4.6   MARINER: Multi-Agent Architecture for Distributed-IN Load Control and Overload Protection

Brendan Jennings, Teltec Ireland DCU, brendan.jennings@teltec.dcu.ie

4.6.1   Agent Technology for Load Control in Intelligent Networks

Rapid technological advances have encouraged ever-greater usage of telecommunications services, both in terms of the number of users and the information volumes that they produce. Continued growth in the market is welcome from the point of view of network operators and service providers, however the task of meeting customers' availability and service quality demands is becoming increasingly challenging. Convergence of fixed, mobile and Internet networks, together with the expected introduction of service delivery and management platforms based on distributed object technology is resulting in an increasingly complex, interconnected infrastructure. This infrastructure will require network management systems that are more responsive, adaptive, proactive and less centralised than those currently deployed. Since these are properties of agents and multi-agent systems many in the telecommunications research community have recognised that agent-based technology has the potential to offer a timely solution to the growing problem of designing efficient and flexible management systems.

ACTS project MARINER (Multi-Agent Architecture for Distributed-IN Load Control and Overload Protection) has been assessing the usefulness of agent technology for the enhancement of the performance and management of complex distributed telecommunications network environments. Recent years have seen the development of a variety of agent languages, architectures and platforms, however efforts towards the standardisation of agent facilities has only commenced, mainly within the Foundation for Intelligent Physical Agents (FIPA) and more recently the Object Management Group (OMG).

MARINER has contributed to the validation of the FIPA model for agent-based systems through the design, implementation and trialling of a prototype multi-agent system for Intelligent Network (IN) load control. This application places very stringent real-time performance constraints on the operation of agents, thus it has served as an excellent basis for the identification of potential enhancements to the FIPA specifications to facilitate their use in high performance, mission critical application domains.

Intelligent Network implementations are the prevalent architecture for the delivery of advanced telephony-based services in today's public network infrastructure and are widely seen as the basis for the service delivery platforms of the future. As customer demands on the IN grow, the task of dimensioning networks to meet the performance requirements of an, as yet unknown, set of services and their associated usage demands becomes very difficult. Because of this, networks are growing more susceptible to overloads that result in customer dissatisfaction and, in cases where service requests are abandoned, loss in revenue for the network operator.
In this context the increasing importance of efficient and flexible load control mechanisms is clear. Traditional load control mechanisms operate by protecting individual network nodes, however the agent-based approach offers the potential to develop more advanced and effective network-based strategies, which take into account the load situation in the network as a whole. The particular strategies developed within the MARINER project demonstrate the enormous potential for agent-based control for IN in particular and telecommunications networks in general.

This article gives an overview of the work on agent-based IN load control strategies undertaken to date by the MARINER project. The next section provides a brief overview of IN and existing load control mechanisms. The third section introduces the MARINER framework for IN load control and is followed by a description of a multi-agent system that achieves load control using the Market-based Control paradigm. The fifth section describes trials of this agent system. The final section draws some conclusions and outlines ongoing work.

4.6.2 INTELLIGENT NETWORK LOAD CONTROL

The Intelligent Network architecture was developed as a means to introduce, control and manage services rapidly, cost effectively and in a manner not dependent on equipment/software from particular equipment vendors. The IN standards specify four network element types: Service Switching Points (SSPs), Service Control Points (SCPs), Service Data Points (SDPs) and Intelligent Peripherals (IPs). These elements typically communicate with each other via a Signalling System No.7 (SS.7) network. SSPs facilitate end user access to services by means of trigger points for detection of service access codes. SCPs form the core of the IN architecture – they receive service requests from SSPs and execute the relevant service logic. SCPs are assisted by SDPs, which store service/customer related data, and by IPs, which provide services for interaction with end-users (for example automated announcements). A much more complete description of the IN architecture can be found in [1].

Traditional approaches to IN load control are ‘node-based’ in nature, being centred on protection of the processors of individual SCPs. A commonly deployed mechanism is based on Automatic Call Gapping. In this approach an overload detection algorithm is located at the SCP and works in conjunction with a service request throttling mechanism located in the SSPs. Once overload is detected, a control message indicating the severity of the overload is sent to SSPs, via the SS.7 network, where an appropriately severe throttle is put in place to restrict the number of accepted IN service requests. The throttling mechanism operates by restricting the number of requests for particular service types that are accepted during set intervals.

