Volume: 03, Issue: 04, Pages:1292-1297 (2012)

Multipath Routing Algorithms for Congestion Minimization

B. Sasthiri
Department of Computer Science and Engineering, CSI College of Engineering, Ketti, The Nilgiris, 643215.
Email: sasthirib@yahoo.co.in
T. Prakash
Department of Mathematics, CSI College of Engineering, Ketti, The Nilgiris, 643215.
Email: prakashthonan@gmail.com

ABSTRACT

Unlike traditional routing schemes that route all traffic along a single path, multipath routing strategies split the traffic among several paths in order to ease congestion. It has been widely recognized that multipath routing can be fundamentally more efficient than the traditional approach of routing along single paths. Yet, in contrast to the single-path routing approach, most studies in the context of multipath routing focused on heuristic methods. We demonstrate the significant advantage of optimal (or near optimal) solutions. Hence, we investigate multipath routing adopting a rigorous (theoretical) approach. We formalize problems that incorporate two major requirements of multipath routing. Then, we establish the intractability of these problems in terms of computational complexity. Finally, we establish efficient solutions with proven performance guarantees.

Keywords : Congestion Avoidance, Multi path routing, NP-hard, Quality of Service, Routing protocols.

Date of Submission: June 08, 2011 Date of Acceptance: September 23, 2011

I. INTRODUCTION

Current routing schemes typically focus on discovering a single "optimal" path for routing, according to some desired metric. Accordingly, traffic is always routed over a single path, which often results in substantial waste of network resources. Multipath Routing is an alternative approach that distributes the traffic among several "good" paths instead of routing all traffic along a single "best" path.

Multipath routing can be fundamentally more efficient than the currently used single path routing protocols. It can significantly reduce congestion in “hot spots,” by deviating traffic to unused network resources, thus, improving network utilization and providing load balancing [16]. Moreover, congested links usually result in poor performance and high variance. For such circumstances, multipath routing can offer steady and smooth data streams [6].

Multipath routing algorithms that optimally split traffic between a given set of paths have been investigated in the context of flow control (e.g.., [14], [19], [20]). Yet, the selection of the routing paths is another major design consideration that has a drastic effect on the resulting performance. Therefore, although many flow control algorithms are optimal for a given set of routing paths, their performance can significantly differ for different sets of paths. Accordingly, in this paper, we focus on multipath routing algorithms that both select the routing paths and split traffic among them.

Accordingly, in this study we investigate multipath routing adopting a rigorous approach, and formulate it as an optimization problem of minimizing network congestion. Under this framework, we consider two fundamental requirements. First, each of the chosen paths should usually be of satisfactory "quality". Indeed, while better load balancing is achieved by allowing the employment of paths other than shortest, paths that are substantially inferior (i.e., "longer") may be prohibited.

Therefore, we consider the problem of congestion minimization through multipath routing subject to a restriction on the "quality" (i.e., length) of the chosen paths.

Another practical restriction is on the number of routing paths per destination, which is due to several reasons [23]. First, establishing, maintaining and tearing down paths pose considerable overhead; second, the complexity of a scheme that distributes traffic among multiple paths considerably increases with the number of paths; third, often there is a limit on the number of explicitly routing paths (such as label-switched paths in MPLS [26]) that can be set up between a pair of nodes. Therefore, in practice, it is desirable to use as few paths as

Volume: 03, Issue: 04, Pages:1292-1297 (2012)

possible while at the same time minimize the network congestion.

II. MODEL AND PROBLEM FORMULATION.

a) ALGORITHMS IN RMP FOR MINIMIZING NETWORK CONGESTION UNDER PATH QUALITY CONSTRAINTS

Solution of problem RMP:

In this section we aim at solving problem RMP, i.e., the problem of minimizing congestion subject to additive QoS requirements. In addition, we present an important application that supports end-to-end reliability requirements. First we establish that the problem is intractable.

A. Intractability of Problem RMP.

We show that Problem RMP can be reduced to the Partition problem [12].

Theorem: Problem RMP is NP-hard.

Suppose there is a path flow that transfers two flow units over paths that are not bigger than L. It is easy to see that all paths in the graph must be simple since the graph is a DAG. Select one path that transfers a positive flow and denote it as p. Define an empty set S. For every link in p, with weight s (a_i), insert the element a_i into S. Since all links in the graph have one unit of capacity, the selected path p is not able to transfer more than one unit of flow. Now, delete all the links that constitute path p. Since p is simple and since it transfers at most one unit of flow, there must be another path that is disjoint to the selected path that transfers a positive flow over the links that were left in the graph. For each link in that path with size s (a_i), insert the element a_i into a different set S'.

