

# Subjective-cost policy routing<sup>☆</sup>

Joan Feigenbaum<sup>a</sup>, David R. Karger<sup>b</sup>, Vahab S. Mirrokni<sup>b</sup>, Rahul Sami<sup>c,\*</sup>

<sup>a</sup> Yale University Computer Science Department, 51 Prospect St., New Haven, CT 06511, USA

<sup>b</sup> MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St, Cambridge, MA 02139, USA

<sup>c</sup> University of Michigan School of Information, 1075 Beal Ave., Ann Arbor, MI 48109, USA

## Abstract

We study a model of path-vector routing in which nodes' routing policies are based on *subjective* cost assessments of alternative routes. The routes are constrained by the requirement that all routes to a given destination must be confluent. We show that it is NP-hard to determine whether there is a set of stable routes. We also show that it is NP-hard to find a set of confluent routes that minimizes the total subjective cost; it is hard even to approximate the minimum cost closely. These hardness results hold even for very restricted classes of subjective costs.

We then consider a model in which the subjective costs are based on the relative importance nodes place on a small number of objective cost measures. We show that a small number of confluent routing trees is sufficient for each node to have a route that nearly minimizes its subjective cost. We show that this scheme is trivially strategy proof and that it can be computed easily with a distributed algorithm. Furthermore, we prove a lower bound on the number of trees required to contain a  $(1 + \epsilon)$ -approximately optimal route for each node and show that our scheme is nearly optimal in this respect.

© 2007 Elsevier B.V. All rights reserved.

**Keywords:** Policy routing; Path-vector routing; Algorithmic mechanism design

## 1. Introduction

In any networked computation environment, routing (finding paths from each node in the network to every other node) is an essential task. Two families of routing protocols are widely used: *link-state* routing protocols and *path-vector* routing protocols. In link-state protocols, the state of the entire network is maintained by every single node; this is updated whenever there is a change anywhere in the network. In a path-vector protocol, each node only maintains a local map of the network that includes the paths currently used by neighboring nodes, leading to significant space and communication savings. In this paper, we focus on path-vector routing protocols.

<sup>☆</sup> This work was supported by the DoD University Research Initiative (URI) administered by the Office of Naval Research under Grants N00014-01-1-0795 and N00014-04-1-0725. An extended abstract of this paper appeared in [J. Feigenbaum, D. Karger, V. Mirrokni, R. Sami, Subjective-cost policy routing, in: Proceedings of the Workshop on Internet and Network Economics, in: Lecture Notes in Computer Science, vol. 3828, 2005, pp. 174–183] [2].

\* Corresponding author.

E-mail addresses: [feigenbaum@cs.yale.edu](mailto:feigenbaum@cs.yale.edu) (J. Feigenbaum), [karger@csail.mit.edu](mailto:karger@csail.mit.edu) (D.R. Karger), [mirrokni@csail.mit.edu](mailto:mirrokni@csail.mit.edu) (V.S. Mirrokni), [rsami@umich.edu](mailto:rsami@umich.edu) (R. Sami).

With path-vector routing, routes are propagated by a node announcing its current path to a given destination to some or all of its neighbors. A node  $i$  with many neighbors may thus receive announcements of many different routes to a given destination  $j$ . It can then select (at least) one of these available routes as the route it will use to send its traffic; subsequently,  $i$  can announce this chosen route (prefixed by  $i$  itself) to its neighbors. Proceeding in this manner, every node in the network can eventually discover at least one route to destination  $j$ . Thus, one of the key decisions that has to be made at each node is that of *route selection*: Given all the currently available routes to destination  $j$ , which one is traffic sent on?

At first glance, it may seem as though nodes should always pick the shortest route; in practice, however, node preferences may be greatly influenced by other factors, such as perceived reliability or preferences over which outgoing link is used. This is particularly important when nodes have some degree of autonomy (i.e., they are controlled by different entities). In this case, each node may independently choose a *routing policy* that governs its route selection. The resulting routing scheme is called *policy-based routing*, or *policy routing* for short. With policy routing, the global routes depend on all the individual nodes' policy choices. Achieving efficient routing thus requires coordination of routing policies as well as attention to the incentives and preferences of autonomous nodes.

### 1.1. Path-vector routing in the Internet

Policy routing has chiefly been studied in the context of *interdomain routing*. The Internet is divided into many *Autonomous Systems* (ASes). Loosely speaking, each AS is a subnetwork that is administered by a single organization. Interdomain routing is the task of routing between different ASes. Currently, the only widely used protocol for interdomain routing is the Border Gateway Protocol (BGP). BGP is a path-vector protocol that allows an AS to “advertise” routes it currently uses to neighboring ASes.

AS route preferences are complex, and are largely dependent on the commercial relationships that an AS has with other ASes. For this reason, BGP allows ASes complete freedom to pick a route according to their own *routing policies*. However, BGP does place one important constraint on routing: it stipulates that an AS can only advertise a route that the advertising AS itself currently uses. This is because of the way traffic is routed in the Internet: routers examine the destination of incoming packets and simply forward the packet to the next hop on the current route to that destination. At a given time, each AS typically has exactly one active route to the destination. Thus, the set of all ASes' routes to a given destination AS  $j$  must be *confluent*, i.e., they must form a tree rooted at  $j$ .

The policy-routing aspect of interdomain routing has recently received a lot of attention from researchers. Varadhan et al. [10] observed that general policy routing could lead to route oscillations. Griffin, Shepherd, and Wilfong [7,6] studied the following abstract model of general policy routing: each AS  $i$ 's policy is represented by a preference ordering over all possible routes to a given destination  $j$ . At any given time,  $i$  inspects the routes all of its neighbors are advertising to  $j$  and picks the one that is ranked highest. AS  $i$  then advertises this route (prefixed by  $i$  itself) to all its neighbors. Griffin et al. proved that, in such a scenario, BGP may not converge to a set of *stable paths*; the routes might keep oscillating as ASes continuously change their selection in response to their neighbors' changes. They further showed that, given a network and a set of route preferences, it is NP-complete to determine whether a set of stable paths exists. In recent work, Feamster et al. [1] showed that instability can arise even for restricted routing policies.

Feigenbaum et al. [4] extended the model of [7] by including cardinal preferences instead of preference orderings. Specifically, they assume that AS  $i$  conceptually assigns each potential route a monetary value and then ranks routes according to their value. The advantage of working with cardinal preferences is that a set of paths can be stabilized by making payments to some of the ASes: although the ASes' a priori preferences may have led to oscillation (in the absence of payments), ASes preferences can be changed if they receive more money for using a less valuable route. This is the basis for the mechanism-design approach to routing, which seeks to structure incentives so as to achieve a stable, globally optimal set of routes; see [4] for further details. In the context of policy routing, the most natural global goal is to select a set of confluent routes that maximizes the *total welfare* (the sum of all ASes' values for their selected routes). However, Feigenbaum et al. showed that, for general valuation functions, it is NP-hard to find a welfare-maximizing set of routes; it is even NP-hard to approximate the maximum welfare to within a factor of  $n^{\frac{1}{4}-\epsilon}$ , where  $n$  is the number of nodes. Thus, in this model too, general routing policies lead to computationally intractable problems.

The natural approach to get around the intractability results is to restrict either the network or the routing policy. Restricting the network alone does not appear to be a very promising direction, because the hardness results hold even for fairly simple networks that cannot be excluded without excluding many “Internet-like” networks. This has led researchers to turn to restricted classes of preferences that can express a wide class of routing policies that ASes use in practice. Feigenbaum et al. [4] study *next-hop* preferences – preferences in which an AS  $i$ ’s value for a path depends only on the next AS on the path – and show that, in this case, a welfare-maximizing set of routes can be found in polynomial time. Next-hop preferences can capture the effects of  $i$ ’s having different commercial relationships with neighboring ASes. Similarly, in the ordinal-preference model, Gao and Rexford [5] show that, with the current hierarchical Internet structure, BGP is certain to converge to a set of stable paths as long as every AS prefers a customer route (i.e., a route in which the next hop is one of its customers) over a peer or provider route; this can also be viewed as a next-hop restriction on preferences.

## 1.2. More expressive preference classes

There are many reasonable policies that cannot be expressed in terms of next-hop preferences alone. Although commercial relationships are the primary determinant of current AS routing policies, understanding other classes of policies that could be handled without abandoning the path-vector approach would be useful as the Internet continues to evolve. In addition, new domains such as wireless ad hoc networks also involve loosely coupled autonomous nodes, and could also benefit from policy-based routing protocols.

