Agent-Enhanced Dynamic Service Level Agreement in Future Network Environments

David Chieng[1], Alan Marshall[1], Ivan Ho[2], and Gerard Parr[2]

[1] Advanced Telecommunication Systems Laboratory,
School of Electrical and Electronic Engineering, Ashby Bld, Stranmillis Road,
The Queen's University of Belfast, BT9 5AH Belfast, Northern Ireland, UK
{d.chieng, a.marshall}@ee.qub.ac.uk
[2] Telecommunication and Distributed Systems Group,
School of Information and Software Engineering,
University of Ulster at Coleraine, BT52 1SA Coleraine, Northern Ireland, UK
{wk.Ho, gp.Parr}@Ulst.ac.uk

Abstract. Current network infrastructures are experiencing rapid transformation from providing mere connectivity, to a wider range of flexible network services with Quality of Service (QoS). We propose an agent-enhanced system that facilitates dynamic Service Level Agreement (SLA) activities, such as end-to-end QoS specifications and service price negotiations in such an environment. A prototype system consisting of real-time Java-based agents that interacts with a simulated network was developed to demonstrate scenarios and enable analysis. The studies show that this form of dynamic SLA negotiation introduces many innovative ways on how network services can be provisioned. This paper also highlights the effects of implementing dynamic connection bandwidth pricing on traffic load and network provider's revenues.

1   Introduction

After years of rapid technical and standardization efforts, various Quality of Service (QoS) architectures such as Differentiated Services (DiffServ) [1], Multi Protocol Label Switching (MPLS) [2] and Resource Reservation Protocol (RSVP) [3] begin to take a foothold in today's networking environments. Over the next few years we are going to witness rapid transformation in current network infrastructures from providing mere connectivity to a wider range of tangible and flexible network services with QoS. Faced with increasingly complex network, to provision, manage and bill services are becoming some of the biggest challenges for the service, network and content providers today.

Issues regarding Service Level Agreement (SLA) arise due to the need to maximize customer satisfaction and service reliability. According to [4], many end users and providers are still unable to specify Service Level Agreements (SLAs) in a way that benefits both parties. Very often, the service or network providers would overprovision their services that leads to degradation or on the contrary, failed to provide serv-

ices to the best of their systems or networks capabilities. Hence, the abilities to deci-pher what the end users want from what they asked for and to response effectively are critical as the competitiveness of future providers does not only rely on the diversity of the services they can offer, but also the ability to meet customers' requirements. As users are more discerned about QoS, it is believed that value-added services and appli-cations are best to be delivered and billed on per session or per-transaction basis [5]. A well-defined pricing scheme is not only important as a tool to enable users to match and negotiate services as a function of their requirements, it can also be a mechanism itself, as the dynamic setting of prices can be used to control the volume of traffic carried [6]. In this paper we introduce an agent-enhanced system that facilitates dy-namic SLA specifications between end users and various providers in order to setup end-to-end connections or Virtual Leased Lines (VLLs) with preferred QoS and prices. Here, agents are employed as autonomous negotiators on behalf of various parties involved.

The rest of the paper is organized as follows: Section 2 summarises some related work. Section 3 gives some background on the definitions of SLA and the metrics used in our work. Section 4 presents some advantages of employing agents as the SLA mediator. This is followed by the description of our agent-enhanced service brokering architecture in Section 5. Section 6 illustrates the components within the agent envi-ronment and their interactions. Since bandwidth is a shared finite resource, a fair dis-tribution policy such as pay as much as you used is proposed. This is highlighted in section 7 where we demonstrated how dynamic bandwidth pricing strategy can affect the overall network utilization as well as the generated revenues if users are putting service quality and cheapest price as their priorities.

2   Related Work

Our current work is motivated by a number of research. First, the researchers in [7] proposed the implementation of bandwidth brokers to support SLA trading which includes resource allocation, path selection and pricing between DiffServ networks. The authors in [8] introduced Virtual Network Service (VNS) that uses a virtualisation technique to customize and support individual VPNs' QoS level. The authors in [9] proposed a QoS management architecture that employs distributed agents to establish and maintain the QoS requirements for various multimedia applications. Here, QoS specification is categorized into 2 main abstraction levels: application and system. The capability of current RSVP signalling is extended in [10] where users are allowed to reserve bandwidth or connections in advance so that blocking probability can de re-duced. Authors in [11] offered a similar idea but use reservation agents to help clients to reserve end-to-end network resource. The Resource Negotiation and Pricing Proto-col (RNAP) proposed by [12] enables network service providers and users to negotiate service availability, price quotation and charging information on per application basis. Researchers in [13] implemented adaptive bandwidth pricing and bidding at the lower level of granularity such as path or link via multi-agent systems (MAS).

