Proceedings of IEEE/IFIP Network Operations & Management Symposium (NOMS 2010), Osaka, Japan, April 2010

Distributed Approaches to S-CSCF Selection in an IMS Network

Plarent Tirana and Deep Medhi
Networking & Telecommunication Research Lab
Computer Science & Electrical Engineering Department
University of Missouri–Kansas City, USA

***Abstract*—The IP Multimedia Subsystem (IMS) is a generic open-systems architecture offering converged multimedia services over IP. A function in the signaling plane of an IMS network is the interrogating Call/Session Control Function (I-CSCF) choosing the serving CSCF (S-CSCF) for a request. In this paper, we investigate distributed S-CSCF selection by considering four schemes: uniform random allocation, round robin, shortest expected delay, and least response time with network feedback. Through our study, we found that the shortest expected delay gives the best performance when the message size is one. However, when the message size of a request is higher, the shortest expected delay and the round robin scheme perform similarly in most cases while they are both better than uniform random allocation and least response time with network feedback schemes.**

I. Introduction

The IP Multimedia Subsystem (IMS) is a generic open-systems architecture offering converged multimedia services over IP [5], [14]. A high level view of the IMS architecture is shown in Fig. 1. The Session Initiation Protocol (SIP) was chosen for the signaling plane in the IMS network mainly due to its scalability and flexibility. The 3rd Generation Partnership Project (3GPP) has defined several mandatory SIP extensions for IMS to the SIP core functionality. Three types of SIP proxies are defined by 3GPP for IMS signaling, known as CSCF (Call/Session Control Function). These proxies are categorized by their signaling functionality as P-CSCF (proxy CSCF), I-CSCF (interrogating CSCF), and S-CSCF (serving CSCF). Note that in an operational environment, multiple instances of CSCFs may be distributed geographically across the network. Two of these types (P-CSCF and S-CSCF) are assigned to the IMS terminal (user) for the entire registration process, while the I-CSCF acts as a load balancer only during the IMS registration phase and selects the best-fit S-CSCF for the user. This process is defined as *S-CSCF Assignment* by 3GPP. The S-CSCF acts as a liaison for the user in the signaling plane. In fact, it intercepts every SIP message destined/originated to/from an IMS terminal, decides which AS (application server) to invoke, checks the user profile, and authorizes the services. It is imperative for the quality of service offered to the user to have the best S-CSCF selected. Our attempt in this work is to consider and compare a number of algorithms to select the best S-CSCF when a request arrives.

There are two algorithms proposed for *S-CSCF Assignment*: one is defined by 3GPP [5] and the other by Motorola [6]. In the 3GPP approach, selection criteria include capabilities that match the user profile against capabilities of the different S-CSCFs in the network, operator preference on a per user basis, topological information (location user/S-CSCF), and availability information. This infers a simple round-robin strategy when all things are equal. Obviously, this approach does not consider either the round trip delay of the SIP messages, or the load of the S-CSCF. Motorola has proposed an approach to this problem based on a heartbeat mechanism by using the SIP OPTIONS method. I-CSCF will periodically ask for all the load related information from the S-CSCF, store this information and plans to apply some sort of a load balancing algorithm (without providing any details) to perform load distribution during user registrations. While this might seem like a good approach (because the load of the S-CSCF could somewhat impact the QoS), it overlooks the main factor contributing directly to the QoS, the end-to-end signaling delay. 3GPP has also discussed the load balancing problem [1]; however, no algorithmic detail nor control trigger points are available as of now.

It may be noted that SIP is transport protocol agnostic; it can be implemented over either UDP or TCP. In addition, SIP signaling can be performed in a stateless or stateful fashion. In the stateless case, every SIP message is routed individually; thus, it is possible that messages that are part of a particular session are routed differently (taking different paths). In the stateful case, once the session is established, all the messages related to this session are routed along the same path. Despite SIP being transport protocol agnostic, 3GPP specifies that TCP be used for SIP for IMS signaling and for stateful treatment of sessions. With this stateful treatment, the I-CSCF will assign the S-CSCF at the registration time (beginning of the session) and stick with it for the entire session duration for a particular request. The P-CSCF will send the SIP messages directly to the S-CSCF without again involving the I-CSCF for this request. In this paper, we follow 3GPP specification of using a TCP implementation and stateful treatment of the SIP requests. Since the number of messages can depend on a particular request type, we consider this as a critical factor in our study by introducing a parameter called the *message size* parameter associated with each request. This parameter represents the number of packets a SIP request may consist of in order to establish a connection. Simply put, the message size represents the number of packets that a SIP request may generate.

