Self-Adaptability and Organization for Pervasive Computing and Sensor Network Environments using a Biologically-inspired Approach

Ehssan Sakhaee, Naoki Wakamiya, Masayuki Murata
Graduate School of Information Science and Technology
Osaka University, Osaka, Japan
{sakhaee, wakamiya, murata}@ist.osaka-u.ac.jp

Abstract

In this paper we propose an architecture which integrates the notion of self-adaptability and self-organization in the pervasive computing architecture. Furthermore we describe how a biologically-inspired approach may be a good candidate for this purpose in order to provide a resilient, self-adaptive system similar to living biological systems. Additionally the emergent intelligence that would ultimately encompass the human being and its environment would inevitably assist us where our consciousness is not present.

1. Introduction

A truly pervasive environment has several characteristics. For a pervasive computing environment to be resilient as much as possible, and for the sake of decreasing constant maintenance of a networked pervasive environment, the pervasive computing environment should be able to adapt to environmental changes and self-organize itself. We propose the idea of the third ICT revolution [1] which encompasses biologically-inspired methods and approaches to future communication protocols, with special focus on self-organization in wireless sensor networks. We believe that in order to achieve the highest level of adaptability in a naturalistic way, is to mirror or at the very least be inspired by nature's inherent mechanisms.

The idea is based on an ambient information society, where the artificial intelligence in the environment (consisting of sensors, self-aware systems, and other electronics and computers) readily provide services to individuals by being aware, and able to respond to environmental needs. Additionally, we consider the unpredictability and variability in the environment and the need for a self-adaptive and resilient system which will continue to work effectively when random and unexpected events occur, such as the failure or destruction of individual nodes, displacement due to node mobility, or other unpredictable behaviour in the system and its environment. A reference architecture

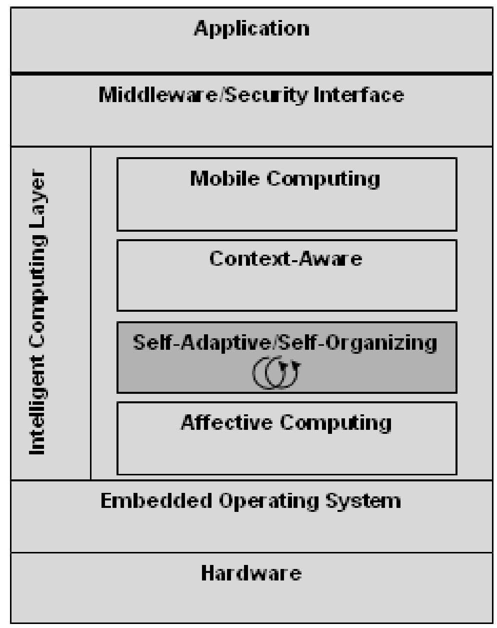

Fig. 1. The extended architecture for pervasive computing.

for pervasive computing was presented in [2]. We would like to extend this architecture by incorporating the "Self-adaptive/Self-organizing" layer into the Intelligent Computing Layer as shown in Figure 1. Some of the features of ambient intelligence (AmI) are but not limited to, embedded, context-aware, personalized, adaptive and anticipatory. This paper introduces a possible approach for self-adaptive and self-organizing networks using the idea of attractors.

The notion of self-organization and attractors is not new. In 1947, Ashby, a British cybernetician proposed "the principle of self-organization" [3] where he introduced the notion of an attractor, stating that any dynamic system will eventually *evolve* towards a state of equilibrium, which he referred to as an *attractor*. The idea of attractors is also central in the field of chaos theory. In general, attractors bring order out of chaos as a system tried to reach a pattern, point of stability or recurrence.

According to [4] a self-organizing system can emerge without a centralised control with simple rules followed by each node (in [4] Kauffman uses synchronizing light bulbs as an example). This involves feedback mechanisms from surrounding nodes, hence the next status of the current node takes into consideration the status of its neighbours. The notion of attractor here is that a repeating sequence of activity will emerge as a result of node interactions. It is claimed that in [4] such attractors are fundamentally the source of intelligence prior to Darwin's theory of natural selection. Furthermore, Kauffman emphasizes the fact that *homeostatis* is another important aspect of stability. *Homeostatis* is the property of resistance to perturbations in the system, and attractors are claimed to also be the basis of such a property. In the case of the light bulb network, by manually interfering and changing the state of one or two light bulbs (i.e. perturbing the system) the system will again return back to its original attractor. In nature this phenomenon can also be seen in the synchronization of the flashing of fireflies.

