Optim Eng (2006) 7:421–444
DOI 10.1007/s11081-006-0348-y

Non-linear game models for large-scale network bandwidth management

Dalia Fayek · George Kesidis · Anthony Vannelli

Received: 2 May 2003 / Revised: 29 July 2004

Abstract The design of dynamic Label-Switched Paths (LSP's) in MultiProtocol Label Switched (MPLS) networks is an NP-hard optimization problem. An LSP is a logical path between two nodes in the network. This path has a pre-reserved amount of bandwidth that defines its size. The LSP design problem consists of determining the number of these logical links and configuring the physical path and the size of each LSP. This paper presents an optimization model based on game theory. In this approach, connection requests are modeled as competitive players in a noncooperative game context. The transport network bandwidth constitutes the resource for which optimization is sought. The outcome of this optimization is a set of LSPs upon which the competing connections are routed.

Keywords Noncooperative games · MPLS tunnel dimensioning · Mixed-integer nonlinear optimization · Nash equilibrium · Stackelberg equilibrium

1 Introduction

Modern computer networks are large-scale systems that carry a wide variety of traffic classes, such as video, audio and computer data, each with its own characteristics and quality of service requirements. Supported applications range from simple file transfers to resource demanding multimedia services such as video-conferencing and network management schemes.

D. Fayek (✉)
School of Engineering, University of Guelph, Guelph, Ontario N1G 2W1
e-mail: dfayek@uoguelph.ca

G. Kesidis
Electrical Engineering, The Pennsylvania State University, University Park, PA 16802
e-mail: kesidis@engr.psu.edu

A. Vannelli
Electrical and Computer Engineering Dept., University of Waterloo, Waterloo, Ontario N2L 3G1
e-mail: vannelli@cheetah.vlsi.uwaterloo.ca

Label-Switched Paths (LSP's) are used in MultiProtocol Label Switched (MPLS) networks as a mechanism for traffic control. The LSP can be viewed as a pre-established logical route through the network with dedicated bandwidth onto which individual connections are multiplexed. LSP's considerably simplify the hardware in the transit nodes, provide simple flow admission control and reduce the set-up delays. However, the use of LSP's also increases the call blocking rate and reduces the degree of capacity sharing in communication networks (Awduche and Rekhter, 2001; Awduche et al., 2000; Ghizzi et al., 2004; Goyal and Bellur, 2005; Hussain, 2004; Medrano et al., 2004; Rosen et al., 2001; Shi and Mohan, 2004).

Pricing in telecommunications networks is an ongoing reseach topic to enhance network throughput and increase revenue. Recent work related to network pricing models can be found in Blefari-Melazzi et al. (2000), Marbukh and Moayeri (1999), Xinjie and Subramanian (2000), Courcoubetis et al. (2001), Campos-Nanez and Patek (2003) and Jin and Kesidis (2005).

In the network design phase, the LSP capacity allocation is an NP-hard optimization problem (Aneroussis and Lazar, 1996; Aydemir and Viniotis, 1996; Fayek, 2001; Gerstel et al., 1996; Lim and Chae, 2001). A variety of approaches that aim to find the near-optimal LSP configuration in MPLS networks is found in the literature. Recently, this problem was viewed from a game perspective, where different connection requests are considered as "selfish" users competing for the network resources (Almendral et al., 2004; Anandalingam, 1997; Aresti et al., 2004; Jin and Kesidis, 2003; Korilis et al., 1997b; Liu and Simaan, 2005; Marbukh, 2001; Yaïche et al., 2000). The self-optimizing behaviour of the users leads to a dynamic behaviour of the network. Game theory provides the systematic framework to study and understand the behaviour of noncooperative users in communication networks. A game models situations where individuals with different, and at times conflicting, interests interact. The operating points of a noncooperative network are the Nash equilibria of the underlying game, that is, the points where unilateral deviation does not help any user to improve his performance. In the case of games with relative priorities, where one or some users have precedence in selecting their game strategy, the operating point is then called a Stackelberg equilibrium (Başar and Jan Olsder, 1995).

In large-scale networks, the information processing becomes unmanageable due to the network size. Conceptually organizing the network into several hierarchical levels has been shown to be beneficial for information handling purposes (The ATM Forum, 1996; Sullivan and Callon, 1994). Decentralized control algorithms provide the overall network management.

In this paper, we present an approach for LSP allocation in large-scale MPLS networks, based on a game-theoretic framework. This paper is organized as follows. An introduction was given in Section 1. Section 2 provides the context and background for the LSP allocation problem found in the literature. Section 3 presents the core of this work by first summarizing some game-theory related definitions. Then it presents the problem formulation of the LSP allocation developed in this research. In Section 4, an analytical comparison between the classical and game-based approaches is conducted. The simulation results of experiments that are using the different models are reported for comparison purposes. Finally, Section 5 summarizes the findings in the previous section and concludes by justifying the benefits of using the presented approach over the classical one.

2 Context and background

MultiProtocal Label Switched networks were brought in to bridge the gap between the Internet Protocol IP running on both Asynchronous Transfer Mode (ATM) and IP routers. The purpose is to offer seamless network management that is independent of the physical layer equipment. Communication in ATM networks is based on fixed size packets called *cells*. In fact, the MPLS label can be carried in the Virtual Path Identifier (VPI) field of ATM cells (Lee and Choi, 2001; Rosen et al., 2001; Trimintzios et al., 2003; Xiao et al., 2000; Xiao and Ni, 1999). In this case, when multiple Virtual Paths are grouped into one, a label merging process occurs on the MPLS level. A connection, on the other hand, is composed of a sequence of cells created by a node in the network, the *source* node, and received by a *destination* node. This renders ATM and hence MPLS networks connection-oriented by design. When a connection is admitted into the network, a *virtual circuit* or *virtual channel* (VC) is established along the path chosen to route the connection between the given Source-Destination (SD) node pair.

In ATM networks, the cell loss requirement is considered the most stringent and usually dominates other performance requirements (Lin and Cheng, 1993). Consequently, the *Virtual Path* (VP) concept was introduced, and was shown to be a powerful transport mechanism for ATM networks (Ahn et al., 1994; Aneroussis and Lazar, 1996; Chlamtac et al., 1993; Gerstel et al., 1996; Hadama et al., 1994; Lin and Cheng, 1993). By allowing VCs to be bundled together in a single larger logical unit, the VP, the results are (i) smaller total processing requirements, (ii) faster processing per circuit and (iii) significant improvement in use of network resources. More than 90% of processing time can be saved when VCs are routed on VPs rather than processed individually (Chlamtac et al., 1993; Xiao et al., 2000). The down-side, however, is an increase in loss rates since the overall resource sharing degree is reduced due to the pre-reserved capacity in VP's, and consequently a decrease in network revenue.

The VP distribution problem was proven to be an NP-hard (Ahn et al., 1994; Aneroussis and Lazar, 1996; Chlamtac et al., 1993) optimization problem. Similarly, the LSP dimensioning problem (that is basically a classical VP distribution in ATM networks) is NP-hard and can be described as follows. Given the network topology, the effective capacities of network links, the capacities of the signalling processors, and the matrix of offered load, calculate the routes and capacities of LSP's in the network such that the following requirements are satisfied:

1. The sum of LSP capacities on each link does not exceed its effective capacity.
2. The signalling load on each signalling processor is below a predefined upper bound.
3. The call blocking rate of each SD pair is below a predefined upper bound (also referred to as the SD blocking constraint).
4. The network revenue is maximized under the above constraints.

The Poisson model is adequate for modeling the connection-level behaviour in broadband networks (Kelly, 1991; Ross, 1995; Wolff, 1989). Therefore, it is widely used to model the call arrival process in current telephone networks. Practically, when sampling over a given period of the day characterized by intense traffic, the interarrival times between successive connections are independent from each other and from the holding times of calls. They are usually modeled by a collection of independent identically distributed random variables following an exponential distribution. The arrival process therefore belongs to a Poisson distribution (Wolff, 1989). The results of interest to our model are summarized in the following discussion.

The Danish mathematician, A. K. Erlang studied the problem of telephone calls arriving at a link comprising C circuits as a Poisson process with rate λ. With each call holding a circuit for an independent exponentially distributed time with mean $1/\mu$ and by defining the "traffic intensity" $\rho = \lambda/\mu$, Erlang published his famous formula,

$$\mathcal{E}(\rho, C) = \frac{\frac{\rho^C}{C!}}{\sum_{n=0}^{C} \frac{\rho^n}{n!}} \tag{1}$$

for the loss probability of a telephone system (Brockmeyer et al., 1948). A call is blocked and lost if all C circuits are occupied, and call holding times are independent of each other and of arrival times, in addition, they are identically distributed with mean $1/\mu$.