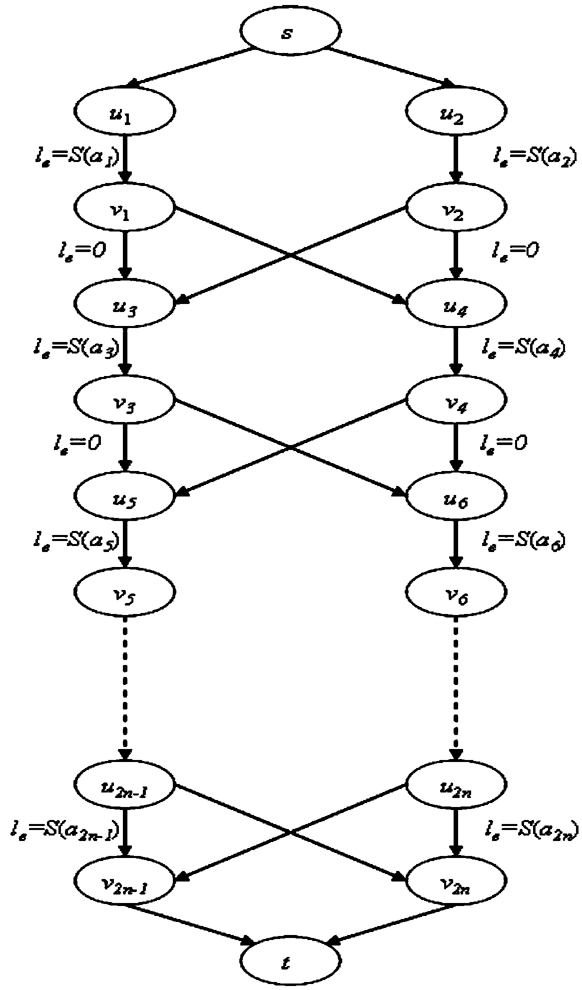

Fig. 1.1 Reduction of Partition to RMP

B. PSEUDO-POLYNOMIAL ALGORITHM

The first step towards obtaining a solution to Problem RMP is to define it as a linear program. To that end, we need some additional notation.

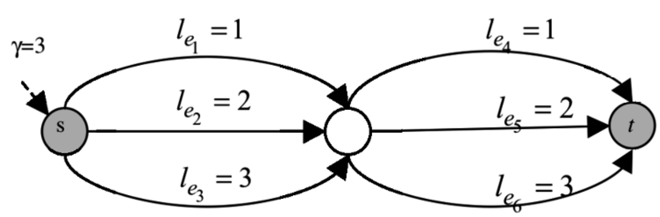

Fig. 1.2 Single link flow Single link flow can be decomposed into several path flows. Some of them satisfy the length restriction and the rest violate it.

We can solve Program RMP as shown in fig 1.1 using any polynomial time algorithm for linear programming [18]. The solution to the problem is then achieved by decomposing the output of Program RMP (i.e., link flow techniques that transforms flows along paths as shown [1]), cannot be used for our purpose since they do not respect the length restrictions.

Algorithm PFC $\left(G(V,E),\{s,t\},\{f_e^{\lambda}\},\gamma\right)$

A. Initialization:

For each path $p \in P^{(s,t)}$: $g_p \leftarrow 0$.

B. While $\gamma > 0$ do:

1. $\mathcal{P} \leftarrow$ Path_Construction $\left(G(V,E),\{s,t\},\{f_e^{\lambda}\}\right)$
2. $f_{e_i}^{\Lambda_i} \leftarrow f_{e_i}^{\Lambda_i} - \min_{(e_i,\Lambda_i)\in\mathcal{P}} \{f_{e_i}^{\Lambda_i}\}$ for each $(e_i,\Lambda_i) \in \mathcal{P}$.
3. $\gamma \leftarrow \gamma - \min_{(e_i,\Lambda_i)\in\mathcal{P}} \{f_{e_i}^{\Lambda_i}\}$.
4. Let p be the path that corresponds to $\mathcal{P}$ i.e., $e_i \in p$ iff $(e_i,\Lambda_i) \in \mathcal{P}$. Perform $g_p \leftarrow \min_{(e_i,\Lambda_i)\in\mathcal{P}} \{f_{e_i}^{\Lambda_i}\}$.

