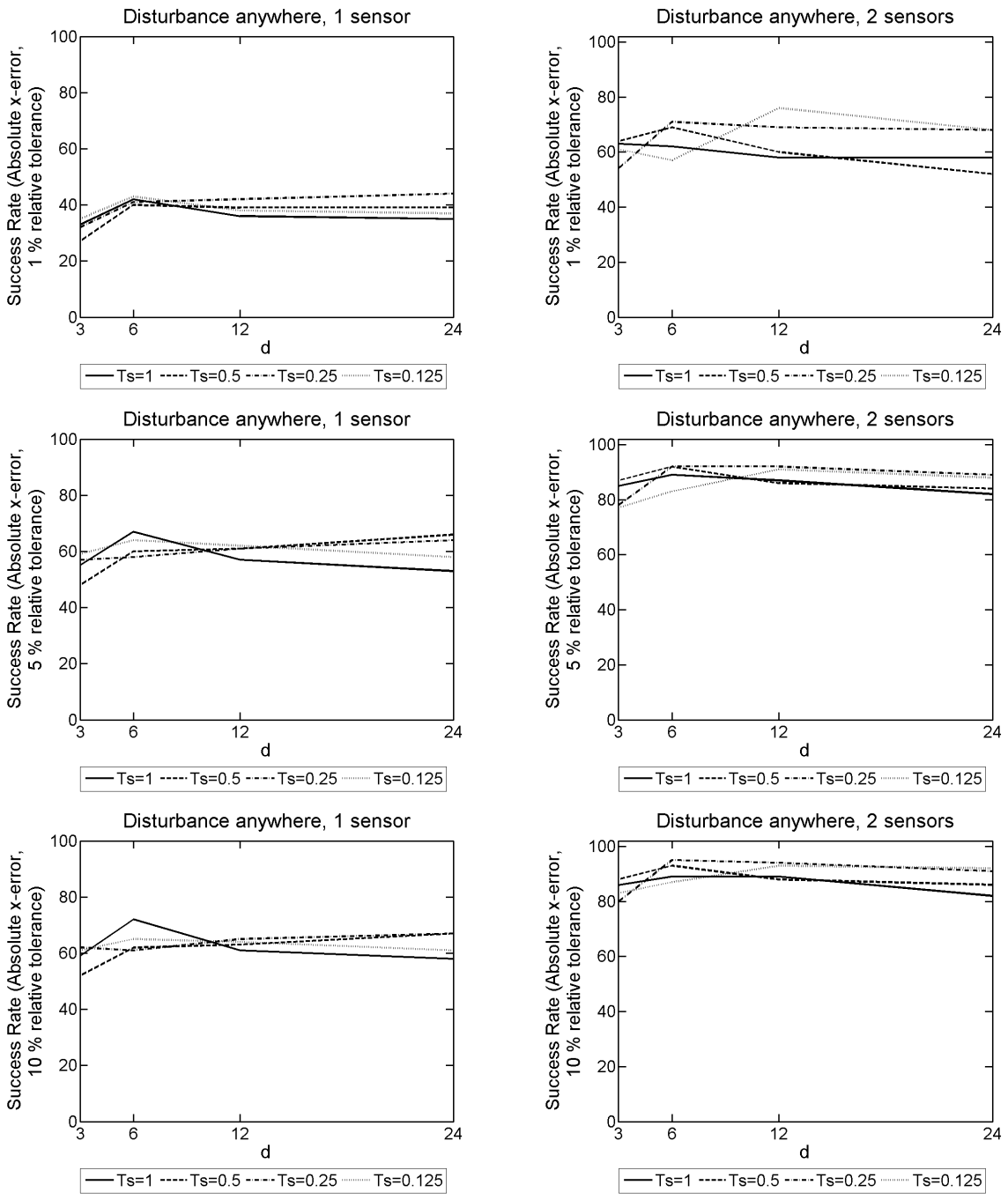


Figure 9.44: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

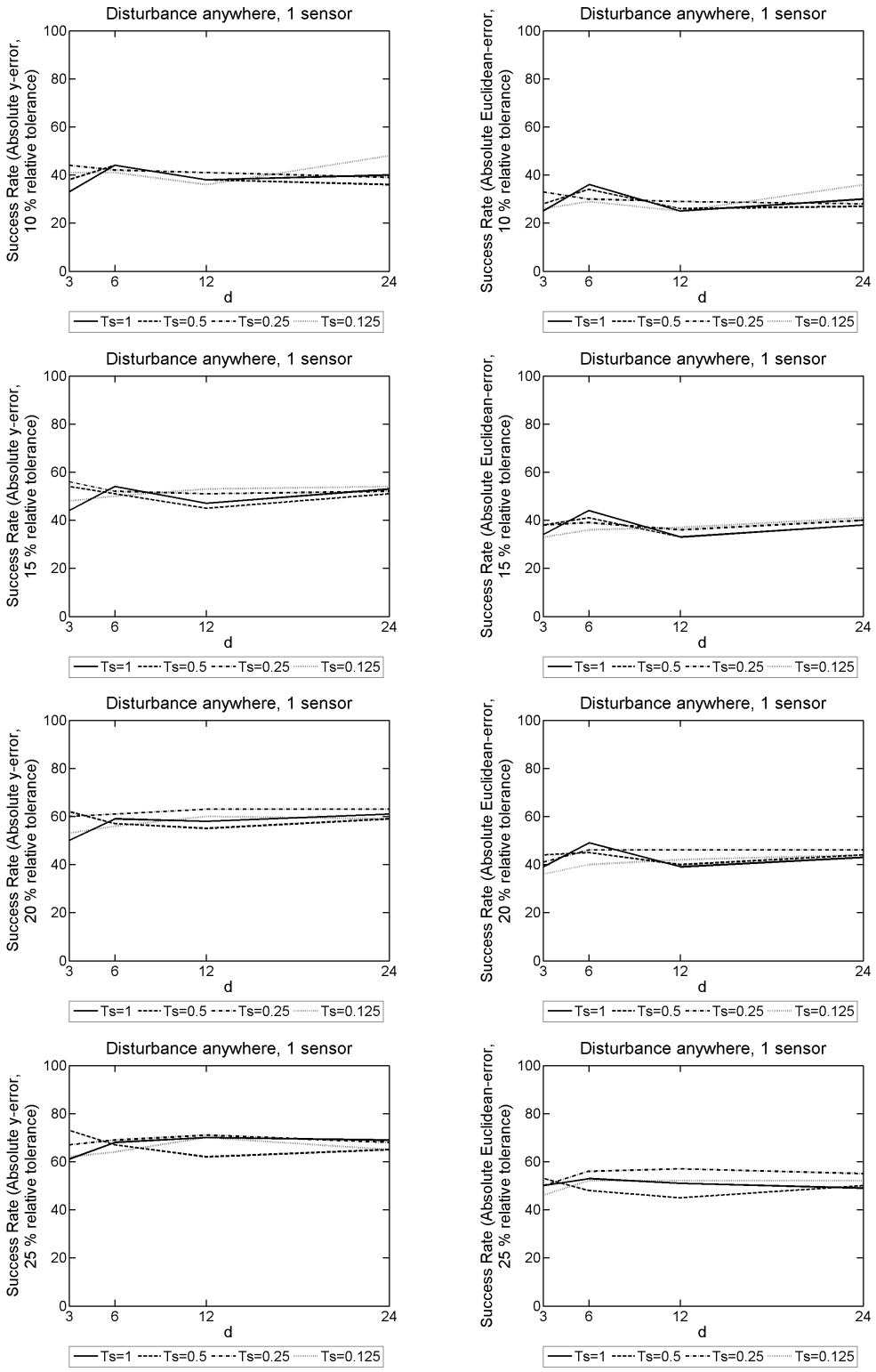


Figure 9.45: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 1 sensor present.

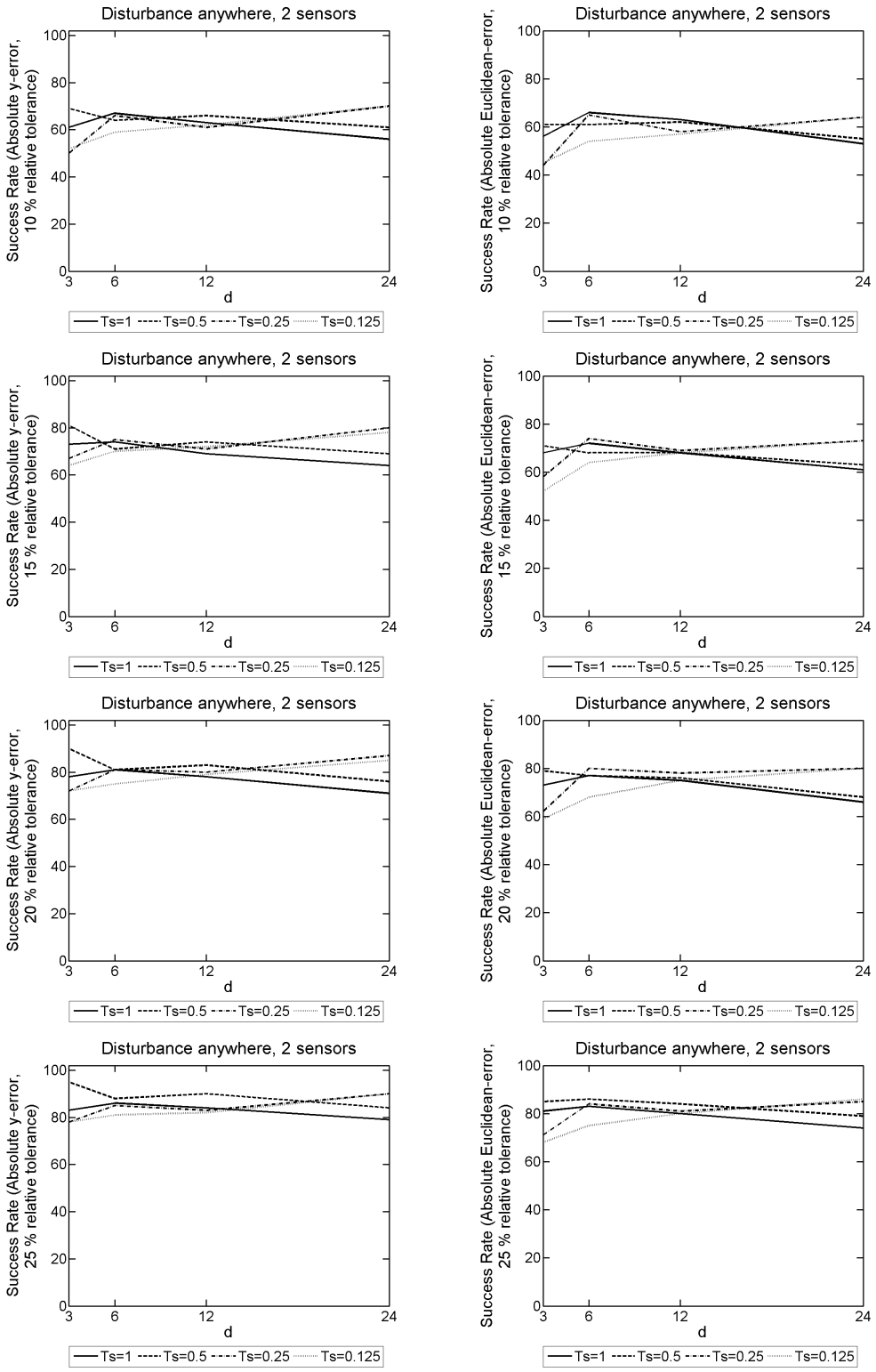


Figure 9.46: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 2 sensors present.

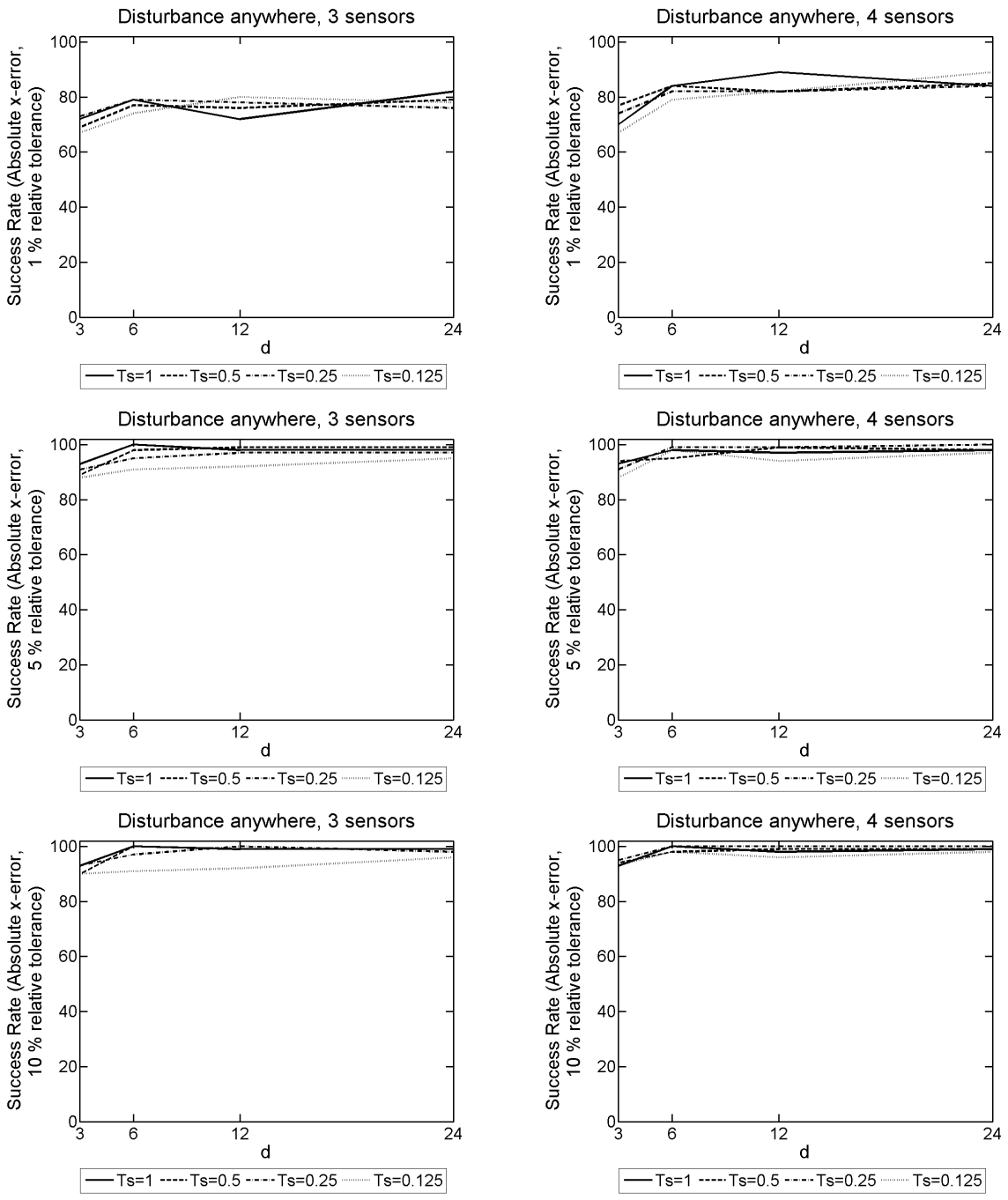


Figure 9.47: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

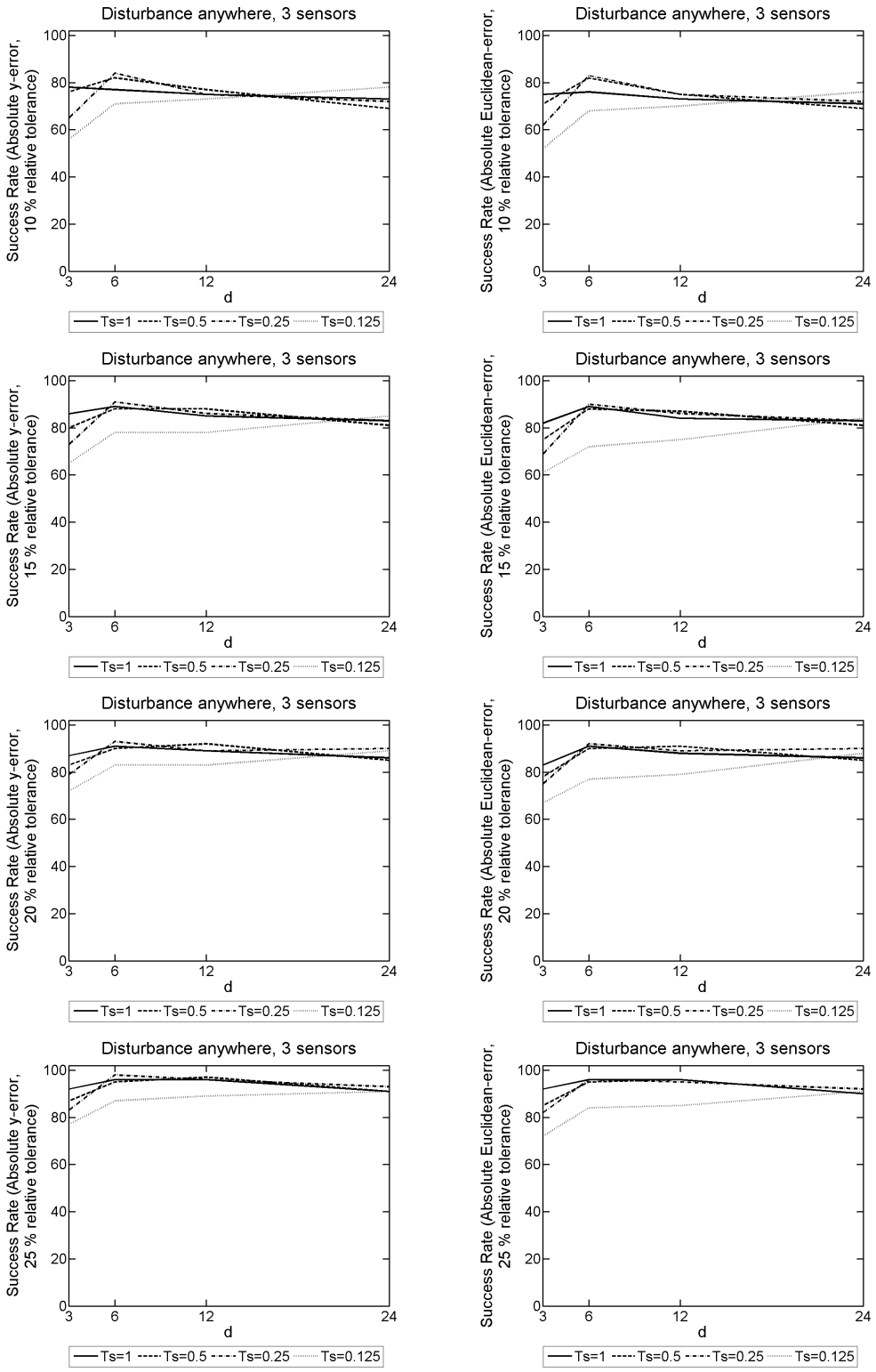


Figure 9.48: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 3 sensors present.

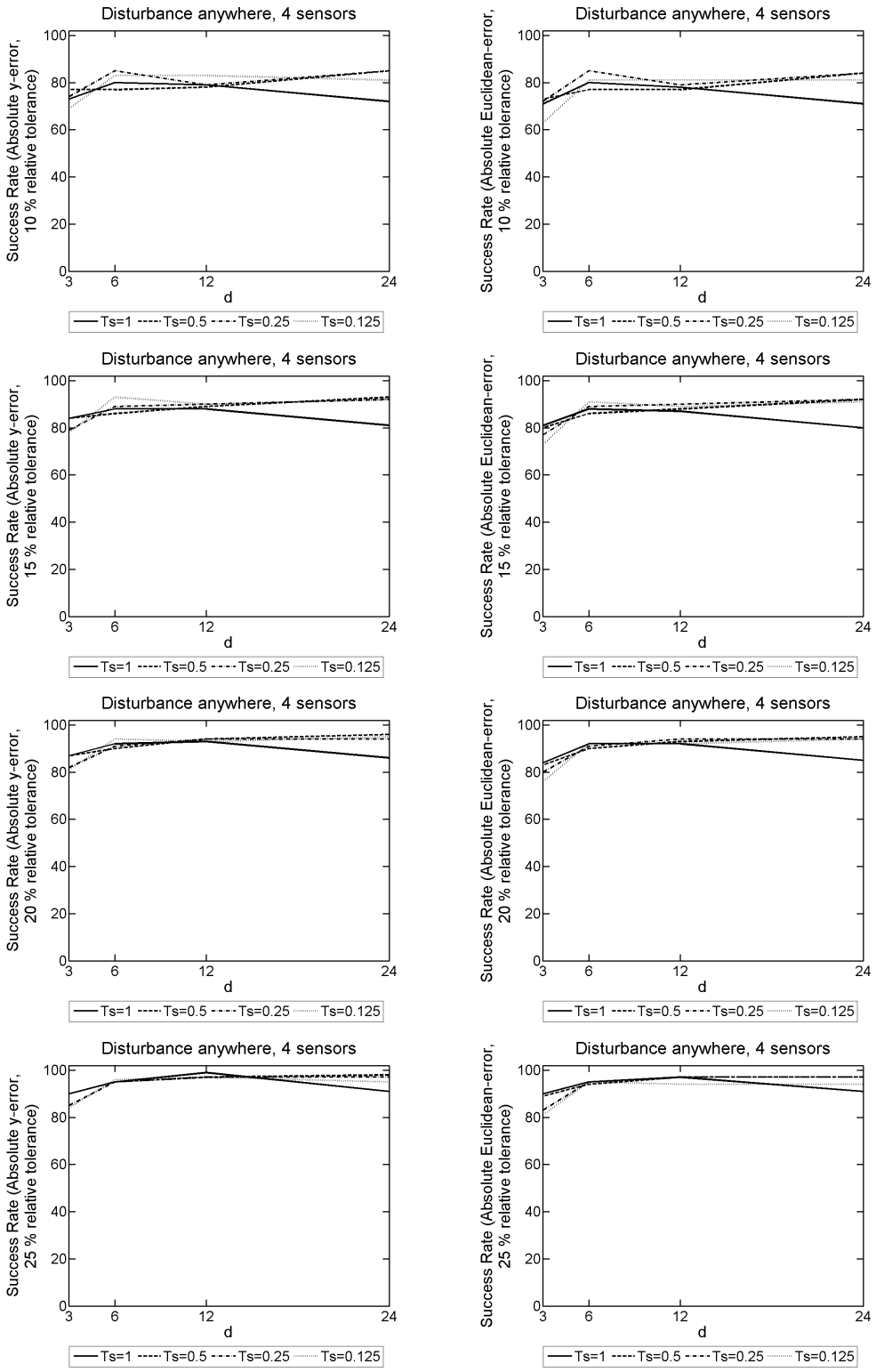


Figure 9.49: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 4 sensors present.

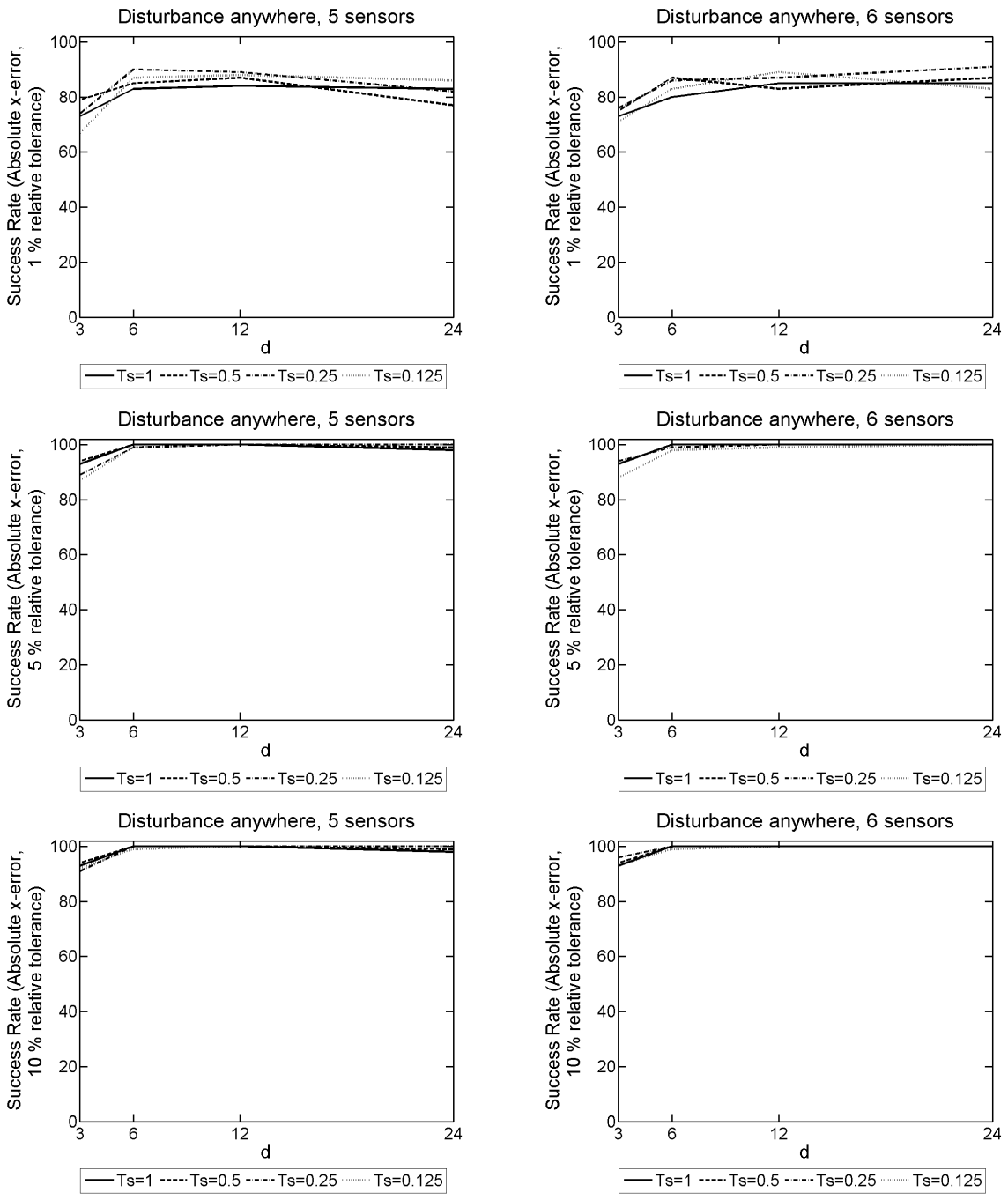


Figure 9.50: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

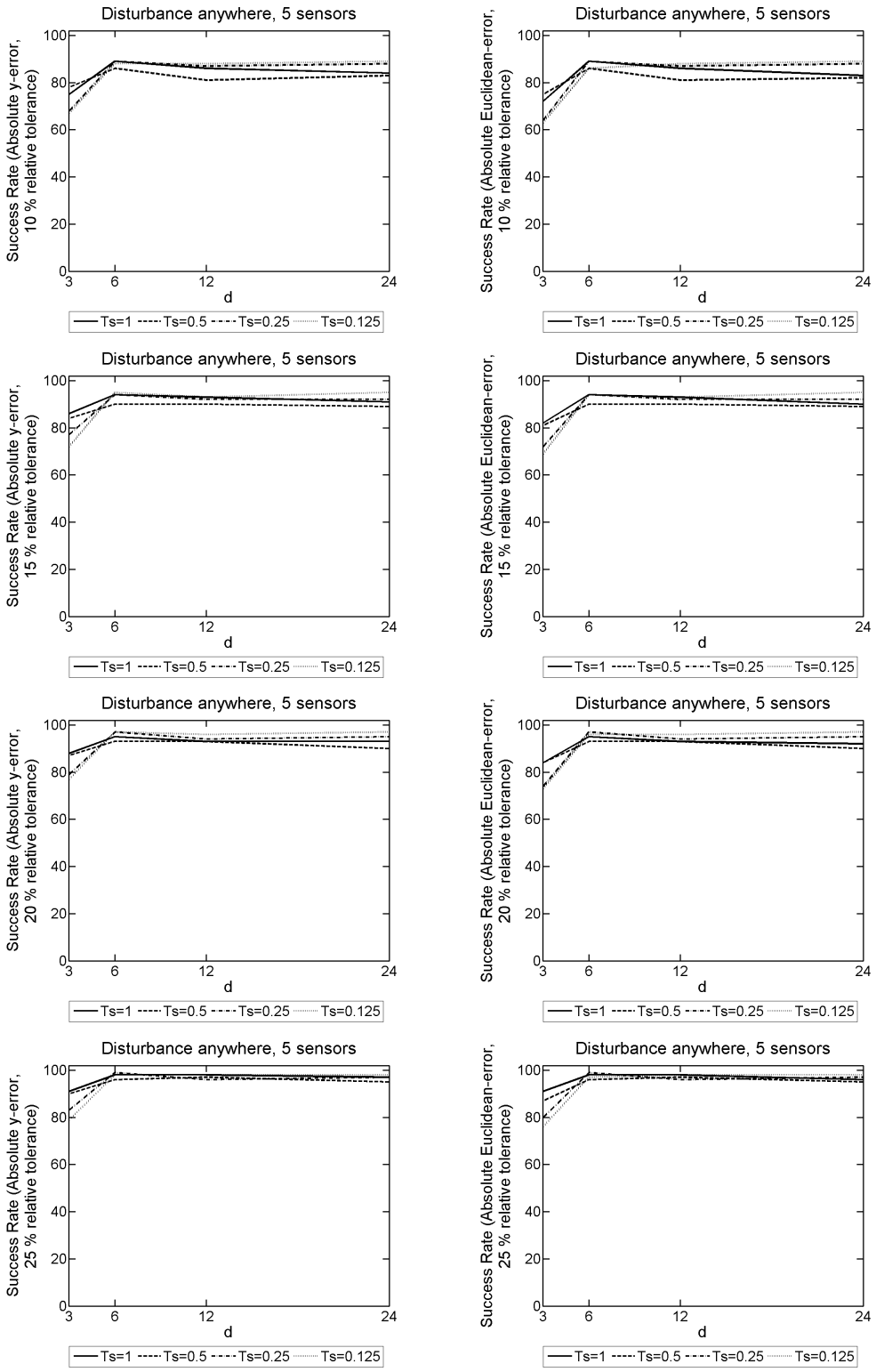


Figure 9.51: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 5 sensors present.

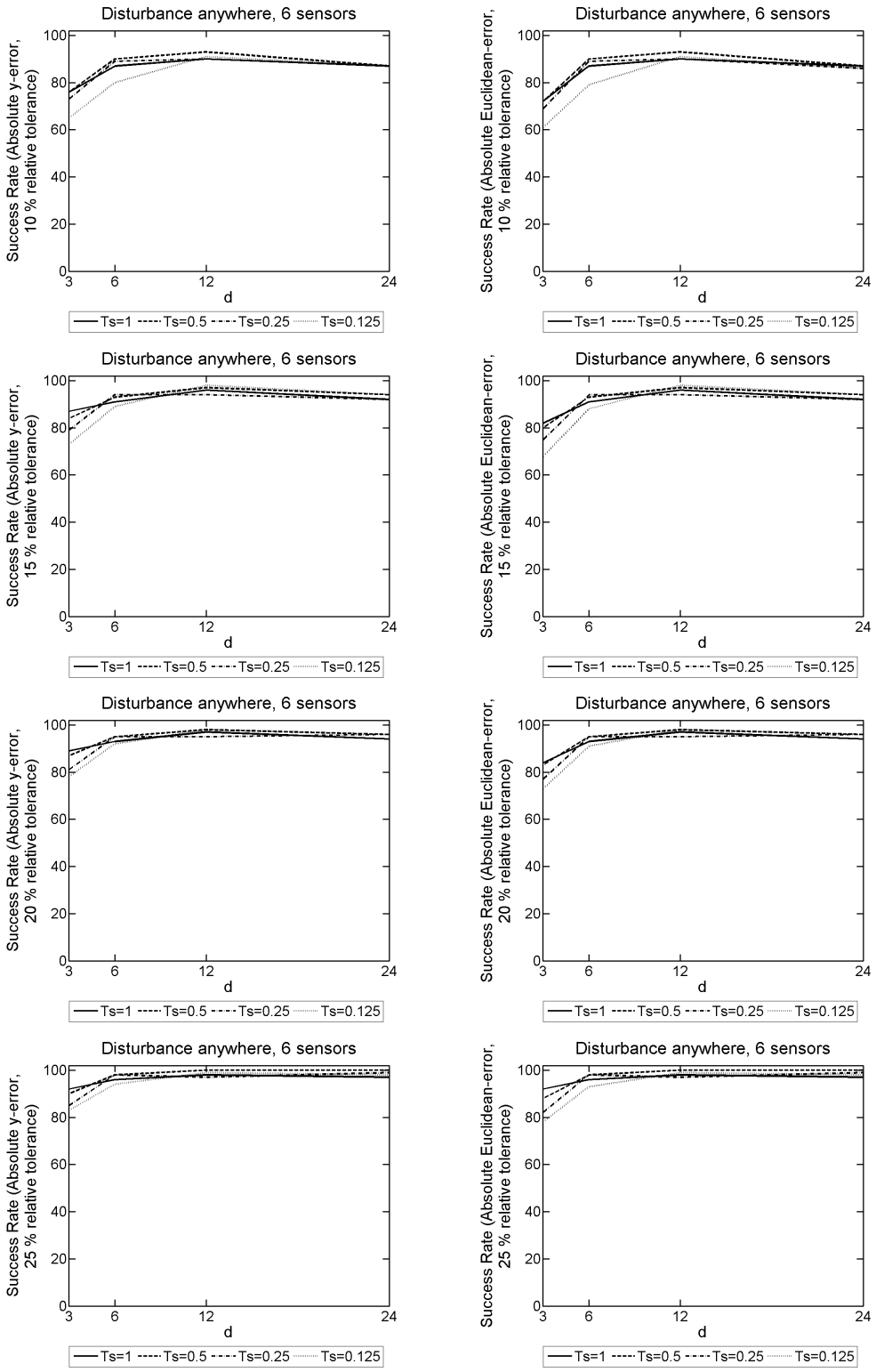


Figure 9.52: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 6 sensors present.

C.4 Use of an SVD to reduce matrix dimensions in the KF, 3NBCs model problem

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	87	91	91	84	84	95	95	80	80	87	91
2	99	100	100	96	96	99	99	96	96	99	99
3	100	100	100	99	99	100	100	99	99	100	100
4	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100

Table 9.75: The success rate of our 2D model problem with 3NBCs. Here we have a mesh dimension of $N = 50$, $M = 5$ and $L = 9000$ over a simulation duration of $T = 3$ seconds, using 100 different disturbance locations positioned where the likelihood function in (2.107) is evaluated. The KF is then run over the final second in an attempt to mimic the scenarios considered when using an SVD.

Number of Sensors	Success Rate										
	|x-error|			|y-error|				|Euclidean-error|		
	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1	55	77	77	38	45	50	55	33	38	43	51
2	67	88	88	40	51	57	62	37	48	54	61
3	78	96	96	58	71	76	79	55	68	73	78
4	74	98	98	57	69	73	79	57	68	72	77
5	76	97	97	55	68	73	77	54	67	72	77
6	82	99	99	65	79	85	87	65	79	85	87

Table 9.76: The success rate of our 2D model problem with 3NBCs. Here we have a mesh dimension of $N = 50$, $M = 5$ and $L = 9000$ over a simulation duration of $T = 3$ seconds, using 100 random disturbance locations. The KF is then run over the final second in an attempt to mimic the scenarios considered when using an SVD.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	23	53	61	39	39	66	66	27	27	36	54
1.000	6	36	53	55	46	46	70	70	29	29	39	51
1.000	12	40	53	54	41	41	77	77	27	27	37	50
1.000	24	34	58	61	50	50	76	76	38	38	49	58
0.500	3	40	49	50	46	46	68	68	34	35	46	51
0.500	6	32	47	49	42	42	70	70	28	28	31	44
0.500	12	30	46	46	38	38	72	72	24	24	39	48
0.500	24	35	52	55	47	47	71	71	30	31	41	52
0.250	3	34	52	57	44	44	75	75	34	34	39	50
0.250	6	30	48	52	38	38	73	73	24	24	30	49
0.250	12	34	55	60	36	36	70	70	29	30	41	50
0.250	24	36	55	56	47	47	80	80	32	32	45	57
0.125	3	43	60	62	40	40	73	73	33	34	46	59
0.125	6	39	57	62	37	37	71	71	28	28	39	50
0.125	12	35	55	58	48	48	78	78	36	36	43	59
0.125	24	44	61	64	39	39	74	74	32	32	47	59

Table 9.77: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	87	93	94	81	81	93	93	78	78	90	90
1.000	6	83	92	92	78	78	89	89	77	77	85	86
1.000	12	81	86	87	71	71	86	86	67	68	79	84
1.000	24	80	87	88	80	80	93	93	77	77	80	85
0.500	3	82	87	87	77	77	92	92	73	73	84	87
0.500	6	76	88	89	81	81	90	90	76	76	82	86
0.500	12	80	91	93	72	72	91	91	70	71	83	89
0.500	24	85	92	93	82	82	93	93	80	80	86	90
0.250	3	95	97	98	97	97	100	100	96	96	97	99
0.250	6	87	94	94	85	85	93	93	85	85	89	91
0.250	12	82	86	86	80	80	95	95	75	75	86	90
0.250	24	88	92	92	87	87	94	94	85	85	88	91
0.125	3	91	92	92	93	93	97	97	90	90	91	93
0.125	6	87	89	90	89	89	95	95	84	84	87	89
0.125	12	86	91	91	90	90	98	98	85	85	88	95
0.125	24	91	96	97	87	87	95	95	86	86	92	94

Table 9.78: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	98	98	100	100	98	98	100	100
1.000	6	96	97	97	97	97	98	98	96	96	97	98
1.000	12	94	99	99	90	90	96	96	90	90	94	95
1.000	24	97	100	100	94	94	99	99	94	94	97	99
0.500	3	98	98	98	99	99	100	100	98	98	98	99
0.500	6	96	99	99	99	99	100	100	98	98	99	99
0.500	12	97	100	100	96	96	99	99	96	96	98	99
0.500	24	98	98	98	97	97	100	100	97	97	98	98
0.250	3	92	95	96	91	91	97	97	90	90	92	95
0.250	6	96	98	98	96	96	99	99	96	96	97	98
0.250	12	96	99	99	95	95	98	98	95	95	96	97
0.250	24	99	100	100	98	98	100	100	98	98	100	100
0.125	3	94	96	97	97	97	97	97	95	95	95	95
0.125	6	99	99	99	100	100	100	100	99	99	99	100
0.125	12	96	97	97	97	97	99	99	96	96	97	98
0.125	24	100	100	100	97	97	100	100	97	97	100	100

Table 9.79: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	99	100	100	99	99	100	100	99	99	99	100
1.000	12	100	100	100	97	97	100	100	97	97	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	99	99	100	100	99	99	100	100
0.500	12	99	100	100	98	98	100	100	98	98	99	100
0.500	24	100	100	100	98	98	100	100	98	98	100	100
0.250	3	99	100	100	99	99	99	99	99	99	99	99
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	99	99	99	100	100	100	100	99	99	100	100
0.125	6	100	100	100	99	99	100	100	99	99	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.80: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	99	99	99	100	100	100	100	99	99	99	100
0.125	6	99	100	100	99	99	99	99	99	99	99	99
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.81: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	98	100	100	98	98	99	99	98	98	99	99
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	98	99	99	97	97	100	100	97	97	99	99
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	93	98	98	98	98	99	99	97	97	97	98
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	86	93	93	86	86	93	93	83	83	88	90
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.82: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	24	58	60	33	40	46	49	21	27	35	44
1.000	6	26	57	59	30	44	51	59	21	30	39	46
1.000	12	28	49	51	33	45	53	58	19	28	36	44
1.000	24	28	54	57	30	43	52	57	21	28	35	45
0.500	3	32	58	61	33	43	52	59	24	31	37	47
0.500	6	29	49	49	39	46	56	62	22	29	38	49
0.500	12	20	43	43	36	44	55	61	17	22	27	41
0.500	24	19	45	48	26	35	45	54	12	17	26	37
0.250	3	23	50	53	38	51	57	60	21	30	35	44
0.250	6	27	44	47	38	50	58	59	20	25	32	41
0.250	12	27	46	50	34	52	62	67	20	26	34	42
0.250	24	25	45	48	31	45	57	62	15	22	31	43
0.125	3	39	60	64	33	49	59	63	20	34	41	51
0.125	6	25	48	51	32	37	48	52	18	22	27	33
0.125	12	25	55	60	31	42	56	60	21	30	41	46
0.125	24	31	52	55	37	50	59	64	24	32	42	50

Table 9.83: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	48	78	78	48	62	70	75	43	55	63	68
1.000	6	58	82	82	54	63	68	74	48	55	60	68
1.000	12	58	79	80	53	68	74	79	46	58	63	72
1.000	24	56	82	85	47	63	71	77	41	56	63	71
0.500	3	52	79	79	42	65	74	79	36	55	62	68
0.500	6	54	76	78	38	53	63	69	35	46	56	60
0.500	12	60	83	84	50	66	75	79	46	59	66	72
0.500	24	65	86	87	46	57	67	74	44	52	63	70
0.250	3	49	79	83	44	59	67	68	37	52	59	61
0.250	6	63	84	87	50	62	68	73	46	57	64	72
0.250	12	56	85	86	46	59	72	77	43	55	67	71
0.250	24	71	90	91	55	71	75	82	51	67	71	79
0.125	3	44	68	71	48	60	63	67	37	47	50	56
0.125	6	44	74	75	50	64	71	71	43	51	57	60
0.125	12	53	79	81	43	60	70	73	36	51	61	67
0.125	24	60	82	84	52	69	73	77	43	58	62	67

Table 9.84: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	67	93	93	68	88	93	93	64	82	87	92
1.000	6	72	92	92	62	83	86	89	60	81	85	89
1.000	12	76	97	97	60	76	84	88	59	74	83	87
1.000	24	78	98	100	59	80	88	93	59	80	88	93
0.500	3	65	93	93	61	82	85	87	59	78	82	85
0.500	6	77	96	97	69	85	87	90	68	84	86	89
0.500	12	67	91	93	61	77	87	89	57	71	81	84
0.500	24	73	95	95	60	75	85	89	60	74	85	88
0.250	3	66	94	95	64	79	82	83	61	75	79	81
0.250	6	71	96	97	67	79	84	86	65	77	82	83
0.250	12	74	98	98	58	75	85	88	58	75	85	87
0.250	24	78	98	98	67	82	86	91	66	81	85	91
0.125	3	69	92	93	61	74	78	80	59	71	75	78
0.125	6	73	91	93	66	80	84	86	63	76	79	82
0.125	12	80	95	96	71	83	90	90	70	79	87	88
0.125	24	76	98	99	67	81	86	89	67	80	86	89

Table 9.85: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	66	93	93	68	87	93	93	64	81	87	91
1.000	6	79	99	99	77	92	95	95	77	92	95	95
1.000	12	78	100	100	65	84	90	93	65	84	90	93
1.000	24	80	99	99	66	83	91	96	65	82	90	96
0.500	3	62	92	92	59	78	85	86	58	75	82	83
0.500	6	81	98	99	77	91	93	94	77	91	93	94
0.500	12	79	98	98	66	84	90	95	66	82	89	94
0.500	24	77	100	100	73	88	94	97	73	87	94	97
0.250	3	70	93	97	65	78	82	84	65	78	82	83
0.250	6	81	99	99	74	90	93	93	74	89	93	93
0.250	12	79	100	100	74	91	92	93	74	91	92	93
0.250	24	78	100	100	74	89	92	93	74	89	92	93
0.125	3	71	94	94	64	77	82	82	62	75	80	81
0.125	6	82	99	99	75	88	91	92	75	88	91	91
0.125	12	89	99	100	79	91	95	96	79	91	95	96
0.125	24	81	99	99	79	90	92	94	79	90	92	94

Table 9.86: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	67	92	92	67	87	92	92	63	81	86	91
1.000	6	84	99	99	85	96	98	98	85	96	98	98
1.000	12	86	100	100	76	90	94	94	76	90	94	94
1.000	24	84	99	99	77	91	93	97	77	90	93	97
0.500	3	67	95	95	63	83	86	88	62	81	85	87
0.500	6	82	99	99	79	94	96	96	79	93	96	96
0.500	12	82	100	100	77	89	93	95	77	89	93	95
0.500	24	87	99	99	81	94	97	97	80	93	96	96
0.250	3	74	96	96	77	87	88	88	74	84	85	86
0.250	6	85	98	98	83	94	95	96	82	93	94	96
0.250	12	88	100	100	83	92	95	96	83	92	95	96
0.250	24	87	100	100	84	95	96	98	84	95	96	98
0.125	3	75	93	93	63	75	80	82	63	74	78	79
0.125	6	87	100	100	82	93	96	96	82	93	96	96
0.125	12	87	100	100	84	95	96	96	84	95	96	96
0.125	24	85	100	100	86	94	96	97	86	94	96	97

Table 9.87: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	68	94	94	69	90	94	94	65	84	89	94
1.000	6	81	99	99	83	95	96	96	83	95	96	96
1.000	12	84	99	99	76	95	97	98	76	95	97	98
1.000	24	92	100	100	80	92	97	98	80	92	97	98
0.500	3	65	94	94	59	82	86	88	58	79	84	85
0.500	6	83	98	98	80	93	97	97	80	93	96	96
0.500	12	85	100	100	82	95	95	96	82	95	95	96
0.500	24	89	100	100	84	95	99	100	84	95	99	100
0.250	3	64	97	98	65	77	80	81	65	77	79	80
0.250	6	84	97	97	81	93	95	95	81	93	95	95
0.250	12	91	100	100	89	99	100	100	89	99	100	100
0.250	24	88	100	100	83	94	98	98	83	94	98	98
0.125	3	69	93	93	67	77	82	84	63	73	77	80
0.125	6	76	98	98	73	89	92	94	73	87	91	93
0.125	12	89	98	99	87	98	100	100	87	97	99	99
0.125	24	86	100	100	86	96	97	97	86	95	97	97

Table 9.88: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

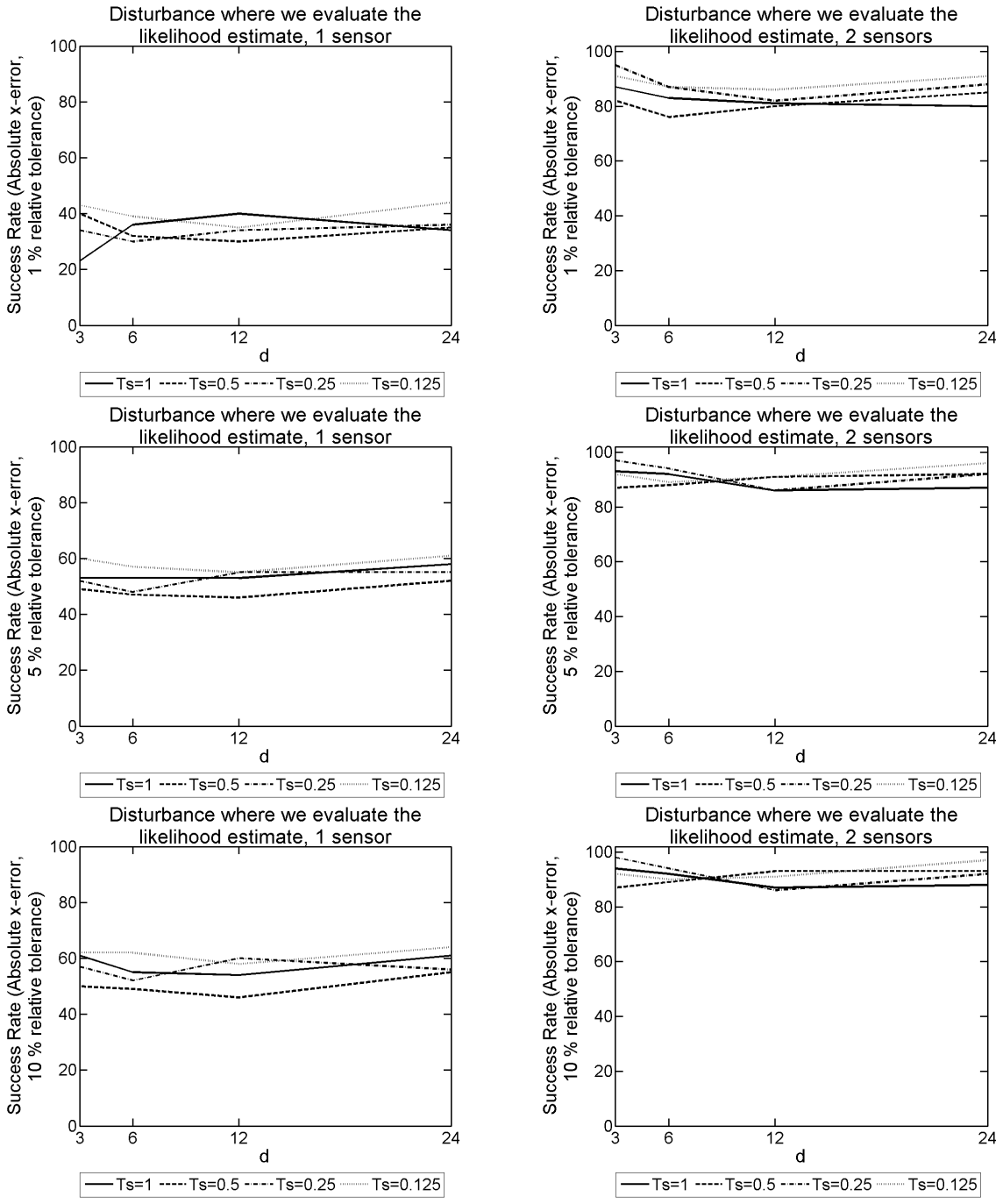


Figure 9.53: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

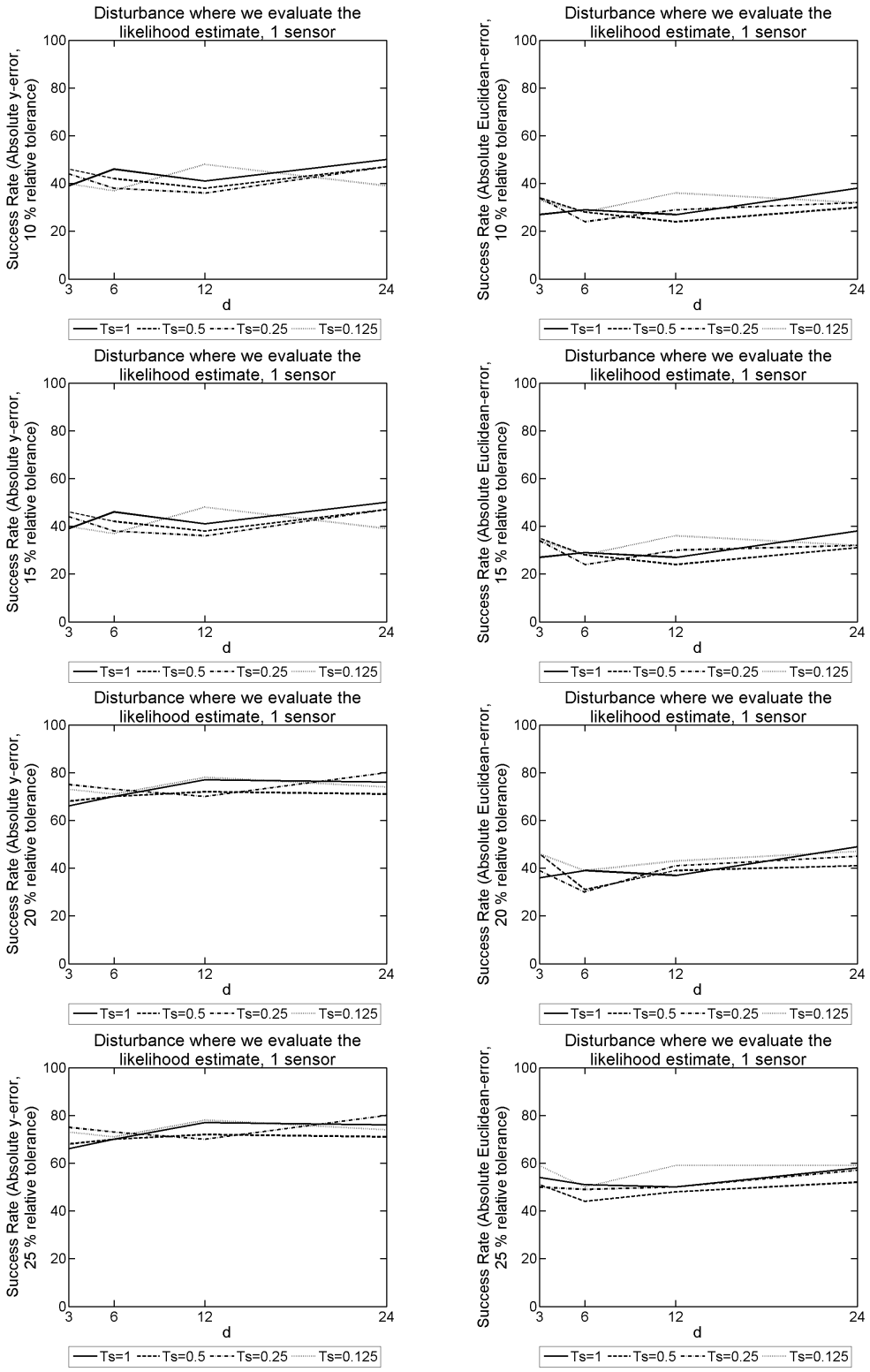


Figure 9.54: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 1 sensor present.

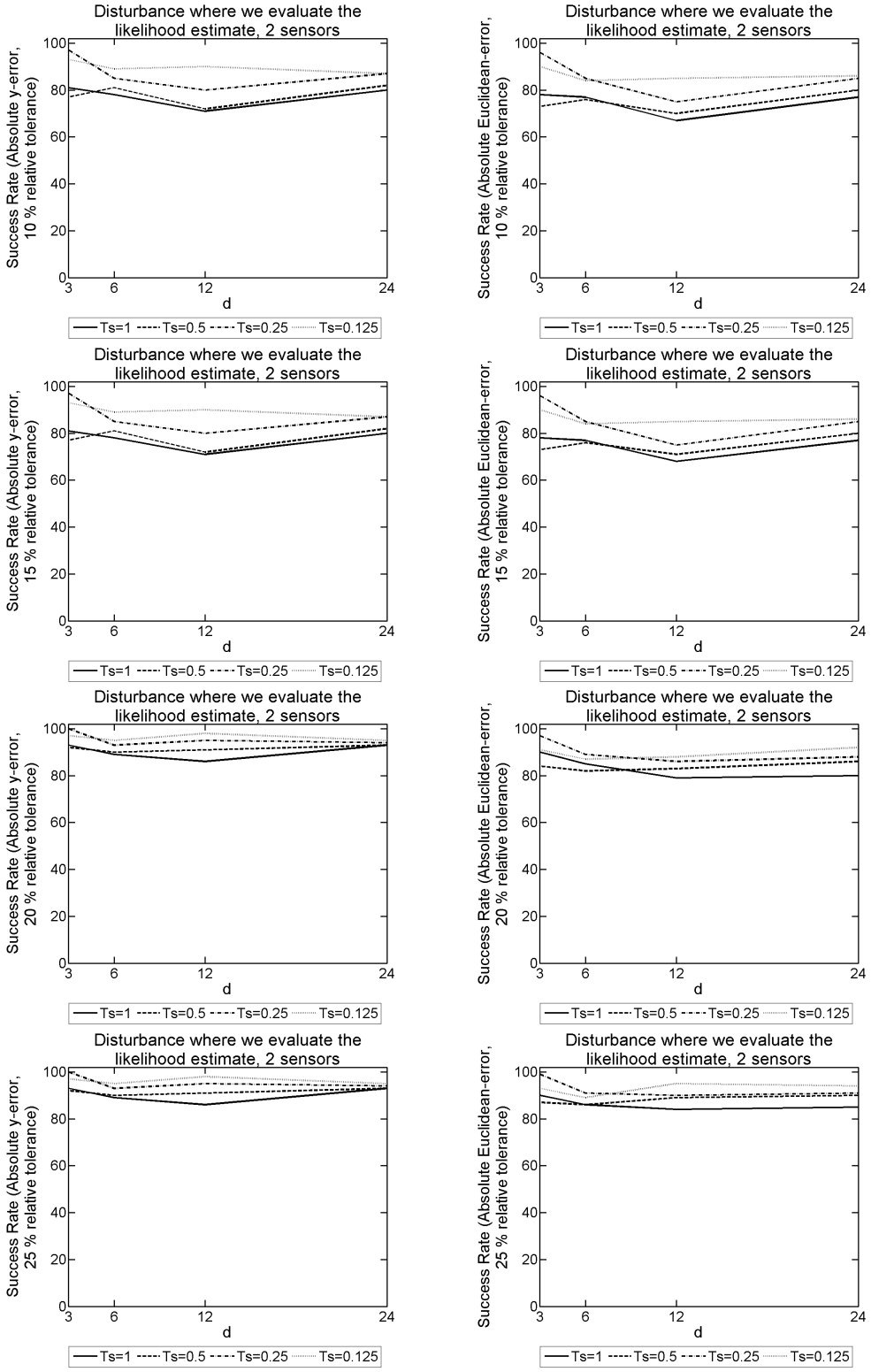


Figure 9.55: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 2 sensors present.