Generally, the S-CSCF selection is a distributed load bal-

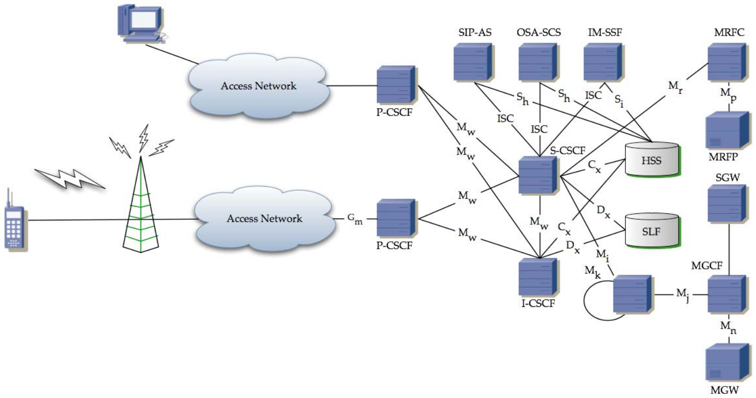

Fig. 1. Reference IMS Architecture

ancing problem. Load balancing is not a new concept in distributed computing. There has been work on web server load balancing in the Internet [3], [11]. However, little work has been done on S-CSCF load balancing in the IMS network. For instance, in an IMS network, because of multiple S-CSCFs for handling requests located in different geographical regions and with I-CSCF being involved as intermediaries, the knowledge available at each node is not instantaneous. Therefore, the latency may be measured at the receiving end, but feeding this back to the entry points has a time lag; thus, a protocol message update system may be needed for distributed control and management for S-CSCF selection. Alternately, a P-CSCF may decide to choose an S-CSCF based on information locally available. It should also be noted that such load balancing, because of geographic distribution and IMS requirements, must take its architectural context into account, rather than using a load balancing approach that may be appropriate for web server load balancing. Thus, we take such issues into account to study S-CSCF selection in a distributed IMS environment.

The rest of the paper is organized as follows. In Section II, we introduce the operational model that is followed by the S-CSCF selection schemes in Section III. An analytical treatment of the S-CSCF selection is presented in Section IV when the message size is one. Section V discusses the simulation results considering a number of different values for the message size parameter. In Section VII, we briefly discuss future work.

II. Operational Model

Our operational model considers an IMS environment where a set of K I-CSCFs can share the load with a set of N S-CSCFs while there are M independent P-CSCFs (input sources). This environment is more generic than the scenario of a single I-CSCF with N S-CSCFs sharing the load considered in [5], [6]. We also assume that the number of sessions accepted at each P-CSCF is finite. As we consider the stateful request, the I-CSCF will assign the S-CSCF at the registration time (beginning of the session) and stick with it for the entire session duration. The P-CSCF sends the SIP messages directly to the S-CSCF without again involving the I-CSCF.

We are primarily interested in the round-trip delay for request routing to understand the user's perception of set-up delay. This is important in the case of the distributed applications in IMS where the clients, load balancers, and application servers are spread over a large geographic area. Note that the role of an I-CSCF is primarily that of a lookup function to forward a request to the right S-CSCF (during the SIP registration only); the lookup time (however small) would be reduced with distributed I-CSCFs across the network.

As we will discuss in the next section, we consider an algorithm that requires delay-related information to be exchanged between P-CSCFs, I-CSCFs, and S-CSCFs. However, such information may not be available instantaneously to all entities in the environment. Thus, to account for this for use in an actual operational environment, we have also designed an update protocol to convey the delay-related information to the I-CSCF (if a single I-CSCF is implemented) or multiple I-CSCFs. This includes both pushes from the P-CSCFs and S-CSCFs toward the I-CSCFs and periodical and/or on-demand pulls from I-CSCFs. We use the SIP INFO field to convey the delay-related information. We use two key message types: Request-I-CSCF-Broadcast and Response-I-CSCF-Broadcast. The first is a message that is sent from any of the I-CSCF and targets all the P-CSCFs and S-CSCFs. In response, the I-CSCF is expecting that each of the P-CSCFs will send the delay-related information from itself to all the S-CSCFs. Similarly, each of the S-CSCFs will reply with their own information. In addition, we have Request-I-CSCF-Single-Destination message type in case of requesting from a specific destination.

III. S-CSCF Selection Schemes

Given the IMS operational environment of M P-CSCFs, K I-CSCFs, and N S-CSCFs, we considered four distributed algorithms for S-CSCF selection: 1) uniform random allocation (URA) where the S-CSCF is chosen randomly between the available ones, 2) round robin (RR) where the next request is routed to the next available S-CSCF, 3) lowest response time with network feedback (LRT-NF), and 4) shortest expected delay (SED). These were chosen due to their intuitive nature on how they might help in reducing the response time. The first two are commonly well-known approaches, while the third one ("lowest response time") handles incoming requests by periodically considering the delay estimate using the delay update protocol.

The fourth algorithm, shortest expected delay algorithm, will assign the S-CSCF to the server that has produced the shortest expected delay. As such, it is similar to the least response time because it takes a direct approach to minimizing the end-to-end signaling delay. However, in this case, the P-CSCF uses local information (such as queue size) without coordinating with I-CSCFs or S-CSCFs.

Note that all of the schemes are distributed approaches. In the case of URA, RR and SED, the decision on selecting an S-CSCF is completely local while with LRT-NF, the decision is based on coordinated information exchange on delay-related information with S-CSCFs through an update protocol. While there are certainly other algorithms possible, we concentrated on these four algorithms to understand how S-CSCF is impacted for stateful requests. This is considered further in term of the message size parameter m, which is discussed further in the next two sections.

