This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2009 proceedings

Service Engineering for Inter-Domain Overlay Networks

Eren Gürses, Jin Xiao, Raouf Boutaba
David R. Cheriton School of Computer Science,University of Waterloo,
Waterloo, ON N2L 3G1, Canada
{egurses, j2xiao, rboutaba}@uwaterloo.ca

***Abstract*—In recent years application-level networking (e.g. overlay networks, P2P networks) has become key enabling architecture for supporting distributed services on the Internet. We consider overlays to have delay and cost requirements and thus there exists an optimal distribution of the overlay traffic (the service engineering problem) across multi-domain underlay networks such that these requirements can be satisfied while achieving fairness among the overlays. To this end, we formulate the problem as a convex optimization problem and propose a distributed solution. We show the efficiency and effectiveness of our approach through simulation studies.**

I. Introduction

In recent years application-level networking (e.g. overlays, P2P networks, etc.) has become a key enabling architecture for supporting distributed services on the Internet. By routing its traffic through relays, an overlay can obtain much higher network performance than underlay routing [1]. However, their uncoordinated and self-serving routing behaviors raises significant concerns over network stability [2] and can cause load balancing issues in the networks [3]. Furthermore, when multiple overlays are supported by the same network infrastructure, there is no guarantee each overlay can obtain their performance objectives and maintain fairness. To date, there are few mechanisms for managing overlay networks in a multi-domain environment, where objectives from a dual facets must be met: 1) from the application service perspective, how can overlays meet their performance objectives; 2) from the network perspective, how to achieve resource load balancing and fairness. We believe the major challenge of this problem lies in the distributedness of the overlays and the large scale of the multi-domain networks. Unlike traditional traffic engineering, centralized planning can not be carried out in practice and the objectives of each overlay must be satisfied.

In this paper we consider the problem of traffic distribution for overlays in multi-domain networks. Figure 1 illustrates a simple example. Two logically separate overlays are supported by the multi-domain networks and consequently share some of the underlay network domains. The traffic demands of an overlay can be represented by a set of *services*. A service $s \in \mathcal{S}$ is the set $\{\mathcal{R}_s, x_s, d_s^{\text{pref}}, c_s^{\text{pref}}\}$, where $\mathcal{S}$ and $\mathcal{R}_s$ are respectively the set of all services and overlay routes used by s, x_s is the total rate of traffic, d_s^{pref} and c_s^{pref} are the respective delay and cost requirements of the service. In the example, three routes are possible for the service (s_1, d_1) in