While mechanisms such as Automatic Call Gapping serve to protect SCPs from overload in most situations we believe that such mechanisms cannot in general guarantee that desired quality of service levels are consistently achieved. The following observations support this viewpoint:
• Most currently deployed node-based mechanisms were designed for standard telephony traffic patterns. Present and future INs support a large number of heterogeneous services, each exhibiting differing traffic characteristics that cannot be effectively controlled using node-based techniques;
• Existing node-based overload protection mechanisms serve to protect individual nodes only and may cause the propagation of traffic congestion, resulting in adverse effects on the service completion rates of the network as a whole;
• Typical node-based mechanisms do not interact effectively with the protection mechanisms that are incorporated into the signalling networks that carry information between the nodes in a network. This lack of co-ordination means that often node-based mechanisms are at best ineffectual and can even serve to exasperate the overload situation;
• Node-based controls typically focus on SCP protection only, however a network operator may wish that network profit is maximised, or may wish to ensure that particular service types are given higher priority in times of overload;
• Telecommunications equipment manufacturers implement node-based mechanisms on a proprietary basis. This can lead to difficulties in effectively controlling traffic in an IN that contains a heterogeneous mix of equipment types.

These inherent drawbacks with node-based mechanisms point towards the need for network-based controls that deal with high offered load levels in a manner such that the entire network benefits. The operation of such controls may result in certain resources in the network operating at a level that is from the local perspective less than optimal, however a more global measure of network performance (such as number of successful sessions or generated revenue) will be maximised.

4.6.3 MARINER Agent Framework for IN Load Control

The agent framework for IN load control developed by the MARINER project is illustrated by Figure 1. As the Figure shows, the project has been addressing load control for both current IN implementations and more future-oriented implementations based on the use of CORBA distributed object technology. The network shown in the bottom half of Figure 1 shows three inter-connected networks, one containing traditional IN elements communicating via SS.7, the others containing IN functional elements implemented using CORBA, which use TCP/IP or SS.7 respectively for inter-node communication. Interworking between the SS.7 and CORBA based IN networks is provided by TC/CORBA gateways. More information on CORBA-based IN can be found in [2].

The top half of Figure 1 depicts the general framework for a FIPA-compliant multi-agent system to perform load control. Individual agents, the FIPA-standards Agent Communication Channel (ACC) and wrappers that allow agents to interact with the system software of the IN elements they are associated with, are shown. The four classes of agent identified as being required to carry out the tasks necessary to control IN load are:
Figure 1: Agent Framework for IN Load Control

- **Quantifier** agents, which monitor and predict the load/performance of SCP processors (and possibly other IN resources) and report this information to other agents;
- **Distributor** agents, which maintain an overview of the load and resource status in the entire network and may play a controlling or supervisory role in resource allocation;
- **Allocator** agents, which are associated with SSPs. They form a view of the load situation in the network and the possibility of resource overload, based on their own predictive algorithm(s) and information received from other agents. If they perceive that there is danger of overload of resources they throttle service requests on a priority basis;
- **Reporter** agents, which operate on a longer time-scale, employing more sophisticated algorithms (possibly based on techniques like pattern recognition) to interface with external network management systems or human users. A Reporter could, for example, provide recommendations regarding the introduction of additional SCP processing capacity in response to observation of recurring overload conditions. Multi-agent systems incorporating a mixture of Quantifier, Distributor, Allocator and Reporter agents could utilise a wide range of control algorithms to carry out the actions described above. Approaches typically associated with agent-based control systems include Market-based Control, Ant Colony Optimisation and Genetic Algorithms. The next section will describe an IN load control strategy based on the Market-based Control paradigm.

### 4.6.4 Multi-Agent System for Market-based IN Load Control

The load control strategy that we describe in this section is an application of *Computational Markets*, also known as *Market-based Control*, which is a distributed resource allocation
paradigm that builds on economic utility theory for electronic markets. Market-based Control has been successfully applied to a diverse range of problems, such as ATM bandwidth allocation [3] and power load management [4]; an overview of this expanding research area can be found in [5].

