PERFORMANCE MODELING OF HYBRID SATELLITE/WIRELESS NETWORKS USING FIXED POINT APPROXIMATION AND SENSITIVITY ANALYSIS OF THE PERFORMANCE MODELS FOR NETWORK DESIGN

Ayan Roy-Chowdhury, Electrical and Computer Engineering and Institute for Systems Research, University of Maryland College Park, MD 20742, USA, +1(301)4056561, ayan@umd.edu
Vahid Tabatabaee, Institute for Systems Research, University of Maryland College Park, MD 20742, +1(301)4057902, vahidt@umd.edu
John S. Baras, Electrical and Computer Engineering and Institute for Systems Research, University of Maryland College Park, MD 20742, USA, +1(301)4056606, baras@isr.umd.edu

Abstract

In this work, we propose a new analytical method for modeling and estimating the performance of hybrid networks that are comprised of terrestrial wireless networks interconnected by a satellite overlay. We take advantage of the natural hierarchy present in the hybrid architecture to split the network into multiple levels based on the node and link characteristics. For the lower level of terrestrial wireless nodes, we use a simple approximate (throughput) loss model that couples the physical, MAC and routing layers. The model provides quantitative statistical relations between the loss parameters used to characterize multiuser interference and physical path conditions on the one hand and traffic rates between origin-destination pairs on the other. For the higher layer consisting of the terrestrial gateway nodes interconnected by the satellite overlay, we use a simple satellite channel bit error loss model coupled with a channel access protocol, to derive a similar relationship between traffic arrival rate and the link loss. We use the technique of hierarchical reduced loss network model, adapted for packet-switched networks, to create a reduced load network model for the hybrid network that connects the packet arrival rates in the different levels with the associated physical and link loss rates. We then apply a fixed point approximation approach for this set of relations to derive a solution that converges to a fixed point for the given set of parameters, while satisfying all the equations in the set. The result is an implicit model of the selected performance metric parameterized by the design variables. We apply the technique of Automatic Differentiation to the performance models obtained through the fixed point approximation and analyze the sensitivity of the performance metrics to variations in the selected network parameter. We thus develop a methodology for parameter optimization and sensitivity analysis of protocols to aid in the design of hybrid networks. We demonstrate the effectiveness of our approach through simulation results, compared with results from discrete event simulations on identical network topology.

1. Introduction

Hybrid networks that interconnect terrestrial wireless segments via satellite overlay, hold the promise of ubiquitous connectivity, but their deployment is still far away. A major reason for this state of affairs is the lack of systematic methodologies and toolkits for the design and dimensioning of such networks so as to have predictable performance bounds that are measured by a few key performance metrics. This is principally due to the inherent uncertainty and variability of the wireless medium and the interdependence between the performance of wireless links.

It is possible to develop packet level simulation tools based on physical (PHY) and medium access control (MAC) layer models using various packages. However, the packet level simulation of satellite/wireless networks with appropriate PHY and MAC layer modeling turns out to be too complex and time consuming for design and analysis in realistic settings. Our objective is to develop low complexity combined analytical and computational (numerical) models, which can efficiently approximate hybrid network performance. Such models have several applications in the design and analysis of hybrid networks, for example in evaluation of protocol performance and robustness, and component-based design and parameter tuning for optimal performance.

We propose a novel approach based on the fixed point method and loss network models for performance evaluation and optimization. Loss network models [1] were originally used to compute blocking probabilities in circuit switched networks [2] and later were extended to model and design ATM networks [3-6]. In [6], reduced load approximations were used to evaluate complex ATM networks, with complex and adaptive routing protocols and multi-service, multi-rate traffic. The primary challenges in developing loss network models for hybrid satellite/wireless networks are (i) the coupling between wireless links in the terrestrial segment due to the transmission interference between neighboring nodes, and (ii) the coupling of loss and data rates between the terrestrial segment and the satellite overlay. In the terrestrial segment, we propose to approximate interference and contention as inter-link traffic dependent loss factors by using probabilistic physical and MAC layer models developed in [13]. Such an approximate model provides a system of equations describing the relations between reduced link rates. For the satellite overlay, we use the Gilbert-Elliot loss model [7-8] to derive a similar relationship between traffic arrival rate and the link loss. We then use the hierarchical reduced loss network model from [9], to create a network model for the hybrid network that couples the arrival rates in each layer with the associated loss rates, and interconnects the arrival rates and loss rates between the layers. We then use a fixed point approximation method for this set of relations to derive a solution that converges to a fixed point for the set of parameters arrival rate and link loss, while satisfying all the equations in the set