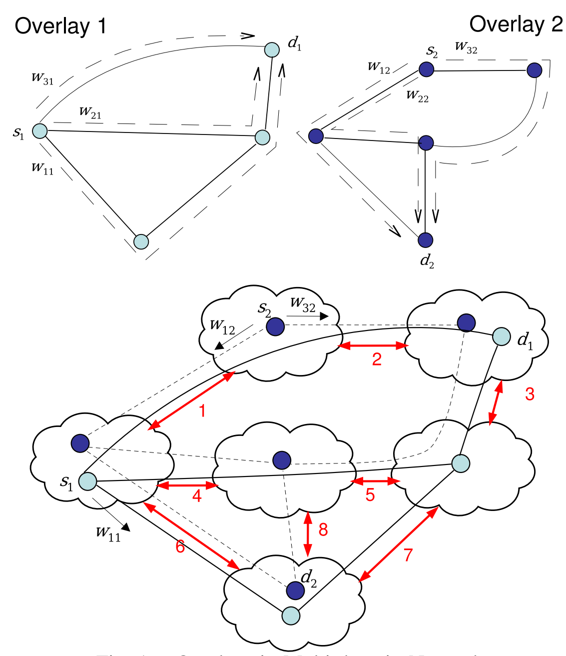

Fig. 1. Overlays in Multi-domain Networks

Overlay 1. We define the *service engineering problem* as: for each service find a distribution of traffic among its routes such that its delay and cost constraints are met and at the same time resource load balancing and fairness is achieved in the system. An effective solution to the service engineering problem is vital as it allows the overlays to achieve their performance objectives without inducing certain adverse side effects to the networks (e.g. network congestion and unfair resource usage). Furthermore, each network domain can manage the overlay traffic traversing its domain by adjusting the pricing parameter. We formulate the service engineering problem as a convex optimization problem with multi-criteria objectives and show that distributed, scalable and implementable solution to this problem can be obtained by applying dual-based optimization technique. The formulation of our multi-criteria objective function for the service engineering problem not only considers hard QoS constraints rather an average performance. And the implementation of the solution does not require global information exchange or explicit synchronization among the services. Experimental results show the effectiveness of our solution. Although we use overlay as the example application in this paper, we note that our approach is general enough to be applicable to other distributed applications such as P2P networks, content-sharing networks and distributed Web services.

The rest of the paper is organized as follows: Section II presents related work. Section III details the optimization model and its distributed implementation. Section IV presents a case study and discuss some implementation issues while Section V describes the simulation results. Section VI concludes the paper with summary and future direction.

II. Related Work

On the topic of overlay networks, Liu et. al. [2] have studied the interaction between overlay routing and traffic engineering in underlay network. They model the selfish behavior of the overlay routing and the system balancing objective of the underlying traffic engineering as a two-player non-cooperative non-zero sum game. They conclude that equilibrium exists in simple networks. In general topologies, oscillations and inefficiency arise due to the different objectives. Using realistic topologies and traffic demands, Qiu et. al. [3] presented through simulation that contrary to theoretical worst case, selfish routing in overlay achieves close to optimal average latency. Although they also note that some links in the network will have significantly increased congestion. Zhang, Kurose and Towsley [1] show that when underlay topology is rich, overlay can compensate underlay routing inefficiencies. Few work has studied the interaction of overlays and underlays in multi-domain networks. QoS assurance of overlays has also been studied in considerable detail in the literature. Gu et. al. [4] proposed a QoS-assured service composition mechanism in service overlay networks. They proposed a heuristic for finding QoS satisfied overlay routes via a linear multi-constraint mapping function. Duan et. al. [5] studied the problem of overlay bandwidth provisioning for QoS assurance. They mathematically estimated the amount of network bandwidth required to support the desired QoS quality of the overlay. Thus, far little work in literature address the service engineering problem as we have presented. To the best of our knowledge, there are currently no distributed optimization technique for solving this problem.

Since the Kelly's seminal paper [6] decomposition based solutions have become popular for distributed optimization. Although there are many published papers on multipath routing [7], [8], they are all about traffic engineering and joint routing and utility maximization. None of these papers from the distributed optimization literature are relevant to service optimization and inter-domain overlay networks.

III. Overlay Service Engineering in Multi-domain Networks

We consider a network consisting of multiple domains where each domain is represented by a set of virtual links. A virtual link abstracts the traversal of a domain (i.e. ingress gateway to egress gateway) through a domain. Thus an overlay hop across multiple domains can be represented as a list of adjoining virtual links. A service $s = \{\mathcal{R}_s, x_s, d_s^{\text{pref}}, c_s^{\text{pref}}\}$ supported by an overlay has a list of overlay routes $\mathcal{R}$ to carry its traffic. It is important then to determine the best distribution of the service traffic x_s such that the maximum delay across any of its route is constrained by d_s^{pref} and the total cost of communication is bounded by c_s^{pref}. Existing optimization technique on bounding the performance of routes usually uses average delay, which is not sufficient for dealing with real-time or multimedia services that requires strict end-to-end delay bound. We consider each domain to have control over the overlay traffics by placing a price on the rate of flow. Thus we denote the unit price of an overlay path by c_r, consisting of the sum of unit price of each domain the overlay route traverses. Furthermore, we associate a setup cost c_r^{setup} with each overlay route as the minimum charge required to keep the overlay route active. This cost model is general enough to capture the interactions between the overlay and the network domains. As the performance of the network is aggregated at the domain level, we model a virtual link as having a fixed capacity b_l (representing the domain's bandwidth provisioning for overlays) and aggregate flow rate f_l (representing the total volume of overlay traffic on the virtual link).

