iREX: Efficient Automation Architecture for the Deployment of Inter-Domain QoS Policy

Ariffin Datuk Yahaya, *Student Member, IEEE,* Tobias Harks, *Student Member, IEEE,* and Tatsuya Suda, *Fellow, IEEE*

***Abstract*—The inter-domain Resource Exchange (iREX) architecture uses economic market mechanisms to automate the ad-hoc negotiation and deployment of end to end inter-domain quality of service policy among resource consumer and resource provider Internet Service Providers.**

In this paper, we explore iREX's network load distribution by comparing its performance to a lower bound for network congestion in two ways. We first present an analytical model of iREX in terms of an online algorithm and analyze its efficiency via competitive analysis. Our main result shows that the efficiency loss of iREX with respect to monetary cost is upper-bounded by a factor of $\frac{8K}{2K+1}$, where K is the number of deployments, provided affine linear price functions are used. When the price functions are used to model congestion in the network, this result implies upper bounds on the efficiency loss of iREX with respect to network congestion.

We then complement the analytical model with a numerical study using simulations. We compare the efficiency of the iREX architecture with optimal solutions derived from unsplittable and splittable multi-commodity flow optimization models. Our numerical results show that for nominal to high traffic loads of 40% or more, iREX deviates a maximum of about 20% from the lower bound, while the current method deviates a maximum of 300%.

***Index Terms*—Inter-domain; QoS policy; resource allocation and management; network control by pricing; economics; online optimization.**

I. Introduction

Managing an Internet domain's policy to offer Quality of Service (QoS) services to selected traffic flows is an important networking research area. Presently, any number of domain Internet Service Providers (ISPs) can create and install policies to selectively support multiple traffic flows with different QoS specifications within the domain(s) that they control. However, because no single ISP controls all the domains in the Internet, deploying end-to-end (E2E) inter-domain (ID) QoS traffic flows (*requirements*) must involve negotiating for, and propagating the policies to support the traffic flows' QoS specifications (QoS policy) with transit ISP domains.

Throughout this paper, we use the term *resource* as an abstract ID QoS network transport service defining ownership and transport responsibility starting from a domain's ingress border router, going through the domain and ending at a neighboring domain's ingress border router.

A. Inter-Domain QoS Policy Automation

Methods to automate inter-domain QoS deployment like Bandwidth Brokers [1]–[4], MESCAL [5] and CADENUS [6], [7] have been suggested. These methods assume that some sort of business level document (e.g. the Service Level Agreement (SLA) [8], [9]) has previously been manually negotiated defining the expectations and responsibilities of both resource consumer and resource provider domains. This manual negotiation process is very slow taking time in the order of days, making it impossible to negotiate for routes according to current network state, which may change on the order of seconds or minutes.

To overcome this manual negotiation problem, the current approach has been either 1) to pre-negotiate for all possible routes even if some (if not most) may never be used, or 2) to pre-negotiate for specific routes and limit deployments to those routes even if there may be others that could at times be better. Our work focuses on a third approach, where there is no pre-negotiation; instead, provider domains are allowed to decide whether or not to make their resource available in an ad-hoc manner, and consumer domains discover, select and negotiate for currently available resources also in an ad-hoc manner.

B. Automated Negotiation Issues

In order to fully automate E2E ID QoS policy negotiations in an ad-hoc manner, in addition to automating the deployment of a resource, domains must first be able to discover, select and negotiate for currently available resources. Since each domain's network resource is owned and managed by different ISPs, negotiating for a QoS policy across ID borders is complicated by the inter-related non-technological issues of "ownership" and "trust". Resource provider domains need to *trust* resource user domains to compensate them for the

Manuscript received April 30, 2007; revised January 3, 2008; approved by IEEE Transactions on Network and Service Management Editor Nikos Anerousis. A part of Ariffin Datuk Yahaya's and Tatsuya Suda's work is supported by the NSF through grants ANI-0083074, ANI-9903427 and ANI-0508506, by DARPA through grant MDA972-99-1-0007, by AFOSR through grant MURI F49620-00-1-0330, and by grants from the California MICRO and CoRe programs, Hitachi, Hitachi America, Hitachi CRL, Hitachi SDL, DENSO IT Laboratory, DENSO International America LA Laboratories, NICT (National Institute of Communication Technology, Japan), NTT Docomo and Novell.

A. Yahaya and T. Suda are with the School of Information & Computer Science, University of California, Irvine, CA 92697-3425 USA (e-mail: {ariffin, suda}@ics.uci.edu).

T. Harks is with the Mathematical Department of Technical University Berlin, Germany (e-mail: harks@math.tu-berlin.de).

Digital Object Identifier 10.1109/TNSM.2008.080105.

resource contributed towards supporting an E2E ID QoS policy, and resource user domains requiring deployment of ID QoS policy need to *trust* resource provider domains to perform as compensated. Competing autonomous domains would also find it difficult to agree on which would actually be in charge of any central entity, therefore the discovery, selection and negotiation process must be done in a fully distributed and autonomic manner.

Additionally, depending on a domain's connectivity within the Internet and its proximity to available resources, there may exist many potential choices of a resource that could be included in an ID QoS policy deployment path. These resources need to be chosen carefully with respect to increasing the number of possible coexisting ID QoS policies and also decreasing the overall congestion of the Internet.

C. Automated Policies with iREX

An automated method for negotiating and deploying E2E ID QoS policy in an ad-hoc manner has been proposed by Yahaya and Suda in the inter-domain Resource Exchange architecture (iREX) [10], [11]. iREX is a fully distributed architecture that empowers Internet domains to self-manage E2E ID QoS policy by enabling an autonomic system based on economics.

Precursors [10] and [11] to this paper have shown that one of the benefits of the iREX architecture is that it enables an increased number of coexisting ID QoS policies to be deployed compared to the current method while still maintaining a lower level of congestion. iREX achieves this lowering of congestion by efficiently taking advantage of unused resources along multiple routes for the same source-destination domain pair. In [12], Yahaya, Harks and Suda explored the issue of iREX's efficiency further by addressing the question "how does iREX's performance compare to centralized optimal solutions?"

In this paper, we detail and strengthen the work on the efficiency of iREX with respect to congestion started in [12] by incorporating an analytical analysis of the iREX architecture based on an *online multi-commodity routing* model. In this model, we prove upper bounds on the efficiency loss of the iREX architecture for *arbitrary* demands and *arbitrary* network topologies under the simplifying assumptions that: (i) reservations are non-expiring, and (ii) capacity constraints are inactive. Our measure of efficiency is defined with respect to total monetary deployment cost and total network congestion.

D. Our Analytical Contributions

Using an analytical methodology, we prove upper bounds on the efficiency loss of the iREX architecture for *arbitrary* demands and *arbitrary* network topologies. Our main analytical results show that the iREX architecture's efficiency loss:

- with respect to total deployment cost is not worse than $\frac{8\,K}{2\,K+1}$ times the optimal offline solution, where K is the number of demands, when non-decreasing affine price functions are used.
- with respect to network congestion is not worse than $\frac{16\,K}{2\,K+1}$ times the optimal offline solution, where K is the number of demands, when price functions are non-decreasing affine and congestion is modeled by the same price function.
- with respect to network congestion is not worse than $\sqrt{\frac{16\,K}{2\,K+1}}$ times the optimal offline solution, where K is the number of demands, when price functions are linear ($p(x) = x$) and congestion is measured by the L_2-norm of the vector of link loads.

E. Our Numerical Contributions

Since our analytical results are based on several simplifying assumptions, we also *empirically* investigate the efficiency of iREX with more realistic assumptions. Specifically, to achieve our numerical results, we consider link capacities and time windows for ID QoS demands, and allow for linear, squared, cubed, and randomly chosen price functions. In this setting, we compare time based snapshots of iREX network congestion simulation results with *numerically calculated* lower bounds specific to each snapshot.

Our main result with the numerical methodology shows that even though in the iREX architecture demands are routed *selfishly* and *online* (i.e. the chosen path is the cheapest and future demands are not known until the time they are routed), the efficiency loss (i.e. the percentage difference of the congestion experienced between iREX and the lower bound) is very low. The numerical results have partially appeared in [12].

F. Paper Organization

The next section (II) will explain the details of the iREX architecture. In Section III, we present bounds on the efficiency loss of the iREX architecture together with details of our assumptions. In Section IV, we complement the analytical results with numerical analysis.

II. iREX Architecture

The iREX architecture contains a broad set of resource advertisement, reservation, and reputation score maintenance protocols based on the "Posted Price Competition" economic model, see Abbink and Brandts [13]. In this model, providers independently choose prices that are publicly communicated to consumers on a take-it-or-leave-it basis. In the ID QoS context, domains are both resource providers and consumers at the same time. While domains have ID QoS resources that they can "sell", they also need to "buy" resources to deploy their own ID QoS requirements.

Domains that support the iREX protocols form a loose community (*iREX market*) existing for the sole purpose of trading in ID QoS network resources by registering offline with a clearinghouse[1] that acts as a bank to handle settlements for resources bought and sold within the iREX market. Members of the iREX market facilitate ID resource selection by maintaining information about the desirability of resources within the market, and by supporting the deployment of E2E ID QoS policy.

[1]Details of this clearinghouse will not be covered in this paper.

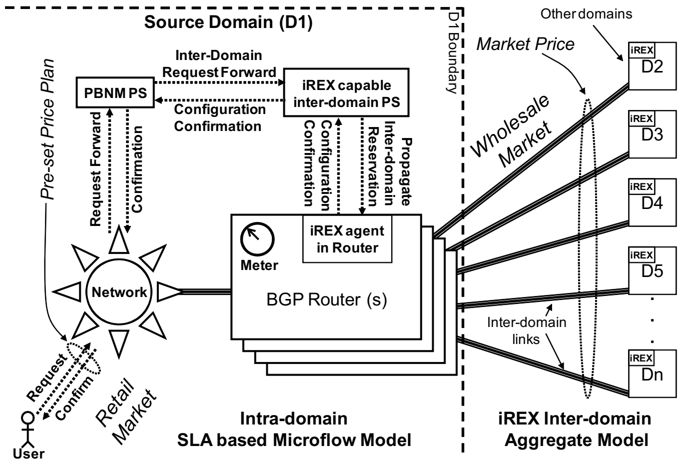


Fig. 1: iREX domain scenario.

A. *iREX Domain Scenario*

In iREX, each domain Di has its own retail market dealing in microflows [14] with its domain users and a wholesale market dealing in aggregated jumbo flows [15] facilitated by connecting to the Internet through n_i domain neighbors which in turn are also connected to n_j domain neighbors and so on. Fig. 1 illustrates this view with domain $D1$ having a retail market with flat pre-set pricing within its own domain, and dealing in the wholesale inter-domain (iREX) market with domains $D2$ to Dn_1 using market pricing.

