Lucrative Reserve Allotment of Superimpose Steering Spread Knobs

[I]B.Saranya, [II]S.Muruganandham
[I]M.Phil Full Time Research Scholar, [II]Assistant Professor
[I,II]Dept. of Computer Science, Vivekananda College of Arts and Sciences For Women (Autonomous), Namakkal, Tamilnadu, India.

Abstract

Delay tolerant networks are characterized by the sporadic connectivity between their nodes and therefore the lack of stable end-to-end paths from source to destination. Since the future node connections are mostly unknown in these networks, opportunistic forwarding is used to deliver messages. However, making effective forwarding decisions using only the network characteristics (i.e. average intermeeting time between nodes) extracted from contact history is a challenging problem. Based on the observations about human mobility traces and the findings of previous work, we introduce a new metric called conditional intermeeting time, which computes the average intermeeting time between two nodes relative to a meeting with a third node using only the local knowledge of the past contacts. We then look at the effects of the proposed metric on the shortest path based routing designed for delay tolerant networks. We propose Conditional Shortest Path Routing (CSPR) protocol that routes the messages over conditional shortest paths in which the cost of links between nodes is defined by conditional intermeeting times rather than the conventional intermeeting times. Through trace-driven simulations, we demonstrate that CSPR achieves higher delivery rate and lower end-to-end delay compared to the shortest path based routing protocols that use the conventional intermeeting time as the link metric.

Key Words

Intermeeting time, Conditional Shortest Path Routing, effective forwarding.

I. Introduction

TCP has been the primary transport protocol used in the Internet for over 20 y ears. Improving TCP throughput is of importance due to the wide variety of applications that use TCP . A performance characteristic of TCP is that the rate at which its window size increases is inversely proportional to the average round trip time (RTT), and the average win- do w size is directly related to the throughput. Thus, reducing the RTT between connection endpoints can increase the window size faster, resulting in increased throughput. An intuitive w a y to reduce the effective RTT of a direct TCP connection between any two hosts is to split that connection in to multiple pipelined sub-connections such that each sub-connection has a lower RTT than the direct connection. We observe that the emergence of overlay net works pro- vides a natural platform that can be leveraged to supp ort splitting TCP connections b y allowing peer no des to select each other as in term diaries. Based on this observation, we propose Overlay TCP (oTCP), an application-level protocol extension to TCP that splits an end-to-end TCP connection with a high RTT in to pipelined sub-connections with low RTTs using in terminate nodes to achieve higher end-to-end throughput. In addition to increasing throughput, oTCP can also route around failures and discover paths that are better than the direct path. Thus, ending end-to-end trans- p ort has the potential to not only improve the throughput of a connection but also improve the reliability and quality of paths between endpoints. We advocate oTCP as a viable architecture for throughput demanding applications.

Overlay routing has been proposed in recent years as an effective way to achieve certain routing properties, without going into the long and tedious process of standardization and global deployment of a new routing protocol. For example, overlay routing was used to improve TCP performance over the Internet, where the main idea is to break the end-to-end feedback loop into smaller loops. This requires that nodes capable of performing TCP Piping would be present along the route at relatively small distances. Other examples for the use of overlay routing are projects like RON and Detour, where overlay routing is used to improve reliability. Yet another example is the concept of the "Global-ISP" paradigm introduced in, where an overlay node is used to reduce latency in BGP routing.