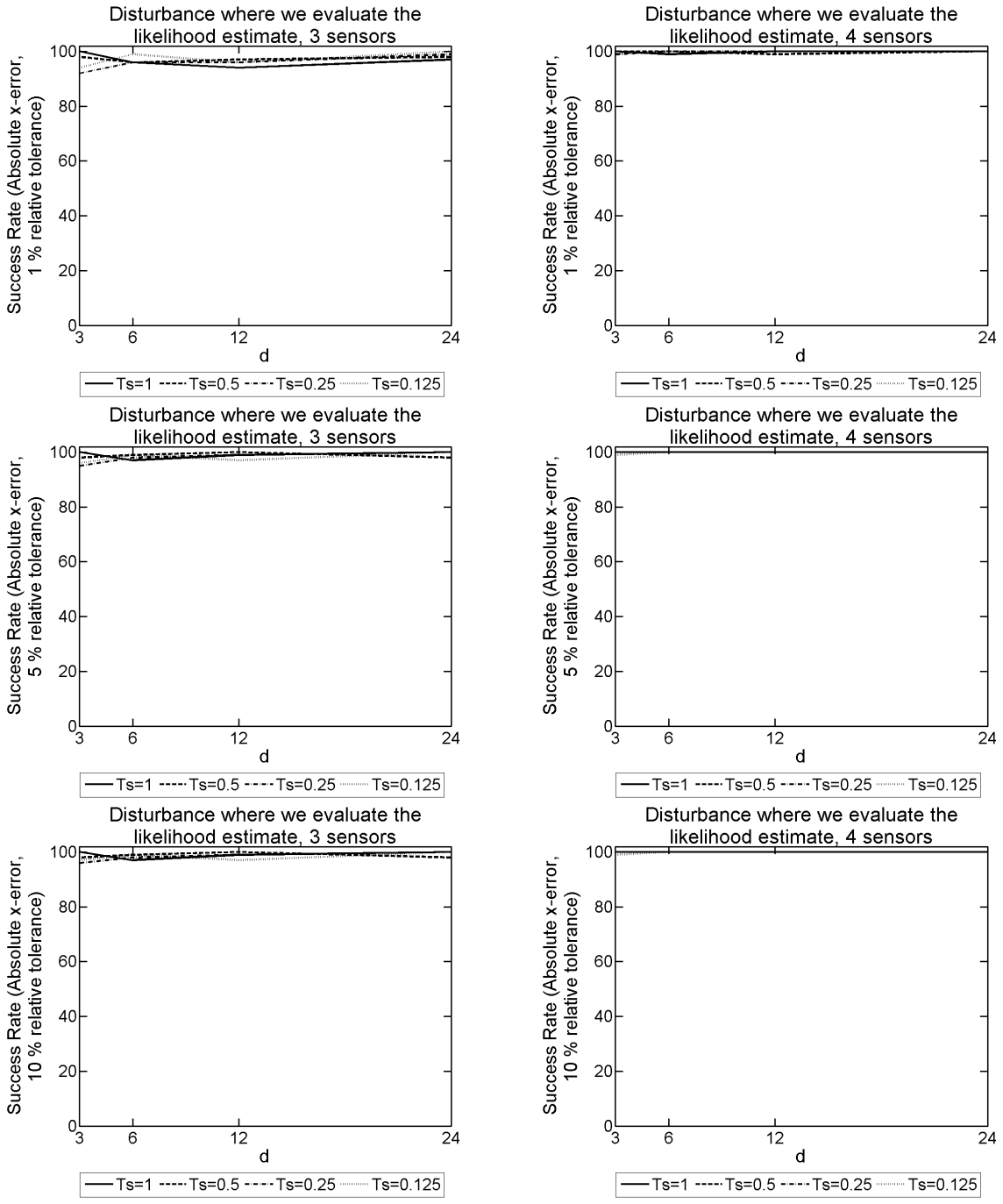


Figure 9.56: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

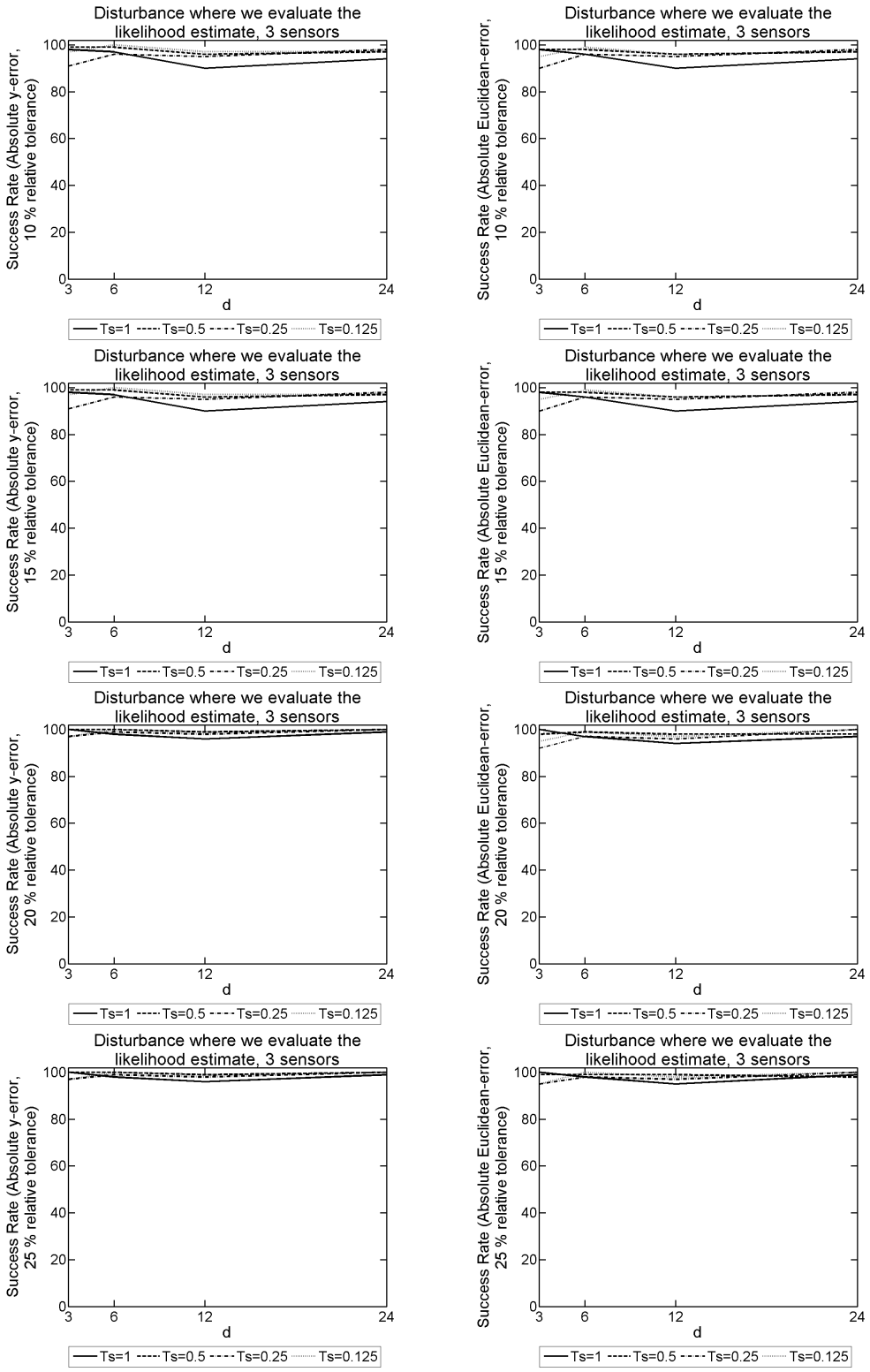


Figure 9.57: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 3 sensors present.

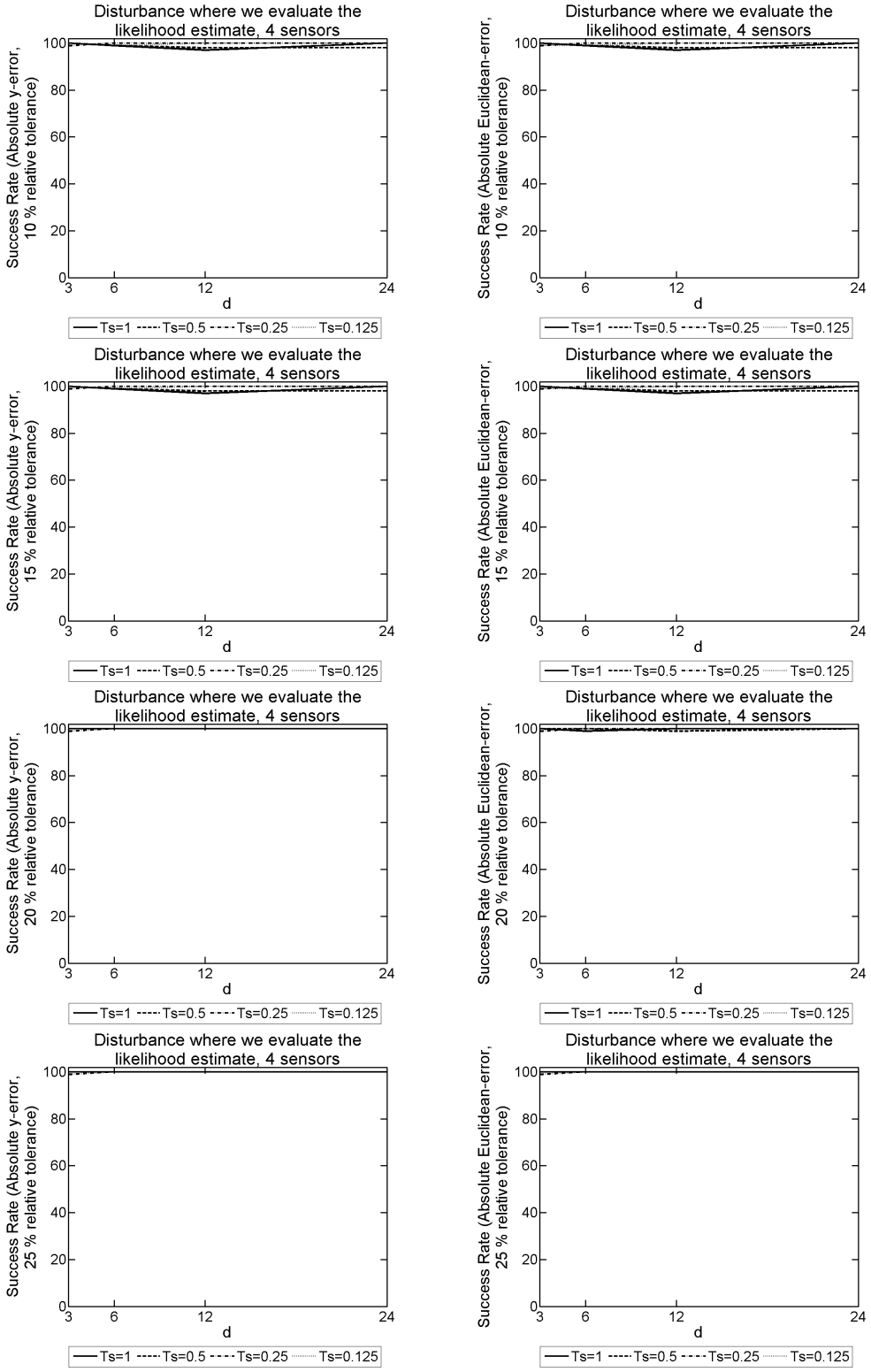


Figure 9.58: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 4 sensors present.

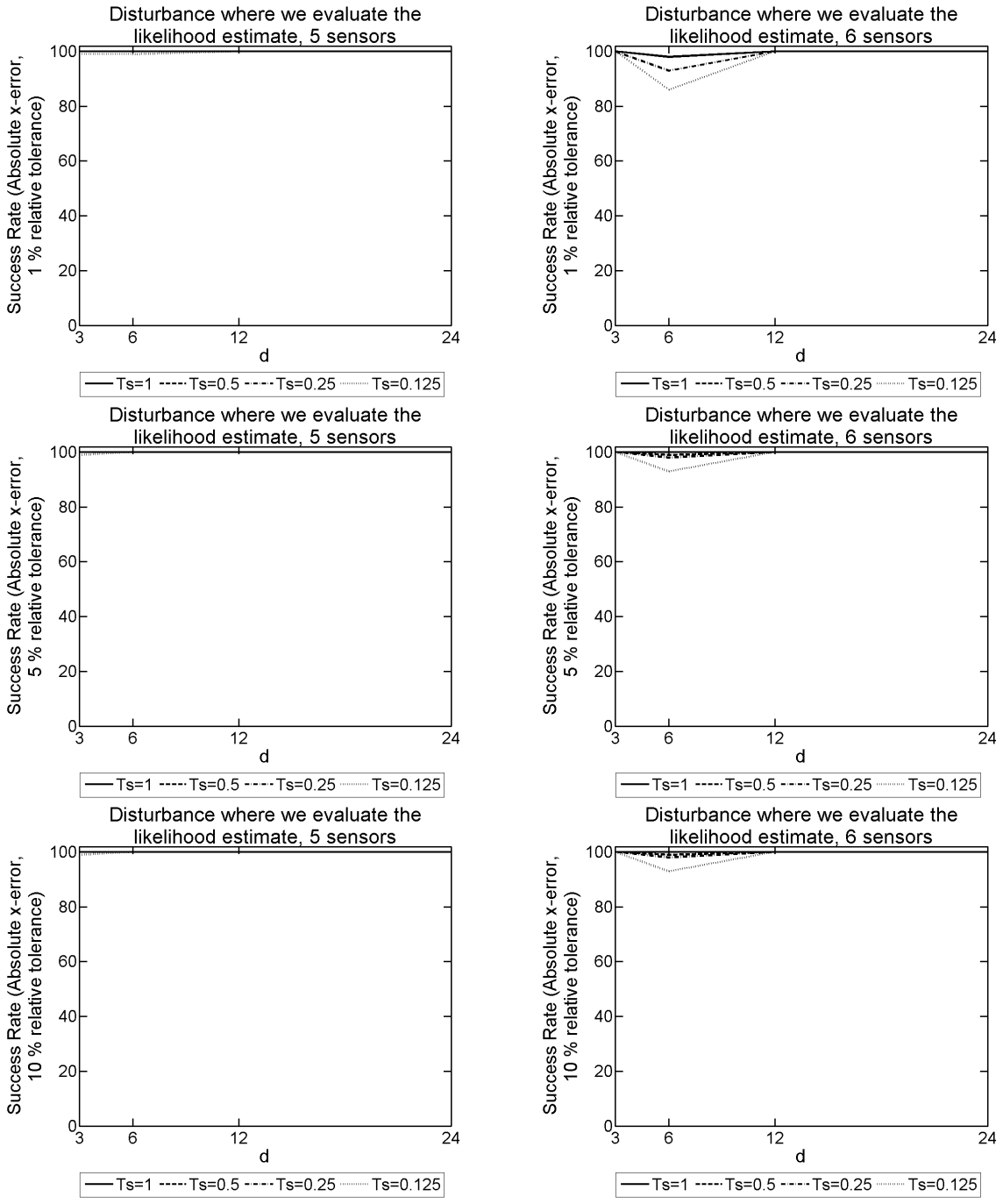


Figure 9.59: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

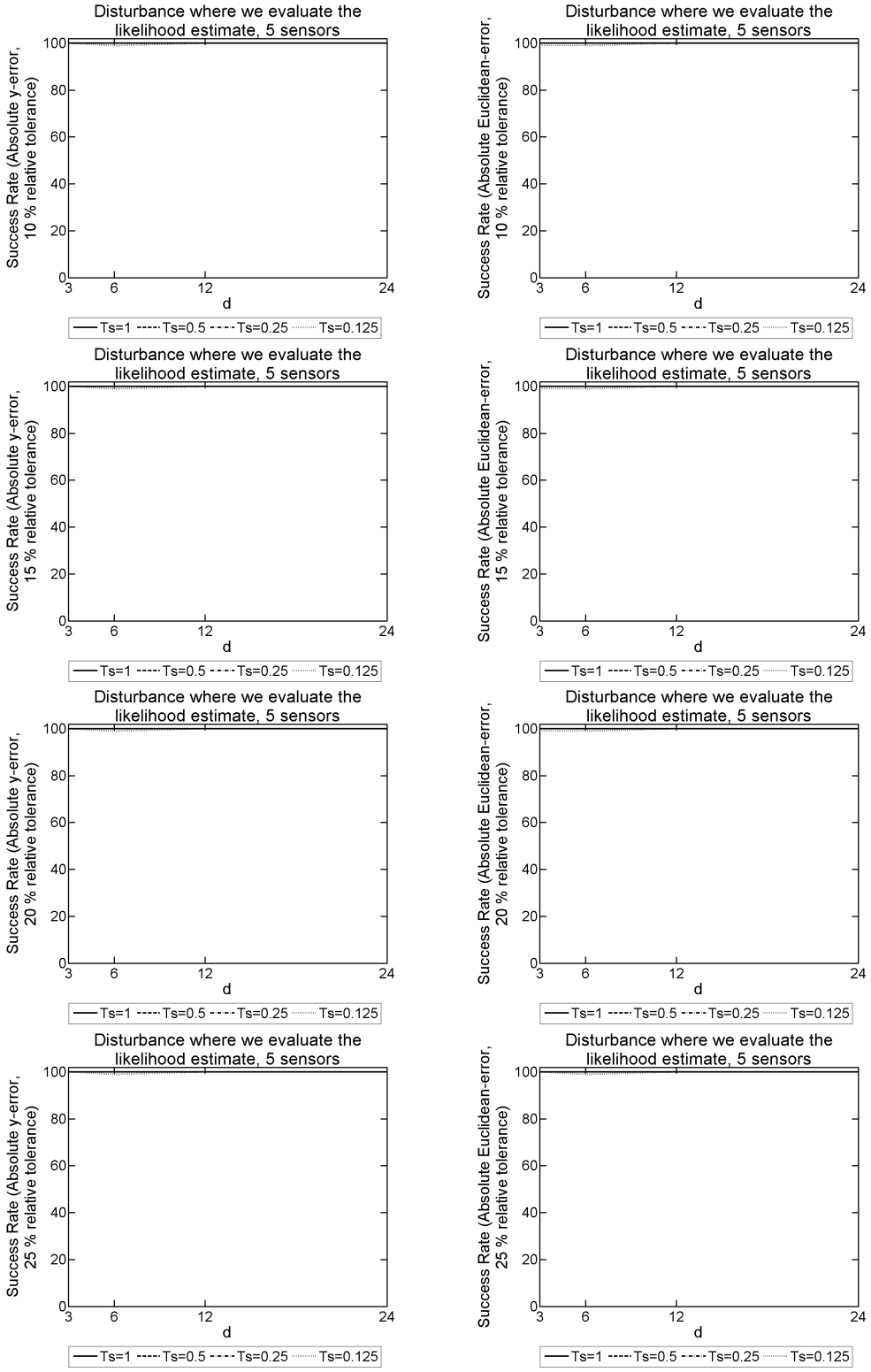


Figure 9.60: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 5 sensors present.

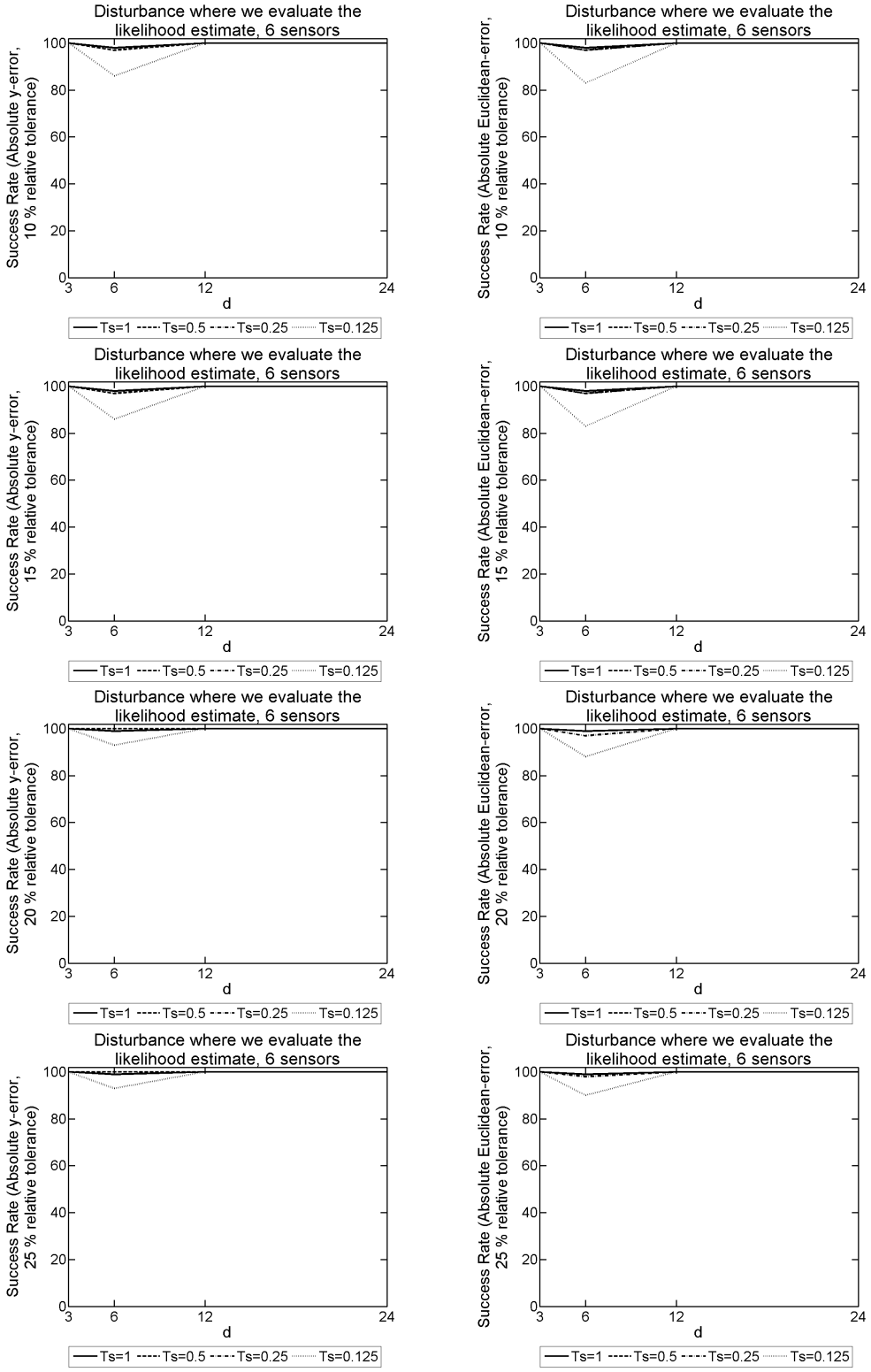


Figure 9.61: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 6 sensors present.

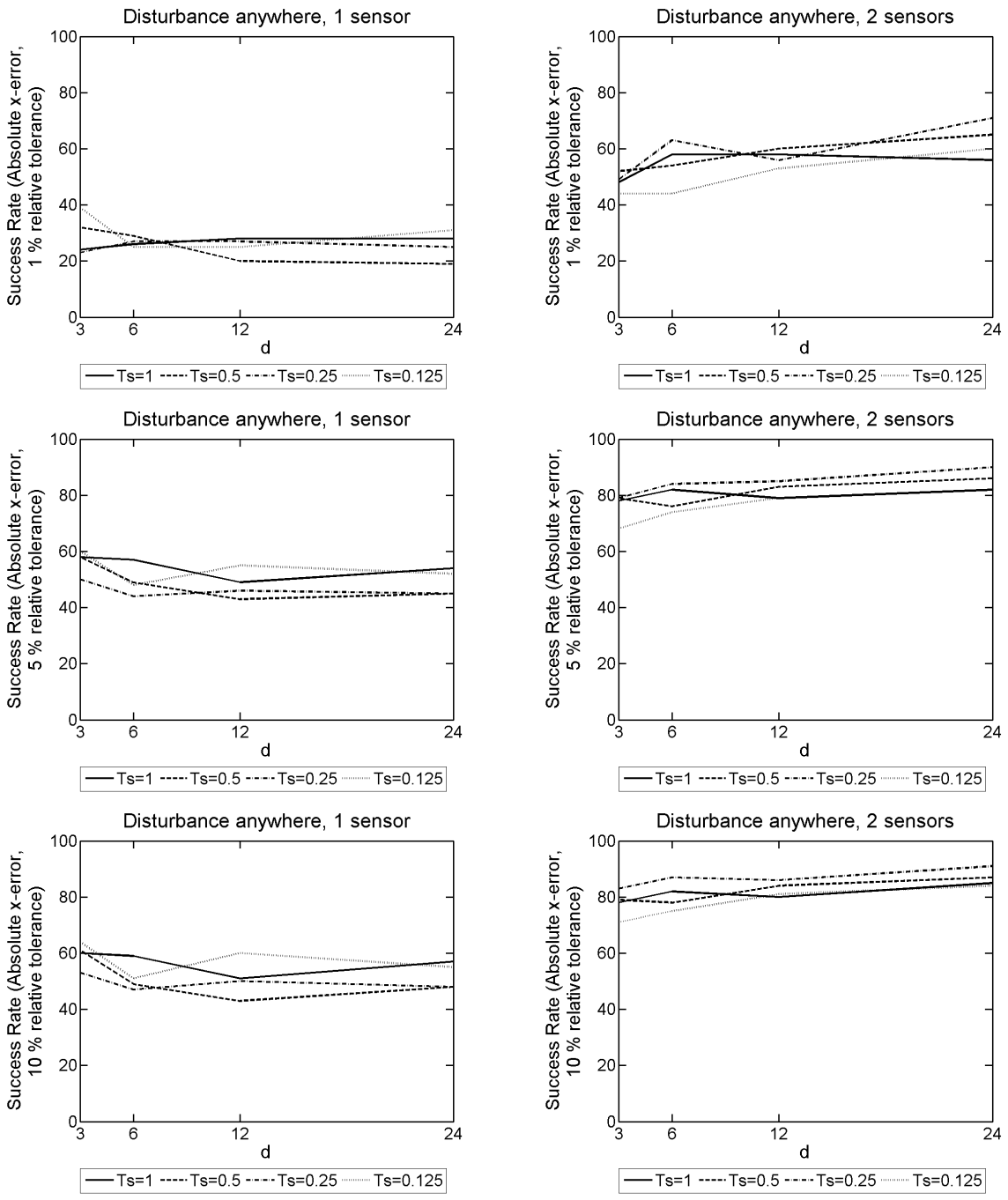


Figure 9.62: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

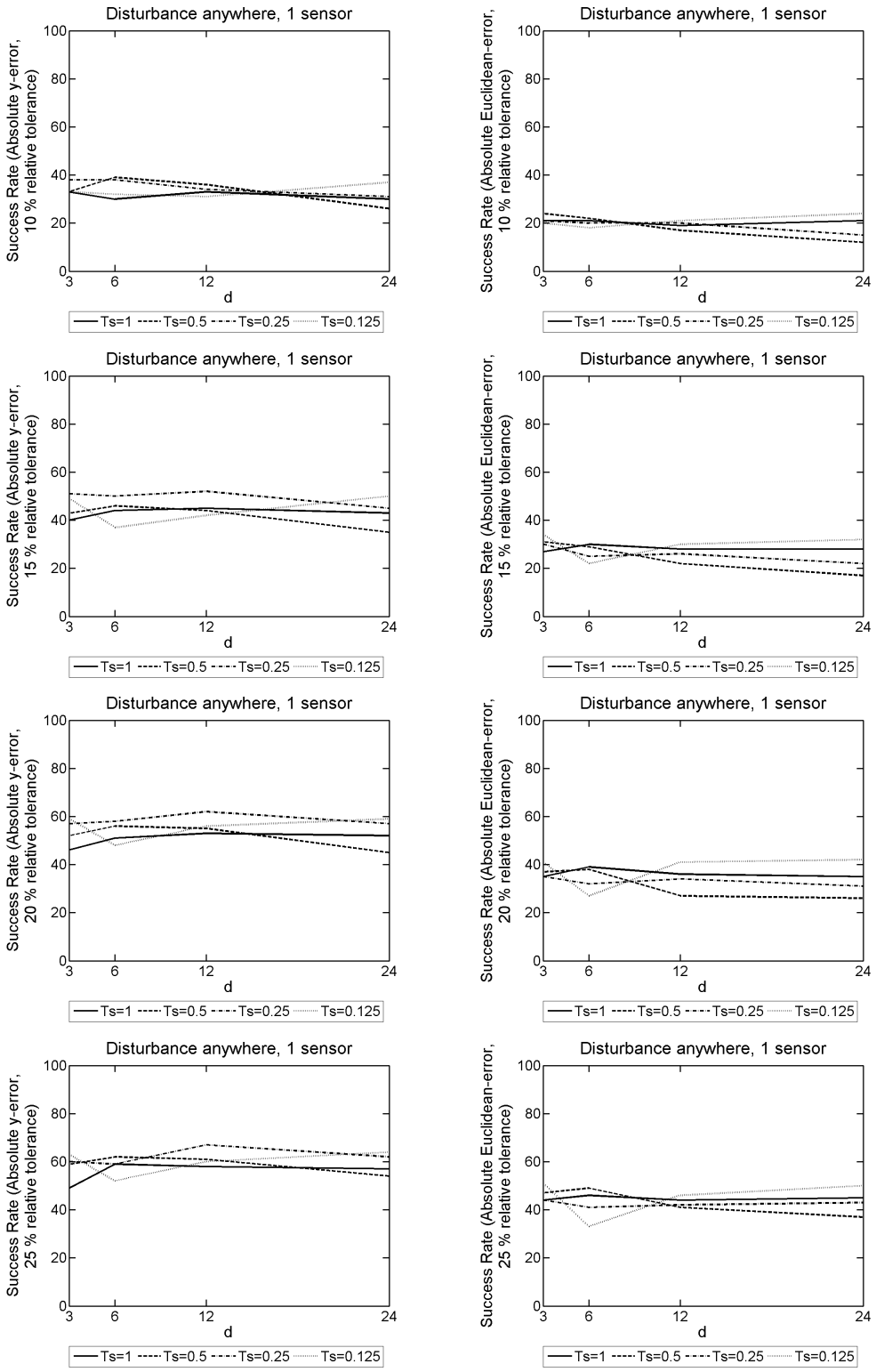


Figure 9.63: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 1 sensor present.

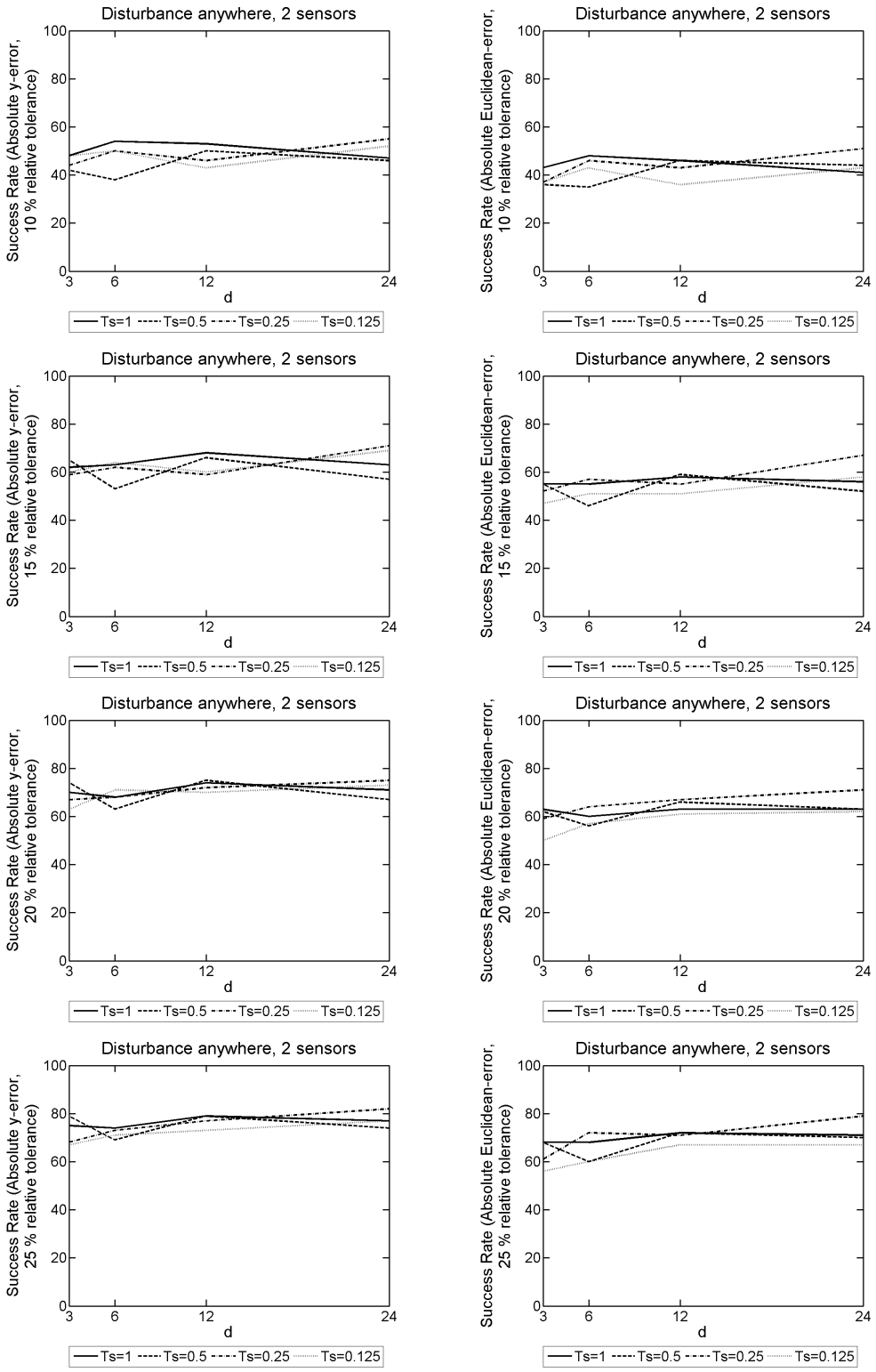


Figure 9.64: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 2 sensors present.

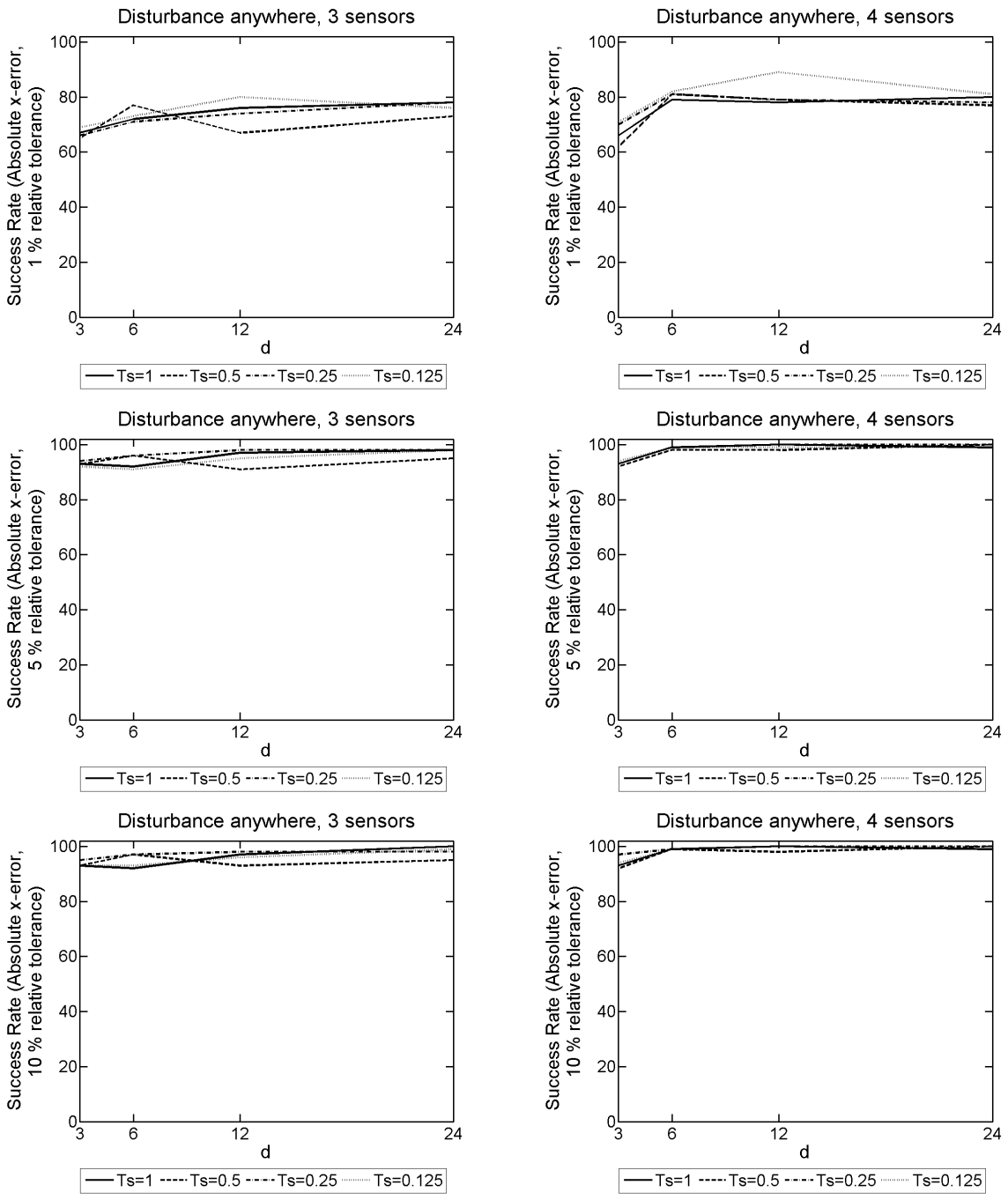


Figure 9.65: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

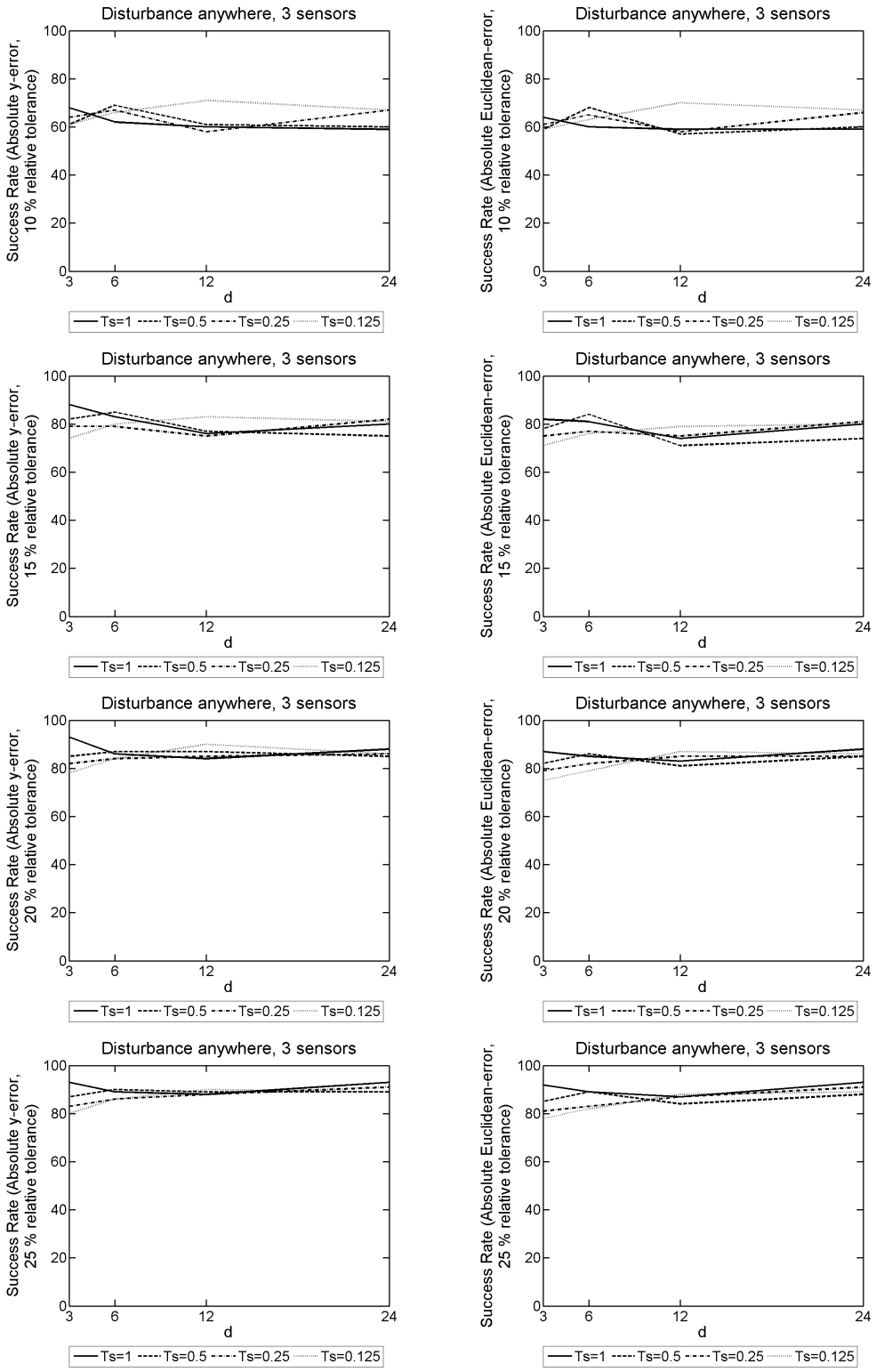


Figure 9.66: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 3 sensors present.

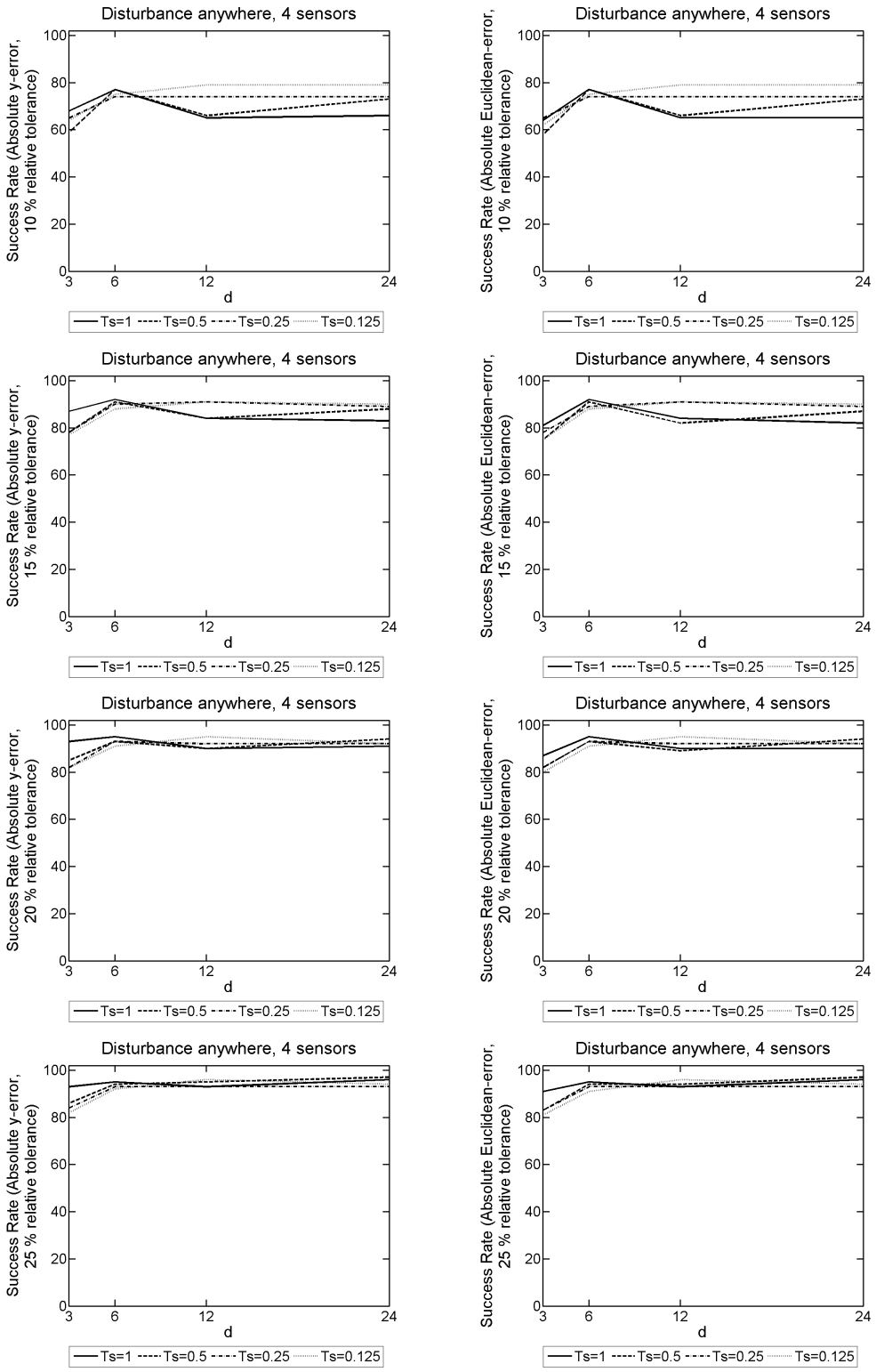


Figure 9.67: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 4 sensors present.

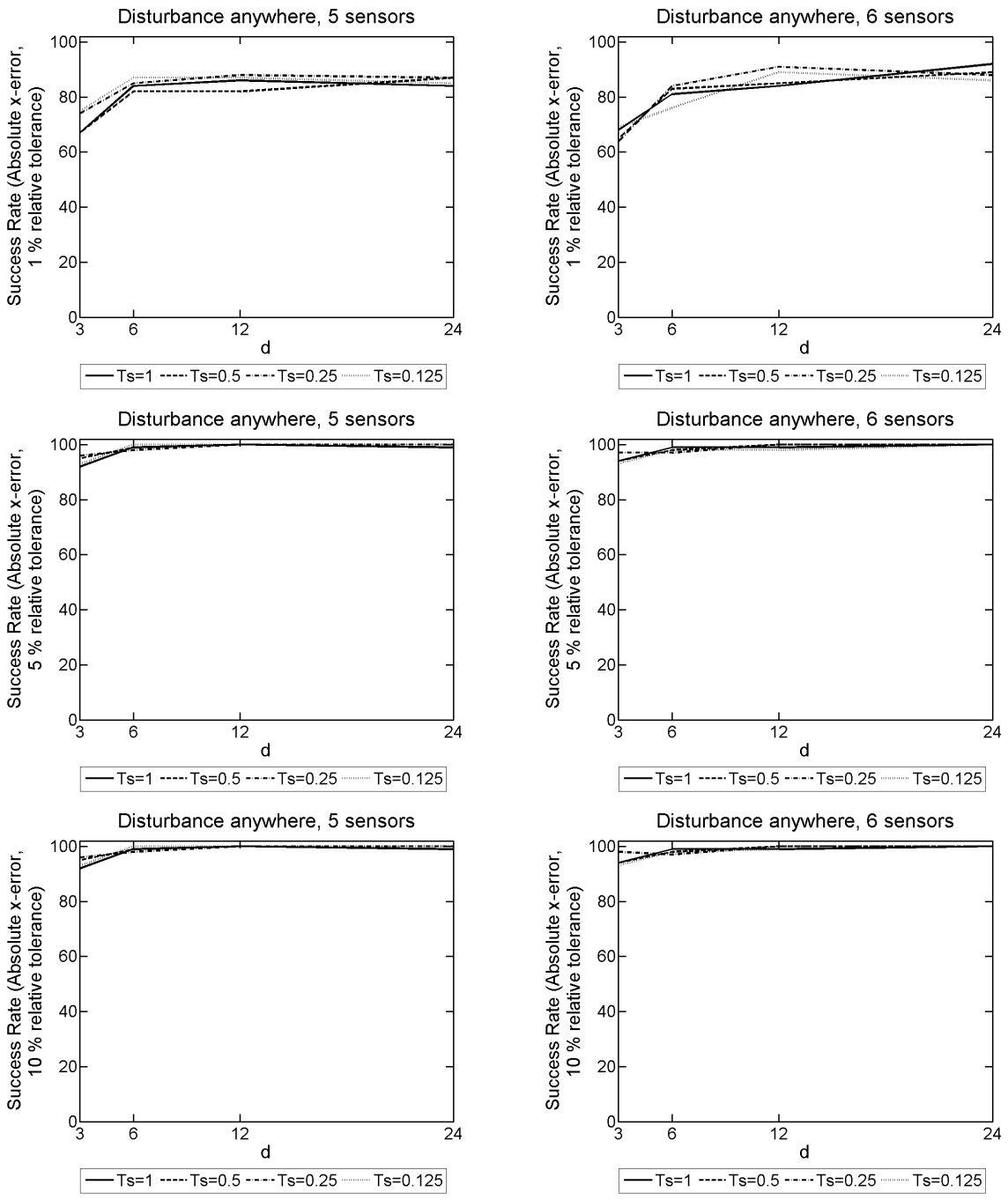


Figure 9.68: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

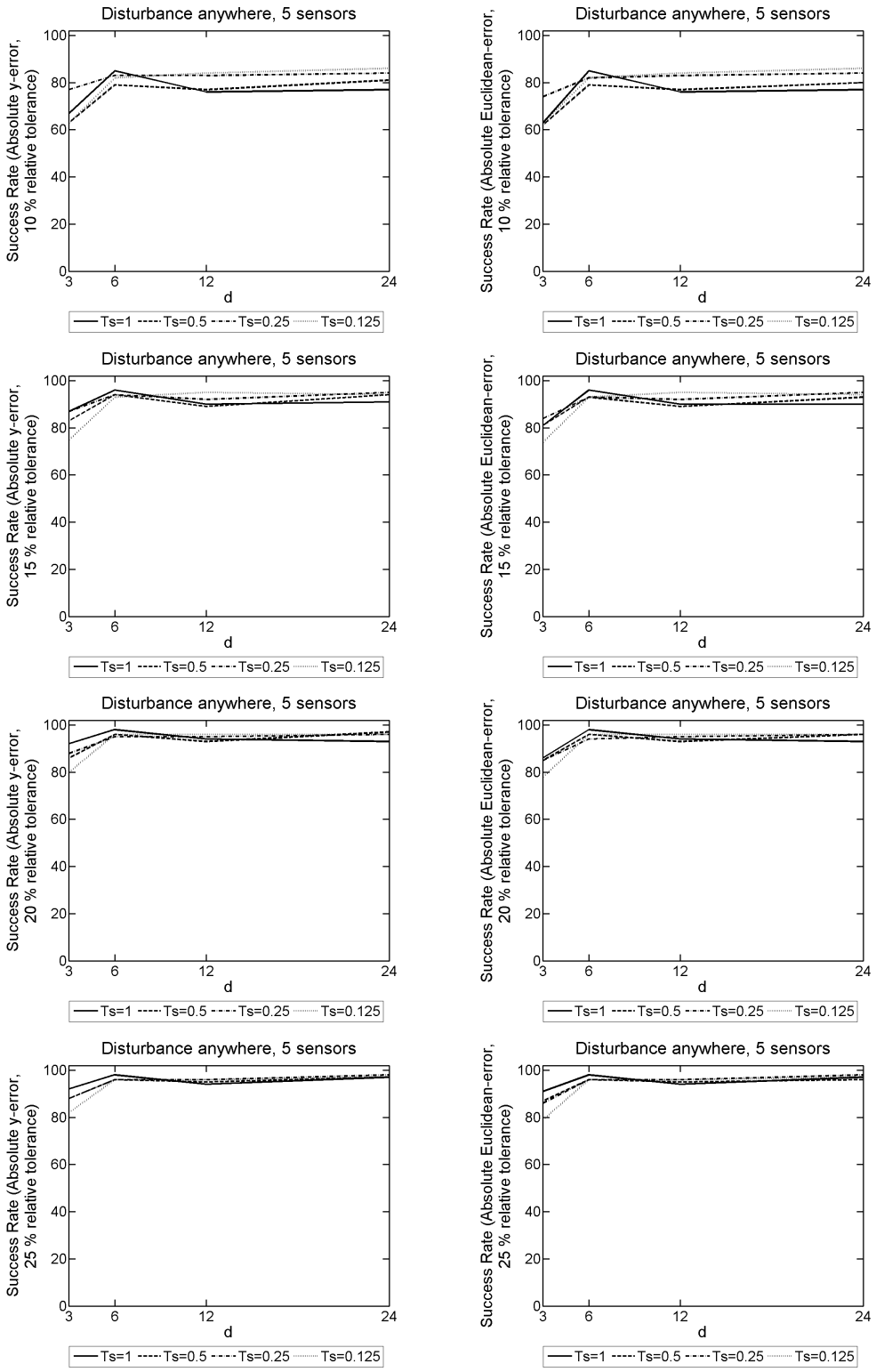


Figure 9.69: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 5 sensors present.

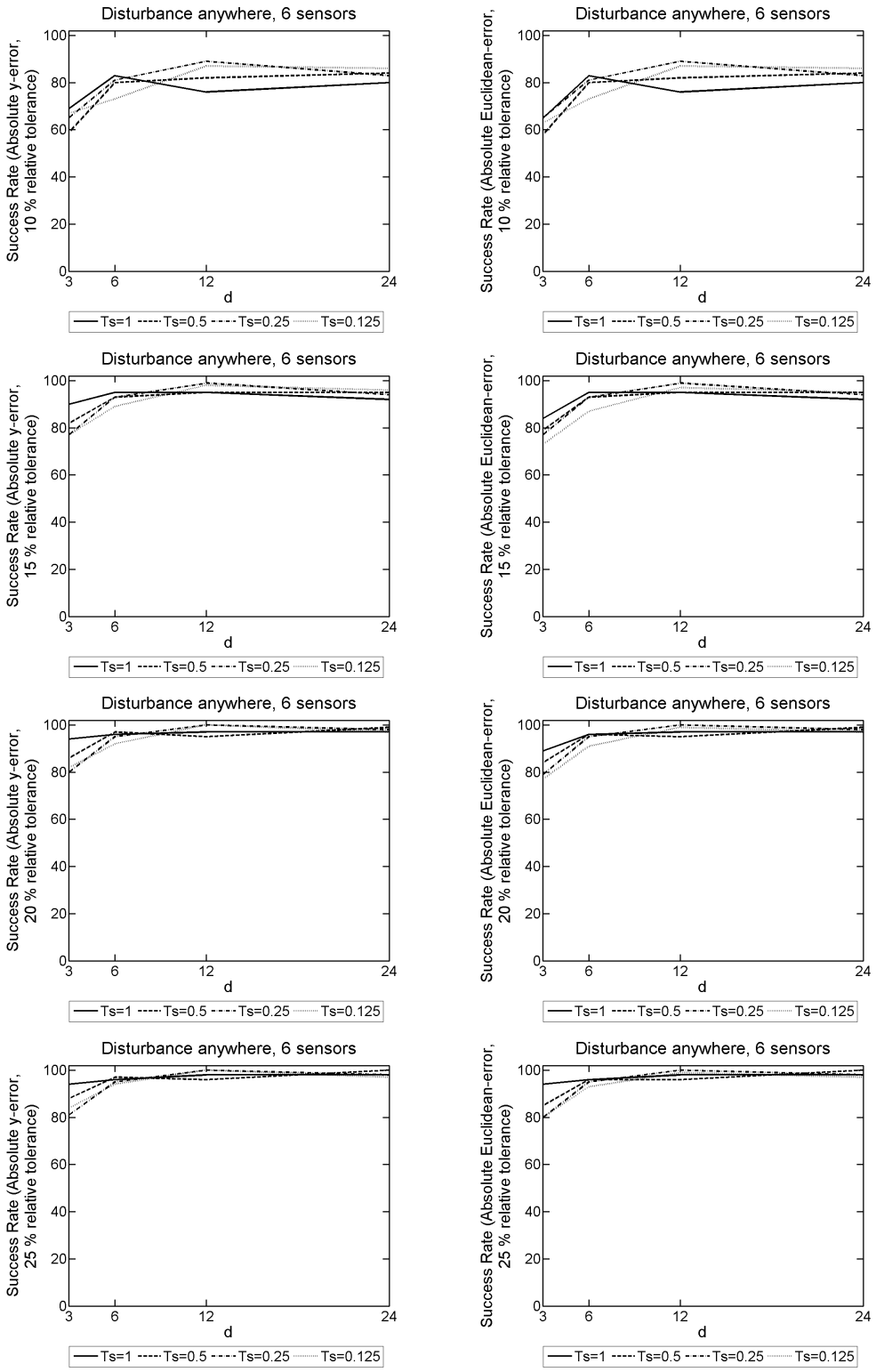


Figure 9.70: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$, $M = 5$ and $L = 9000$, and $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 6 sensors present.