Fig. 1 also illustrates how iREX fits into the IETF's Policy Based Network Management (PBNM) admission control architecture presented by Yavatkar *et al.* in [16]. Intra-domain QoS policy deployment is done by the PBNM Policy Server (PS) according to the respective domain's management practices without iREX. If the PS requires ID policies to be deployed, it will communicate the requirements to the iREX-capable ID Policy Server (iIDPS). The iIDPS then uses the iREX reservation protocol to negotiate for and deploy the ID QoS policy. If there are ID QoS policies from another domain that require intra-domain routing either to transit or terminate within the domain, the iIDPS will in turn communicate this to the PS. iREX inter-domain signaling is facilitated by the iREX agent in the Border Gateway Protocol (BGP) [17] router. BGP routers will also have access to traffic meters similar to those introduced by Brownlee *et al.* in [18] to enable flow policing by a provider within a domain and meter reading by consumer domains from outside the domain. iREX uses standard TCP for signaling, so inter-domain messages between iIDPSs rely on BGP routing.

While iIDPS in neighboring domains form connections to each other, any iIDPS may directly send messages to any other iIDPS. This capability is important due to the "loose" assumptions of an iREX market where domains have registered with a clearinghouse and agreed to use the iREX protocols to participate in the market, but may yet to have any other relationship. The iREX architecture allows each consumer domain to create ad-hoc bilateral relationships with each and every provider domain along its deployed ID QoS path; thus

- empowering the source domain to directly make resource selection decisions without a proxy,
- increasing transparency by making each provider domain accountable directly to the source domain, and
- introducing new fault recovery possibilities where intermediate domains may now suggest recovery solutions directly to the source.

The iREX architecture's "direct" nature is in contrast with the use of proxies at ISP borders when using SLAs. Provider domains that negotiate agreements using proxies promise ID resources which are not under their direct control. The use of proxies also weakens accountability and limits an intermediate domain's access to the source domain and vice versa – an example of a problem this causes is when an ID resource involved in a QoS policy fails and a new ID path to the destination converges through the normal process of BGP; in this case, there is no guarantee that ISPs along the newly converged route (that were not along the initial route) will honor the QoS policy negotiated for in the initial SLA.

B. *Selling: iREX Path Vectors*

iREX domains use "real" *resource price* in the form of "*monetary unit* per *time unit* per *bundle of resource*" and *resource reputation score* in the form of "*number of complaints* against a resource" to evaluate resource *desirability*.

Domains choose desirable resources to form chains of ID resources leading to destination domains in the form of path vectors (*iREX path vectors*). iREX path vectors are similar to those used in BGP, but with two differences. The first difference is that instead of using minimum ID hops as an evaluation metric, iREX path vectors use the current *total price* per unit time. The total price per unit time for an ID QoS policy to the destination is determined by adding up the component prices of all the ID resources used in the iREX path vector to that destination. The second difference is that since iREX uses source routing, the iREX path vectors are used only as hints, making total network convergence for an iREX path vector unnecessary.

To sell transport services on an iREX resource, a provider domain decides for each of its own iREX resource a minimum price that the provider domain is willing to accept from other domains to reserve the resource, and the maximum reservation duration that it is willing to accept at that price. A provider domain decides on the selling price for its resources based on current market prices and the inherent risk of providing service on that resource. Market prices are dependent on a resource's location in reference to other similar resources (supply) and the distribution of resource consumers and their destinations (demand) within the network topology. Risk is dependent on the provider domain's previous commitments on its resources as domains are expected to use *statistical multiplexing* (i.e. failure on previously negotiated QoS deployments due to multiplexing collisions will be detrimental to the domain.)

After deciding on prices for its resources, a provider domain incorporates these prices into the current cheapest known iREX path vectors before advertising the path vectors to its neighbors. All prices are public once advertised; the public nature of these prices is necessary in iREX to provide information about a resources' desirability and demand. Every price quoted by a provider domain is also accompanied by a

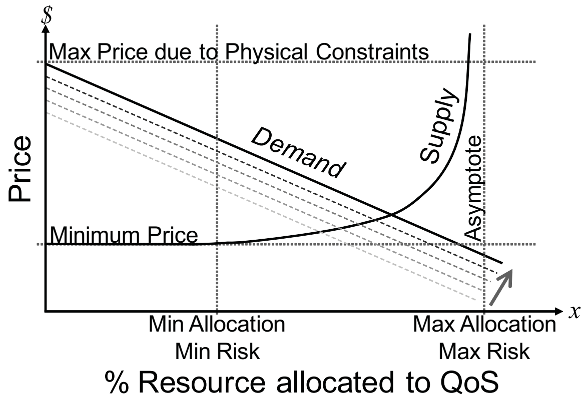


Fig. 2: iREX pricing heuristic example.

price version code that identifies when the quote was made. Additional security measures to promote non-repudiation and authentication have been considered through the use of asymmetric keys but is not covered here.

Domains receiving these advertisements evaluate and filter the received iREX path vectors by first excluding those that use resources with reputation scores worse than the particular domain's tolerance. Domains then select the path vectors with the cheapest total price to their corresponding destination domains.

After this filtering process, domains incorporate their own resource prices into the selected paths for their next advertisement. iREX advertisements are done in reaction to a change in price information and also implements Hold Time and MinRouteAdvertisementInterval timers [17] similar to BGP to keep the minimum advertisement interval on the scale of minutes.

By the repeated advertisement of known iREX path vectors and the filtering of received advertisements, iREX path vectors formed using the cheapest reputable ID QoS resources propagate to all domains within the iREX market. As we will see later in this section, forming path vectors with the cheapest total price also translates into choosing path vectors that use underutilized resources.

As mentioned earlier, each iREX provider domain has to decide prices for its resources, and in Section III-E we will show that determining optimal values for these resource prices is an $\mathcal{NP}$-hard optimization problem even when only two providers exists in the network. This "hardness" stems from the bi-level nature of determining optimal prices. Every provider has to set and advertise prices on its own links, while other domains filter the advertisements to select the cheapest paths with respect to these set prices. Each provider must strike the right balance between low prices that allow inclusion into many iREX price vectors that will generate high reservation volume but low revenue, and high prices which could also result in low revenue because the provider's resources may only be included in a few iREX path vectors (or be excluded entirely) due to being expensive. As noted in [19], bi-level programs are generally non-convex and non-differentiable – to all practical extent, intractable.

In light of this hardness result, we expect domains to use heuristics for pricing their resources.

Fig. 2 illustrates a simple pricing heuristic in the form of a price function. The price function models a provider domain's basic physical constraints when allocating resources for trading in response to the associated risks. When an iREX market deems a resource desirable by increasing the price that it is willing to pay for the resource, the demand curve for the resource moves up the supply curve between the $MinRisk$ and $MaxRisk$ points on the x axis. The provider domain of this desirable resource will then respond by demanding a higher price for the resource in compensation for the risks they are experiencing. Less desirable resources will be underutilized and cheaper.

The iREX architecture assumes that domains will view resources as an economic good, as is illustrated by the non-decreasing supply curve in Fig. 2. An iREX domain that does not conform to this assumption (i.e. views its resource as an economic bad) will have non-increasing supply curves that will at first increase the domain's number of reservations as the price of competing resources increase; but then when the reservations cannot be fulfilled due to lack of resource, the domain's reputation will worsen and the iREX market will exclude the domain. iREX's reputation system will be covered in Section II-D.

Domains wanting to leave an iREX market (e.g. during busy or crisis periods) can transmit iREX "withdraw" messages to its neighbors. Once withdrawn, a domain no longer participates in propagating iREX path vectors. To rejoin the iREX market, a domain has only to start advertising its resources again.

The perpetuation of the iREX path vector as a means of resource discovery is a key concept in iREX. This path vector determines the information made available to consumer domains about available resources and directly determines which provider domains receive deployment revenues. Domains participate in the perpetuation of iREX path vectors through selfish self-interest to ensure that their own resources remain in the path vector, and therefore in the iREX market.

iREX assumes that each different QoS traffic specification, which may include a minimum available resource duration, is standardized into its own resource commodity. The QoS traffic specification will determine the metric used to evaluate the commodity, and the minimum available resource duration will determine a domain's temporal policy deployment behavior (i.e. short, medium or long term) when using that commodity. While this paper may use bandwidth as a metric, any pre-agreed metric may be used for an iREX market since only the price is advertised, and not the metric.

The use of commodity prices to represent information about the scarcity of resources affords iREX the flexibility to be applied to many QoS specifications. Multiple iREX markets may independently exist, each specializing in a particular QoS traffic specification and maintaining a set of iREX path vectors formed using resources deemed to be most desirable for that specification.

C. Buying: Resource Reservation

iREX consumer domains have total autonomy to determine and manage their own ID QoS policy deployment path(s). To buy transport services for deploying an ID QoS policy using iREX, a source domain first identifies the entire ID path and initiates negotiation with the domains along the identified path through reservation request signaling. Initiating this reservation request implies a commitment to pay the price that each resource provider along the identified ID path has requested for the use of their advertised resource. Responding positively to the initiated reservation request signal implies an agreement by the provider of a resource to deploy network support and actively participate to maintain the policy referred to by the reservation request for the full duration of time that the resources have been reserved.

A source domain primarily identifies an ID path by referring to the current iREX path vector(s) to the destination and selecting a path with an acceptable price. Secondarily, a consumer domain may identify a path with an acceptable price by either constructing a path from recent historical advertisements or by initiating an iREX path request message to its neighbors and selecting a path from the resulting information. If a reservation is initiated using an iREX path vector with stale information, iREX allows resource provider domains to directly update the consumer domain with current information during the reservation process – at which time a consumer may reconsider its resource choices and deploy on another ID path. A consumer domain may also choose to split his requirement and deploy on multiple ID paths. Expiring deployed reservations can be extended at the most current total real resource price for the path. Since iREX domains makes greedy (i.e. cheapest) deployment path decisions without knowing about future deployments, iREX's process for identifying a deployment path can be abstracted as an *online* algorithm.

Resource prices quoted and advertised by provider domains can fluctuate continuously leading to dynamics in the iREX path vector, but a successful reservation request freezes the reserved ID path and its associated prices for the duration of the reservation. This "freezing" is necessary to avoid the instability that may happen if successful reservations follow the dynamics of the iREX path vector. Multiple ID QoS policies using different paths may be concurrently deployed for the same source destination pair depending on the iREX path vector at the time of each policy's reservation.

Within an iREX market, an ID QoS policy is uniquely identified by the combination of the source domain's Autonomous System (AS) number and a unique deployment code generated by the source domain. Provider domains will associate each policy with a source domain selected ID path and QoS specification. This allows iREX to have multiple policies deployed at the same time, each using an independent path between source and destination domain pairs. iREX deployments for the same source destination domain pair may use different paths depending on the iREX path vector at the time of a policy's deployment.

