Synaptic Connection Autonomic Networks

Fang Wang[1], Yaoru Sun[2] and Robert Ghanea-Hercock[1]

[1] Pervasive ICT Research Center, British Telecom, Orion 1/12, Ipswich IP5 3RE, UK

[2] Behaviour and Brain Science Centre, School of Psychology, University of Birmingham,

Birmingham B15 2TT, UK

(fang.wang@bt.com, Y.Sun.1@bham.ac.uk, robert.ghanea-hercock@bt.com)

Abstract

This paper proposes a novel approach to form weighted peer-to-peer networks in a self-organising and decentralised way, termed Synaptic Connection Autonomic Networks (SCAN). Distributed peers in SCAN establish and update their connections or associations based on resource sharing results by following Hebbian learning in a related manner to that in biological neural systems. The strengths of peer associations reflect the utility of one peer to another and continuously adapt over time. In operation SCAN constructs resilient peer-to-peer networks in real time. The result is a more efficient and effective resource sharing mechanism between distributed peers. Simulated experiments verified that SCAN successfully formed P2P networks with correct peer associations and the resource search based on it was continuously improving as the networks were correctly formed.

1. Introduction

The emerging Peer-to-Peer (P2P) systems have brought a new way to harness scattered resources at the edge of networks. For successful remote resource sharing, distributed peers in P2P systems need to preserve information about others so as to direct a search for resources. The associations established between peers create a logical connection of peers and form a peer overlay network on top of the underlying network architecture such as the Internet. As P2P networks are dynamic complex systems involving widely spread and continuously changing peers and resources, how to construct appropriate peer connections to support efficient and effective resource sharing is a significant challenge.

Current P2P systems can be classified into two types, structured and unstructured, according to the methods used to organize peer connections [8]. In structured P2P systems, peers are assigned with static identifiers. Routing tables built on identity distances are distributed onto some if not all of the peers. Typical examples of structured P2P systems include Freenet [4], Chord [11], Pastry [10], Tapestry [13], and P-Grid [1]. These systems usually have pre-defined network topologies and resource placement schemes. Though certain heuristic information is used in Freenet for resource allocation and P-Grid for peer path maintenance, the underlying network structures of these two systems are determined. In structured P2P systems peers are well organised and resource search is relatively straightforward, but substantial knowledge and experience are required for system design and this is always at a cost of increased maintenance to deal with various changes caused by peers/resources joining and leaving. Generally speaking, structured P2P systems target for high resource availability in a relatively persistent environment. In order to do so, they usually put extra and sometimes strict requirements (e.g., fixed data placement) on participating peers. This is sometimes difficult to achieve especially when peers come from heterogeneous and dynamic organizations or locations.

In contrast to structured systems, unstructured P2P systems usually have no global control or layout of the whole system. Peers in unstructured P2P systems manage their own associations during run time. The formation of the overlay network is therefore decentralised and self-organising. In Gnutella [5], for example, peers are self-organised into an overlay network with a power-law degree distribution. In

Anthill [2], peers construct or remove connections to other relevant peers (called neighbours) according to instantaneous information discovered by a special kind of ant agent during search for resources. Because unstructured P2P systems organise peers adaptively and autonomously, less prior knowledge or pre-design is required so the resulting networks can be more resilient to dynamic changes present in the system. Unstructured P2P systems target for flexibility but cannot always guarantee resource availability. The search of resources based on these systems can be very inefficient. For example, the search process in Gnutella is a type of broadcasting, which often causes message flooding over the network. Anthill adopts a search strategy similar to Freenet, again based on predefined peer identifiers. This inevitably restricts search flexibility. Yang and Garcia-Molina [12] improved search speed on Gnutella by dynamically controlling search steps or utilising neighbour information in route selection. This work, together with many others, assumes P2P systems as an undirected graph (i.e., the relations between pairs of peers are symmetric), despite varying peer relationships and conditions. It therefore cannot make the most of any content relationships between peers for resource search and sharing.