Computational Markets as applied to resource allocation problems are generally implementations of General Equilibrium Theory (developed in the field of microeconomics), whereby agents in the market set prices and create bids for resources, based on supply and demand functions. Once equilibrium has been computed from the bids of all the agents, the resources are allocated in accordance with the bids and the equilibrium prices. The search for the market equilibrium can be implemented so that the customer and/or producer submit bids to an auctioneer. From these bids, the auctioneer updates its information and requests new bids in an iterative fashion. Once the market equilibrium has been found, the reallocation of goods is performed in accordance with the bids and market prices.

In our strategy, load control is carried out by means of ‘tokens,’ which are ‘sold’ by the Quantifiers of ‘providers’ (SCPs) and ‘bought’ by the Allocators of ‘customers’ (SSPs). The amount of tokens sold on behalf of an SCP controls the load offered to it, and the amount of tokens bought on behalf of an SSP determines how many IN service requests it can accept. ‘Trading’ of tokens (in an ‘auction’) is carried out such that the ‘common good’ is maximised. We define common good as the maximisation of the profit generated for the network operator – note that here profit is an abstract concept, it need not be based solely on generated revenue but may also incorporate other factors such as customer perceived importance of a particular service.

All SSPs contain a number of pools of tokens, one for each SCP and service class pairing. Each time an SSP feeds an SCP with a service request one token is removed from the relevant pool. An empty pool indicates that the associated SCP cannot accept more requests of that type from the SSP. Tokens are periodically assigned to pools by a Distributor, which runs an auction algorithm to calculate token allocations. Auctions are centrally implemented by a single Distributor using ‘bids’ (received in the form of messages sent at set intervals interval) from all the Allocators and Quantifiers in the system.

Quantifier bids consist of the unclaimed processing capability for the coming interval and the processing requirements for each service class. In a similar manner, Allocator bids consist of the number of expected IN requests over the next interval for each service class. These values are simply set to the number that arrived in the previous interval – with small interval durations this will be a reasonably accurate estimate.

The objective of the auction process is to maximise expected network profit over the next interval. To do this, it maximises the increase in expected marginal utility, measured as marginal gain over cost, for every token issued. The expected marginal gain associated with allocating an additional token to a particular Allocator is defined as the profit associated with consuming it times the probability that it will be consumed over the auction interval. The expected marginal cost associated with issuing a token from a particular Quantifier is defined
as the ratio between the processing time consumed and the remaining processing time. Based on these values the Distributor implements a maximisation algorithm that is iterated to allocate all the available tokens. Tokens will typically be first allocated to Allocators with ‘higher’ bids (i.e. those which expect greater numbers of requests for service sessions that result in high profits) in preference to those with ‘lower’ bids.

The operation of the auction algorithm is illustrated for simple case where there is only one service class supported by the network by Figure 2 below. In step (1) Quantifiers and Allocators submit their bids to the Distributor, which runs the auction process – step (2). In the Figure dark circles represent tokens, whereas light circles represent token requests, the auction algorithm assigns tokens to token requests using the algorithm described above. Once the auction completes the values of token assignments are reported to the Allocators, which use them to admit service requests during the next time period.

Figure 2: Operation of Market-based Control Strategy

The net effect of the auction process is that tokens are allocated in a manner that will serve to balance the arriving traffic load across all SCPs, subject to maximising the overall network profit.

4.6.5 MARINER Project Trials
MARINER is performing trials of both an operational and simulation nature that test the operation of a FIPA-compliant agent system for the Market-based Control strategy described
above, and assess whether such a system would deliver acceptable performance when deployed in a real network environment. The trials have been developed in two phases, the first involving simulation of the operation of the load control strategy, the second involving the implementation of this strategy using real agents.

Simulation-based trials involved the use of event-driven simulation techniques to model the operation of a number of versions of the Market-based Control strategy and assess its performance and efficiency under a range of conditions. The use of a simulation-based approach allowed analysis of the strategy under a range of network configurations and traffic conditions that would have been impossible to emulate using a prototype agent system implementation.

In the second, more substantial, trial phase the Market-based Control strategy is being implemented using real agents that interact with a virtual multi-node IN platform. This platform is comprised of a number of CORBA-based SSP and SCP prototypes, one ‘real’ SSP realised by a programmable switch controlled by an SDL-based software system. The agents themselves are implemented using the Grasshopper platform and the April agent environment. The goal of this trial phase is to provide a qualitative study of the issues involved in using the FIPA specifications to achieve interoperability between agent systems and for achieving inter-agent communication. Through this work a number of areas where the FIPA specifications required clarification, modification or enhancement have been identified by the project and reported to FIPA. Specifically these relate to the semantics originally outlined for the FIPA Agent Communication Language (ACL) [6], agent naming [7] and potential enhancements to the FIPA specifications to support interoperability between real-time constrained agent systems [8].