In addition to the performance model we are interested in developing a methodology for design of hybrid networks, through sensitivity and gradient analysis for parameter optimization and robustness evaluation. We use Automatic Differentiation (AD) [10] for sensitivity analysis. AD is a powerful method to numerically compute the derivatives of a software-defined function (i.e. a computer program implementation of the function, which we also consider a 'model' following modern systems engineering and hybrid systems formalisms [11, 12]. The analysis model that we generate based on the fixed point iterations and loss models is the input function to the AD and the output of the AD is the partial derivative of the performance metric (e.g. throughput) with respect to defined input parameters (i.e. design variables or parameters). It is important to note that the method allows for very complex design parameters to be implicitly embedded in the input function to the AD module (see for example the work of Liu and Baras in [6]). As an example, we show how we can use this methodology to find the optimal load distribution among multiple paths in the network to maximize throughput. In this example the gradient projection algorithm is used to find the optimal load distribution, and AD is used to compute the gradient of the network throughput with respect to the load distribution parameters.

The rest of the paper is organized as follows. In section 2 we describe the proposed hierarchical loss network model for the hybrid satellite/wireless network and the fixed point method for performance modeling. In section 3, we discuss how to use Automatic Differentiation for sensitivity analysis of the performance metrics. Section 4 gives numerical results of simulations to validate our approach and highlight its advantages. We conclude this paper with a summary in section 5.

4 The Fixed Point Algorithm for the Hybrid Network

We consider a generic hybrid network comprising terrestrial wireless clusters interconnected by a satellite overlay. We follow the methodology for hierarchical loss network models, described in [9], for the performance model of the hybrid network. We assume unicast communication with the source and destination nodes being in separate terrestrial networks (i.e., clusters/segments), as shown in the network representation in figure 1, where we have a source S in cluster 1.1 and a destination D in cluster 1.3. We assume IEEE 802.11 RTS/CTS as the link layer protocol in each terrestrial cluster and use Bianchi's 802.11 performance model [13], to derive a set of fixed point equations relating the packet arrival rates to the network loss.

4.1 Lower Layer Fixed Point Model

The network topology graph consisting of *N* nodes is given. Each node has the ability to transmit packets at the rate of Λ bits/second to the nodes that are connected to it, i.e., for simplicity, we assume that the physical layer capacity of all links is fixed and equal to Λ. All nodes use omnidirectional antennas, and all neighbor nodes of the transmitting node receive the signal. The connectivity and interfering property between each node pair is decided by the Signal-to-Noise (SNR) ratio (transmission power, distance, modulation, etc). Note that two nodes are neighbors and connected if they can directly communicate, and two nodes are interfering if one of them can not receive data, while the other one is transmitting data to a third node. Let $c = 1, \cdots, C$ be the set of ***commodities*** in the network. Each commodity is specified by its source-destination pair ***(I(c),O(c))***, and traffic demand rate r_c between them. Network links (connections between neighbor nodes) are specified either by their index $l = 1, \cdots, L$ or their source-destination pair ***(i,j)***. Note that r_c is the average traffic rate generated at the source node (the incoming rate), which may not be equal to the traffic received at the destination node (the outgoing rate) if there is packet loss in the network. We assume that η_l, the PHY layer link loss probability is known and fixed.

