Selfish Overlay Network Creation and Maintenance

GEORGIOS SMARAGDAKIS
Deutsche Telekom Labs/TU Berlin
georgios.smaragdakis@telekom.de

NIKOLAOS LAOUTARIS
Telefonica Research
nikos@tid.es

VASSILIS LEKAKIS
University of Maryland
lex@umd.de

AZER BESTAVROS
Boston University
best@cs.bu.edu

JOHN W. BYERS
Boston University
byers@cs.bu.edu

MEMA ROUSSOPOULOS
University of Athens
mema@di.uoa.gr

***Abstract*— A foundational issue underlying many overlay network applications ranging from routing to peer-to-peer file sharing is that of the network formation, *i.e.*, folding new arrivals into an existing overlay, and re-wiring to cope with changing network conditions. Previous work has considered the problem from two perspectives: devising practical heuristics for the case of cooperative peers, and performing game theoretic analysis for the case of selfish peers. In our work, we unify the aforementioned thrusts by defining and studying the Selfish Neighbor Selection (SNS) game and its application to overlay routing. At the heart of SNS stands the restriction that peers are allowed up to a certain number of neighbors. This makes SNS substantially different from existing network formation games that impose no bounds on peer degrees. Having bounded degrees has important practical consequences as it permits the creation of overlay structures that require $O(n)$ instead of $O(n^2)$ link monitoring overhead.**

We show that a node's "best response" wiring strategy amounts to solving a k-median problem on asymmetric distance. Best response wirings have substantial practical utility as they permit selfish nodes to reap substantial performance benefits when connecting to overlays of non-selfish nodes. A more intricate consequence is that even non-selfish nodes can benefit from the existence of some selfish nodes since the latter, via their local optimizations, create a highly optimized backbone, upon which even simple heuristic wirings yield good performance. To capitalize on the above properties we design, build and deploy, EGOIST, an SNS-inspired prototype overlay routing system for PlanetLab. We demonstrate that EGOIST outperforms existing heuristic overlays on a variety of performance metrics, including delay, available bandwidth, and node utilization, while it remains competitive with an optimal, but unscalable full-mesh overlay.

***Index Terms*— Overlay networks, overlay routing, selfish neighbor selection, network formation.**

I. INTRODUCTION

Motivation: Overlay networks [3] are used for a variety of popular applications including routing [4], content distribution [5], [6], peer-to-peer (P2P) file sharing [7], [8] and streaming [9], [10], [11], data-center applications [12], and online multi-player games [13]. A foundational issue underlying many such overlay network applications is that of connectivity management. Connectivity management is called upon when having to wire a newcomer into the existing mesh of nodes (bootstrapping), or when having to rewire the links between overlay nodes to deal with churn and changing network conditions. Connectivity management is particularly challenging for overlay networks because overlays often consist of nodes that are distributed across multiple administrative domains, in which auditing or enforcing global behavior can be difficult or impossible. As such, these nodes may act selfishly and deviate from the default protocol, by utilizing knowledge they have about the network, to maximize the benefit they receive from it. Selfish behavior has been reported in studies relating to selfish (source) routing [14] and free riding [15] in P2P file-sharing networks. Selfish behavior also has many implications for connectivity management. In particular, it creates additional incentives for nodes to rewire, not only for operational purposes (bootstrapping and substituting nodes that went off-line), but also for seizing opportunities to incrementally maximize the local connection quality to the overlay. While much attention has been paid to the harmful downsides of selfish behavior in different settings [14], [16], [17], the impact of adopting selfish connectivity management techniques in real overlay networks has been an open problem [18].

Selfish Neighbor Selection: In a typical overlay network, a node must select a fixed number (k) of immediate overlay neighbors for routing traffic. Previous work has considered this problem from two perspectives: (1) Devising *practical heuristics* for specific applications in real deployments, such as bootstrapping by choosing the k closest links (*e.g.*, in terms of TTL or IP prefix distance), or by choosing k random links in a P2P file-sharing system. Notice here that DHTs like Chord [8] solve a different problem. They route queries, not data traffic. The latter is left to a separate subsystem [19] that typically opens a direct connection to the target host. (2) Providing abstractions of the underlying fundamental neighbor selection problem that are analytically tractable, especially via game-theoretic analysis [20], [21], [22]. To date, however, the bulk of the work and main results in this area have centered on strategic games where edges are undirected, access costs are based on hop-counts, and nodes have potentially unbounded degrees [20], [23], [21], [24], [22]. While this existing body of work is extremely helpful for laying a theoretical foundation and for building intuition, it is not clear how or whether the guidance provided by this prior work generalizes to situations of practical interest, in which underlying assumptions in these prior studies are not satisfied. Another aspect not considered in

G. Smaragdakis was supported by Deutsche Telekom Laboratories under a Strategic Research Grant. N. Laoutaris is supported by the NANODATA-CENTERS EU project. A. Bestavros and J. Byers are supported by a number of NSF awards, including CISE/CSR #0720604, ENG/EFRI #0735974, CISE/CNS #0952145, #1012798, #1040800 and CISE/CCF #0820138. The work of M. Roussopoulos was supported by NSF CAREER Award #0446522. Parts of this work appeared in the proceedings of the IEEE INFOCOM '07 [1] and ACM CoNEXT '08 [2].

previous work is the consideration of settings in which some or even most players do not play optimally – a setting which we believe to be typical. Interesting questions along these lines include an assessment of the advantage to a player from employing an optimizing strategy, when most other players do not, or more broadly, whether employing an optimizing strategy by a relatively small number of players could be enough to achieve global efficiency.

Paper Scope and Contributions: In this paper, we formulate and answer such questions using a combination of modeling, analysis, and extensive simulations using synthetic and real datasets. Our starting point is the definition of a network creation game that is better suited for settings of P2P and overlay routing applications – settings that necessitate the relaxation and/or modification of some of the central modeling assumptions of prior work. In that regard, the central aspects of our model are bounded degree, directed edges, non-uniform preference vectors, and representative distance functions.

Our first technical contribution within this model is to express a node's "best response" wiring strategy as a k-median problem on asymmetric distance [25], and use this observation to obtain pure Nash equilibria through iterative best response walks via local search. We then experimentally investigate the properties of stable wirings using link weights obtained from PlanetLab and the AS-level topologies maps. Here, we find that selfish nodes can reap substantial performance benefits when connecting to overlay networks composed of non-selfish nodes. On the other hand, in overlays that are dominated by selfish nodes, the resulting stable wirings are already so highly optimized that even non-selfish newcomers can extract near-optimal performance through heuristic wiring strategies.

Motivated by the above positive results, we design, implement, and deploy EGOIST, a prototype overlay routing network built around best response wiring strategies. EGOIST serves as a building block for the construction of efficient and scalable overlay applications consisting of (potentially) selfish nodes. We first demonstrate through real measurements on PlanetLab that overlay routing atop EGOIST is significantly more efficient than systems utilizing common heuristic neighbor selection strategies under multiple performance metrics, including delay, system load and available bandwidth. Second, we demonstrate that the performance of EGOIST approaches that of a (theoretically-optimal) full-mesh topology, while achieving superior scalability, requiring link announcements proportional to nk compared to n^2 for a full mesh topology. Our experimental results show that EGOIST remains highly effective under significant churn and incurs minimal overhead. Our evaluation includes among others, a case study in which EGOIST is used for routing the traffic generated by an online multi-player P2P game.

II. Overlay Network Model and Definitions

Previous work on overlay network creation [20], [23], [21], [24], [22] has focused on physical telecommunication networks and primarily the Internet. Overlay networks are substantially different [26], [27] which prompts us to consider the following overlay network model.

A. Overlay Network Model

We start by relaxing and modifying some of the central modeling assumptions of previous work. In that regard, the central aspects of our model are:

Bounded Degree: Most protocols used for implementing overlay routing or content sharing impose hard constraints on the maximum number of overlay neighbors. For example, in popular versions of BitTorrent a client may select up to 50 nodes from a neighbors' list provided by the *Tracker* of a particular torrent file [28]. In overlay routing systems [29], the number of immediate nodes has to be kept small so as to reduce the monitoring and reporting overhead imposed by the link-state routing protocol implemented at the overlay layer. Hard constraints on the number of first hop neighbors are also imposed in most P2P systems to address scalability issues, up-link fragmentation, and CPU consumption due to contention [30]. Motivated by these systems, we explicitly model such hard constraints on node degrees. Notice that in the prior studies cited above, node degrees were *implicitly bounded* (as opposed to *explicitly constrained*) by virtue of the trade-off between the additional cost of setting up more links and the decreased communication distance achieved through the addition of new links. We also note that some of these earlier network creation games were proposed in the context of physical communication networks [20], [23]. In such networks, the cost of acquiring a link is instrumental to the design and operation of a critical infrastructure. Such concerns do not apply in the case of overlay networks such as those we consider in this paper.

Directed Edges: Another important consideration in the settings we envision for our work relates to link directionality. Prior models have generally assumed bi-directional (undirected) links [20], [23], [21], [24], [22]. This is an acceptable assumption that fits naturally with the unbounded node degree assumption for models that target physical telecommunication networks because actual wire-line communication links are almost exclusively bidirectional. In overlay settings we consider, this assumption needs to be relaxed since the fact that node v forwards traffic or requests to node u does not mean that node u may also forward traffic or requests to v. Undirected links are created by the establishment of two directed links.

Non-uniform preference vectors: In our model, we supply each node with a vector that captures its local preference for all other destinations. In overlay routing such preference may capture the percentage of locally generated traffic that a node routes to each destination, and then the aggregation of all preference vectors would amount to a origin/destination traffic matrix. In P2P overlays such preference may amount to speculations from the local node about the quality of, or interest in, the content held by other nodes. Other considerations may also include subjective criteria such as the perceived capacity of the node, its geographic location, or its availability profile.

B. Definitions

Let $V = \{v_1, v_2, \ldots, v_n\}$ denote a set of nodes. Associated with node v_i is a preference vector $p_i = \{p_{i1}, p_{i2}, \ldots, p_{ii-1}, p_{ii+1}, \ldots, p_{in}\}$, where $p_{ij} \in [0, 1]$ denotes the preference of v_i for v_j, $i \neq j$: $\sum_{j=1, j\neq i}^{n} p_{ij} = 1$. Node v_i establishes a *wiring* $s_i = \{v_{i_1}, v_{i_2}, \ldots, v_{i_{k_i}}\}$

by creating links to k_i other nodes (we will use the terms link, wire, and edge interchangeably). Edges are *directed* and *weighted*, thus $e = (v_i, v_j)$ can only be crossed in the direction from v_i to v_j, and has cost d_{ij} ($d_{ji} \neq d_{ij}$ in the general case). Let $S = \{s_1, s_2, \ldots, s_n\}$ denote a *global wiring* between the nodes of V and let $d_S(v_i, v_j)$ denote the cost of a shortest directed path between v_i and v_j over this global wiring; $d_S(v_i, v_j) = M \gg n$ if there's no directed path connecting the two nodes. If the links are also annotated, then $M \gg \max_{i,j} d_{ij}$. For the overlay networks discussed here, the above definition of cost amounts to the incurred end-to-end delay when performing shortest-path routing along the overlay topology S, whose direct links have weights that capture the delay of crossing the underlying IP layer path that goes from the one end of the overlay link to the other. Let $C_i(S)$ denote the cost of v_i under the global wiring S, defined as the weighted (by preference) summation of its distances to all other nodes, *i.e.*, $C_i(S) = \sum_{j=1, j\neq i}^{n} p_{ij} \cdot d_S(v_i, v_j)$.