In this paper, we study other classes of routing policies that capture plausible node preferences. We examine the global routing behavior under the confluence constraint that requires all routes to a destination to form a tree; as discussed above, this is a natural requirement when packets are forwarded based on the destination alone. For example, a node  $i$  might wish to avoid any route that goes through node  $k$ , either because it perceives  $k$  to be unreliable or because  $k$  is a malicious competitor who would like to drop all of  $i$ ’s traffic. This leads to the *forbidden-set* class of routing policies: for each node  $i$ , there is a set of nodes  $S_i$  such that  $i$  prefers any route that avoids  $S_i$  over any route that uses a node in  $S_i$ . We can then ask the following questions: (1) If each node uses a forbidden-set routing policy, will the path-vector routing protocol converge to a set of stable paths? and (2) Can we find a welfare-maximizing routing tree, i.e., a set of confluent routes that maximizes the number of nodes  $i$  whose routes do *not* intersect the sets  $S_i$ ? If the latter optimization problem were tractable, then this class of routing policies would be a candidate for a mechanism-design solution as in [3].

Forbidden-set policies (and many others) can be framed in terms of *subjective costs*. Each node  $i$  assigns a cost  $c_i(k)$  to every other node  $k$ . Then, the “cost” perceived by node  $k$  for a route  $P$  is  $\sum_{k \in P} c_i(k)$ ; node  $i$  prefers routes with lower subjective cost. Subjective-cost routing is a natural generalization of lowest-cost routing (in which there is a single objective measure of cost that all nodes agree upon). It is well known that lowest-cost routes can be computed easily, and hence we hope that some more general class of subjective-cost routing policies will also be tractable.

However, we find that even very restricted subsets of subjective-cost policies lead to intractable optimization problems. We show that, if all nodes rank paths based on subjective-cost assignments, it is still possible to have an instance in which there is no stable-path solution. Further, given a network and subjective costs, it is NP-complete to determine whether there is a set of stable paths. Moreover, the NP-completeness reduction only requires subjective costs in the range  $\{0, 1, 2\}$  for each node. In the cardinal utility model, the outlook is not much brighter. We show that, even if all subjective costs are either 0 or 1, it is NP-hard to find a set of routes that maximizes the overall welfare; indeed, it is NP-hard even to approximate maximum welfare to within *any* factor. The forbidden-set routing policies can be formulated in terms of 0-1 subjective costs, and hence optimizing for this class is also difficult. We then turn to subjective costs with bounded ratios. We show that, if the subjective costs are restricted to lie in the range  $[1, 2]$ , the problem of finding a confluent tree with minimum total subjective cost is APX-hard; thus finding a solution that is within a  $(1 + \epsilon)$  factor of optimal is intractable. In this case, however, an unweighted shortest-path tree provides a trivial 2-approximation to the optimization problem.

In light of all these hardness results, we consider a more restricted scenario in which the differing subjective cost assignments arise from differences in the relative importance placed on two *objective* metrics, such as latency and reliability. Thus, we suppose that every path  $P$  has two objective costs  $l_1(P)$  and  $l_2(P)$ . We assume that node  $i$  evaluates the cost of path  $P$  as the convex combination  $\lambda_i l_1(P) + (1 - \lambda_i) l_2(P)$ , where  $\lambda_i \in [0, 1]$  reflects the importance  $i$  places on the first metric. Here, too, it is NP-hard to find a routing tree that closely approximates

the maximum welfare. However, if we slightly relax the constraint that each node stores only a single route to the destination, we show that it is possible to find a nearly optimal route, as follows. Given any  $\epsilon > 0$ , we can find a set of  $O(\log n)$  trees<sup>1</sup> rooted at  $j$  with the following property: if each node  $i$  chooses the route it likes best among the  $O(\log n)$  alternatives, the overall welfare is within a  $(1 + \epsilon)$  factor of optimal. This solution can be implemented by replacing each destination with a set of  $O(\log n)$  logical destinations and then finding a lowest-cost routing tree to each of these logical destinations. The results generalize to the convex combinations of  $d > 2$  objective metrics;  $O(d3^d \log^{d-1} n)$  trees are required in this case. This scheme is trivially strategy proof, and, further, it can be implemented with an efficient distributed algorithm.

The rest of this paper is structured as follows. In Section 2, we introduce the subjective-cost model of routing preferences. In Section 3, we study the stable-paths problem for path rankings based on subjective costs. In Sections 4 and 5, we study the problem of finding a routing tree that minimizes the total subjective cost.

## 2. Subjective-cost model for policy routing

In this section, we present the subjective-cost model of node preferences. The model involves each node  $i$ 's assigning a cost  $c_i(k)$  to every other node  $k$ . These costs are *subjective*, because there is no requirement that  $c_i(\cdot)$  and  $c_k(\cdot)$  be consistent. We assume that each subjective cost  $c_i(k)$  is non-negative. The total cost of  $n$  node  $i$  for a route  $P_{ij}$  to destination  $j$  is

$$c_i(P_{ij}) = \sum_{k \in P_{ij}} c_i(k).$$

Here, the notation  $k \in P_{ij}$  is used to indicate that  $k$  is a *transit node* on the path  $P_{ij}$ ;  $i$  and  $j$  are thus excluded from the summation. Node  $i$  wants to use a route  $P_{ij}$  that minimizes the cost  $c_i(P_{ij})$ .

The subjective-cost model can be used to express a wide range of preferences, but it does place some restrictions on node preferences. For instance, a node  $i$  cannot prefer a path  $P$  over a path  $P'$  whose nodes are a strict subset of  $P$ . The class of preferences that can be expressed as subjective costs includes

- **Lowest-cost routing**  
If  $c_i(k)$  is the actual cost of transiting node  $k$ , minimizing the path cost is exactly lowest-cost routing.
- **Routing with a forbidden set**  
Let  $c_i(\cdot)$  take the following form: If  $k \in S_i$ ,  $c_i(k) = 1$ , else  $c_i(k) = 0$ . Then any route that avoids nodes in  $S_i$  is preferred by  $i$  over any route that involves a node in  $S_i$ .

Subjective costs can form the basis for either ordinal preferences or cardinal utilities. In Section 3, we study the stable-paths problem for path rankings based on subjective costs. In Sections 4 and 5, we study the problem of finding a routing tree that minimizes the total subjective cost.

## 3. Stable paths with subjective costs

The *Stable Paths Problem (SPP)*, introduced by Griffin et al. [6], is defined as follows. We are given a graph with a specified destination node  $j$ . Each other node  $i$  represents a node; there is an edge between two nodes if and only if they exchange routing information with each other. Thus, a path from  $i$  to  $j$  in the graph corresponds to a potential route from node  $i$  to the destination. Each node  $i$  ranks all potential routes to destination  $j$ . A *route assignment* is a specification of a path  $P_{ij}$  for each node  $i$  such that the union of all the routes forms a tree rooted at  $j$  (i.e., the confluence property is satisfied). A route assignment is called *stable* if, for every node  $i$ , the following property holds: for every neighbor  $a$  of  $i$ , node  $i$  does not strictly prefer the path  $aP_{aj}$  over the path  $P_{ij}$ ; in other words,  $i$  would not want to change its current route to any of the other routes currently being advertised by its neighbors. The stable-paths problem is *solvable* if there is a stable route assignment.

Griffin et al. [7,6] have shown that there are instances of SPP that are unsolvable, and, further, that it is NP-complete to determine whether a given SPP is solvable. Their constructions used preferences that cannot be directly expressed

<sup>1</sup> The dependence on  $\epsilon$  is detailed in Section 5.

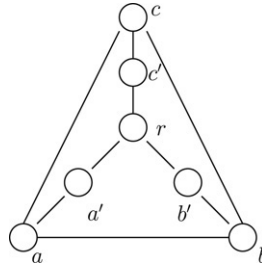


Fig. 1. A bad triangle.

as subjective-cost preferences. This leads us to hope that, for subjective-cost preferences, the stable-paths problem might be tractable. Unfortunately, this is not the case. In this section, we prove that these hardness results extend to subjective-cost preferences.

