Models, Methods and Algorithms for Supply Chain Planning

A thesis submitted

for the degree of

Doctor of Philosophy

by

Deborah Chippington Derrick

School of Information Systems,

Computing and Mathematics

Brunel University

September 2011

Abstract

An outline of supply chains and differences in the problem types is given. The motivation for a generic framework is discussed and explored. A conceptual model is presented along with it application to real world situations; and from this a database model is developed.

A MIP and CP implementations are presented; along with alternative formulation which can be use to solve the problems. A local search solution algorithm is presented and shown to have significant benefits.

Problem instances are presented which are used to validate the generic models, including a large manufacture and distribution problem. This larger problem instance is not only used to explore the implementation of the models presented, but also to explore the practically of the use of alternative formulation and solving techniques within the generic framework and the effectiveness of such methods including the neighbourhood search solving method.

A stochastic dimension to the generic framework is explored, and solution techniques for this extension are explored, demonstrating the use of solution analysis to allow problem simplification and better solutions to be found. Finally the local search algorithm is applied to the larger models that arise from inclusion of scenarios, and the methods is demonstrated to be powerful for finding solutions for these large model that were insoluble using the MIP on the same hardware.

Acknowledgements

Cormac Lucas who posed many questions that led to formulations and analyses that have been included in this work. For his patience in going through many drafts over a very long period of time and aiding me in moving the large quantities of results, that now make up the current appendices, from the main text.

My husband and business partner Tim, who has been steadfast in his support from the moment we first discuss the idea of my doing a research degree. The development of the Neighbourhood Search method has been a joint venture and one which we have also had success in employing in commercial settings. He has provided constructive ideas and criticisms throughout on all aspect of this work as well as making it possible for me to study on both a practical and emotional level.

Brunel colleagues Angela Koller and Marian Guertler for their encouragement and their confidence that I would eventually following them to this point.

To my four children who have made the task even more of a challenge, but have never seriously complained about the level of neglect they may have suffered as a consequence and in fact may have benefitted by being expected to cook meals and do other tasks that I may well have done for them otherwise.

Contents

Figures

Tables

1 Introduction and Background

1.1 Background

Many projects and studies, both within the business world and the academic arena have investigated and implemented supply chain models.

Much of the research focuses on specific models for specific situations, typically a particular company, within a particular field, focusing on answering a particular question. These involve problem analysis, abstraction of a model for the representation of the specific real world system, the construction of a problem specific data model and problem specific solution model. The other area of focus is the solution techniques employed and particular aspects of how the method was implemented and often novel variation in approaches to the implementation.

In this work we focus on the common features that are seen in most supply chain problems, and consider modelling techniques and solution methods that could be applicable to a wide range of supply chain problem instances. We show that it is practical to create an abstract and generic framework that can then be implemented and used to solve a variety of SCPPs in a way that is independent of any particular solver or mathematical representation. We also describe a simple neighbourhood search algorithm that allows solution of bigger SCP instances with stochastic features.

1.2 Different Problem Types

1.2.1 Strategic

A strategic view provides the overview, it is usually the view of the whole enterprise and it is over a relatively long timescale. When taking this view we are able to ignore many detailed variations and interactions which are only relevant on short timescales because over a longer timescales these average out. The view is much smoother and more continuous. This is the dimension of the problem where the granularity is larger; the time units considered will typically be numbers of months or even years.

Decisions taken in strategic planning will usually involve large financial investments. These decisions are likely to leave little flexibility for reversal of the decisions, with high financial penalties if the wrong decisions are taken; however, the benefits for making good strategic decisions are typically considerable.

1.2.2 Tactical

Tactical problems lie between the extremes of strategic and operational, with finer granularity and timescales that are not as long as years, but not as short and fine grained as those in operational problems. It is usually necessary to consider some tactical details when modelling a strategic problem; this provides the necessary underlying detail for the strategic model to be correctly modelled.

1.2.3 Operational

The Operational view is the detailed view. This view is over a short period of time; usually only a few days or weeks, and may provide details of what is happening minute by minute in some cases. The operational view will usually focus on a small area of the supply chain, even down to the scheduling of a particular machine or production line in a factory, for example. When considering problems in this detail the differences are more significant and the specifics of the individual situation are not averaged out as they would have been in a longer term view. The relationships and interactions between the various parts of a system in an operational view are usually much more complex, with logical dependences and exclusions which can usually be ignored or neglected in tactical and strategic views. This presents a situation where there is greater complexity, and it is crucial that the huge variation in the detail is taken into account in carrying out planning at this level. Decisions made at this level are often required to be updated if there is a change in requirements, or the previous plan has run into problems.

These types of problems are typically non linear and require the use of strategies and models that are specific to the problem instance in order to reduce the complexity of the problem, Hence the specifics of the problem become crucial in finding ways to reduce the complexity whilst providing solutions.

This is where supply chain planning starts to address issues such as scheduling, routing, and rostering. Often these different aspects of the same supply chain problem are addressed in different ways, producing several operational models for one area within a supply chain.

1.2.4 Uncertainty

Real world problems will have uncertain data because we need to plan far enough ahead to be able to implement our decisions in time for them to be effective. Demands for products, costs of materials, fuel, labour and transport costs are all uncertain. So significant quantities of the data used in supply chain models is inevitably based upon forecasts. There has been considerable research into the use of stochastic models to provide more robust solutions rather than ones that are solely optimal for a single set of forecast data. Some supply chain models have also incorporated stochastic features into the models; mainly in terms of demand scenarios. However, most stochastic models rely on a deterministic model that is used as the basis of the stochastic procedures.

1.3 Motivations for a Generic Model

There are a number of advantages and benefits for creating a generic model and supporting framework.

1.3.1 Faster and Cheaper Results

For many organisations the supply chain model is only one of several possible decision support tools, so it may be difficult to justify huge investment in building a model. A pre-built framework offers the potential for much quicker and cheaper development of such models.

A generic model offers the potential to demonstrate the benefits of such supply chain models at a considerably lower cost. For a large organisation considerable spending on data and optimisation can be justified when small percentage savings can lead to substantial benefits, but for smaller organisations these same real up-front costs may not be justified by corresponding benefits that may be achieved.

A generic model also offers increased accessibility of optimisation techniques to other decisions, where the potential costs and benefits would not justify investment in constructing a bespoke supply chain specific model from scratch.

1.3.2 Reduction of Risk

Those that have developed such models previously may be confident of a cost benefit to the supply chain planner, but for the business planner this will often be unproven technology and can be viewed as a high risk investment itself. The availability of a pre-built framework that has been shown to work before can help to reduce the apparent risk.

1.3.3 Earlier Delivery of Results

There is also an issue of the long development cycle for many optimization problems including supply chain models. Models need to be completed and further analysis to be carried out, before the knowledge gained from the model can be used to aid decision making. It may be months before even the study phases for these data and optimization models are completed. A generic model offers the potential for drastically reducing this development cycle and delivering results that can be understood and evaluated by the business quicker in the project cycle.

This reflects the common practice in industry such as the providers of Enterprise Resource Planning (ERP) systems such as SAP (2011), QAD (2011), Caliach (2011), Infor, IFS etc. who provide large and sophisticated software suites which are typically based on relational databases and may include a number of more specialised applications. These suites are typically applicable across many different industries through customisation. In practice, these companies have pre-configured versions of their modules which are specialised for different industries, so that they can be implemented more quickly.

1.3.4 Guidance for Data Gathering

With the development of these systems, data is usually a significant and difficult issue. Just defining the data required can be an extensive task. A clear framework allows much of the necessary data and its structure and relationships to be better understood at an early stage of the development cycle.

A set of simple pre-built data structures and models allows a rapid turnaround in being able to construct such problem instances, making this information more accessible to decision makers.

1.3.5 Common Terminology

A generic framework provides a business-independent terminology which can be used as a basis to map the business specific-terminology. Often during the development considerable effort is required in providing a communications bridge between the business owners and the analyst, database and optimisation specialists.

1.3.6 Clarify General and Specific Problem Aspects

A generic framework for a class of problems helps to clarify those aspects which are common across a range of problem instances in that class of problems, and in consequence highlights where particular problems are different from the generic template problem.

1.3.7 Solver Independence

We are interested in generating a robust SCPP model that is independent of the details of any single supply chain. Along with that objective, we also want to make sure that this

SCPP model is also stable and robust with respect to different solvers, even different solving technologies.

Independence of any particular solver, allowing the modelling framework to be validated on a variety of different solvers as well as with a variety of different problems and problem instances improves confidence in the approach before even starting on a specific problem instance. This typically simplifies the validation of the approach for real problem instances.

The ability to explore the supply chain model with several different solvers allows the system to be evaluated and/or proved with one or several solvers, before making a decision about choosing any particular solver.

1.3.8 Practical Implementation

A major objective of this work is to make sure that we have a relatively pure and clean supply chain model that does not become specific to a particular solving technique or algorithm. However, we are aware that to be of value, a generic abstract model needs to lead to practical implementations and useful solutions.

1.3.9 Do it right once and reuse

One mechanism for achieving wider use of the technology is to reduce the entry barrier by making it easier, quicker and cheaper to build good solutions. Do the analysis once in such a way that it can be re-used in the form of a generic model that fits a range of specific problem instances. If the generic analysis and modelling approach is good enough, then it should be possible to use the generic framework as a basis to build extensions to cope with any aspects of a specific problem instance which are not covered by the generic modelling framework. Having a proven generic model would allow more companies to make that move with confidence.

1.4 This Thesis

The thesis is organised as follows: In chapter two we present a literature review of supply chain planning and identify key features.

In chapter three a representation for generic supply chain planning models and supporting data models are presented.

Chapter four describes two problems and gives details of the problem instances of each of these. The implementation of the warehouse location problem are discussed and the results compared with results available in the literature and from two website which host the problem data

In chapter five we present MIP and CP implementations for the generic model.

Chapter six is an investigation into a number of alternative reformulations of the model. Details of result from these experiments provided.

Chapter seven describes how the generic model has been successfully extended to allow it to solve multiple scenario models and methods of analysis of the strategic problem are presented.

Chapter eight describes an approach for solving the mathematical programming problem using a neighbourhood search technique and results obtained are presented.

Finally conclusions are presented in chapter 9.

2 Literature Review

2.1 Supply Chain Planning

There is an extensive and growing literature on supply chain management, covering a wide range of problem types from short-term operational decisions such as scheduling production, through mid-term tactical decisions such as supplier selection, to long-term strategic decisions such as facility location decisions.

First, a number of existing reviews of the supply chain planning field were analysed, and these provided a basis for a classification of supply chain planning problems that can be used to classify other published papers, identifying where common issues are addressed and where there are gaps in the published literature.

2.2 Existing Reviews of the Literature

A number of extensive literature review papers have been published, which cover many contributions to the field and provide useful guidance on classifying other papers.

Mula *et al* (2010) is an extensive review of 44 papers on supply chain production and transport planning. They describe a taxonomy that they used to classify the literature, based on supply chain structure, decision level, modelling approach, purpose, shared information, limitations, novelty and practical applications. They found that of the 44 papers they considered, the majority dealt primarily with tactical decisions; only seven of those papers considered strategic decisions, and of those only four considered both tactical and strategic decisions. The problem addressed in this thesis includes a combination of features and characteristics that none of the 44 papers reviewed covers.

Melo *et al* (2009) also provide a review of papers on facility location and supply chain management. All of the papers reviewed can be assumed to include strategic decisions as they all address the facility location problem to a greater or lesser extent. They classify the 139 papers reviewed according to several criteria, such as single vs multiple echelons, single vs multiple commodities (products), single vs multiple time periods and deterministic vs stochastic data. The majority of papers reviewed address single period deterministic problems; only one paper was identified that covers three or more echelons, multiple commodities and multiple periods, but that paper does not address any stochastic issues of uncertain data. They also classify the papers based on the other decisions that are considered along with the facility location decisions, looking at decisions for capacity, inventory, procurement, production, routing and mode of transport. Again none of the papers reviewed is identified as discussing all of capacity, inventory, procurement, production and routing together which are addressed in this thesis.

Min and Zhou (2002) review contributions from over 70 papers and identify a number of key components of supply chain modelling, including the supply chain drivers (customer service, financial value, information exchange, risk management) and the variables and constraints which define the decisions that are being addressed. They also provide a taxonomy of supply chain modelling which over generalises to a large extent but which does recognise contributions from a diverse range of fields of study and acknowledges the importance of Information Technology developments to this field. They identify a number of research areas that they consider likely to be of growing importance, including the application of mathematical programming techniques to cover multi-period, multi-echelon problems, and the use of other techniques such as game theory, simulation and Theory of Constraints to widen the range of issues than can be included in supply chain models.

Beamon (1988) uses a simple four-way categorisation of the papers reviewed as a starting point: deterministic analytic, stochastic analytic, economic and simulation. They also discuss supply chain performance measures including both qualitative (e.g. customer satisfaction) and quantitative measures (e.g. cost, profit) as another categorisation. Finally they discuss the types of decisions and decision variables in each paper as another categorisation (scheduling, inventory levels, assignment of DC to customer etc.). Of the papers reviewed, most are analytical (about half deterministic, half stochastic) with only three papers using simulation and just one using an economic model. Most papers use cost as the main performance measure. Most papers address inventory levels and ordering (batch) size as the main decisions, with a smaller number addressing scheduling issues; just one or two papers address the issue of assignment of production or storage to a site, or assignment of which DC will serve each customer. No papers were identified that address the assignment of functions to all the sites in the supply chain while minimising cost or similar measures with uncertain stochastic data.

Narasimhan and Mahapatra (2004) list 37 papers with the problem area addressed in each case (grouped into strategic, tactical and operational areas) and identify in each case the issues addressed. They discuss five “illustrative” models, addressing: (a) investment implications of innovation-based competition between buyer and supplier, (b) bidding by a prospective supplier of a product, (c) bid evaluation and supplier selection by a buyer dealing in multiple products, (d) integrated operations in a supply chain, and (e) market integrated distribution.

Erengüc *et al* (1999) breaks the field into three distinct stages; the supplier stage, the production (plant) stage, and the distribution stage. They give a very clear explanation of the sorts of decisions that need to be addressed in each of these stages of supply chain planning, with proposed outline mathematical models. They identify several areas for future research, including:

Current work has primarily focused on single stage or at most, two stage models and has indicated that there are substantial benefits which can be achieved by coordinating inventory decisions in light of demand uncertainties as well as capacity constraints. However, there is a lack of approaches which explore these decisions simultaneously at all three stages. Such approaches can explore the impact of shifting capacity within the stages of the chain as well as how an uninterrupted flow of materials (due to better management of inventories) could lead to cost reductions for the entire chain.

2.3 Classification Scheme for Published Papers

The following classification dimensions have been extracted from the other literature reviews described above. The supply chain planning literature can be categorised according to these dimensions, many of which are widely discussed and used by many authors, some of which are inter-related:

- Strategic vs Tactical vs Operational decisions
- Purpose: facility location, supplier selection, scheduling, product routing etc
- Fixed or flexible sites (choose the functions available at each facility)
- Single vs multi-product
- Simple vs complex products (Bill of Materials, raw materials, assembly etc)
- Simple vs complex processes (multiple steps, assembly, packing, storage etc)
- Single vs multiple echelons or layers
- Simple vs complex transport routes (inter-layer only, intra-layer allowed, reverse routes)
- Single vs multiple suppliers for each product
- Single or multiple suppliers to each site or customer
- Handling of inventory at sites/facilities and across time periods
- Financial considerations (budget constraints, tax, transfer prices, fixed and variable cost components)
- Customer service issues (shortfall allowed or not, flexible delivery routes)

- Deterministic vs stochastic (uncertainty in demands, supplier costs, transport costs, final product value)
- Uni-directional vs bi-directional (reverse logistics, recovery, repair, rework etc.)
- Objective (maximise profit, minimise costs, robustness)
- Single vs multi-objective

2.4 Specific Papers

Many of the papers referenced by the other literature reviews listed above overlap with the work described in this thesis on one or more of the dimensions listed above; however only a small number are similar on most of these dimensions. None share all the characteristics of the problem and model described in this thesis. Those that are most similar along with some later papers are reviewed below.

All the most similar papers found through the literature searches are listed in Table 2-1, where the main characteristics on the dimensions listed above are summarised. In this table, it is possible to see most clearly on which characteristics each paper differs from the work described here.

Dogan and Goetschalckx (1999) develop a MILP formulation for a strategic supply chain problem, and a decomposition solution method based on Benders' decomposition. They use the problem structure which showed an accelerated convergence when solving their larger problem instances. They describe a real-world case study in the packaging industry.

Hung *et al* (2006) describe a generic supply-chain node to capture the features present in all supply-chain entities and demonstrate it use in a multi-echelon multi-product supply chain with five sites and four products. They model the behaviour of this supply chain over a two-year period using simulation.

Timpe and Kallrath (2000) describe a MILP formulation for a deterministic supply chain planning, including lot-sizing and inventory issues. All the facilities are known and fixed but the production processes can be configured to produce one or several different products. Their original target is the chemical process industry. They allow for only one change of operating mode per facility per production period. The first feasible solution was usually accepted as the objective value was close to that of the LP relaxation (within a few percent, well within the error associated with the forecast input data).

Sabri and Beamon (2000) describe a four echelon supply chain problem with multiple products for a single time period, with lead-times. The overall problem is solved iteratively by alternating between strategic MILP and non-linear operational sub-models. They use a multi-objective approach, and explicitly include consideration of delivery time and volume flexibility. Five scenarios were examined.

Jayaraman and Pirkul(2001) describe a single-period facility location model for a four supply chain problem. Their problems are from health-care products manufacturing and result in large MILPs. They present a MILP and an heuristic solution method that uses Lagrangian relaxation.

Goetschalckx *et al* (2002) extend the problems and approach of e.g. Dogan and Goetschalckx (1999) to include more international considerations including tax and transfer pricing resulting in a non-convex problem with a linear objective function. The bilinear equality constraints are linearised, some by fixing values of variables. A heuristic solution procedure is presented which it is observed was highly dependent on the starting point and experiments were carried out with alternative procedures for determining the starting points.

Jang *et al* (2002) describe a system of four modules, model management and data management modules which link a supply network design optimization module which uses a Lagrangian heuristic and a production and distribution planning module solved which uses genetic algorithm which generates real time production and distribution plans. They develop mathematical models and solution methodologies for each of these separately. The customer demand is deterministic and the problems are considered for a single time period and they use a backward planning process, solving the outbound, the distribution and finally the inbound sub-problem.

Kallrath (2002) describes a multi-facility production network problem in the chemical industry with facility selection, allowing both opening and closing of facilities. They include nonlinear pricing structures for the purchase of raw materials and multi-criteria objectives. Production facilities have operating modes which can be changed once per time period. They solve several deterministic problems to compare different scenarios and perform sensitivity analyses.

Aghezzaf (2005) describes a strategic capacity planning and warehouse location problem with demand uncertainty and a solution method that uses a decomposition algorithm using a Lagrangian relaxation method. The problem is given as a MILP formulation. Their proposed decomposition algorithm does not seem to provide much reduction in the solution gap or solution time.

	Strategic and Tactical	Multiple Period	Facility Assignment	Facility Selection	Facility Commissioning	Facility Decommissioning	Facility Configuration	Facility reconfiguration	Routing Inter-echelon	Routing Intra-echelon	Multiple suppliers	Inventory	Multiple Products	BOM	Multiple facility production	Investment budget	Stochastic Data	Transfer prices	Taxation
Aghezzaf 2005	Y	Y	Y	Y	Y	Y	X	X	Y	X	X	X	X	X	X	X	Y	X	X
Bidhandi *et al* 2009	Y	X	Y	Y	X	X	X	X	Y	X	Y	X	Y	Y	X	X	X	X	X
Bidhandi & Yusuff 2011	Y	X	Y	Y	X	X	X	X	Y	X	Y	X	Y	Y	X	X	Y	X	X
Canel *et a*l 2001	Y	Y	Y	Y	Y	Y	X	X	Y	X	X	X	Y	X	Y	X	X	X	X
Cohen & Lee 1988	X	Y	Y	X	X	X	X	X	Y	X	Y	Y	Y	Y	Y	X	Y	X	X
Dogan & Goetschalckx 1999	Y	Y	Y	Y	X	X	Y	X	Y	Y	Y	Y	Y	X	X	X	X	X	X
Fleischmann *et al* 2006	Y	Y	Y	X	X	X	Y	Y	X	X	Y	X	Y	Y	X	Y	X	Y	X
Georgiadis *et al* 2011	X	Y	Y	Y	X	X	X	X	Y	X	X	Y	Y	X	X	X	Y	X	X
Goetschalckx *et al* 2002	Y	Y	Y	Y	X	X	Y	X	Y	Y	Y	Y	Y	X	X	X	X	Y	Y
Guillen *et al* 2005	Y	Y	Y	Y	Y	Y	X	X	Y	X	X	Y	Y	X	X	X	Y	Y	X
Gupta & Maranas 2003	X	Y	Y	X	X	X	X	X	Y	Y	X	Y	Y	Y	Y	X	Y	X	X
Hinojosa *et al* 2000	Y	Y	Y	Y	Y	Y	X	X	Y	X	X	X	Y	X	X	X	X	X	X
Hinojosa *et al* 2008	Y	Y	Y	Y	Y	Y	X	X	Y	X	Y	Y	Y	X	X	X	X	X	X
Hugo & Pistikopoulos 2005	Y	Y	Y	X	X	X	Y	Y	Y	X	X	X	Y	Y	X	X	X	X	X
Hung 2006	X	Y	X	X	X	X	X	X	X	X	Y	Y	Y	Y	Y	X	X	X	X
Jang *et al* 2002	X	X	Y	Y	X	X	X	X	Y	X	Y	X	Y	Y	X	X	X	X	X
Jayaraman & Pirkul 2001	X	X	Y	Y	X	X	X	Y	Y	X	Y	X	Y	Y	X	X	X	X	X
Jiao *et al* 2009	X	X	Y	X	X	X	X	X	Y	X	X	X	Y	Y	Y	X	X	X	X
Jung *et al* 2008	X	Y	Y	X	X	X	X	X	Y	X	X	Y	Y	X	X	X	X	X	X
Kallrath 2002	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	Y	Y	Y	Y	X	X	X	X
Ko & Evans 2005	Y	Y	Y	Y	Y	X	X	X	Y	X	X	X	Y	X	X	X	X	X	X
Lejeune 2006	X	Y	Y	Y	X	X	X	X	Y	X	Y	Y	Y	Y	X	X	X	X	X

	Strategic and Tactical	Multiple Period	Facility Assignment	Facility Selection	Facility Commissioning	Facility Decommissioning	Facility Configuration	Facility reconfiguration	Routing Inter-echelon	Routing Intra-echelon	Multiple suppliers	Inventory	Multiple Products	BOM	Multiple facility production	Investment budget	Stochastic Data	Transfer prices	Taxation
Listes & Dekker 2005	X	X	Y	Y	X	X	X	X	Y	X	X	X	X	X	X	X	Y	X	X
Martel 2005	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	X	X	X
Melachrinoudis 2000	Y	Y	Y	Y	X	X	Y	Y	Y	X	X	X	X	X	X	Y	X	X	X
Melo *et al* 2005	Y	Y	Y	Y	Y	Y	X	X	Y	Y	Y	Y	Y	X	X	Y	X	X	X
MirHassani *et al* 2000	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	X	X	Y	X	Y	X	Y	X	X
Mirzapour 2011	X	Y	Y	X	X	X	Y	Y	Y	X	Y	Y	Y	X	X	X	Y	X	X
Ouhimmou *et al* 2008	X	Y	Y	X	X	X	X	X	Y	X	Y	Y	Y	Y	X	X	X	X	X
Perron *et al* 2010	X	X	X	X	X	X	X	X	Y	X	Y	Y	Y	Y	Y	X	X	Y	Y
Sabri & Beamon 2000	Y	X	Y	Y	X	X	X	X	Y	X	Y	X	Y	Y	X	X	X	X	X
Saldanha da Gama 1998	X	Y	X	Y	Y	Y	X	X	X	X	X	X	X	X	X	X	X	X	X
Salema *et al* 2007	Y	X	Y	Y	X	X	X	X	Y	X	X	X	Y	X	X	X	Y	X	X
Santoso *et al* 2005	Y	X	Y	Y	X	X	Y	X	Y	X	Y	X	Y	X	X	X	Y	X	X
Sodhi & Tang 2009	X	Y	Y	X	X	X	X	X	X	X	Y	Y	Y	X	X	Y	Y	X	X
Srivastava 2008	X	Y	Y	Y	Y	X	X	X	Y	X	X	X	Y	X	X	X	X	X	X
Thanh *et al* 2008	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	X	X	X
Timpe & Kallrath 2000	X	Y	Y	X	X	X	Y	Y	Y	Y	X	Y	Y	Y	X	X	X	X	X
Tsiakis *et al* 2001	Y	X	Y	Y	X	X	X	X	Y	X	X	X	Y	X	X	X	Y	X	X
Ulstein *et al* 2006	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	X	X	Y	X	X	X	X	X	X
Vidal 2001	X	X	X	X	X	X	X	X	Y	X	Y	Y	Y	Y	Y	X	X	Y	Y
Vila *et al* 2006	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	X	Y	X
Yan 2003	X	X	Y	Y	X	X	X	X	Y	X	Y	X	Y	Y	X	X	X	X	X

Table 2-1 – Supply chain features documented in papers

Canel *et al* (2001) describes an algorithm for solving a deterministic capacitated multi-product multi-period and multi-stage facility location problem. The approach makes extensive use of the known structure of the problem to simplify the solution process, which is in multiple phases. Firstly finding which facilities must be open or closed throughout the full time horizon of the problem. Then using a branch and bound procedure to repeatedly solve the single time period facility location problem to generate a set of candidate configurations of the sets of facilities. Finally dynamic programming is use to select one of the facility generated configurations for each of the time periods in order to minimise cost.

Guillén *et al* (2005) describes a stochastic multi-echelon supply chain problem with multiple products and multiple time periods. A multi-objective two-stage stochastic optimisation approach is used. Consideration is given to the financial aspects, including interest rates, tax rates, depreciation, capital investments, fixed costs and indirect expenses. The multiple objectives include maximising net present value, maximising demand satisfaction and minimising financial risk. The authors perform extensive analysis of the impact of choosing the supply chain design from a single scenario compared to the design from the stochastic 100-scenario case, showing benefit in terms of improved net present value and reduced financial risk from using the design generated by the stochastic approach.

Fleischmann *et al* (2006) describe a load-planning model for BMW which includes supply of materials, three production departments at eight facilities. It determines the investments needed for the departments and impact on cash flows. The MILP determines the allocation of known production levels to the facilities based on the demands of eight or 10 sales regions allowing transport cost and taxes for each to be calculated. The model used cash flow before tax, but the authors provide details of four assumptions they would need to make to incorporate the approximate tax impact.

Hinojosa *et al* (2000) describes a deterministic multi-period multi-product multi-echelon capacitated facility location problem, allowing new facilities to be opened and existing ones closed. They model flows of products to customers as a fraction of the demands, and the amount transported to warehouses as a fraction of the capacity. The authors state that the generated MILP models are very large and solving them directly takes a prohibitive amount of CPU time, so they present an approach that uses Lagrangean relaxation with a heuristic solution construction phase to generate good feasible solutions.

Hinojosa *et al* (2008) generalises Hinojosa *et al* 2000 to include modelling of inventory to be carried over from one time period to the next, and outsourcing of materials and products where the supply chain capacity is insufficient. They report that direct solution of the MILP problem was faster than their heuristic for the small problems, but that the heuristic algorithm was substantially faster for the largest problems even though the solutions generated do not appear to be as good as those from the direct MILP solver.

Hugo and Pistikopoulos (2005) describe a model including life cycle assessment (LCA) criteria as part of the strategic investment decisions. The problem model includes multiple time periods and multiple products, with a single supplier. The problem is from the chemical industry and includes network configuration and capacity planning strategy that minimize the environmental impact of the entire supply chain. They describe a six step Multi-objective optimisation algorithm.

Ko and Evans (2007) describe a mixed integer nonlinear programming model for the design of a distribution network including reverse logistics for third party logistics providers. It includes warehouse and collection facilities. The problem is divided into two sub-problems to model the forward and reverse logistics. A genetic algorithm-based heuristic is presented.

Listes and Dekker (2005) describe a stochastic programming approach to extending a deterministic location model for product recovery network design, with a case study for the recycling of sieved sand including storage and cleaning. They address uncertainty of supply and demand of a single product through a three-stage stochastic programming model, with location decisions in the first stage, high or low supply being revealed in the second stages and the flow decisions made in the third based on seven demand scenarios.

Melo *et al* (2005) describe a supply chain planning problem including dynamic planning horizon, generic supply chain network structure, external supply of materials, inventory, distribution, facility configuration, availability of capital for investments, and storage limitations. Capacity expansion and reduction for the distribution centres only are addressed by two extensions of the base model where new facilities can be selected and remain until the end of the time horizon, similarly existing facilities that are closed do not reopen. The authors provide a table of comparisons with other models.

Ulstein *et al* (2006) describes a multi-product, multi-period supply chain planning problem in the high-quality metals manufacturing industry, specifically silicon and related products. The model makes an explicit distinction between the facilities and the machinery installed at each facility. Complex products and processes mean that some products need specific equipment and experience at the facility, and there are complex relationships between customers and products (a customers could even specify a particular furnace for a specific product).

Salema *et al* (2007) describe single and two product reverse distribution network with capacity limits and demand and returns uncertainty. A mixed integer formulation is developed. They extended the model to include minimum and maximum capacity constraints for each type of facility, which due to constraint redundancy reduces the model size, and then extend this to a three-scenario model.

Santoso *et al* (2005) describe a supply chain planning problem with multiple products, multiple echelons, and configuration of the machines and equipment at each facility. They present an initial deterministic MILP formulation of the problem and description of a two stage stochastic programming model, with a first stage facility selection and configuration and second stage tactical decisions. They use a sample average approximation (SAA) scheme to allow them to take potentially very many scenarios into consideration, and provide a detailed discussion of the characteristics of this scheme. They solve this problem using a Benders decomposition scheme with various additional acceleration schemes used singly and in combination.

Srivastava (2008) describes an integrated holistic conceptual framework for network design for reverse logistics that combines descriptive modelling with optimization techniques at the methodological level. The model is a multi-product, multi-echelon, profit maximizing model for the recovery of televisions, cars, refrigerators, washing machines, mobile phones and computers. The demand uncertainty in this case is the likelihood of products being taken to a recovery centre.

Sabri and Beamon (2000) describe a single-period multi-objective four echelon supply chain model. The focus on operational performance, such as service levels, inventory levels and lead times. They split their problem into a strategic sub-model that optimises the set of facilities that will be used, and an operational sub-model that includes the uncertainty and non-linearity of variable production, distribution and transportation costs. The two sub-models were solved alternately, starting with the strategic sub-model. Finally the authors performed some sensitivity analysis by fixing or varying the bounds on volume flexibility, delivery flexibility and/or customer service levels.

Vila *et al* (2006) describe a supply chain planning problem in the softwood lumber industry. The BOM is a divergent with one-to-many relationships starting with trees and generating many products. The facilities are configurable, including the selection of the technologies and seasonal shutdowns are included. A mixed-integer program with after tax profit objective function was generated.

Bidhandi *et al* (2009) describe a deterministic single period four-echelon supply chain network problem. The model is split into two parts, covering the strategic and tactical decisions. They use Benders' decomposition, with the strategic decisions in the master problem and the tactical decision in the sub-problem.

Bidhandi and Yusuff (2011) extend their earlier work to include planning under uncertainty using a two-stage stochastic program. The first stage includes the strategic decisions of facilities selection and the second stage the tactical decisions of processing and transportation. Uncertainty is included for operational costs and customer demand. They use the SAA (Sample Average Approximation) method, using Bender's decomposition to solve the MIP generated.

Gupta and Maranas (2003) describe a tactical multi-site, multi-product and multi-period supply chain planning problem with uncertain demands. The authors describe a two-stage stochastic programming model, where the manufacturing decision variables are considered as the first-stage here-and-now decisions while the logistics, inventory and transport decisions are modelled as second-stage wait-and-see decisions, given the production levels and demand realisations for each product.

Mirzapour Al-e-Hashem *et al* (2011) describe a multi-site, multi-period, multi-product production planning problem under uncertainty. Their model is a multi-objective mixed integer nonlinear programming model including uncertainty in both costs and demands. They transform this to a multi-objective linear problem and the multiple objectives are handled by optimising each objective separately and then formulating a new objective function that minimises the total normalised deviation from the optimal value for each objective. They also account for uncertainty in the data through the use of robust optimisation techniques, solving over a set of discrete scenarios. They include the costs of hiring, firing and training workers in the model.

Jung *et al* (2008) describe a negotiation model for tactical planning. The model involves a producer and a distributer, and there is incomplete exchange of information. The distributer is trying to meet a known demand, and the producer making best decisions about where to sell their products. The negotiation process is to determine supply quantities when the price is set by local markets. The objective is to improve the profit for both partners, with iterative optimisation steps taken on both sides, starting with the distributer. The model is multi-period, with both the distributer and the supplier having multiple facilities with transfers and inventory.

Lejeune (2006) describes a deterministic three-stage multi-period supply chain planning. The three stages are suppliers, production and distribution. A solution algorithm based on a variable neighbourhood decomposition search metaheuristic is used, with increasingly large neighbourhoods. The integrality requirements on some of the variables in the problem are relaxed; CPLEX is used to solve the sequence of sub-problems with increasing numbers of the integrality requirements enforced; hence the integrality of the initial relaxed solution is incrementally increased.

Ouhimmou *et al* (2008) describes a tactical multi-product, multi-period, multi-echelon supply chain planning problem for the wooden furniture industry. The problem was formulated as a MIP which was then solved with both CPLEX 9.1.2 and a time-decomposition heuristic that solves for increasing time horizons while fixing the decision variables for earlier time periods.

Sodhi and Tang (2009) describe the extension of a deterministic multi-period supply-chain planning problem to include demand uncertainty and cash flows, the focus on inventory costs and cash flows and the tactical production plan is not scenario dependent. They discuss the role of decomposition, aggregation and scenario sampling in solving these.

Thanh *et al* (2008) describe a MILP model for a strategic deterministic multi-echelons, multi-periods supply chain planning model with inventory, multiple suppliers, and BOM. The authors suggest several directions for further research including the addition of budget constraints in the problem and the use of decomposition and metaheuristics to solve larger instances.

Yan *et al* (2003) describe a single-period multi-product multi-echelon supply chain model with supplier and facility selection and BOM. There is also a lot of discussion of how to model the logical relationships inherent in many BOM relationships, and how these can be linearised.

Cohen and Lee (1988) describe a model framework for a stochastic multi-product, multi-echelon tactical/operational supply chain planning problem with BOM. There is an extensive discussion of the stochastic modelling required, and they note that minimising the costs of the whole supply chain would be the ideal approach but that this was computationally infeasible (in the late 1980s). Instead they decompose the problem into four linked sub-models (material control, production control, finished goods stockpile and distribution) which are then solved in sequence.

MirHassani *et al* (2000) describe a stochastic multi-period, multi-product supply chain network planning problem. The problem is decomposed into the strategic "here and now" decisions and the tactical "wait and see" decisions. Two approaches were taken for solving the problem. First a detailed scenario analysis was carried out, evaluating a range of strategic "here and now decisions" by solving the tactical "wait and see" models for many scenarios. Secondly a two-stage stochastic programming formulation was solved using Benders decomposition, while fixing the strategic decision variables to the values found in the earlier scenario analysis.

Tsiakis *et al* (2001) describe a MILP model for a single-period, multi-product, multi-echelon supply chain network problem with fixed manufacturing facilities and customers, and dynamic location of warehouses and distribution centres.

Georgiadis *et al* (2011) describe a stochastic four echelon, multi-product, three-period supply chain network planning model. The plant and customer locations are fixed, whilst the locations and capacities of the warehouses and distribution centres are to be decided by the model, but those decision are identical throughout the planning horizon. Tactical decisions can be reconfigured in each of the three time periods and for each scenario

Melachrinoudis and Min (2000) addresses the problem of relocating a single facility within a current network. It describes a case study of a firm which plans to move its current manufacturing and warehousing facility within the USA. They describe the multi-period, multi-echelon, single-product, multiple objective, MILP model. Transport costs are included, but no supplier selection, allocation to customers or facility configuration is considered. Unusually, the demand per customer is dependent on the choice of new facility location.

Saldanha da Gama and Captivo (1998) describe a dynamic multi-period facility location problem which includes opening and closing costs. This is not a supply chain problem; rather it is a more general facility location problem that would be suitable for other problems such as choosing school locations. The facilities are all identical and demand to be met by each facility is based upon the cost of meeting the demand of one or more communities from that facility. They use a two-phase heuristic approach. The first phase starts with all facilities open in all time periods and removes them one time period at a time whilst maintaining feasibility. The second phase is a local search which allows a single switch in facility opening variables in each step.

Martel (2005) gives a very detailed description of a model for multi-period, multi-product, multi-echelon supply chain planning. The problem model includes configuration and layout decisions for the facilities. The modelling of the inventory throughput functions gives rise to non-linearities in the model. They solve successive linear MIP problems with added cuts to strengthen the formulation. The approach did not try to achieve an optimum solution: *"...rather, it is perceived as a practical scenario improvement method based on reasonable approximations of the inventory-throughput functions"*.

2.5 Literature summary

The existing literature includes a very large number of papers describing variations of the supply chain planning problem. However, when a more detailed analysis is done, many of the published papers focus on specific aspects of the supply chain planning problem and do not include the range of issues addressed in this thesis; for example, many papers are focussed on tactical and/or operational decisions and do not include strategic issues of supply chain design and configuration.

Table 2-1summarises the results of the literature review above. The most directly relevant (40 in total) have been classified under a set of headings to try to identify what they address.

Table Headings:

- **Strategic and tactical** – whether the model supports both strategic and tactical decisions at the same time
- **Multiple period** – does the model support multiple time periods, or just a single time period
- **Facility assignment** – allocating of production, storage or other function to a facility
- **Facility selection** – solving the location selection problem of where to put facilities, usually from a finite set of discrete choices
- **Facility commissioning** – opening of new facilities during the planning horizon, after the start
- **Facility decommissioning** – closing of existing facilities during the planning horizon, after the start

- **Facility configuration** is the issue of whether the capabilities at a facility also need to be decided. At a minimal level this may be deciding the capacity for each product at a facility, or more generally (as in this thesis) the decisions on what equipment (e.g. production, packing lines, storage) should be installed at a facility. Specifically this refers to the internal configuration decisions at a facility rather than the overall supply chain network configuration
- **Facility reconfiguration** Changes to facility configuration during the planning horizon but after the start
- **Routing Inter-echelon** – does the model include transport decisions between facilities in different echelons
- **Routing intra-echelon** – routing within an echelon. Most published papers assume all transport and transfers of products and materials are from suppliers to plants, plants to warehouse etc., and do not allow transfer of e.g. intermediate products between production plants in the same echelon
- **Multiple suppliers** – does the model include selection decisions between multiple suppliers, or are they assumed fixed either explicitly or implicitly (not mentioned)
- **Inventory** – does the model include inventory decisions, such as how much material or product to store and where to store it
- **Multiple products** – does the model include multiple products
- **BOM** refers to modelling of the bill of materials relationships, e.g. the consumption of sub-assembly intermediate products to produce a final assembled product. In general this can feature either convergent (assembly of several intermediate products) or divergent (sawing of timber, paper deckling), one-to-one or a combination.
- **Multiple facility production** is the issue of production of a final product being carried out in multiple stages at multiple facilities, e.g. manufacturing sub-assemblies at one facility before transport to another facility for final assembly.
- **Investment budget** refers to constraints on what facilities can be opened or closed and what configuration changes allowed due to a limited budget during each time period. Most published papers do not directly limit the changes to the supply chain network structure in this way.
- **Stochastic data** – is the data assumed to be fully known in advance, or is there modelling of uncertainty with multiple scenarios and/or stochastic data

Several interesting observations can be made from this analysis of the published literature.

- The most common shared feature is the allocation of production and/or storage to facilities. Only one paper did not do this and that was more related to service provision such as school location.
- The second most common feature is consideration of the routing decisions between the echelons in the supply chain – all but three of the papers include this. In contrast, only 9 of the papers seem to model transport within an echelon.
- Almost all the papers identified feature multiple products (36 papers out of 40). However only 18 of the papers include modelling of the bill of materials relationships; and only 9 support the transfer of intermediate products between facilities to allow a product to be made in stages at multiple sites.
- 75% of the papers include the selection of facility locations (opening new sites). However, only 15 of the papers allow the opening of facilities at an intermediate time period during the planning horizon rather than at the start, and fewer still (13) allow closure of facilities during the planning horizon rather than at the start.
- Most of the papers (29) cover multiple time periods, allowing changes in costs, demands etc. The others are mostly single time period models or steady-state models, often associated more with tactical or operational decisions than strategic decisions.
- Only 18 of the 40 papers address the issue of inventory, which is perhaps surprising.
- 14 of the papers model the internal configuration of the facilities at some level, from simplistic (capacity can vary) through to more detailed (choice between types of production line technologies with different costs, throughputs etc).
- 13 of the papers address the issues of uncertainly, either through multiple scenarios or use of stochastic programming techniques. The other 27 papers deal with deterministic data where the costs, demands etc are all known.
- Only 4 of the papers address the issues of budget constraints which limit the changes that can be made to the supply chain network.
- Solving Techniques

2.5.1 MILP

By far the most common technique for solving supply chain planning problems is the use of a mixed integer linear programming (MILP) solver. There are a number of commercial and non-commercial MILP solvers, and the better-known commercial solvers are now very powerful and capable of solving a variety of large MILP problems. A necessary restriction for solving large problems in reasonable time is to keep the objective and all the constraints linear; this is not usually a problem as most relations are inherently linear, but means that to include some constraints or objectives, linearization methods or approximation may have to be used. (Williams 1990, McKinnon and Williams, 1989, Yan and Hooker 1999)

Since many of the supply chain planning problems modelled with MILP techniques create MILP problem instances which are large and hard to solve, a number of authors have used additional techniques to accelerate the solution process, notably decomposition techniques like Benders' decomposition (Benders 1962, Bidhandi 2011). Another approach to decomposition uses a "natural" decomposition into sub-problems, such as the Alternate heuristic (Cooper 1964) which alternately solved the location and allocation components of a combined location-allocation problem. The same approach has been extended to tackle bi-linear problems by alternately solving linear sub-problems while fixing the other part of the overall problem (Vidal and Goetschalckx 2001, Perron *et al* 2010).

2.5.2 Constraint Programming

Constraint programming is a powerful modelling and problem solving technique. It has the advantage over MILP modelling in that it has no reliance on ensuring that the model is linear, so almost any arbitrary logical or non-linear constraint can be directly modelled. Many of the principles used for MILP modelling will also apply when implementing a constraint programming model, but the ability to directly model more complex relationships means that the model is often more compact, without needing to add auxiliary variables and constraints to model non-simple parts of the problem (Smith *et al*, 1996). The disadvantage for solving these problems is that the mathematical techniques underlying these constraint solvers are inherently much weaker than those which underlie MILP solvers, and it is common to have to code significant parts of the search process in order to achieve good levels of performance. Even then, finding an optimal solution and proving optimality can be very difficult.

Jiao *et al* (2009) address a factory loading allocation problem (FLAP) for the case of single period, deterministic multi-site, multi-echelon, multi-product supply chain planning problem manufacturing. They address the problem of 'configuration' of a supply chain in terms of allocating which families of products will be processed at each site. The authors discuss issues of complicating constraints which they claim may be important to correct modelling of all the necessary relationships in the problem and which cannot be easily represented in a linearised formulation. Hence they formulate their problem as a constraint satisfaction problem, and solve it using constraint heuristic search (CHS) and a decision propagation structure (DPS). They only attempt to find feasible solutions rather than optimal solutions. The approach is illustrated on a problem with two product families, two sub-assembly facilities and four final assembly sites. This appears to be only a tactical/operational single time period problem, and no computational results are presented.

2.5.3 Local and Neighbourhood Search Heuristics

Local and Neighbourhood search heuristics are a powerful technique. These rely on the idea that, given a solution to a problem, then there is often a better solution to that problem that can be found "near" to that known solution. Conceptually, a "neighbourhood" about the current solution is defined, i.e. the set of solutions that can be reached from the current solution by making small changes to that solution. The neighbourhood is explored for a better solution and the processes repeated, defining a new neighbourhood around the new current solution. A sequence of improving steps, moving from one solution to the next, will converge to at least a local optimum. If no improving solution can be found in the neighbourhood, we can change the neighbourhood (e.g. enlarge it, or change how the neighbourhood is defined). The solution search process can also be allowed to accept non-improving moves to escape the (supposed) local optimum. Techniques such as simulated annealing (Kirkpatrick *et al* 1983) and tabu search (Glover 1989, Glover 1990) have been developed to manage the acceptance of non-improving steps to help the search escape from local optima.