This paper proposes a new approach to form weighted peer networks, by drawing inspiration from biological neurons. In the brain, numerous neurons cooperate in a decentralized manner to achieve specific tasks. The connections between neurons are also adjusted in real time through neural synapses. It is obvious that network peers and biological neurons share the same motivation, both of which are task driven to seek an appropriate set of active (social) connections for task achievement. Following Hebbian learning rules found in biological neuron activity, we propose that computing peers in distributed P2P networks can achieve adaptive and efficient resource sharing by actively exploiting peer associations and self-organising synaptic type processes.

In the next section we provide a brief overview of the plastic nature of neural interconnectivity in the brain, which provides a basis of our work. The proposed SCAN system and its associated rules are described in Section 3. Section 4 shows some experimental results which demonstrate the formation of weighted P2P networks using SCAN.

2. Plastic neural interconnectivity in the brain

The mammalian brain, and in particular the highly developed human brain, may be the most complicated network in the world, composed of more than 10^{12} neurons and 10^{23} synapses. The diverse neurons are connected by neural synapses to implement immense capabilities and activities in the brain. Due to the huge number of neurons and synapses, the potential for synaptic interconnectivity may be nearly infinite. Traditional neural models have been structured as a hierarchical structural organization based on which information is transmitted from an origin sensor to target neurons in a specialized region of the brain. Newly emerging studies have recently revealed that the brain does not have a fixed functional or organisational structure. Instead, information flows or communications between neurons can be highly diversified and modulated in nearly real time. Particularly, additional neural clusters such as dormant or under-utilised neuronal-synaptic and dendritic clusters may be recruited instantly in response to afferent information and various environmental or systemic physiological factors. Some synaptic clusters can even be re-adapted from existing functions to achieve other emergent on-demand functionality. An example is the neurons in the regions within the V5 area of the medial superior temporal cortex which respond dynamically and selectively to changes in afferent optic flow [9].

While multiple neuron ensembles participate in task achievement, these ensembles may be located in separate and even widely distant and unrelated regions. It is surprising that sometimes there are no clear definable regulatory centres presented in the brain to centrally control neural coordination [6]. These neural ensembles appear to function in a decentralised and highly collaborative manner through the versatility of ongoing synaptic interconnectivity. Neurons exhibit extensive changes in behaviour by sprouting dendritic protrusions and establishing new synaptic terminal connections. The synaptic pathways involved in a task may respond separately or concurrently with different overlap patterns. When some areas are out of function, new or even unusual patterns may be rapidly tried out for routing and prioritizing information flows. The underlying mechanisms about when and how to change synaptic interconnectivity and implement synaptic coordination remain unclear. A few attempts have used variant Hebbian learning or synaptic-plasticity theory models to explain this kind of decentralised symphonic collaboration between neurons and their neural information fusion.

Hebbian learning proposed by Donald O. Hebb describes how repeated stimulation of specific receptors leads to the formation of neural assemblies that can act as a closed system after stimulation has ceased [7]. This continuous cerebral activity serves as a prolonged duration for neural structural changes to occur during learning. Hebbian learning is therefore a time-dependent mechanism that modifies synaptic efficacy as a function of pre- and post-synaptic activity. Specifically it means that the strength of the action of neuron A onto neuron B increases as A repeatedly participates in firing B.

In connectionism, Hebbian learning and its variations constitute a type of simple but important learning which adjusts a network's weights to reflect its familiarity with inputs in an unsupervised and competitive manner. Procedures similar to Hebbian learning was also used to form knowledge structures on the web [3]. In this paper, we introduce extended Hebbian learning with time decay to organise distributed peers for resource sharing.

3. Synaptic Connection Autonomic Networks

The Synaptic Connection Autonomic Network (SCAN) proposed in this paper provides a novel approach to construct weighted peer networks, inspired by synaptic neural networks in the brain. Peers in SCAN form their associations in real time and in a decentralised and collaborative manner. The peer networks constructed in SCAN are similar to a directed graph, meaning that the connection from peer A to peer B is treated differently from the connection from B to A. The creation of new peer connections, removal of existing connections and adjustment of the strengths of peer connections follows Hebbian learning. Moreover, a decay rule is introduced to simulate the alienation of peers when one has no contribution in resource sharing to another with time. The strengths of peer connections therefore provide an indication of peer utility of one to another. This reflects not only the familiarity of two peers but also a number of factors which may affect the quality of services such as network bandwidth, data transmission speed, peer work load, and value of resources delivered. The strengths of peer connections are essential in resource sharing as it can help choose the most suitable routes for resource search or delivery.