Assume that the rankings assigned by nodes are based on an underlying subjective-cost assignment. Then, the stable paths problem can be viewed in terms of a strategic game, as follows. The players of this game are the nodes. Given a graph  $G(V, E)$  with a specific destination  $j$  and a subjective-cost function  $c : V(G) \times V(G) \rightarrow \mathcal{R}$ , the *next-hop game* is defined as follows. Nodes correspond to the vertices of graph  $G$ . The strategy space for node  $i$  is the set  $N(i)$  of neighboring nodes in the graph; thus, node  $i$ 's picking the route advertised by a neighboring node  $a$  corresponds to  $i$ 's playing strategy  $a$ . Given a vector of strategies (one for each player), the cost incurred by player  $i$  is the subjective cost of its route to the destination; if there is no route from  $i$  to the destination,  $i$ 's cost is  $\infty$ . A vector of strategies is a pure-strategy Nash equilibrium if, given the strategies of all the other nodes, no node could decrease its subjective cost by changing its strategy. A pure-strategy Nash-equilibrium strategy profile must result in every node having some route to  $j$ , and, hence, it must correspond to a valid route assignment. Thus, proving that an SPP is solvable is equivalent to proving that the corresponding next-hop game has a pure-strategy Nash equilibrium.

**Definition 1.** The *bad triangle* is defined as follows. It is a graph  $G$  with vertex set  $\{a, b, c, a', b', c', r\}$  and edge set  $\{aa', a'r, cc', c'r, bb', b'r, ab, bc, ca\}$ . Set  $c_a(c) = c_a(c') = 0$ ;  $c_b(a) = c_b(a') = 0$ ; and  $c_c(b) = c_c(b') = 0$ . All other subjective costs are set to 1. A bad triangle is shown in Fig. 1. (This construction is based on the *bad gadget* defined in [7].)

In the bad triangle, node  $a$  prefers the path  $(a, c, c', r)$  to the path  $(a, a', r)$ , node  $b$  prefers the path  $(b, a, a', r)$  to the path  $(b, b', r)$ , and node  $c$  prefers the path  $(c, b, b', r)$  to the path  $(c, c', r)$ . It follows from the arguments in [7] that this network is not solvable.

We now show that, as in the case of unrestricted routing policies, it is NP-complete to determine if an SPP based on subjective-cost preferences is solvable.

**Theorem 1.** *Given an instance of the next-hop game, it is NP-complete to decide whether it has a pure Nash equilibrium or not.*

**Proof.** The proof is based on the corresponding NP-completeness proof in [7]. We give a reduction from the 3SAT problem to this problem. In an instance of the 3SAT problem, we are given a set of variables  $\{x_1, \dots, x_n\}$  and a set of clauses  $\{C_1, \dots, C_m\}$ . Each clause  $C_j$  contains three literals  $x_{i1}$ ,  $x_{i2}$ , and  $x_{i3}$ . Given an instance  $(\{x_1, \dots, x_n\}, \{C_1, C_2, \dots, C_m\})$  of the 3SAT problem, we construct the following next-hop game. For each variable  $x_i$  ( $1 \leq i \leq n$ ), we put three vertices  $x_i$ ,  $\bar{x}_i$ , and  $y_i$  in the graph. For each clause  $C_j$  ( $1 \leq j \leq m$ ), we put a bad triangle. We also put a separate vertex  $r$  in the graph. Let the center and three outer vertices of the bad triangle corresponding to the  $j$ th clause be  $v_j$ ,  $v_{j1}$ ,  $v_{j2}$ , and  $v_{j3}$  respectively. The edge set of the graph is  $E(G) = \{y_i x_i, y_i \bar{x}_i, x_i y_{i-1}, \bar{x}_i y_{i-1}, x_1 r, \bar{x}_1 r, v_j r, v_{j1} y_n, v_{j2} y_n, v_{j3} y_n | 1 \leq i \leq n, 1 \leq j \leq m\}$ . The structure of the graph is depicted in Fig. 2. If vertex  $v_{kj}$  corresponds to the literal  $x_i$ , then  $c_{v_{kj}}(\bar{x}_i) = 2$  and  $c_{v_{kj}}(x_i) = 0$ . If it corresponds to  $\bar{x}_i$ , then  $c_{v_{kj}}(x_i) = 2$  and  $c_{v_{kj}}(\bar{x}_i) = 0$ .

Further, we set  $c_{x_i}(y_i) = c_{\bar{x}_i}(y_i) = 1$  and  $c_{y_i}(x_{i+1}) = c_{y_i}(\bar{x}_{i+1}) = 1$ . Finally, we set  $c_{y_n}(c_j) = c_{c_j}(y_n) = 1$  for each clause  $j$ . These assignments are to ensure that, in any Nash equilibrium, all the bad-triangle centers  $v_j$  connect directly to  $r$ , and all the  $x_i$ ,  $\bar{x}_i$ ,  $y_i$  nodes use paths directed towards the left of Fig. 2. Note that, in any pure Nash

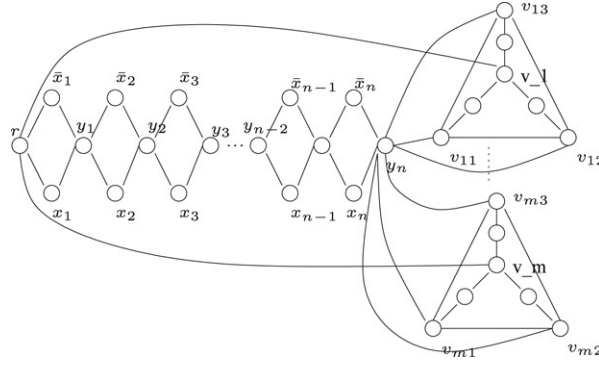


Fig. 2. The reduction from 3SAT.

equilibrium, there is a unique path from  $y_n$  to the root  $r$ . For each  $i$ , this path passes through either  $x_i$  or  $\bar{x}_i$ , indicating an assignment of  $x_i$  in the 3SAT instance.

The subjective costs of vertices inside bad triangles follow the pattern of the bad-triangle example. All other subjective costs are set to zero. Thus, for a set of routes to be stable, for each clause  $j$ , at least one of the vertices  $\{v_{j1}, v_{j2}, v_{j3}\}$  must use the path through  $y_n$  to the root, because there is no stable set of routes through  $v_j$ .

We now show that, if there exists an assignment of variables that satisfies all the clauses, then there exists a pure Nash equilibrium in the next-hop game. We can find this Nash equilibrium by putting a path through the true literals from  $y_n$  to  $r$ . We then assign all nodes  $v_{jk}$  corresponding to a true literal the route through  $y_n$ . This route has zero subjective cost, and so  $v_{jk}$  has no incentive to change routes. Each clause contains at least one true literal, and hence each bad triangle is broken up, and the remaining vertices can be assigned stable paths to  $r$  through the center  $v_j$ .

Conversely, if there is a pure Nash equilibrium in the next-hop game, assign all literals corresponding to the path from  $y_n$  to  $r$  the value true. In this way, for each  $i$ , either  $x_i$  is true or  $\bar{x}_i$  is true, but not both, and hence we have a valid variable assignment. There are no bad triangles in the Nash equilibrium, and so each clause must contain a vertex  $v_{jk}$  that uses a route through  $y_n$  to  $r$ . Note that  $v_{jk}$  always has the option of switching to a path through  $c_j$  with cost 1. The path stability implies that  $v_{jk}$  does not have a subjective cost of 2 for the path through  $y_n$ ; hence, the literal corresponding to  $v_{jk}$  must be assigned true. There is such a vertex for each clause  $j$ , and hence this variable assignment satisfies all the clauses. This shows that there exists a pure Nash equilibrium in the next-hop game if and only if there exists a satisfying assignment in the 3SAT instance.  $\square$

#### 4. The minimum-subjective-cost tree (MSCT) problem

In this section, we assume that the subjective cost  $c_i(k)$  is an actual monetary amount that is measured in the same unit across all nodes. A natural overall goal is then to minimize the sum of subjective costs, i.e., to pick a set of routes  $\{P_{ij}\}$  that minimizes  $\sum_j \sum_i c_i(P_{ij})$ . However, there is a constraint that all the routes to a single destination  $j$  must form a tree, because the packets are actually sent by forwarding. This constraint applies independently to each destination, and so we can consider the simpler problem of routing to a single destination  $j$ .

Thus, we can frame the subjective-cost minimization problem as follows.

**Subjective-cost minimization:** We are given a graph  $G$ , a set of cost functions  $\{c_i(\cdot)\}$ , and a specific destination  $j$ .

