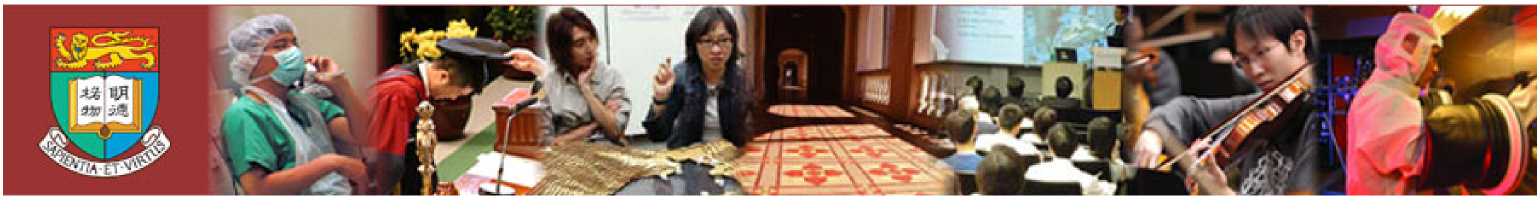

Dynamic Bandwidth Allocation Using Infinitesimal Perturbation Analysis

Ning Xiao
Qualcomm Inc.

Felix F. Wu
Department of Electrical Enginerring and Computer Sciences
University of California at Berkeley

Shau-Ming Lun
Optimal Networks Inc.

Abstract

Advances in network management and switching technologies make dynamic bandwidth allocation of logical networks built on top of a physical network possible. Previous proposed dynamic bandwidth allocation algorithms are based on simplified network model. The analytical model is valid only under restrictive assumptions. Infinitesimal Perturbation Analysis, a technique which estimates the gradients of the functions in discrete event dynamic systems by passively observing the system, is used to estimate delay sensitivities under general traffic patterns. A new dynamic bandwidth allocation algorithm using on-line sensitivity estimation is proposed. Simulation results show that the approach further improves network performance. Implementation of the proposed algorithm in operational networks is also discussed.

I. Introduction

Many networks in large corporations are built on digital bandwidth multiplexers connected by high speed transmission lines [1][2] in order to enhance the flexibility and reliability of the networks and take advantage of economy of scales. This network is called a physical transmission network, and the transmission lines the physical links in [16]. The physical links can be multiplexed into basic channel speeds using a time-division multiplexing technique.The communication equipment are connected to multiplexers through serial interfaces. Serial lines connecting communication equipment, which we call logical links, can be formed by grouping multiple basic channels from physical links. The physical network is thus further divided into a logical packet switched network and a logical circuit-switched network. A typical example of this type of network is shown in Figure 1. Figure 2 shows a logical packet switched network for LAN interconnection built on top of the physical transmission network of Figure 1.

The phenomenal growth in the amount of applications requires frequent bandwidth allocation. Advances in switching technologies and network management make dynamic bandwidth allocation possible. A centralized network management system can use standard network management protocols to control the routers, bandwidth multiplexers and other equipment [1] [2] [3]. The inverse multiplexing technique [3], where one logical link can be formed by multiple basic channels from different physical paths, further improves flexibility.