A. *The service engineering optimization problem*

In this section we introduce the service engineering optimization problem. Given a set of services as described above. The goal is to find a distribution of traffic for all of the services such that a multi-objective function is optimized which approximates a min-max problem. The multi objective function has two parts. The first part consider the end-to-end *delay* of service s, and the second part considers the total *cost* of supporting the service. The amount of traffic allocated to a route r of the service s is denoted by w_{rs} which is bounded with a minimum non-zero value of $w_{\min} = \frac{c_r^{\text{setup}}}{c_r x_s}$ in order to maintain the convexity of the problem.

$$\min_{w,f,d} \quad \sum_s \left(\frac{d_s}{d_s^{\text{pref}}}\right)^2 + \sum_s \left(\frac{\sum_{r\in\mathcal{R}_s} c_r x_s w_{rs}}{c_s^{\text{pref}}}\right)^2 \tag{1}$$

$$\text{s.t.} \quad \sum_{r\in\mathcal{R}'_l} \sum_{s\in\mathcal{S}_r} x_s w_{rs} \le f_l \qquad , \forall l\in\mathcal{L} \tag{2}$$

$$\sum_{l\in\mathcal{L}_r} \frac{1}{b_l - f_l} \le d_s \qquad , \forall s\in\mathcal{S}, r\in\mathcal{R}_s \tag{3}$$

$$\sum_{r\in\mathcal{R}_s} w_{rs} = 1 \qquad , \forall s \in \mathcal{S} \tag{4}$$

$$w_{\min} \le w_{rs} \le 1\,;\; 0 \le f_l \le b_l\,;\;\; d_s \ge 0 \tag{5}$$

Eq. 1 is formulated such that the delay and cost preference of the service becomes a "benchmark" for the optimizer. The resulting configuration will favor traffic distributions with low delay and cost ratio with respect to the preference bounds. Eq.

3 gives the relation between virtual link congestion level and virtual link delay under M/M/1 queuing. Eq. 5 is the fairness constraint. Firstly, because there exists a minimum charge c_r^{setup} for each route of the service, it is sensible to send at least enough traffic on the routes to compensate for the setup charge. Secondly, this constraint introduces a desirable fairness effect to the optimization that prevents overlays from unfairly abuse network links against other overlays. We will explain this effect in detail in Section IV. Finally, the flow constraint f_l is introduced in order to simplify the equations for clarity. It can in fact be substituted by $\sum_{r\in\mathcal{R}'_l}\sum_{s\in\mathcal{S}_r} x_s w_{rs}$ for equivalent results. The dual-based solution to the original problem in (1)-(5) is obtained as follows after writing the Lagrangian $L(w, f, d, \lambda, \mu, \nu)$ in (6).

$$L(w,f,d,\lambda,\mu,\nu)=\sum_{s\in\mathcal{S}}\Big(\frac{d_s}{d_{\text{pref},s}}\Big)^2+\Big(\frac{\sum_{r\in\mathcal{R}_s} c_r x_s w_{rs}}{c_{\text{pref},s}}\Big)^2 + \sum_{l\in\mathcal{L}}\lambda_l\Big\{\sum_{r\in\mathcal{R}'_l}\sum_{s\in\mathcal{S}_r} x_s w_{rs}-f_l\Big\}+\sum_{s\in\mathcal{S}}\nu_s\Big\{\sum_{r\in\mathcal{R}_s} w_{rs}-1\Big\} + \sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}_s}\mu_{rs}\Big\{\sum_{l\in\mathcal{L}_r}\frac{1}{b_l-f_l}-d_s\Big\} \tag{6}$$