Fig. 3 illustrates how a source domain sends information about a resource reservation to the destination domain in iREX. The source domain supplies a unique deployment code, destination, quantity reserved, reservation duration and sufficient ID path information to the next hop domain so that the reservation may be relayed along the source domain's chosen path. Each price that the source domain uses for the

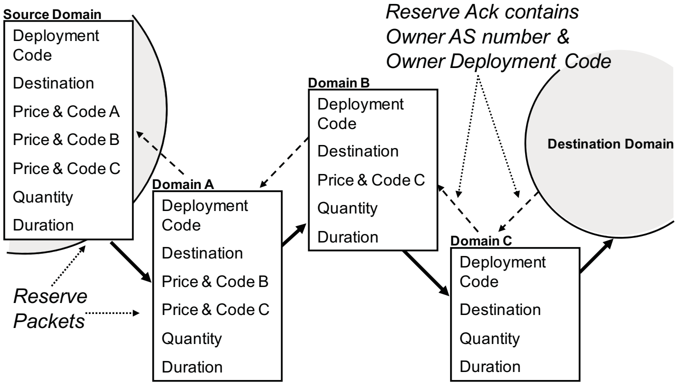


Fig. 3: iREX reservation information passing.

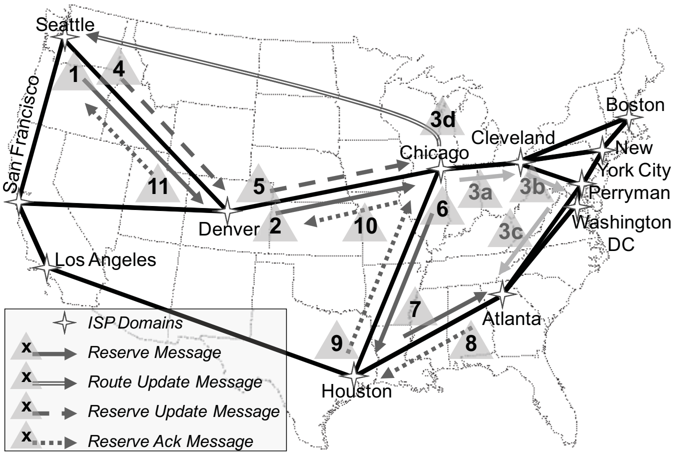


Fig. 4: Example reservation in the vBNS network topology.

reservation is accompanied by the price's version code. Upon receiving a reservation message, a provider domain retrieves its own price information and checks it for correctness; if satisfied, the domain then checks on its resource availability. Once the domain has ascertained that it is able to support the reservation, it deletes its price information from the reservation message and relays the resulting reservation message to the next hop domain – this continues until the destination domain is reached and an acknowledgment is sent in the reverse direction, at which time the policy is installed.

Fig. 4 illustrates an example of iREX resource reservation signaling using the Very High Performance Backbone Network Service (vBNS) topology with each point of presence representing an ISP domain. The source domain (Seattle) is trying to set up ID QoS policy to Atlanta on the ID path Seattle $\mapsto$ Denver $\mapsto$ Chicago $\mapsto$ *Cleveland* $\mapsto$ *Perryman* $\mapsto$ Atlanta and messages 1 and 2 are Reserve messages. When the Reserve message arrives at Chicago, the reservation is supposed to continue with Reserve messages 3a, 3b and 3c to Atlanta, but Chicago has newer information than the source domain and decides to suggest another route through Houston to the source domain by sending Route Update message 3d instead. In this case, source domain Seattle decides to accept Chicago's suggestion, resulting in Reserve Update messages 4 and 5 first informing Denver of the change in routing, and then informing Chicago to use the suggested path to continue the reservation, resulting in Reserve messages 6 and 7. The ID QoS policy is then successfully deployed with Reserve

Ack messages 8, 9, 10, and 11, with resultant ID QoS path Seattle $\mapsto$ Denver $\mapsto$ Chicago $\mapsto$ *Houston* $\mapsto$ Atlanta.

In [11], our simulations have shown iREX's worse case reservation time for the vBNS topology to be about 500 milliseconds with the source and destination domains being about 6,000 kilometers apart; the average total control packet overhead for this simulation was 5 packets per successful reservation including advertising packets.

Reservation signaling may fail due to many reasons including changing network conditions and technical difficulties experienced by domains. To mitigate these problems, all reservations have a time-out mechanism where the consumer domain sets a target time for reservation completion. Upon reaching the target time without completing the reservation, the consumer domain has the autonomy to decide on its course of action, which could be to try a reservation on a different path, or to retry the same path. Continued reservation failure after multiple tries on the same path may lead to the consumer domain complaining to the iREX reputation system. iREX's reputation system will be covered in the next (II-D) subsection.

Provider domains agreeing to support ID QoS policy deployed with iREX actively maintain the deployed policy and participate in its recovery from faults. Fault recovery is achieved either by suggesting to reroute a failed deployment through other available paths, or by signaling the fault to an upstream neighboring domain (i.e. towards the direction of the source domain) so that another intermediate domain involved in the deployment may attempt a recovery. If an intermediate domain decides that a recovery is possible, it will contact the source directly to suggest an alternate route. Should a network fault occur, a source domain will either receive a suggestion to reroute or a fault signal, and can then make a decision on recovering the policies affected.

Domains support the deployment of E2E ID QoS to earn revenue, and maintain good quality service to maintain untarnished reputations so that they are not blocked from supplying the market.

D. Domain Conformance: Reputation Score

iREX uses a reputation system to evaluate an iREX market's cumulative perception of a resource's ability to conform to its service expectations. The reputation system is intended only for the selfish self-preservation of an iREX market. Resources may be non-conforming, fall into ill repute, and then be excluded for a variety of reasons including simple system failure or while experiencing a denial of service (DOS) attack. The effect of each complaint expires with time, and we expect domains to set this expiration time in the order of an hour.

A reputation score is a tally of the current number of unique complaints against a resource. Each iREX market maintains a distributed and redundant database where member domains affected by "bad" resource can register complaints. Multiple complaints against a resource from the same domain (i.e. non-unique) will only count as one complaint, but will reset the specific complaint's expiration timer. Each domain has the autonomy to decide on a preference for a reputation score threshold count, which if exceeded will cause the resource to be excluded from that domain's outgoing iREX path vector advertisements – thus precluding that resource from selling any transport services. We expect domains to set their threshold to be between 5 to 10 complaints. Domains also have the autonomy to independently use other heuristics to determine which resources to include in its iREX path vector advertisements.

To find out where complaints will be sent and registered, or where a reputation score request must be queried to, iREX uses a well known function at each domain that, given the two AS numbers that a resource connects, produces three AS numbers where the complaint should be kept – this redundancy makes it possible to use voting to minimize the effects of tampering and collusion.

To complain about a bad resource, a domain first needs to identify the offending resource. One approach to achieve this is by deploying multiple test reservations to each domain along the current cheapest path. Each reservation can then be monitored on both the satisfactory completion of the reservation signaling process and the "within specification" metered performance of the deployed reservation. In [11], our simulations have shown the reputation system to be about 70% effective in avoiding the use of the bad resource within 3 minutes of a resource becoming unresponsive.

Domains support the maintenance of the reputation system because a domain's revenue depends on advertising good resources.

E. Congestion Avoidance

Lower demand for an ID QoS resource will result in an abundance of that resource, which is accompanied by a lowering of the risk of impacting the domain's current customers. Additionally, lower demand results in lower revenue as domains in the iREX market use cheaper alternative resources. To increase revenue, the domain will seek to increase the demand for its ID resources by lowering the price of its resource.

Higher demand for an ID QoS resource will result in resource scarcity, which is accompanied by the risk that a domain's current customers will receive decreased QoS and complain, so the domain will seek to be compensated for the increased risk by increasing the price of its ID resources. Increasing the price also effectively decreases demand.

Lower prices accompany lower resource use, and higher prices accompany higher resource use, therefore choosing reputable resources that are cheaper translates directly to choosing conforming resources that are less congested. In this manner, economics and reputation are used to dynamically change iREX path vectors to include the cheapest reputable resources, which also translates into the least congested conforming resources.

F. iREX Data Plane Requirements

iREX uses economics to automate E2E ID QoS policy on the management and control planes, imposing three requirements on the data plane of each participating ISP's domain:

(i) flow identification to support each iREX reservation deployment,

(ii) measurable (metered) QoS differentiation of flows to support an iREX market's service level assumption, and
(iii) reconfigurable ad-hoc routing for flows to a next hop domain to support iREX's ad-hoc reservations using consumer domain selected ID paths. Note that flows to the same destination domain may have different next hop domains.

We note that the current Internet does not yet fully support these requirements. Work on these issues is the subject of current research and not covered in this paper.

III. Bounding the Efficiency Loss of iREX

As noted in the Section II-C, the iREX architecture can be interpreted as an *online* routing algorithm, which routes every demand along the currently cheapest (shortest) path. That is, iREX solves a min-cost flow problem for every demand without considering future demands. Our goal is to investigate the *efficiency* of iREX with respect to (1) monetary cost and (2) network congestion. In the first variant, we compare the solution of iREX to an *offline* solution with minimum total cost. In the second variant, we compare the solution of iREX to an offline solution with minimum network congestion. In particular, for the second variant, we discuss and characterize the conditions on price functions leading to an efficient routing. Note that for obtaining an offline optimum, all demands have to be known *a priori*. This type of efficiency analysis is called *competitive analysis* coming from the online optimization field; see Borodin and El-Yaniv [20] for an overview of this concept.

To keep the mathematical analysis tractable, we make several simplifying assumptions. We consider a sequence of non-expiring demands that are released one after another and have to be routed in an online fashion: by the time demand i has to be routed, no information about future demands is available. Furthermore, we assume fixed non-decreasing price functions and assume that capacity constraints on the network links are inactive for any routing decision of the demands. Even though the assumption of fixed price functions may appear quite restrictive, we show in Section III-E that the problem of determining optimal prices for a resource provider is $\mathcal{NP}$-hard.

A. Network Model

We use standard notation for the considered online multi-commodity routing problem, see also Harks *et al*. [21]. An instance of the *Unsplittable Online multi-commodity Routing Problem* (OnlineUMCRP) consists of a directed network $D = (V, A)$ and continuous and non-decreasing price functions $p_a : \mathbb{R}_+ \to \mathbb{R}_+$ for each link $a \in A$. Every node $v \in V$ (or a partition $W_i \subseteq V, i = 1, \dots, n$) represents a provider domain. The price functions define the price of reserving capacity on a link depending on the current load. Furthermore, a sequence $\sigma = 1, \dots, K$ of commodities must be routed one after the other. We assume that $K \geq 2$ and denote the set of commodities by $[K] := \{1, \dots, K\}$. Each commodity $k \in [K]$ has a demand $d_k > 0$ that has to be routed from a source $s_k \in V$ to a destination $t_k \in V$. When we speak of a sequence $\sigma = 1, \dots, K$ of commodities, we refer to the full specification $(d_1, s_1, t_1), \dots, (d_K, s_K, t_K)$.