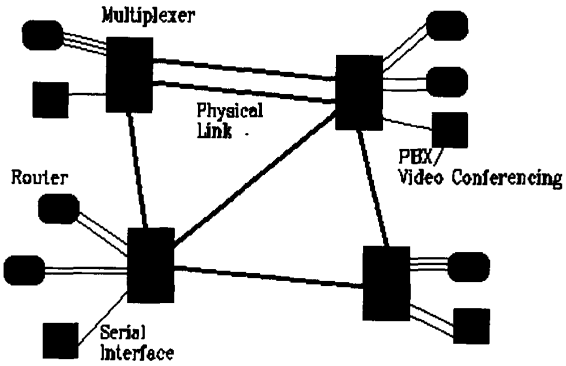

Figure 1. An Example of a Physical Network

Dynamic bandwidth allocation is a scheme which has not been fully explored. Bandwidth allocation of a (logical) packet switched network given the topologies of the logical network and the physical network is discussed in [4]. Flow assignment is assumed given in that work. It is also assumed that a logical link is assigned channels from redefined physical links in one physical path. Related work done for ATM networks also assumes channel assignment of logical links from predefined physical links [5]. A dynamic bandwidth allocation algorithm is recently proposed in [16]. The algorithm integrates the bandwidth allocation of packet switched network with that of circuit-switched network. Flexible channel assignment and

0743-166X/94 $3.00 © 1994 IEEE

inverse multiplexing are also incorporated in the algorithm. Simulation results show that the proposed algorithm improves network performance significantly.

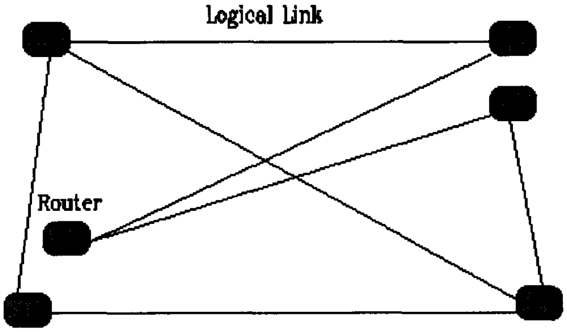

Figure 2. An Example of a Logical Network

The widely-used Jackson model [7] is used in [4][5][16] to model the networks. The model is an approximation to real networks. It assumes external Poisson packet arrivals with exponentially distributed packet lengths. It also assumes that packets are reorganized in every link queue to form exponentially distributed packet lengths [7]. Infinitesimal perturbation analysis (IPA) [13], a technique which estimates the gradients of the functions in discrete event dynamic systems by passively observing the system, is introduced in this paper to allocate link bandwidth under general traffic patterns. The IPA algorithm can either be implemented in communication equipment in operational networks or in the network models in discrete event simulation systems. Simulations have been conducted to demonstrate the advantages of using IPA to estimate the gradients instead of calculating them from the simplified model.

Since the emphasis of this paper is on the application of Infinitesimal Perturbation Analysis technique to the dynamic bandwidth allocation problem, a simplified version of problem formulation and algorithm in [16] is presented in section 2 briefly. Infinitesimal Perturbation Analysis is introduced and applied to the dynamic bandwidth allocation problem in section 3. Simulation results are also presented in Section 3. Section 4 is the conclusion.

II. Problem formulation and algorithm

Given physical network topology and link bandwidth, the topology of logical packet switched network and data traffic in the logical packet switched network assigned by a specific routing algorithm, a dynamic bandwidth allocation algorithm can be used to minimize the average packet delay in the logical packet switched network by properly adjusting the link bandwidths in the logical network and their corresponding channel assignment from physical links. The logical link bandwidth allocation should obey the physical link bandwidth constraint, i.e. the sum of basic channels assigned to logical links in a physical link should not exceed the bandwidth of the physical link. For a logical link through a multiplexer, the corresponding number of channels assigned from the incoming physical links and the outgoing physical links of the multiplexer should be equal. This is called channel assignment constraint. The serial interface constraint specifies that the bandwidth of a logical link should be smaller than the highest speed of its corresponding serial interfaces. The flow-capacity constraint guarantees that the traffic flow in a local link is smaller than the link bandwidth.