Appendix D: Results for chapter 6

D.1 2D model problem with 1NBC and a disturbance frequency $F = 150\mathbf{Hz}$

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	89	90	90	90	90	94	94	89	89	90	90
1.000	6	55	57	64	64	64	82	82	56	56	56	58
1.000	12	41	43	49	53	53	75	75	40	41	44	48
1.000	24	30	35	43	44	44	75	75	33	36	39	47
0.500	3	94	95	95	95	95	98	98	93	94	97	97
0.500	6	58	63	69	65	65	85	85	58	58	61	66
0.500	12	43	46	53	51	51	72	72	46	46	47	55
0.500	24	36	37	41	51	51	68	68	38	40	41	44
0.250	3	90	92	93	91	91	99	99	90	90	91	93
0.250	6	83	85	87	83	83	91	91	82	82	85	85
0.250	12	70	71	78	76	76	88	88	71	71	75	76
0.250	24	53	55	58	66	66	78	78	54	55	56	61
0.125	3	71	73	75	75	75	83	83	71	71	71	74
0.125	6	92	96	96	94	94	98	98	92	92	94	96
0.125	12	94	95	95	93	93	98	98	91	91	93	94
0.125	24	93	94	94	95	95	99	99	92	92	95	95

Table 9.89: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	97	97	97	97	97	98	98	97	97	97	97
1.000	12	84	87	89	85	85	94	94	84	84	86	87
1.000	24	72	74	83	77	77	87	87	76	76	77	82
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	93	93	94	94	94	99	99	93	93	93	94
0.500	12	93	94	95	97	97	99	99	94	94	94	94
0.500	24	92	92	92	93	93	97	97	92	92	92	92
0.250	3	99	99	99	99	99	100	100	99	99	99	99
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	96	96	96	97	97	99	99	96	96	96	96
0.250	24	96	97	97	94	94	98	98	94	94	96	97
0.125	3	84	84	86	90	90	93	93	85	85	85	86
0.125	6	99	99	99	99	99	99	99	99	99	99	99
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.90: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	99	99	100	100	99	99	100	100
1.000	24	98	98	99	98	98	99	99	98	98	98	98
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	99	99	99	99	99	100	100	99	99	99	99
0.125	3	97	97	97	97	97	98	98	97	97	97	97
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.91: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	99	99	99	99	99	100	100	99	99	99	99
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.92: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.93: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	100	100	100	100	100	100	100	100	100	100	100
1.000	24	100	100	100	100	100	100	100	100	100	100	100
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	100	100	100	100	100	100	100	100	100	100	100
0.500	12	100	100	100	100	100	100	100	100	100	100	100
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	100	100	100	100	100	100	100	100	100	100	100
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.94: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	73	92	92	79	92	93	96	77	91	92	94
1.000	6	38	46	58	38	51	63	72	36	41	45	52
1.000	12	21	35	45	38	46	54	59	27	29	33	38
1.000	24	15	19	33	28	36	47	53	17	19	25	30
0.500	3	76	91	91	84	90	90	91	84	90	90	90
0.500	6	44	52	57	49	57	64	69	46	48	52	59
0.500	12	32	40	43	38	48	57	62	32	36	38	42
0.500	24	27	33	43	39	47	55	60	28	34	38	41
0.250	3	79	89	91	83	89	91	91	83	89	90	90
0.250	6	70	83	85	75	80	88	92	73	78	81	83
0.250	12	53	64	67	64	72	81	83	54	59	64	69
0.250	24	39	52	61	53	60	69	72	45	50	53	59
0.125	3	48	56	63	60	68	75	77	51	54	59	63
0.125	6	83	92	93	87	93	95	97	85	92	92	93
0.125	12	81	90	90	83	89	92	94	82	87	88	90
0.125	24	79	87	88	80	84	90	92	78	82	86	88

Table 9.95: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	92	99	99	92	99	99	99	92	99	99	99
1.000	6	72	81	84	76	83	86	88	71	78	81	83
1.000	12	50	61	64	54	69	70	77	49	55	56	64
1.000	24	33	40	55	49	55	59	60	38	41	46	49
0.500	3	92	99	99	93	98	99	99	93	98	99	99
0.500	6	78	86	88	81	90	91	92	77	85	85	86
0.500	12	70	81	84	72	79	85	85	71	78	80	82
0.500	24	59	68	75	65	72	78	78	62	68	70	71
0.250	3	89	94	94	90	94	94	95	89	93	93	93
0.250	6	86	97	98	86	95	97	98	86	94	96	97
0.250	12	89	94	94	89	94	97	97	88	93	94	94
0.250	24	73	82	84	78	86	89	89	77	82	82	83
0.125	3	69	77	78	68	74	77	78	65	69	71	71
0.125	6	90	96	97	91	95	96	96	91	95	96	96
0.125	12	85	93	94	88	92	93	94	88	92	93	93
0.125	24	89	95	96	90	95	96	96	90	94	95	95

Table 9.96: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	95	100	100	95	100	100	100	95	100	100	100
1.000	6	84	92	93	86	96	97	98	84	91	92	92
1.000	12	64	72	76	69	73	77	77	64	66	69	71
1.000	24	56	71	74	68	72	75	76	64	67	69	70
0.500	3	97	100	100	96	100	100	100	96	100	100	100
0.500	6	89	96	96	92	97	97	97	92	96	96	96
0.500	12	83	91	92	82	89	92	92	82	88	89	90
0.500	24	81	84	84	80	86	90	92	79	83	83	84
0.250	3	89	94	94	92	96	96	96	90	94	95	95
0.250	6	94	100	100	95	99	100	100	95	99	100	100
0.250	12	93	97	97	93	98	99	99	93	96	97	97
0.250	24	90	95	95	91	96	97	97	90	95	95	95
0.125	3	71	78	79	68	74	76	78	67	72	74	75
0.125	6	89	98	99	93	97	98	98	93	97	98	98
0.125	12	91	97	98	95	97	97	97	95	97	97	97
0.125	24	93	98	98	93	98	98	98	93	98	98	98

Table 9.97: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	95	100	100	95	100	100	100	95	100	100	100
1.000	6	91	95	95	90	96	96	96	89	95	95	95
1.000	12	74	83	86	79	82	83	84	78	81	81	83
1.000	24	69	78	81	74	78	78	79	71	76	76	77
0.500	3	98	100	100	96	100	100	100	96	100	100	100
0.500	6	96	98	98	94	99	99	99	93	98	98	98
0.500	12	88	93	94	90	94	94	94	89	93	93	93
0.500	24	83	91	92	85	92	92	92	84	91	91	91
0.250	3	90	94	95	92	96	96	97	90	94	94	94
0.250	6	97	100	100	97	100	100	100	97	100	100	100
0.250	12	94	99	99	93	99	100	100	93	99	99	100
0.250	24	93	94	96	91	98	98	98	91	95	96	96
0.125	3	69	76	77	70	75	76	78	68	72	73	75
0.125	6	95	99	99	95	98	99	99	95	98	99	99
0.125	12	90	98	98	95	98	98	98	95	98	98	98
0.125	24	92	99	99	96	99	99	99	96	99	99	99

Table 9.98: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	94	100	100	95	100	100	100	95	100	100	100
1.000	6	96	100	100	96	100	100	100	96	100	100	100
1.000	12	78	85	89	82	85	89	91	79	82	82	84
1.000	24	71	77	83	73	78	81	85	69	74	77	80
0.500	3	98	100	100	97	100	100	100	97	100	100	100
0.500	6	98	100	100	97	100	100	100	97	100	100	100
0.500	12	94	99	99	91	98	99	99	91	98	99	99
0.500	24	88	94	95	87	93	96	97	87	92	93	94
0.250	3	90	94	94	92	96	96	96	90	94	94	95
0.250	6	97	100	100	97	100	100	100	97	100	100	100
0.250	12	96	100	100	98	100	100	100	98	100	100	100
0.250	24	94	98	98	93	97	99	99	93	97	98	98
0.125	3	71	82	83	71	76	79	79	70	75	78	78
0.125	6	95	100	100	98	100	100	100	98	100	100	100
0.125	12	93	98	99	95	98	98	98	95	98	98	98
0.125	24	90	99	99	95	99	99	99	95	99	99	99

Table 9.99: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	94	100	100	95	100	100	100	95	100	100	100
1.000	6	97	100	100	95	100	100	100	95	100	100	100
1.000	12	88	93	95	85	91	93	93	85	91	92	92
1.000	24	75	84	88	76	81	83	84	73	77	79	80
0.500	3	98	100	100	97	100	100	100	97	100	100	100
0.500	6	97	100	100	95	100	100	100	95	100	100	100
0.500	12	94	99	99	95	99	99	99	95	99	99	99
0.500	24	92	97	97	91	95	97	97	91	95	96	96
0.250	3	90	94	95	92	96	96	96	90	94	95	95
0.250	6	98	100	100	98	100	100	100	98	100	100	100
0.250	12	96	100	100	97	100	100	100	97	100	100	100
0.250	24	93	98	98	95	99	99	99	95	98	98	99
0.125	3	72	81	82	71	76	78	81	70	75	77	78
0.125	6	95	100	100	97	99	100	100	97	99	100	100
0.125	12	96	100	100	99	100	100	100	99	100	100	100
0.125	24	95	99	99	93	100	100	100	93	99	99	99

Table 9.100: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

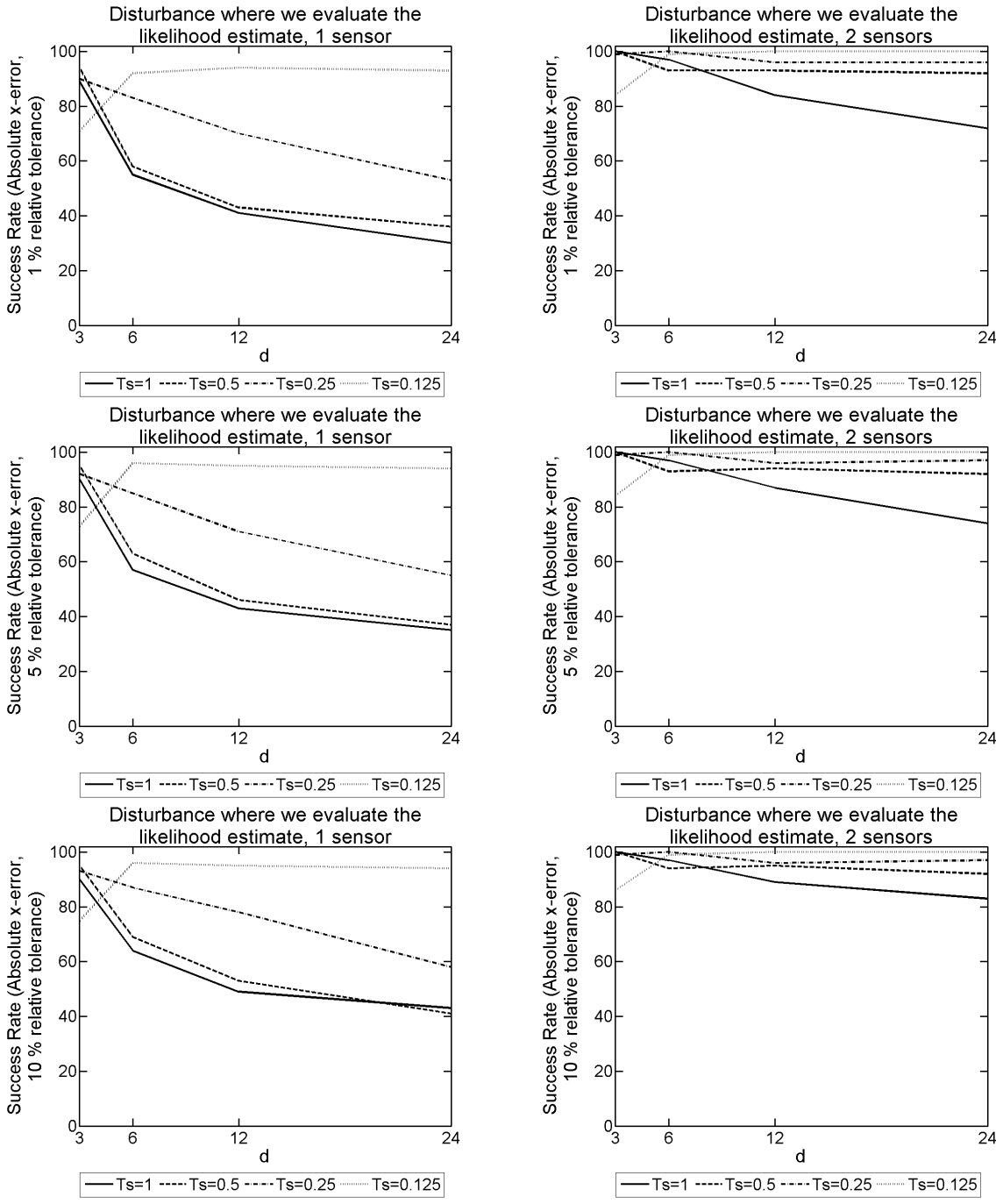


Figure 9.71: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

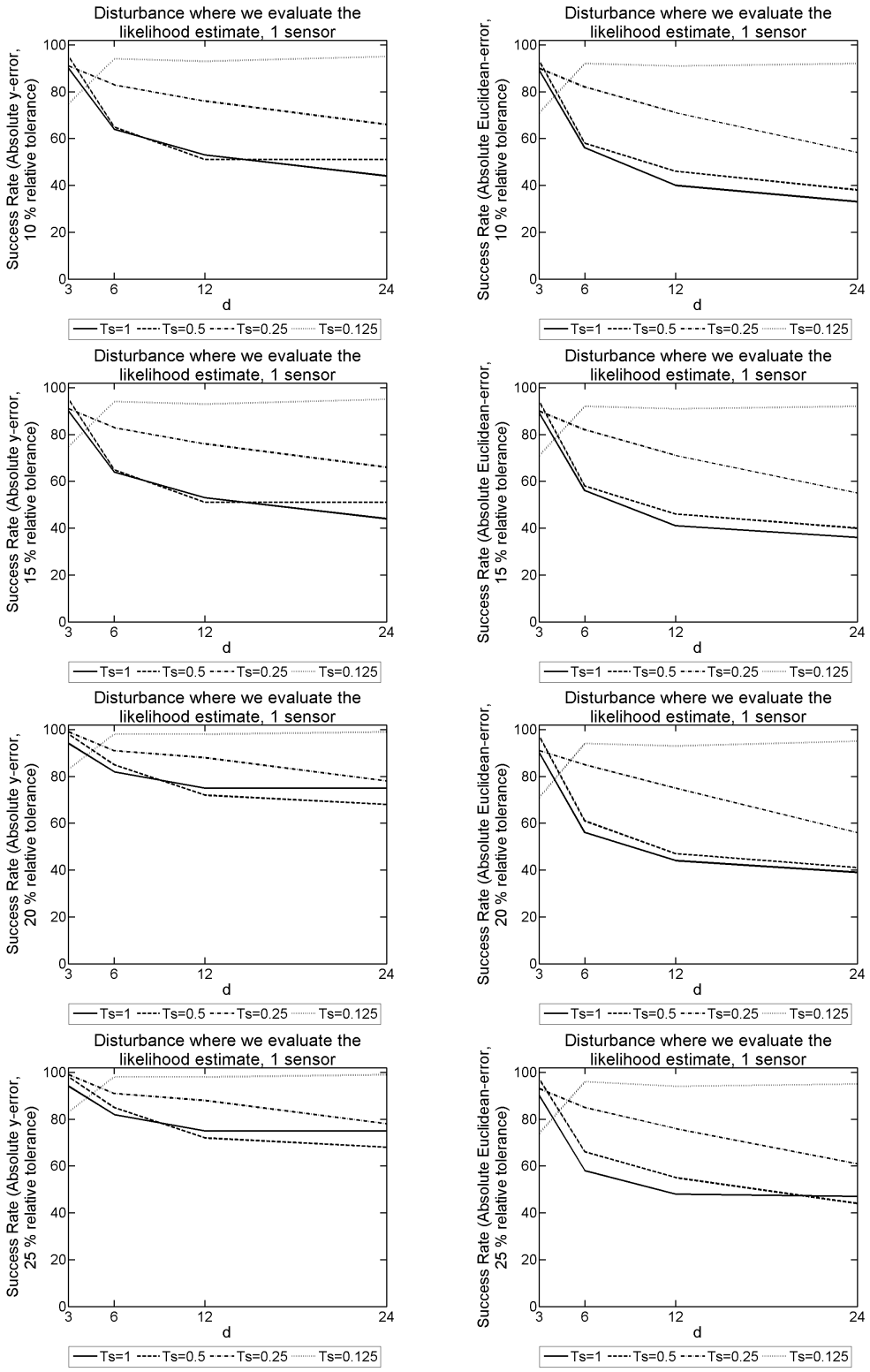


Figure 9.72: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 1 sensor present.

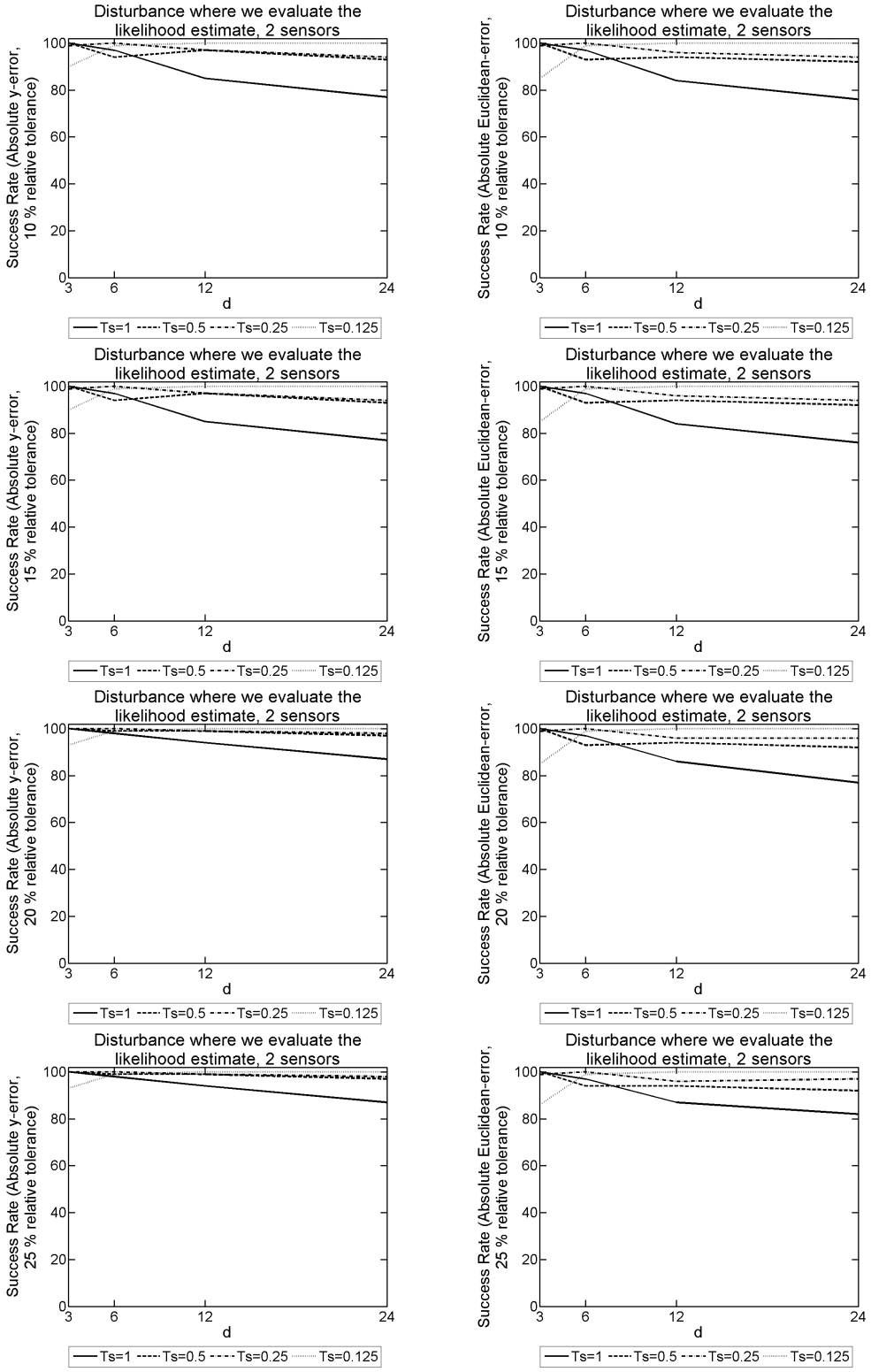


Figure 9.73: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 2 sensors present.

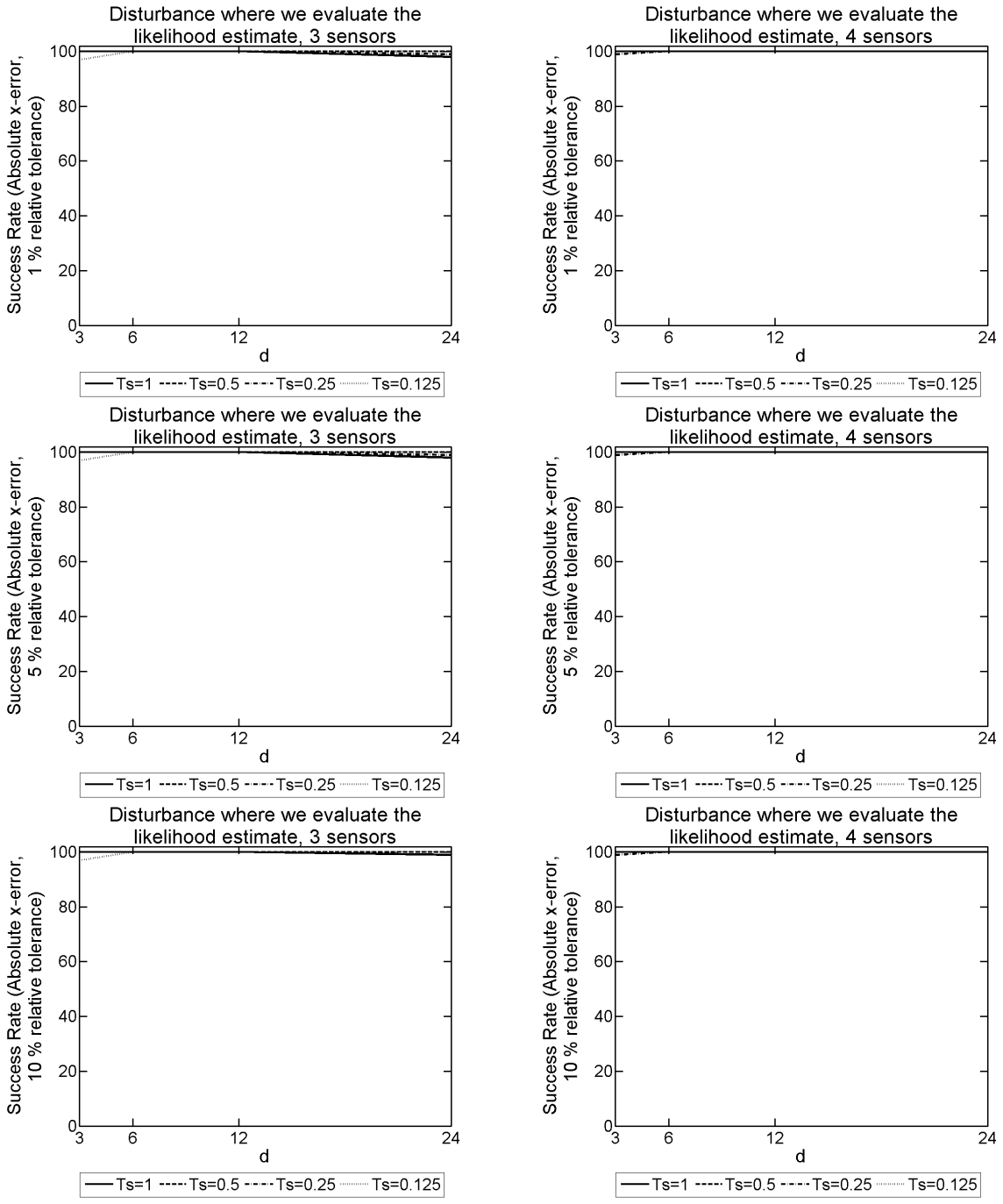


Figure 9.74: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

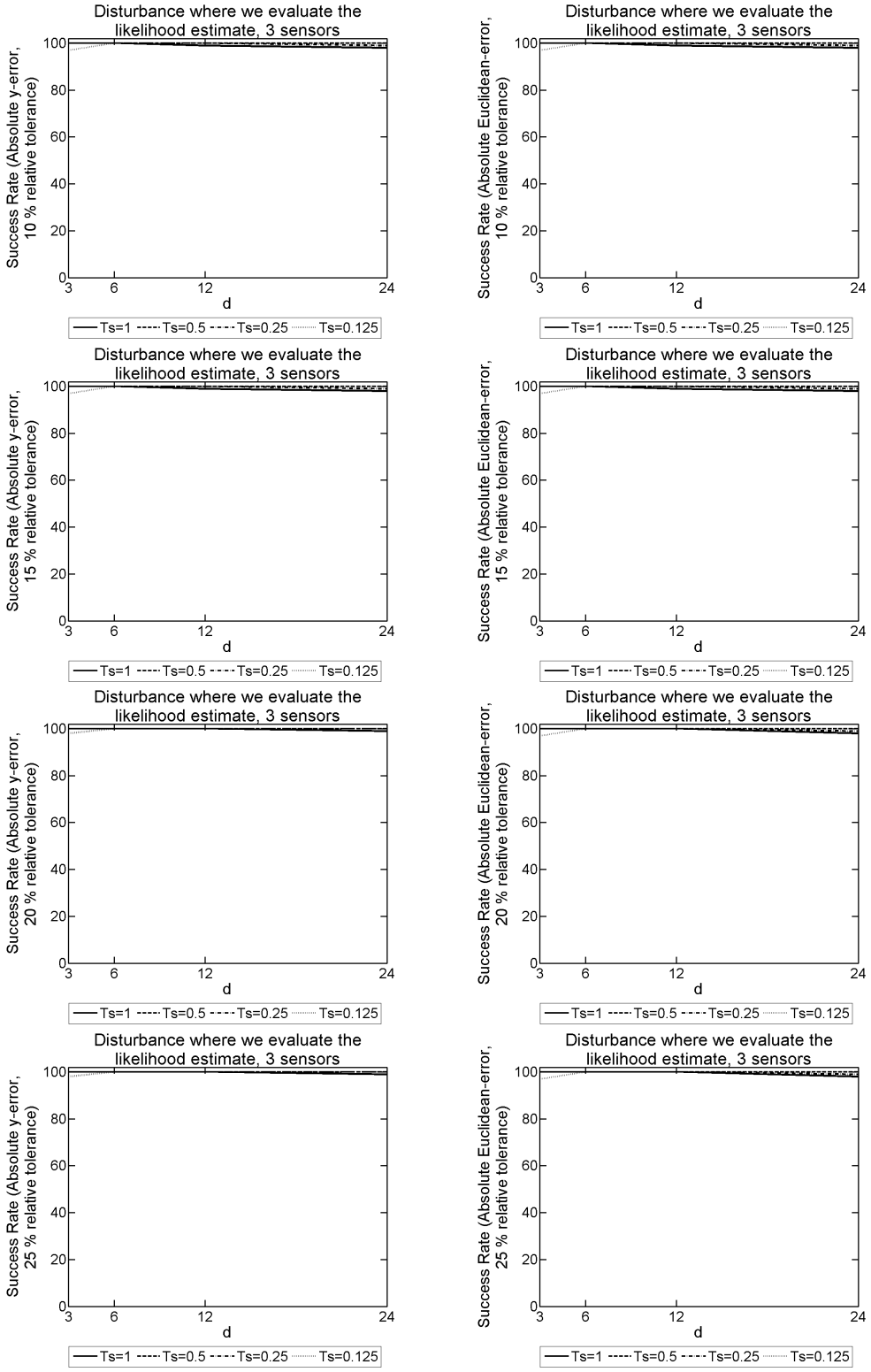


Figure 9.75: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 3 sensors present.

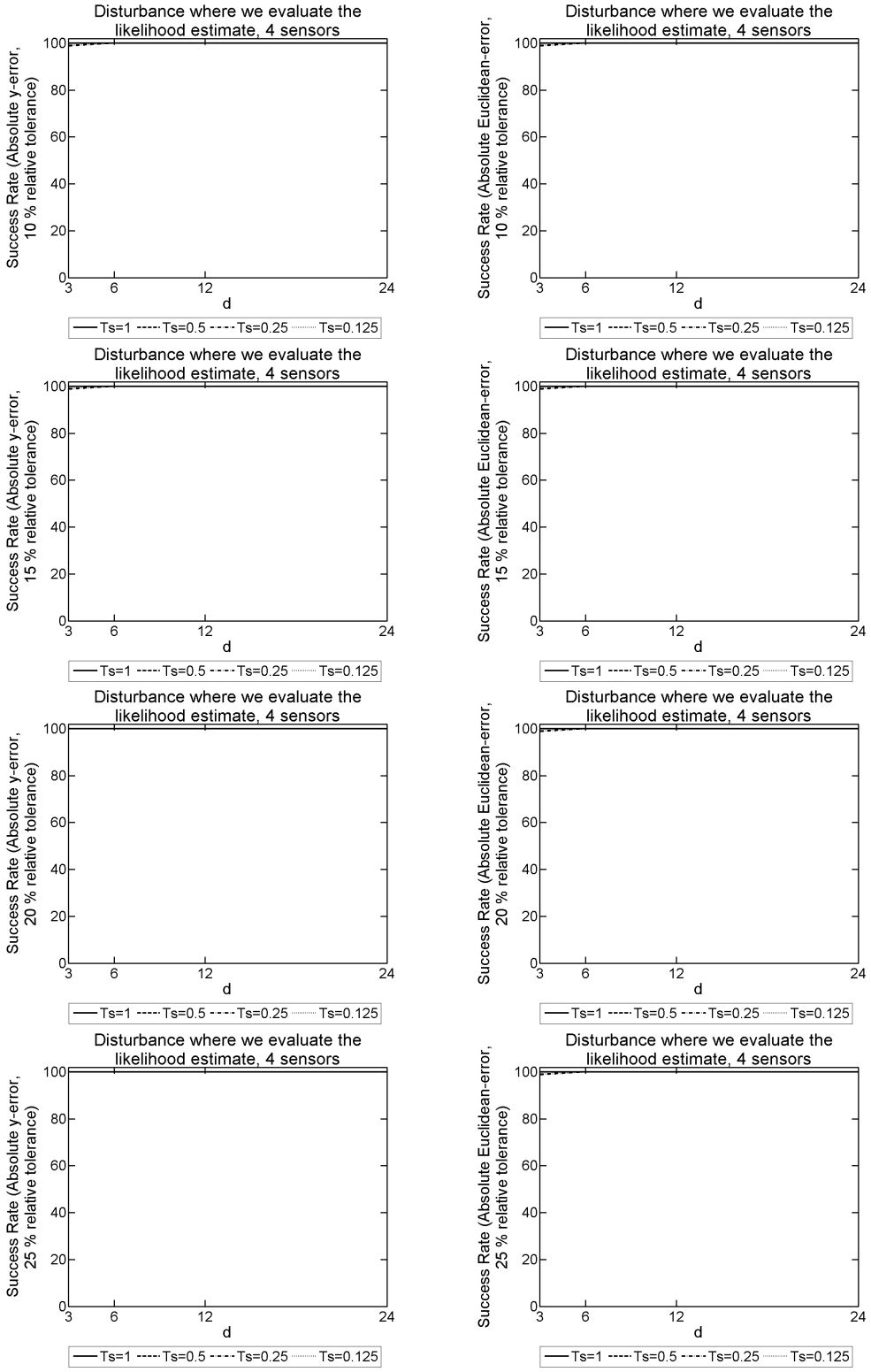


Figure 9.76: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 4 sensors present.

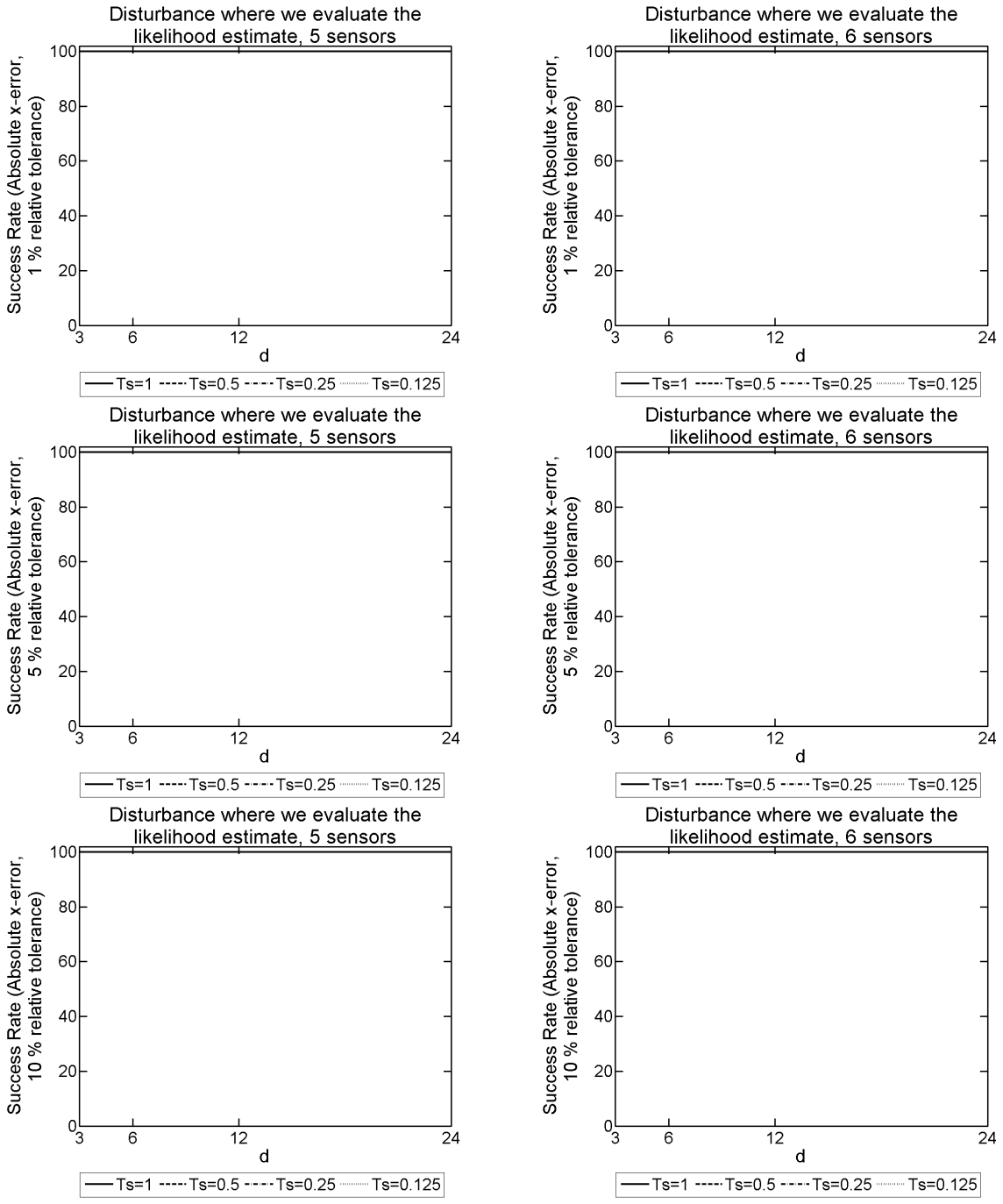


Figure 9.77: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

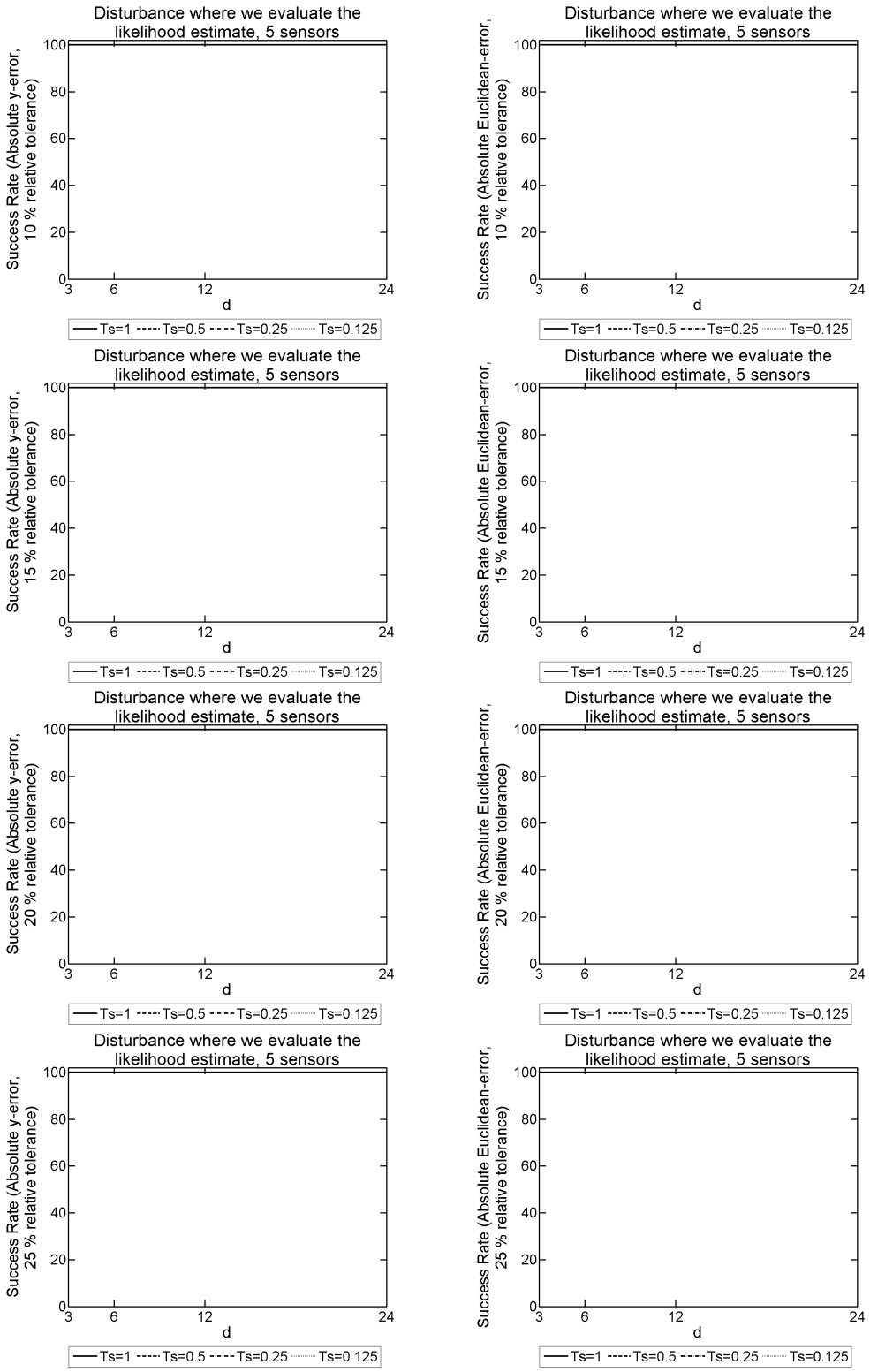


Figure 9.78: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 5 sensors present.

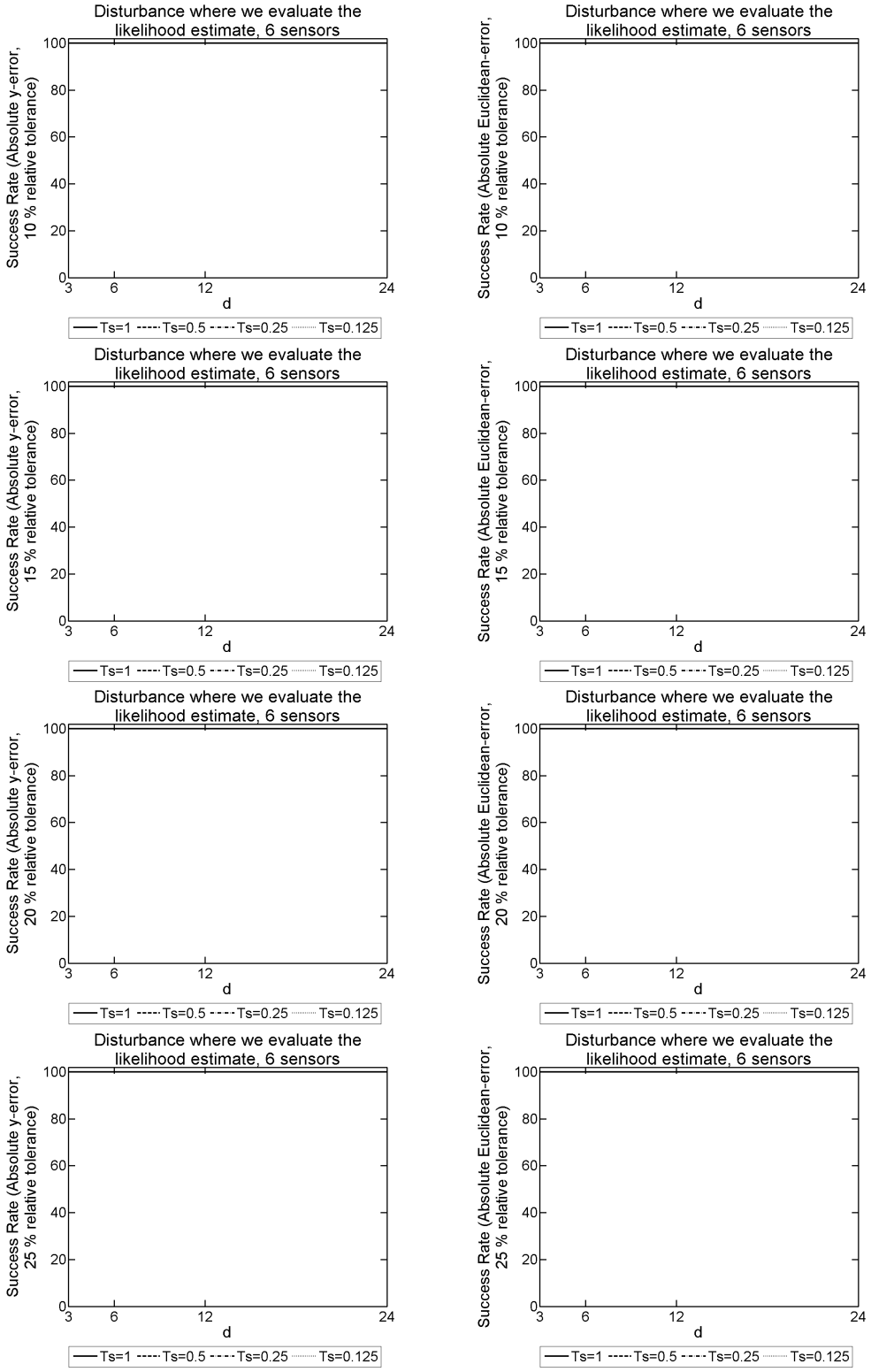


Figure 9.79: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 6 sensors present.

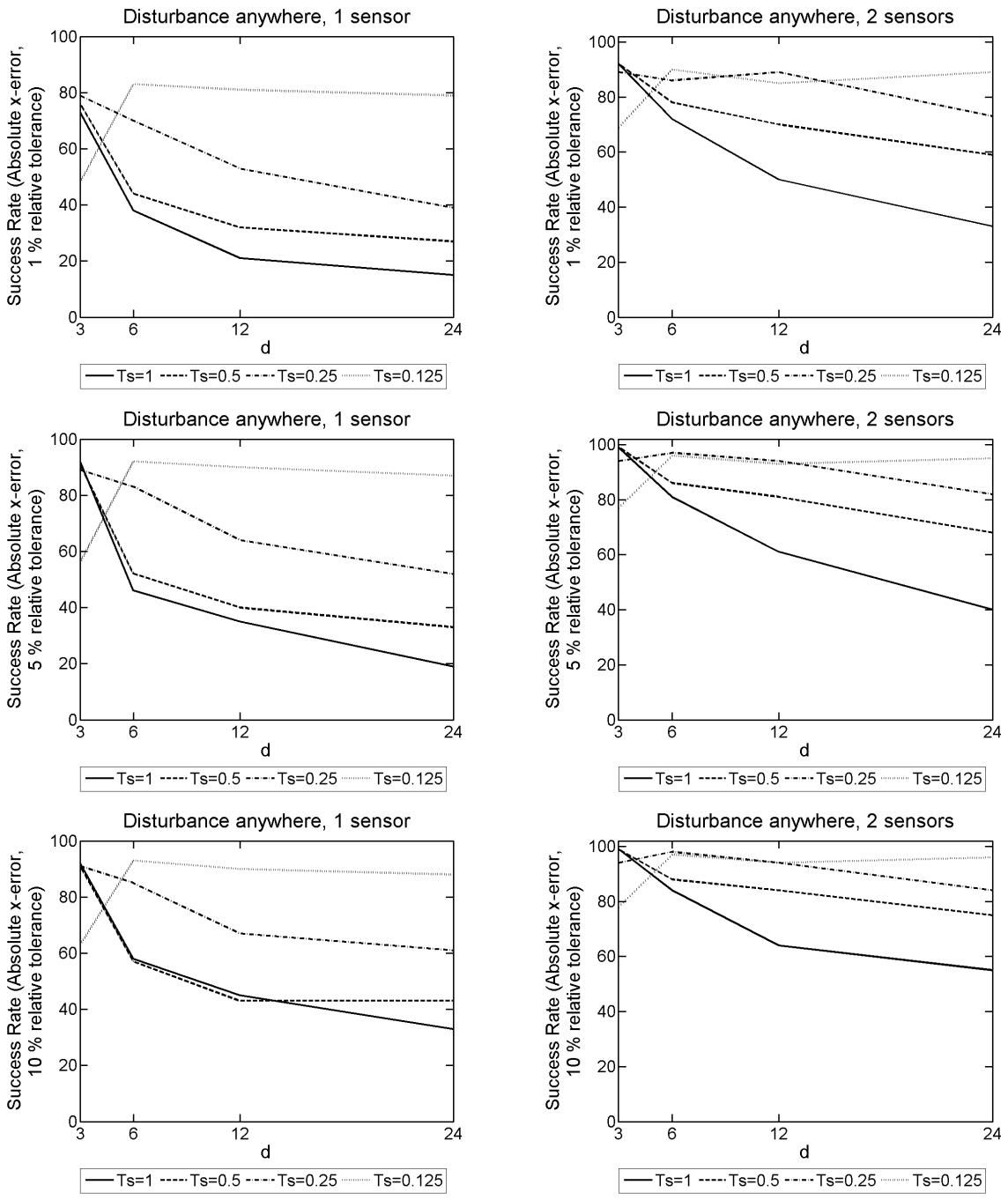


Figure 9.80: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

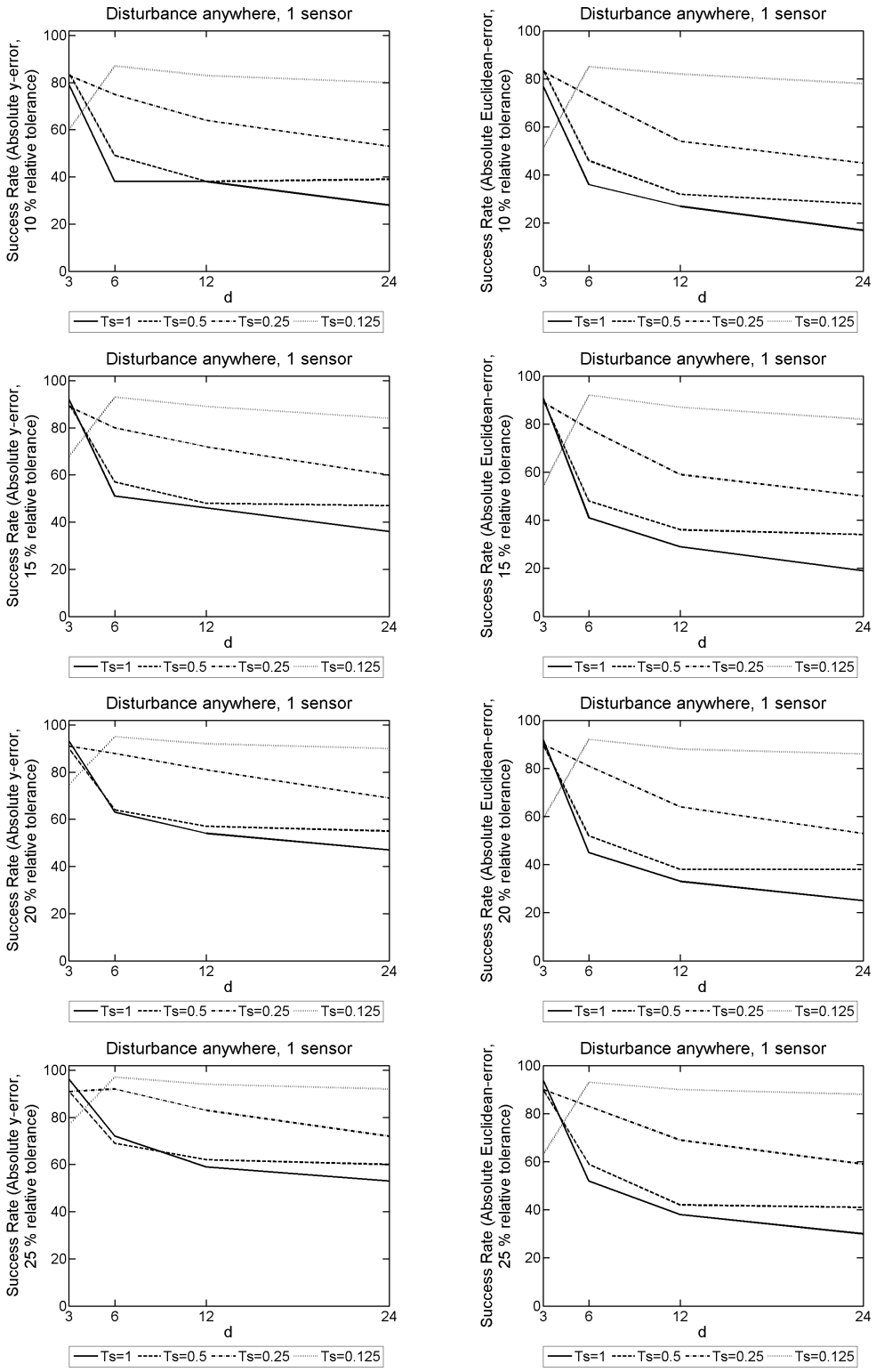


Figure 9.81: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 1 sensor present.

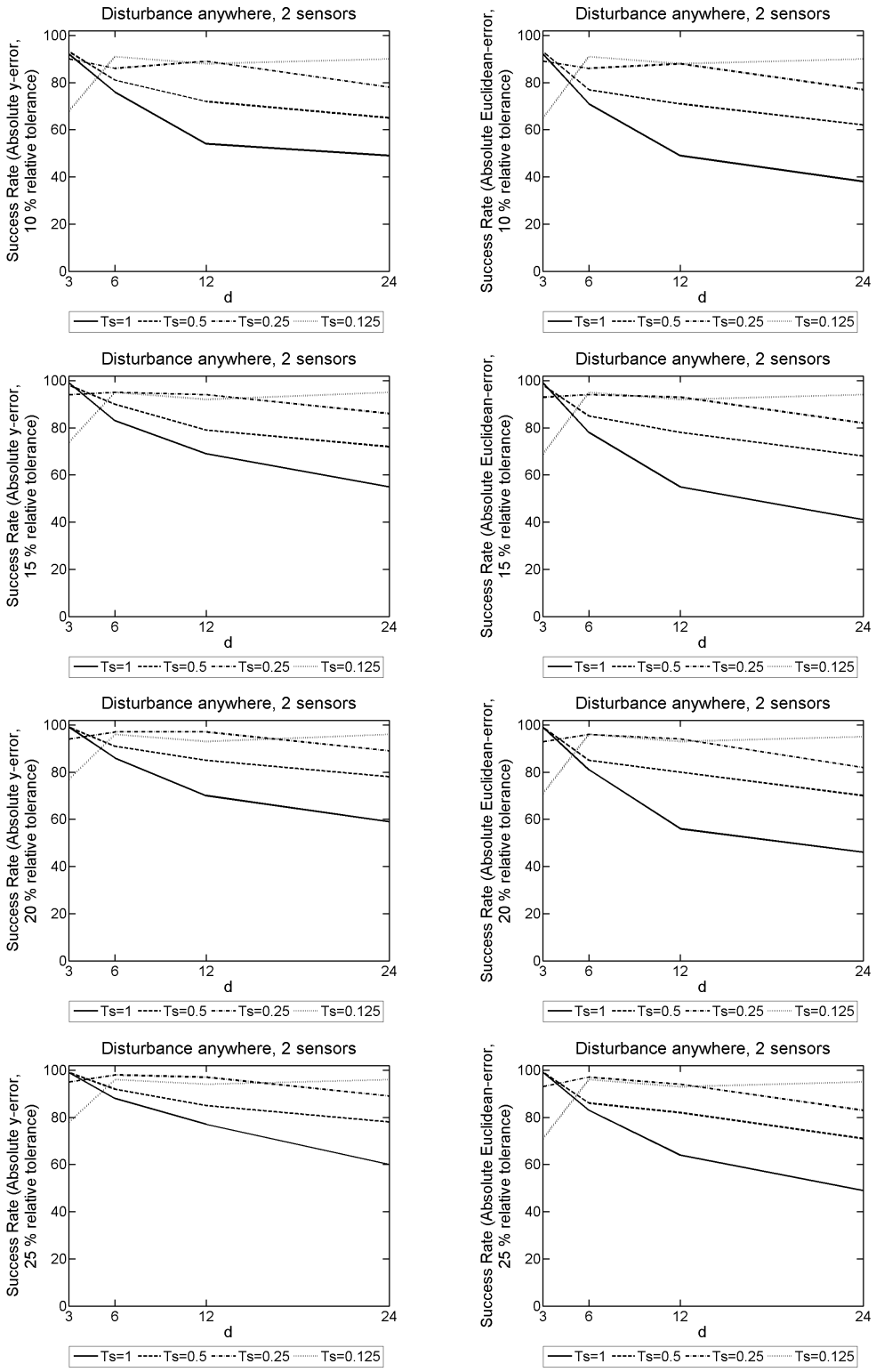


Figure 9.82: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 2 sensors present.

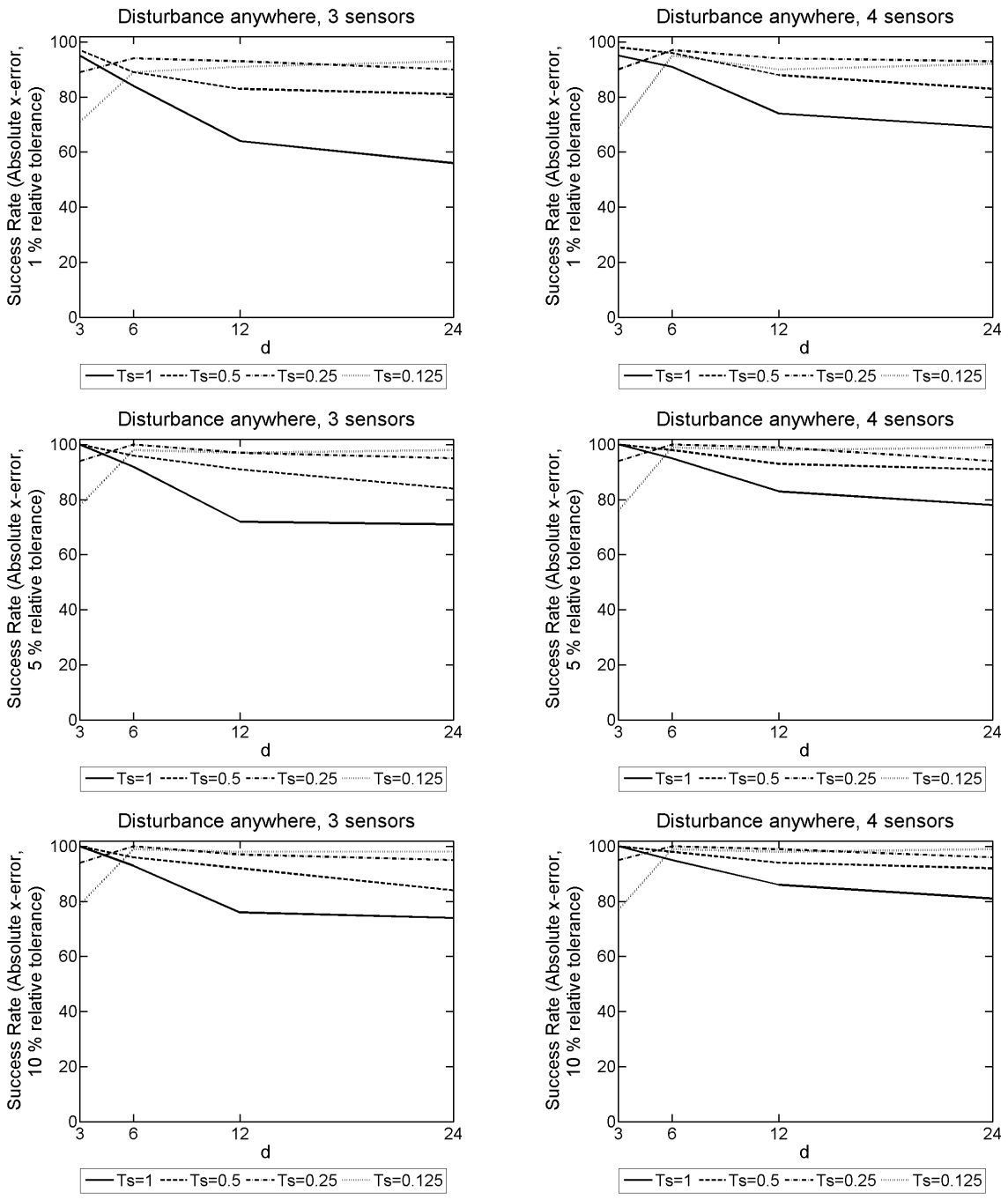


Figure 9.83: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

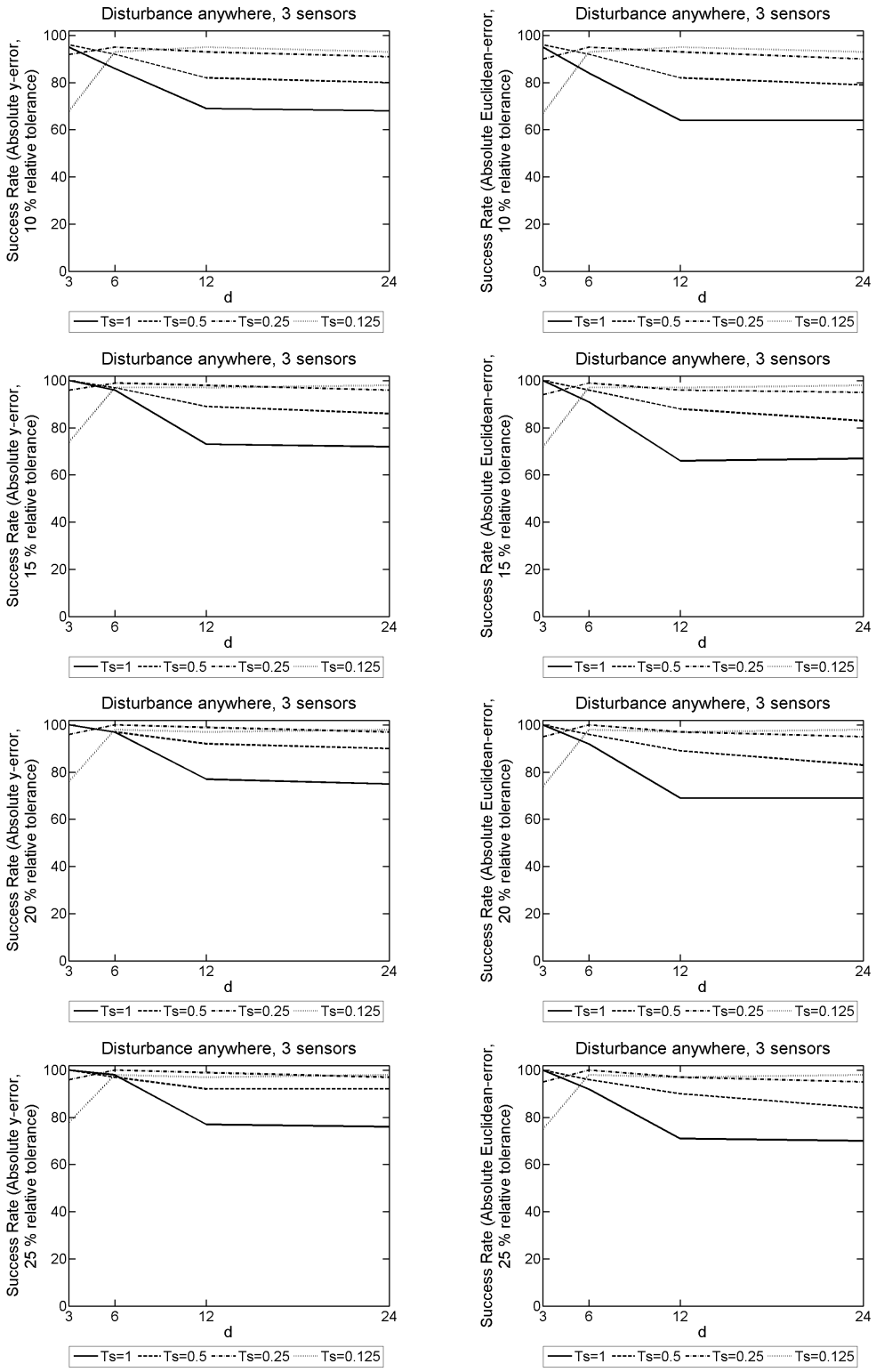


Figure 9.84: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 3 sensors present.