Definition 1: (The SNS Game) The selfish neighbor selection game is defined by the tuple $\langle V, \{S_i\}, \{C_i\}\rangle$, where:

- V is the set of n players, which in this case are the nodes.
- $\{S_i\}$ is the set of strategies available to the individual players. S_i is the set of strategies available to v_i. Strategies correspond to wirings and, thus, player v_i has $\binom{n-1}{k_i}$ possible strategies $s_i \in S_i$.
- $\{C_i\}$ is the set of cost functions for the individual players. The cost of player v_i under an outcome S, which in this case is a global wiring, is $C_i(S)$.

The above definition amounts to a local connection [17], non-cooperative, non-zero sum, n-player game [31]. Let $S_{-i} = S - \{s_i\}$ denote the *residual wiring* obtained from S by taking away v_i's outgoing links.

Definition 2: (Best Response) Given a residual wiring S_{-i}, a best response for node v_i is a wiring $s_i \in S_i$ such that $C_i(S_{-i} + \{s_i\}) \leq C_i(S_{-i} + \{s'_i\})$, $\forall s'_i \neq s_i$.

Definition 3: (Stable Wiring) A global wiring S is stable iff it is composed of individual wirings that are best responses.

Therefore stable wirings are pure Nash equilibria of the SNS game, *i.e.*, they have the property that no node can re-wire unilaterally and reduce its cost. Fundamentally different is the work on Selfish Routing [14], [16], in which the network topology is part of the input to the game, and selfish source routing is the outcome. In a way, this is the inverse of our work, in which network-based (shortest-path) routing is an input of the game, and topology is the outcome.

III. Deriving Stable Wirings

A wiring for a node v_i can be defined using $n-1$ binary unknowns Y_l, $1 \leq l \leq n, l \neq i$: $Y_l = 1$ iff v_i wires to v_l, and 0 otherwise. Define also the binary unknowns X_{lj}: $X_{lj} = 1$ iff v_i has v_l as a first-hop neighbor on a shortest path to v_j. A best response for v_i under residual wiring S_{-i} can be obtained by solving the following Integer Linear Program (ILP):

Minimize:

$$C_i(S_{-i}, X) = \sum_{j=1, j\neq i}^{n} p_{ij} \sum_{l=1, l\neq i}^{n} X_{lj} \cdot (d_{il} + d_{S_{-i}}(v_l, v_j)) \quad (1)$$

Subject to:

$$\sum_{l=1, l\neq i}^{n} X_{lj} = 1, \forall j \neq i \text{ and } \sum_{l=1, l\neq i}^{n} Y_l = k_i \text{ and } X_{lj} \leq Y_l, \forall l, j \neq i, \quad (2)$$

where d_{il} is the cost of a wire from v_i to v_l, and $d_{S_{-i}}(v_l, v_j)$ is the cost of a shortest path from v_l to v_j over the wiring S_{-i}. For the special case where the link costs are identical the best response of a node is the solution of the k-median problem on asymmetric distance as we show in the next section. For general link costs, as we showed in the ILP formulation, the link cost of a node to connect to other nodes has to be taken into account.

A. Connections between the SNS game and Facility Location

When all the wires have the same unitary weight, then the distances d_S are essentially "hop counts", in which case there is an interesting relationship between finding a node's best response wiring and solving a *k-median problem on asymmetric distance* [25], [32]. The latter is defined as follows:

Definition 4: (Asymmetric k-median) Given a set of nodes V', weight's w_j, $\forall v_j \in V'$, and an asymmetric distance function $d_{S'}$ (meaning that in general $d_{S'}(v, u) \neq d_{S'}(u, v)$), select up to k nodes to act as medians so as to minimize $C(V', k, w)$, defined as follows:

$$C(V', k, w) = \sum_{\forall v_j \in V'} w_j \cdot d_{S'}(v_j, m(v_j)),$$

where $m(v_j)$ is the median that is closest to v_j.

Proposition 1: The best response of node v_i to S_{-i} under uniform link weights ($d_{ij} = 1, \forall i, j \in V$) can be obtained by solving an asymmetric k-median problem, in which:

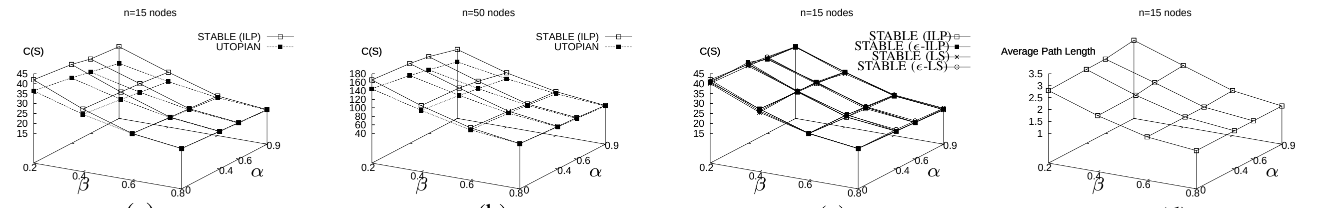
1) $V' = V - \{v_i\}$
2) $k = k_i$
3) $w_j = p_{ij}$, $v_j \in V'$
4) $d_{S'}(u, w) = d_{S_{-i}}(w, u)$, $u, w \in V'$,

Proof: Let s_i denote v_i's response to S_{-i}. The resulting cost will be:

$$\begin{aligned}
&C_i(S_{-i} + \{s_i\}) = \sum_{v_j \in V'} p_{ij} d_{S_{-i}+\{s_i\}}(v_i, v_j) \\
&= \sum_{v_j \in V'} p_{ij} (d_{S_{-i}+\{s_i\}}(v_i, m(v_j)) + d_{S_{-i}+\{s_i\}}(m(v_j), v_j)) \\
&= \sum_{v_j \in V'} p_{ij} d_{S_{-i}+\{s_i\}}(v_i, m(v_j)) + \sum_{v_j \in V'} p_{ij} d_{S_{-i}+\{s_i\}}(m(v_j), v_j) \\
&= \sum_{v_j \in V'} p_{ij} + \sum_{v_j \in V'} p_{ij} d_{S_{-i}+\{s_i\}}(m(v_j), v_j) \\
&= \sum_{v_j \in V'} w_j + \sum_{v_j \in V'} w_j d_{S_{-i}}(m(v_j), v_j) \\
&= c + \sum_{v_j \in V'} w_j d_{S'}(v_j, m(v_j))
\end{aligned} \quad (3)$$

where c is a constant and $m(v_j)$ is v_i's next-hop neighbor on a shortest path to v_j under the global wiring $S_{-i} + \{s_i\}$. The transition from the third to the fourth line of Equation (3) relies on the fact that all distances to first hop neighbors are equal to 1 under hop-count distance. Obtaining the best response requires minimizing $C_i(S_{-i} + \{s_i\})$. Equation (3) shows that this is equivalent to minimizing $\sum_{v_j \in V'} w_j d_{S'}(v_j, m(v_j))$, which is exactly the objective function of the above mentioned asymmetric k-median problem. ∎

Proposition 1 suggests that v_i's best response is to wire to the k_i medians of a distance function obtained by reversing the end-to-end distances of the residual wiring S_{-i}. Since even the metric version of k-median is NP-hard [32], so is



(a) (b) (c) (d)

Fig. 1. (a) Comparison of the social cost $C(S)$ of stable wirings to the cost of a socially optimal (utopian) solution for $n = 15$. Stable wirings obtained using exact best (b) same as (a) with $n = 50$. (c) Comparison of the social cost of stable wirings obtained by using exact (ILP) and approximate (LS) best response with $\rho = 1$ and corresponding $\epsilon = 5\%$ versions. (d) Average path length for the stable graph obtained by using exact (ILP) best response.

its asymmetric version, and through Proposition 1 the best response of the SNS game as well. For the metric version of the k-median there exist several algorithms that provide constant-factor approximations (in a polynomial number of iterations) of an exact solution [33], [34], [35], [36]. These guarantees do not hold for the asymmetric case. For the asymmetric k-median, Lin and Vitter [37] have given a bi-criterion approximation that blows up the number of used medians by an $O(\log n)$ multiplicative factor to provide a cost that exceeds the optimal one by an additive factor. Archer [25] has shown that this is the best attainable approximation for this problem unless $NP \subseteq DTIME(n^{O(\log\log n)})$. Despite this negative result, simple heuristics like ρ-swapping local search [35] perform typically very well as we will show later.

B. Stable Wirings through Iterative Best Response

We obtain stable wirings through a simple iterative best response method in which nodes apply iteratively their best response until no unilateral improvement can be obtained. In Section IV we present synthetic results based on hop-count distance. We take advantage of the connections established through Proposition 1, and we employ exact (ILP) and approximate (ρ-swapping local search [35]) solutions for the directed k-median in order to obtain best responses. In Section V employ the ILP formulation of Section III in order to obtain best responses in several real topologies.

IV. Performance Evaluation of Stable Wirings

In this section we assume that establishing a direct overlay link between any two nodes incurs unit cost and, therefore, the cost between any pair of nodes equals the number of hops along any shortest, directed path that connects these nodes at the overlay layer. Our goal is to evaluate the performance of stable wirings with respect to two key scaling parameters.

The first parameter, $\alpha \in [0, 1]$, reflects the skew in the popularity of different destinations. The space of possible combinations of pair-wise preference is large. To quantify the effect of preference profile on stable wiring performance we assume that a homogeneous preference profile. We will relax this assumption by using passive and active network measurements in Sections V and VII respectively. The popularity of the ith most popular node is $q_i = \Lambda/i^\alpha$, where $\Lambda = (\sum_{k=1}^{n} \frac{1}{k^\alpha})^{-1}$. We construct the preference vector p_i of node v_i by setting $p_{ij} = q_j/(1 - q_i), \forall v_j \in V : v_j \neq v_i$.

The second parameter, $\beta \in [0, 1]$, determines the *link density* of a regular graph, which relates to the fanout (out-degree) of each node as follows: $k = \lceil n^\beta \rceil$.

For a given pair (α, β) we obtain the corresponding stable wiring by using the iterative best response method of Section III-B, where the best response amounts to a solution of a directed k-median problem. Here, it is worthwhile to notice that different node orderings in the iterative best response search may lead to different stable wirings.We have found that different stable wirings perform approximately the same. We also observed that the stable wirings obtained for the same value of β have similar structure for different values of α [38].