IV. Analytical Representation

Recall that a particular request may generate more than one message depending on the type of a request that requires SIP signalling; this is transmitted using TCP from a P-CSCF to the same S-CSCF for a specific request. We denote the size of a request in terms of the number of packets as *message size* by m. Thus, when $m = 1$, this means that each request generates only one message that is sent in a single packet from a P-CSCF to an S-CSCF. This special case can be analytically represented. In this case, the packet arrival rate is nothing but the input rate of requests at each P-CSCF. For the IMS network environment we are considering, we assume the following:

- The input to P-CSCFs are Poisson processes with rates $\lambda_p, p = 1, 2, ..., N$.
- The service times for P-CSCFs are exponential $\mu_p, p = 1, 2, ..., N$.
- For each of the P-CSCFs we assume utilization stability, i.e., $\rho_p = \lambda_p/\mu_p < 1, i = 1, 2, ..., N$.
- The input to S-CSCFs are Poisson processes with rates $\lambda'_s = \sum_{p=1}^{N} \lambda_p q_{ps}, s = 1, 2, ..., M$, where q_{ps} represents the proportion of requests from p to s.
- The service times for S-CSCFs are exponential $\mu'_s, s = 1, 2, ..., M$.
- For each of the S-CSCFs, we assume stability $\rho'_s = \lambda'_s/\mu'_s < 1, s = 1, 2, ...M$.

A request arrival in an IMS request is usually a call (either a phone or a multimedia call); thus, it is reasonable to assume that the request arrival is Poisson. While the output from a P-CSCF after being served could affect the arrival distribution to an S-CSCF depending on the service rate at a P-CSCF, we make the simplifying assumption that the arrival to S-CSCF is also Poisson. This is also reasonable as, in most realistic cases, it is not desirable to have highly utilized P-CSCFs as this will add to the latency during the call set up time.

Under the independence assumption, the delay performance response, W, going through the system when $m = 1$, is given by

$$W = \frac{\sum\limits_{p=1}^{N} \frac{\rho_p}{1-\rho_p}}{\sum\limits_{p=1}^{N} \lambda_p} + \frac{\sum\limits_{s=1}^{M} \frac{\rho'_s}{1-\rho'_s}}{\sum\limits_{s=1}^{M} \lambda'_s} + \bar{\Delta}_{ps}, \tag{1}$$

where $\bar{\Delta}_{ps}$ is the average *in-network* delay going from P-CSCF p to S-CSCF s. It may be noted that

$$\sum_{s=1}^{M} q_{ps} = 1, \quad p = 1, 2, ..., N \tag{2}$$

The main difference between different schemes can be represented through q_{ps}. In the case of URA, q_{ps} represents the probability of the uniform proportion of traffic that is allocated to each S-CSCF s by each P-CSCF p. Thus, this quantity is given by

$$q_{ps} = 1/M, \quad p = 1, 2, ..., N \tag{3}$$

For URA, the response W can be analytically calculated when the message size, m, is one.

For RR, selection of q_{ps} is based on allocating in a round-robin manner. If all arrivals and service rates were the same in the system, then in the limiting case, RR is the same as URA, but not in general.

For LRT-NF, the selection q_{ps} is time-dependent, which is dependent on knowing the feedback from each S-CSCF. If we denote the feedback on delay at time t by $d_s(t)$ ($s = 1, 2, ...N$), then $q_{p\hat{s}}(t)$, $\hat{s} = 1, 2, ..., N$, is given by

$$q_{p\hat{s}}(t) = \begin{cases} 1, & d_{\hat{s}}(t) < d_s(t), s = 1, 2, ..., N; \\ 0, & \text{Otherwise.} \end{cases} \tag{4}$$

In the case of SED, the decision is local. If we denote the observed delay locally at p at time t by $T_{ps}(t)$, then $q_{p\hat{s}}(t)$, $\hat{s} = 1, 2, ...N$, is given by

$$q_{p\hat{s}}(t) = \begin{cases} 1, & T_{p\hat{s}}(t) < T_{ps}(t), s = 1, 2, ..., N; \\ 0, & \text{Otherwise.} \end{cases} \tag{5}$$

Although we represent the LRT-NF and SED above with just a single time-dependent parameter t, in practice, the estimation is based on periodic measurements that go through a smoothing operation using an exponential weighted moving average approach that can be adjusted by using a smoothing weight, α ($0 < \alpha < 1$); our simulation presented later uses the smoothing approach. Thus, in general, even for $m = 1$, the response W for schemes other than URA are hard to determine analytically.

Now, consider the case when $m > 1$, i.e., each request generates a number of messages due to SIP signalling that are