Perron *et al* (2010) model a multi-echelon multi-product supply chain planning problem which includes transfer pricing and which leads to a bi-linear mathematical model. They solve it using both the Alternate heuristic (Cooper 1964) and Variable Neighbourhood Search (VNS) (Mladenovic *et al* 2003). The approach taken in this work is most closely related to Large Neighbourhood Search. The term Large Neighbourhood Search seems to have originated in Shaw 1998 for solving vehicle routing problems with constraint programming and local search.

Ahuja *et al* (2002) is a review of very large-scale neighbourhood (VLSN) search techniques, but focuses more on the use of move-based neighbourhoods rather than implicit method-based neighbourhood definitions.

Pisinger and Ropke (2010) gives a good recent overview of the field of large neighbourhood search, which they identify as being a sub-class of VLSN search but which doesn't fit well into the three categories identified by Ahuja *et al* (2002). They characterise the LNS technique as an iterative sequence of destroy and repair operations on parts of the solution. Similar approaches have also been called “Ruin and Recreate” (Schrimpf *et al* 2000).

Caserta and Voß (2009) provide another review of a wide variety of metaheuristics including several approaches similar to VLSN and LNS.

Taillard and Voß (2002) describe the POPMUSIC conceptual framework for local search by iterated partial optimisation of sub-problems which can be defined as large neighbourhoods. The approach uses a particular scheme to ensure that all sub-parts of the overall problem are covered and prevent re-visiting the same neighbourhood.

Sniedovich & Voß (2006) describe the corridor method which performs a LNS using a method-based neighbourhood definition, implementing the neighbourhoods by adding constraints to fix the values of (most of) the variables in the problem.

There has been considerable work on scheduling problems. Applegate and Cook (1991) describe the use of a shifting bottleneck heuristic to solve job-shop scheduling problems. Danna and Perron (2003) compare the use of structured and unstructured neighbourhoods for job-shop scheduling problems, and show that using the problem structure to guide neighbourhoods gives better results. The introduction gives a particularly clear explanation of the LNS approach. Perron *et al* (2004) describe the use of information from constraint propagation to automatically select good neighbourhoods to explore. Godard *et al* (2005) use information from the current solution in terms of a partial order schedule to select neighbourhoods to search for cumulative job-shop scheduling problems. Carchrae and Beck (2005) use a measure of the likely impact on the overall objective to define cost-based neighbourhoods to explore for a scheduling problem. Laborie and Godard (2007) describes a self-adapting LNS approach for scheduling problems. They use a portfolio of neighbourhood definitions within a LNS framework with ideas from machine learning to automatically converge on the most effective neighbourhoods for each problem to give a robust scheduling algorithm. Cipriano *et al* (2009) used a local search library (EasyLocal++) with constraint programming (Gecode) to build a hybrid LNS solver for an asymmetric TSP, and report that even simplistic random neighbourhoods significantly outperform the pure CP approach. All of the above apart from Applegate and Cook (1991) used a constraint-programming approach.

Thompson and Orlin (1989) and Thompson and Psaraftis (1993) describe the use of large and complex neighbourhood definitions for vehicle routing problems which perform cyclic transfers of parts of the solution between multiple routes. Pisinger and Ropke (2007) describe an adaptive LNS which performs well on a large number of benchmark problems of several different types, by using a portfolio of neighbourhoods which are selected automatically. He and Qu (2009) describes an approach similar to that used by Perron *et al* (2004) to help guide the selection of neighbourhood using constraint propagation, and compare results with simpler structure-driven neighbourhoods for the nurse rostering problem.

3 Problem Representation

In this chapter, the Supply Chain Planning Problem (SCPP) is discussed and analysed, and a common set of terms, entities and relationships are defined that can be used to describe a wide range of supply chain planning problem instances. This abstract framework has been created so as to avoid dependence on any specific industry or type of product or process in order for it to be generic and flexible. It also does not make reference to or use any particular mathematical solver or algorithm, or any particular mathematical representation or technology. It will then be shown that this abstract conceptual model for SCPPs can be implemented in a reasonably direct manner as a mathematical model, and also as a database schema and as an object-oriented model.

In order for modern optimisation techniques to be made practically available for widespread use in commerce and industry, it is important that the initial implementation costs and timescales are kept as small as is practically possible. One way to achieve this is to standardise on the modelling approach and the supporting tools so that as much as possible of the necessary structures and algorithms are available with minimal cost and delay. Another benefit of this approach of standardising the terminology and modelling approach is that it becomes possible to more directly compare different SCPPs and SCPP instances since they can be put into the same representation; and where a SCPP or instance cannot currently be represented, that too gives us a better understanding of where and how those problems differ and what extensions would be required to include them.

It is important that a Generic SCPP model incorporate common features seen in supply chains, but to avoid hard coding features that are specific to any particular supply chain in a way that may lead to the exclusion of other supply chains problems. We aim to demonstrate that the SCPP model devised is problem independent and stable in the face of changes to the problem instance, mathematical model or underlying solver.

We describe a conceptual model for the SCPP. Since an abstract conceptual model is only a theoretical entity, we also show how the conceptual model can be implemented in terms of:

- a mathematical model suitable for solving using a mathematical modelling tool
- a relational database schema,
- an object-oriented model suitable for implementation in C++ or Java

Actually several different mathematical models are generated which are shown to be able to solve several different problem instances consistently using several different solution methods (MIP, CP, Neighbourhood Search).

In addition we show how this generic model can be made accessible to those with both simple and complex problems, by making the views of the data applicable to the problem instance.

3.1.1 Problem Characteristics

In the literature review a list of key features of supply chain planning problems were identified. The problems addressed in this thesis have the following characteristics:

- Strategic and tactical decisions solved together
- Site (facility) selection together with assignment of production and routing for each product
- Facilities with one or more different functions with a choice of different equipment
- Multiple products
- Complex products which can have multiple raw materials, parts or partial products which need to be combined or assembled at each stage
- Complex multi-stage processes that can be split across sites
- Multiple echelons, including suppliers, production/manufacturing/assembly, packaging, storage and customers
- Complex routing within and between each echelon
- Multiple suppliers for each facility and customer
- Inventory handling
- Investment budgets, fixed and variable costs for facilities
- Facilities can be opened or closed just once in the planning horizon, with some modelling of ramp-up of availability after each facility is opened
- Stochastic data handled through multiple scenarios; optimising the strategic decisions across multiple scenarios for robustness

3.2 Conceptual Model

Our conceptual model describes the supply chain and the supply chain planning problem in terms of some well-defined entities, based upon the concept of transformation processes. The model is dependent on a set of underlying dimensions, the detail of which changes the strategic and tactical balance of the problem. They allow the description of what is transformed, where and when; and of the flows between these points of transformation.

The main entities in the conceptual model are what is transformed which we will refer to as products, these may include raw material and intermediate parts and product, packaged, unpackaged, unfinished goods, etc. The products are transformed in facilities and an additional level of detail is added within the facilities, referred to as a technology which allow the detail of not only production lines, but of storage space, human resources etc. which process the products to be modelled. The other dimension is the time periods, which defines the time horizon and the level of detail in which the planning is required to take place. These provide the main dimensions of the model.

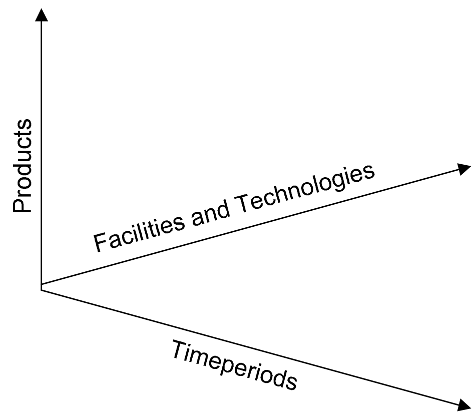

Figure 3-1 - Dimensions of the Model

The flow of products within the framework is described by

- the products,
- the facilities and technologies,
- the time periods,

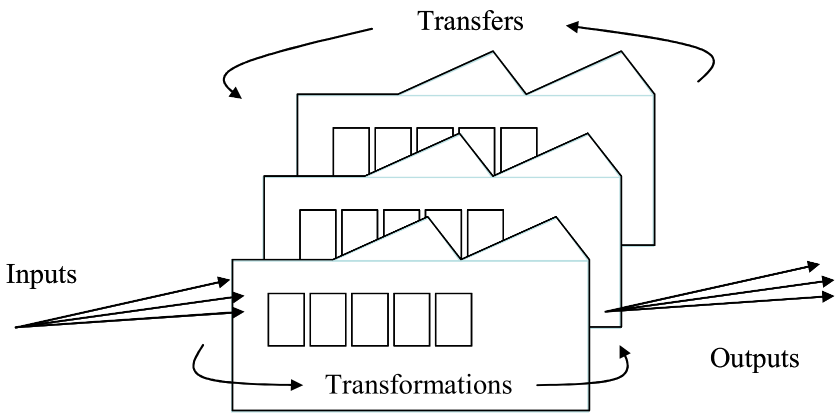

Figure 3-2 - Flow of Products

At the facilities these flows are driven by the need to meet the demands for each product at customer facilities in each time period. The flows of the products in the generic model are described by:

- the transformations of products at the facilities in each time period,
- the transfer of products, from supplier facilities into the system,
- the transfer of products between facilities
- the transfer of products to customer facilities.

3.2.1 Model Framework

3.2.1.1 Products

Products range from the initial raw materials that enter the supply chain, through the intermediate products that are moved within the supply chain, to the final products that leave the supply chain. Products may all be considered separately or grouped, changing the level of tactical detail that is described within the model. In some cases products within one area of the business or trading area may be considered separately to the rest of the supply chain, drilling down to the detail in just one or more aspects of the products.

3.2.1.2 Time Periods

The chosen time periods determine the granularity of the model, the timescales within which events are modelled. The chosen length of the time periods changes whether the model is strategic, tactical or operational, with shorter timescales typically providing a more operational planning model, through levels of tactical planning to long timescale strategic planning models.

The length of the time periods do not need to be uniform enabling longer later time periods as well as being able to look in greater detail over some period of time, allowing drilling down to the details of the model in this dimension in a similar manner to the other dimensions.

3.2.1.3 Facilities and Technologies

Facilities include the sites at which transformation of products takes place, along with the end point facilities, the customers' facilities, where the products leave the supply chain and the start point facilities, the supplier facilities, where the raw materials and intermediate products enter the supply chain.

A supply chain problem may be more directed to facility location or to configuration depending on the facility and technology decision included. The separate decisions for technologies allow the configuration to change over time.

Technologies provide the means of describing the detail within a facility, allowing the modeller to include decisions about equipment and infrastructure. Examples of real life items within a facility that will be modelled by technologies are production processes such as:

- Batch processes such as brewing or smelting
- Factory production lines
- Packaging and sorting
- Warehousing and human resources at a distribution centre

Even when a specific technology is not required to carry out processing within a facility there will be specific costs associated and capacity limitations as products pass through facilities and technologies allow these to be modelled. The detail with which the technologies are modelled within the facilities controls the amount of tactical detail that is included.

Most models include a rigid network structure with a set number of echelons often modelling suppliers, factories, warehouses and customers.

3.2.1.4 Demands

This is the amount of each product that the customers are expected to purchase. Along with this is the required price that it is expected they will pay for the products. This is data which must be forecast, and by considering a range of different forecasts it is possible to incorporate risk models into the generic framework.

3.2.1.5 Transformations

Following a product from its raw material requirements through to the finished product, it goes through a series of transformations. It is common to have a BOM (Bill of Materials) to describe these and it can be encoded by the use of predecessor products that are required for the production of a product. Each transformation will require the use of a technology of a specified type, each of these technologies will usually have limited capacities and may only be available in certain facilities. The cost of transformations can then be modelled via the cost of providing the technology that carries out the transformation. Technology capabilities at facilities, may be added or removed, and some technologies may have limited lives. Products may also be transferred from one facility to another. Transformations and transfer of products will both incur costs.

This description of production allows for a significant amount of flexibility. The simplest production case will be when a single product is processed into another single product directly, but more complex flows can also be described, for example cases where a number of products are required to produce the next stage product.

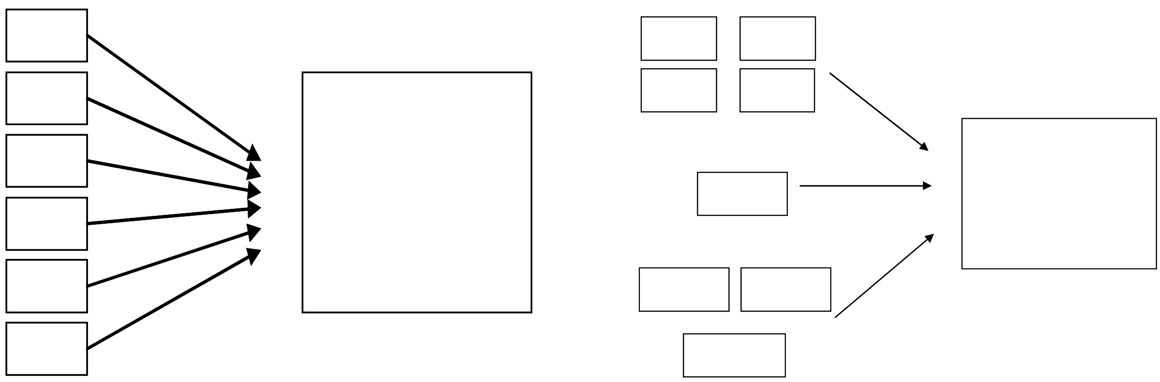

Figure 3-3 - Flows with Multiple Predecessor Products

These sorts of transformations would be found in a situation such as packing boxes of a product, although if the consumption of packaging material was also to be included this would be a required predecessor product too.

This more complex system of transformation, not only would be able to describe a simple transformation such as packaging including the packing materials, but the assembly of complex products from a variety of predecessor products.

Products can also be produced by splitting a single predecessor product into multiple resulting products.

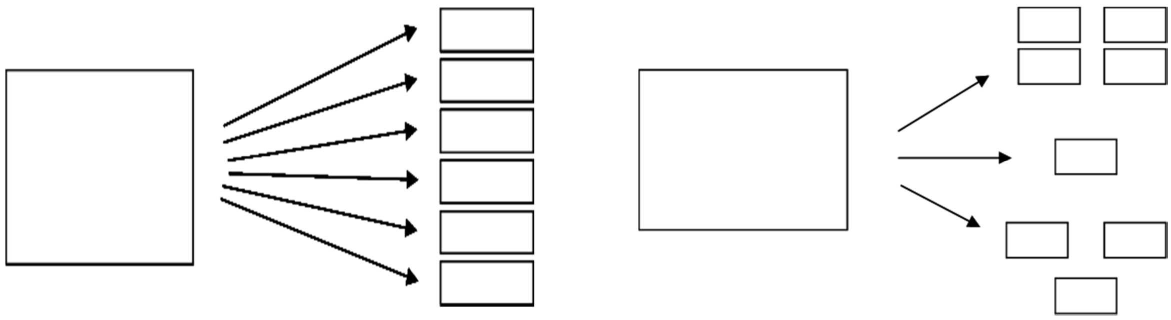

Figure 3-4 - Multiple Successor Products

These transformations may produce a set of products which may be identical or different. A combination of these transformations allows the description of complex sequences of processes, with some processes combining products and some splitting them that may be required to describe the production processes in many places.

Non-concrete products, such as energy requirements, can be included in these transformations in the same manner; this can be used when energy is consumed within the supply chain, but also if energy is an output too. This may become an issue of greater importance when companies may be distributing energy by products, as environmental pressures mount for what has previously been considered a waste product to be recycled.

3.2.1.6 Transfers

This is the transportation of products within the supply chain, between facilities and from facilities within the supply chain to the customers. Facilities may have different technologies available and products may be required to go through several facilities in order to be transformed into a final product. Some supply chains will be able to perform the same transformations at different facilities and modelling of these alternative routes of production allows investigation of the efficiency of such transfers of products. Examples of these types of transfers may be between factories and distribution centres, or different production stages such as production of paper and cutting it or production of metal slabs and the casting and machining that may follow, which will commonly be undertaken at different facilities.

Each possible transfer route would be described in the data model, allowing a balance between production, consumption, and transfer in and out of a facility to be maintained. Also each transfer would have an associated cost.

3.2.1.7 Additional Detail

In order to avoid replicated data, several of the entities have additional higher-level entities allowing grouping of lower-level entities and allowing additional details to be added at that level, without altering the main structure of the conceptual model.

In the datamodel implementation description we discuss our choices of these types, some of which have been chosen to be modelled by using separate database entities and others which are described by the specification of different types where the differences are solely in the problem instance data describing the types, for example Facility or Technology types.

3.2.1.8 Facility Detail

Types of facilities that are included in supply chains are suppliers, customers and those of the supply chain under consideration, which include factories, warehouses, distribution centre, etc. There may be further categorisation within each of these groups, for example facilities that are currently in use and potential facilities. These provide a hierarchy that can be useful in constructing models.

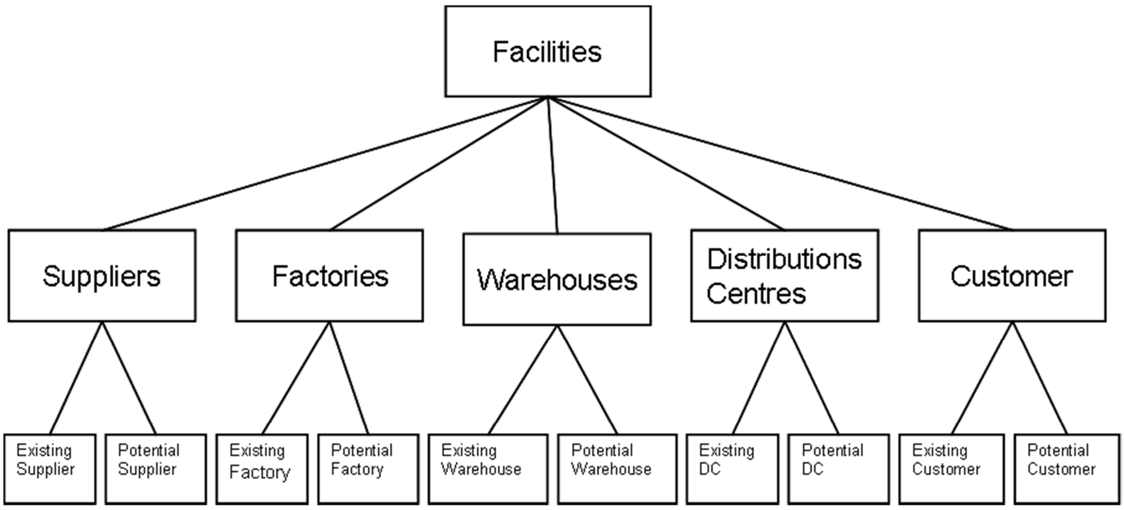

Figure 3-5 - Facility Detail

3.2.1.9 Other Entity Detail

Similarly, other entities may need categorisation and benefit from using different type descriptions within either the data model or the mathematical model. For example technology types may require different types for the technologies that carry out manufacturing, assembly or packing of products.

Product details required may vary, for example between raw materials, outsourced products, intermediate products, packaging materials and final products.

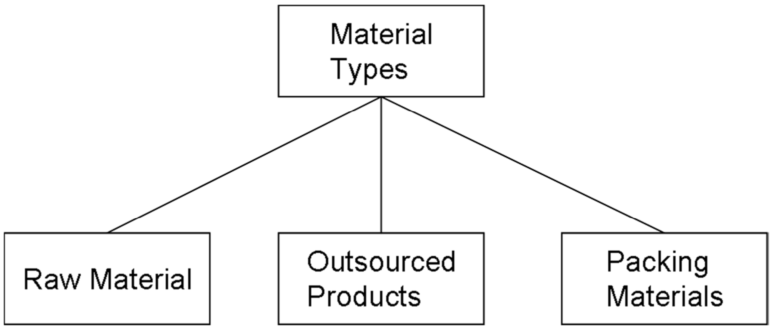

Figure 3-6 - Product Detail

3.2.1.10 Relationships between the Entities

There are clear relationships between the main entities in the generic model. The following diagram (Figure 3-7) gives an overview of these main entities and what can be considered the main relationships between them that underpin the generic model.

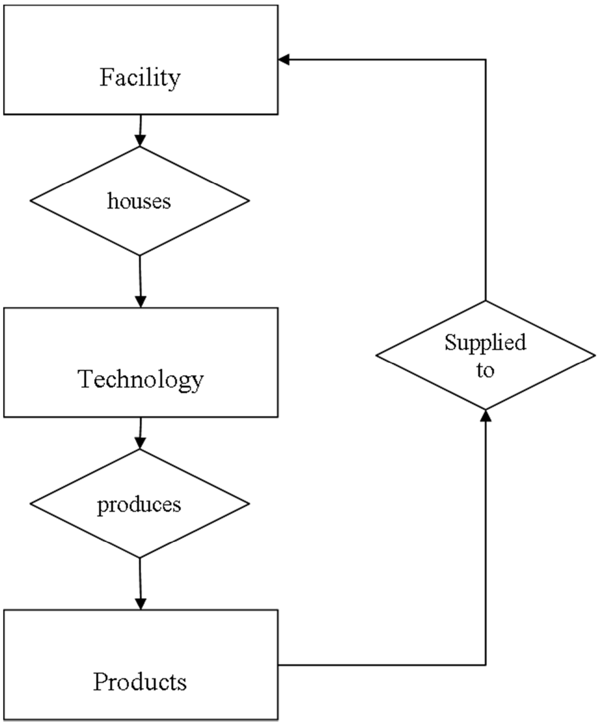

Figure 3-7 - Entity Overview

The Facilities house the technologies, which in turn produce products which are supplied to another facility, this may be one that further processes a product or an end point customer facility.

3.3 Problem Specification within the Generic Framework

The generic model was built using the conceptual model described previously, in order to facilitate easy encoding of problem instances by supply chain planners or analysts.

The data required to fully describe a Supply Chain can be divided into two categories:

The problem instance data which defines the Supply Chain infrastructure. This is typically the factories, warehouses, supply routes and so on, and may include some choices about closing existing facilities or opening new ones. Most of this information is usually known with relatively high certainty, but some may need to be predicted or forecast.

The problem instance data that defines supply and demand, prices and costs. This data is not known with any certainty, is more prone to variation, and is forecast; it relates to decisions at the tactical planning level, but is essential in understanding what effects it has on the strategic solution.

The supply chain infrastructure for the problem instance is defined in terms of the main entities within the Generic Model, these are:

- Facilities (factories, warehouses, distribution centres etc.),
- Technologies within these Facilities (e.g. production, packing etc),
- Products produced,
- Customers who are supplied with the Products from the Facilities,
- Suppliers who provide the raw materials used in production.

The tactical planning data would include:

- Product Demands
- Production Costs
- Transportation costs

This data is not known with any certainty and needs to be predicted or forecast. This can be considered as scenario data and for a robust solution to a Supply Chain problem, more than one set of this data is likely to be considered.

3.3.1 Data Model Implementation

Here we describe in detail the main entities from the conceptual model, and the associative entities that link them and how they are implemented in the relational model. The use of specialisation and generalisation of entities was discussed in the conceptual model, considering whether these allowed the data model to be better matched to the supply chain instances whilst being more efficient, by preventing the duplication of data without compromising the generic nature of the model. Both of the strategies have been employed in producing the relational model that we have developed.

There were three clear groups of Facilities that as discussed previously would be better modelled with specialised entities. The data requirements of each of these types of facilities were clearly different, so the following were modelled as separate entities in the relational model that we developed.

- Supplier Facilities,
- Facilities belonging to the supply chain problem owner
- Customer Facilities

This also led to the specialisation of the relational entities for the transportation of products between the supply chain facilities being considered separately from the cost of supply to customers and suppliers.

A simplistic Entity relationship diagram was given in the conceptual model description in Figure 3-7, the following diagram shows the relationships with the distinct types of entities for the supplier and the customer facilities.

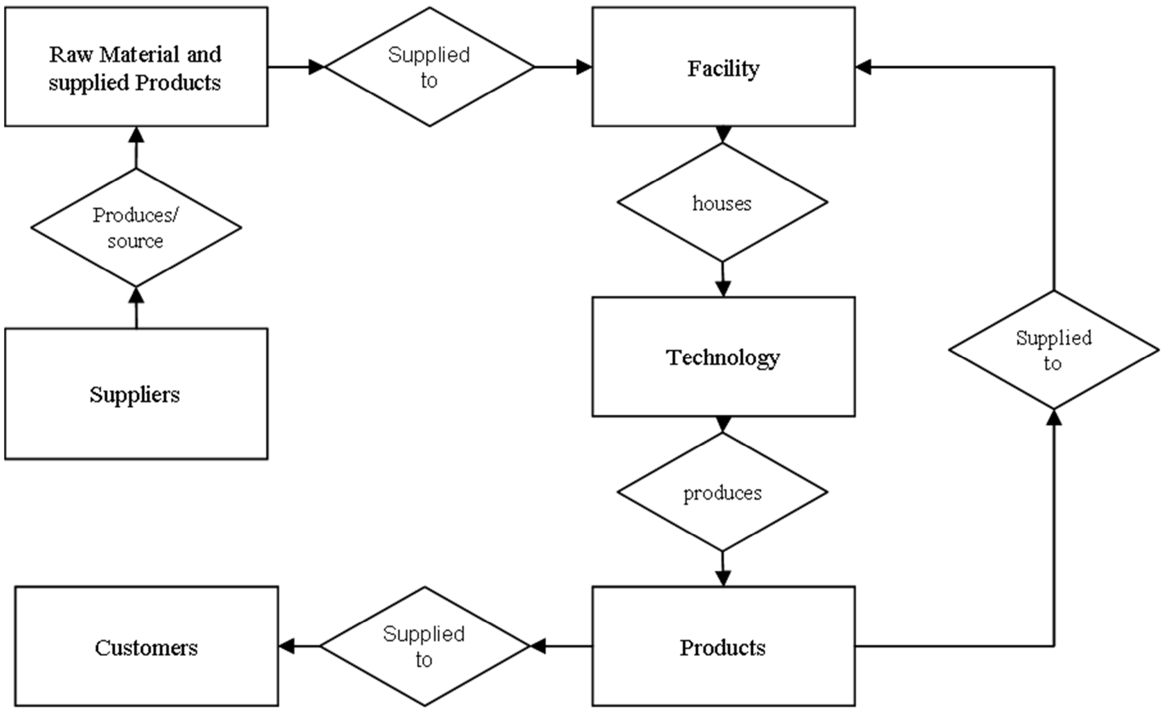

Figure 3-8 - Specialised Entity Overview

The following shows the completed Entity Relationship Diagram (ERD) view of the generic database that was developed, which is followed with an explanation of each of the main entities, and the associative entities linking them. The data fields included below do not include the soft information such as names of facilities, customers, products, etc. which are included in the database; although these are of great importance to humans in interpreting the data and results, they are not important to the functionality of the system.

This framework also requires the time dimension to be incorporated because the time periods underlie all the relationships within the model, as illustrated below. For example the Products are supplied to a Supply Chain Facility from a Supplier Facility in a particular time period; the cost of doing so may be dependent on the time period in which this happens, as well as the product that is supplied and the facility to which it is supplied. The framework also allows for a scenario dimension, this allows alternative sets of forecast data to be incorporated into the model.

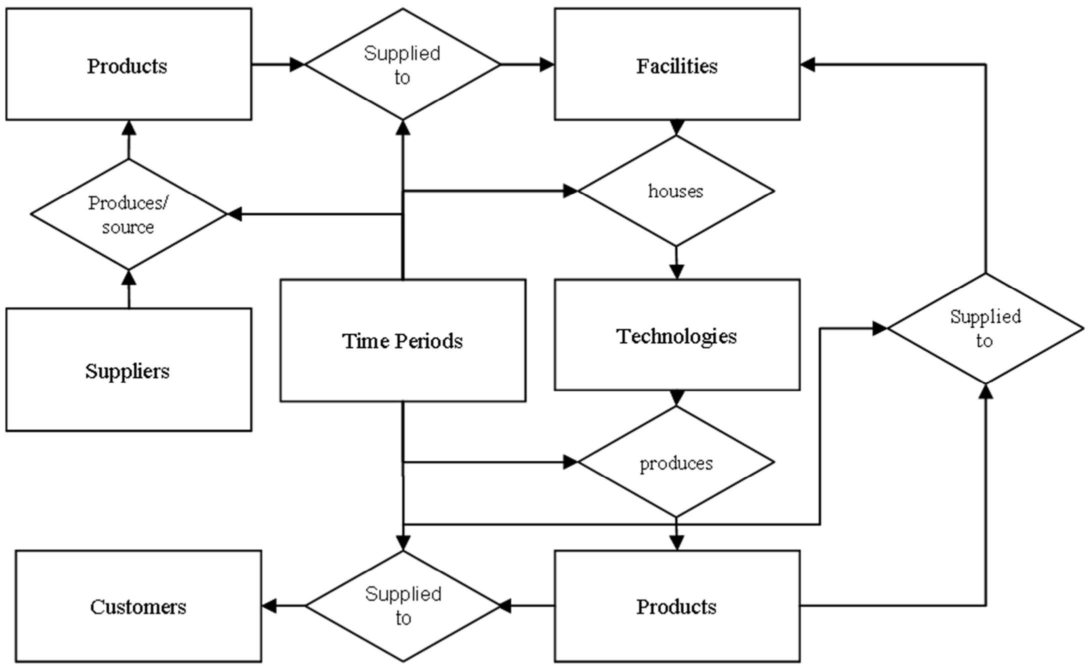

Figure 3-9 - Generic Entity Overview

3.3.2 Main Entities

First we describe each of the main entities in the above model and then consider how the relational entities were modelled in the database.

Facilty	
PK PK	<u>scenarioId</u> <u>facilityId</u>
 FK1	description facilityTypeId status landCapCost revenueFromSale

Each facility has a unique identifier, *facilityId*, and a *facilityTypeId* linking it to a *FacilityType*. The status specifies whether the facility is currently open or a potential facility that could be used. The capital cost of the site, the costs of constructing and the revenue that could be expected from the sale of the Facility are given.

Figure 3-10 - Facility Data

Facility Type	
PK PK	<u>scenarioId</u> <u>facilityTypeId</u>
	description maxCapacity timeToSetUp runningCost inventoryCost perCentRate

The *Facility Type* models a group of similar facilities, with the same maximum possible capacities, the same time to be set up, and the same running and inventory costs. As the percentage rate for payment made on the capital investment are the same.

Figure 3-11 - Facility Type Data

Technology Line	
PK	<u>lineId</u>
 FK1 FK2	description facilityId TechTypeId status

Each instance of a technology line is uniquely identified by a *lineId*, specifies which facility the technology is operating at or where it can be installed, the status indicates whether it is currently operational. Each technology line has technology type.

Figure 3-12 - Technology Lines Data

Technology Type	
PK	<u>TechTypeId</u>
	description timeToSetUp maxLines capitalCost

The *Technology Type* contains data about the capital and running costs. It also allows the maximum number of this type of lines to be the running cost of a line to be specified.

Figure 3-13 - Technology Type Data

Customer	
PK PK	<u>scenarioId</u> <u>customerId</u>
	description singleSupplier

Each customer is identified by a *CustomerId*. An option is provided to select whether the customer must be supplied from a single source within the supply chain.

Figure 3-14 - Customer Data

Although products are central to supply chain problems and hence form the central thread through the groups of data, little information is required about each product, and the only field other than the description is the option of specifying the units which the product will be measured, which could be a weight, items, bottles, boxes of items or SKUs; we discuss later the use of these units in providing visibility of the flow and cost of product through the system, and the importance of being able to specify quantities of products in a variety of units, in order to provide inbuilt scaling and improve numerical stability of models.

Most of the data relating to products will be related to the relationships between the products, and how they are transformed and transported, and from whom they are obtained as raw materials, or to whom they are supplied as finished products. The costs of products are all in relation to how much the raw material and equipment to produce them cost, the transportation costs given in relation to the transport routes in a time period and how much they are sold for given in relation to which customer they are sold to and in which time period.

Product	
PK	**productId**
	description **units**

Figure 3-15 - Product Details

Supplier	
PK	**supplierId**
	description

Similarly the Supplier entity only describes the suppler without any other data.

Figure 3-16 - Suppliers Details

The Time Periods provide the framework in which the supply chain model instance exists; they specify the length of time over which the operation of the supply chain will be considered, along with the detail in which it is modelled across this time, the sum of the duration of these time periods giving the time horizon for the problem. The time periods have a unique identifier, *timePeriodId*, and an investment amount can be specified to give the maximum investment that can be made in each time period on new Facilities and Technologies.

Time Periods	
PK,FK1 PK	<u>scenarioId</u> <u>timeperiodId</u>
	investmentAllowance duration

Figure 3-17 – Time Period Details

The duration of the time periods do not have to be identical, and there are advantages to be gained in modelling if the granularity is varied across the time horizon; allowing more detail about what will happen in the near future, but including a large time horizon without the detail and consequent increase in data and mathematical model size that would follow if the later part of the horizon were modelled in the same detail.

Scenario	
PK	<u>scenarioId</u>
	description

Figure 3-18 - Scenario Details

The scenario entity only has a unique identifier, *scenarioId*, and a description. It allows several sets of data to be specified from forecasts, such as customer demands, transportation or production cost, or even production rates.

3.3.3 Relational Entities

Each of these entities is linked in the database by relational entities. We will now describe how these have been derived, and how they model the details of the supply chain.

3.3.3.1 Material Costs

Consider the relationship between the Suppliers of products, the products which they supply, the facilities to which they supply them, the time periods in which this happens, and the scenario which is under consideration. This relationship is modelled by the Material Costs relational entity, which specifies the cost of a particular Supplier supplying a *unit* of a particular Product to a particular Facility, in a particular Time Period, under a particular scenario. There is also a data field for the specification of the units in which the product supplied from this manufacturer is measured, allowing visibility of this data to help prevent inconsistencies in the data being used.

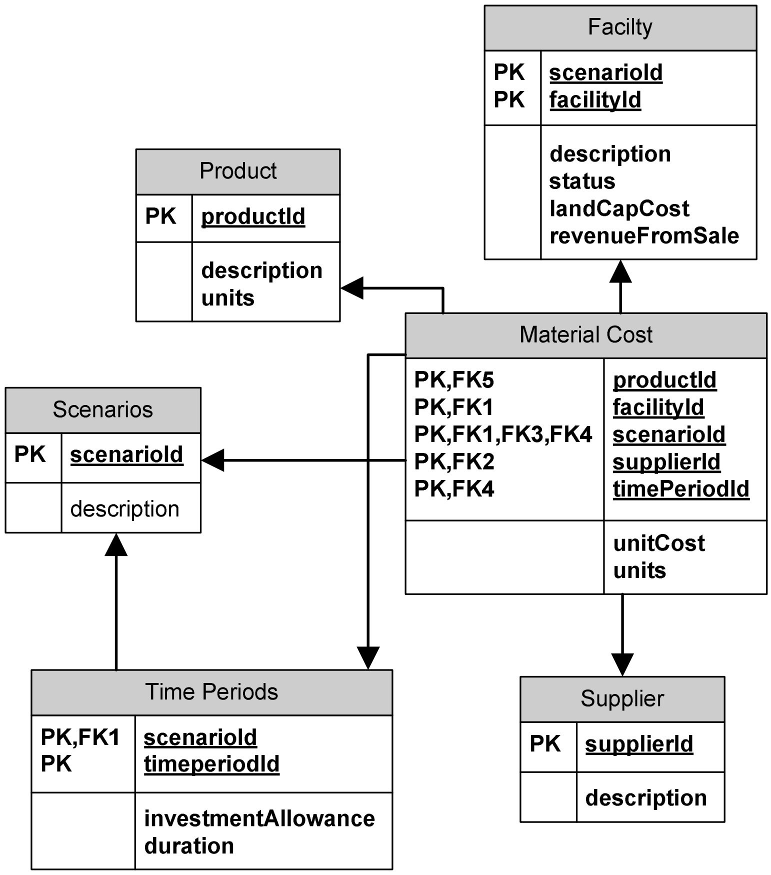

Figure 3-19 - Material Cost Data

3.3.3.2 Technology Throughput

The relational entity, *TechnologyThroughput*, provides the data relating to the flow of products through a technology. It gives the rate at which a particular technology can produce a particular product and the units for this rate can also be stated, which allow the users clarity about what the cost data actually means and whether it has been specified correctly for the product that is being considered. This cost data is both time period and scenario dependent, as well as being dependent on the product produced and the technology type being used to produce it. This allows changes that may come about because of improvements in production, or changes in working practices, to be considered across the time periods or within different scenarios.

The costing data is not included here but is included in the cost of running the Technology of that type, along with the cost of running the Facility, and of any predecessor products that will be used in the production.

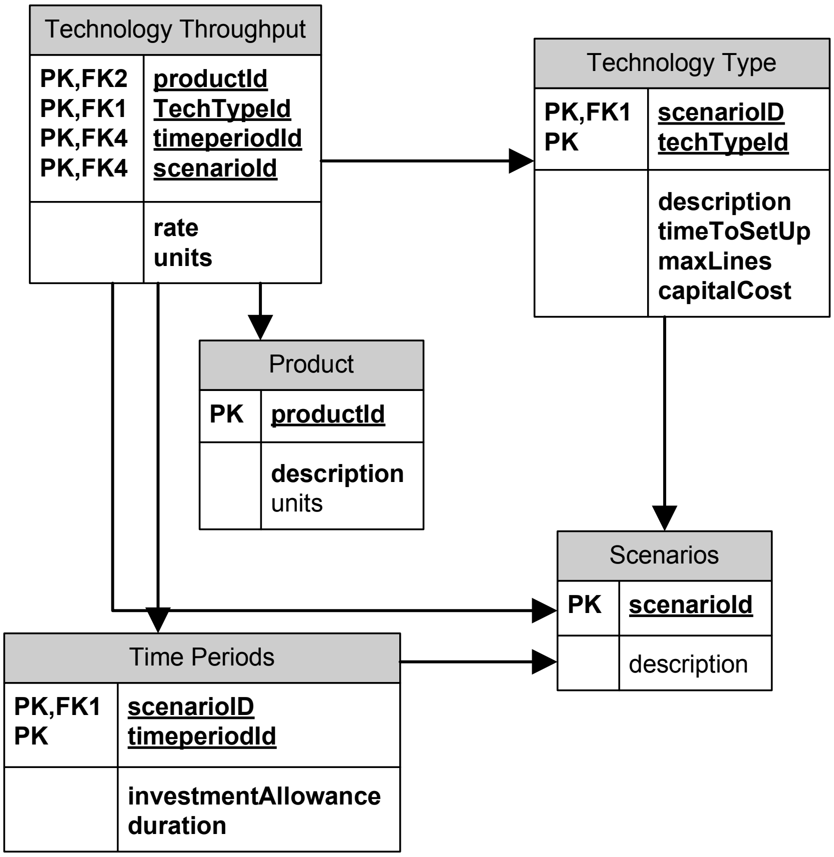

Figure 3-20 - Technology Throughput Data

3.3.3.3 Predecessor Requirements

As discussed previously the relational entity, Predecessor, provides the information about what is used to produce a product. Each product may have more than one predecessor product, and different quantities of each predecessor product may be required for each product, therefore providing the recipe for the production of the product. The amount of each predecessor along with the specified ratio; this gives the number of units of the product that are produced from a unit of the predecessor product. This value can be one (ie. one unit of the product for each unit of the predecessor); greater than one (ie. several units of the predecessor are required to produce a single unit of the product) or less than one (ie. several units of the product are produced from a single unit of the predecessor).

The units are specified for the products, these could be kilos, litres, tonnes, etc. or items, which could include bottles, boxes, SKUs etc. This information allows the user to easily see that the ratios provided when specifying the predecessor relationships between products and its predecessor products, allowing the user to be confident that units are used correctly in all the data specification. For example if a product had been specified as being measured in kilos, and the predecessor in tonnes, then the predecessor relationship should have a ratio that is specified in kilos per tonne. This may also be of particular use when more complex units are required to be used through the evolution of an end product, consider a liquid that is then bottled, and then a number of these bottles are boxed together. The ratios for the predecessors may need to be in terms of litres per bottle, and bottles per box; with the visibility of this unit data providing a confidence in the entered data that would not be possible otherwise.

Specifying products in terms of a variety of units allows an inbuilt scaling in the problem. It is common for some materials to be required in significantly smaller quantities to others in a production process and if all products were measured throughout the system in the same units, for example kilos, requirements for small volume products may not retain the precision that they should, providing the provision for this scaling should allow better numerical stability in the models.

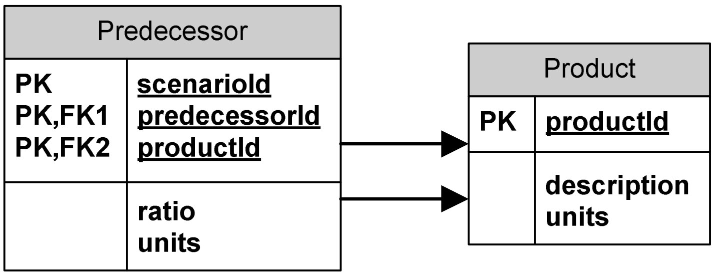

Figure 3-21 - Predecessor Requirement Data

3.3.3.4 Transport of Products within the Supply Chain

The relational entity, *TransportInternal*, provides the permitted transfer of particular products between Facilities within the supply chain and the cost of transporting the products. The unit of the cost can be specified allowing the expected units in which the quantities of product transported are expected to be measured: £ per kilo, £ per box, £ per litre, etc. Again this statement of the units is not intended to provide any conversion facilities, just a visibility to the meaning of the data.

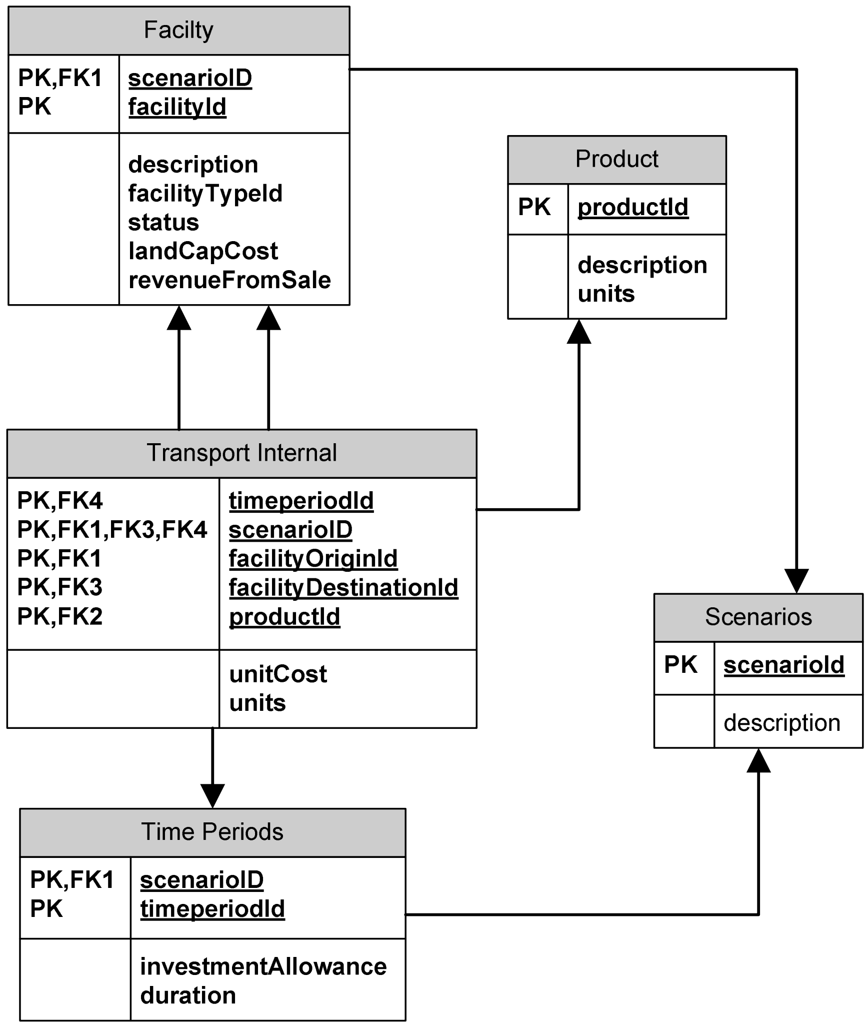

Figure 3-22 - Transport Internal Data

This is also Time period and Scenario dependent data, allowing cost to be varied over time or by scenario. This allows changes in cost to be considered, the company may need to consider the impact of increases in fuel cost, or taxation on fuel and vehicles, or the impact of changing a fleet of vehicles for more efficient ones.