The routing decision for commodity k is *online*, that is, it only depends on the routings of commodities $1, \dots, k-1$. Once a commodity has been routed it remains unchanged, thus modeling iREX's behavior of "freezing" the reserved ID path as mentioned in subsection II-C.

A routing assignment, or *flow*, for commodity $k \in [K]$ is a nonnegative vector $\boldsymbol{f}^k \in \{0, d_k\}^A$, where every entry f_a^k describes the load of commodity k on link a. This flow is *feasible* if for all $v \in V$

$$\sum_{a \in \delta^+(v)} f_a^k - \sum_{a \in \delta^-(v)} f_a^k = \gamma(v), \tag{1}$$

where $\delta^+(v)$ and $\delta^-(v)$ are the links leaving and entering v, respectively; furthermore, $\gamma_k(v) = d_k$ if $v = s_k$, $\gamma_k(v) = -d_k$ if $v = t_k$, and $\gamma_k(v) = 0$ otherwise. We say that $(\boldsymbol{f}^1, \dots, \boldsymbol{f}^K)$ form a *multi-commodity flow*. Note that the condition $f_a^k \in \{0, d_k\}$ accounts for an *unsplittable routing*. We define $\mathcal{F}_k$ with $k \in [K]$ to be the set of vectors $(\boldsymbol{f}^1, \dots, \boldsymbol{f}^k)$ such that $\boldsymbol{f}^i$ is a feasible flow for commodity $i = 1, \dots, k$. If $(\boldsymbol{f}^1, \dots, \boldsymbol{f}^k) \in \mathcal{F}_k$, we say that it is *feasible* for commodities $1, \dots, k$. The entire flow for the sequence $1, \dots, K$ of commodities is denoted by $\boldsymbol{f} = (\boldsymbol{f}^1, \dots, \boldsymbol{f}^K)$. The aggregated flow on link a is defined as $f_a := \sum_{k \in [K]} f_a^k$.

The cost for routing demand d_k is defined as

$$C^k(\boldsymbol{f}^k; \boldsymbol{f}^1, \dots, \boldsymbol{f}^{k-1}) := \sum_{a \in A} p_a\big(\sum_{i=1}^{k} f_a^i\big)\, f_a^k. \tag{2}$$

In this context, we will sometimes write $C^k(\boldsymbol{f}^k)$ to stress the fact that we consider $f_a^1, \dots, f_a^{k-1}$ as fixed. We assume that once the demand d_k is routed, the prices along the used path are updated. As a result, future demands will possibly be routed over different cheaper paths, balancing the load.

The total cost of a flow $f \in \mathcal{F}_K$ is given by

$$C(\boldsymbol{f}) = \sum_{k=1}^{K} C^k(\boldsymbol{f}^k; \boldsymbol{f}^1, \dots, \boldsymbol{f}^{k-1}).$$

A feasible flow $\boldsymbol{x}^* \in \mathcal{F}_K$ that minimizes $C(\boldsymbol{f})$ is called the *offline optimum*. In order to compute the offline optimum, the entire demand sequence has to be known a priori. According to the iREX architecture each demand is routed selfishly along the cheapest path without taking future demands into account. Hence, given a feasible flow $\boldsymbol{f}^1, \dots, \boldsymbol{f}^{k-1} \in \mathcal{F}_{k-1}$, the flow $\boldsymbol{f}^k$ for the kth commodity is determined by the solution of the following program:

$$\min_{\boldsymbol{f}^k \geq 0} C^k(\boldsymbol{f}^k), \tag{3}$$

subject to the canonical flow constraints (1). The above problem can be computed in polynomial time since it can be reduced to a shortest path problem, see [22]. We denote the greedy online algorithm, which solves problem (3) upon arrival of commodity k by iREX.

B. Characterizing the Online Algorithm iREX

In the following, we characterize iREX in terms of its optimality condition for choosing the cheapest path. This inequality captures necessary and sufficient optimality conditions of problem (3).

Lemma 3.1: Let $\boldsymbol{x} \in \mathcal{F}_K$ be any feasible flow. A flow $\boldsymbol{f} = (\boldsymbol{f}^1, \ldots, \boldsymbol{f}^K)$ is generated by iREX if and only if for all $k \in [K]$ the following condition is satisfied:

$$\sum_{a \in A} p_a\Big(\sum_{i=1}^{k} f_a^i\Big) f_a^k \leq \sum_{a \in A} p_a\Big(\sum_{i=1}^{k-1} f_a^i + x_a^k\Big) x_a^k. \qquad (4)$$

This inequality stresses the fact that iREX produces a flow $\boldsymbol{f}^k$ for demand k with less cost $C^k(\boldsymbol{f}^k)$ compared to any other feasible flow $\boldsymbol{x}^k$, *given* previous routings $\boldsymbol{f}^1, \ldots, \boldsymbol{f}^{k-1}$ of iREX.

Remark 1: The above Lemma also holds if we replace the unsplittable flows $\boldsymbol{f}$ and $\boldsymbol{x}$ with a splittable flows $\boldsymbol{f}'$ and $\boldsymbol{x}'$ such that $\boldsymbol{f}'$ is a feasible splittable flow minimizing $C^k(\boldsymbol{f}^k)$ and $\boldsymbol{x}'$ is an arbitrary feasible splittable flow.
Summing inequality (4) over $k \in [K]$ yields

$$\sum_{k \in [K]} \sum_{a \in A} p_a\Big(\sum_{i=1}^{k} f_a^i\Big) f_a^k \leq \sum_{k \in [K]} \sum_{a \in A} p_a\Big(\sum_{i=1}^{k-1} f_a^i + x_a^k\Big) x_a^k.$$

Using an instance presented in Harks *et al.* [21], it can be shown that if the offline optimum can split the flow, no unsplittable online algorithm is competitive, even for linear price functions.

C. Efficiency Loss–Competitive Analysis

For a given sequence of commodities $\sigma = 1, \ldots, K$ and a solution $\boldsymbol{f}$ produced by an online algorithm ALG, we denote by $\text{ALG}(\sigma) = C(\boldsymbol{f})$ its cost. The online algorithm ALG is called (strictly) *c-competitive*, if the cost of ALG is never larger than c times the cost of an optimal offline solution. The *competitive ratio* of ALG is the *infimum* over all $c \geq 1$ such that ALG is c-competitive, see for instance Borodin and El-Yaniv [20] and Fiat and Woeginger [23]. We make use of the following identity:

$$(f_a)^2 = \sum_{k=1}^{K} \sum_{i=1}^{K} f_a^i f_a^k = 2 \sum_{k=1}^{K} \sum_{i=1}^{k} f_a^i f_a^k - \sum_{k=1}^{K} (f_a^k)^2. \qquad (5)$$

Theorem 3.2: Let the price functions $p_a(z) = q_a z + r_a, \; q_a, r_a \geq 0$ be affine linear for all $a \in A$. Then, the competitive ratio of the online algorithm iREX for the ONLINEUMCRP is bounded by $\frac{8K}{2K+1}$, where K denotes the number of commodities. Furthermore, for $K \in \mathbb{N} \cup \{\infty\}$, iREX is 4-competitive.

Proof: We start with the cost of the flow $\boldsymbol{f}$ produced by iREX.

$$\begin{aligned} C(\boldsymbol{f}) &= \sum_{k \in [K]} \sum_{a \in A} q_a\Big(\sum_{i=1}^{k} f_a^i\Big) f_a^k + r_a f_a^k \\ &\leq \sum_{a \in A} \sum_{k \in [K]} q_a\Big(\sum_{i=1}^{k-1} f_a^i + x_a^k\Big) x_a^k + r_a x_a^k \qquad (6) \\ &\leq \sum_{a \in A} q_a f_a x_a + \sum_{k \in [K]} q_a x_a^k x_a^k + r_a x_a^k. \end{aligned}$$

Inequality (6) follows from Lemma 3.1 and the last inequality follows since price functions are non-decreasing. Next, we add and subtract $\sum_{a \in A} q_a x_a^2$, which yields

$$C(\boldsymbol{f}) \leq \sum_{a \in A} q_a\big(f_a - x_a\big)x_a + q_a x_a^2 + \sum_{k \in [K]} q_a x_a^k x_a^k + r_a x_a^k.$$

The first expression in the sum can be bounded by $\frac{q_a}{4} f_a^2$, because $q_a\left(\frac{1}{2} f_a - x_a\right)^2 \geq 0$:

$$C(\boldsymbol{f}) \leq \sum_{a \in A} \frac{1}{4} q_a f_a^2 + q_a x_a^2 + \sum_{k \in [K]} q_a x_a^k x_a^k + r_a x_a^k.$$

Using equation (5), we can write

$$C(\boldsymbol{f}) \leq \sum_{a \in A} \sum_{k=1}^{K} \sum_{i=1}^{k} \frac{1}{2} q_a f_a^i f_a^k - \frac{1}{4} q_a (f_a^k)^2 + 2\, q_a x_a^i x_a^k + r_a x_a^k.$$

To bound the term $\sum_{k \in [K]} \frac{1}{4} q_a (f_a^k)^2$ from below, we use the inequality of Cauchy-Schwarz as follows:

$$\langle \boldsymbol{f}_a, \mathbb{1} \rangle^2 \leq \|\boldsymbol{f}_a\|^2 \cdot \|\mathbb{1}\|^2 \quad \Leftrightarrow \quad \frac{1}{K}\Big(\sum_{k=1}^{K} f_a^k\Big)^2 \leq \sum_{k=1}^{K} (f_a^k)^2,$$

where $\mathbb{1}$ is the vector of all ones and $\boldsymbol{f}_a := (f_a^1, \ldots, f_a^K)$. Then, adding $\sum_{a \in A} \left(\frac{1}{2} f_a - \frac{1}{4K} f_a + x_a\right) r_a \geq 0$ yields:

$$C(\boldsymbol{f}) \leq \frac{1}{2} C(\boldsymbol{f}) - \frac{1}{4K} C(\boldsymbol{f}) + 2\, C(\boldsymbol{x}).$$

Rewriting and taking $\boldsymbol{x}$ as the optimal offline solution proves the first claim. Taking the limit $K \to \infty$ proves the second claim. ■

The above theorem establishes the first constant factor bound for the unsplittable online multi-commodity routing problems with affine linear price functions.