3   SLA Metrics

SLA provides a means of quantifying service definitions. In networking environment, it specifies what an end user wants and what a provider is committing to provide such as the definitions for QoS, performance levels, etc [14]. The definitions of SLA vary at business, application or network level. Business level SLA involves the issues such as pricing schemes and contract. Application level SLA concerns the issues of server availability (e.g., 99% during normal hours and 97% during other hours). Network level SLA or often referred as Service Level Specification (SLS) involves lower layer parameters such as throughput, latency, packet loss and jitter. In our work we first focus on the basic SLA metrics involved in establishing Virtual Leased Line (VLL) on demand type of services which are described in table 1 as follows:

Table 1. SLA metrics and descriptions

SLA metrics	Description
Guaranteed BW (b_i) *for request i	The amount of guaranteed/reserved bandwidth allocated to VLL_i. We only consider this metric at present due the ease of its configuration. It is also the single most important factor (not always the case) that affects other lower level QoS parameters such as delay, jitter, etc. Guaranteed BW can be quantified in units of 1kb, 10kb, etc.
Start Time (Ts_i)	This is applicable to scheduled services such as VoD, video conferencing, news broadcast, MTV, etc. For instant access, this metric is simply assigned as the current time.
Session Length (T_i)	The duration required for this VLL. This is applicable to Video on Demand (VoD) or news broadcast type of services where the session time is known in prior. However the session time can be extended automatically depending on the availability, policy, etc.
Price (P_i)	This can be the maximum price a user is willing to pay for this service.
Option (Ω_i)	It consists preferences and priorities in the form of rules. This is useful at times when not all the requested SLA metrics can be granted. User can specify which parameter is the priority and which is tolerable.

* Other hardware and software specifications such as application types, IP/MAC addresses, port numbers and other resources though not explicitly mentioned, are assumed to be taken care of by lower level agents.

From the metrics described above a SLA request i can then be represented by:

$$Request_i(b_i, Ts_i, T_i, P_i, \Omega_i)$$

4   Why Agent-Mediated SLAs?

Software agents offer many advantages in this kind of environment. Agents can assist the end users, the service and network providers, to perform expertise brokering tasks. These include service selection, QoS specification, pricing negotiations, etc. Agents are particularly suited for the tasks where fast decision making is critical. These satisfy the two most important aspects of performance in an SLA; availability and responsiveness [14]. The following attributes highlight the capabilities of agents in carrying out brokering tasks [15].

- **Autonomy**. Agents can be either responsive or proactive. It is able to carry out tasks autonomously under pre-defined rules or constraints. The level of their intelligence depends on the given roles or tasks.

- **Communication**. With this ability, negotiations can be established between two agents. Agent Communication Language (ACL) from Foundation for Intelligent Physical Agent (FIPA) has been widely adopted as the de-facto language.

- **Cooperation**. Agents can cooperate to achieve a single task. Hence, agents representing end users, service providers and network providers are able to cooperate to setup an end-to-end VLL that spans across multiple domains.

- **Mobility**. Java-based agents can migrate across heterogeneous networks and platforms. This attribute differentiates mobile agents from the other forms of static agents. Mobile agents can migrate their executions and computation processes to remote hosts. This saves shared and local resources such as bandwidth and CPU usage compared to conventional client/server systems. Thus, intensive SLA negotiation processes can be migrated to the service provider or network provider's domain.

ObjectSpace™ Voyager ORB 3.0 [16] is used to implement of our agent system. Voyager Agent API offers the ability to construct remote objects in the remote host and a set of control mechanisms that offer more flexible instructions on how the agent should terminate itself. There are two types of communication mechanisms, namely Method Calling and ObjectSpace™ by which Voyager agents interact with each other. The former mechanism enables an agent to call methods of another agent. This is provided if the calling agent knows a-priori the method interface of the called agent. The latter mechanism enables voyager agents to multicast an event message to other agents for example advertising for new services or offers.