A. Social Cost of Stable Wirings

To study the quality of stable wirings, we compare their social cost with that of socially optimal wirings. Let S^* denote a socially optimal (SO) wiring, *i.e.*, a global wiring that minimizes the *social cost* $C(S) = \sum_{\forall v_i \in V} C_i(S)$. Let $S^{U,i}$ denote the *utopian* wiring for v_i, *i.e.*, the global wiring that minimizes $C_i(S)$ over all possible global wirings S (this should not be confused with a best response s_i that minimizes $C_i(S_{-i} + \{s_i\})$ granted a particular residual wiring S_{-i}). Due to lack of space, we show how we obtain a lower bound of the social cost of the above mentioned utopian wiring in [38].

As can be seen for the examples depicted in Figure 1 (a) and (b), which are representative of a much larger set of simulations we conducted [38], the gap between the stable solution and the utopian solution is small, and this result holds across a wide range of settings for α and β, and for various values of n for which simulation was tractable. In terms of absolute values, the social cost decreases with both the skew in popularity and link density. In particular, a highly-skewed popularity profile ensures that shorter paths to the most popular destinations are realized, whereas higher link densities reduces the average length of shortest paths, and thus the social cost as well. Turning our attention on the structure of stable wirings, we found that popular nodes have high in-degree, but non-popular nodes may also have high in-degree in order to provide good global connectivity to the rest of the nodes [38].

Since computing exact best response wirings is NP-hard, even under hop-count distance, we turn to *approximate best responses* and corresponding *approximately stable wirings*. For this purpose, we used the ρ-swapping Local Search (LS) heuristics, where each node can replace up to ρ of its neighbors [35], to solve the k-median problem which yields the best response wiring by virtue of Proposition 1. We also considered ϵ-stable versions of the problem in which nodes do not re-wire unless they can reduce their current cost by *at least* a multiplicative factor ϵ (we combined ϵ-stability with both exact (ILP) and approximate (LS) best responses). As evident from Figure 1 (c), we found that ϵ-stable wirings have similar social costs [38].

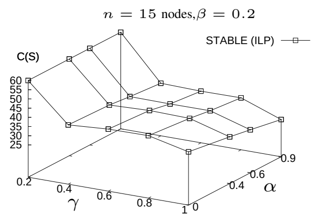


Fig. 2. The social cost of doubly capacitated stable wirings for $\beta = 0.2$ and variable α, γ.

To summarize, stable wirings have performance close to the socially optimal wirings. Moreover, approximate best response wirings can be computed fast with LS and ϵ approximations. On a computational note, in the presence of non-uniform preference profile stable graphs were found within a small number of iterations. Due to lack of space we provide more details on the convergence speed and the structure of stable wirings in [38]. In support to our results, we note that in a recent work [39], it has been established analytically that provably existent stable wirings are guaranteed to perform approximately as well as socially optimal solutions under uniform node popularity. A similar conclusion is reached in the next section (albeit experimentally) for the case of non-uniform popularity. We also find that the average path length slowly increases with α for a given β (see Figure 1 (d)).

B. Constraining the In-degree: A Doubly Constrained Overlay

We next examine the effects of constraining the maximum in-degree of nodes so that they never have more than ν incoming links, while maintaining also the constraint on the out-degree. We can enforce this constraint by including in the definition of $C_i(S)$ a large penalty for connecting to nodes that have more than $\nu - 1$ incoming links. We can define a scaling factor γ for the in-degree as done previously with β for the out-degree.

In Figure 2, we fix the out-degree scaling parameter to $\beta = 0.2$, and present the social cost for different values of the in-degree scaling parameter γ. Low values of γ increase the social cost under skewed popularity profiles, as in these cases, the highly-popular nodes quickly reach their maximum in-degree and thus, many nodes have to reach them indirectly through multi-hop paths. Note that without in-degree constraints most nodes would access them in a single hop by establishing a direct overlay link to them. When γ is low, *e.g.*, $\gamma = 0.2$, the resulting graph looks much like a ν-regular graph. With large values of γ, *i.e.*, γ approaching 1, the in-degree constraints become too loose and, thus, the corresponding stable graphs become similar to their unconstrained counterparts.

V. Best Response vs. Heuristics in Mixed Overlays

In this section we take a closer look at the performance benefits from employing best response instead of heuristic wiring strategies. We also depart from simplistic unit-distance, the homogeneous preference profile, and the assumption that all nodes apply the same wiring strategy. With regards to the preference profile, we let it derive directly from the pair-wise distance of nodes. We do this by setting $p_{ij} = 1$ in Equation 1.

A. Description and Design Methodology

In Section III, we defined the best response strategy for a node entering a given network. Now, we consider three other natural alternatives. Let d_{ij}^X denote the cost associated with creating a direct overlay link between nodes v_i and v_j under a model X for end-to-end IP layer distances. We say that a "newcomer" node v_i employs a k-*Closest* wiring strategy under the model X when it establishes a wiring s_i such that $d_{ij}^X \leq d_{ij'}^X$ for all $v_j \in s_i, v_{j'} \notin s_i$. We say that a newcomer node v_i employs a k-*Random* wiring strategy when it chooses a wiring s_i uniformly at random from the space of all valid wirings of cardinality k_i. A newcomer node v_i employs a k-*Regular* wiring strategy if it follows a pre-defined wiring pattern, based on node identifiers (IDs), like every other node in the network. Unless otherwise noted, the common wiring pattern is the one described in Chord [8].

To substantiate the benefits of best response, we consider the initial graph awaiting a "newcomer" upon its arrival. We assume that this initial graph has resulted from having its constituent nodes apply a specific wiring strategy.[1] We refer to an instance of an n node graph for which each of the n nodes employed a k-Closest strategy as a k-*Closest* graph, and attribute similar meanings to a k-*Random* graph, a k-*Regular* graph and a *Best Response (BR) graph.*

B. Description of the Datasets

In this section we describe the IP-layer end-to-end distance models X from which we obtain the d_{ij}^X's that are used as weights for direct overlay links between nodes v_i and v_j. Overlay nodes that do not have a direct link communicate through a shortest-path on the overlay topology. The following three datasets are used:

BRITE: The first dataset is synthetically generated from the BRITE topology generator [40] following a Barabási-Albert model [41] with N=1000 nodes and incremental growth parameter μ=2. The nodes were placed on the plane according to a heavy tail model that creates high density clusters. Based on the observation that the delay between two nodes in high speed networks is highly correlated to their physical distance [42], we assigned weights on the links at the physical layer by calculating the pairwise Euclidean distance.

PlanetLab: PlanetLab is an overlay testbed network of approximately 700 nodes in more than 300 academic, industrial, and government sites around the world. We used a publicly available dataset[2] containing delays obtained using *pings* between all pairs of PlanetLab sites (inter-site delays are more representative than inter-node delays for overlay applications).

AS-level map: As a third dataset, we use the relation-based AS topology map of the Internet.[3] It includes 12779 unique ASes, of which 1076 are peers (joined by at least one peer-peer link), and the remaining 11703 are customers. These ASes are connected through 26387 directed and 1336 undirected links.

C. Comparison of Different Graphs

Using as input the weighted graphs from our three datasets, we obtained the social costs resulting from applying the various wiring strategies under consideration, for different

[1] To guarantee connectivity, nodes that participate in a k-Random or a k-Closest graph, donate one link in order to create a ring. We note that a ring is a feature common to many other overlays, such as the Chord DHT [8].

[2] `http://ping.ececs.uc.edu/ping`

[3] `http://www.cc.gatech.edu/~mihail/ASdata.html`

	$\beta = 0.1$			$\beta = 0.2$			$\beta = 0.4$			$\beta = 0.6$			$\beta = 0.8$		
	k-Random/BR	k-Regular/BR	k-Closest/BR	k-Random/BR	k-Regular/BR	k-Closest/BR	k-Random/BR	k-Regular/BR	k-Closest/BR	k-Random/BR	k-Regular/BR	k-Closest/BR	k-Random/BR	k-Regular/BR	k-Closest/BR
BRITE	1.44	3.61	1.53	1.52	2.31	1.84	1.38	1.50	2.07	1.28	1.11	1.46	1.09	1.03	1.16
PlanetLab	2.23	3.84	1.48	1.75	2.74	1.23	1.37	2.10	1.13	1.09	1.41	1.16	1.04	1.18	1.06
AS-level	2.04	4.78	1.90	1.83	2.86	1.61	1.58	2.37	1.39	1.24	1.10	1.23	1.12	1.12	1.16

TABLE I

SOCIAL COST RATIOS BETWEEN HEURISTIC WIRING STRATEGIES AND BEST RESPONSE.

values of β. The BR graph (resulting from having all nodes apply the best response wiring strategy) was by far the most optimized wiring, thus providing a lower-bound for the simpler k-Random and k-Closest strategies. Table I summarizes our results by providing the ratios of the social costs of the heuristic wiring strategies (k-Random, k-Regular, k-Closest) to that of the BR wiring. These results suggest that the premium provided by BR is highest for lower link densities (*i.e.*, when β is small). This is an intuitive result since in denser graphs, there is less of an opportunity for optimization.

The results in this section give us a baseline for the efficiency of the wirings that result from the adoption by all nodes in the graph of the same strategy. This sets up the stage for our next set of questions: Given such an initial wiring, what is the marginal utility to a newcomer from executing each one of the three wiring strategies under consideration?

D. The Value of Best Response

Given an initial wiring created by having n overlay nodes follow one of our four wiring strategies, we quantify the benefit to a "newcomer" from choosing its neighbors using one of the four neighbor selection strategies. Twelve possibilities exist for applying strategy S1 over a wiring obtained using S2, where S1 and S2 could be k-Random, k-Closest, k-Regular or BR. We use $c(w)$ to denote the cost of a newcomer using wiring strategy w on a pre-existing graph.

In the results presented below, we set $n = 50$ and evaluate the performance for 200 newcomers on the BRITE and AS dataset and 100 newcomers for the PlanetLab dataset (which is smaller). Our main results are shown in Figure 3, where each column corresponds to an underlying graph model, and each row corresponds to a strategy employed by the n newcomers. Within each plot, we vary the link density β along the x-axis, and plot the cost ratio of the newcomer for a given strategy versus the cost of the newcomer if it were to use BR.

Connecting to a k-Random Graph: The plots in the top row of Figure 3 show the case in which the first n arrivals use k-Random, and thus the underlying graph is poorly optimized.

With such an initial graph, the k-Random wiring is a poor choice for the newcomer, as it could lead to significantly higher costs. This performance gap closes, as one would expect, when β (and therefore k) becomes large. In fact this trend holds in all cases because finding a closer approximation to BR is easier when each node has more links — and therefore ample opportunity to make good connections, even when using simple strategies. The performance of k-Regular wiring is similar to the k-Random one, as IDs are randomly assigned.