Remark 2: Since Lemma 3.1 also holds for the splittable variant of iREX (see Remark 1) the competitive ratio of the online algorithm iREX with splittable flow is also bounded by $\frac{8K}{2K+1}$, where K denotes the number of commodities. Furthermore, for $K \in \mathbb{N} \cup \{\infty\}$, iREX is 4-competitive in this case.

The above theorem shows that iREX routing strategy for deploying E2E ID QoS is provably efficient in terms of monetary deployment cost. In the following, we investigate the implications of this result on an alternative performance metric – network congestion.

D. Implications for Network Congestion

Besides monetary cost, network congestion is also an important metric for E2E ID QoS deployment as suggested by Yahaya and Suda [10], [11]. Our goal in this section is to investigate conditions on price functions, under which the efficiency loss of iREX with respect to network congestion is bounded. We will first introduce network congestion models and then characterize such conditions, which induce an efficient network load distribution of the iREX architecture.

Congestion in transportation networks is usually modeled by non-decreasing congestion functions ℓ_a for each arc $a \in A$. These functions are typically nonlinear, positive, and strictly increasing with flow, see Patriksson [24] and Fortz and Thorup [25]. In practical applications, the most frequently used functions are polynomials, whose degrees and coefficients are determined from real-world data through statistical evaluation methods. The total congestion cost for a flow $\boldsymbol{f}$ is defined as $\Phi(\boldsymbol{f}) = \sum_{a\in A} \ell_a(f_a)\, f_a$. The idea is that it will be cheap to send traffic over an underutilized arc, but as the load on the arc increases the cost for this arc will grow superlinearly – penalizing high congestion. Hence, minimizing convex load dependent cost functions are well suited to balance the load in a network, see also Fortz and Thorup [25]. An unsplittable flow that minimizes congestion in a network solves the following optimization problem:

$$(P) \qquad \min \quad \Phi(\boldsymbol{f}),$$

subject to the standard flow constraints. We denote the splittable variant by SP. A similar congestion metric, which is used for load balancing problems in the context of machine scheduling is the L_p-norm of the vector of the link states, see Awerbuch *et al*. [26]. An unsplittable (splittable) flow that minimizes the L_p-norm ($p \in \mathbb{N}$) of the link loads solves

$$(L_p) \qquad \min \quad L_p(\boldsymbol{f}) = \big(\sum_{a\in A} (f_a)^p\big)^{1/p}.$$

Here, we denote by SL_p the splittable variant. An important special case of the L_p-norm is the case, where $p \to \infty$. In this case, the problem of minimizing the L_∞-norm corresponds to minimizing the most congested arc:

$$(L_\infty) \qquad \min \quad L_\infty(\boldsymbol{f}) = \max_{a\in A} f_a,$$

Again, we denote the splittable variant by SL_∞. It is well known that unsplittable multicommodity flow problems with convex objectives are $\mathcal{NP}$-hard as shown by Kleinberg in [27]. Since the splittable variants SP and SL_p have a convex objective and linear constraints, a global optimum exists and can be computed with arbitrary precision in polynomial time, see Grötschel, Lovasz and Schrijver [22]. If SP and SL_p have a strictly convex objective, the global optimum is also unique.

Before we characterize *efficient* price functions with respect to a given congestion metric, we start with a simple example demonstrating the relationship between congestion efficiency and cost efficiency.

Example 1: We are given a network consisting of two nodes connected by two parallel links. Price functions on both links are constant, 1 on the lower and $1-\epsilon$ on the upper arc, respectively. Hence, w.l.o.g. iREX routes every demand that is released along the upper link. Note that there is no incentive to distribute the traffic since prices remain constant, while the cheapest arc is the upper. Hence, for latency functions $\ell(x) = x$ on both links the ratio $\Phi(\boldsymbol{f})/\Phi(\boldsymbol{f}^*)$, where $\boldsymbol{f}^*$ is the offline optimum, can be made arbitrarily large. Thus, in contrast to the total cost of a solution produced by iREX, there exist instances with affine price and affine latency functions, where the efficiency loss of iREX with respect to network congestion is unbounded.

In light of this negative example, we characterize restricted sets of price functions, which are provably efficient with respect to a given congestion metric.

Theorem 3.3: Given affine congestion functions $\ell_a(z) = q_a\, z + r_a$, where $q_a \geq 0, r_a \geq 0,\ a \in A$. If $p_a(z) = \ell_a(z), \forall a \in A$, then, the competitive ratio of the online algorithm iREX with respect to $\Phi(\boldsymbol{f})$ is bounded by $\frac{16\,K}{2\,K+1}$, where K denotes the number of commodities. Furthermore, for $K \in \mathbb{N} \cup \{\infty\}$, iREX is 8-competitive.

Proof: First, we use (5), which implies:

$$\Phi(\boldsymbol{h}) \leq 2\, C(\boldsymbol{h}) \leq 2\Phi(\boldsymbol{h}), \tag{7}$$

for arbitrary flows $\boldsymbol{h}$. Then, the following set of inequalities is valid:

$$\Phi(\boldsymbol{f}) \leq 2\, C(\boldsymbol{f}) \leq 2\, \frac{8\,K}{2\,K+1} C(\boldsymbol{x}^*) \leq \frac{16\,K}{2\,K+1} \Phi(\boldsymbol{x}^*),$$

where $\boldsymbol{x}^*$ denotes the offline optimal flow minimizing $\Phi(\cdot)$. The first inequality follows from (7). The second inequality follows from the bound in Theorem 3.2, which also holds for $\boldsymbol{x}^*$ as a feasible flow. The last inequality follows again from (7). ■

Corollary 3.4: If the used price functions are given by $p_a(x) = x,\ a \in A$ then, the competitive ratio of the online algorithm iREX for the L_2-norm is bounded by $\sqrt{\frac{16\,K}{2\,K+1}}$, where K denotes the number of commodities. Furthermore, for $K \in \mathbb{N} \cup \{\infty\}$, iREX is $\sqrt{8}$-competitive.

Proof: We consider latency functions $\ell_a(z) = z, a \in A$. Then, we have

$$L_2(\boldsymbol{f}) = \sqrt{\Phi(\boldsymbol{f})} \leq \sqrt{\frac{16\,K}{2\,K+1} \Phi(\boldsymbol{x}^*)} = L_2(\boldsymbol{x}^*),$$

where the first inequality follows from Theorem 3.3. The last equality follows from the monotonicity of the root. ■

Remark 3: The results of Theorem 3.3 and Corollary 3.4 also hold for the splittable variant of iREX.

The above results characterize conditions on price functions that are sufficient to achieve an efficient resource allocation with respect to network congestion. In practice, however, iREX allows resource providers the autonomy to choose their private price functions according to selfish profit maximizing interests. Natural question then are: (i) can we efficiently calculate optimal prices? (ii) what is the effect on network congestion, if price functions are based on private choices and do not satisfy the assumptions of Theorem 3.3?

In the following section (III-E), we answer the first question in terms of the complexity of the underlying optimization problem. And in Section IV we try to answer the second question empirically by simulating the iREX architecture for a large set of price functions (linear, squared, cubic, or chosen randomly), which are not necessarily related to a given congestion metric. For an analytical answer to the second question, we point the reader to recent work by Harks *et al*. [28], which builds upon this paper and studies a more general setting including general price functions.

E. The Complexity of Optimal Pricing

In this section, we briefly discuss the complexity of determining optimal prices for resource sellers. Instead of fixed price functions $p_a(x)$ for every link $a \in A$, we assume that providers determine individual prices p_a so as to maximize their individual profit.

We are given a set of $[n] = \{1, \dots, n\}$ providers, each of them owning a subset A_i of resources. These subsets are mutually disjoint and satisfy $\cup_{i\in[n]} A_i = A$. We denote by $p \in \mathbb{R}^{|A|}$ the vector of all prices, and by $p^i \in \mathbb{R}^{|A_i|}$ the vector of link prices set by provider i on the links A_i that this provider owns. Furthermore, $p^{-i} \in \mathbb{R}^{|A|}$ denotes the vector of prices except for prices $p_a, a \in A_i$. Suppose, we are given a single ID QoS request (d, s, t) starting from node s, ending at node t, and having a demand request of size d. For a price vector p, we introduce $z(p) \in \{0, 1\}^{|A|}$ and write $z(p) \in S(p)$ if and only if $z_a(p) = 1$ when link a is contained in the cheapest path for demand d, and $z_a(p) = 0$ otherwise. If the cheapest path is not unique, ties are broken arbitrarily. The payoff for provider i is defined as $P^i(p^i; p^{-i}) := \sum_{a \in A_i} p_a\, z_a(p)\, d$ with $z(p) \in S(p)$. For fixed prices p^{-i} of other provider domains, the best pricing strategy for provider i is to solve the following optimization problem:

$$\max_{p^i \geq 0} P^i(p^i; p^{-i}), \quad \text{s.t.:} \quad z(p) \in S(p). \tag{8}$$

The challenge of this problem is that a provider must strike the right balance between low prices, which generate low revenue, and high prices, which could also result in low revenue as the demand would follow a cheaper path (i.e. $z_a(p) = 0$ for $a \in A_i$) as is also explained in Section II-B. In the following, we show that this optimization problem is strongly $\mathcal{NP}$-hard, even when only two providers own the network resources. Our proof uses a reduction of the MAXTOLL problem to a special case of our problem. Note that MAXTOLL is known to be strongly $\mathcal{NP}$-hard, see Roch *et al.* [19].

Theorem 3.5: Problem (8) is strongly $\mathcal{NP}$-hard, even for the case of two resource providers.

Proof: An instance of MAXTOLL consists of a partition of the arcs A of a directed graph $D = (V, A)$ into two sets $A_1 \cup A_2 = A$. The arcs of the first set A_1 have fixed nonnegative costs $p^1 = (p_a)_{a \in A_1}$, while the arcs of the second set A_2 can be priced with nonnegative tolls $p^2 = (p_a)_{a \in A_2}$. We are given a single commodity $(1, s, t)$ with a demand of size 1 that has to be routed from s to t along the cheapest path with respect to the fixed costs p^1 and tolls p^2. The (MAXTOLL) problem is to find a toll vector p^2 maximizing the raised profit $P^2(p^2; p^{-2})$ subject to $z(p) \in S(p)$. The constraint $z(p) \in S(p)$ ensures that the demand is routed along the shortest (cheapest) paths with respect to $p = p^1 \cup p^2$.

The preceding discussion shows that MAXTOLL is a special case of problem (8) when we set $n = 2$, $d = 1$, and $i = 2$. Thus, using that MAXTOLL is strongly $\mathcal{NP}$-hard, the claim follows directly. ■

This hardness result supports our assumption in Section II-B that resource providers will rely on heuristics (e.g. fixed price functions) to price their resources. Another implication of the above result concerns a game theoretic model of the pricing problem. For a given demand, we can define a non-cooperative pricing game, where the resource providers set prices so as to maximize their individual profit. Theorem 3.5 establishes the hardness of determining a pure strategy for every provider.

