Performance Evaluation of a SLA Negotiation Control Protocol for Grid Networks

Igor Cergol[1], Vinod Mirchandani[1], and Dominique Verchere[2]

[1] Faculty of Information Technology
The University of Technology, Sydney, Australia
{icergol,vinodm}@it.uts.edu.au
[2] Bell Labs, Alcatel-Lucent
Route de Villejust, 91620 Nozay, France
Dominique.Verchere@alcatel-lucent.com

Abstract. A framework for an autonomous negotiation control protocol for service delivery is crucial to enable the support of heterogeneous service level agreements (SLAs) that will exist in distributed environments. We have first given a gist of our augmented service negotiation protocol to support distinct service elements. The augmentations also encompass related composition of the services and negotiation with several service providers simultaneously. All the incorporated augmentations will enable to consolidate the service negotiation operations for telecom networks, which are evolving towards Grid networks. Furthermore, our autonomous negotiation protocol is based on a distributed multi-agent framework to create an open market for Grid services. Second, we have concisely presented key simulation results of our work in progress. The results exhibit the usefulness of our negotiation protocol for realistic scenarios that involves different background traffic loading, message sizes and traffic flow asymmetry between background and negotiation traffics.

Keywords: Grid networks, Simulation, Performance Evaluation, Negotiation protocols.

1 Introduction

Network forums like IETF have given rise to several different negotiation protocols such as COPS-SLS [1], DSNP [2], SrNP QoS-NSIS and RNAP [3]. Apart from these protocols, FIPA Contract Net [4] is a generic interaction protocol popular within the multi-agent community. The negotiation protocols enables the service providers to dynamically tune the network related parameters as well as take into account the level of pricing that can actually be afforded by the consumers.

From the afore-mentioned negotiation protocol proposals published, we have identified scope for improvement in the individual mechanisms by means of consolidation; addressing the key areas of efficiency, reliability, flexibility and completeness.

First we discuss our protocol for autonomous dynamic negotiation within a distributed open market environment. The protocol can operate on any underlying network

P. Primet et al. (Eds.): GridNets 2008, LNICST 2, pp. 81 – 88, 2009.

switching technologies ie L1 services (e.g. fiber or TDM), L2 services (e.g. Carrier Grade Ethernet) or L3 services (e.g. IP/MPLS) . Then we present and discuss OMNeT++ based models and related stochastic simulation results that reflect the impacts on the performance of our negotiation protocol from different background traffic loads, message sizes and traffic flow asymmetry.

This paper is organized as follows: Section 2 gives an outline of the distributed Grid market framework in open environments. It also gives an essence of our negotiation protocol proposed in [5] first for single service element between two parties and then for a bundle of heterogeneous service elements from multiple service providers that can complement or compete for the service elements offers. Performance evaluation results are presented and discussed also in Section 2. The conclusions that can be drawn from this paper are stated in Section 3.

2 Negotiation Framework Overview

The framework for managing **Q**uality of experience **D**elivery **In** **N**ew generation telecommunication networks with **E**-negotiation (QDINE) [6] is shown in Figure 1.

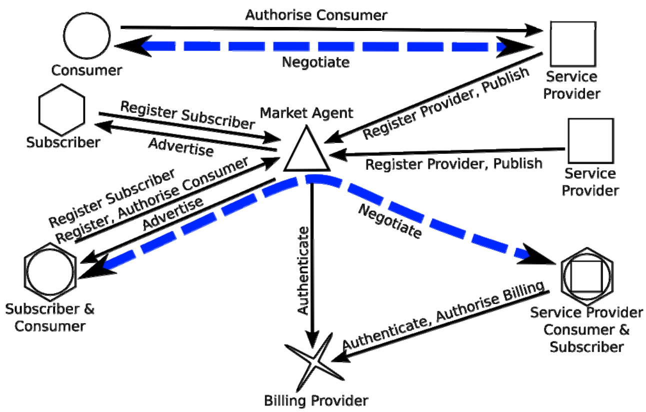

Fig. 1. The QDINE Negotiation Framework

The QDINE approach adopts a distributed, open market approach to service management. It makes services available via a common repository with an accompanying list of negotiation models that are used to distribute service level agreements (SLAs) for the services. A QDINE market is a multiagent system with each agent adopting one or more market roles. Every QDINE market has exactly one agent acting in the market agent (MA) role and multiple service providers (SPs), consumers and billing providers.