3.3.3.5 Transport of Products to the Customer

The relation entity *TransportCustomer* provides the link between the products with which a customer is supplied and the facilities from which they are supplied. The costs given for transporting a particular product from each facility to each customer not only provides the data about how much it costs to take the product to the customer, but also whether the supply of any given customer is permissible from each Facility.

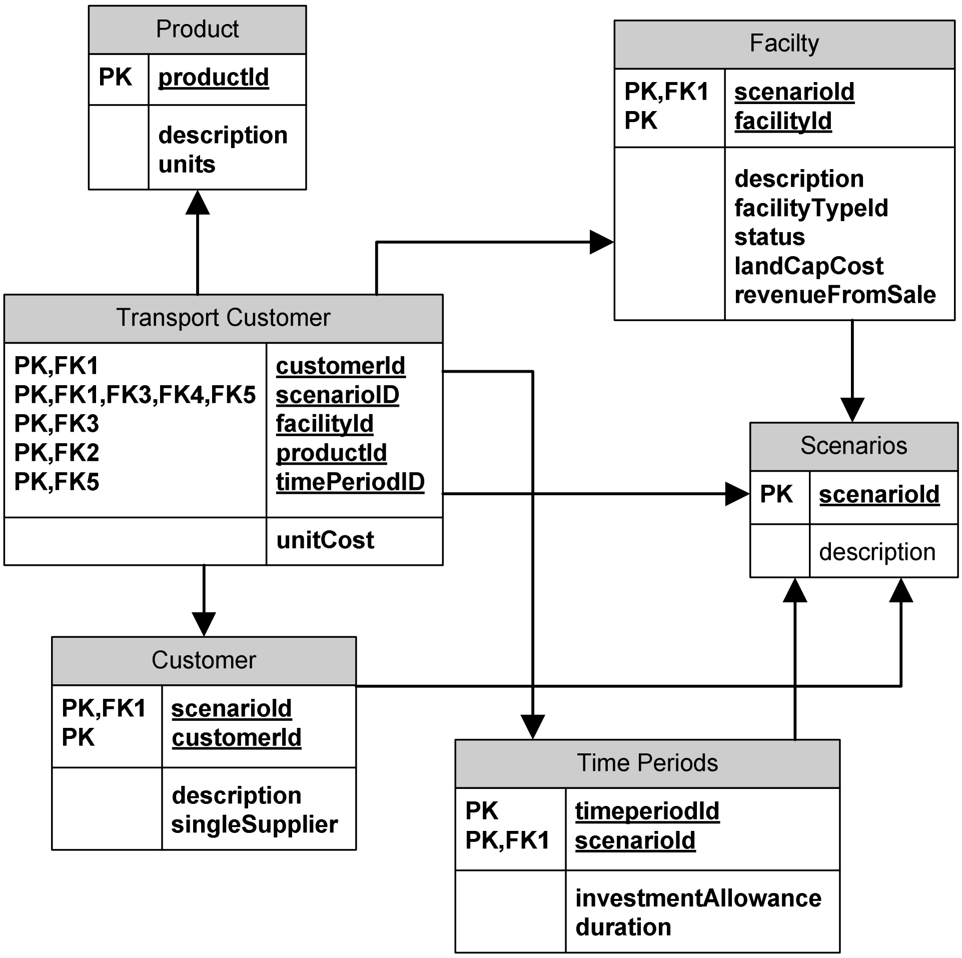

Figure 3-23 - Transport Customer Data

For routes between a facility and a customer that are not permitted, a cost is not specified. Should a zero cost be required, then this can be explicitly entered, making this route permissible at zero cost. These costs can also be varied by time period and by scenario allowing consideration of changing costs over time, or different forecasts for the costs to be considered. Again the units that are being used to measure the various transport costs can be specified allowing a visibility of this information.

3.3.3.6 Demand of Products by the Customers

The relation entity, *Demand*, provides the amount of a product that is to be provided for a customer. The quantity, specifies the amount of a product that is required by a customer in a time period, as with the preceding relations this can be specified by scenario, to allow different demand forecasts to be considered. Again the units of this demand can be specified allowing visibility of the true meaning of the data.

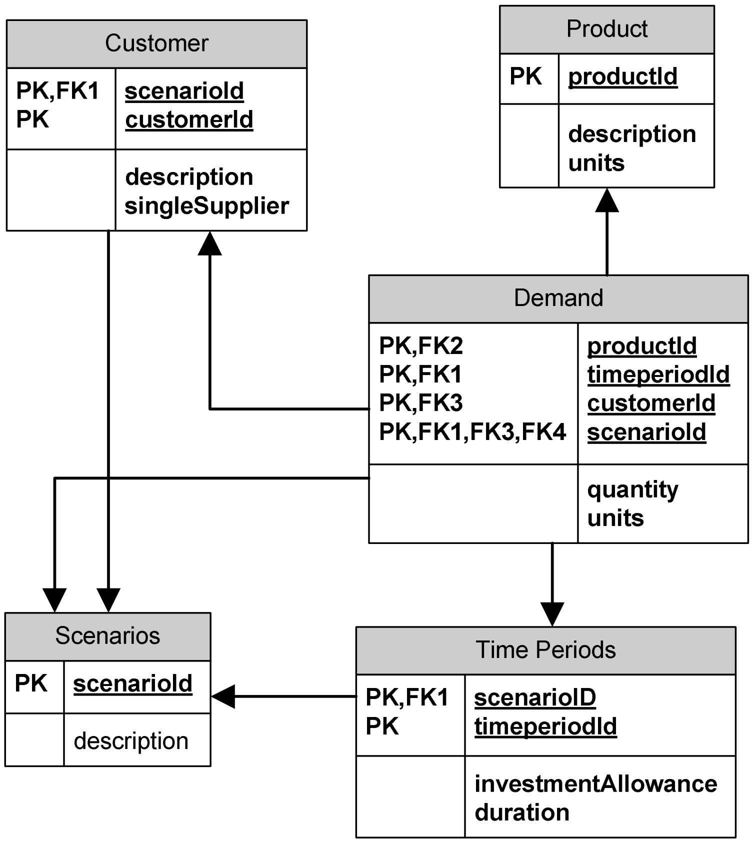

Figure 3-24 - Demand Data

3.3.4 Summary

The combination of these main entities and the relational entities described above produces the following relational database, which has been used for the rest of our investigations.

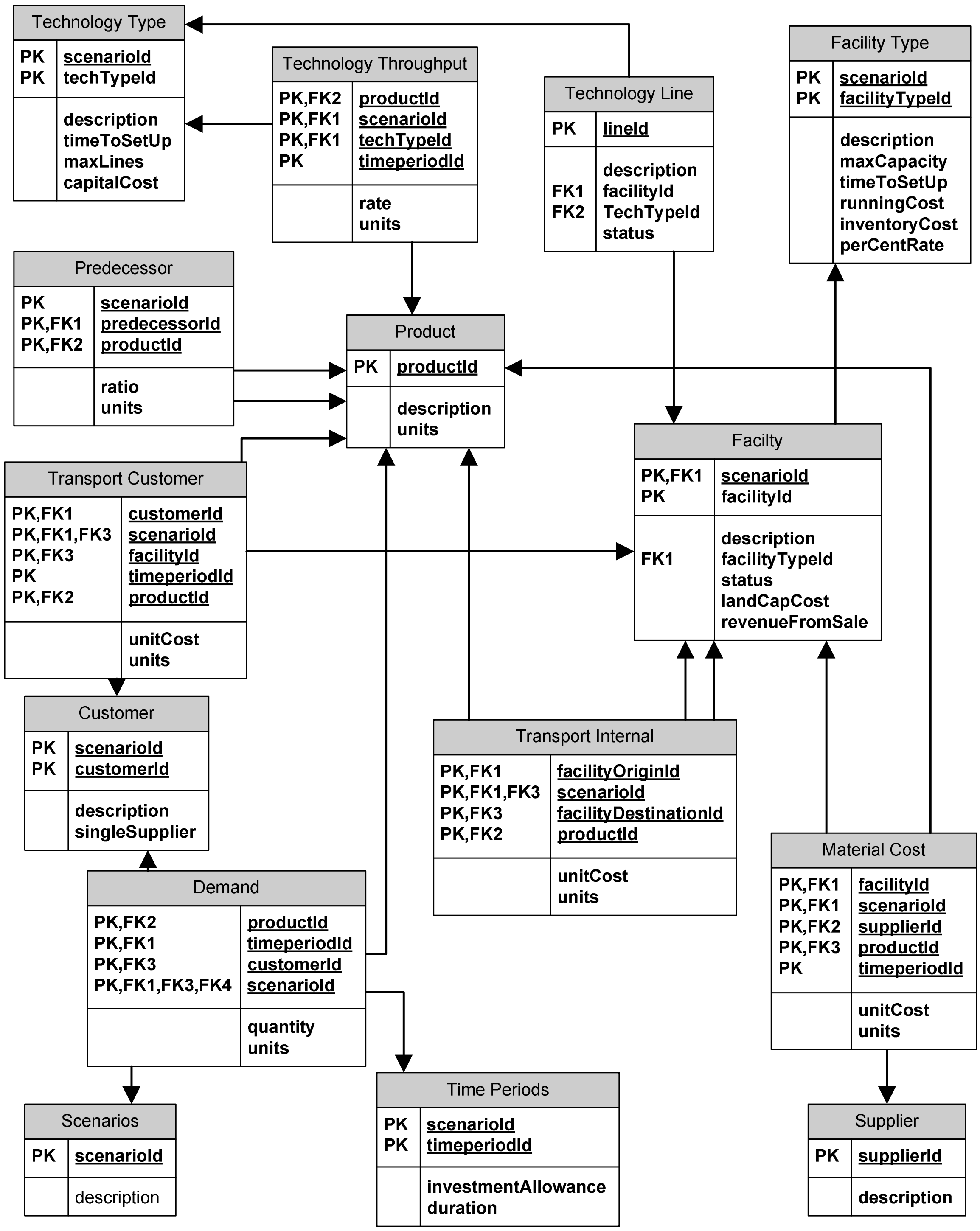

Figure 3-25 - Full Relational Model

4 Problem Instances

4.1 Introduction

Here we describe the problem instances for which we were able to obtain data.

4.2 Osiris Problem Instance

4.2.1 The problem

The main realistic-size set of data that was available for this work was the Osiris data Baricelli (1996). This problem has been translated into this framework as a supply chain which includes factories, distribution centres and customers. At the factories intermediate and final products are produced. These are then sent to distribution centres where they are labeled and dispatched to the customer.

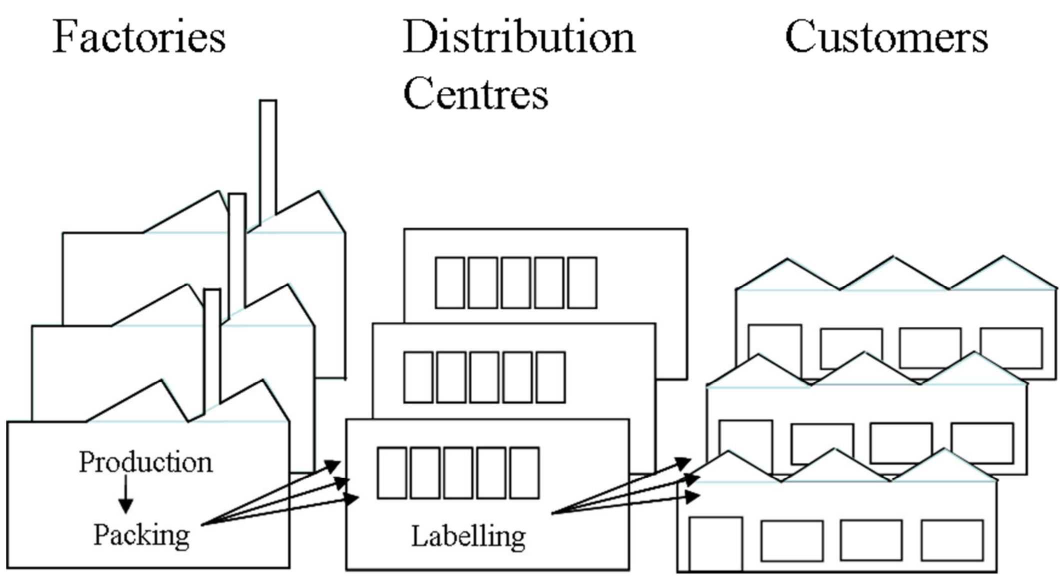

Figure 4-1 - Problem Instant Entities

The problem is strategic, but includes the underlying tactical decisions. It aims to identify the best long term decisions about which factories and distribution centres to open, retain or close, and within each of these which production lines should remain in use and for how long and which new ones should be selected and when they should be installed.

The Strategic decisions need to be made within the deterministic framework of tactical/operational decisions including:

- Meeting customer demands for each product in each time period. A shortage quantity is included in the model and this allows penalties to be imposed for shortages. The problem we have been studying has such penalties imposed.
- Production flows: each product has a set of predecessor products, which are represented in the model using a consumed quantity for that predecessor product. The current problem data has three production stages and the products and their predecessors have one-to-one relationships. The model will allow data with much more complexity and could deal with multiple parts being assembled, multiple products being packed or items being cut or moulded into multiple products. The data required to specify these flows is the requirement of any predecessor product for each product, and the possible throughput of a product by a production line.
- Transportation: this involves the cost of different routes both internal and external to the customer. These transportation costs between the sites are also used to specify whether a particular route is permissible either between any two sites or when supplying a customer from a particular site.

4.2.1.1 Problem size

The problem included 23 sites. There are 8 factory sites, of which 5 are currently operational and 3 were potential sites. There are also 15 distribution centres which also were able to carry out packaging tasks, 10 of these are currently operational and 5 are potential sites.

13 products are produced, packed and labeled and distributed to 30 customers. This leads to 39 intermediate and final products to model. All products are produced at a factory site, but they can then either be packed at the same site and then shipped, or shipped and then packed once they reach the distribution centre. All products, wherever they have been packed will be labeled at the distribution centre.

There were 8 different technology types which carry out the three different processing stages.

The model statistics are as follows:

Facilities	23
Existing Technology Lines	82
Potential Technology Lines	564
Technology Line Costs	94
Technology Line Throughput Rates	103
Technology Line Production Costs	1794
Products	39
Predecessor relationships	26
Transport Internal Costs	2639
Time Periods	6
Customer Zones	30
Transport Customer Costs	5785
Demands in each Scenario	1380
Price and Shortage Penalties	2340

Table 4-1 - Model Statistics

4.2.2 Complexity of the problem

The data included 100 different customer demand scenarios. In all of the scenarios for this problem instance the customer demands varied only in terms of the quantity of the products required and not the range of products to be supplied, so the set of products being supplied to a customer were the same in all cases and are shown below (Table 4-2).

Customer Zone	Number of Products	Products
1, 2	5	29, 33, 34, 35, 38
3, 4	6	29, 30, 32, 33, 34, 36
5	3	29, 30, 32, 33, 34, 35, 36
6	3	29, 35, 36
7, 8	4	28, 32, 35, 36
9 - 17	9	27, 29, 30, 34, 35, 36, 37, 38, 39
18 - 26	10	27, 29, 31, 33, 34, 35, 36, 37, 38, 39
27 - 29	6	27, 29, 30, 35, 36, 39
30	5	27, 29, 30, 38, 39

Table 4-2 - Customer Details

The number of products that need to be supplied to customers varies from between 3 and 10, but remained the same for each customer across different time periods and demand scenarios, although the quantity of the demands varied.

Product	Number of Customers with demand for the product
27	22
28	2
29	28
30	15
31	9
32	4
33	13
34	22
35	27
36	27
37	18
38	21
39	22

Table 4-3 - Product Demands

The demands of 30 customers for these products have to be fulfilled. We have 100 different demand scenarios including 1380 demand amounts, 230 for each of 6 time periods, the first five covering periods of a year each and a longer final time period of 10 years.

Transport is restricted to movement of products from a factory site to a distribution centre (8 starting points and 15 destinations), reducing the theoretical number of transport routes to 3120. However, there are additional restrictions encoded in the transport cost data, with limits these to 2639 routes. There are 5850 theoretical product transports of the 13 final products from the 15 sites to 30 customers, but the cost data reduces these to only 5785.

4.2.3 Results for Osiris Problem Instance

These results obtained from this problem instance are described in other chapters.

4.3 Warehouse Location Problem

This problem is described in CSPLib (Gent 2011) ; problem is listed as Prob034 and the data for 7 problem instances is provided, the cost in these problems have been rounded to integer values. It is also described by Beasley (1988) and data, description and results can be found on the OR-Library pages Beasley (2011). There are two variants of these problems, depending on whether each customer's demand can be satisfied from multiple warehouses or must be supplied by a single warehouse.

4.3.1 Problem Specification

In the *Warehouse Location problem* (WLP) or *Facility Location Problem* (FLP), a company considers opening warehouses (facilities) at some candidate locations in order to supply its existing stores (customers). Each potential warehouse has a maintenance cost, and a designated maximum capacity for the number of stores that it can supply. For the single source problems each store must be supplied by exactly one open warehouse. The supply cost to a store depends on the warehouse.

The goal is to determine which warehouses to open, and which of these warehouses should supply the various stores, such that the sum of the maintenance and supply costs is minimized.

4.3.2 Specification of the Problem using the Generic Model

A direct implementation of these problem instances using our generic supply chain model was relatively straightforward. The warehouses are modelled as potential facilities and the stores as customer facilities, whilst the supply costs become the transfer costs from the supply chain facilities to the customer facilities.

There are no predecessor requirements as there is only a single product, nor any raw material products or inventory to be considered. Only a single line is required at each facility, to model the flow of product through that warehouse to each store that the warehouse is to supply.

The warehouse costs were attributed to the sites, and the capacities which were identical for all warehouses in each problem were attributed to the lines, allowing a line type to used

Shortage penalty costs of a sufficient size were added to make the cost of not supplying a store greater than the costs of supplying the stores. These need to be greater than the cost the maximum of the warehouse fixed cost plus the supply cost to a store for any warehouse store combination, ensuring that the minimisation of the total cost can be used as the objective function.

4.3.3 Results for Warehouse location Problem Instances

A number of the benchmark problem instances were solved for both the standard and single source variants of the problems. Table 4 gives a comparison of the results from the model described here and those quoted in Beasley (1988) for the standard version of the problem in which each customer can be supplied by more than a single supplier warehouse. The original paper Beasley (1988) quotes the number of warehouses opened in each case in the optimal solution, but does not give objective values.

Problem	Cost	Warehouses	Number of Warehouses	Quoted number of warehouses
CapA1	19240822	16,60,62,68,70,79,89	7	7
CapA2	18438047	59,65,70,79,83,89	6	6
CapA3	17828010	16,59,60,70,79	5	5
CapA4	17346752	16,59,70,79	4	4
CapB1	13656380	12,14,37,59,60,69,70,76,88,90,97	11	11
CapB2	13368137	37,46,60,69,70,82,88,90,98	9	9
CapB3	13234409	37,46,59,60,70,82,85,90	8	8
CapB4	13082516	37,57,59,60,70,88,90	7	7
CapC1	11646597	6,9,14,24,35,42,48,66,70,79,87	11	11
CapC2	11570340	6,9,14,24,35,42,66,70,79,89	10	10
CapC3	11572855	6,9,12,14,24,35,66,70,79	9	9
CapC4	11505767	6,14,24,35,53,70,79,81,89	9	9

Table 4-4 - Results for the Capacitated Warehouse Location Problems

Table 5 gives comparison of results obtained for the single source problem and results from OR Group, Pisa (2011) and OR-Lib (Beasley 2011).

Problem	Best Integer Solution	Warehouses	Number of Warehouses	Quoted Optimal Solution

Cap61	932616	1,2,3,4,6,7,8,9,11, 12,13	11	932616
Cap62	977799	1,2,3,4,6,7,8,11,13	9	977799
Cap63	1014100	2,3,4,6,7,8,11,13	8	1014062
Cap64	1053197	2,3,6,11,12,13	6	1045650
Cap71	932616	1,2,3,4,6,7,8,9,11, 12,13	11	932616
Cap72	977799	1,2,3,4,6,7,8,11,13	9	977799
Cap73	1010641	3,7,8,11,13	5	1010641
Cap74	1034977	3,11,12,13	4	1034977
Cap91	691648	1,2,4,6,7,8,9,11,13,17, 18,20,23,24,25	15	796648
Cap92	858109	1,2,4,6,7,11,12,13,17, 23,24,25	12	855734
Cap93	900760	4,6,7,11,13,17,18, 23,24	9	896618
Cap94	950608	6,11,12,13,17,18,24	7	946051
Cap101	796648	1,2,4,6,7,8,9,11,13,17, 18,20,23,24,25	15	796648
Cap102	854704	1,4,6,7,11,12,13,17,23, 24,25	11	854704
Cap103	893782	4,7,11,13,17,23,24,25	8	893782
Cap104	928942	11,13,18,24	4	928942
Cap121	793440	6,7,11,13,15,16,18,23, 27,34,37,41,45,46,49	15	793440
Cap122	854900	6,7,11,13,15,23,25, 27,34,45,46,49	12	852525
Cap123	898266	6,11,12,15,23,27,34, 37,45,46	10	895302
Cap124	950608	13,23,25,27,34,37,46	7	946051
Cap131	793440	6,7,11,13,15,16,18,23, 27,34,37,41,45,46,49	15	793440
Cap132	851495	6,11,13,15,23,25,27,34 ,45,46,49	11	851495
Cap133	893077	6,23,25,27,34,45,46,49	8	893077
Cap134	928942	23,27,37,46	4	928942
CapA1	19263319	16,36,60,62,68,70,79	7	19240822

CapA2	18440383	59,65,70,79,83,89	6	18438047
CapA3	17765202	34,36,59,65,79	5	17765202
CapA4	17163722	34,59,70,79	4	17160439
CapB1	13659766	12,14,37,59,60,69,70,76,88,90,97	11	13656380
CapB2	13362779	24,37,46,59,60,69,70,82,90	9	13361927
CapB3	13201626	37,48,57,59,60,69,70,9	8	13198556
CapB4	13091335	37,57,59,60,70,99,90	7	13082516
CapC1	11647534	6,9,14,24,35,42,48,66,70,79,87	11	11646597
CapC2	11570438	6,9,14,24,35,42,66,70,79,89	10	11570340
CapC3	11522067	6,14,24,35,53,70,79,81	9	11518744
CapC4	11509396	6,9,14,24,58,66,70,79	8	11505767

Table 4-5 - Results for the Single Source Capacitated Warehouse Location Problems

The results show that when these benchmark problems are represented in the generic modelling framework, the resulting models give the same results as the published results for those same problems. This helps to confirm that the modelling framework does give a faithful translation from the generic representation through to mathematical model.

4.4 Summary

The Osiris dataset provided the main validation for the generic framework and the data for testing the extension to this described in this thesis. The warehouse location datasets provided validation, unfortunately it did not prove possible to obtain another large data set to provide further validation, however the framework is consistent with other descriptions of SCPPs.

5 The Mathematical Models

5.1 Introduction

In order to demonstrate the solution models within the Generic Framework a mathematical model description is first given that is not specific to any solution method. This Model is then used as the framework for both Mixed Integer Programming (MIP) and Constraint Programming (CP) models. Both of these models have been implemented and successfully used to solve the problem instances described in the previous chapter.

The MIP model is used in the following chapters. In chapter 5 it is used to consider the potential of model adaptation in solution improvements, in chapter 6 the use of the model within a local search method is investigated and finally in chapter 7 the extension of the model to incorporate uncertainty is considered.

5.2 Generic Framework

For the generic model to be a correct representation of a general supply chain there are a minimal set of indices, variables and constraints that must be defined; along with mappings from the conceptual and data models that have previously been described. Although implementations of a specific model in a mathematical programming language are likely to have additional indices, variables and constraints declared, along with objective functions that allow certain measure of the quality of the solution to be maximised or minimised, they will not need to be included in this minimal framework.

5.2.1 Indices

A minimal set of indices are required for Facilities, Technology Lines, Customers, Products, Suppliers, Time Periods and Scenarios. In the cases of Facilities and Technology Lines two subsets are defined for Existing and Potential Facilities or Technology Lines, where status indicates whether the Facility or Technology Line is an Existing (status 1) or a Potential (status 0) one. There are also subsets defined for the sets of Technology Lines that are located at each specific Facility.

$\mathbf{F} = \{ \text{Facilities}\}$

$\mathbf{F_f} = \{ (f, f_1) \mid f \in \mathbf{F}, f_1 \in \mathbf{F}, f \neq f_1\}$

$\mathbf{F_e} = \{f \in \mathbf{F} \mid status_f = 1\}$

$\mathbf{F_p} = \{f \in \mathbf{F} \mid status_f = 0\}$

$\mathbf{L}$ = {Technology Lines}

$\mathbf{L_e}$ = {l ∈ $\mathbf{L}$ | $status_l$ = 1}

$\mathbf{L_p}$ = {l ∈ $\mathbf{L}$ | $status_l$ = 0}

$\mathbf{L_f}$ = {l ∈ $\mathbf{L}$ | $Facility_l$ = f}

$\mathbf{C}$ = {Customers}

$\mathbf{P}$ = {Products}

$\mathbf{P_p}$ = {(p, p_1) | p ∈ $\mathbf{P}$, p_1 ∈ $\mathbf{P}$, p_1 is the predecessor product of p}

$\mathbf{S}$ = {Suppliers}

$\mathbf{T}$ = {Time Periods}

$\mathbf{T_t}$ = { (t, t_1) | t ∈ $\mathbf{T}$, t_1 ∈ $\mathbf{T}$, t_1 is earlier than t}

$\mathbf{T_n}$= { (t, t_1) | t ∈ $\mathbf{T}$, t_1 ∈ $\mathbf{T}$, t_1 is the time period before t}

$\mathbf{Z}$ = {Scenarios}

5.2.2 Variables

$x_{f,t}$ — Indicating whether a Facility, ***f***, is open in Time Period, ***t***. 1 indicating an open Facility and 0 a closed Facility.

$z_{l,t}$ — Indicates whether a Technology Line, ***l***, is open in the Time Period, ***t***. 1 indicating an open Line and 0 a closed Line.

$q_{f_1,f_2,p,t,z}$ — Transport Internal quantities - the amount of Product, ***p***, transported from Facility, $\boldsymbol{f_1}$, to Facility, $\boldsymbol{f_2}$, at time, ***t***, under Scenario ***z***.

$q_{l,p,t,z}$ — Production quantities – the amount of Product, ***p***, produced on a Line, ***l***, at time, ***t***, under Scenario ***z***.

$q_{f,c,p,t,z}$ — Transport external quantities – the amount of Product, ***p***, transported from Facility, ***f***, to Customer, ***c***, at time, ***t***, under Scenario ***z***.

$i_{f,p,t,z}$ — Inventory quantities – the amount of a Product, ***p***, held in storage at Facility, ***f***, from Time Period, ***t***, to the next Time Period, under Scenario ***z***.

$m_{f,p,s,t,z}$ Raw material Product quantities – the amount of a raw material Product, $\boldsymbol{p}$, supplied by supplier, $\boldsymbol{s}$, to Facility, $\boldsymbol{f}$, in Time Period, $\boldsymbol{t}$, under Scenario $\boldsymbol{z}$.

5.2.3 Constraints

For all Facilities that are initially open, if they become closed in any Time Period then they must remain closed in all the subsequent Time Periods, in the time horizon.

$$f_{f_e,t} \leq f_{f_e,t_1} \qquad \forall f_e \in F_e, \forall (t,t_1) \in T_t$$

For all Facilities that are not initially open, if they are open in a particular Time Period then they must remain open in all the subsequent Time Periods.

$$f_{f_p,t} \geq f_{f_p,t_1} \qquad \forall f_p \in F_p, \forall (t,t_1) \in T_t$$

Similarly for the Technology Lines, we have.

$$l_{l_e,t} \leq l_{l_e,t_1} \qquad \forall l_e \in L_e, \forall (t,t_1) \in T_t$$

$$l_{l_p,t} \geq l_{l_p,t_1} \qquad \forall l_p \in L_p, \forall (t,t_1) \in T_t$$

A Technology Line cannot be opened at a Facility that is not itself open.

$$l_{l,t} \leq f_{f,t} \qquad \forall l \in L_f, \forall f \in F, \forall t \in T$$

The number of Technology Lines cannot exceed the capacity of the Facility.

$$\sum_{l \in L_f} l_{l,t} \leq M_f \qquad \forall f \in F, \forall t \in T$$

Production can only be carried out on a Technology Line that has been installed, and the production is limited by the capacity of the Technology Line.

$$\sum_{p \in P} p_{l,p,t,z} \le l_{l,t} \; M_{l,t} \qquad \forall l \in L, t \in T, z \in Z$$

The quantity of a Product supplied, produced or retrieved from inventory must be at least as much as is used in production of successor Products, transfers to other Facilities, including Customer Facilities and kept as inventory. The quantity of product, ***p***, is given by multiplying the quantity if the predecessor product, $\boldsymbol{p_1}$, used in producing product ***p.***

$$\sum_{l \in L_f} p_{l,p,t,z} + \sum_{(f,f_1) \in F_f} q_{f_1,f,p,t,z} + \sum_{s \in S} m_{p,f,s,t,z} + i_{f,p,t_1,z}$$

$$\ge \sum_{l \in L_f, (p,p_1) \in P_p} R_{p,p_1} \; p_{l,p_1,t,z} + \sum_{(f,f_1) \in F_f} q_{f,f_1,p,t,z}$$

$$+ \sum_{c \in C} q_{f,c,p,t,z} + i_{f,p,t,z}$$

$$\forall f \in F, \forall (t,t_1) \in T_n, \forall z \in Z, \forall p \in P$$

Note: In some cases such as the inventory quantity

$$i_{f,p,t_1,z}$$

t_1 may be defined for a time period before the first time period in order to provide starting conditions.

5.3 The MIP and CP Implementations

The MIP and CP models were implemented using OPL Studio and solved from within OPL using the inbuilt call to CPLEX and ILOG Solver. Later the MIP model was rewritten using C++ and calling CPLEX directly, this allowed the generation of MPS files which provided the means of using alternative MIP solvers to solve the same problem instance.

The following indices, data and variables are used in the definition of the model. Scenario indices have not been included in the model definition for simplicity to improve clarity and because the model was initially used for solving only a single selected Scenario. Inclusion of scenarios will be considered separately in Chapter 7. Some alternative definitions of indices and variables are considered later in this chapter and in Chapter 5.

The two formulations had much in common, so we present both of them here, describing the common features and the differences for each part of the formulation.

In the CP implementation conditional constraints are used, which evaluate to one when the condition is met, and to zero when it is not. For example the conditional $(\boldsymbol{T \geq fc})$ evaluates as one when $\boldsymbol{T \geq fc}$ and to zero when $\boldsymbol{T < fc}$. This allows such conditionals to be used directly in constraints as a binary variable would be used, or in pairs of constraints where if one condition is met then another must also be enforced.

5.3.1 Indices

All of the indices were common across the two formulations

F	set of ***f*** Facilities, {1...|F|}; comprising of Existing Facilities $\boldsymbol{f_e}$, {1...|EF|}; and Potential Facilities $\boldsymbol{f_p}$, {1...|PF|}.
L	set of ***l*** Lines, {1...|L|}; comprising of Existing Lines $\boldsymbol{l_e}$, {1...|EL|}; and Potential Lines $\boldsymbol{l_p}$, {1...|PL|}.
C	set of ***c*** Customers, {1...|C|}..
P	set of ***p*** Products, {1...|P|}.
S	set of ***s*** Suppliers, {1...|S|}.
T	set of ***t*** Time Periods, {1...|T|}.

5.3.2 Subsets

The subset were common across the two formulations.

EF Existing Facilities - subset of ***F*** where the Facility is initially open.

PF Potential Facilities - subset of ***F*** where the Facility is not initially available and can be opened.

EL Existing Technology Lines - subset of ***L*** where the Line exists initially, each Line will be of a specified Technology Line Type ***lt***.

PL Potential Technology Lines - subset of ***L*** where the Line is not available initially, requires the Potential Line to be installed, each Line will be of a specified Line Technology Type ***lt***.

L_{lt} subset of Technology Lines that are of type ***lt***.

L_f subset of Technology Lines that can be installed at Facility ***f***.

PL_f subset of the Potential Technology Lines, ***pl***, that are available for installation at Facility ***f***.

EL_f subset of Existing Technology Lines that are installed at Facility ***f***.

5.3.3 Data

Most data was common across the two formulations, with one additional set of data in the MIP giving the length of each Time Period, allowing any number of variable length Time Periods. Whilst in the CP formulation there were two additional single data items, the number of Time Periods and the length of the full time horizon. In the CP formulation the only variable length Time Period was the final one, and this was given by the difference between the time horizon and the number of Time Periods. Both these formulations allowed the main problem instance that is described in Chapter 3 to be modelled as all the Time Periods except the final one were of the same length.

$Q_{c,p,t}$ Demand quantity of Product, ***p,*** by Customer, ***c***, at time***, t***.

$P_{c,p,t}$ Price of a Product, ***p***, demanded by Customer, ***c***, at time, ***t***.

$W_{c,p,t}$ Penalty for not meeting the demand for Product, p, by Customer, c, at time, t.

$C_{f1,f2,p}$ The cost of transporting Product, p, from Facility, f_1, to Facility, f_2.

$C_{f,c,p}$ The cost of transporting Product p, from Facility f, to Customer c.

$C_{l,p}$ The cost of producing Product p, on Technology Line l.

C_l The running cost of a Technology Line, l.

C_f The running cost of a Facility f.

$C_{f,p,s,t}$ The cost of a Product, p, that is brought into the system at Facility, f, as a raw material Product from supplier, s, in time period, t..

$C_{f,p}$ Inventory cost, at Facility, f for a Product p.

$R_{p1,p2}$ The conversion ratio; the amount of Product, p_1, that is required to produce one unit of Product, p_2.

R_l The production rate of a Technology Line, l.

R_f The percentage rate, the fraction of the capital cost of a Facility, f, to be assigned to a time period.

R_p Inventory capacity of a Product , the amount of capacity that a Product p takes to store when transferred to inventory.

L_l The capital cost of a Technology Line, l.

L_f The land and capital cost for a new Facility, f.

V_l The revenue received from the disposal of a Technology Line l.

$V_{f,t}$ The revenue received from the disposal of a Facility, ***f*** in time period ***t.***

N_f The capacity of a Facility, the number of Technology Lines that can be operational at a Facility, ***f,*** in any Time Period.

M_f The maximum capacity for Products transferred to inventory at a Facility, ***f***, in any Time Period.

M_t The maximum capital available to invest in new Facilities and Technology Lines in Time Period, ***t***.

5.3.3.1 MIP Formulation

D_t The duration of a Time Period, ***t***.

5.3.3.2 CP Formulation

T The number of Time Periods

H The total time horizon length

5.3.4 Variables

5.3.4.1 Binary and Integer Variables

5.3.4.1.1 Binaries in the MIP Implementation

$x_{f,t}$ Whether a Facility, ***f***, is open in Time Period, ***t***.

$y_{f_p,t}$ Whether a Potential Facility, ***p***, is opened in the Time Period***, t***.

$y_{f_e,t}$ Whether a Existing Facility, ***e,*** is closed in the Time Period, ***t***.

$z_{l,t}$ Whether a Technology Line, ***l***, is open in the Time Period***, t***.

$w_{l_p,t}$ Whether a Potential Technology Line, ***p***, is opened in the Time Period***, t***.

$w_{l_e,t}$ Whether an Existing Technology Line, ***e***, is closed in the Time Period, ***t.***

5.3.4.1.2 Binary variables in the CP Implementation

A single binary variable was included for each Facility and each Technology Line to indicate whether the status of the Facility or Line was changed, this is an additional variable that was added to the model for ease of modelling.

x_f Whether the status of the Facility, f, changes in any of the Time Periods, an open Facility closes or a Potential Facility opens.

z_l Whether the status of the Technology Line, l, changes in any of the Time Periods, an Existing Technology is removed or and Potential Technology Line is installed.

5.3.4.1.3 Integer variables in CP Implementation

In the CP implementation the variables describing when a Facility is opened or closed are described by a single variable for each Facility which indicates which Time Period the change in status occurs, so the variables can take any value between 1 and the number of Time Periods T+1, which indicates that the Facility or Technology Line remains in its initial state, and is not opened or closed.

y_{f_p} Which Time Period Potential Facility, f_p, is opened.

y_{f_e} Which Time Period Existing Facility, f_e, is closed

w_{l_p} Which Time Period Potential Technology Line, l_p, is opened. Takes values from 1 to T+1, where the value T+1 indicates that the line does not open.

w_{l_e} Which Time Period Existing Technology Line, l_e, is closed. Takes values from 1 to T+1, where the value T+1 indicates that the line does not close.

5.3.4.2 Continuous Variables

All the continuous variables were common across the two formulations.

$s_{c,p,t}$ Shortage quantities - the amount by which the demand amount is short for each Product, p, that should be delivered to Customer, c, in Time Period, t.

$q_{f1,f2,p,t}$ Transport Internal quantities - the amount of Product, ***p***, transported from Facility, f_1, to Facility, f_2, in Time Period, ***t*.**

$p_{l,p,t}$ Production quantities – the amount of Product, ***p***, produced on Technology Line, ***l***, in Time Period, ***t*.**

$c_{f,p,t}$ Consumed quantities – the amount of Product, ***p***, consumed at Facility, ***f*,** in Time Period, ***t***, in producing successor Products**.**

$q_{f,c,p,t}$ Transport external quantities – the amount of Product, ***p***, transported from Facility, ***f***, to Customer, ***c***, in Time Period, ***t*.**

$i_{f,p,t}$ Inventory quantities – the amount of a Product, ***p***, held in storage at Facility, ***f*,** from Time Period, ***t***, to the next Time Period**.**

$m_{f,p,s,t}$ Raw material Product quantities – the amount of a raw material Product, ***p***, supplied by supplier, ***s***, to Facility, ***f***, in Time Period, ***t*.**

v_1 The total cost of the loss in income due to shortage.

v_2 The total penalty imposed for items of shortage, not a real cost but one that is intended to reflect the impact on the business in terms of lack of service to the Customer.

v_3 The total cost of transporting Products between Facilities belonging to the supply chain owner.

v_4 The total cost of transporting Product to Customers.

v_5 The total cost of producing all the Products that were produced at all the Facilities.

v_6 The total cost of running the Existing Technology Lines.

v_7 The cost of opening and running the Potential Technology Lines

v_8 The cost of opening and running the Facilities.

v_9 The total cost of holding Products in inventory.

v_{10} The total cost of the supply of raw materials.

u The maximum possible revenue, if all the demand is met.

5.3.5 Objective Function Constraints

The default objective function consists of maximising the profit, which is calculated as the maximum possible revenue calculated in constraint (1), less the costs, which are calculated in constraints (2) – (11). The cost of different constituents are modelled separately not only to ease the modelling but also to allow easy analysis of the relative costs.

$$u - v_1 - v_2 - v_3 - v_4 - v_5 - v_6 - v_7 - v_8 - v_9 - v_{10}$$

This separation of these objective cost constituents not only simplifies the modelling process, but it provides a breakdown of the costs from the objective, which is of significant importance in a model that is part of a decision making process. This separation also offers the option of scaling or ignoring one of more of these costs in the objective function by adding a proportion data element for each objective function constituent giving the following objective formulation:

$$u - \delta_1 v_1 - \delta_2 v_2 - \delta_3 v_3 - \delta_4 v_4 - \delta_5 v_5 - \delta_6 v_6 - \delta_7 v_7 - \delta_8 v_8 - \delta_9 v_9 - \delta_{10} v_{10}$$

Where the following give the proportion of the each cost to be included in the objective function

δ_1 – proportion of the shortage costs

δ_2 – proportion of the shortage penalty

δ_3 – proportion of the internal transport costs

δ_4 – proportion of the external transport costs

δ_5 – proportion of the production costs

δ_6 – proportion of the Existing Technology Line costs

δ_7 – proportion of the Potential Technology Line costs

δ_8 – proportion of the Facility costs

δ_9 – proportion of the Inventory costs

δ_{10} – proportion of the Material costs

This allows information to be easily obtained about each aspect of the solution, for example the minimum transport or production cost that are possible, should all the demand be met. It also allows for comparisons to be made between each component of the objective when solving a set of different scenarios for a problem instance.

This also made it possible to enforce a zero shortage cost using the shortage constraint from this set of constraints to force this to zero, this is discussed further in Chapter 5.

The formulation of the objective function and calculation of the maximum revenue were the same in both the MIP and CP formulations. The calculation of the shortage costs and penalties, internal and external transport, production, inventory and material cost were also the same in both formulations, but the cost relating to Facilities and Technology Lines were formulated differently.

5.3.5.1 Maximum possible revenue

A constant value, the summation (1) below gives the amount that would be paid, if all the demands by all of the Customers for all of the Products were met; the product of ***P*** which gives the amount that a Customer, ***c***, would be expected to pay for Product, ***p***, in Time Period ***t*** and ***Q*** the demand for Product, ***p***, made be Customer, ***c***, in Time Period, ***t***.

$$u = \sum_{c,p,t} Q_{c,p,t} \; P_{c,p,t} \tag{1}$$

5.3.5.2 Shortage Costs

The amount (2) that is deducted for shortage, a straight deduction of the amount that would have been paid ***P***, for Product, ***p***, by Customer, ***c***.

$$v_1 = \sum_{c,p,t} P_{c,p,t}\ s_{c,p,t} \qquad (2)$$

5.3.5.3 Shortage Penalty

An additional penalty, the sum (3) of all the penalties, ***W,*** for failing to meet demand of Customer, ***c***, for Product, ***p***, in Time Period, ***t***.

$$v_2 = \sum_{c,p,t} W_{c,p,t}\ s_{c,p,t} \qquad (3)$$

5.3.5.4 Internal Transportation Costs

The total cost (4) of transporting Products from one Facility to another, which is calculated on a linear basis only, using the cost, ***C*** , of transferring a single unit of a Product, ***p***, from one Facility, ***f***, to another Facility, f_1.

$$v_3 = \sum_{\substack{p,t,f \\ f_1 \in F \\ f_1 \neq f}} C_{f,f_1,p}\ q_{f,f_1,p,t} \qquad (4)$$

5.3.5.5 External Transportation Costs

The total cost (5) of transporting the Products to the Customers, again this transportation cost is on a linear basis only, using the cost, ***C***, of transferring a single unit of a Product, ***p***, from one Facility, ***f***, to a Customer, ***c***.

$$v_4 = \sum_{f,c,p,t} C_{f,c,p}\ q_{f,c,p,t} \qquad (5)$$

5.3.5.6 Production Costs

The cost of producing Products on each Line at each Facility (6) is calculated using the quantity, ***p***, of each Product produced and the cost, $C_{l,p}$, of producing the Product, ***p***, on each Technology Line, ***l***, at the Facility, ***f.***

$$v_5 = \sum_{l,p,t} C_{l,p} \; p_{l,p,t} \tag{6}$$

5.3.5.7 Existing Line Technology Costs

There are two parts to this cost. There is the running cost of the Technology Line, l_e, per time unit, is given by $C_{l,p}$, this is multiplied by the time for which the Technology Line is open, ***d,*** whilst any revenue that is received if the Technology Line is closed, V_l, is deducted from this cost .

