Can an overlay compensate for a careless underlay?

Honggang Zhang, Jim Kurose, Don Towsley
Dept. of Computer Science
University of Massachusetts, Amherst, MA 01003
Email: {honggang, kurose, towsley}@cs.umass.edu

Technical Report 2005-49
Revised on July 15, 2005

***Abstract*— We investigate the ability of an overlay network to compensate for "careless" routing in the native network layer, i.e., for network-layer routes not optimized for a given cost function or traffic matrix. We consider cost/traffic-dependent, overlay routing on top of an underlay routing based on randomly-generated, cost/traffic-independent link weights, and determine the extent to which overlay-over-careless-underlay can achieve performance close to that attainable when underlay routing is performed in an optimal cost/traffic-dependent ("careful") manner. We identify three graph-theoretic metrics that collectively characterize the *richness* of the underlay network: the tightness (characteristic path length, or CPL), the thickness (cut average) and the weighted sum of node degrees. We find that only when the underlay graph is rich, the overlay can compensate for careless underlay. If network links have linear costs, we prove that overlay cost is a linear increasing function of CPL under some assumptions. Finally, we investigate heuristics (based on the notion of richness) for choosing a set of overlay nodes of fixed size that is likely to result in good overlay performance.**

I. INTRODUCTION

While much of the past and current research in overlay networking has focused on techniques for building overlay networks and evaluating their performance(e.g., [2], [13]), several recent efforts have begun to investigate the interaction between the underlay network and the overlay network, including the stability of the interaction between changes in underlay and overlay routing [11], [15], [16] and the influence of the underlay topology [8], [10] on overlay performance.

In this paper, we too consider the interaction between underlay and overlay networks, and consider the ability of an overlay network to compensate for "careless" routing in the native network layer, i.e., for network-layer routes that have not been optimized for a given cost function or input traffic matrix.

Our work is based on the conjecture that when applications are able to adapt their routing at the application layer, it is less important that the network layer optimizes its routing, since this additional level of control and flexibility allows applications to compensate for routing inefficiencies at the network layer. Indeed, we find that this conjecture is often (but not always) true, and identify characteristics of the underlay topology and underlay routing that determine when this conjecture holds true.

A careless underlay[1] might arise in several situations. For example, the underlay might set the physical routes based on some internal policies [16]. Or the underlay tries to optimize its routes but is unable to achieve it due to deficiencies of its routing algorithm [6]. Or there might be multiple overlays with different traffic demands such that it is difficult for the underlay to optimize physical routes for all of them simultaneously. Or an inter-AS routing protocol, e.g. BGP, simply follows some policy which completely ignores the routing performance for any overlay network. Another situation might be that other considerations such as reliability restrict the underlay's capability of making optimal routes.

[1] We will use the terms *underlay* and *overlay* to refer to the underlay routing controller (or algorithm or protocol) and the overlay routing controller, respectively.

The central question we address is *whether an application level overlay routing structure can compensate for a careless underlay.* We decompose the central question into two sub-problems: one for the underlay and the other one for the overlay. First, since the overlay is built on top of the underlay network, a logical route on the overlay corresponds to some physical routes set by the underlay. Thus, we cannot expect that the overlay can achieve near optimal routes on any arbitrary set of underlay routes. Thus, the first sub-problem is: what basic requirements should the physical routes satisfy in order for the overlay to have reasonably good performance. Second, we assume that the overlay is restricted to choosing only a subset of nodes as relay nodes, then our second sub-problem is: how to construct an optimal overlay topology on top of a set of underlay routes. This overlay topology is optimal in the sense that the overlay routing algorithm on this topology will have the lowest cost among all topologies.

Unlike [8], we do not restrict ourselves to improving path diversity in order to achieve a better failure recovery ratio. Instead, we are interested in improving the routing performance for given traffic matrices. Our work also differs from [10] in that our goal is not to evaluate specific common overlay topologies and routing protocols. Instead, we are interested in investigating fundamental topological properties that significantly influence the routing cost.

We have the following contributions and results.

First. We use randomly generated sub-graphs of the original physical network to mimic the carelessness of the underlay. Specifically, these sub-graphs are obtained by randomly assigning link weights and running shortest path (least weight) routing algorithm. We show that an overlay can always improve on a careless underlay, but it is not necessarily true that the overlay can achieve near optimal routing performance on top of any set of underlay routes. Optimal routing performance is achieved when the total network resource is available to the overlay, which happens when the overlay includes all physical nodes and the underlay exposes all physical links to the overlay.

Second. To tackle the first sub-problem, we let the overlay include all the physical nodes and vary the sub-graphs to explore the impact on overlay's routing from the topological properties of these sub-graphs. We identify three key graph-theoretic metrics which have the most significant influence on routing performance. These metrics can be used to measure the richness of graph in terms of routing performance. Specifically, we find that the characteristic path length (CPL) of a graph is a reasonable measure of the tightness of a graph, and the average cut-set size can measure the thickness of a graph. More precisely, the CPL of a graph is defined as the average of the means of the shortest path lengths connecting each vertex to all other vertices [5]. The average cut-set size is the average size of all cut-set sizes. Weighted node degree sum (WNDS) is a weighted sum of all nodes' degrees of a graph. We give a larger weight to a node with smaller degree since the smaller degree node is more likely to have its links to be highly utilized.

We study two link cost functions. One is a linear cost function, for which we prove that there is a linear functional relationship between overlay routing cost and the CPL (tightness) when the traffic matrix is homogeneous (all nodes have equal amount of traffic demands to all other nodes) and link capacities are homogeneous. We also show that this linear function asymptotically holds true with probability one even if the traffic matrix has variation. The other cost function is a piece-wise linear convex increasing function (an approximation of M/M/1 queuing delay function[6].) For this function, we find experimentally that the thickness of the graphs should also be considered. We find that when the load level is either light or heavy, the CPL is strongly correlated with the routing performance, and there exists an almost perfect monotonic decreasing function relationship between routing performance and the CPL. This is also true for average cut-set size, except that it is an increasing function. However, when the load level is moderate, weighted node degree sum must be considered in order to predict routing performance. Thus, in order to provide richness for routing, we recommend that the underlay should generate sub-graphs having low CPL and high average cut size and low weighted node degree sum. We investigate both homogeneous and bimodal traffic matrix [12], and ten different load levels from light link utilization to utilization above one. Our results hold in a large range of traffic matrix variation that is introduced by varying the coefficient of variation (ratio of the standard deviation to the mean) of the bimodal traffic matrix.

Third. Based on the findings we recommend a heuristic to construct an overlay topology with good richness when this overlay cannot have all nodes included. Experiments on some inferred real ISP topologies demonstrate the effectiveness of our heuristic.

In summary, our answer to the central question is affirmative given that the careless underlay provides a rich sub-graph and the overlay constructs a rich topology.

The rest of the paper is organized as follows. Routing models and analysis are presented in Section II. In Section III we present the experimental study on identifying the key graph-theoretic metrics. In section IV, a overlay constructing heuristic is given and its effectiveness is demonstrated through experiments on some inferred ISP topologies. Related work is given in Section V. Finally, we conclude the paper in Section VI.

II. Models and Analysis

In this section, we first present the overlay and underlay routing models studied in this paper. Then, we introduce several graph-theoretic metrics related to overlay routing performance. Following that, we then give some analytic results on relating routing performance to graph-theoretic metrics.

A. Network routing models

We model the physical network as a connected undirected graph $G = (V, E)$ with node set V and edge or link set E. The overlay network is denoted by another connected undirected graph $G_o = (V_o, E_o)$. We have $V_o \subseteq V$ but logical link set E_o may not be a subset of E. In fact, logical link $(s, t) \in E_o$ may correspond to one or multiple physical paths (sequences of physical links) in G. This is determined by the underlay routing controller. Take a physical link e, we use $f_e^{(s,t)}$ to denote the fraction of traffic rate on logical link (s, t) allocated on physical link e. In [11], $f_e^{(s,t)}$ is a real number in $[0, 1]$ because the traffic is treated as splittable fluid. However, in this paper, $f_e^{(s,t)}$ only evaluates to either 0 or 1, [2] namely, the traffic is not splittable.