5   Agent-Enhanced Service Brokering Architecture

In this architecture, the network provider has the control over all the access points within its managed domain. To setup a VLL, an end-to-end path must be configured in advanced before the actual traffic can be admitted. Hence, the network provider must

maintain a global view of the network resources which can be realized using Internet Gateway Protocol (IGP) link state algorithm [17]. This is contrary to RSVP [3] peer-to-peer approach, where network nodes such as routers or switches can decide locally whether to accept or reject resource reservation. Distributed admission control complicates the service management task and especially when comes to billing. Figure 1 illustrates the open architecture:
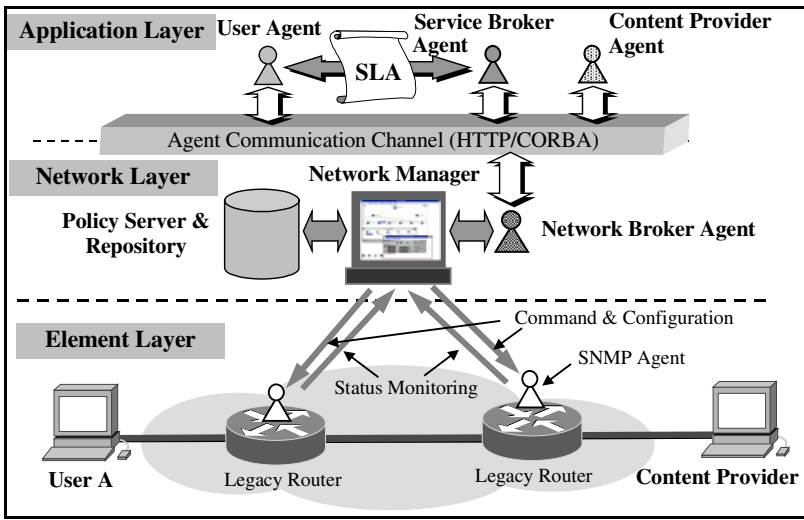

Fig. 1. Agent-Based Service Brokering Architecture [18]

Here, agents act as autonomous negotiators on the application layer. Conventional HTTP or CORBA can be utilized as the communication channel for agents. The role of an User Agent (UA) is to automate the service request procedures on the behalf of the end user. The Content Provider Agent (CPA) provides information or advertising facilities for it services such as news broadcast, Video on Demand (VoD), etc. The Network Provider Agent (NPA) acts as an access broker for its domain. Logically it is a component of the network manager but physically it may reside at the domain access router. The Network Manager has the direct access to a policy server which administrates policies rules and actions for different services' SLAs. The policy repository stores defined policies as sets of rules for this domain. It may be a single physical site or replicated at several places in the form of databases, files, an administrative server or a directory server. Currently, Light Lightweight Directory Access Protocol (LDAP) directory is favoured by most vendors [19]. The Policy manager within the policy server validates policies in the policy repository so that it is mutually consistent in the network. The Network manager can also be an agent manager that assigns NPAs to serve the incoming agents. The Service Provider Agents (SPA) acts as a mediator between multiple parties involved. An UA first asks SPA for a service with various QoS preferences. The SPA then negotiates with respective CPA and NPA before setting up a VLL. The NPA then makes decisions based on the requested SLA and

propagates the required configurations down to the element layer. Conventional SNMP agents can be employed at the element layer to configure queues and monitor links and flows' states. The multi-party relationship model is illustrated in figure 2. If the route spans across multiple domains, the SPA may need to negotiate with different network providers and content providers in order to set up a VLL.