Two major protocols are designed to implement SCAN. The maintenance protocol constructs a weighted, time-dependent peer network, while the search protocol provides a means to locate shared resources across the constructed peer network.

3.1. Network maintenance protocol

SCAN maintains a form of autonomic connectivity between distributed peers. This type of autonomic connectivity is learned and adjusted by following extended Hebbian rules. Specifically, the learning process includes:

- creation of new connections,
- elimination of existing connections, and
- changes in the connection strengths

Suppose w_{ij} is the strength of an association, or virtual connection, from peer *i* to peer *j*. It will be adjusted by a set of rules: the frequency rule, feedback rule, symmetry rule, decay rule, new peer rule and connection removal rule.

Frequency Rule

This rule is deployed when a neuron is fired in response to the firing of another neuron. In SCAN, this happens when a peer *j* provides required resources to a requesting peer *i*. The strength of the connection from peer *i* to *j* is then updated according to equations (1) and (2):

$$w_{ij} = w_{ij} + \Delta w_{ij} \qquad (1)$$

where $\Delta w_{ij} = \gamma \cdot x_i \cdot x_j \qquad (2)$

where γ is the learning rate, x_i is the output of peer *i* which sends a request for particular resources. It has a value of 1 from time *t* when the request is sent to time *t*+*T* and 0 otherwise. x_j is the output of peer *j* replying to the request sent. If peer *j* provides search results within a time period *T*, its output is 1 from time *t'* when peer *j* sends the results to peer *i* to time *t'*+*T*. Otherwise, x_j has a value of 0.

The learning rate is defined as:

$$\gamma = \begin{cases} q \cdot (1 - \frac{\tau}{T+1}) \cdot \sigma & \tau \le T \\ 0 & \tau > T \end{cases} \quad (3)$$

where τ is the time spent for searching a resource, $q \in [-1,1]$ indicates the value or quality of the resources provided, examined by the receiving peer *i*. *T* is a constant indicating the maximum time period allowed for the search. σ is a learning parameter.

During a search, if a solution is found within limited time, the connection from the original requesting peer *i* to peer *k* providing the resource will be strengthened in accordance with Equations (1) (2) (with *k* written instead of *j*). A new connection will be constructed from peer *i* to *k* if they have no link before and the initial strength of the connection is:

$$w_{ik} = \Delta w_{ik} \quad (4)$$

It is worth noting that the frequency rule will reduce the connection strength if the value *q* of the resources provided is negative to peer *i*, that is, the resources provided are not relevant to the search. If no solution is found for a search in limited time period T or a maximum number of forwarding steps *F*, the search is terminated. If desired, the failure information may be sent back to the original requesting peer *i*, so the strengths of peer connections involved will be updated with a negative *q* (e.g. -1). A connection will be eliminated if its strength is below a threshold value ε (e.g., ε=0.001), as explained in the following connection removal rule.

Feedback rule

When a resource search involves more than one peer, all the peers contributing to the search will update their connections involved, after the original requesting peer evaluates the resources provided. The evaluation information works as feedback to reinforce the strengths of the connections involved. The feedback may be positive or negative depending on whether the resources provided are useful. Equation (5) describes the feedback rule.

$$w_{kl} = w_{kl} + \eta_f \cdot \Delta w_{ij} \quad (5)$$

where η_f is a parameter to fine-tune the feedback rule, *k* is any peer involved in the search except peer *j* which provides the resources and *l* is the peer to which the search request was forwarded by peer *k*.

Symmetry Rule

This rule is based on the assumption that if peer *i* is related to peer *j*, then peer *j* is possibly related to peer *i*. The symmetry rule is triggered to enforce the inverse link w_{ji} after w_{ij} is updated, but to a lesser degree. This rule is described as follows.

$$w_{ji} = w_{ji} + \eta_s \cdot \Delta w_{ij} \quad (6)$$

where η_s is a parameter to fine-tune the symmetry rule.

If the feedback rule (or symmetry rule, if used) implies adjustment of a weight for an association that does not currently exist, a new entry is added by following equation (4).

In addition to the above adjustment rules of strengths, SCAN possesses a decay rule to simulate decaying connectivity over time.