Using the k-Closest wiring, on the other hand, turns out to be a very reasonable choice, as it achieves a cost comparable to that achieved by BR (typically within 15% low link densities). This finding suggests that in poorly optimized random graphs, simply connecting to your nearby neighbors (at low cost), is a good rule of thumb, especially when edge density is high.

Connecting to a k-Regular Graph: The plots in the second row of Figure 3 show the case in which the first n arrivals use k-Regular, and thus the underlying graph is a structured one, where each node follows the same wiring pattern. Here we see again that a BR wiring pays off. The performance of k-Closest and k-Random improve as the graph becomes denser. k-Regular turns out to be a the worst choice (the range on values of newcomer's cost ratio is now higher). because structured graphs seem to eliminate the number of shortcuts.

Connecting to a k-Closest Graph: The plots in the third row of Figure 3 show the case in which the first n arrivals use k-Closest, and thus the underlying graph consists mostly of local edges with few shortcuts. Here we see that it is considerably more important for newcomers to behave strategically. For example, on the BRITE topology, using k-Closest is a poor choice that perpetuates the lack of shortcuts in the underlying graph to the point that even using k-Random or k-Closest turns out to be a better choice. In the other topologies, k-Closest, k-Random, and k-Regular are comparable, and the improvement in quality relative to BR as β increases is much more modest.

The above suggest that although *it pays to "cheat"*, and *e.g.*, ping the possible neighbors and connect to the k-Closest ones, instead of k random ones as as the other nodes do, if the other nodes also cheat, then a new node may actually be better of by sticking to the protocol and getting neighbors randomly.

Connecting to a Stable Graph: Finally, the plots in the bottom row of Figure 3 show the case in which the first n arrivals use BR, and thus the underlying graph ends up being highly optimized, prior to the arrival of newcomers. In this case, the graph is so much optimized for the newcomer that any reasonable strategy might well have good performance. Surprisingly, while the k-Closest strategy does indeed perform well for the newcomer across the three topologies, the alternative strategies of k-Random and k-Regular do not. This seemingly odd result could be explained by noting that given the very low overall costs between nodes in the optimized initial graph, the cost to the newcomer from selecting its own neighbors plays an important role.

General Observation: In conclusion, we find common trends across the three topologies with respect to strategic neighbor selection behavior. At the two extremes where the other players are playing completely at random or completely selfishly (top and bottom rows, respectively), the underlying graphs are either too poorly constructed, or too well constructed, for an uninformed newcomer to be at a significant disadvantage. In either of these two situations, the myopic strategy of k-Closest

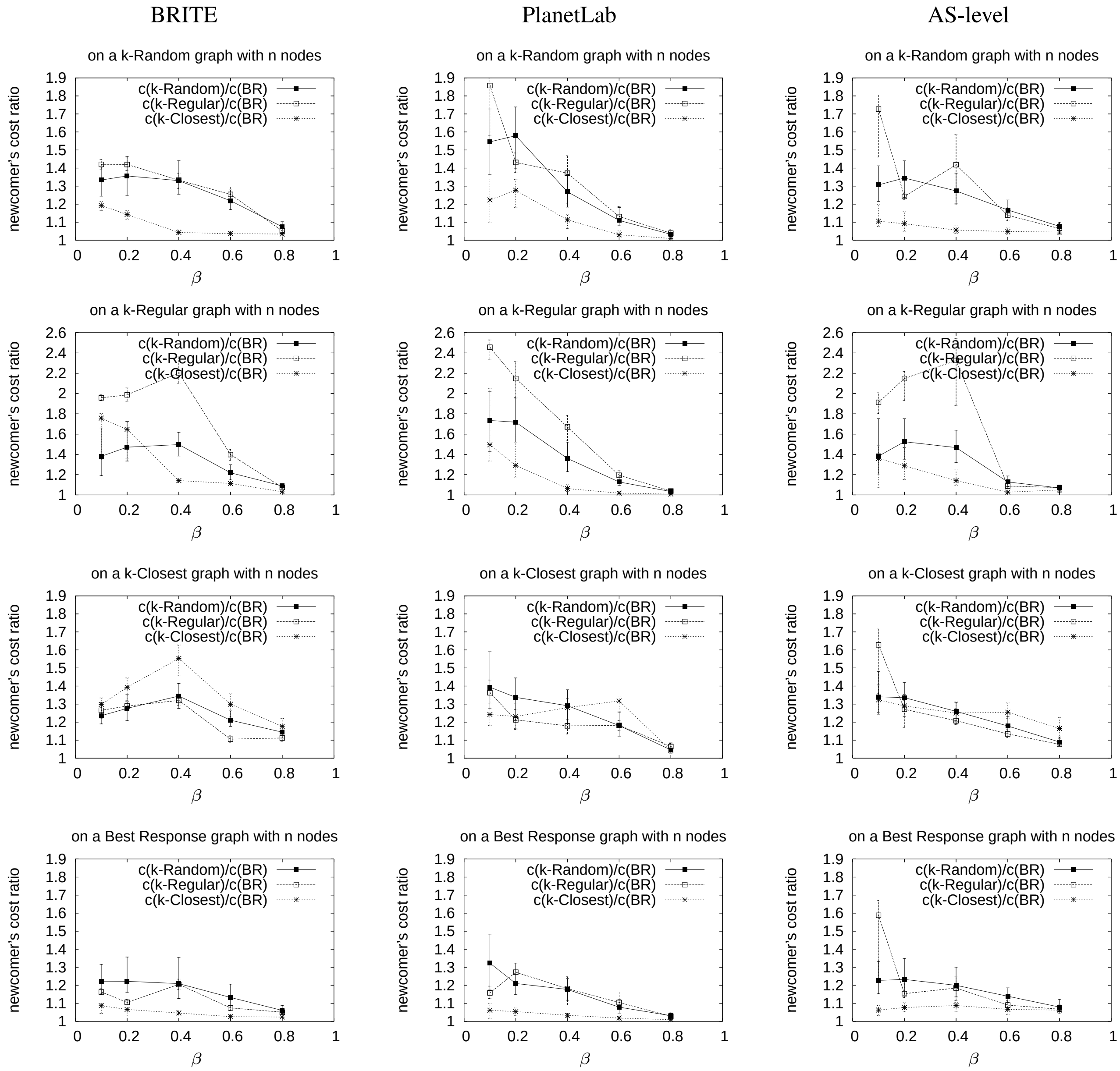


Fig. 3. The cost ratio between heuristic wirings and BR wiring for a newcomer node that connects to a pre-existing network of n nodes that was wired using k-Random, k-Regular, k-Closest, or BR. We present the 25-, 50-, 75-quartiles for the aforementioned ratios using three different data sets.

is generally competitive to BR, especially under stable graphs. Picking links at random in these situations however, is unlikely to work well, unless the graph is already dense (large β).

But in the middle regime, in which all the other players adopt k-Closest the newcomer must be much more careful. Here, there is much to be gained by the optimal shortcuts selected in BR, which neither k-Closest nor k-Random typically selects. Our experimental results suggest that k-Closest is one of the worse the possible strategies considered for the newcomer to adopt in this situation. Strikingly, our results advocate that the k-Regular is actually the worst of the possible strategies considered for the newcomer. Structured overlays seem to reduce to the minimum the number of shortcuts. Due to lack of space we provide a larger set of experiments in [38].

VI. The EGOIST Overlay Routing System

The previous results have shown that no simple heuristic strategy can keep up with the performance of best response across the entire range of considered scenarios. What is not clear, however, is whether it is practical to build overlays to support best response and how to incorporate additional metrics other than delay, *e.g.*, bandwidth. It is also unclear what is the average performance gain when SNS wiring strategies are used in highly dynamic environments, whether such overlays are robust against churn, and whether they scale. We address the questions mentioned above by describing the design and implementation of EGOIST: an SNS-inspired prototype overlay routing network. EGOIST serves as a building block for the distributed construction of efficient and resilient overlays where both individual and social performance is close to optimal.

A. Basic Design

EGOIST is a distributed system that allows the creation and maintenance of an overlay network, in which every node selects and continuously updates its k overlay neighbors in a selfish manner—namely to minimize its (weighted) sum of distances to all destinations under shortest-path routing. For ease of presentation, we will assume that *delay* is used to reflect the cost of a path, noting that other metrics – which we will discuss later in the Section and which are incorporated in EGOIST's implementation – could well be used to account for cost, including bandwidth and node utilization.

In EGOIST, a *newcomer* overlay node v_i connects to the system by querying a *bootstrap* node, from which it receives a list of *potential* overlay neighbors. The newcomer connects to at least one of these nodes, enabling it to participate in the link-state routing protocol running at the overlay layer. As a result, after some time, v_i obtains the full residual graph G_{-i} of the overlay. By running all-pairs shortest path algorithm[4] on G_{-i}, the newcomer is able to obtain the pair-wise distance (delay) function $d_{G_{-i}}$. In addition to this information, the newcomer estimates d_{ij}, the weight of a potential direct overlay link from itself to node v_j, for all $v_j \in V_{-i}$. Using the values of d_{ij} and $d_{G_{-i}}$, the newcomer connects to G_{-i} using one of of the wiring strategies discussed in Section V. In our implementation, each node acts as a server that listens to all the messages of the link state protocol and propagates them only to its immediate neighbors. In order to reduce the traffic in the system, each node propagates only unique messages by dropping messages that have been received more than once or have been superseded. There are also two threads, one for estimating d_{ij}, and one responsible for estimating the new wiring and propagating the wiring to the immediate neighbors. In order to minimize the load in the system, a node propagates its wiring to its immediate neighbors only if this changes.

B. Dealing with Churn

EGOIST's BR neighbor selection strategy assumes that existing nodes never leave the overlay. Therefore, even in an extreme case in which some nodes are reachable through only a unique path, a node can count on this path always being in place (re-wirings by other nodes will not tear it down as this would also disconnect them [39]). Overlay routing networks (*e.g.*, RON [4]) are not inherently prone to churn to the extent that file-sharing P2P networks [43], [44] are. Nonetheless, nodes may occasionally go down, or network problems may cause transient disconnections until successive re-wirings establish new paths. One could re-formulate the BR objective function used by a node to take into account the churning behavior of other nodes. This, however, requires modeling of the churn characteristics of various nodes in an overlay, which is not feasible in large networks [27].

In EGOIST, we follow a different approach reminiscent of how k-Random and k-Closest strategies ensure overlay connectivity. We introduce a hybrid wiring strategy (HybridBR), in which each node uses k_1 of its k links to selfishly optimize its performance using BR, and "donates" the remaining $k_2 = k - k_1$ links to the system to be used for assuring basic connectivity under churn. We call this wiring "hybrid" because, in effect, two wiring strategies are in play – a selfish BR strategy that aims to maximize local performance and a selfless strategy that aims to maintain global connectivity by providing redundant routes.