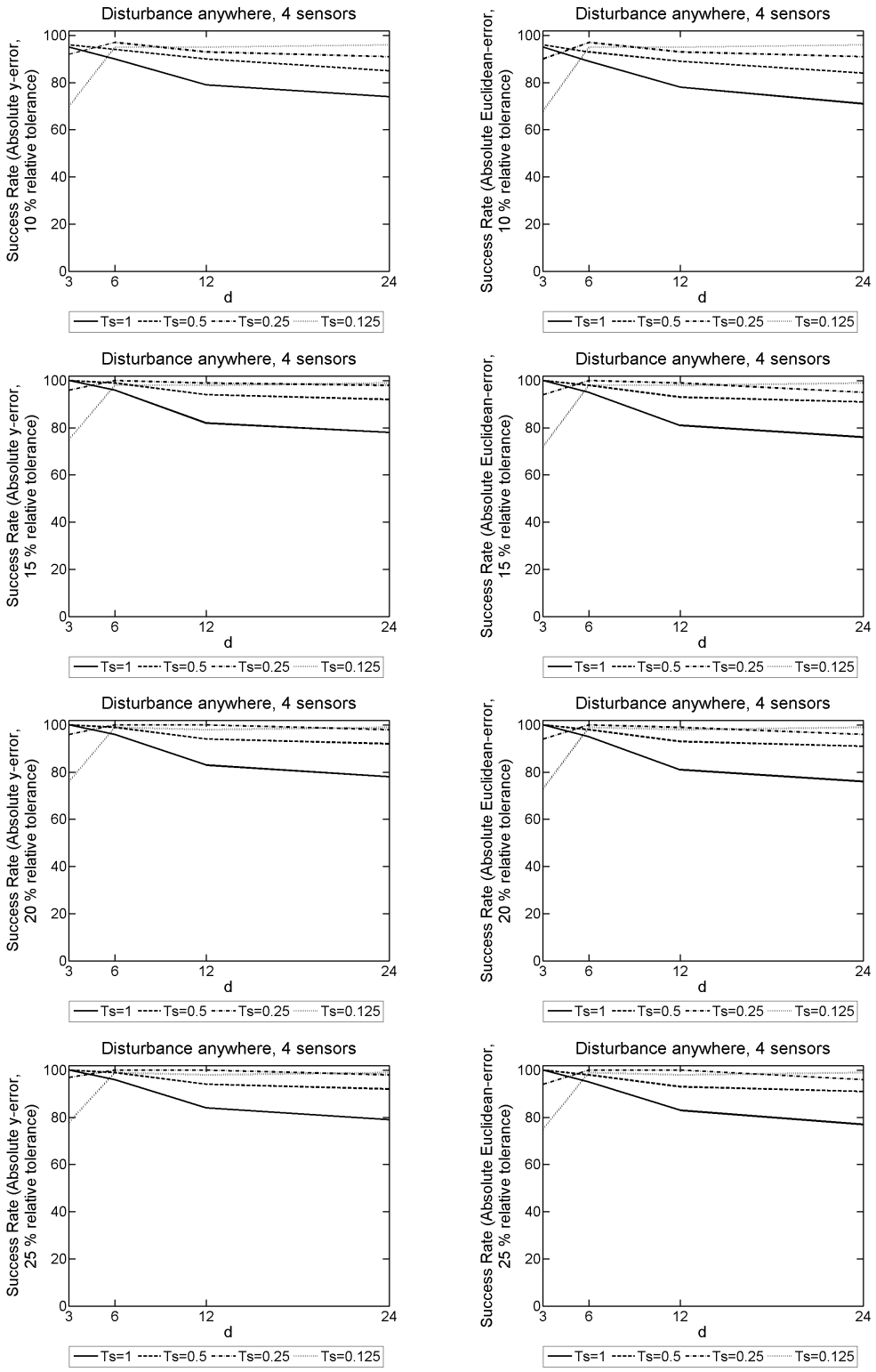


Figure 9.85: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 4 sensors present.

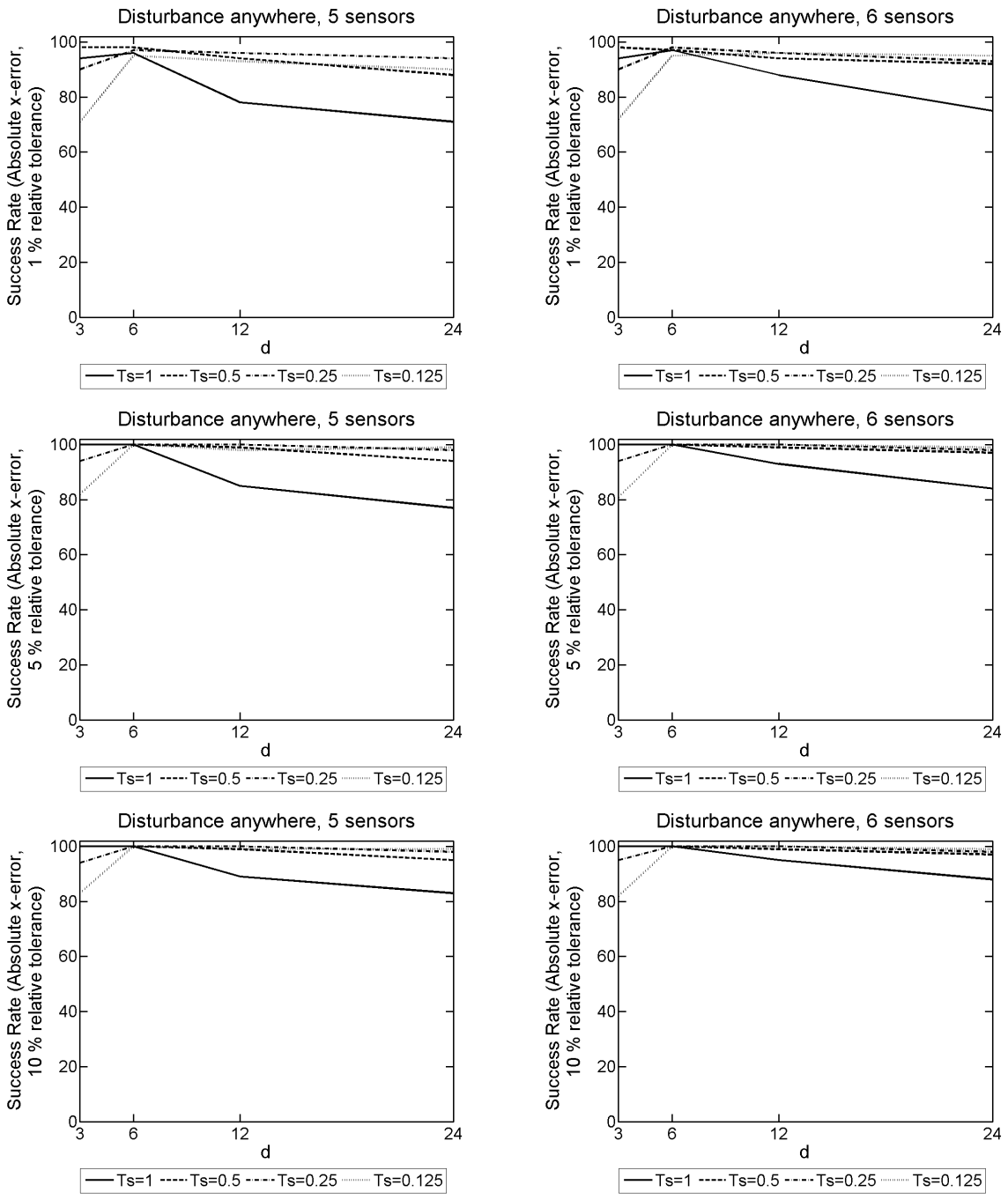


Figure 9.86: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

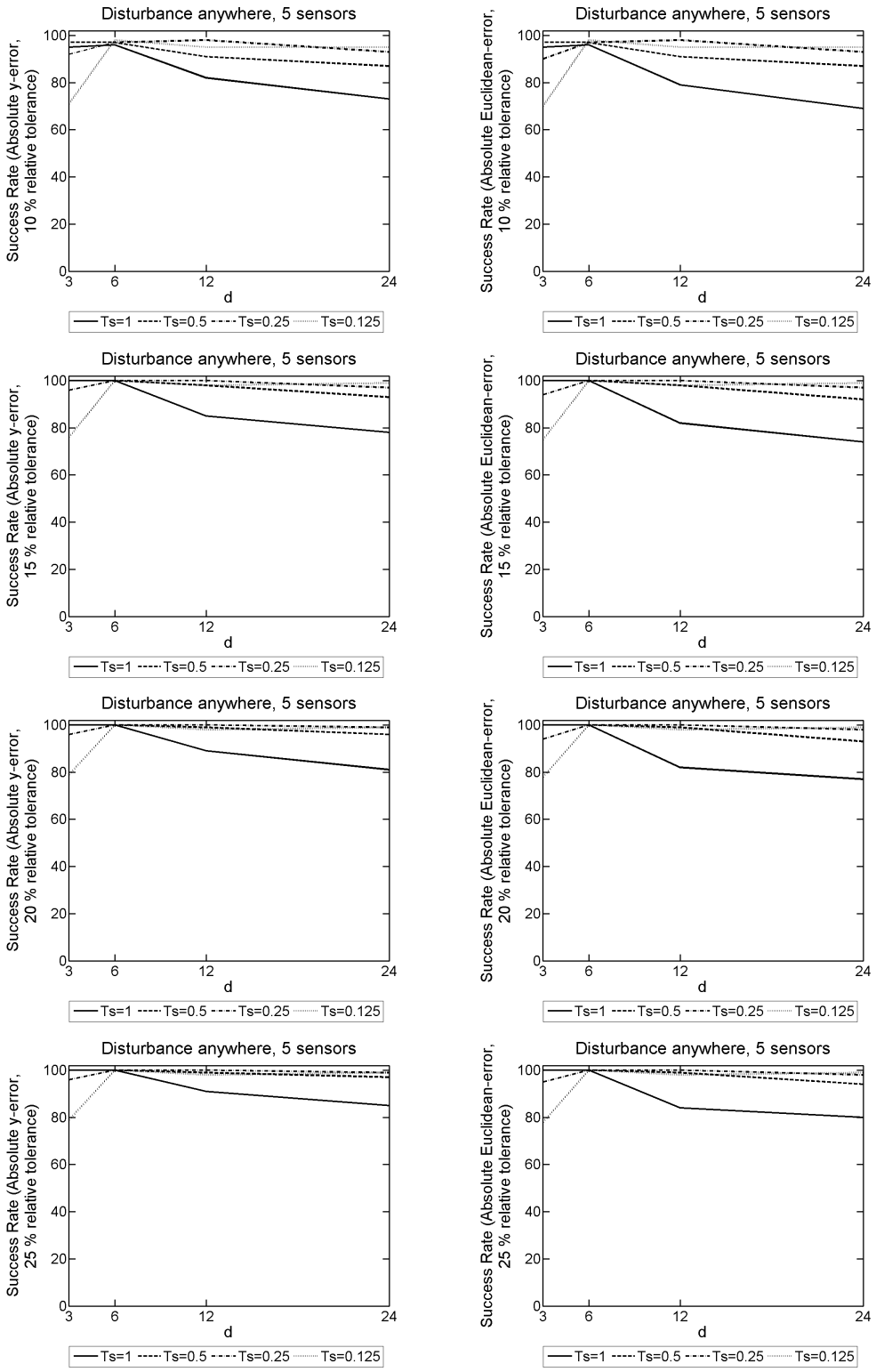


Figure 9.87: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 5 sensors present.

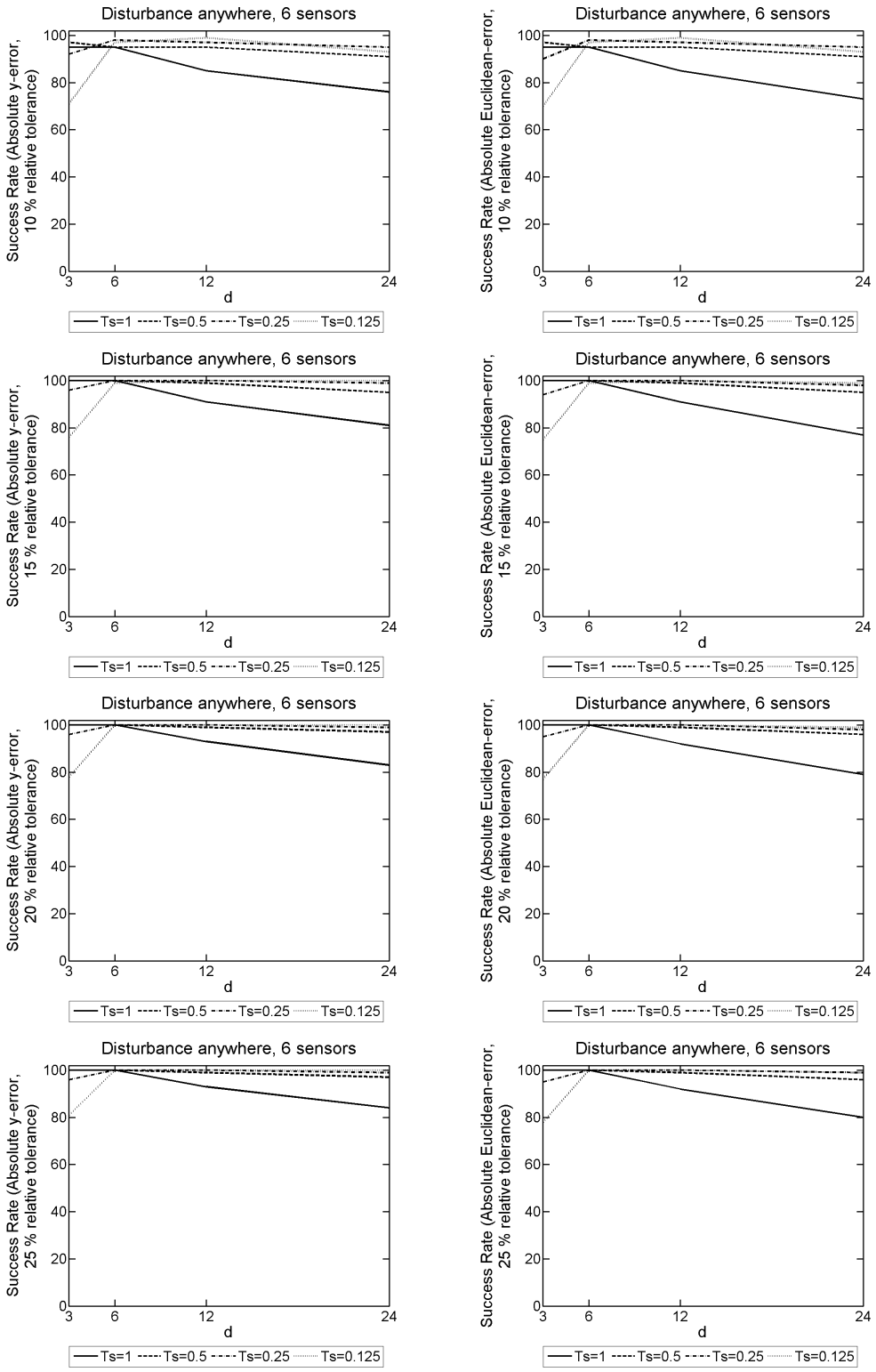


Figure 9.88: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 6 sensors present.

D.2 2D model problem with 3NBCs and a disturbance frequency $F = 150\mathbf{Hz}$

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	55	60	64	62	62	80	80	53	53	55	61
1.000	6	22	32	36	33	33	60	60	22	22	28	37
1.000	12	19	27	32	26	26	52	52	18	19	26	28
1.000	24	22	26	30	28	28	56	56	16	17	24	30
0.500	3	59	68	70	61	61	81	81	55	55	63	66
0.500	6	26	30	35	33	33	65	65	22	22	25	30
0.500	12	12	18	29	30	30	60	60	15	15	17	24
0.500	24	20	23	34	37	37	64	64	20	21	27	32
0.250	3	70	75	76	70	70	89	89	65	65	72	75
0.250	6	51	56	60	53	53	82	82	48	49	57	60
0.250	12	53	60	62	61	61	76	76	56	57	60	63
0.250	24	41	43	49	49	49	70	70	43	43	46	50
0.125	3	42	48	53	53	53	74	74	43	43	45	51
0.125	6	59	67	68	69	69	83	83	60	61	62	67
0.125	12	67	71	74	72	72	88	88	65	65	69	74
0.125	24	64	69	73	74	74	89	89	68	69	72	76

Table 9.101: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	93	94	94	93	93	97	97	92	92	93	94
1.000	6	48	57	58	56	56	80	80	48	49	53	59
1.000	12	20	33	36	35	35	60	60	25	30	34	39
1.000	24	24	36	46	38	38	63	63	27	29	33	37
0.500	3	94	96	96	94	94	98	98	93	93	95	96
0.500	6	60	63	63	65	65	84	84	57	57	64	65
0.500	12	45	52	57	50	50	70	70	44	46	49	52
0.500	24	44	47	58	51	51	68	68	45	45	49	52
0.250	3	94	99	100	95	95	99	99	95	95	97	99
0.250	6	83	90	91	86	86	95	95	83	83	88	90
0.250	12	75	77	79	75	75	88	88	74	74	75	77
0.250	24	69	71	75	77	77	88	88	70	71	74	76
0.125	3	74	75	77	75	75	87	87	71	71	75	77
0.125	6	84	86	86	82	82	95	95	77	77	84	85
0.125	12	87	91	92	88	88	96	96	86	87	90	94
0.125	24	95	95	95	97	97	97	97	95	95	96	97

Table 9.102: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	61	65	68	67	67	80	80	63	63	65	65
1.000	12	41	46	50	47	47	67	67	41	43	44	48
1.000	24	36	39	46	42	42	61	61	39	40	41	44
0.500	3	99	99	99	99	99	100	100	99	99	99	99
0.500	6	86	88	89	89	89	94	94	87	87	88	88
0.500	12	74	77	79	76	76	87	87	75	75	78	79
0.500	24	71	74	79	77	77	90	90	72	72	74	79
0.250	3	99	99	99	98	98	99	99	98	98	99	99
0.250	6	98	99	99	99	99	100	100	99	99	99	99
0.250	12	95	96	96	97	97	99	99	96	96	96	96
0.250	24	90	94	95	91	91	96	96	91	91	93	94
0.125	3	86	86	88	89	89	90	90	86	86	86	86
0.125	6	94	95	95	96	96	97	97	95	96	96	96
0.125	12	98	98	99	99	99	100	100	99	99	99	100
0.125	24	99	99	99	99	99	100	100	99	99	99	99

Table 9.103: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	86	89	90	90	90	94	94	87	88	89	89
1.000	12	50	54	56	65	65	75	75	54	55	56	57
1.000	24	56	58	62	62	62	79	79	57	57	57	61
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	98	98	98	98	98	100	100	98	98	98	98
0.500	12	88	89	89	88	88	92	92	87	87	88	88
0.500	24	95	96	96	97	97	98	98	96	96	97	97
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	99	99	99	99	99	99	99	99	99	99	99
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	97	98	98	98	98	99	99	98	98	98	98
0.125	3	88	90	92	91	91	95	95	88	88	89	91
0.125	6	95	98	98	96	96	99	99	96	96	97	97
0.125	12	100	100	100	100	100	100	100	100	100	100	100
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.104: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	99	100	100	100	100	100	100	100	100	100	100
1.000	12	87	90	91	89	89	93	93	87	88	90	90
1.000	24	78	83	86	82	82	88	88	81	82	83	86
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	99	99	99	99	99	99	99	99	99	99	99
0.500	12	99	99	99	99	99	100	100	99	99	99	99
0.500	24	97	99	99	100	100	100	100	99	99	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	97	97	97	96	96	99	99	96	96	97	97
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	93	93	94	94	94	97	97	93	93	94	94
0.125	6	85	90	91	83	83	94	94	83	83	87	89
0.125	12	98	99	99	100	100	100	100	99	99	99	99
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.105: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	100	100	100	100	100	100	100	100	100	100	100
1.000	6	100	100	100	100	100	100	100	100	100	100	100
1.000	12	97	97	97	98	98	99	99	96	96	97	98
1.000	24	89	91	92	91	91	95	95	90	90	90	91
0.500	3	100	100	100	100	100	100	100	100	100	100	100
0.500	6	98	98	98	97	97	99	99	97	97	98	98
0.500	12	99	99	99	99	99	100	100	99	99	99	99
0.500	24	100	100	100	100	100	100	100	100	100	100	100
0.250	3	100	100	100	100	100	100	100	100	100	100	100
0.250	6	80	81	83	83	83	92	92	79	80	81	84
0.250	12	100	100	100	100	100	100	100	100	100	100	100
0.250	24	100	100	100	100	100	100	100	100	100	100	100
0.125	3	97	97	97	100	100	100	100	97	98	98	98
0.125	6	72	74	78	78	78	86	86	71	72	78	79
0.125	12	99	99	99	99	99	100	100	99	99	99	99
0.125	24	100	100	100	100	100	100	100	100	100	100	100

Table 9.106: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	61	72	73	58	72	77	79	55	69	71	72
1.000	6	26	36	41	36	44	51	58	26	31	35	37
1.000	12	5	13	19	18	25	34	46	4	7	12	16
1.000	24	9	17	24	25	37	44	52	11	13	21	23
0.500	3	47	66	68	60	72	75	76	52	63	65	67
0.500	6	18	26	34	21	34	45	53	15	21	27	32
0.500	12	22	37	46	28	34	45	52	20	26	31	38
0.500	24	17	27	38	31	42	49	51	12	19	27	34
0.250	3	57	70	71	58	68	75	78	51	58	65	68
0.250	6	48	60	65	58	65	71	77	48	57	60	65
0.250	12	49	56	64	62	70	75	77	49	58	62	64
0.250	24	39	51	52	47	62	69	71	37	44	48	51
0.125	3	18	32	41	34	47	54	59	20	28	32	36
0.125	6	40	59	64	51	64	70	73	42	55	58	62
0.125	12	63	78	79	68	81	85	87	61	73	78	80
0.125	24	61	80	83	68	80	82	83	63	76	78	78

Table 9.107: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	91	97	98	87	94	97	97	87	94	97	97
1.000	6	37	51	57	39	53	61	68	30	42	47	50
1.000	12	12	26	36	30	37	38	45	14	21	22	24
1.000	24	15	28	41	27	35	37	43	17	24	29	29
0.500	3	88	98	98	93	99	100	100	93	97	98	98
0.500	6	52	61	61	55	63	66	70	47	53	55	58
0.500	12	32	40	49	40	50	57	60	28	33	38	42
0.500	24	33	43	44	38	47	54	60	29	34	38	39
0.250	3	85	93	93	86	93	94	95	84	89	90	92
0.250	6	79	87	88	73	84	88	90	70	81	85	86
0.250	12	64	72	72	71	83	86	88	62	71	74	76
0.250	24	52	59	61	57	66	71	74	47	56	57	62
0.125	3	38	47	53	45	55	60	65	37	45	47	48
0.125	6	71	82	85	73	85	88	90	67	80	82	83
0.125	12	85	91	91	81	93	94	94	79	90	91	92
0.125	24	86	93	94	80	94	95	95	79	92	93	94

Table 9.108: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	96	100	100	96	100	100	100	96	100	100	100
1.000	6	52	69	73	55	67	71	71	50	60	64	65
1.000	12	29	45	55	42	44	48	51	31	35	36	37
1.000	24	26	44	50	38	49	56	63	29	34	37	42
0.500	3	97	100	100	96	100	100	100	96	100	100	100
0.500	6	65	73	74	72	78	78	79	65	71	72	74
0.500	12	57	63	67	53	67	71	73	44	57	60	61
0.500	24	59	71	73	65	72	80	81	60	66	71	73
0.250	3	94	98	98	94	99	99	99	94	98	98	98
0.250	6	90	98	98	87	97	98	98	87	97	98	98
0.250	12	82	93	95	83	94	94	95	81	93	93	93
0.250	24	75	85	86	81	89	89	91	77	84	84	84
0.125	3	60	65	66	61	68	69	73	59	65	65	65
0.125	6	86	94	95	81	94	96	96	81	92	94	95
0.125	12	88	95	95	91	95	96	97	90	94	95	95
0.125	24	86	95	96	88	95	95	95	88	95	95	95

Table 9.109: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	98	100	100	97	100	100	100	97	100	100	100
1.000	6	84	89	89	85	92	93	93	82	88	88	88
1.000	12	52	62	68	59	66	69	72	50	56	60	63
1.000	24	36	46	55	45	53	56	61	36	42	45	48
0.500	3	98	100	100	96	100	100	100	96	100	100	100
0.500	6	88	94	95	85	94	95	95	84	91	91	91
0.500	12	79	85	88	76	87	89	89	74	84	85	85
0.500	24	75	83	85	73	84	87	88	73	81	84	84
0.250	3	93	97	97	91	99	99	99	91	97	97	97
0.250	6	97	100	100	95	99	99	99	95	99	99	99
0.250	12	93	100	100	94	99	99	99	94	99	99	99
0.250	24	87	96	96	91	100	100	100	88	96	97	97
0.125	3	62	68	70	62	69	71	73	60	65	66	66
0.125	6	93	97	97	87	96	97	97	86	95	96	96
0.125	12	93	99	99	93	100	100	100	93	99	99	99
0.125	24	91	96	97	90	98	98	99	90	96	96	96

Table 9.110: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	98	100	100	97	100	100	100	97	100	100	100
1.000	6	94	99	99	88	96	98	98	88	96	98	98
1.000	12	66	73	77	68	77	78	80	61	70	72	72
1.000	24	48	59	67	53	59	65	70	46	53	57	60
0.500	3	98	100	100	95	100	100	100	95	100	100	100
0.500	6	96	99	99	92	99	99	99	92	99	99	99
0.500	12	92	98	98	90	100	100	100	88	98	98	98
0.500	24	86	92	94	86	95	96	96	85	90	91	92
0.250	3	93	97	97	92	99	99	99	92	97	97	97
0.250	6	91	99	99	89	99	99	99	89	99	99	99
0.250	12	91	100	100	93	100	100	100	93	100	100	100
0.250	24	93	99	99	87	99	99	99	87	99	99	99
0.125	3	61	65	66	63	69	70	73	59	63	65	65
0.125	6	80	94	94	82	93	94	94	81	92	93	93
0.125	12	89	98	98	89	97	98	98	89	97	98	98
0.125	24	93	98	98	87	99	99	100	86	98	98	98

Table 9.111: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
1.000	3	98	100	100	97	100	100	100	97	100	100	100
1.000	6	98	100	100	96	99	100	100	96	99	100	100
1.000	12	85	90	92	85	94	94	94	81	89	89	90
1.000	24	73	80	84	74	80	83	86	71	76	79	81
0.500	3	98	100	100	96	100	100	100	96	100	100	100
0.500	6	93	96	96	90	96	96	96	88	94	94	95
0.500	12	91	98	98	90	98	99	99	90	97	98	98
0.500	24	92	96	97	91	98	99	99	90	96	97	97
0.250	3	94	98	98	93	99	99	99	93	98	98	98
0.250	6	77	87	88	78	91	94	94	76	86	89	90
0.250	12	97	100	100	90	100	100	100	90	100	100	100
0.250	24	86	96	96	90	99	100	100	87	95	96	96
0.125	3	66	69	71	67	75	77	79	61	68	71	72
0.125	6	54	71	75	66	78	81	84	58	68	71	74
0.125	12	82	98	98	83	96	98	98	83	96	98	98
0.125	24	93	99	99	86	99	99	99	86	99	99	99

Table 9.112: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

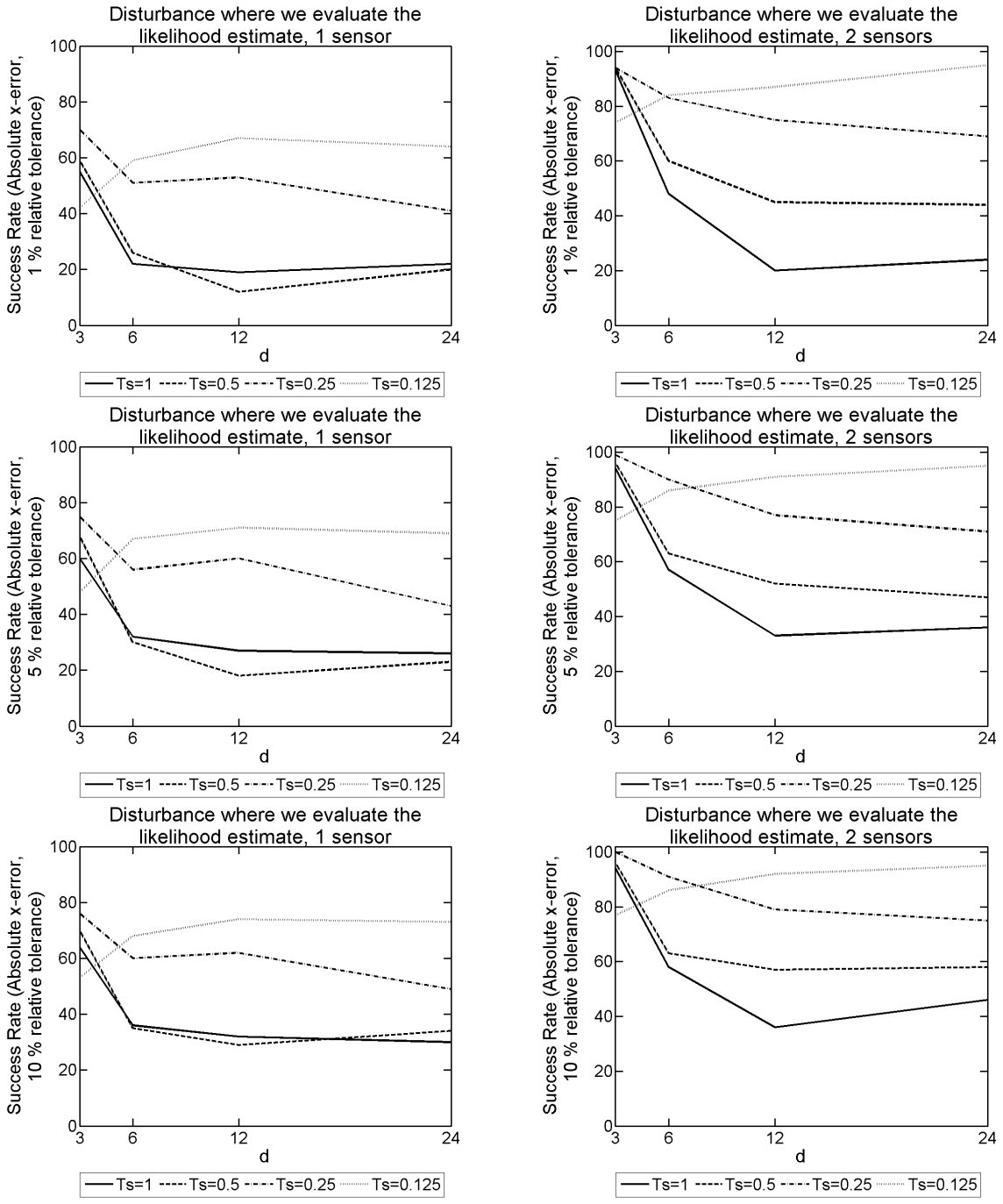


Figure 9.89: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

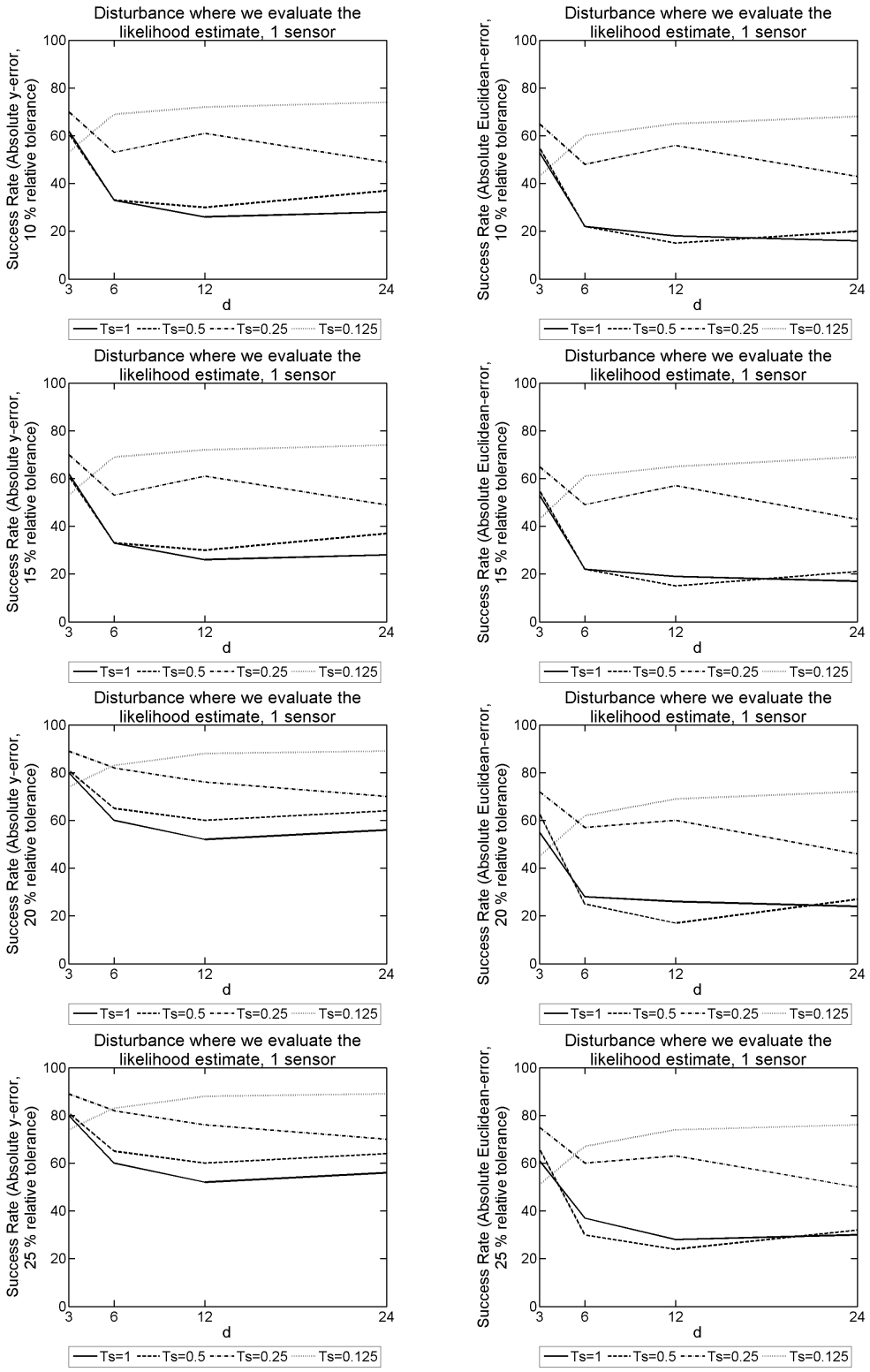


Figure 9.90: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 1 sensor present.

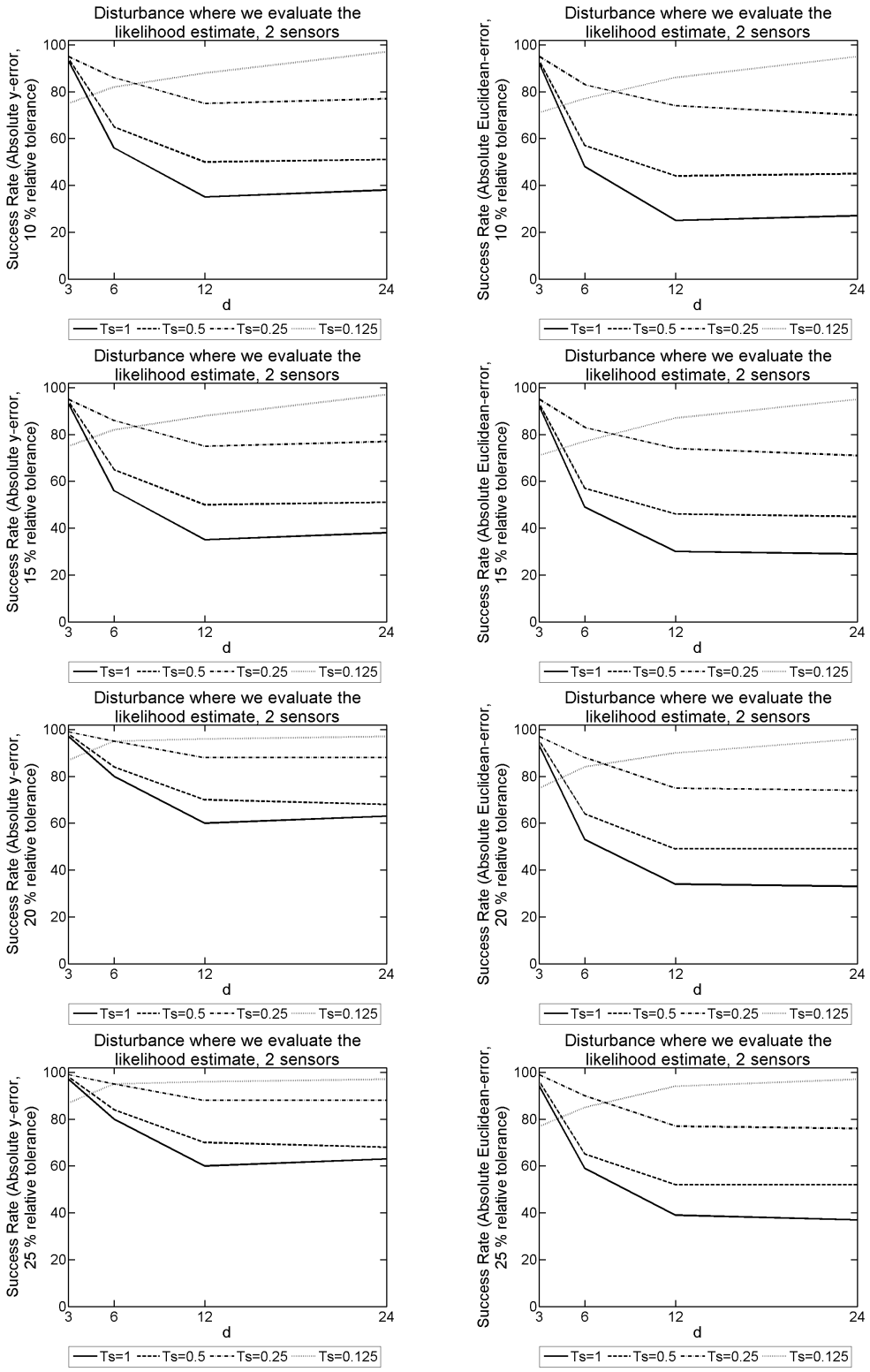


Figure 9.91: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 2 sensors present.

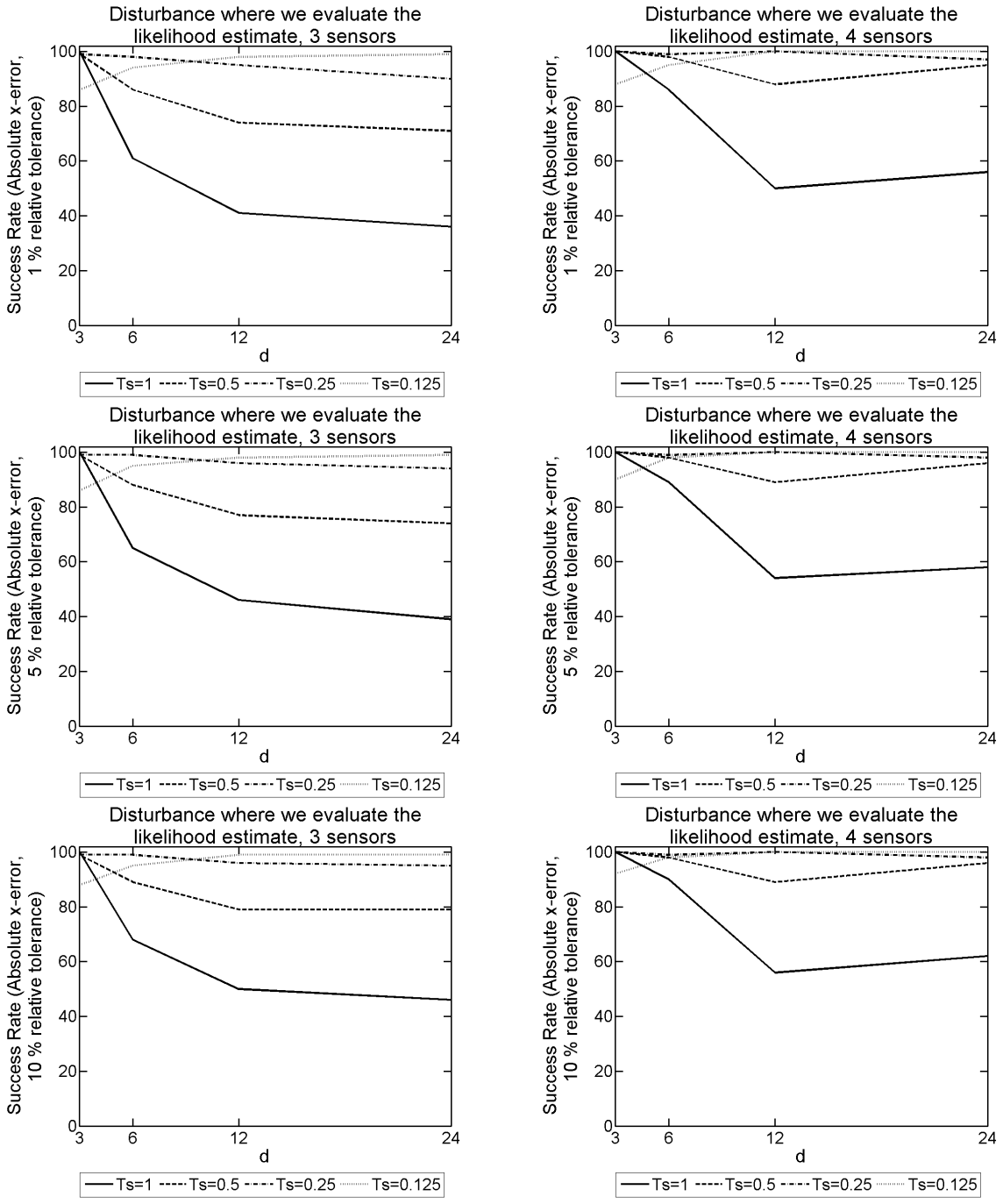


Figure 9.92: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

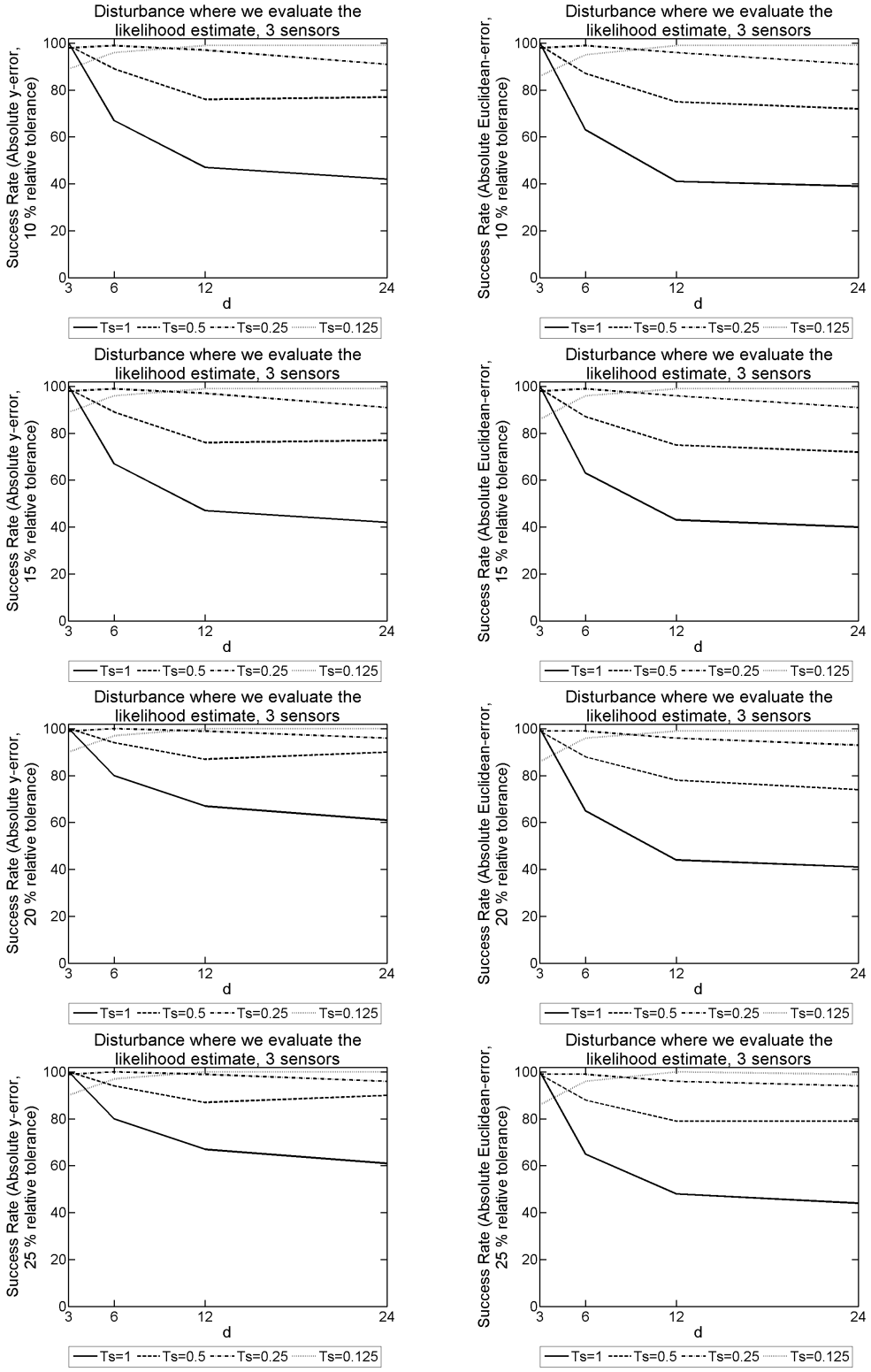


Figure 9.93: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 3 sensors present.

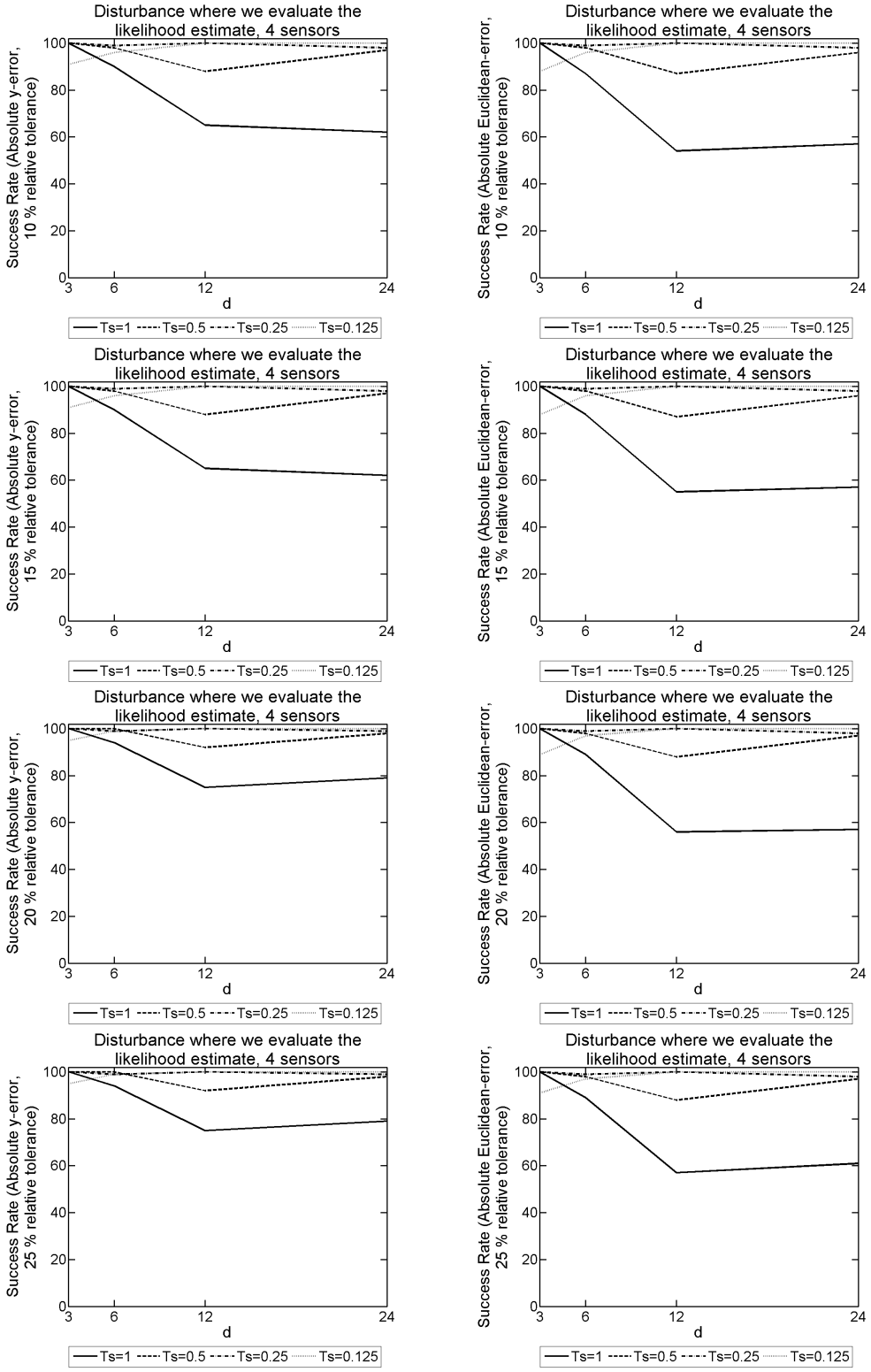


Figure 9.94: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 4 sensors present.

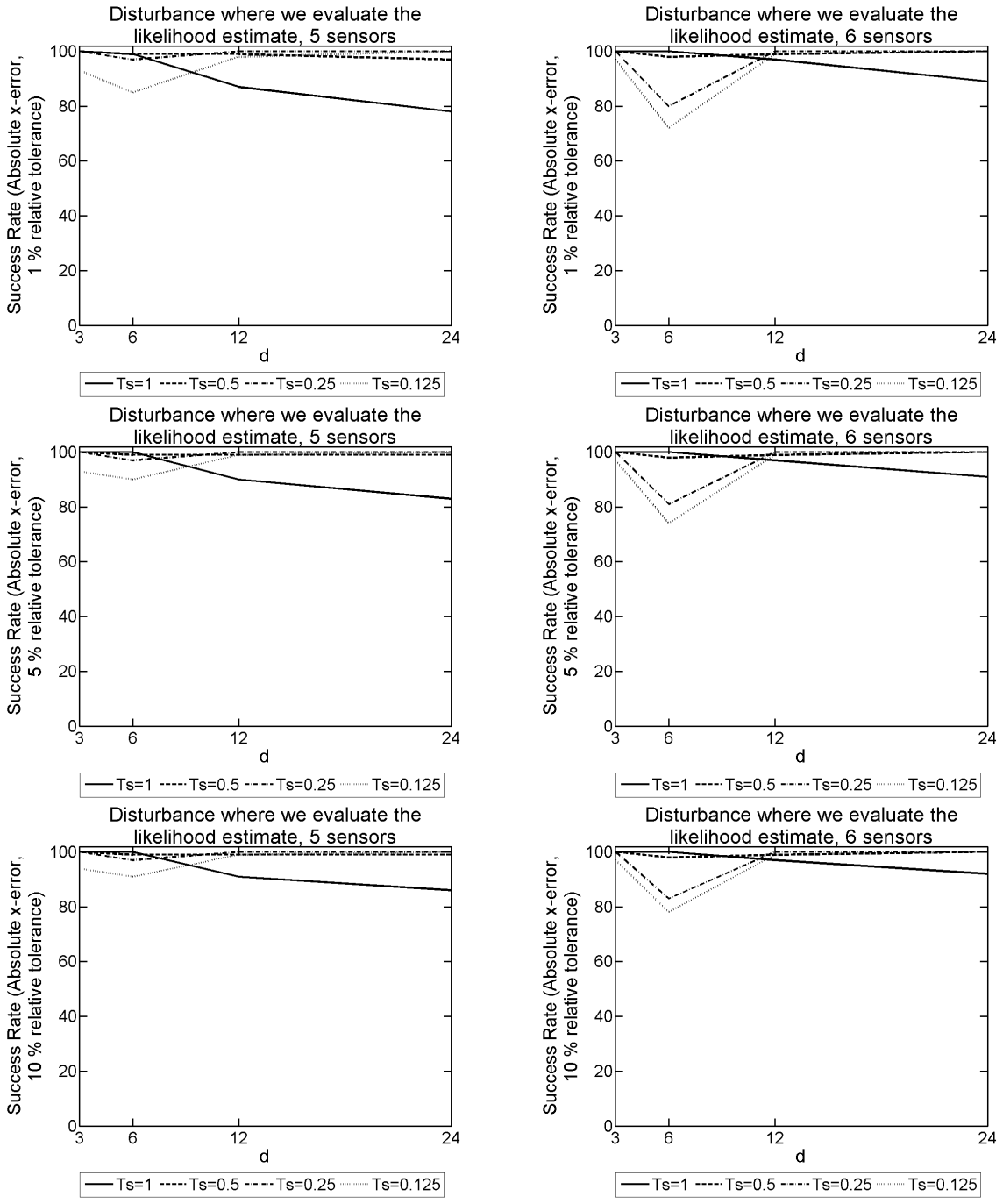


Figure 9.95: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

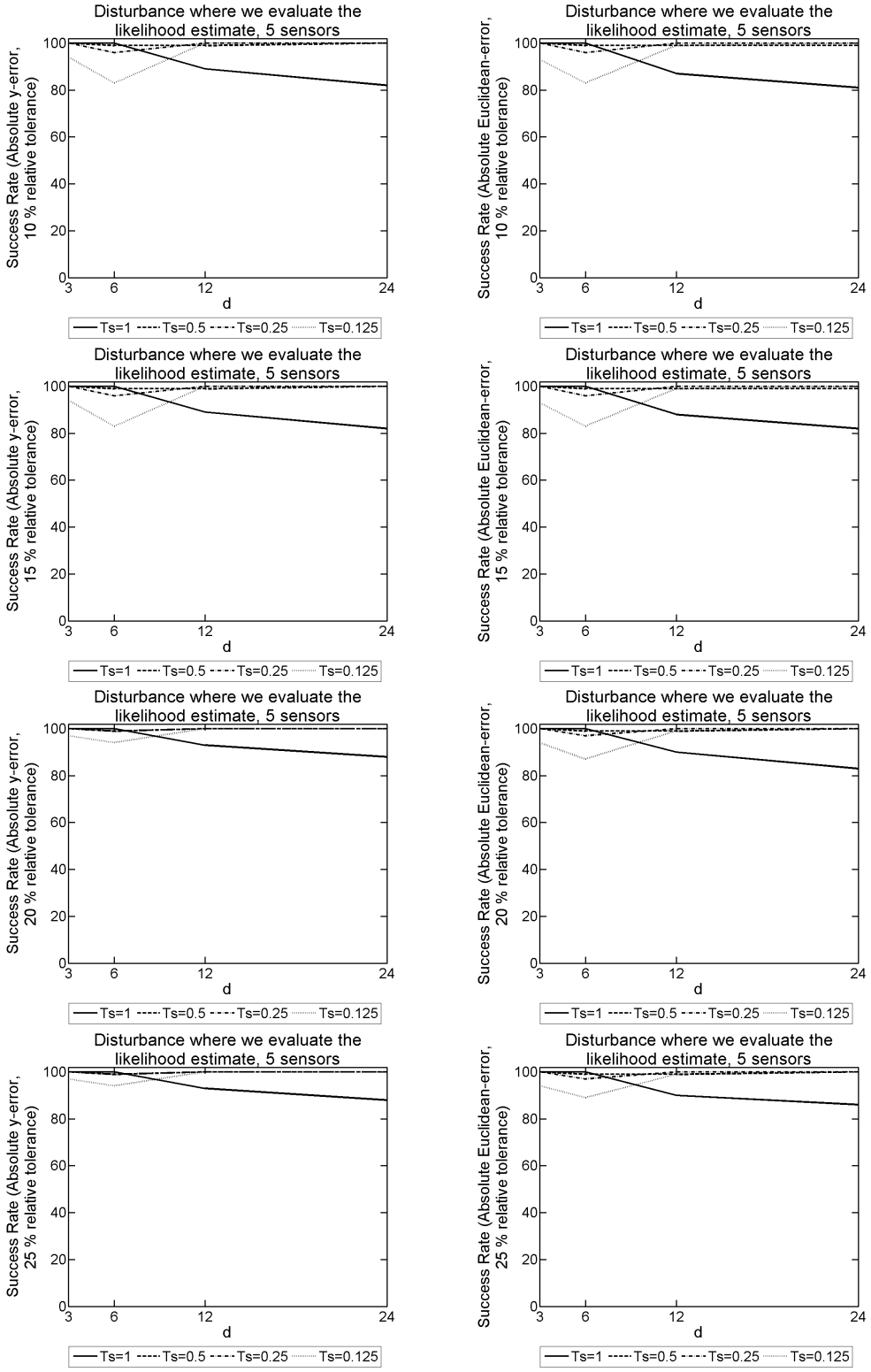


Figure 9.96: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 5 sensors present.