The routing is known and fixed and it is defined by the set of end-to-end paths (i.e. the paths between the origin and destination of each commodity) and the fraction (probability) of the incoming traffic that is transmitted on each of these paths. Let Π_c be the set of the paths that are used for commodity ***c***. Consider a path $\pi_{c,k} \in \Pi_c$, then $\alpha_{\pi_{c,k}}$ is the fraction (probability) of commodity ***c*** traffic transmitted over path $\pi_{c,k}$ at ***I(c)***, the source node of commodity ***c***. We have:

$$\sum_{\pi_{c,k} \in \Pi_c} \alpha_{\pi_{c,k}} = 1, \quad for \quad each \quad c = 1, \cdots, C \tag{1}$$

Our goal is to find a consistent set of link loss parameters and traffic rate parameters that satisfy two sets of equations. The first set is derived from the network loss model and computes the link outgoing traffic rates $\lambda_l, \ l = 1, \cdots, L$ from the MAC layer effective loss parameters $\epsilon_l, l = 1, \cdots, L$.. The second set is based on Bianchi's model and computes the loss parameters ϵ_l from the rates λ_l. The fixed point method applies these two mappings iteratively, until convergence to a consistent solution $\langle \epsilon^*, \lambda^* \rangle$ that satisfies both mappings is achieved. The existence of a consistent solution follows from the facts that both mappings are continuous and bounded and map a compact subset of R^{2L} into itself, via an application of a fixed point theorem [14]. In the rest of this section we derive the two mappings.

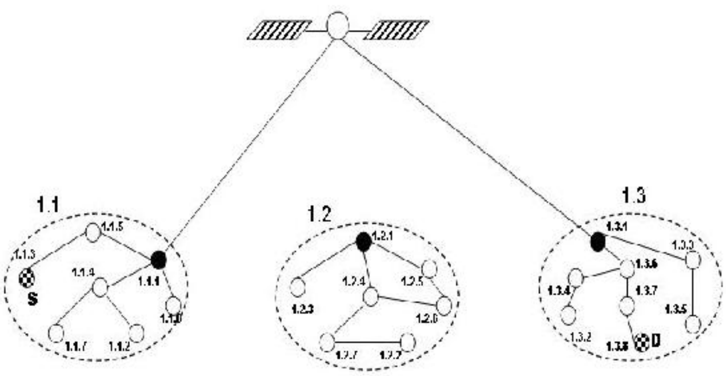

Figure 1: Hybrid network with three wireless clusters and satellite - layer one

4.1.1 First Mapping: From Є to λ

We assume that the packet loss probabilities due to MAC layer contention are given. We compute the λ_l's, which are the *effective* (outgoing) data rates of the network links. Let $(l^{\pi}(1), \cdots, l^{\pi}(k_{\pi}))$ be the set of the links in path π, which are ordered from the first to the last hop in the path. Let s_{π} be the commodity that path π is serving. Then $\lambda^{\pi}_{l^{\pi}(i)}$, the outgoing data rate of the ***i*** th link of path π is,

$$\lambda^{\pi}_{l^{\pi}(i)} = r_{s_{\pi}} \alpha_{\pi} \prod_{j=1}^{i} (1 - \eta_{l^{\pi}(j)})(1 - \epsilon_{l^{\pi}(j)}). \tag{2}$$

Note that r_c is the corresponding source incoming traffic rate for commodity ***c***, α_{π} is the fraction of that traffic routed on path π. The two terms in parentheses specify the percentage of the traffic that is successfully transmitted over the first ***i*** links of the path π. Let ***E(l)*** be the set of paths that share link ***l***. Then the total traffic rate on link ***l*** is,

$$\lambda_l = \sum_{\pi : \pi \in E(l)} \lambda^{\pi}_{l^{\pi}(j_{\pi})} \tag{3}$$

The notation convention in the last equation is that there exists an index j_{π} for path π such that the link ***l*** whose total traffic rate we calculate is identical with the link $l^{\pi}(j_{\pi})$ of path π. Equations 2 and 3, taken together, provide the desired mapping from the vector of ϵ's to the vector of λ's. These computations are executed throughout the network synchronously.