First, the dual function $D(\lambda, \mu, \nu)$ is found by minimizing the Lagrangian w.r.t. the primal variables w, f and d:

$$D(\lambda,\mu,\nu) = \min_{\substack{w_{\min}\le w_{rs}\le 1\ ;\\ d_s\ge 0\ ;\ 0\le f_l\le b_l}} L(w,f,d,\lambda,\mu,\nu) \tag{7}$$

Due to the linearity of the differentiation operator, this minimization problem is separable into subproblems as below after properly arranging the terms in the Lagrangian.

$$\min_d \sum_{s\in\mathcal{S}}\Big\{\frac{d_s^2}{d_{\text{pref},s}^2}-\sum_{r\in\mathcal{R}_s}\mu_{rs}d_s\Big\} \tag{8}$$

$$\min_f \sum_{l\in\mathcal{L}}\Big\{\sum_{r\in\mathcal{R}'_l}\sum_{s\in\mathcal{S}_r}\frac{\mu_{rs}}{b_l-f_l}-\lambda_l f_l\Big\} \tag{9}$$

$$\min_w \sum_{s\in\mathcal{S}}\Big\{\Big(\frac{\sum_{r\in\mathcal{R}_s} c_r x_s w_{rs}}{c_{\text{pref},s}}\Big)^2+\sum_{r\in\mathcal{R}_s} w_{rs}\{\nu_s+x_s\sum_{l\in\mathcal{L}_r}\lambda_l\}\Big\} \tag{10}$$

Then the second step is to maximize the dual function w.r.t. the dual variables (11) where the ones corresponding to inequality constraints (i.e. λ, μ) take values larger or equal to zero and otherwise unconstrained (i.e. ν).

$$\mathbf{D}=\max_{\lambda,\mu\ge 0\ ;\nu} D(\lambda,\mu,\nu) \tag{11}$$

These two steps can be iteratively solved as given by the algorithm in Table I. The details of the algorithm is described in the following section. Before concluding this section we will introduce Theorem 1 in order to specify the necessary conditions on the selection of step sizes κ_w, κ_λ, κ_μ and κ_ν.

Theorem 1 *The distributed service engineering algorithm for supporting inter-domain overlay networks converge for step size* $0 \le \kappa_w \le 2/\max_{r,s}\Big(\sum_{r'\in\mathcal{R}_s}\frac{2c_r c_{r'} x_s^2}{c_s^{\text{pref}2}}\Big)$ *with* κ_λ, κ_μ, κ_ν *small enough [9]*

TABLE I
DISTRIBUTED SERVICE ENGINEERING FOR OVERLAY NETWORKS

Initialize: Set $t{=}0$ and $\lambda_l(t), \mu_{rs}(t), \nu_s(t), w_{rs}(t{-}1) = 0$

A.1. Alg. for source domain (primal variable updates):
Given the dual variables $\lambda_l(t)$, $\mu_{rs}(t)$, $\nu_s(t)$, sources solve problems (8) and (10) respectively as given in; (12) by direct differentiation and (13) by using gradient based methods with constraint $w_{\min} \le w_{rs}(t) \le 1$.

$$d_s(t)=\frac{d_s^{\text{pref}2}}{2}\sum_{r\in\mathcal{R}_s}\mu_{rs}(t) \tag{12}$$

$$w_{rs}(t)=\Big[w_{rs}(t{-}1)-\kappa_w\Big(\frac{2c_r x_s^2}{c_s^{\text{pref}2}}\sum_{p\in\mathcal{R}_s} c_p w_{ps}(t{-}1) + \nu_s(t)+x_s\sum_{l\in\mathcal{L}_r}\lambda_l(t)\Big)\Big]_{w_{\min}}^{1} \tag{13}$$

Entities $x_s.w_{rs}(t)$ and $\mu_{rs}(t)$ are communicated over routes $r \in \mathcal{R}_s$ in order to be used in A.2 by the link algorithm of intermediate domains.