Decay rule

In the absence of any stimulus (i.e., lack of contributions in resource sharing in SCAN), the strength of a connection from one peer to another will decay over time. The decay rate can be simulated as the leakage current of a capacitance

$$w_{ij}(s+1) = w_{ij}(s) \cdot \exp\left(-\frac{s}{\eta_d}\right) \qquad (7)$$

where s is the time measured from the point at which the last update in accordance with Equation (1), (5) or (6) occurred.

The decay rule can also be simplified as a linearly regressive function as shown in equation (8).

$$w_{ij}(s+1) = w_{ij}(s) - \eta_d \cdot s \qquad (8)$$

where η_d is a constant between [0,1].

A strength w_{ij} can be considered as 0 if w_{ij} is below the threshold value ε (e.g. 0.001). The connection from peer i to j will be removed from the system if its strength is decreased to zero.

SCAN also has a new peer rule and a connection removal rule to deal with peers entering and leaving the network.

New peer rule

This rule is applied when a new peer joins the network. In order to join the network, this peer should make connections to at least one peer already existing. This peer can be a peer known to the new peer before, or some major peers whose information is bootstrapped by the system.

The new connections between the new peer i and an existing peer j are given a default value if the utility of these two peers are unknown to each other, that is:

$$w_{ij} = \eta_0 \text{ and } w_{ji} = \eta_0 \qquad (9)$$

where η_0 is a predefined constant.

The strengths, however, will be updated in the future through resource sharing. The new peer also may establish new connections to other peers via the Frequency rule, Feedback rule and Symmetry rule.

Connection removal rule

The connection removal rule will be applied if a peer j is found no longer reachable when peer i forwards a search request to it. As mentioned before, the connection removal rule will also be applied when the strength of the link from peer i to j is reduced to a small value ε or less. In both situations, the strength of connection from i to j will be set to zero and the connection will be removed from the network.

3.2. Resource search protocol

By using the above maintenance protocol, peers at distributed locations will form a time-varying, weighted network whose connections indicate the correlations or utility of the peers to each other. A wide range of search protocols can be applied on the formed SCAN architecture to search for suitable resources held by distributed peers, such as those used in Gnutella, Anthill, and [12]. In this section, we introduce a simple but effective search technique to exploit the weighted SCAN network.

Suppose a peer *i* initiates a request for a resource *x*. Peer *i* will first check its neighbouring peers to see if any of them possesses the requested resource. This can be achieved by sending the request directly to the neighbours for an immediate answer or by inspecting the index of neighbours' resources if peer *i* maintains such information.

If the requested resources are owned by some of the neighbours, the resources will be sent to peer *i* and the search is completed successfully. The selection of neighbouring peers to forward the request to is based on their varying strengths from peer *i* by following simple selection rules. The neighbouring peers can even compete for the request. For simplicity we use roulette-wheel selection to choose neighbouring peers, so those peers to which peer *i* has stronger weighted connections will have a higher probability of being selected. After receiving the request, a neighbouring peer *j* checks with its own peer associations to see if any has the required resources. If any neighbour has, the resources will be sent back to peer *i* and the search is terminated.

If the neighbours of peer *j* do not have the required resources either, peer *j* will forward the request to one or more peers from its own contacts. Peers receiving the request will examine their neighbours first and if the required resources are unavailable, they will continue the search by forwarding the request to one or more of their neighbours. This process is repeated until the required resources are found (indicating a search success) or a 'stop' criterion is met (indicating a search failure).

Once the search is finished, the strengths of the peer connections involved during the search will be updated according to the maintenance protocol.

4. Experiments

A set of experiments has been executed to test the SCAN algorithm. In this paper, we focus on SCAN's ability to form a pre-defined target peer network and assess the improved search based on the discovered network.

In the experiments, a set of peer nodes were bootstrapped with a set of randomly initialised connection weights. Every peer held a small database of information (represented by a hashtable of strings). At each step every peer initiated a query for a piece of information and this request was sent to selected neighbouring peers. The 'time-to-live' of a search was defined as 7 hops. Based on search results the simulated peers managed their associations and adjusted the association strengths according to the maintenance protocol introduced in Section 3. Table 1 lists the parameter values used in the experiments.