We want to find a set of routes  $\{P_{ij}\}$  and payments  $p_i$  to each node  $i$  such that:

- (1) The routes  $\{P_{ij}\}$  form a tree rooted at  $j$ .
- (2) Among all such trees, the selected tree minimizes the sum  $\sum_i \sum_{k \in P_{ij}} c_i(k)$ .

We first prove that, for arbitrary cost functions, the MSCT problem is NP-hard to approximate within any multiplicative factor. Let  $c_{\max} = \max_{v,u \in V(G)} c_v(u)$  and  $c_{\min} = \min_{v,u \in V(G)} c_v(u)$ . Then, we have the following result:

**Theorem 2.** *It is NP-hard to approximate the MSCT problem within a factor better than  $\frac{c_{\max}}{c_{\min} n^2}$ , where  $n$  is the number of vertices. In particular, it is NP-hard to approximate MSCT within any factor if  $c_{\min} = 0$  and  $c_{\max} > 0$ .*



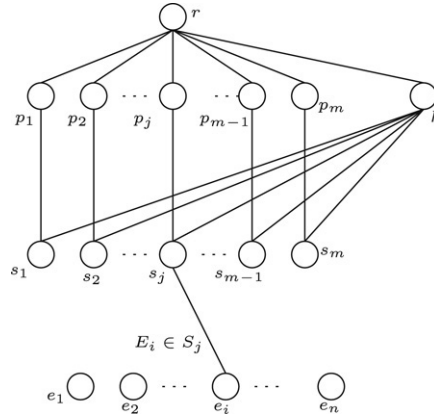


Fig. 3. The reduction from the set-cover problem.

**Proof.** We prove that, if we can approximate MSCT in polynomial time, then we can solve the  $k$ -DISPATH problem in polynomial time. In the  $k$ -DISPATH problem, we are given a graph  $G = (V, E)$  and  $k$  pairs of vertices  $(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)$ , and we want to find  $k$  vertex disjoint paths from  $s_i$  to  $t_i$  for  $1 \leq i \leq k$ . The  $k$ -DISPATH problem is among the ones that Karp originally showed to be NP-hard [8]. From an instance  $\mathcal{I} = (G, \{(s_i, t_i) | 1 \leq i \leq k\})$ , we construct the following instance  $\mathcal{J} = (G', r, \{c_v : v \in V(G')\})$  of the MSCT problem:  $V(G') = V(G) \cup \{r\}$  and  $E(G') = E(G) \cup \{t_i r | 1 \leq i \leq k\}$ . We define  $c_{s_i}(t_j) = c_{\max}$  if  $i \neq j$ . For all the other cases  $c_v(u) = c_{\min}$ . Thus the subjective cost of a tree in which  $t_i$  is not on the path from  $s_i$  to the root is at most  $c_{\min}n^2$ , because each path has at most  $n$  vertices on it. The subjective cost of a tree in which  $s_i$  and  $t_j$  are on the same path in the tree, for  $i \neq j$ , is at least  $c_{\max}$ . It is apparent that there is a tree without a pair  $(s_i, t_j)$  for  $i \neq j$  on a path to the root if and only if  $\mathcal{I}$  is a yes instance of the  $k$ -DISPATH problem. Thus, distinguishing between the case in which the total subjective cost is at most  $c_{\min}n^2$  and the case in which it is at least  $c_{\max}$  is NP-hard. In particular, if  $c_{\min} = 0$  and  $c_{\max} > 0$ , we can use any finite approximation algorithm for MSCT to solve the  $k$ -DISPATH problem.  $\square$

Note that the above theorem does not show hardness for the special cases in which  $\frac{c_{\max}}{c_{\min}}$  is not large. This may be a reasonable restriction; however, we now show that this also yields an intractable optimization problem. In particular, we study the special case in which all subjective costs are either 1 or 2. We call this problem the (1, 2)-MSCT problem. In the following, we give a hardness result for the (1, 2)-MSCT problem.

**Theorem 3.** *The (1, 2)-MSCT problem is APX-hard.*

**Proof.** We first prove that the problem is NP-hard and then modify the reduction to prove APX-hardness. We give a reduction from the set-cover problem to (1, 2)-MSCT. Consider an instance  $\mathcal{I}$  of the set-cover problem with  $n$  elements  $\{E_1, E_2, \dots, E_n\}$  and  $m$  sets  $\{S_1, S_2, \dots, S_m\}$ . We construct an instance  $\mathcal{J} = (G, r, \{c_v : V(G) \rightarrow \{1, 2\} | v \in V(G)\})$  of (1, 2)-MSCT problem as follows. Vertices  $s_j$  and  $p_j$ , for  $1 \leq j \leq m$ , correspond to the set  $S_j$  in  $\mathcal{I}$ . Vertex  $e_i$ , for  $1 \leq i \leq n$ , corresponds to the element  $E_i$  in  $\mathcal{I}$ . There are two other vertices, the root  $r$  and a helper vertex  $h$ . Thus,  $V(G) = \{r, h\} \cup \{s_j, p_j | 1 \leq j \leq m\} \cup \{e_i | 1 \leq i \leq n\}$ . Vertex  $e_i$  is connected to all vertices  $s_j$  such that  $E_i \in S_j$  in  $\mathcal{I}$ . There is an edge between  $s_j$  and  $p_j$ , for all  $1 \leq j \leq m$ , and there is an edge from each  $s_j$  to the helper vertex  $h$ . All vertices  $p_j$ , for  $1 \leq j \leq m$ , and vertex  $h$  are connected to  $r$ . We also set  $c_{e_i}(h) = 2$ , for all  $1 \leq i \leq n$ , and  $c_{s_j}(p_j) = 2$ . For all other  $v$  and  $u$ ,  $c_v(u) = 1$ . Graph  $G$  is depicted in Fig. 3.

We claim that there is a set cover of size  $k$  in  $\mathcal{I}$  if and only if there is tree with total subjective cost  $2n + m + k$  in  $\mathcal{J}$ .

If there is a family of sets  $F$  of size  $k$  that covers all the elements in  $\mathcal{I}$ , then we can construct the following solution in  $\mathcal{J}$ . If  $S_j \in F$  then connect  $s_j$  to  $p_j$  to  $r$ . Each  $e_i$  is connected to some  $S_j \in F$  in the tree; because all elements are covered in the set cover, all vertices  $e_i$  will be included in the tree. For any vertex  $s_j$  such that  $S_j \notin F$ , connect  $s_j$  to  $h$ , and finally, connect  $h$  to  $r$ . It is straightforward to check that the subjective cost of this tree is exactly  $2n + m + k$ .

Conversely, if there is a tree  $T$  of cost  $2n + m + k$  in  $\mathcal{J}$ , then there is a set cover of size  $k$  in  $\mathcal{I}$ . First we can assume that all the edges to  $r$  are in  $T$ , because, if they were not, we could add them and remove an edge from  $T$  to decrease the total subjective cost. Also we can assume that, if there is an edge  $e_i s_j$  in  $T$ , then  $s_j p_j \in E(T)$ , because otherwise

$s_j h$  would be in  $E(T)$ , and we could remove the edge  $s_j h$  and add  $s_j p_j$  to  $T$  to get another tree with lower or equal subjective cost. Knowing these properties of the tree  $T$ , we can easily construct the set cover of size  $k$  from  $T$ . This completes the NP-hardness proof.

For APX-hardness, we need to look at a special case of the set-cover problem. The case in which each element occurs two times and each set is of size at most 3 is APX-complete [9].<sup>2</sup> For this special case of the set-cover problem, we know that  $n \leq m \leq 1.5n$  and that  $k \leq m$ ; thus, there is a constant  $c$  such that  $2n + m + k \leq ck$ . This means that a  $1 + \epsilon$ -approximation for the (1, 2)-MSCT problem gives a  $(1 + \epsilon c)$ -approximation for this special case of set cover.  $\square$

**Theorem 3** shows that, for sufficiently small  $\epsilon$ , it is hard to find a  $(1 + \epsilon)$ -approximation for the (1, 2)-MSCT problem. However, we note that finding a 2-approximation is easy: simply ignore the costs, and construct an unweighted shortest-path tree with destination  $j$ . This is optimal to within a factor of 2, because the number of nodes on the shortest path from  $i$  to  $j$  is a lower bound on the subjective cost  $c_i(P_{ij})$  for any path  $P_{ij}$  from  $i$  to  $j$ .