In order to deploy overlay routing over the actual physical infrastructure, one needs to deploy and manage overlay nodes that will have the new extra functionality. This comeswith a nonnegligible cost both in terms of capital and operating costs. Thus, it is important to study the benefit one gets from improving the routing metric against this cost. In this paper, we concentrate on this point and study the minimum number of infrastructure nodes that need to be added in order to maintain a specific property in the overlay routing. In the shortest-path routing over the Internet BGP-based routing example, this question is mapped to: What is the minimum number of relay nodes that are needed in order to make the routing between a group of autonomous systems (ASs) use the underlying shortest path between them? In the TCP performance example, this may translate to: What is the minimal number of relay nodes needed in order to make sure that for each TCP connection, there is a path between the connection endpoints for which every predefined round-trip time (RTT), there is an overlay node capable of TCP Piping? Regardless of the specific implication in mind, we define a general optimization problem called the Overlay Routing Resource Allocation (ORRA) problem and study its complexity. It turns out that the problem is NP-hard, and we present a nontrivial approximation algorithm for it. Note that if we are only interested in improving routing properties between a single source node and a single destination, then the problem is not complicated, and finding the optimal number of nodes becomes trivial since the potential candidate for overlay placement is small, and in general any assignment would be good. However, when we consider one-to-many or many-to-many scenarios, then a single overlay node may affect the path property of many paths, and thus choosing the best locations becomes much less trivial. We test our general algorithm in three specific such cases, where we have a large set of source–destination pairs, and the goal is to find a minimal set of locations, such that using overlay nodes in these locations allows to create routes (routes are either underlay routes or routes that use these new relay nodes) such that a certain routing property is satisfied. The first scenario we consider is AS-level BGP routing, where the goal is to find a minimal number of relay node locations that can allow shortest-path routing between the source–destination pairs. Recall that routing in BGP is policy-based and depends on the business relationship between peering ASs, and as a result, a considerable fraction of the paths in the Internet do not go along a shortest path. This phenomenon, called path inflation, is the motivation for this scenario.

II. Related Work

Using overlay routing to improve network performance is motivated by many works that studied the inefficiency of varieties of networking architectures and applications. Analyzing a large set of data, Savage explore the question: How “good” is Internet routing from a user’s perspective considering round-trip time, packet loss rate, and bandwidth? They showed that in 30%–80% of the cases, there is an alternate routing path with better quality compared to the default routing path. In that TCP performance is strictly affected by the RTT. Thus, breaking a TCP connection into low-latency sub connections improves the overall connection performance. In that in many cases, routing paths in the Internet are inflated, and the actual length (in hops) of routing paths between clients is longer than the minimum hop distance between them. Using overlay routing to improve routing and network performance has been studied before in several works. In authors studied the routing inefficiency in the Internet and used an overlay routing in order to evaluate and study experimental techniques improving the network over the real environment. While the concept of using overlay routing to improve routing scheme was presented in this work, it did not deal with the deployment aspects and the optimization aspect of such infrastructure. A resilient overlay network (RON), which is an architecture for application-layer overlay routing to be used on top of the existing Internet routing infrastructure, has been presented. Similar to our work, the main goal of this architecture is to replace the existing routing scheme, if necessary, using the overlay infrastructure. This work mainly focuses on the overlay infrastructure (monitoring and detecting routing problems, and maintaining the overlay system), and it does not consider the cost associated with the deployment of such system. In authors study the relay placement problem, in which relay nodes should be placed in an intradomain network. An overlay path, in this case, is a path that consists of two shortest paths, one from the source to a relay node and the other from the relay node to the destination. The objective function in this work is to find, for each source–destination pair, an overlay path that is maximally disjoint from the default shortest path. This problem is motivated by the request to increase the robustness of the network in case of router failures. In authors introduce a routing strategy, which replaces the shortest-path routing that routes traffic to a destination via predetermined intermediate nodes in order to avoid network congestion under high traffic variability. Roy were the first to actually study the cost associated with the deployment of overlay routing infrastructure. Considering two main cases, resilient routing, and TCP performance, they formulate the intermediate node placement as an optimization problem, where the objective is to place a given number intermediate nodes in order to optimize the overlay routing, and suggested several heuristic algorithms for each application. Following this line of work, we study this resource allocation problem in this paper as a general framework that is not tied to a specific application, but can be used by any overlay scheme. Moreover, unlike heuristic algorithms, the approximation placement algorithm presented in our work, capturing any overlay scheme, ensures that the deployment cost is bounded within the algorithm approximation ratio.

A. Disadvantages of Existing System

In order to deploy overlay routing over the actual physical infrastructure, one needs to deploy and manage overlay nodes that will have the new extra functionality. This comes with a non negligible cost both in terms of capital and operating costs. Our proposed algorithmic framework that can be used in order to deal with efficient resource allocation in overlay routing.

III. Proposed System:

We propose Conditional Shortest Path Routing (CSPR) protocol that routes the messages over conditional shortest paths in which the cost of links between nodes is defined by conditional intermeeting times rather than the conventional intermeeting times. Through trace-driven simulations, we demonstrate that CSPR achieves higher delivery rate and lower end-to-end delay compared to the shortest path based routing protocols that use the conventional intermeeting time as the link metric.