6   Agent Environment and Multi-party Interactions

A prototype multi-operator network model has been built using the Block Oriented Network Simulator (BONeS) [20] as a testbed for our proposed architecture. This allows functional and dynamic behaviour of the network under various agent-supported scenarios to be investigated. Figure 2 illustrates the interactions between various agent components. In our framework we have real-time agents running in LAN and virtual networks in simulation environment. Our agents use BONeS' Inter Process Communication (IPC) protocol to interface with the Sun Solaris-based Network Simulator via TCP/IP socket.
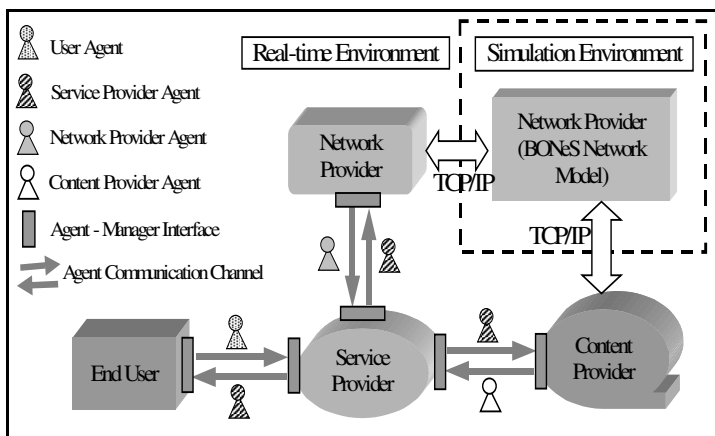

Fig. 2. Agent Components Interaction [15].

There are four main parties involved this scenario: the End User, Service Provider, Content Provider, Network Access Broker and the Network Provider. Each component consists of its own database, agent(s) and a manager. The manager's job is to provide service for any arriving agents, handle data transactions, storage retrieval, agent creation and task assignment. Figure 2 shows that when requesting a service, an end user can launch its agent to the central meeting point (SP) to interact with the local object (SP Manager). The manager then asks its server to create an agent with a task list. These agents can request for content listing from the CPA and then setting up a VLL with the NPA according to user SLA specification. At this stage, our agent security is based on 'trusted hierarchy' scheme where agents only communicate with trusted

remote hosts. However, it is anticipated that future development of the agent system will need to implement a much more rigorous security policy [15].

7   Demo – Agent Mediated SLA and Dynamic Bandwidth Pricing Scenario/Game

A demonstration on the prototype system was carried out during the technical visit session at Fujitsu Telecommunications Europe Ltd. headquarter in Birmingham in conjunction with World Telecommunication Congress/Integrated Signal Symposium (WTC/ISS2000). In each session, three volunteers were invited from the audience to assume the role of the future network operators. Their common goal was basically to maximize their network revenue by dynamically price the bandwidth under agent brokering environment. The scenario presents three individual networks owned by different operators; each is identical in terms of the number of devices, topology and available resources. Each network consists of nodes connected in a mesh topology. As a means of creating competition, all three networks offer the same access to a remote content provider that provides multimedia services as shown in figure 3 as follows:
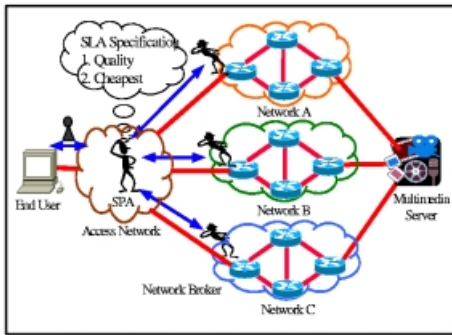

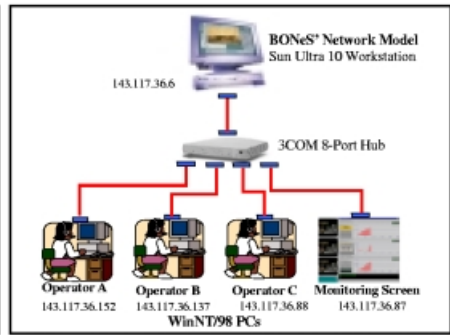**Fig. 3.**   Agent Brokering Scenario   **Fig. 4.**   System Set-up

A simple LAN was set-up for this demo as shown in figure 4. The multi-operator network model coupled with internal agent brokering mechanisms was run on the Sun Ultra 10 workstation. A few PCs were set-up for the competing network operators and a separate monitoring screen was provided (Fig. 6).

