Multi-agent System Architectures for Wireless Sensor Networks

Richard Tynan, G.M.P. O'Hare, David Marsh, and Donal O'Kane

Adaptive Information Cluster (AIC),
University College Dublin
Belfield, Dublin 4, Ireland

{richard.tynan, gregory.ohare, david.marsh, donal.okane}@ucd.ie

Abstract. Traditionally Multi-Agent Systems have been thought of in terms of devices that possess a relatively rich set of resources e.g. power, memory, computational ability and communication bandwidth. This is usually necessary due to the complex deliberation and negotiation processes they require to fulfil their goals. Recently, networked devices have become available on the millimeter scale called Wireless Sensor Networks (WSNs), which pose new challenges because of their constrained resources. However what these devices lack in resources they make up for in numbers due to their small inexpensive nature. In this paper we identify some of the existing MAS architectures for WSNs, and we propose some novel architectures of our own.

1 Introduction

Modern computing networks are comprised of a wide range of heterogeneous devices that may possess various levels of resources and are interconnected using differing networking technologies. This gives rise to a device continuum with high-powered servers connected using high-speed GB/s LANs at one extreme and Mobile Phones connected using GPRS at the other. Multi-Agent Systems for subsets of this spectrum of devices are numerous. Diao et al [1] use autonomic management agents for Web server optimization. At the other end of the scale, Gulliver's Genie uses agents for context sensitive tourist information delivery in the mobile computing arena [2].

Contemporary advances in microprocessor fabrication has lead to a dramatic reduction in the size and power consumed by computational devices. Sensing technology, batteries and radio hardware have also followed a similar minaturisation trend. These factors combined have enabled the production of match-box scale, battery powered devices capable of complex processing tasks called Wireless Sensor Networks, thus heralding a new era for ubiquitous sensing technology. Previous deployments of these networks have been used in many diverse fields such as wildlife habitat monitoring [3], traffic monitoring [4] and lighting control [5].

The device spectrum for computer networks now has a new resource challenged device on which an agent could potentially be deployed - a WSN node. Agents have been deployed previously on WSNs primarily to process the raw data in an intelligent fashion with a view to reducing transmissions - the single biggest factor in determining

the longevity of a network [6], [7]. Agents have also been used in a lighting control application for energy conservation [5]. The research to date has mainly focussed on the application and not on the potential architectures for the MASs. In this paper we identify some existing MAS architectures and propose some new architectures.

2 Device Hierarchy

When considering the diverse range of devices that can comprise a modern computer network where an agent may be deployed it is useful to take into account of the resources available at each device. For our purposes we have identified a five tier hierarchy of devices. Figure 1 depicts this hierarchy and a device typical of each layer.

Fig. 1. Hierarchy of devices according to resources available

A more precise description of this hierarchy can be seen in table 1. A Level 4 device we consider to have unbounded resources in terms of the 4 criteria. Devices of this nature are connected to a mains supply and have vast memory and computational resources. Also their communication occurs over high-speed networks so network bandwidth is very high. Below this level the first distinction we make is power. We now enter the mobile computing arena based on battery powered devices. The next distinction that occurs is based on communication capabilities i.e. a laptop is usually connected to other nodes through a wireless LAN whereas PDAs and mobile phones use 3G technology and GPRS. The criteria separating level 2 and level 1 devices are CPU constraints. Due to the more complex operation of a PDA, compared to the mobile phone, it requires a faster CPU which would allow deeper agent deliberation. The final distinction between devices is that of memory. Modern mobile phone memory is of the order of megabytes whereas, a WSN node typically only has a few kilobytes.