2.1 Proposed Service Negotiation Protocol

Our negotiation protocol [5] is session-oriented and client-initiated protocol that can be used for intra as well as inter-domain negotiation in a Grid environment. A key distinguishing feature of our protocol is that it enables the negotiation of a single service as well as the composition of several service elements from multiple service providers (SPs) within one session. There are two possible scenarios as shown in Figs. 2 & 3 that we consider for the negotiation of services between the consumer agent (CA) and SP.

2.1.1 Negotiation of a Single Service

The sequence diagram for the negotiation of a single service is explained with the help of Fig. 2 when there is a direct negotiation between the service provider (SP) and consumer agent (CA). The numbers in the parenthesis that precede the messages are used to describe the order of the interactions. Upon receiving the Request message (1), the SP has several options:

- It can immediately provide the service by sending an Agree message (2a) back to the CA. This forms the beginning of the SLA binding process of the service element.
- If the SP cannot deliver the service requested, it chooses to end the negotiation, it may reply with a Refuse message (2b), stating the reason for refusal.

The SP may choose to continue the negotiation sequence by offering an alternative of the SLA parameters proposal, in which case it responds with a Propose message (2c), containing the modified SLA and any additional constraints for an acceptable contract agreement.

The consumer may reply to the new proposal with another Request message (3), or accept the proposal (5b). Several Request and Propose messages (3) and (4) may be exchanged between the CA and the SP.

If the consumer chooses to end the negotiation, it can reply with a Reject-proposal message (5a). Alternatively, if the consumer accepts the proposal, it responds with an Accept-proposal message (5b) indicating the accepted SLA. After receiving the Accept-proposal message from the CA the SP can bind the SLA by sending an Acknowledge message (6). The SLA under negotiation can be validated at any time by participants, using the appropriate service specification(s) and current constraints. Invalid proposals may result in a Reject-proposal message (5a) or Refuse message (2b) sent by the consumer or provider, respectively.

2.1.2 Negotiation of Multiple Service Elements

We illustrate the protocol sequence diagram in Figure 3 for a multi-service negotiation using the following scenario: A QDINE market consumer needs to perform a highly demanding computational fluid dynamics (CFD) simulation workflow on a specific computer infrastructure and then to store the processed data results in other remote locations. The consumer therefore needs to negotiate the bundled service from several service providers (SPs): (i) a high performance computing (HPC) SP to execute the workflow of the simulation program hosted on the machine, (ii) a storage SP to store the results produced by the computational application, and (iii) a network (NW) SP to provide the required bandwidth for transmitting the data between the different resource centers within a time interval window that is able to avoid internal delays of the execution of global application workflow. In the configuration phase the consumer sends a