4.1.2 Second Mapping: From λ to ϵ

This mapping is based on Bianchi's 802.11 performance model [13]. Let H_i be the set of interfering nodes with node *i*. The set of nodes that interfere in the transmission from node *i* to *j* is the union of node *i*'s and *j*'s sets of interfering nodes. The total traffic exiting nodes in the set $\{H_i \bigcup H_j \bigcup \{i\} \bigcup \{j\}\}$ contend with each other and share the same multi-access channel. The total traffic demand for this channel is,

$$X_{ij} = \sum_{n=1}^{N} \sum_{m \in \{H_i \bigcup H_j \bigcup \{i\} \bigcup \{j\}\}} \lambda_{mn} \quad (4)$$

For now assume that we have calculated *S* for this channel. Recall that *S* is the fraction of time that the channel is successfully used to transmit data packets. We assume that the contending links have equal capacity Λ. Therefore, the channel capacity is $S\Lambda$. We assume that the channel capacity is divided proportionally between all contending connections. Therefore, the link *(i,j)* effective data rate is,

$$u_{ij} = \begin{cases} \lambda_{ij} & \text{if } X_{ij} \leq S\Lambda \\ \frac{S\Lambda}{X_{ij}} \lambda_{ij} & \text{if } X_{ij} > S\Lambda \end{cases} \quad (5)$$

The intermediate MAC layer loss factor for link *(i,j)* is,

$$\epsilon'(i,j) = \frac{\lambda_{ij} - u_{ij}}{\lambda_{ij}} \quad (6)$$

Assume now that we are at iteration ***k+1***, the new value for MAC layer loss factor of link ***(i, j)*** is the weighted average of the previous iteration value and the intermediate value computed in equation 6:

$$\epsilon_{(i,j)}^{k+1} = \beta \epsilon_{(i,j)}^{k} + (1-\beta)\epsilon'(i,j) \quad (7)$$

The weighted average is introduced to avoid rapid changes and oscillations in the computation of ϵ and λ in the fixed point iterations.

In summary, for each connection (link) in the network we have defined and computed the channel capacity. The channel capacity is based on the Bianchi's saturation model, and hence a function of the number of interfering nodes with the corresponding link. After convergence, the total effective (outgoing) data rate of each channel is less than its throughput. Equations 4 to 7 taken together, provide the desired mapping from the vector of λ's to the vector of ϵ's. These computations are executed throughout the network synchronously.

In addition to the throughput, the fixed point method provides the loss rate on every link of the network. Note that in equation 5, we assume that the channel capacity is proportionally fairly distributed among the contending links. However, the 802.11 protocol is not a fair protocol, and it is even possible that some links face rate starvation in an 802.11 network [15]. To avoid this problem, we make the additional assumption that the incoming traffic of each link is controlled by a rate-controller, so that it does not exceed its allocated rate λ_{ij}.

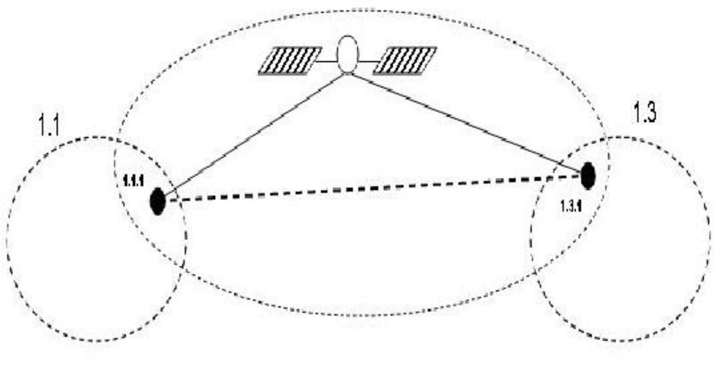

Figure 2: Hybrid network layer two - source, destination and intermediate segments

4.2 Higher Layer Fixed Point Model

For the higher layer performance model, we consider the overlay comprising the satellite and the two gateway nodes 1.1.1 and 1.3.1 logically connected to one another via satellite links, as shown in figure

2. This forms the intermediate segment at the higher layer. We assume symmetric links for the satellite uplink and downlink.

In the higher layer, our goal is to find a consistent set of traffic rate parameters and satellite channel loss parameters that satisfy two sets of equations. The first set is derived from the loss model in the satellite layer and computes the outgoing traffic rate as a function of the loss and the incoming traffic rate. The second set is derived from the satellite channel capacity model and computes the satellite channel loss from the traffic rate. The fixed point method applies the two mappings iteratively until the convergence to a consistent solution that satisfies both set of equations. We subsequently feed the convergence values to the offered load in the lower layer, namely to equation 2 and iterate local and higher layer computations till the differences between successive iterations are within certain criteria.