There are several ways in which a system can use the k_2 donated links of each node to build a connectivity backbone. Young et al. [45] proposed the use of k Minimum Spanning Trees (k-MST). Using k-MST (a centralized construction) to maintain connectivity is problematic, as it must always be updated (due to churn and to changes in edge weights over time), not to mention the overhead and complexities involved in establishing $(k_2/2)$-MSTs. To avoid these complexities, EGOIST uses a simpler solution that forms $k_2/2$ bidirectional cycles. For $k_2 = 2$, it allows for the creation of a single bidirectional cycle. For higher k_2, the system decides $k_2/2$ *offsets* and then each node connects to the nodes taken by adding (modulo n) its id to each offset. If k_2 is small (*e.g.*, 2) then the nodes will need to monitor (*e.g.*, `ping`) the backbone links closely so as to quickly identify and restore disconnections. With higher k_2 the monitoring can be more relaxed due to the existence of alternative routes through other cycles. Computing BR using k_1 links *granted* the existence of the k_2 links can be achieved by restricting the set candidate candidate immediate neighbors for swapping.

We have implemented HybridBR in EGOIST. As hinted above, donated links are monitored aggressively so as to recover promptly from any disconnections in the connectivity backbone through the use of frequent heartbeat signaling. On the other hand, the monitoring and upkeep of the remaining BR links could be done lazily, namely by measuring link costs, and recomputing BR wirings at a pace that is convenient to the node—a pace that reduces probing and computational overheads without risking global connectivity.

To differentiate between these two types of link monitoring strategies (aggressive versus lazy), in EGOIST we allow re-wiring of a dropped link to be performed in one of two different modes: *immediate* and *delayed*. In immediate mode, re-wiring is done as soon as it is determined that the link is dropped, whereas in delayed mode re-wiring is only performed (if necessary) at the preset *wiring epoch* T. Unless otherwise specified, we assume a delayed re-wiring mode is in use.

C. Cost Metrics

As alluded earlier, the choice of an appropriate "cost" of traversing a link depends largely on the application at hand. In EGOIST we consider the following metrics:

Link and Path Delays: Delays are natural cost metrics for many applications, including real-time ones. To obtain the delay cost metric, a node needs to obtain estimates for its own delay to potential neighbors, and for the delay between pairs of overlay nodes already in the network. In EGOIST, we estimate directed (one-way) link delays using two different methods: an active method based on `ping`, and a passive method using the `pyxida` virtual coordinate system [19]. Using `ping`, one-way delay is estimated to be one half of the measured `ping` round-trip-times (RTT) averaged over enough samples. Clearly, a node is able to measure such a value for all of its direct (overlay) neighbors, and is also able to relay such information to any other nodes through the overlay link-state routing protocol. To estimate the distance to nodes that were configured not to reply to `ping`, we used application layer

[4] Given than the graph is sparse, we used the most efficient implementation of Dijkstra algorithm using Fibonacci heap that requires $O(|E|+|V|\log|V|)$ amortized time, where $|E|$ is the number of edges in the graph.

ping. Using pyxida, delay estimates are available through a simple query to the pyxida system. Using ping produces more accurate estimates, but subjects the overlay to added load, whereas using pyxida produces less accurate estimates, but consumes much less bandwidth.

Node Load: For many overlay applications, it may be the case that the primary determinant of the cost of a path is the performance of the nodes along that path—*e.g.*, if traversal of nodes along the path incur significant overhead due to (say) context switching and frequent crossing of user/kernel spaces. Thus, in EGOIST, we allow the use of a variation of the delay metric in which all outgoing links from a node are assigned the same cost, which is set to be equal to the measured load of the node. When applicable, the estimation of such a metric is straightforward as it requires only local measurements. In EGOIST, we did this by querying the CPU load of the local PlanetLab node, and computing an exponentially-weighted moving average of that load calculated over a given interval (taken to be 1 minute in our experiments querying the loadavg reports).

Available Bandwidth: Another important cost metric, especially for content delivery applications, is the available bandwidth on overlay links. Different available bandwidth estimation tools have been proposed in the literature [46]. In EGOIST, we used pathChirp [47], a light-weight, fast and accurate tool, which fits well with PlanetLab-specific constraints, namely: it does not impose a high load on PlanetLab nodes since it does not require the transmission of long sequences of packet trains, and it does not exceed the max-burst limits of Planetlab. pathChirp is an end-to-end active probing tool, which requires the installation of sender and receiver pathChirp functionality in each EGOIST node. The available bandwidth between a pair of nodes $v, u \in V_{-i}$ is given by: $AvailBW(v, u) = \max_{p \in P(v,u)} AvailBW(p)$, where the available bandwidth for a path p is given by: $AvailBW(p) = \min_{e \in p} AvailBW(e)$, and $P(v, u)$ denotes the set of paths that connects v to u. Thus, finding $P^*(v, u)$ that maximizes the available bandwidth between v and u, and the bottleneck edge, is a "Maximum Bottleneck Bandwidth" [48] problem which can be solved using a simple modification of Dijkstra's algorithm.

VII. Performance Evaluation of EGOIST

In this section, we present performance results obtained through measurement of EGOIST. These results allow us to make comparisons between the neighbor selection strategies described in Section V for the various cost metrics described above. At first, we present our results assuming that there is no node churn. Results showing the impact of node churn on EGOIST performance are presented in Section VII-B.

Experimental Setting: We deployed EGOIST on n=50 PlanetLab nodes (30 in North America, 11 in Europe, 7 in Asia, 1 in South America, and 1 in Oceania) and collected performance statistics for more than a year. Each of these nodes is configured to recompute its wiring every wiring epoch T=60 seconds. EGOIST nodes are not synchronized, thus on average a re-wiring by some EGOIST node occurs every T/n=1.2 seconds. Whether a node ends up re-wiring or not depends on the neighbor selection strategy. For k-Random and k-Regular strategies, and since our baseline experiments do not feature any node churn, it follows that these strategies will not exhibit any re-wiring. For k-Closest, re-wiring would only be the result of dynamic changes in PlanetLab that result in changes to the cost metric in use. For BR, a node may re-wire due to changes in PlanetLab conditions, but may also re-wire simply as a result of another node's re-wiring.

To be able to compare the impact of neighbor selection on the quality of the resulting overlay, throughout this paper we use the *routing cost* (for an individual node or averaged over all nodes) as the main performance metric. For each experiment, an individual cost metric is calculated for every one of the n=50 nodes in the system. The individual cost metric for a node reflects the cost of routing from that node to all other 49 nodes in the system, assuming a uniform routing preference over all destinations (the preference vector depends on the value of the metric that is used). For each experiment we report the mean of all n=50 individual costs, as well as the 95^{th}-percentile confidence interval. To facilitate comparisons between various neighbor selection strategies, we often report the *normalized routing cost* (and the 95^{th}-percentile confidence interval), which is the ratio of the cost achievable using a given strategy to that achievable using BR.

Control Variables: In our first set of experiments, our aim is to identify for the three metrics of interest the payoff (if any) from adopting a selfish neighbor selection strategy, *i.e.*, using a BR strategy in EGOIST. This payoff will depend on many variables. While some of these variables are *not* within our control (*e.g.*, the dynamic nature of the Internet as reflected by variability in observed PlanetLab conditions), others are within our control, *e.g.*, n, T, and the various settings for our active measurement techniques.

One control variable that is particularly important is the number of direct neighbors, k, that an EGOIST node is allowed to have. In many ways, k puts a premium on the significance of making a judicious choice of neighbors. For small values of k, choosing the right set of neighbors has the potential of making a bigger impact on performance, when compared to the impact for larger values of k.

In order to neutralize the effect of extrinsic variables that are not within our control, experiments reporting on different neighbor selection strategies were conducted *concurrently*. To do so, we deploy concurrent EGOIST agents on each of the n=50 PlanetLab nodes we use in our experiments, with each agent using a different selection strategy. In effect, each experiment compares the performance of a *set* of concurrently deployed EGOIST overlay networks, each resulting from the use of a particular neighbor selection strategy.

Overview of Performance Results: Before presenting specific performance results, we make two broad observations: first, in all of our experiments, using a BR strategy in EGOIST consistently yields the best performance. While such an outcome was anticipated by virtue of findings reported in the previous Section V-D for a static setting, the results we present here are significant because they underscore the payoff in a *real* deployment, where the modeling assumptions made in prior work do not hold. Second, in all of our experiments, with the exception of BR, no single neighbor selection strategy was consistently better than all others across all metrics. While the

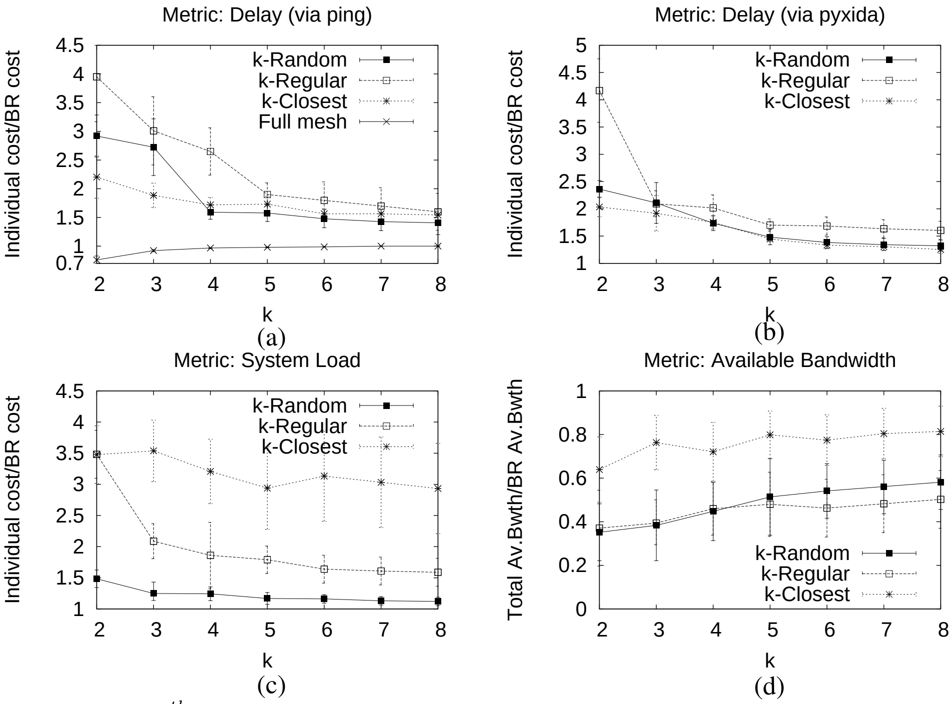


Fig. 4. Normalized individual costs and 95^{th}-percentile confidence intervals with respect to BR cost, under different metrics, of various neighbor selection strategies in a 50-node EGOIST overlay.

performance of a given strategy may approach that of BR for one metric while dominating all other strategies, such strategy dominance does not hold across all the metrics.