C. Convert each non-simple path that carries a positive flow in g into a simple path as follows:

a. Let $h: P^{(s,t)} \rightarrow P_{simple}^{(s,t)}$ map each path $p = (v_0, v_1, \cdots v_{h-1}, v_h, \cdots, v_h, v_{h+1}, \cdots v_n) \in P^{(s,t)}$ into the corresponding simple path $p' = (v_0, v_1, \cdots v_{h-1}, v_h, v_{h+1}, \cdots v_n) \in P_{simple}^{(s,t)}$.

b. For each $p \in P_{simple}^{(s,t)}$, $f_p \triangleq \sum_{\hat{p}|h(\hat{p})=p} g_{\hat{p}}$.

D. Return path flow f.

Fig.1.3 Algorithm PFC

Procedure Path_Construction $\left(G(V,E),\{s,t\},\{f_e^{\lambda}\}\right)$

Initialization

$\mathcal{P} \leftarrow \phi$, $u_0 \leftarrow s$, $\Lambda_0 \leftarrow 0$, $i \leftarrow 0$

While $u_i \neq t$ do

1. Select a positive variable $f_{e_i}^{\Lambda_i}$ such that $e_i \triangleq (u_i, u_{i+1}) \in E$.
2. $\mathcal{P} \leftarrow \mathcal{P} \cup \{(e_i,\Lambda_i)\}$, $\Lambda_{i+1} \leftarrow \Lambda_i + l_{e_i}$, $i \leftarrow i+1$.

Return $\mathcal{P}$

Fig.1.4 Procedure Path Construction

The linear program can be solved within time complexity that is polynomial in the number of variables. Therefore, the complexity incurred by solving the linear program is polynomial in L [18], [12].

Algorithm RMP $\left(G(V,E),\{s,t\},\{l_e\},\{c_e\},\gamma,L\right)$

1. $\{f_e^{\lambda}\} \leftarrow$ **Program RMP** $\left(G(V,E),\{s,t\},\{l_e\},\{c_e\},\gamma,L\right)$
2. $f \leftarrow$ **Algorithm PFC** $\left(G(V,E),\{s,t\},\{f_e^{\lambda}\},\gamma\right)$
3. **Return** path flow f.

Fig.1.5 Algorithm RMP

C. ε-OPTIMAL APPROXIMATION SCHEME FOR PROBLEM RMP

In Section B, we established an optimal polynomial solution to Problem RMP for the case where the length restrictions are sufficiently small. In this section, we turn to consider the solution to Problem RMP for arbitrary length restrictions. We focus on the design of an efficient algorithm that approximates the optimal solution.

Our main result is the establishment of an optimal approximation scheme, which is termed the RMP approximation scheme. This scheme is based on Algorithm RMP, specified in section B, the RMP approximation scheme is specified in Fig. 1.6.

RMP Approximation Scheme $\left(G,\{s,t\},\{c_e\},\{l_e\},\gamma,L,\varepsilon\right)$

2. $\Delta \leftarrow \frac{L \cdot \varepsilon}{N}$.
3. Let $\left\langle G(V,E),\{s,t\},\{\tilde{c}_e\},\{\tilde{l}_e\},\gamma,\tilde{L}\right\rangle$ be an instance of Problem RMP such that:
   a. $\tilde{L} \leftarrow \left\lceil \frac{L}{\Delta} \right\rceil$
   b. For each link $e \in E$:
   $\tilde{l}_e \leftarrow \left\lfloor \frac{l_e}{\Delta} \right\rfloor$, $\tilde{c}_e \leftarrow c_e$.
4. Invoke *Algorithm RMP* over the instance $\left\langle G(V,E),\{s,t\},\{\tilde{c}_e\},\{\tilde{l}_e\},\gamma,\tilde{L}\right\rangle$ of Problem RMP. Let path flow f represent the output.
5. Return Path flow f.

Fig. 1.6 RMP approximation scheme

Volume: 03, Issue: 04, Pages:1292-1297 (2012)

II.b) ALGORITHMS IN KPR FOR MINIMIZING CONGESTION WHILE ROUTING ALONG AT MOST K DIFFERENT PATHS

Theorem: The minimum congestion of a γ/K-integral flow is at most twice the congestion of the optimal solution.

- γ/K- integral flows that minimize congestion
- An optimal γ/K- integral flow is a 2-APX scheme.
- Computing optimal γ/K- integral flows.