A.2. Alg. for intermediate domain (primal and dual varable updates):
Each intermediate domain, for each of the outgoing overlay links l, solves problem (9) as in (14) for primal variable $f_l(t)$. Similarly dual variables $\lambda_l(t+1)$ that maximize (11) are iteratively solved by gradient based methods in (15) for primal variables $w_{rs}(t)$ and $f_l(t)$.

$$f_l(t) = b_l - \left|\sqrt{\frac{\sum_{r\in\mathcal{R}'_l}\sum_{s\in\mathcal{S}_r}\mu_{rs}(t)}{\lambda_l(t)}}\right| \tag{14}$$

$$\lambda_l(t+1) = \Big[\lambda_l(t)+\kappa_\lambda(\sum_{r\in\mathcal{R}'_l}\sum_{s\in\mathcal{S}_r} x_s w_{rs}(t)-f_l(t))\Big]^+ \tag{15}$$

While packets are traversing links, partial sums $\sum_{l\in\mathcal{L}_r}\lambda_l(t{+}1)$ and $\sum_{l\in\mathcal{L}_r} 1/(b_l{-}f_l(t))$ are calculated and communicated back to source.

A.3. Alg. for source domain (dual variable updates):
Dual variables $\mu_{rs}(t{+}1)$ and $\nu_s(t{+}1)$ that maximize (11) are iteratively solved by gradient based methods (16)-(17) for fixed primal variables $d_s(t)$, $w_{rs}(t)$ and $f_l(t)$.

$$\mu_{rs}(t+1) = \Big[\mu_{rs}(t)+\kappa_\mu(\sum_{l\in\mathcal{L}_r}\frac{1}{b_l-f_l(t)}-d_s(t))\Big]^+ \tag{16}$$

$$\nu_s(t+1) = \nu_s(t)+\kappa_\nu\{\sum_{r\in\mathcal{R}_s} w_{rs}(t)-1\} \tag{17}$$

A.4. Set time $t{=}t{+}1$, if primal and dual variables have not converged, continue with step A.1, otherwise stop.

Proof: By the descent lemma, convergence of the gradient-based optimization of a function is guaranteed if the function has the Lipschitz continuity property. And the function has Lipschitz continuity property if it's Hessian is bounded in l_2 norm. Let $\mathbf{H}$ denote the Hessian matrix for the objective function in problem (10), then the entries H_{ij} are calculated as follows. Indices are found by the function $g(.)$ as $i = g(r, s)$, $j = g(r', s')$ for any route-source pairs (r, s) and (r', s').

$$H_{g(r,s),g(r',s')} = \begin{cases}\frac{2c_r c_{r'} x_s x_{s'}}{c_s^{\text{pref}} c_{s'}^{\text{pref}}} & \text{, for } s = s' \\ 0 & \text{, otherwise}\end{cases} \tag{18}$$