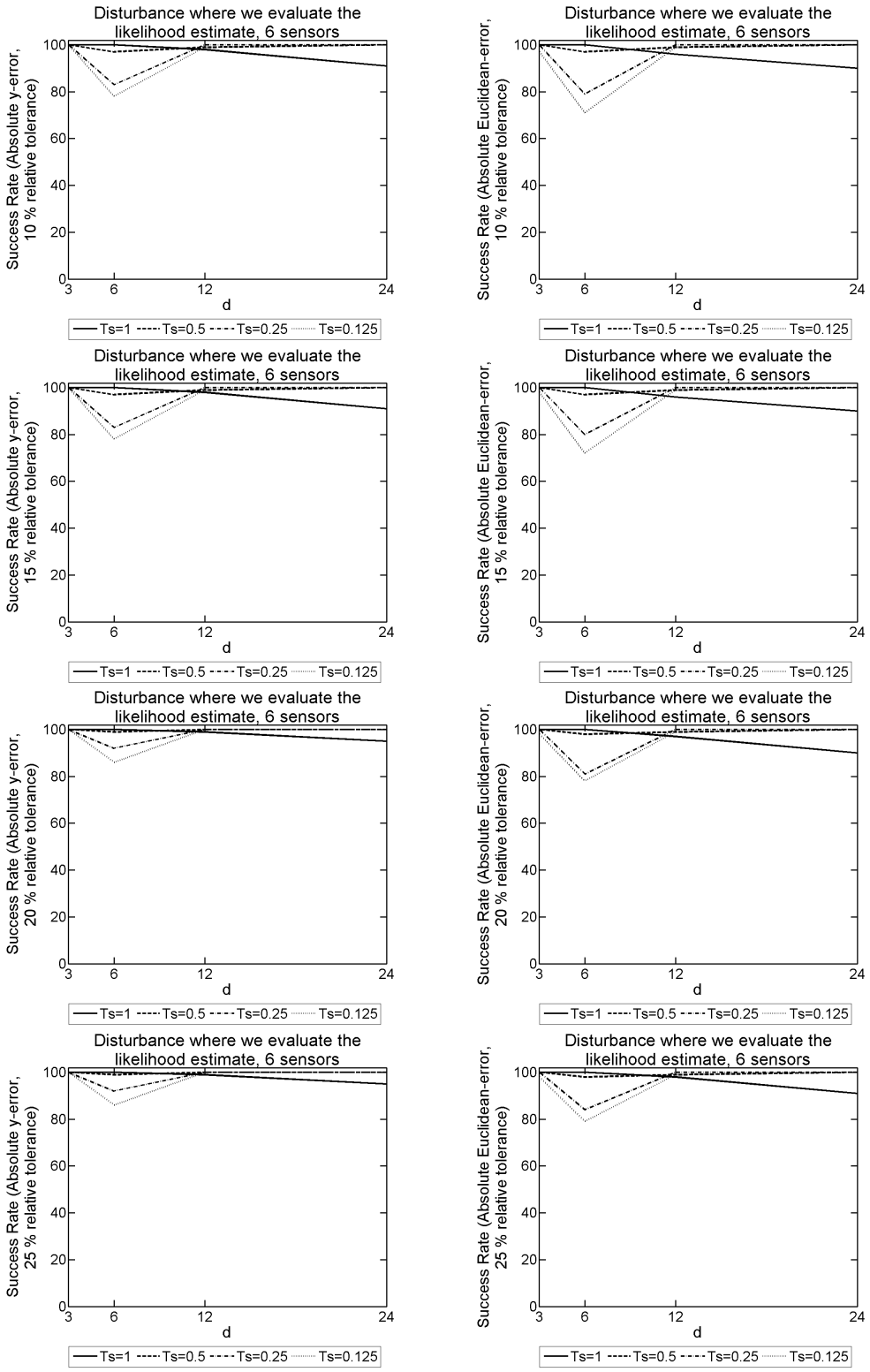


Figure 9.97: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 6 sensors present.

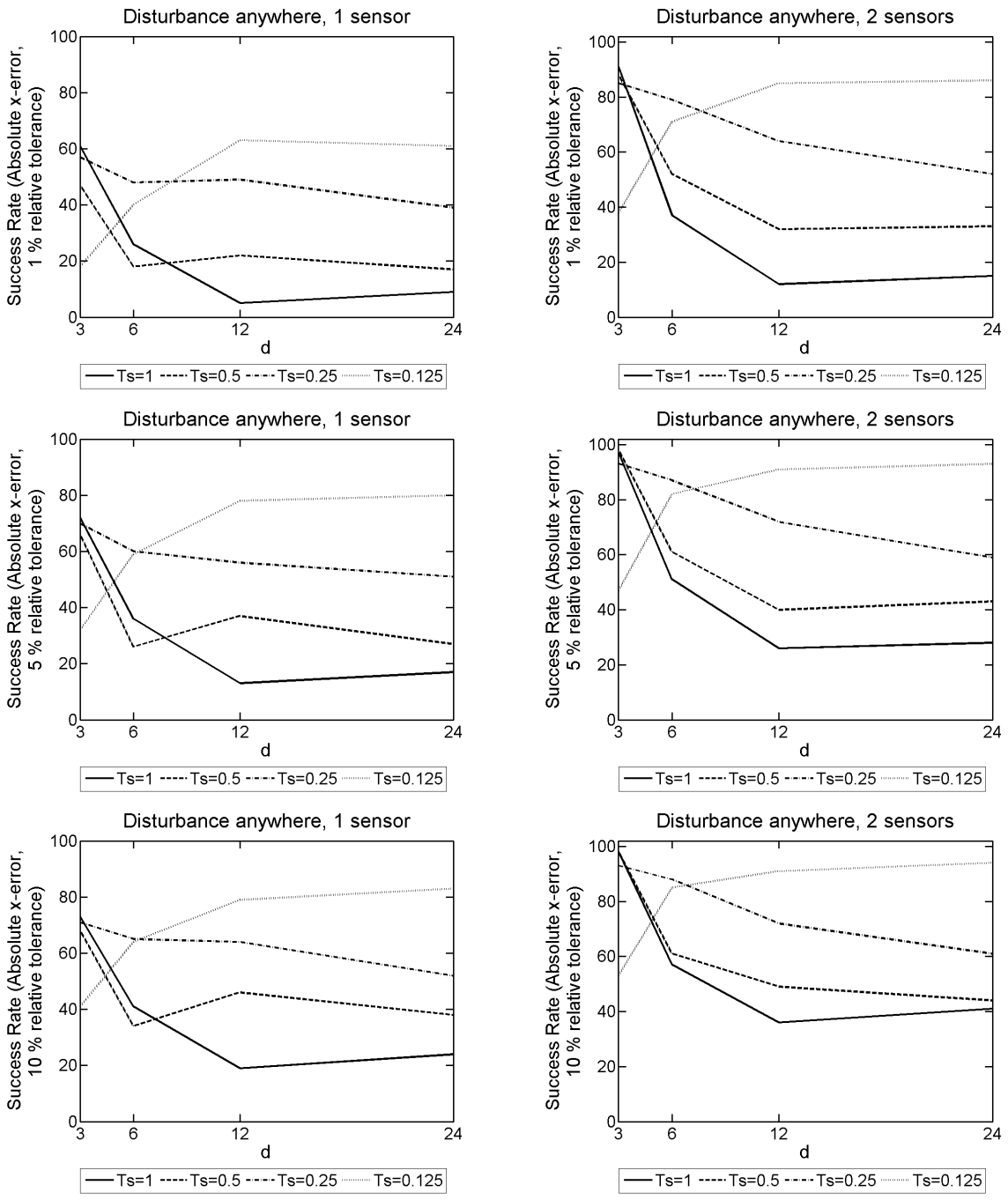


Figure 9.98: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

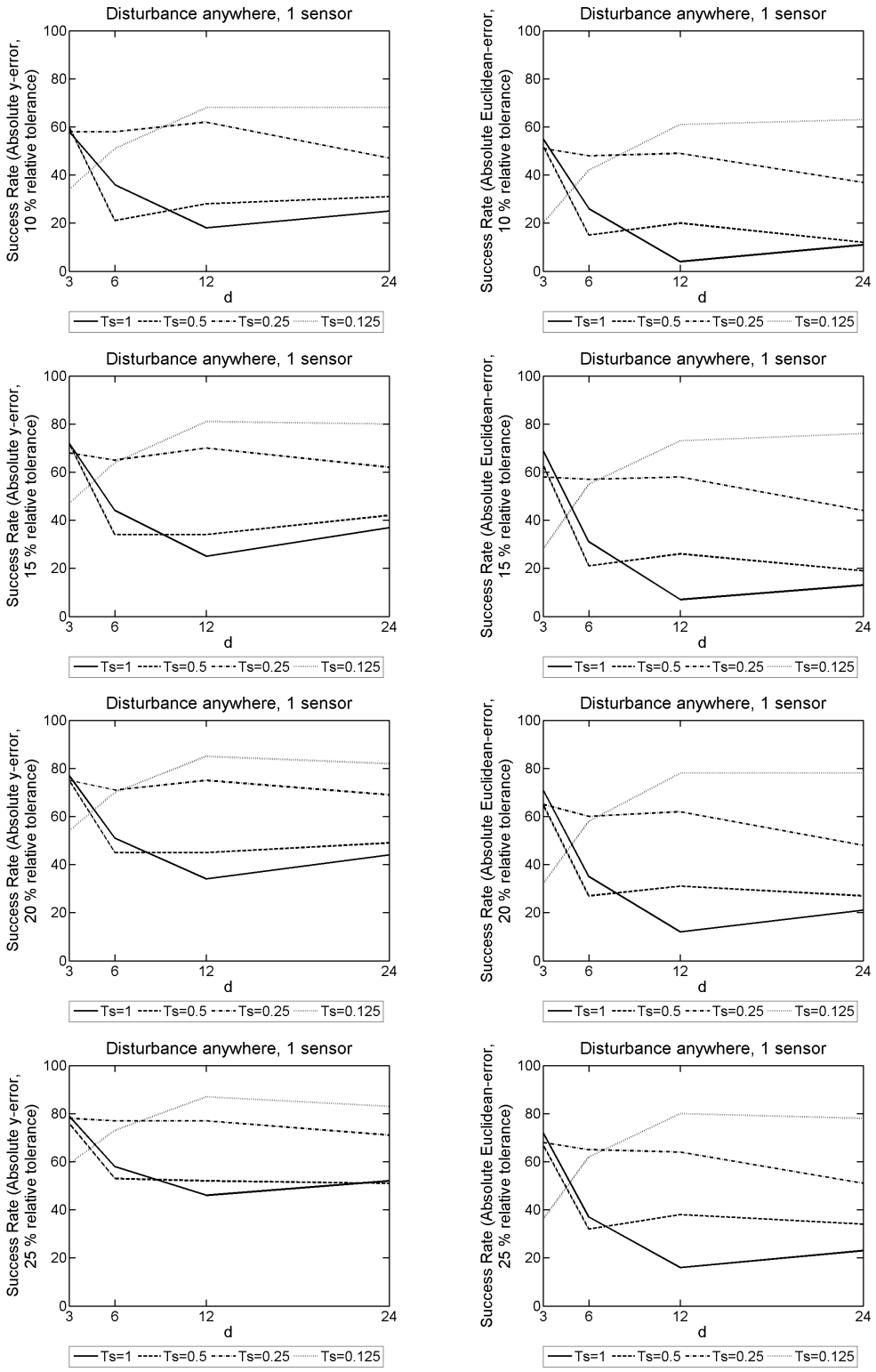


Figure 9.99: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 1 sensor present.

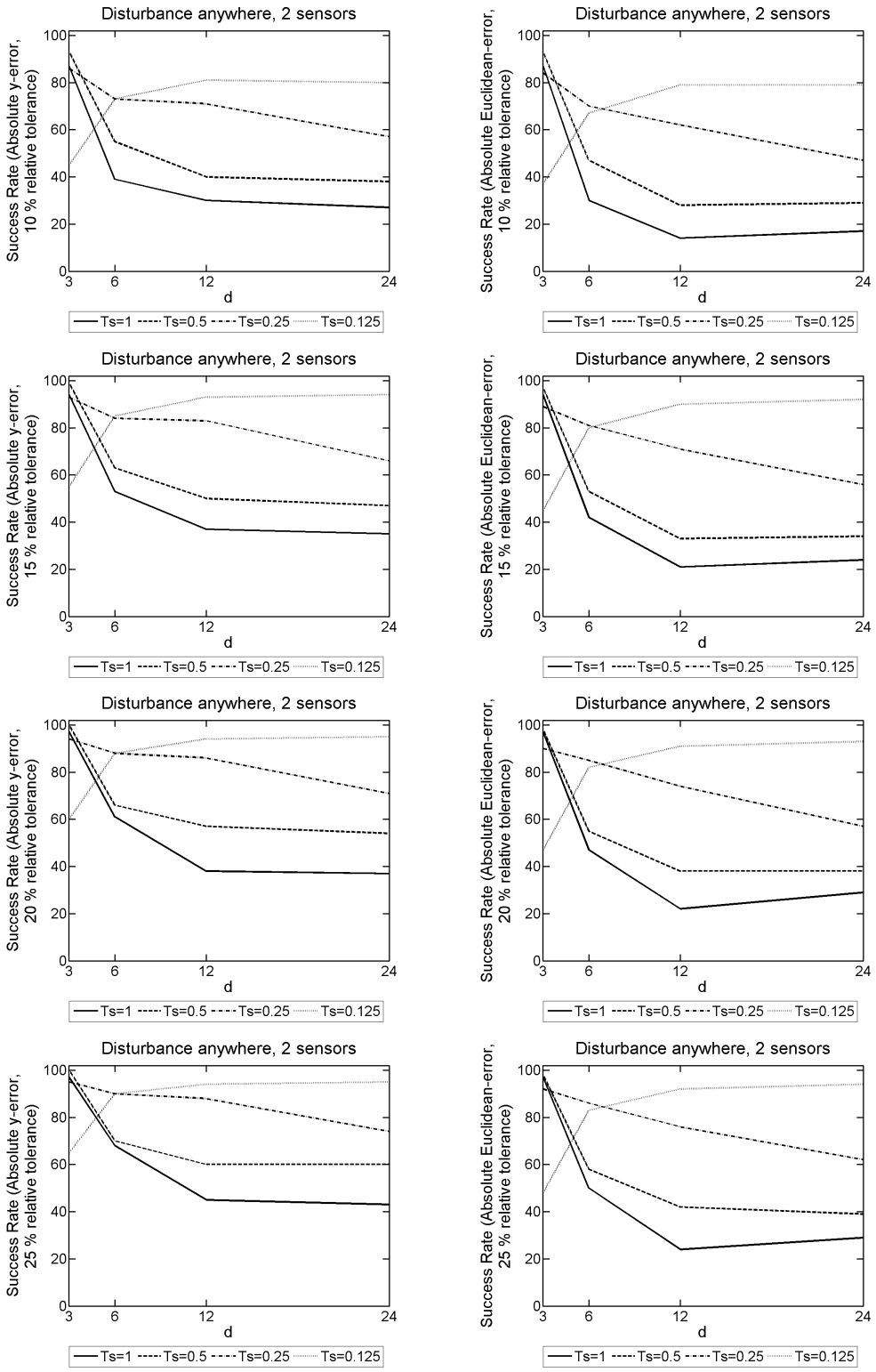


Figure 9.100: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 2 sensors present.

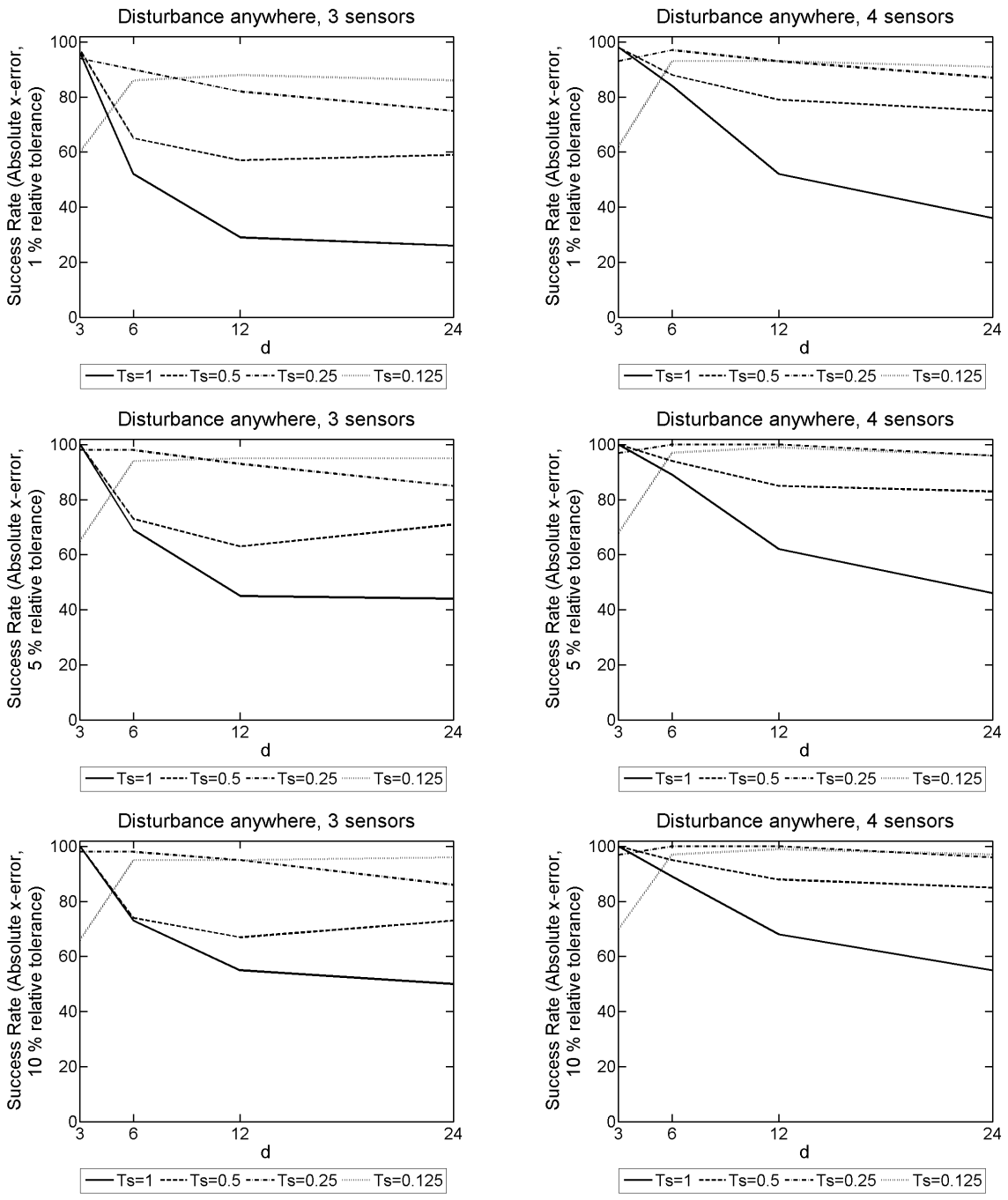


Figure 9.101: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

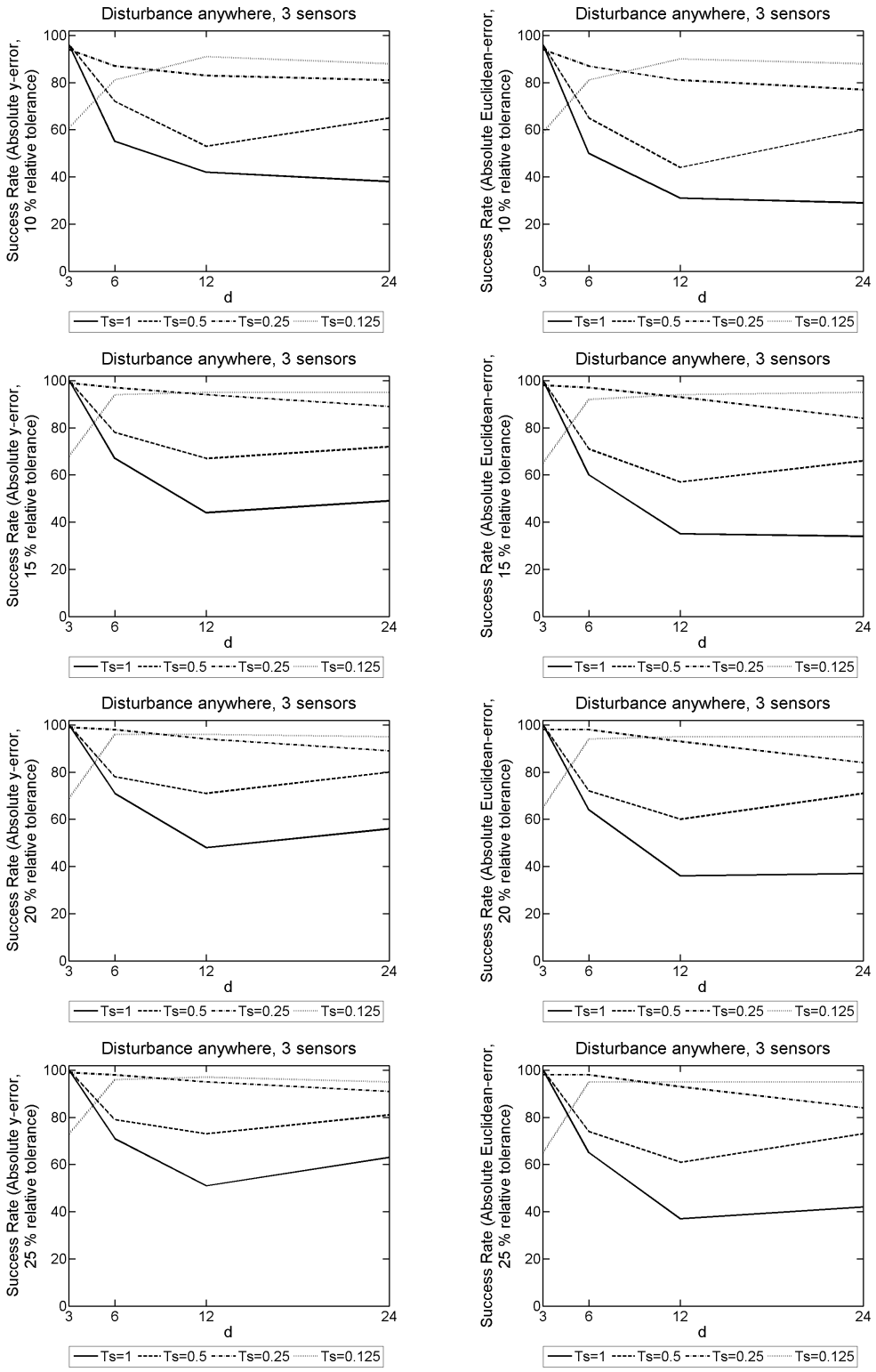


Figure 9.102: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 3 sensors present.

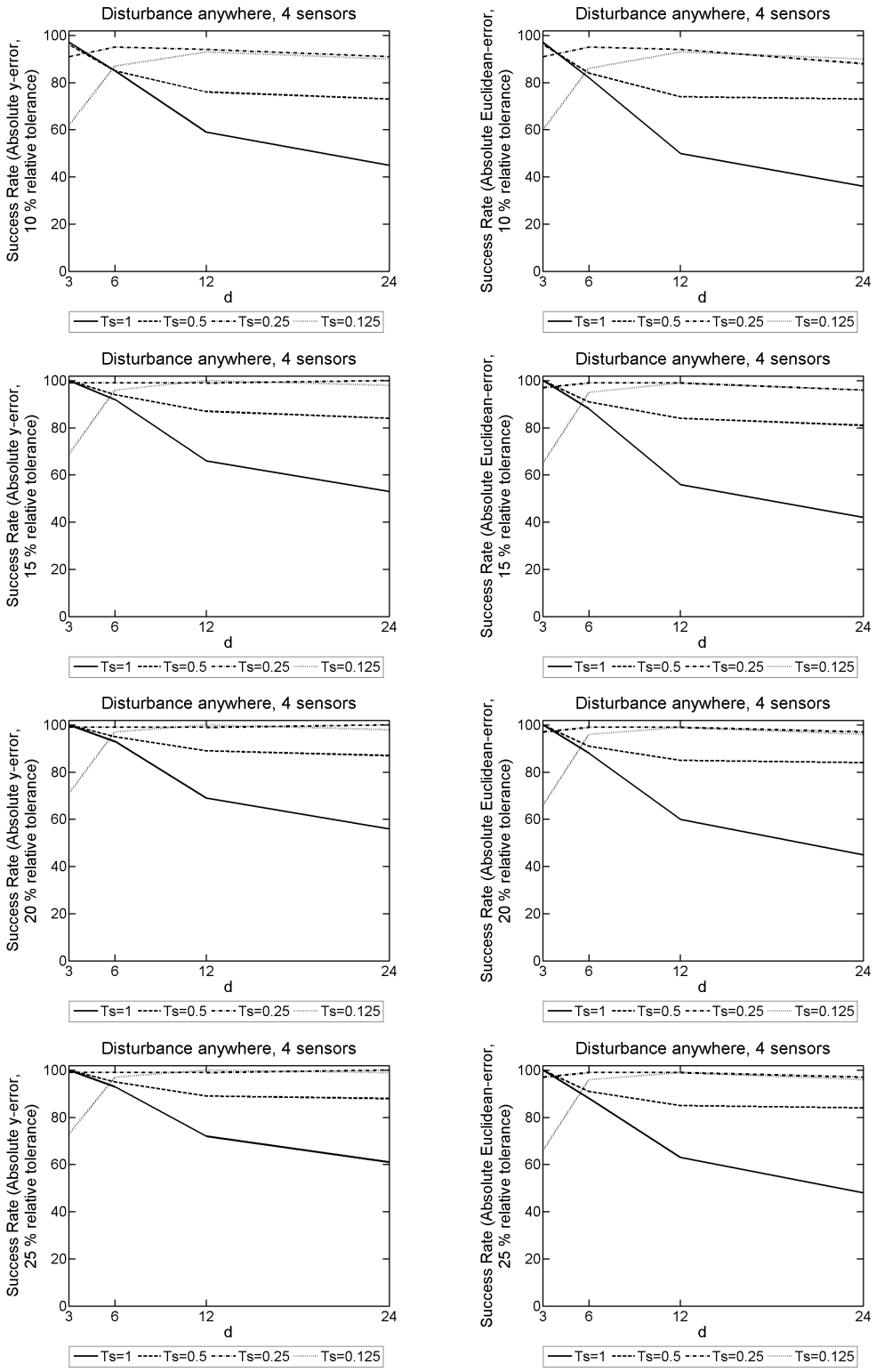


Figure 9.103: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 4 sensors present.

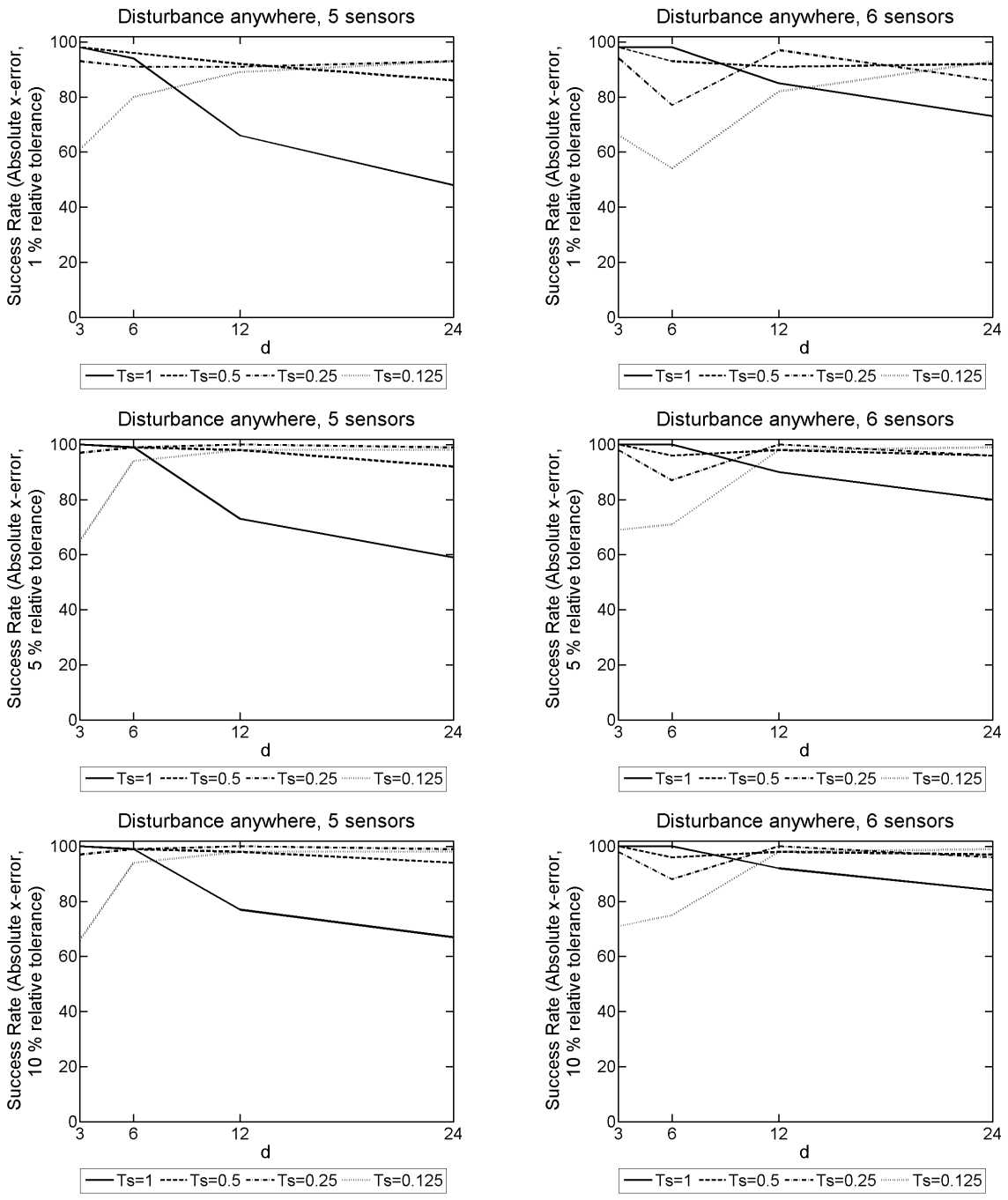


Figure 9.104: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

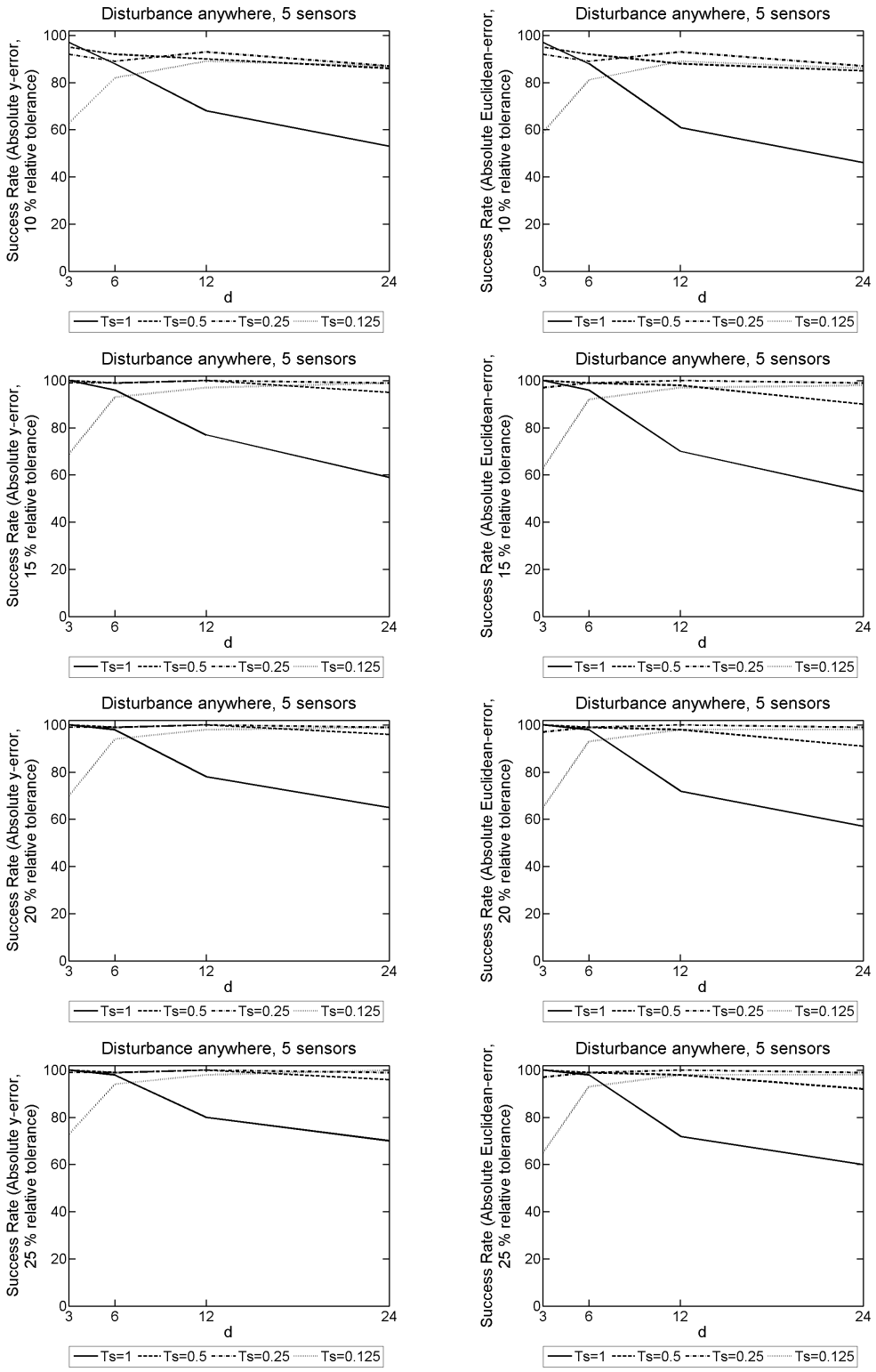


Figure 9.105: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 5 sensors present.

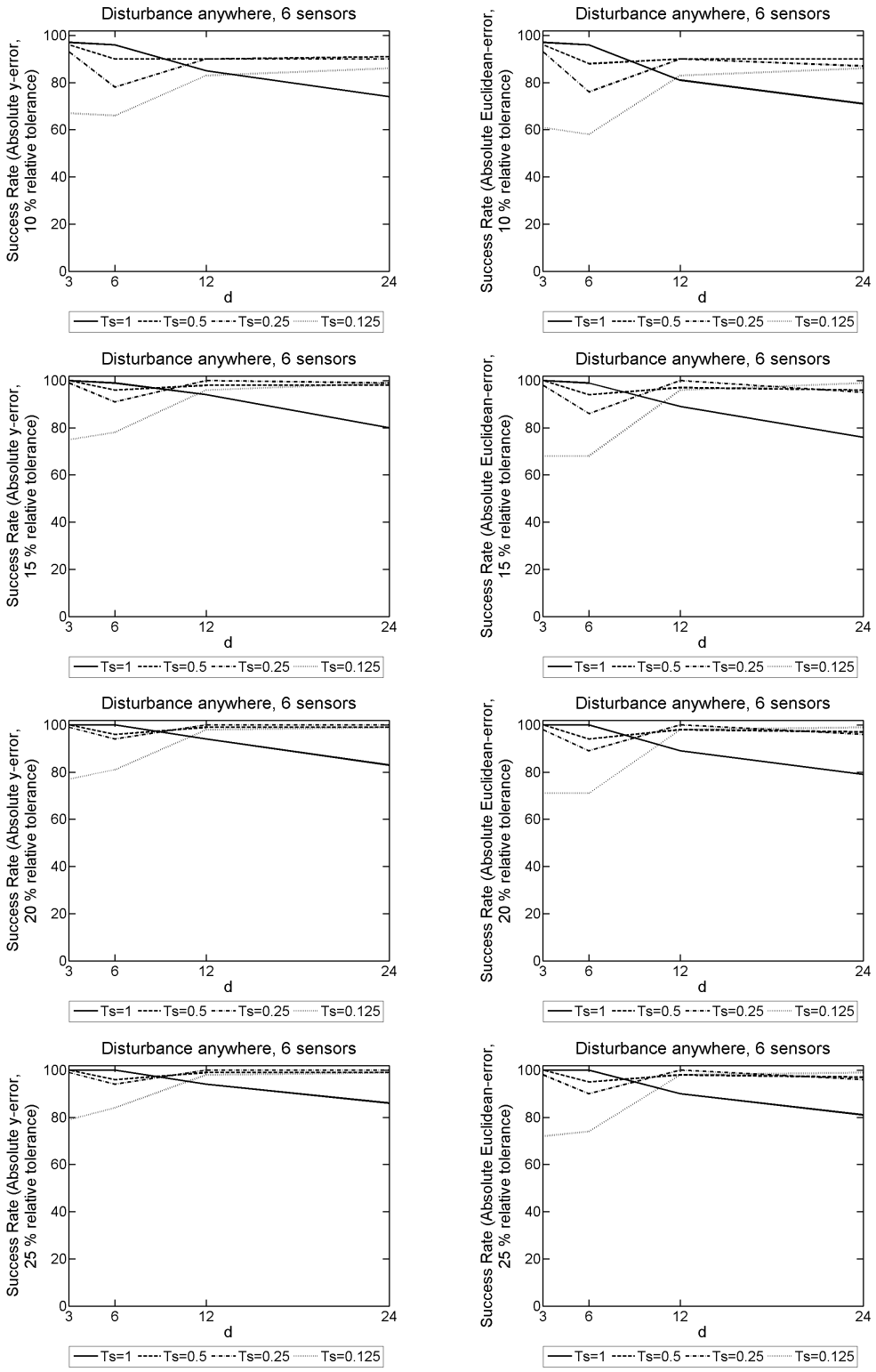


Figure 9.106: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 150$, $M = 15$ and $L = 18000$, and $F = 150$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 6 sensors present.

D.3 2D model problem with 1NBC and a disturbance frequency $F = 300\mathbf{Hz}$

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	89	98	99	82	82	91	91	81	82	90	91
0.125	6	86	94	95	91	91	95	95	88	88	88	91

Table 9.113: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	99	99	100	98	98	99	99	98	98	99	99
0.125	6	97	100	100	92	92	97	97	92	92	96	97

Table 9.114: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100

Table 9.115: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100

Table 9.116: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100

Table 9.117: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	100	100	100	100	100	100	100	100	100	100	100
0.125	6	100	100	100	100	100	100	100	100	100	100	100

Table 9.118: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	37	81	89	46	58	72	78	40	54	63	72
0.125	6	28	74	82	43	55	65	73	38	52	60	70

Table 9.119: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	57	93	99	64	71	74	80	62	71	74	79
0.125	6	59	85	90	63	74	79	82	60	72	78	81

Table 9.120: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	53	91	100	59	73	80	84	56	73	80	84
0.125	6	76	96	97	77	88	92	95	76	87	92	94

Table 9.121: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	63	96	100	66	76	79	86	65	76	79	86
0.125	6	76	98	98	83	92	92	94	83	92	92	93

Table 9.122: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	69	94	100	75	83	84	89	73	83	84	89
0.125	6	85	100	100	87	93	95	97	86	93	95	97

Table 9.123: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	71	95	99	68	77	82	86	67	77	82	86
0.125	6	82	98	100	87	96	98	99	87	95	98	99

Table 9.124: 2D model problem with 1NBC: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

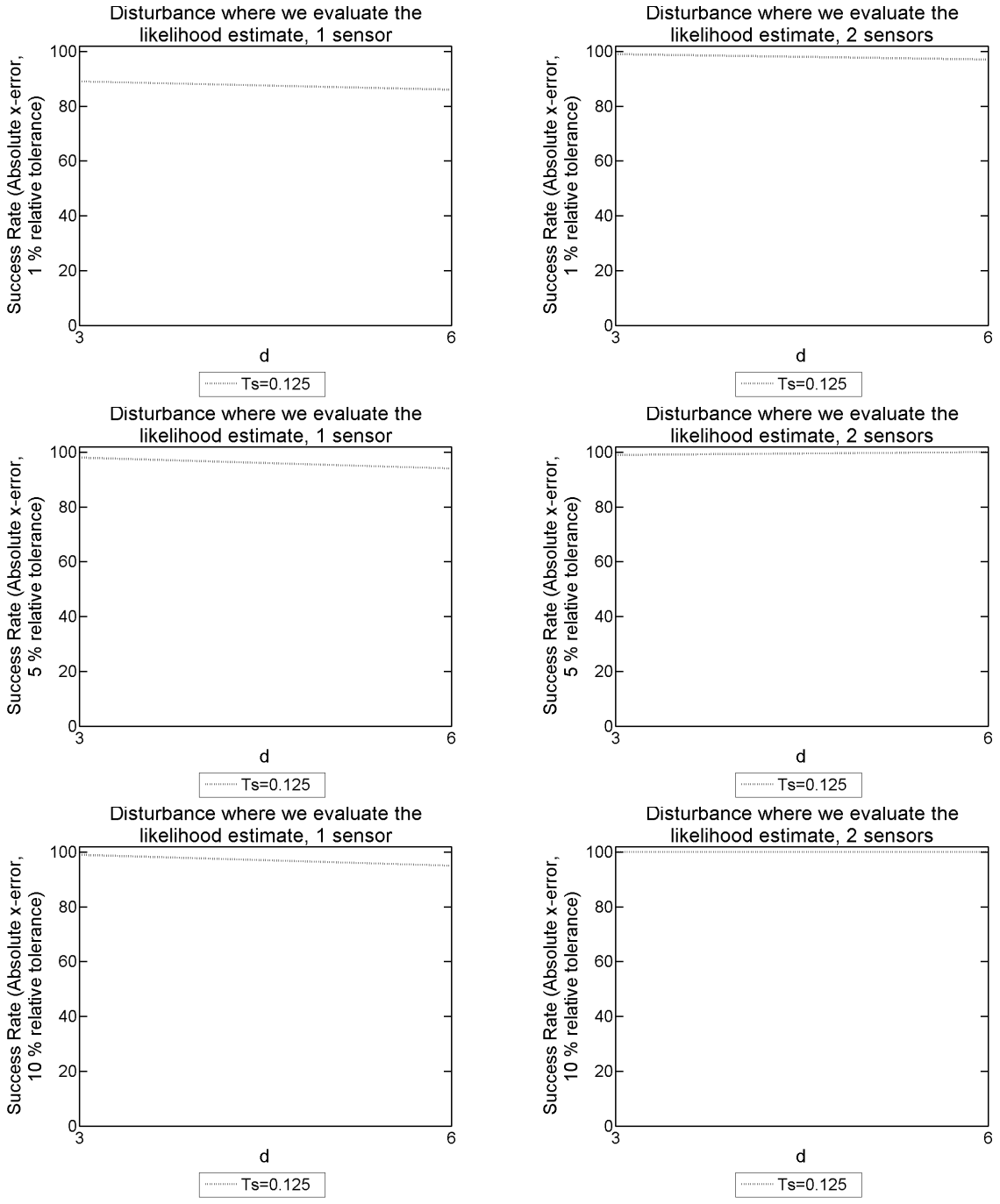


Figure 9.107: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

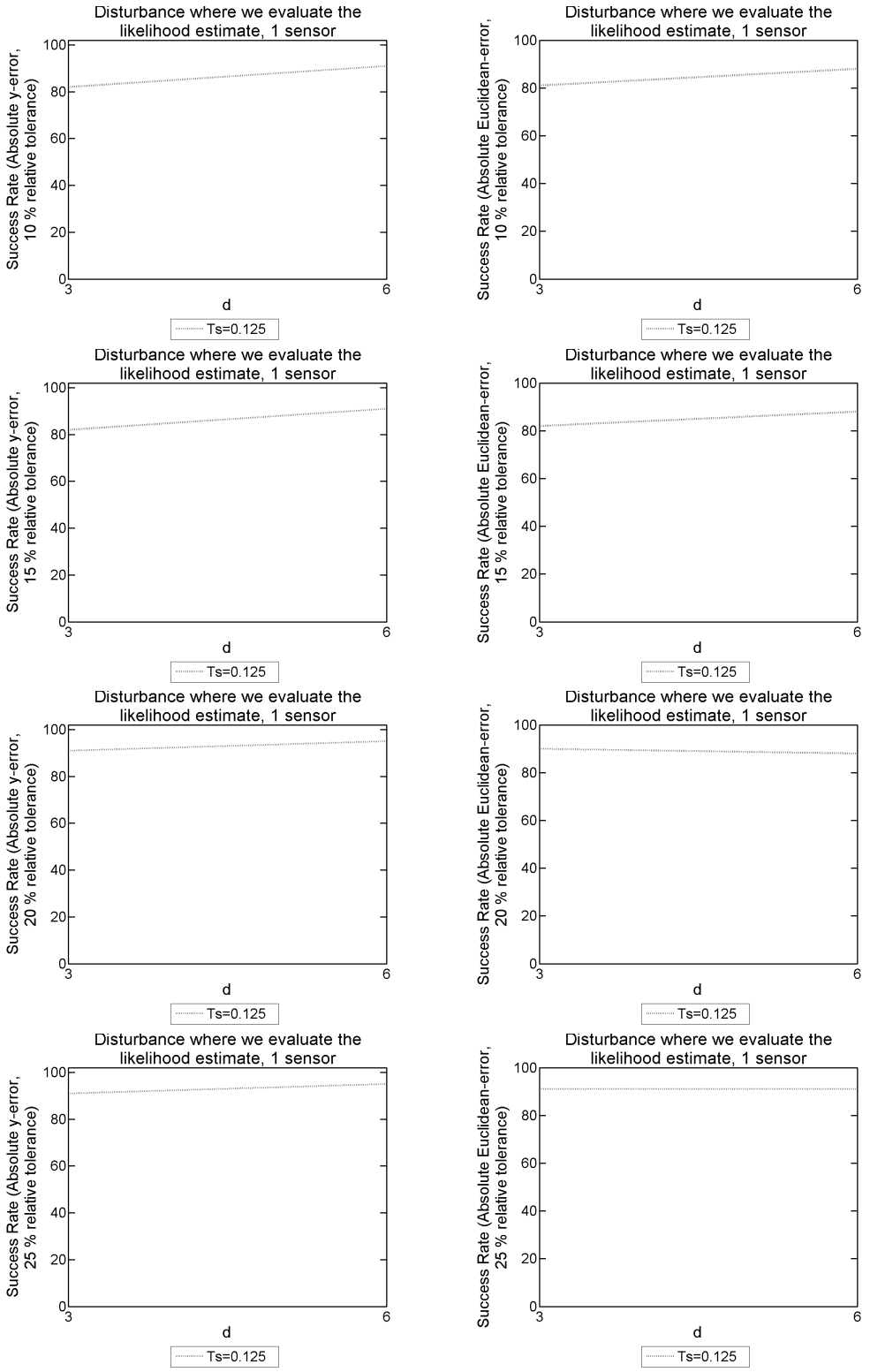


Figure 9.108: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 1 sensor present.

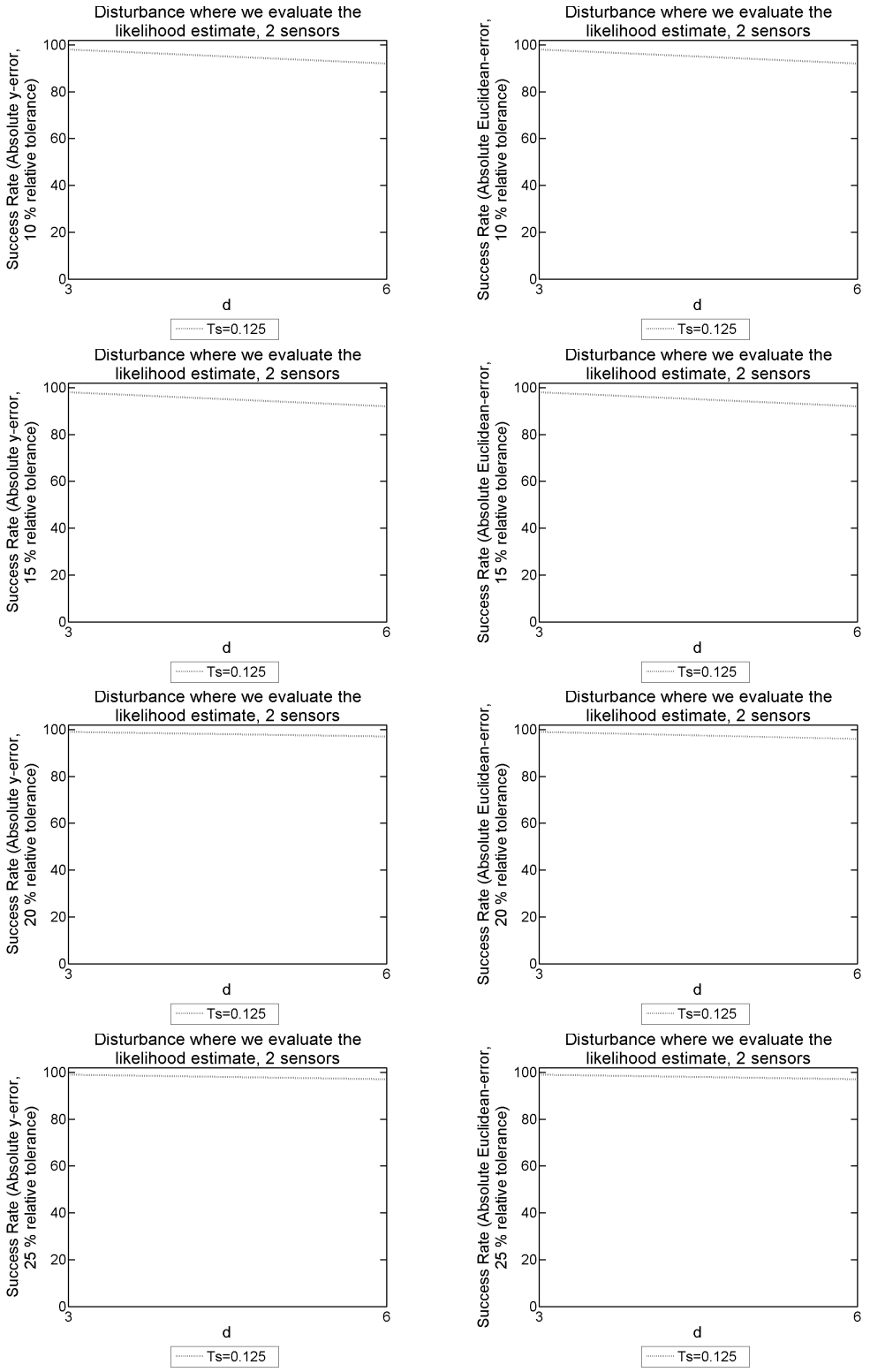


Figure 9.109: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 2 sensors present.

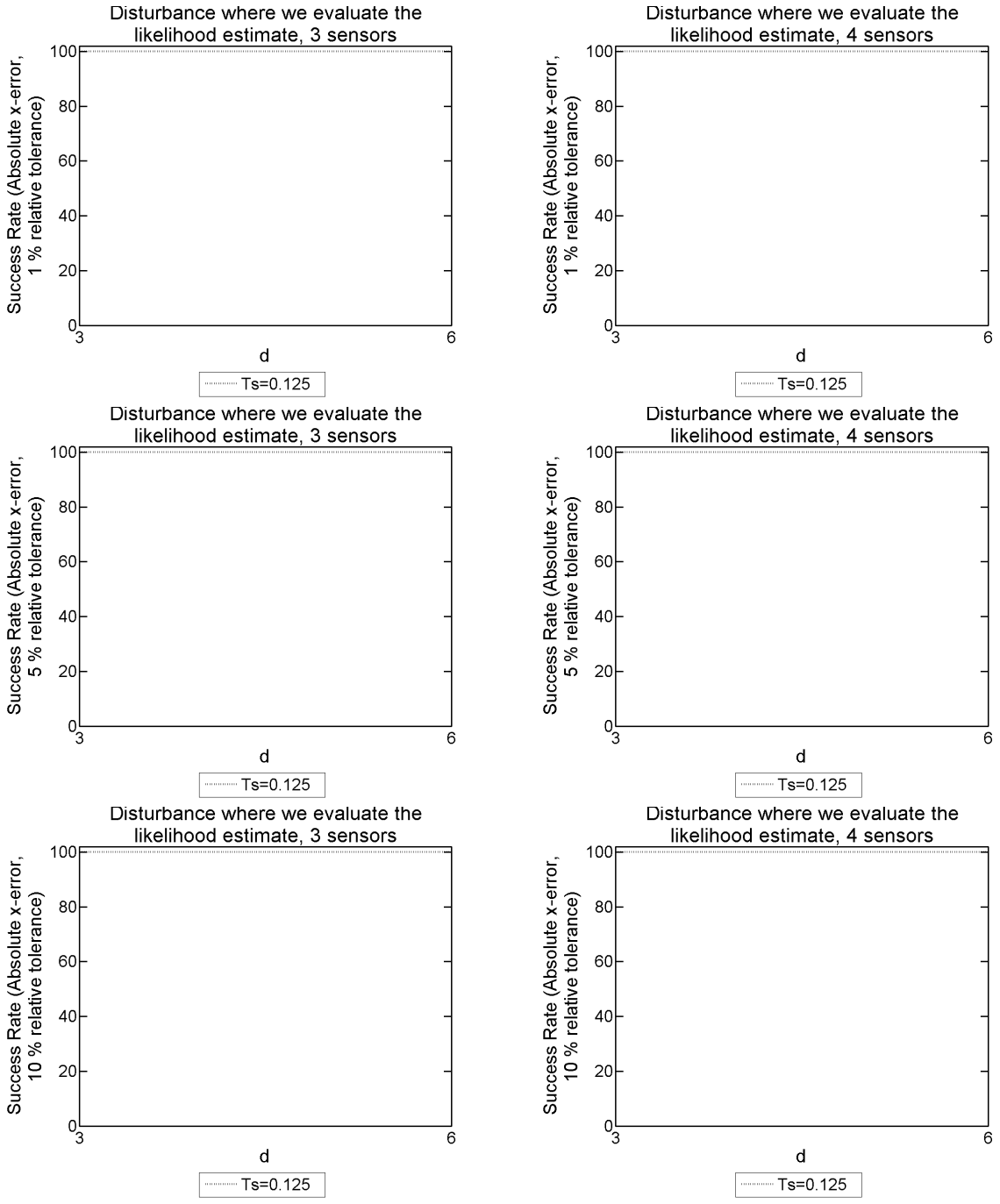


Figure 9.110: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

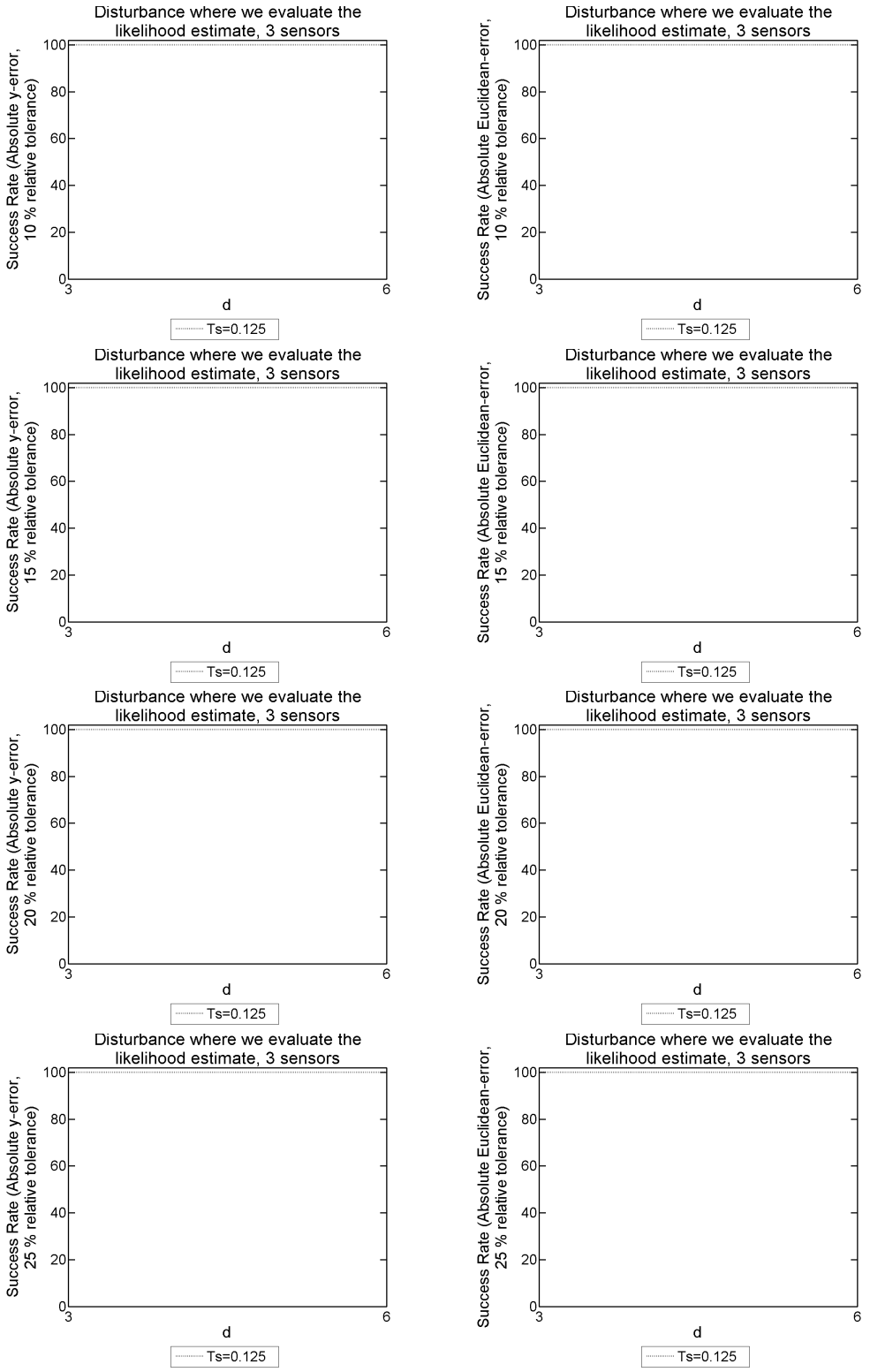


Figure 9.111: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 3 sensors present.