In this game we considered some universal customer preferences such as best QoS (guaranteed BW) and cheapest price. We assumed most users would want QoS as their first priority and the cheapest offering price as the second priority. Hence the options for User SLA request are:

Ω_i (b_i, Ts_i, T_i = NOT negotiable AND First Priority, P_i = Cheapest AND Second Priority)

If the cheapest network provider could not provide the required guaranteed VLL BW, the service provider would opt for the second cheapest one. During the game, SPA continually requested NPAs to setup VLLs to the multimedia server. In order to show the effects of dynamic pricing, we allowed the invited audience/acting network operator to manually change the BW price. For this demo we did not provide different pricing schemes for different user classes. The table below describes various billing parameters for the network operators in this demonstrator.

Table 2. Billing Parameters Description

Parameters	Description
BW selling price, P_i	The selling price for a BW Unit per minute for VLL i.
Cost Price, θ_i	The cost price for a BW Unit per minute for VLL i that changes according to link's reservation load status. This loosely represents the management overhead cost.
Operation Cost, σ	This represents the overall maintenance cost, hardware cost, labour cost etc per minute.
Guaranteed BW, b_i	This is the size of Guaranteed unit bandwidth allocated to VLL i.
No. of links, ℓ_i	The number of links used by VLL i. Here all the possible paths are pre-computed. Since no. of links is considered in the charging equation, the shortest available path is therefore preferred.
Session, T_i	The session length in minute subscribed by VLL i.

*Note: In this game, price and bandwidth were simply given in term of units. Some attributes are specifically created for this game.

In this game, users were charged at the end of their sessions. The calculation for gross revenue earned by a network operator from each VLL i is based on the following equations:

$$\text{Rev}_i = (P_i - \theta_i) \cdot b_i \cdot \ell_i \cdot T_i \tag{1}$$

Therefore the total gross revenue, $\text{Rev}_{\text{gross}}$ after period t hence:

$$\text{Rev}_{\text{gross}}(t) = \sum_{i=1}^{n(t)} \text{Rev}_i \tag{2}$$

Where
  t = simulation time elapsed in minute.
  $n(t)$ = total number of VLLs sold after t.

Total net revenue Rev_{net} after t hence:

$$\text{Rev}_{\text{net}}(t) = \sum_{i=1}^{n(t)} \text{Rev}_i - \sigma.t \tag{3}$$

Where

σ = operating cost per minute.

Each player or acting network operator could monitor his/her competitors' offered BW prices and set their own price at the console shown in figure 5. Network links' reservation load, cost and network topology showing current network utilisation were also displayed. Users and network blocking statistics were also reported. For this game, the operators' revenues were solely generated from VLL sales. The monitoring window (figure 6) displayed the total revenues (profit/lost) generated by each network operator. We associated the link QoS level in terms of Gold, Silver and Bronze by referring to link's reservation load of 0-50%, 50-75% and 75-100% respectively. This is different from per-user's QoS as each user's VLL was already assigned an amount of guaranteed BW. Therefore link reservation load is the aggregation of all the independent VLLs' guaranteed BW.
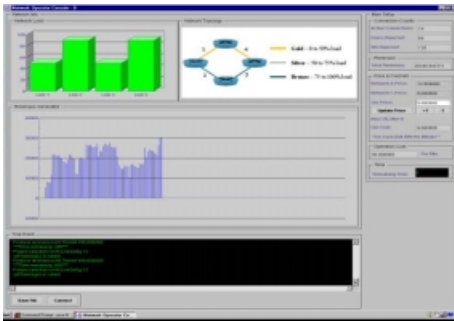

Fig. 5. Network Operator Console   **Fig. 6.** Monitoring Window

The users' service request arrival rates were generated according to different Poisson arrival distributions. Table 3 shows the characteristics set for the three basic classes of VLL subscribers.

Table 3. User Characteristics

User Classes	Mean Request Arrival Rate (per hour)	Mean b (Guaranteed BW unit)	T (Session in mins)	Example Applications
1	70	2	3-10	VOIP/Live Music
2	15	30	10-60	VoD/Conferencing
3	28	20	1-10	File Transfer