Suppose that there is a traffic matrix TM consisting of traffic demands for multiple source and destination (SD) node pairs. Let $d^{(s,t)}$ denote the traffic demand between SD pair (s, t) in TM. $d^{(s,t)}$ includes the demand of the overlay $r^{(o,d)}$ (if overlay node pair (o, d) is the same as (s, t)) and all other non-overlay demand $d^{(s,t)non-over}$. That is,

$$d^{(s,t)} = d^{(s,t)non-over} + r^{(o,d)}; \quad if \quad (o, d) = (s, t)$$

The objective of an overlay routing algorithm is to allocate $r^{(o,d)}$ on logical links so as to minimize cost. The resulting allocation $g_{(s,t)}^{(o,d)} \cdot r^{(o,d)}$ will be interpreted as traffic demands by the underlay. Here, $g_{(s,t)}^{(o,d)}$ is the fraction of traffic demand $r^{(o,d)}$ allocated to logical link (s, t). Let $d^{(s,t)over}$ denote the traffic rate on logical link (s, t) from all overlay source destination pairs, after the overlay runs its routing algorithm. That is, we have $d^{(s,t)over} = \sum_{(o,d)} g_{(s,t)}^{(o,d)} \cdot r^{(o,d)}$. Then, the total traffic demand for each node pair of the underlay can be denoted as $d^{(s,t)under} = d^{(s,t)non-over} + d^{(s,t)over}$. Let TM^{under} denote the underlying traffic matrix, $TM^{under} = [d^{(s,t)under}]$. Note that $d^{(s,t)under}$ might not be the same as $d^{(s,t)}$. TM^{under} is the traffic matrix to the underlay after the overlay runs its routing algorithm, whereas TM is the original traffic matrix when the overlay does not run its routing algorithm. If there is no overlay demand ($r^{(o,d)} = 0, \forall (o, d) \in V_o \times V_o$), then $d^{(s,t)} = d^{(s,t)under} = d^{(s,t)non-over}$.

In this paper, we are interested in a situation where the underlay modifies its routes infrequently and the overlay routes the original traffic matrix TM on top of the physical routes set by the underlay. Furthermore, we assume that the underlay does not have as its objective to optimize its routes for any given TM, but that the overlay does have this objective subject

[2] This is valid for OSPF in intra-AS graph and BGP in inter-AS graph.

to the constraint that it must work on the logical level on top of the given physical routes set by the underlay.

Link cost functions. The flow rate on physical link e is:

$$l_e = \sum_{(s,t)} f_e^{(s,t)}[d^{(s,t)non-over} + d^{(s,t)over}] \quad (1)$$

This flow will incur some delay or cost on physical link e. We consider two link cost functions. One is a linear cost function, defined as

$$\Phi_e(l_e) = a \cdot l_e/C_e \quad (2)$$

where $a > 0$ is common to all links.

The other is a piece-wise linear approximation of M/M/1 delay function [6], [11], [15], defined as

$$\Phi_e(l_e) = \begin{cases} l_e & , l_e/C_e \in [0, 1/3) \\ 3l_e - 2/3C_e & , l_e/C_e \in [1/3, 2/3) \\ 10l_e - 16/3C_e & , l_e/C_e \in [2/3, 9/10) \\ 70l_e - 178/3C_e & , l_e/C_e \in [9/10, 1) \\ 500l_e - 1468/3C_e & , l_e/C_e \in [1, 4/3) \\ 5000l_e - 19468/3C_e & , l_e/C_e \in [4/3, \infty) \end{cases} \quad (3)$$

Underlay Routing Optimizer.

Its objective is to minimize the overall cost $\sum_{e \in E} \Phi_e$. The decision variable is $f_e^{(s,t)}$ (recall (1)).

Careless Underlay Routing Controller (Careless Underlay).

A careless underlay routing controller (or simply called a careless underlay) is a routing algorithm that does not have the objective to optimize its routes to minimize the cost of routing a given traffic matrix. As mentioned in Section I, this might happen in various situations. For example, an intra-AS routing algorithm might have to protect some links for some policy reasons [16], or its routing algorithms (e.g. OSPF) cannot achieve an optimal routing due to the fundamental deficiency [6]. Another case is that an inter-AS routing protocol, e.g. BGP, simply follows some routing policy which completely ignores the routing cost for any traffic matrix. [3]

In this paper, we use randomly generated sub-graphs of the original physical network to mimic this carelessness. Specifically, these sub-graphs are obtained by randomly assigning link weights and running shortest path routing algorithm (in this sense, it is an OSPF-like shortest path routing [1].) This careless underlay does *not* assign weights with the aim to minimize the routing cost of a given traffic matrix. We assume that it assigns link weights *uniformly at random*. We use W to denote a weight assignment profile. We hope this will be a reasonable approximation of the OSPF routing in a single ISP network where link weights are assigned based on some policy constraint or some other reasons. For example, [16] mentioned that an ISP might assign very large weights to some links to protect them. Note, these link weights are *not* link capacities. They might be related to link capacities. For example, Cisco [3] recommends to set link weights to be the inverses of link capacities. And a physical path obtained by the shortest path routing of the careless underlay is the least "weight" path, *not* the least cost path in terms of the link cost functions defined before.

[3]Note, a careless underlay is not limited to an intra-AS routing underlay.

Thus, the careless underlay routing cost $\Phi_u(G, M, W)$ is a function of the traffic matrix M and a link weight assignment profile W for a physical network G.

Overlay Routing Optimizer.

Part of the traffic rate (given in (1)) from the overlay network on a physical link is:

$$l_e^{over} = \sum_{(s,t)} f_e^{(s,t)} \sum_{(o,d)} g_{(s,t)}^{(o,d)} r^{(o,d)} \quad (4)$$

The goal of an overlay routing optimizer is to minimize the overall cost incurred to the overlay, shown as $\Phi_o = \sum_e \Phi_e^{over}(l_e^{over})$. The decision variable is $g_{(s,t)}^{(o,d)}$.

The careless underlay generates a set of routes such that each physical node has a shortest path tree to all other nodes in the physical graph G. If the overlay has a node set V_o (with $V_o \subseteq V$), then the overlay routing cost Φ_o is computed by the overlay routing optimizer described before. Each virtual link of the overlay is mapped to a physical path according to the shortest path trees. Note, if $V_o = V$, then all these physical routes form a sub-graph of the original network, which can also be thought of as the packing of all these shortest path trees. Thus, the overlay routing cost Φ_o is a function of G, M, W, V_o.

B. Overlay compensation is not always effective

In this section, we consider the case where the overlay includes all nodes in the network. This is the best case for the overlay. We refer to this as a *full-sized* overlay. We show through a simple example that, even in this case, there are some sets of underlay routes on which the overlay cannot provide good (near optimal) routing performance. Furthermore, this example suggests to us that the topological properties of the sub-graphs formed by the underlay routes can have large impact on the overlay routing performance.

First, we know that any physical routes set by the careless underlay can potentially result in some links being missing in the reduced sub-graph. This is first observed in [15] in studying selfish routing. Thus, this means that more constraints are added into the optimization problem to be solved by the overlay. Adding constraint reduces solution space and then increases the cost. [4] This implies that the best the underlay can do is to do "nothing", i.e., expose all links to the overlay. This is possible only when all physical links are assigned equal weights, then the underlay routing becomes a shortest hop-count routing. If in the mean time, we have $V_o = V$, then the overlay routing optimizer can attain the lowest possible cost Φ^*. We refer to the overlay in this case as the *optimal overlay*. In this case, the overlay routing optimizer is equivalent to the underlay routing optimizer. However, the careless underlay does not expose all links to the overlay due to various "careless" reasons. Those sub-graphs formed by different sets of underlay routes can have quite different topologies. Then, we suspect that the full-sized overlay can have quite different routing performance on different sub-graphs, since different graph topologies produce different constraint sets for the overlay routing optimizer. This is illustrated in the following simple example.

[4]Note, this is different from Braess's paradox [17] in which adding more links might increase the cost.

We conducted experiments on a 9-node network [15] (Figure 1) in which all links have the same capacity. We consider a homogeneous traffic matrix in which there are identical traffic demands between all pairs of nodes. For each link weight setting, we obtain a reduced sub-graph, and compute the cost of the shortest path routing Φ_u. We then let the overlay include all nodes, and compute the routing cost Φ_o for each sub-graph. We run many instances of such careless underlay routing algorithms to get pairs $(\Phi^*/\Phi_u, \Phi^*/\Phi_o)$, the *routing performance* P of both underlay and overlay. By scaling up and down all link capacities, we distinguish three load settings: light, moderate, and heavy.[5] For each load setting, we run 2000 instances and compute the underlay and overlay routing performances.