Results for Delay Metric: Figures 4 (a) and (b) show the performance of the various neighbor selection strategies in EGOIST normalized with respect to that achievable using BR when the metric of interest is the overlay path delay over a range of values for k (using `ping` and `pyxida`). These results show that BR outperforms all the other wiring strategies, especially when k is small. For k=2, the average delay experienced by an individual node could be anywhere between 200% and 400% *higher* than that achievable using BR. The performance advantage of BR in terms of routing delay stands, even for a moderate number of neighbors. For example, for k=5, BR cuts the routing delay almost by half.

These results confirm the superiority of BR relative to other strategies, but do not give us a feel for how close is the performance of EGOIST using BR wiring to the "best possible" performance. To do so, we note that by allowing nodes to connect to all other nodes in the overlay, we would be creating a complete overlay graph with $O(n^2)$ overlay links, obviating the need for a neighbor selection strategy. Clearly, the performance of routing over such a rich overlay network gives us an *upper bound* on the achievable performance, and a lower bound on the delay metric. Thus, to provide a point of reference for the performance numbers we presented above, in Figure 4 (a) we also show the performance achieved by deploying EGOIST and setting k=n-1. Here we should note that this lower bound on delay is what a system such as RON [4] would yield, given that routing in RON is done over shortest paths established over a full mesh, and assuming that any of the $O(n^2)$ overlay links could be used for routing. These results show that using BR in EGOIST yields a performance that is quite competitive with RON's lower bound. As expected, the difference is most pronounced for the smallest k we considered—namely, the lowest delay achievable using 49 overlay links per node is only 30% lower than that achievable using BR with 2 overlay links per node. BR is almost indistinguishable from the lower bound for slightly larger values of k (*e.g.*, k=4).

With respect to the other heuristics, the results in Figures 4 (a) and (b) show that k-Closest outperforms k-Random when k is small, but that k-Random ends up outperforming k-Closest for slightly larger values of k. This can be explained by noting that k-Random ends up creating graphs with much smaller diameters than the grid-like graphs resulting from the use of k-Closest, especially as k gets larger. In all experiments, k-Regular performed the worst. In [38] we also show that BR wiring strategy is robust to cheating.

Results for Node Load: Figure 4 (c) shows the results we obtained using the node load metric, where the path cost is the sum of the loads of all nodes in the path. These results show clear delineations, with BR delivering the best performance over all values of k, k-Random delivering the second-best performance, and k-Closest delivering the worst performance as it fails to predict anything beyond the immediate neighbor, especially in light of the high load variance in PlanetLab.

Results for Available Bandwidth: Figure 4 (d) shows the results we obtained using available bandwidth as the cost metric. Recall that, here, the objective is to get the highest possible *aggregate* bandwidth to all destinations (again, assuming a uniform preference for all destinations) – thus, larger is better. These results show trends that are quite similar to those obtained for the delay metric, with BR outperforming all other strategies—delivering a two-fold to four-fold improvement over the other strategies, over a wide range of values of k.

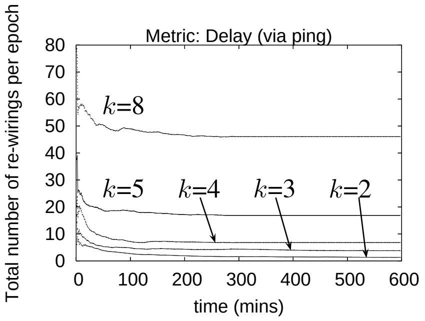


Fig. 5. Number of re-wirings in EGOIST.

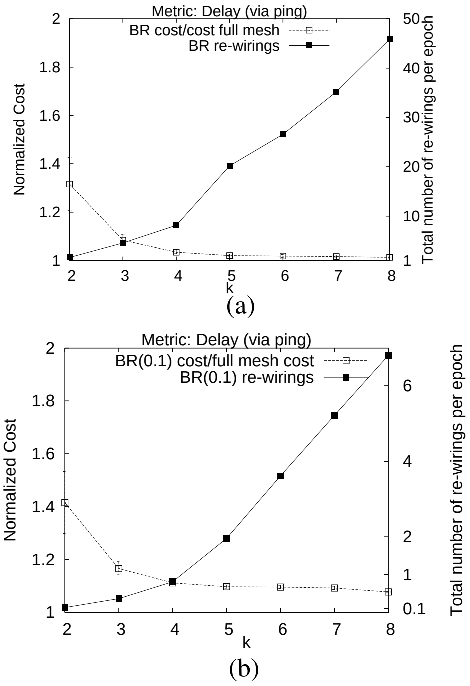


Fig. 6. Trade-off between performance and number of re-wirings in EGOIST.

A. Measurement and Re-wiring Overheads

In this section we show experimentally that EGOIST introduces a small amount of overhead for maintaining the overlay.
Link-State Protocol Load: The overhead (in terms of additional injected traffic) imposed by the link-state protocol is also low. Each node broadcasts a packet with its ID, its neighbors' IDs and the cost of the established links to its k neighbors every $T_{\mathrm{a}} < T$. The header and padding of the link-state protocol messages require a total of 192 bits, and the payload per neighbor requires 32 bits. Thus, the overhead in terms of injected traffic on the overlay is $\approx (192{+}32k)/T_{\mathrm{a}}$ bps per node. In our experiments we set T_{a}=20 secs. The above can be seen as an upper limit, as only unique link state messages forwarded in the overlay (as mentioned in Section VI-A). In our implementation, no node spent more than 1 Kbps to maintain the network.
Re-wirings Overhead: Figure 5 shows the total number of re-wirings per (one minute) epoch for the entire overlay over time. The results suggest that the re-wiring rate decreases fast as EGOIST reaches a "steady state" and that the re-wiring rate is minimal for small values of k. Here we note that as k increases the re-wiring rate increases, but the improvement (in terms of routing cost) is marginal, as a small number of outgoing links is sufficient to significantly decrease the cost. This is evident in Figure 6 (a). Finally, we also note that the re-wiring rate can significantly be decreased (with marginal impact on routing cost) by requiring that re-wiring be performed only if connecting to the "new" set of neighbors would improve the local cost to the node by more than a given threshold ϵ. We refer to this modified version of BR as BR(ϵ). Figure 6 (b) confirms this by showing the number of re-wirings and resulting performance when ϵ = 10%. We also measured the memory and CPU consumption using `time` of Unix. Our experimental results show that both the CPU and memory consumption is close to 0%, and the bandwidth consumption per node is negligible [38]. It is worth mentioning that the in-degree was quite uniform in all our experiments, thus no node allocated significantly more CPU power, memory, or bandwidth than any other in the overlay.

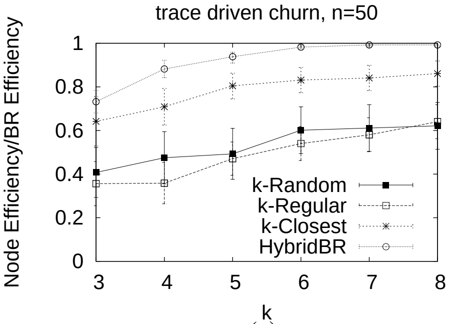
B. Effect of Churn

In the original SNS formulation,the graphs resulting from the SNS-game as well as from the empirical wiring strategies were guaranteed to be connected, so they could be compared in terms of average or maximum distance. Node churn, however, can lead to disconnected graphs, therefore we have to use a different metric. For that purpose, we choose the *efficiency* metric [49], where the efficiency ϵ_{ij} between node i and j ($j \neq i$) is inversely proportional to the shortest communication distance d_{ij} when i and j are connected. The efficiency ϵ_i of a node i defined as: $\epsilon_i = 1/(n-1)\sum_{j\neq i}\epsilon_{ij}$. The less is the cost to reach a node in the network, the higher is the value of node efficiency. If there is no path in the graph between node i and j then $\epsilon_{ij} = 0$, thus a disconnected graph yields reduction node efficiency.

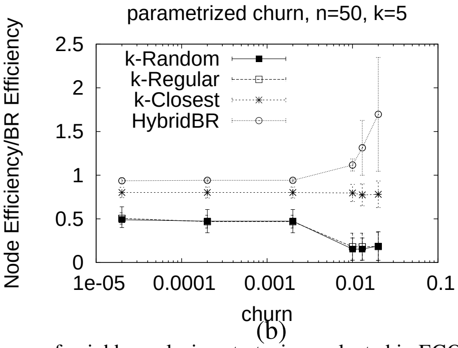
To evaluate the efficiency of nodes in EGOIST overlays under churn, we allow each of the n=50 nodes in the overlays to exhibit ON and OFF periods. During its ON periods, a node "joins" the overlay, performs re-wiring according to the chosen strategy, and fully participates in the link-state routing protocol. During its OFF periods, a node simply drops out from any activity related to the overlay. The ON/OFF periods we use in our experiments are derived from real data sets of the churn observed for PlanetLab nodes [43], with adjustments to the timescale to control the intensity of churn. In addition to evaluating the efficiency of various neighbor selection strategies we have considered so far, we also evaluate the efficiency of HybridBR, which allows a node to donate k_2=2 of its links to ensure connectivity (*i.e.*, boost the overlay efficiency) while using BR for the remaining links.

Figure 7 (a) shows the achievable efficiency of the various neighbor selection strategies when churn is present. As before, the efficiency of the various strategies is normalized with respect to that achievable using BR, and is shown as a function of k. As with all the metrics we considered so far, BR outperforms all other strategies (including HybridBR), but as EGOIST nodes are allowed to have more neighbors (*i.e.*, as k increases), the efficiency of the HybridBR approaches that of BR, with the efficiency of k-Closest decisively better than k-Random and k-Regular.

The above results imply that under the level of churn in



(a)



(b)

Fig. 7. Efficiency of neighbor selecion strategies evaluated in EGOIST under churn.

these experiments, it is not justifiable for BR to donate two of its links simply to ensure connectivity, especially when k is small. Notice that BR overlays that get disconnected due to churn will naturally heal as soon as any of its active nodes decides to re-wire. This is so because the (infinite) cost of reaching the disconnected nodes will act as an incentive for nodes to choose disconnected nodes as direct neighbors, thus reconnecting the overlay. As noted earlier, re-wiring occurs every T/n units of time on average, which implies that the vulnerability of BR to disconnections due to churn is highest for smaller overlays and if re-wiring is done infrequently. Our results also showed that adding or removing a node from the overlay does not increase the number of re-wires in the system. Under moderate churn, and random selection of a node to add or delete, the number of re-wirings in the system are similar to those reported in Section VII-A.

Our last question then is whether at much higher churn rates, it is the case that the use of HybridBR would be justified. To answer this question, we changed the timescale of the ON/OFF churn processes to emulate more frequent joins and leaves. Figure 7 (b) shows the results by plotting the efficiency metric for the various strategies as a function of the churn rate (on the x-axis), which we define (as in [43]) to be the sum of the fraction of the overlay network nodes that changed state (ON/OFF), normalized by time T, *i.e.*, Churn = $1/T \sum_{events\ i} \frac{|U_{i-1} \ominus U_i|}{max\{|U_{i-1}|, |U_i|\}}$, where U_i is the new set of nodes in the overlay following an event i that alters the membership in the set of nodes that participate in the overlay, and $\ominus$ is the symmetric set difference. Thus, a churn rate of 0.01 implies that, on average, 1% of the nodes join or leave the overlay per second. For an overlay of size n=50, this translates to a join or leave event every two seconds.