5.3.5.7.1 MIP Implementation

In the MIP formulation this constraint is formulated by considering the cost for each Time Period, ***t***, during which the Technology Line is open which is given by ***w***, multiplied by the cost per unit time and the duration of the Time Period, D_t, giving:

$$\sum_{l_e,t} C_{l_e} \; D_t \; w_{l_e,t}$$

Any Technology Line that closes will be indicated by binary ***w*** and hence the total revenue received will be given by

$$\sum_{l_e,t} V_{l_e} \; w_{l_e,t}$$

This gives the overall cost (7) for Existing Technology Lines as

$$v_6 = \sum_{l_e,t} D_t \; C_{l_e} \; z_{l_e,t} \; - \sum_{l_e,t} V_{l_e} \; w_{l_e,t} \tag{7}$$

5.3.5.7.2 CP implementation

In the CP implementation the length of time that the Existing Technology Line remains open is obtained directly from the value of the variable $\boldsymbol{w_{l_e}}$. The binary variable value gives the Time Period in which the Existing Technology Line closes with the value T+1 indicating that the line does not close, so the expression $(\boldsymbol{w_{l_e}} - \mathbf{1})$ give the number of Time Period which any Existing Technology Line is open, which is multiplied by the running cost of an Existing Technology Line $\boldsymbol{C_{l_e}}$. For Existing Technology Lines which remain open until the end of the time horizon, the increased cost to include any extended length of the final Period, this only occurs when $\boldsymbol{z_{l_e}}$ is zero, as the status of the Technology Line does not change, so need to be included when $\boldsymbol{(1 - z_{l_e})}$ is one, this is multiplied by the additional time which is given by $\boldsymbol{(H - T)}$ and the running cost per unit of time $\boldsymbol{C_{l_e}}$; giving the following expression.

$$\sum_{l_e} C_{el}\ (w_{l_e} - 1) + \sum_{l_e} C_{l_e} (H - T)\ (1 - z_{l_e})$$

The potential saving from decommissioning of Lines is calculated by summing all the revenues of these Lines when there is a change of status, given by the constraint below.

$$\sum_{l_e} V_{l_e}\ z_{l_e}$$

Giving the total cost (7a) of the Existing Technology Lines

$$\begin{aligned} v_6 = & \sum_{l_e} C_{l_e}\ (w_{l_e} - 1) \\ & + \sum_{l_e} C_{l_e}\ (H - T)\ (1 - z_{l_e}) \\ & + \sum_{l_e} V_{l_e}\ \ z_{l_e} \end{aligned} \qquad (7a)$$

5.3.5.8 Potential Technology Line Costs

For a Technology Line to be opened there is a capital cost of the Line, $\boldsymbol{L}$, and then a running cost, $\boldsymbol{C_l}$. The running cost incurred is dependent upon the duration that the Technology Line is in use.

5.3.5.8.1 MIP Implementation

In the MIP formulation we formulate this constraint by considering the cost for each Time Period during which the Technology Line is open which is given by $\boldsymbol{z}$, multiplied by the cost per unit time and the duration of the Time Period giving:

$$\sum_{l_p,t} C_{l_p} \; D_t \; z_{l_p,t}$$

Any Technology Line that is opened for the first time will be indicated by binary $\boldsymbol{w_{l_p}}$ and the cost incurred will be given by

$$\sum_{l_p,t} L_{l_p} \; w_{l_p,t}$$

giving the following overall cost (8) for Potential Technology Lines

$$v_7 = \sum_{l_p,t} C_{l_p} \; D_t \; z_{l_p,t} \;\; + \sum_{l_p,t} L_{l_p} \; w_{l_p,t} \tag{8}$$

5.3.5.8.2 CP implementation

In the CP implementation the number of Time Period that a Potential Technology Line will be open is given by the total number of Time Periods, $\boldsymbol{T}$, minus the number of Time Period for which the Potential Technology Line is closed which is given by $\boldsymbol{w_{l_p} - 1}$ giving

$$\sum_{l_p} C_{l_p} \; (T + 1 - w_{l_p})$$

This needs to be extended by any length of time that is in the final Time Period which is given by ***H* - *T***, which only is required when the Potential Technology Line is opened, so this additional running cost is given by

$$\sum_{l_p} C_{l_p} \; (H - T) \; z_{l_p}$$

The cost of opening the Technology Line is given by the sum of the costs

$$\sum_{l_p} L_{l_p} \; z_{l_p}$$

giving the total cost (8a) of the Potential Technology Lines

$$v_7 = \sum_{l_p} C_{l_p} \; [(T + 1 - w_{l_p}) + (H - T) \; z_{l_p}]$$

$$+ \sum_{l_p} L_{l_p} \; z_{l_p} \qquad (8a)$$

5.3.5.9 Facility Costs

The cost incurred for each Time Period that a Facility, ***f***, is open is determined by the running cost, ***C***, of the Facility and the duration for which it is open. The capital cost that is incurred for the Facility, ***f***, in each Time Period, ***t***, is calculated using the percent cost, R_f, applied to the capital expenditure, ***L***. This allows the capital costs of a site to be spread across or beyond the time horizon, whilst the investment limit is still applied strictly to the Time Period in which the capital expenditure was made. It also allows the capital cost of sites already in existence to be incorporated if applicable.

5.3.5.9.1 MIP implementation

In the MIP formulation this constraint is formulated as a summation across the Facilities, ***f***, and the Time Periods, ***t***. The sum of costs incurred in each Time Period is multiplied by the duration of the Time Period, ***D***, and these are summed for all the Time Periods in which the Facilities are open. It includes the facility percentage rate, $\boldsymbol{R_f}$, which gives the cost of the capital investment for that facility that is incurred for one unit of time, which when multiplied by the duration of the time period gives the cost for that time period.

$$v_8 = \sum_{f,t} (L_f R_f + C_f)\, D_t \, x_{f,t} \qquad (9)$$

5.3.5.9.2 CP implementation

In the CP formulation the summation is only across the Facilities, however, the constraint is constructed in three parts, the first two giving the cost for Existing Facilities and the third the cost for Potential Facilities.

As the CP formulation only included fixed length Time Periods for all but the final Time Period, the duration that an Existing Facility is obtained directly from $\boldsymbol{y_{f_e}}$, unless the Existing Technology Line remains open into the final Time Period, when an adjustment is made. The number of Time Periods that the Existing Facility is open is given by $\boldsymbol{y_{f_e} - 1}$ and hence the facility cost for this time will be given by

$$\sum_{f_e} (L_{f_e} R_{f_e} + C_{f_e})\,(y_{f_e} - 1)$$

This needs to be adjusted to incorporate any additional time in the final time period when the Existing Facility does not close, when the variable ***y*** indicating the status of the line has not changes takes the value 0, therefore the term ***(1-x)*** is multiply by the cost and the addition length of time ***H-T*** giving

$$\sum_{f_e} (L_{f_e} R_{f_e} + C_{f_e})\,(H - T)\,(1 - x_{f_e})$$

The time that a Potential Facility is open is given by the number of Time Period which it is open, which is given by $\boldsymbol{T+1-y_{f_p}}$ which is then multiplied by percentage of the capital cost to be repaid in the Time period which is given by the product of the Facility capital cost $\boldsymbol{L_f}$ and the Facility percentage rate $\boldsymbol{R}$, plus the running costs $\boldsymbol{C}$ giving

$$\sum_{f_p} (L_{f_p} \; R_{f_p} + C_{f_p})\,(T + 1 - y_{f_p})$$

This is then extended by any additional length of time that is in the final Time Period which is given by $\boldsymbol{H - T}$, when the Potential Technology Line remains open, this is indicated by the status, $\boldsymbol{x_f}$, of the Facility having change and therefore having a value of 1 and gives the following additional running cost.

$$\sum_{f_p} (L_{f_p} \; R_{f_p} + C_{f_p})\,(H - T) \; x_{f_p}$$

Finally the cost of the Lines is given by all that are opened will remain open in the final Time Period so just a single expression with the adjustment is required. The time that the Facility is open is the whole time horizon, ***H,*** minus the time before the opening of the Facility.

$$\sum_{f_p} (L_{f_p} \; R_{f_p} + C_{f_p})\,(H + 1 - y_{f_p})$$

Hence the Facility costs (9a) are given by

$$v_8 = \sum_{f_e} (L_{f_e} \; R_{f_e} + \; C_{f_e})\;(y_{f_e} - 1)$$

$$+ \sum_{f_e} (L_{f_e} \; R_{f_e} + C_{f_e})\,(H - T) \; (1 - x_{f_e})$$

$$+ \sum_{f_p} (L_{f_p} \; R_{f_p} + C_{f_p})\,(H + 1 - y_{f_p}) \qquad \text{(9a)}$$

5.3.5.10 Inventory Cost

The costs, $\boldsymbol{C_{f,p}}$, of storing Products, $\boldsymbol{p}$, at Facilities, $\boldsymbol{f}$, in the Time Periods, $\boldsymbol{t}$.

$$v_9 = \sum_{f,p,t} C_{f,p} \; i_{f,p,t} \tag{10}$$

5.3.5.11 Raw Material Product Costs

The cost of acquiring Raw Material Products, $\boldsymbol{p}$, which includes both the purchase and the transportation from supplier, $\boldsymbol{s}$, for use at Facility, $\boldsymbol{f}$, in Time Period, $\boldsymbol{t}$.

$$v_{10} = \sum_{p,f,s,t} C_{f,p,s,t} \;\; m_{p,f,s,t} \tag{11}$$

5.3.6 The Model Structure Constraints

The model structure constraints have the most significant differences between the MIP implementation and those in the CP implementation. The CP framework allows non linear constraints which more clearly express the requirements of the real world situation.

In the CP implementation conditional constraints are used, which evaluate to one (true) when the condition is met, and to zero (false) when it is not. For example $(\boldsymbol{T \geq y})$ evaluates as one when $\boldsymbol{T \geq y}$ and to zero when $\boldsymbol{T < y}$. This allows such conditionals to be used directly in constraints as a binary variable would be used, or in pairs of constraints such as:

$$(T \geq y\;) \Leftrightarrow (x = 1)$$

which forces $\boldsymbol{x}$ to have the value one if $\boldsymbol{T \geq y}$ and $\boldsymbol{T}$ to be greater than $\boldsymbol{y}$ if $\boldsymbol{f = 1}$.

5.3.6.1 Existing Facilities

5.3.6.1.1 MIP Implementation

If the Existing Facility, $\boldsymbol{f_e}$, cannot be open in Time Period, $\boldsymbol{t}$, if it has been closed in that or any preceding Time Period, $\boldsymbol{t_1}$. Therefore the value of each open variable, $\boldsymbol{x}$, for Time Period, $\boldsymbol{t}$, will be less than or equal to one minus the newly closed variables, $\boldsymbol{y}$, for this and each of the previous Time Periods, $\boldsymbol{t_1}$.

$$x_{f_e,t} \le 1 - y_{f_e,t_1} \qquad t_1 = 1..t, \forall t, \forall f_e \qquad (12)$$

For every Existing Facility, f_e, will still be open in a Time Period, t, if it has not been closed in that or any preceding Time Period, t_1. Therefore the open variable must be greater than or equal to one minus the sum of all the newly closed variables for the previous Time Periods, t_1, where t_1 is the same Time Period or precedes t. An alternative formulation of this constraint is considered in Chapter 5 where an equality constraint is used.

$$x_{f_e,t} \ge 1 - \sum_{t_1=1}^{t} y_{f_e,t_1} \qquad \forall f_e, t \qquad (13)$$

5.3.6.1.2 CP Implementation

A single constraint in the CP formulation is used to link the value of the variable, $\boldsymbol{x_{f_e}}$, which indicates whether an Existing Facility has been closed at any point during the time horizon with the value of the variable, $\boldsymbol{y_{f_e}}$ which specifies the Time Period in which the Existing Facility closed, but takes the value ***T+1*** when the Existing Facility is not closed.

$$(y_{ef} \le T) \Leftrightarrow (x_{ef} = 1) \qquad (12a)$$

5.3.6.2 Potential Facilities

5.3.6.2.1 MIP Implementation

Any Potential Facilities, f_p, which are closed initially, must be open in Time Period, t, if it has been opened in any preceding Time Period, t_1. Therefore the open variable, x will be greater or equal to each of the newly opened variables, y, for each Time Period pair, t, t_1, where t_1 is the same Time Period or precedes Time Period t.

$$x_{f_p,t} \ge y_{f_p,t_1} \qquad t_1 = 1..t, \forall t, \forall f_p \qquad (14)$$

A Potential Facility, f_p, will remain closed in Time Period, t, if it has not been opened in the current or any of the preceding Time Periods, t_1. Therefore the value of each open variable, $\boldsymbol{x}$, will be less than or equal to the sum of all the previous Facility newly opened variables, $\boldsymbol{y}$. An alternative formulation of this constraint is considered in chapter 5 where an equality constraint is used.

$$x_{f_p,t} \leq \sum_{t_1=1}^{t} y_{f_p,t_1} \qquad \forall f_p, t \qquad (15)$$

In the MIP formulation we also require a constraint for each Facility to prevent Facilities being opened or closed more than once.

Each Existing Facility, $\boldsymbol{f_e}$, can only be closed once, therefore the sum of the newly closed variable, $\boldsymbol{y}$, over all the Time Periods, $\boldsymbol{t}$, cannot exceed 1.

$$\sum_{t} y_{f_e,t} \leq 1 \qquad \forall f_e \qquad (16)$$

Each Potential Facility can only be opened once, therefore the sum of the newly opened variables, $\boldsymbol{y}$, over all the Time Periods, $\boldsymbol{t}$, cannot exceed 1.

$$\sum_{t} y_{f_p,t} \leq 1 \qquad \forall f_p \qquad (17)$$

Alternative formulations of these constraints are considered in Chapter 5.

5.3.6.2.2 CP Implementation

A single constraint in the CP formulation is used to link the value of the variable $\boldsymbol{x_{f_p}}$ which indicates whether a Potential Facility has been closed at any point during the time horizon with the value of the variable $\boldsymbol{y_{f_p}}$ which specifies the Time Period in which the Potential Facility opened, again with the value $\boldsymbol{T+1}$ indicating that the Facility is not opened. .

$$(T \leq y_{f_p}) \Leftrightarrow (x_{f_p} = 1) \qquad (14a)$$

5.3.6.3 Existing and Potential Technology Line constraints

5.3.6.3.1 MIP Implementation

Every Potential Technology Line, $\boldsymbol{l_p}$, is closed initially, therefore any Potential Technology Line must be open in Time Period, $\boldsymbol{t}$, if it has been opened in any preceding Time Period, $\boldsymbol{t_1}$. Therefore the open variable, $\boldsymbol{z}$ will be greater or equal to each of the newly opened variables, $\boldsymbol{w}$, for each Time Period pair, $\boldsymbol{t}$, $\boldsymbol{t_1}$, where $\boldsymbol{t_1}$ is the same Time Period or precedes Time Period $\boldsymbol{t}$.

$$z_{l_p,t} \geq w_{l_p,t_1} \qquad t_1 = 1..t, \forall t, \forall l_p \tag{18}$$

A Potential Technology Line, $\boldsymbol{l_p}$, will remain closed in Time Period, $\boldsymbol{t}$, if it has not been opened in the current or any of the preceding Time Periods, $\boldsymbol{t_1}$. Therefore the value of each open variable, $\boldsymbol{z}$, will be less than or equal to the sum of all the previous Facility newly opened variables, $\boldsymbol{w}$.

$$z_{l_p,t} \leq \sum_{t_1=1}^{t} w_{l_p,t_1} \qquad \forall l_p, t \tag{19}$$

Every Existing Technology Line, $\boldsymbol{l_e}$, is open initially, therefore every Existing Technology Line that is still open in a Time Period, $\boldsymbol{t}$, must not have been closed in any preceding Time Period, $\boldsymbol{t_1}$. Therefore the open variable, $\boldsymbol{z}$, must be less than or equal to one minus the newly closed variable, $\boldsymbol{w}$, for each pair of Time Periods, $\boldsymbol{t}$, $\boldsymbol{t_1}$, where $\boldsymbol{t_1}$ is the same Time Period or precedes $\boldsymbol{t}$.

$$z_{l_e,t} \leq 1 - w_{l_e,t_1} \qquad t_1 = 1..t, \forall t, \forall l_e \tag{20}$$

The Existing Technology Line, $\boldsymbol{l_e}$, remains open in Time Period $\boldsymbol{t}$ if not closed in that or a preceding Time Period, $\boldsymbol{t_1}$,. Therefore the value of each open variable, $\boldsymbol{z}$, for Time Period, $\boldsymbol{t}$, will be greater than one minus the sum of all the newly closed variables, $\boldsymbol{w}$, for the previous Time Periods, $\boldsymbol{t_1}$.

$$z_{l_e,t} \geq 1 - \sum_{t_1=1}^{t} w_{l_e,t_1} \qquad \forall l_e, t \qquad (21)$$

Potential Technology Lines can only be opened once, therefore the sum of the newly opened variable, ***w***, for all the Potential Technology Lines, $\boldsymbol{l_p}$, over all the Time Periods, ***t***, cannot exceed one.

$$\sum_t w_{l_p,t} \leq 1 \qquad \forall l_p$$

(22)

Existing Technology Lines, are in use initially and can only be closed once, therefore the sum of the newly closed variable, ***w,*** for all the Existing Lines, $\boldsymbol{l_e}$, over all the Time Periods, ***t***, cannot exceed one.

$$\sum_t w_{l_e,t} \leq 1 \qquad \forall l_e \qquad (23)$$

Alternative formulations of these constraints are considered in Chapter 5.

5.3.6.3.2 CP Implementation

In the CP formulation constraints similar to those specified for the Facilities were required to link the variables indicating which Time Period a Technology Line opened or closed with the variables indicating whether the status of the Technology Line had changed.

A Potential Technology Line has a status one, indicating a change of status, if it is closed in a Time Period before the end of the time horizon, so if $\boldsymbol{w_{l_e}}$ has a value greater or equal to the number of Time Periods, ***T***, then it has been closed and the binary variable $\boldsymbol{z_{l_e}}$.

$$(T \geq w_{l_p}) \Leftrightarrow (z_{l_p} = 1) \qquad \forall l_p \qquad (18a)$$

An Existing Technology Line has a status one, indicating a change of status, if it is opened in a Time Period before the end of the time horizon, so if $\boldsymbol{w_{l_e}}$ has a value greater or equal to the number of Time Periods, ***T***, then it has been opened and the binary variable $\boldsymbol{z_{l_e}}$, indicates a change of status.

$$(T \geq w_{l_e}) \Leftrightarrow (z_{l_e} = 1) \qquad \forall\, l_e \tag{20a}$$

5.3.6.4 Facility Technology Line Capacity Constraint

There is a limit on the number of Technology Lines that can be accommodated at a Facility. The sum of the Existing Technology Lines still in use in a Time Period and new Technology Lines that have been opened, in the Time Period or preceding Time Periods, must not exceed the limit, $\boldsymbol{N_f}$, for a Facility, $\boldsymbol{f}$. These constraints also force the number of Technology Lines at the Facility to be zero if the Facility is not open.

5.3.6.4.1 MIP Implementation

The MIP implementation sums the binary variables indicating the Technology Lines open at a Facility in a Time Period and constrains this to be less than the Facility Capacity when the Facility is open.

$$\sum_{l \in L_f} z_{l,f} \leq N_f \; x_{f,t} \qquad \forall f, t \tag{24}$$

This constraint can easily be extended to allow limits to be placed on groups of lines, for example to place a limit on how much packing capacity could be placed at a facility.

$$\sum_{l \in L_f \cap L_{lt}} z_{l,f} \leq N_f \; x_{f,t} \qquad \forall f, \forall lt, \forall t \tag{24a}$$

5.3.6.4.2 CP Implementation

The CP implementation uses conditional expression for Technology Lines and Facilities being open or closed. The following constraint limits the number of Technology Lines open at a Facility to $\boldsymbol{N_f}$ when the Facility is open and to zero when it is closed.

$$\sum_{l_1 \in EL_f} (w_{l_1} \geq t) \; + \; \sum_{l_2 \in PL_f} (z_{l_2} \leq t)$$

$$\leq N_f\,[(y_{f_e,t} \geq t) \;+ (y_{f_p,t} \leq t)\,]$$

$$\forall f, t \qquad (24a)$$

5.3.6.5 Inventory Capacity Constraint

There is a limit on the amount of Product that can be held as inventory at each Facility. The sum of all the Products, $\boldsymbol{p}$, that are carried over from Time Period, $\boldsymbol{t}$, to the next Time Period, at Facility, $\boldsymbol{f}$, cannot exceed the specified limit, $\boldsymbol{M_f}$, and must be zero if the Facility is not open in the Time Period.

5.3.6.5.1 MIP Implementation

In the MIP formulation the variable, $\boldsymbol{x_{f,t}}$, is one if the Facility is open

$$\sum_p i_{f,p,t} \leq M_f \; x_{f,t} \qquad \forall f, t \qquad (25)$$

5.3.6.5.2 CP Implementation

In the CP model a conditional expression is used, and required different formulations for Existing and Potential Facilities. For Existing Facilities this evaluates to one if the Facility closes in a Time Period after the one under consideration.

$$\sum_p i_{f_e,p,t} \leq M_{f_e} \; (y_{f_e} > t) \qquad \forall f_e, t \qquad (25a)$$

For Potential Facilities this evaluates to one if the Facility opens in the Time Period under consideration, or any preceding Time Period.

$$\sum_p i_{f_p,p,t} \leq M_{f_p} \; (y_{f_p} \leq t) \qquad \forall f_p, t \qquad (25b)$$

5.3.6.6 Investment Constraint

There is a limit on the amount of investment, $\boldsymbol{M_t}$, that can be made in each Time Period, $\boldsymbol{t}$. The investment costs are incurred by opening Potential Facilities and Technology Lines, however additional capital can be retrieved for other investments by the disposal of Facilities and Technology Lines. The amount, $\boldsymbol{V_f}$, that may be retrieved is dependent on which Existing Facility, $\boldsymbol{f_e}$, is closed. The capital costs of each Potential Facility, $\boldsymbol{f_p}$, is given by, $\boldsymbol{L_f}$, and of each Technology Line by, $\boldsymbol{L_f}$, which is dependent upon the Potential Line, $\boldsymbol{l_p}$; these are the full capital costs and not the percentage of the cost that is assigned to the cost for each Time Period.

5.3.6.6.1 MIP Implementation

The MIP formulation uses the binary variables in summing the investment costs.

$$\sum_{f_p} L_{f_p} \; y_{f_p,t} + \sum_{l_p} L_l \; w_{l_p,t} - \sum_{f_e} y_{f_e,t} \; V_{f_e} - \sum_{l_e} w_{l_e,t} \; V_{l_e} \le M_t \qquad \forall t \qquad (26)$$

5.3.6.6.2 CP Implementation

The CP constraint formulation has the same structure as the MIP formulation but uses conditional expression instead of the binary variables in summing the investment costs.

$$\sum_{f_p} L_{f_p} (y_{f_p} = t) + \sum_{l_p} L_{l_p} (w_{l_p} = t) - \sum_{f_e} V_{f_e} (y_{f_e} = t) - \sum_{l_e} V_{l_e} (w_{l_e} = t) \le M_t \qquad \forall t \qquad (26a)$$

5.3.6.7 Production Rate Constraint

The amount of each Product, ***p***, that can be produced is limited by the availability of Technology Lines and their production rate. Each Technology Line has a production rate, $R_{l,p}$, for each of the Products that the Line is able to produce; this rate is dependent on the Product, ***p***, and the Technology Line, ***l***. The sum of the available time for each Line, provided by Existing and Potential Technology Lines, must be greater or equal to the time that would be required for the production that is carried out on that Line Type.

5.3.6.7.1 MIP Implementation

The MIP implementation sums the number of Lines open in a Time Period directly from the binary variables indicating whether a site has opened or closed in a preceding Time Period.

$$z_{l,t} \; D_t \geq \sum_p (p_{l,p,t} \,/\, R_{l,p}) \qquad \forall\, l, t \qquad (27)$$

5.3.6.7.2 CP Implementation

The CP formulation has a similar structure, but uses conditional constraints to identify whether a Line is open in the Time Period under consideration. Two constraints for Existing Technology Lines and two constraints for Potential Technologies are required in order to incorporate the longer final Time Period which was used in the CP formulation.

For the both the Potential Technology Lines, l_p, and Existing Technology Lines, l_e, conditionals are used to identify whether each Technology line is in use in each Time Period, ***t***. These are multiplied by the expression ***(H – T)*** for the final Time Periods to incorporate any extended length final time periods. In each case the amount of Product, ***p***, divided by the production rate must be less than this.

$$(w_{l_p} < t) \geq \sum_p (p_{l_p,p,t} \,/\, R_{l_p,p}) \qquad \forall\, l_p, t < T \qquad (27a)$$

$$(H-T)\ (w_{l_p} < \mathrm{T}) \geq \sum_p (p_{l_p,p,T} / R_{l_p,p})$$

$$\forall l_p \tag{27b}$$

$$(w_{l_e} < t) \geq \sum_p (p_{l_e,p,t} / R_{l_e,p})$$

$$\forall l_e, t < T \tag{27c}$$

$$(H-T)\ (lc_{l_p} \geq \mathrm{T}) \geq \sum_p (p_{l_p,p,T} / R_{l_p,p})$$

$$\forall l_e \tag{27d}$$

5.3.6.8 Predecessor Product Constraint

The amount of a Product consumed is dependent on the quantity of a successor Product that is produced and the conversion ratio ***r*** between the consumed Product, ***p***, and the produced Product, $\boldsymbol{p_1}$. The constraint forces the consumed amount to be equal to the amount required for the production of the successor. The consumed Product can include Raw Material Products or intermediate Products depending on the problem instance.

This constraint is the same in both the MIP and CP formulations.

$$c_{f,p,t} = \sum_{l \in L_f, p_1 \in P} R_{p,p_1}\ p_{l,p_1,t} \qquad \forall f, p, t$$

p is the predecessor product of $\boldsymbol{p_1}$ (28)

5.3.6.9 Product Flow Constraint

This constraint balances the flow of Products through the system. At any Facility, ***f***, in any Time Period, ***t***, the amount of a Product, ***p***, that is produced on each Technology Lines, ***l***, at the Facility; plus the amount, ***q***, that is transported to the Facility, ***f***, from other Facilities, f_1, plus the amount of the Product that enters the Facility as Raw Material Product, ***m***, from the suppliers, plus the amount that was placed in inventory, ***i***, in the previous Time Period, must be at least as great as the amount of a Product consumed, ***c***, plus the sum of the amount ***q*** of the Product transferred from the Facility, ***f***, to all the other Facilities, f_2, , plus the amount, ***q***, of the Product transferred to Customers, ***c,*** plus the amount that is transferred to inventory, ***i***, to be used in next Time Period. Most instances of this constraint will simplify, giving a significantly simpler constraint; for example, Products that are consumed are predecessor Products and as such are not usually final Products that would be transferred to a Customer, but this is dependent on the data determining the problem instance. Conversely Raw Material Products that are obtained from suppliers are not usually going to be Products that are produced within the system, transferred between sites, passed into inventory or transported to Customers.

Again this constraint is the same in both the MIP and CP formulations.

$$\sum_{l} p_{l,p,t} + \sum_{f_1} q_{f_1,f,p,t} + \sum_{s} m_{p,f,s,t} + i_{f,p,t-1}$$

$$\geq c_{f,p,t} + \sum_{f_2} q_{f_2,f,p,t} + \sum_{c} q_{f,c,p,t} + i_{f,p,t}$$

$$\forall f,p,t \qquad (29)$$

5.3.6.10 Transport Internal Constraint

This constraint prevents any Products, ***p***, being transported to or from a Facility, ***f,*** unless the Facility is open. There are no differences in the formulation of this constraint in the MIP and CP implementations. The constraint uses a Big M value, ***B***, which is sufficiently large to prevent this constraint limiting the transport of products when a Facility is open.

$$\sum_{\substack{f_1 \in F \\ f_1 \neq f \\ p \in P}} q_{f,f_1,p,t} + \sum_{\substack{f_2 \in F \\ f_2 \neq f \\ p \in P}} q_{f_2,f,p,t} \leq \mathrm{B}\, x_{f,t}$$

$$\forall f, t \qquad (30)$$

5.3.6.11 Demand and Shortage Balance Constraint

This constraint determines the shortage of a Product in respect of the demand for the Product, ***p***, by the Customers, ***c***, in the Time Period, ***t***. The amount delivered from all the Facilities, ***f***, plus the shortage must be equal to the demand. This constraint is the same in the MIP and the CP formulations.

$$\sum_{f} q_{f,c,p,t} + s_{c,p,t} = Q_{c,p,t} \qquad \forall c, p, t \qquad (31)$$

5.3.7 Modelling changes

In both the MIP and the CP approaches several different implementations were used that reduced the number of variables or constraints that would be generated by a problem instance.

5.3.7.1 Removing some Facilities and Technology Line variables from the model

In the MIP both Facilities and Technology Lines have variables which specify whether the Facility can be open in a specific Time Period and which specify which Time Period a Facility and Technology Lines is first opened or closed. All the constraints can be reformulated to just use the variables which specify when Facilities and Technology Lines are first opened or closed.

First consider the Facility variables, ***x***, $\boldsymbol{y_{f_p,t}}$ and $\boldsymbol{y_{f_e,t}}$. The relationship between ***x*** and $\boldsymbol{y_{f_p,t}}$ is given by constraint (13) which can be tightened to an equality relationship.

$$x_{f_e,t} = 1 - \sum_{t_1=1}^{t} z_{f_e,t_1} \qquad \forall f_e, t$$

Whilst the relationship between $\boldsymbol{x}$ and $\boldsymbol{y_{f_e,t}}$ is given by constraint (15) again tightening to an equality relationship.

$$x_{f_p,t} = \sum_{t_1=1}^{t} y_{f_p,t_1} \qquad \forall f_p, t$$

This expression can then be used as to substitute for $\boldsymbol{y}$ in all the other constraints. The Facility cost (9) which was:

$$v_8 = \sum_{f,t} (L_f \; R_f + C_f)\, D_{\mathrm{t}} \; x_{f,\,t}$$

This is replaced by the following two constraints, as the Existing and Potential Facility formulation are different:

$$v_8 = \sum_{f,t} \left[(L_f \; R_f + C_f)\, D_{\mathrm{t}} \; (1 - \sum_{t_1=1}^{t} y_{f_e,t_1}) \right]$$

$$v_8 = \sum_{f_p,t} \left[(L_f \; R_f + C_f)\, D_{\mathrm{t}} \; \sum_{t_1=1}^{t} y_{f_p,\,t_1} \right]$$

Similarly the Inventory Capacity constraint (25)

$$\sum_{p} i_{f,p,t} \le M_f \; x_{f,t} \qquad \forall f, t$$

Can be reformulated to use $\boldsymbol{y_{f_e,t}}$. and $\boldsymbol{y_{f_p,t}}$. instead of $\boldsymbol{x}$ giving the following pair of constraints:

$$\sum_{p} i_{f_e,p,t} \le M_{f_e} \; (1 - \sum_{t_1=1}^{t} y_{f_e,t_1}) \qquad \forall f_e, t$$

$$\sum_{p} i_{f_p,p,t} \le M_{f_p} \sum_{t_1=1}^{t} y_{f_p,t_1} \qquad \forall f_p, t$$

Next consider the technology Line variables, z, $w_{l_p,t}$, and $w_{l_e,t}$. The relationship between z and w is given by constraint (19) which can be tightened to an equality relationship.

$$z_{lp,t} = \sum_{t_1=1}^{t} w_{l_p,t_1} \qquad \forall l_p, t$$

Whilst the relationship between z and w is given by constraint (21) again tightening to an equality relationship.

$$z_{l_e,t} = 1 - \sum_{t_1=1}^{t} w_{l_e,t_1} \qquad \forall l_e, t$$

This leads to the reformulation of the Existing Technology Line cost constraint (7) and Potential Technology Line cost constraint (8) as:

$$v_6 = \sum_{l_e,t} C_{l_e} \; D_t \; (1 - \sum_{t_1=1}^{t} w_{l_e,t_1}) \; - \sum_{l_e,t} V_{l_e} \; w_{l_e,t}$$

$$v_7 = \sum_{l_p,t} \sum_{t_1=1}^{t} C_{l_p} \; D_t \; w_{l_p,t_1} \; + \sum_{l_p,t} L_{l_p} \; w_{l_p,t}$$

Whilst the production rate constraint (27) becomes the following for Existing and Potential Technology Lines:

$$D_t \; (1 - \sum_{t_1=1}^{t} w_{l_e,t_1}) \; \ge \sum_{p} (p_{f,l_e,p,t} \,/\, R_{l_e,p})$$

$$\forall l_e, t$$

$$\sum_{t_1=1}^{t} w_{l_p,t_1} \; D_t \geq \sum_{p} (p_{f,l_p,p,t} \, / \, R_{l_p,p})$$

$$\forall l_p, t$$

The Facility Technology Line Capacity constraint (24) includes both the variable ***l*** and the variable ***f***, substituting for both of these gives rise to the following constraint.

$$\sum_{l_e \in EL_f} (1 - \sum_{t_1=1}^{t} w_{l_e,t_1}) + \sum_{1 \in PL_f} \sum_{t_1=1}^{t} w_{l_p,t_1}$$

$$\leq N_f \; x_{f,t} \qquad \forall f, t$$

5.3.7.2 Grouping of Technology Lines by Type

In both the MIP and the CP implementations Technology Lines can be grouped by Line Types allowing a reduction in the number of variables generated by a problem instance. The variable $p_{l,p,t}$ which is the amount of each Product produced on each Technology Line in each Time Period is replaced by the variable set $p_{f,k,p,t}$ where the quantity of Product produced is not modelled separately for each Technology Lines, but is accumulated for all Technology Lines of Line Types, ***k***, at a Facility, ***f***.

This leads to a number of constraint changes, which make the constraints more complex algebraically, but reduce the number of variables or number of constraints used in many instances.

The production cost constraint (6) becomes

$$v_5 = \sum_{f,k,p,t} p_{f,k,p,t} \; C_{k,p}$$

The Production Rate Constraint (27) becomes a constraint that sums across all the Technology Lines of a Type at a Facility, $\boldsymbol{L_{k,f}}$.

$$\sum_{l \in L_{lt,f}} D_t \; z_{l,t} \geq \sum_{p} (p_{f,k,p,t} / R_{k,p}) \qquad \forall f, k, t$$

The Predecessor Product Constraint (28)

$$c_{f,p,t} = \sum_{l \in L_{k,f}, p_1} p_{f,k,p_1,t} \; R_{p,p_1} \qquad \forall f, p, t$$

$\boldsymbol{p_1}$ is predecessor product of $\boldsymbol{p}$

Whilst the Product Flow Constraint (29) becomes

$$\sum_{l \in L_{k,f}} p_{f,k,p,t} + \sum_{f_1} q_{f,f_1,p,t} + \sum_{s} m_{p,f,s,t} + i_{f,p,t-1}$$

$$\geq c_{f,p,t} + \sum_{f_2} q_{f_2,f,p,t} + \sum_{c} q_{f,c,p,t} + i_{f,p,t}$$

$$\forall f, p, t$$

Consider the implications of this change for the main problem instance described in Chapter 4. Table 5-1 gives the details of the Technology Line and Technology Lines Types that need to be considered at each of the Facilities.

Facility	Technology Lines	Technology Types
1	41	6
2	46	6
3	44	6
4	43	6
5	41	6
6	36	6
7	36	6
8	36	6
9	42	6
10	45	6
11	43	6
12	43	6
13	14	2
14	17	2
15	15	2
16	15	2
17	14	2
18	15	2
19	12	2
20	12	2
21	12	2
22	12	2
23	12	2
All Facilities	646	94

Table 5-1 – Technologies and Technology Types at a Facility

In this problem instance there are 13 products that can be produced on the Technology Lines. If these are indexed by lines then this leads to 8398 production quantity variables, however if these are indexed Technology Lines Types and Facilities then there are only 1222 variables, a reduction by almost a factor of seven. This leads to a reduction of the number of variables in the Production Cost constraint, each of the Predecessor Product constraints and each of the Product Flow constraints, without altering the number of these constraints.

However for the Production Rate Constraint the number of constraints generated is affected. If Technology Lines are used as the index then with the 6 Time Periods that are included in this problem instance 50388 constraints are obtained, whilst only 7332 of these constraints are required when the production quantity is indexed by Technology Line Type, again a difference of just less than a factor of seven.

5.3.8 Problem Structure Changes

Additional optional constraints have been implemented to allow specification of particular requirements of problem instances within the MIP.

5.3.8.1 Limit the number of Facilities that Supply a Customer

A limit can be specified for how many Facilities, f, can supply each Customer, c

5.3.8.1.1 Data

This requires the addition of a set of data, M_c, giving the maximum number of Facilities supplying each Customer.

5.3.8.1.2 Variables

An additional binary variable $r_{f,c}$ is required to specify whether a Facility, f, supplies Customer, c.

5.3.8.1.3 Constraints

The amount that can be transferred from a Facility, f, must be less than or equal to the demand quantity if that route of supply, cf, is chosen and zero if not.

$$q_{f,c,p,t} \leq Q_{c,p,t} \; r_{f,c} \qquad \forall f, c, p, t$$

$$\sum_f r_{f,c} \leq M_c \qquad \forall c$$

5.3.8.2 Restricting the Facilities supplying Customers each Product

A limit can be imposed on how many Facilities, ***f***, are supplying each product to each Customer, ***c***.

5.3.8.2.1 Data

This requires the addition of a set of data, $\boldsymbol{M_{p,c}}$, giving the maximum number of Facility, ***f***, supplying Product, ***p***, to Customer, ***c***.

5.3.8.2.2 Variables

An additional binary $\boldsymbol{r_{f,c,p}}$ variable is required to specify whether a Facility, ***f***, supplies Customer, ***c***, with Product, ***p***.

5.3.8.2.3 Constraints

The amount that can be transferred from a Facility, ***f***, must be less than or equal to the demand quantity if that route of supply, $\boldsymbol{r_{f,c,p}}$, is chosen, and zero if it is not chosen.

$$q_{f,c,p,t} \leq Q_{c,p,t}\ r_{f,c,p} \qquad \forall f,c,p,t$$

The number of route of supply, ***cfp***, that can be chosen to a Customer for a Product is less than the specified maximum, $\boldsymbol{fmx_{p,c}}$.

$$\sum_{f} r_{f,c,p} \leq M_{p,c} \qquad \forall c,p$$

5.3.9 Model Simplification

We have specified the problem using the most general form of each constraint. This has the advantage that there is a high degree of uniformity and consistency in the model, with no special cases having to be specified. This may appear to create many unnecessary variables and constraints, and to make the constraints unnecessarily complex. However the data values from a problem instance are used to control which parts of each constraint are included in each case, using only the terms that are relevant and have non zero coefficients; and only generating the required constraints. We will illustrate two aspects of simplification of the general formulation of the constraints that are driven by the data using the product flow constraint. The first of these is the simplification that comes about due to the structure of different parts of the supply chain and the second is the simplification which comes about if the supply chain under consideration has a less complex structure.

5.3.9.1 Product Flow Constraint

Here we consider the formulation using Technology Line Type, *k*, as an index for production quantities. The most general formulation of this constraint includes production and consumption of multiple Products, on multiple Technology Lines, at multiple Facilities; supply of multiple Products to multiple Customers; and the acquisition of multiple Raw Material Products from multiple suppliers, all of these in multiple Time Periods, with inventory carried between consecutive Time Periods.

$$\sum_{k} p_{f,k,p,t} + \sum_{\substack{f_1 \in F \\ f_1 \neq f}} q_{f,f_1,p,t} + \sum_{s} m_{p,f,s,t} + i_{f,p,t-1}$$

$$\geq c_{f,p,t} + \sum_{\substack{f_2 \in F \\ f_2 \neq f}} q_{f_2,f,p,t} + \sum_{c} q_{f,c,p,t} + i_{f,p,t}$$

$$\forall f, p, t$$

5.3.9.2 Constraint complexity within different parts of the Supply Chain

However, even in the most complex of problem instances, all instances of this constraint should simplify because many of the terms will not exist or have zero coefficients. In order for all the component variables of the constraint to be included in one instance the same Product would have to be able to be acquired by the Facility from a supplier as a raw material, then be consumed and produced in the production process, able to be transported to the Facility from at least one other Facility within the supply chain and to be transported to at least one other Facility in the supply chain; it would also have to be possible to transfer it in and out of inventory and to supply the Product to at least one Customer from this same Facility. Which parts of this constraint become active will be dependent on the function of the Facility in respect of each Product, as defined by the data.

For example, a raw material is likely to be acquired from a supplier and then consumed in the production process, without any other transfers in or out of the Facility. This leads to the following simplification of the product flow constraint for all Products of this sort:

$$\sum_{s} m_{p,f,s,t} \geq c_{f,p,t} \qquad \forall f, p, t$$

or at most the following constraint if inventory balance is included for that Product:

$$\sum_{s} m_{p,f,s,t} + i_{f,p,t-1} \geq c_{f,p,t} + i_{f,p,t} \qquad \forall f, p, t$$

For a Product produced at a Facility, the flow constraint would not usually include any acquisition of the Product as a raw material and no consumption of the Product, and this Product is likely to be transported either to another Facility within the supply chain or to an end Customer, but not usually both, leading to the following constraint if there is no inventory included:

$$\sum_{k} p_{f,k,p,t} \geq \sum_{\substack{f_2 \\ f_2 \neq f}} q_{f_2,f,p,t} \qquad \forall f, p, t$$

$$\sum_{k} p_{f,k,p,t} \geq \sum_{c} q_{f,c,p,t} \qquad \forall f, p, t$$

Not only will the number of variable groups included in a constraint vary depending on the structure of the flow of Products through the supply chain, but the complexity of the supply chain itself, in terms of how many Facilities, Products and Customer are included, along with how many Time Periods are modelled.

Considering the production of a Product that is produced at a Facility and then supplied directly from that Facility to Customers, there will be ***f.t*** constraints of this form, modelling this production and distribution; where ***f*** is the number of Facilities that produce and distribute this Product to Customers and ***t*** is the number of Time Periods considered for the problem instance, as below:

$$\sum_k p_{f,k,p,t} \geq \sum_c q_{f,c,t} \qquad \forall f,t$$

The number of variables in this constraint will be dependent on the number of different Technology Types at a Facility that can produce the Product under consideration, giving the number of production variables that must be summed to give the total production of the Product at the Facility, and upon the number of Customers that are supplied, giving ***c + k*** variables in the constraint. If there is only a single Line Type capable of this type of production then this constraint has only ***c + 1*** variables, and reduces to the following:

$$p_{f,p,t} \geq \sum_c q_{f,c,p,t} \qquad \forall f,p,t$$

The following table (Table 5-2) shows the number of Products involved in each stage of the supply chain, acquisition of raw materials, first, second and final stage production, for the Osiris problem instance when 10 raw materials are acquired from 3 suppliers.

The first line in the table gives the number of Products of each type included in the problem. The supplier line shows the number of suppliers that are involved with the Products. Then the number of Facility Types involved with the Product Type, in this case there is a Facility Type, which we will call a “Factory”, involved with the production of stage one Products from raw materials, and then the production of stage two Products, and another Facility Type, which we will refer to as a “Distribution Centre” involved with the conversion of stage 2 to final Products and capable of production of stage 2 Products. The number of Line Types involved with these conversions is shown, and finally the number of Customers who purchase these final Products. This allows us to give an estimate of the number of product flow constraints generated and their complexity.

		Products			
		Raw Materials	**First Stage Products**	**Second Stage Products**	**Final Products**
	Quantity	10	13	13	13
Number of Facilities dealing with Products	Suppliers	3			
	Factories	8	8		
	Distribution Centres			15	15
	Customers				30

Table 5-2 - Product used at stages of the Supply Chain

The first line in the table gives the number of Products of each type included in the problem. The supplier line shows the number of suppliers that are involved with the Products. Then the number of Facility Types involved with the Product Type, in this case there is a Facility Type, which we will call a "Factory", involved with the production of stage one Products from raw materials, and then the production of stage two Products, and another Facility Type, which we will refer to as a "Distribution Centre" involved with the conversion of stage 2 to final Products and capable of production of stage 2 Products. The number of Line Types involved with these conversions is shown, and finally the number of Customers who purchase these final Products. This allows us to give an estimate of the number of product flow constraints generated and their complexity.

$$\sum_{k} p_{f,k,p,t} + \sum_{\substack{f_1 \in F \\ f_1 \neq f}} q_{f,f_1,p,t} + \sum_{s} m_{p,f,s,t} + i_{f,p,t-1}$$

$$\geq c_{f,p,t} + \sum_{\substack{f_2 \in F \\ f_2 \neq f}} q_{f_2,f,p,t} + \sum_{c} q_{f,c,p,t} + i_{f,p,t}$$

$$\forall f, p, t$$

The total number of constraints that will be generated to model the product flow for this problem instance will be calculated by ***f.p.t***. With 23 Facilities, 8 Technology Line Types, 52 Products and 6 Time Periods, a naive view would give a constraint count of ***f.p.t.***, which could give a count of 7176 constraints for this problem instance and the variable count for each constraint would be $k + f + s + 1 + 1 + f + c + 1 = 2f + k + s + c + 3$, which would give 90 variables in each constraint.