Figure 7 shows the accumulated bandwidth request (offered load) profile with this specification. The profile shows that Class 1 users (e.g. audio) requested for relatively

low BW but arrived very frequently. Class 2 users (e.g. VoD) required high bandwidth VLLs over long periods of time and they were the dominant traffic contributors. Class 3 users requests produced bursty type of traffic profiles that suited applications such as FTP.
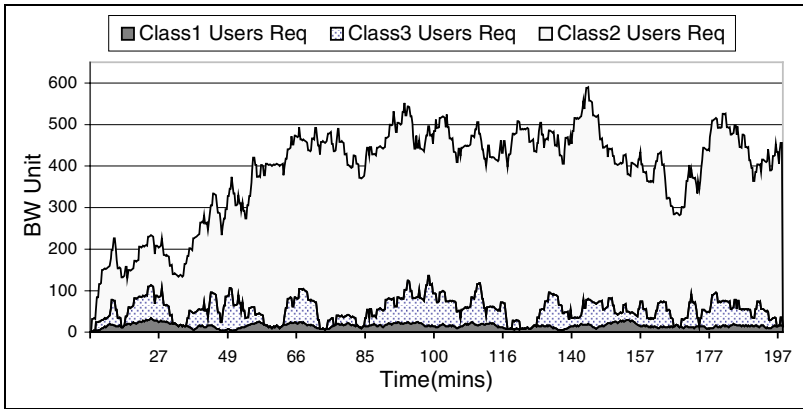

Fig. 7. Multimedia Users Bandwidth Request Profile (Accumulated Bandwidth)

8   Result of Dynamic Pricing

The results from one of the sessions were collected and analyzed in figures 8 through figure 11. Figure 8 shows the pricing history of the three acting network operators.
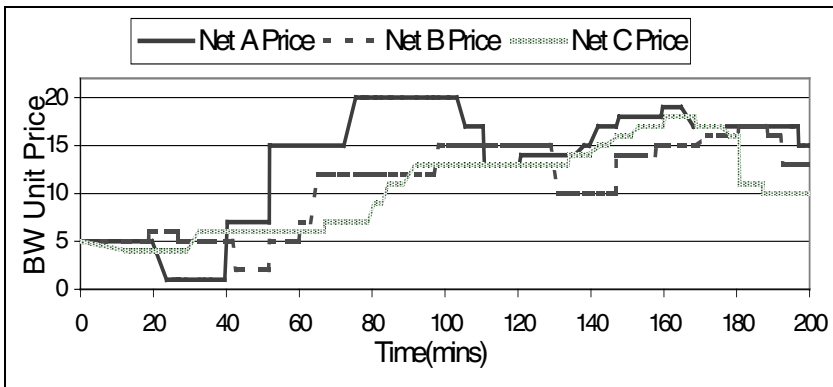
Fig. 8. Bandwidth Price bidding vs. Time (minutes)

Here, the three network operators were trying to maximise the revenues by setting different BW prices dynamically. At t>20mins, network A lowered its bandwidth

price to 1 and caused a sharp increase in load over the measured link (see figure 9). At t>50mins, network A increased its price dramatically, and soon became much more expensive than the others. As a result, a significant drop in traffic was observed after t>75mins. This was most likely due to class 2 subscribers leaving the network. At another time instant, t>110mins when network B's price remained constant, network A beat network B in price and attracted traffics to its network.
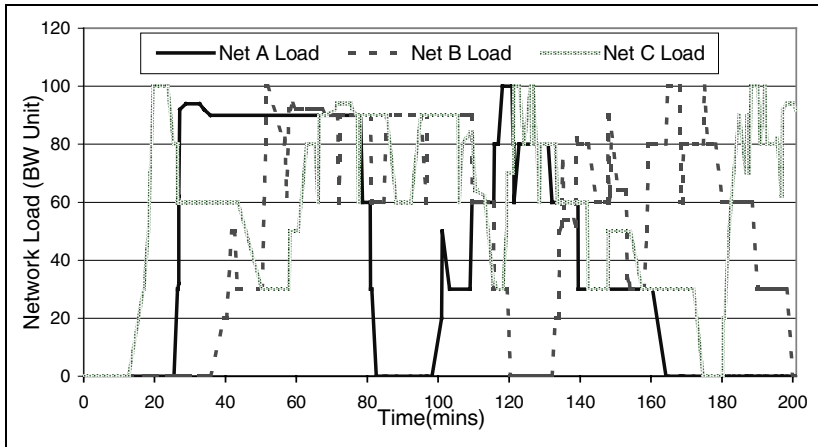

Fig. 9. Network load measured at one of the links(link2) vs. Time

Notice that at time t>100mins, when network A was still the most costly network, traffic was coming into the network because the other two networks were saturated and were unable to provide the required bandwidth. This earned network A a sharp rise in revenue (see figure 10) and a short lead in revenue race as a result of users buying high cost connections at bulk volume. In figure 11, we can observe a close relationship between load and price. In this case, it seemed that the cheapest provider earned the most revenues. However if we look at figure 10, network B was just marginally lost to network A. This means network B can actually bid a higher average price and win the game because network C had a significantly higher average bandwidth price compared to network B. Nevertheless, this strategy is only applicable for this scenario.