Table 1. Table representing the various constraints of a device at a given level. An x indicates that this constraint is not a concern for an agent on a device in this layer. An o implies that the agent must be aware of this constraint

	Power	Communication	CPU	Memory
Level 4	x	x	x	x
Level 3	o	x	x	x
Level 2	o	o	x	x
Level 1	o	o	o	x
Level 0	o	o	o	o

So what implication does this have for MASSs? An agent must be aware of its resident devices abilities and factor these into its deliberation about future actions. However this hierarchy becomes more interesting when we look at mobile agents. When an agent migrates between devices of different levels in the hierarchy the agent may opportunistically take advantage of the differing resources on a device. An agent migrating to a lower level device must be aware of this and tailor its reasoning process accordingly. For instance, when migrating from a level 4 to a level 3 device, an agent must be aware that, should the device’s power become critically low, it should have a route available to it to leave the device. In the case of moving from a level 2 to a level 1 device, the agent may have to adapt to a simplified version of the agent platform that does not provide services common to larger platforms. When moving to a level 0 device, it is likely that the encoding of the agent’s mental state will have to be radically reduced. Of course, when moving up through the hierarchy, an agent should become aware, or be made aware, that its operation is no longer as restricted as before, and tailor its behavior accordingly.

3 Wireless Sensor Networks

Wireless Sensor Networks (WSNs) are made possible by the minaturisation of their components, namely microprocessors, sensors and radio communication hardware. The main components for a WSN are a base-station and one or more sensor nodes. The sensor nodes relay their sensed data either directly to the base station or through each other (termed multi-hopping) depending on the scale of the network. In turn the base station sends commands to the nodes to, for example, increase their sampling frequency.

Multi-hopping, while useful in extending the reach or scale of a WSN, does have its pitfalls. The cost of transmitting a packet can be greatly increased depending on the distance a node is away from the base station. Secondly, since nodes nearest the base station, i.e. 1 hop away, will not only have to send their data but also that of all other nodes greater than a single hop, there will be a greater demand placed on the power supply of these nodes. It means that, in general, a nodes lifespan is inversely proportional to the number of hops it is away from the base station. To alleviate this problem, multiple base stations can be used, with the nodes only transmitting data to their local station. A second solution creates a hierarchy of nodes with varying power

and transmission capabilities. Higher power nodes act as a networked backbone for the lower powered sensors. One final alternative is to populate the area closest to the base station more densely than in outlying areas.

In some WSNs the nodes of the network can perform considerable processing tasks. This facilitates the examination of raw sensed data in an intelligent fashion with transmissions only containing summarized information rather than the raw data i.e. a single packet containing the average temperature over say ten seconds could be transmitted instead of ten individual packets containing the temperature as sampled at each second. Processing capabilities can vary from platform to platform. We have chosen the Mica2 Mote [8] for our experimentation. It has 4KB of RAM, 128KB of instruction memory and 512KB of flash memory for data storage purposes. They are also equipped with a suite of sensory modalities, heat, light, sound, barometric pressure and humidity.

4 Agents for Wireless Sensor Networks

Having seen that current research endeavors have shown not only the feasibility but also the utility for agents on WSNs we now turn our attention to candidate architectures that may prove appropriate.

4.1 WSN Nodes as Perceptors

The first architecture model considers individual nodes as perceptors for a single agent, as depicted in figure 2. On the perceive part of the perceive-deliberative-act cycle, the agent requests a sensed value from each node, which it then uses as input to the deliberative part of that cycle. This has a number of benefits and restrictions. Chief among the benefits is ease of programming, since there is a single control point in the system. The system will also be more synchronized with regard to the timing of transmissions, and hence more efficient. This is because any collisions during transmission necessitates a costly retransmission, so synchronization brings direct energy benefits.

The restrictions of this architecture are typical of centralized systems. The primary disadvantage is that the system is not scalable. There are two main reasons for this. Firstly, since each node must be queried to generate a perceptor value, as the number of nodes increases so does the length of the deliberative cycle. With current node radio speeds, the length of the cycle could quickly grow past one second. This is compounded by the second scalability problem, which is that with larger numbers of nodes, multi-hop communication is increasingly likely to be needed. This invariably leads to even greater time delays for the agent while it gathers its data, while energy consumption also rises. An additional problem is that if two or more agents wish to use the network as a set of perceptors, they will have to coordinate with each other to synchronize access, and of course they will be subjected to additional delays while waiting for the other agents to finish with the network. Larger networks also increase the likelihood of having unconnected parts of the network. In this architecture, these parts would remain inactive as long as they are unable to receive messages from the agent. This sort of architecture is probably most suited to small-scale networks which are not likely to have more than

Fig. 2. WSN as perceptor to central base station agent (left), collaborative agent approach where agents are considered peers (right)

a single data sink (i.e. agent that wishes to use the network). As such it rarely exists in practice.