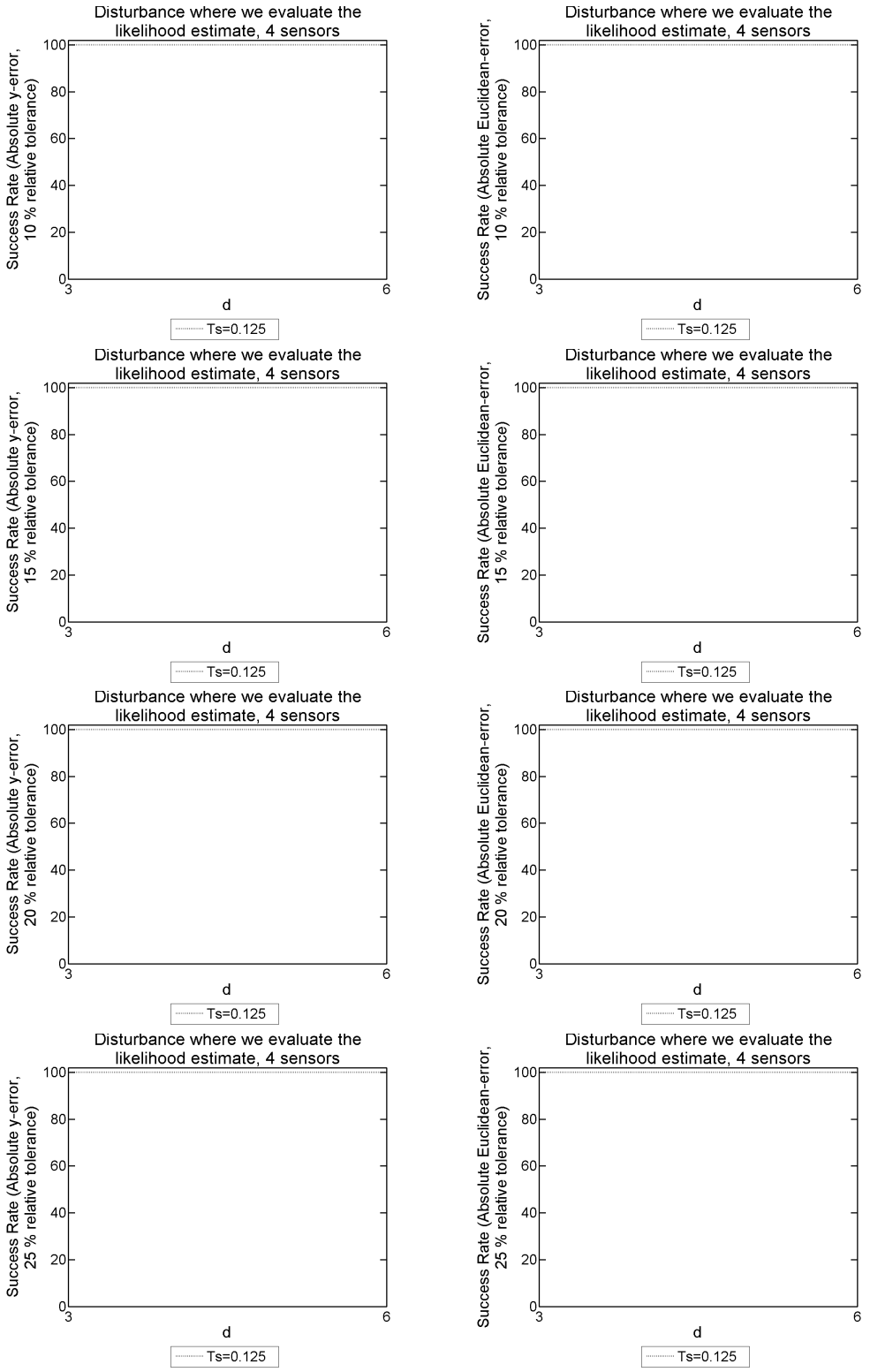


Figure 9.112: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 4 sensors present.

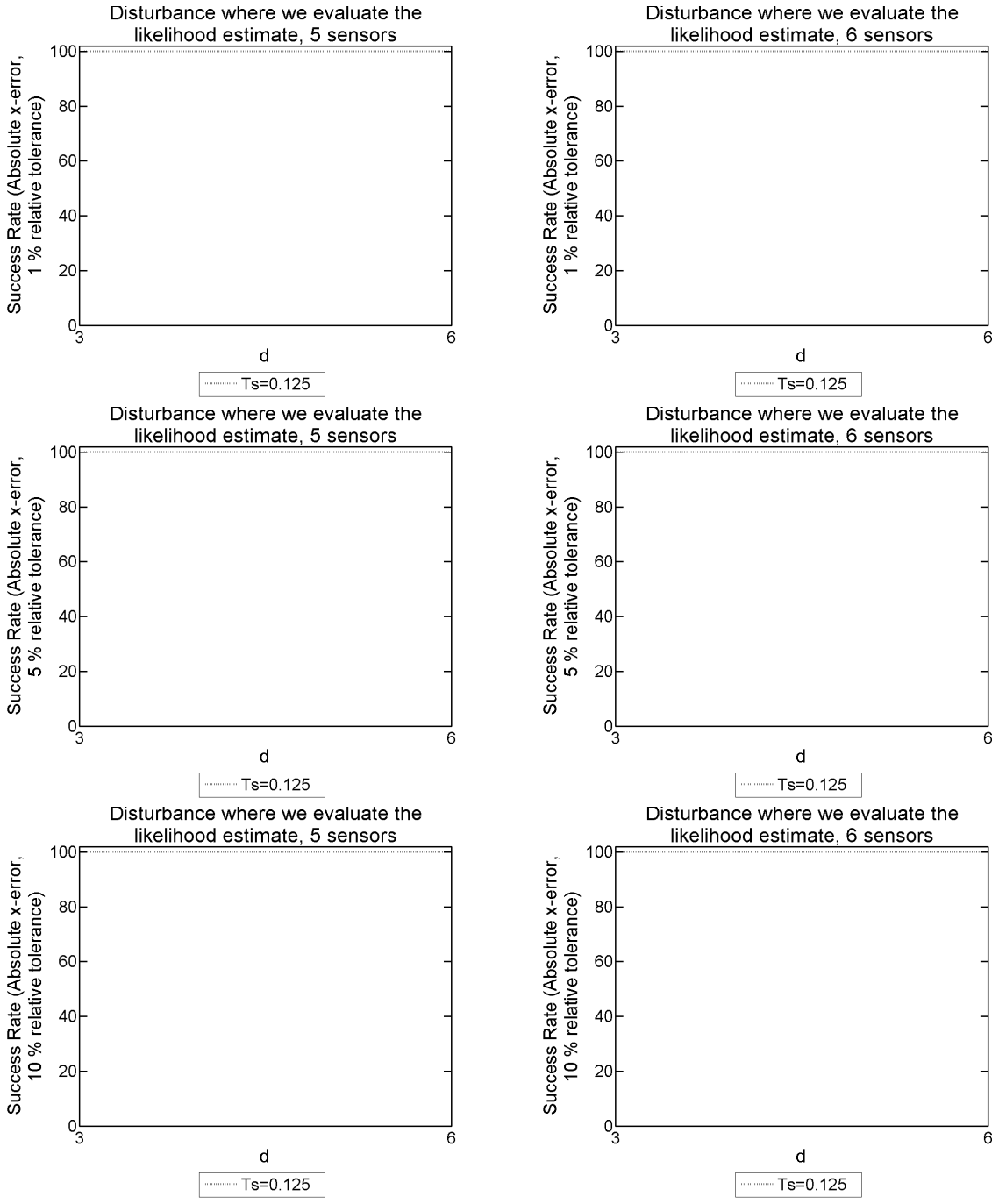


Figure 9.113: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

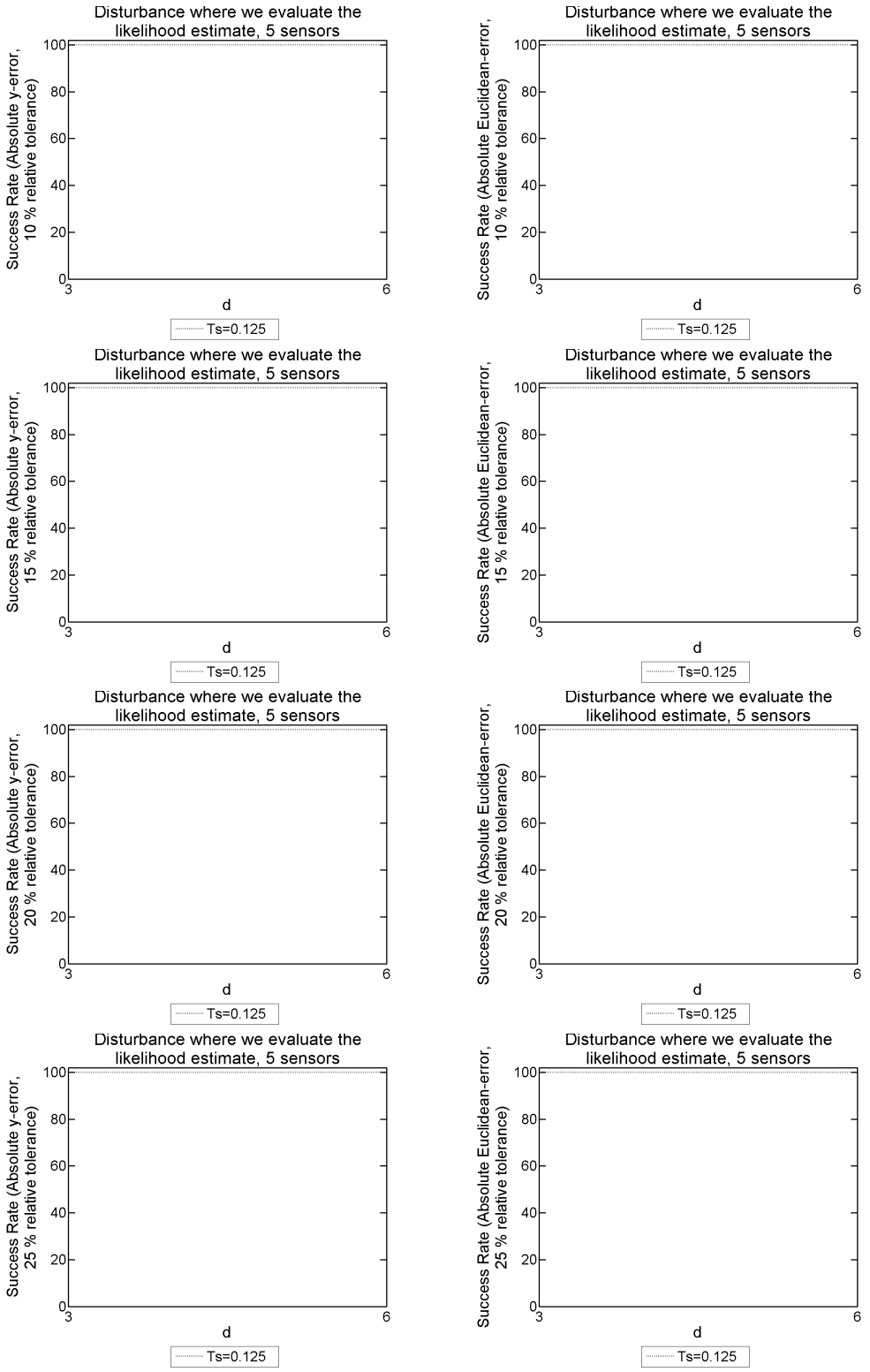


Figure 9.114: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 5 sensors present.

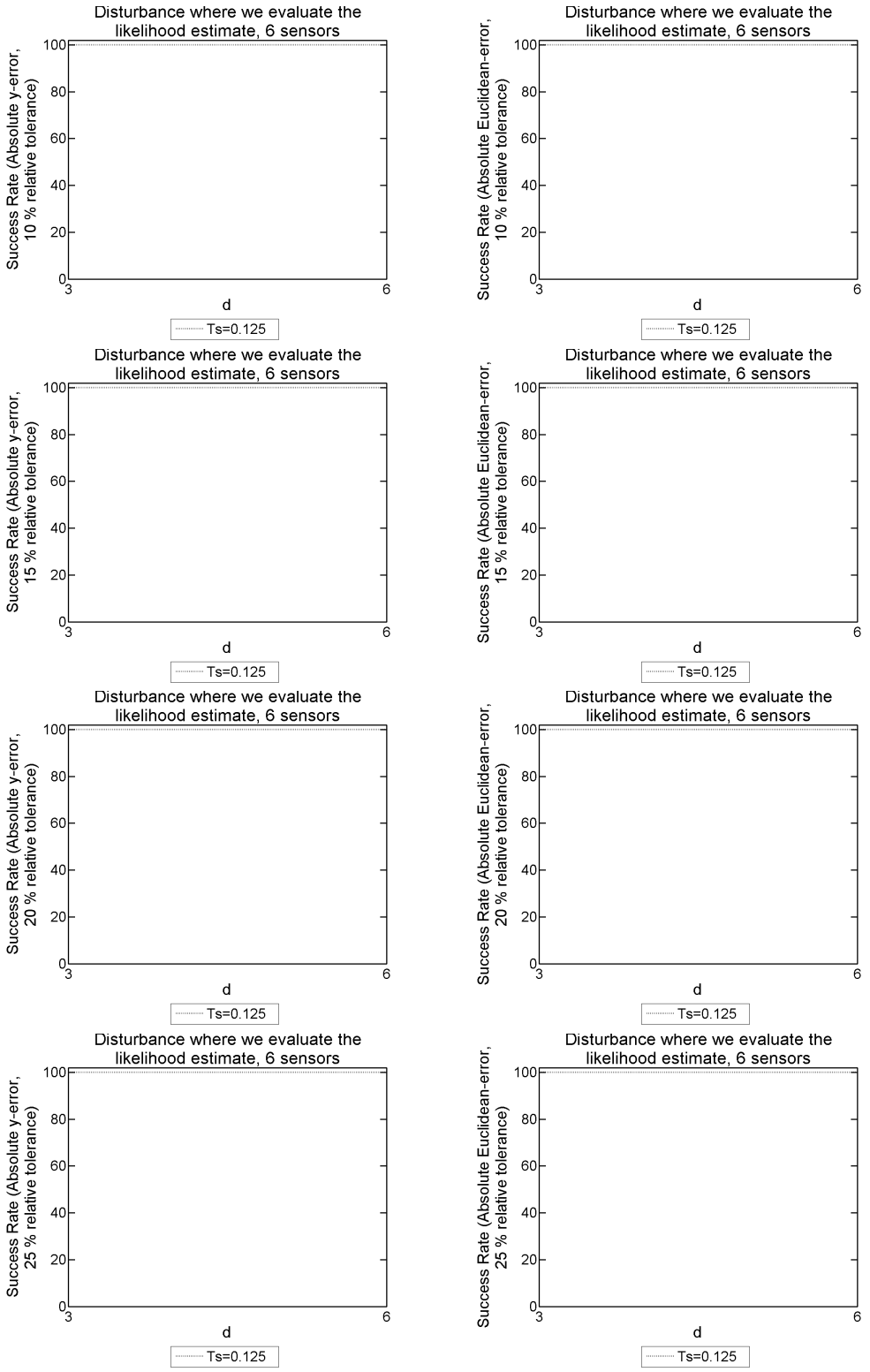


Figure 9.115: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 6 sensors present.

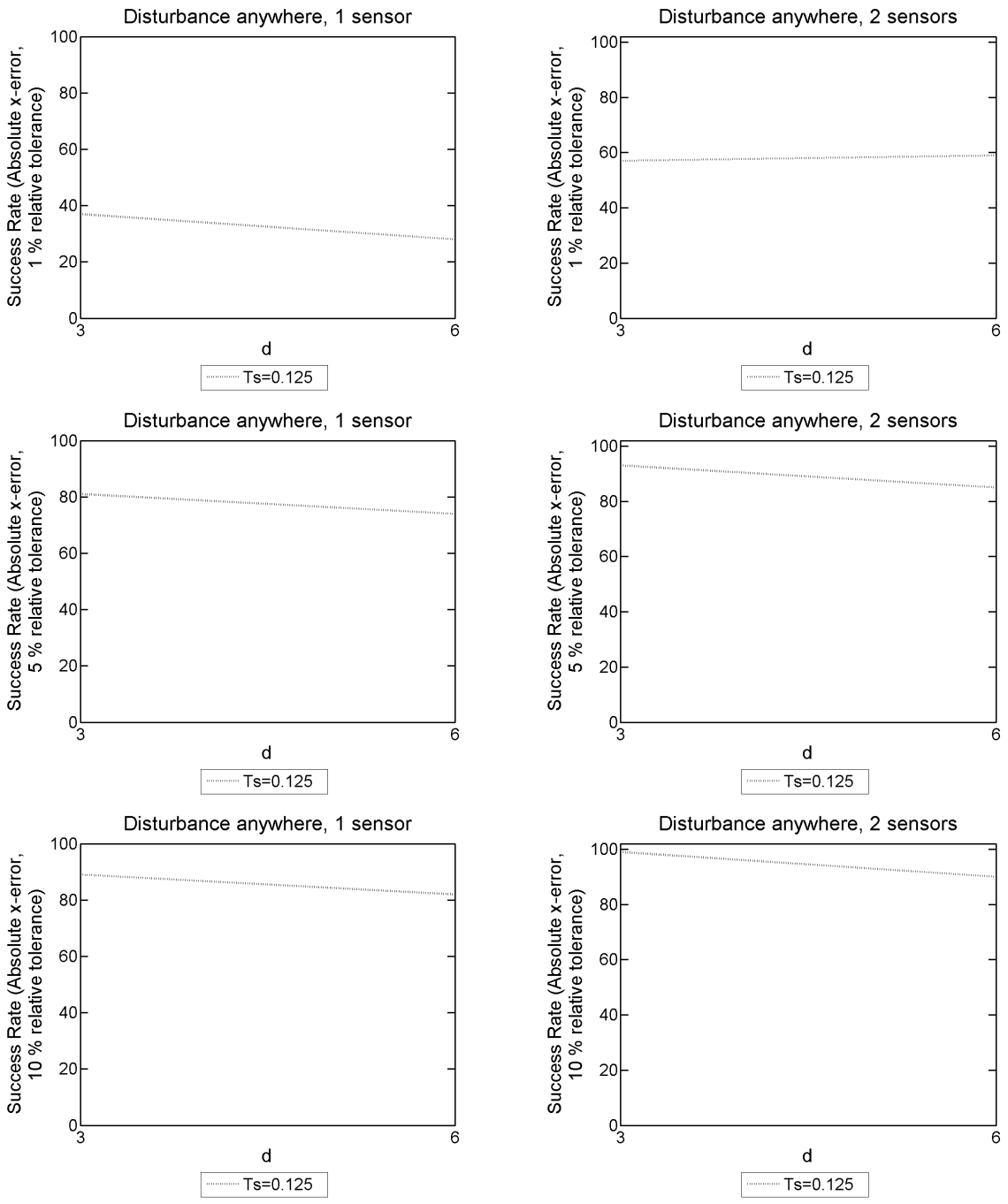


Figure 9.116: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

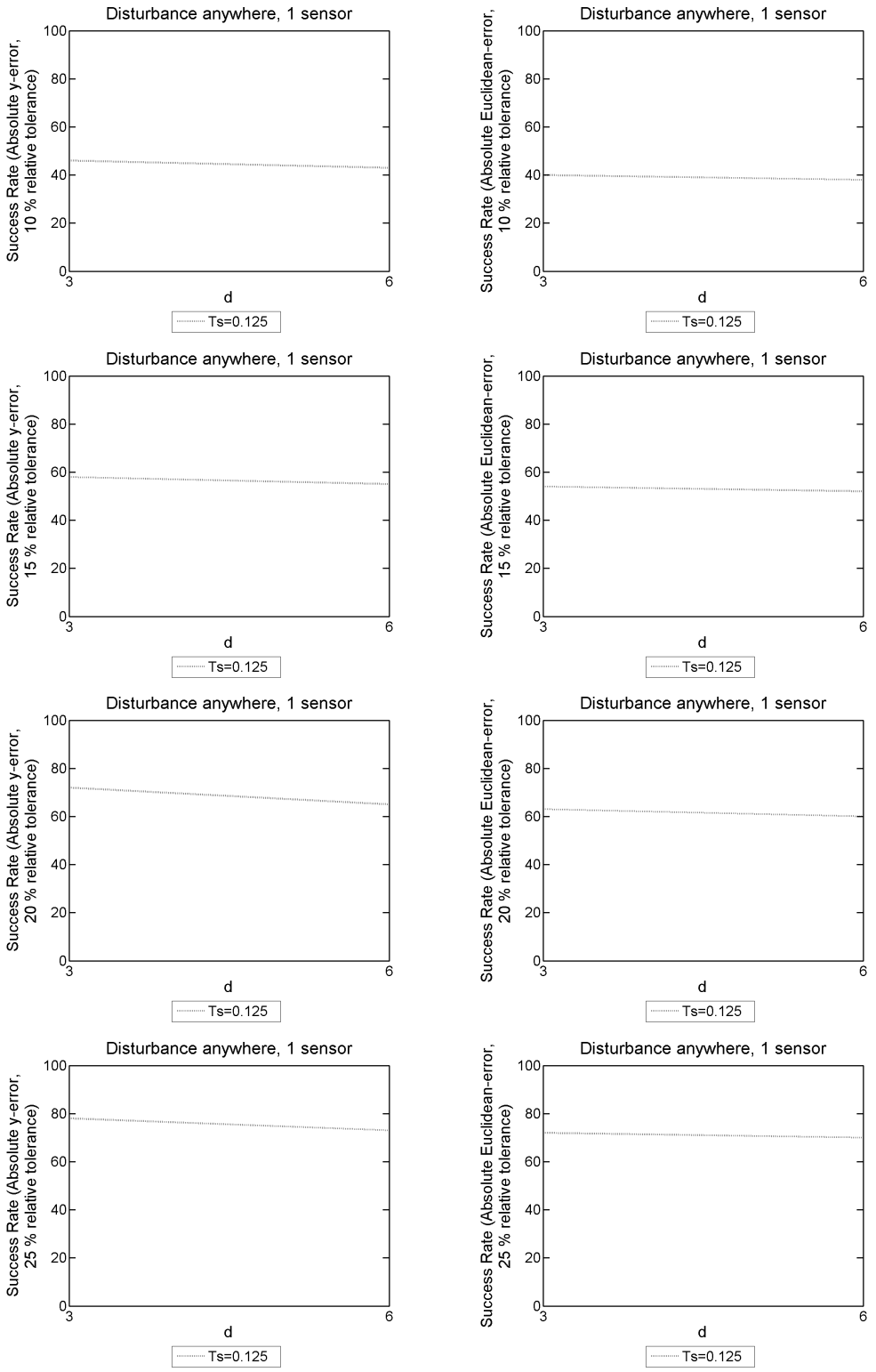


Figure 9.117: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 1 sensor present.

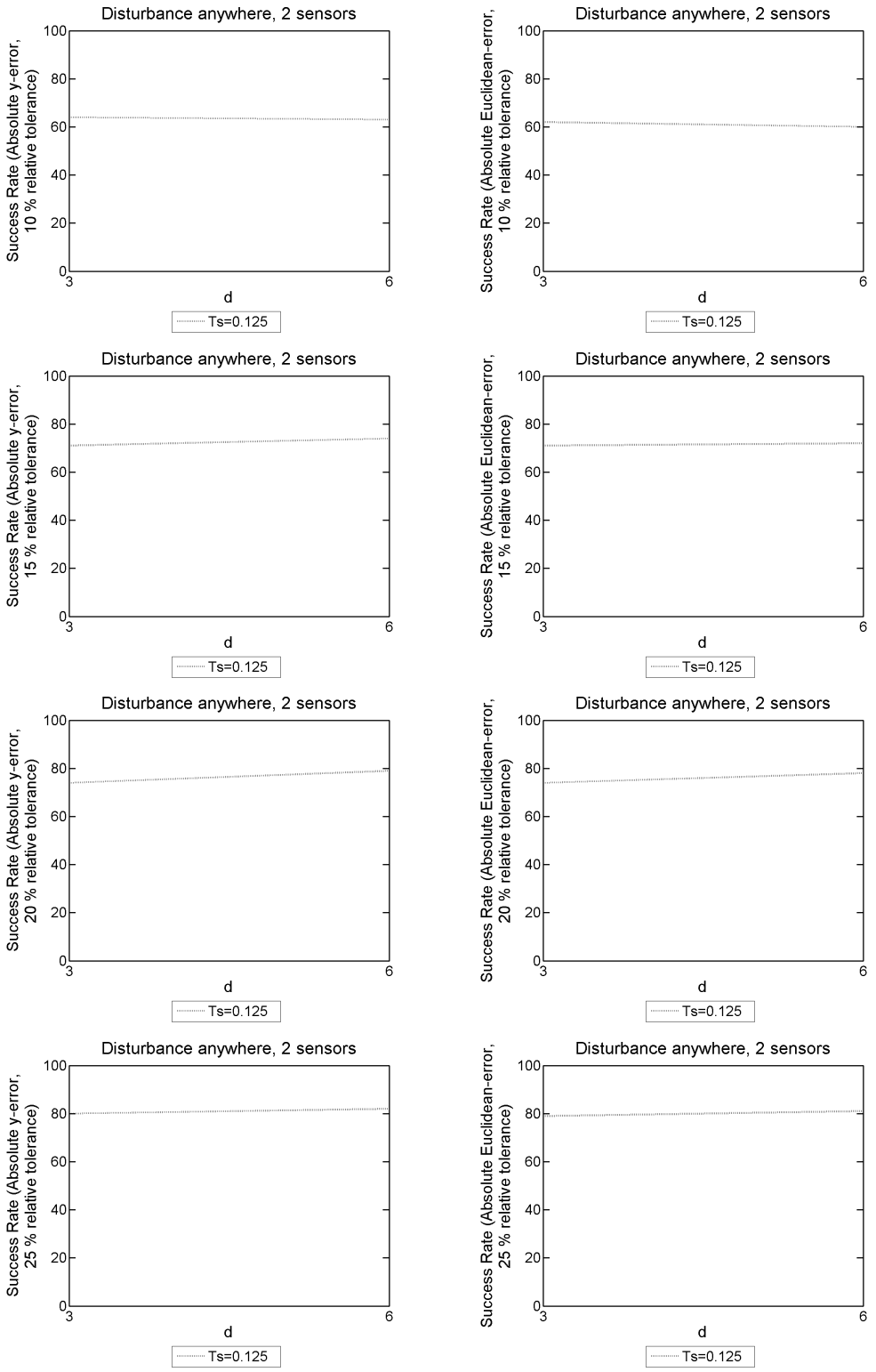


Figure 9.118: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 2 sensors present.

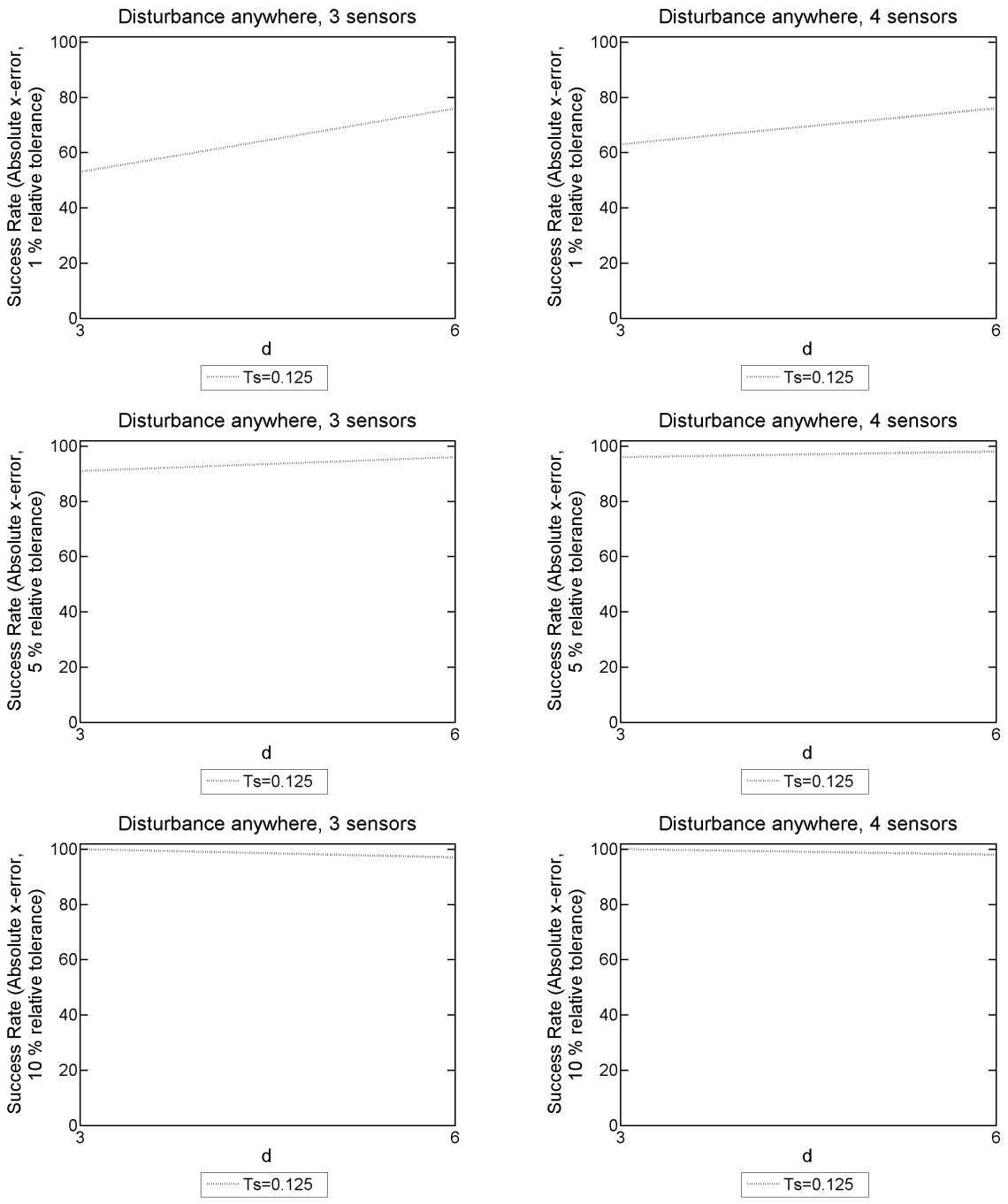


Figure 9.119: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

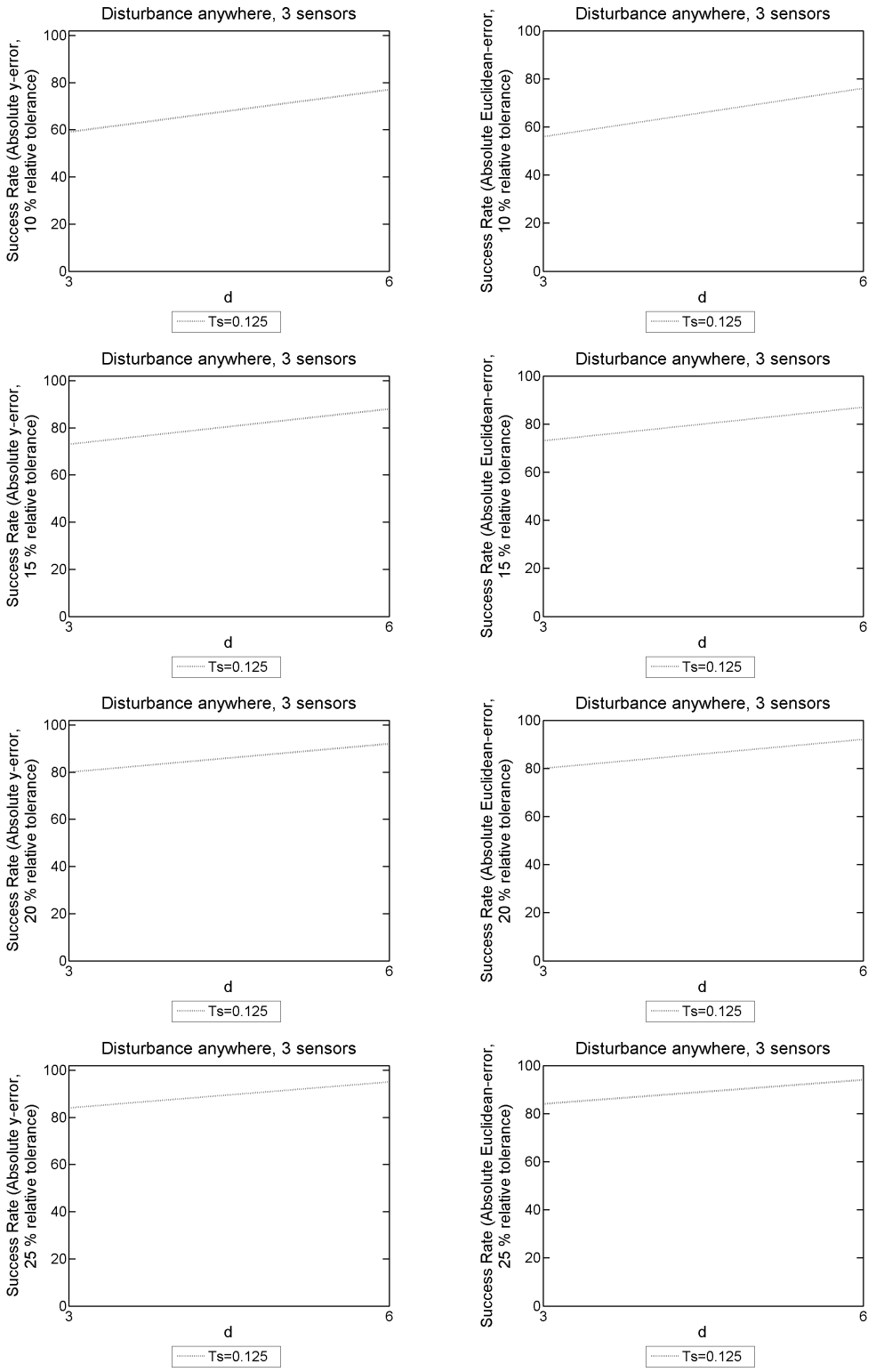


Figure 9.120: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 3 sensors present.

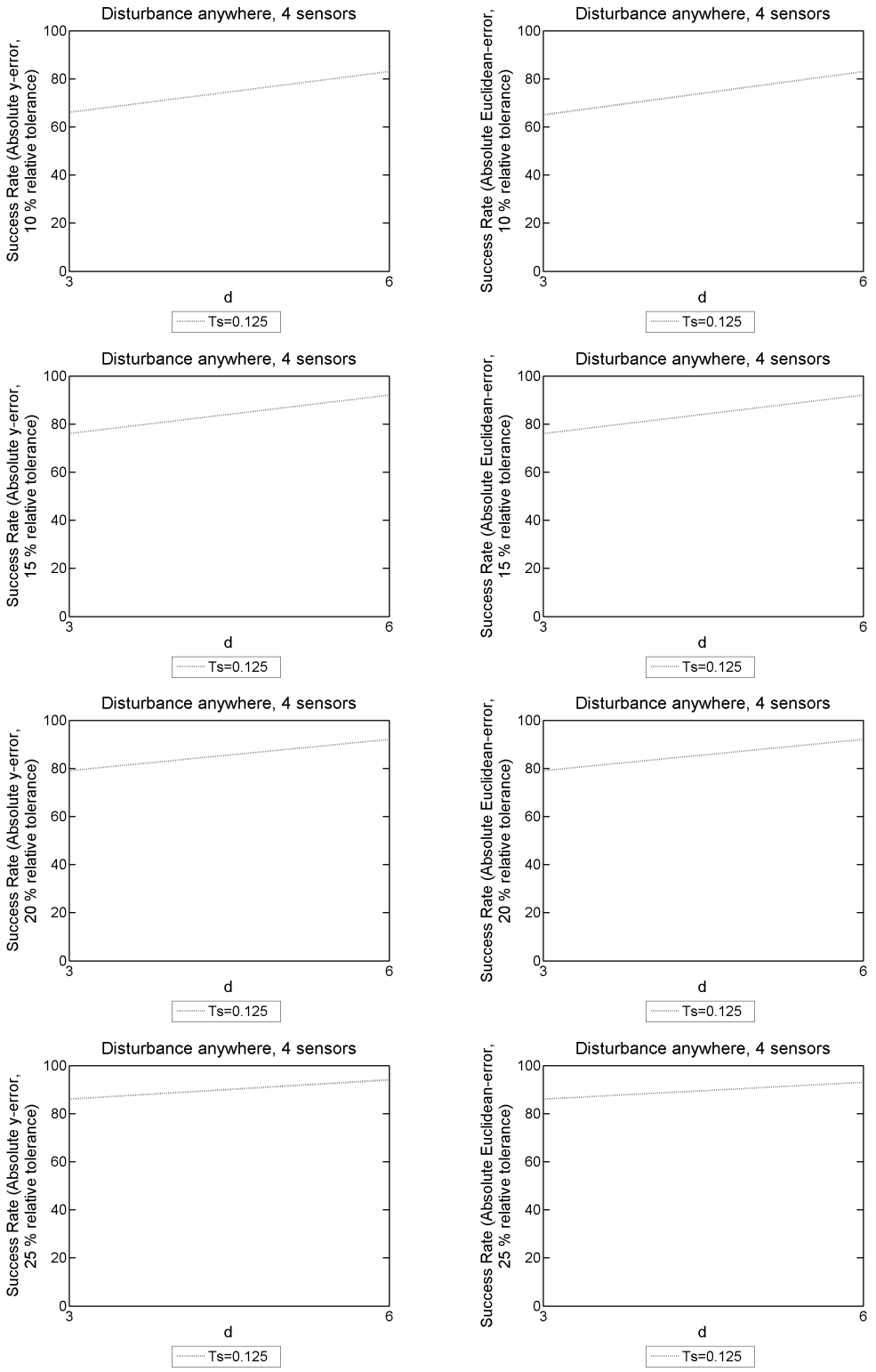


Figure 9.121: 2D model problem with 1NBC: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 4 sensors present.

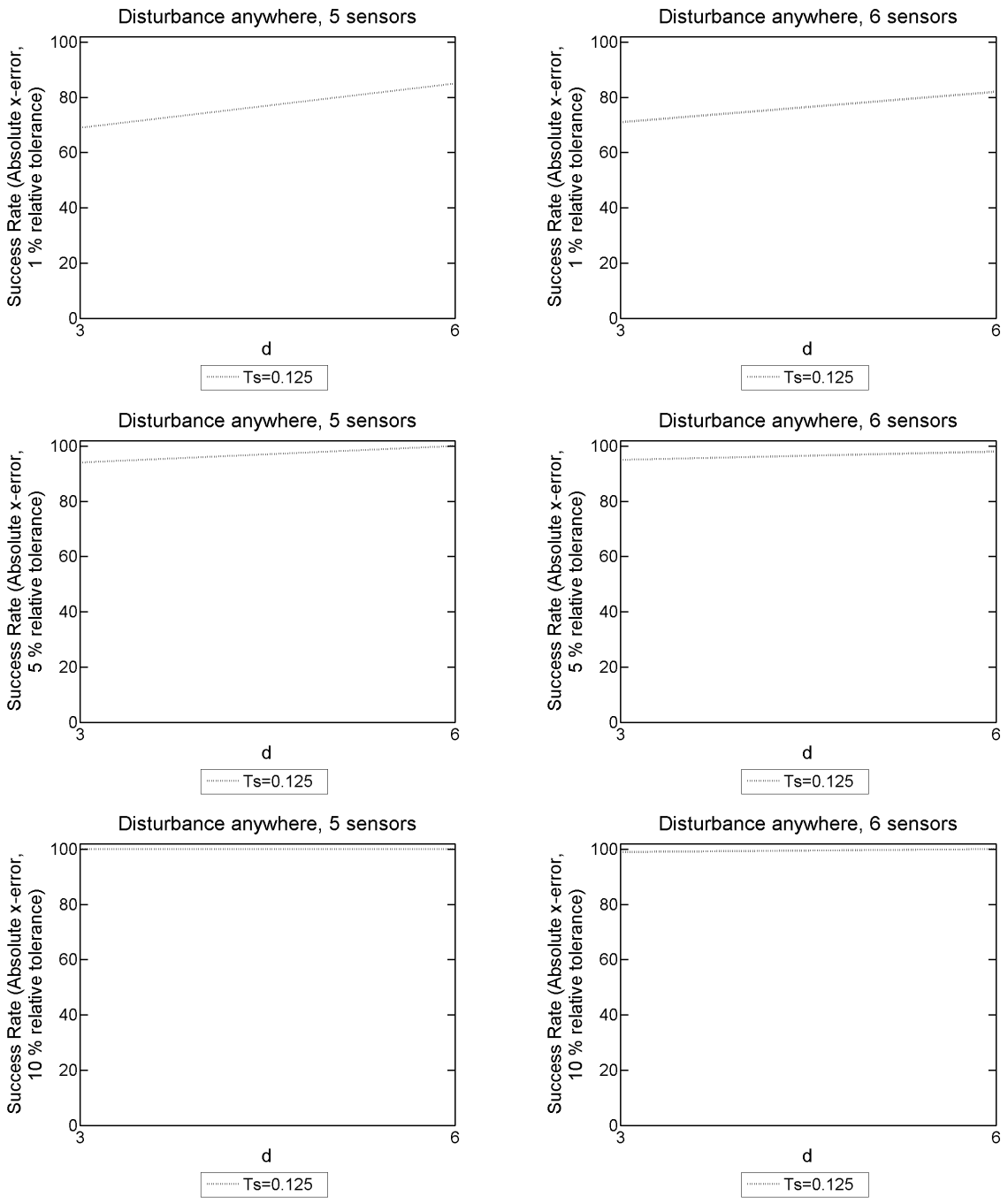


Figure 9.122: 2D model problem with 1NBC: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

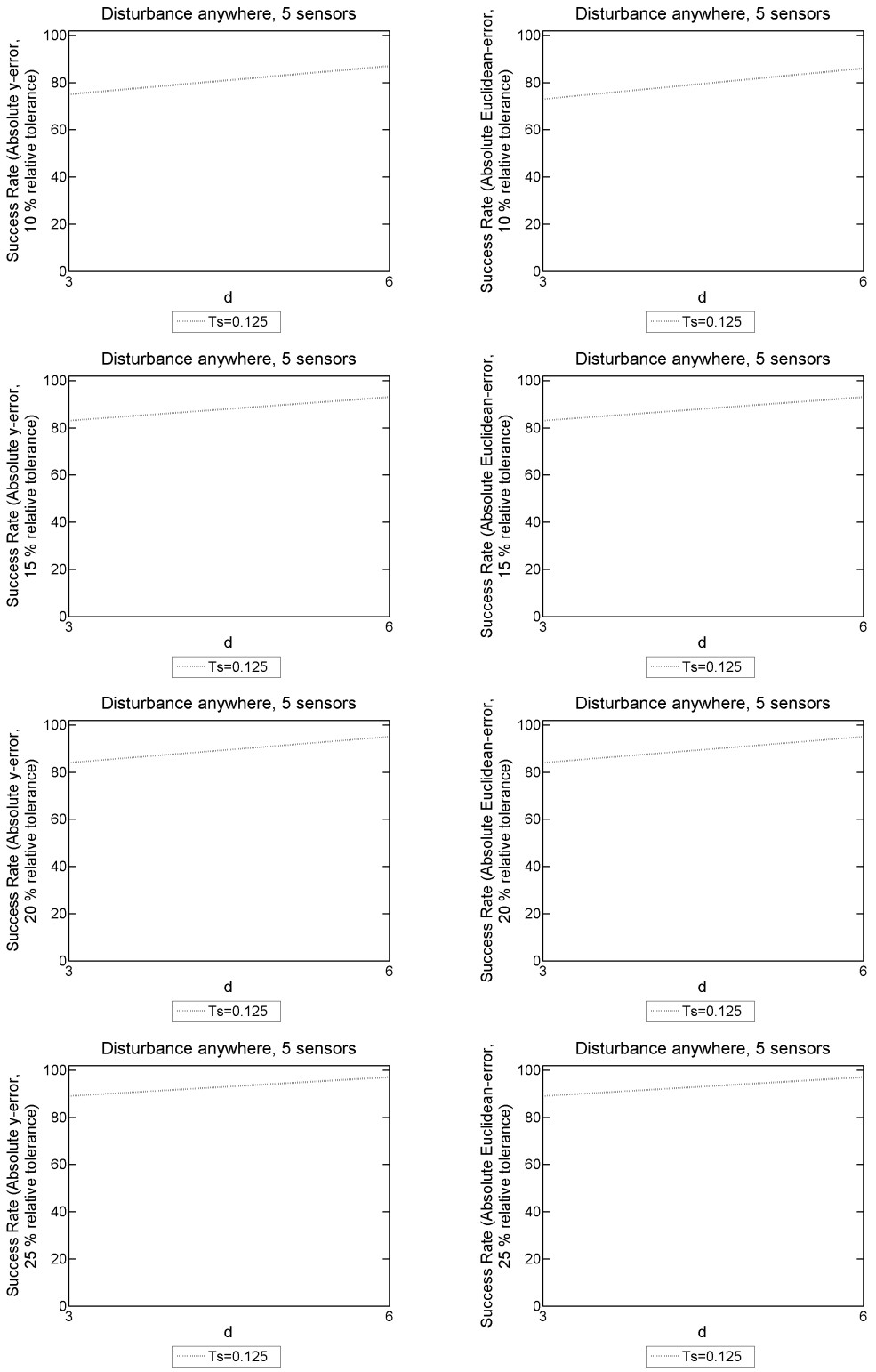


Figure 9.123: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 5 sensors present.

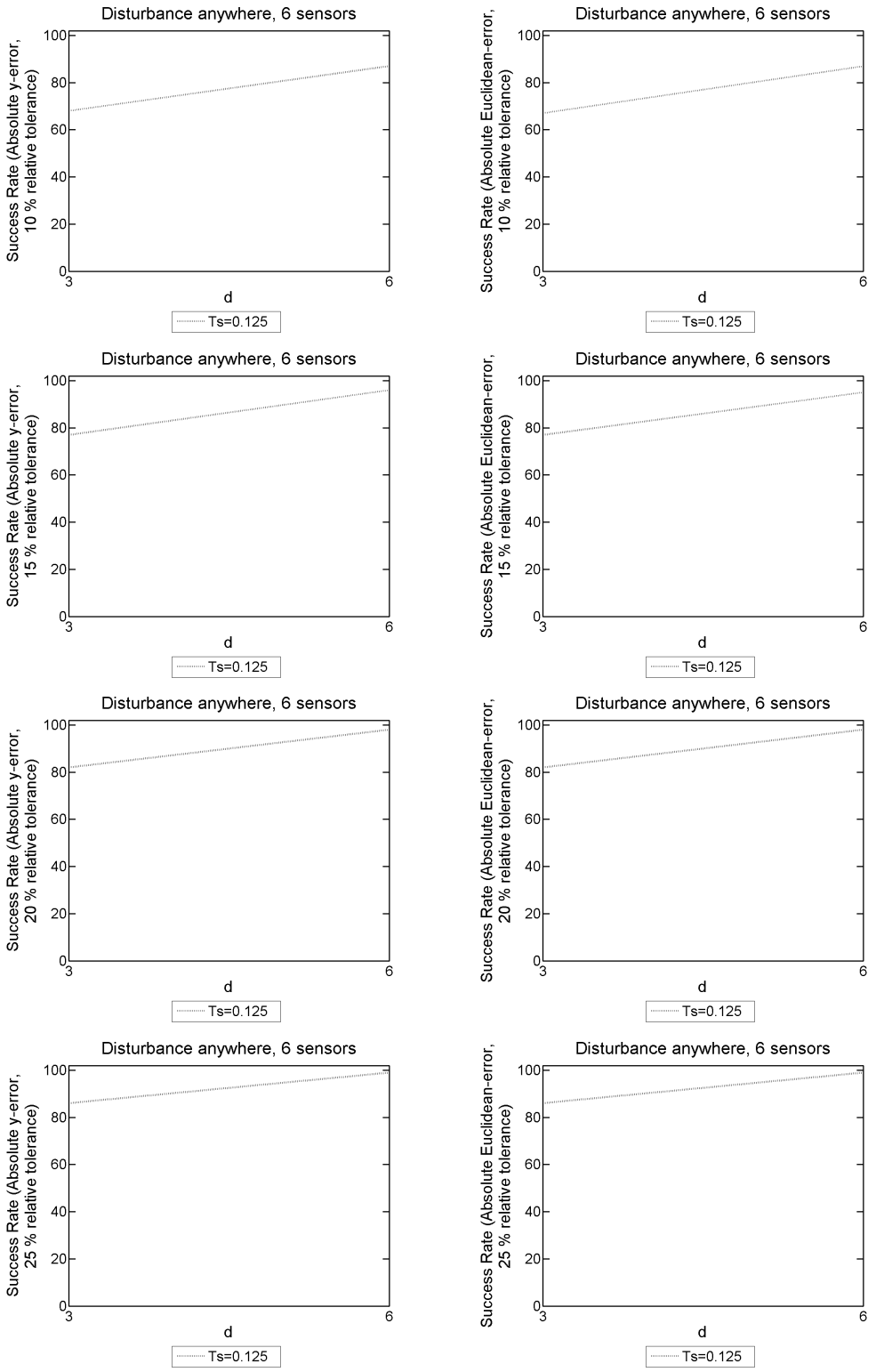


Figure 9.124: 2D model problem with 1NBC: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 6 sensors present.

D.4 2D model problem with 3NBCs and a disturbance frequency $F = 300\mathbf{Hz}$

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	68	87	90	70	70	89	89	69	69	74	86
0.125	6	72	80	84	78	78	90	90	76	76	78	83

Table 9.125: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	85	97	99	85	85	91	91	85	85	88	91
0.125	6	93	97	97	95	95	95	95	95	95	95	95

Table 9.126: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	75	99	99	66	66	93	93	66	66	78	93
0.125	6	100	100	100	99	99	100	100	99	99	100	100

Table 9.127: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	89	100	100	75	75	92	92	75	75	88	92
0.125	6	99	100	100	99	99	100	100	99	99	100	100

Table 9.128: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	85	100	100	82	82	98	98	82	82	90	98
0.125	6	97	99	100	97	97	100	100	97	97	99	100

Table 9.129: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	96	100	100	92	92	97	97	92	92	94	97
0.125	6	82	98	100	76	76	92	92	76	76	87	92

Table 9.130: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	32	73	81	32	46	54	61	29	39	51	54
0.125	6	23	60	67	31	48	62	68	26	39	53	58

Table 9.131: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 1 sensor present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	41	79	85	47	60	66	70	44	56	62	65
0.125	6	49	73	81	48	65	71	74	43	59	66	70

Table 9.132: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 2 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	60	97	98	57	71	77	82	56	70	75	81
0.125	6	57	79	82	48	58	66	71	46	55	61	69

Table 9.133: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 3 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	69	100	100	46	56	62	64	46	56	62	64
0.125	6	73	96	98	62	74	79	84	62	73	78	83

Table 9.134: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 4 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	62	99	100	46	55	62	67	45	55	62	66
0.125	6	82	97	98	68	84	87	90	68	84	87	90

Table 9.135: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 5 sensors present to record data from the FDM approximation of u.

		Success Rate										
		|x-error|			|y-error|				|Euclidean-error|		
Ts	d	1%	5%	10%	10%	15%	20%	25%	10%	15%	20%	25%
0.125	3	65	99	100	50	63	68	70	49	63	68	70
0.125	6	69	98	99	68	81	86	87	68	80	86	87

Table 9.136: 2D model problem with 3NBCs: The success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, and 6 sensors present to record data from the FDM approximation of u.

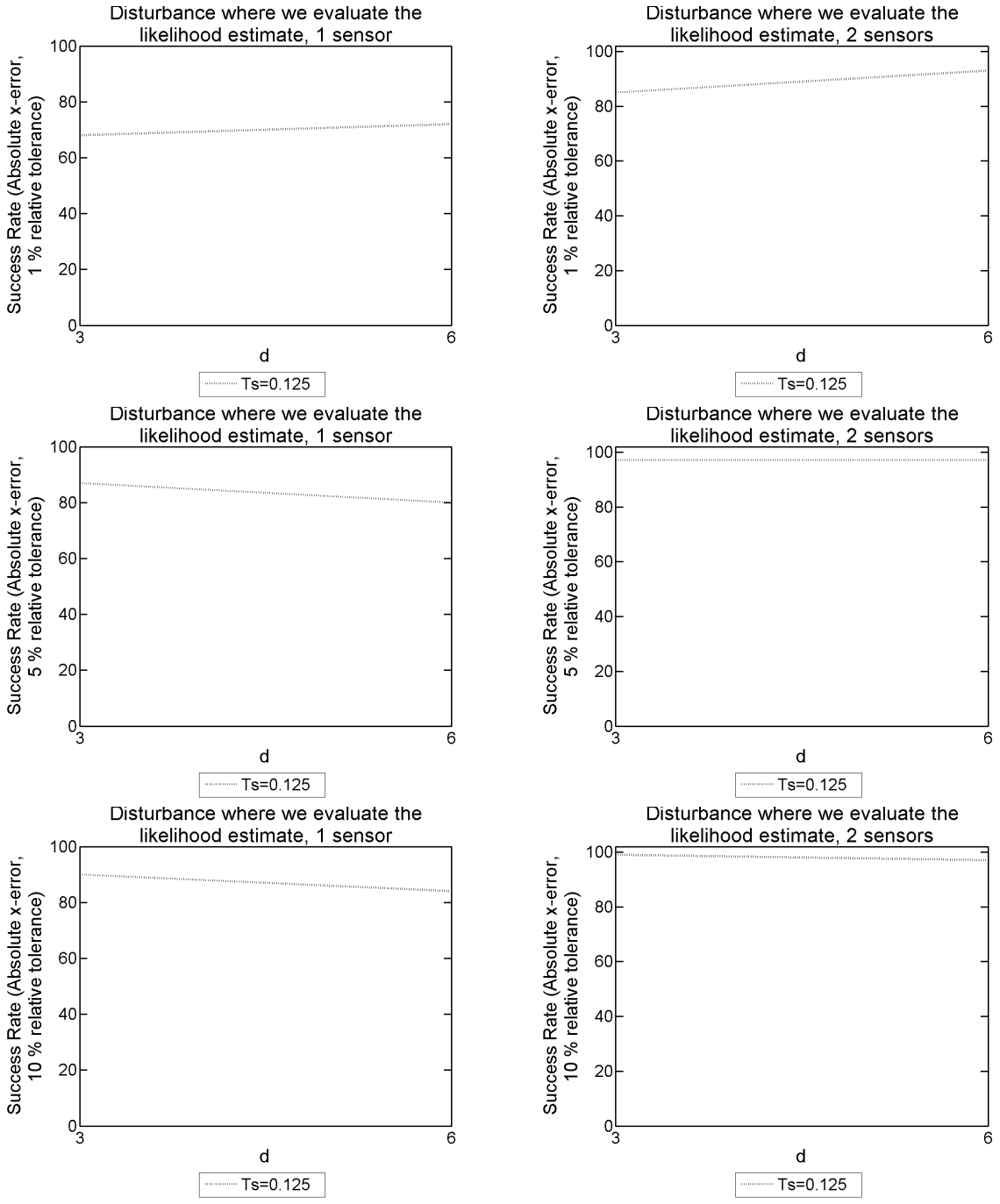


Figure 9.125: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

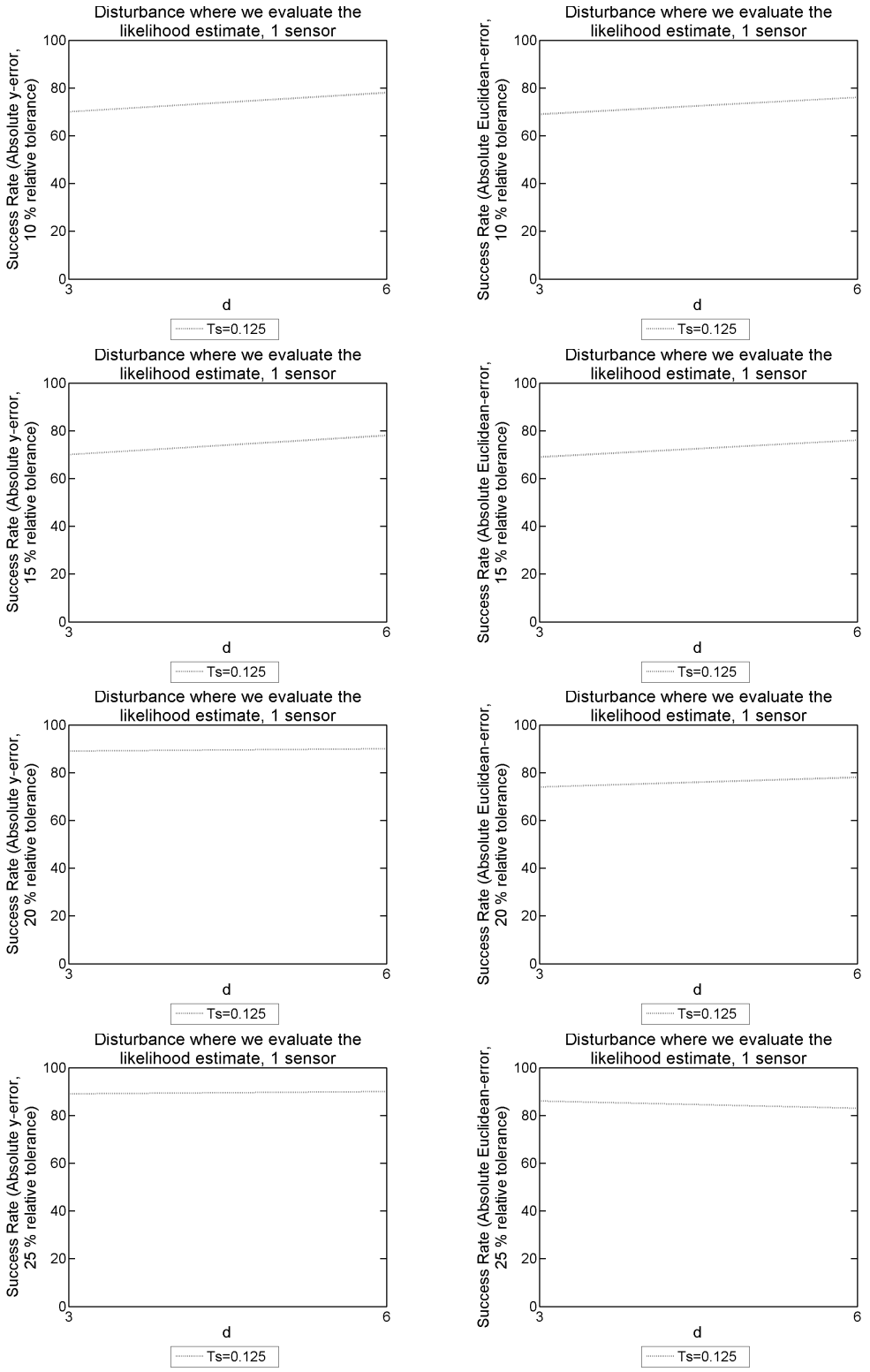


Figure 9.126: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 1 sensor present.