As shown in Figure 2, for each instance of underlay routes, the overlay is always able to achieve better performance than the underlay, which can be verified by observing that all points lie above the diagonal line. We also see that some careless underlay routes are so bad that the overlay cannot compensate much. For example, a large part of the light load curve is below $y = 0.6$, indicating that the full-sized overlay on those sub-graphs can only achieve less than 60% performance of the optimal overlay.

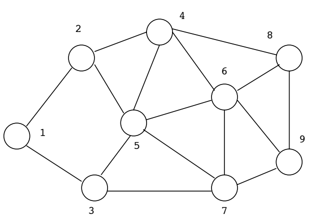

Fig. 1. A nine-node underlay network topology.

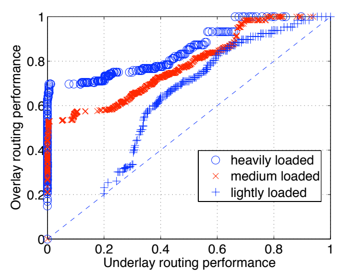

Fig. 2. Overlay routing performance versus underlay routing performance.

To further understand the statistical properties of the careless underlay, we classify underlay routes into ten levels based on their performance, and within each level, we use boxplots to plot the range of overlay routing performance. These are shown in Figures 3, 4 and 5. First we note that for most levels, there is a range of overlay routing performances. This implies that for a given level of carelessness, there might exist different topologies such that the overlay is able to achieve better performance in some topologies than in some other topologies. Thus, it is important to study the topological properties of the underlay routing graphs when proposing basic requirements for the underlay. In addition, we notice that in the heavy load setting, the range of overlay routing performances tends to be smaller. This is understandable, since in a heavy load setting, the overlay has less flexibility to improve its performance.

C. Graph-theoretic metrics related to routing performance

Our objective is to identify a minimal set of graph-theoretic metrics as reasonable predictors of overlay routing performance. We mainly consider three metrics: characteristic path length (CPL)[5], average cut-set size, and weighted node degree sum.

[5]By light load setting, we refer to the case in which the maximum link utilization is 20% when Φ^* is achieved. Similarly, for the medium and heavy load setting, the maximum link utilization is 30% and 40% respectively.

Characteristic Path Length. In [5], the *characteristic path length* (CPL) of a graph is defined as the median of the means of the shortest path lengths connecting each vertex to all other vertices. More precisely, in a graph $G = (V, E)$, for each node $u \in V$, let $d(u)$ denote the average shortest path length [6] to all other possible nodes in V. Then, the CPL is the median of $d(u), \forall u \in V$. In this paper, we choose to use the average of $d(u)$. Intuitively, a graph with small CPL produces low routing cost because traffic traverse short paths on average.

Average Cut-Set Size. Let $\delta(S, \bar{S})$ denote the set of links of a cut-set $(S, \bar{S})$ of a graph $G = (V, E)$. Then the average cut-set size is the average over all cut-set sizes: $\bar{A} = \frac{1}{2^{|V|}-2} \sum_{(S,\bar{S})} |\delta(S, \bar{S})|$. The average cut-set size gives a rough measure of the number of available alternative resources for all demand pairs. The higher the average cut-set size is, the lower the routing cost will be. Note that we can also define a generalized version of cut-set size. Let $C(S)$ denote the aggregate link capacity of all links connecting this cut, namely, $C(S) = \sum_{e\in\delta(S,\bar{S})} C_e$. Let $D(S)$ denote the total traffic demand between nodes in S and $\bar{S}$. Then the generalized average cut-set size will be $\text{mean}_{S\subset V} C(S)/D(S)$. A related metric is *cut sparsity*[7], defined as $\min_{S\subset V} C(S)/D(S)$. Cut sparsity is a natural upper bound for the maximum link congestion [7] in the multi-commodity flow problem, which studies how large one can increase concurrently all traffic demands of a given traffic matrix on a given network and obey the link capacity constraint. However, the routing model studied in this paper differs from the multi-commodity flow problem in that the routing cost defined in this paper can be thought of as the average cost for each traffic demand pair or the average link cost. Thus, average cut-set size or its generalized version is more closely related to the routing cost studied in this paper. We are mainly interested in the average cut-set size instead of its generalized version, since it is traffic matrix independent. In the rest of the paper, for brevity, we simply refer to the average cut-set size as cut average.

Weighted node degree sum (WNDS) is a weighted sum of the degrees of all nodes of a graph. More precisely, WNDS is

$$wnds = \sum_{u\in V} w_u(d_u)d_u$$

where d_u is the degree of node u and $w_u(d_u)$ is the weight of a node, a function of its degree:

$$w_u(d) = \begin{cases} 1, & d \geq 6 \\ 3, & d = 5 \\ 10, & d = 4 \\ 70, & d = 3 \\ 500, & d = 2 \\ 5000, & d = 1 \end{cases} \quad (5)$$

We give larger weights to nodes with smaller degrees since a small degree node is more likely to have highly utilized links. If we know the traffic demand matrix, then we can modify this metric such that a node will have larger weight if the ratio of traffic demands over the sum of its degrees is larger.

[6]Note that the path length is the number of hops and differs from that defined for the careless underlay for which the path length is the sum of link weights.

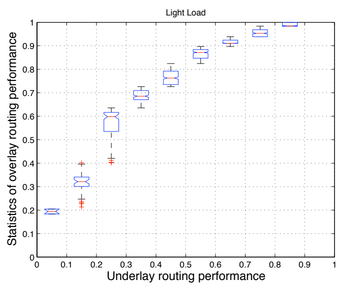

Fig. 3. Boxplot for overlay routing performance in ten levels of underlay routes. Load level is light.

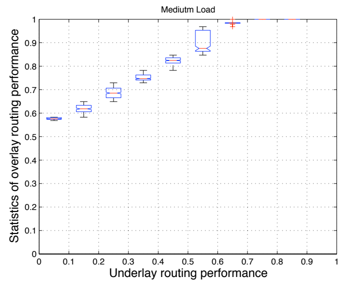

Fig. 4. Boxplot for overlay routing performance in ten levels of underlay routes. Load level is medium.

Fig. 5. Boxplot for overlay routing performance in ten levels of underlay routes. Load level is heavy.

Remarks. First, we note that the characteristic path length measures the *average tightness* of a graph, namely, how close a node is to each other node on average. On the other hand, the cut average measures the *average thickness* of a graph, that is, how many links exist within a graph for a given number of nodes. Second, finding the cut sparsity of a graph is known to be NP-hard [7]. In addition, calculating the cut average requires to check all possible cut-sets, thus, it also takes exponential time. Thus, we use a sampled average as an approximation.

D. *Analysis of linear cost function*

In this section, we present an analytic study to relate routing cost with the characteristic path length (CPL) in the case of linear cost function.

We show first that the CPL completely determines the cost of an overlay routing optimizer under some assumptions.

Theorem 1: For the linear link cost function defined in (2), if all links have the same capacity and the traffic matrix to be routed by the overlay is homogeneous, and if the overlay includes all nodes in the network, then the cost of the overlay routing optimizer is a linear function of the graph's characteristic path length (CPL).

Proof: Since the overlay includes all nodes, the overlay routing optimizer is equivalent to the underlay routing optimizer. Thus, in this proof, our arguments are directly on the physical network.

Suppose that the demand for SD pair i is d^i. The percentage of demand SD pair i allocated on the k-th link of path p_j is f_k^{i,p_j}. This k-th link must be some physical link e in the network. Let $v_k^{-(i,p_j)}$ denote the sum of all other traffic rates of other SD pairs allocated on this link or the traffic rates of this SD pair i on different paths also using this link. Let C be the capacity of all links. Since the link cost function is linear, the cost of link e can be divided into two parts. One part is contributed by SD pair i's path p_j:

$$\Phi_e^{i,p_j} = d^i f_k^{i,p_j} a/C$$

The other part of the cost is contributed by $v_k^{-(i,p_j)}$: $v_k^{-(i,p_j)} a/C_e$. Since the link cost function is linear, the outcome of the overlay routing optimizer will be such that SD pair i always sends its traffic on the shortest path among n^i available paths. Then, we have

$$\Phi^i = \sum_{j=1}^{n^i} \Phi^{i,p_j} = \Phi^{i,p^*} = d^i |m^{i,p^*}| a/C$$

where p^* is the shortest path for SD pair i and $|m^{i,p^*}|$ is the length or hop-count of p^*.

Since this is true for all SD pairs, then the total cost of the overlay routing optimizer is

$$\Phi = \sum_{i=1}^{|I|} \Phi^i = \sum_{i=1}^{|I|} \Phi^{i,p^*} = a \sum_{i=1}^{|I|} |m^{i,p^*}| d^i / C \tag{6}$$

where I is the set of all source destination demand pairs.