4.2 One Agent per Sensor Node

In this architecture, each node in the network hosts an agent which is responsible for the behavior of that node (figure 2). This approach represents a distributed method of network control. The agents can co-ordinate their behavior to achieve both local and global goals, and so the network may be employed in more useful ways than in the simple perceptor case. For example, network services such as packet routing can be built within the framework of collaborating agents, whereas this is not possible in the centralized case where a layer independent of the controlling agent would be required.

One of the primary benefits of this architecture is scalability. Local tasks can be performed without multi-hopping messages across the network to a central controller, which leads to immediate energy and latency gains. Concurrency is another gain associated with decentralizing the responsibility for processing. This allows the network to perform more work in a given time period than could be achieved by other means. The effects of network partitioning are also alleviated somewhat by using this architecture, since partitioned regions can still perform local tasks.

Disadvantages of this method include the difficulty of creating systems of agents that can realize global goals using only local information. The cost of negotiation between agents is an overhead avoided by the centralized architecture. In a network of heterogeneous nodes, i.e. nodes with varying degrees of sensing and in particular computational resources, an agent on a faster node may experience little performance gain because of the limitations of the surrounding agents that must run on slower nodes. This architecture is most suitable when the sensors have some form of local actuation, as seen in [5], and [9].

4.3 Mobile Agents

This scheme uses an agent with migratory capabilities which visits a set of nodes, accumulating state as it does so. This is a typical method for data aggregation/fusion applications and is shown in figure 3. Benefits include low overall computational costs, as no more than one node is active at a time. Since a single transmission and reception event is all that is required of an individual node in order to facilitate the agent's migration, communication is also generally quite low. Simplicity is another benefit, since the agent manages both the task of collecting data and its visitation itinerary. The negative aspects of network partitioning have less impact on this method since an agent can simply alter its visitation itinerary to go to the nodes it can reach until connectivity is restored.

Fig. 3. Agent navigates the network for its control and data harvesting requirements

One disadvantage of this architecture is that if the node the agent is currently on fails (for example the battery depletes before the agent can migrate elsewhere), all data that has been accumulated by it so far would be lost. Methods to counter this include leaving copies of the state of the agent on previously visited nodes, which could "revive" the agent using this information in the event that the original was lost. Secondly, if latency is an important concern then this method is likely to be less efficient compared to the nodes-as-agents architecture because of the lack of concurrency. If a network is very large, it may take too much time for a single agent to gather all the data. This scheme is likely to be useful where nodes collect a lot of data before processing it [7].

Our initial experience with implementing MASs for WSNs has highlighted the advantages and disadvantages of the previous architectures, and motivated the development of the following 2 combined approaches. These attempt to harness the advantages of a number of the previous architectures while minimizing their disadvantages.

4.4 Hybrid Approach

If we roughly describe the advantages of the three methods above as being simplicity, robustness and efficiency respectively, and their weaknesses as being scalability, complexity and latency, then we can attempt to combine them in such a way as to reduce the

impact of the weaknesses while retaining the advantages. This leads to the following hybrid approaches as illustrated in figure 4.