Each γ/K- integral flow satisfies the requirement to ship the demand γ on at most K paths.

Corollary: minimizing the congestion while restricting the flow to be integral in γ/K is a 2-approximation scheme for the original problem

COMPUTING OPTIMAL γ/K-INTEGRAL FLOWS

The network congestion factor of **each** γ/K-integral flow belongs to $\{n.\gamma/K.c_e | e\in E,\ n \in [0, K]\}$

.

- The flow over each link is integral in γ/K and is at mostγ.
- Hence, for each e™E it holds that $f_e \in \{n.\gamma/K. | n\in [0,K]\}$
- Thus, for each e™E it holds that

$f_{e/}c_e \in \{n.\gamma/K.c_e | n\in [0,K]\}$

- In particular,

Max $\{f_{e/}c_e\} \in \{n.\gamma/K.c_e | e\in E, n\in [0,K]\}$

Procedure Test $(G,\{s,t\},\{c_e\},\gamma,K,\alpha)$

1. **Multiply all link capacities by α and round down each capacity to the nearest multiple of $\frac{\gamma}{K}$ i.e., set $\tilde{c_e} \leftarrow \frac{\gamma}{K}\cdot\left\lfloor\frac{\alpha\cdot c_e}{\gamma/K}\right\rfloor$ for each $e\in E$.**
2. **Solve the instance $\langle G,(s,t),\{\tilde{c_e}\}\rangle$ of the Maximum Flow Problem using the *Push Relabel Algorithm* [13]. Let the link flow $\{f_e\}$ represent the solution, and let F be the total transferred flow from s to t.**
3. **If $F \geq \gamma$**
   Return the link flow $\{f_e\}$.
   Else
   Return Fail.

Fig 2.1 Procedure Test

In this section, we investigate Problem KPR, which minimizes congestion while routing traffic along at most different paths. we prove that Problem KPR is NP-hard in the general case but admits a (straightforward) polynomial solution when the restriction on the number of paths is larger than the number of links K>M.

A. Round down the capacity of each link to a multiply of γ/K.
   - Since the flow must be γ/K-integral, such a rounding has no affect.

B. Apply a maximum flow algorithm.
   - Since all capacities are integral in γ/K, the algorithm returns a γ/K-integral flow.

C. If the γ/K-integral flow fails to transfer γ flow units repeat the process with a larger; otherwise repeat the process with a smaller.

D. Output the flow that transfers γ flow units and has the smallest m.

Since the set *A* is polynomial the complexity of the solution is polynomial. Thus, we established a polynomial algorithm that admits at most K paths and has a network congestion factor that is at most twice larger than the optimum

1. A special case of our problem: Is there a path flow that transfers γ flow units from s to t such that if path p transfers a positive amount of flow then D (p) ≤D?
2. The partition problem: Given an ordered set of elements $a_1, a_2, \ldots, a_{2n}$ that constitute a set A with a size $s(a)^{TM}\mathbb{Z}^+$ for each a ^{TM}A, is there a subset A⊆A such that A′contains exactly one element of a_{2i-1}, a_{2i} for 1≤i≤n such that $\sum_{a™A′}s(a)=\sum_{a™A\backslash A′}s(a)$?
3. All link capacities are 1.
4. Claim: It is possible to transfer 2 flow units over paths whose end-to-end delays are not larger than $½\sum_{a™A}$ s (a) iff there is a subset A′⊆A such that A′ contains exactly one element of a_{2i-1}, a_{2i} for 1≤i≤n and $\sum_{a™A′}$ s (a) $=\sum_{a™A\backslash A′}$s (a).

There is a subset A′⊆A such that A′ contains exactly one element of a_{2i-1}, a_{2i} for 1≤i≤n and $\sum_{a™A′}$s (a) $=\sum_{a™A\backslash A′}$ s (a).

Volume: 03, Issue: 04, Pages:1292-1297 (2012)

The selection of the links that correspond to the elements of A′ and the zero delay links that connect these links constitutes a path *p*.
Path p is disjoint to the path that the complement subset A\A′ defines.
Since all capacities are equal to 1, we have two disjoint paths that can transfer together 2 units of flow.
The end-to-end delay of each path is $½\sum_{a™A}$ s(a).