## 5. An alternative model: Subjective choice of metrics

In this section, we consider a more restricted preference model. We assume that there are multiple objective metrics on routes (e.g., cost and latency), and nodes' preferences differ only in the relative importance they accord to different metrics. This is a non-trivial restriction only when the number of objective metrics is small; here, we first consider the case in which there are only two objective metrics on a route. The results are generalized to  $d > 2$  objective metrics in Section 5.1.

Formally, suppose that any transit node  $k$  has two associated objective “length” values  $l_1(k)$  and  $l_2(k)$ . Both the length values can be extended to additive path metrics, i.e., we can define  $l_1(P_{ij}) = \sum_{k \in P_{ij}} l_1(k)$  and  $l_2(P_{ij}) = \sum_{k \in P_{ij}} l_2(k)$ . Note that we use the term “metric” for the ease of presentation and that we do not impose the triangle equality on the length functions  $l_1$  and  $l_2$ .

Each node  $i$  has a private parameter  $\lambda_i$ ,  $0 \leq \lambda_i \leq 1$ . Node  $i$ 's subjective cost for the route  $P_{ij}$  is given by  $c_i(P_{ij}) = \lambda_i l_1(P_{ij}) + (1 - \lambda_i) l_2(P_{ij})$ , i.e., Node  $i$ 's preferences are modeled as a convex combination of the two path metrics.

It is easy to show that the APX-hardness proof for the (1, 2)-MSCT problem (**Theorem 3**) can be adapted to the two-metric routing problem as well:

**Theorem 4.** *In the subjective-metric model, it is APX-hard to find a tree  $T$  that minimizes total subjective cost.*

**Proof.** In the construction of **Theorem 3**, assign  $l_1(h) = 2, l_2(h) = 1; \forall j \ l_1(p_j) = 1, l_2(p_j) = 2; \forall j \ l_1(s_j) = 2, l_2(s_j) = 1$ . Assign  $\lambda_{e_i} = 1$  for all  $i$ , and  $\lambda_{s_j} = 0$  for all  $j$ . Then, the subjective costs are exactly as specified in the proof of **Theorem 3**, and the APX-hardness follows.  $\square$

We now investigate whether relaxing the confluent-tree routing constraint would lead to stronger results. If we allowed the routes to be completely arbitrary, then clearly we could have optimal routing: each node could simply use the route it liked the best. However, supporting these routes would either require source routing (i.e., the packet header contains a full path) or a massive increase in storage at each router to record the forwarding link for each source and destination. Instead, we ask whether we can get positive results with only a small growth in routers' space requirements.

Our approach is to use a *small number*  $r$  of confluent routing trees  $T_1, T_2, \dots, T_r$  to each destination  $j$ . Then, each node  $i$  evaluates its subjective cost to  $j$  in each of the routing trees and picks a tree  $T_{t_i}$  that minimizes this subjective cost. Node  $i$  then marks each packet it sends with the header  $\langle j, t_i \rangle$ . Each node *en route* stores its route to  $j$  along each tree  $T_j$ ; thus, it can inspect the header of each incoming packet and forward along the appropriate route.

We can prove the following result:

**Theorem 5.** *Suppose that, for transit node  $k$ ,  $l_1(k)$  and  $l_2(k)$  are integers bounded by a polynomial, i.e.,  $l_1(k), l_2(k) < n^c$  for some constant  $c$ . Then, for any given  $\epsilon > 0$ , there is a set of routing trees  $T_1, T_2, \dots, T_r$  with  $r = O(\frac{1}{\epsilon} [\log n + \log(\frac{1}{\epsilon})])$  such that*

<sup>2</sup> In fact, this is the vertex-cover problem in bounded-degree graphs.



For each node  $i$ , there is a tree  $T_i$  such that  $c_i(T_i) \leq (1 + \epsilon)c_i(P_{ij}^*)$ , where  $P_{ij}^*$  is the minimum-subjective-cost route from  $i$  to  $j$ .

Further, this set of trees can be constructed in polynomial time.

**Proof.** Let  $\alpha = (1 + \epsilon)$ . We now show how to construct an appropriate set of trees. Each tree  $T_t$  in our collection is the shortest-path tree for a specific convex combination of the two metrics. We name the trees after the metrics they optimize:

$T_\infty$ :  $l_1(\cdot)$ , with ties broken by minimum  $l_2(\cdot)$ .

$T_{-\infty}$ :  $l_2(\cdot)$ , with ties broken by minimum  $l_1(\cdot)$ .

$T_t$ :  $l_t(\cdot) = \frac{\alpha^t}{1+\alpha^t}l_1(\cdot) + \frac{1}{1+\alpha^t}l_2(\cdot)$  for  $t \in \{-k, -(k-1), \dots, -1, 0, 1, \dots, k\}$ , where  $k = \lceil \log_\alpha(2\epsilon^{-1}n^{c+1}) \rceil$ .

Thus, there are a total of  $r = 2k + 3 = O(\log n)$  trees. These trees can be constructed with  $r$  shortest-path computations and hence can be done in polynomial time. Now, consider a node  $i$ , and let  $\lambda_i \in [0, 1]$  denote its private choice of metric. If  $\lambda_i = 1$ , then clearly  $T_\infty$  will contain the optimal route from  $i$  to  $j$ . Similarly, if  $\lambda_i = 0$ ,  $T_{-\infty}$  will contain the optimal route from  $i$  to  $j$ . Hence, we restrict our attention to the case in which  $\lambda_i \in (0, 1)$ .

Let  $P_{ij}^t$  denote the path from  $i$  to  $j$  in  $T_t$ , and  $P_{ij}^*$  denote the path from  $i$  to  $j$  to which  $i$  assigns minimum subjective cost. We divide all values of  $\lambda \in (0, 1)$  into 4 cases.

**Case (i):**  $\frac{\lambda_i}{1-\lambda_i} > 2\epsilon^{-1}n^{c+1}$

Now, if  $l_1(P_{ij}^*) = 0$ , then  $T_{-\infty}$  must contain  $P_{ij}^*$ , because it is the optimal path under the  $l_2$  metric. If  $l_1(P_{ij}^*) \geq 1$ , we get

$$\begin{aligned} c_i(P_{ij}^\infty) &= \lambda_i l_1(P_{ij}^\infty) + (1 - \lambda_i) l_2(P_{ij}^\infty) \\ &\leq \lambda_i l_1(P_{ij}^*) + (1 - \lambda_i) l_2(P_{ij}^\infty) \\ &\leq \lambda_i l_1(P_{ij}^*) + \epsilon/2 \quad (\text{using the bound on node costs}) \\ &\leq (1 + \epsilon) c_i(P_{ij}^*). \end{aligned}$$

**Case (ii):**  $1 \leq \frac{\lambda_i}{1-\lambda_i} \leq 2\epsilon^{-1}n^{c+1}$

Let

$$t = \left\lfloor \log_\alpha \frac{\lambda_i}{1-\lambda_i} \right\rfloor.$$

Then, we have

$$\alpha^t \leq \frac{\lambda_i}{1-\lambda_i} \leq \alpha^{t+1}.$$

Now, because  $P_{ij}^t$  is the optimal tree under metric  $l_t$ , we have

$$\alpha^t l_1(P_{ij}^t) + l_2(P_{ij}^t) \leq \alpha^t l_1(P_{ij}^*) + l_2(P_{ij}^*).$$

Then,

$$\begin{aligned} \frac{1}{1-\lambda_i} c_i(P_{ij}^t) &= \frac{\lambda_i}{1-\lambda_i} l_1(P_{ij}^t) + l_2(P_{ij}^t) \\ &\leq \alpha^{t+1} l_1(P_{ij}^t) + l_2(P_{ij}^t) \\ &\leq \alpha(\alpha^t l_1(P_{ij}^t) + l_2(P_{ij}^t)) \\ &\leq \alpha(\alpha^t l_1(P_{ij}^*) + l_2(P_{ij}^*)) \\ &\leq \alpha \left( \frac{\lambda_i}{1-\lambda_i} l_1(P_{ij}^*) + l_2(P_{ij}^*) \right) \\ &= \frac{1}{1-\lambda_i} \alpha c_i(P_{ij}^*). \end{aligned}$$

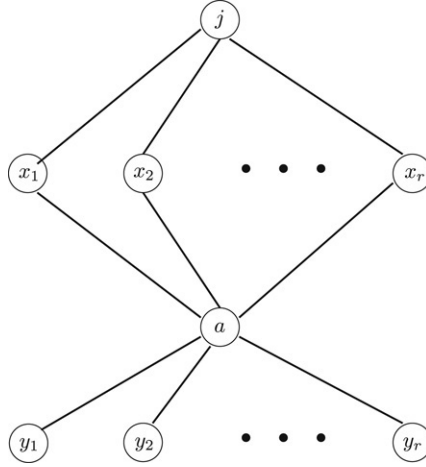


Fig. 4. Network used in Theorem 6.