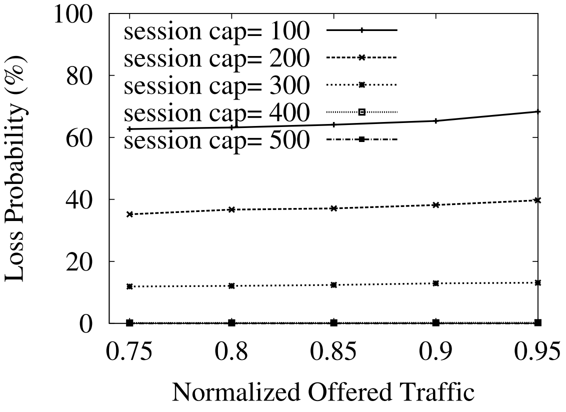

Fig. 2. Loss probability for different session buffer size/capacity

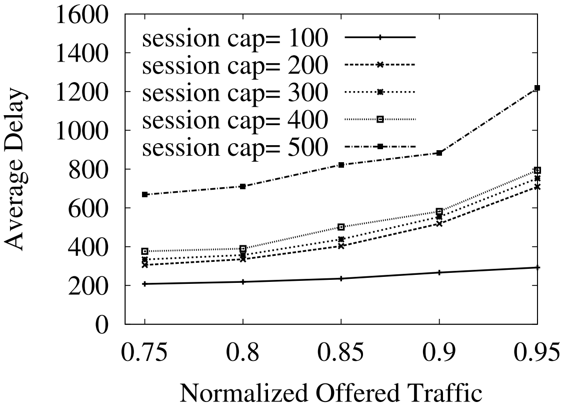

Fig. 3. Average delay for different session buffer size/capacity

to be sent on the same TCP connection between a P-CSCF and an S-CSCF. Since this is related to a particular request, all these m messages for this request must go from a P-CSCF to the same S-CSCF, while for a different request arriving at this P-CSCF may choose another S-CSCF. The case for $m > 1$ is difficult to track analytically due to packet sequence dependencies for an arriving request. While there has been much work on TCP modeling, there are some important differences here. Most TCP modeling studies consider a dumbbell topology to understand active queue management, i.e., a number of connections are tunneled through just a single congested link. There are two key differences here. First, in our case, we have load distribution through nodes in the IMS network where request routing takes place. Second, due to the importance of the real-time requirement of the signalling responses, this part of the network (which is usually in a private IP network) is engineered so that this does not become the dominant factor in latency in the SIP signal communication for a request in the IMS network (as opposed to the notion of congested link in most TCP modeling study). Finally, it is tempting to extend the above analytical representation by scaling the packet arrival rates when the message size m is multiplied by the request arrival rate. Through comparison with simulation, we found that multiplicative representation is quite off the mark.

V. Study Model and Results

We developed a simulation model to study S-CSCF selection schemes. For our simulation, the request input rates, λ_p, for different P-CSCFs are randomly chosen. The service rates of P-CSCFs and S-CSCFs are chosen in such a way that the system stability is maintained.

Each scenario in our simulation handled arrivals in the tune of a million incoming requests to be routed to the N S-CSCFs. In our simulation, the S-CSCF assignment is dynamic and the choice made when a session started might not be the best one for subsequent requests, depending on the state of the system as perceived by a selection method at that instant.

For all our studies, we have used a number of different cases by changing the number of P-CSCFs (M) and S-CSCFs (N), while keeping the actual number of P-CSCFs and S-CSCFs the same for each case; this range for $M \times N$ is varied from 4×4 to 12×12. We decided to keep both M and N the same value so that it is easier for us compare for different cases. In an actual network, they do not need to be the same.

Our simulation model has the ability to limit the number of incoming session requests that can be handled at any time at each P-CSCF; this parameter will be referred to as *session buffer size* or *session capacity* (not to be confused with the message size, m). Note that for the finite session buffer size model, it is possible to allow the system to reach a normalized load of one or beyond, unlike the infinite buffer system because requests can be rejected when the buffer capacity is reached.

Our first study was to determine the appropriate session buffer size. For this, we tested the algorithms by changing the number of sessions allowed in the buffer from 100 to 500 for a number of different message sizes and normalized loads. For illustration, we report the results for the lowest response time scheme with network feedback (LRT-NF) when the session length is fixed at 5 while varying the normalized load from 0.75 to 0.95. From Fig. 2, we can see that when the session buffer size is 100, the incoming request loss is very high, it then drops to essentially zero when the session buffer size is increased to 500. As expected (Fig. 3), the average delay is low when the session buffer size is 100 (where the high loss occurs), and the delay increases when the buffer size is increased. The same basic pattern is observed with the other load distribution algorithms. Heretofore, we focused our main study when the session buffer size was set to 500 so that the loss probability was in the order of 10^{-3}; with this being fixed, we attempted to understand the impact on the response time delay with various distribution algorithms for S-CSCF selection. In a future work, we will address in detail the trade-off between buffer size and delay performance, especially for overloaded situations, by considering different dropping strategies.

The simulation code was also validated against the analytical model discussed in the previous section using uniform random allocation for S-CSCF selection when the session

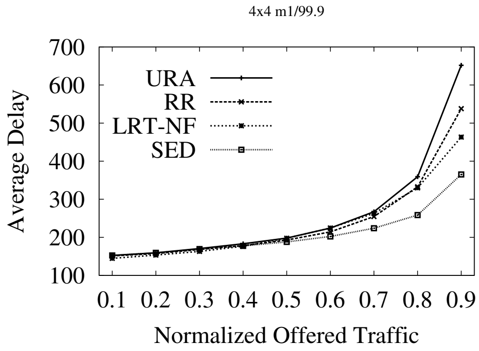

Fig. 4. Average Delay (4×4, $m = 1$, $\alpha = 0.01$)