IV. NUMERICAL ANALYSIS OF iREX

In the previous sections, we introduced the framework ONLINEUMCRP in order to analyze the efficiency of online multi-commodity routing strategies for networks with given non-decreasing price functions. In particular, we made several simplifying assumptions (demands persist forever, no arc capacities) for our analytical study. Additionally, the derived analytical results are based on competitive analysis coming from the classical toolbox of the online optimization field. It is inherent to this concept that the competitive ratio of an online algorithm holds for *every* instance. However, worst-case instances may be very rare or even impossible to construct in practice.

Therefore, we present in the following a numerical analysis with more realistic settings so as to assess the efficiency of the routing strategies in practical environments. In particular, we present the numerical analysis under the following assumptions:

(i) we consider the vBNS network topology illustrated by Fig. 4 with each point of presence representing an ISP domain,

(ii) within the vBNS topology, ISP domains are connected to its neighbors with optical fiber links of finite capacity equivalent to an OC48 and the length of each link is calculated to be the actual beeline distance between the cities,

(iii) traffic demands are generated over time stochastically based on a simple Poisson arrival model with parameters derived from a $M/M/\infty$ analysis, and

(iv) deployed demands expire over time.

The vBNS topology was chosen because it is small enough for us to get useful data, and also represents many structures found at the core of inter-domain power law networks – where ISPs connect to each other. We expect that iREX's performance gain, in comparison to other methods, to be proportional to the number of neighbors connected to an iREX domain.

In the remainder of this section, we empirically quantify the efficiency loss of a greedy online routing algorithm modeling the iREX architecture.

A. Online Routing with Expiration

As in the previous sections, we consider a set $[K] := \{1, \ldots, K\}$ of source-destination pairs that represent the inter-domain reservation requests (demands). For each $k \in [K]$, a demand of d_k must be routed from the source s_k to the destination t_k. Without loss of generality, we assume that all demands have the same normalized bundle size. A demand value that is larger than this bundle size can be represented by several demands of this bundle size. We introduce a starting time τ_k that specifies the time commodity k is revealed to the system. Furthermore, every demand k has a duration time E_k. Without loss of generality, we assume the time points are ordered $\tau_1 < \cdots < \tau_K$. We define $[K(\tau)] \subseteq [K]$ to be the subset of commodities that are active at time τ. Formally, the set is defined as $[K(\tau)] := \{i \in [K] \,|\, \tau \in [\tau_i, \tau_i + E_i]\}$. The resource links $a \in A$ of the network are equipped with finite resource capacities $c = (c_a, a \in A)$. iREX routes every commodity $k \in [K]$ that is released at time τ_k along the cheapest available path. This is equivalent to solving the following linear min-cost flow problem:

$$\min \quad \sum_{a \in A} p_a \Big(\sum_{i \in [K(\tau_k)]} f_a^i \Big) f_a^k$$

subject to the standard flow constraints for single path routing and capacity constraints $\sum_{i\in[K(\tau_k)]} f_a^i \leq c_a$.

B. The Offline Optimum

Since the total traffic load varies over time, we evaluate the efficiency of the iREX architecture at different time points τ. Our measure of efficiency is again based on competitive analysis coming from online optimization. We will present two variants of an offline optimum, where we consider both, the *unsplittable* (single path) offline optimum and the *splittable* (multi-path) offline optimum. In both variants, we minimize network congestion for commodities $k \in [K(\tau)]$ as defined by the problems P, SP and L_∞, SL_∞, respectively. Note that for both problems, we have additional capacity constraints. When we speak of the P and L_∞ metric, our reference solution is the offline optimum for problem P and L_∞, respectively.

The following trivial bounds characterize the relation between a solution produced by iREX and the unsplittable (splittable) offline optimum for P:

Proposition 4.1: Let $\boldsymbol{f} = (\boldsymbol{f}^k, k \in [K(\tau)])$ be a feasible flow that is produced by the solutions of problem iREX$_\tau$ at time τ. Let $\boldsymbol{h}$ and $\boldsymbol{g}$ be optimal flows of P and SP, respectively. Then, the following inequalities are satisfied:

$$\Phi(\boldsymbol{g}) \leq \Phi(\boldsymbol{h}) \leq \Phi(\boldsymbol{f}). \tag{9}$$

Proof: Each flow $\boldsymbol{f}^k$ routes the demand d_k on a single path. Hence, $\boldsymbol{f}$ is feasible for problem P, i.e., $\Phi(\boldsymbol{h}) \leq \Phi(\boldsymbol{f})$. Furthermore, $\boldsymbol{h}$ is a feasible flow for SP. Therefore, $\Phi(\boldsymbol{g}) \leq \Phi(\boldsymbol{h})$. ■

To evaluate the performance of the solutions of iREX$_{\tau_k}$, we numerically solve P and SP, which provides us with the lower bounds $\Phi(\boldsymbol{g}) \leq \Phi(\boldsymbol{h}) \leq \Phi(\boldsymbol{f})$. Furthermore, we can empirically quantify the gain of the fractional routing compared to the unsplittable variant.

We have a similar relationship for the L_p metric:

Proposition 4.2: Let $\boldsymbol{f} = (\boldsymbol{f}^k, k \in [K(\tau)])$ be a flow produced by the solutions of iREX$_\tau$ at time τ. Let $\boldsymbol{h}$ and $\boldsymbol{g}$ be optimal flows of L_p and (SL_p), respectively. Then, the following inequalities are satisfied:

$$L_p(\boldsymbol{g}) \leq L_p(\boldsymbol{h}) \leq L_p(\boldsymbol{f}). \tag{10}$$

Here problem SL_p is the splittable variant of problem L_p.

C. The Simulator

The iREX simulator (available at [29]) implements a simplified BGP protocol, the iREX protocols and the SLA framework. The simulator performs packet level simulation for control packets used for iREX and BGP signaling, and flow level simulation for the deployment of flows with QoS constraints.

We simulated three iREX simulation sub-configurations based on the type of heuristic (price function) used by domains to price their resources. The *linear* sub-configuration prices resources uniformly according to the affine linear function $p(z) = a_0 + a_1 z$. The *squared* sub-configuration prices resources uniformly according to the squared polynomial $p_a(z) = a_0 z + a_1 z + a_2 z^2$. The *random* sub-configuration randomly assigns each domain one of three price functions – linear, squared or cubed ($p(z) = a_0 z + a_1 z + a_2 z^2 + a_3 z^3$). All coefficients a_i are assumed to be nonnegative and are randomly assigned.

Inter-domain reservation requirements within the simulator are viewed as bundles of traffic sized 0.1% of line speed (about 2.4mb/sec) with a 5 minute average reservation duration (E_k). The traffic load (total projected bandwidth usage) is determined according to a percentage of each domain's actual total egress capacity in the topology from 40% to 100% in 4% steps. We chose this traffic range to simulate nominal to heavy traffic loads.

D. Metrics

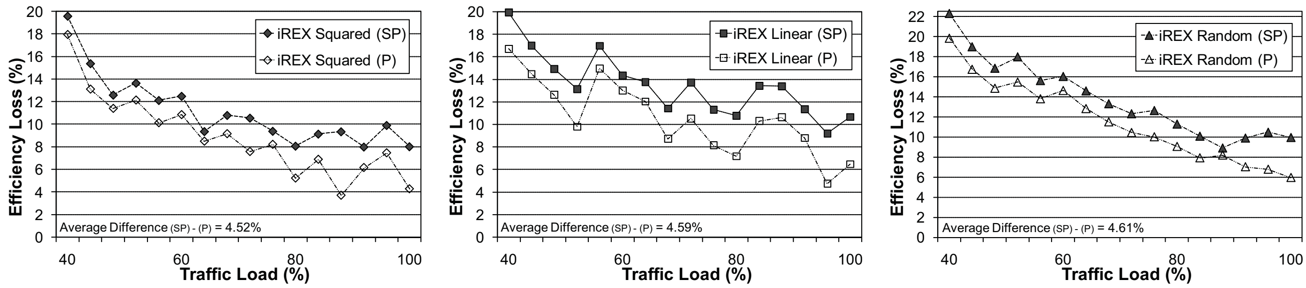
We present efficiency results using two metrics, the efficiency loss compared to solutions of P and SP, and compared to solutions of L_∞ and SL_∞. These results are from 4 simulation runs for the *linear* and *squared* sub-configurations, and 16 runs for the *random* sub-configurations, with individual runs having approximately 500,000 simulated reservations. To compare the simulation results with the lower bounds derived from solving P, SP, and L_∞ and SL_∞, we evaluated congestion for a simulated flow $\boldsymbol{f}$ by evaluating $\Phi(\boldsymbol{f})$ and $L_\infty(\boldsymbol{f})$ at time points τ. For the evaluation of $\Phi(\boldsymbol{f})$, we used linear latency functions. For all graphs, we define *efficiency loss* to be the percentage difference between the network congestion of the iREX architecture simulation results and the computed optimal solutions as defined by the problems P, SP and L_∞, SL_∞. That is, if the iREX architecture produces a flow $\boldsymbol{f}$, and the optimal flow for problem P is denoted by $\boldsymbol{f}^*$, the efficiency loss with respect to P is defined as:

$$\text{Efficiency loss} = \frac{\Phi(\boldsymbol{f})-\Phi(\boldsymbol{f}^*)}{\Phi(\boldsymbol{f}^*)} \times 100.$$

We show the numerical results in reference to the multiple (splittable flow) and single (unsplittable flow) path solutions. Each graph in this section has two curves, which show the efficiency loss with respect to a solution that uses splittable flow, and one that only uses a unsplittable flow. The simple average of the difference between the two curves is also included. While the single path routing describes the iREX architecture, the multi-path solution is an absolute reference bound for all possible methods (including future multi-path iREX architecture improvements).

E. Mathematical Solutions

To efficiently compute solutions for all problems of type P, SP, and L_∞, SL_∞ we used CPLEX 10.0, that is equipped with Linear (LP), Quadratic (QP), Mixed Integer Problem (MIP), and Quadratic Integer (QIP) solvers. For modeling purposes, we used the ZIMPL modeling language, see Koch [30]. In total, more than 2600 problems of type SP, and SL_∞ are solved to optimality. We solved the problems P, L_∞ involving integer constraints within 1% of optimality. Average running time on a Pentium 4 (3GHz) for the problem type SP, SL_∞ was about 1 second and for problem types P, L_∞ about 30 seconds.



(a) *Squared* sub-configuration (b) *Linear* sub-configuration (c) *Random* sub-configuration

Fig. 5: iREX Efficiency loss with respect to P and SP.