For clarity of mathematical formulation, the notation used in the mathematical formulation is listed in the Appendix. The dynamic bandwidth allocation problem can then be formulated as follows

Mathematical Formulation

Minimize

Average packet delay D

Subject to the constraints:

$$\sum_{i} c_{st}^{ij} - \sum_{k} c_{st}^{jk} = \begin{cases} -c_{st} & s\in L(j) \\ 0 & s,t\notin L(j) \\ c_{st} & t\in L(j) \end{cases}$$

$$\text{for } all\ (s,t)\in PN \quad j\in M \qquad (1)$$

$$\sum_{(s,t)\in PN} c_{st}^{ij} \le p^{ij} \qquad (2)$$

$$c_{st}^{ij} \ge 0 \qquad (3)$$

$$0 \le c_{st} \le \overline{c}_{st} \qquad (4)$$

$$f_{st} \le c_{st} \qquad (5)$$

Equations (1) is the channel assignment constraint for logical packet switched network. Constraint (2) states the physical link bandwidth constraint. The serial interface constraint is stated in (4). Constraint (5) is the flow-capacity constraint.

If link bandwidth c_{st} in logical packet switched network and its corresponding channel assignment c_{st}^{ij} are treated as continuous variables, then the problem has a a polyhedron constraint set. Frank-Wolfe's steepest descent algorithm [8] can be used to solve the problem. The method converges to a local minimum of a continuous objective function for any starting point [9].

The critical part of the Frank-Wolfe method is to find a feasible descent direction of the objective function. Starting from a feasible solution c^k, a feasible direction can be found by solving the following problem since constraint (5) is included in the objective function as a penalty function:

$$\text{minimize } \nabla D^T(c^k)\ c$$
$$\text{subject to all the constraints (1) -- (4)}$$

The problem is a linear programming problem which can be solved by standard linear programming techniques such as the simplex method [10]. If c^* is a solution of the problem, then c^* - c^k is a feasible direction.

A dynamic bandwidth allocation algorithm is then proposed as follows:

Algorithm 1

```
Given initial feasible solution $(c^0, f^0)$ and tolerance $t$.
step 0: Set $k = 0$.
step 1: Set $k = k + 1$. find the solution $c^*$ of above linear programming problem.
Step 2: Find the value $\lambda^*$ that minimizes
        $D[\lambda c^* + (1-\lambda)\ c^{k-1}]$ .
step 3: Set $c^k = \lambda^* c^* + (1-\lambda^*)\ c^{k-1}$ .
        if $|D(c^k) - D(c^{k-1})| \geq t$ go to step1,
        otherwise go to step 4.
step 4: Round $c_{st}^{ij}$ and $c_{st}$ to discrete values of multiple basic channel speeds using a heuristic algorithm [16].
```

The proposed algorithm can be implemented in a network management system as a dynamic bandwidth allocation tool. The management system monitors network performance and measures traffic. When performance of the packet switched network deteriorates to a certain degree, the algorithm will be applied to the network using current bandwidth allocation as an initial solution. The new bandwidth allocation will then be sent to routers and multiplexers to improve network performance.The algorithm can also be implemented as a design tool in a network planning system. A heuristic algorithm is designed to find a feasible initial solution in [16].

An integrated network planning environment called NetPlan [6] has been developed at the University of California at Berkeley. NetPlan integrates software tools that analyze, simulate and design communication networks. The object-oriented data model in NetPlan shared by all tools provides a flexible data access for tool development. NetPlan provides a block-oriented discrete event simulation system based on Ptolemy [14]. The design tools and the simulation tools can be integrated in the NetPlan environment.

The proposed algorithm has been implemented in NetPlan as a design tool. Extensive simulation results are presented in [16]. The results show that the proposed algorithm improves network performance significantly. The relatively quick convergence of the algorithm, e.g., about 3 seconds for a 24 router and 12 multiplexer network, makes it possible for on-line implementation.

III. Sensitivity estimation using Infinitesimal Perturbation Analysis (IPA)

The packet switched network is modeled as a network of independent M/M/1 queues in [16]. The following expression for the average packet delay in the packet switched network is [7]:

$$D = \frac{1}{\gamma} \sum_{(s,t) \in PN} \frac{f_{st}}{c_{st} - f_{st}}$$

The average packet delay in the logical packet switched network is a convex function of link bandwidth.Therefore, Algorithm 1 converges to the global minimum of the objective function for any starting point [9].