If all traffic demands are the same, then (6) becomes

$$\Phi = ad|I|\bar{m}/C \tag{7}$$

where $\bar{m}$ is the average shortest path length, i.e., the characteristic path length. ∎

Remarks.

This theorem provides us the underlying reason why the CPL is important for determining routing cost. Note that the M/M/1 queuing cost function (approximated by a piece-wise linear function) is approximately equivalent to the linear cost function when the link utilization is small, which is common in the current Internet backbones [14], and note that the links in the backbone are approximately homogeneous. Thus, we believe the above result is practically meaningful. As we show later in the experimental studies, when the network load level is light, indeed we observe a nearly linear functional relationship between cost and CPL when using the piece-wise linear function. However, if the link cost function is not linear, then the overlay routing optimizer will allocate traffic demand of a SD pair to multiple paths. Thus, the CPL (tightness) of a graph is no longer the only factor to affect the routing cost. We might need to consider the thickness as well. We currently do not have an analytic solution for this problem, so, we investigate it by using experimental studies in the following section.

Another interesting question is: is there still a linear functional relationship between routing cost and CPL if there are variations in traffic demands? We can show that this is actually true asymptotically with probability one, and that the coefficient of this linear function can be computed from the sampled average of traffic demands $\bar{d}$. If we assume that traffic demands are varying but stationary and they are generated from distribution d, then we can take samples of demand d_j $(j = 1, 2, ...n)$ and compute the sample means $\bar{d}_n = (\sum_{j=1}^{n} d_j)/n$. Here we sample from the distribution that generates demands. And if we assume that the shortest path length m between a node pair is a random variable, then we can also compute the sample means $\bar{m}_n = (\sum_{j=1}^{n} m_j)/n$. We assume that d and m are independent. The weak law of large numbers indicates that these sample means converge in probability to the population means. That is, for all $\epsilon > 0$,

$$\lim_{n\to\infty} P(|\bar{d}_n - E[d]| < \epsilon) = 1; \quad \lim_{n\to\infty} P(|\bar{m}_n - E[m]| < \epsilon) = 1$$

Then we get ([18], page 262)

$$\lim_{n\to\infty} P(|\bar{d}_n \bar{m}_n - E[d]E[m]| < \epsilon) = 1$$

where $E[d]$ and $E[m]$ are population means of random variable d and m. That is, $\bar{d}_n \bar{m}_n$ converges in probability to $E[d]E[m]$ (or $E[dm]$ since d and m are independent.) In our case, $E[m]$ is actually the CPL.

Then the cost computed from the sample means $\bar{d}_n$ and $\bar{m}_n$

$$\Phi_n = (a/C)|I|\bar{d}_n \bar{m}_n$$

will converge in probability to the true cost:

$$\Phi = (a/C)|I|E[dm]$$

This argument indicates that, even if traffic demands vary, the CPL can still be a reasonably good predictor of the routing cost. The linear coefficient between cost and CPL is $(a/C)|I|\bar{d}_n$. This is actually verified in the following experimental studies in which we find that the linear regression between cost and CPL models the experimental data almost perfectly, and the slope of regression function is very close to what we compute from $\bar{d}_n$.

III. Experimental Study

We experimentally explore the relationship between the above graph-theoretic metrics and the overlay routing performance. In the case of linear link costs, our experimental results verify the analysis presented in the previous section. In the case of piece-wise linear increasing and convex link costs, we rely on a large number of experiments to study those important graph-theoretic metrics. We believe that these experimental results provide evidence for the significant correlations between the above metrics and the overlay routing performance. In all our experiments, we let the overlay include all nodes of the network. Thus, this is the best case for the overlay to compensate for the underlay on a given sub-graph produced by the underlay routes.

A. Experimental methodology

Experiment settings: graphs.

We consider three canonical graphs, and one inferred real ISP network by Rocketfuel [19]. The first graph is a completely connected 30-node graph. The second graph is a 30-node purely random graph (Erdos-Renyi random graph [4]) generated by GT-ITM [21]. The probability to create a link between a node pair is 0.2. The third graph is a mesh graph (rectangular 5 by 5 grid). The inferred ISP topology is Cable&Wireless ISP with 33 nodes. We expect that the larger number of nodes a graph has, the more likely we can get rich topology such that the impact of the graph metrics will be revealed if this impact indeed exists. However, we cannot have too large graphs because of the computation constraints. Recall that the routing cost of the set of routes generated by the overlay routing optimizer are obtained by solving a linear programming (LP) problem. The larger the graph is, the more time-consuming it is to solve a LP problem. In our simulations, we find that about 30 nodes are rich enough to find important information.

Experimental settings: network load.

We assume that all traffic demands are routed by the overlay. We first consider a homogeneous traffic demand model in which every node has the same traffic demand d for all other nodes. The network load level is based on a *nominal* load computed as the ratio between the aggregate traffic demand over the aggregate link capacity of the network. More precisely, load level is

$$L = \sum_{i=1}^{|I|} d^i / \sum_{e \in E} C_e$$

Based on this nominal load, we classify network load into ten levels from about 10% to about 100%. [7] We also consider a bimodal traffic matrix [12], and use it to generate a large variation in the demands of the traffic matrix.

Experimental procedures.

A problem instance is a tuple (G, L) where G is a graph and L is the load level. For each problem instance (G, L), we assume link weights are from a uniform distribution $U(1, 100)$. and then determine the shortest paths to get a reduced sub-graph with all nodes preserved but some links missing. We get 500 sub-graphs from the same graph. For each sub-graph, the overlay includes all nodes as overlay nodes and runs the overlay routing optimizer to obtain overlay routes and the corresponding cost.[8] For linear link costs, we only study the relationship between overlay routing cost and CPL, because it is predicted to have a linear functional relationship in Theorem 1. For piece-wise linear costs, we investigate the statistical correlations between routing performance P with graph metrics CPL and cut average. Recall P (defined in Section II-B) is the overlay routing cost normalized by the optimal overlay routing cost. If these correlations are statistically significant, then we compute their correlation coefficients r and their confidence intervals. For each problem instance (G, L), we also visually present the relationships between P and each graph metric by using the scatter plots. Since CPL and cut average only capture the average behavior of a graph, to investigate some potential bad extreme cases, we also studied the importance of weighted node degree sum.

B. Experiments with linear cost function

Homogeneous traffic demands.

[7] This nominal load is just an average value, so it is very likely that some link might have over 100% utilization even if this load level is below 100%.

[8] This overlay routing optimizer is equivalent to the underlay routing optimizer working on the reduced sub-graph.

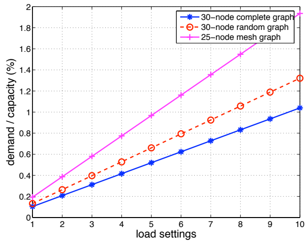

Fig. 6. Load levels for three canonical graphs.

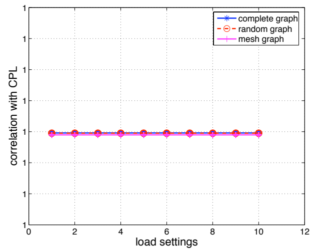

Fig. 7. Correlation coefficients of overlay routing cost with CPL.

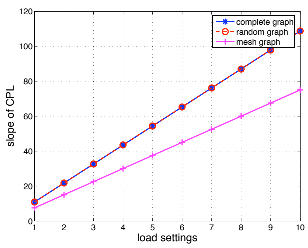

Fig. 8. Slope of the linear regression of overlay routing cost over CPL.

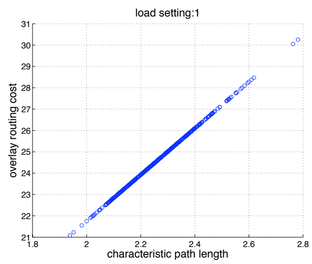

Fig. 9. Scatter plot of overlay routing cost versus CPL when load level is 1, on 30-node complete graph.

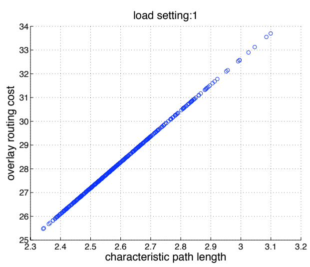

Fig. 10. Scatter plot of overlay routing cost versus CPL when load level is 1, on 30-node random graph.