Routing in delay tolerant networks (DTN) is a challenging problem because at any given time instance, the probability that there is an end-to-end path from a source to a destination is low. Since the routing algorithms for conventional networks assume that the links between nodes are stable most of the time and do not fail frequently, they do not generally work in DTN’s. Therefore, the routing problem is still an active research area in DTN’s.

We introduced a new metric called conditional intermeeting time inspired by the results of the recent studies showing that nodes’ intermeeting times are not memory less and that motion patterns of mobile nodes are frequently repetitive. Then, we looked at the effects of this metric on shortest path based routing in DTN’s. For this purpose, we updated the shortest path based routing algorithms using conditional intermeeting times and proposed to route the messages over conditional shortest paths. Finally, we ran simulations to evaluate the proposed algorithm and demonstrated the superiority of CSPR protocol.

Advantages of Proposed System

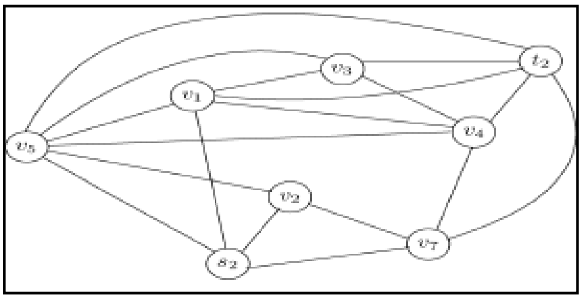
We are only interested in improving routing properties between a single source node and a single destination, then the problem is not complicated, and finding the optimal number of nodes becomes trivial since the potential candidate for overlay placement is small, and in general any assignment would be good.

However, when we consider one-to-many or many-to-many scenarios, then a single overlay node may affect the path property of many paths, and thus choosing the best locations becomes much less trivial.

IV. Implementation

A. Finding Overlay Vertex Cut

While finding a minimal Overlay Vertex Cut required by the algorithm presented above seems like a nontrivial task in the general case, in turns out that when the overlay routing is given explicitly, it is simple as finding a minimal vertex cut. This is done by building a graph G’= (V’, E’) such that a vertex cut in G’ represents an Overlay Vertex Cut in the original graph G.



Finding an overlay cut

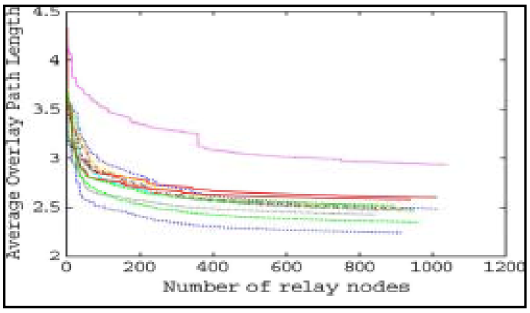
B. BGP Routing Scheme:

BGP is a policy-based interdomain routing protocol that is used to determine the routing paths between autonomous systems in the Internet. In practice, each AS is an independent business entity, and the BGP routing policy reflects the commercial relationships between connected ASs. A customer– provider relationship between ASs means that one AS (the customer) pays another AS (the provider) for Internet connectivity, a peer–peer relationship between ASs means that they have mutual agreement to serve their customers, while a sibling–sibling relationship means that they have mutual- transit agreement (i.e., serving both their customers and providers). These business relationships between ASs induce a BGP export policy in which an AS usually does not export its providers and peers routes to other providers and peers, the authors showed that this route export policy indicates that routing paths do not contain so-called valleys nor steps.

In other words, after traversing a provider–customer or a peer–peer link, a path cannot traverse a customer–provider or a peer-peer link. This routing policy may cause, among other things, that data packets will not be routed along the shortest path. For instance, consider the AS topology graph depicted. In this example, a vertex represents an AS, and an edge represents a peering relationship between ASs. While the length of the physical shortest path between AS6 and AS4 is two (using the path AS6, AS7, AS4), this is not a valid routing path since it traverses a valley. In this case, the length of the shortest valid routing path is five (using the path AS6, AS5, AS1, AS2, AS3, AS4).