**Case (iii):**  $1 < \frac{1-\lambda_i}{\lambda_i} \leq 2\epsilon^{-1}n^{c+1}$

The same argument as Case (ii) shows that some  $T_t$  is a good approximation.

**Case (iv):**  $\frac{1-\lambda_i}{\lambda_i} > 2\epsilon^{-1}n^{c+1}$

The same argument as Case (i) shows that either  $T_\infty$  or  $T_{-\infty}$  is a good approximation.

Thus, in each case, one of the routing trees contains a route to  $j$  that has a subjective cost of at most  $(1 + \epsilon)c_i(P_{ij}^*)$ .  $\square$

There are several points worth noting about this scheme. (1) It achieves a result that is slightly stronger than our initial goal — it approximately maximizes each individual node's welfare, not just the sum of all nodes' welfare. (2) The computation of the trees is oblivious to the nodes' preference information. Thus, if we assume that the objective costs are common knowledge (or verifiable), this scheme is trivially a strategy-proof mechanism. (3) Each tree computation involves computing lowest-cost routes for a specific objective metric. Thus, it is easily computed within the framework of the Border Gateway Protocol (BGP). (In the terminology of Feigenbaum et al. [3,4], there is a natural BGP-based distributed algorithm for this scheme.)

We now prove a corresponding lower bound that shows that Theorem 5 is nearly optimal.

**Theorem 6.** Let  $\epsilon > 0$  be given. There is a family of instances of the subjective-metric routing problem, with all weights in  $[0, n^c]$  for some constant  $c$ , such that the following property holds:

Any set of routing trees that contains a  $(1 + \epsilon)$ -approximately optimal path  $P_{ij}$  for each  $i$  must have  $\Omega(\log n/\epsilon)$  trees.

(Here,  $n$  is the number of nodes of the network.)

**Proof.** First, consider the network shown in Fig. 4: The destination  $j$  has  $r$  neighbors  $x_1, x_2, \dots, x_r$ . There is a node  $a$  that is adjacent to each of these  $r$  nodes, and there are  $r$  more nodes  $y_1, \dots, y_r$  that are adjacent to  $a$ .

Let  $\alpha = 1 + \epsilon$ . The transit lengths are set as follows. For each node  $x_k$ , set  $l_1(x_k) = \alpha^{2r-2k}$  and  $l_2(x_k) = \alpha^{2r+2k}$ . For  $a$ , set  $l_1(a) = l_2(a) = 0$ ; the lengths of nodes  $y_i$  are irrelevant, because they cannot be transit nodes. Finally, for each node  $y_i$ , define  $\lambda_{y_i} = (1 - \alpha^{-4i})$ . Hence, for a route  $P$ , we have

$$c_{y_i}(P) = \frac{\alpha^{4i}}{\alpha^{4i} + 1} l_1(P) + \frac{1}{\alpha^{4i} + 1} l_2(P).$$

We now show that, for these costs and preferences, the minimum-subjective-cost route from  $y_i$  to  $j$  is  $ax_i j$ , and that, further, no other path is within a  $(1 + \epsilon)$  factor of optimal.

Let  $P_k$  denote the path  $ax_k j$ . Then,  $y_i$ 's subjective cost for this path is

$$c_{y_i}(P_k) = \lambda_{y_i} l_1(x_i) + (1 - \lambda_{y_i}) l_2(x_k)$$

$$= \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2k} + \frac{1}{\alpha^{4i} + 1} \alpha^{2r+2k}.$$

Thus,

$$c_{y_i}(P_i) = \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2i}.$$

Now, we evaluate  $c_{y_i}(P_k)$  for  $i \neq k$ . Consider two cases:

**Case (i):**  $k > i$

In this case,

$$\begin{aligned} c_{y_i}(P_k) &= \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2k} + \frac{1}{\alpha^{4i} + 1} \alpha^{2r+2k} \\ &= \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2k} \left[ 1 + \alpha^{4(k-i)} \right] \\ &> \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2i} \alpha^{2(k-i)} \\ &> \alpha c_{y_i}(P_i). \end{aligned}$$

**Case (ii):**  $k < i$

In this case, we have

$$\begin{aligned} c_{y_i}(P_k) &= \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2k} + \frac{1}{\alpha^{4i} + 1} \alpha^{2r+2k} \\ &= \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2k} \left[ 1 + \alpha^{4(k-i)} \right] \\ &> \frac{\alpha^{4i}}{\alpha^{4i} + 1} \alpha^{2r-2i} \alpha^{2(i-k)} \\ &> \alpha c_{y_i}(P_i). \end{aligned}$$

Thus, if a solution contains a  $(1+\epsilon)$ -approximately optimal path for every  $y_i$ , it must contain  $P_i$ . In any one routing tree,  $a$  can have only a single route to  $j$ ; thus, at least  $r$  routing trees are required to achieve this.

Finally, observe that the costs are all bounded below by 1 and bounded above by  $\alpha^{4r}$ . For an  $n$ -node graph, we can set  $r = c \frac{\log n}{4 \log \alpha} = O(\frac{\log n}{\epsilon})$  to ensure that all costs are at most  $n^c$ ; then, the graph in Fig. 4 can be embedded into the  $n$ -node graph.  $\square$

### 5.1. Generalization to more than 2 metrics

In this section, we show that Theorems 5 and 6 generalize to the case in which there are  $d > 2$  objective metrics, and a node's subjective cost is a convex combination of these metrics.

**Theorem 7.** Suppose that, for transit node  $k$ , all lengths  $l_1(k), l_2(k), \dots, l_d(k)$  are integers bounded by a polynomial, i.e.,  $l_j(k) < n^c$  for some constant  $c$ . Then, for any given  $\epsilon > 0, \epsilon < 1$ , there is a set of routing trees  $T_1, T_2, \dots, T_r$  with total number of trees  $r = O(d 3^d \lceil \frac{(c+1) \log n + \log(\frac{4}{\epsilon})}{\log(1+\epsilon)} \rceil^{d-1})$  such that:

For each node  $i$ , there is a tree  $T_{i_j}$  such that  $c_i(T_{i_j}) \leq (1+\epsilon)c_i(P_{ij}^*)$ , where  $P_{ij}^*$  is the minimum-subjective-cost route from  $i$  to  $j$ .

Further, this set of trees can be constructed in polynomial time for any constant  $d$ .

**Proof.** The proof, like Theorem 5, uses the idea of rounding the preferences to a bounded grid. The only subtle detail is to handle the case in which  $i$ 's optimal path has zero costs in some metrics. We do this by including, for every subset of metrics, a tree that preferentially selects paths with zero cost on that subset.

Let  $\alpha = \sqrt[3]{1+\epsilon}$ . First, let us define the set of trees constructed. Let  $k = \lceil \log_\alpha(4\epsilon^{-1}n^{c+1}) \rceil$ . Each such tree is indexed by a pair  $(\hat{j}, \vec{h})$ , where  $\hat{j} \in \{1, 2, \dots, d\}$ ,  $\vec{h} \in \{-\infty, -k, -(k-1), \dots, -1, 0, \infty\}^d$ , and  $h_{\hat{j}} = 0$ . The corresponding cost function is defined by

$$c_{(\hat{j}, \vec{h})}(P) = \sum_j \frac{\alpha^{h_j} l_j(P)}{\sum_{a, h_a \neq \infty} \alpha^{i_a}}.$$

For each such cost function, we construct the corresponding minimum-cost tree  $T_{(\hat{j}, \vec{h})}$ . Note that the cost of a path may be infinite; if there is no path from  $i$  that has a finite cost under  $T_{(\hat{j}, \vec{h})}$ , we can include an arbitrary path from  $i$  in this tree.