Figure 12 shows the importance of setting the right price at the right time. Network A made a loss at 20-40 minutes interval due to low offer price. However much of the loss was compensated at 100-120 interval due to bulk bandwidth sales at high price. On the whole, network C managed to maintain a good stream of revenue generated throughout the session. It is also observed from this simple experiment that it is more profitable to get revenue from high BW, long session stream such as video conferencing.
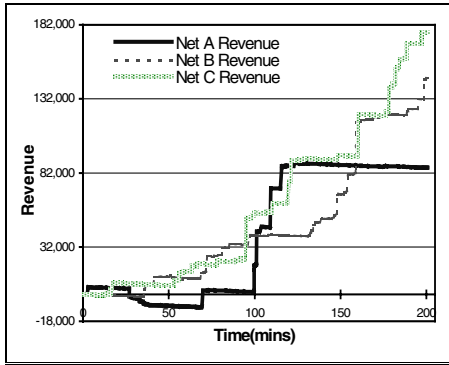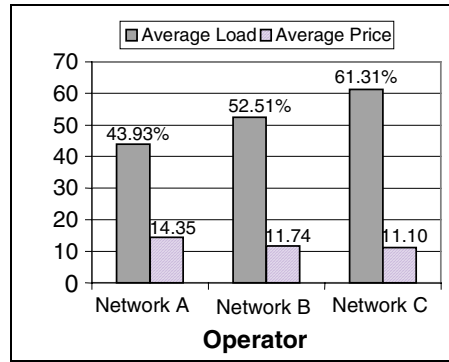
Fig. 10. Revenue Generated vs. Time

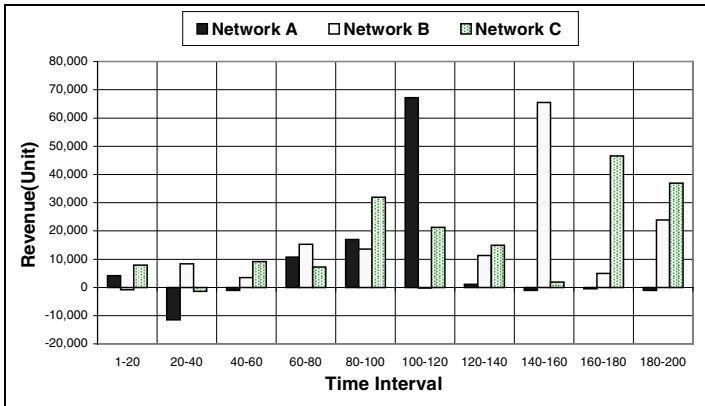**Fig. 11.** Average Load Vs. Price

Fig. 12.   Revenue Generated Per Time Interval vs. Time

9   Conclusions and Future Work

In this paper we demonstrated a futuristic scenario where agents can be employed to support dynamic SLA specification for network services. Various aspects of SLA especially regarding the guaranteed BW and pricing were investigated. A demonstrator consisting of real-time agents and simulated networks was built to support our analysis. The studies show that dynamic SLA negotiation introduces many innovative ways on how the future network services can be provisioned and priced. At this stage, our agents only exercise the simplest form of negotiation such as resource query and pricing comparison. Hence, BW consumption during the negotiation process is negligible whether client-server based agents or mobile agents were used. Nevertheless it is

believed that when agents have acquired a higher level of negotiation capabilities and intelligence, this issue must be further addressed.

During the demonstration, we allowed the audience to manually set the BW price. In the future, operator agent can potentially take over this task where it can set the right price at the right time based on a more sophisticated pricing model or mechanism i.e. different pricing schemes for different service classes. Advance reservation option and various tolerance parameters can be incorporated as part of the SLA metrics. Links segregation can be also implemented to facilitate lower granularity BW resource control such as preventing high BW services e.g. VoD or FTP flows to starve all the bandwidth.