The model is an approximation to the real network. It assumes external Poisson packet arrivals with exponentially distributed packet lengths. It also assumes that packets are reorganized in every link queue to form exponentially distributed packet lengths [7].

In order to further test the accuracy of the model and the effectiveness of algorithm 1 under general traffic pattern, a simulation model of a 12 router logical network built on top of a 6 multiplexer physical network is implemented in NetPlan. The network has 47 randomly assigned traffic pairs with an average rate of 600 packets per second. The traffic generators built in NetPlan can generate packets according to Poisson arrivals with exponentially or uniformly distributed packet length (PE/PU) or uniform arrival rates with exponentially or uniformly distributed packet length (UE/UU). The algorithm is applied to the network with traffic generated according to various combinations of the traffic models. The average packet delay obtained from discrete event simulations before and after the algorithm is applied to the network under various traffic patterns are shown in Table 1. The initial packet delay calculated according to the analytical model is 129.87 ms. The packet delay calculated according to analytical model after the algorithm is applied is 4.234 ms. All simulation results are with 99% confidence level.

The results in Table 1 shows that the dynamic bandwidth allocation algorithm improves network delay performance under various traffic patterns. However the delays obtained from simulations are different from analytical values in various degrees, especially when network is more congested or packets are not generated according to Poisson arrivals.

Delay (ms)	Analytical	PE/UU	PU/PE	PU/UU	PU/PU	PE/PE
Initial	129.87	6.677+/-0.165	5.210+/-0.089	3.754+/-0.086	3.545+/-0.050	6.686+/-0.113
Final	3.972	2.608+/-0.030	3.362+/-0.031	1.644+/-0.014	2.322+/-0.022	4.285+/-0.047

Table 1. Comparison of average packet delays

If we assume that the distributions of packet rates and packet lengths are general, The average packet delay can then be determined to be [7]

$$D = 1/\gamma \sum \lambda_{st} T_{st}$$

where T_{st} is the average packet delay in link st. This formula is very general and can be applied to networks with general traffic patterns.

This formula can be used as the objective function in the dynamic bandwidth allocation algorithm in replace of the formula obtained from the simplified analytical model. If we assume the independence of packet arrivals in the link queues, the average packet delay sensitivity with respect to the capacity of link *st* can be obtained as follows:

$$\partial D/\partial c_{st} = 1/\gamma \, \lambda_{st} \; \partial T_{st}/\partial c_{st}$$

The sensitivities can be used in the linear programming problem to find the feasible descent direction of average network delay in the network.

Unfortunately, the analytical expression of T_{st} is very difficult to derive since the distribution of packet arrivals and packet lengths in every link is unknown and changes with capacity assignment. New techniques should be investigated to estimate the sensitivities from simulation systems or operational networks in order to overcome the inaccuracy of modeling.

Infinitesimal Perturbation Analysis (IPA) [13][17] technique has recently been developed as an effective method for sensitivity analysis of complex discrete event systems. It enables the sensitivity of a performance measure to be calculated while passively observing a sample path of a (real or simulated) system. An important advantage of the IPA approach is that it is based on real data, and is therefore independent of many restrictive assumptions required for stochastic modeling. It can also provide on-line estimation of performance sensitivities without actually implementing perturbations and with little computational overhead.

IPA has been shown to generate strongly consistent and asymptotically unbiased estimates of average packet delay sensitivities with respect to packet arrival rate and service time in an M/G/1 queue [11]. An extension is given to the GI/G/1 queue [12].