However, more detailed analysis requires the consideration of each type of Product and the Facilities that will be involved with the Product. For the 3 raw material Products there are 13 Facilities involved, giving a constraint count of 234, for stage one Products there are 8 Facilities involved giving a constraint count of 624, for stage two Products there are 23 Facilities and hence a constraint count of 1794, and for the final Product only 15 Facilities are involved giving a count of 1170. Hence the total constraint count is actually only 3198, less than half the 'naïve' figure above.

The complexity of these constraints can also be estimated. For raw materials the constraint reduces to

$$\sum_{s} m_{p,f,s,t} \geq c_{f,p,t} \qquad \forall f, p, t$$

Giving a variable count of $s + 1$; only 4 variables with non zero coefficients in each of the 234 product flow constraints of this type.

For the stage one Products the constraint will have reduced to the form below, with production of the Products taking place at factory type Facilities, and being used for production of the stage two Products at that Facility or transferred to another factory type Facility for use there.

$$\sum_{k} p_{f,k,p,t} \geq c_{f,p,t} + \sum_{\substack{f_2 \in F \\ f_2 \neq f}} q_{f_2,f,p,t} \qquad \forall f, p, t$$

This then gives a variable count of $k_1 + 1 + (f_2 - 1)$; where k_1 is the number of Technology Lines that produce stage one Products and f is the number of factory type Facilities, which in the variable count must be reduced by one, so as not to include the Facility under consideration, as that would be modelling the transfer of Product to itself. This gives a variable count estimate of only 2 + 1 + 7 = 11 for each of these 624 constraints.

For stage two Products, the constraint will simplify to the following for factory type Facilities, where the Product can be produced, but must be transferred out to a distribution centre type Facility for the next stage of production.

$$\sum_{k} p_{f,k,p,t} \geq \sum_{\substack{f_2 \in F \\ f_2 \neq f}} q_{f_2,f,p,t} \qquad \forall f,p,t$$

This gives a variable count estimate of $\boldsymbol{k + f_2}$, where $\boldsymbol{k}$ is number of Technology Lines that produce stage two Products, and $\boldsymbol{f}$ is the number of distribution centre type Facilities; and for this problem instance that is 4 + 15 = 19 for the first 624 of these 1794 constraints. The constraints for the distribution centre type Facilities only includes transfer in of these Products and not transfer out, along with consumption and production; giving the following constraint structure

$$\sum_{k} p_{f,k,p,t} + \sum_{\substack{f_1 \in F \\ f_1 \neq f}} q_{f_1,f,p,t} \geq c_{f,p,t} \qquad \forall f,p,t$$

and a constraint count estimate of $\boldsymbol{k_1 + f_1 + 1}$ where $\boldsymbol{k_1}$ is the number of Technology Line Types producing stage two Products and $\boldsymbol{f_1}$ is the number of distribution centre type Facilities; giving a variable count estimate of 4 + 15 + 1 = 20 for the remaining 1170 of the 1794 constraints for this stage of Products. Again these figures are considerably smaller than the ‘naïve’ figure of 97 variables with non zero coefficients in each constraint, almost reduce to a fifth of the possible terms.

In the case of the sums of transfers between Facilities there may be further reduction in the complexity of the constraint in cases where specific transfers are not permissible. If the cost of transfer is not specified in the data then it is assumed that this transfer route is not permissible, and the variables for this transfer will not be added to the model, reducing the complexity of constraint where they would have been included. This will also be the case where a specific Line Type is not available at a specific Facility, and if there are no Line Types available for the production of a particular Product at a Facility, then the variable will not be created for the relevant production and consumption quantities, and the constraints modelling that Product and that Facility will simplify accordingly.

The final stage Products have the following product flow constraint structure as below:

$$\sum_{k} p_{f,k,p,t} + \sum_{\substack{f_1 \in F \\ f_1 \neq f}} q_{f,f_1,p,t} \geq \sum_{c} q_{f,c,p,t} \qquad \forall f,p,t$$

The constraint count estimate becomes $\boldsymbol{k_1}+\boldsymbol{f_1}+\boldsymbol{c}$, where $lt_1$is the number of Technology Line Types that can produce final Products, $\boldsymbol{f_1}$ is the number of Facilities that can transfer Product to this Facility, which in this case will be the total number Facilities less one, whilst ***c*** is the number of Customer to who the Products can be supplied, giving an estimate of 2 + 22 + 30 = 54 for the 1170 constraints for these final stage Products.

By taking the product of rows and columns give us a measure of the non zeros in model matrix, which give us some measure of the size and complexity of the mathematical model that is generated. Instead of the 7176 constraints each with 97 variables, which would be 696072 non zeros, we have:

- 624 constraints with 13 variables
- 624 constraints with 19 variables
- 1170 constraints with 21 variables
- 1170 constraints with 54 variables

Which gives 107718 non zeros for our more detailed estimate, with the reductions due to model structure, compared to 696072 without this reduction. This is about 15% of the original size. In each case, the values in the data drive the form and the number of constraints, so that the model contains the minimum number of constraints with the minimum number of variables necessary to correctly model this aspect of the problem instance, and as said before this may lead to further reductions in the model size where data does not permit certain transport routes.

5.3.9.3 Constraint complexity within less complex problem instances

In many supply chains the structure will be simpler than the generic structure we have defined. There may be situations where there is only a single Product, a single Technology Line, a single Facility or single Customer to be considered, or the problem only needs to be considered over a single Time Period, and inventory and raw material acquisition may not be included. In these cases the complexity of complete sets of constraints is reduced as the set indices which define them are reduced, removing a whole dimension from the model when the problem is reduced in this way.

For example consider the case where the problem instance only included the production and the supply of a single Product to a set of Customer from a set of Facilities, without any transfer between the Facilities. The resulting constraints would give the amount of the Product produced at the Facility must be at least the amount supplied to the Customer from that Facility, as below:

$$p_f \geq \sum_c q_{f,c} \qquad \forall f$$

This gives a set of *f* constraints of this type, each including $c + 1$ variables.

Another example may be where there is only a single Facility producing a set of Products, in which case the constraint would be more complex but there would only be a single instance of the constraint, as below

$$\sum_k p_{k,p,t} \geq \sum_c q_c$$

It is clear that different supply chain instances will give rise to different complexity of constraints. What we have achieved with the specification we have defined and the way it has been implemented, is to permit the modelling of complex structure when necessary, but for simplification to be driven by the data specification of the model and not to require additional inputs for the problem owner to keep the constraint complexity or the number of constraints generated in the model to a minimum.

5.4 Conclusion

These models along with the database design given in chapter 2 provide the basis for the implementation of this generic supply chain framework and the further investigations that are reported in the following chapters. In chapter 5 it is used to consider the potential of model adaptation in solution improvements, in chapter 6 the use of the model within a local search method is investigated and finally in chapter 7 the extension of the model to incorporate uncertainty is considered.

This implementation, unlike other implementations, is not problem-specific and provides the basis of a system that can be used for the representation and solution of a whole class of problems and not just simple variations of a single problem instance.

The resulting mathematical models are solvable using current hardware and software, even for significant real-world problems.

6 Alternative MIP Formulation, Solution Methods and Extensions

Alternative model formulations can affect how quickly optimisation problems take to solve. The aim of the following investigations was to demonstrate the potential for integrating a variety of formulations into the generic framework, and to show that these different formulations have the potential for improved solution of specific problem instances.

In this chapter seven variations to the original MIP formulations described in the previous chapter are considered. Each of the alternative formulations is described along with the results that were realised when they were employed with the main problem instance which is described in Chapter 3.

In each case, the variation in the generated mathematical model formulations were produced without changing the original generic SCPP representation. This demonstrates the separation of the problem representation and description from the transformation into a mathematical problem formulation. The consistency of the solutions between the different formulations of the problem demonstrates the robustness of the generic framework approach in that the SCPP representation is not dependent on the details of the mathematical formulation, but is still solvable using current hardware and software.

6.1 The Formulations

The seven adaptations of the original MIP formulation investigated were:

- Fractional Variable Formulation, where the variables for transfer of a Product to a customer are in terms of the proportion of the demand that is met, rather than the quantity supplied.
- Linking Constraints, additional constraints providing a direct link between the variables for the transfer of a Product from a Facility and the variables for the Facility being in use.
- Subsets of Linking Constraints, the use of a selection of the linking constraint described above.
- Alternative Formulations for Facility Logical Constraints.
- Special Ordered Sets, as additional constraints or as an alternative to the constraints on sets of variables where only one of the set can be non-zero at one time.

- Symmetry Breaking Constraints, these constraints were added to remove solutions that are effectively identical, such as occur if two identical Technology Lines are available to install at a Facility.
- Forcing Facilities Closed.

6.1.1 Fractional Variable Formulation

The original formulation uses variables which represented the quantity of each Product shipped from a facility to a customer; these are continuous variables with imposed upper bounds. This alternative formulation uses a continuous variable with a lower bound of 0 and an upper bound of 1 for the fraction of the customer demand that is met for a Product by a Facility in a Time period. The shortage variables are also changed to model the fraction of the demand which is not met.

The change in the transportExternal, q, variables leads to the need for changes in constraint groups (29) and (31) as follows:

$$\sum_{f} q_{c,f,p,t} + s_{p,c,t} = 1 \qquad \forall t, p, c \tag{31a}$$

$$\sum_{l \in L_f} p_{l,p,t} + \sum_{\substack{f_1 \in F \\ f_1 \neq f}} q_{f,f_1,p,t} + \sum_{s} m_{s,p,t} + i_{f,p,t-1}$$

$$\geq c_{f,p,t} + \sum_{\substack{f_2 \in F \\ f_2 \neq f}} q_{f_2,f,p,t} + \sum_{c} q_{c,f,p,t}\, Q_{c,p,t} + i_{f,p,t}$$

$$\forall f, t, p \tag{29a}$$

Three constraints (2), (3) and (5) used for construction of the objective function also required changes giving the new formulation below:

$$v_1 = \sum_{c,p,t} P_{c,p,t}\, Q_{c,p,t}\, s_{c,p,t} \tag{2a}$$

$$v_2 = \sum_{c,p,t} W_{c,p,t}\; Q_{c,p,t}\; s_{c,p,t} \tag{3a}$$

$$v_4 = \sum_{c,f,p,t} C_{c,f,p}\; Q_{c,p,t}\; q_{c,f,p,t} \tag{5a}$$

6.1.2 Linking Constraints

An additional group of redundant constraints (in terms of the modeller) directly linking the Facility opening variables with products supplied to Customers were added. This group of constraints may have implications for the performance of the optimisation; they offer the potential of tightening of the model bounds, but add large numbers of constraints to the model.

Demand met from Facility, f, can only be greater than zero if the Facility is open, therefore a constraint can be added to give a direct link between the Facility open value and the quantity of the demand, Q, for Product, p, met by the Facility, in time period, t. The link in the original formulation of the model is indirect, through constraints (24) which prevent a Technology Line being used at a Facility which is not open, constraints (27) which prevents production of a Product on a Technology Line that is not open, and constraints (29) which deals with the transport of Products from Facilities where they have undergone their final transformation.

The formulation of the linking constraint was initially proposed within the fractional formulation that was discussed previously.

$$q_{c,f,p,t} \le f_{f,t} \qquad \forall c,f,p,t \tag{32a}$$

However a second formulation was considered so that these linking constraints could be used in the original formulation.

$$q_{c,f,p,t} \le Q_{c,p,t}\; f_{f,t} \qquad \forall c,f,p,t \tag{32}$$

Tests were carried out to see whether there was more advantage to be gained by tightening the formulation than disadvantage from increasing the size of model by the addition of these constraints.

6.1.3 Subsets of Linking Constraints

Selection of subsets of the linking constraints were considered, in order to try to maintain the positive effect on the lower bounds, whilst still achieving good quality integer solutions. This would need a process which could be automated and used within the generic framework. Hence the selection of this subset needed to be based on some data specified criteria, either from the original problem data, or from some calculated result that could be obtained quickly and easily. We show how the solution of the LP relaxation can be used.

The values of the Facility opening variables for those Facilities from the LP relaxation for our large problem instance that were involved with the supply of Products to customers were considered. Details for Facilities 9-23 only are shown in the table Table 6-1, as facilities 1-8 do not produce Products that are supplied to customers.

Values of $f_{f,t}$						
	Time Periods					
Facilities	1	2	3	4	5	6
9	0	0	0	0	0	0
10	0.175172	0.130887	0.130887	0.130937	0.130968	0.130968
11	0.274056	0.233472	0.233699	0.234625	0.235089	0.235089
12	0.511556	0.327747	0.327747	0.327747	0.327747	0.327747
13	0.389249	0.301909	0.305372	0.33514	0.389249	0.389249
14	0.00295	0	0	0	0	0
15	0.059572	0.011928	0.011928	0.011928	0.011928	0.011928
16	0.292901	0.292901	0.292901	0.292901	0.292901	0.292901
17	0.102395	0.071442	0.071442	0.08475	0.102395	0.102395
18	0.085347	0.085347	0.085347	0.085347	0.085347	0.085347
19	0	0	0	0	0	0
20	0	0	0	0	0	0
21	0.09634	0.105685	0.105918	0.105918	0.105918	0.105918
22	0	0	0	0	0	0
23	0	0	0	0	0	0

Table 6-1 - Facility opening Variables Values from the LP Solution

It can be seen that the opening variables for five of the Facilities were zero in all six time periods. Ordering the Facilities according to the maximum value of the opening variables gives the results shown below.

Opening variable maximum value	
Facilities	Maximum $f_{f,t}$
9	0
19	0
20	0
22	0
23	0
14	0.00295
15	0.059572
18	0.085347
17	0.102395
21	0.105918
10	0.175172
11	0.274056
16	0.292901
13	0.389249
12	0.511556

Table 6-2 - Facility Opening Variable Maximum Value

If these requirements from the LP solution are summed, a total Facility requirement of less than 2 Facilities is indicated. In order to meet other requirements and minimise other costs, more than 2 are likely to be selected in any practical solution; however, it is unlikely that all 10 will be required, and this seems to lend weight to the importance of this decision in the solution process as well as to the problem owner.

A series of experiments were carried out to test different selection criteria for the addition of the linking constraints to be added to the model and the effects on the integer solutions and the lower bounds that were obtained.

6.1.3.1 Selection Criteria

We considered some selection criteria for grouping these Facilities:

- Facility opening values of zero, giving the set of Facilities {9, 19, 20, 22, 23}.
- Facility opening values greater than zero, but less than 0.2, giving the set of Facilities {10, 14, 15, 17, 18, 21}.
- Facility opening values of greater than 0.2, giving the set of Facilities {11, 12, 13, 16}.

6.1.4 Alternative Formulations for Facility Logical Constraints

Two alternative formulations of these constraints were considered, an equality formulation where inequality constraints were replaced by equality constraints; and a difference formulation where constraints used the difference of 'adjacent' variables.

6.1.4.1 Equality Formulation for the Facility Logical Constraints

It is possible to replace the constraint groups (12), (13) and (16) in the original formulation, with a single set of equality constraints (33). Constraints (13) are relaxed versions of these new constraints (33).

$$x_{f_e,t} \le 1 - y_{f_e,t_1} \qquad t_1 = 1..t \qquad \forall f_e, t \qquad (12)$$

$$x_{f_e,t} \ge 1 - \sum_{t_1=1}^{t} y_{f_e,t_1} \qquad \forall f_e, t \qquad (13)$$

$$\sum_{t} y_{f_e,t} \le 1 \qquad \forall f_e \qquad (16)$$

replaced by

$$x_{f_e,t} = 1 - \sum_{t_1=1}^{t} y_{f_e,t_1} \qquad \forall f_e, t \qquad (33)$$

Similarly the constraint groups (14), (15) and (17), again repeated below for clarity, can be replaced with the single set of constraints (34). Similarly Constraints (15) are relaxed versions of these new constraints (34).

$$x_{f_p,t} \geq y_{f_p,t_1} \qquad t_1 = 1..t \qquad \forall f_p, t \qquad (14)$$

$$x_{f_p,t} \leq \sum_{t_1=1}^{t} y_{f_p,t_1} \qquad \forall f_p, t \qquad (15)$$

$$\sum_{t} y_{f_p,t} \leq 1 \qquad \forall f_p \qquad (17)$$

replaced by

$$x_{f_p,t} = \sum_{t_1=1}^{t} y_{f_p,t_1} \qquad \forall f_p, t \qquad (34)$$

6.1.4.2 Difference Formulation for Facility Logical Constraints

The Facility logical constraints (13) and (14) can be replaced by the following formulation, which links the opening or closing of the Facility to the change in the open variable between the previous and current time periods. Again this relies on the additional restriction constraints (16) and (17) on the Facility only opening or closing once.

$$y_{f_e,t} = x_{f_e,t-1} - x_{f_e,t} \qquad t = 1..T, \forall f_e \qquad (13c)$$

$$y_{f_e,0} = 1 - x_{f_e,0} \qquad \forall f_e \tag{13d}$$

$$y_{f_p,t} = x_{f_p,t} - x_{f_p,t-1} \qquad t = 1..T, \forall f_p \tag{14c}$$

$$y_{f_p,0} = x_{f_p,0} \qquad \forall f_p \tag{14d}$$

6.1.5 Special Ordered Sets

A special ordered set of type one (SOS1) was a concept first introduced by Beale and Tomlin (1969). The set as proposed by Beale and Tomlin and described by Williams (1990) places the restriction that one of the variables in the set must be non-zero, and that none of the other variables in the set can be non-zero; this does not allow all of the variables in the set to be zero.

The implementation of SOS1 in the main proprietary software packages is a less rigid constraint on the set; that at most one of the variables can be non-zero, allowing the case where all the variables are zero valued. The solvers CPLEX (IBM 2009), XpressMP (FICO 2009), GAMS (Rosenthal 2011), all describe a SOS1 in this way, as do the modelling languages AIMMS (Bisschop 2011) , with MPL (Maximal, 2008) using the definition of the underlying solver if it supports SOS sets, and ignoring defined SOS in a model where the solver employed does not provide this support.

The effectiveness of the use of SOS was evaluated for the set of variables with and without the other Facility logical constraints.

6.1.6 Formulation

There are several sets of variables in the model where constraints could be formulated using SOS:

- Facility newly opened and newly closed variables for each Facility
- Technology newly open and newly closed variables for each Technology
- Hence constraints (16), (17), (22) and (23) can all be replaced or supplemented by declaring the sets of variables for each Facility or each Technology Line as a SOS1.

6.1.7 Symmetry Breaking

Observation of the sequence of MIP solutions obtained when solving for a single scenario showed that alternative equivalent solutions were being generated due to symmetry. The data for the main problem instance considered contains Technology Lines with identical capabilities and costs that could be used at each site. We carried out further investigations and consider whether such symmetry could be broken and whether this would have a positive impact on the solution process.

6.1.7.1 Formulation

The technology lines were given an ordering based on the ID numbers that they had been given and an additional constraint was added to the model to prevent subsequent Technology Lines in a set of Lines of the same Technology Type being opened before a previous one.

For each pair in the ordering we impose a constraint that the sum of the opening variables for that time period and all the preceding time period for the first line must be greater or equal to the equivalent sum of opening variables for the subsequent line:

$$\sum_{l_1 \in PL_f} w_{l_1,t} \leq \sum_{l_2 \in PL_f} w_{l_2,t} \qquad l_2 > l_1, \forall f, \forall t \qquad (35)$$

6.1.8 Forcing Facilities to be closed

It was observed that some Facilities were not being employed at all in any of the solutions, and when the solutions from the linear relaxation was examined these same Facilities had zero valued variables in that solution as well. It was felt that using the linear relaxation to remove what was effectively a redundant part of the model may be an effective way to improve the solving process, and that this would be a mechanism that could be incorporated within a generic solution framework.

It was observed that seven Facilities {6, 7, 9, 19, 20, 22, 23} had zero valued opening variables in the relaxed solution. These variables were then forced to zero in the model, the problem solved and the solution obtained compared with the original formulation.

Table 6-3 gives the model dimensions for this formulation and for the original formulation.

Model Dimensions				
	Original Formulation		Formulation with Facilities Forced Closed	
	Before presolve	After presolve	Before presolve	After presolve
Constraints	12439	7243	12481	5626
Variables	58080	41926	58080	27805

Table 6-3 - Model Dimensions

The number of variables in the model before pre-solve show the additional 42 constraints added to the model to prevent the opening of the seven Facilities in each of six time periods. However, there is a clear reduction in the problem size after pre-solve.

6.1.9 Forcing Shortage to be Zero

It was observed that for good solutions to the problem there was no shortage, so the effect on the solution process of forcing the shortage to be zero was investigated. Setting the shortage to zero did not reduce the number of constraints in the model, but a small decrease in the number of variable in the model was seen after pre-solve.

6.2 Experimental Results

A brief overview of the main results for all the adaptations is given, followed by a discussion of the main findings for each adaptation; with more detailed results for each of the formulation given in appendix 1.

The comparisons that have been made for these adaptations to the MIP implementation are reported for a single problem instance.

Access to alternative problem instances with sufficiently large data sets was limited to a set of problem instances that varied solely in the demand data that had originally been produced to use for stochastic modelling. The results obtained from these alternative problem instances were similar and hence have not been reported.

Clear differences between the use of formulation was more likely to have been demonstrated if true sets of different problem instances of sufficient size had been available, it is likely that problem instances with different structures would have led to constraints having been generated with different structures and the main variation not having just been the difference in the coefficients of the variables in the constraints.

It is acknowledged that comparisons of these techniques using a single problem instance were not ideal, but as the main aim of these experiments were to demonstrate the feasibility and usefulness of such adaptations to a model within the generic framework this was considered sufficient for the purpose. It provides sufficient evidence that a variety of formulations could be incorporated into the generic framework allowing access to different model formulations that could be tested to find the best choice in a practical situation where solution times are likely to vary between formulations.

The results presented here are from the solution of these models using CPLEX 9.0.

The effectiveness of the formulation is considered in light of the integer solutions and the lower bounds obtained with respect to CPU time and the number of nodes explored in the branch and bound.

A corrected CPU time was used which removed the model build time; this was felt to be a separate issue of software engineering which we were not intending to include in this comparison.

A comparison was also made of the solutions obtained with respect to the number of nodes explored in the Branch and Bound search. This is a commonly reported result in discussion of performance of optimisation problems, and does provide information about whether changes to the model have been effective in pruning the search space and helping to find solutions and bounds without such an extensive search. It also allows comparison to be made between solutions obtained on different hardware. However improvements in pruning the search space are often seen to increase the time taken for each step of the branch and bound search; an indication of this increased time can often be seen in the initial root relaxation time for the problem.

In order to make a combined comparison of the effect of the model changes on the best integer solution and the lower bound we also include observations of the optimality gap obtained with each of these formulations. This is a measure of the relative difference between the best integer solution and the best bound; it is sometimes given as a fractional value and sometimes as a percentage, as we will use here.

$$gap = \frac{|\text{best bound - best integer}|}{\text{best integer}}$$

This gives a measure of the quality of the solution we have obtained and of how confident we are about the quality of the solution. This is a very important measure when optimising real world problems, allowing the user confidence that the solution is of adequate accuracy for the practical application.

A limit was placed on the CPU time and the number of nodes explored for each experiment, but convergence after this time and node limits were very slow and hence this was considered a practical action which would not have an adverse effect on the comparisons that could be made. Also comparisons were made of the model sizes and the root relaxation times.

6.2.1 Summary

The following tables give a summary of the results of using the alternative formulations that have been described previously. Table 6-4Table 6-5 give the best integer solutions that have been found with each formulation at the given corrected CPU times. In this pair of tables objective values that were better at the same corrected CPU time have been highlighted in green. The only case where there was a clear improvement on the original formulation in all cases was when some facilities were forced closed and this is the one case included here which is not a true alternative formulation, but a problem simplification method.

Corrected CPU Time	Original Formulation	Fractional Formulation	Linking Constraints		Linking Constraints Subsets			Facility Logical Constraints	
			Non Fractional	Fractional	Facilities with zero variable values	Facilities with low variable values	Facilities with higher variable values	Difference Formulation	Equality Formulation
100	3423660	8406660			6276380	2435690		4424400	8406660
200	3423660	4924400	6208990		6276380	2435690	3697750	4424400	4924400
500	3423660	4924400	2357310	7013710	6276380	2435690	3491090	4424400	4924400
1000	3423660	4254240	1283250	1592860	6276380	2435690	2020910	4424400	4254240
1500	3423660	1264680	1283250	1044600	6276380	1057280	2020910	4424400	1264680
2000	3423660	1264680	1283250	1044600	6276380	1057280	2020910	2299630	1264680
2500	3423660	1228660	1283250	1044600	6276380	1057280	1766540	2299630	1228660
3000	727310	1228660	1283250	1044600	6276380	1057280	1766540	2299630	1228660
3500	727310	1228660	1283250	1044600	708045	1057280	1766540	2299630	1228660
4000	727310	701212	1283250	1044600	708045	1057280	1766540	2299630	701212
4500	727310	699938	1283250	1044600	708045	1057280	1766540	698378	699938
5000	727310	695701	1283250	1044600	708045	1057280	1766540	698378	695701
5500	685963	695701	1283250	1044600	708045	1057280	1766540	698378	695701
6000	685963	683200	1283250	1044600	708045	1057280	714429	698378	683200
6500	685963	682211	1283250	1044600	708045	696620	714429	698378	682211
7000	685963	681905	1283250	1044600	708045	696620	714429	685546	681905
7500	685963	681905	1283250	1044600	708045	696620	714429	685159	681905
8000	685963	681905	711623	1044600	708045	685322	714429	685159	681905
8500	685963	681905	711623	1044600	694577	684651	714429	685159	681905
9000	685963	681905	692650	1044600	687490	684651	714429	685159	681905
9500	685963	681905	692650	1044600	687490	684651	714429	685159	681905
10000	685963	681905	692650	935249	687490	684651	714429	685159	681905

Table 6-4 - Best Integer Solutions with alternative MIP formulations

Corrected CPU Time	Original Formulation	SOS1 in Addition			SOS1 as Alternative			Symmetry Breaking	Facilities Forced Closed	Shortage Forced to Zero
		On Facilities variables only	On Line Technology variable only	On Facility and Line technology variables	On Facilities variables only	On Line Technology variable only	On Facility and Line technology variables			
100	3423660	6290240	3423660	6290240	4299050	7920470	8530200			
200	3423660	2903910	3423660	2903910	4299050	5297930	8530200		4117350	
500	3423660	2903910	3423660	2903910	4299050	5297930	6063310	9326270	1218870	
1000	3423660	2903910	3423660	2903910	4299050	5297930	6063310	8649350	746940	
1500	3423660	2903910	3423660	2903910	4299050	5297930	6063310	8649350	746940	
2000	3423660	2903910	3423660	2903910	4299050	5297930	6063310	8649350	682235	
2500	3423660	2903910	790621	2903910	4299050	3564000	6063310	6181020	681505	
3000	727310	2903910	790621	2903910	4299050	788935	6063310	6181020	681505	
3500	727310	2903910	790621	2903910	4299050	788935	5895060	1165330	681505	
4000	727310	1521990	695461	1521990	2659980	788935	3062850	1165330	681505	
4500	727310	1521990	695461	691159	1168380	788935	3062850	1165330	681505	698322
5000	727310	1521990	695461	691159	1168380	684071	3062850	717496	681505	684608
5500	685963	1521990	695461	691159	730238	682740	3062850	707648	681505	684608
6000	685963	1517250	695461	691159	730238	682740	1547490	707648	681505	684608
6500	685963	1059280	695461	691159	730238	682740	1547490	707648	681505	684608
7000	685963	752733	695461	691159	690432	682740	1547490	686971	681505	684608
7500	685963	734148	695461	691159	690432	682740	699278	686971	681505	684608
8000	685963	734148	695461	691159	690432	682740	699278	686971	681505	684608
8500	685963	691110	685250	691159	690432	682740	699278	686971	681505	684608
9000	685963	691110	685250	691159	690071	682740	699278	686971	681505	684608
9500	685963	691110	685250	690220	690071	682740	699278	686971	681505	684608
10000	685963	685537	685250	688996	688553	682740	697845	686971	681505	684608

Table 6-5 - Best Integer Solutions with alternative MIP formulations

Alternative Formulations		Root relaxation time	Number of constraint after presolve	Number of variables after presolve	Best Integer solution in time limit (10000s)	Best Integer Solution in node limit (3500)
Original Formulation		38s	7243	41928	685963	684107
Fractional Variable Formulation		18s	7226	41911	681905	681905
Linking Constraints		128s	26548	41911	692650	686956
Fractional Variable and linking Constraints		171s	26548	41911	935249	681906
Subsets of Linking Constraints	Zero opening values	37s	13359	41924	687490	687490
	Smaller opening values	51s	16176	41921	684651	684651
	Larger opening values	87s	12647	41920	714429	683410
Alternative Formulations for Facility Logical Constraints	Difference Formulation	42.7s	6797	41928	685159	685159
	Equality Formulation	71.0s	9360	41928	762066	762066
Special Ordered Sets used as alternative	Facilities only	31.2s	7243	41928	688553	683693
	Technology Lines only	34.8s	6679	41951	682740	682740
	Facilities and Technology Lines	32.4s	6679	41951	697845	681321
Special Ordered Sets used in addition	Facilities only	35.6s	7266	41928	685537	685537
	Technology Lines only	36.2s	7243	41951	685250	685250
	Facilities and Technology Lines	35.0s	7266	41951	688996	682708
Symmetry Breaking Constraints		71s	9829	41928	686971	686971
Forcing Facilities Closed		13s	5626	27805	681505	681505
Forcing Shortage to Zero		37.2s	7423	40548	681843	681843

Table 6-6 - Summary of Model Statistic and Best Integer Solutions

Alternative Formulations		Lower bound in time limit (10000s)	Lower bound in node limit (3500)	Early lower bound (before 500s)	Optimality Gap in time limit (10000s)	Optimality Gap in node limit (3500)
Original Formulation		647926	647926	616629	5.6%	5.3%
Fractional Variable Formulation		663586	663586	633100	2.7%	2.7%
Linking Constraints		647926	663592	663592	6.5%	3.4%
Fractional Variable and linking Constraints		663326	663331	638213	29.1%	2.7%
Subsets of Linking Constraints	Zero opening values	637482	647276	615613	7.3%	5.7%
	Smaller opening values	646948	648443	626475	5.5%	5.3%
	Larger opening values	654750	656460	635961	8.4%	3.9%
Alternative Formulations for Facility Logical Constraints	Difference Formulation	649846	656169	656169	5.2%	4.2%
	Equality Formulation	633421	650243	650243	16.9%	14.7%
Special Ordered Sets used as alternative	Facilities only	664150	666638	623078	3.54	2.5%
	Technology Lines only	645129	645129	621966	5.5%	5.5%
	Facilities and Technology Lines	660052	660300	621026	5.4%	3.1%
Special Ordered Sets used in addition	Facilities only	664677	664851	623117	3.0%	3.0%
	Technology lines only	647696	647696	623688	5.5%	5.5%
	Facilities and Technology Lines	664596	664596	623117	3.5%	2.7%
Symmetry Breaking Constraints		650275	654096	616081	5.3%	4.8%
Forcing Facilities Closed		657065	656937	637145	3.6%	3.6%
Shortage Forced to Zero		655834	655834	617113	3.8%	3.8%

Table 6-7 - Summary of Lower Bounds and Optimality Gaps

The following tables (Table 6-6 and Table 6-7) gives the model statistics, best integer solutions, lower bound and optimality gaps for each of the formulations, and again improvements have been highlighted in green.

6.3 Discussion of Results

Further details, including graphs of results can be found in appendix 1.

6.3.1 Fractional Variable Formulation

The fractional formulation performed better than the original formulation during much of the search time, and ultimately found a better integer solution within the time limit, 681905 in comparison to 685963, a difference of just over 4000 (about 0.6%). When this comparison was made by the number of nodes explored the difference was not as great, as the original formulation obtained a solution of 684107 within the node limit. This was despite the root relaxation time being shorter for the fractional formulation in comparison to the original formulation.

The lower bounds obtained were better regardless of whether by corrected CPU times or numbers of nodes explored were seen beyond 4000 CPU seconds and beyond 150 nodes explored.

6.3.2 Linking Constraints

Adding all the linking constraints increased the size of the model considerably and this persisted after the application of pre-solve as is shown in Table 6-8.

Model Sizes			
Original Formulation		Formulation with linking constraints	
Before Pre-solve	After Pre-solve	Before Pre-solve	After Pre-solve
12439	7243	54304	26548

Table 6-8 - Rows in the model for the two formulations

The root relaxation times were increased, however, integer solutions with objective values under 1,000,000 were obtained earlier with the linking constraint formulations, but the formulation with linking constraints and fractional variables failed to achieve a good integer solution (under 700,000) in the search time.

The effect on the lower bound was similar to the lower bounds that were seen with the use of fractional variables. The optimality gap comparisons reflected the very poor performance in obtaining good integer solutions.

6.3.3 Subsets of Linking Constraints

None of these formulations gave clear improvements in performance in terms of integer solutions, although when the constraints were applied to the set of Facilities with smaller non zero values in the LP solution gave a better integer solution by the end of the search time.

None of the lower bounds achieved were as good as was obtained when all the linking constraint were used, but significant early improvement with the convergence of the lower bound was seen when linking constraints were applied to the two set of Facility variables which had non zero values in the LP solution. The bound achieved using linking constraints on the Facility variable with the larger values in the LP solution, gave an improved bound throughout the search time.

When the results were compared by nodes explored there was little difference to be seen in either the integer solution found or the lower bounds obtained.

6.3.4 Alternative Formulations for Facility Logical Constraints

These alternative formulations had a detrimental effect on the ability to find integer solutions for this problem instance within the specified time limits; with better solutions being obtained at respective times by the original formulation.

The lower bound was also not improved by these changes; hence a better convergence of the optimality gap was seen with the original formulation. Further details of these results can be found in appendix 1.

6.3.5 Special Ordered Sets

Using the SOS1 Constraints on the three sets of variables below, in addition to the constraints in the original formulation and as an alternative gave six different SOS1 formulations to investigate.

- both the Facility and Technology Line variables
- just the Facility variables
- just the Technology Line variables

Comparisons made of the model sizes of each formulation showed only small differences. The addition of SOS1 to just the Technology Line constraints proved the best of these alternatives and although there was no early improvement, better solutions were obtained after 5000s. The formulation with SOS1 constraints as an alternative on both Facility and Technology Line variables also produced a better final integer solution.

The addition of SOS1 constraints to the original formulation constraints on the Facility variables, with or without the Technology Line variables improved the lower bound.

Little difference was seen when the comparison was made with respect to number of nodes explored. However, the solver was able to find better solutions below 700,000 for all three cases where SOS1 constraints were used as an alternative to the original formulation constraints.

The optimality gaps do not show any clear difference due to the improvement in integer solutions obtained coinciding with a degradation of the lower bounds obtained, and visa versa. Root relaxation times were all very similar to those seen with the original formulation.

6.3.6 Symmetry Breaking

The inclusion of these symmetry-breaking constraints increased the model size by over 2500 constraints and this increase in problem size was not reduced at all by the application of the CPLEX Presolve,. The constraints did remove the symmetry from the solutions obtained.

It had been noted that the root relaxation time was increased considerably by the addition of the symmetry breaking constraints, rising from 42 seconds with the original formulation to 71 seconds with the addition of the symmetry breaking constraints. This raised the question as to whether the search had been improved by the addition of the symmetry breaking constraints, but the increased root relaxation time was masking any positive effect on the branch and bound search.

No improvement was seen in the integer solution obtained, but the lower bound was considerably better by this measure.

6.3.7 Forcing Facilities to be closed

In this formulation, constraints were added to the model to force the values of some of the Facility variables to zero, this obviously increased the number of constraints in the model, but following CPLEX Pre-solve there was a considerable reduction in model size. There was a reduction in the number of constraints from 7243 in the original formulation to 5626 and a reduction in the number of variables from 41926 to 27805.

Better integer solutions were found at all times during the search, whether the comparisons were made by corrected CPU time or by the number of nodes explored, and the root relaxation time had been reduced from 38s to only 13s. Lower bounds obtained were also improved regardless of which method of comparison was made.

6.3.8 Forcing Shortage to Zero

Forcing the shortage to zero meant that earlier integer solutions which have been seen to contain high levels of shortage were not found, so obtaining a first integer solutions took longer. The best integer solutions found later in the search were a little better than was seen in the original formulation.

6.4 Summary

The approach taken, to separate the SCPP representation issues as a generic framework, and to explicitly handle the transformation of the SCPP representation into a mathematical formulation has been shown to be robust in that the resulting mathematical problems can be solved to give consistent solutions across a range of different formulations without changing the original SCPP representation in the generic framework.

The different mathematical formulations do exhibit a variety of performance benefits although a clear advantage of one particular formulation over another was not shown. It was shown that different formulation could be incorporated into a generic framework and allow some tuning of the solver for different SCPPs.

7 Strategic Planning with Multiple Scenarios

In this chapter the practicalities of including multiple scenarios in the generic model are considered, along with the implications on the model size and the ability to find good integer solutions.

Analysis of the scenarios is carried out and methods for using multiple scenarios to find more robust solutions are considered, while analysis of the solutions obtained from a range of scenarios offer the potential to simplify the model to be solved.

7.1 Background

When undertaking supply chain planning, most of the data used to make such a plan cannot be known with certainty, and forecasts have to be relied on to provide this missing data. This is true for the demand levels of the products, the cost of facilities and technologies, and also of selling prices for produced goods, buying prices for materials, manufacturing overheads and for transport costs.

We propose a method for incorporating these stochastic features, in terms of scenario specifications, into our generic supply chain model and demonstrate their use within our main problem instance for demand scenarios and demonstrate that they can be solved practically with current hardware and software.

We acknowledge that the addition of such scenario data for all these aspects is likely to be impractical, not only in terms of data generation, but in terms of the size and complexity of the mathematical models that would be generated, and the resultant solution times that would be required. The size of the single scenario model is potentially large and building a combined model for the number of scenarios required to represent real world variability in just one of the sets of the data makes resulting models much larger. Extending the model to include variability in several sets of these aspects makes the model very much bigger again.

Although in most cases it will not be practical to include stochastic variation in all the areas where it occurs, it was considered to be important that these were included within the generic framework. Therefore provision of a means of including scenario data for customer demands of products, costs of raw materials, cost of transportation of products were added to the generic model. Even if it was not practical to include uncertainty in all of these areas, it allows the supply chain planner or analyst to investigate the impact of variations in any one or several of the factors. For example as well as knowing what the effects of changes in demand may have, there may be concerns about the impact of the changes of transportation cost. Some supply chain planners will be well aware of where their vulnerabilities to change lie, whilst others may wish to investigate several areas of change to determine which aspect's variability has most impact upon their problem.

Although for practical reasons the problem instances would currently only be able to include minimal scenario data, the continued improvement of solvers and hardware and the development of the use of techniques such as neighbourhood search as described in chapter 8 is likely to provide the ability to solve larger and larger. The discovery of improved techniques for combining solutions makes the inclusion of all these scenario-dependent sets of data crucial for a truly generic model. Such a generic framework allows investigation of how much detail can practically be incorporated. This has a huge advantage of being able to work at the limit of what is practical, rather than having to make assumptions about whether or not to include such an aspect in a system during the design phase.

There are many well developed techniques in the field of stochastic programming for gaining insight into the effects of uncertainty. These make use of "Wait and See", "Expected value" and "Here and Now" approaches. In the "Wait and See" approach we solve for the best solution to our strategic problem based on a set of scenarios and the probabilities of them occurring. It is the best solution for decision through the time horizon assuming that the assumptions are correct.

7.2 Experiments

Some simple techniques for inclusion of scenario data are considered. However, the intension was not to focus on this area of research other than to clearly demonstrate that our generic framework would be applicable in that respect and that some simple techniques may provide useful information to the supply chain planner or analyst and that the resulting models can be solved in realistic times.

The problems were solved by the mathematical models described in chapters 4 and in chapter 7 we demonstrate the use of neighbourhood search to solve these multi-scenario models.

Since the future is uncertain, making decisions about long-term investment always involves risk. Evaluating a number of possible scenarios is one way of reducing that risk, or at least being able to go some way to evaluating the risk. For the main SCPP instance there was data available to describe 100 different demand scenarios. The framework was used to solve all 100 scenarios separately to get a good solution in each case, providing a range of different investment decisions. The impact of choosing each of these solutions was investigated for a range of other scenarios. For example, if the investment decisions from a high demand scenario are used, but the actual demand is low, what is the impact on production and transportation costs as well as overall profit. The strategic solutions from the 100 scenarios were also each imposed on all the other scenarios and re-solved (requiring the solution of 10000 SCPPs) to get a better understanding of the range of impact of choosing the 'wrong' scenario as a basis for making strategic decisions.

Another way of producing solutions that are less risky is to solve a number of scenarios together and hence find a single strategic solution which is good for all the scenarios chosen. Obviously solving more scenarios together should give more reliable decisions; however, the problem size grows with the number of scenarios solved together. The framework was extended to explore these issues.

7.2.1 Strategic and Tactical Solutions

One approach to finding a robust solution is to find a strategic solution which is good for a wide selection of scenarios. In order to plan across multiple scenarios, a single strategic solution (plan) for all the scenarios is required whilst allowing different tactical solutions for the different scenarios within the framework of that strategic plan. Within the mathematical model this meant that the discrete decisions, the first stage decisions of which facilities and technologies are to be used remained common across all the scenarios, whilst the continuous decisions, the second stage decisions of which products to produce at each facility and the transfers of products between facilities and customers, were allowed to vary for each scenario. This parallels the practical implementation of a supply chain plan where the long term decisions have to be made and can’t be changed later whereas the shorter term tactical decisions can be varied within the set strategic framework.

7.3 Scenario Dependent Model

A model was constructed which enforced a common strategic solution across a set of scenarios but still calculated separate tactical solutions for each scenario. The combined model contained scenario-specific production quantities, internal product transfer rates and quantities supplied to the customer (i.e. the scenario-specific tactical solutions), whilst the facility and technology availability was shared across all the scenarios (the common strategic framework) providing common first stage decisions.

The following sets of variables in the mathematical model had an additional index (z) added for scenario:

Production Quantity	$p_{l,p,t,z}$
Consumed Quantity	$c_{f,p,t,z}$
Transport Internal Quantity	$q_{f1,f2,p,t,z}$
Transport Customer Quantity	$q_{f,c,p,t,z}$
Shortage Quantity	$s_{c,p,t,z}$
Inventory Quantity	$i_{f,p,t,z}$
Raw Material Quantity	$m_{f,p,s,t,z}$

The constraints linking these quantities in the model also had to incorporate the multiple scenarios. The constraints summing the separate costs that make up the objective are easily extended to include all the scenario dependent values. This was implemented by summing each of these separate costs for each scenario; see below for a description of how the shortage cost constraint (2) was adapted. The Total Cost was then constructed from these individual scenario costs, this allowed easy access to the cost of shortage, transportation, production etc, for each individual scenario.

$$v_{1,z} = \sum_{c,p,t,z} s_{c,p,t,z} \; P_{c,p,t,z} \qquad \forall z \qquad (2)$$

Similarly the inventory constraint (22) needs only the inclusion of the scenario specific inventory quantity, and the summation being carried out for each scenario, as well as facility and time period.

$$\sum_{p} i_{f,p,t,sc} \leq M_f x_{f,t} \qquad \forall f,t,z \qquad (25)$$

Similar changes were required to the Production Rate Constraint (27), Predecessor Constraint (28), Product Flow Constraint (29), and the Demand and Shortage Balance Constraint (31).

7.4 Solving Multiple Scenarios

The investigation of the impact of the inclusion of multiple scenarios was carried out using the main problem data set. 100 scenarios were available for this problem, each with different forecasts for customer demand, across the products, customer facilities and time periods, giving 1380 different demand quantities in each scenario.

The impact on the model to be solved when multiple scenarios were included was investigated. The model sizes and the solution times were recorded for increasing numbers of scenarios.

7.4.1 Model Size

Table 7-1shows the effect on the size of the model generated when increasing numbers of these demand scenarios are included in the model.