As expected, when churn rate increases significantly to the point where the average time between churn events approaches T/n, the efficiency of HybridBR eventually surpasses that of BR. The results also suggest that under such conditions, the efficiency of both k-Random and k-Regular fall dramatically, whereas that of k-Closest remains level with that of BR.

VIII. Applications and Artifacts

EGOIST is a general purpose overlay routing system that can be used by applications to supplement traditional IP routing. To demonstrate its potential value we consider the case of real-time applications with hard end-to-end requirements and the case of online multi-player P2P games.

Real-time Applications: In many real-time application, *e.g.*, voice conference, a selfish node would strive to minimize the maximum delay to all the other nodes in the overlay. Given a residual wiring the best response of a node, henceforth called min-max BR, is such that the maximum delay to any other node in the network is minimized. Figure 8 (a) shows the performance of various neighbor selection strategies in the 50-node EGOIST overlay when normalized with respect to that achieved by min-max BR. The maximum delay to an individual node is anywhere between 50% and 350% (in low link density overlays) higher than that achieved by min-max BR.

In other real-time applications, *e.g.*, transactions that require consistency among distributed databases, hard quality-of-service requirements must be satisfied. The estimation of a minimum out-degree that is needed to satisfy the application requirements is hard to be estimated a-priori by the system designer especially in a highly dynamic environment. A selfish node would strive to satisfy the application requirement while keeping its out-degree as small as possible. The best response of a node, henceforth called variable-degree BR, can be materialized by a local search heuristic where each node can swap or incrementally add or drop out-going wirings. We consider a real-time application where each node has to communicate with any other node in less than 125 msecs. We run the application over the 50-node EGOIST overlay where initially each node selects uniformly at random five other nodes as neighbors and we set T_{a}=5 seconds. Figure 8 (b) shows the maximum delay in the overlay over time. Within 35 seconds the application requirement is satisfied. It is worth mentioning that out of the 50 nodes 35 nodes have out-degree 2, 12 nodes have out-degree 3, and three out-degree 5. This counts up to only 121 links compared to 250 links established initially. Multiple establishments or drops of wirings may lead to faster convergence time but it might be unfair for some of the nodes as we show in a larger set of experiments in [38].

Multi-player P2P Games: We obtained from [13] a trace containing the movements of 100 players (AI bots) participating in a game of Quake III. In Quake III, players are located in a virtual 3D world and interact frequently as they come into contact to fight each other by by sending event updates (packets). We distributed the 100 players among our 25 EGOIST nodes on PlanetLab and used the EGOIST overlay to deliver the updates. We set k=2 and mapped the L_3 distance of players i and j in the virtual world into the preference weight p_{ij} that defines the preference that the local EGOIST node of i has for sending messages to the local EGOIST node

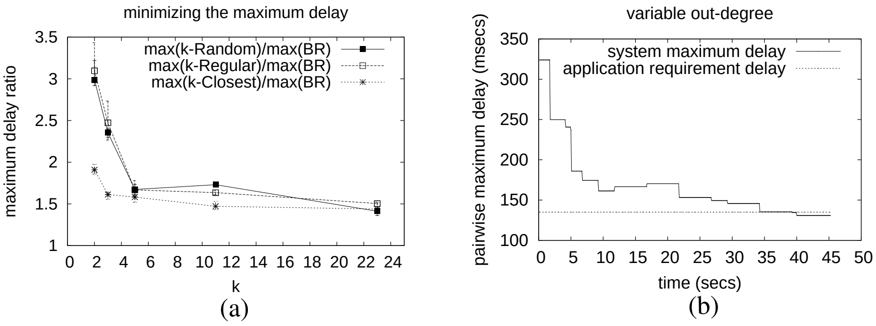


Fig. 8. Achieving the minimum pairwise delay with constant and with variable out-degree in EGOIST.

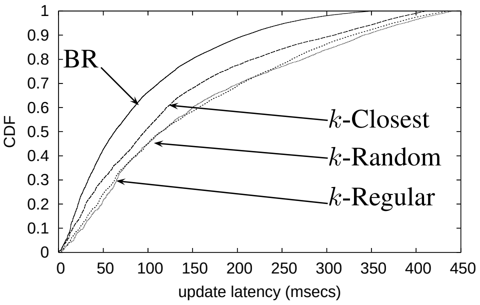


Fig. 9. Comparison of update latencies for various neighbor selection strategies.

of j. We replayed the trace, that involved more than $108,000$ events, and compared the update latencies when sent over EGOIST and over heuristic wiring strategies. The cumulative distribution function of update latencies is illustrated in Fig. 9. Both the median (∼65 msecs) and the 95^{th}-percentile update latency over EGOIST is less than half of the corresponding latencies over k-Random and k-Regular, and less than two-thirds of those over k-Closest. Experimentally, it has been shown that update latency higher than 200 msecs may effect the quality of user's experience [13]. More than 90% of packets sent over EGOIST were delivered earlier than 200 msecs and only 60-70% under the other topologies.

Artifacts: Variations of EGOIST may significantly improve the performance of popular applications, including multi-path routing and content search. In multi-path bulk transfers a selfish node would strive to maximize the set of disjoints paths to the destination that maximize its up-link capacity. In multi-path routing for time-sensitive applications, *e.g.*, Voice-over-IP, a selfish node would strive to maximize the set of disjoints paths that minimize the loss rate. In the content search context, *e.g.*, scoped flooding, a selfish node would strive to maximize the number of similar-profile nodes to query. All the abovementioned best responses are easily implementable in EGOIST. We implemented them and evaluated their performance against heuristic wirings. In all the cases the performance of best response was way higher, especially for large values of out-degree. In [38] we provide a detailed presentation of all our experimental results.

Our EGOIST prototype is currently deployed on PlanetLab. A live demonstration of the overlay routing topology maintained by EGOIST and the source code can be accessed from the project web site at `http://csr.bu.edu/sns/`.

IX. Related Work

Selfish neighbor selection for overlay networks was first mentioned by Feigenbaum and Shenker [18]. Fabrikant et al. [20] studied an unconstrained undirected version of the game in which nodes can buy as many links as they want at a fixed per link price α. Chun et al. [23] studied experimental an extended version of the problem in which links prices need not be the same. The work by Rocha et al. [24] was in the same spirit. Corbo and Parkes [21] studied bilateral network formation games. Demaine et al. [22] proved tighter bounds on the price of anarchy [50] in the aforementioned games. In practice, however, important constraints on node degrees, not captured by these models, lead to richer games with substantively and fundamentally different outcomes. Only recently Laoutaris et. al [39] studied the fractional bounded budget connection games and Kintali et al. [51] studied the complexity of Nash equilibria in such games.

Bindal et al. [28] proposed a locality-enhanced version of BitTorrent in which only m out of the total k neighbors of a BitTorrent node are allowed to belong to a different ISP. Although the capacitated selection of neighbors is a central aspect of this work, their treatment is fundamentally different from ours in several regards: (i) there's no contention between selfish peers, (ii) the minimization objective is on inter-AS traffic therefore only two levels of communication distance are modeled, intra and inter-AS (we use finer topological information that includes exact inter-peer distances), and (iii) their "reachability" constraint amounts to asking for a similar level of data availability as the original one under the standard random neighbor selection mechanism of BitTorrent (we have fundamentally different reachability constraints, expressed as general preference functions over the potential overlay neighbors). Smaragdakis et al. [52] proposed neighbor selection strategies to create optimized graphs for n-way broadcast applications. Another recent work on neighbor selection is from Godfrey et al. [43]. It aimed at selecting neighbors in a way that minimizes the effects of node churn (appearance of new nodes, graceful leaves and sudden malfunctions), but unlike our work, it did not focus on the impact of competing selfish nodes. Aggarwal et al. [53] evaluated ISP-assisted neighbor selection strategies in P2P systems. The effect of selfishly constructed overlays to traffic engineering in the native layer was studied in [54].

X. Conclusion

Our work has started with a study of selfish neighbor selection under strictly enforced neighbor budgets and has come up with a series of findings with substantial practical value for real overlay networks. First, we have shown that a best response (*i.e.*, selfish) selection of neighbors leads to the construction of overlays with much better performance than those constructed by simple random and myopic heuristics. The reason is that by being selfish, nodes embark on a distributed optimization of the overlay that turns out to be beneficial for all. Secondly, we have demonstrated through the design, implementation, and deployment of EGOIST, that it is indeed feasible to apply our best response wiring in practice and that the obtained benefits are actually much larger under dynamic environments where the simple heuristics lag even more. Finally, we have used our EGOIST prototype for achieving real-time requirements and carrying the traffic generated by an online multi-player P2P game and have verified all our above observations.

REFERENCES

[1] N. Laoutaris, G. Smaragdakis, A. Bestavros, and J. W. Byers, "Implications of Selfish Neighbor Selection in Overlay Networks," in *Proc. IEEE INFOCOM'07*.

[2] G. Smaragdakis, V. Lekakis, N. Laoutaris, A. Bestavros, J. W. Byers, and M. Roussopoulos, "EGOIST: Overlay Routing using Selfish Neighbor Selection," in *Proc. ACM CoNEXT'08*.

[3] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: Informed Internet routing and transport," *IEEE Micro*, vol. 19, no. 1, pp. 50–59, 1999.

[4] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient Overlay Networks," in *Proc. ACM SOSP'01*.

[5] O. Ercetin and L. Tassiulas, "Market-based resource allocation for content delivery in the internet," *IEEE Trans. on Computers*, vol. 52, no. 12, pp. 1573–1585, 2003.

[6] L. Wang, K. Park, R. Pang, V. Pai, and L. Peterson, "Reliability and Security in the CoDeeN Content Distribution Network," in *Proc. USENIX'04*.

[7] B. Cohen, "Incentives Build Robustness in BitTorrent," in *Proc. of the 1st Workshop on Economics of Peer-to-Peer Systems*, 2003.

[8] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, , M. F. Kaashoek, F. Dabek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications," *IEEE/ACM Trans. on Networking*, vol. 11, no. 1, pp. 17–32, 2003.

[9] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "SplitStream: High-bandwidth Multicast in Cooperative Environments," in *Proc. ACM SOSP'03*.

[10] D. Kostic, A. Rodriguez, J. Albrecht, and A. Vahdat, "Bullet: High Bandwidth Data Dissemination using an Overlay Mesh," in *Proc. ACM SOSP'03*.

[11] N. Magharei and A. H. Rasti, "Prime: Peer-to-peer Receiver-driven Mesh-based Streaming," in *Proc. IEEE INFOCOM'07*.

[12] N. Laoutaris, P. Rodriguez, and L. Massoulie, "ECHOS: Edge Capacity Hosting Overlays of Nano Data Centers," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 1, pp. 51–54, 2008.