- There is a path flow that transfers two flow units over paths that are not larger than $½\sum_{a™A}$ s (a).
- Let *p* be a path that carries a positive flow; by construction, p contains exactly one element of a_{2i-1}, a_{2i} for $1 \le i \le n$.
- Since all the links have one unit of capacity *p* can transfer at most 1 flow unit.
- Therefore, there exists a path p′ that is disjoint to *p* that transfers a positive flow; by construction, p′=A\p
- Hence, D (p) $\le ½\sum_{a™A}$ s (a) and D (p′) $\le ½\sum_{a™A}$ s (a).
- Therefore, since D(p)+ D(p′)=$\sum_{a™A}$ s(a) it follows that $\sum_{a™p}$ s(a)=$\sum_{a™p′}$s(a)=$½\sum_{a™A}$ s(a).

III. ADVANTAGES

Multipath routing can be fundamentally more efficient than the currently used single path routing protocols. It can significantly reduce congestion in "hot spots," by deviating traffic to unused network resources, thus, improving network utilization and providing load balancing. Moreover, congested links usually result in poor performance and high variance. For such circumstances, multipath routing can offer steady and smooth data streams.

Applications for Program RMP

Problem RMP may arise in several forms. In the single-commodity case, it adds an additive restriction to the well-known Maximum Flow Problem, which applies to paths that carry a positive flow. This restriction may be important in multipath routing schemes where additive QoS metrics, such as delay and jitter, are considered. In this section, we show that Program RMP can be used in order to support multipath routing with end-to-end reliability requirements, i.e., when we need multipath routing schemes that choose paths with a "good" probability of success.

The notion of reliability can be implemented by assigning to each link in the network a failure probability and restricting all paths that carry positive flow to have an end-to-end success probability that is larger than some given lower bound. This is formulated by the following problem.

IV. IMPLEMENTATION AND RESULTS

This implementation and results chapter gives the overview about how this Proposed Project System has been implemented with all the above mentioned software and hardware facilities and the corresponding Screenshots and the Performance Measures have been taken for the particular application.

V. OVERVIEW

During this work we observed that multipath routing offers many advantages in contexts that are not necessarily related to congestion avoidance or load balancing. In the following we present a brief description of this research.

Multipath routing and survivability

Multipath routing can be used in order to improve resilience and avoid congestion. The combination of both benefits can be obtained by employing the idea of diversity coding, which adds redundant information to the data stream, like error detection and correction codes. Then, in order to increase fault tolerance, the redundant information is routed along paths that are disjoint to the paths that are used to transfer the original data stream. Therefore, it is desired to develop new multipath routing schemes that also engage the diversity coding concept. For example, it is desired to develop schemes for multipath routing that maximize the total flow (or minimize the congestion) and satisfy a fundamental property that restricts each path that transfer positive data flow to have an adequate set of disjoint paths with enough bandwidth to protect this flow.

VI. CONCLUSION

Previous multipath routing schemes for congestion avoidance focused on heuristic methods. Yet, our simulations indicate that optimal congestion reduction schemes are significantly more efficient. Accordingly, we investigated multipath routing as an optimization problem of minimizing network congestion and considered two fundamental problems. Although both have been shown to be computationally intractable, they have been found to admit efficient approximation schemes. Indeed, for each problem, we have established a polynomial time algorithm that approximates the optimal solution by a (small) constant approximation factor.

A common feature that both approximations share is the discretization of the set of feasible solutions. Whereas the solution to Problem KPR is established by restricting the flow along each path to be integral in some common scaling factor, (i.e. γ/K) the solution to Problem RMP is established by restricting all lengths to be integral in some common scaling factor. These discretizations enable to reduce the space of feasible solutions and therefore obtain polynomial running time algorithms.

VII. FUTURE ENHANCEMENTS

While this study has laid the algorithmic foundations of two fundamental multipath routing problems, there are still many challenges to overcome. Since algorithm integral routing (that is used to solve Problem KPR) invokes a set of successive computations of a max-flow algorithm, its distributed implementation is straightforward due to [3] that provides distributed implementations for max-flow algorithms. The distributed implementation of Algorithm RMP remains an open issue for future investigation. Finally, as discussed in [4], multipath routing offers a rich ground for research also in other contexts, such as survivability, recovery, network security, and energy efficiency. We are currently working on these issues and have obtained several results regarding survivability [5].

REFERENCES

[1]. R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, Network Flows: Theory, Algorithm, and Applications. sEnglewood Cliffs, NJ: Prentice-Hall, 1993.

[2]. D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for traffic engineering over MPLS," IETF RFC 2702, Sep. 1999.