Number of Scenarios	Before pre-solve		After pre-solve		Integer Variables
	Rows	Columns	Rows	Columns	
1	12439	58080	7243	41928	4014
2	23001	111920	12859	79642	4014
3	33560	165757	18465	117359	4014
4	44119	219594	24081	155076	4014
5	54678	273431	29691	192787	4014
6	65237	327268	35311	230502	4014
7	75796	381105	40931	268217	4014
8	86355	434942	46539	305932	4014
9	96914	488779	52168	343656	4014

Table 7-1 - Problem size for multiple demand scenarios

The addition of each scenario adds approximately 10,500 constraints which is reduced to 5,600 additional constraints after CPLEX pre-solve, and adds approximately 53,800 variables or about 37,700 after CPLEX pre-solve has been applied. The number of integer variables remains constant as these represent the common strategic solution shared by all the scenarios in the model.

7.4.2 Solution Times

Table 7-2shows the impact of this additional model size on the solution time. The time to reach a gap of 10% was used to give a measure of solution time, as in most cases with a single scenario the gap was reduced below this level relatively quickly (within 5000s) with convergence becoming very slow around a 4-6% gap.

The root relaxation times in each case also show the dramatic increases in solving time as more scenarios are included in the model.

Number of Scenarios	Root Relaxation Time	Time to 10% Gap	Increased time for additional Scenario
1	35.54	4607	
2	189.00	15735	11128
3	830.85	28109	12375
4	1526.56	74773	46663
5	2960.96	> 75000	-
6	3863.97	> 75000	-
7	4969.88	> 75000	-
8	8459.4	> 75000	-
9	10725.71	> 75000	-

Table 7-2 - Solution times for multiple demand scenarios

This confirmed that the increased problem sizes and hence solution times were not going to allow the solution of a model which included more than a few scenarios and it was necessary to consider other methods that could provide reasonable methods for considering multiple scenarios.

7.5 Strategic Solution Analysis

Solving the most demanding scenario should give a strategic solution that covers all the scenarios but is likely do so by using a level of investment in facilities and technologies that is excessive for the majority of less demanding scenarios; whilst solving the least demanding scenario will give a strategic solution that does not provide sufficient capacity and leads to shortfalls for the more demanding scenarios. The implication of choosing a high demand strategic plan and the actual demand being low, or choosing a low demand strategic plan and the actual demand being high, can be tested. These results can then be used to give a measure of the risk being taken in each case.

We conducted two series of experiments in which we imposed the strategic solutions from each scenario upon the total set of scenarios.

Each scenario was solved independently generating a set of strategic solutions as the best solutions found for each of these scenarios; where the strategic solutions specify the opening and closing of facilities and technology lines. These scenarios were then solved again forcing the strategic decisions from each of the scenarios upon them in turn. Two approaches were taken. In the first these decisions were imposed in their entirety which we will refer to as the full strategic solution. For the second set of experiments just the facility decisions were imposed which we will refer to as the partial strategic solution.

The full strategic solution provides values for all the integer variables in the problem and hence its imposition reduced the problem to a large LP problem which can be solved relatively quickly and the solutions that were obtained were optimal solutions of these reduced problems. However, when the partial strategic solution was imposed the resulting problem was still a MIP, although much reduced in size.

7.5.1 Results of Imposing Scenario Solutions

When imposing strategic solutions two effects are seen, one is limiting the possible solutions and the other is to reduce the search space and in some cases allow the solver to find better solutions than were found when the scenario was solved independently.

When the full Strategic solution is imposed upon a scenario the main effect is to limit the possible solutions and it is possible to use this to give a measure of how the strategic solution may have an impact on the supply chain's performance in different situations. However, when imposing partial strategic solutions the effect of reducing the search space allows the solver to find better solutions than were found solving the scenario independently in a significant number of cases. The independent solution is the same as is obtained when a strategic solution to a scenario is imposed upon itself.

7.5.1.1 Effects on Scenarios of Imposing the Full Strategic Solution

The following box and whisker plots showing range, inter-quartile range and the median of the solutions (costs) obtained for each scenario when the full strategic solutions were imposed upon the scenario. The independent solutions' costs are shown in black and in all cases can be seen to be the lowest cost solutions, whilst the median costs of the imposed solutions are marked in red. Figure 7-1 is ordered by scenario number, whilst Figure 7-2 by the objective value when the scenario was solved independently.

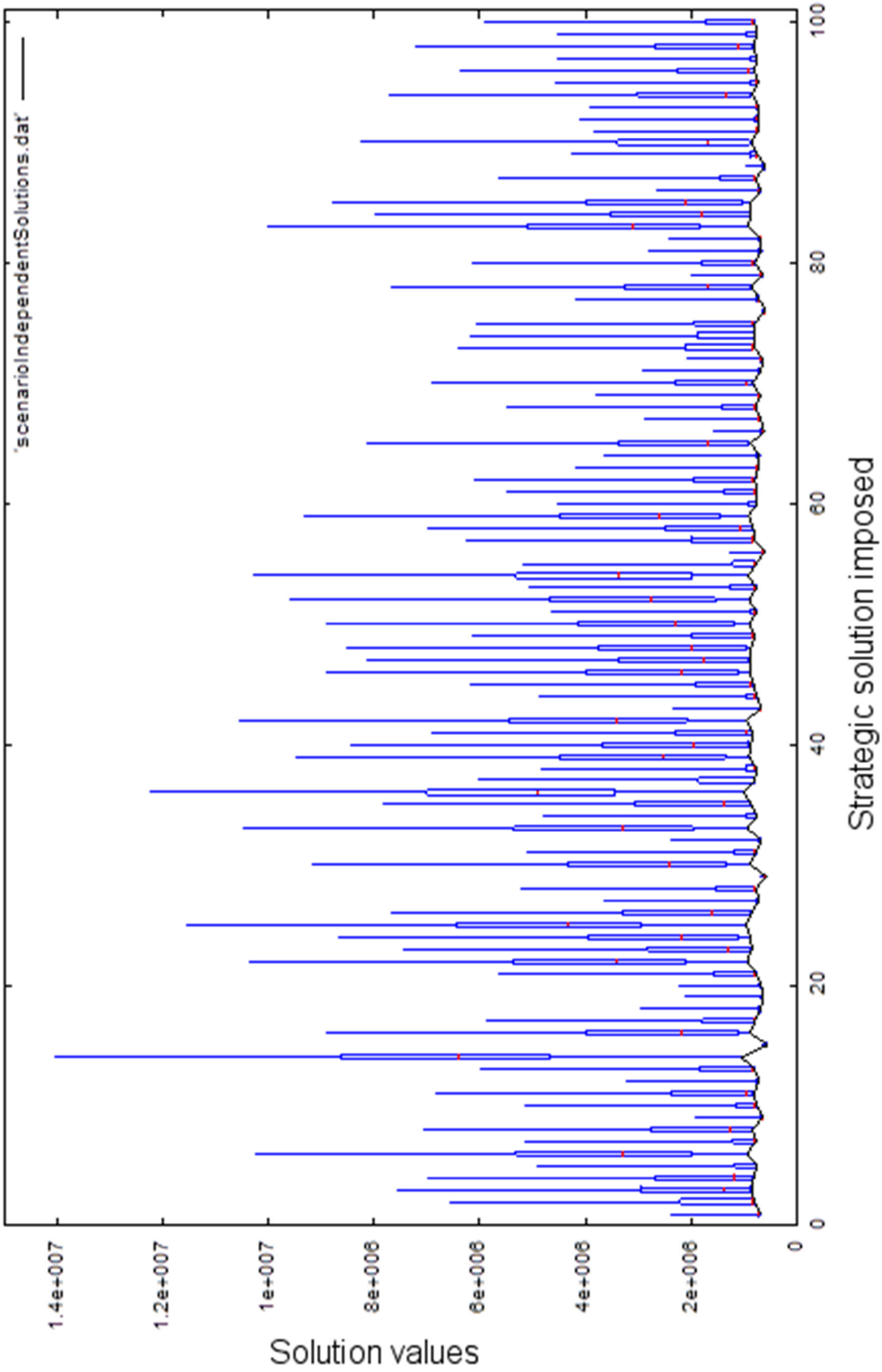

Figure 7-1 - Scenario Solutions when Full Strategic Solution imposed ordered by Scenario number

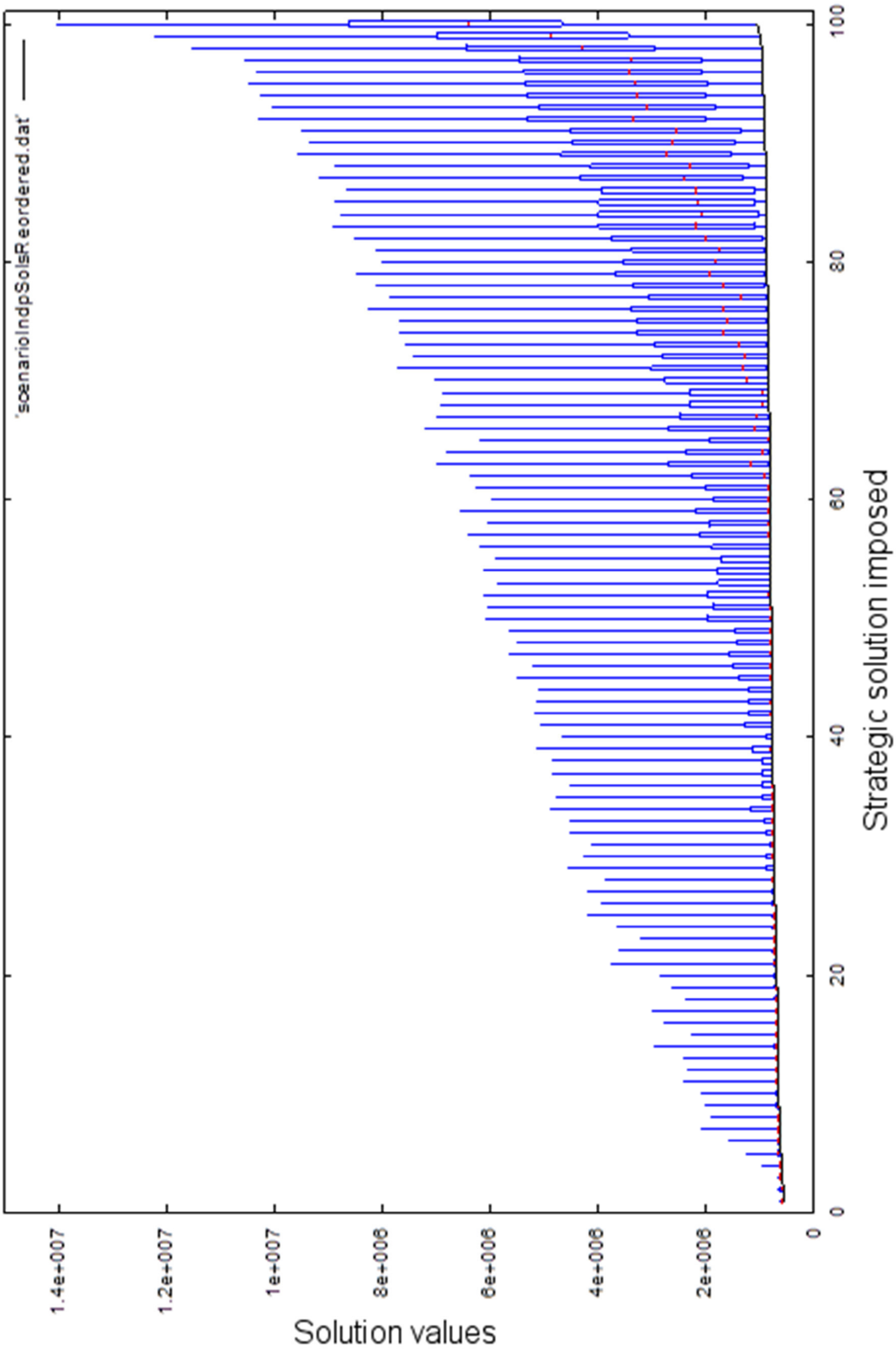

Figure 7-2 - Scenario Solutions when Full Strategic Solution imposed ordered by independent solution values

In all cases the imposition of the strategic solution from a scenario on the same scenario led to the same solution as when the scenario was solved independently, providing validation of the process.

If the strategic solutions had been obtained from optimal solutions then it would not be possible to find better solutions when imposing different strategic solution. However, in a few cases, better solutions were obtained when imposing a strategic solution from another scenario than when solving the scenario independently. For 69 of the 100 scenarios there was a least one case where imposing another strategic solution allowed a lower cost solution to be found, although the largest improvements were only 2.16%, 1.68%, 1.23% and 1.09%, with another 16 showing an improved solution that was better by between 0.5% and 1%, and 48 showing an improvement of under 0.5%.

Hence this imposition of the full strategic solution could be used to give a measure of the impact of choosing a particular strategic decision, and this will be considered later.

7.5.1.2 Effects on Scenarios of Imposing the Partial Strategic Solution

The following graphs shows range, inter-quartile range and the median of the solutions obtained for each scenario when the partial strategic solutions were imposed upon the scenario. The independent solutions are shown by the black line and these are not always the best cost that were obtainable. The median solutions are again shown in red, but it can be seen that the upper and lower quartile, and median values were all very similar with higher and lower outlying values.

Figure 7-3 show the solution values ordered by scenario number and

Figure 7-4 the same solution values order by the solution values of the scenarios when solved independently.

The highest objective values were obtained when solving scenario14, but this scenario had the highest demand and led to a much highest costs when solved independently too. The solutions were examined to see whether it was the strategic solution from low demand scenarios that were leading to the poorest solutions for scenario 14, but it was not this simplistic and the issues of imposing a low demand strategic solution on high demand scenario and visa versa will be considered in more detail later.

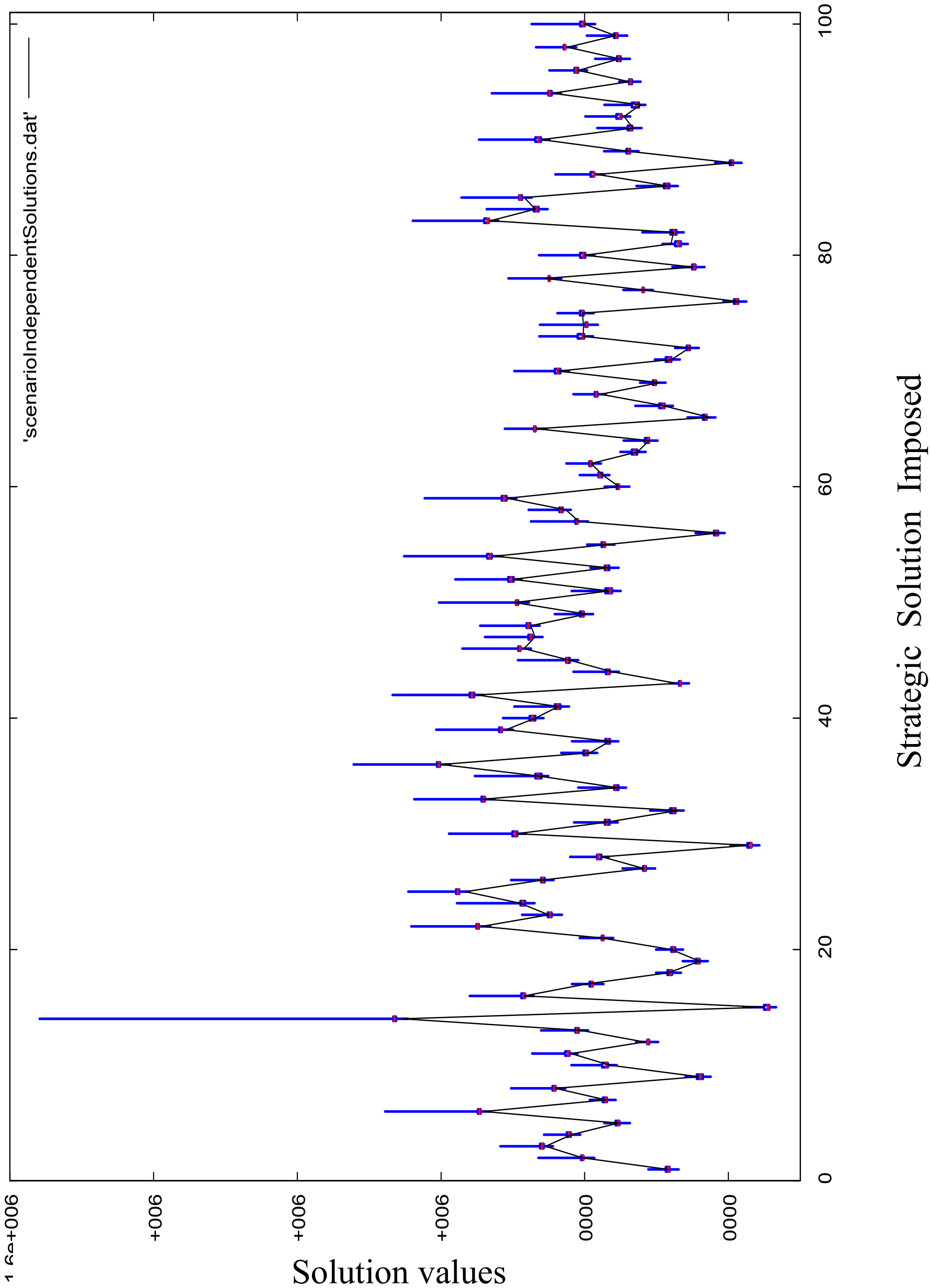

Figure 7-3 - Scenario Solutions when Partial Strategic Solution imposed, ordered by Scenario number

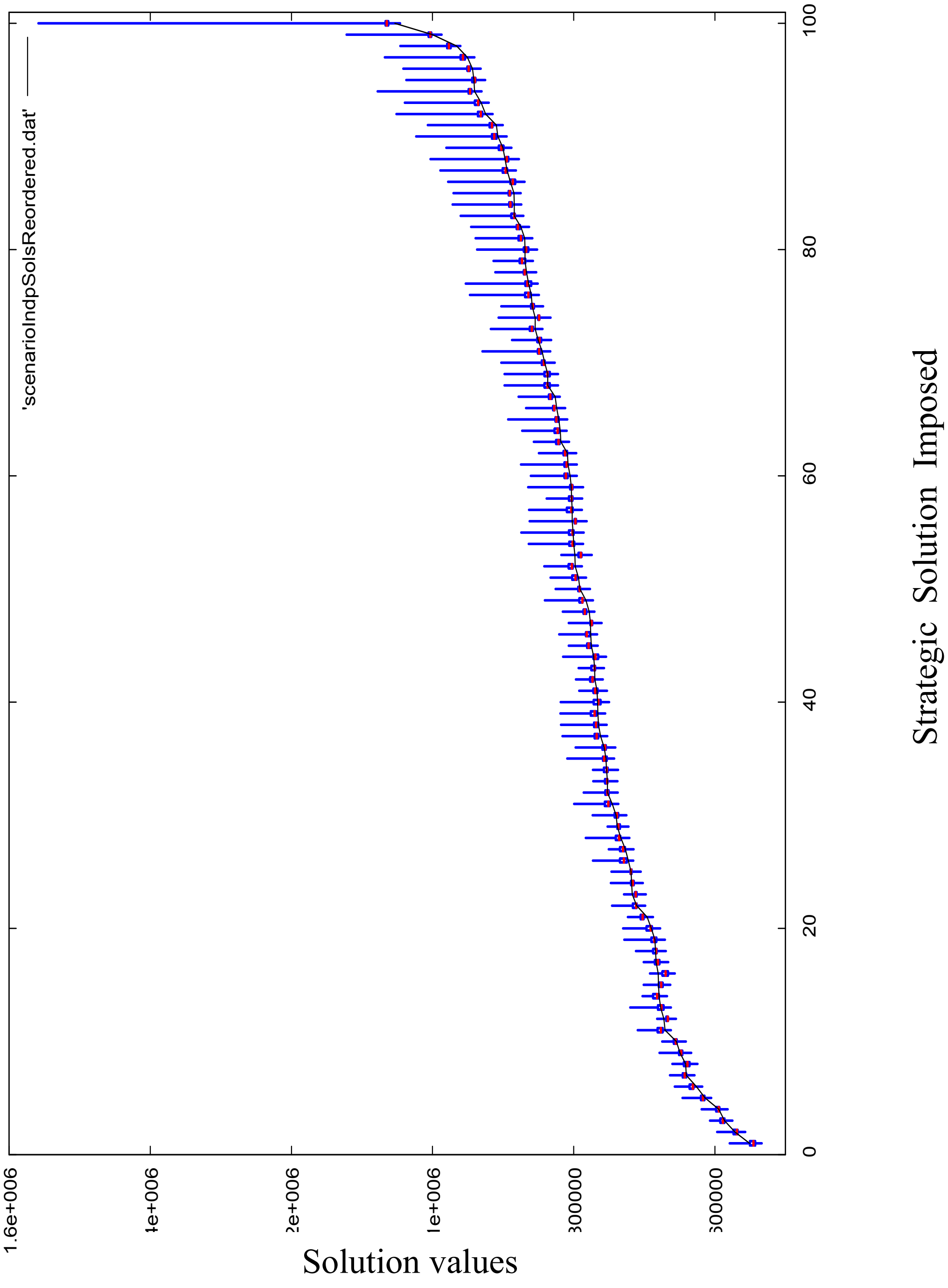

Figure 7-4 - Scenario Solutions when Partial Strategic Solution imposed, ordered by independent solution costs

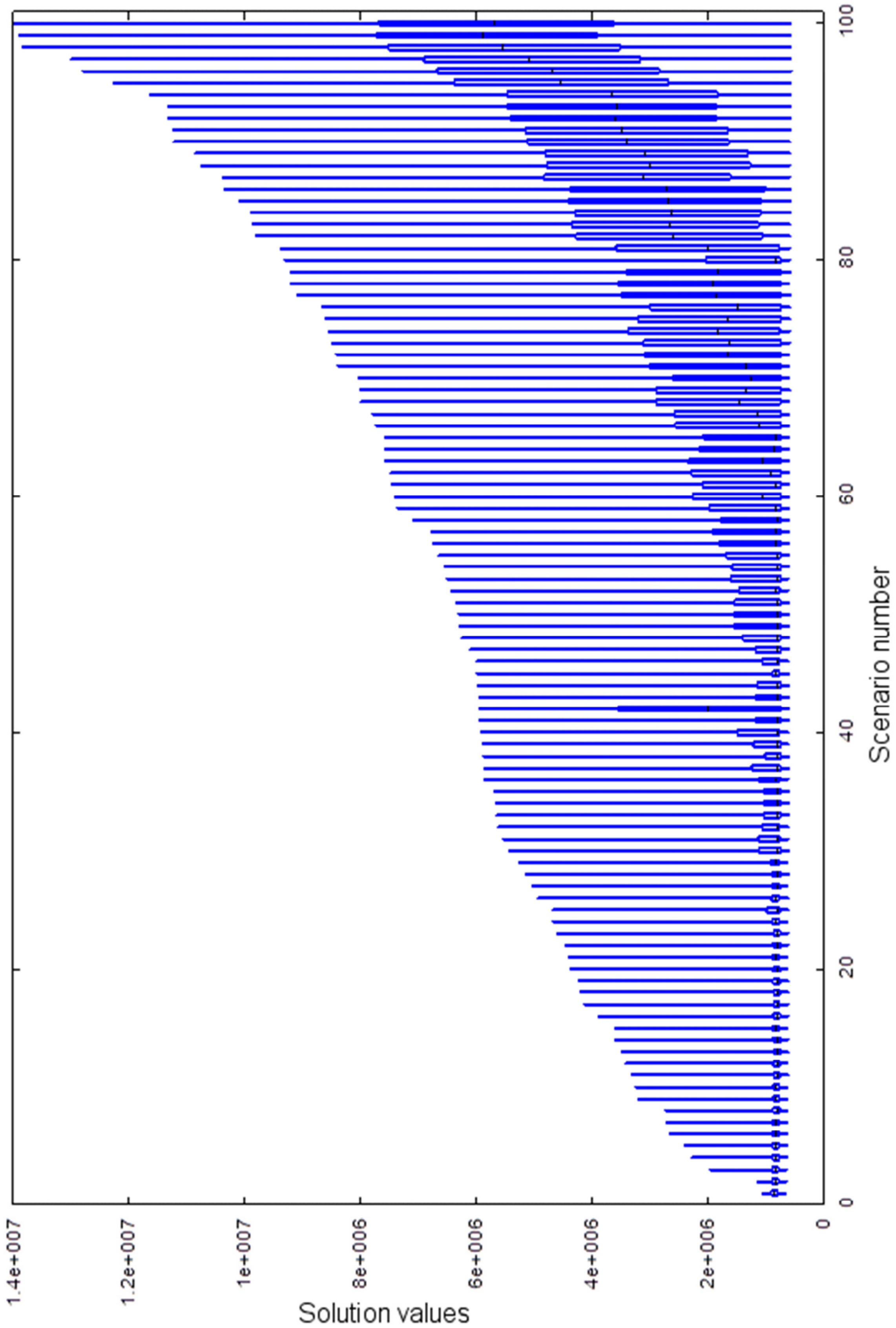

Figure 7-5 - Range of Solutions obtained for each Scenario with Full Strategic Solution imposed

7.5.1.3 The impact of different Strategic Solutions

The impact of each strategic solution on each scenario was considered.

The following graph (Figure 7-5) shows the range, inter-quartile range and median values for each strategic solution when used with each scenario ordered by the maximum objective value obtained for the strategic solution. These results allow consideration of the impact of selecting a particular strategic solution.

The two strategic solutions which lead to the smallest costs across the scenarios were the strategic solutions from the two highest demand scenarios 14 and 36. The strategic solutions from the two lowest demand scenarios 15 and 29 led to the highest costs. This result was consistent with what was expected as the problem data specifies high penalties for shortage. Further detailed analysis of these extreme cases is considered later.

7.5.1.4 Imposition of Partial Strategic Solution

The following graph (Figure 7-6) shows the range, inter-quartile range and median values for each strategic solution when used with each scenario ordered by the maximum objective value obtained for the strategic solution.

Imposing most of the partial strategic solutions led to a very similar distribution of solution values, with the imposition of scenario 32 (a fairly low demand scenario) leading to a better range of solutions than when the scenarios were solved independently. However, for nine of the Strategic solutions much higher maximum costs were seen for some scenarios, although there was not a large rise in either the median or the upper quartile costs.

7.5.2 Analysis of High and Low Demand Scenario Interaction

The results from imposing high demand strategic solution on low demand scenarios and vice versa were analysed further.

The objective values obtained when imposing the strategic solutions would be affected by excess capacity cost or shortfall for some of the products. It was expected that the shortfall costs would be magnified when low demand strategic solutions were imposed on high-demand scenario, and that excess capacity cost would be seen when high demands strategic solution were imposed on low demand scenarios, some of these cases were analysed in more detail.

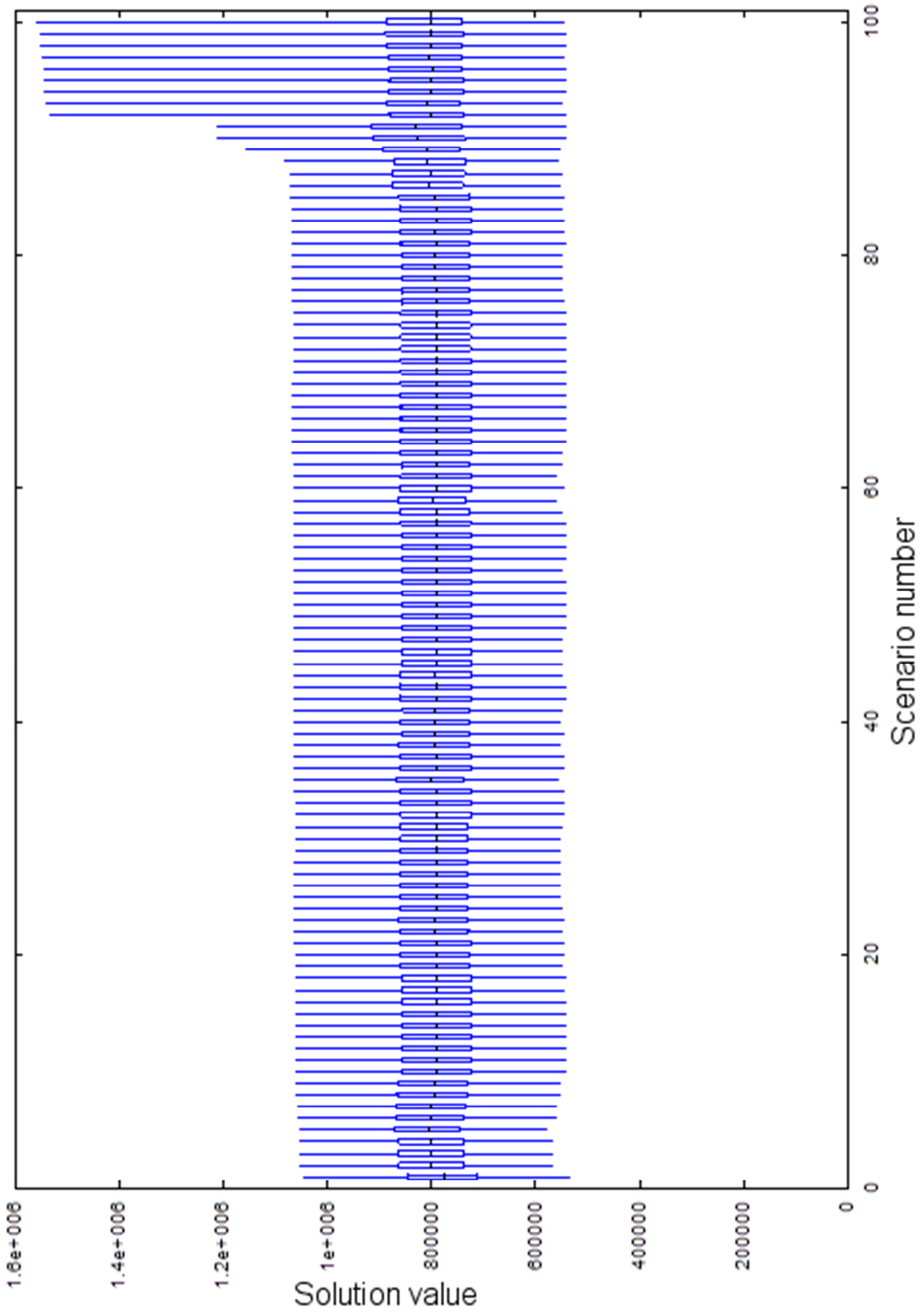

Figure 7-6 - Range of Solutions obtained for each Scenario with Partial Strategic Solution imposed

7.5.2.1 Identifying High and Low Demand Scenarios

The demand data for the 100 different demand scenarios was analysed. The demand profiles were found to all be similar, but giving a range of levels of demand. The following graph (Figure 7-1) shows the total demand for each of the products in each time period, for all of the customers. The final time period in the demand scenarios had ten year duration, so the demand illustrated below for that time period was the average annual demand over that ten year period.

Three high demand and three low demand scenarios were identified; the high demand ones were 14, 25, 36 and the low demand ones 15, 29 and 88.

7.5.2.2 Full Strategic Solution

When the scenarios that led to high cost solutions were examined it was found that that the worst solutions occurred when the three low demand strategic solutions were imposed on the highest demand scenario 14, and that their imposition on the high demand scenario 25 led to the second or third worst solution, however the same was not seen for scenario 36 which had a lower level of demand. When the strategic solution from scenario 14 was imposed on the other scenarios the first, second and fourth worst outcomes were seen with scenarios 15, 29 and 88.

Observation of how these increases were distributed across the different types of costs showed that in the case of imposing low demand strategic solution on the high demand scenarios led to very large shortage cost which was greater than the total cost increase, as reductions were seen in other costs, reducing the overall cost.

Conversely when high demand strategic solutions were imposed on the low demand scenarios the increased costs were distributed across the different types of costs and there was no shortage component in any of these cases, due to the high capacities from the high demand strategic solution being more than adequate for the production and distribution required. Increases of between 33.01% and 69.77% were seen for facilities costs and between 71.68% and 98.84% for the technology costs, due to the additional capacity enforced in the solution. However, the additional capacity allows lower cost production and lower cost transport routes to be utilised, so all the production costs and all but one of the transport costs were reduced. Reductions of between 9% and 30.42% were seen for production costs and between 8.74% and 42.42% for transport costs, except in one instance when an increase of 7.37% was observed.

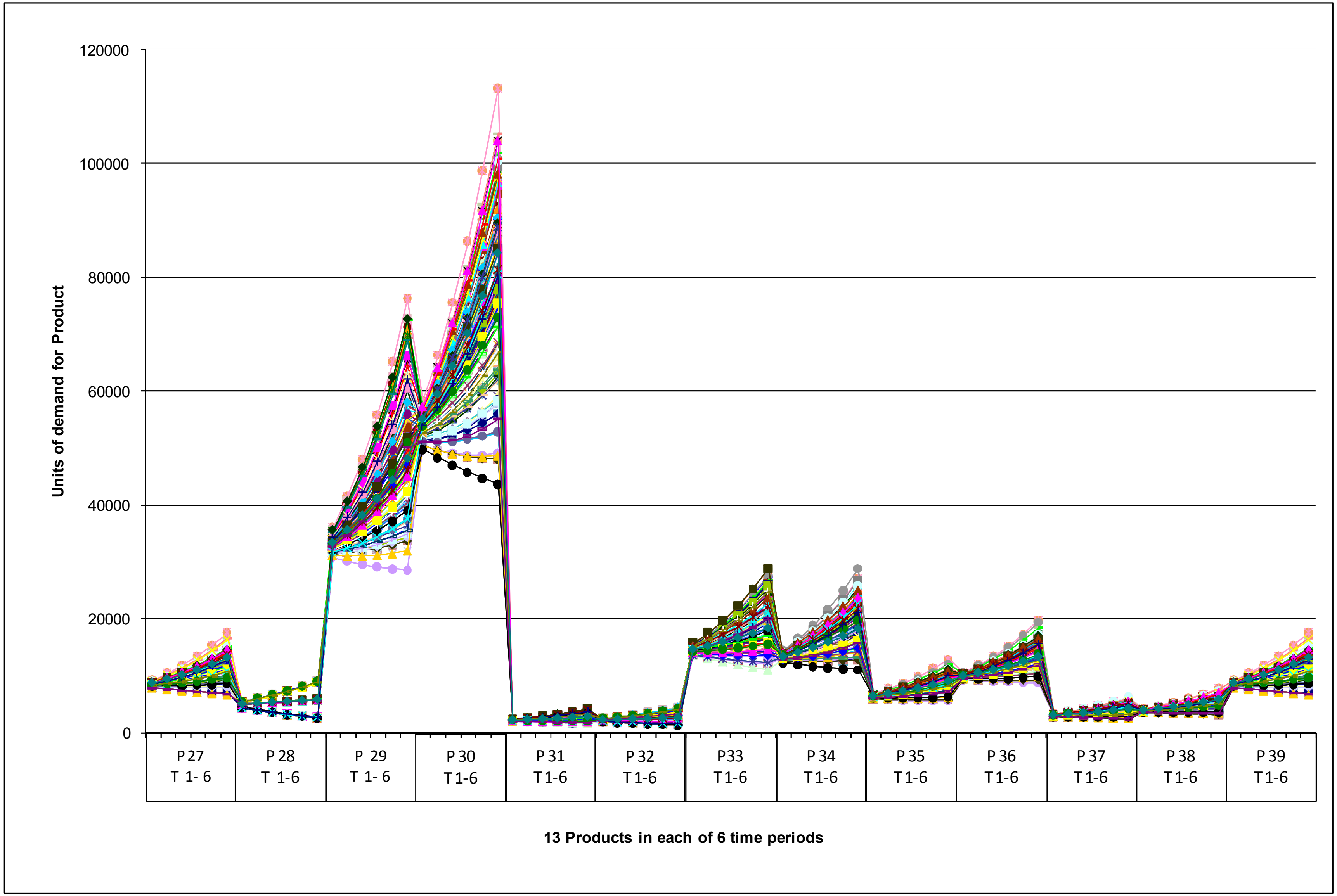

Figure 7-7 - Demand Quantities by products and time periods.

It could be seen from these two sets of comparisons that much more detrimental effects of imposing the low demand strategic solutions on the high demand scenarios were many due to the severe shortage costs and give a clear indication of which would be the better strategic policy to adopt.

Further details of all of these points can be found in Appendix 3.

7.5.2.3 Partial Strategic Solution

In most cases most the cost increases were from shortage costs in the same way as seen when the full strategic solution was imposed. However, when imposing scenario 88 on two of the high demand scenarios no shortage was seen, as additional capacity was available by adding additional technology lines at the facilities which had been included.

The contributions of other components of the increased cost for transport, production and technologies all varied from very small increases or decreases to 101.6%, 63.6% and 47.9% respectively. The facility costs were reduced in all cases, providing reductions that were 1.3% and 61.0% the magnitude of the increase seen in costs.

In the cases where high demand strategic solutions for facilities only were imposed on the low demand scenarios only very small increases in costs were seen in comparison to solving the scenario independently, the greatest of which was only 4.1%.

Further details can be found in Appendix 3.

7.6 Analysis of Facilities used

As the imposition of partial strategic solution which fixes just the facility decisions, had led to the ability to find better solutions further investigation were made of the facilities that were used in each case.

When the solutions of each of the scenario solved independently were analysed it became apparent that there were similarities in the facilities selected to use. It was seen that facilities 1, 6, 7, 14, 19, 20, 21, 22 and 23 were not opened in any time period in either the low or high demand scenarios. In comparison facilities 3, 5, 12 and 13 were used in all six Time periods in both the high and low demand scenarios. This left all of the variation in the facility strategic solutions in just 10 of the 23 facilities that were available: facilities 2, 4, 8, 9, 10, 11, 15, 16, 17 and 18.

Table 7-3shows the variation in the facility solutions between the three high demand scenarios 14, 25 and 36; and the three low demand scenarios 15, 29 and 88.

	Scenario					
Facility	**14**	**25**	**36**	**15**	**29**	**88**
2	1-6	0	1-6	1-2	1-2	1-3
4	1-6	1-6	1-6	1	0	1
8	1-6	1-6	1-6	1	5	1-6
9	0	0	0	1-2	0	0
10	1-6	0	1-6	0	1-2	1-3
11	1-6	1-6	1-6	0	0	0
15	0	0	0	1	1	0
16	1-6	1-6	1-6	1	0	1-6
17	1	0	0	1	0	0
18	0	1-6	1	0	1	1
Total used	7	5	7	7	5	6

Table 7-3 - Facilities used in each Scenario

Including the 4 facilities that were included in all these solutions, the variation in the number of facilities used was a minimum of 9 and a maximum of 11. Reducing the problem by forcing those facilities that are always used to be open and removing those that are never used makes the problem easier to solve as was demonstrated in (**Error! Reference source not found.**) and provides a way to simplify the problem across a group of scenarios.

7.6.1 Imposing Facility and Shortage decisions on multiple Scenarios

All the facilities that were seen to be unused in all the scenarios of this problem instance were forced to be closed in all time periods and those facilities that were seen to be used in all the time period were forced to be open in all time periods.

It had also been observed that shortage was zero in the best integer solutions found for 98 of the 100 scenarios. In the two cases where shortage was not zero it was small and feasible solutions near to the best integer solutions with shortage could be obtained for both cases; the cost savings that were being made seemed to be mainly due to additional line costs that were incurred. The impact of forcing the shortage to zero, as a strategy on its own, and in combination with forcing the facility decisions was investigated on a small set of scenarios that did not include the two which found best solutions that included shortage.

When Shortage was forced to zero for a single scenario a small improvement was seen in the later solutions found (**Error! Reference source not found.**). Table 7-4 gives the number of rows and columns after pre-solve showing reduction in problem sizes that were seen.

	Original MIP		Forcing Facilities Open/Closed		Forcing Shortage to Zero		Forcing Facilities & Shortage	
Number of Scenarios	Rows	Columns	Rows	Columns	Rows	Columns	Rows	Columns
2	12859	79642	9724	51981	12859	76882	9724	49221
3	18465	117359	14149	79715	18465	113220	14149	72575
4	24081	155076	18522	101419	24081	149556	18522	95899
5	29691	192787	22900	126103	29691	185887	22900	119203

Table 7-4 - Problem size with different numbers of Scenario

It can be seen that forcing the facilities reduced the number of variables in the model by approximately 25%, whilst forcing the shortage has no impact on the number of variables. The number of constraints in the model after pre-solve is reduced by both methods, with forcing facilities leading to a reduction of about a third, whilst forcing shortage lead to reductions of less than 5%, using both did lead to a larger combined effect that was approaching a 40% reduction. The root relaxation times were also compared and the results in Table 7-5 showing large reductions when the facilities are forced open and closed and small reduction when the shortage is forced to zero.

	Root Relaxation Times			
Number of Scenarios	**Original MIP**	**Forcing Facilities Open/Closed**	**Forcing Shortage to Zero**	**Forcing Facilities & Shortage**
2	189	77	164	72
3	831	301	720	280
4	1527	680	1501	619
5	2961	1006	2563	1048

Table 7-5 - Root relaxation times when forcing Facility usage and shortage

Solutions found using these methods were then compared. It can be seen that forcing the facilities has allowed better solutions to be found within the time limit, but that forcing shortage to zero did not improve solutions found and actually made it more difficult for the solver to find an integer solution. However, combined with forcing facilities it can be seen than better solutions were found when solving two and three scenarios. With Five scenarios the model with the shortage forced to zero still failed to find a solution within 50000s, although both formulations with the facility usage fixed did find good solutions given this additional time.

	Best Solutions within a 10000s time limit			
Number of Scenarios	**Original MIP**	**Forcing Facilities Open/Closed**	**Forcing Shortage to Zero**	**Forcing Facilities & Shortage**
2	1527268	1501830	1544930	1496260
3	2569870	2400690	No solution	2361350
4	8835621	6919100	No solution	No solution
5	No solution	14561500	No solution	No solution

Table 7-6 - Solution when forcing Facility usage and shortage

Significant benefit was seen from forcing the facility decisions, this reduced the size of the model, and improvements were seen in solve times, integer solutions found and the size of the gap. However, when shortage was forced to zero, this seemed to make finding solutions more difficult, as many of the poorer solutions seen early in the MIP search process have significant levels of shortage.

7.6.2 Conclusion

The generic model has been successfully extended to use it to solve multiple scenarios models, so that a single strategic solution can be found which is provides a best fit for all the included scenarios.

8 Neighbourhood Search

8.1 Background

Neighbourhood search is a technique for obtaining good quality solutions to large optimisation problems, providing an effective method of finding solutions when the problem is so large that solving it directly is computationally very expensive or not achievable.

The starting assumption is that we have a problem and a feasible solution (current solution), which will usually be sub-optimal for that problem. The approach is to repeatedly explore neighbourhoods of the current solution by solving the sub-problem made up of a subset of the variables from the original problem whilst other variables are set to values from a known feasible solution.

There are considerable variations in the way this method is applied and these come from the possible variations in how the sub-problem to solve on each cycle is chosen, the method used to solve the sub-problem, and how the results of the sub-problem solution are integrated back into the solution for the overall problem. This framework is very general and can encompass a wide range of apparently unrelated approaches and algorithms.

A simple neighbourhood search algorithm was devised and implemented as part of the generic framework and its performance was compared with that achieved with a standard commercial MIP solver for a number of problem instances.

The work reported here was presented at 9th Metaheuristics International Conference (Chippington Derrick 2011a) and submitted as a paper to JORS (Chippington Derrick 2011b).

8.2 Method Description

The local search method employed here first determines a feasible solution to the problem, then selects a subset of the variables to be fixed at values from the feasible solution, resulting in a smaller sub-problem which is then solved to look for improved solutions. Any solution to the sub-problem will also satisfy the parent problem. The new values for the variables then provide an updated set of values for the whole problem. This process is then be repeated in order to find improving solutions to the parent problem.

8.2.1 Mathematical Formulation

Consider a MIP minimisation problem which can be defined as follows:

$$\text{Min} \quad \sum_j c_j x_j$$

$$\text{Subject to} \quad \sum_j a_{ij} x_j = b_i \qquad \forall i$$

The set J can be partitioned into two disjoint sets J' and J'' where $J = J' \cup J''$ and $J' \cap J'' = \phi$.

Given a feasible solution to J, $x_j = x_j^*$ where $j \in J$; then this can be decomposed into $x_{j'} = x_{j'}^*$ where $j' \in J'$ and $x_{j''} = x_{j''}^*$ where $j'' \in J''$

A sub-problem is constructed by simplifying the main problem by fixing a set of solution values J' in J. Thus:

$$\text{Min} \quad \sum_j c_j x_j$$

$$\text{Subject to} \quad \sum_j a_{ij} x_j = b_i \qquad \forall i$$

$$x_{j'} = x_{j'}^* \qquad \forall j'$$

This sub-problem can be more efficiently written as:

$$\text{Min} \quad \sum_{j''} c_{j''} x_{j''} + \sum_{j'} c_{j'} x_{j'}^*$$

$$\text{Subject to} \quad \sum_{j''} a_{ij} x_j = b_i - \sum_{j'} a_{ij} x_j^* \qquad \forall i$$

where $I' \subseteq I$ is the subset of constraints involving decision variables $x_{j''}$, $j'' \in J''$.