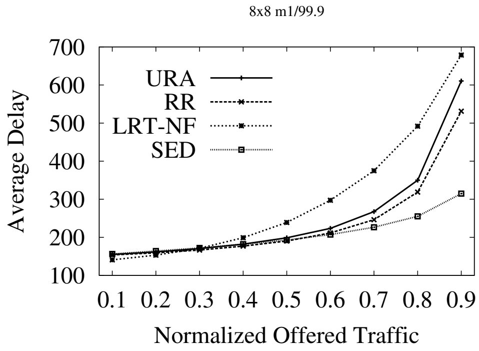

Fig. 5. Average Delay (8×8, $m = 1$, $\alpha = 0.01$)

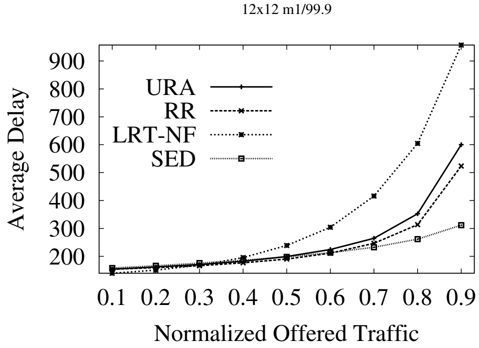

Fig. 6. Average Delay (12×12, $m = 1$, $\alpha = 0.01$)

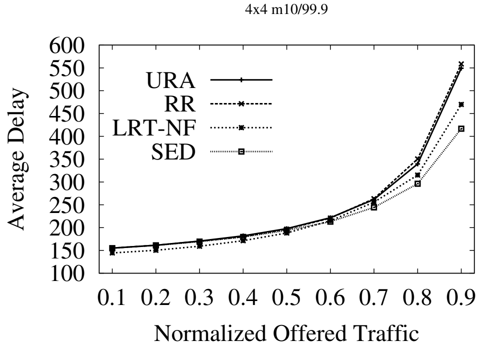

Fig. 7. Average Delay (4×4, $m = 10$, $\alpha = 0.01$)

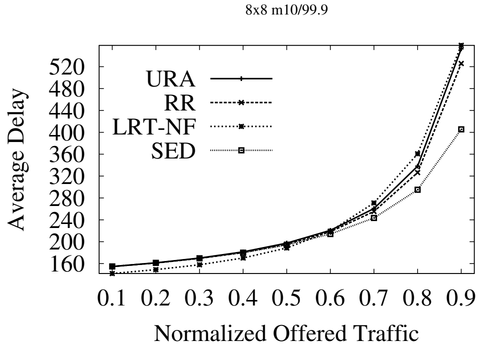

Fig. 8. Average Delay (8×8, $m = 10$, $\alpha = 0.01$)

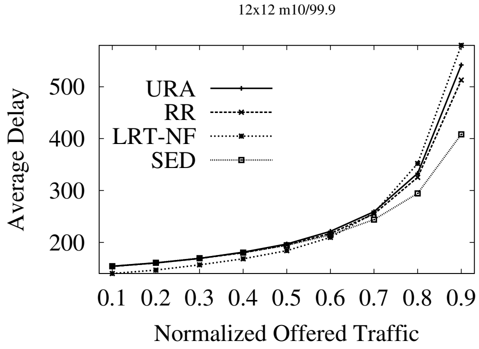

Fig. 9. Average Delay (12×12, $m = 10$, $\alpha = 0.01$)

length of a request is set to one (i.e., $m = 1$) and the session buffer size is set to 500. The results between the analytical model and the simulation model were found to differ by only 1% to 2% for this case.

For the remainder of our discussion, there are three main parameters that we focused on in our study for the best S-CSCF selection: 1) normalized offered load, 2) topology, expressed as the number of ingresses (M)–egresses (N) to the simulation, and 3) message size (m). The results are presented for normalized offered loads varying from 0.1 to 0.9. For statistical measures, we collected the average, the standard deviation, and the 95th percentile of delay, on the system response time; in this paper, we report all our results for average delay.

Consider the size of ingresses/egresss ($M \times N$) grids for P-CSCFs and S-CSCFs. In Fig. 4 to Fig. 12, we show the average delay for three cases of ingresss/egresses: 4×4, 8×8 and 12×12. We first consider the case when $m = 1$ (refer to Fig. 4 to Fig. 6). As expected, in low load, all schemes perform about the same. As the normalized load increases, we note that the URA has the highest average delay for 4×4, but LRT-NF has the highest average delay for 8×8 and 12×12, while SED has the least delay in all these cases. While this may seem counter-intuitive, for heterogeneous traffic rates for different inputs, λ_p, and a larger network grid, SED achieves better load balancing, thereby reducing the average delay compared

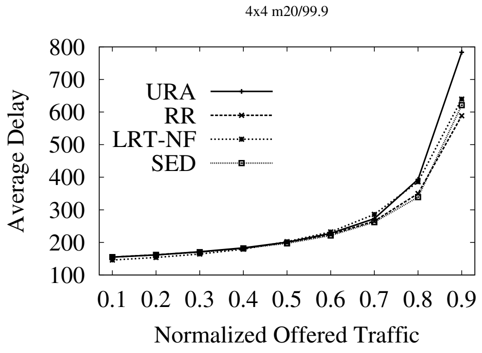

Fig. 10. Average Delay (4×4, $m = 20$, $\alpha = 0.01$)