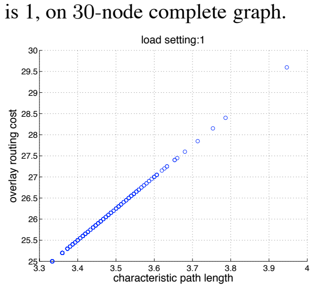

Fig. 11. Scatter plot of overlay routing cost versus CPL when load level is 1, on 25-node mesh graph.

We have 30 problem instances, based on the combination of graphs and network load levels. Figure 6 shows the nominal load levels realized in our experiments. For each problem instance, we generate 500 instances of careless underlay sub-graphs using the procedure described in Section III-A. We compute the slope of linear regression and the correlation coefficient between overlay routing cost and the CPL.

Recall that a linear regression relates the dependent variable y (cost in our case) to independent variable x (CPL in our case) through a linear function plus a normal random variable $\epsilon \in N(0, \sigma^2)$: $y = \beta x + \epsilon$. Correlation coefficient measures how well this linear function models the relationship between y and x. If the correlation coefficient is close to 1, then we say that there exists an almost perfect linear correlation between y and x. In addition, the linear regression technique estimates slope β to get $\hat{\beta}$. The larger the $\hat{\beta}$ is, the more impact the x will have on y.

Figure 7 shows that there are perfect linear correlations between the overlay routing cost and CPL for all three graphs, which are also verified by the sample scatter plots in Figure 9, 10 and 11. Recall that Theorem 1 predicts that the slope between the cost and CPL is $\beta = ad|I|/C$. In our experiments, we choose $a = 1.5$ and $C = 120$. For a 30-node graph, I is 870, then for load level 1 with $d = 1$, we get $\beta = 10.875$, which is what we obtain from the linear programming solver and is shown in Figure 8. For all other graphs and other load levels, we can also easily verify these slopes.

Bimodal traffic matrix.

To study the effectiveness of the CPL when the traffic matrix has widely varying demands, we consider a bimodal traffic matrix [12] and vary the coefficient of variation (CV) of the generated traffic matrix to simulate a large range of variations. More precisely, coefficient of variation (CV) is defined as the ratio of the standard deviation to the mean of a random variable. The bimodal traffic matrix in [12] is generated from two normal distributions and CVs of these two distributions are $(0.125, 0.05)$. We scale up the CVs by $1, 5, 10, 15, 20$ to create five levels of variability. Within each level of variability, we again conduct experiments on 10 different load levels (500 sub-graphs are generated for each level) as before.

Figure 13 illustrates results for a 30-node complete graph. First, we see that the sample CVs of different variability levels vary from 0.48 to 1.05 (shown in the legends of the figure.) Second, we note that, for different levels of variability, the correlation coefficients between overlay cost and CPL are consistently close to 1. And they are similar to those obtained previously from the homogeneous traffic matrix. These almost perfect linear functional relationships can also be verified by looking at an example scatter plot of routing cost versus the CPL when the CV is the largest and the load level is the smallest, shown in the fourth plot of Figure 13. These results are also true for a 30-node random graph, shown in Figure 14.

Analysis in Section II-D predicts that the routing cost asymptotically exhibits a linear function relationship with average demand $\bar{d}_n$ as $\Phi = (a\bar{d}_n|I|/C)\bar{m}$. This can be verified for this complete graph and random graph. Let k_n denote the coefficient $(a\bar{d}_n|I|/C)$. For example, Figure 12 shows that the actual regression slopes and k_n(with $n = 870$) are very close when the experiments are conducted on the random graph at 4 different load levels and when the variability level is the highest. We find that this is also true for all other settings.

	level 1	level 2	level 3	level 4
Regression slope	8.43	16.86	25.29	33.72
Predicted slope	8.39	16.78	25.17	33.56

Fig. 12. Example comparison between slopes obtained by linear regression and predicted slopes.

In Figure 13 and 14 we see that within each variability level (a given CV), as load increases, slope of the linear regression (or coefficient of the linear function between overlay cost and the CPL) will increase. This is consistent with the homogeneous demand case. Furthermore, we see that as variability increase, the slopes increase also, which means that the linear impact of CPL on routing cost becomes more dramatic. This is expected for the following reasons. Analysis in Section II-D predicts that as the average load level increases the coefficient of the linear function increases also, and we see that the realized load level increases as the variability increases, based on the first plot on either Figure 13 or 14.

C. *Experiments with piece-wise linear convex costs*

Homogeneous traffic matrix.

The network load levels in simulations are the same as for the case of linear costs before. Again, we have 30 problem

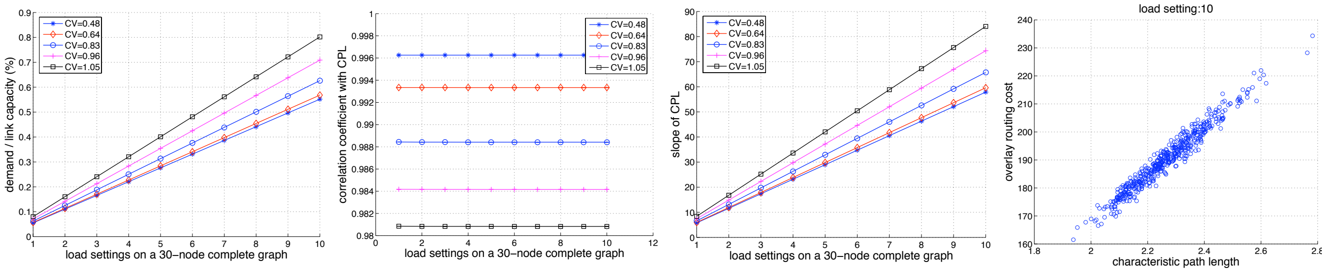

Fig. 13. Experimental results of linear cost function on a 30-node complete graph with bimodal traffic matrices with five different levels of coefficients of variations (CV). The first plot shows the nominal load levels for each level of CV. The second plot shows the correlation coefficients between the overlay cost and the CPL. The third plot shows the slopes of the linear regressions between the overlay cost and the CPL. The fourth plot is a scatter plot of routing cost versus the CPL in the case of the largest variability and load level of 1.

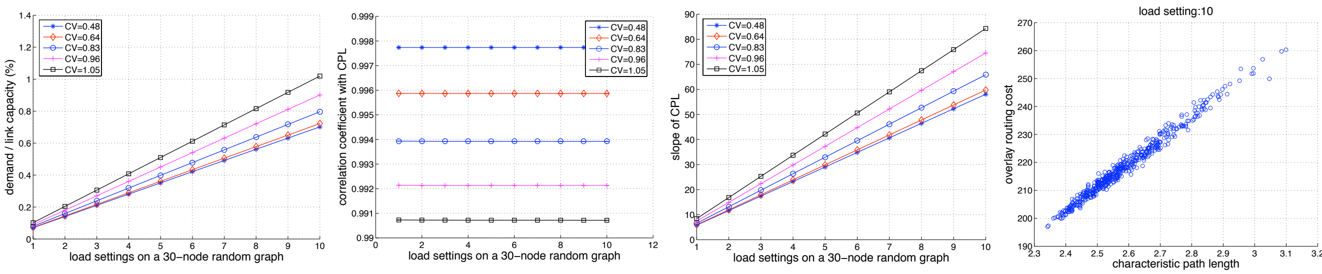

Fig. 14. Experimental results of linear cost function on a 30-node random graph with bimodal traffic matrices with five different levels of coefficients of variations (CV). The first plot shows the nominal load levels for each level of CV. The second plot shows the correlation coefficients between the overlay cost and the CPL. The third plot shows the slopes of the linear regressions between the overlay cost and the CPL. The fourth plot is a scatter plot of routing cost versus the CPL in the case of the largest variability and load level of 1.

instances. Recall that for piece-wise linear convex link cost function, the overlay routing optimizer will very likely split the demand of a SD pair on multiple paths. Thus, we need also to examine the thickness (average cut-set size) of the graph.

For each problem instance, we compute the correlation coefficients between the normalized overlay routing performance P and two metrics: CPL and cut average, shown in Figures 15 and 16. For all problem instances, these two graph metrics are significantly correlated with overlay routing performance P.

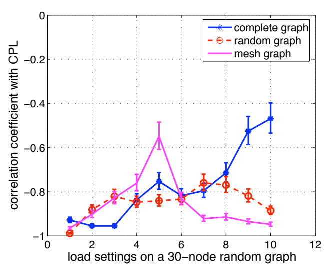

Fig. 15. Correlation coefficient (with 95% confidence interval) between routing performance and the CPL.