Similar to Kauffman, the notion of attractor-selection from a gene's perspective was demonstrated in [5] and used in network routing in [6]. In this model the notion of attractor selection is defined by a function, an activity, and noise term. The principal equation is given by the following stochastic differential equation:

$$\frac{dn}{dt} = f(n)\alpha + \eta \qquad (1)$$

In this case, n represents a property for selection, $f(n)$ is a function defining the attractors, α is the activity (where $0 < \alpha < 1$) which defines how suitable the current (found) solution is for attractor selection, and η is the noise term introduced into the system so that random selection is induced in order for an attractor to be selected once the system falls into an unstable state. The activity increases as the solution becomes better, and smaller when the solution is not suitable. Hence as the activity becomes smaller, the noise term dominates and allows random selection to take place until a better solution is found (in which case activity will grow again) and the solution converges to the newly found attractor. The idea behind this mechanism is to induce inherent adaptability and resilience in a network rather than pure optimization which traditional methods seek to achieve.

2. Layered Attractor Selection

We believe that the notion of attractor selection is an effective way to establish a stable system in an adaptable manner, where any perturbations or events that influence a system's current equilibrium state will eventually direct it to a stable attractor, i.e. a new equilibrium state. Furthermore, it may be important to consider a layered approach to attractor selection. To cater for different objectives of the network, both locally and globally, we introduce the layered attractor model which allows independent objectives to balance each other in order to meet a higher order purpose (such as those that occur in biological systems). Furthermore when such a system is subjected to unexpected changes (node failure/destruction, changing network dynamics and so forth) the system is able to reliably recover itself in a self-organized manner. In the proposed model we use the *layered attractor-selection* where the attractor selection is based on that introduced in [5]. The *activity* can be based on different metrics depending on the application and scenario. We will consider two primary forms of activity: global and local activity. In order to demonstrate this concept, we apply it to two clustering problems.

2.1 Attractor Selection for Clustering in a Sensor Network

Firstly, let us consider a clustering scheme aimed at a sensor network, where sensor nodes are grouped into clusters, where a clusterhead is in charge of gathering data from the other sensors, aggregating and reporting this to the sink. In this approach we consider a local objective, e.g. clusterhead election based on a certain metric, for example, energy, and a global objective, such as, minimum hopcount from all nodes to the sink, or some criteria which all clusterheads follow in order to achieve global stability (e.g. a defined number of clusters within the network). In such a model we will have a local attractor and separate activities for each of these objectives. The local objective, is to choose suitable clusterheads, and repair a cluster locally upon clusterhead failure, whereas the global objective will take over where local repair fails, i.e. when the scale of the recovery is too great for local repair to take place. Hence these two separate objectives interact independently, however their goal is *interdependent*. This is illustrated in Figure 2. This system can be described using two stochastic differential equations defined in Equation 1, as follows.

$$\frac{dx}{dt} = f(x)\alpha + \eta \quad (2)$$

$$\frac{dy_i}{dt} = g(y_i)\alpha_i + \eta \quad (3)$$

where x defines the global object's parameter (e.g. number of clusterheads), and y_i defines the local parameter (e.g. the clusterhead selection within the local cluster). Each of these layers have their own independent attractors defined by the functions $f(x)$ and $g(y_i)$ and their own attractors, however the local activity α_i may be derived from the global activity α, i.e. α_i is a function of α, as only the global activity is known and propagated through the network, via the sink. In Figure 2, clusters also exchange their local activity with that of their neighbours, and when the current cluster's activity or the average sum of the adjacent clusters (local) falls below a certain threshold, reclustering of the whole network may need to take place, which is triggered by the global activity. For instance if the activity is based on the residual energy of the network, then nodes may need to reduce their transmission range, and hence more clusters are formed (a global property). However this was a consequence of the local activity.

Often we may not know the optimal performance as conditions of the network and environment change. Particularly when there are more than one objective in a network. Hence having predefined parameters for QoS, or other greedy approaches, may hinder and limit network performance. It is best to allow the network to self-adapt to its changing requirements, and changing environment and find a good solution and selection on its own.

2.2 Attractor Selection for Clustering for Selecting Cluster Information Retrieval

Let us consider another scenario, where nodes are used for obtaining some information from their environment where a user is interested in some certain information. The nodes do not have the intelligence to understand the type of information the user is looking for. The user also does not know to what degree the information it can obtain correlates with the actual information available. Furthermore nodes are arranged in clusters similar to that of Fig. 3. We assume there is one node that can provide the required data, and node data are independent of their neighbouring nodes. Furthermore, clusters are mutually exclusive. A procedure that can be undertaken is as follows.

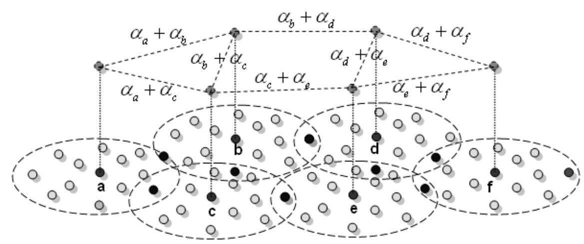

Fig. 2. Activity exchange between clusters.