4.2.1 Mapping from Loss Parameter to Traffic Rate

The aggregate offered load (for class *s* of traffic) between the gateway nodes *i* and *j* in the overlay is:

$$\lambda^1_{ij_s} = \lambda^0_{ij_s} + \sum_p \lambda^0_{ps}.(1-\beta^{src}_p) \quad (8)$$

where ***p*** refers to any source-destination pair whose traffic passes through the logical link connecting the gateway nodes ***i,j***. The first term on the RHS of equation 8 is the initial load between the gateway nodes, while the second term is the initial load for each source-destination pair obtained from the lower layer iterations, thinned by physical and MAC layer losses β^{src}_p in the source cluster.

Let ϵ be the satellite link loss due to physical and MAC layer conditions, computed using the loss model described in section \ref{sat_loss_model}. Therefore, the reduced load in the satellite overlay is:

$$\lambda^1_{ij_s}{}' = \lambda^1_{ij_s}(1-\epsilon)(1-\epsilon'_{ij}) \quad (9)$$

where ϵ'_{ij} is the loss due to data drop in the satellite link between ***i,j***, computed from equation 12. In the initial run of the algorithm, we take its value to be zero.

4.2.2 Mapping from Traffic Rate to Loss Parameter

From equation 8, $\lambda^1_{ij_s}$ is the offered load on the connection between one pair of gateway nodes ***i,j***. Let us express this as λ^1_k. Assuming there are ***L*** such connections between pairs of gateway nodes, the total traffic demand for the satellite channel is:

$$\lambda_L = \sum_{k=1}^{c} \lambda^1_k \quad (10)$$

We assume that the multiple-access channel capacity C_a is divided proportionally between all ***L*** contending connections. Then the effective data rate on the connection between *i* and *j* is:

$$\nu^1_{ij_s} = \begin{cases} \lambda^1_{ij_s}{}' & \text{if } \lambda_L \le C_a \\ \frac{\lambda^1_{ij_s}{}'}{\lambda_L} C_a & \text{otherwise} \end{cases} \quad (11)$$

Therefore, the intermediate satellite link loss factor on the connection between ***i,j*** is:

$$\epsilon'_{ij} = \frac{\lambda^1_{ij_s}{}' - \nu^1_{ij_s}}{\lambda^1_{ij_s}{}'} \quad (12)$$

In order to avoid rapid changes and oscillations in the computation of the fixed point, we update the value of the loss factor as the weighted average of its value in the previous iteration and the intermediate value computed in equation 12. Therefore the value of the loss factor at iteration *k* is:

$$\epsilon_{ij}^{k} = \gamma \epsilon_{ij}^{k-1} + (1-\gamma)\epsilon_{ij}^{'} \qquad (13)$$

The value of the dampening factor γ is chosen such that the algorithm converges to a fixed point.

The fixed point values $\langle \epsilon^*, \lambda^* \rangle$ are fed back to the incoming traffic rate of link *i* in equation 2 and the lower layer and higher layer computations are repeated till the difference in the arrival rates between successive iterations is within pre-determined limits.

Upon convergence of the fixed point iterations, denoting $\lambda_{first,p}$ and $\lambda_{last,p}$ to be respectively the arrival rate of packets of the source or destination of path *p*, we define the throughput of a source-destination pair *c* to be:

$$T_c = \frac{\sum\limits_{p \in P_c} \lambda_{last,p}}{\sum\limits_{p \in P_c} \lambda_{first,p}} \qquad (14)$$

5 Automatic Differentiation for Hybrid Network Design

The fixed point algorithm enables us to do performance analysis for a given network configuration. However, analysis alone is not enough, and we need to develop a methodology for network configuration and optimization too. We use optimal routing design as an example to illustrate our proposed design methodology. Assume that we want to find optimal values for the routing parameters $(\alpha_1, \cdots, \alpha_K)$ to maximize the network throughput. The fixed point method provides a computational scheme that, after convergence (i.e. the fixed point), describes the performance metric as an implicit function of the design parameters. Thus, we do not have (or obtain) analytic expressions of the performance metric evaluations, but instead, we have a program that computes the values of the performance metric, while implicitly providing the dependence of the values on the design parameters.