Note that  $\hat{j}$  does not influence the cost function; it merely indicates one metric that has the maximum weightage under  $c_{(\hat{j}, \vec{h})}$ . It follows that multiple index pairs  $(\hat{j}, \vec{h})$  may correspond to the same cost metric, and hence the same tree. As we are proving an upper bound on the number of trees required, this redundancy is not problematic. For fixed  $\hat{j}$ , there are  $(k+3)^{d-1}$  legal values for  $\vec{h}$ . Thus, the total number of trees constructed is bounded by  $d(k+3)^{d-1} = O(dk^{d-1}) = O(d3^d \lceil \frac{(c+1)\log n + \log(\frac{4}{\epsilon})}{\log(1+\epsilon)} \rceil^{d-1})$ .

We now show that this set of trees contains, for each node  $i$ , a path that approximately minimizes  $i$ 's subjective cost.  $i$ 's subjective cost is a convex combination of the metrics, and hence can be represented as

$$c_i(P) = \sum_j \lambda_{i,j} l_j(P)$$

where the  $\lambda_{i,j}$  values are non-negative and sum to 1. Let  $P^*$  be the path from  $i$  that minimizes  $i$ 's subjective cost. We need to prove that there is a path in one of the constructed trees that approximates the subjective cost (to  $i$ ) of  $P^*$ .

First, consider the case  $c_i(P^*) = 0$ . Consider the tree with index  $(\hat{j}, \vec{h})$  defined as follows:  $h_j = -\infty$  if  $\lambda_{i,j} = 0$ , and  $h_j = 0$  otherwise.  $\hat{j}$  is picked such that  $h_{\hat{j}} = 0$  (there is at least one such  $\hat{j}$ , because not all  $\lambda_{i,j}$  can be zero). It follows that  $c_{(\hat{j}, \vec{h})}(P^*)$  must be 0. The optimal path  $P_{\vec{h}}$  from  $i$  in  $T_{(\hat{j}, \vec{h})}$  must hence have zero cost under  $c_{(\hat{j}, \vec{h})}$ . Therefore,  $l_j(P_{\vec{h}})$  must be zero whenever  $\lambda_{i,j} > 0$ , and so  $P_{\vec{h}}$  must also have zero cost under metric  $c_i$ . Thus, in this case, the constructed paths contain an exactly optimal path for  $i$ .

Now, suppose that  $c_i(P^*) > 0$ . Let  $S = \{j | l_j(P^*) > 0\}$  be the set of base metrics under which  $P^*$  has positive cost. Now, pick  $\hat{j}$  to maximize  $\lambda_{i,j}$  within  $S$ :

$$\hat{j} = \operatorname{argmax}_{j \in S} \lambda_{i,j}.$$

Note that  $\lambda_{i,\hat{j}} > 0$ , because  $c_i(P^*) > 0$ . Let  $i_j = \log_\alpha \frac{\lambda_{i,j}}{\lambda_{i,\hat{j}}}$ ; then,  $i_j \in [-\infty, 0]$ .

We now select a vector  $\vec{h}$  as follows:

$$\begin{aligned} \forall j \notin S, & \quad h_j = \infty \\ \forall j \in S, i_j \geq -k, & \quad h_j = \lfloor i_j \rfloor \\ \forall j \in S, i_j < -k, & \quad h_j = -\infty. \end{aligned}$$

Let  $P_{\vec{h}}$  be the optimal path from  $i$  to  $j$  in  $T_{(\hat{j}, \vec{h})}$ . We now show that it is nearly optimal in terms of  $i$ 's subjective cost.

First, observe that  $P^*$  has finite cost under  $c_{(\hat{j}, \vec{h})}$ , because, by construction,  $l_j(P^*) = 0$  whenever  $h_j = \infty$ . Next, observe that we do not alter the cost much by rounding all sufficiently small  $i_j$  to  $-\infty$ , because  $l_{\hat{j}}(P^*) \geq 1$ :

$$\sum_{j | i_j < -k} \alpha^{i_j} l_j(P^*) \leq d\alpha^{-k} n^{c+1} \leq (\epsilon/4) \leq (\epsilon/4) l_{\hat{j}}(P^*).$$

(Here, we have used the fact that the maximum cost of a path, in any base metric, is at most  $n \cdot n^c = n^{c+1}$ .)

This gives us a bound:

$$\begin{aligned} \sum_j \alpha^{i_j} l_j(P^*) & \leq (1 + \epsilon/4) \sum_{j | i_j \geq -k} \alpha^{i_j} l_j(P^*) \\ \sum_j \alpha^{i_j} l_j(P^*) & \leq \alpha(1 + \epsilon/4) \sum_{j | i_j \geq -k} \alpha^{h_j} l_j(P^*) \quad (\text{because } \alpha^{i_k} \leq \alpha \cdot \alpha^{h_k}). \end{aligned}$$

Now, we can bound the cost of path  $P^*$  in the cost metric  $c_{(\hat{j}, \vec{h})}$ .

$$\begin{aligned} c_{(\hat{j}, \vec{h})}(P^*) &= \frac{\sum_j \alpha^{h_j} l_j(P^*)}{\sum_{a, h_a \neq \infty} \alpha^{h_a}} \\ &\geq \frac{\sum_j \alpha^{h_j} l_j(P^*)}{\sum_a \alpha^{i_a}} \quad (\text{increasing the denominator}) \\ &\geq \frac{1}{\alpha(1 + \epsilon/4)} \frac{\sum_j \alpha^{i_j} l_j(P^*)}{\sum_a \alpha^{i_a}} \\ &\geq \frac{1}{\alpha(1 + \epsilon/4)} c_i(P^*). \end{aligned}$$

Note that  $c_{(\hat{j}, \vec{h})}(P_{\vec{h}}) \leq c_{(\hat{j}, \vec{h})}(P^*)$ , because the tree  $T_{(\hat{j}, \vec{h})}$  is a lowest-cost tree. Further,  $P_{\vec{h}}$  must have finite cost in this metric, and hence  $l_j(P_{\vec{h}}) = 0$  whenever  $l_j(P^*) = 0$ . Thus, we get

$$\begin{aligned} c_i(P_{\vec{h}}) &= \frac{\sum_j \alpha^{i_j} l_j(P_{\vec{h}})}{\sum_a \alpha^{i_a}} \\ &\leq \frac{\sum_j \alpha^{i_j} l_j(P_{\vec{h}})}{\sum_{a, h_a \neq \infty} \alpha^{i_a}} \quad (\text{reducing the denominator}) \\ &\leq \frac{\sum_j \alpha^{i_j} l_j(P_{\vec{h}})}{\sum_{a, h_a \neq \infty} \alpha^{h_a}} \quad (\text{reducing the denominator}) \\ &\leq \alpha \frac{\sum_j \alpha^{h_j} l_j(P_{\vec{h}})}{\sum_{a, h_a \neq \infty} \alpha^{h_a}} \quad (\text{using } \alpha^{i_j} < \alpha \cdot \alpha^{h_j}) \\ &\leq \alpha c_{(\hat{j}, \vec{h})} P_{\vec{h}}. \end{aligned}$$

Combining, we get that  $c_i(P_{\vec{h}}) \leq \alpha^2(1 + \epsilon/4)c_i(P^*)$ . For  $\epsilon < 1$ , we have  $(1 + \epsilon/4) \leq \sqrt[3]{1 + \epsilon}$ , so  $c_i(P_{\vec{h}}) \leq (1 + \epsilon)c_i(P^*)$ . Thus,  $P_{\vec{h}}$  is an approximately optimal path for  $i$ .  $\square$

**Theorem 8.** Let  $\epsilon > 0$  be given and  $d > 2$  be given. There is a family of instances of the subjective-metric routing problem, with all weights in  $[0, n^c]$  for some constant  $c$ , such that the following property holds:

Any set of routing trees that contains a  $(1 + \epsilon)$ -approximately optimal path for each  $i$  must have  $\Omega((\frac{\log n}{d \log d + d \log(1 + \epsilon)})^{d-1})$  trees.

(Here,  $n$  is the number of nodes of the network.)

**Proof.** Consider the network shown in Fig. 5. The destination  $w$  has  $m$  neighbors  $x_1, x_2, \dots, x_m$ . There is a node  $w'$  that is adjacent to each of these  $m$  nodes, and there are  $n$  more nodes  $y_1, \dots, y_n$  that are adjacent to  $w'$ . Consider all integer vectors  $\vec{i} = (i_1, i_2, \dots, i_{d-1}, 1) \in [1, r]^d$ . There are  $r^{d-1}$  such vectors. We set  $m = r^{d-1}$  and correspond each vector  $\vec{i}$  to one of the vertices  $x_j$ . Thus, we denote by  $x_{\vec{i}}$  the vertex  $x_j$  corresponding to vector  $\vec{i}$ .