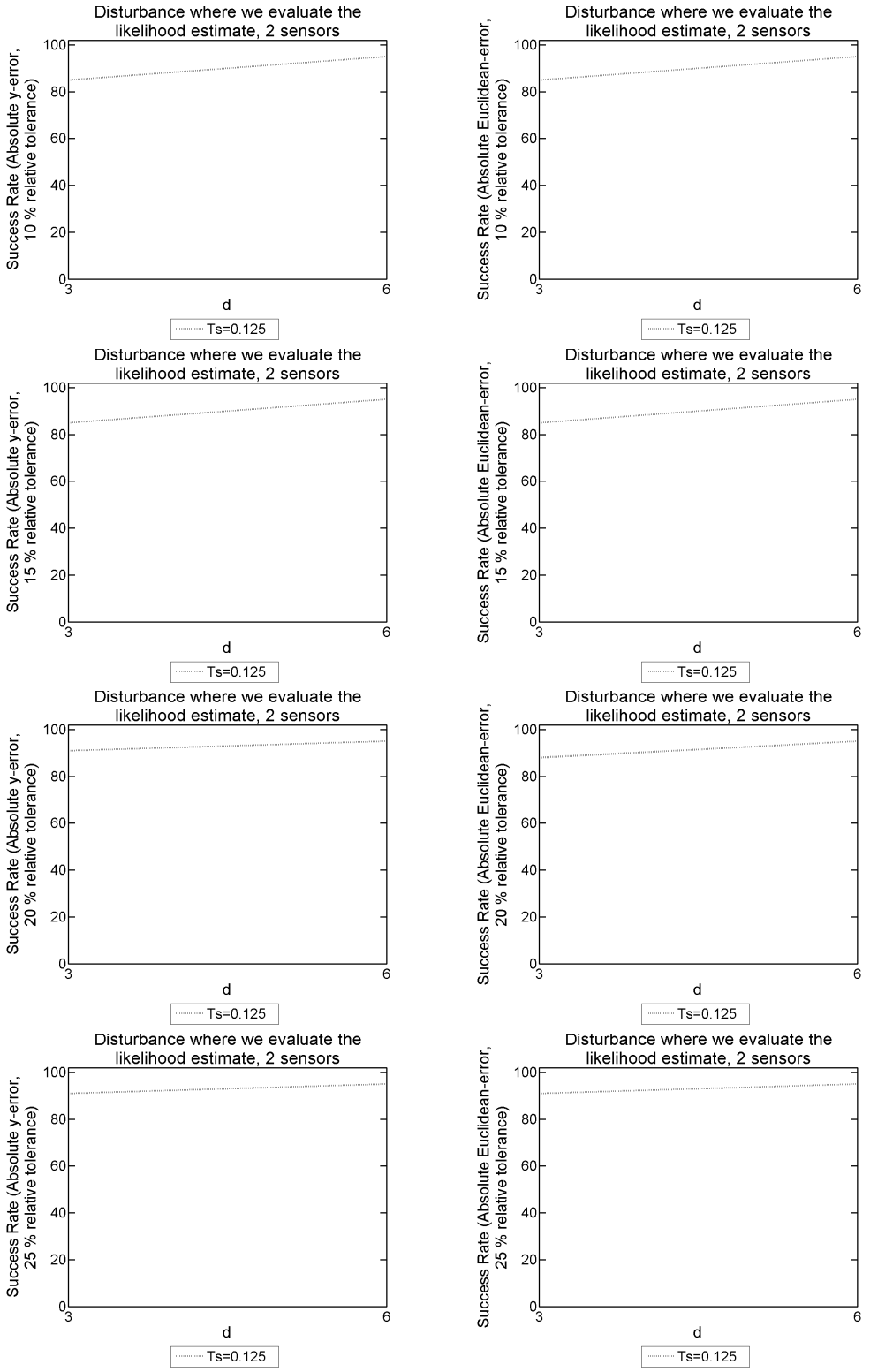


Figure 9.127: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 2 sensors present.

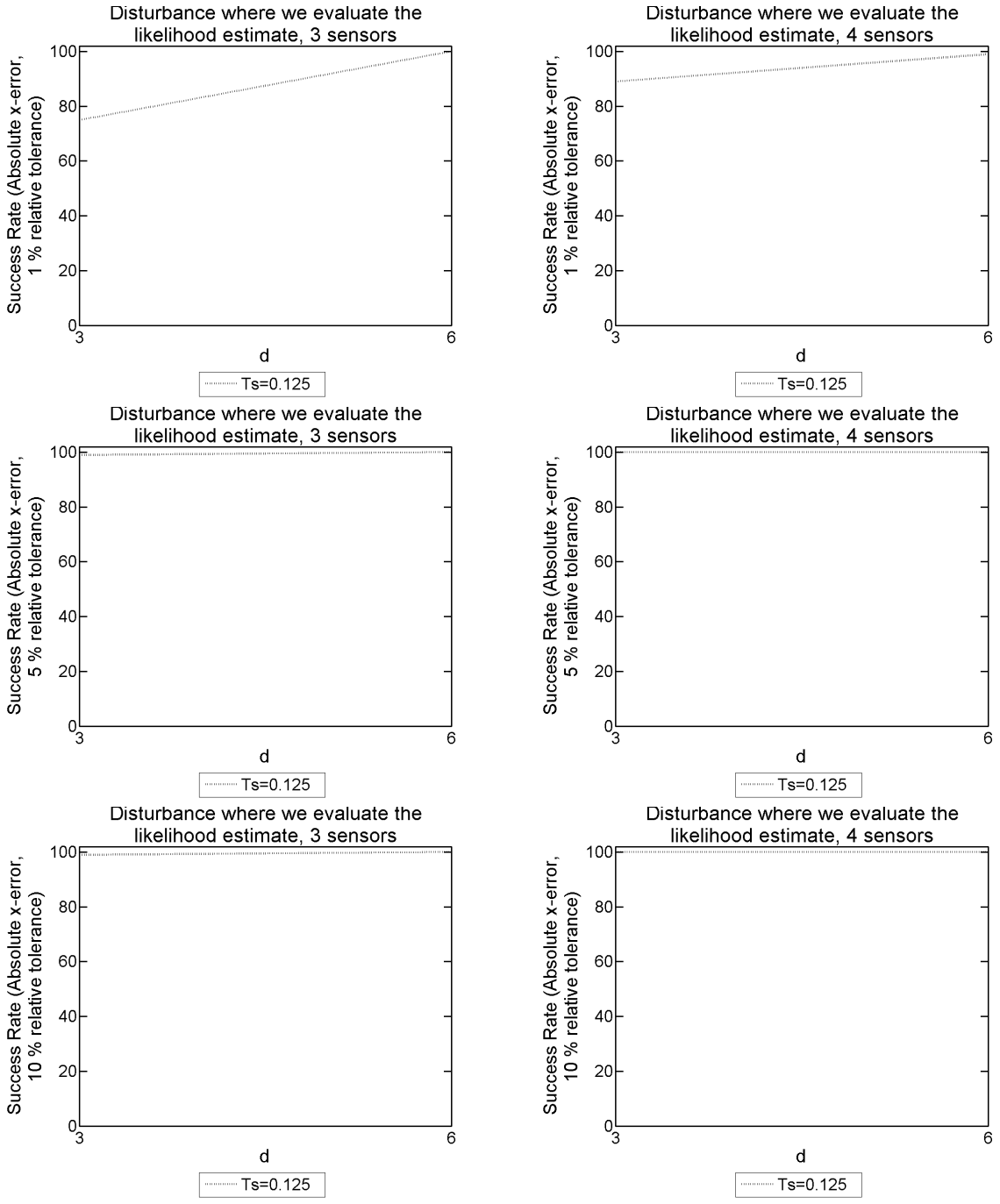


Figure 9.128: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

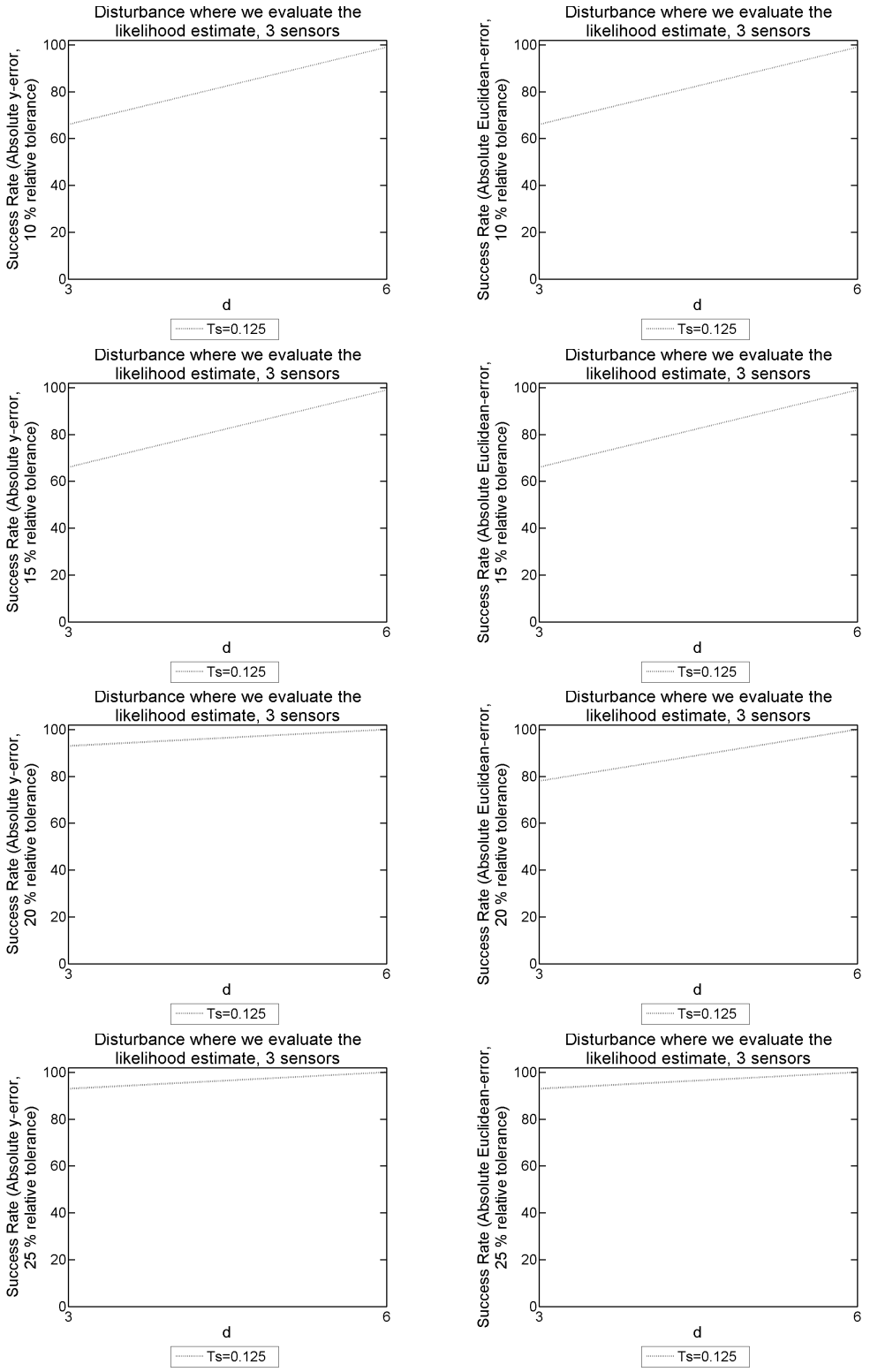


Figure 9.129: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 3 sensors present.

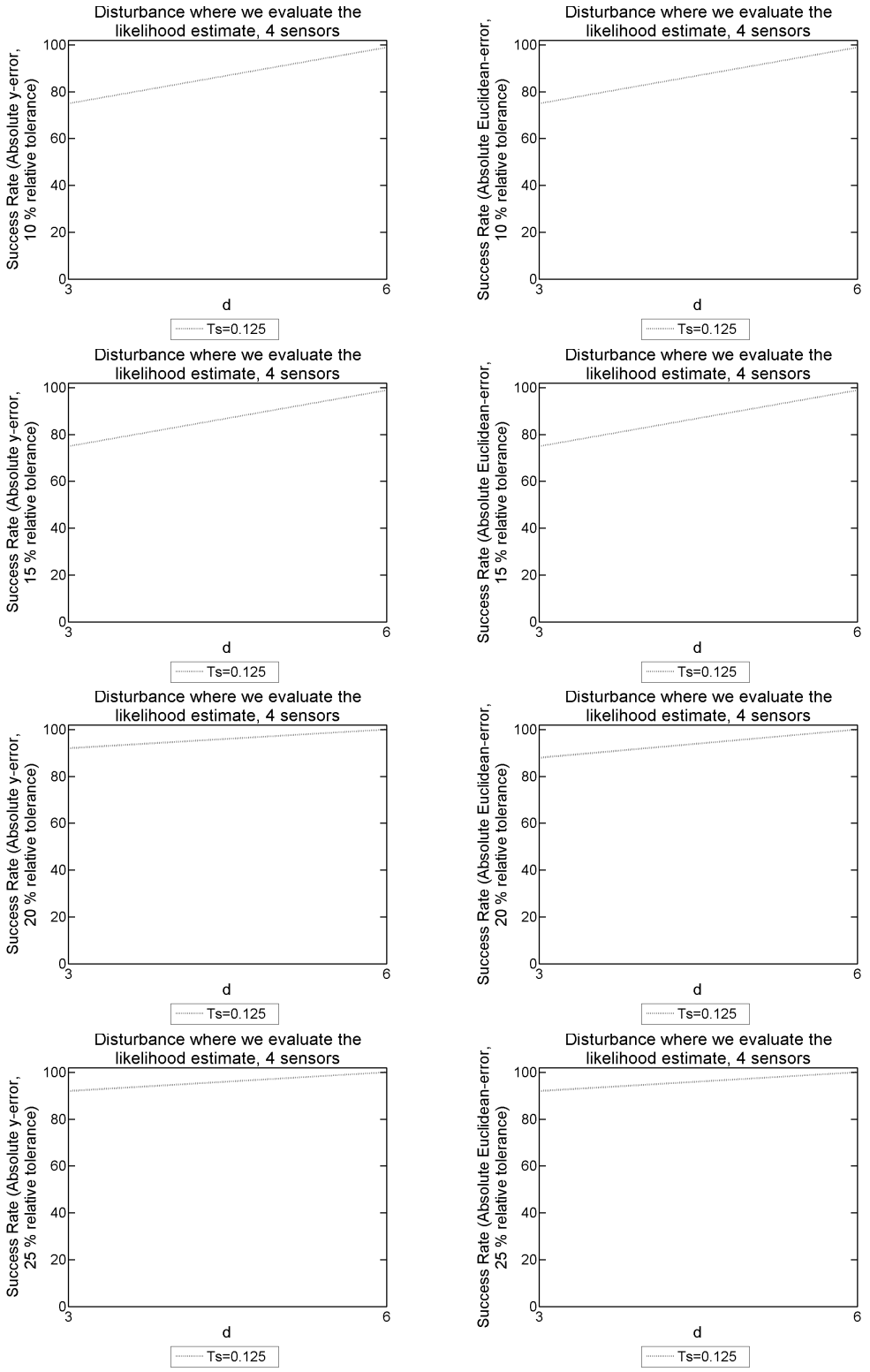


Figure 9.130: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 4 sensors present.

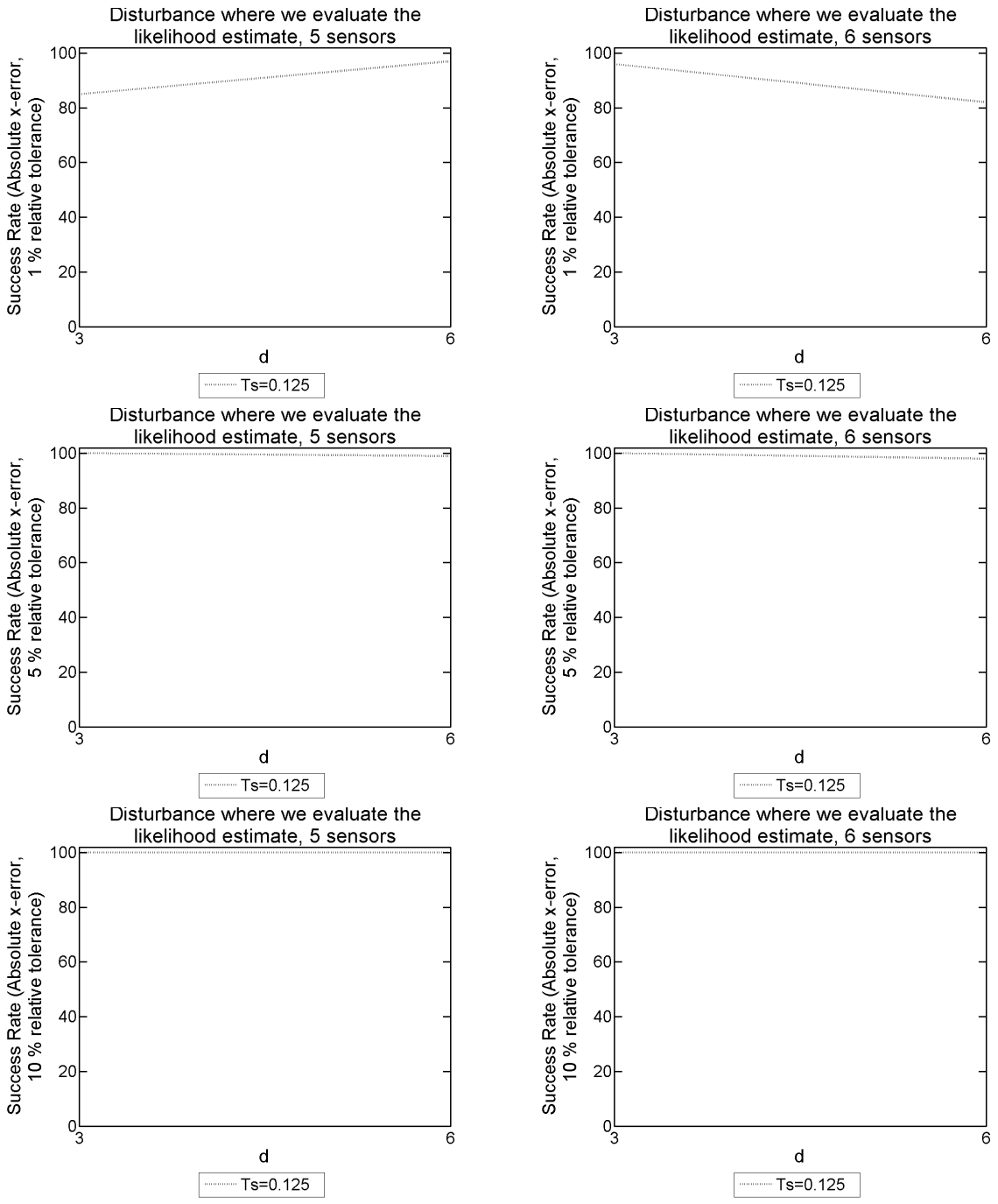


Figure 9.131: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

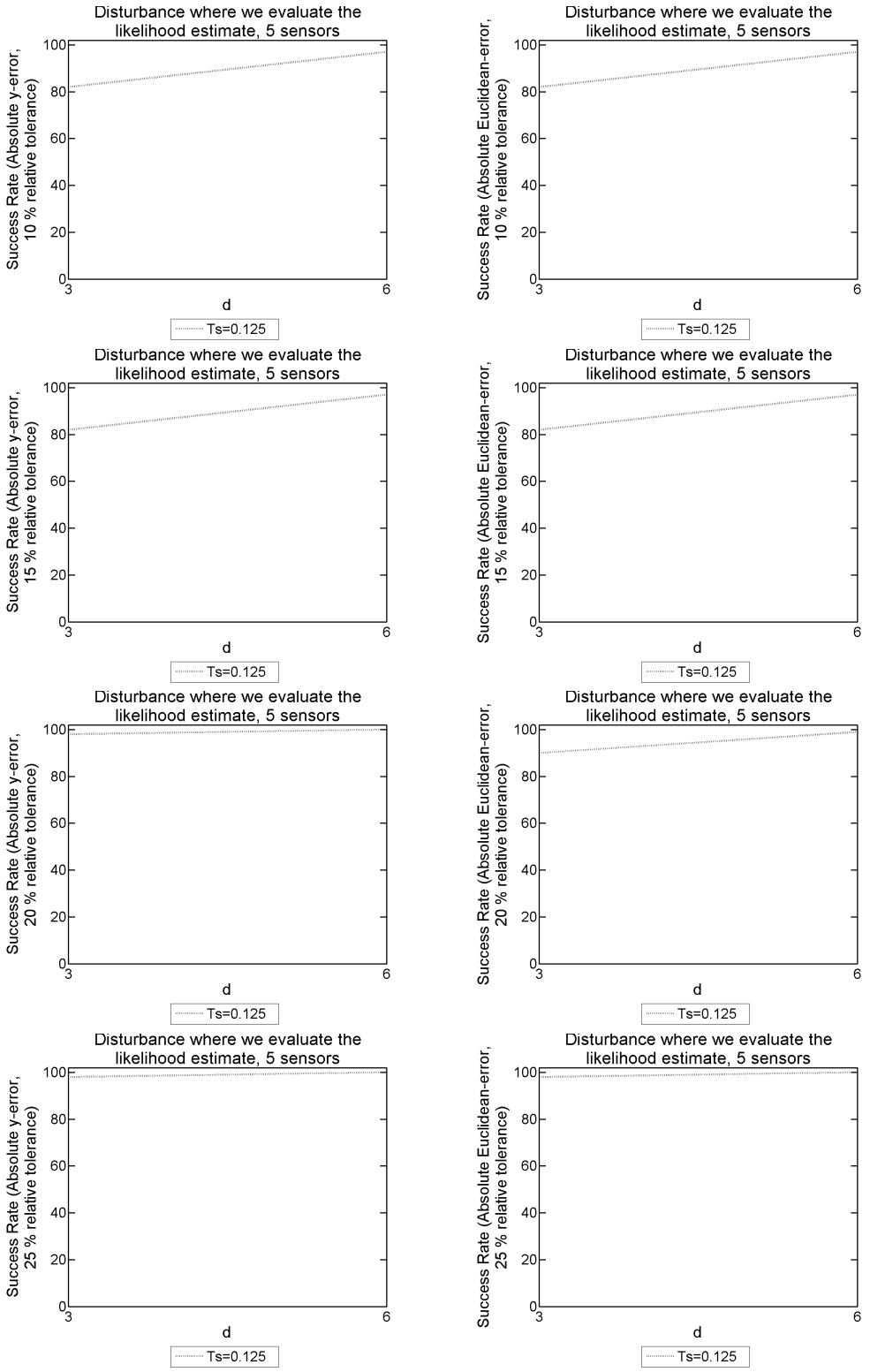


Figure 9.132: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 5 sensors present.

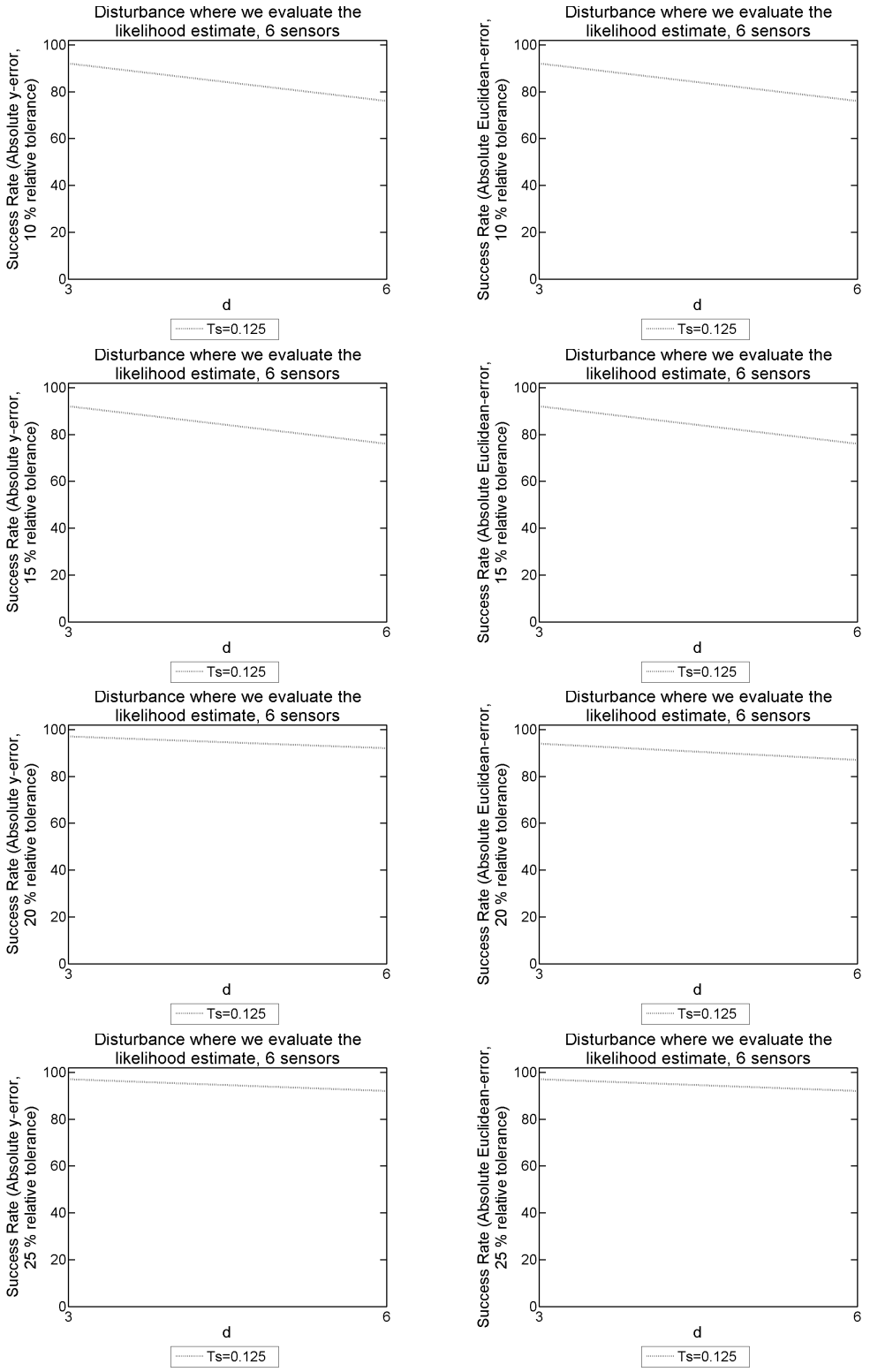


Figure 9.133: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 disturbance locations positioned where the likelihood function is evaluated, with 6 sensors present.

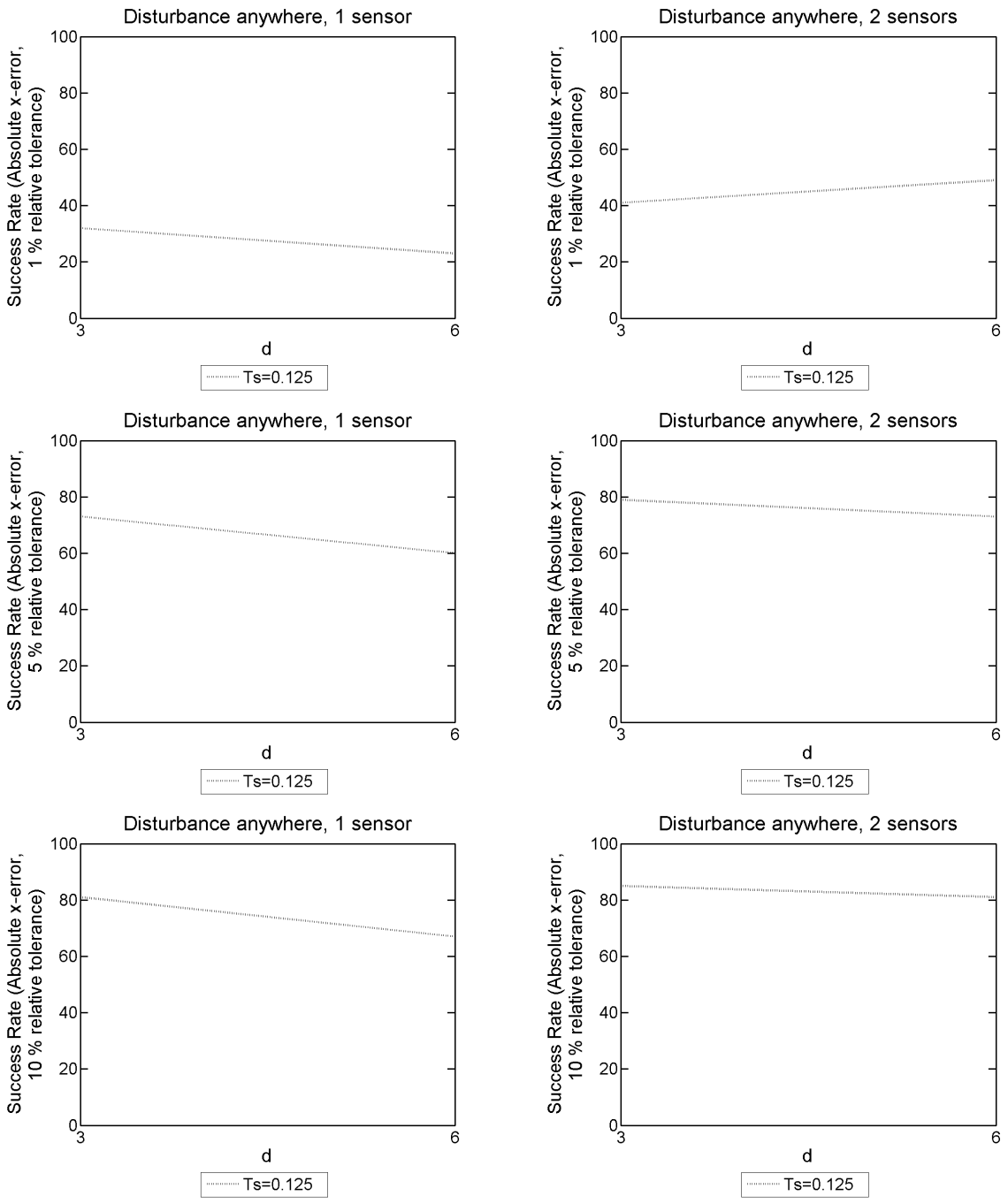


Figure 9.134: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 1 sensor present, whereas on the RHS there are 2 sensors present.

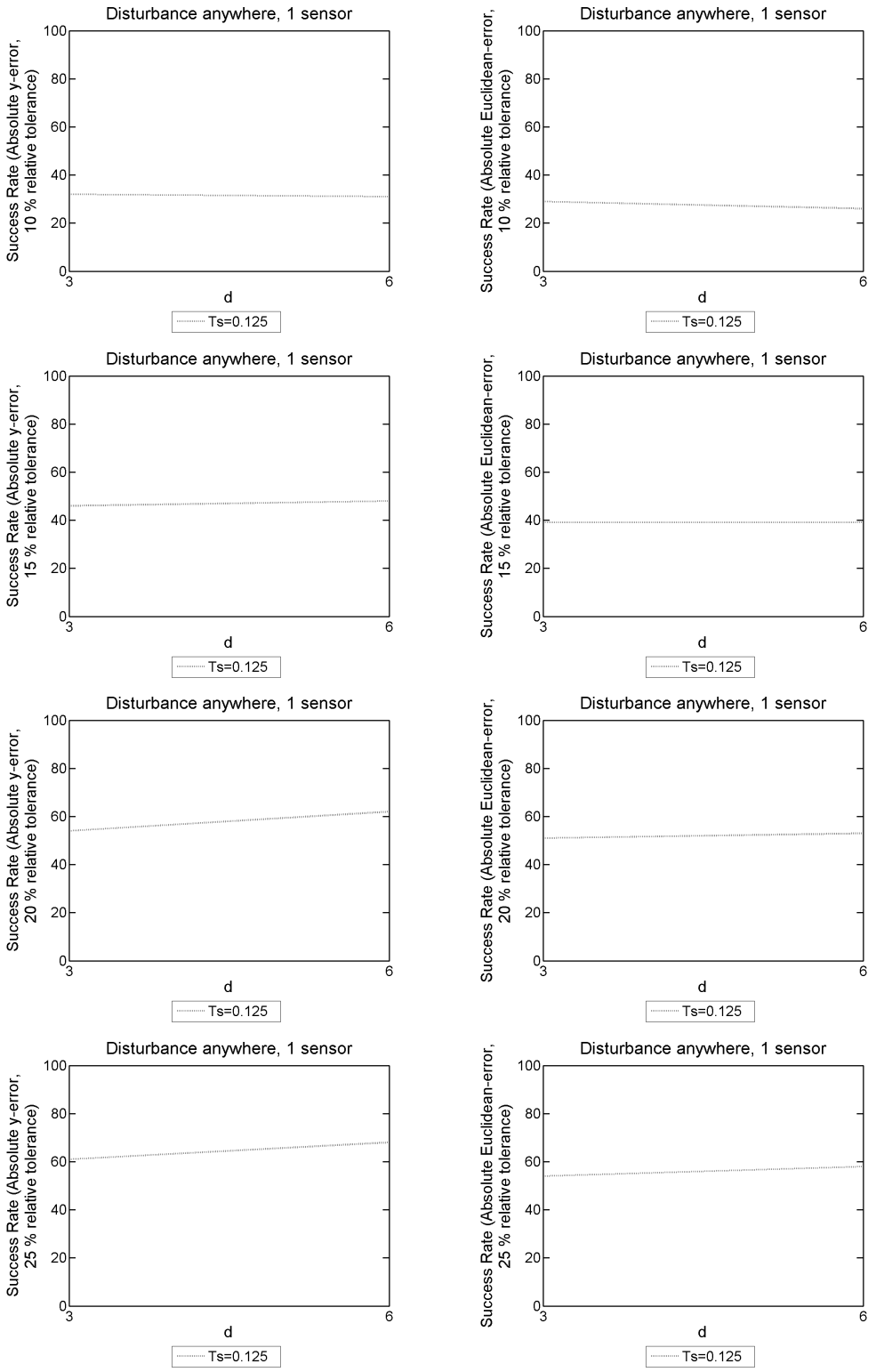


Figure 9.135: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 1 sensor present.

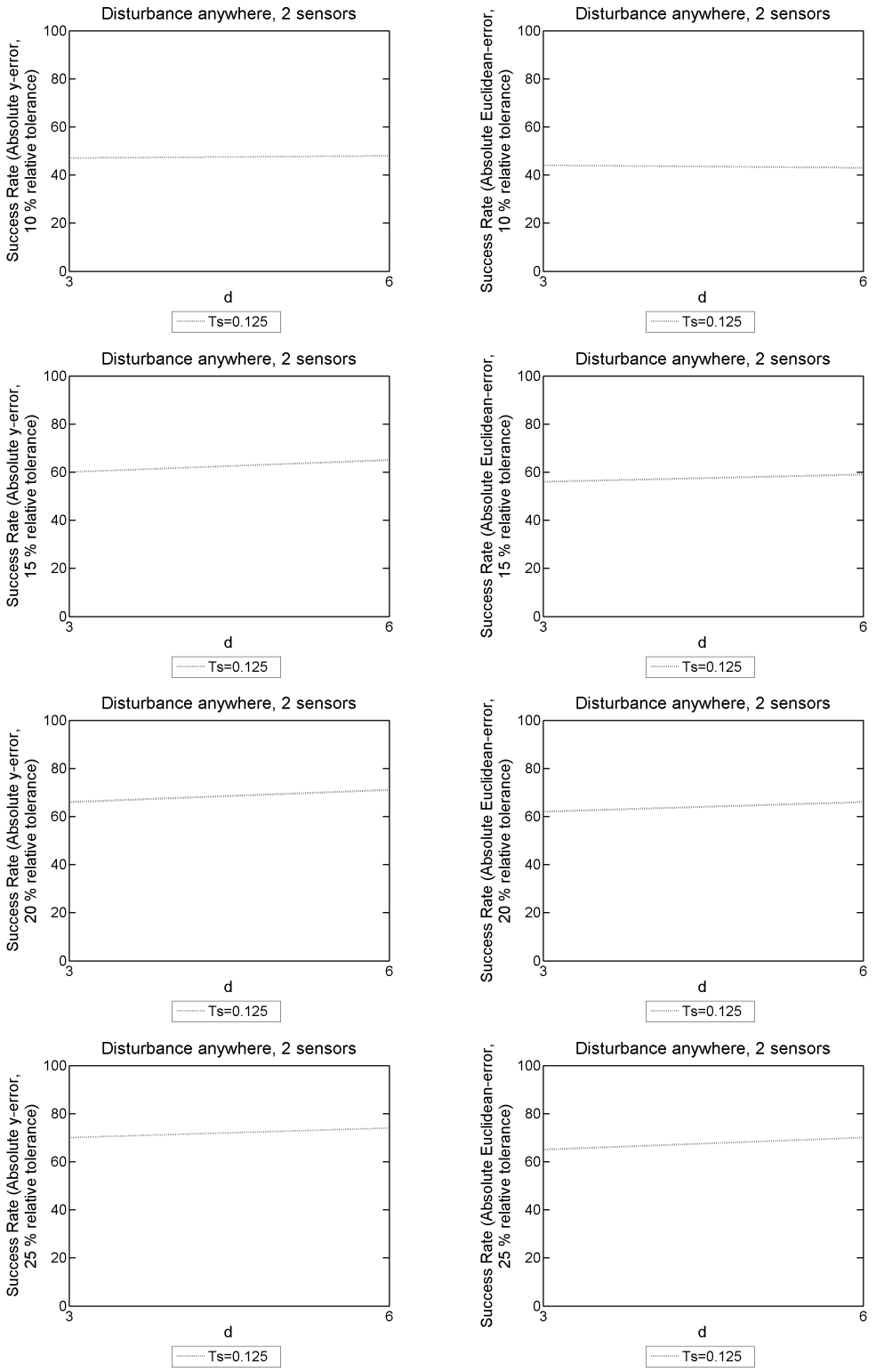


Figure 9.136: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 2 sensors present.

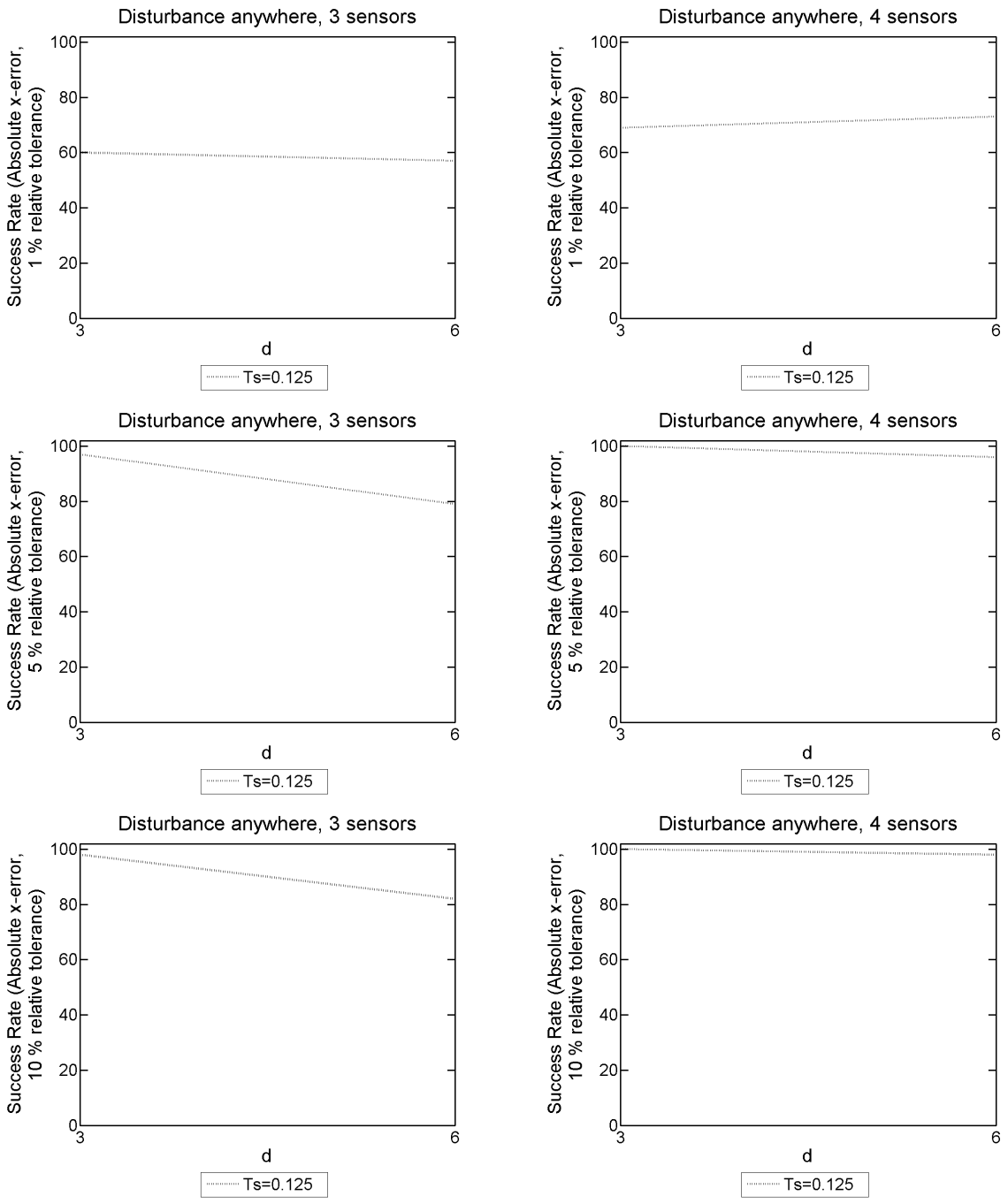


Figure 9.137: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 3 sensors present, whereas on the RHS there are 4 sensors present.

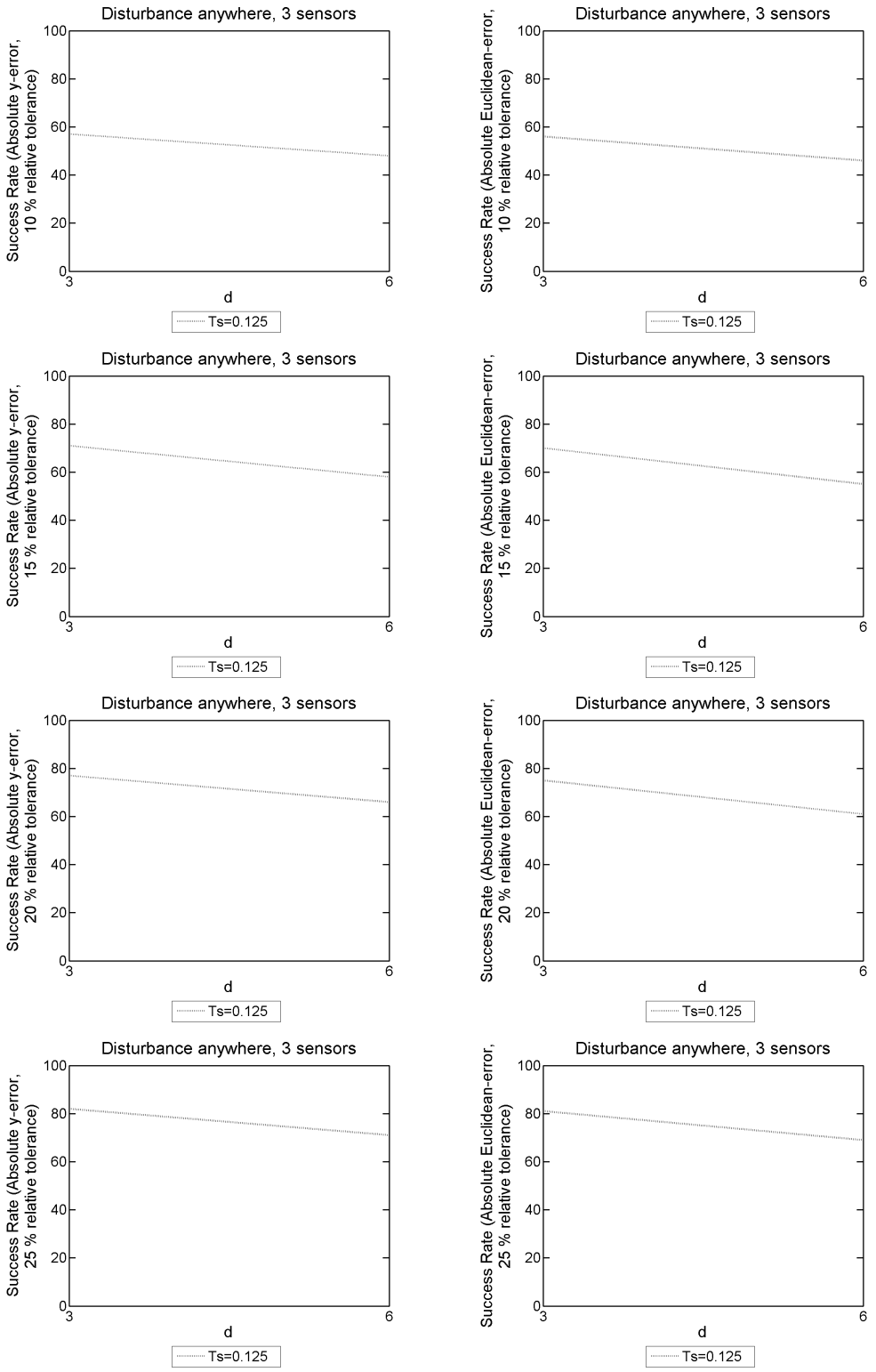


Figure 9.138: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 3 sensors present.

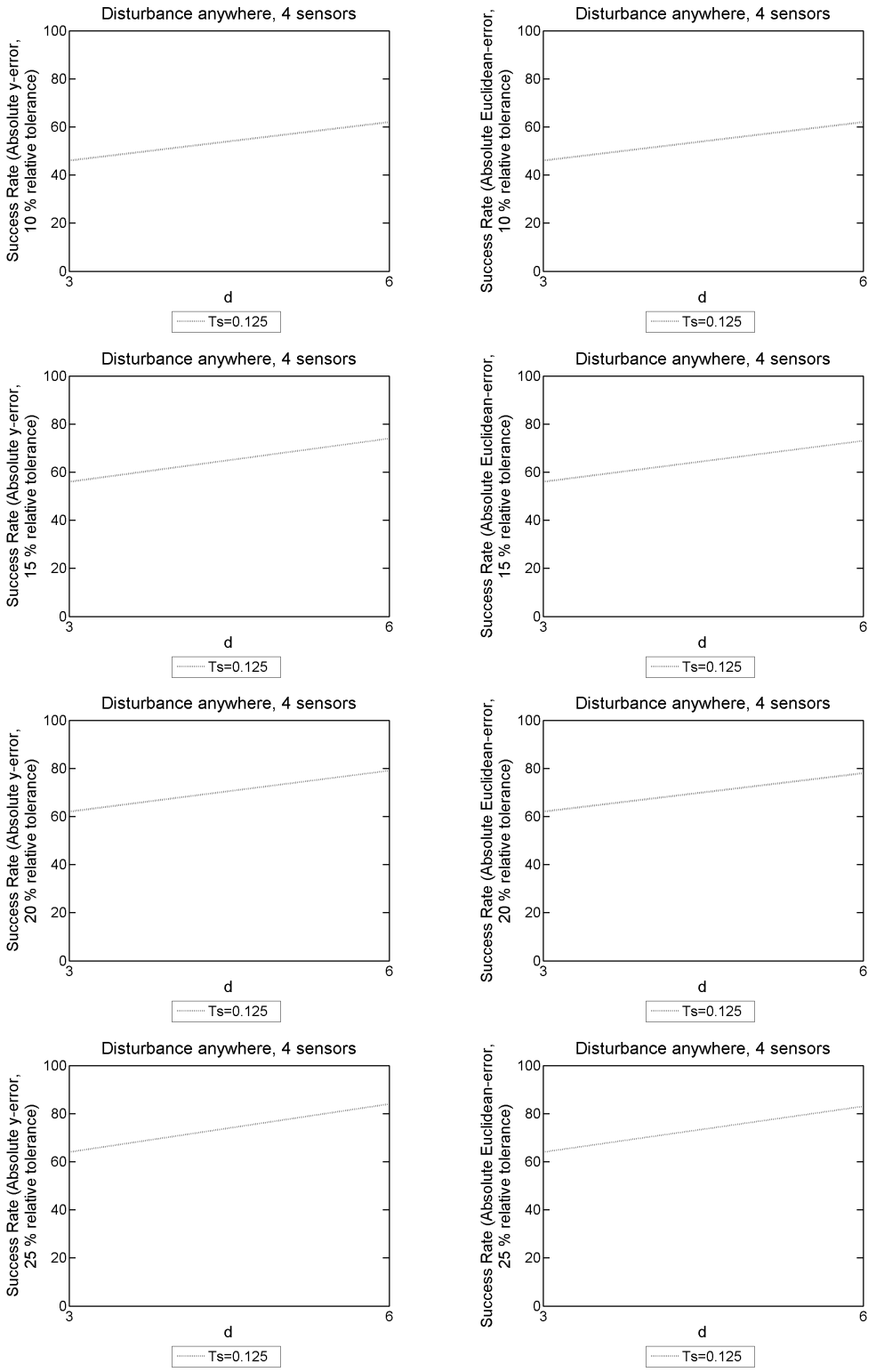


Figure 9.139: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 4 sensors present.

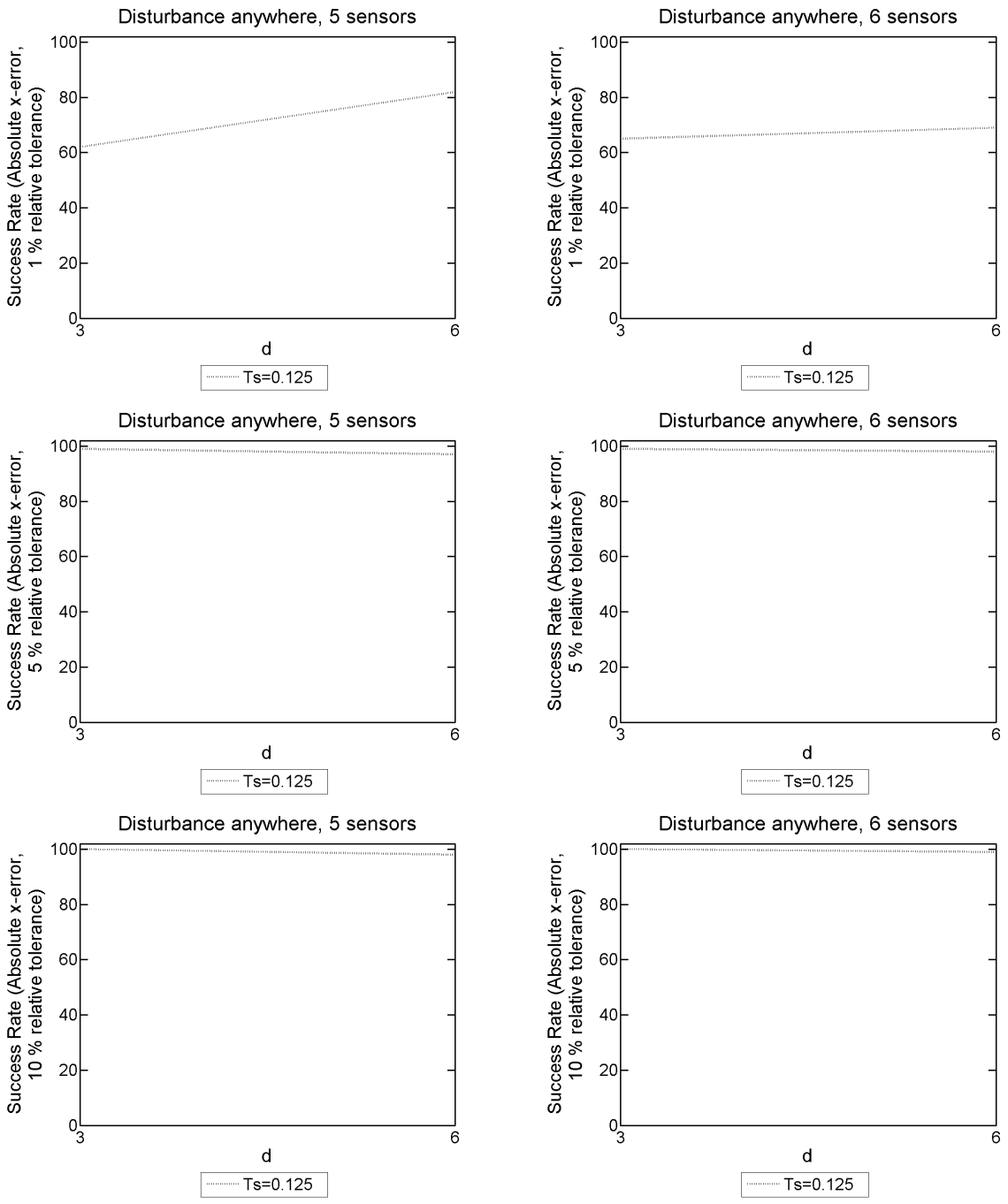


Figure 9.140: 2D model problem with 3NBCs: The success rate given $|x\text{-error}|$ for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations. The results on the LHS have 5 sensors present, whereas on the RHS there are 6 sensors present.

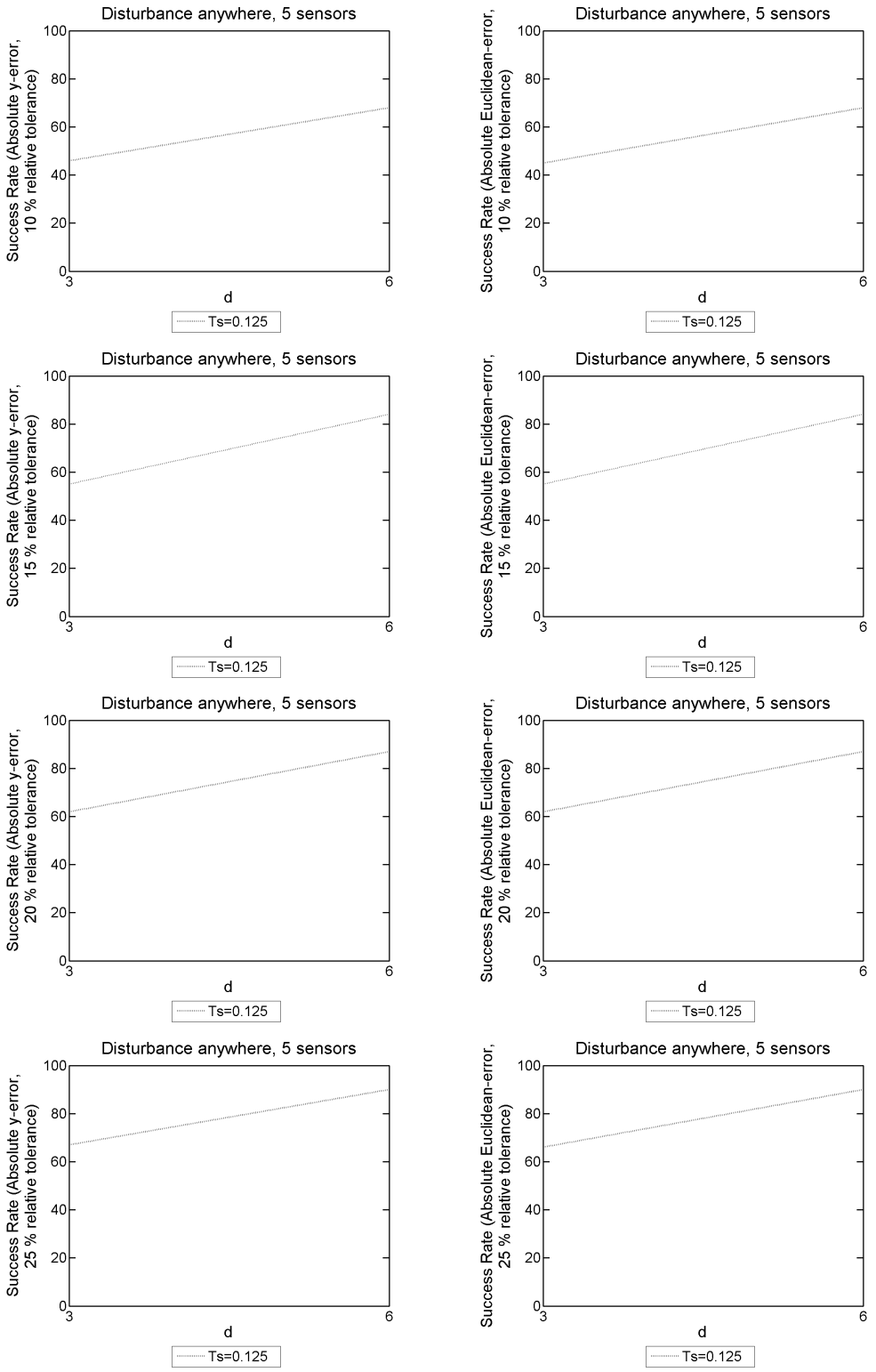


Figure 9.141: 2D model problem with 3NBCs: The success rate given $|y\text{-error}|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 5 sensors present.

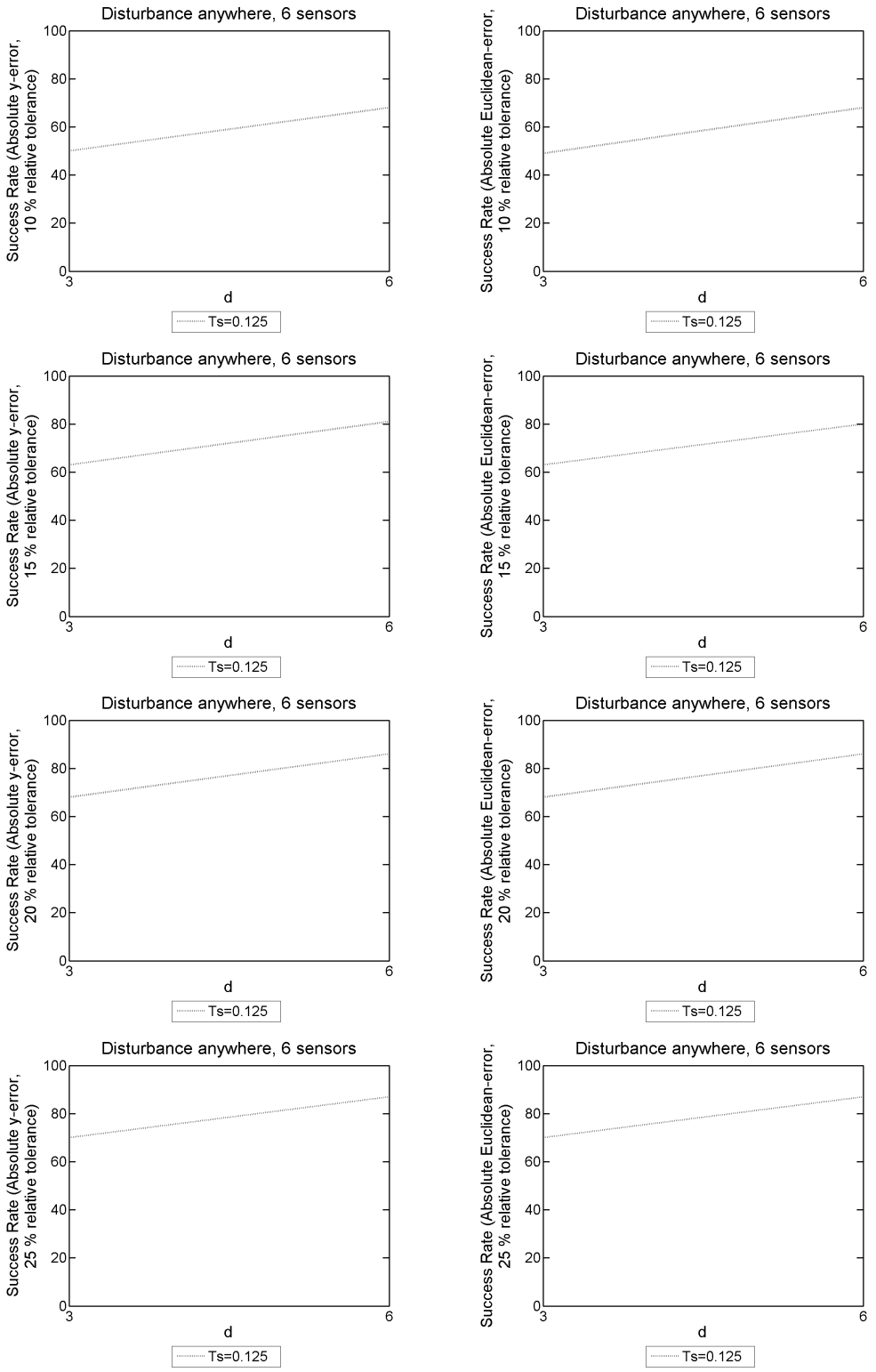


Figure 9.142: 2D model problem with 3NBCs: The success rate given $|y$-error$|$ on the LHS, and |Euclidean-error| on the RHS. We use an array of different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 500$, $M = 50$ and $L = 45000$, and $F = 300$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 100 random disturbance locations, with 6 sensors present.