For parameter optimization, we need to compute the sensitivities of the performance metric(s) (here the throughput) with respect to the design parameters (here the routing parameters). Many powerful methods for design and sensitivity analysis are based on gradient-based schemes. Since we do not have an explicit functional description of the network performance metric, we need to rely on computational methods that numerically approximate the gradients. Automatic Differentiation (AD) [16, 17] is a method to numerically evaluate the derivative of a function specified by a computer program. In order to compute the gradient values, we use the ADIC package [18] that is a source translator augmenting ANSI-C programs with statements for the computation of derivatives using the AD method.

ADIC uses the source-to-source transformation method [20] to implement the AD algorithm. In our case, the input to the ADIC is the fixed point algorithm that we developed in ANSI C code. ADIC generates a new version of the program that computes both the original result, which is throughput, and its derivatives with respect to the input parameters, which are the path routing probabilities. We then use the computed gradients to

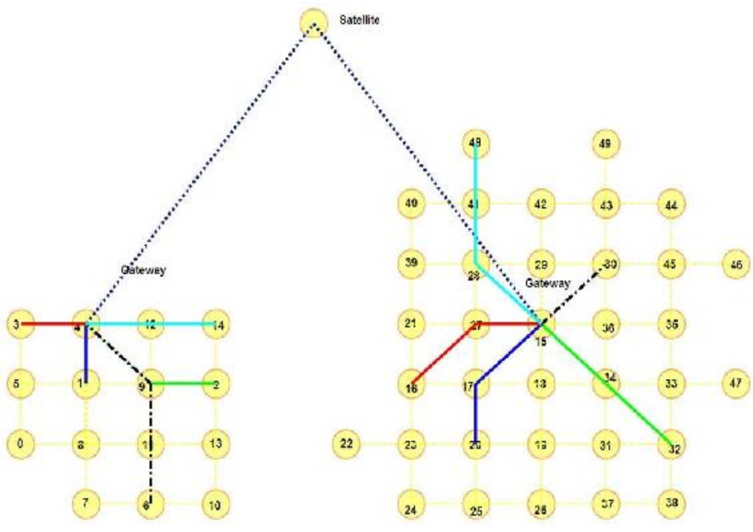

Figure 3: Simulation setup network topology with 5 connections across 2 clusters

find the optimal routing parameters which maximize the throughput. To that end, we use the gradient projection method which is explained in the section 5.1.

5.1 Optimization and Path Selection Algorithms

We use the gradient projection method to maximize the throughput. In this approach, we iteratively compute or estimate the gradients of the throughput with respect to the routing parameters and based on that will update the path probabilities. In addition, we have to project the computed values back into the constraint set, i.e. we have to make sure that all probabilities are positive and the path probabilities of every source-destination pair sum up to one. The explicit computation of the gradients is clearly impossible, hence we use the code generated by ADIC from our fixed point program which numerically computes the throughput and its gradients.
We have implemented the Dreyfus K-shortest path algorithm [19] for path selection. For a given set of link weights and integer value *k* and source-destination pair, this algorithm finds *k* loop free paths with minimum total weight. We set all link weights to one, but it is possible to use other weights based on the distance, bandwidth, interference or other performance related criteria.

6 Results

The first experiment compares the variation of the throughput computed by our fixed point method according to the desired load in the network with the same metric estimated by discrete event simulations using the Opnet Modeler. We set up a simple two-cluster network, presented in figure 3. There are five connections across the two clusters. Since the connection is across clusters, all the paths go through the two gateway nodes, node 4 in cluster 1 and node 15 in cluster 2. Our routing algorithm finds the shortest paths between the source and destination nodes. Using these paths we then employ our set of fixed point equations to compute the throughput of these connections according to the desired load. As can be seen in figure 4 the fixed point model results are close to the Opnet results.