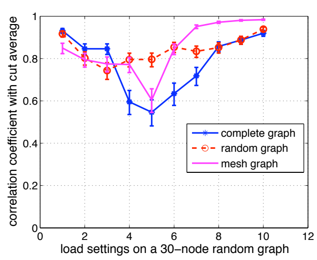

Fig. 16. Correlation coefficient (with 95% confidence interval) between routing performance and cut average.

Figure 15 shows that when load level is low, for all three graphs, both CPL and cut average have close to one correlation with P which indicates that there exist almost perfect linear functional relationships between P and these metrics. We also observe that, for the complete graph, there is a peak when the load level is at 50%. To understand this, we provide scatter plots for all load levels in Figure 17. Notice that when the load level is low $(1, 2, 3)$, overlay routing performance exhibits an almost perfect linear correlation with the CPL. This is similar to the case of linear costs in the last section. As the load level increases up to 50%, some sub-graphs with really bad performance appear. When the load level is moderate ($L = 5, 6, 7, 8$), all sub-graphs divide into two clusters: one with good performance (called the good cluster) and one with bad performance (called the bad cluster). However, within each cluster, there are still strong correlations between performance and these two metrics. And the graphs in the bad cluster in general have worse performance than those graphs in the good cluster. As the load level increases, graphs in the good cluster shift to the bad cluster. Eventually, as the load level becomes high, only the bad cluster is left and within this cluster the correlation is still very strong. We will revisit this cluster phenomenon later. Similarly, Figure 16 shows that there is a large correlation coefficient between P and cut average at all load levels, for all three graphs. This is also verified by the scatter plots in Figure 17.

Strong correlations between overlay routing performance and the CPL or cut average show that these two metrics suffice to explain most of the variability of the overlay routing performance. However, the presence of two clusters at moderate load levels indicates that these two metrics are not always sufficient to predict the performance. Possible reasons are given in the below.

To understand the two cluster phenomenon, we focus on two graphs with the same CPL and with dramatically different performance (i.e., they are in separate clusters.) We plot the histograms of node degrees of these two graphs in Figure 18. It is interesting to note that the graph with bad performance has a node with degree 1. This means that the utilization on this link is well above 1. Thus, the cost on this link is dramatically higher than other links. Recall that when the link utilization is above 1, the slope of the piece-wise linear cost function is as large as 5000. Note that the cut sparsity can capture this difference between these two graphs. However, since finding the cut sparsity is NP-hard, we propose a related metric to capture this dramatic difference in link utilization. The idea is to compute

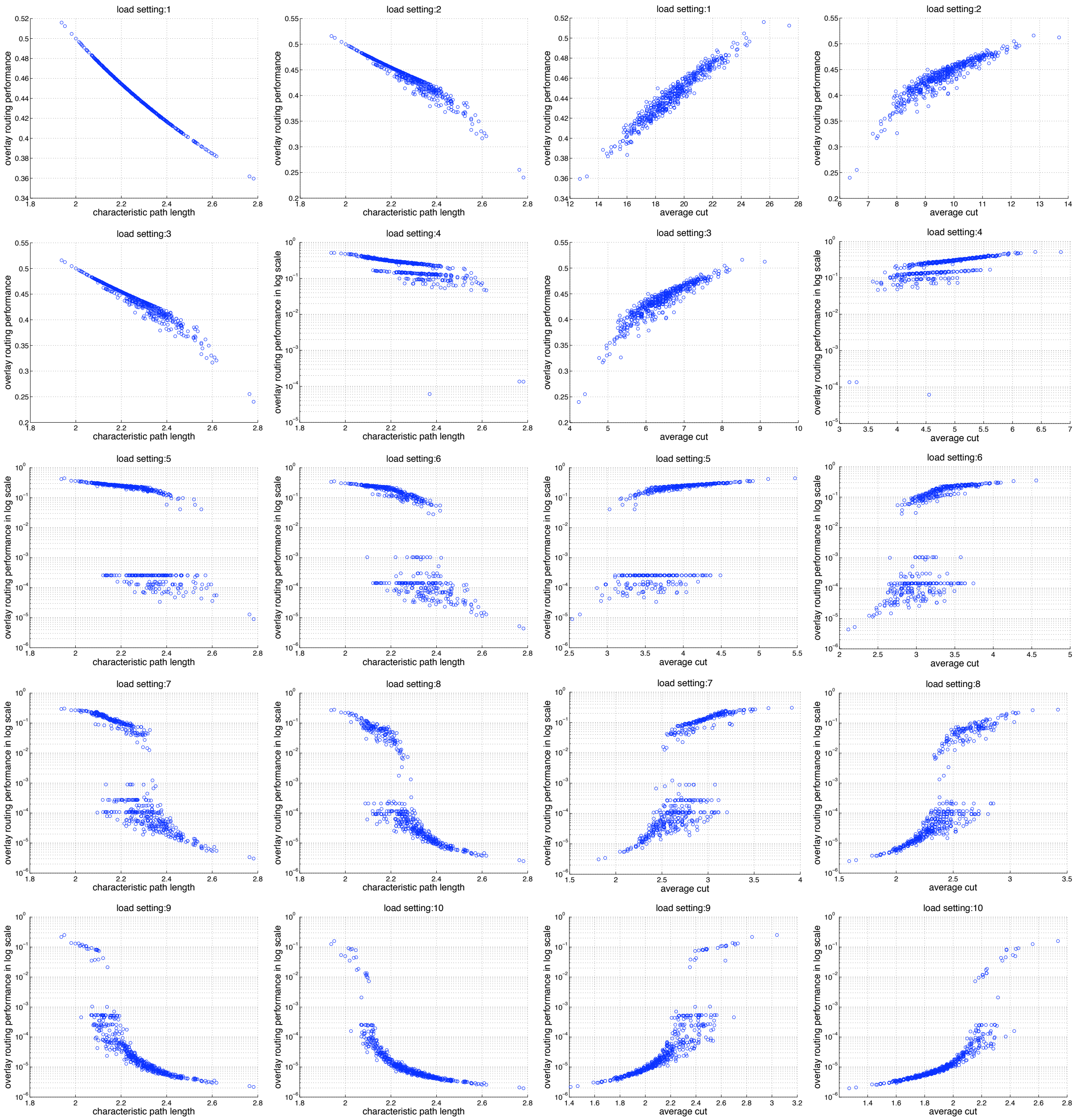

Fig. 17. Completely connected 30-node graph. The left two columns are the scatter plots for overlay performance vs. **characteristic path length**. The right two columns are the scatter plots for overlay performance vs. **cut average**. **The y-axes of load level 4 to 10 are in log scale.** Note, performance of the original complete graph is omitted since it is 1, much higher than those of all other sub-graphs.

the *weighted node degree sum* (WNDS). Recall that we give a larger weight to a node with small degree since the small degree node is more likely to have its links to be highly utilized. Intuitively, WNDS indicates a weighted influence of node degree on the routing performance. We can now set different threshold values based on WNDS to divide graphs into different clusters. For example, a graph with WNDS larger than 5000 is very likely to have a degree one node. As shown in Figure 19, 5000 seems to be a reasonable threshold to classify graphs into two clusters (good and bad performance graphs) when the load level is moderate. Only 3% of the graphs are mis-classified by this metric. Thus, WNDS can be used together with CPL and cut average to differentiate graphs when the load level is moderate.

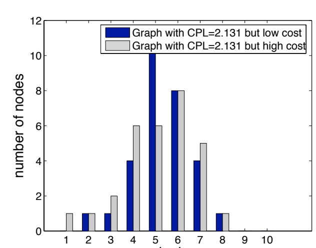

Fig. 18. Comparison of node degrees of two graphs with the same CPL but dramatically different costs.

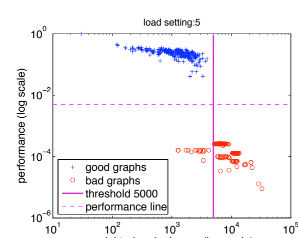

Fig. 19. Sub-graph clusters of 30-node complete graph when $L = 5$ (medium). 5000 is the threshold for weighted node degree sum.

Bimodal traffic matrices

We use the same bimodal traffic matrices as those in studying linear link costs. Recall that we generate five levels of variability. Within each level of variability, we conduct experiments on 10 different load levels. Figure 20 illustrates the results on a 30-node complete graph. We note that, for different levels of variability, the correlation patterns between overlay performance and CPL (or cut average) are very similar. And they are similar to those obtained previously from the homogeneous traffic matrix. Similarly, experimental results on a 30-node random graph are shown in Figure 21. We again see that the correlation coefficients between CPL (or cut average) and overlay performance show consistent patterns across different variability levels of traffic matrices.