Since the IPA estimates of the gradients are available as the experiments is being observed, the estimates can be used in an optimization algorithm to improve the parameter value while the system is operating on [13]. IPA has also been introduced in the context of estimating delay sensitivity with respect to link capacity and suggested to be used in the network performance optimization problems [7]. An approximation of the IPA technique is introduced in [18] to estimate the delay sensitivities with respect to link traffic flow. The estimates are used in a distributed routing algorithm to optimize the network delay performance. However, it is the first time that state-of-the-art IPA technique is attempted to be used in bandwidth allocation of communication networks.

The Algorithm 1 in [13] is implemented in NetPlan to estimate the delay sensitivity in a link queue where the link capacity is c:

IPA Algorithm

```
0) Initialize: xj =0, xsum=0,observation stop time.
1) Update: At departure of a packet (with service time observed to be xj)
    1.1) j++;
    1.2) xsum += xj;
    1.3) hsum += xsum;
    1.4) If the link is idle, xsum = 0.
2) Test: If packet arrival time ≥ observation stop time, go to Output. Otherwise go to Update.
3) Output: The IPA estimation of the gradient $dT/dc$ is -hsum/(j*c).
```

In an operational network, the IPA algorithm can be implemented in every router to estimate the sensitivities. The feasible direction finding and linear search algorithm can be implemented in a centralized network management system. In every iteration of the bandwidth allocation algorithm, the IPA algorithm is used to estimate the delay sensitivities in every link. The average packet arrival rate and average packet delay in this period for each outgoing link are also collected. The average packet delay sensitivities with respect to the link capacities $\partial D/\partial c_{st} = 1/\gamma \, \lambda_{st} \; \partial T_{st}/\partial c_{st}$ are then sent to network management system to find the feasible descent direction of the average packet delay in the network. A golden-section linear search method is applied to the network after the feasible direction is found. Measurement is conducted at each step of the linear search method to obtain the average packet delay of the network. The measured delay can be used to determine when to terminate the algorithm.

Note that the sensitivities can be obtained in parallel during one observation period in each router in operational networks and only rather simple computation is required for the estimation. The average packet delay of the network can also simply be calculated from average packet delays measured in link queues in the routers. The amount of data needed to be sent to the network management system is rather limited. Furthermore, no extra disturbance to the system is needed. Therefore IPA is well suited for on-line estimation.

As observed in many simulations, exhaustive linear search at each iteration is not effective since the stochastic behavior of the system. It is also not practical in simulation since large amount of computation time is needed. A heuristic method is used to determine a descent point at each iteration in the algorithm. After a feasible direction is found by solving the linear programming problem, the step size λ is set to a constant, typically between 0.02 and 0.04 and a set of new link bandwidths is obtained. If the average delay obtained from the simulation is at least t percent lower than the previous one, the set of bandwidths is accepted as a descent point. Otherwise the linear search is continued until a descent point is reached. In order to prevent wrong descent direction to cause excessive number of simulations, a maximum number of M iterations is set in linear search. When Mth iteration is reached and there is still no t percent lower delay point reached, the lowest delay point in the linear search is then picked as a starting point of the next descent direction finding observation. M is typically chosen between 4 and 8. The algorithm stops when no improvements in average packet delay can be made.

The algorithm is summarized as follows:

Algorithm 2

Given:

Initial feasible solution (c^0, f^0), tolerance t, initial step size λ^*,Iteration number M and observation period N

Step 0: Set $k = 0$.

Step 1: Set $k = k + 1$. IPA algorithm is used in every router to estimate delay sensitivities during one observation period.

Step 2: The sensitivities are sent to network management system to find the solution c^* of the linear programming problem.

Step 3: $j = 0, \quad c = \lambda^* c^* + (1-\lambda^*)\, c^{k-1}$

Step 3.1 Discrete event simulation is conducted to measure the average packet delay.

If $D(c) < (\,1 - t/_{100}\,) * D(c_{k-1})$ or $j > M$, go to step 4. Otherwise, $j = j+1$.

Step 3.2 Apply golden section linear search method and go to step3.1

Step 4: If $D(c^k) < D(c^{k-1})$, go to step1. Otherwise stop.