[3]. B. Awerbuch, "Reducing complexities in the distributed max-flow and breadth-first-search algorithms by means of network synchronization," Network, vol. 15, 425–437, 1985.

[4]. R. Banner and A. Orda, "Multipath routing algorithms for congestion minimization," Dept. Electr. Eng., Technion, Haifa, Israel, CCIT Rep. 429, (2005). [Online]. Available: http://www.ee.technion.ac.il/people/ron/Congestion.pdf

[5]. R. Banner and A. Orda, "The power of tuning: A novel approach for the efficient design of survivable networks," in Proc. IEEE ICNP, 2004, 2–11.

[6]. D. Bertsekas and R. Gallager, Data Networks. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[7]. A. Borodin and R. El Yaniv, Online Computation and Competitive Analysis. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[8]. I. Cidon, R. Rom, and Y. Shavitt, "Analysis of multi-path routing," IEEE Trans. Netw., vol. 7, no. 6, 885–896, Dec. 1999.

[9]. S. Chen and K. Nahrstedt, "An overview of quality-of-service routing for the next generation high-speed networks: Problems and solutions," IEEE Trans. Netw., vol. 12, no. 6, 64–79, Dec. 1998.

[10]. T. Cormen, C. Leiserson, and R. Rivest, Introduction to Algorithms. Cambridge, MA: MIT Press, 2001.

[11]. M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relation- ships of the Internet topology," in Proc. ACM SIGCOMM, 1999, 251–262.

[12]. M. R. Garey and D. S. Johnson, Computers and Intractability. San Francisco, CA: Freeman, 1979

[13]. A. V. Goldberg and R. E. Tarjan, "A new approach to the maximum Flow problem," J. ACM, vol. 35, no. 4, pp. 921–940, 1988.

[14]. H. Han, S. Shakkottai, C. V. Hollot, R. Srikant, and D. Towsley, "Multi- path TCP: A joint congestion control and routing scheme to exploit path Diversity in the Internet," IEEE/ACM Trans. Netw., vol. 14, no. 6, 1260–1271, Dec. 2006.

[15]. C. Huitema, Routing in the Internet. Englewood Cliffs, NJ: Prentice- Hall, 1995.

[16]. S. Iyer, S. Bhattacharyya, N. Taft, N. McKeoen, and C. Diot, "A measurement based study of load balancing in an IP backbone," Sprint ATL, Tech. Rep. TR02-ATL-051027, May 2002.

[17]. Y. Jia, I. Nikolaidis, and P. Gburzynski, "Multiple path QoS routing," in Proc. ICC, 2001, 2583–2587.

[18]. N. Karmarkar, "A new polynomial-time algorithm for linear program- ming," Combinatorica, vol. 4, 373–395, 1984.

[19]. F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control in communication networks: Shadow prices, proportional fairness and stability," J. Oper. Res. Soc., vol. 49, 237–252, 1998.

[20]. F. P. Kelly and T. Voice, "Stability of end-to-end algorithms for joint routing and rate control," Comput. Commun. Rev., vol. 35, no. 2, 5–12, 2005.

[21]. J. Kleinberg, "Single-source unsplittable flow," in Proc. 37th IEEE FOCS, 1996, 68–77.

[22]. J. Moy, "OSPF Version 2," IETF RFC 2328, Apr. 1998.

[23]. S. Nelakuditi and Z.-L. Zhang, "On selection of paths for multipath routing," in Proc. IWQoS, 2001, 170–186.

[24]. V. Paxson, "End-to-end routing behavior in the Internet," in Proc. ACM SIGCOMM, 1996, 25–38.

[25]. P. Papadimitratos and Z. J. Haas, "Secure routing for mobile Ad hoc networks," in Proc. CNDS, 2002, 70–82.

[26]. E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label Switching architecture," IETF RFC 3031, 2001.

[27]. D. Thaler and C. Hopps, "Multipath issues in unicast and multicast next-hop selection," IETF RFC 2991, 2000.

[28]. C. Villamizar, "OSPF optimised multipath (OSPF-OMP)," Internet Draft, 1998.

[29]. Y. Wang and Z. Wang, "Explicit routing algorithms for Internet traffic engineering," in Proc. ICCN'99, 1999, 582–588.

[30]. B. M. Waxman, "Routing of multipoint connections," IEEE J. Sel. Areas Commun, vol. 6, no. 9, 1617–1622, Dec. 1988.

[31]. A. E. I. Widjaja, "Mate: MPLS adaptive traffic engineering," Internet Draft, Aug. 1998.