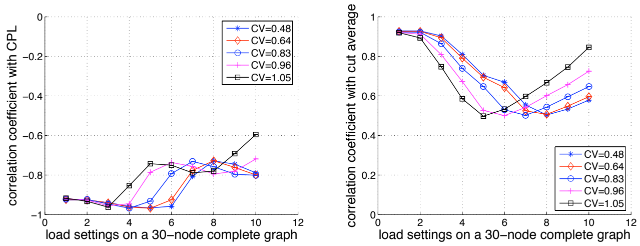

Fig. 20. Experimental results on a 30-node complete graph with bimodal traffic matrices with five different levels of coefficients of variations (CV). The first plot shows the correlation coefficients between the overlay performance and the CPL. The second plot shows the correlation coefficients between the overlay performance and the cut averages.

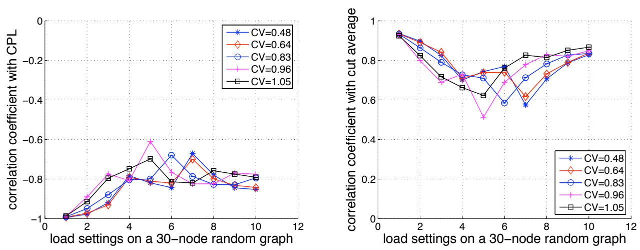

Fig. 21. Experimental results on a 30-node random graph with bimodal traffic matrices with five different levels of coefficients of variations (CV). The first plot shows the correlation coefficients between the overlay performance and the characteristic path length. The second plot shows the correlation coefficients between the overlay performance and the cut averages.

Remark. Experiments and analysis so far suggest that three graph metrics, characteristic path length (CPL) and cut average and weighted node degree (WNDS), are important predictors for routing performance. In general, a graph will have good performance if it has small CPL and large cut average and small WNDS. In the following, we will further verify this conclusion through experiments on an inferred ISP topology.

D. Experiments on an inferred ISP topology

We consider the pop-level network of Cable&Wireless ISP inferred by Rocketfuel project [19]. There are 33 nodes and 116 links. All links are OC-12 with capacity of 622Mbps except for two links having capacity of 1244Mbps. We consider piece-wise linear link costs. We use a homogeneous traffic matrix. Let d denote each of these demands. We present results when network load is light. Specifically, we take $d = 5$Mbps and nominal load level of 9.6%. We generate 1000 sets of underlay routes, and then generate optimal routes based on the sub-graphs. The scatter plots of the routing performance versus the characteristic path length (CPL) and cut average are shown in Figure 22. Note that CPL and cut average are both significantly correlated with the routing performance. The correlation structures are very similar to what we observed before for those three canonical graphs. This verifies that the graph metrics we choose are reasonable predictors of the routing performance.

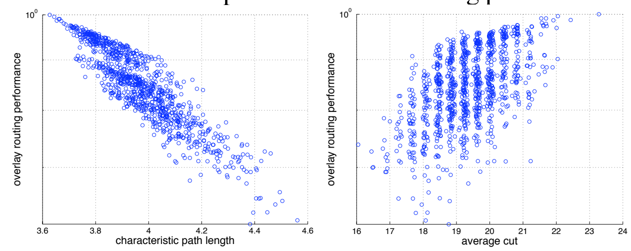

Fig. 22. Cable&Wireless topology. The left column is the scatter plot for overlay performance vs. **characteristic path length**. The right column is the scatter plot for overlay performance vs. **cut average**. **The y-axes of are in log scale.**

E. Requirements to the careless underlay

Based on the previous experiments and analysis, we think that a routing rich underlay graph should have small CPL, large average cut-set size, and small WNDS. Here, we avoid giving any specific threshold values to these three metrics, because the richness of a reduced sub-graphs also depends on the original underlay graph.

All of our studies in this section are concerned with the best-case for the overlay, i.e., the overlay has all nodes in control. Thus, these studies remove the effect of overlay topologies and give the basic requirements for the underlay. That is, *in order for the overlay to effectively compensate for the carelessness of the underlay, the underlay should generate rich sub-graphs, even though it can ignore the optimal routing for any traffic matrix.*

IV. Compensation when overlay size is limited

As we mentioned before, the effectiveness of an overlay compensating for the careless underlay depends on both the overlay and the underlay. From previous sections, we learn that three graph metrics are very strongly correlated with routing performance. Thus, the underlay should choose the richest sub-graph. Otherwise, the carelessness of the underlay cannot be effectively compensated by the *full-sized overlay* (including all nodes.) On the other hand, when an overlay constructs a virtual routing topology, if it can only choose a subset of physical nodes, it should make sure that the virtual topology has the best possible values of these three graph metrics. This is the problem studied in this section. We call this kind of overlay *size-limited overlay*. In the following, we first formulate this problem and then illustrate the benefits through experiments on the inferred Cable&Wireless topology.

A. Problem formulation

Consider a graph $G = (V, E)$. We distinguish two subsets of the vertex set V. One is a set (V_b) of *basic* overlay nodes which have traffic demands to each other. The other is a set (V_c) of available *candidate* nodes, which are potentially to be chosen by the overlay as relay nodes for overlay traffic. From V_c, an overlay selects a subset V_r as the set of relay nodes. Then we have a overlay network G_o formed by the overlay nodes in $V_o = V_b \bigcup V_r$. As for the overlay link set, we can construct a fully connected overlay graph for a small scale overlay, whereas, for a large scale overlay, we can choose some set of logical links. The objective of an overlay is to construct an overlay topology G_o so as to minimize the average cost. If we do not limit the number of relay nodes available to the overlay, the best an overlay can do is to select all nodes in V_c as relay nodes. However, in practice, an overlay may not be able to put an application level relay node along with each physical router. To capture this, we let the overlay choose only $|V_r|$ nodes as relay nodes.

As discussed previously, since a graph with a small CPL, large cut average, and small WNDS provides good routing performance, the objective of overlay topology construction should be to optimize these three metrics. Since the overlay cannot include all nodes to obtain the sub-graph produced by the underlay routes, these metrics cannot be defined directly on the sub-graph. Thus, we need to re-define them as follows.

First, for an overlay graph $G_o = (V_o, E_o)$, we associate a weight to each virtual link equal to the number of hops of the corresponding physical path. This weight is determined by the underlay routes. Then each virtual path length is the sum of virtual link weights. For each overlay node, we compute its shortest paths to all other nodes. The average of all those paths' lengths is recorded as the overlay characteristic path length cpl_o. Second, for a cut S of G_o with $S \cup \bar{S} = V_o$, let δ_S denote the set of virtual links or physical paths connecting nodes in S to nodes in $\bar{S}$. Note that several physical paths might share a common physical link. We define the size of an overlay cut as the number of *disjoint* paths between nodes in S and nodes in $\bar{S}$. Then the average of these cut sizes is the overlay average cut size c_o. Third, we define the number of disjoint physical paths from an overlay node to all other overlay nodes as the degree of that overlay node. Then, we can define the weighted node degree sum $wnds_o$ similarly as in last section. It can be verified that these definitions for size-limited overlay are consistent with those for full-sized overlay.

Then, constructing a rich size-limited overlay topology is equivalent to searching for a topology with a small cpl_o, large c_o, and small $wnds_o$. Recall that, for linear cost or nonlinear convex cost with light load, the CPL alone can determine routing cost. However, in other cases, we need to consider all three metrics. Thus, an ideal measure of richness might be some combination of all three metrics. We do not have a complete solution for this at this time and choose it as a future research topic. Nevertheless, we still can show the benefits of considering richness by just looking at cpl_o and $wnds_o$. To this end, we give a randomized heuristic as follows.