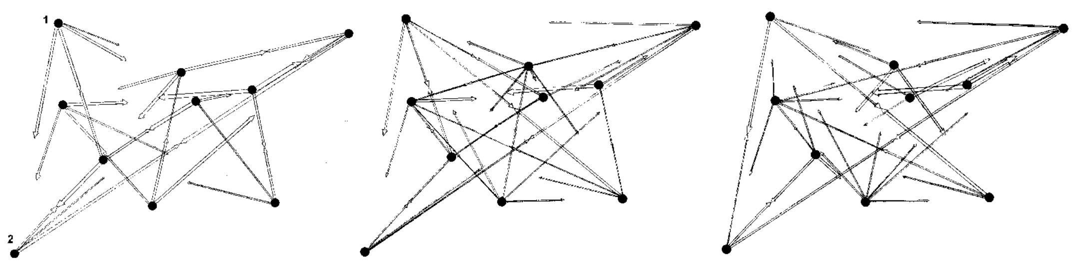
Table 1 Parameters used in experiments

Parameter	Value
T	7 hops
σ	0.25
η_f	0.25
η_s	0.05
η_d (eq. 8)	100
η_0	0.1

4.1. SCAN Formation of P2P networks

The experiments described above were tested on a numbers of peers, ranging from 10 to 100. In each case SCAN successfully learned the initialised connection weights between the peers in a few steps. Figure 1 shows a learning sequence for SCAN with 10 peers in the system. The picture shown in Figure 1(a) is the arbitrarily connected network at step 1. The arrow from peer 1 to peer 2 indicates a connection from peer 1 to peer 2. The width of the arrow implies the strength of the connection. The wider an arrow is, the greater the connection strength. Because there is no connection from peer 2 to 1 at step 1, the other half of the line between peers 1 and 2 is blank.

As learning proceeds, the peer network is modified continuously. Hence peer connections that are unrecognised will be discovered and low value peer connections will be removed. Moreover, the strengths of peer connections will be corrected to reflect the real correlation of two peers. Figure 1 (b) shows the peer network formed after 6 learning steps. Figure 1 (c) gives a picture of the finally learned network after 20 learning steps, which has learnt the correct network topology.



(a) step=1 (b) step=6 (c) step=20

Figure 1 SCAN learning the structure of a P2P network

(number of peers = 10)

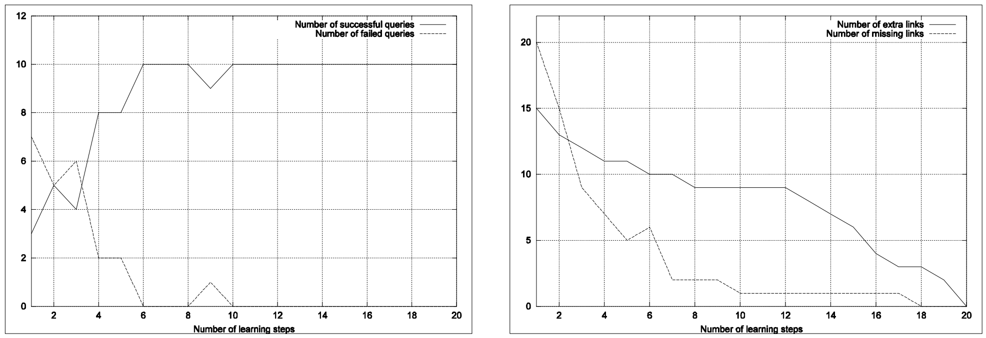
4.2. Improved search results

As correct peer associations are gradually identified and constructed in SCAN, the search for resources becomes easier. Peers that hold resources of interest to other peers will be included in the contacts of these peers. A requested resource can therefore be easily found from a peer’s neighbour or neighbour’s neighbour. Though there is no central control or any global information about the system present, the search request is rapidly spread to the destination peers. As a result, search requests can be answered with an increased success rate.

Figure 2 gives an example of the improved search results when SCAN learned a 10-peer network. Every peer initiated a query at each step, so there were 10 queries at each step in total. From Figure 2 (a), we can see that only 3 out of 10 queries were answered at the beginning of the test, while the other 7 all resulted in a failure. Meanwhile, there were 20 peer connections missing from the network and 15 redundant connections which should not exist, as shown in Figure 2 (b). However, by step 10 the peer network is nearly correctly formed, with 9 extra links and one link missing. The peer queries, at this stage, were all processed successfully. After 20 learning steps, the peer network was correctly formed with correct peer connections, and all launched queries answered satisfactorily.