$$\|\mathbf{H}\|_2 \le \sqrt{\|\mathbf{H}\|_1\|\mathbf{H}\|_\infty} \tag{19}$$

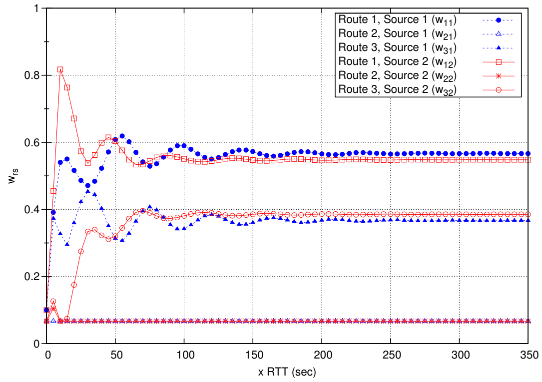

Fig. 2. Traffic distribution w_{rs} for the two source case in Fig. 1

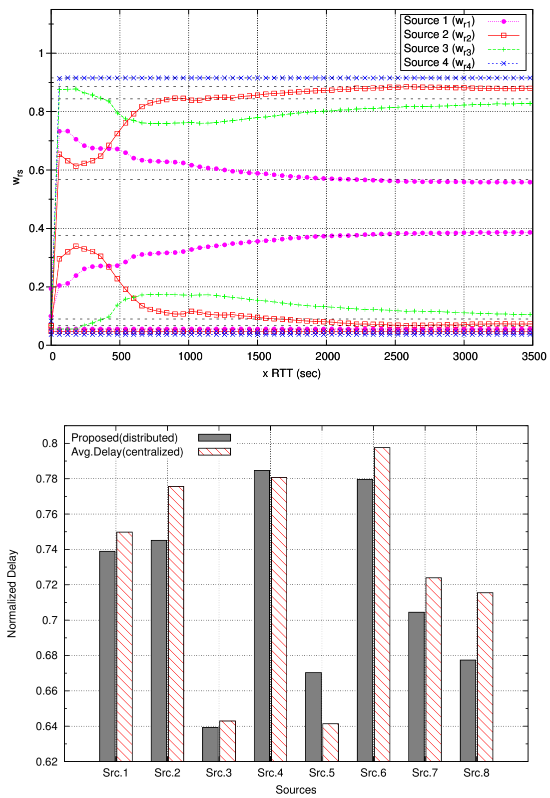

Fig. 3. For 50 node case (a) convergence behaviour of services (random four), (b) normalized delay for max. and average delay algorithms at optimum

The above inequality relates the Euclidian norm of matrix $\mathbf{H}$ with entries H_{ij} to the the maximum column-sum and row-sum matrix norms which are respectively defined as $\|\mathbf{H}\|_1 = \max_j \sum_{i=1}^{R} |H_{ij}|$ and $\|\mathbf{H}\|_\infty = \max_i \sum_{j=1}^{R} |H_{ij}|$

$$(L')^2 = \left[\max_{r,s} \Big(\sum_{r' \in \mathcal{R}_s} \frac{2c_r c_{r'} x_s^2}{c_s^{\text{pref}\,2}} \Big) \right]^2 \tag{20}$$

Step size κ_w should be selected as below for the gradient method with fixed step size to converge where ϵ is any small number satisfying $0 < \epsilon \leq \frac{2}{1+L'}$:

$$\epsilon \leq \kappa_w \leq \frac{2-\epsilon}{L'} \tag{21}$$

■

IV. Case Study

In the previous section we proposed a dual-based algorithm which decomposes the Lagrange dual of the problem in (1)-(5) into independently solvable subproblems as given in (8)-(10). After calculating Lagrange multipliers (11) (i.e. prices or dual variables) then the dual problem coordinates the subproblems by communicating prices back to the them. By means of the proper exchange of prices, the original problem is solved in a distributed way [6].

In A.1 of Table I, subproblems are solved for w_{rs} and d_s at the source nodes of s using (12)-(13). Similarly in step A.2 the intermediate nodes solve f_l using (14) for each of its links. Note that these subproblems are coordinated by dual variables which represent the price per unit use of related resource. Each overlay node updates in A.2 the price per unit use of its virtual links in (15). Similarly in A.3, source node of s updates the prices on its routes per unit delay per packet in (16) and per unit flow distribution in (17). Then these prices are communicated to the subproblems by simply piggybacking to the packets to minimize communication overhead. Hence, congestion prices λ_l are sent back to the service domains that have a route on that virtual link. Similarly, end-to-end delay price μ_{rs} of route r of service s is transmitted to all links on that route. For example if the the delay price of r of service s is increasing, which means end-to-end delay on r is getting higher than the service s requires, then the target utilization values f_l of links on r are decreased according to (14) and results in a lower delay.