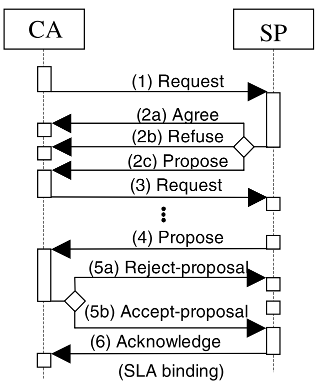

Fig. 2. Direct negotiation of a single service

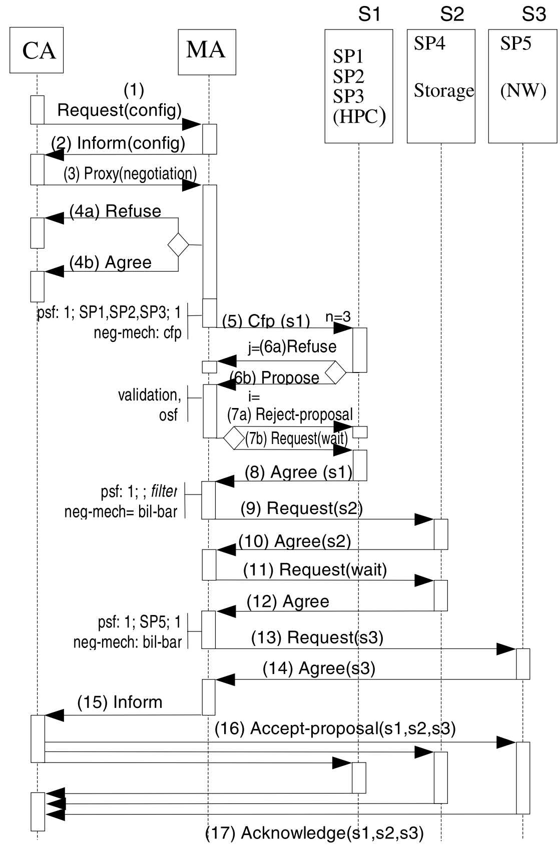

Fig. 3. Negotiation of multiple services

Request message (1) to the MA, to get the necessary information to enable the consumer agent (CA) to fill the proxy message in a template defined by the MA.

The MA responds to the request with an Inform message (2), carrying any updated configuration information since consumer registration.

In this example scenario, after obtaining the configuration information, the CA sends a Proxy message (3) to the MA, requesting it to select the appropriate SPs and negotiate the three afore-mentioned services. All of these services are required for a successful negotiation in this example:

- a call-for-proposal (cfp) for a service S1 (HPC) with three preferred providers (SP1, SP2 and SP3), returning the best offer.
- a bilateral bargaining with any appropriate providers for service S2 (storage), returning the best offer, and
- a bilateral bargaining with service provider SP5 for the network service S3 (constraints expressed with bandwidth), returning the final offer.

(The SPs for common services are shown collocated in Figure 3. due to space limitations.) Upon receiving the Proxy message, the MA assesses the proxy message and if it validates the request it replies with an Agree message (4b) to CA.

According to the request for service element S1, the MA issues a cfp (5) to service providers SP1, SP2 and SP3. One of them refuses to provide the service, sending a Refuse message (6a), the other two HPC service providers submit each a proposal, using a Propose message (6b). The MA validates the proposals and uses the offer selection function (osf) to pick the most appropriate service offer. The MA requests the chosen SP to *wait* (7b), as it doesn't want to bind the agreement for service element S1 until it has collected acceptable proposals for all three service elements.

According to the "any appropriate" provider selection function submitted by the CA for service S2, the MA then selects the only appropriate SP (SP4) for the storage service. The MA then sends a Request message (9) to SP4 by invoking a bilateral bargaining based mechanisms. For simplicity, we assume that the chosen SP immediately agrees to provide the requested service by replying with an Agree message (10). In reality, the negotiation sequence could be similar to the one depicted in Figure 2. When an agreement is formed, the MA requests SP4 to *wait* (11) while the other service negotiations are completed. SP4 replies with an Agree message (12). Steps 13-16 can be referred to in [5]. The Service providers bind the agreement by replying with an Acknowledgement (17) for the bundled service (s1, s2, s3).

2.2 Performance Evaluation

An important part of the specification phase of the negotiation protocol is to pre-validate its sequences and evaluate its performance through simulations. Metro Ethernet Network (MEN) architecture was modelled as an underlying transport network infrastructure in the QDINE service negotiation framework. This section documents the results of the simulation study of the negotiation protocol performance for different realistic parameters such as negotiation message sizes, negotiation frequency and input load. We believe this study is valuable as it considers the performance during the negotiation phase over MEN rather than the end-to-end performance of the transport of data traffic.

The study investigates the following negotiation sequence: (i) the negotiation of a single service element where the negotiation is direct between the SP and CA and (ii) the negotiation of multiple service elements simultaneously from different types of SPs via the MA. To the best of our knowledge, whilst there have been other studies conducted for negotiation protocols, they neither evaluate the performance for different negotiation message sizes nor for varying background input load from other stations connected to the network.

Simulation Model. The simulations were performed by using OMNeT++ tool [7], the framework INET was used for the simulation of wired (LAN, WAN) and was implemented in our model environment. The QDINE framework with MEN infrastructure was modeled and simulated distinctly for (i) direct negotiation and (ii) indirect negotiation.