A simulation system is implemented in NetPlan to test the algorithm. The IPA algorithm is implemented in every router model in the discrete event simulation system. The feasible direction finding and linear search algorithm are implemented in a centralized network management system model. In every iteration of the bandwidth allocation algorithm, the IPA algorithm is used to estimate the delay sensitivity in every link in the router model. The average packet rate and delay in this period in each outgoing link is also collected. The average packet delay with respect to the link capacity

$$\partial D/\partial c_{st} = 1/_{\gamma}\, \lambda_{st}\ \partial T_{st}/\partial c_{st}$$

is then sent back to network management system model to find the feasible descent direction of the average packet delay in the network. A golden-section linear search method is applied to the network after the feasible direction is found. Discrete event simulation is conducted at each step of the linear search method to measure the average packet delay of the network. The results are used to find a descent point of the objective function.

Table 5.2 shows the simulation results when algorithm 2 is applied to the network compared with the simulation results when link bandwidth of the logical network is allocated by algorithm 1. A 3 second of real time in network is used in every simulation, which requires about 30 minutes of elapsed time in simulation in a SUN SPARC2 workstation. The parameters in the algorithm are set as follows: $\lambda^* = 0.04$, $M = 4$, $t = 1$. All average delays of the network are with 99% confidence level. Delays are in milliseconds.

The algorithm is applied to the network from two different sets of starting bandwidths. Results in case B are obtained by applying Algorithm 2 starting from the bandwidths obtained from Algorithm 1. Results in case C are obtained by applying algorithm 2 starting from the same initial bandwidths as Algorithm 1. All results are improved compared with results of case A, where link bandwidths are assigned by Algorithm 1. The tolerance in Algorithm 1 is chosen 0.001 so that the algorithm reaches a value very close to a global minimum of the objective function.

As shown in Table 2, the performance of the network is further improved after IPA is used to estimate the delay sensitivities in all test scenarios. However, the approach used in case B generally generates better results and requires fewer numbers of simulations than the approach used in case C. It suggests that applying Algorithm 1 to a network first and then applying Algorithm 2 to further improve the network performance is a more efficient and effective approach.

Traffic	PE/UU	PU/PE	PU/UU	PU/PU	PE/PE
Result A	2.608+/-0.030	3.362+/-0.032	1.644+/-0.0015	2.322+/-0.022	4.285+/-0.047
Result B	2.204+/-0.024	3.196+/-0.031	1.515+/-0.014	2.219+/-0.021	4.090+/-0.042
Improvement	15.5 %	4.9 %	7.8 %	4.4 %	4.6 %
Iteration #	5	5	4	10	5
Simulation #	13	28	12	28	24
Result C	2.351	3.268	1.572	2.255	4.076
Improvement	9.9 %	2.8 %	4.4 %	2.9 %	4.9 %
Iteration #	6	10	5	10	11
Simulation #	14	28	12	28	41

Table 2. Comparison of network delay when IPA is applied

IV. Conclusion

A dynamic bandwidth allocation algorithm using Infinitesimal Perturbation Analysis is proposed. Simulations are conducted in a network with various traffic patterns. Results show that the approach further improves network performance.

Other traffic patterns, especially traffic pattern measured from operational networks [15] should be modeled in NetPlan to further test the algorithm.

Feasible direction algorithms which converge fast [10] should be tried. Fast Convergence can further decrease the computation time required for the approach.

The IPA technique can further be explored to apply to other network design problems such as capacity assignment in packet switched networks and ATM networks.