Appendix E: Results for chapter 7

E.1 Using approximations of $\partial u/\partial t$ for sensor traces

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	28	44	48
1.000	6	32	42	42
1.000	12	44	48	48
1.000	24	42	46	46
0.500	3	36	42	46
0.500	6	38	46	46
0.500	12	48	52	52
0.500	24	46	52	52
0.250	3	38	50	50
0.250	6	30	44	46
0.250	12	46	58	58
0.250	24	42	50	52
0.125	3	38	48	52
0.125	6	34	44	46
0.125	12	50	60	64
0.125	24	46	58	58

Table 9.137: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	60	72	78
1.000	6	66	68	70
1.000	12	48	56	56
1.000	24	62	70	74
0.500	3	70	78	80
0.500	6	52	60	68
0.500	12	44	54	56
0.500	24	50	60	62
0.250	3	60	68	70
0.250	6	66	72	74
0.250	12	50	66	70
0.250	24	56	66	70
0.125	3	56	66	66
0.125	6	58	74	76
0.125	12	54	68	68
0.125	24	62	66	74

Table 9.138: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	92	94	94
1.000	6	98	98	98
1.000	12	98	98	98
1.000	24	98	98	98
0.500	3	96	96	96
0.500	6	100	100	100
0.500	12	98	98	98
0.500	24	98	98	98
0.250	3	92	92	92
0.250	6	98	98	98
0.250	12	92	98	98
0.250	24	98	98	98
0.125	3	86	88	90
0.125	6	96	96	96
0.125	12	96	98	98
0.125	24	96	100	100

Table 9.139: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	98	98	98
0.500	6	98	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	98	100	100
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	94	98	98
0.125	6	98	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.140: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	54	58	60
1.000	6	98	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	64	74	76
0.500	6	98	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	70	74	76
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	70	82	82
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.141: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	100	100	100
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	98	98	98
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	94	94	94
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.142: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	30	46	48
1.000	6	32	46	46
1.000	12	54	62	62
1.000	24	44	60	60
0.500	3	20	38	48
0.500	6	38	52	52
0.500	12	42	64	66
0.500	24	34	50	50
0.250	3	26	46	50
0.250	6	36	56	58
0.250	12	40	58	58
0.250	24	40	46	48
0.125	3	28	36	38
0.125	6	26	38	40
0.125	12	32	48	50
0.125	24	52	54	56

Table 9.143: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 1 sensor present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	38	50	52
1.000	6	54	64	64
1.000	12	36	62	64
1.000	24	38	58	60
0.500	3	38	46	54
0.500	6	50	58	58
0.500	12	42	56	62
0.500	24	40	60	64
0.250	3	38	58	58
0.250	6	42	48	56
0.250	12	28	42	46
0.250	24	52	64	66
0.125	3	34	50	52
0.125	6	30	52	54
0.125	12	36	56	56
0.125	24	34	54	58

Table 9.144: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 2 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	70	74	74
1.000	6	74	84	84
1.000	12	82	88	90
1.000	24	86	90	90
0.500	3	58	70	72
0.500	6	80	84	86
0.500	12	76	84	86
0.500	24	86	94	94
0.250	3	50	54	54
0.250	6	78	84	84
0.250	12	82	92	92
0.250	24	82	86	86
0.125	3	66	76	76
0.125	6	78	88	88
0.125	12	82	84	88
0.125	24	78	84	84

Table 9.145: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 3 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	70	78	78
1.000	6	82	84	84
1.000	12	98	98	98
1.000	24	94	96	96
0.500	3	70	76	80
0.500	6	96	96	96
0.500	12	98	98	98
0.500	24	90	94	94
0.250	3	66	66	68
0.250	6	90	96	96
0.250	12	94	98	98
0.250	24	98	100	100
0.125	3	68	76	80
0.125	6	84	94	94
0.125	12	90	92	92
0.125	24	90	90	90

Table 9.146: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 4 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	38	46	50
1.000	6	98	98	98
1.000	12	96	100	100
1.000	24	96	100	100
0.500	3	42	54	58
0.500	6	92	100	100
0.500	12	96	100	100
0.500	24	92	98	98
0.250	3	48	56	58
0.250	6	84	96	96
0.250	12	100	100	100
0.250	24	96	100	100
0.125	3	54	56	60
0.125	6	88	98	98
0.125	12	96	98	98
0.125	24	98	100	100

Table 9.147: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 5 sensors present to record the approximations of $\partial u/\partial t$ only.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	66	74	76
1.000	6	98	100	100
1.000	12	96	100	100
1.000	24	98	100	100
0.500	3	72	76	84
0.500	6	88	96	96
0.500	12	98	100	100
0.500	24	100	100	100
0.250	3	64	66	68
0.250	6	96	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	72	74	76
0.125	6	90	98	98
0.125	12	98	98	98
0.125	24	96	98	98

Table 9.148: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 6 sensors present to record the approximations of $\partial u/\partial t$ only.

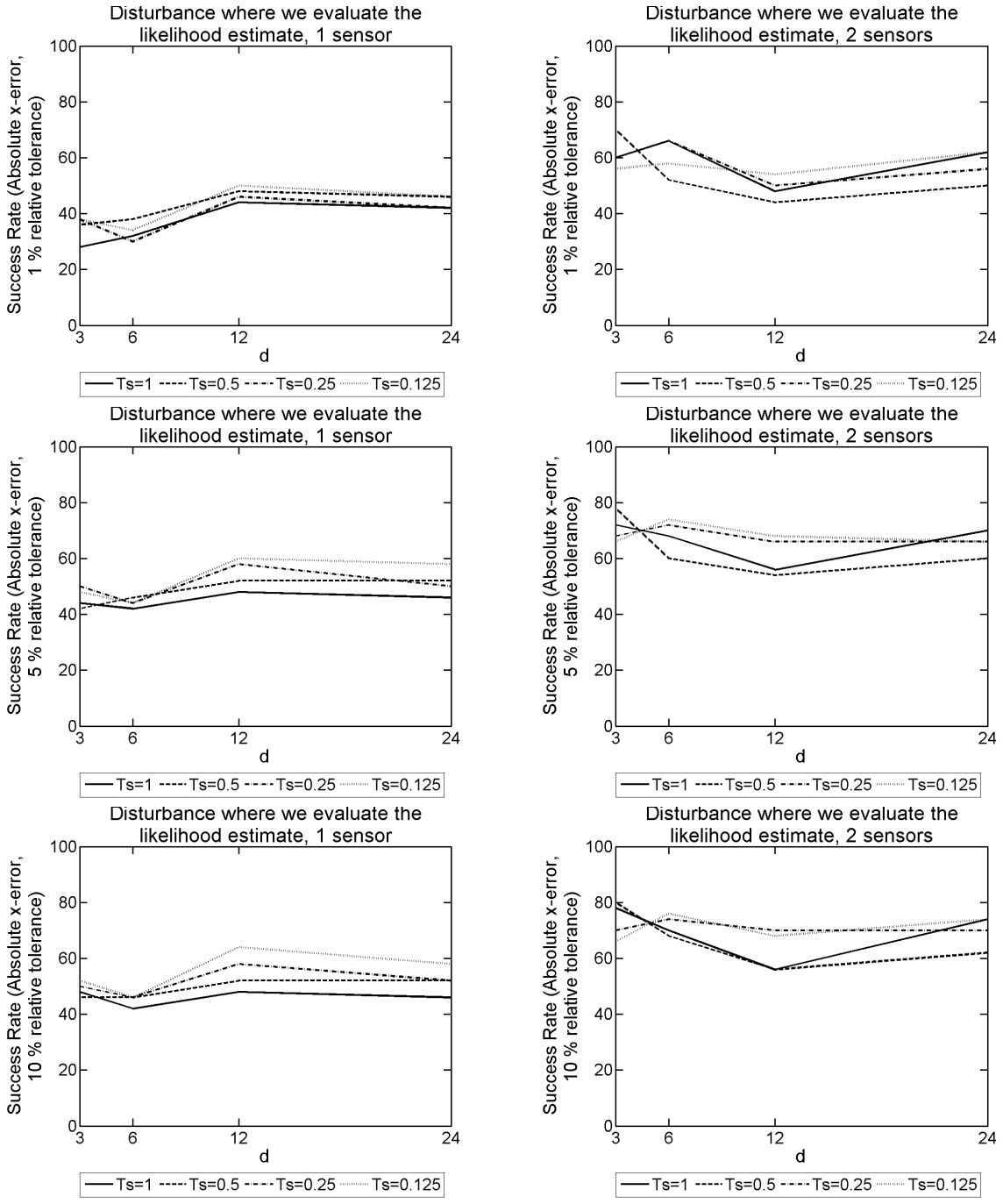


Figure 9.143: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 1 sensor present, and on the right-hand side 2 sensors were present with approximations of $\partial u/\partial t$ only.

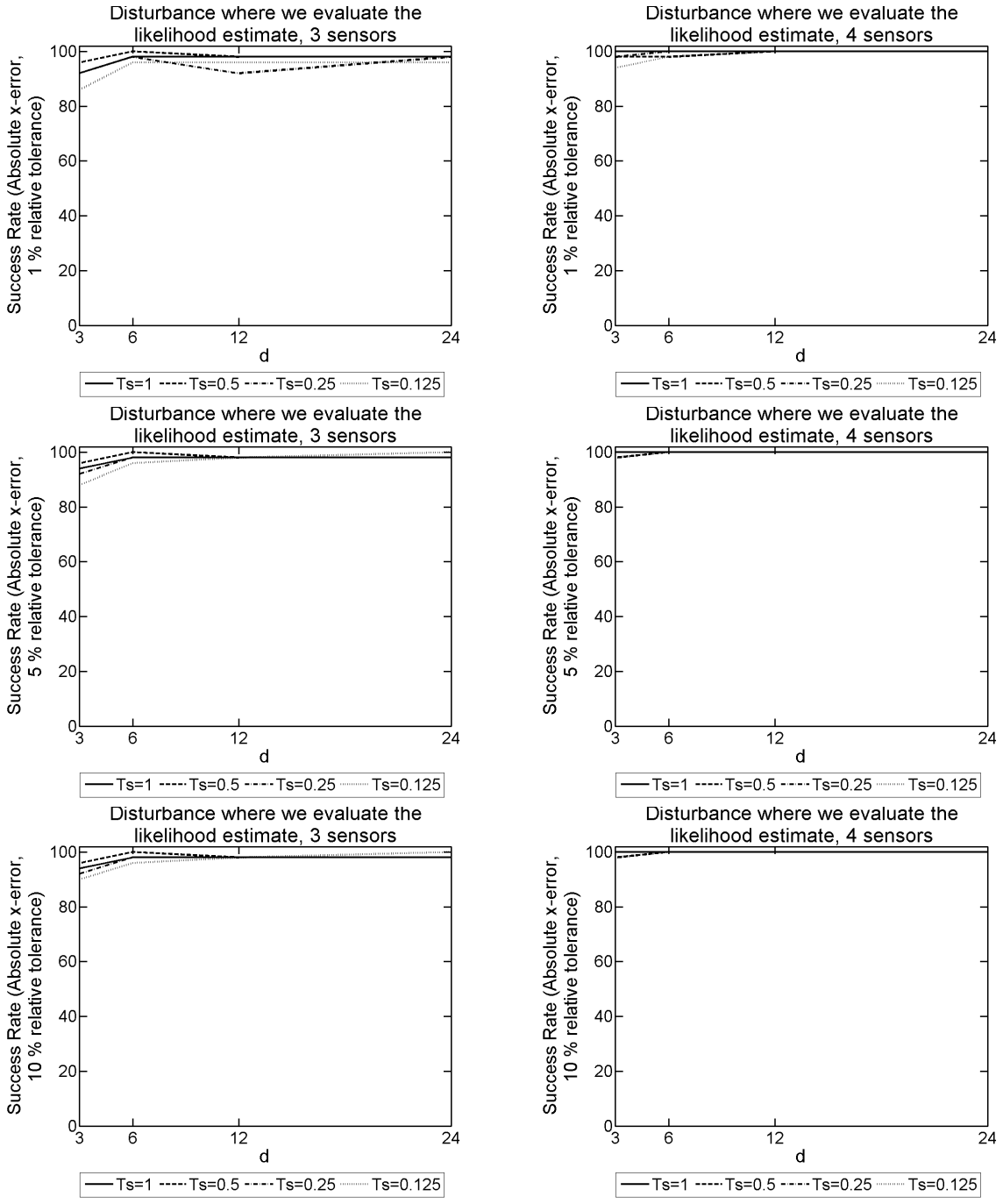


Figure 9.144: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 3 sensors present, and on the right-hand side 4 sensors were present with approximations of $\partial u/\partial t$ only.

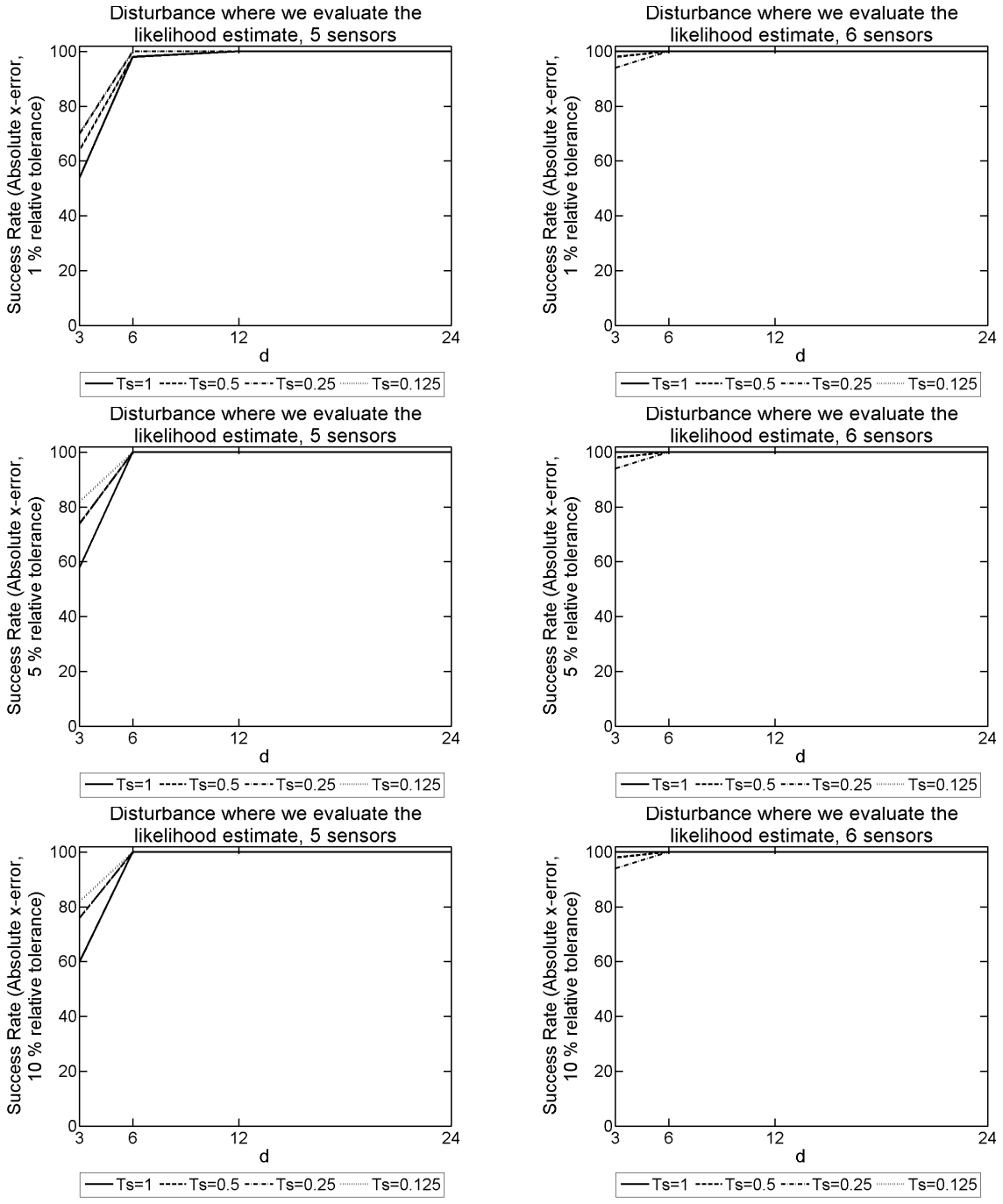


Figure 9.145: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 5 sensors present, and on the right-hand side 6 sensors were present with approximations of $\partial u/\partial t$ only.

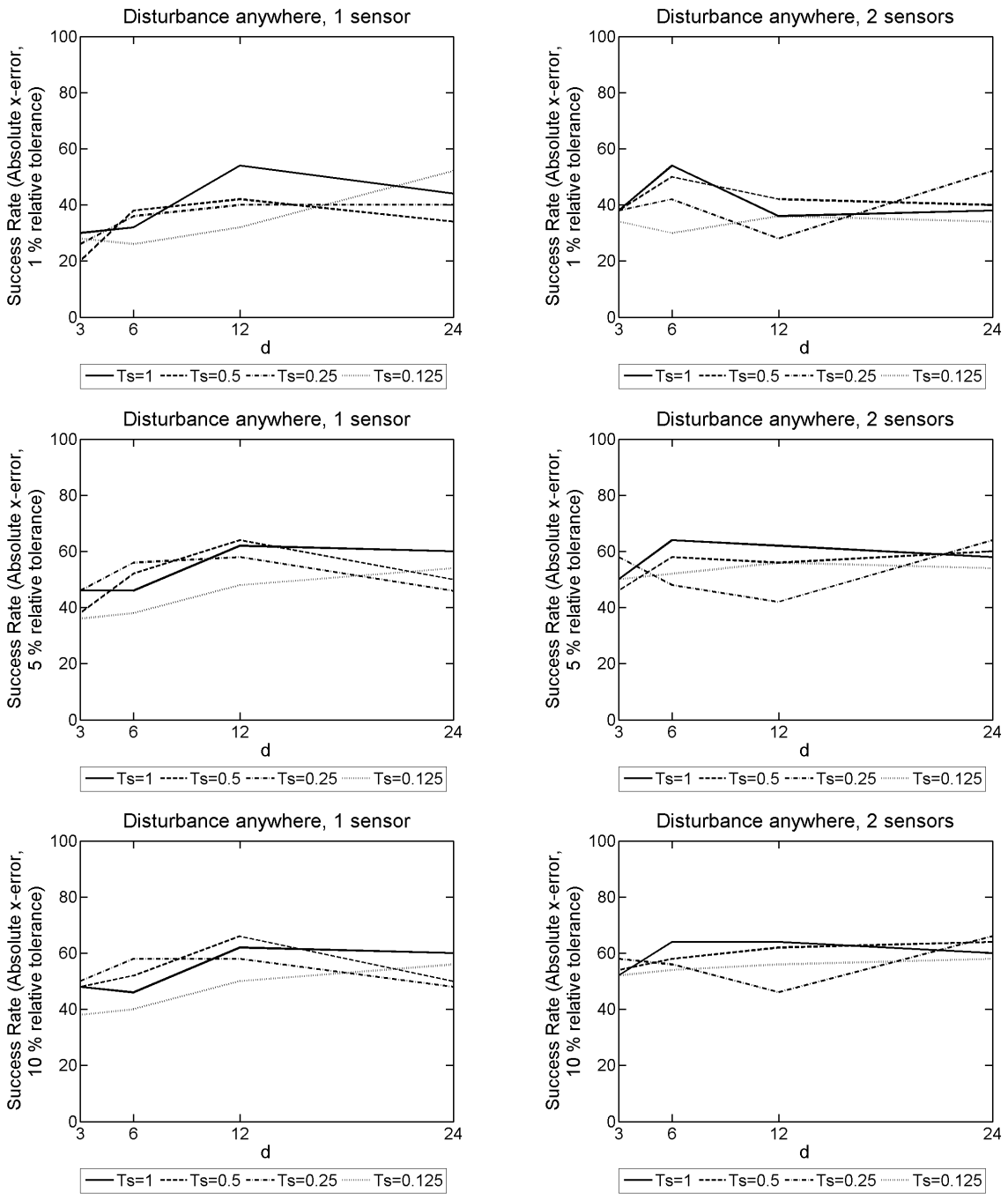


Figure 9.146: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations. The results on the left-hand side have 1 sensor present, and on the right-hand side 2 sensors were present with approximations of $\partial u/\partial t$ only.

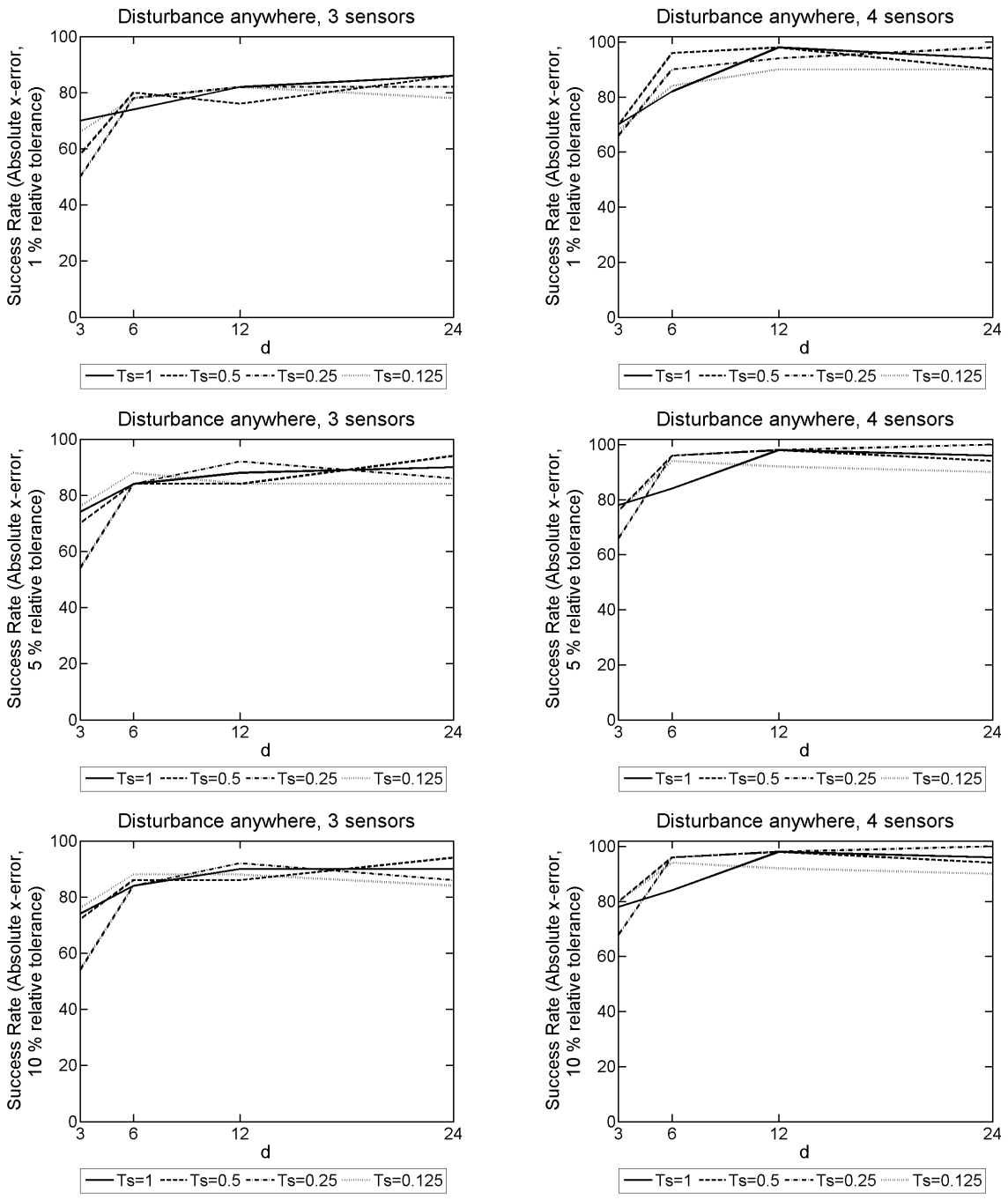


Figure 9.147: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations. The results on the left-hand side have 3 sensors present, and on the right-hand side 4 sensors were present with approximations of $\partial u/\partial t$ only.

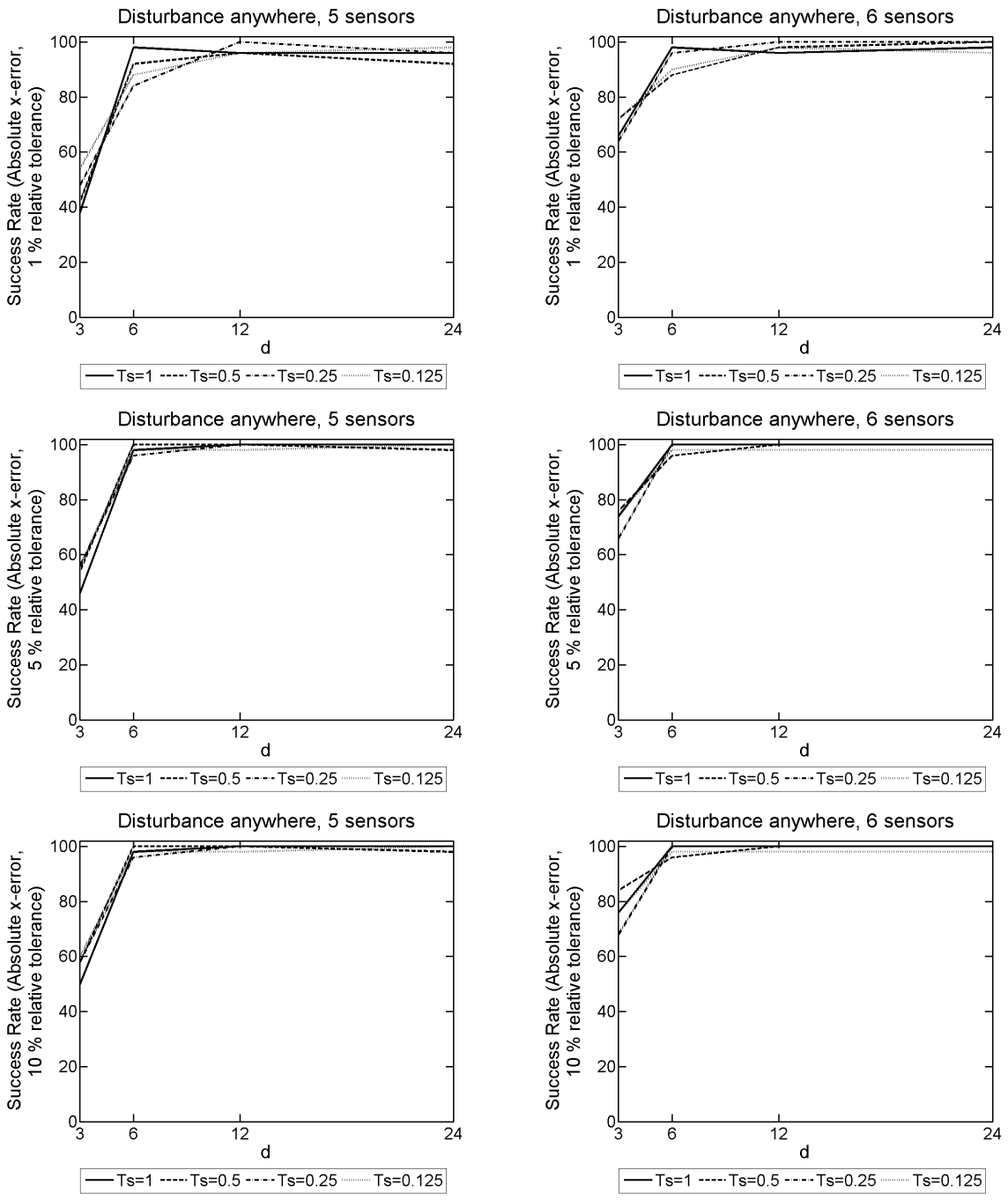


Figure 9.148: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations. The results on the left-hand side have 5 sensors present, and on the right-hand side 6 sensors were present with approximations of $\partial u/\partial t$ only.

E.2 Using approximations of u and $\partial u/\partial t$ for sensor traces

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	44	56	60
1.000	6	32	36	38
1.000	12	36	42	44
1.000	24	50	56	58
0.500	3	22	36	40
0.500	6	38	48	48
0.500	12	50	60	60
0.500	24	52	58	60
0.250	3	42	50	50
0.250	6	36	52	52
0.250	12	56	66	66
0.250	24	52	66	66
0.125	3	30	54	54
0.125	6	42	58	58
0.125	12	38	52	54
0.125	24	44	52	56

Table 9.149: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 1 sensor present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	74	76	78
1.000	6	50	60	64
1.000	12	58	68	70
1.000	24	62	68	70
0.500	3	76	80	80
0.500	6	60	66	66
0.500	12	54	60	62
0.500	24	52	62	64
0.250	3	58	68	68
0.250	6	48	54	58
0.250	12	46	62	62
0.250	24	56	66	70
0.125	3	44	64	64
0.125	6	60	70	72
0.125	12	48	60	62
0.125	24	62	68	74

Table 9.150: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 2 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	94	96	96
1.000	6	98	98	98
1.000	12	98	100	100
1.000	24	96	100	100
0.500	3	96	96	96
0.500	6	98	98	98
0.500	12	94	94	94
0.500	24	94	96	96
0.250	3	94	94	94
0.250	6	96	96	96
0.250	12	98	98	100
0.250	24	98	98	98
0.125	3	96	98	100
0.125	6	96	98	98
0.125	12	98	100	100
0.125	24	98	98	98

Table 9.151: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 3 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	96	96	96
1.000	6	100	100	100
1.000	12	98	100	100
1.000	24	96	98	98
0.500	3	96	100	100
0.500	6	100	100	100
0.500	12	100	100	100
0.500	24	98	100	100
0.250	3	94	94	94
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	100	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.152: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 4 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	58	64	68
1.000	6	98	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	52	64	68
0.500	6	96	98	98
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	70	74	74
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	64	70	70
0.125	6	98	98	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.153: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 5 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	98	98	98
1.000	6	100	100	100
1.000	12	100	100	100
1.000	24	100	100	100
0.500	3	100	100	100
0.500	6	98	100	100
0.500	12	100	100	100
0.500	24	100	100	100
0.250	3	98	98	98
0.250	6	100	100	100
0.250	12	100	100	100
0.250	24	100	100	100
0.125	3	100	100	100
0.125	6	98	100	100
0.125	12	100	100	100
0.125	24	100	100	100

Table 9.154: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated, and 6 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	24	38	38
1.000	6	36	42	42
1.000	12	38	46	48
1.000	24	42	60	62
0.500	3	20	44	54
0.500	6	32	52	52
0.500	12	32	42	44
0.500	24	46	58	58
0.250	3	30	42	46
0.250	6	36	58	58
0.250	12	38	54	58
0.250	24	38	54	54
0.125	3	28	46	50
0.125	6	36	54	60
0.125	12	28	40	42
0.125	24	44	58	60

Table 9.155: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 1 sensor present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	36	46	52
1.000	6	36	52	54
1.000	12	38	58	58
1.000	24	40	62	64
0.500	3	44	48	56
0.500	6	36	52	58
0.500	12	42	54	56
0.500	24	42	60	62
0.250	3	42	54	54
0.250	6	48	62	62
0.250	12	44	56	56
0.250	24	40	58	60
0.125	3	46	60	62
0.125	6	38	60	62
0.125	12	36	52	54
0.125	24	48	68	68

Table 9.156: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 2 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	62	66	66
1.000	6	70	78	78
1.000	12	74	86	88
1.000	24	70	78	80
0.500	3	64	76	78
0.500	6	74	78	78
0.500	12	76	90	90
0.500	24	84	88	90
0.250	3	66	68	70
0.250	6	78	84	84
0.250	12	82	94	94
0.250	24	80	96	96
0.125	3	52	64	64
0.125	6	72	76	80
0.125	12	78	84	84
0.125	24	74	78	78

Table 9.157: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 3 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	68	74	76
1.000	6	90	90	90
1.000	12	96	100	100
1.000	24	98	98	100
0.500	3	78	84	84
0.500	6	88	92	92
0.500	12	94	98	98
0.500	24	88	96	96
0.250	3	58	66	66
0.250	6	90	98	98
0.250	12	88	96	98
0.250	24	92	96	98
0.125	3	60	70	72
0.125	6	92	94	94
0.125	12	96	100	100
0.125	24	92	96	96

Table 9.158: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 4 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	40	48	52
1.000	6	90	100	100
1.000	12	94	100	100
1.000	24	96	98	98
0.500	3	48	54	58
0.500	6	92	100	100
0.500	12	94	100	100
0.500	24	98	100	100
0.250	3	36	44	52
0.250	6	84	92	92
0.250	12	100	100	100
0.250	24	96	100	100
0.125	3	50	62	64
0.125	6	86	92	92
0.125	12	92	100	100
0.125	24	92	96	96

Table 9.159: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 5 sensors present to record the approximations of u and $\partial u/\partial t$.

		Success Rate		
Ts	d	1%	5%	10%
1.000	3	76	80	80
1.000	6	92	98	98
1.000	12	98	100	100
1.000	24	98	98	98
0.500	3	70	76	80
0.500	6	94	98	98
0.500	12	96	100	100
0.500	24	96	98	98
0.250	3	60	60	62
0.250	6	94	100	100
0.250	12	98	100	100
0.250	24	98	100	100
0.125	3	70	76	78
0.125	6	90	96	98
0.125	12	94	98	98
0.125	24	98	100	100

Table 9.160: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations, and 6 sensors present to record the approximations of u and $\partial u/\partial t$.

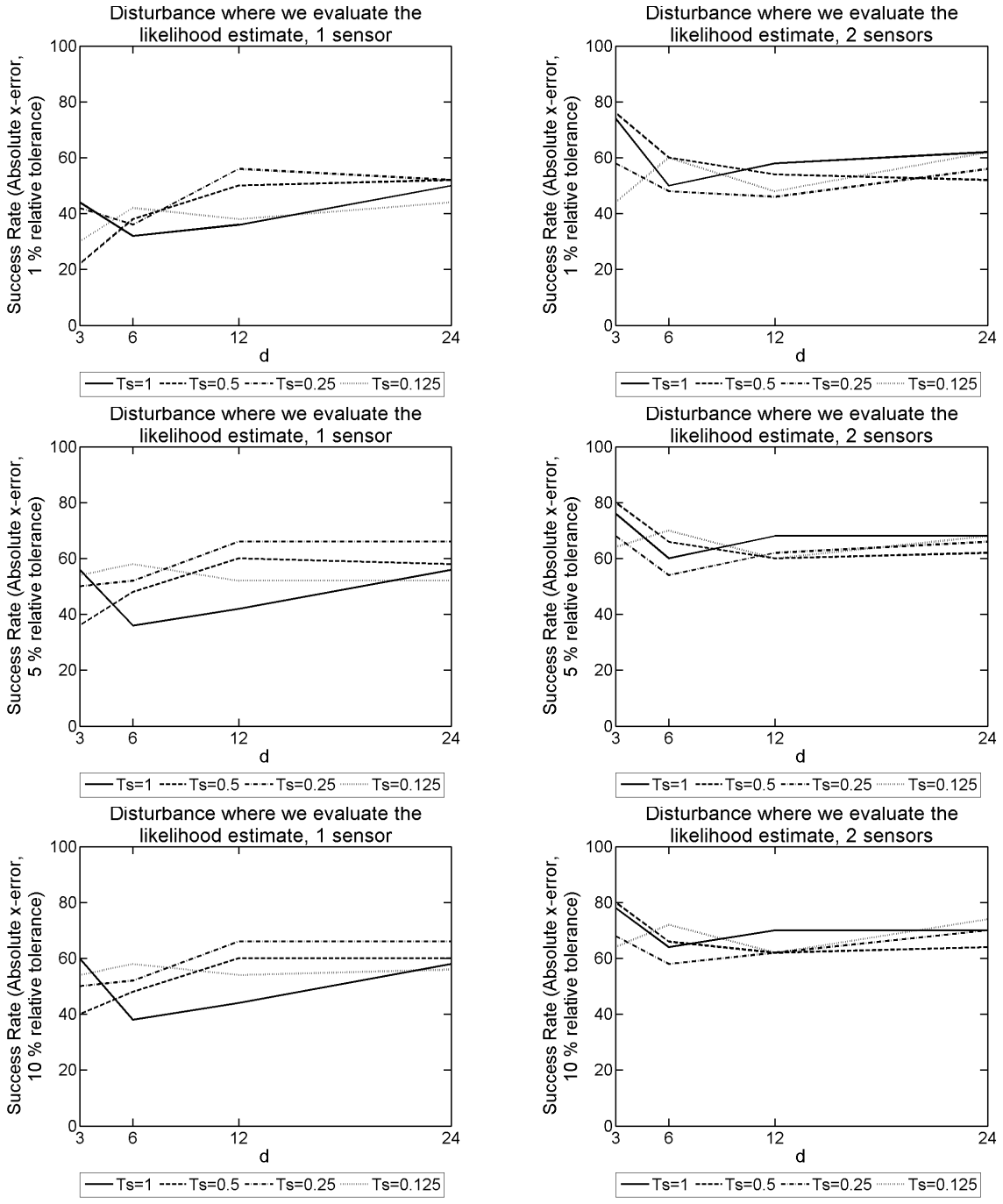


Figure 9.149: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 1 sensor present, and on the right-hand side 2 sensors were present with approximations of u and $\partial u/\partial t$.

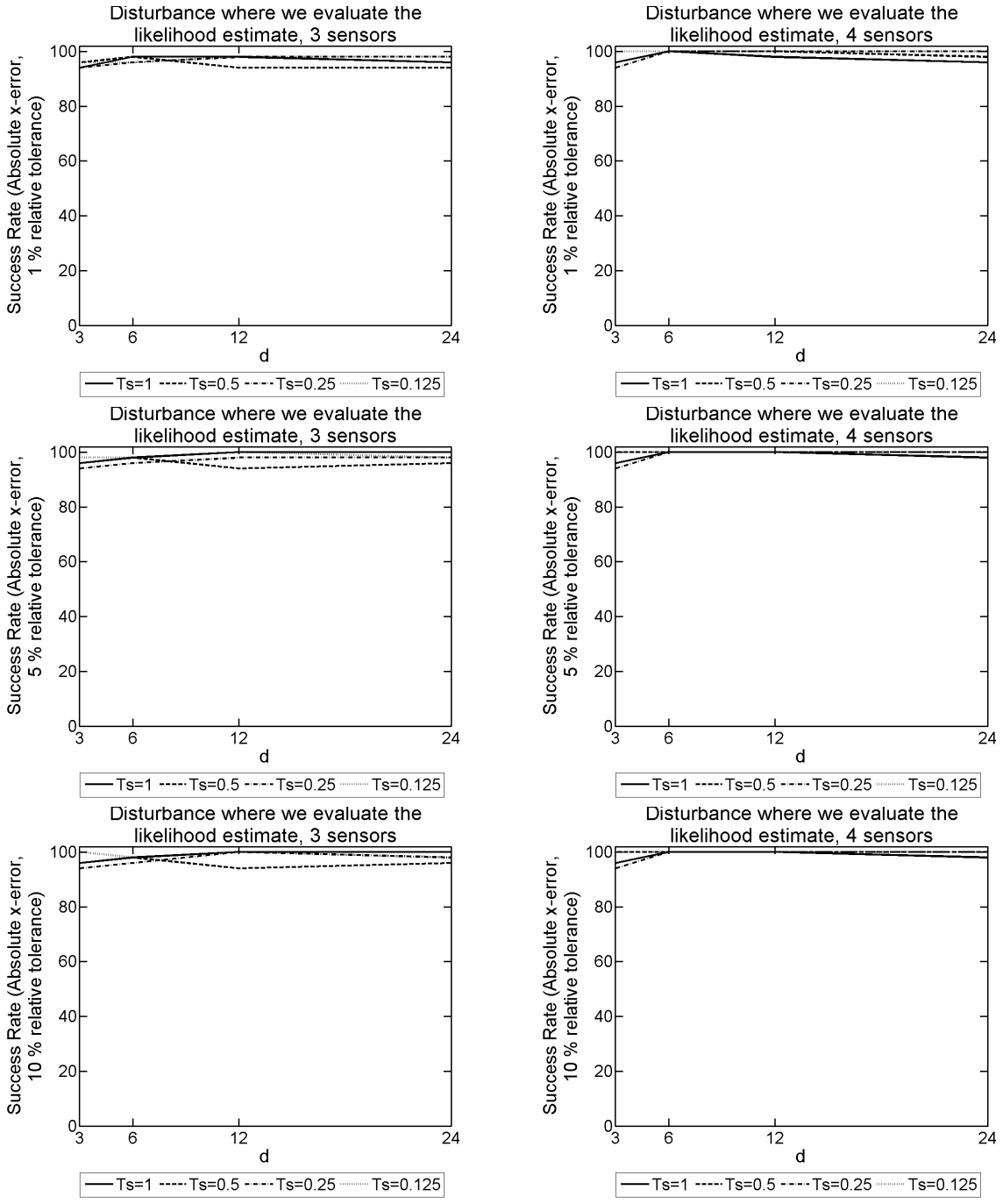


Figure 9.150: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 3 sensors present, and on the right-hand side 4 sensors were present with approximations of u and $\partial u/\partial t$.

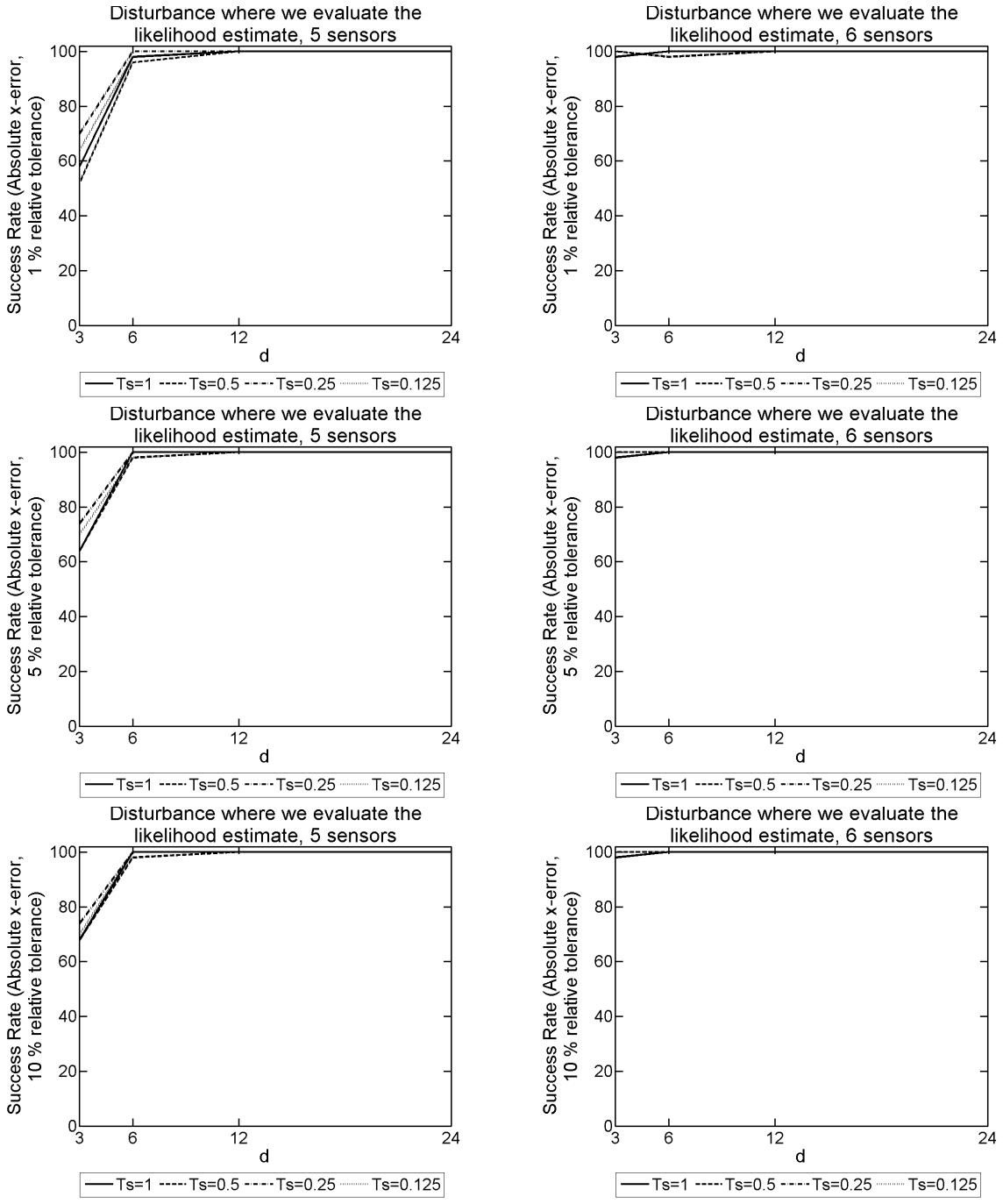


Figure 9.151: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 disturbance locations positioned where the likelihood function is evaluated. The results on the left-hand side have 5 sensors present, and on the right-hand side 6 sensors were present with approximations of u and $\partial u/\partial t$.

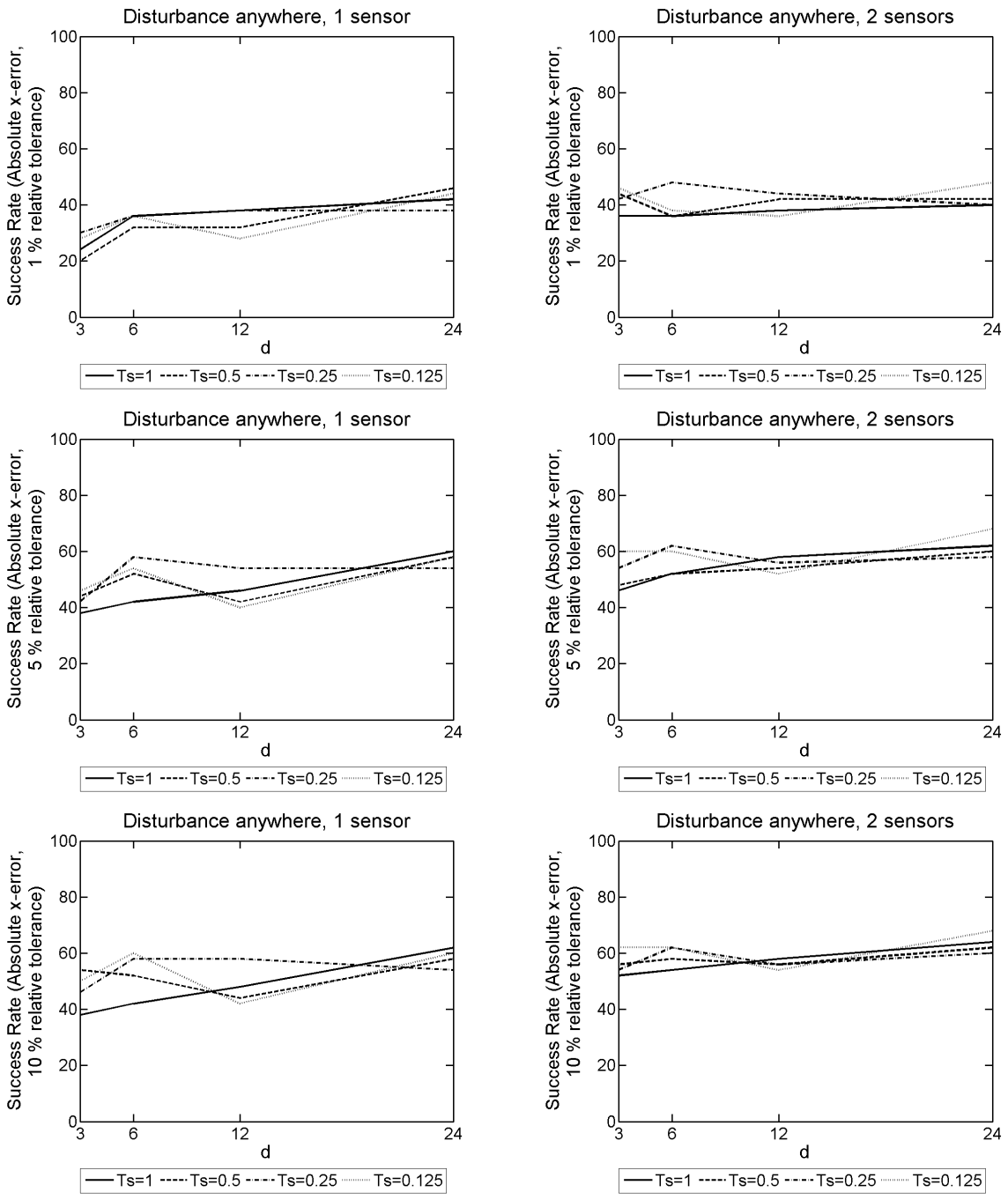


Figure 9.152: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations. The results on the left-hand side have 1 sensor present, and on the right-hand side 2 sensors were present with approximations of u and $\partial u/\partial t$.

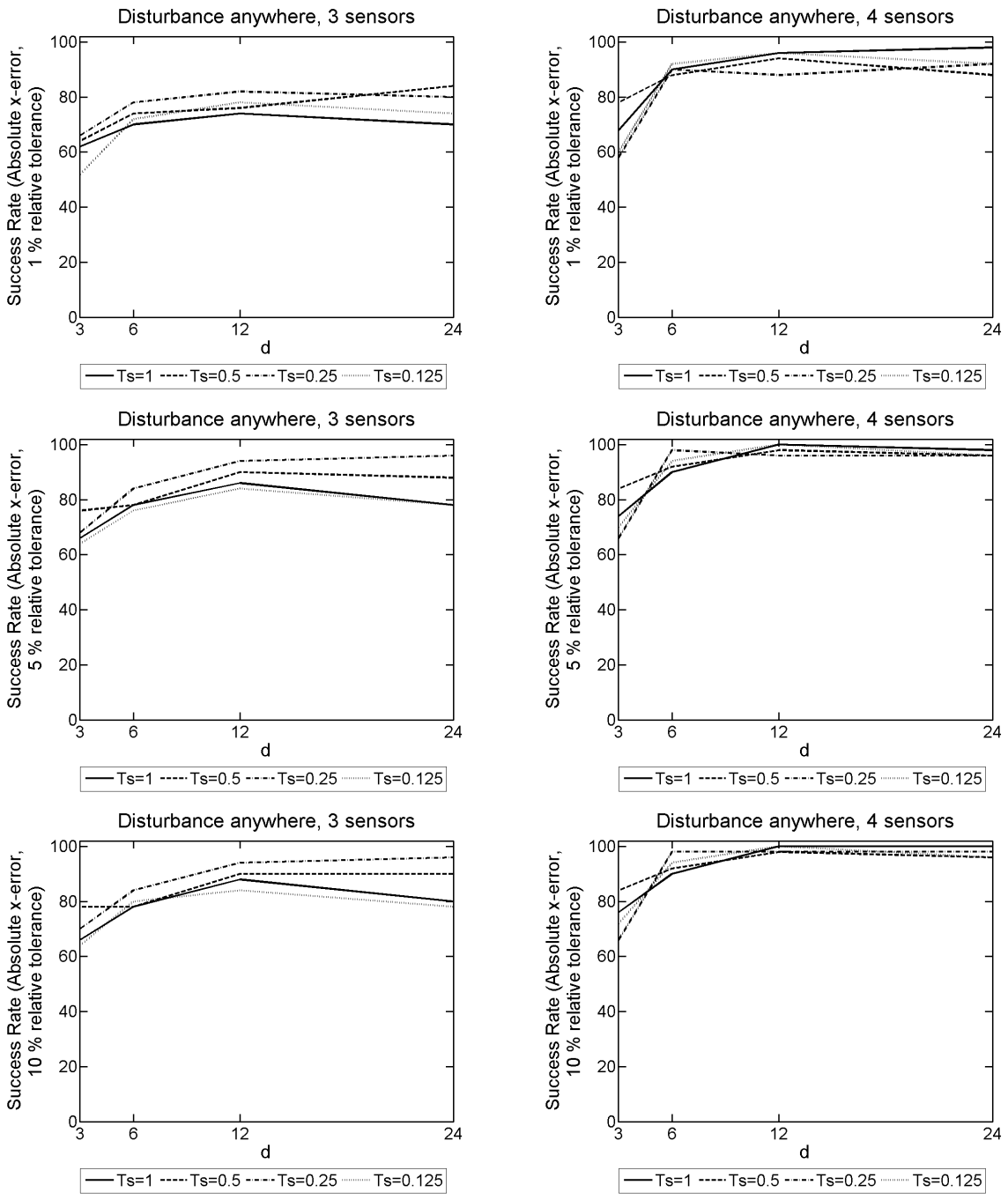


Figure 9.153: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations. The results on the left-hand side have 3 sensors present, and on the right-hand side 4 sensors were present with approximations of u and $\partial u/\partial t$.

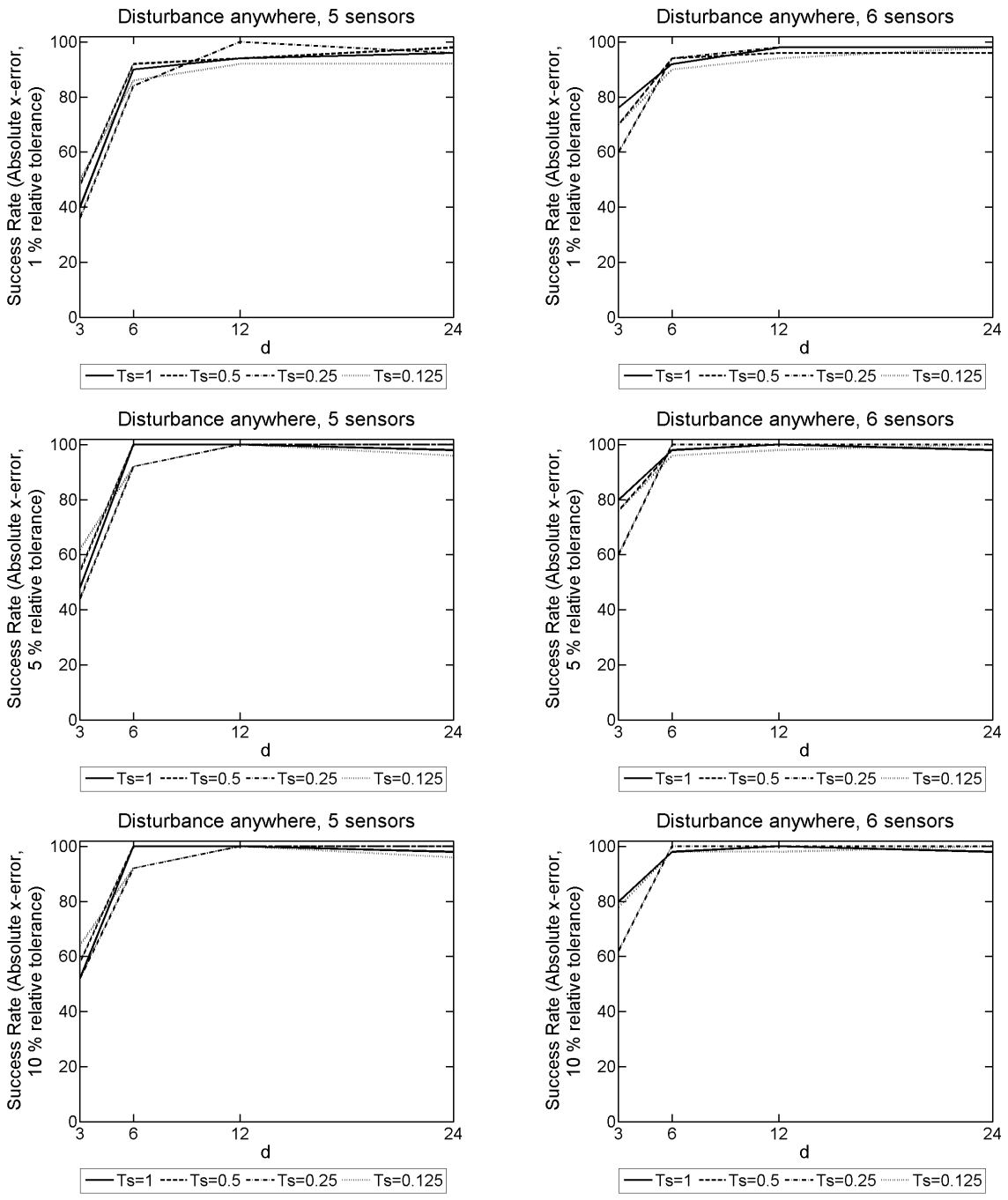


Figure 9.154: The 1D model success rate for different T_s and d values, used to form our SVD from the explicit FDM approximation of u on a mesh with dimensions of $N = 50$ and $L = 9000$, a disturbance frequency of $F = 25$Hz over a simulation duration of $T = 3$ seconds. These probabilistic results come from 50 random disturbance locations. The results on the left-hand side have 5 sensors present, and on the right-hand side 6 sensors were present with approximations of u and $\partial u/\partial t$.