To find the blocking probability L_r on a route r comprised of J links, Erlang also established the following relation known as the "*Erlang fixed-point approximation*" (Kelly, 1991):

$$1 - L_r \approx \prod_{j \in r} (1 - \mathcal{E}_j) \tag{2}$$

where

$$\mathcal{E}_j = \mathcal{E}(\rho_j, C_j), \quad j = 1, 2, \ldots, J$$

and $\mathcal{E}$ is Erlang's formula given by Eq. (1).

3 Non-linear programming model

In the previous section, we presented the Lable-Switched Path optimization problem using a classical approach in solving LSP dimensioning. This model is based on maximizing a single objective function representing the total network revenue.

In the present section, we present a different approach to solve the same problem. The approach discussed here is based on game-theoretic formulation. The framework is based on hierarchical organization of large-scale MPLS networks (Fayek et al., 1999). A large network is composed of subnetworks that can themselves be composed of even smaller subnetworks. A conceptual layered (hierarchical) organization results in which each subnetwork is represented by a node in a parent hypernode. In essence, the original physical network is at the bottom of the hierarchy, i.e., at layer 1.

Each subnetwork (or hypernode) is optimized individually based on the traffic pattern where connections can be either: (i) generated by a source node in the subnetwork,

(ii) arriving at a destination node or (iii) transiting the subnetwork. For the last category, the transit traffic is passed down by the subnetwork's parent node for resolving the routing details within the subnetwork. The optimization within a subnetwork is described in Section 3.2. Section 3.1 provides an introductory summary of some definitions related to game theory that are relevant to the problem formulation described subsequently.

3.1 Nash and Stackelberg equilibria

In the subsequent definitions for N-players noncooperative games, the following notation is used:

Definition 1. In a N-player finite game, we call the set-valued function R^i the *optimal response function* of player i. It is defined as follows:

$$R^i(\gamma_{-i}) = \{\varepsilon \in \Gamma^i : J^i(\varepsilon, \gamma_{-i}) \leq J^i(\gamma^i, \gamma_{-i}) \ \forall \gamma^i \in \Gamma^i\}$$

where

$\mathbf{N}$	$\{1, \ldots, N\}$ players' set; $i \in \mathbf{N}$
P_i	player i
J^i	cost function of P_i
γ^i	strategy (decision rule) of P_i
γ_{-i}	the $(N-1)$-tuple $\{\gamma^1, \gamma^2, \ldots, \gamma^{i-1}, \gamma^{i+1}, \ldots, \gamma^N\}$ of all players' strategies except P_i
Γ^i	strategy space of P_i

To show that a game has a Nash equilibrium, it suffices to show that there is a profile $\gamma^* = \{\gamma^{1*}, \gamma^{2*}, \ldots, \gamma^{N*}\}$ of strategies such that (Başar and Jan Olsder, 1995)

$$\gamma^{i*} \in R^i(\gamma_{-i*}) \quad \forall i \in \mathbf{N} \tag{3}$$

Equation (3) states that the intersection of the $\mathbf{N}$ R^i sets $\forall i \in \mathbf{N}$ constitutes the set of Nash equilibrium points. Fixed point theorems (Smart, 1974) provide conditions on R under which there exists a value of γ^* for which $\gamma^* \in R(\gamma^*)$, i.e. the intersection of the R^is is a nonempty set.

Theorem 1. *Let Γ be a compact convex subset of $\mathbf{R}^n$ and let $R : \Gamma \to \Gamma$ be a set-valued function for which Joshi and Bose* (1985) *and Smart* (1974)

- *for each $\gamma \in \Gamma$, the function $R(\gamma)$ is nonempty, convex and assigns a closed and convex subset of Γ,*
- *the function $R(\gamma)$ is upper-semicontinuous for all $\gamma \in \Gamma$,*

then there exists at least one $\gamma^* \in \Gamma$ such that $\gamma^* \in R(\gamma^*)$.

The Nash equilibrium solution concept presented above provides a reasonable noncooperative equilibrium solution for nonzero-sum games when the roles of the players are symmetric and there is no single player dominating the decision process. However, there are yet other types of noncooperative decision problems wherein one (or more) player has the ability to enforce his strategy on the other players, and for such decision problems one has to introduce a hierarchical equilibrium solution concept. The player who holds the powerful position in such a decision problem is called the *leader* and the other players who react rationally to the leader's decision (strategy) are called the *followers*. In games with multi-levels of hierarchy in decision making, the equilibrium solution is called a *"Stackelberg"* equilibrium.

Definition 2. In a two-player finite game with P_1 acting as the leader and P_2 as the follower, the optimal response of P_2 for any strategy γ^1 of the leader is $R^2(\gamma^1)$ given by Definition 1. $\gamma^{1*} \in \Gamma^1$ is called a *Stackelberg equilibrium strategy* for the leader, if

$$J^{1*} \triangleq \max_{\gamma^2 \in R^2(\gamma^{1*})} J^1(\gamma^{1*}, \gamma^2) = \min_{\gamma^1 \in \Gamma^1} \max_{\gamma^2 \in R^2(\gamma^1)} J^1(\gamma^1, \gamma^2)$$

The quantity J^{1*} is the *Stackelberg cost* of the leader, and any element $\gamma^{2*} \in R^2(\gamma^{1*})$ is an *optimal strategy* for the follower that is *in equilibrium* with γ^{1*}. The pair $\{\gamma^{1*}, \gamma^{2*}\}$ is a *Stackelberg solution* for the game.

In an attempt to minimize his cost, the leader P_1 accounts for the worst case scenario in response to a given strategy γ^1, by assessing his maximum cost for each strategy $\gamma^2 \in R^2(\gamma^1)$ and selecting the strategy γ^{1*} that results in the least maximum possible. This is what the "min max" term in Definition 2 refers to. Here, J^{1*} is no longer only a secured equilibrium cost level for the leader, but it is the cost level that is actually attained. Furthermore, if $R^2(\gamma^1)$ is singleton for each $\gamma^1 \in \Gamma^1$, then

$$J^{1*} \leq J^1{}_N$$

In other words, the leader never does worse in a "Stackelberg game" than in a "Nash game" (Başar and Jan Olsder, 1995).

3.2 Resolving the competition

Let us consider a local subnetwork $G(V, L)$, where V is a finite set of nodes and $L \subset V \times V$ is a set of directed links, such that each link $l \in L$ has a bandwidth C_l. Without loss of generality, we assume that at most one link exists between each pair of nodes, in each direction. For each link $l \in L$, let $S(l)$ denote the node at the starting point of l and $D(l)$ denote the node at the ending point.

Furthermore, let $\mathcal{I}_k = \{1, 2, \ldots, N_k\}$, $k \in \{1, 2, \ldots, K\}$ denote the set of layer (k) demands competing for the bandwidth within the subnetwork G. In a local layer (k) subnetwork, the network manager deals with $k, k+1, \ldots, K$ demand types. Recall that a type-k connection means that it is administered by a layer (k) network manager. Figure 1 depicts a plausible pattern of traffic demands found in a layer (1) subnetwork.

Let us first define the following variables:

$c_i{}^l$ is the amount of capacity reserved by connection i on link l.
$c_t{}^l$ is the total capacity reserved by all connections on link l, $c_t{}^l = \sum_{k_l=k}^{K} \sum_{i \in \mathcal{I}_{k_l}} c_i{}^l$
c_i is the total capacity reserved by the ith connection on all simple paths (routes)

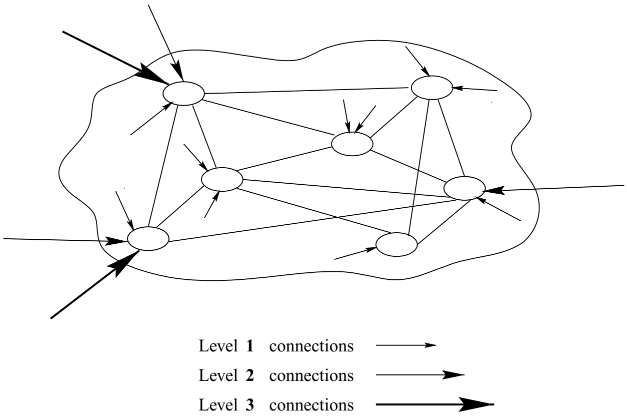


Fig. 1 Example of traffic demands in a layer (1) subnetwork, with $K = 3$

joining the node pair (s_i, d_i), $c_i = \sum_{l:S(l)=s_i} {c_i}^l$.