In practice, using real data gathered from BGP routing tables, Gao and Wand showed that about 20% of AS routing paths are longer than the shortest AS physical paths. While routing policy is a fundamental and important feature of BGP, some application may require to route data using the shortest physical paths. In this case, using overlay routing, one can perform routing via shortest paths despite the policy. In this case, relay nodes should be deployed on servers located in certain carefully chosen ASs.

Average path length versus number of relay nodes,
BGP scenario



C. TCP Throughput

Using overlay routing to improve TCP performance has been studied in several works in recent years. In particular, the TCP protocol is sensitive to delay, and there is a strict correlation between TCP throughput and the RTT. Thus, it may be beneficial to break high-latency TCP connections into a few concatenated low-latency subconnections. In this case, the set of relay nodes is used as subconnection endpoints, and the objective is to bound the RTT of each one of these subconnections. For instance, assuming that each link in the network depicted in Fig. 7 has a similar latency, the TCP connection between and can be broken using the relay node located in into two subconnections (between and , and between and) reducing the maximum RTT of the connection (although the total length is increased).

ALGORITHM:

Our algorithm basically finds conditional shortest paths (CSP) for each source-destination pair and routes the messages over these paths. We define the CSP from a node n0 to a node nd as follows:

$$\mathrm{CSP}\ (n0,\ nd) = \{n0,\ n1,\ \ldots,\ nd-1,\ nd \mid Rn0\ (n1|t) + \sum_{i=1}^{d-1} Tni\ (ni+1|ni-1) \text{ is minimized.}\}$$

Here, t represents the time that has passed since the last meeting of node n0 with n1 and Rn0 (n1|t) is the expected residual time for node n0 to meet with node n1 given that they have not met in the last t time units. Rn0 (n1|t) can be computed as in with parameters of distribution representing the intermeeting time between n0 and n1. It can also be computed in a discrete manner from the observed intermeeting times of n0 and n1. Then, discrete computation of $Rn_0(n_1|t)$ can be defined formally as follows:

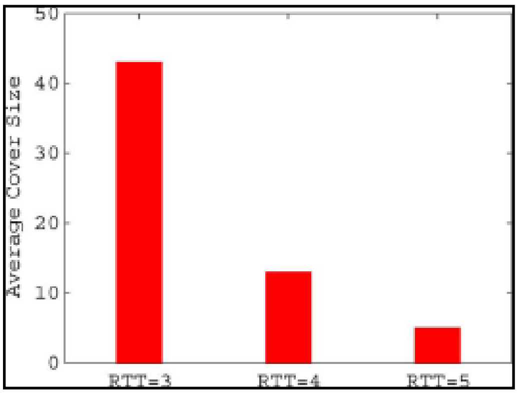
$$Rn_0(n_1|t) = \frac{\Sigma^{k}_{s=1} f^{s}_{n0}(n_1)}{|\{\tau^{s}_{n0}(n_1) \geq t\}|} \quad \text{where,}$$

$$f^{s}_{n0}(n_1) = \begin{cases} \tau^{s}_{n0}(n_1) - t & \text{if } \tau^{s}_{n0}(n_1) \geq t \\ 0 & \text{otherwise} \end{cases}$$

If none of the k observed intermeeting times is bigger than t (this case occurs less likely as the contact history grows), $Rn_0(n_1|t)$ becomes 0, which is a good approximation.

V. Experimental Evaluation

We simulated the TCP scenario using random graphs (i.e., a graph in which a pair of nodes is connected randomly with an equal independent probability) containing 1000 nodes with equal weight. While the RTT depends, among other things, on the routing distance, we have considered the case where most of the delay is contributed equivalently by the network nodes. Thus, the maximum RTT is defined by the path length in hops. Considering a single source serving the entire network, the set of vertices contains all the pairs from a single random source to the rest of the network.



Algorithm coverage for different RTT values.