F. Simulation Results

Figs. 5(a), 5(b), and 5(c) show the efficiency loss to P and SP for iREX using the *squared*, *linear*, and *random* price functions respectively under varying traffic load. The worst case (*random*) efficiency loss to the single path P metric is about 20%, and the "best" worse case among the three sub-configurations is about 17% (*linear*).

We observe that throughout the traffic loads, the *squared* sub-configuration (Fig. 5(a)) performs better than the *linear* (Fig. 5(b)), and *random* (Fig. 5(c)) sub-configurations. Price functions determine the speed and aggressiveness of a domain's response to a market situation, and the "faster" *squared* price function allows for faster use of alternative paths, thereby making the *squared* sub-configuration perform better. To further expose this behavior, we refer to the *squared* (Fig. 5(a)) sub-configuration's smaller distance to the optimal solution (i.e. x axis) in comparison to the *linear* (Fig. 5(b)) at traffic loads 60%, 72% and 88%.

The *random* (Fig. 5(c)) sub-configuration, which we feel is the most realistic scenario, performed worse than the domains in the uniform price function sub-configurations. This may be caused by this sub-configuration allowing for more response differences by allowing for a multitude of domain price function choices. We note however that the worse efficiency loss difference between the *random* and the best (*squared*) sub-configuration is about 10%.

We also observe that iREX's efficiency loss to the single path metric P is consistently scaled lower than its efficiency loss to the multi path (splittable-flow) metric SP with the difference averaging between 4.52% to 4.61%.

In all cases, efficiency loss decreases with increased traffic load, this is because as traffic load increases, the search space for "good" ID paths decrease.

In contrast to the iREX method results, the SLA method exhibits a minimum efficiency loss of 100% which increased to a maximum of 340% with respect to P as seen in Fig. 6. The constant increase in efficiency loss is due to the static nature of this method.

Fig. 7(a), 7(b), and 7(c) show the efficiency loss to L_∞ and SL_∞ for iREX using the *squared*, *linear*, and *random* price functions respectively under varying traffic load. The worst case efficiency loss to the single path L_∞ metric is about 48% for all the three sub-configurations. The differences in the sub-configurations are small due to the nature of the metric that tabulates only the most congested of links.

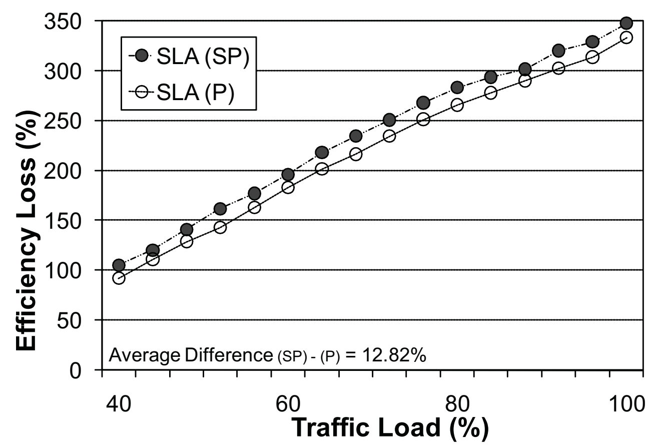


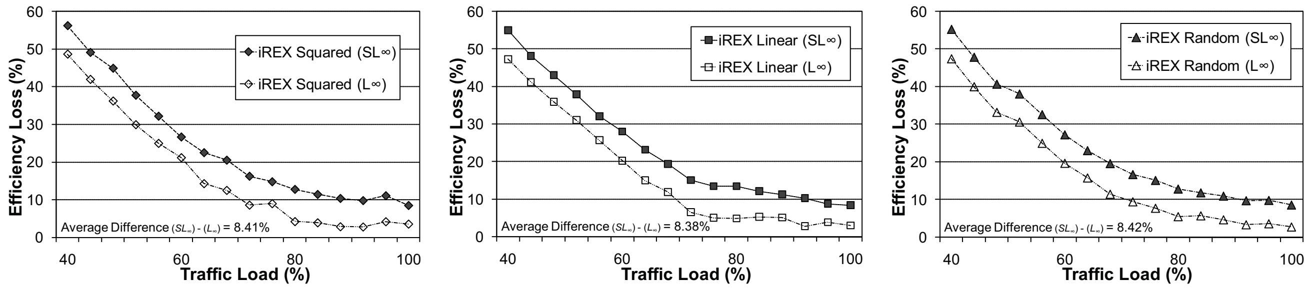
Fig. 6: SLA Efficiency loss with respect to P and SP.

We again observe that iREX's efficiency loss to the single path metric L_∞ is consistently scaled lower than its efficiency loss to the multi-path (splittable-flow) metric SL_∞ with the difference between L_∞ and SL_∞ averaging between 8.38% to 8.42%. And again we see that efficiency loss decreases with increased traffic load due to the decreasing search space for "good" ID paths.

The SLA method stays at about 72% efficiency loss across the same traffic load ranges as seen in Fig. 8. This efficiency loss does not increase because usage on the most congested link has reached maximum capacity.

V. Related Work

Applying economics and the concept of pricing within networking has been studied in Mackie-Mason *et al.* in [31] and [32], Yang in [33], and Shenker *et al.* in [34] but work in ID policy within an economic environment has been sparse. Fankhauser *et al.* [35] proposed an economics based SLA trading system, Koistenen *et al.* [36] proposed a protocol for peers to negotiate prices and Wang *et al.* proposed RNAP [37], but in these systems, policy deployment is done bilaterally among neighboring peers whereas in iREX, the source domain deploys policy bilaterally with all domains involved in the deployment. Inter-AS pricing has been proposed by Mortier *et al.* in [38], but this method only uses pricing between AS to simplify the gauging of congestion and does not deploy E2E QoS policy. He and Walrand studied the interactions of service providers under a pricing rule in which revenues are shared fairly [39]. Ebata *et al.* proposed a system for ID provisioning



(a) *Squared* sub-configuration (b) *Linear* sub-configuration (c) *Random* sub-configuration

Fig. 7: iREX Efficiency loss with respect to L_∞ and SL_∞.

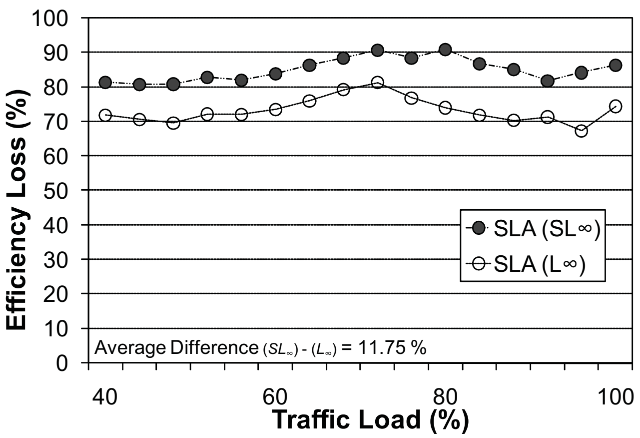


Fig. 8: SLA Efficiency loss with respect to L_∞ and SL_∞.

and accounting in [40], but they focus on the dissemination of information about services available, whereas iREX focuses on providing information about the desirability of available ID transport services. Bandwidth Brokers [1]–[4] address some of the same issues as iREX, but iREX promotes direct bilateral relationships between the consumer domain and all provider domains to facilitate economic incentives. Choice in ID paths has been commercially offered by Internap [41] and Avaya [42], but these are just a start – iREX would offer more than just a choice of the first hop ISP. Bandwidth switching exchanges like Tradingcom Europe [43] and Enron Broadband Services [44] (currently in the midst of restructuring) are centralized services that operate similar to stock exchanges where ISPs trade excess capacity – iREX is a fully distributed architecture that can be used for similar purposes, but without the use of any centralized entity.

Work on the efficiency of routing problems have been extensively studied using game theoretic concepts, cf. Correa [45] and Roughgarden [46], [47], and references there in. However, the characterization of a Nash equilibrium is not applicable to our problem setting. In iREX, once a routing decision has been made this routing remains unchanged, that is, the routing is irrevocable. For the duration of this deployment all succeeding demands pay increased cost. In the concept of a Nash equilibrium, if a new demand is released, all demands are possibly rerouted according to the selfish interest of travel time minimization. Harks *et al*. study the problem of routing commodities online [21]. In contrast to our work, they focus on *splittable* routings and prove upper bounds on the competitive ratio of a greedy online algorithm and a different cost function.

Initial work on iREX was previously published by Yahaya and Suda in [10], work defining the iREX architecture was previously published by Yahaya and Suda in [11], preliminary work exploring iREX efficiency was previously published by Yahaya, Harks and Suda in [12], and work suggesting a multi-path extension to iREX was previously published by Yahaya and Suda in [48].

VI. Conclusions and Future Work

We have presented two methods that explore iREX as an architecture that efficiently automates the deployment of ID QoS policy. Using the analytical methodology outlined in Section III, for general networks and non-expiring demands we have shown that the worst case cost efficiency loss is bounded by a factor of $\frac{8K}{2K+1}$, where K is the number of deployments, provided *affine linear* price functions are used. Furthermore, we characterized classes of price functions that are provably efficient with respect to network congestion. Using the numerical analysis methodology outlined in Section IV, we evaluated iREX with respect to two types of derived reference solutions. Our numerical results have shown that even when disadvantaged by doing reservations sequentially in time and not splitting demands, at nominal to heavy traffic loads of 40% or more where efficiency is most important, the iREX architecture still exhibits a low worse case efficiency loss of about 20% with respect to the single path P reference solution. These results let us conjecture that iREX performs well in arbitrary environments.

We believe that research into autonomic systems dealing with non-technical issues such as *ownership* and *trust* to solve the *human* aspect of Internet management problems is important and we see iREX as the first step in this direction. Our future work will be to continue to prove iREX's merits by investigating the architecture from the standpoint of economics theory and further simulations, and to build iREX into a real system by addressing issues raised by iREX's data plane requirements outlined in subsection II-F.

Finally, we close with a comment on the research area in which iREX resides. In our analysis contained in this paper, we simplified the competition within the market by assuming fixed price functions defining the price for a unit resource. In practice, resource providers determine prices depending on the current market situation and their position with respect to the network topology. If the provider domain's link is a bottleneck, the demand would become somewhat inelastic leading to a monopolistic situation. For a fully connected network (i.e. perfect competition in the network), the demand is at a minimum when the offered price is above the current market price and at maximum when below. The infrastructure of the Internet today is more related to an oligopolistic market where the network is not fully connected. We are only aware of few works on this complex topic. Acemoglu and Ozdaglar [49] studied the competition of service providers for very simple network topologies such as parallel links or serial links. However, the outcome of competition between service providers for general network topologies, where demand is elastic remains tantalizingly open.