In Fig. 1, the set of demands constitutes the set of players in a noncooperative Stackelberg game:

- The strategy space of player i is the capacity vector $C_i = \{c_i^1, c_i^2, \dots, c_i^L\}$.
- The set of players $\mathcal{I}_3$ are the leaders of the game, followed by the set $\mathcal{I}_2$ players, and finally by the followers in the $\mathcal{I}_1$ set. The reason for this ordering comes from the fact that the higher the index of the call, the more revenue it produces. Therefore, the network manager, at any level, trying to maximize the network's revenue, assigns different priorities to the calls according to their index.
- A Nash game occurs between the first set of players. The resulting Nash Equilibrium Point (NEP) is denoted by $\mathbf{C}^3$. The links capacities are updated accordingly.
- The equilibrium point strategy $\mathbf{C}^3$ becomes the leader's strategy for the followers in the subsequent $\mathcal{I}_2$ set. A second NEP point is reached and is denoted by $\mathbf{C}^2$. Note that $\mathbf{C}^2 \in R(\mathbf{C}^3)$ (recall Definition 2). Once again, the links capacities are updated.
- Finally, the NEP of the $\mathcal{I}_1$ players is $\mathbf{C}^1 \in R(\mathbf{C}^2, \mathbf{C}^3)$.

Figure 2 outlines the algorithm of LSP dimensioning executed by each local network manager. In the previous discussion, we assumed that there exists at least one NEP at each level. In the sequel, we present the validity of this assumption.

Each player i in the set $\mathcal{I}_k = \{1, 2, \dots, N_k\}$ tries to reserve capacity ${c_i}^l$ between source node s_i and destination node d_i on link l subject to the following:

- ${c_i}^l \geq 0$,
- ${c_i}^l \leq C_l$ where C_l is the capacity of link l,
- $\forall u \neq s, d$: $\sum_{l:S(l)=u} {c_i}^l = \sum_{l:D(l)=u} {c_i}^l$, i.e., the reserved capacity outgoing from a tandem node must be equal to that ingoing to that node, since an excess in capacity in any direction is of no benefit.

From the restrictions above, the capacity reservation problem is a network flow problem (Gibbons, 1985), with the maximum flow being the solution to the problem of the total capacity that player i can reserve in G.

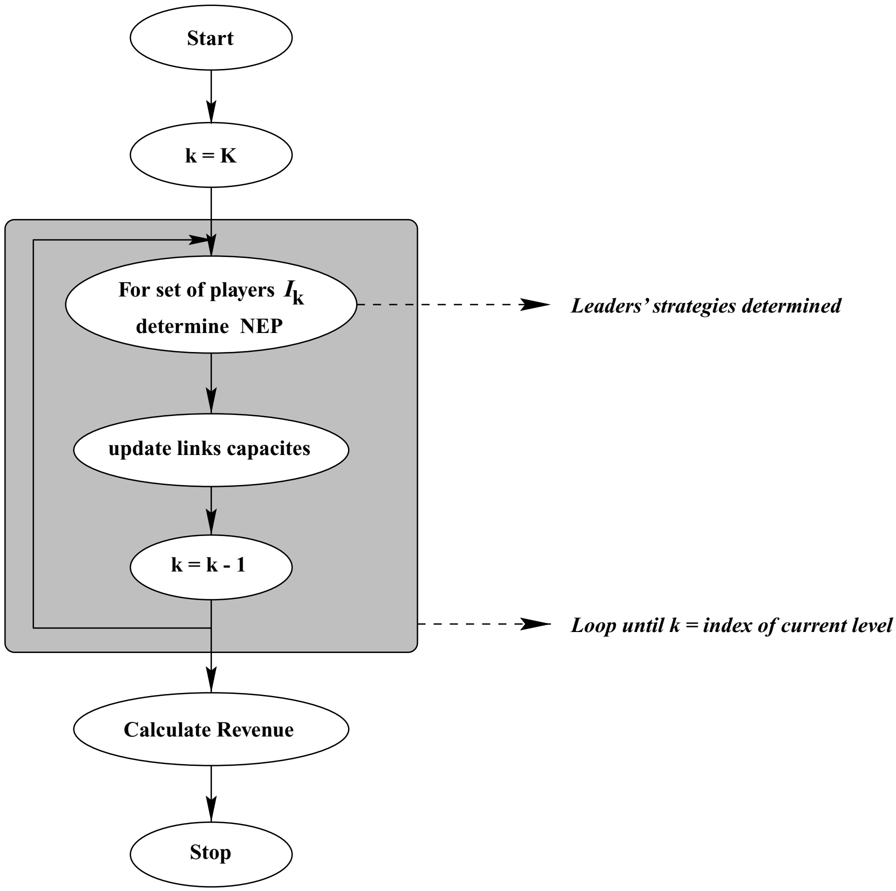


Fig. 2 Functional description of a local network manager

In the game, each player tries to optimize its objective function. In our case, the objective function for player i is described by:

$$J_i(\mathbf{C}) = \sum_{l \in L} F_i{}^l\left(c_i{}^l, c_t{}^l\right) + G_i(c_i), \qquad \forall i \in \mathcal{I}_{k_l}, \quad k_l = k, k+1, \ldots, K. \tag{4}$$

where the $F_i{}^l$ function accounts for the availability of resources on link l perceived by the ith connection, whereas the function G_i accounts for the effect that the amount of reserved capacity has on the performance of that particular connection. The definition of these two functions is as follows:

$$F_i{}^l\left(c_i{}^l, c_t{}^l\right) = c_i{}^l \cdot \frac{\alpha_i}{1 - c_t{}^l / C_l} \tag{5}$$

and

$$G_i(c_i) = \begin{cases} \dfrac{1}{\kappa_i - p_i}, & \text{if } p_i < \kappa_i \\ \infty, & \text{otherwise.} \end{cases} \tag{6}$$

where

$$p_i = 1 - \sum_r \lambda_{ir}(1 - L_r)$$

α_i	is the price per unit reserved capacity on link l by connection i.
λ_{ir}	the portion of connection i routed on route r .
L_r	is the blocking probability on route r (Eq. (2)).
$\kappa_i < 1$	is an upper bound on connection i's call blocking probability, p_i, as determined by its QoS requirements.

In the above formulation, the cost function of each player $J_i(\mathbf{C})$ is designed to possess the following properties. The F function is convex and continuously differentiable with respect to ${c_i}^l$. ${F_i}^l({c_i}^l, {c_t}^l)$ is also monotonically increasing in each of its two arguments and $\partial {F_i}^l({c_i}^l, {c_t}^l)/\partial {c_i}^l$ is nondecreasing with respect to ${c_t}^l$. Finally, we can notice that as ${c_i}^l \to C_l$, ${F_i}^l({c_i}^l, {c_t}^l) \to \infty$. This limit on the value of ${c_i}^l$ prevents any connection from exhausting the link resources on its own (Korilis et al., 1997a; Lazar et al., 1995).

A connection can be routed on any path (or combination of paths) between the source and destination on which the amount of reserved and unused capacity can accommodate its demand size. This means that the loss process depends only on the total amount of capacity reserved by that connection on all paths, and not on the precise distribution of that capacity among the paths. Consequently, the cost function G_i takes as its argument the total capacity c_i. Due to the nodal conservation of capacities, the total capacity c_i is equal to the sum of capacities reserved on links outgoing from the source node s_i (equivalently, the sum of capacities reserved on links ingoing the destination node d_i).

Thus, the G_i function has the following properties (Messerli, 1972). $G_i(c_i)$ is convex, strictly decreasing and continuously differentiable with respect to c_i. Moreover, $\lim_{c_i \to 0} G_i(c_i) = \infty$ which indicates that each connection needs some positive amount of capacity.

By inspection, we can see that the F function guards against the violation of the link capacity constraint, while the G function is responsible for the blocking constraint found in the more traditional formulations of the LSP allocation problem (Aneroussis and Lazar, 1996). In other words, the key elements in this optimization problem are the link congestion control and the connection's QoS guarantees expressed in the functions F and G, respectively.

From the above definitions, we can see that the cost function of each connection depends on the strategies of all the other connections and we are dealing with a noncooperative game situation. Recall that the set of players is composed of the connections belonging to the same hierarchical kth level. Hence, we are seeking a Nash Equilibrium solution in each level, starting by the highest level and working through till the lowest.

At each layer (k) of the game, we find that:

- By its definition, the game strategy space $\mathcal{C}$ is a convex, closed and bounded set.
- Furthermore, the cost function of each connection i ${J_i}^k(\mathbf{C}^k)$ is, whenever finite, continuous in ${c_i}^l$ and convex in ${c_i}^l$ for every fixed value of C^l (Lazar et al., 1995).

Therefore, based on Theorem 1, the game possesses at least one Nash Equilibrium point (Başar and Jan Olsder, 1995).

It is worthy to note that in the above formulation, a player can be also thought of as a user (a major client, or a service provider, renting a portion of the network bandwidth). Once the NEP solution is reached, the user's equilibrium strategy c_i can be shared by various types of flows belonging to different traffic classes (e.g., voice, video, etc, ...).

4 Analysis

In this section, we present the simulation results comparing the performance of the classical versus the *competitive* LSP allocation models (Fayek, 2001). For convenience, these two models are summarized here.