The MEN service type that we used is the Extended Virtual PrivateLAN (EVP-LAN) service and the background traffic generated was best-effort service, which is carried by the EVP-LAN. Diverse actors in the QDINE negotiation framework – CA, SP and MA – were modeled as network stations, i.e. Ethernet hosts. Each Ethernet host was modeled as a stack that in a top to bottom order comprised of: (a) two application models – a simple traffic generator (client side) for generating request packets and a receiving entity (server side), which also generates response packets, a LLC module and a CSMA/CD MAC module.

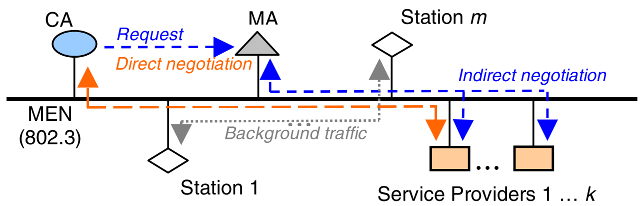

Fig. 4. Metro Ethernet Network (MEN) model

2.2.1 Results and Discussion – Direct Negotiation for a Single Service

In direct negotiation (refer Fig. 4), the negotiation messages flow directly between the consumer agent (CA) and the service provider (SP). The input data traffic load on the MEN was generated by the linked nodes (stations) 1…*m*. other than the negotiating control stations. The input load is expressed as a fraction of the 10Mbps capacity of the MEN. In the experiments performed, we varied the input load by keeping the packet sending rate constant and increasing the number of stations.

Table 1. Attributes specific for direct negotiation

Parameter	Value
Packet size – background traffic	1000 bytes
Packet arrival rate – background traffic	5, 10 packets./sec
Negotiation frequency	1, 10 negotiation./sec
Packet generation	Exp. Distributed
Negotiation message sizes	100-300 bytes

A) Impacts of negotiation frequency and back-ground traffic packet arrival rate: The attributes used for this simulation are given in Table 1. Fig. 5 shows the impact of the negotiation frequency on the traffic control as well as the generated background data traffic's packet arrival rates on the mean transfer delay of the negotiation packets received at the CA.

The negotiation control protocol used was bilateral bargaining. From Fig. 5a it can be seen that the increase in the negotiation frequency has a marginal effect on the mean delay up to the input load where the mean delay increases asymptotically (0.7 in Fig. 5a). This shows that the negotiation protocol is more influenced by input data traffic load on the network relative to negotiation control frequency, which will facilitate to improve scalability of negotiation stations on the underlying network.

B) Impact of different negotiation messages sizes: We studied this to determine the impact on performance at the SP, CA and overall network due to the following negotiation control messages sizes (i) 100 – 300 bytes (ii) 300 – 900 bytes (iii) 300 – 1800 bytes.

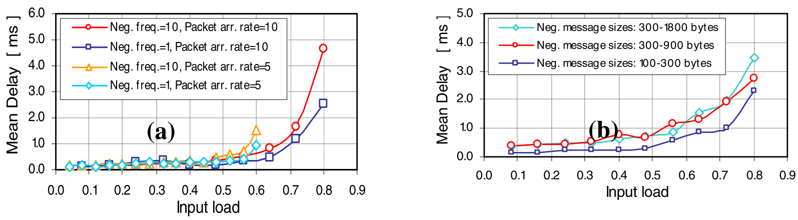

Fig. 5. (a) Time Performance at the consumer agent (CA) (b) Performance at the CA – Mean delay vs. Input load

The impact of negotiation message size on the mean transfer delay obtained at the CA is shown in Fig. 5b. The impact was found to be more pronounced at the SP than at the CA. The results of the impact at SP are not shown here due to space limitations. Most of the large size messages are received at SP relative to the CA, which causes a larger overall delay. Note: the total end-to-end delay of a control frame transfer is the sum of queuing delay, transmission delay, propagation delay and contention time. Except for the propagation delay (fixed for a medium of length L) all other delays are higher for larger frames.