Collaborative Agents: Each agent is a perceptor type group. As was mentioned above, the first architecture is not suitable for larger networks of sensors. However, if the network is divided into numerous small groups of nodes, each one could contain a single agent that is responsible for several nodes. The system can be viewed at a high level as a collection of collaborating agents, and at a low level as a collection of nodes being managed by one agent. At the level of the individual agent, most of the problems associated with the lack of scalability will no longer apply once the number of nodes the agent is responsible for is low enough. While network partitions may still cause problems, the chance of them occurring in a small group of nodes is much reduced, and the groups of nodes could be organized around such disturbances. The disadvantages of the collaborative agent architecture are lessened by combining it with the first approach. Since there are less agents, co-ordination cost between agents are reduced. Heterogeneous networks could be organized in such a way that every agent operates on an approximately equivalent set of nodes, which should mask the inequalities present when each node has its own agent.

Fig. 4. Agents are allotted a subset of the network for control based on the perceptor model. Agent cooperation then performs high level control and harvesting (left). The agent controls and senses its environment by migrating to the sensor nodes under its control (right)

Collaborative Agents: Each agent is the domain for a mobile agent. Combining the second and third architectures also reduces some of the problems that they have on their own. Once again, the network can be divided into small groups of nodes, but in this instance each group has a single mobile agent that migrates between the constituent nodes. The latency issue for the migrating agent architecture is reduced by reducing the number of nodes it has to traverse. The probability of an agent being trapped on a node that is going to fail is also reduced as the number of nodes it travels among lessens. For the same reasons as in the first combined architecture, the disadvantages of the collaborative approach are reduced when combined with the mobile agent case.

5 Conclusions

Recent work has shown that multi-agent systems can be deployed on the new wave of computationally limited devices that have emerged in the last few years. One of the latest classes of these devices is the wireless sensor node. In line with the general trend of adapting agents to work on decreasingly powerful equipment, the use of agents on such simple platforms brings a new set of challenges. We detailed the restrictions that exist at the different levels of sophistication in order to illuminate the problems faced by an agent when executing at the various levels. We then provided a brief introduction to wireless sensors networks, specifying their components, the capabilities of these components, the pressing research issues and an example of some WSN hardware. Finally we outline MAS architectures that we believe are appropriate to use on WSNs. We give details of their respective advantages and disadvantages, and with these points as a guideline we posit the situations in which they are likely to be useful. Included is a pair of hybrid architectures, with the elements combined in order to minimize their negatives while maintaining their positive qualities.

Acknowledgements

This material is based on works supported by Science Foundation Ireland under Grant No. 03/IN.3/I361. We would also like to thank the Irish Research Council for Science, Engineering and Technology (IRCSET) for their support.

References

1. Diao, Y., Hellerstein, J.L., Parekh, S., Bigus, J.P.: Managing Web Server Performance with AutoTune Agents. *IBM Systems Journal* **42** (2003) 136–149
2. O'Grady, M.J., O'Hare, G.M.: Gullivers's genie: Agency, mobility and adaptivity. *Computers and Graphics Journal, Special Issue on Pervasive Computing and Ambient Intelligence - Mobility, Ubiquity and Wearables Get Together*. **28** (2004)
3. Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., Anderson, J.: Wireless Sensor Networks for Habitat Monitoring. In: *International Workshop on Wireless Sensor Networks and Applications*. (2002)
4. Coleri, S., Cheung, S.Y., Varaiya, P.: Sensor Networks for Monitoring Traffic. In: *Allerton Conference on Communication, Control and Computing*. (2004)
5. Sandhu, J., Agogino, A., Agogino, A.: Wireless Sensor Networks for Commercial Lighting Control: Decision Making with Multi-agent Systems. In: *AAAI Workshop on Sensor Networks*. (2004)
6. Marsh, D., Tynan, R., O'Kane, D., O'Hare, G.: Autonomic wireless sensor networks. *Engineering Applications of Artificial Intelligence* (2004)
7. Qi, H., Wang, X., Iyengar, S.S., Chakrabarty, K.: Multisensor Data Fusion in Distributed Sensor Networks Using Mobile Agents. In: *4th International Conference Information Fusion*. (2001)
8. Hill, J.: System Architecture for Wireless Sensor Networks. PhD thesis, UC Berkeley (2003)
9. Braginsky, D., Estrin, D.: Rumor routing algorithm for sensor networks. In: *WSNA*. (2002)