Revenue Model

$$(P1) \qquad \max \quad Rev = \sum_{i \in \mathcal{I}_{k_l}} \alpha_i (1 - p_i)$$

$$\begin{aligned} \text{s. t.} \quad 1. \quad & c_t{}^l = \sum_{i \in \mathcal{I}_{k_l}} c_i{}^l \le C_l && \forall l \in L \\ 2. \quad & p_i < \kappa_i && \forall i \in \mathcal{I}_{k_l} \\ 3. \quad & 0 \le c_i{}^l \le C_l && \forall i \in \mathcal{I}_{k_l} l \in L \end{aligned}$$

Game Model

$$(P2) \qquad \min \quad \sum_{i \in \mathcal{I}_{k_l}} J_i(\mathbf{C})$$

$$\text{or} \quad \min \quad \sum_{i \in \mathcal{I}_{k_l}} \sum_{l \in L} F_i{}^l\left(c_i{}^l, c_t{}^l\right) + G_i(c_i)$$

$$\begin{aligned} \text{s. t.} \quad & 1.\ c_t{}^l = \sum_{i \in \mathcal{I}_{k_l}} c_i{}^l \le C_l && \forall l \in L \\ & 2.\ 0 \le c_i{}^l \le C_l && \forall i \in \mathcal{I}_{k_l} \ \ l \in L \end{aligned}$$

where $$F_i{}^l\left(c_i{}^l, c_t{}^l\right) = c_i{}^l \cdot \frac{\alpha_i}{1 - c_t{}^l / C_l}$$

and $$G_i(c_i) = \begin{cases} \dfrac{1}{\kappa_i - p_i}, & \text{if } p_i < \kappa_i \\ \infty, & \text{otherwise} \end{cases}$$

Both models are solved in the discrete domain. That is, the $c_i{}^l$ variables take integer values only. Moreover, the *Game Model* was solved in the continuous domain ($P2'$)

There are three networks used to test the models, with three traffic sets each. The description of both the networks and their traffic sets are summarized in Tables 1 and 2, respectively.

Table 1 Networks data

Network data	Net (1)	Net (2)	Net (3)
Number of Level (1) Nodes	99	295	432
Number of HyperNodes (Fig. 4):	21	21	21
Number of switches	33	59	72
Number of External Nodes	66	236	360
Total Number of Nodes	120	316	453
Number of level (1) Edges	183	547	783
Number of HyperEdges	47	56	52
Total number of Edges	230	603	835

Table 2 Traffic data for Net (1), Net (2) and Net (3)

	Net (1)			Net (2)			Net (3)		
	T1	T2	T3	T1	T2	T3	T1	T2	T3
Level (1) connections	67	65	48	359	308	251	553	513	374
Level (2) connections	12	17	32	46	69	117	72	107	177
Level (3) connections	6	13	19	23	47	70	36	72	108
Total	85	95	99	428	424	438	661	692	659

The first network, Net(1), is depicted in Fig. 3. The two conceptual layers of hypernodes and hyperedges built on top of the physical network, Net(1), are shown in Fig. 4.

The optimization problems, $P1$, $P2$ and $P2'$, were solved using the Tomlab (Holmström, 1999) software package. Tomlab is an optimization library that runs in a Matlab environment (Holmström, 1999). The routine *glcSolve* implements an extended version of the DIRECT method presented in Jones et al. (1993). DIRECT handles problems with both nonlinear and integer constraints (Holmström, 1999).

DIRECT is a modification of the standard Lipschitzian approach that eliminates the need to specify a Lipschitz constant. Since no such constant is used, there is no natural way of defining convergence, except when the optimal function value is known. Therefore, *glcSolve* is run for a predefined number of function evaluations and considers the best function value found as the optimal one. The optimality percentage was set to 0.01%, i.e., the best solution is 0.01% from the optimal solution.

There is also the option to restart *glcSolve* with the final status of all parameters from the previous run (Holmström, 1999). In general, *glcSolve* solves global mixed-integer nonlinear programming problems of the form:

$$
\begin{array}{l}
\qquad\qquad \min_x f(x) \\
\text{subject to:} \\
\begin{array}{rccl}
x_L \leq & x & \leq & x_U \\
c_L \leq & c(x) & \leq & c_U \\
b_L \leq & Ax & \leq & b_U \\
x_i & \text{integer} & & i \in \text{Integers}
\end{array}
\end{array}
$$

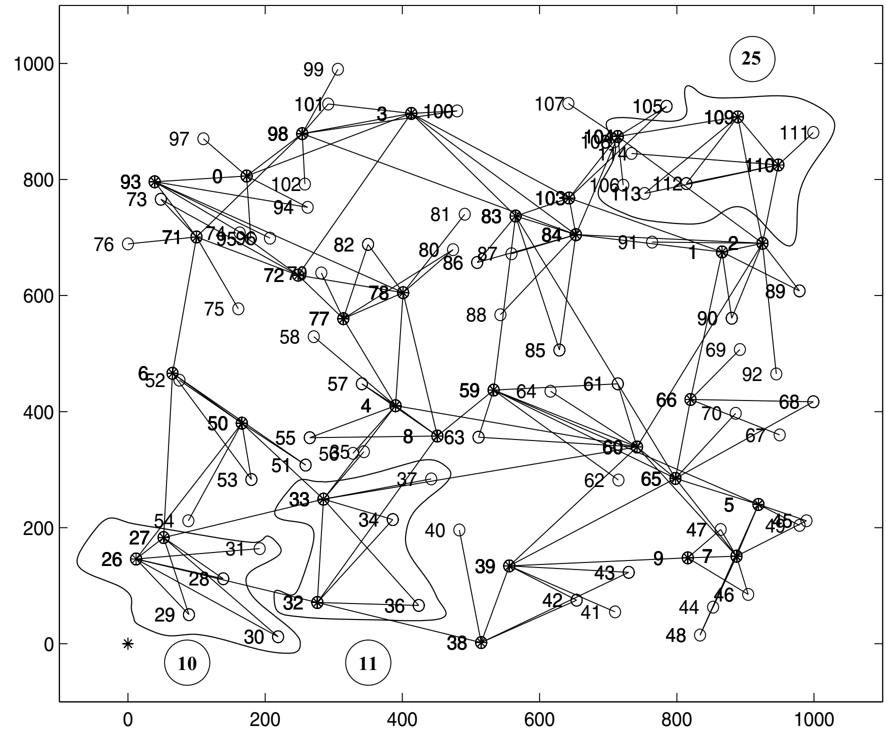


Fig. 3 Network (1), with 16 hypernodes (nodes (10) to (25))

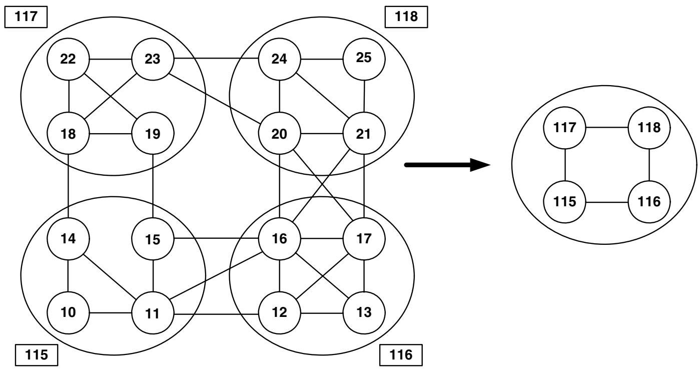


Fig. 4 First and second layers on top of the physical network Net (1) (Fig. 3)

where $x, x_L, x_U \in \mathcal{R}^n$, $c(x), c_L, c_U \in \mathcal{R}^{m_1}$, $A \in \mathcal{R}^{m_2 \times n}$ and $b_L, b_U \in \mathcal{R}^{m_2}$.

For each network/traffic-set pair, the *glcSolve* was run 10 times with “warm-start-ups”. That is the starting point was deduced from the previous runs. When the best value for the objective function stops improving, the runs would stop indicating the best solution obtained. The optimality percentage was set to 0.01%, i.e., the best solution

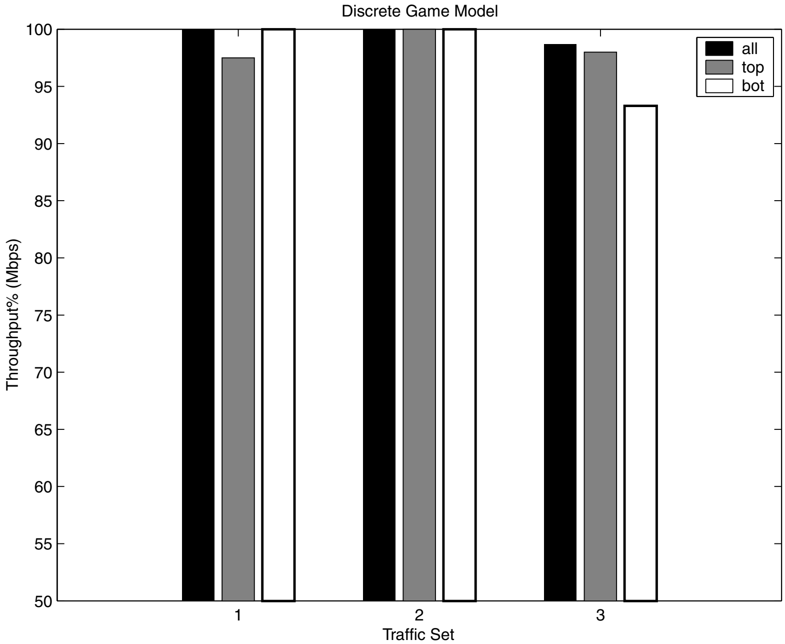


Fig. 5 Throughput values for Net (1) using the discrete game model

is 0.01% from the optimal solution. In most cases, when a feasible solution exists, the average number of runs did not exceed 5 times.