2.2.2 Results and Discussion – Indirect Negotiation for Multiple Services

This subsection provides an evaluation study of our service negotiation control protocol for multiple service elements that can be negotiated between a SP and a CA. This study uses the parameters values similar to that given in Table 1. As the negotiation messages and the background traffic are of different packet sizes and have relative traffic flow asymmetry it makes the performance study more realistic.

The method for the indirect negotiation of multiple services has been proposed in section 2.1.2. The operation of the indirect negotiation that was modelled incorporated negotiations between the CA and the SP for multiple services indirectly via the MA. The MA can also be configured to possess the functionality of a resource broker, which will facilitate to identify the state and the services available at the various SPs. This will help to reduce the overall time of negotiation between the CA and the SPs via the MA. The mean delay of the messages received at the MA is presented in Fig. 6a, when the MA actively negotiates with the SPs on behalf of the CA. The SPs on receipt of the cfp respond in a distributed manner, which causes relatively lower

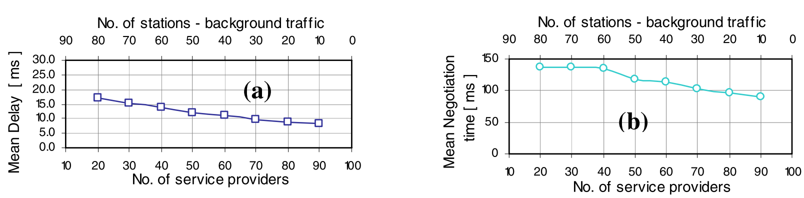

Fig. 6. (a) Mean delay vs. No. of neg. stations (No. of background stations) at the MA. (b) Mean negotiation time vs. No. of neg. stations (No. of background stations).

contention delays hence mean delay than it would be if all the SPs were to respond simultaneously. Due to this the mean delay of the messages at the MA is decreased. Also, background traffic has more impact on the performance than the number of SPs.

In the indirect negotiation cases it was considered SPs to be grouped in blocks of 10 each forming a bundled service. The SPs within each SP block were candidates for providing a distinct service element. The mean negotiation time for this scenario is shown in Fig. 6b. It can be seen that in this case also the background traffic influences the delay. The result in Fig. 6b indicates that the increase in number of the negotiating stations has relatively lesser impact than the background traffic. This means that a higher capacity network such as 100 Mbps/1 Gbps/10 Gbps will be a more suitable infrastructure to limit the effect of the background traffic.

3 Conclusions

We have proposed a dynamic negotiation protocol that was explained in a generic QoS framework for realistic scenarios that involved intra-domain negotiation. Salient performance evaluation results obtained from extensive simulations showed the effectiveness of the protocol for both direct and indirect negotiations. In particular, the negotiation protocol was shown to be feasible and scalable i.e. the negotiation protocol messages did not limit the scalability.

Acknowledgements. This research was performed as part of an ARC Linkage Grant, between Bell Labs (Alcatel-Lucent) and UTS, Sydney. We express our thanks to Les Green and Prof. John Debenham for their contributions to the QDINE framework.

References

1. Nguyen, T.M.T., Boukhatem, N., Doudane, Y.G., Pujolle, G.: COPS-SLS: A Service Level Negotiation Protocol for the Internet. IEEE Communications Magazine (2002)
2. Chen, J.-C., McAuley, A., Sarangan, V., Baba, S., Ohba, Y.: Dynamic Service Negotiation Protocol (DSNP) and Wireless Diffserv. In: Proceedings of ICC, pp. 1033–1038 (2002)
3. Wang, X., Schulzrinne, H.: RNAP: A Resource Nego-tiation and Pricing Protocol. In: Proceedings of NOSSDAV, NJ, USA, pp. 77–93 (1999)
4. FIPA Contract Net Interaction Protocol Specification (December 2003), `http://www.fipa.org`
5. Green, L., Mirchandani, V., Cergol, I., Verchere, D.: Design of a Dynamic SLA Negotiation Protocol for Grids. In: Proc. ACM GridNets, Lyon, France (2007)
6. Green, L.: PhD Thesis, Automated, Ubiquitous delivery of Generalised Services in an Open Market (April 2007)
7. Discrete Event Simulation Environment OMNeT++, `http://www.omnetpp.org`