8.2.2 Search Procedure

The following flow chart (**Error! Reference source not found.**) shows the procedure that was used in applying Local Search using a MIP model instance within the generic framework which has been developed.

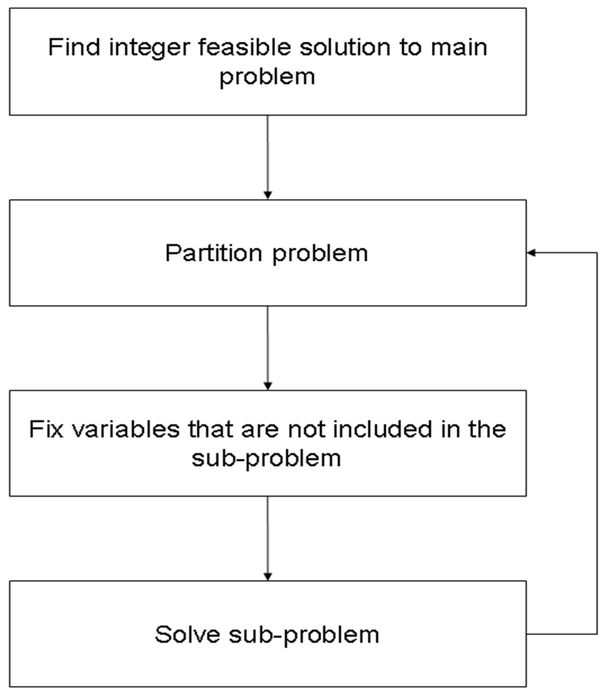

Figure 8-1 - Flow Diagram for Neighbourbood Search Procedure

The initial step is to determine an integer feasible solution to the main problem. This is then followed by three steps iterations of the neighbourhood search. The problem is partitioned to produce a smaller neighbourhood to search; the sub-problem is created by fixing all the values of the variables outside that are not in the neighbourhood and then the sub-problem is solved to provide another integer feasible solution. Hopefully a better objective value is obtained and the values of the variables from this new solution update the incumbent solution to be use in the next iteration.

There are decisions to be made about how each of these steps will be implemented. The decision about how to obtain the initial integer feasible solution, about how the problem is partitioned at each iteration, how the sub-problem is solved at each iteration and how much effort is expended on solving each sub-problem before moving on to solving a new sub-problem are detailed in the following section.

8.2.2.1 Choosing the sub-problem

How the problem will be partitioned and the sub-problem constructed at each iteration need to be considered. Below is a list of possible alternatives.

We can make use of our knowledge about the problem structure to help us define our sub-problems in ways that are likely to make them easier to solve and/or more likely to include the potential for improving solutions.

We can make use of relaxations of the overall problem to guide the choice of sub-problem.

We can make use of heuristics to guide the choice of sub-problem

We can control the size of the sub-problem. Smaller sub-problems can often be solved to optimality, but may be too limited in their scope to allow any room for finding improving solutions. Larger sub-problems may be too computational intensive to solve to optimality in reasonable time, but may offer more opportunities for finding improving solutions.

8.2.2.2 Solving the sub-problems

We used a MILP solver to solve the sub-problem, although it would also be possible to employ another method, such as Heuristics, Local search or constraint programming.

One or more stopping criteria must be selected:

- A measure of the quality of the solution
- A number of solution steps
- A time limit

8.2.2.3 Summary

We have a structure for Neighbourhood Search that can be used with the Generic Supply Chain framework. The MIP solution method that has been defined in chapter 4 can be used to find the initial solution and to solve the sub-problems; the following section gives further details of this implementation.

8.3 Integrating Neighbourhood Search into our Generic Supply Chain Model

The local search method was implemented in the C++ framework that had been developed for solving the MIP with CPLEX. This allowed different initial solutions to be used, either by choosing the values of the integer variables by hand or by finding an integer solution by solving the MIP, either a poor solution that may be found relatively quickly or a solution that could be found quickly by adding additional constraints to the problem.

8.3.1 Choice of Initial Solution

Integer-feasible solutions are often found quickly and these can be used as an initial solution for this process. Alternatively it is possible to find solutions that would not be good solutions to the problem, but would be known to be feasible such as keeping all the existing Facilities and Technology Lines open and all the potential Facilities and Technology Lines closed. These two methods of finding an initial solution for the process were both explored.

8.3.2 Choice of Sub-problem

Finding values for the continuous variables was very fast and hence all these variables could be resolved for each sub-problem. Finding the feasible integer values is the time consuming part of the solution process, so how many and which of these variables were included in the sub-problem was an area for exploration. The more variables that are included in the sub-problem the greater scope for finding a better solution, however, a smaller sub-problem whilst having less potential for improving the solution will be quicker to solve.

There are two groups of integer variables, those indicating which facilities are open in which time periods, and those indicating which technologies are open at each facility in each time period.

The decisions between opening and closing the facilities and the technologies are closely linked; we consider several methods of selection of facilities along with their technologies to be included in the sub-problems.

8.3.3 Solving the sub-problems

CPLEX was used to find initial solutions and also to solve the sub-problems. Constraints were added to the master problem setting the values of the integer variables which were not included in the sub-problem, and all other integer variables and the continuous variables were not fixed. However, CPLEX also offers the option of providing an initial solution and this was done at each cycle using the variable values from the current solution to warm start the search process for the sub-problem.

Investigations were also made of a variety of stopping criteria used for the solution of each sub-problem:

- Number of nodes explored
- Time limit
- Optimality Gap

8.4 Results of the use of Neighbourhood Search

8.4.1 Comparison of Neighbourhood Search with MIP

The results obtained using neighbourhood search were compared with the results obtained using the original MIP formulation. These results were compared for the main problem when solving single and multiple scenarios.

Even for some of the single scenario cases the neighbourhood search performed better than the MIP, particularly in finding early good solutions; but once several scenario were included in the problem the neighbourhood search consistently performed better.

The choice of the sub-problem, the stopping criteria and the initial solution were important to being able to obtain better solutions when using the Neighbourhood Search method; choosing these parameters can be considered as tuning the search and in the rest of this chapter we report on the experiments carried out on tuning the search.

8.4.2 Tuning Neighbourhood Search

Experiments were carried out to compare how changing the selection of the sub-problem, the stopping criteria and the initial solution affected the results obtained, and to demonstrate the importance of considering the impact of these choices. The different criteria that were included in these experiments were:

Initial Solution Selection

- Good Integer Solution
- Feasible Integer Solution

Sub-problem selection criteria

- Problem segments for example a selection of Facilities
- Random selection of percentage of the integer variables

Stopping criteria for each Iteration

- Number of Nodes explored
- Time limit
- Optimality Gap

Comparisons of these were made by considering the solutions obtained by number of iterations carried out and elapsed corrected CPU time. Corrected CPU time allows for the removal of the build time from the models in the comparison. This is because the build time is highly dependent on the implementation of the program, which is a software engineering issue and was not the focus of this study. However, it did allow for the inclusion of the iteration set ups. Although these were negligible in most cases, in some cases where there were many iterations with no improvements in the solutions then significant differences were seen between the CPU times and elapsed times. Hence as the build times for iterations were mainly involved with selecting and setting variable values, and unlikely to be able to be reduced by different implementations it was felt that this should be retained in the corrected CPU times that were compared

8.4.2.1 Initial Solution Selection

There are two issues which need to be considered when obtaining an initial solution; how long it takes to find the initial solution and whether the starting point has an impact on the solutions found in the subsequent neighbourhood search.

The initial solutions considered were

- first integer solution obtained when solving the MIP
- all the facilities and technologies closed in all time periods
- all the facilities and technologies that were already in use left open in all time periods and all potential facilities and technologies closed for all time periods.

The times taken to find these three initial solutions were not significantly different in comparison to the times that are required to find good integer solutions. Times to find initial solutions along with the root relaxation times, in brackets, are given in the table below.

	Corrected CPU time (s)	
Initial Solution	Scenario 1	Scenario 2
First Integer Solution	78.46 (40.87)	76.27 (38.66)
Solution with all facilities forced closed	0.34 (0.01)	0.33 (0.0)
Solution with facilities and technologies	16.43 (13.33)	13.08 (7.66)

Table 8-1 - Corrected CPU Times

As shortage has a significant impact on the objective, the solutions for both scenarios with the facilities closed have large costs. The impact of shortage on the cost also accounts for why the costs seen for scenario 1 are significantly lower than those seen for scenario 2 when the solution with the current facilities and technologies is imposed; this is because most of the demand can still be met for scenario 1 with this limit on resources, but scenario 2 has a higher demand which requires more resources.

	Initial Solution Objective Values	
Initial Solution	Scenario 1	Scenario 2
First Integer Solution	3,423,661	3,635,649
Solution with all facilities forced closed	22,698,800	27,812,400
Solution with facilities and technologies retaining current status	837,439	3,616,479

Table 8-2 - Initial Solution Objective Values

The difference in the time taken to find these initial solutions was small and hence had little impact on the time taken to find solutions using the Neighbourhood Search method. For Scenario 1 better results than were obtained with the MIP when using the first integer solution, but this gave worse results with Scenario 2. Although the quality of the different initial solutions were very different, the impact of these on the quality of the subsequent solutions found was small. When the Neighbourhood Search was given a poor initial solution, rapid improvement in the early iterations was seen. For detailed results see appendix 2.

8.4.2.2 Sub-problem Selection

8.4.2.2.1 Random Selection of Percentage of Variables

Comparisons were made when the sub-problems for the Neighbourhood Search was created by selecting different percentages of the integer variables to be included in the sub-problem.

A simple process was used to randomly select variables for each sub-problem, with each variable being included or excluded on the basis of whether a random number selected fell into the specified range. The pseudo random numbers generator gave values between 0 and

1; so when aiming to select 10% of the variables, if the random number was less than 0.1 the variable was included in the sub-problem and if it was greater or equal to 0.1 then it value was fixed and it did not become a variable in the sub-problem. This method provided a simple method of iterating through the variables and fixing values or provide starting values for the sub-problem model that were being solved by CPLEX at each iteration.

Although this did not allow the selection of precise percentages of variables at each iteration, it did allow selection of percentages of variables that were near to the aim value. In addition to this it provided simplicity of implementation and repeatability, but with the option of running the pseudo random number generation with a different seed to provide a different selection of the same approximate percentages of variables.

The inclusion of various percentages of variables in the sub-problem was investigated; results for sub-problems containing 5%, 10%, 15%, 20%, 25%, 30%, 40% and 50% of the integer variables are reported.

8.4.2.2.1.1 Quality of Integer Solutions for different sub-problem sizes

Considerable differences were seen in the solutions obtained with the different sized sub-problems. The smaller sub-problems were fast to solve, but the neighbourhood search algorithm was then slow to achieve good integer solutions as can be seen in the graph below of the solution obtained for the first problem instance.

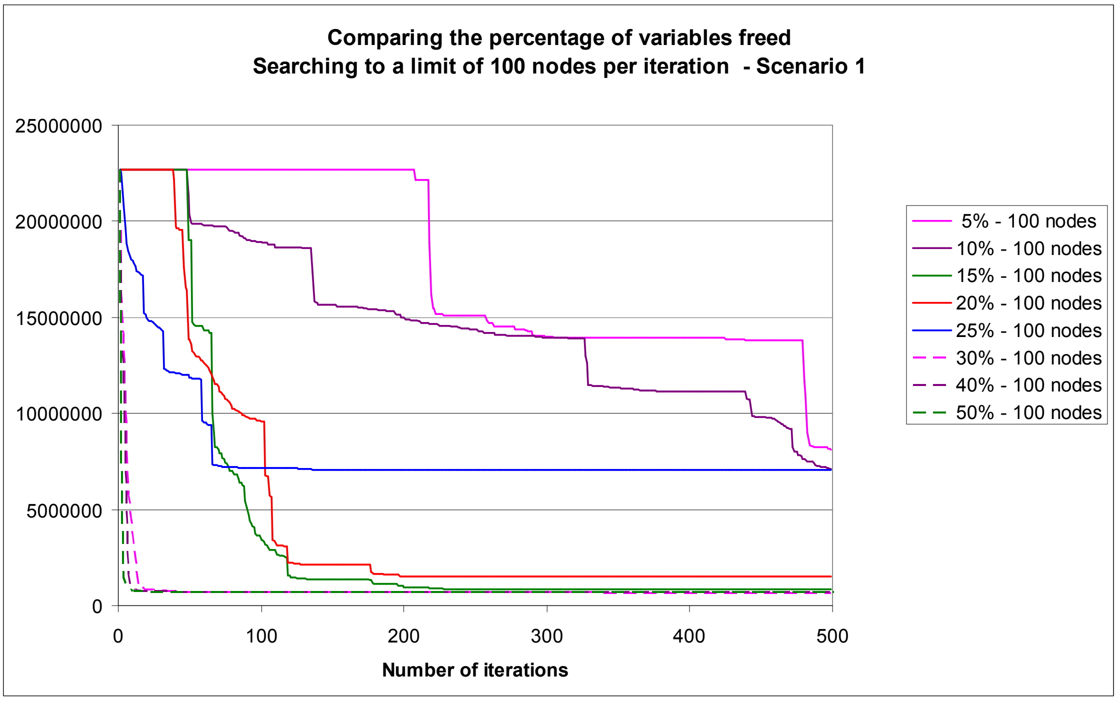

Figure 8-2 - Integer Solution for Different Sized Sub-Problems

This illustrates that a suitably large sub-problem needed to be selected in order for this method to be effective, but if the sub-problem became too large then the speed of the convergence, and the quality of the solutions could become less good.

8.4.2.2.1.2 Implications of the small sub-problem

Further evidence for the importance of using a suitably large sub-problem was seen in the results obtained. It was seen that there were iterations where no improvement was made, which occurred more frequently with smaller sub-problems. It was seen that in some cases no improvement was possible for the sub-problem which had been constructed, as the starting solution proved to be optimal solution for the sub-problem. In these cases CPLEX pre-solve was sufficient to find the optimum solution, and although the solution of the sub-problem was very rapid, at each iteration time was expended setting up the sub-problem. The following graph shows the number of occurrences of sub-problems where the starting solution was proven optimal and provided no scope for improvement to be made to the main problem. It can be seen that these occurrences decreased as the sub-problem size increased.

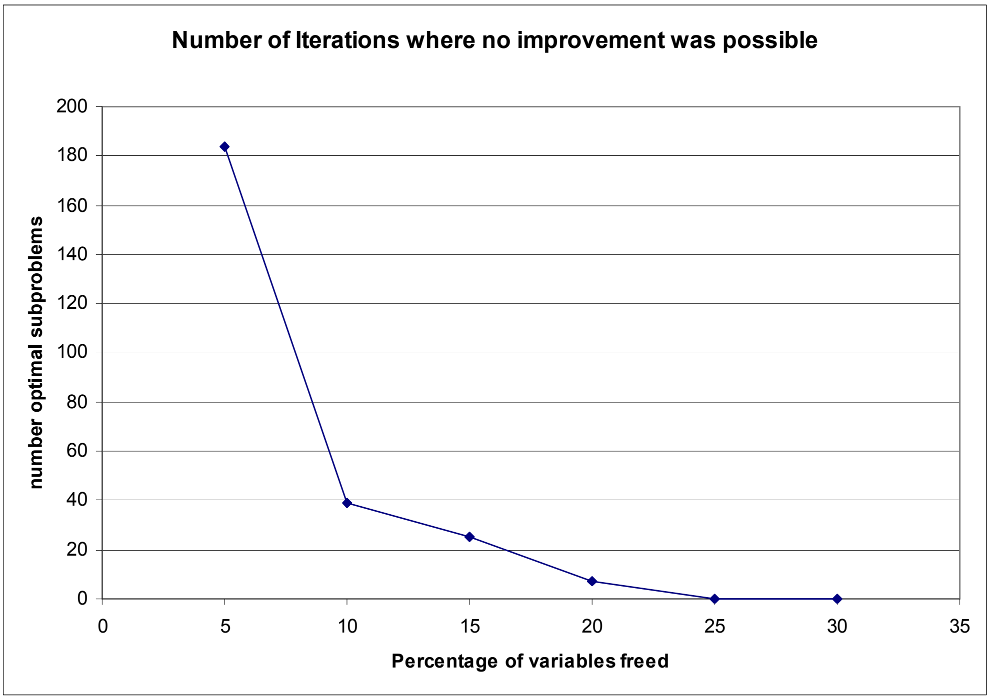

Figure 8-3 - Iterations with no improvement seen

Most of these occurrences occurred early in the sequence of iterations and before there was any movement from the initial solution. The following table shows the number of iterations in which no improvement was possible, and the number of iterations before a change from the initial solution was achieved.

Percentage of Variables freed to create sub-problems	Number of iterations with optimal sub-problem starting solution	Number of iterations before sub-problems starting solutions all non optimal	Percentage of early iterations with optimal sub-problem starting solution
5%	184	207	89%
10%	39	48	81%
15%	25	48	52%
20%	7	39	18%

Table 8-3 - Sub-Problem Details

It was also observed that the as the sub-problem size increased so did the size of the optimality gaps at the beginning of the search for each iteration. The following table shows the maximum and the average gap that were seen for the first 50 iterations for each of the neighbourhoods. This gives a measure of the scope for finding an improved solution to the main problem.

Percentage of Variables freed to create sub-problems	Maximum Optimality gap at the beginning of a sub-problem search.	
	Scenario 1	Scenario 2
5%	14.87	15.14
10%	10.67	26.83
15%	37.59	40.70
20%	47.72	39.25
25%	22.81	38.51
30%	50.61	51.74
40%	71.98	66.02

Table 8-4 - Optimality Gap for Different Sub-Problems

Percentage of Variables freed to create sub-problems	Average of the Optimality gap at the beginning of the first 50 iterations.	
	Scenario 1	Scenario 2
5%	1.33	1.64
10%	0.53	1.36
15%	2.46	1.95
20%	2.01	2.19
25%	1.73	2.72
30%	3.35	2.64
40%	3.99	3.76

Table 8-5 - Average Optimality Gap

Further analysis of the larger sub-problems led to further information about the upper limits that needed to be placed on sub-problem size for the effective use of this method.

The early solutions obtained when 30%, 40% or 50% of the integer variables included in the sub-problem were similar, but comparing the later solutions indicated that around 30% we had reached a limit on improvements seen with increasing sizes of sub-problem. The graph below shows the better solutions that are found after the same elapsed time with the smaller 30% sub-problems in comparison to the larger 40% sub-problems.

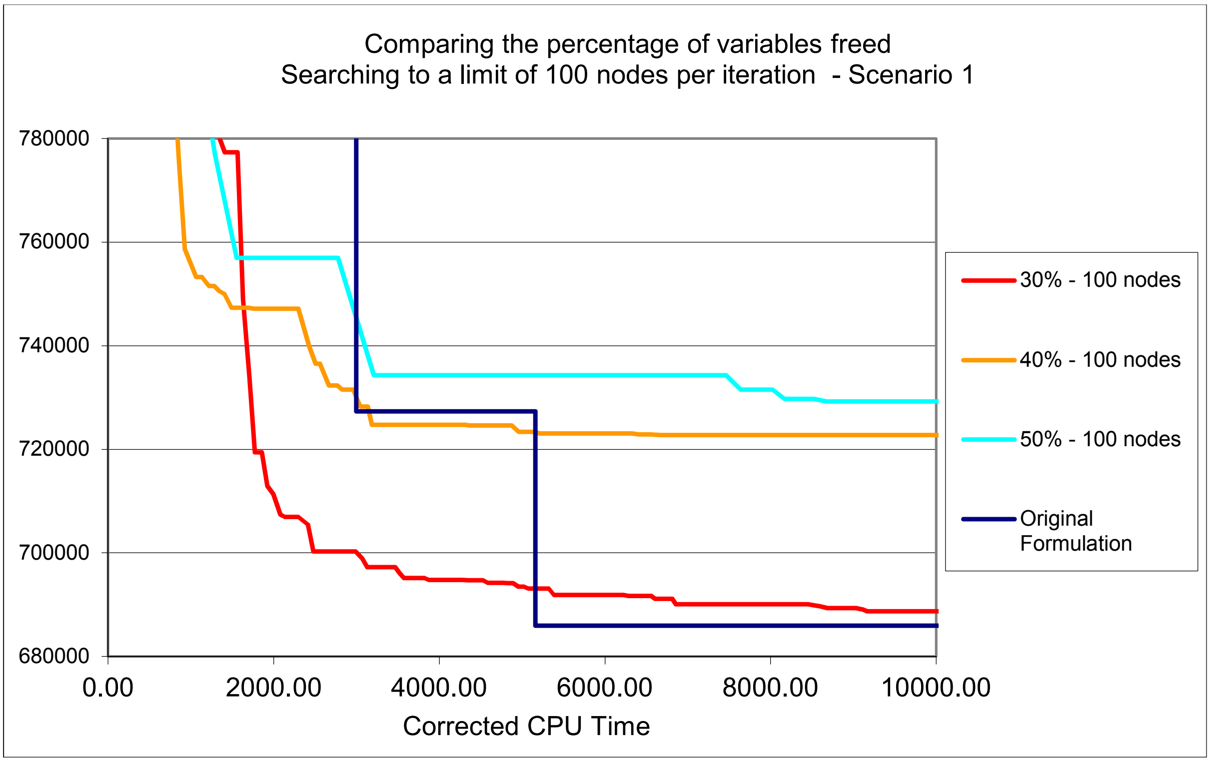

Figure 8-4 - Comparison of Solutions under 780000 by corrected CPU Time

Although some of this difference was due to the longer solution times taken with the larger sub-problems this was not the only reason for 40% sub-problems not performing as well, as was apparent when a comparison was made of the solutions obtained after a given number of iterations. It can be seen in the graph below that although there is less difference between the two cases the 30% sub-problems still perform better at finding good solutions to the overall problem.

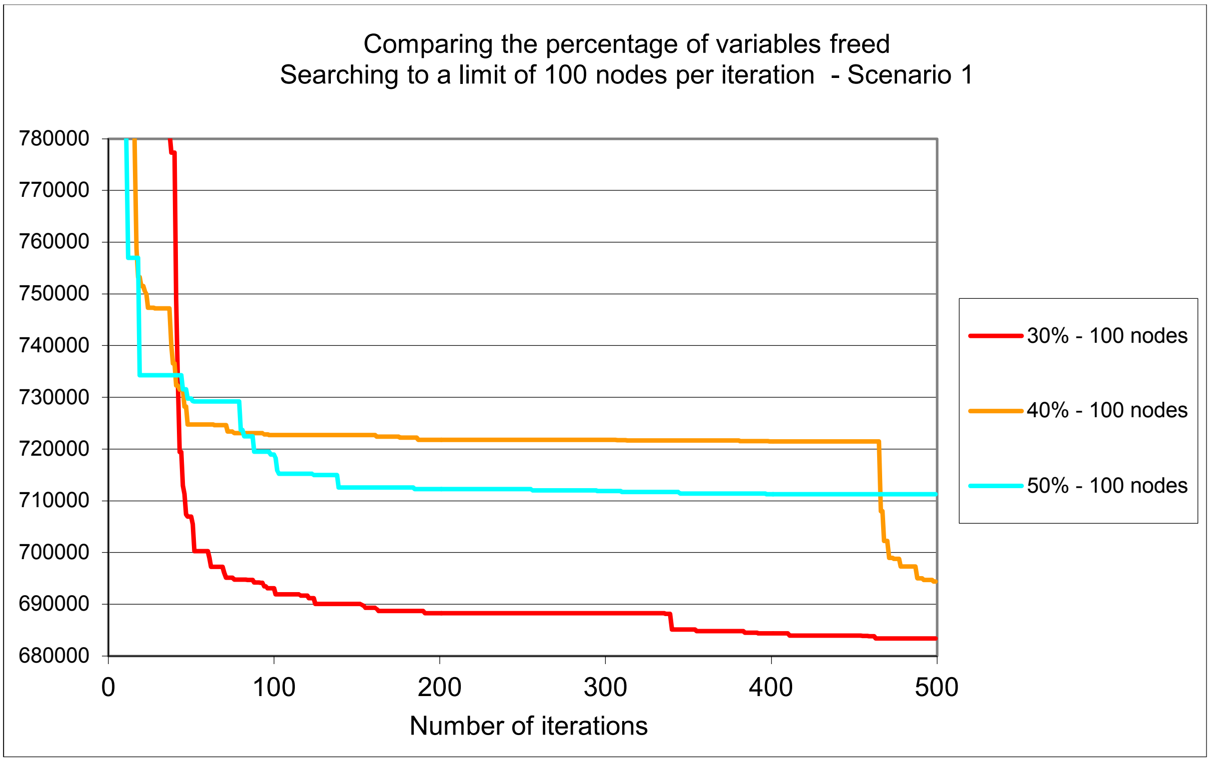

Figure 8-5 - Comparison of Solutions by Number of Iterations

This stagnation of the solution improvement is thought to be due to the solution becoming too tight in some neighbourhoods to allow improvements in other neighbourhoods later in the process.

8.4.2.2.2 Selection of a Set of Facility Variables

As an alternative to a purely random selection of integer variables to be freed for each successive sub-problem, we also investigated selecting the variables belonging to a set of facilities.

Again a pseudo-random number was to select a facility until the required number of facilities had been selected. All the facility and technology line integer variables associated with the selected facilities were then included as variables in the sub-problem, with the remaining variables becoming the fixed part of the problem.

8.4.2.2.2.1 Quality of Integer Solutions for different sub-problem sizes

Using a single facility was not expected to be a good technique, as it would not allow any switching of processing from one facility to another; however, the ineffectiveness was striking with no improvements seen in any of the iterations.

Differences were seen in the quality of the solutions obtained when different numbers of facilities were included in the sub-problem selection process, although improved solutions were seen with the local search method when compared with the MIP, these improvements were seen across the range of large and small sub-problems, although the small sub-problem comprising of 4 or less facilities seemed to perform less well.

The following graph shows the solutions obtained with different numbers of facilities used to select the sub-problem for scenario 1. It can be seen that the local search gave good results early in the search for all the sub-problem sizes, but for the smaller sub-problem less good final solutions were achieved. The solution achieved for scenario 2 were better at all times for all the sub-problem sizes, although using more than 4 sets of facility variables to construct the sub-problem again led to better solutions.

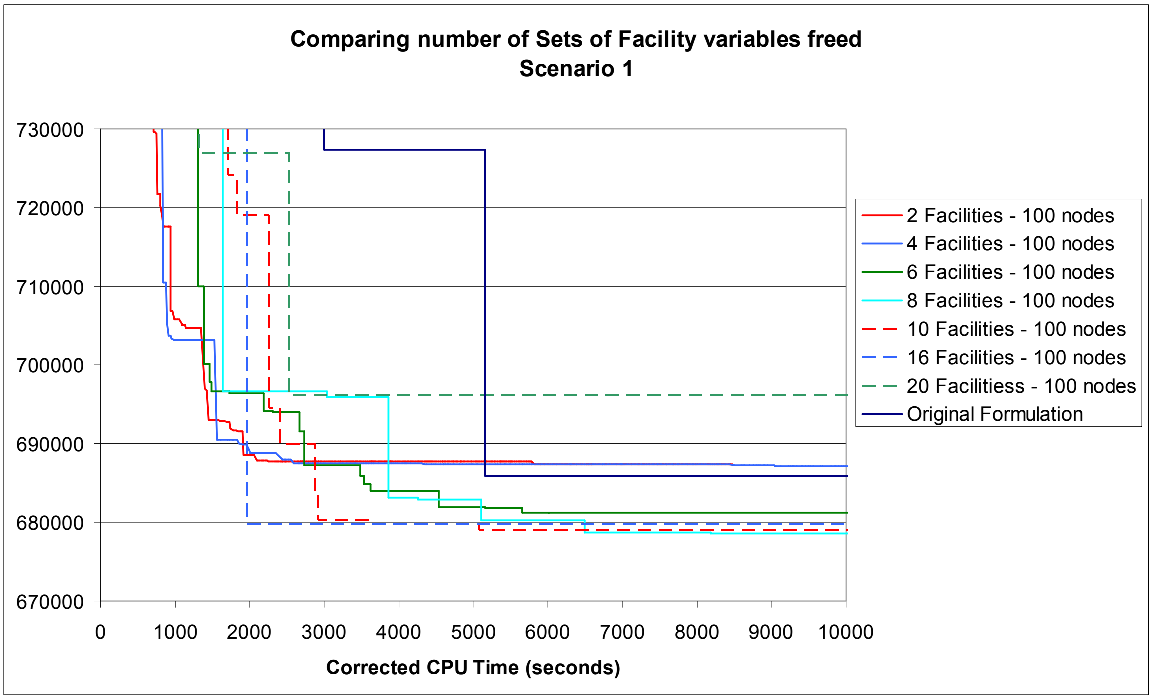

Figure 8-6 - Integer solutions when selecting a number of facilities

Further analysis was carried out on a larger MIP to consider the impact of changing the sub-problem size in this case.

The following graph shows the results that were obtained when a larger main problem was solved. The example used below is a multiple Scenario problem and these are discussed in more detail in chapter 7.

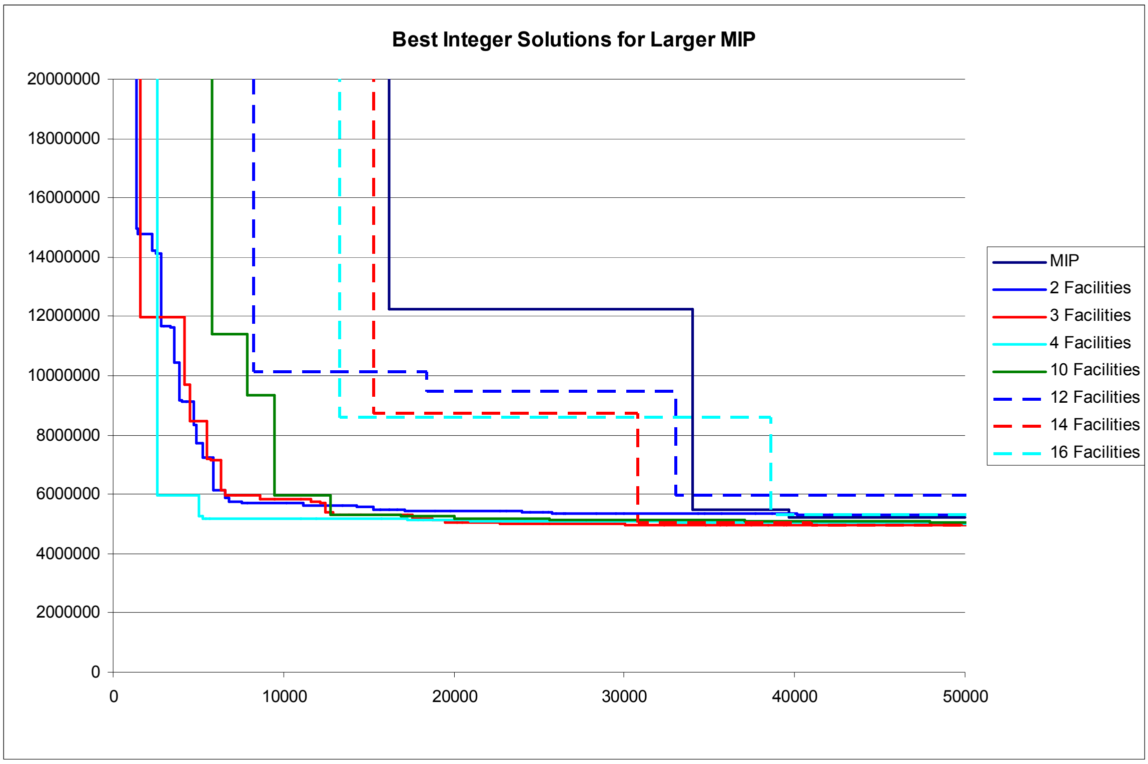

Figure 8-7 - All Integer Solutions for Larger MIP

The graphs (

Figure 8-7 and Figure 8-8) show the results obtained for the various choices of sets of facility variables. Similar results were seen for different numbers of sets of facility variables, but it was seen that too small a set of variables led to poor final results, whereas letting the set of variables included in the sub-problem become too large led to poor early solution improvement and a less predictable convergence of the integer solutions later in the search process (similar to what was seen when solving the main problem).

The time taken for each iteration increased as the sub-problems became larger and long times were being taken to find any solution to the sub-problems once they became large. Iterations were being solved in an average time of approximately 300 seconds for the 2 facility sub-problem, approximately 350 seconds for the 3 facility sub-problems and over 700 seconds per iteration for the 4 facility sub-problems, rising to over 4000 seconds for sub-problem using 10 facilities, over 12000 seconds per iteration for sub-problem using 10 or 12 facilities; and to over 17000s for 16 facilities. When solving sub-problems for 16 facilities, improved solutions were being found with similar frequency to that when solving the whole problem.

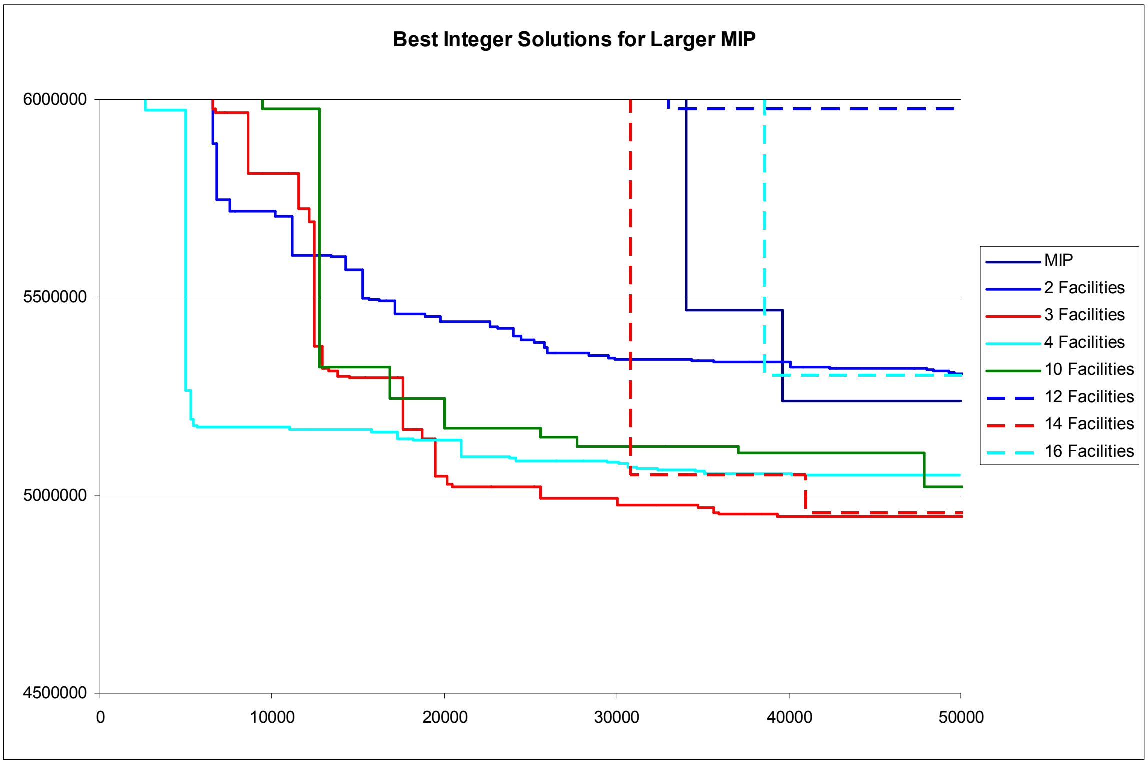

Figure 8-8 - Good Integer Solutions for Larger MIP

8.4.2.2.3 Comparison of sub-problem selection methods

The number and percentage of the integer variables that were included in the sub-problems when variables were selected by facilities was examined.

Table 8-6 gives the maximum, minimum and average numbers of variables included in the sub-problem in each of the first 500 iterations, along with what percentage of the total integer variables this represented.

When using the neighbourhood search algorithm which frees variables by selecting facilities, better integer solutions were obtained with much smaller percentages of variables than when the variables were all selected at random. This is thought to be because the sets of variables chosen for this approach are much more related and allow for a wider variety of changes in the sub-problem solution than when the sets of variables are chosen completely at random from across all the facilities. When choosing values completely at random, for example, the choice of opening or closing a facility will be fixed in many cases by having just one variable value fixed in the sequence of variables coving all the time periods. This situation is avoided in the case when the approach frees up all the variables for a facility.

	Percentage of variables freed during iterations when solving Scenario 1		
Number of set of facilities	**Minimum**	**Maximum**	**Average**
1	78 (1.93%)	282 (7.03%)	169.4 (4.22%)
2	156 (3.88%)	552 (13.75%)	350.1 (8.72%)
3	234 (5.83%)	822 (20.48%)	528.4 (13.16%)
4	330 (8.22%)	1074 (26.76%)	703.9 (17.54%)
5	432 (10.76%)	1350 (33.63%)	878.0 (21.90%)
6	528 (13.15%)	1554 (38.71%)	1053.2 (26.24%)
7	750 (18.86%)	1668 (41.55%)	1229.6 (30.63%)
8	846 (21.07%)	1974 (49.17%)	1399.0 (34.85%)
9	972 (24.22%)	2166 (53.96%)	1575.8 (39.26%)
10	1194 (29.75%)	2304 (57.40%)	1751.5 (43.63%)

Table 8-6 - Percentage of Variables Freed During Iterations

8.4.2.2.4 Mixing these selection methods

Changing the way that the program selects the variables to be freed changes the structure of the neighbourhoods that are used for the local search, and this directly limits the possible changes that can be made to the solution to the main problem. There is no reason for the method of selecting the sub-problem to be the same for each iteration. The practicality and effectiveness of varying the sub-problem selection method between the two previous methods was investigated and shown to be practical. The results obtained on this small problem did not provide improved solutions, but there is potential for them to do so.

8.4.2.3 Stopping criteria for each Iteration

Given that each sub-problem only represents a small part of the overall problem, there may be little point in solving each sub-problem to optimality; it takes time and may not produce sufficient benefit in practice. Solving to optimality may also make the problem too tight in some areas and make it more difficult to find improvement in subsequent iterations.

Therefore the sub-problem search is terminated early in order to move onto another sub-problem. A number of alternative methods of deciding when a sub-problem should be terminated were considered.

8.4.2.3.1 Number of Nodes explored

The impact of limiting the number of nodes explored at each iteration was investigated using the neighbourhoods selected using 3 sets of facility variables, and an initial solution with all the facilities closed. Limits of 50, 100, 200, 500, 1000 and 2000 nodes were imposed for each of the iterations in each solution process.

The following graph clearly illustrates that a node limit of 50 nodes explored in solving each sub-problem was insufficient to allow good improvement in the solution in each iteration and that the 2000 node limit even when it had comparable convergence by number of iterations, had less good results as each iteration was long. Allowing a node limit between 200 and 1000 showed much better convergence to a good solution

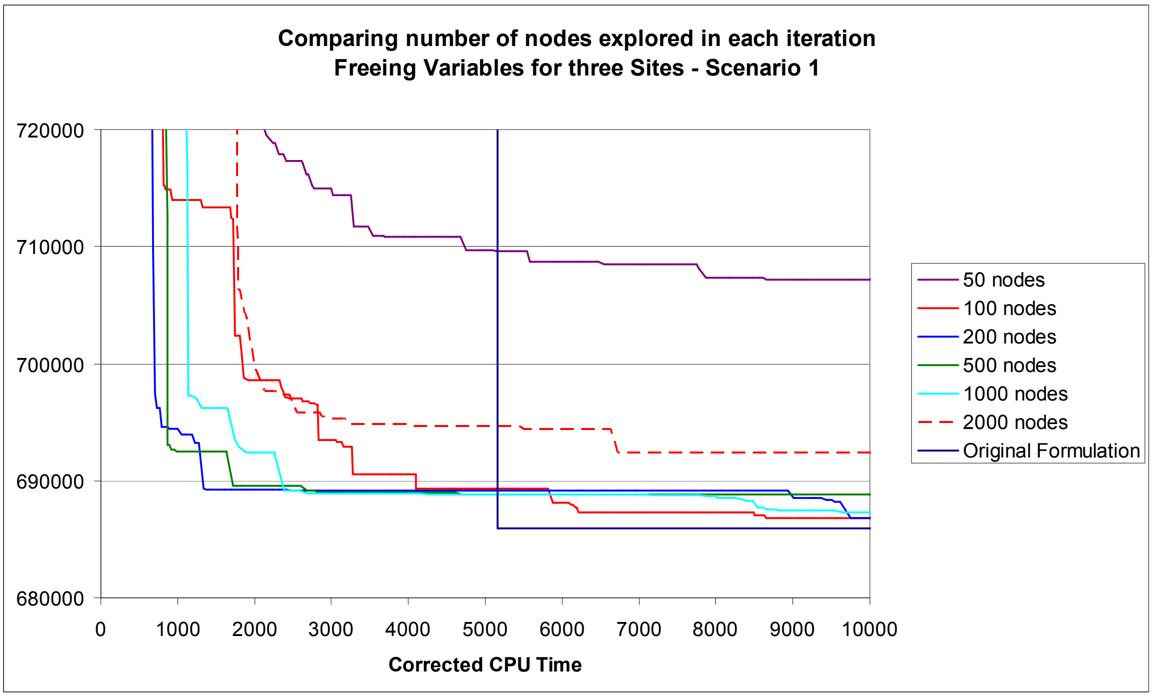

Figure 8-9 - Different number of nodes as the sub-problem stopping criteria

8.4.2.3.2 Time limit

The impact of imposing different time limits on each iteration sub-problem was investigated using the neighbourhoods constructed using 3 set of facility variables, and an initial solution with all the facilities closed. Limits of 20, 30, 40, and 50 seconds were imposed on all of the iterations in each solution process.

Better solutions were seen with intermediate time limits, allowing a sufficient time to let an improvement be found, but short enough that excess time is not wasted in searching for improvements to the sub-problem solution that do not improve the overall search process.

Detailed results for imposing nodes and time limits on two problem instances can be found in Appendix 2.

8.4.2.3.3 Comparison of use of node limits and time limits on iterations.

Comparisons were made of the results achieved using the more successful time and node limit, the following graphs show the 30, 40 or 50 second time limit compared to imposing a limit of 100, 500 or 1000 nodes.

Figure 8-10 shows similar results were obtained using both methods.

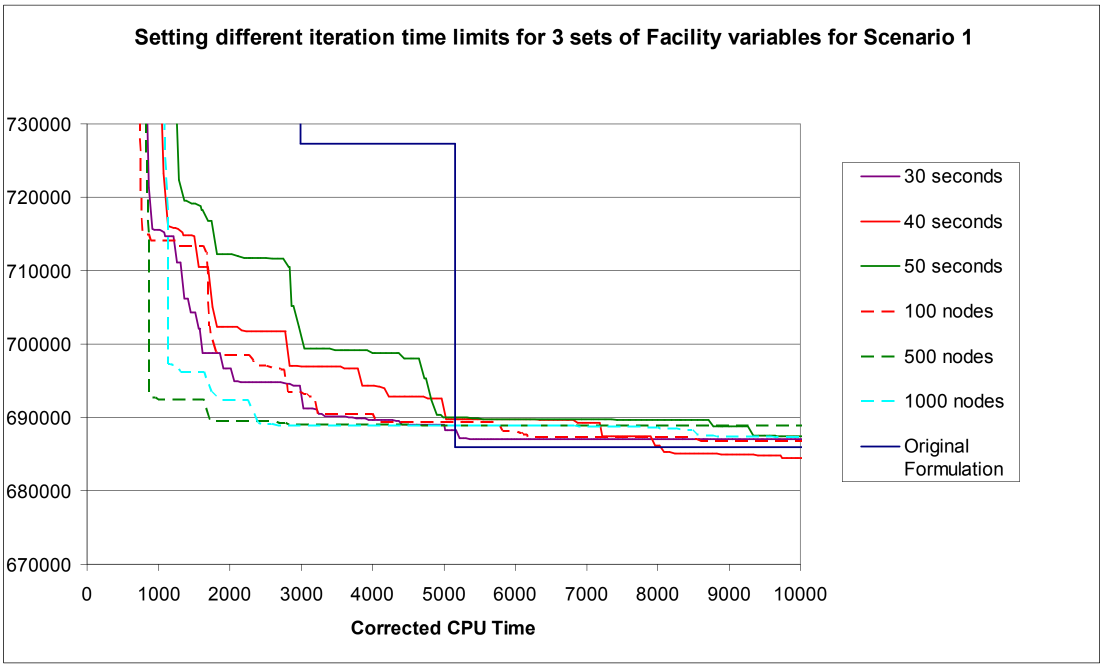

Figure 8-10 - Integer Solutions when using Different Sub-Problem Stopping Criteria

8.4.2.3.4 Optimality Gap

A series of experiments were carried out setting different optimal gap limits for the sub-problems. Setting a single optimality gap as a limit for all the iterations in local search was not effective. Setting a small optimality gap as the termination condition leads to very long solution times for the early iterations, so the overall algorithm is able to explore fewer sub-problems and convergence to a good solution is not very good. Setting a larger optimality gap as the terminating condition allows the overall algorithm to complete more iterations early in the overall search process, but then leads to the search being terminated too early in later iterations with little improvement in the solutions found in those later iterations; again the overall convergence is not as good as the other approaches tried.