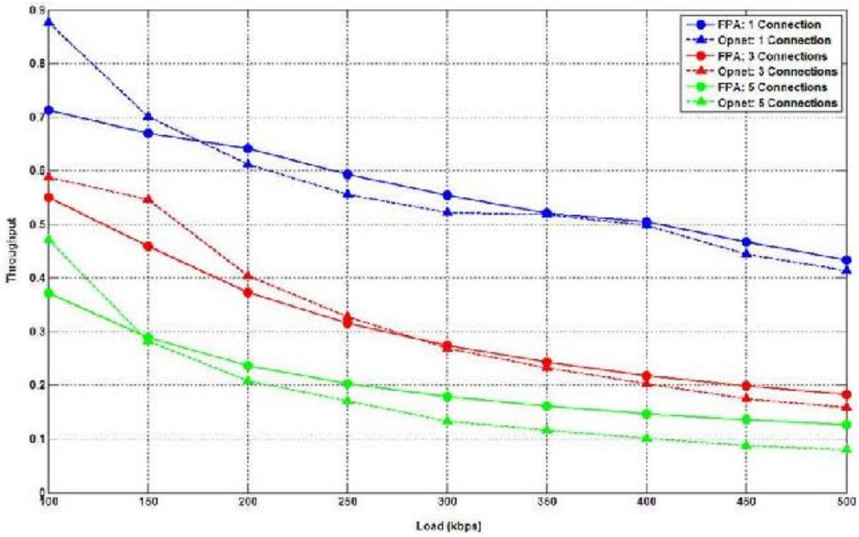

Figure 4: Comparison between fixed point approximation model and Opnet discrete event simulation.

The main advantage of the proposed fixed point model over discrete event simulation platforms, such as Opnet, is the computation time. While our fixed point converges on the order of seconds, Opnet often requires several minutes to compute the throughput. This makes our model more suitable to compute approximations of throughput for network management and design which require fast and/or multiple simulations. Table 1 compares the time needed by our model and by Opnet as a function of the number of active connections in the network.

Connections	1	3	5	7	9	11
Fixed Point	0.089	0.124	0.702	0.907	1.103	1.295
Opnet	229	218	228	242	265	265

Table 1: Comparison of computation time (seconds) between Fixed Point Model and Opnet simulation

Next we use the fixed point model with AD to enhance the routing performance. Here we assume that a fixed set of paths are given and we want to tune the probabilities (portions) of sending traffic over the paths to maximize the throughput. We consider three alternative routings: (1) using shortest path only, (2) using all available paths with equal probability, and (3) using AD and gradient projection method to find the optimal probabilities. Figure 5 shows the aggregate network throughput for the three connections, versus the number of available paths. The performance of the optimization algorithm improves as the number of available paths increases and it clearly outperforms other policies.

7 Conclusion

We have developed a novel method for performance analysis of hybrid satellite/wireless networks, by combining loss network models for the MAC and PHY layers with routing, through fixed point iterations. The method computes sensitivities of performance metrics with respect to design parameters using automatic differentiation. We illustrated applications in a few examples. Future work includes considering the hidden node problem in the wireless MAC layer, derivation of other performance metrics such as delay and buffer over-flow, scalability analysis for large networks, and accuracy bounds on various metrics.

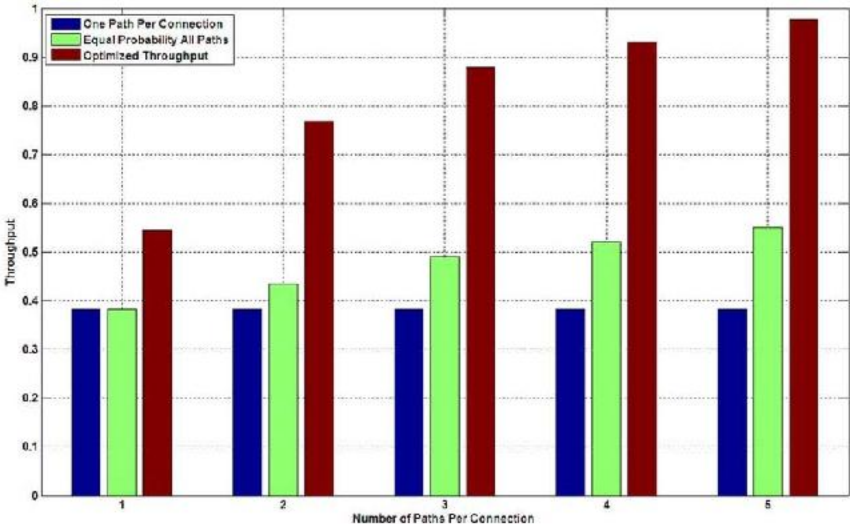

Figure 5: Comparison of aggregate network throughput for number of available paths, 500kbps input load