Appendix

D = average packet delay in the logical packet switched network

PN = the index set of all links in logical packet switched network

M = the set of all multiplexers in the physical network

c_{st} = the bandwidth of logical link st in packet switched network

c^{ij}_{st} = the total channel speeds in physical link ij assigned to logical link st in packet switched network

f_{st} = the data flow in logical link st in packet switched network

$L(j)$ = the set of routers connected to multiplexer j

p^{ij} = the bandwidth of physical link ij

$\bar{c}_{st}$ = the capacity limit of serial interface for logical link st in packet switched network

γ = the total external packet arrival rate (packets/second) in the packet switched network

TM = the set of all traffic pairs in logical packet switched network

$c = (..., c_{st}, ...)^T$ = the vector of logical link capacities in the packet switched network

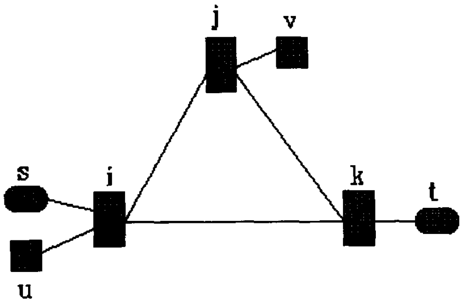

Figure 3. A simple physical network

The notation can be explained through Figure 3. For example, the logical link st in the packet switched network can be formed by channels from physical paths ik and ijk.

References

1. P. Zanella, "Customer Network Reconfiguration Applications Utilizing Digital Cross-connect Systems", GLOBECOM'88, pp1538-1543

2. T. Zerbiec, " Considering the Past and Anticipating the Future for Private Data Networks", IEEE communications Magazine, March 1992, pp36-46

3. R. Palmer, "High Speed Serial Interface (HSSI): Design Considerations For Network Equipment", Issues in Broadband Networking, January 1992

4. R.A. Pazos-Rangel and M.Gerla, "Express Pipe Networks", Global Telecomm. Conf., Dec. 1992, pp B2.3.1-5

5. J. Monteiro and M. Gerla, "Topological Reconfiguration of ATM Networks", INFOCOM 90, pp207-213

6. S. Lun, F. Wu, N. Xiao and P. Varaiya, "NetPlan: An Integrated Network Planning Environment ", to appear in "State-of-the-art in Performance Modeling and Simulation, Vol. 1: computer and Communication Networks", Kluwer Publisher, 1993

7. L. Kleinrock, "Queuing Systems", Volume II, John Wiley & Sons

8. J. Abadie, editor, "Integer and Nonlinear Programming", North-Holland

9. M. Minoux, "Mathematical Programming -- Theory and Algorithm", John Wiley & Sons

10. D. Luenberger, "Linear and Nonlinear Programming", Addison Wesley

11. R. Suri and M. Zazanis, "Perturbation Analysis Gives Strongly Consistent Sensitivity Estimates For the M/G/1 Queues", Management Science, January 1988, pp39-64

12. M. Zazanis and R. Suri, "Estimating First and Second Derivatives of Response Time for G/G/1 Queues from a Single Sample Path", preprint

13. R. Suri, "Perturbation Analysis: The State of the Art and Research Issues Explained via the GI/G/1 Queue", Proc. IEEE, January 1989, pp114-137

14. C. Cassandras and S. Strickland, "Perturbation Analytic Methodologies for Design and Optimization of Communication Networks", JSAC, January 1988, pp158-171

15. R. Caceres, P. Danzig, S. Jamin and D. Mitzel, "Characteristics of Wide-Area TCP/IP Conversations", Proc. ACM SIGCOMM'91 Symp., Zuerich, Switzerland, Sept. 1991, pp101-112

16. N. Xiao, F.F. Wu and S.M. Lun, "Dynamic Bandwidth Allocation in Integrated Communication Networks", GLOBECOM'93

17.Y.C.Ho and X. Cao, "Perturbation Analysis of Discrete Event Dynamic Systems", Kluwer Publisher, 1991

18. C. Cassandras, M. Abidi and D. Towsley, "Distributed Routing with On-line Marginal Delay Estimation", IEEE Transactions on Communicaton, Vol. 38, No.3, March 1990