When a final configuration for LSP allocation was reached, the following performance measures were of interest: the total network throughput, the link utilization distribution and the average run-time values for each model. The throughput is calculated by:

$$\mathcal{TH} = \left(1 - \frac{\text{Blocked requests (Mbps)}}{\text{Total bandwidth request (Mbps)}}\right) \times 100\% \tag{7}$$

Equation (7) measures the throughput as the ratio of the total bandwidth that is routed (not blocked) through the network to the total bandwidth requests. The revenue generated is hence proportional to this quantity.

In Figs. 5 to 7 for each model, discrete revenue, discrete game and continuous game, three flavours of optimization orders were used. Namely, fair optimization, in which all connection requests are considered simultaneously during optimization, i.e., finding a Nash equilibrium (or equilibria) in the case of game models. This first approach is labeled “all”. The second approach, labeled “top”, is a Stackelberg model in which a multi-level optimization is used. It begins with connection requests with higher priorities as the leaders and proceeds through the prioritized list of connections until they are all considered. The last approach, labeled “bot” for bottom-up, is also a multi-level optimization with reversed order, i.e., the Stackelberg model treats the lowest priority connections as the leaders of the game. In the case of the Revenue

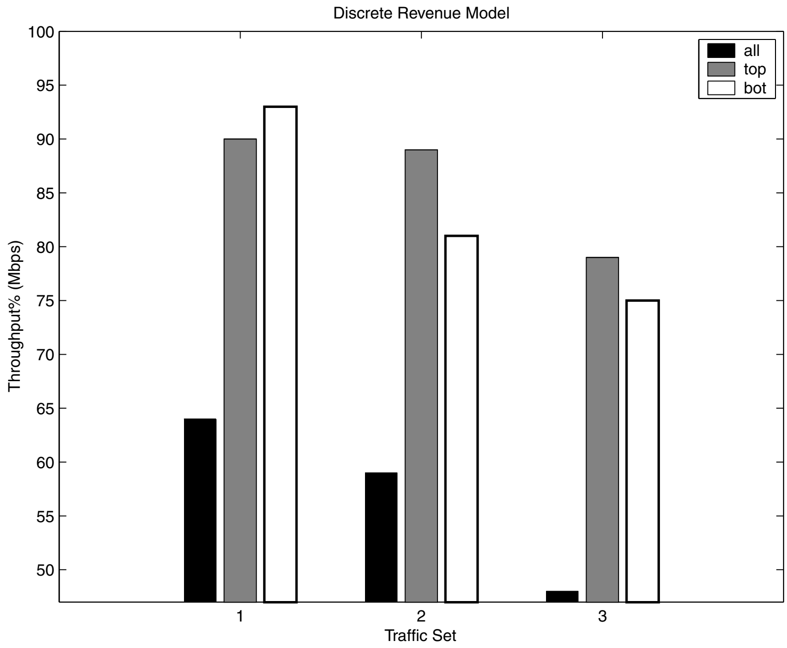


Fig. 6 Throughput values for Net (1) using the discrete revenue model

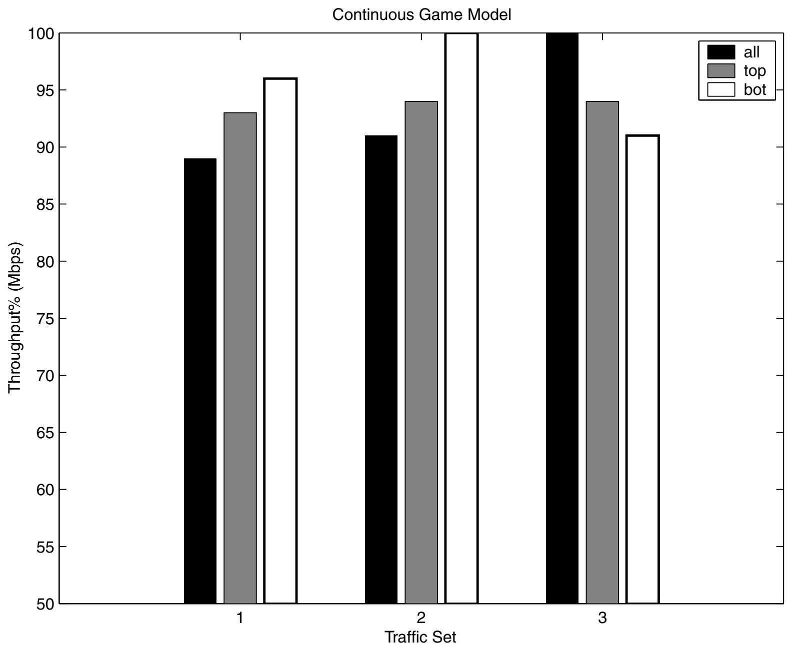


Fig. 7 Throughput values for Net (1) using the continuous game model

model, the same prioritization scheme was used as in the game models for the sake of comparison.

The link utilization is a performance measure that indicates the flow distribution in the network. The "smoother" the distribution is, the less the tendency of the *critical* links to become congested. In other words, when the network links are not loaded evenly, some links will service greater flow demands than others and will be prone to congestiont. The traffic distribution in this case will show spikes at those overloaded links. Figure 8, gives an illustrative description of the links utilization for Network (1) and using the Top-Down approach in all three cases. When it comes to the overall links utilization, the continuous model was found to produce the smoothest distribution among the three models. This is to be expected since the capacity allocated to each LSP is not restricted to take on integer values. Therefore, any connection is routed over more links than in the other two discrete cases. The discrete revenue model comes in second place in that regard. Naturally, as the traffic load increases, the links utilization distribution gets smoother for all three models.

The run-time for the experiments conducted for Net (1) are shown in Fig. 9. The following can be observed in that figure:

- In all three cases, the run-time of the "All" order of optimization is greater than the other two.
- The run-time of both the "Top" and "Bot" techniques are similar more or less.
- The revenue model is much more time-consuming than the discrete and continuous game models.

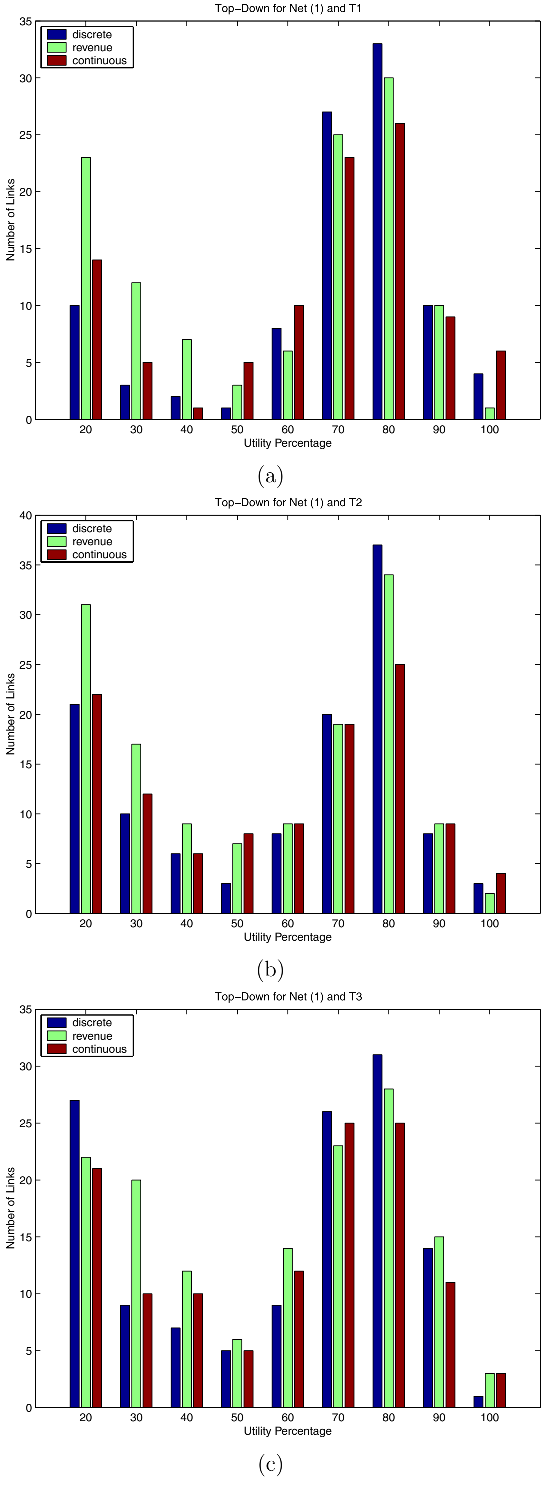
It is worth noting that approximately 80% of the run-time recorded here is due to file read/write operations. The interface between the Tomlab solver (in Matlab) and the LSP allocation engine (in a C++ environment) was through input/output files.