1. The user's device randomly selects one of the clusters (a node is randomly selected within the cluster) and obtains the reading of that node. If the reading is not satisfactory (low activity), the user's device chooses another cluster. The activity increases when a node within the currently chosen cluster provides the required data (or part of the data). The activity is proportional to the degree of agreeable data. The activity will be highest when the node provides the full data and lower if it is partial.
2. The user's device adaptively chooses clusters (nodes) which progressively provide the satisfactory information. This is further supported by the adaptive selection of nodes within the cluster.

Figure 3 illustrates this idea. The clusters may represent independent networks which are accessible by a user. We note that the approach presented differs from the broadcast query approach in traditional networks. The advantage of the presented approach is as follows.

1. Not all nodes respond to queries, saving energy.
2. The user is in control of the information it needs, not the provider of information.
3. Instead of user saying "give me X", it says "give me what you have" or "give me what you have, close to X". Hence there is no hard-bound rule specifying the exact desired data.
4. The user adaptively decides what is good, and whether a new selection is necessary.
5. The user selects cluster (not individual nodes). The cluster chooses the individual nodes (members of the cluster). This reduces the need for the user's device to possess information about every node in every cluster. This is especially effective with large number of clusters having large number of nodes.

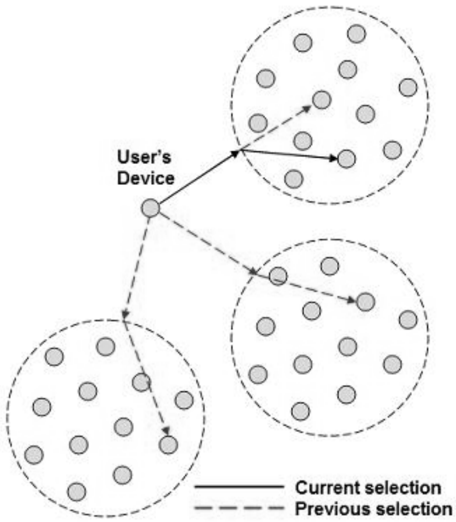

Fig. 3. Selection made by a user's device.

In the above example, each cluster (lower layer) locally selects a "representative" node which offers its information to the user (the higher layer).

The local attractor defines the goodness of the representative node within the cluster. The higher layer (user) chooses among different clusters looking for the information of interest. The higher (global) layer, which is based on the user selection has its own independent activity. The lower layer activity within the cluster has an option of either

1. Independent activity – local metrics define the representative node.
2. Dependent activity – the global metric and requirements affects local selection.

For instance if node within the cluster are restrained by their own limitations (such as energy in case of sensor networks), a local selection is more suitable. However if the limitations are either relaxed or non-existent, then global dependent activity may be applicable. An example of independent activity is the local selection of a representative, or clusterhead node based on energy of the nodes. In such a case the highest energy node is most likely (locally) selected as the representative node to the higher layer (user), and the user will simply choose among the various clusters which best correlate with the information (independent higher layer activity) back to the user. On the other hand, a dependent local activity would consider the user-dependent (higher layer) activity in its local representative node selection. The visible effect of this is that the representative node selected locally is partially dominated and favored by the *user*, and in many cases the same node may be chosen as the representative node if the node is capable of providing the desired data to the higher layer. This of course will happen in an adaptive manner. Hence as the same cluster/node continuously provide the desired data; the selection of the same cluster (and same node) becomes higher, at each round of user's access. When the node can no longer provide the desired data, or the desired data changes, the local (lower layer) and global (higher layer) activities decrease, and random selection will begin again, until the activity increases once again for selecting a certain cluster and local node.

3. Conclusion

In this paper, we have introduced the concept of a biologically-inspired approach for self-adaptation and self-organization in a pervasive environment. This is based on a layered attractor selection model. It is believed that this approach will provide a platform for pervasive computing and sensor network environments to effectively create self-organized and adaptive networks that can surpass the limitation of traditional approaches. Although the ideas presented in this paper are still in their infancy, it is believe their applications may be far-reaching and would have favorable advantages over traditional approaches.

Acknowledgement

This research was supported in part by "Global COE (Centers of Excellence) Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

[1] H. Miyahara, "Center of excellence for founding ambient information society infrastructure technical report," 2007.

[2] Y. Liu, and F. Li, "A reference architecture for pervasive computing," 1st International Symposium on Pervasive Computing and Applications, pp. 99-103, Aug. 2006.

[3] R. Ashby, "Principles of the self-organizing dynamic system," Journal of General Psychology, vol. 37, pp. 125-128, 1947.

[4] S. Kauffman, "At Home In The Universe : The Search for the Laws of Self-Organization and Complexity," Oxford University Press, Nov. 1996.

[5] A. Kashiwagi , I. Urabe, K. Kaneko, and T. Yomo, "Adaptive response of a gene network to environmental changes by fitness-induced attractor selection," PLoS ONE, Dec. 2006.

[6] K. Leibnitz, N. Wakamiya, M. Murata, "Biologically-inspired self-adaptive multi-path routing in overlay networks," Communications of the ACM, Special Issue on Self-Managed Systems and Services, vol. 49, no.3, pp. 62-67, Mar. 2006.