Let  $\alpha = 1 + \epsilon$ . Now we define the transit lengths as follows. For each node  $x_{\vec{i}}$  with  $\vec{i} = (i_1, \dots, i_{d-1}, 1)$ , we set  $l_j(x_{\vec{i}}) = t^{i_j + r - a(\vec{i})}$  for  $1 \leq j \leq d$  where  $a(\vec{i}) = \frac{\sum_{j=1}^{d-1} i_j}{d}$  and  $t$  is a constant that is fixed later. For  $w'$ , set  $l_j(w') = 0$  for all  $1 \leq j \leq d$ ; the lengths of nodes  $y_{\vec{i}}$  are irrelevant, because they cannot be transit nodes. Finally, for  $m$  nodes  $y_{\vec{i}}$ , define  $\lambda_{y_{\vec{i}}, j} = t^{-i_j}$  for  $1 \leq j \leq d-1$  and  $\lambda_{y_{\vec{i}}, d} = 1$ . For the other  $n - m$  vertices,  $\lambda_{y_k, j} = \frac{1}{d}$  for  $1 \leq j \leq d$  for  $m < k \leq n$ .

Let  $P_{\vec{i}}$  be the path  $(w', x_{\vec{i}}, w)$ . For a route  $P_{\vec{i}}$ , we have

$$c_{y_{\vec{i}}}(P_{\vec{i}}) = dt^{r - a(\vec{i})}.$$

We now show that, for these costs and preferences, and by setting  $t$  such that  $\log t \geq d(\log d + \log \alpha)$  the minimum-subjective-cost route from  $y_{\vec{i}}$  to  $j$  via  $P_{\vec{i}}$ , and that, further, no other path is within a  $(1 + \epsilon)$  factor of optimal.  $y_{\vec{i}}$ 's

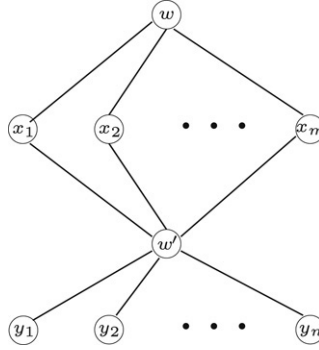


Fig. 5. Network used in Theorem 8.

subjective cost for path  $P_{\vec{i}}$ , we claim that

$$\begin{aligned} c_{y_{\vec{i}}}(P_{\vec{i}}) &\geq \max_{1 \leq j \leq d} t^{i'_j - i_j} \cdot t^{r - a(\vec{i})} \\ &\geq \alpha d t^{r - a(\vec{i})} \\ &\geq \alpha c_{y_{\vec{i}}}(P_{\vec{i}}). \end{aligned}$$

To see the second inequality, we need to show that for a number  $j$  ( $1 \leq j \leq d$ ),  $t^{i'_j - i_j + a(\vec{i}) - a(\vec{i}')} \geq \alpha d$ . Assuming  $\vec{i} \neq \vec{i}'$ , there exists an index  $j$  such that  $i'_j - i_j + a(\vec{i}) - a(\vec{i}') \geq \frac{1}{d}$ . Thus, in order to prove the claim it is sufficient to prove that  $t^{\frac{1}{d}} \geq \alpha d$ , but this is an immediate consequence of  $\log t \geq d \log \alpha + d \log d$ .

This shows that if a solution contains a  $(1 + \epsilon)$ -approximately optimal path for every  $y_{\vec{i}}$ , it must contain  $P_{\vec{i}}$ . In any one routing tree,  $w'$  can have only a single route to  $w$ ; thus, at least  $m = r^{d-1}$  routing trees are required to achieve this.

Now, if there is an upper bound of  $n^c$  on the subjective costs, we can set  $t^r = \Theta(n^c)$ . By setting  $\log t = \Theta(d \log d + d \log \alpha)$ ,  $r = \Theta(\frac{c \log n}{\log t}) = \Theta(\frac{c \log n}{d \log d + d \log \alpha})$ . Therefore, we need  $m = r^{d-1} = \Theta([\frac{c \log n}{d \log d + d \log \alpha}]^{d-1})$  routing trees.  $\square$

## 6. Conclusion

In this paper, we have studied classes of ordinal and cardinal preferences based on subjective costs. The subjective-cost preference model is intuitively appealing, and it is very expressive. However, our results show that, even if the costs are restricted to a very small range, unstructured subjectivity leads to intractable problems in both models: NP-completeness of the stable paths problem for ordinal preferences and APX-hardness of the minimum-subjective-cost tree problem for cardinal preferences.

The root cause of these hardness results appears to be the high dimension of the space of node preferences. Thus, it is necessary to work with models that provide a more consistent global structure. In Section 5, we consider the case in which there are two objective cost metrics, and nodes differ in the relative importance they place on the first metric. For example, nodes may agree on the latency and packet-loss rate of each node in the network but have subjective opinions about the relative importance of latency and loss rate. Thus, in this model, the space of all node types is one-dimensional. We showed that it is possible to select a small number ( $O(\frac{1}{\epsilon} [\log n + \log(\frac{1}{\epsilon})])$ ) for a  $(1 + \epsilon)$ -approximation of representative types such that every nodes' preferences are closely approximated by one of the representatives; then, by picking a set of routing trees, each of which is optimized for a specific representative type, we can guarantee each node a route that  $(1 + \epsilon)$ -approximately minimizes its subjective cost. Further, this scheme is easy to implement, even in the distributed-computing context: each destination can be replaced by a small number of logical destinations, and a lowest-cost routing algorithm (e.g., the Bellman–Ford algorithm) can be used for each logical destination.

It is also possible that other models that restrict the subjectivity of the costs in some way may yield positive results. For example, the nodes' subjective costs for a given transit node  $k$  are random variables drawn from a specific distribution. Finding such models that are both realistic and tractable is an interesting avenue for future research.



## Acknowledgements

We thank Shay Kutten for suggesting the forbidden-set routing problem and Hari Balakrishnan for pointers to the literature. We are grateful to Andrew Twigg and Vijay Ramachandran for their comments on an earlier version of this paper.

Joan Feigenbaum was supported in part by ONR grants N00014-01-1-0795 and N00014-04-1-0725 and NSF grants ANI-0207399 and CNF-0428422. David R. Karger and Vahab S. Mirrokni were supported in part by NSF grant ANI-0225660. Rahul Sami was supported in part by NSF grant ITR-0219018. Most of this work was done while he was a postdoctoral associate at MIT. Vahab Mirrokni is currently at Microsoft Research.

## References

- [1] N. Feamster, R. Johari, H. Balakrishnan, The implications of autonomy for stable policy routing, in: *Proceedings of the ACM SIGCOMM '05*, 2005, pp. 25–36.
- [2] J. Feigenbaum, D. Karger, V. Mirrokni, R. Sami, Subjective-cost policy routing, in: *Proceedings of the Workshop on Internet and Network Economics*, in: *Lecture Notes in Computer Science*, vol. 3828, 2005, pp. 174–183.
- [3] J. Feigenbaum, C. Papadimitriou, R. Sami, S. Shenker, A BGP-based mechanism for lowest-cost routing, *Distributed Computing* 18 (1) (2005) 61–72.
- [4] J. Feigenbaum, R. Sami, S. Shenker, Mechanism design for policy routing, *Distributed Computing* 18 (4) (2006) 295–306.
- [5] L. Gao, J. Rexford, Stable internet routing without global coordination, *IEEE/ACM Transactions on Networking* 9 (6) (2001) 681–692.
- [6] T. Griffin, B. Shepherd, G. Wilfong, The stable paths problem and interdomain routing, *IEEE/ACM Transactions on Networking* 10 (2) (2002) 232–243.
- [7] T. Griffin, G. Wilfong, An analysis of BGP convergence properties, in: *Proceedings of ACM SIGCOMM '99*, 1999, pp. 277–288.
- [8] R. Karp, Reducibility among combinatorial problems, *Complexity of Computer Computations* (1972) 85–103.
- [9] C. Papadimitriou, M. Yannakakis, Optimization, approximation, and complexity classes, *Journal Computer and System Sciences* 43 (1991) 425–440.
- [10] K. Varadhan, R. Govindan, D. Estrin, Persistent route oscillations in inter-domain routing, Technical Report 96-631, USC/ISI, February 1996.