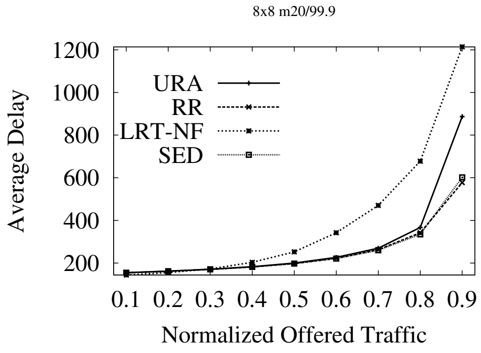

Fig. 11. Average Delay (8×8, $m = 20$, $\alpha = 0.01$)

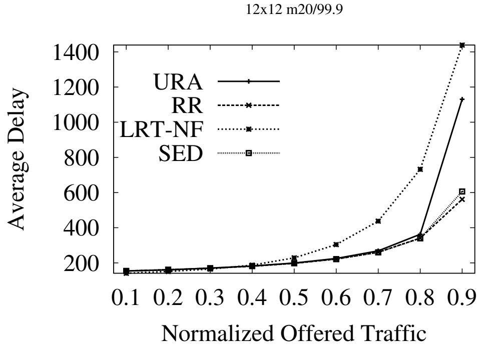

Fig. 12. Average Delay (12×12, $m = 20$, $\alpha = 0.01$)

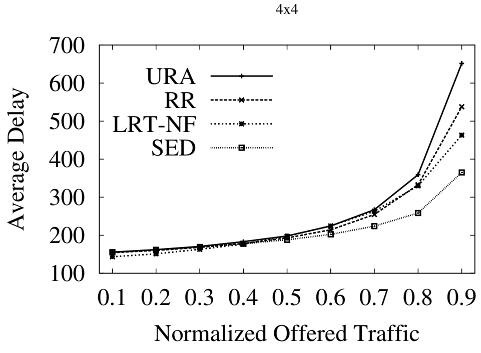

Fig. 13. Average Delay (4×4, $m = 1$, $\alpha = 0.05$)

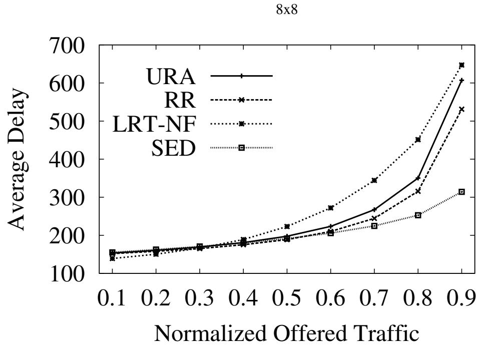

Fig. 14. Average Delay (8×8, $m = 1$, $\alpha = 0.05$)

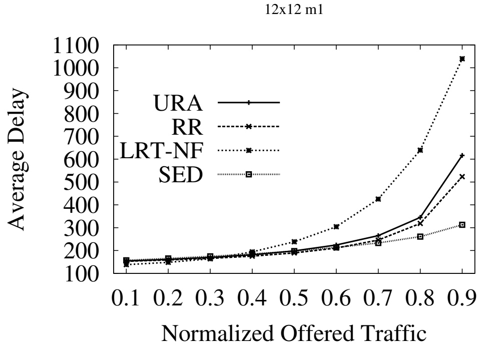

Fig. 15. Average Delay (12×12, $m = 1$, $\alpha = 0.05$)

to the other schemes. We note that LRT-NF, despite receiving feedback from the S-CSCF, cannot do better than SED when the message size (m) is just one. This means that the local information as in SED is sufficient for obtaining good global load balancing as the load increases.

Next we consider when the message size of a request is increased. We show results for two values of m; $m = 10, 20$ (Fig. 7 to Fig. 12). There are several observations. First, we note that there is virtually little difference between URA and RR for $m = 10$, while SED still shows the lowest delay as the load increases. On the other hand, if the message size is high ($m = 20$), URA has the worst delay for the smaller topology (4×4), while LRT-NF has the worst delay for larger topology (8×8 and 12×12). Furthermore, we note that with $m = 20$, both RR and SED result in a lower delay while RR edges out SED in some instances. This points out that when a request has multiple messages, and since they are to be routed to the same S-CSCF for a specific connection request, packets cannot take advantage of updated information as in LRT-NF; rather, RR gives the best performance.

In order to see if smoothing makes any difference, we also changed the value of the smoothing weight, α (for the results described thus far, α was set to 0.01; for the remaining discussion, α is set to 0.05 in Fig. 13 to Fig. 21). Results for 4×4, 8×8, and 12×12 are reported again for $m = 1, 10, 20$. We note that the general behavior remains the same regardless

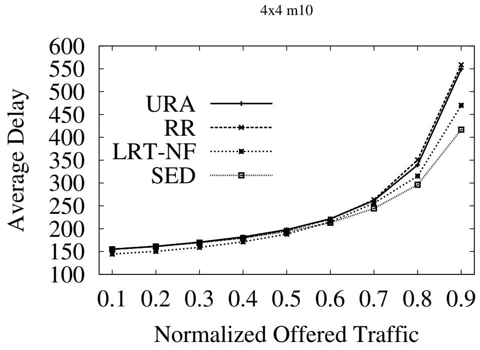

Fig. 16. Average Delay (4×4, $m = 10$, $\alpha = 0.05$)