REFERENCES

[1] A. Terzis, L. Wang, J. Ogawa, and L. Zhang, "A two-tier resource management model for the internet," in *Proceedings of Global Internet 99*, Rio de Janeireo, Brazil, Dec 1999.

[2] V. Sander, W. Adamson, I. Foster, and A. Roy, "End-to-end provision of policy information for network qos," in *IEEE Symposium on High Performance Distributed Computing*, San Francisco, CA, USA, August 2001.

[3] H. Mantar, J. Hwang, I. Okumus, and S. Chapin, "A scalable model for interbandwidth broker resource reservation and provisioning," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 10, pp. 2019–2034, Dec 2004.

[4] ——, "A bandwidth-broker based inter-domain sla negotiation," *Lecture Notes in Computer Science*, vol. 4267/2006, pp. 134–140, 2006.

[5] M. Howarth, P. Flegkas, G. Pavlou, N. Wang, P. Trimintzios, D. Griffin, J. Griem, M. Boucadair, P. Morand, H. Asgari, and P. Georgatsos, "Provisioning for inter-domain quality of service: the MESCAL approach," *IEEE Communications Magazine*, vol. 43, no. 6, pp. 129–137, June 2005.

[6] O. Dugeon and A. Diaconescu, "From SLA to SLS up to QoS control: The CADENUS framework," *XIII World Telecommunications Congress*, September 2002.

[7] "IST-1999-11017 Service Scenarios," *CADENUS White Paper*, 2003.

[8] Frame Relay Forum, "FRF.13 service level definitions implementation agreement," August 1998.

[9] TeleManagement Forum, "GB917 Eva.Ver.1.5 SLA management handbook," June 2001.

[10] A. Yahaya and T. Suda, "iREX: Inter-domain QoS Automation using Economics," in *Proceedings of IEEE CCNC*, Las Vegas, Nevada, USA, January 2006.

[11] ——, "iREX: Inter-domain Resource Exchange Architecture," in *Proceedings of IEEE INFOCOM*, Barcelona, Spain, April 2006.

[12] A. Yahaya, T. Harks, and T. Suda, "iREX: Efficient inter-domain QoS policy architecture," in *Proceedings of IEEE Globecom*, San Francisco, California, USA, November 2006.

[13] K. Abbink and J. Brandts, "Price competition under cost uncertainty: A laboratory analysis," *UFAE and IAE Working Papers 550.02*, 2002.

[14] R. Braden, Ed., L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP) — version 1 functional specification," RFC 2205, Sep. 1997, status: PROPOSED STANDARD.

[15] W. Fang and L. Peterson, "Inter-AS traffic patterns and their implications," *Proceedings of IEEE Global Internet Symposium*, vol. 5, pp. 68–75, December 1999.

[16] R. Yavatkar, D. Pendarakis, and R. Guerin, "A framework for Policy-based Admission Control," IETF, RFC 2753, Jan 2000.

[17] Y. Rekhter and T. Li, "A border gateway protocol 4 (BGP-4)," IETF, RFC 1771, March 1995.

[18] N. Brownlee, C. Mills, and G. Ruth, "Traffic Flow Measurement: Architecture," IETF, RFC 2722, Oct 1999.

[19] S. Roch, G. Savard, and P. Marcotte, "An approximation algorithm for Stackelberg network pricing," *NETWORKS*, vol. 46, no. 1, pp. 57–67, 2005.

[20] A. Borodin and R. El-Yaniv, *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[21] T. Harks, S. Heinz, and M. E. Pfetsch, "Online multicommodity routing problem," in *Proceedings of the Fourth Workshop on Approximation and Online Algorithms*, Zurich, Switzerland, September 2006.

[22] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, 2nd ed., ser. Algorithms and Combinatorics. Heidelberg: Springer-Verlag, 1993, vol. 2.

[23] A. Fiat and G. J. Woeginger, Eds., *Online Algorithms: The State of the Art*, ser. Lecture Notes in Computer Science. Springer, 1998, vol. 1442.

[24] M. Patriksson, *The Traffic Assignment Problem—Models and Methods*. Topics in Transportation, VSP, 1994.

[25] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in *Proceedings of IEEE INFOCOM*, Tel-Aviv, Israel, March 2000, pp. 519–528.

[26] B. Awerbuch, Y. Azar, E. Grove, M.-Y. Kao, P. Krishnan, and J. Vitter, "Load balancing in the L_p norm," in *Proceedings of 36th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, Milwaukee, Wisconsin, USA, 1995, pp. 383–391.

[27] J. M. Kleinberg, "Single-source unsplittable flow," in *Proceedings of 37th IEEE Conference on Foundations of Computer Science*, Burlington, VT, USA, October 1996.

[28] T. Harks, S. Heinz, M. E. Pfetsch, and T. Vredeveld, "Online multicommodity routing with time windows," ZIB, Tech. Rep. 07-22, 2007.

[29] iREX WEB, "http://netresearch.ics.uci.edu/irex/."

[30] T. Koch, "Rapid mathematical programming," Ph.D. dissertation, Technische Universität Berlin, 2004, zIB-Report 04-58.

[31] J. MacKie-Mason and H. Varian, "Some economics of the internet," *10th Michigan Public Utility Conference*, March 1993.

[32] ——, "Pricing the internet," in Public Access to the Internet, Prentice-Hall, 1995.

[33] X. Yang, "Nira: A new internet routing architecture," in *Proceedings of ACM SIGCOMM FDNA 2003 Workshop*, Karlsruhe, Germany, August 2003.

[34] S. Shenker, D. Clark, D. Estrin, and S. Herzog, "Pricing in computer networks: reshaping the research agenda," *ACM Computer Communication Review*, vol. 26, no. 2, pp. 19–43, April 1996.

[35] 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.

[36] J. Koistinen and A. Seetharaman, "Worth-based multi-category quality-of-service negotiation in distributed object infrastructures," *Hewlett Packard Software Technology Laboratory Technical Report HPL-98-51 (R. 1)*, July 1998.

[37] X. Wang and H. Schulzrinne, "RNAP: a resource negotiation and pricing protocol," Basking Ridge, NJ, USA, June 1999.

[38] R. Mortier and I. Pratt, "Incentive based inter-domain routing," *Proceedings of Internet Charging and QoS Technology Workshop (ICQT'03)*, Sep 2003.

[39] L. He and J. Walrand, "Pricing and revenue sharing strategies for internet service providers," *IEEE Journal on Selected Areas in Communications*, May 2006.

[40] T. Ebata, M. Takihiro, S. Miyake, M. Koizumi, F. Hartanto, and G. Carle, "Interdomain QoS provisioning and accounting," *Internet-Draft draft-ebata-interdomain-qos-acct-00.txt*, November 1999.

[41] INTERNAP, "http://www.internap.com."

[42] AVAYA, "http://www.avaya.com."

[43] Tradingcom Europe, "http://www.tradingcomeurope.com."

[44] Enron Broadband, "http://www.enron.com/corp/products/broadband/."

[45] J. R. Correa, A. S. Schulz, and N. E. Stier-Moses, "Selfish routing in capacitated networks," *Mathematics of Operations Research*, vol. 29, pp. 961–976, 2004.

[46] T. Roughgarden, *Selfish Routing and the Price of Anarchy*. The MIT Press, 2005.

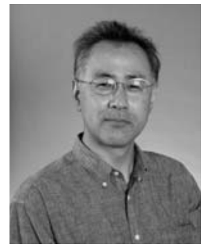
[47] T. Roughgarden and E. Tardos, "How bad is selfish routing?" *Journal of the ACM*, vol. 49, no. 2, pp. 236–259, 2002.

[48] A. Yahaya and T. Suda, "iREX MPO: A multi-path option for the iREX inter-domain QoS policy architecture," in *Proceedings of IEEE ICC*, Beijing, China, May 2008.

[49] D. Acemoglu and A. Ozdaglar, "Competition in parallel-serial networks," *IEEE Journal on Selected Areas in Communications, Special Issue on Non-Cooperative Behavior in Networking*, vol. 25, no. 6, pp. 1180–1192, August 2007.

Ariffin Datuk Yahaya (S'00) received the B.Sc. degree in computer science from California State University, Northridge, California, USA, in 2001 and the M.S. degree in computer science in 2006 from the University of California, Irvine, USA. He is currently working towards the Ph.D. degree at the School of Information and Computer Science, University of California, Irvine, USA. His research interests include Network Management, Autonomic Computing, Policy Based QoS Networks, QoS Routing and the general question of how to apply economic principles to network management. Mr. Yahaya has worked in industry first in deploying enterprise level computing systems, and then later in enterprise level network and database disaster recovery.

Tobias Harks (S'00) received the diploma degree and Ph.D. in mathematics from the University of M¨unster and the Berlin Institute of Technology, Germany, in 2003 and 2007, respectively. He is currently working as a Postdoctoral Research Associate at the Berlin Institute of Technology. His research interests include Combinatorial Optimization, Algorithmic Game Theory, Online Optimization, and Network Management.



Tatsuya Suda (S'80-M'82-SM'97-F'01) received the B.E., M.E., and Dr.E. degrees in applied mathematics and physics from Kyoto University, Kyoto, Japan, in 1977, 1979, and 1982, respectively. From 1982 to 1984, he was with the Department of Computer Science, Columbia University, New York, as a Postdoctoral Research Associate. Since 1984, he has been with the Department of Information and Computer Science, University of California, Irvine, where he is currently a Professor. He has also served as a Program Director of the Networking Research Program, National Science Foundation from October 1996 to January 1999. He was a visiting Associate Professor at the University of California, San Diego, a Hitachi Professor at the Osaka University, and currently is a NTT Research Professor. He is an Area Editor of the INTERNATIONAL JOURNAL OF COMPUTER AND SOFTWARE ENGINEERING. He is a member of the Editorial Board of the *Encyclopedia of Electrical and Electronics Engineering* (Wiley).

He has been engaged in research in the fields of computer communications and networks, high-speed networks, multimedia systems, ubiquitous networks, distributed systems, object oriented communication systems, network applications, performance modeling and evaluation, and application of biological concepts to networks and network applications. Dr. Suda is a member of the Association for Computing Machinery (ACM). He received an IBM Postdoctoral Fellowship in 1983. He was the Conference Coordinator from 1989 to 1991, the Secretary and Treasurer from 1991 to 1993, the Vice Chairman from 1993 to 1995, and the Chairman from 1995 to 1997 of the IEEE Technical Committee on Computer Communications. He was also the Director of the U.S. Society Relations of the IEEE Communications Society from 1997 to 1999. He is an Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING, a Senior Technical Consultant to the IEEE TRANSACTIONS ON COMMUNICATION, and a former Editor of the IEEE TRANSACTIONS ON COMMUNICATION. He was the Chair of the 8th IEEE Workshop on Computer Communications and the TPC Co-Chair of the IEEE INFOCOM 1997.