To discuss the algorithm on an example, we use the simple topology in Fig. 1 with links $b_l = 1$ Mbps where there are $S = 2$ services with three routes each. For a set of two identical services with $x_s = 0.5$ Mbps, $d_s^{\text{pref}} = 0.054$ sec., $c_s^{\text{pref}} = 0.6$ unit price/pkt and $\mathcal{R}_s$ (in figure), we obtain w_{rs} as in Fig. 2. We note that the longest routes $(r, s) = (2, 1)$ and $(r, s) = (2, 2)$ are minimally used by both overlays 1 and 2 respectively. In this way min-max of the normalized delay values for all services are obtained. Note that while obtaining this solution, services respect the requirements (like delay) of other services (i.e. fairness). For example, although only a minimal amount of flow is assigned to w_{12} by overlay 2, overlay 1 does not dominate links 1 and 4 by increasing rates on w_{21} and w_{31} where *i)* w_{21} could be advantageous since links 4 and 5 are underutilized, *ii)* w_{31} could be better since it is the shortest path. However both of these actions will lead to an increased delay on route $(r, s) = (1, 2)$ and $(r, s) = (2, 2)$ of overlay 2. Due to the min-max nature of the problem in (1) which is as a result of the second order polynomial in the objective function, overlay 1 will avoid these paths in order to minimize the maximum delay of the services and mostly use w_{11}.

V. Simulation Results

In the simulations we fix virtual link bandwidths to $b_l = 1$ Mbps and service parameters to $x_s = 0.3$ Mbps, $d_s^{\text{pref}} = 0.144$ sec., $c_s^{\text{pref}} = 0.009$ unit price/pkt where packet size is 1500 bytes. For these fixed parameters we evaluate the algorithm

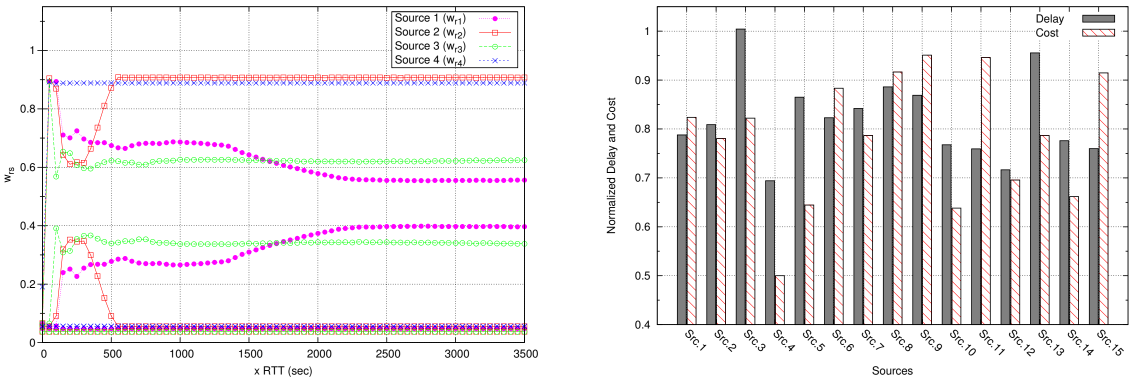

Fig. 4. For 100 node case (a) convergence behaviour of services (random four), (b) normalized delay and cost at optimum.

for randomly generated networks with $N = 50$, 100 nodes where each node represents a domain. We fix the number of routes for each service to $|\mathcal{R}_s| = 3$ where all of them are disjont paths per service s. We set κ_w to the maximum value using the bound given in Theorem 1 and we select sufficiently small values for step sizes κ_λ, κ_μ, κ_ν the dual updates.

In the first experiment we run the distributed algorithm in Table I for $N = 50$ nodes case with $|\mathcal{S}| = 8$ services. In order to keep the figure simpler, we randomly picked up four sources and plotted them only. In Fig. 3(a) we make two major obeservations. First, the distributed algorithm converges to the optimal values calculated by MATLAB using centralized optimization assuming global coordination. The dashed lines in the figure show the optimal values calculated by MATLAB. Second we observe, within few thousand RTT (iteration) times algorithm converges to the optimal values. By finding strict bounds on dual update step sizes, faster convergence could be achieved, however we leave it as a future work.