The following results are related to the second and third networks. In the case of the continuous game model, the number of variables was prohibitively large to the point that the program could not be run to completion as it ran out of memory in some situations. In some other situations, the discrete revenue model failed to produce more than 50% throughput. The results of these runs were discarded from the comparison of link utilizations.

In Fig. 10, the throughput values under different load conditions are compared for each mathematical model. As can be verified from the figure, the "Top-Down" order of optimization faired better than the "All" approach and better or equal to the "Botton-Up" approach in both discrete models. In the continuous case, however, the "Bottom-Up" approach was found to be better or equal to the "Top-Down" approach.

If the throughput comparison is based on the "Top-Down" approach only between the three mathematical models, it can be observed, from Fig. 11, that the discrete game model is consistently superior to the other two models under any load condition. From the experiments, we observed that the amount of routing information was larger in the Continuous Game model which unavoidably worsened the algorithm performance. In the case of the Revenue model, the nature of the optimization problem affected the performance. The Revenue model is a

Fig. 8 Link utilization for Top-Down approach when used for Net (1) and under traffic loads: (a) T1, (b) T2 and (c) T3



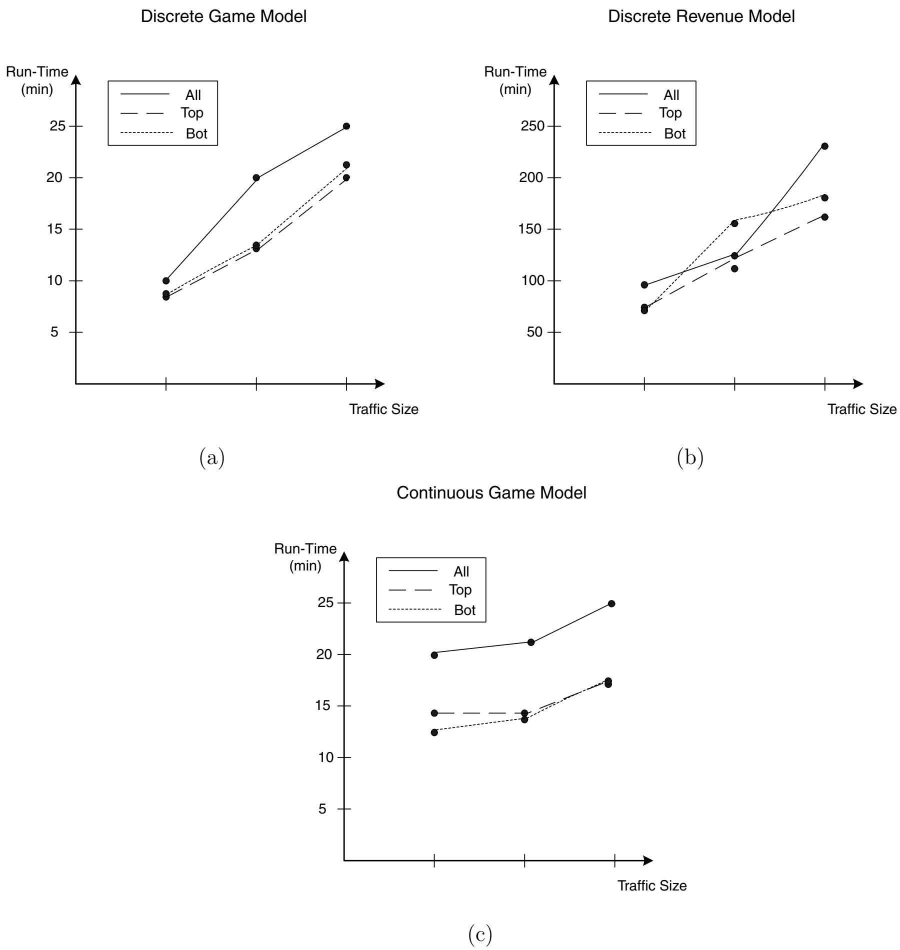


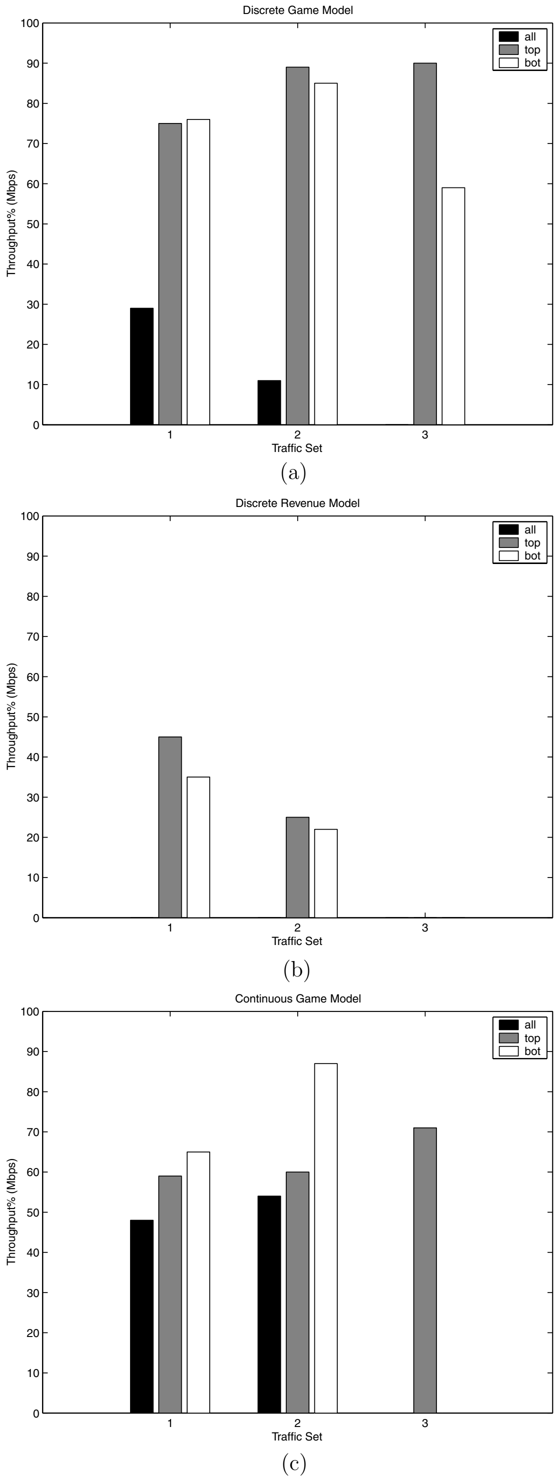
Fig. 9 Run-Time values for: (a) Discrete Game model, (b) Discrete Revenue model and (c) Continuous Game model

maximization-based objective function subject to non-linear constraints. In contrast, the Discrete Game model, the non-linearity of the constraints is transported into the objective function. The minimization-based objective is now subject to linear constraints.

The link utilizations are compared and Fig. 12 outlines this comparison. As in the previous scenario, the continuous model offered a smoother distribution than the discrete game model. The discrete revenue model values can be discarded since some of them resulted in less than 50% throughput. Therefore they do not reflect the real figure of the total number of links utilized nor the utilization distribution, for that matter.

The run-time for the experiments conducted for Net (2) are shown in Fig. 13. Due to the fact that the "A" experiment in the continuous case under traffic load T3 did not run to completion, only the run-time data for the discrete models is shown in the figure.