5. Conclusions

The rapid development of P2P networks urgently requires a more effective and efficient method to organise distributed peers and their resources. However, the heavy dependency on prior knowledge and the inflexibility of existing P2P systems make them incapable of dealing with the inherent distributed and dynamic nature of P2P networks. This paper proposes a novel approach to form Synaptic Connection



(a) (b)

Figure 2 Improved search results in a SCAN learning (number of peers is 10)

Autonomic Networks (SCAN), which builds resilient peer networks in a self-organising and decentralised manner.

The SCAN algorithm proposed in this paper draws inspirations from the adaptive and plastic properties of biological neurons. By following Hebbian learning, distributed peers in SCAN establish their associations according to feedback from resource sharing results. The strengths of peer associations reflect the familiarity or usefulness of one peer to another, which continuously change with time. SCAN can therefore support more efficient and effective resource sharing between distributed peers due to its flexible weighted network structure. Simulated experiments proved that SCAN was able to form P2P networks with pre-set peer associations and the resource search based on this continuously improved as the network was formed.

We believe that SCAN is also adaptive to changing network environments, including the addition of new peers, removal of stale peers and changes of state in existing peers. Further simulation and real-world tests are planned to verify the SCAN algorithm on large-scale P2P networks.

6. References

[1] K. Aberer, P. Cudrĕ-Mauroux, A. Datta, Z. Despotovic, M. Hauswirth and M. Punceva, “P-Grid: A Self-organizing Structured P2P System”, ACM SIGMOD Record, 32(3), 2003.

[2] O. Babaoglu, H. Meling and A. Montresor, “Anthill: A Framework for the Development of Agent-Based Peer-to-Peer Systems”, Proc. of the 22nd Int. Conf. on Distributed Computing Systems, Vienna, Austria, 2003.

[3] J. Bollen, and F. Heylighen, “Algorithms for the self-organisaton of distributed, multi-user networks. Possible application to the future world wide web”, Cybernetics and Systems, 1996, pp. 911-916.

[4] I. Clarke, and O. Sandberg, “Freenet: A distributed anonymous information storage and retrieval system”, ICSI Workshop on Design Issues in Anonymity and Unobservability. California, 2000, pp. 25-26.

[5] Gnutella, http://www.gnutella.com

[6] R.V. Hanwehr, “Information fusion in the human brain: the complexity of de-centralized symphonic collaboration”, in A.K. Hyder, et al. (eds.) Multisensor Fusion, Kluwer Academic Publishers, 2002 pp. 1-36.

[7] D.O. Hebb, The Organization of Behaviour. John Wiley & Sons, New York, 1949.

[8] Q. Lv, P. Cao, E. Cohen, K. Li and S. Shenker, “Search and replication in unstructured peer-to-peer networks”, Proceedings of the 16th international conference on Supercomputing, 2002, pp. 84-95.

[9] M.C. Morrone, M. Tosetti, D. Montanaro, et al., “A cortical area that responds specifically to optic flow, revealed by fMRI”, Nature Neuroscience 3(12), 2002, pp. 1322-1328.

[10] A. Rowstron and P. Druschel, “Pastry: scalable, decentralized object location and routing for largeScale Peer-to-Peer systems”, Proceedings of 18th Conference on Distributed Systems Platforms, Heidelberg, 2001.

[11] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek and H. Balakrishnan, “Chord: A scalable peer-to-peer lookup service for internet applications”, SIGCOMM, 2002, pp. 149-160.

[12] B. Yang and H. Garcia-Molina, “Improving search in peer-to-peer networks”, Proceedings of 22nd International Conference on Distributed Computer Systems, 2002.

[13] B. Zhao, L. Huang, J. Stribling, S.C. Rhea, A.D. Joseph and J. Kubiatowicz, “Tapestry: A resilient global-scale overlay for service deployment”, IEEE Journal on Selected Areas in Communications, 22(1), 2003, pp. 41-53.