In the second part of the first experiment in Fig. 3(b), for each source we compare the normalized delay d_s/d_s^{pref} of the proposed algorithm to the normalized average delay minimization case. Note that our algorithm minimizes the maximum delay of all routes per service, whereas in the average delay case, the average delay of all three routes are minimized. Hence average delay algorithm can not give a bound on delay of routes. As it can be observed from the Fig. 3(b), in the overall, bounding the maximum delay (the proposed algorithm) yields a lower delay than the average delay case for services s. Furthermore average delay problem is not a convex optimization problem, and therefore solving it (without a duality gap) in a distributed way is not feasible without global coordination.

In the second experiment, we run the distributed algorithm for a randomly generated network with $N = 100$ nodes (domains) with $|\mathcal{S}| = 15$ services. Besides these values we keep the other parameters same and using Theorem 1 we set κ_w and some smaller values to κ_λ and κ_μ for the dual updates. We observe in Fig. 4(a) that the algorithm still converges in few thousand RTTs (iteration) and this indicates that the proposed distributed algorithm easily scales with the increasing network size, as long as the step sizes are properly selected. Finally in Fig. 4(b) we plot the normalized delay d_s/d_s^{pref} and the normalized cost $\sum_{r\in\mathcal{R}_s} c_r x_s w_{rs}/c_s^{\text{pref}}$ from the objective function. We observe from the figure that the proposed algorithm achieves the preferred delay and cost values (i.e. both normalized entities are always ≤ 1) by optimally distributing the total flow x_s on its routes.

VI. Conclusion

In this paper, after defining service as $\{\mathcal{R}_s, x_s, d_s^{\text{pref}}, c_s^{\text{pref}}\}$, we considered the service enginering problem in inter-domain overlay networks by finding the best distribution of the traffic such that the service rate, delay and the cost constraints of the services are met. We formulated the problem as a convex optimization problem, and proposed a distributed and scalable algorithm to solve it. The simulation results show the effectiveness and the scalability of the algorithm up to 100 domains.

As an immediate next step, strict bounds on all step sizes will be calculated and then the algorithm should be tested in larger networks. Furthermore, the proposed model is general enough and can be extended to different architectures like P2P networks and content-sharing networks.

References

[1] H. Zhang, J. Kurose, and D. Towsley, "Can an overlay compensate for a careless underlay?" in *IEEE INFOCOM 2006*, April 2006.

[2] Y. Liu, H. Zhang, W. Gong, and D. Towsley, "On the interaction between overlay routing and underlay routing," in *IEEE INFOCOM 2005*, vol. 4, 2005, pp. 2543–2553.

[3] L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker, "On selfish routing in internet-like environments," *IEEE/ACM Transactions on Networking*, vol. 14, no. 4, pp. 725–739, August 2006.

[4] X. Gu, K. Nahrstedt, R. N. Chang, and C. Ward, "QoS-assured service composition in managed service overlay networks," in *Proc. of the 23rd Int. Conference on Distributed Computing Systems*. ACM, May 2003.

[5] Z. Duan, Z. Zhang, and Y. Hou, "Service overlay networks: SLAs, QoS, and bandwidth provisioning," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, 2003.

[6] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *Journal of Operations Research Society*, vol. 49, no. 3, pp. 237–252, March 1998.

[7] X. Lin and N. B. Shroff, "Utility maximization for communication networks with multipath routing," *IEEE/ACM Transactions on Automatic Control*, vol. 51, no. 5, pp. 766–781, May 2006.

[8] D. Xu, M. Chiang, and J. Rexford, "Link-state routing with hop-by-hop forwarding can achieve optimal traffic engineering," in *Proc. IEEE INFOCOM*, April, Phoenix, AZ 2008.

[9] S. H. Low and D. E. Lapsley, "Optimization flow controli: Basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, 1999.