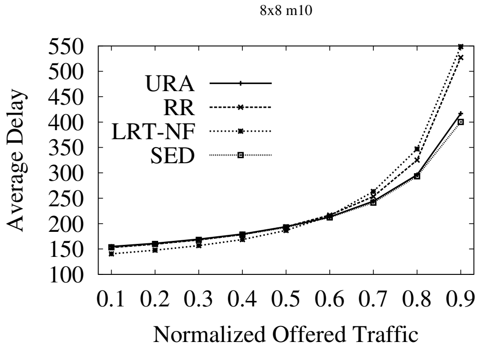

Fig. 17. Average Delay (8×8, $m = 10$, $\alpha = 0.05$)

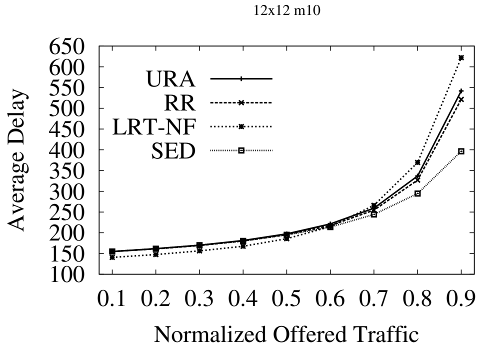

Fig. 18. Average Delay (12×12, $m = 10$, $\alpha = 0.05$)

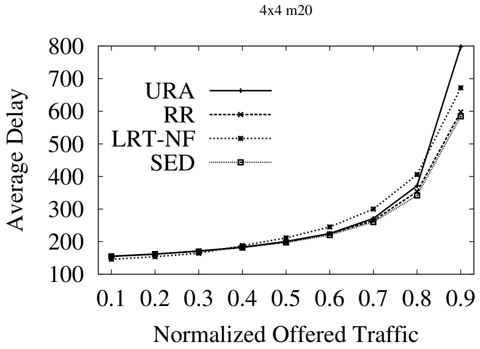

Fig. 19. Average Delay (4×4, $m = 20$, $\alpha = 0.05$)

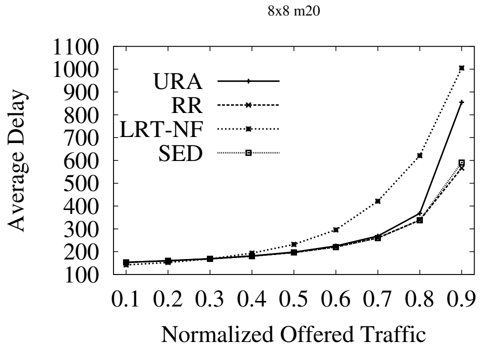

Fig. 20. Average Delay (8×8, $m = 20$, $\alpha = 0.05$)

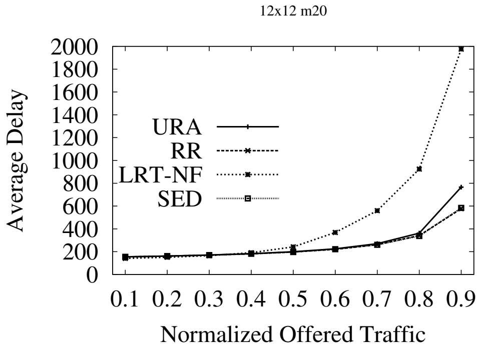

Fig. 21. Average Delay (12×12, $m = 20$, $\alpha = 0.05$)

of the smoothing weight.

Based on the above observations, we can infer that SED is the best load distribution algorithm when $m = 1$. However, if the message size increases, LRT-NF has the worst performance while RR and SED provide the lowest delay. Note that LRT-NF has the overhead of protocol exchange messages due to the update protocol; the actual overhead depends on the periodicity of updates exchanged between different CSCFs, and this need not be on a per request basis. Considering that LRT-NF does not perform well in the cases we studied and it has some overhead, it is questionable whether this scheme is worth considering at all. Finally, local decision as in RR or SED work quite well. This is somewhat counter-intuitive; we are doing further investigation to understand the dynamics and benefit of these schemes and get a better insight.

VI. Related Work

For the IMS architecture, there has been work on the scheduler and improvement of the IMS presence service [2], [17] and on an application server workload [15]. However, these works are orthogonal to our work. In a somewhat related work, SIP server overload has been studied [9]; but, this is not directly applicable to the IMS environment where multiple CSCFs are present. In our earlier work [16], we assumed stateless treatment of requests to understand the basic dynamics of the problem and presented preliminary results. Contrary to that, this paper considers stateful requests since

SIP signalling is required to use TCP in an IMS environment and we present work in the light of how the message size due to such signaling impacts the load balancing decision and system performance. Finally, the focus of the paper is distributed S-CSCF selection in order to achieve distributed management about CSCFs, ideally with little communication overhead. While two schemes have been proposed earlier [5], [6], the details are sketchy and no performance results are available.