Despite these issues there may be several methods that would still allow the optimality gap to be used as a stopping criterion, methods that allow the search for the early iterations to be limited in a different manner and the optimality gap only used for later iterations, or for a variable optimality gap to be used that was reduced as the search proceeds through the iterations, or for a limit on the optimality gap to be used in conjunction with a node or time limit.

8.5 Using Neighbourhood Search to solve multiple Scenarios

The neighbourhood search method described above improves the time taken to find good solutions with a single scenario and also offered a way to make it possible to find solution to the larger models that were obtained when trying to solve multiple scenario problems.

8.5.1 Comparisons of MIP and Neighbourhood Search methods

Increasing numbers of scenarios were solved using both the standard MIP approach and a neighbourhood search method. The examples given here were with the neighbourhood search using three facilities chosen at random, an initial solution obtained by setting all sites to be closed in the first time period and an iteration stopping criterion of 100 nodes or an optimal solution to the sub-problem.

The solution times for the MIP and the Local Search method were compared for increasing numbers of scenarios. Detailed examples of these comparisons are included in Appendix 3

With two scenarios in the model the local search method found better integer solutions more quickly than the MIP, however after around 10000s (almost 3 hours) the MIP was able to find better solutions. A similar effect was seen with models with three scenarios but the time before the MIP was able to achieve comparable integer solutions was extended to around 30000s (more than 8 hours).

Figure 8-11 and Figure 8-12 show the integer solutions obtained using the neighbourhood search method compared with the MIP solutions when solving 3or 4 scenarios.

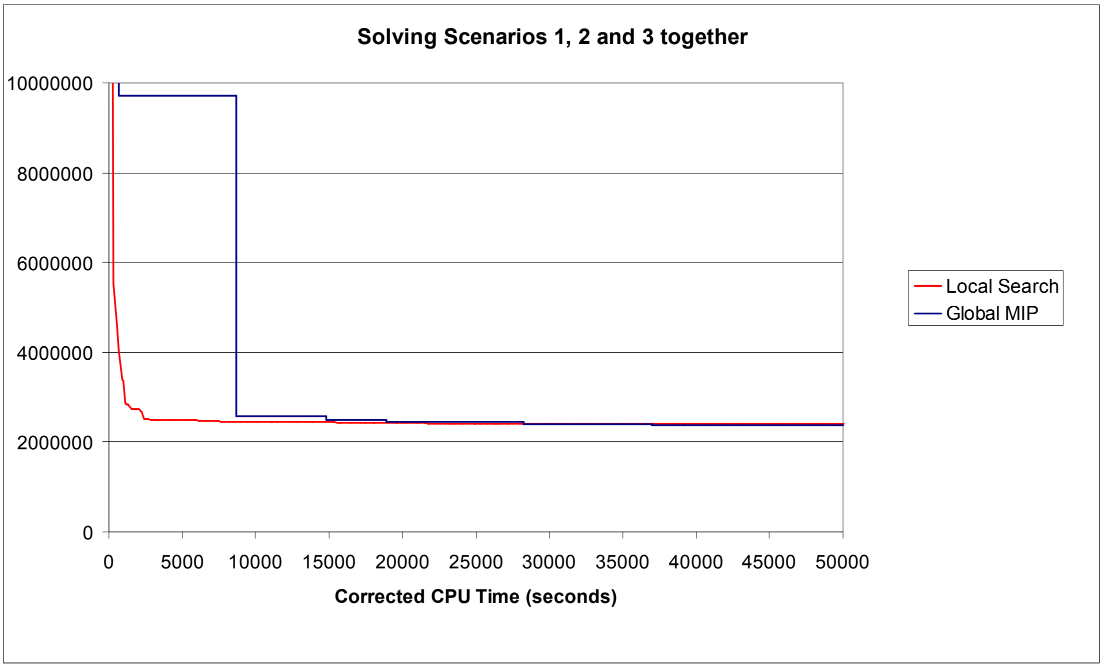

Figure 8-11 - Integer Solution for 3 Scenarios

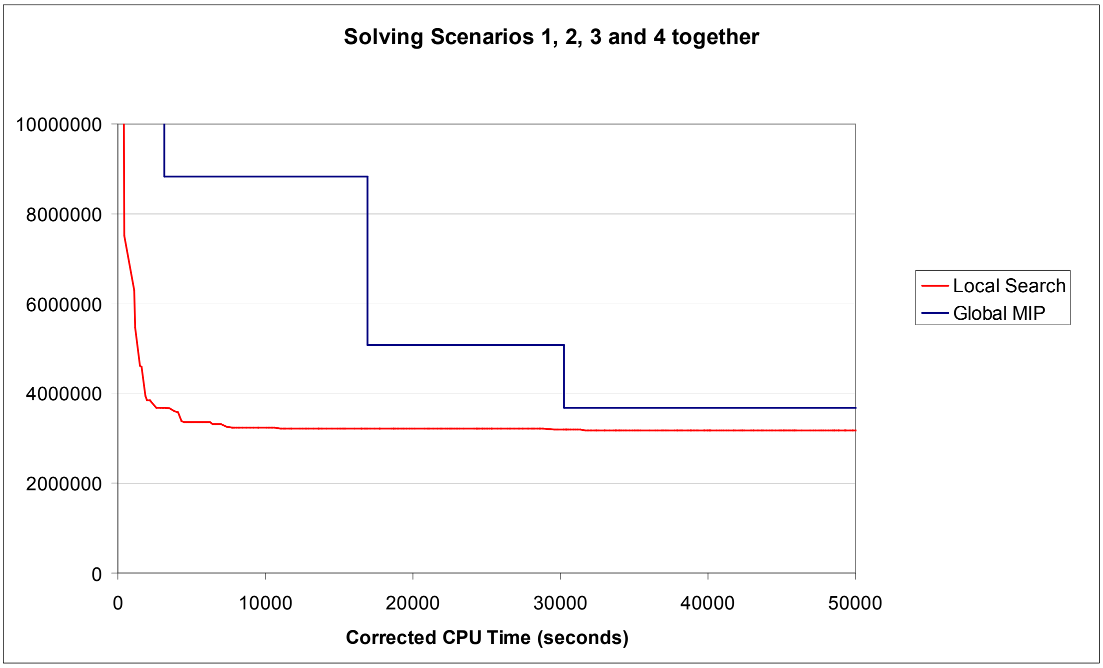

Figure 8-12 - Integer Solution for 4 Scenarios

With 3 scenarios the MIP finds a solution close to that found with the neighbourhood search method at around 9000s, but with 4 scenarios the quality of the solutions found using the neighbourhood search method are much better up to the 50000s time limit.

Although solutions were found using the MIP for models including between 5 and 9 scenarios, the local search approach found better solutions at all times throughout the search. Once 10 scenarios or more were included in the model the MIP failed to find any integer solutions within the 50000s time limit. However, with the local search, solutions were still found when the model contained over 20 scenarios.

8.5.1.1 Comparison of Results for different numbers of Scenarios

The follow set of graphs shows the best MIP and neighbourhood search after 10000, 20000, and 50000s, for 1-10, 15 and 20 scenarios.

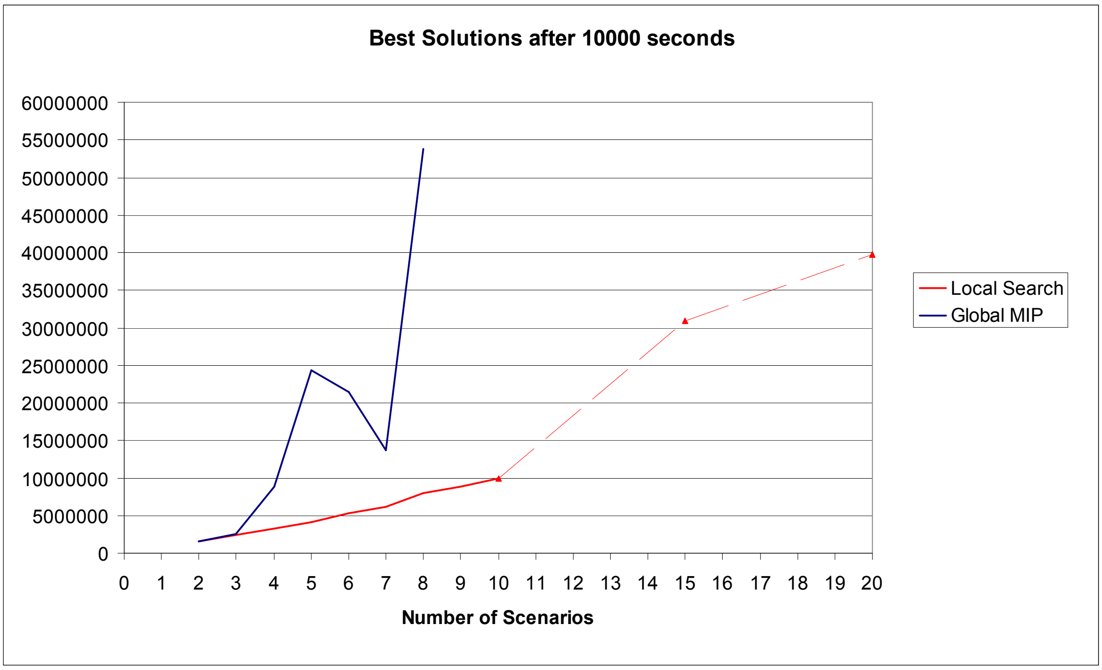

Figure 8-13 - Best Integer Solution after 10000 seconds

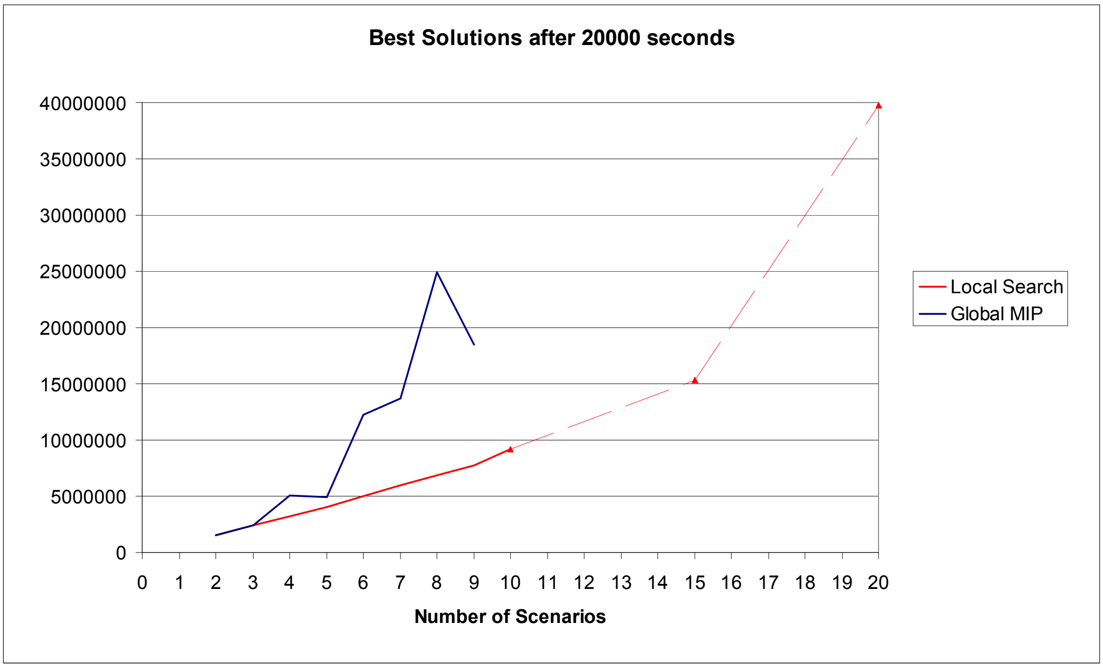

Figure 8-14 - Best Integer Solution after 20000 seconds

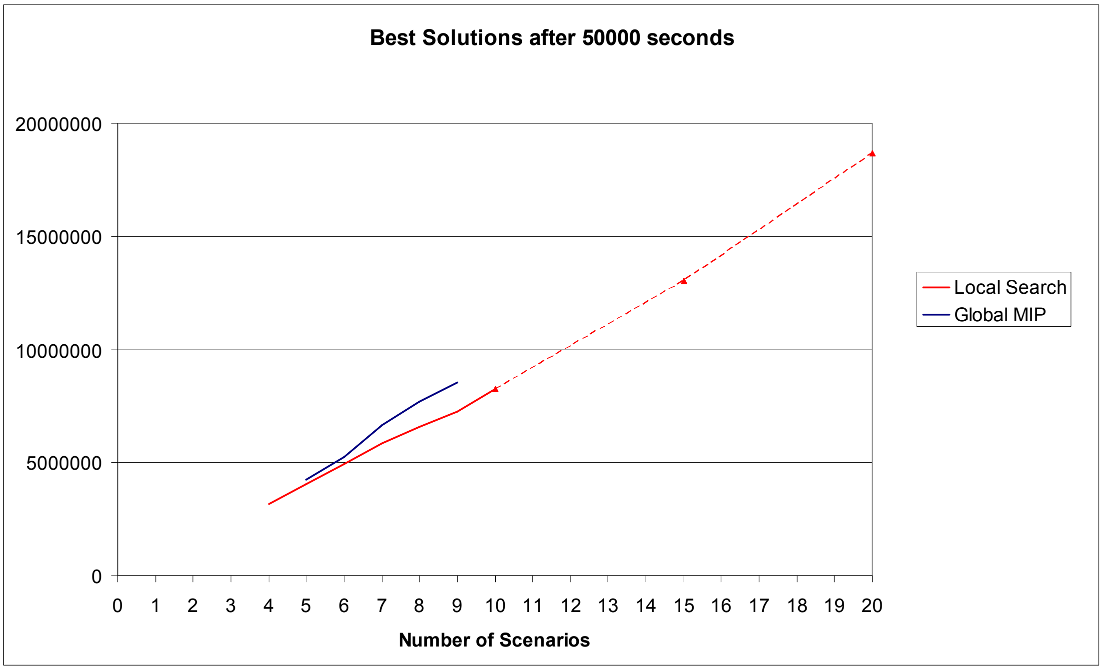

Figure 8-15 - Best Integer Solution after 50000 seconds

The best solutions found show an almost linear relationship between the number of scenarios and the total objective value. This is what we might expect if we are reaching a similar quality of solution because of the increasing costs due to the additional scenarios.

Figure 8-16 shows the average cost per scenario when different numbers of scenario are solved together, this also seems to indicate that good solutions are being obtained for larger numbers of scenarios solved in combination using the neighbourhood search method.

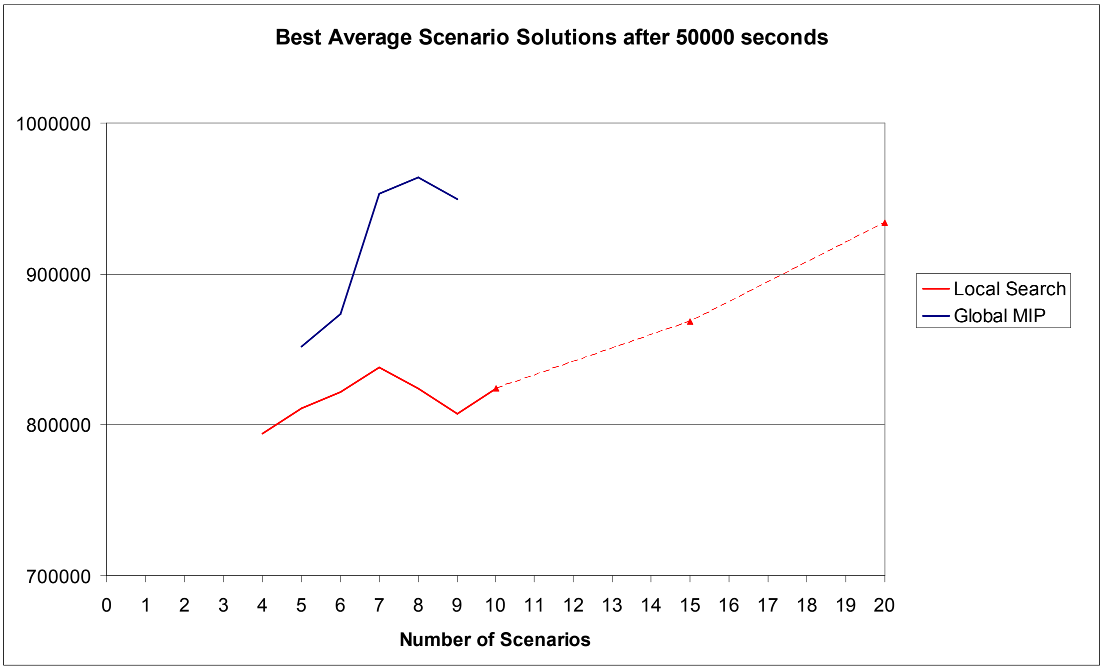

Figure 8-16 - Best Scenario Average after 50000 seconds

8.6 Comparison with Alternate Heuristic

The alternate heuristic described by [Cooper 1964] alternately solved the location and allocation components of a combined location-allocation problem. This approach was extended to tackle bi-linear problems by (Vidal and Goetschalckx 2001, Perron et al 2010). This can be considered as a particular formulation of the neighbourhood search method where the problem is partitioned into two parts and they are repeatedly solved alternately.

Comparisons were carried out for a five scenario problem; all the sub-problems for the experiment were created by selecting a set of facilities and freeing the variables associated with each facility and its associated technologies.

The pair of sub-problems for the alternate heuristic were constructed by randomly selecting 12 facilities and selecting the remaining 11 facilities (Alternate 12|11). This alternating pair of sub-problems were solved repeatedly keeping the same facilities in each of the two sub-problems. Two partitioned sequences of sub-problems were also constructed; firstly 12 facilities were randomly selected at every other step to create the sub-problems, and the remaining 11 facilities were solved in the following step, creating a sequence of partitioned problems (partitioned 12|11); this is similar to the alternate approach but changing which facilities are in each sub-problem after each pair has been solved. Secondly a partitioning created by selecting three groups of 7 facilities until only 2 remained, creating a sequence of sub-problems (partitioned 7|7|7|2) these four sub-problem were solved sequentially and then the selection process repeated to create further different four sub-problem sequences which were solved.

Figure 8-17 clearly shows that all these formulations provided much better solutions than were obtained using the MIP within the 50000s time limit and that all these solutions were converging well toward the best known solution for this problem. This best known solution was obtained by solving the MIP over many days.

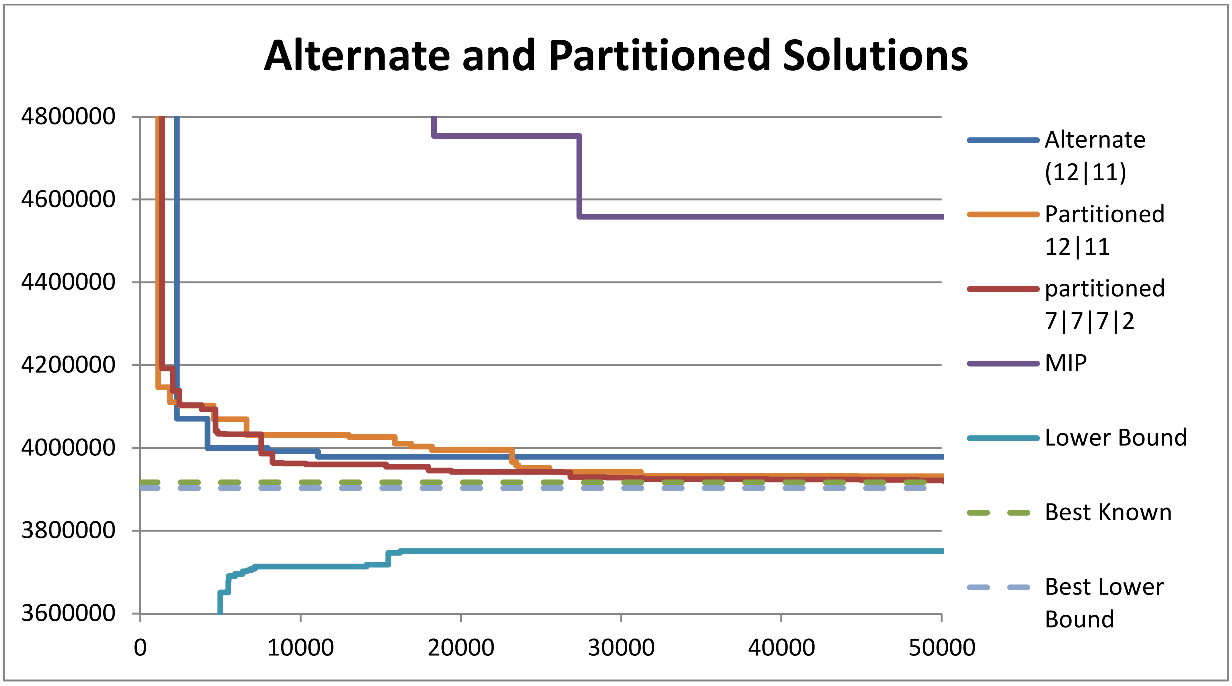

Figure 8-17 - Results for alternate and partitioned sub-problems

Figure 8-18 shows the results for the solutions under 4200000 to allow a more detailed comparison of the convergence. It can be seen that better solutions were obtained with the partitioned sets in comparison the alternate sets, but that partitioning the problem into smaller sub-problems provided a better convergence.

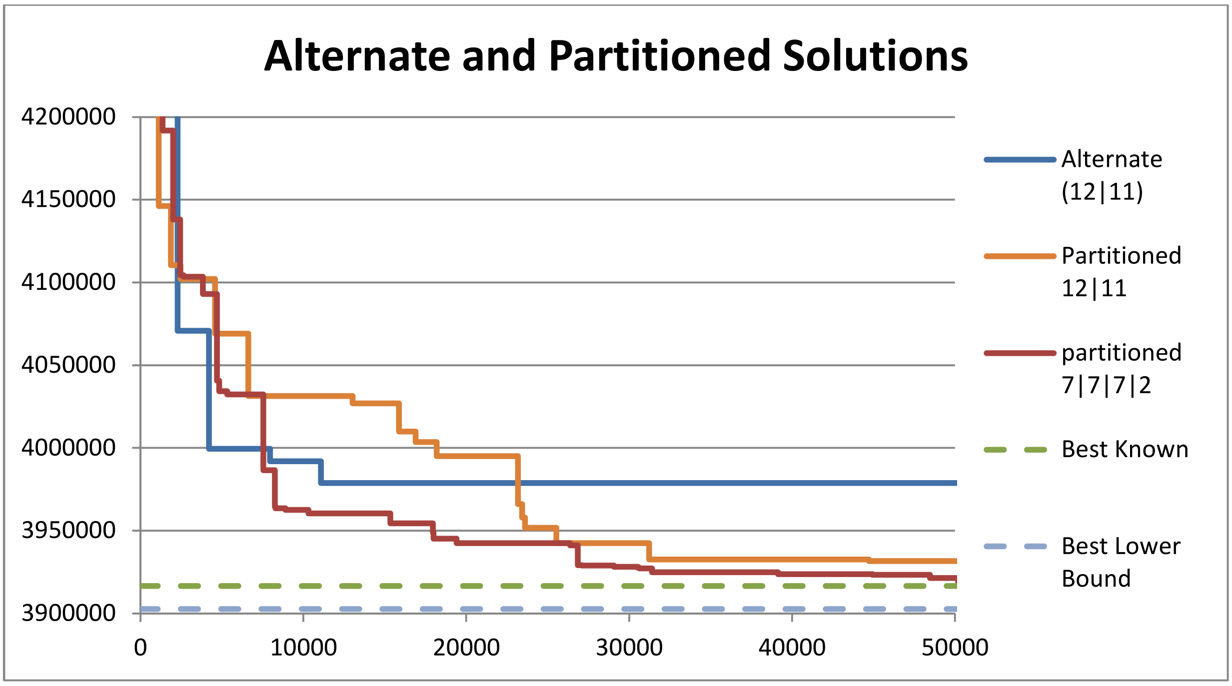

Figure 8-18 - Results under 420000 for alternate and partitioned sub-problems

The inclusion of the alternate heuristic as a solution method that can be used within the generic framework provide the potential to extend the problems that can be solved to those with bi-linear constraints and objectives, such as would be expected if including transfer prices and taxation.

8.7 Conclusions

In most cases neighbourhood search finds much better early solution than are found using the MIP and even for smaller models the later solutions found are comparable with the original MIP formulation.

The neighbourhood search method with adequate tuning provides a set of improving feasible solutions, whereas the MIP has proved much less predictable and considerable times have been seen between improving solutions, making decisions about when to curtail a search very difficult. The more gradual convergence of solutions obtained using neighbourhood search allows better decisions to be made about when a good enough solution has been obtained and about when to curtail the solution process. With good tuning better integer solutions were able to be obtained than could be obtained in the same time using the MIP.

The methods of selecting sub-problems and terminating the search at each iteration have been shown to have a large impact on the effectiveness of this method. It is also likely that some problems will be more sensitive to changes in the initial solution than has been demonstrated for the problem instances used for the experiments described here.

The consistency between solutions found by the neighbourhood search algorithm and the MIP solver again demonstrates the robustness of the generic framework approach and the separation of the representation from the solution technique.

The neighbourhood search method makes it possible to solve the larger problems that are obtained when including many scenarios and the alternate heuristic provides the potential to extend the problems to those with bi-linear constraints and objectives, such as would be expected if including transfer prices and taxation.

9 Conclusion

In this PhD research investigation many aspects of supply chain planning have been studied. We first reviewed the literature on supply chain planning and presented all findings in chapter two. Chapter three describes our formal representation of a generic supply chain model with the relevant data representation. To illustrate the benefits of our generic model, two problem instances have been modelled and described in chapter four. The results of the resulting problems were shown to reproduce those already known. By looking at different technologies for solving the problem we detailed both a MIP and CP reformulation from representing the generic problem. As with all MIP and CP problems there is a trade-off between a more sharp representation of the constraints at the cost of increasing the model dimensions. In chapter five we describe and analyse a number of alternative model representations to gain insight into the trade-offs and give details of our results. As with most planning problems, the data is estimated and in reality it is uncertain. In chapter seven we investigate how this uncertainty could be represented by discrete scenarios about the data. We introduced this to our problem and although faced with the curse of dimensionality, we illustrate how solutions could be achieved accounting for this random data.

In all our solution approaches, it is impossible to achieve proven optimality as it is very time consuming to solve these problems. As a consequence, we have developed a neighbourhood search algorithm which we describe in chapter eight. We have compared this approach with a number of other solution approaches and have demonstrated how effective this method is in terms of solution quality and computation time.

In summary, the main conclusions of this PhD research are:

- the development of a generic SCP problem that is able to represent a wide variety of problems
- the incorporation of multiple features that have only been investigated one by one in the past
- the ability to implement the model in contemporary technologies (relational databases, object oriented programming, mathematical programming model, etc.)
- the capturing and solving of the problem under uncertainty
- the development of a neighbourhood search algorithm which performed better than all other solution approaches in terms of both solution quality and computational time

References

Aghezzaf, E., 2005. Capacity planning and warehouse location in supply chains with uncertain demands. Journal of the Operational Research Society, 56: 453–462.

Ahuja, R.K., Ergun, Ö., Orlin, J.B., Punnen, A.P., 2002. A survey of very large scale neighborhood search techniques. Discrete Applied Mathematics, 123: 75–102.

Applegate, D., Cook, W., 1991. A computational study of the job-shop scheduling problem. ORSA Journal On Computing, 3: 149–156.

Baricelli, P., 1996. An Investigation in a Manufacturing and Supply Chain Planning problem under Uncertainty. MPhil Thesis, Department of Mathematic and Statistics, Brunel University.

Beale, E.M.L., Tomlin, J.A., 1969. Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables, Proceedings of the 5th International Conference on Operations Research, Tavistock, London.

Beamon, B.M., 1998. Supply chain design and analysis: Models and methods, Int. J. Production Economics 55:281-294.

Beasley, J.E., 1988. An algorithm for solving large capacitated warehouse location problems, European Journal of Operational Research 33: 314-325.

Beasley, Accessed Sept 2011.
http://people.brunel.ac.uk/~mastjjb/jeb/info.html
http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/capopt.txt

Benders, J.F., 1962. Partitioning procedures for solving mixed-variables programming problems, Numerische Mathematik, 4: 238-252.

Bidhandi, H.M., Yusuff, R.M., Ahmad, M.M.H.M., Bakar, M.R.A., 2009. European Development of a new approach for deterministic supply chain network design, Journal of Operational Research, 198: 121–128.

Bidhandi, H.M., Yusuff, R.M., 2011. Integrated supply chain planning under uncertainty using an improved stochastic approach, Applied Mathematical Modelling, 35: 2618–2630.

Bisschop, J. 2011. AIMMS Optimization Modeling. http://www.aimms.com/aimms/download/manuals/aimms_modeling.pdf

Caliach, 2011, http://www.caliach.com/caliach/whatiserp/primer3.html

Canel, C., Khumawala, B.M., Law, J., Loh, A., 2001. An algorithm for the capacitated, multi-commodity multi-period facility location problem, Computers & Operations Research, 28: 411-427.

Carchrae, T., Beck. J. C., 2005. Cost-based large neighborhood search. In Workshop on the Combination of Metaheuristic and Local Search with Constraint Programming Techniques.

Caserta, M. and Voß, S., 2009. Metaheuristics: Intelligent problem solving. In Maniezzo, V., Stutzle, T. and Voß, S. (eds). Matheuristics: Hybridizing Metaheuristics and Mathematical Programming, Annals Of Information Systems, Volume 10, Springer, 1-38.

Chippington Derrick, D., Chippington Derrick T.C., 2011a. Experimental Results with Problem Structure-Driven Large Neighbourhood Search for Large Real-World Problems, 9th Metaheuristics International Conference, Udine, Italy,

Chippington Derrick, D., Chippington Derrick T.C., Submitted Aug 2011b. A simple but effective approach to neighbourhood search for large MILP problems. Journal of the Operational Research Society.

Cipriano, R., Di Gaspero, L., Dovier , A. 2009. A Hybrid Solver for Large Neighborhood Search: Mixing Gecode and EasyLocal++ in Proceedings of the 6th International Workshop on Hybrid Metaheuristics, Udine, Italy, Lecture Notes in Computer Science, Vol 5818,. pp. 141-155, Springer-Verlag

Cohen, M.A., Lee, H.A., 1988. Strategic analysis of integrated production-distribution systems: models and methods, Operations Research, 36 (2): 216-228.

Cooper, L., 1964. Heuristic Methods for Location-Allocation Problems, Siam Review, 8(1): 37-53.

Danna, E., Perron, L., 2003. Structured vs. unstructured large neighborhood search, In F. Rossi (Ed.), CP 2003, Lecture Notes in Computer Science, Vol 2833, 817 – 821, Springer-Verlag

Dogan, K., Goetschalckx, M., 1999. A primal decomposition method for the integrated design of multi-period production-distribution systems, Transactions, 31(11): 1027-1036.

Erengüc, S.S., Simpson, N.C., Vakharia, A.J., 1999. Integrated production/distribution planning in supply chains: An invited review, European Journal of Operational Research, 115: 219-236.

FICO, 2009, Getting Started with Xpress, 95-96.
http://www.fico.com/en/FIResourcesLibrary/Xpress_getstart.pdf

Fleischmann, B., Ferber, S., Henrich, P., 2006. Strategic Planning of BMW's Global Production Network Interfaces, 36(3): 194–208.

Georgiadis, M.C., Tsiakis, P., Longinidis, P., Sofioglou, M.K., 2011. Optimal design of supply chain networks under uncertain transient demand variations Omega, 39, pp. 254–272.

Gent, I.P., Walsh, T., Hnich, B., Miguel, I,. Accessed Sept 2011. CSPLib: a problem library for constraints. http://www.csplib.org/

Glover, F., 1989. Tabu Search – Part 1, ORSA Journal on Computing, 1(3): 190-206.

Glover, F., 1990. Tabu Search – Part II, ORSA Journal on Computing, 2(1): 4-32.

Godard, D., Laborie, P., Nuijten, W., 2005. Randomized Large Neighborhood Search for Cumulative Scheduling, In S.Biundo et al (Eds), Proc. International Conference on Automated Planning and Scheduling ICAPS-05, pp81 – 89. AAAI Press.

Goetschalckx, M., Vidal, C.J., Dogan, K., 2002. Modeling and design of global logistics systems: A review of integrated strategic and tactical models and design algorithms, European Journal of Operational Research, 143: 1–18.

Guillén, G., Mele, F.D., Bagajewicz, M.J., Espuña, A., Puigjaner, L., 2005. Multiobjective supply chain design under uncertainty, Chemical Engineering Science, 60: 1535 – 1553.

Gupta, A., Maranas, C.D., 2003. Managing demand uncertainty in supply chain planning, Computers and Chemical Engineering, 27: 1219-1227.

He, F., Qu, R., 2009. A constraint-directed local search approach to nurse rostering problems, 6th International Workshop on Local Search Techniques in Constraint

Satisfaction. Principles and Practice of Constraint Programming, CP2009, Lisbon, Portugal., http://www.cs.nott.ac.uk/~rxq/files/LSCS09.pdf

Hinojosa, Y., Puerto, J., Fernandez, F.R., 2000. A multiperiod two-echelon multicommodity capacitated plant location problem, European Journal of Operational Research, 123: 271-291.

Hinojosa, Y., Kalcsics, J., Nickel, S., Puerto, J., Velten S., 2008. Dynamic supply chain design with inventory, Computers & Operations Research, 35: 373 – 391.

Hugo, A., Pistikopoulos, E.N., 2005. Environmentally conscious long-range planning and design of supply chain networks, Journal of Cleaner Production, 13: 1471-1491.

Hung, W.Y., Samsatli, N.J., Shah, N., 2006. Object-oriented dynamic supply-chain modelling incorporated with production scheduling, European Journal of Operational Research, 169: 1064–1076.

IBM, 2009, IBM ILOG CPLEX V12.1 User's Manual for CPLEX. 626-628. ftp://ftp.software.ibm.com/software/websphere/ilog/docs/optimization/cplex/ps_usrmancplex.pdf

Jang, Y.-J., Jang, S.-Y., Chang, B.-M., Park, J., 2002. A combined model of network design and production/distribution planning for a supply network, Computers & Industrial Engineering, 43: 263-281.

Jayaraman, V., Pirkul, H., 2001. Planning and coordination of production and distribution facilities for multiple commodities, European Journal of Operational Research, 133: 394-408.

Jiao, J., Xu, Q., Wu, Z., Ng, N., 2009. Coordinating product, process, and supply chain decisions: A constraint satisfaction approach, Engineering Applications of Artificial Intelligence, 22: 992–1004.

Jung, H., Jeong, B., Lee, C., 2008. An order quantity negotiation model for distributor-driven supply chains, Int. J. Production Economics, 111: 147–158.

Kallrath, J., 2002. Combined strategic and operational planning – an MILP success story in chemical industry, OR Spectrum, 24: 315–341.

Ko, H.J., Evans, G.W., 2007. A genetic algorithm-based heuristic for the dynamic integrated forward/reverse logistics network for 3PLs, Computers & Operations Research, 34, pp. 346–366.

Laborie, P., Godard, D., 2007. Self-adapting large neighborhood search: Application to single-mode scheduling problems. In Proceedings MISTA-07, Paris. 276–284.

Lejeune, M.A., 2006. A variable neighborhood decomposition search method for supply chain management planning problems, European Journal of Operational Research, 175: 959–976.

Listes, O., Dekker, R., 2005. A stochastic approach to a case study for product recovery network design, European Journal of Operational Research, 160: 268–287.

McKinnon, K.I.M., Williams, H.P., 1989. Constructing integer programming by the predicate calculus, Annals of Operations Research, 21: 227-246.

Martel, A., 2005. The design of production-distribution networks: A mathematical programming approach, In: Geunes, J., Pardalos, P. (Eds.), Supply Chain Optimization. Applied Optimization, Volume 98, pp. 265-305, Springer.

Maximal, 2008, MPL for Windows Manual, section 8.6
http://www.maximal-usa.com/mplman/mpw08060.html

Melachrinoudis, E., Min, H., 2000. The dynamic relocation and phase-out of a hybrid, two-echelon plant/warehousing facility: A multiple objective approach, European Journal of Operational Research, 123: 1-15.

Melo, M.T., Nickel, S., Saldanha da Gama, F., 2005. Dynamic multi-commodity capacitated facility location: a mathematical modeling framework for strategic supply chain planning, Computers & Operations Research, 33: 181–208.

Melo, M.T., Nickel, S., Saldanha-da-Gama, F., 2009. Facility location and supply chain management - A review, European Journal of Operational Research, 196: 401–412.

Mladenovic, N., Labbe, M., Hansen, P., 2003. Solving the p-Center problem with Tabu Search and Variable Neighborhood Search, Networks, 42(1): 48–64.

Min, H., Zhou, G., 2002. Supply chain modeling: past, present and future, Computers & Industrial Engineering, 43: 231-249.

MirHassani, S.A., Lucas, C., Mitra, G., Messina, E., Poojari, C.A., 2000. Computational solution of capacity planning models under uncertainty, Parallel Computing, 26: 511-538.

Mirzapour Al-e-hashem, S.M.J., Malekly, H., Aryanezhad, M.B., 2011. A multi-objective robust optimization model for multi-product multi-site aggregate production planning in a supply chain under uncertainty Int. J. Production Economics (in press).

Mula, J., Peidro, D., Diaz-Madronero, M., Vicens, E., 2010. Mathematical programming models for supply chain production and transport planning, European Journal of Operational Research, 204: 377–390.

Narasimhan, R., Mahapatra, S., 2004. Decision models in global supply chain management, Industrial Marketing Management, 33: 21– 27.

OR Group, Pisa, Accessed Sept 2011. Operations Research Group. Department of Computer Science. University of Pisa, http://www.di.unipi.it/optimize/
http://www.di.unipi.it/optimize/mexch/TabBeasleyMed.pdf
http://www.di.unipi.it/optimize/mexch/TabBeasleyLarge.pdf

Ouhimmou, M., D'Amours, S., Beauregard, R., Ait-Kadi, D., Chauhan, S. S., 2008. Furniture supply chain tactical planning optimization using a time decomposition approach, European Journal of Operational Research, 189, pp. 952–970.

Perron, L., Shaw, P., Furnon, V., 2004. Propagation Guided Large Neighborhood Search, in Wallace, M. (Ed.): CP 2004, Lecture Notes in Computer Science Vol. 3258, pp. 468–481, Springer-Verlag Berlin Heidelberg 2004

Perron, S., Hansen, P., Le Digabel, S., Mladenovic, N., 2010. Exact and heuristic solutions of the global supply chain problem with transfer pricing, European Journal of Operational Research, 202: 64–879.

Pisinger, D. and Ropke, S., 2007. A general heuristic for vehicle routing problems. Computers & Operations Research, 34(8): 2403–2435.

Pisinger, D, Ropke, S., 2010. Large Neighborhood Search in: Gendreau, M. and Potvin, J.-Y. (Eds), Handbook of Metaheuristics, Vol 146 of International Series in Operations Research & Management Science, Springer, Boston, 399-419.

QAD, Accessed 2011. QAD On Demand
http://www.qad.com/Public/Collateral/QAD_On_Demand_fact_sheet.pdf

SAP, Accessed Sept 2011. SAP Business All-In-One; http://www.sap.com/solutions/sme/businessallinone/solutionoverview/index.epx?tab=Fast Start

Sabri, E.H., Beamon, B.M., 2000. A multi-objective approach to simultaneous strategic and operational planning in supply chain design, Omega, 28: 581-598.

Saldanha da Gama, F., Captivo, M.E., 1998. A heuristic approach for the discrete dynamic location problem, Location Science, 6: 211-223.

Salema, M.I.G., Barbosa-Povoa, A.P., Novais, A.Q., 2007. An optimization model for the design of a capacitated multi-product reverse logistics network with uncertainty, European Journal of Operational Research, 179: 1063–1077.

Santoso, T., Ahmed, S., Goetschalckx, M., Shapiro, A., 2005. A stochastic programming approach for supply chain network design under uncertainty, European Journal of Operational Research, 167: 96–115.

Schrimpf, G., Schneider, J., Stamm-Wilbrandt, H., and Dueck, G., 2000. Record breaking optimization results using the ruin and recreate principle. Journal of Computational Physics, 159(2): 139–171.

Shaw, P., 1998. Using constraint programming and local search methods to solve vehicle routing problems. In M. Maher and J-F. Puget (Eds), CP-98 (Fourth International Conference on Principles and Practice of Constraint Programming), Lecture Notes in Computer Science, Vol. 1520, 417–431.

Smith, B., Brailsford, S.C., Hubbard, P.M., Williams, H.P., 1996. The progressive party problem: integer linear programming and constraint programming compared, Constraints, 1: 119-138.

Sniedovich, M. and Voß, S., 2006. The corridor method: a dynamic programming inspired metaheuristic, Control and Cybernetics, 35: 551-578.

Sodhi, M.S., Tang, C.S., 2009. Modeling supply-chain planning under demand uncertainty using stochastic programming: A survey motivated by asset–liability management, Int. J. Production Economics, 121: 728–738.

Srivastava, S.K., 2008. Network design for reverse logistics, Omega, 36: 535–548.

Taillard, E. D., Voß, S., 2002. POPMUSIC – Partial Optimisation Metaheuristic Under Special Intensification Conditions. In C. C. Ribeiro and P. Hansen (eds). Essays and Surveys in Metaheuristics, Operations Research/Computer Science Interfaces Series, Vol. 15, 613-629, Springer.

Thanh, P.N., Bostel, N., Pe´ton, O., 2008. A dynamic model for facility location in the design of complex supply chains, Int. J. Production Economics, 113: 678–693.

Thompson, P. M., Orlin, J. B., 1989. Theory of Cyclic Transfers, Working Paper, Operations Research Center, MIT, Cambridge, Mass.

Thompson, P. M., and Psaraftis, H. N., 1993. Cyclic transfer algorithms for multi vehicle routing and scheduling problems. Operations Research, 41: 935-946.

Timpe, C.H., Kallrath, J., 2000. Optimal planning in large multi-site production networks, European Journal of Operational Research, 126: 422-435.

Tsiakis, P., Shah, N., Pantelides, C. C., 2001. Design of Multi-echelon Supply Chain Networks under Demand Uncertainty, Ind. Eng. Chem. Res., 40: 3585-3604.

Ulstein, N. L., Christiansen, M., Grønhaug, R., Magnussen, N., Solomon, M.M., 2006. Elkem Uses Optimization in Redesigning Its Supply Chain, Interfaces 36(4): 314–325.

Vidal, C.J., Goetschalckx, M., 2001. A global supply chain model with transfer pricing and transportation cost allocation, European Journal of Operational Research, 129: 134-158.

Vila, D., Martel, A., Beauregard, R., 2006. Designing logistics networks in divergent process industries: A methodology and its application to the lumber industry, J. Production Economics, 102: 358–378.

Williams H, P. Model Building in Mathematical programming. Third Ed. 1990, Wiley.

Yan, H., Hooker, J. N., 1999. Tight representation of logical constraints as cardinality rules, Math. Program, 85: 363–377.

Yan, H., Yu, Z., Cheng, T.C.E., 2003. A strategic model for supply chain design with logical constraints: formulation and solution, Computers & Operations Research, 30: 2135–2155.