B. A randomized heuristic and experimental results

A randomized heuristic. If the overlay can afford $|V_r|$ relay nodes, then it can exhaustively search all possible combinations to find out the best topology, which can take exponential time. So, we rely on a randomized heuristic shown in Figure 23. The idea is to randomly pick a set of nodes from the candidate set to form a potential overlay node set, and compute its cpl_o. This process executes a pre-determined number of times. We keep a record of the set of graphs with the smallest cpl_o and then pick the one with the lowest $wnds_o$. Here, we assume that the overlay graph is always fully connected.

input:
G: graph on which to setup overlay
V_b: set of basic overlay nodes
V_c: set of candidate overlay nodes
n: number of loops
δ: relaxation factor in $(0, 1)$
output:
$V_o^{opt} = V_b \bigcup V_r$: set of overlay nodes with the best richness
start:
$\mathbb{V} = \emptyset$: initialize the set of candidate node sets;
repeat n times
  randomly choose a set V_r' of nodes from set V_c;
  construct overlay topology with $V_o' = V_b \bigcup V_r'$;
  compute weight for each virtual link;
  compute the cpl_o for this topology;
  add V_o' into $\mathbb{V}$;
  find the V_o' with the smallest $cpl_{o,\min}$;
  remove any V_o' in $\mathbb{V}$ if its $cpl_o > (1+\delta)cpl_{o,\min}$;
  find V_o^{opt} with minimum wnd_o in $\mathbb{V}$;
return V_o^{opt}

Fig. 23. Randomized heuristic for constructing overlay topology.

Experiments on Cable&Wireless network. To demonstrate the benefit of considering richness when constructing a size-limited overlay topology, we experiment with the inferred Cable&Wirelss network. We randomly choose 5 nodes as basic overlay nodes and all nodes have the same demands to each other. Suppose that we want to choose another 5 nodes as relay nodes from the remaining 33 nodes. First, we use the proposed random heuristic to construct the richest overlay topology. We only consider minimizing cpl_o in this example since the network load is light. Then, we randomly choose 5 relay nodes to generate 100 random overlay topologies. For each topology, we compute the routing cost for five different load levels.

The average cost of all 100 random overlay topologies and the cost of the richest constructed overlay topology are plotted in Figure 24. We see that the heuristically identified richest overlay topology is always better than the average overlay cost. In Figure 25, we plot the percentile of the richest overlay's cost among all overlay topologies. We see that the richest overlay always has a very low cost compared with other topologies at all load levels.

This example demonstrates the usefulness to consider the richness when constructing a size-limited overlay topology. Further study is still needed to refine these metrics and constructing better algorithms.

V. Related work

Many works recently have focused on the routing problems associated with underlay networks or overlay networks. These works can be differentiated from each other as follows: 1) performance objective, either minimizing the average cost (delay) or finding better alternative for a source destination pair; 2) inter-AS or intra-AS routing; 3) layers, either underlay network or overlay network. Regarding the underlay routing in a single ISP, [6] proposes a technique to optimize OSPF link weights for achieving optimal routing of a given traffic matrix. Following

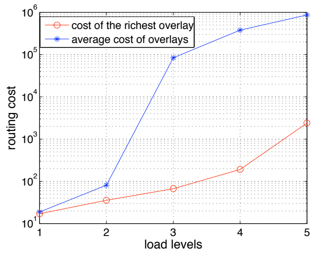

Fig. 24. Comparison of costs between the richest overlay topology and the average overlay topology.

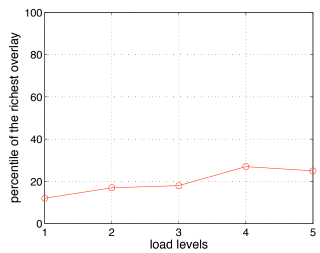

Fig. 25. Percentile of the cost of the richest overlay topology out of all constructed overlay topologies.

that, there have been many works on this topic. As for overlay routing, most of the works concentrate on finding better alternate paths across different ISPs (e.g. RON [2].) There are also some works considering the underlay topology in constructing the overlay topology. For example, [8] studies how to place overlay nodes to maximize path independence without degrading performance (the fraction of successfully recovered path outrages.) [10] considers two overlay routing protocols (link-state and feedback) and several common topologies. They use simulations to evaluate all possible combinations of topologies and protocols in terms of failure recovery ratio, routing overhead, and overlay path penalty. The interaction between the overlay routing and the underlay routing is introduced by [15] in which performance measure is either the average cost or cost of the traffic equilibrium (i.e. selfish routing [17]). [11] studies this interaction problem in a game-theoretic framework. [16] studies a similar problem by using simulations on both intra- and inter-AS underlay networks.

Since our work is concerned with topological issues, we find several works on network topology quite relevant, e.g., [5], [20]. There are also some works relating the network topology with the routing performance. For example, [7] presents some results relating some graph-theoretic metric with multi-commodity flow problem, in which they study how large one can scale up all traffic demands simultaneously on a given network and obey link capacity constraint. The performance measure is the maximum scaling factor. This performance measure is also used in [9] to evaluate several ISP topologies at the router level.

VI. Conclusions and Future Work

In this paper, we study whether an overlay routing controller can compensate for a careless underlay routing controller.

The careless underlay is modeled as a shortest path routing algorithm applied to a network graph with randomly assigned link weights independent of network cost and link capacities. We have identified three graph-theoretic metrics that collectively characterize the richness of the sub-graphs formed by the physical routes set by the careless underlay. They are the characteristic path length (CPL, or tightness), the average cut-set size (or thickness), and the weighted node degree sum. For linear link cost function, we have proved that the CPL can determine the cost of the overlay routing optimizer through a linear function under some assumptions. These results are verified in our experimental studies. For piece-wise linear convex increasing costs (approximations of nonlinear convex costs), our experimental studies have revealed a significant correlation between the overlay routing cost and the CPL (and the average cut-set size.) When the network load is light or heavy, the correlation structures associated with the piece-wise linear link costs are similar to those with the linear link costs. However, when the network load is moderate, the nonlinearity of the cost function requires us to consider node degree as well, since highly utilized links contribute significantly more cost than lightly utilized links. Finally, we study how to apply the notion of richness to the constructing of a fixed size overlay topology. We show that a simple heuristic considering richness helps to improve the overlay performance dramatically on some inferred ISP topology.

Our study is a starting point for further investigation on this topic. There are many interesting open questions. For example, an analytic study of the nonlinear cost function is needed to find better metrics to more precisely characterize the richness. It is also interesting to extend our study to inter-AS graphs.

VII. Acknowledgment

The authors would like to thank Sharad Jaiswal, Yong Liu and Kyoungwon Suh for their help.

References

[1] Open Shortest Path First (OSPF), http://www.ietf.org/html.charters/ospf-charter.html.
[2] D. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proc. 18th ACM Symposium on Operating Systems Principles*, Banff, Canada, October 2001.
[3] D. Applegate and E. Cohen. Making intra-domain routing robust to changing and uncertain traffic demands: Understanding fundamental tradeoffs. In *ACM SIGCOMM 2003*, Karlsruhe, Germany.
[4] B. Bollobas. *Random Graphs*. Academic Press, Orlando, Florida, 1985.
[5] T. Bu and D. Towsley. On distinguishing between internet power law topology generators. In *IEEE INFOCOM 2002*.
[6] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. In *IEEE INFOCOM 2000*.
[7] C. Gkantsidis, M. Mihail, and A. Saberi. Conductance and congestion in power law graphs. In *ACM SIGMETRICS 2003*.
[8] D. W. Junghee Han and F. Jahanian. Topology aware overlay networks. In *IEEE INFOCOM 2005*.
[9] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the internet's router-level topology. In *SIGCOMM 2004*.
[10] Z. Li and P. Mohapatra. The impact of topology on overlay routing service. In *IEEE Infocom 2004*.
[11] Y. Liu, H. Zhang, D. Towsley, and W. Gong. On the interaction between overlay routing and underlay routing. In *IEEE INFOCOM 2005*.
[12] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: Existing techniques compared and new directions. In *ACM SIGCOMM 2002*.
[13] A. Nakao, L. Peterson, and A. Bavier. A routing underlay for overlay networks. In *ACM SIGCOMM 2003*.
[14] A. M. Odlyzko. Data networks are mostly empty and for good reason. In *IT Professional 1 (no. 2)*, March/April 1999.
[15] L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker. On selfish routing in internet-like environments. In *ACM SIGCOMM 2003*.
[16] C. C. R. Keralapura, N. Taft and G. Iannaconne. Can isp's take the heat from overlay networks? In *ACM HotNets Workshop 2004*.
[17] T. Roughgarden and E. Tardos. How bad is selfish routing? In *IEEE Symposium on Foundations of Computer Science*, pages 93–102, 2000.
[18] A. Shiryaev. *Probability*. Springer-Verlag New York Inc., NY, 1995.
[19] N. Spring, R. Mahajan, and D. Wetherall. Measuring isp topologies with rocketfuel. In *ACM SIGCOMM 2002*.
[20] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. Network topology generators: Degree-based vs structural. In *ACM SIGCOMM 2002*.
[21] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee. How to model an internetwork. In *IEEE INFOCOM 2006*.