A. CPSR (Conditional Shortest Path Routing:

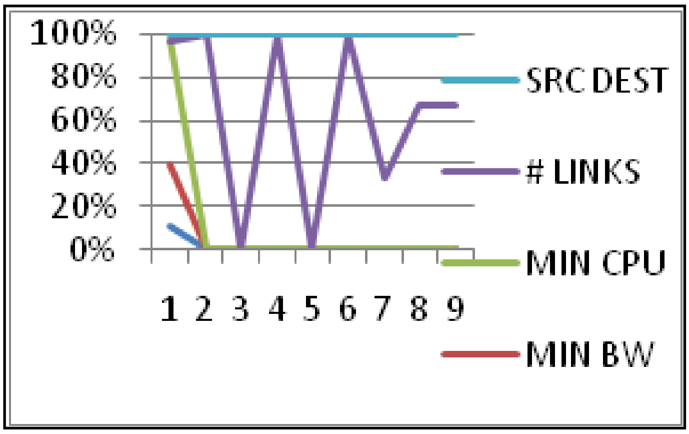
We propose Conditional Shortest Path Routing (CSPR) protocol that routes the messages over conditional shortest paths in which the cost of links between nodes is defined by conditional intermeeting times rather than the conventional intermeeting times. Through trace-driven simulations, we demonstrate that CSPR achieves higher delivery rate and lower end-to-end delay compared to the shortest path based routing protocols that use the conventional intermeeting time as the link metric.

Routing in delay tolerant networks (DTN) is a challenging problem because at any given time instance, the probability that there is an end-to-end path from a source to a destination is low. Since the routing algorithms for conventional networks assume that the links between nodes are stable most of the time and do not fail frequently, they do not generally work in DTN's. Therefore, the routing problem is still an active research area in DTN's.

We introduced a new metric called conditional intermeeting time inspired by the results of the recent studies showing that nodes' intermeeting times are not memory less and that motion patterns of mobile nodes are frequently repetitive. Then, we looked at the effects of this metric on shortest path based routing in DTN's. For this purpose, we updated the shortest path based routing algorithms using conditional intermeeting times and proposed to route the messages over conditional shortest paths. Finally, we ran simulations to evaluate the proposed algorithm and demonstrated the superiority of CSPR protocol.

B. Simulations Result

To evaluate the performance of our algorithm, we have built a discrete event simulator in Java. In this section, we describe the details of our simulations through which we compare the proposed *Conditional Shortest Path Routing* (CSPR) algorithm with standard *Shortest Path Routing* (SPR). To collect several routing statistics, we have generated traffic on the traces of these two data sets.



Overlay request

For a simulation run, we generated 5000 messages from a random source node to a random destination node at each seconds. We assume that the nodes have enough buffer space to store every message they receive, the bandwidth is high and the contact durations of nodes are long enough to allow the exchange of all messages between nodes.

VI. Conclusion And Future Work

While using overlay routing to improve network performance was studied in the past by many works both practical and theoretical, very few of them consider the cost associated with the deployment of overlay infrastructure. In this paper, we addressed this fundamental problem developing an approximation algorithm to the problem. Rather than considering a customized algorithm for a specific application or scenario, we suggested a general framework that fits a large set of overlay applications. Considering three different practical scenarios, we evaluated the performance of the algorithm, showing that in practice the algorithm provides close-to-optimal results. Many issues are left for further research. One interesting direction is an analytical study of the vertex cut used in the algorithm. It would be interesting to find properties of the underlay and overlay routing that assure a bound on the size of the cut. It would be also interesting to study the performance of our framework for other routing scenarios and to study issues related to actual implementation of the scheme. In particular, the connection between the cost in terms of establishing overlay nodes and the benefit in terms of performance gain achieved due to the improved routing is not trivial, and it is interesting to investigate it. The business relationship between the different players in the various use cases is complex, and thus it is important to study the economical aspects of the scheme as well. For example, the one-to-many BGP routing scheme can be used by a large content provider in order to improve the user experience of its customers. The VoIP scheme can be used by VoIP services (such as Skype) to improve call quality of their customers. In both these cases, the exact translation of the service performance gain into actual revenue is not clear and can benefit from further research.

References

[1] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient Overlay Networks. In Proc. of ACM SOSP, 2001.

[2] Q. He, C. Dovrolis, and M. Ammar. On the predictability of large transfer TCP throughput. In Proc. of ACM SIGCOMM, 2005.

[3] H. Pucha and Y. C. Hu. Overlay TCP: Multi-Hop Overlay Transport for High Throughput Transfers in the Internet. Technical Report TR-ECE-05-08, Purdue University, 2005.