[13] A. Bharambe, J. R. Douceur, J. R. Lorch, T. Moscibroda, J. Pang, S. Seshan, and X. Zhuang, "Donnybrook: Enabling Large-Scale, High-Speed, Peer-to-Peer Games," in *Proc. ACM SIGCOMM'08*.

[14] L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker, "On Selfish Routing in Internet-like Environments," in *Proc. ACM SIGCOMM'03*.

[15] M. Feldman, K. Lai, I. Stoica, and J. Chuang, "Robust Incentive Techniques for Peer-to-peer Networks," in *Proc. ACM EC'04*.

[16] T. Roughgarden and Éva Tardos, "How Bad is Selfish Routing?" *Journal of the ACM*, vol. 49, no. 2, pp. 236–259, 2002.

[17] N. Nisan, T. Roughgarden, Éva Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press, 2007.

[18] J. Feigenbaum and S. Shenker, "Distributed Algorithmic Mechanism Design: Recent Results and Future Directions," in *Proc. of the 6th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, 2002.

[19] J. Ledlie, P. Pietzuch, and M. Seltzer, "Network Coordinates in the Wild," in *Proc. USENIX/ACM NSDI'07*.

[20] A. Fabrikant, A. Luthra, E. Maneva, C. H. Papadimitriou, and S. Shenker, "On a Network Creation Game," in *Proc. ACM PODC'03*.

[21] J. Corbo and D. C. Parkes, "The Price of Selfish Behavior in Bilateral Network Formation," in *Proc. ACM PODC'05*.

[22] E. D. Demaine, M. Hajiaghayi, H. Mahini, and M. Zadimoghaddam, "The Price of Anarchy in Network Creation Games," in *Proc. ACM PODC'07*.

[23] B.-G. Chun, R. Fonseca, I. Stoica, and J. Kubiatowicz, "Characterizing Selfishly Constructed Overlay Routing Networks," in *Proc. IEEE INFOCOM'04*.

[24] B. G. Rocha, V. Almeida, and D. Guedes, "Improving Reliability of Selfish Overlay Networks," in *Proc. WWW'06*.

[25] A. Archer, "Inapproximability of the Asymmetric Facility Location and k-median Problems," 2000, unpublished manuscript, available from the author's web-page.

[26] X. Yang and G. de Veciana, "Performance of Peer-to-peer Networks: Service Capacity and Role of Resource Sharing Policies," *Perform. Eval.*, vol. 63, no. 3, pp. 175–194, 2006.

[27] Z. Yao, D. Leonard, X. Wang, and D. Loguinov, "Modeling Heterogeneous User Churn and Local Resilience of Unstructured P2P Networks," in *Proc. IEEE ICNP'06*.

[28] R. Bindal, P. Cao, W. Chan, J. Medval, G. Suwala, T. Bates, and A. Zhang, "Improving Traffic Locality in BitTorrent via Biased Neighbor Selection," in *Proc. IEEE ICDCS'06*.

[29] Y. Liu, H. Zhang, W. Gong, and D. F. Towsley, "On the Interaction between Overlay Routing and Underlay Routing," in *Proc. IEEE INFOCOM'05*.

[30] Y. Tian, D. Wu, and K.-W. Ng, "Analyzing Multiple File Downloading in BitTorrent," in *Proc. ICPP'06*.

[31] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. MIT Press, 1994.

[32] P. B. Mirchandani and R. L. Francis, *Discrete Location Theory*. John Wiley and Sons, 1990.

[33] M. Charikar, S. Guha, Éva Tardos, and D. B. Shmoys, "A Constant-factor Approximation Algorithm for the k-median Problem," *Journal of Computer and System Sciences*, vol. 65, no. 1, pp. 129–149, 2002.

[34] M. R. Korupolu, C. G. Plaxton, and R. Rajaraman, "Analysis of a Local Search Heuristic for Facility Location Problems," in *Proc. ACM-SIAM SODA'98*.

[35] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit, "Local Search Heuristics for k-Median and Facility Location Problems," *SIAM J. on Computing*, vol. 33, no. 3, pp. 544–562, 2004.

[36] K. Jain and V. V. Vazirani, "Primal-Dual Approximation Algorithms for Metric Facility Location and k-Median Problems," in *Proc. IEEE FOCS'99*.

[37] J.-H. Lin and J. S. Vitter, "ϵ-Approximations with Minimum Packing Constraint Violation," in *Proc. ACM STOC'92*.

[38] G. Smaragdakis, "Overlay Network Creation and Maintenance with Selfish Users," *Ph.D. Dissertation, Boston University*, 2009.

[39] N. Laoutaris, L. Poplawski, R. Rajaraman, R. Sundaram, and S.-H. Teng, "A Bounded-Degree Network Formation Game," in *Proc. ACM PODC'08*.

[40] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An Approach to Universal Topology Generation," in *Proc. IEEE/ACM MASCOTS'01*.

[41] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[42] A. Ziviani, S. Fdida, J. F. de Rezende, and O. C. M. B. Duarte, "Toward a Measurement-based Geographic Location Service," in *Proc. PAM'04*.

[43] P. B. Godfrey, S. Shenker, and I. Stoica, "Minimizing Churn in Distributed Systems," in *Proc. ACM SIGCOMM'06*.

[44] S. Rhea, D. Geels, T. Roscoe, and J. Kubiatowicz, "Handling Churn in a DHT," in *Proc. USENIX'04*.

[45] A. Young, J. Chen, Z. Ma, A. Krishnamurthy, L. L. Peterson, and R. Wang, "Overlay Mesh Construction Using Interleaved Spanning Trees," in *Proc. IEEE INFOCOM'04*.

[46] A. Shriram, M. Murray, Y. Hyun, N. Brownlee, A. Broido, M. Fomenkov, and K. C. Claffy, "Comparison of Public End-to-End Bandwidth Estimation Tools on High-Speed Links," in *Proc. PAM'05*.

[47] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cottrell, "pathChirp: Efficient Available Bandwidth Estimation for Network Paths," in *Proc. PAM'03*.

[48] N. Deo, *Graph Theory with Applications to Engineering and Computer Science*. Prentice Hall, 1994.

[49] V. Latora and M. Marchiori, "Economic Small-World Behavior in Weighted Networks," *The European Physical J. B*, vol. 32, pp. 249–263, 2003.

[50] E. Koutsoupias and C. Papadimitriou, "Worst-case Equilibria," in *Proc. STACS'99*.

[51] S. Kintali, L. J. Poplawski, R. Rajaraman, R. Sundaram, and S.-H. Teng, "Reducibility Among Fractional Stability Problems," in *Proc. IEEE FOCS'09*.

[52] G. Smaragdakis, N. Laoutaris, P. Michiardi, A. Bestavros, J. W. Byers, and M. Roussopoulos, "Swarming on Optimized Graphs for n-way Broadcast," in *Proc. IEEE INFOCOM'08*.

[53] V. Aggarwal, A. Feldmann, and C. Scheideler, "Can ISPs and P2P users cooperate for improved performance?" *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 3, pp. 29–40, 2007.

[54] Srinivasan Seetharaman, Voler Hilt, Markus Hofmann, and Mostafa Ammar, "Preemptive Strategies to Improve Routing Performance of Native and Overlay Layers," in *Proc. IEEE INFOCOM'07*.

Georgios Smaragdakis received the Diploma in electronic and computer engineering from the Technical University of Crete, Greece, the Ph.D. degree in computer science from Boston University, MA, and he interned at Telefónica Research, Barcelona, Spain. He is a Senior Research Scientist at Deutsche Telekom Laboratories and the Technical University of Berlin, Germany. His research interests include the measurement, performance evaluation and optimization of content distribution systems and overlay networks with main applications in overlay network creation and maintenance, service deployment, server selection, network storage management, distributed caching, and ISP-Applications collaboration.

Nikolaos Laoutaris is a senior researcher at the Internet research group of Telefonica Research in Barcelona. Prior to joining the Barcelona lab he was a postdoc fellow at Harvard University and a Marie Curie postdoc fellow at Boston University. He got his PhD in computer science from the University of Athens in 2004. His general research interests are on system, algorithmic, and performance evaluation aspects of computer networks and distributed systems. Current projects include: Efficient inter-datacenter bulk transfers, energy-efficient distributed system design, content distribution for long tail content, transparent scaling of social networks, pricing of broadband services and ISP interconnection economics.

Vassilis Lekakis received the M.Sc. degree in computer science from the University of Crete, Crete, Greece, in 2007, and is currently pursuing the Ph.D. degree in computer science at the University of Maryland, College Park.

Azer Bestavros received the Ph.D. degree in computer science from Harvard University in 1992. He is Founding Director of the Hariri Institute for Computing and a Professor in the Computer Science Department at Boston University, which he joined in 1991 and chaired from 2000 to 2007. Prof. Bestavros' research interests are in the broad areas of networking and real-time embedded systems. His contributions include his pioneering the distribution model adopted years later by CDNs, his seminal work on Internet and web characterization, and his work on compositional certification of networked systems and software. He is the chair of the IEEE Computer Society TC on the Internet, served on the program committees and editorial boards of major conferences and journals in networking and real-time systems, and received distinguished service awards from both the ACM and the IEEE. He received the United Methodist Scholar Teacher Award at B.U. and the 2010 ACM SIGMETRICS Inaugural Test of Time Award (with M. Crovella).

John W. Byers is an Associate Professor of Computer Science at Boston University. He is also Chief Scientist and a member of the Board of Directors at Adverplex, Inc., a quantitative online advertising firm based in Cambridge, MA. Dr. Byers joined B.U. in 1999 and has had an executive role at Adverplex since the company's founding in 2005. Prior to his appointment at B.U., he completed his Ph.D. in Computer Science at the University of California at Berkeley in 1997. Professor Byers' research interests include algorithmic and economic aspects of networking, large-scale data management, and e-commerce. He received the ACM SIGCOMM Test of Time Award in 2009 for his work on erasure-encoded content delivery, the IEEE ICDE Best Paper Award in 2004 for his work on sensor databases, and a National Science Foundation CAREER Award in 2001. He served terms on the editorial board of IEEE/ACM Transactions on Networking and on the executive committee of ACM SIGCOMM. He serves as the co-chair of the SIGCOMM 2011 technical program committee.

Mema Roussopoulos received the B.S. degree in computer science from the University of Maryland, College Park, and the Ph.D. degree in computer science from Stanford University, Stanford, CA. She is an Assistant Professor of computer science with the Department of Informatics and Telecommunications, National Kapodistrian University of Athens, Athens, Greece. She was an Assistant Professor of computer science on the Gordon McKay Endowment with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA. She was then a faculty member with the Computer Science Department, University of Crete, Crete, Greece, and an Associated Researcher with the Institute of Computer Science at FORTH. Her research interests are in the areas of distributed systems, networking, mobile computing, and digital preservation. Dr. Roussopoulos is a recipient of the CAREER Award from the National Science Foundation and the Best Paper Award at the ACM SOSP 2003.