References

1. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. "An Architecture for Differentiated Services", RFC 2475, Network Working Group, IETF, December 1998. (www.ieft.org)
2. E. C. Rosen, A. Viswanathan and R. Callon. "Multiprotocol Label Switching Architecture", Internet Draft, Network Working Group, IETF, July 2000. (www.ieft.org)
3. R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, Network Working Group, IETF, September 1997. (www.ietf.org)
4. Fred Engel, Executive Vice President and CTO of Concord Communications. "Grasping the ASP means service level shakedown", Communications News, Aug 2000, pp. 19-20.
5. Roland Courtney, founder of ViewGate Networks, "IP QoS: Tracking the different levels", Telecommunications Magazine International Edition, January 2001, pp. 58-60. (www.telecommagazine.com)
6. Roch Guerin et.al., "Quality-of-Service in the Internet", Guest Editorial, IEEE Journal on Selected Areas in Communications, December 2000, Vo. 18, No. 12, pp 2485-2487.
7. G. Fankhauser, D. Schweikert, and B. Plattner, "Service Level Agreement Trading for the Differentiated Services Architecture", Swiss Federal Institute of Technology, Computer Engineering and Networks Lab, Technical Report No. 59. November 1999.
8. L.K. Lim, J.Gao, T.S. Eugene Ng, P. Chandra, "Customizable Virtual Private Network Service with QoS", Computer Networks Journal, Elsevier Science, Special Issue on "Overlay Networks", to appear in 2001.
9. N. Agoulmine, F. Nait-Abdesselam and A. Serrhrouchni, "QoS Management of Multimedia Services Based On Active Agent Architecture", Special Issue: QoS Management in Wired & Wireless Multimedia Communication Networks, ICON, Baltzer Science Publishers, Vol 2/2-4, ISSN 1385 9501, Jan 2000.

10. M. Karsen, N. Beries, L. Wolf, and R. Steinmetz, "A Policy-Based Service Specification for Resource Reservation in Advance", Proceedings of the International Conference on Computer Communicatons (ICCC'99), September 1999, pp. 82-88.

11. O. Schelen and S. Pink, "Resource sharing in advance reservation agents", Journal of High Speed Networks: Special issue on Multimedia Networking, vol 7, no. 3-4, pp. 213-28, 1998.

12. X. Wang, , H. Schulzrinne, "RNAP: A Resource Negotiation and Pricing Protocol", Proc. International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'99), New Jersey, Jun. 1999.

13. M.A.Gibney, N.R.Jennings, N.J. Vriend and J.M.Griffiths, "Market-Based Call Routing in Telecommunications Networks using Adaptive Pricing and Real Bidding", IATA'99, Stockholm, Sweden, 1999.

14. Dinesh Verma, "Supporting Service Level Agreements on IP Networks", Macmillan Technical Publishing, ISBN: 1-57870-146-5.

15. David Chieng, Alan Marshall, Ivan Ho, Gerald Parr, "A Mobile Agent Brokering Environment for The Future Open Network Marketplace", Seventh International Conference On Intelligence in Services and Networks (IS&N2000), Athen, 23-25 February 2000, pp 3-15. (Springer-Verlag LNCS Series, Vol. 1774, ISBN 3-540-67152-8).

16. www.objectspace.com/products/voyager

17. William A. Shay, "Understanding Data Communications & Networks" 2nd Ed, International Thomson Publishing Inc., ISBN 0-534-95054-X.

18. A. Marshall, S. A. Hussain, D. Chieng, and Q. Gu, "Dynamic Network Adaptation Techniques In An Open Network Environment", International Conference on IT and Communications at the dawn of the New Millennium (AIT2000), August 2000, Bangkok, Thailand

19. R. Rajan, D.Verma, S.Kamat, E.Felstaine, S.Herzog, "A Policy Framework for Integrated and Differentiated Service in the Internet", Technology Articles, Allot Communications Inc., Aug 15, 2000. (www.allot.com/technology/PBN.htm)

20. BONeS DESIGNERâ Ver 4.01, Alta GroupTM of Cadence Design Systems, Inc.