An important focus of our study is when the message size is more than one due to SIP signalling of a request. Since TCP is used for transport in an IMS environment, it seems to give the impression that TCP modeling could be applicable here. On the other hand, while there has been much work on TCP modeling, there are some important differences here. Most TCP modeling studies consider a dumbbell topology to understand active queue management, i.e., a number of connections are tunneled through just a single congested link (see, for example, [4], [7], [8], [10], [12], [13], [18]). An important difference is that load distribution through nodes in the IMS network results in request routing that is not possible in a dumbbell topology with only a congested link. Furthermore, due to the importance of the real-time requirement of the signalling responses, this part of the IMS network is provided through a private IP network that is engineered so that this does not become the dominant factor in latency in the SIP signal communication for a request in the IMS network (as opposed to the notion of congested link in most TCP modeling study).

VII. Summary and Future Work

In this work, we consider the problem of signaling delay due to distributed S-CSCF selection when a request weaves through a network of P-CSCFs and S-CSCFs in an IMS network. This problem is then reduced to a stateful load distribution problem due to SIP signalling in an IMS network. We analyzed four methods: uniform random allocation, round-robin, least response time with network feedback, and shortest expected delay. We have designed a simple delay update protocol to convey the delay information to the I-CSCF that includes both pushes from the P-CSCFs and S-CSCFs towards the I-CSCFs and periodical and/or on-demand pulls from the I-CSCF. We use SIP extra-headers to convey the delay-related information. This mechanism is used in the LRT-NF scheme. In general, SED and RR are better than LRT-NR and URA when the load is high and the message size (m) is high.

There are two additional directions we plan to pursue. First, our work assumes the request arrival to be Poisson. We are interested in the impact on the distributed algorithms when the arrival distribution changes.

Secondly, in the case of overload in the network, P-CSCFs may turn down newly arriving requests so as to not impact the response time for the ones that are already admitted. While we briefly discussed this in terms of the session buffer size, this issue requires further detailed study to better understand the impact and the trade-off in an overloaded situation. We are currently investigating these two directions.

References

[1] 3rd Generation Partnership Project, "Technical specification group services and system aspects; feasibility study on IMS evolution (release 9)," Technical Report 3GPP TR 23.812 V0.5.1 (2009-04). http://www.3gpp.org

[2] M. T. Alam, "Design and analysis for the 3g ip multimedia subsytem," Ph.D. dissertation, Bond University, Australia, 2007.

[3] T. Bourke, *Server Load Balancing*. O'Reilly, 2001.

[4] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang, "Recommendations on queue management and congestion avoidance in the Internet," *IETF RFC 2309*, April 1998. http://www.rfc-editor.org/rfc/rfc2309.txt

[5] G. Camarillo and M. A. García-Martín, *The 3G IP Multimedia Subsystem (IMS), 2nd Edition*. John Wiley & Sons, 2006.

[6] B. Chattopadhayay and M. A. Muñoz de la Torre, "S-CSCF load balancing," Motorola, Inc., April 17, 2006, http://www.priorartdatabase.com/IPCOM/000136381/.

[7] W. Feng, D. D. Kandlur, D. Saha, and K. G. Shin, "Stochastic fair blue: A queue management algorithm for enforcing fairness," in *Proc. IEEE INFOCOM'2001*, Anchorage, AK, 2001, pp. 1520–1529.

[8] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. on Networking*, vol. 1, pp. 397–413, August 1993.

[9] V. Hilt and I. Widjaja, "Controlling overload in networks of SIP servers," in *Proc. of IEEE ICNP'2008*.

[10] C. V. Hollot, Y. Liu, V. Misra, and D. Towsley, "Unresponsive flows and AQM performance," in *Proc. IEEE INFOCOM'2003*, San Francisco, CA, 2003.

[11] C. Kopparapu, *Load Balancing Servers, Firewalls and Caches*. John Wiley & Sons, 2002.

[12] D. Lin and R. Morris, "Dynamics of random early detection," in *Proc. ACM SIGCOMM'97*, Cannes, France, September 1997, pp. 127–137.

[13] V. Misra, W.-B. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," in *Proc. ACM SIGCOMM'2000*, Stockholm, Sweden, August-September 2000, pp. 151–160.

[14] M. Poikselka, A. Niemi, H. Khartabil, and G. Mayer, *The IMS: IP Multimedia Concepts and Services*. John Wiley, 2006.

[15] N. Rajagopal and M. Devetsikiotis, "Modeling and optimization for the design of IMS networks," in *Proc. of 39th Annual Simulation Symposium, 2006 (ANSS'06)*, April 2006, 7 pages.

[16] P. Tirana and D. Medhi, "The effects of load distribution algorithms in application's response time in the IMS architecture," in *Proc. of 18th ITC Specialist Seminar on Quality of Experience*, Karlskrona, Sweden, May 2008, 9 pages.

[17] C. Urrutia-Valdés, A. Mukhopadhyay, and M. El-Sayed, "Presence and availability with IMS: Applications architecture, traffic analysis, and capacity impacts," *Bell Labs Technical Journal*, vol. 10, no. 4, pp. 101–107, 2006.

[18] L. Zhang, "A new architecture for packet switching network protocols," Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA, Tech. Rep. MIT-LCS-TR-455, 1989.