Figure 3 presents load traces from trial experiments, which illustrate the operation of the Market-based Control agent system. The Figure shows two traces, one for the total load offered to a network containing eight SCPs, the other showing the mean load carried by these SCPs. The offered load was varied to emulate two separate focussed overloads occurring due to media-stimulated mass call-ins. In the scenario shown here eight of the thirty-two SSPs in the network receive a much higher than normal number of requests for a particular service type (for example a televoting service) over a sixty second duration. For the first overload, the increase in volume of service requests corresponds to an offered load of 95%, for the second it corresponds to an offered load of 200%.

As can be seen from the mean SCP load trace the agent system serves to ensure that in conditions where offered load exceeds a target load for SCPs of 90% the mean load never exceeds this threshold. In addition results not provided here show that the spread of loads between the SCPs does not exceed 2%. It is noted that the mean load trace in Figure 1 actually falls a couple of percent below the target value during overload; at the time of writing a number of simple mechanisms to allow SSPs to exchange unused tokens between auctions are being developed in order to bring the mean load closer to the target during overload.
4.6.6 **CONCLUSIONS AND ONGOING WORK**

This contribution has provided an overview of some of the work undertaken by the MARINER project in relation to the development and testing of agent-based strategies that can achieve, in an efficient and flexible manner, load balancing and avoidance of harmful overload in complex and dynamic network environments. As an example of the application of the agent-based approach a Market-based Control load control strategy for IN was presented, along with results showing how it performs in response to media-stimulated overloads.

While the strategy presented in this paper provides an effective and scalable solution to the problem of IN network-based load control, we believe that it can be used as a base for the development of more-advanced strategies that would provide a more convincing demonstration of the value of an agent-based approach. In particular it is hoped to incorporate some degree of learning behaviour into the agents, thus allowing them adapt to long term trends and changes in traffic patterns. The potential for human interaction with the agents is also being investigated. For example a human user could warn the agents of an impending call-in event that is likely to cause an overload, allowing them to put into place the necessary throttles required to protect the network and ensure quality of service for all its users.

In addition to work on making the agents more ‘intelligent’ the scope of the load control strategy is being widened to address the following three problem areas:

**Load Control for SDPs/IPs** In the strategy described above it was assumed that neither the SDPs nor the IPs required to support SCPs in the provision of IN services are a bottleneck in
the system, i.e. they will never overload. In real systems, however, it is possible that a service session may be blocked due to the overload of resources at either an SDP or an IP. To combat this potential problem it is possible to modify the auction algorithm to allocate all resources (SCP, SDP and IP) required for the successful completion of a service session. For example each time it decides to allocate resources for that service type to one of the SSPs in the system the Distributor can allocate tokens not only for SCP capacity, but also for each interaction required with an SDP/IP.

**Load Control for CORBA-based IN** The project is also addressing the development of load control strategies specifically targeted towards CORBA-based IN environments. In a CORBA-based implementation the IN functional elements and service logic are realised by a collection of CORBA objects that may be distributed over a number of physical nodes. Clearly different distributions of these objects will result in better overall performance under different traffic conditions. Means by which agents can detect changes in traffic patterns and where appropriate re-distribute objects are currently being investigated. This enhancement adds a new dimension to the load control strategy, by offering the potential to use available resources in a more optimal manner during periods of overload.

**Load Control for Multi-Operator Networks** As the telecommunications market opens further, the scenario involving one operator buying access to the resources of another operator in order to provide its services is becoming increasingly likely. For example, in periods of high loading of its SCPs an operator may buy a portion of excess SCP capacity currently available in the network of another operator. In the Market-based Control strategy this can be achieved through communication between the two Distributors reside in the two operator domains: when a Distributor detects that it needs extra SCPs capacity it can send a bid to another Distributor in order to buy any excess capacity that may be available in that operator's network. In addition it would be straightforward to introduce a pricing mechanism that would mirror the actual contract between the two operators in order to control this transaction.

### References