Fig. 10 Throughput values (Eq. (7)) for Net (2) using: (a) the discrete game model, (b) the Discrete Revenue model and (c) the Continuous Game model



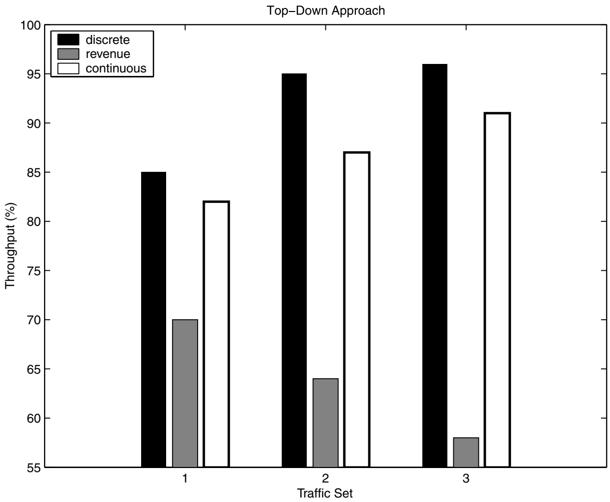


Fig. 11 Throughput values for Net (2) when using the Top-Down Approach and under increasing traffic demands

As is observed from Fig. 13(b), the sequential operation of the proposed system under the discrete revenue model is prohibitively time-consuming, whereas, for the same problem size, the discrete game model exhibited, not only better throughput, but also a considerably faster operation, especially when using the multi-level optimization approach versus the single-level counterpart.

Finally, for the largest network, Net (3), based on the experimental data of the previous two networks, only the discrete game model was considered and used in the experiments shown in Fig. 14. In general, it can be observed that the overall performance in terms of throughput becomes significantly poorer with the increased network size.

Figure 14 demonstrates the results obtained when comparing the three different orders of optimization. It can be noticed that the results for the “All” approach and the traffic set T3 are not reported. This is due to the problem size; a run-time error of insufficient memory resulted in each time this experiment was conducted. Due to the network size and the different traffic sizes, the “All” approach is deemed inadequate. This is due to the fact that, at each iteration, it optimizes the resource distribution among all traffic classes combined. Whereas, the “Top-down” or “Bottom-up” approaches proceed in a prioritized fashion, optimizing resources for the flows belonging to a certain traffic class, one class at a time.

Once again, it can be observed that both the “Top-down” and “Bottom-up” approaches are equally good in terms of throughput, with the “Bottom-up” being better in the low traffic loads and the “Top-down”, on the other hand, being better in heavy traffic load conditions. The reason for this observation is that, in low traffic loads, the lower-priority flows are considered first by the optimization algorithm which ensures they get serviced in the beginning. Whereas in the “Top-down”, these lower-priority flows can experience a percentage of denied-access thus increasing their blocking rate. On the other hand, in high traffic loads, the majority of flows belong to the lower-priority

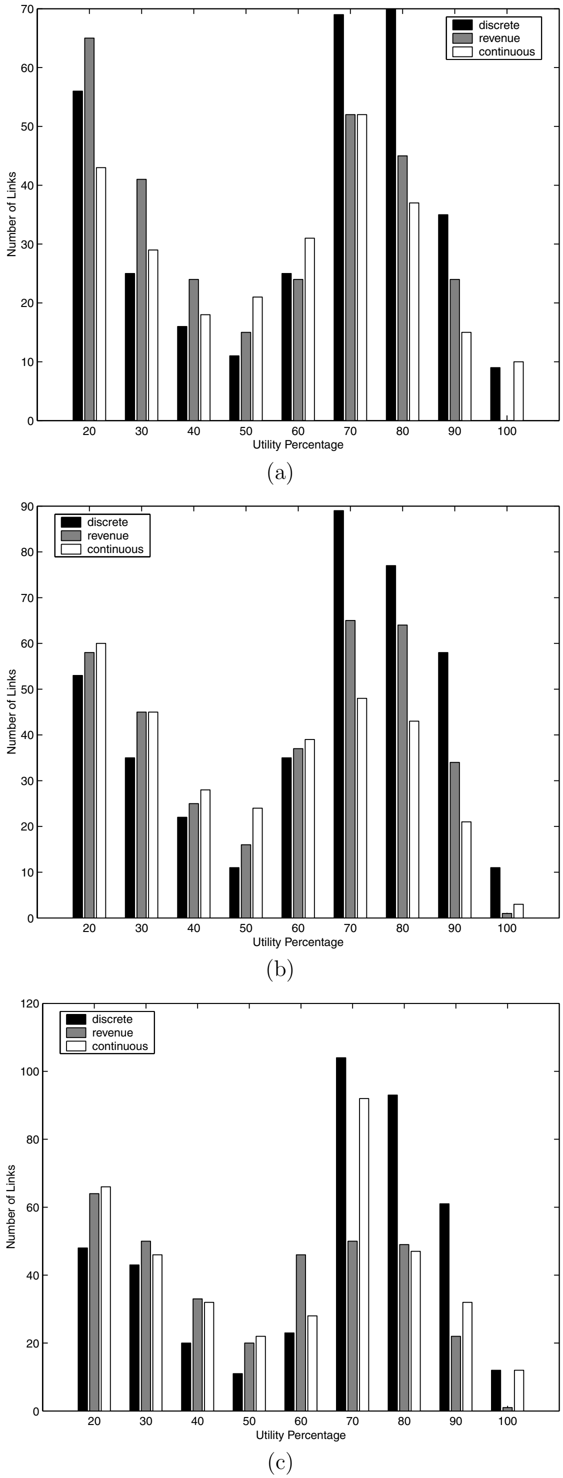


Fig. 12 Link utilization for Top-Down approach when used for Net (2) and under traffic loads: (a) T1, (b) T2 and (c) T3

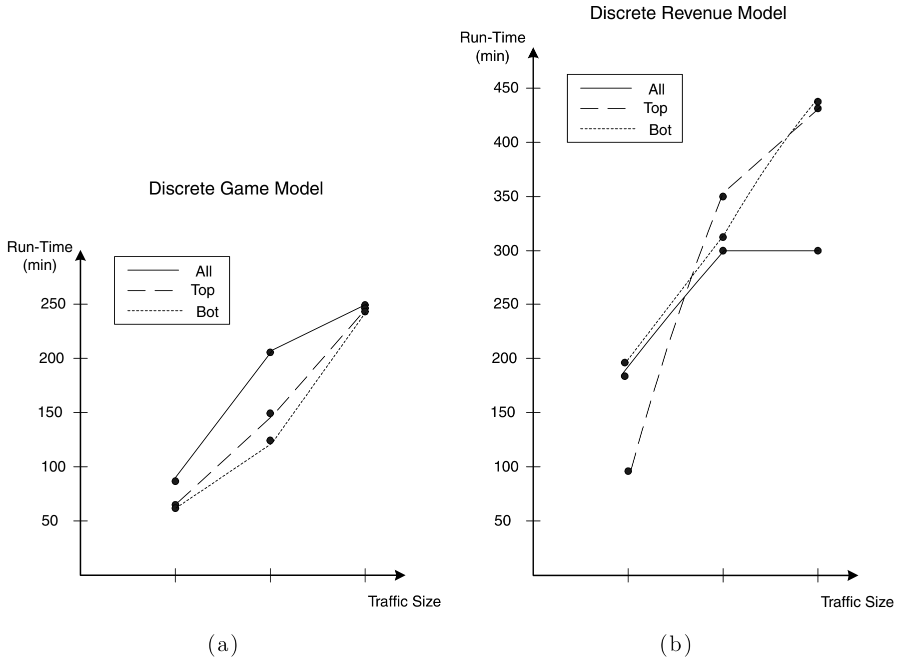


Fig. 13 Run-Time values for: (a) Discrete Game model and (b) Discrete Revenue model

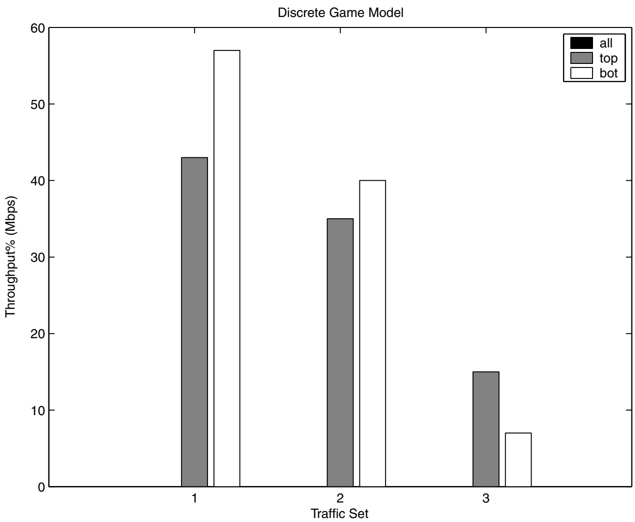


Fig. 14 Throughput values for Net (3) (Eq. (7))

classes. Since the higher priority flows are serviced first in the "Top-down" approach, the overall blocking rate is reduced since the high-volume, low-priority flows do not start by monopolizing the network resources as they would have in a "Bottom-up" approach.