[4] PlanetLab. http://www.planet-lab.org.

[5] L. Qiu, Y. R. Yanga, Y. Zhang, and S. Shenker. On sel_sh routing in internet-like environments. In Proc. of ACM SIGCOMM, 2003.

[6] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Ho_man, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan. Detour: A Case for Informed Internet Routing and Transport. IEEE Micro, 19:50{59, 1999.

[7] R. Cohen and S. Ramanathan, "Using proxies to enhance TCP performance over hybrid fiber coaxial networks," Comput. Commun., vol. 20, no. 16, pp. 1502–1518, Jan. 1998.

[8] N. Spring, R. Mahajan, and T. Anderson, "The causes of path inflation," in Proc. ACM SIGCOMM, 2003, pp. 113–124.

[9] H. Tangmunarunkit, R. Govindan, S. Shenker, and D. Estrin, "The impact of routing policy on Internet paths," in Proc. IEEE INFOCOM, 2001, pp. 736–742.

[10] M. Cha, S. Moon, C.-D. Park, and A. Shaikh, "Placing relay nodes for intra-domain path diversity," in Proc. IEEE INFOCOM, Apr. 2006, pp. 1–12.

[11] M. Kodialam, T. V. Lakshman, and S. Sengupta, "Efficient and robust routing of highly variable traffic," in Proc. HotNets III, 2004.

[12] S. Roy, H. Pucha, Z. Zhang, Y. C. Hu, and L. Qiu, "On the placement of infrastructure overlay nodes," IEEE/ACM Trans. Netw., vol. 17, no. 4, pp. 1298–1311, Aug. 2009.

[13] L. Qiu, V. N. Padmanabhan, and G. M. Voelker, "On the placement of Web server replicas," in Proc. IEEE INFOCOM, 2001, vol. 3, pp. 1587–1596.

[14] E. Cronin, S. Jamin, C. Jin, A. R. Kurc, D. Raz, and Y. Shavitt, "Constrained mirror placement on the Internet," in Proc. IEEE INFOCOM, 2001, vol. 1, pp. 31–40.

[15] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. San Francisco, CA, USA: Freeman, 1979.

[16] U. Feige, "A threshold of ln n for approximating set cover," J. ACM, vol. 45, no. 4, pp. 634–652, 1998.

[17] R. Bar-Yehuda and S. Even, "A local-ratio theorem for approximating the weighted vertex cover problem," Ann. Discrete Math., vol. 25, pp. 27–46, 1985.

[18] Y. Rekhter, T. Watson, and T. Li, "Border gateway protocol 4," RFC 1771, 1995.

[19] C. Alaettinoglu, "Scalable router configuration for the Internet," in Proc. IEEE IC3N, Oct. 1996.

[20] G. Huston, "Interconnection, peering and settlement," Internet Protocol J., vol. 2, no. 1, Mar. 1999.

[21] L. Gao and J. Rexford, "Stable Internet routing without global coordination," IEEE/ACM Trans. Netw., vol. 9, no. 6, pp. 681–692, Dec. 2001.

[22] L. Gao, T. Griffin, and J. Rexford, "Inherently safe backup routing with BGP," in Proc. IEEE INFOCOM, 2001, pp. 547–556.

[23] University of Oregon, Eugene, OR, USA, "University of Oregon Route Views Project," [Online]. Available: http://www.routeviews.org

[24] Merit Network, Inc., Ann Arbor, MI, USA, "Internet Routing Registry," [Online]. Available: http://www.irr.net

[25] "DIMES—Distributed Internet Measurements and Simulations," [Online]. Available: http://www.netdimes.org

[26] Y. Shavitt and E. Shir, "DIMES: Let the internet measure itself," Comput. Commun. Rev., vol. 35, no. 5, pp. 71–74, Oct. 2005.

[27] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra, "Routing policy specification language (RPSL)," Internet RFC 2622, 1999.

[28] R. Cohen and D. Raz, "The Internet dark matter—On the missing links in the as connectivity map," 2006.

[29] T. S. S. O. Itu, "ITU-T recommendation G.114," International Telecommunication Union, Tech. Rep., Mar. 1993.