Reference

1. F. P. Kelly, “Loss networks,” The Annals of Applied Probability, vol. 1, no. 3, pp. 319–378, Aug. 1991.
2. —–, “Blocking probabilities in large circuit switched networks,” Advances in Applied Prob., vol. 18, no. 2, pp. 473–505, June 1986.
3. D. Mitra, J. A. Morrison, and K. G. Ramakrishnan, “ATM network design and optimization: A multirate loss network framework,” IEEE/ACM Trans. Net., vol. 4, no. 4, Aug. 1996.
4. S. Chung, A. Kashper, and K. W. Ross, “Computing approximate blocking probabilities for large loss networks with state-dependent routing,” IEEE/ACM Trans. Net., vol. 1, no. 1, Feb. 1993.
5. A. G. Greenberg and R. Srikant, “Computational techniques for accurate performance evaluation of multirate, multihop communication networks,” IEEE J. Select. Areas Commun., vol. 5, no. 2, pp. 266–277, Feb. 1997.
6. M. Liu and J. S. Baras, “Fixed point approximation for multirate multihop loss networks with adaptive routing,” IEEE/ACM Trans. Networking, vol. 12, no. 2, pp. 361–374, Apr. 2004.
7. E. N. Gilbert, “Capacity of a bursty-noise channel,” Bell System Technical Journal, 39(9):1253–65, September 1960.
8. E. O. Elliot, “Estimates of error rates for codes on bursty noise channels,” Bell System Technical Journal, 42(9):1977–97, September 1963.
9. M. Liu, “Network Performance Modeling, Design and Dimensioning Methodologies,” PhD thesis, University of Maryland College Park, 2000.
10. C. H. Bischof, L. Roh, and A. J. Mauer-Oats, “ADIC: an extensible automatic differentiation tool for ANSI-C,” Software Practice and Experience, vol. 27, no. 12, pp. 1427–1456, Dec. 1997.
11. M. A. Austin and J. S. Baras, “An introduction to information centric systems engineering,” in Proc. of INCOSE’04, Toulouse, France, June 2004.
12. A. Bemporad, A. Bicchi, and G. Buttazzo, Eds., Proceedings of the 10th International Conference on Hybrid Systems: Computation and Control. Pisa, Italy: Springer, Apr. 2007.
13. G. Bianchi, “Performance analysis of the IEEE 802.11 distributed coordination function,” IEEE J. Select. Areas Commun. vol. 18, no. 3, pp. 535–547, Mar. 2000.
14. D. R. Smart, “Fixed Point Theorems,” London, U.K.: Cambridge University Press, 1974.
15. M. Garetto, T. Salonidas, E.W. Knightly, “Modeling Per-flow Throughput and Capturing Starvation in CSMA Multi-hop Wireless Networks,” Proc. of the IEEE Conference on Computer Communications, (INFOCOM 2006), Barcelona, Spain, Apr. 2006.
16. A. Griewank, Evaluating Derivatives; Principles and Techniques of Algorithmic Differentiation, ser. Frontiers in Applied Mathematics. Philadelphia, PA: SIAM, 2000.

17. M. Bucker, G. Corliss, P. Hovland, and U. N. and Boyana Norris., Eds., Automatic Differentiation: Applications, Theory and Implementations, ser. Lecture Notes in Computational Science and Engineering. Berlin, Germany: Springer, 2006, vol. 50.
18. ADIC Resource Center, "Argonne National Laboratory, University of Chicago," http://www-new.mcs.anl.gov/adic.
19. S. E. Dreyfus, "An appraisal of some shortest path algorithms," Operations Research, vol. 17, no. 3, pp. 395–412, May 1969.