5 Conclusion

In the present work, we presented a description of the problem of Label-Switched Paths dimensioning in large-scale MPLS networks. The problem formulation for this work depended on a game-theoretic framework. In this framework, the requests for bandwidth, i.e., the connections, were modeled as non-cooperative players competing for the network resources. Two types of game models were used, namely the fair game or Nash model and the prioritized game or Stackelberg model. The optimization problem formulated after each of these models was solved in the discrete and continuous domain. In addition, this approach to LSP dimensioning was compared with the classical approach in which some network revenue function is maximized.

A summary of the experimental results of the previous section is given here:

- Under any traffic load and using any of the three described mathematical models, the multi-level optimization was found to exhibit consistently a better performance in terms of higher throughput and lower run-time values than the single-level optimization.
- Among the three mathematical models, namely, the discrete game model, the discrete revenue model and the continuous game model, the first model was found to be superior to the other two.
- If the single-level approach is discarded from the comparison, then it can be deduced from the previous analysis that:

  1. Under low traffic loads, the "Bottom-Up" approach behaves better than the "Top-Down" approach for the reasons discussed previously.
  2. As the traffic size increases, the "Top-Down" approach performs better than the "Bottom-Up".

  Therefore, when solving the LSP dimensioning problem of a network under moderate to high traffic conditions, it is recommended to use the discrete game model with a "Top-Down" order of optimization.

References

Ahn S, Tsang RP, Tong S, Du DHC (1994) Virtual path layout design on ATM networks. In: Proceedings of IEEE INFOCOM'94, pp 192–200

Almendral LL, Fernandez JA, Cholvi V, Sanjuan MAF (2004) Oblivious router policies and Nash equilibrium. In: International symposium on computers communication volume 2, pp 736–741

Anandalingam K, Nam G (1997) Conflict and cooperation in designing international telecommunication networks. J. Oper. Res. Soc. 48(6):600–611

Aneroussis NG, Lazar AA (1996) Virtual path control for ATM networks with call level quality of service guarantees. In: Proceedings of IEEE INFOCOM'96

Aresti A, Ninan BM, Devetsikiotis M (2004) Resource allocation games in connection-oriented networks under imperfect information. In: IEEE international conference on communications (ICC), pp 1060–1064

Awduche D, Rekhter Y (2001) MultiProtocol lambda switching: Combining MPLS traffic engineering control with optical crossconnects. IEEE Communications Magazine 39(3):111–116

Awduche D et al. (2000) RSVP-TE: Extensions to RSVP for LSP Tunnels. Internet Draft draft-ietf-mpls-rsvp-lsp-tunnel-07.txt

Aydemir M, Viniotis Y (1996) Deterministic algorithm for VP assignment in ATM networks. Comput Commun 19:1036–1050
Başar TJ, an Olsder G (1995) Dynamic Noncooperative Game Theory. Academic press
Blefari-Melazzi N, Femminella M, Reali G (2000) Dynamic bandwidth allocation in a circuit-switched network: provision of deterministic and statistical QoS guarantees. In: INFOCOM
Brockmeyer E, Halstrom HL, Jensen A (1948) The life and Works of A. K. Erlang. Academy of Technical Sciences, Copenhagen.
Campos-Nanez E, Patek SD (2003) On-line tuning of prices for network services. In: IEEE INFOCOM volume 2, pp 1231–1241
Chlamtac I, Faragó A, Zhang T (1993) How to establish and utilize virtual paths in ATM Networks. In: Proceedings of IEEE ICC'93, pp 1368–1372
Courcoubetis C, Dimakis A, Reiman M (2001) Providing bandwidth guarantees over a best-effort network: call-admission and pricing. In: INFOCOM
Fayek D (2001) Hierarchical virtual paths allocation in large-scale ATM networks using noncooperative game models. Ph.D. thesis, Electrical and Computer Engineering, University of Waterloo
Fayek D, Kesidis G, Vannelli A (1999) Hierarchical virtual path allocation in larg-scale ATM networks using noncooperative game models. In: Canadian Conference on Broadband Research (CCBR) Ottawa, pp 222–233
Gerstel O, Cidon I, Zaks S (1996) The layout of virtual paths in ATM networks. IEEE/ACM Trans Netw 4(6):873–884
Ghizzi M, Cerutti I, Castoldi P, Fumagalli A (2004) Performance of label stacking capable MPLS reconfigurable networks. In: IEEE workshop on local and metropolitan area networks, pp 129–132
Gibbons A (1985) Algorithmic graph theory. Cambridge University Press
Goyal S, Bellur U (2005) Mapping application QoS to network configurations for MPLS networks. In: Consumer Communications and Networking Conference, CCNC pp 562–564
Hadama H, Kawamura R, Izaki T, Tokizawa T (1994) Direct virtual path configuration in large-scale ATM networks. In: Proceedings of IEEE INFOCOM'94, pp 201–207
Holmström K (1999) TOMLAB User's Guide http://www.ima.mdh.se/tom
Hussain I (2004) Overview of MPLS technology and traffic engineering applications. In: Internation conference on networking and communication, ICNN
Jin Y, Kesidis G (2003) Nash equilibria of a generic networking game with applications to circuit-switched networks. In: INFOCOM, pp 1242–1249
Jin Y, Kesidis G (2005) Dynamics of usage-priced communication networks: the case of a single bottleneck resource. IEEE Transactions on Networking
Jones DR, Perttunen CD, Stuckman BE (1993) Lipschitzian optimization without the Lipschitz constant. J Optim Theory Appl 79(1):157–181
Joshi MC, Bose RK (1985) Some topics in nonlinear functional analysis. John Wiley, New York
Kelly FP (1991) Loss networks. The ann Appl Prob 1(3):319–378
Korilis YA, Lazar AA, Orda A (1997a) Achieving network optima using stackelberg routing strategies. IEEE/ACM Trans Netw 5(1) 161–173
Korilis YA, Varvarigou TA, Ahuja SR (1997b) Incentive-compatible pricing strategies in noncooperative networks. In: INFOCOM
Lazar AA, Orda A, Pendarakis DE (1995) Virtual path bandwidth allocation in multi-user networks. In: Proceedings of IEEE INFOCOM'95, pp 312–320
Lee GM, Choi JK (2001) A study of flow-based traffic admission control algorithm in the ATM-based MPLS network. In: 15th International conference on information networking, pp 213–218
Lim JY, Chae KJ (2001) Differentiated link based QoS routing algorithms for multimedia traffic in MPLS networks. In: Information networking, pp 587–592
Lin F, Cheng K-T (1993) Virtual path assignment and virtual circuit routing in ATM Networks. In: GLOBECOM'93, pp 436–441
Liu Y, Simaan MA (2005) Noninferior nash strategies for routing control in parallel-link communication networks. In: IEEE consumer communications and networking conference, CCNC, pp 510–514
Marbukh V (2001) Minimum regret approach to network management under uncertainty with application to connection admission control and routing. In: Internation conference on networking (ICN), pp 309–318
Marbukh V, Moayeri N (1999) A framework for throughput and stability analysis of a DS-CDMA network. In: IEEE-VTC, pp 596–600

Medrano MS, Trindade MB, De Chaves NSA, Fernandez MD, Filho HJM (2004) An optimization model for MPLS networks. In: IEEE International symposium on telecommunications network strategy and planning, pp 285–290

Messerli EJ (1972) Proof of a convexity property of the Erlang B formula. Bell Syst Technol J 51(4):951–953

Rosen E, Viswanathan A, Callon R (2001) Multi-protocol label switching architecture. IETF RFC 3031 rfc3031.txt

Ross KW (1995) Multiservice loss models for broadband telecommunication networks. Springer-Verlag

Shi TJ, Mohan G (2004) An efficient traffic engineering approach based on flow distribution and splitting in MPLS networks. In: IEEE International conference on networks, ICON, pp 99–103

Smart DR (1974) Fixed point theorems. London: Cambridge University Press

Sullivan E, Callon R (1994) P-NNI draft specification. ATM Forum

The ATM Forum (1996) Private network-network interface specification version 1.0. The ATM Forum

Trimintzios P, Pavlou P, Flegkas G, Georgatsos P, Asgari A, Mykoniati E (2003) Service-driven traffic engineering for intradomain quality of service management. IEEE Network 17(3):29–36

Wolff RW (1989) Stochastic modeling and the theory of queues. Englewood Cliffs, N.J.: Prentice Hall

Xiao X, Hannan A, Bailey B, Ni LM (2000) Traffic Engineering with MPLS in the Internet. IEEE Network 14(2):28–33

Xiao X, Ni LM (1999) Internet QoS: a big picture. IEEE Network, pp 8–18

Xinjie C, Subramanian KR (2000) An optimal admission control scheme for the wireless ATM networks. In: NOM2000

Yaïche H, Mazumdar R, Rosenberg C (2000) Distributed algorithms for fair bandwidth allocation to elastic services in broadband networks. In: INFOCOM