Dynamic Wavelength Allocation in IP/WDM Metro Access Networks

Emre Yetginer, *Student Member, IEEE,* and Ezhan Karasan, *Member, IEEE*

***Abstract*— Increasing demand for bandwidth and proliferation of packet based traffic represent a challenge for today's metro networks, which have been traditionally designed to carry circuit-switched connections. The problem is further complicated by the constraints of cost efficiency and traffic adaptability, imposed by the limited customer base in the metro area. Recently, several architectures have been proposed for future metro access networks. Nearly all of these solutions support dynamic reconfigurability, however reconfiguration policies have not been fully explored yet. In this paper, reconfiguration policies for IP/WDM metro access networks with switching delays are considered, where dynamic reconfiguration corresponds to dynamic allocation of wavelengths to access nodes. Exact formulation of the dynamic wavelength allocation (DWA) problem is developed as a Markov Decision Process (MDP) and a new cost function is proposed to attain both throughput efficiency and fairness. For larger problems, a heuristic approach based on first passage probabilities is developed and shown to yield nearly optimum performance through simulations.**

***Index Terms*— Metro access networks, IP over WDM, dynamic wavelength allocation, reconfiguration, Markov Decision Process, switching delay.**

I. Introduction

THE steady increase of the Internet traffic has caused architectural and conceptual changes in communication networks. The infrastructure, once designed to carry legacy voice services, is no longer able to put up with this ever increasing packet-based traffic. Long-haul backbone networks have been adapted to this change using the optical transmission technology and dense wavelength division multiplexing (DWDM). In the future, core networks are expected to evolve towards a fully optical transport network architecture [1]. Meanwhile, in the access side service rates have increased to megabits level. With the penetration of optical fibers down to the premises of end users, the target is to offer gigabit per second rates directly to the customers [2]. But, metro networks, that are in between access and core networks lag behind in terms of speed and capacity. Hence, they constitute a barrier for this large volume of traffic to be transmitted from access networks to the high speed backbone networks. The pressure from the access side forces metro networks, most of which still rely on legacy time division multiplexing (TDM) based technologies, into an evolutionary process [3]. High capacity, protocol transparency, cost efficiency and dynamic traffic adaptability are major requirements in this transformation [4].

The proximity of metro networks to the end users differentiates them from core networks. Large volumes of traffic aggregation in core networks results in slowly changing and hence to a large extent stable and predictable traffic patterns [5]. Therefore, static design of the logical topology and over-provisioning the capacity to handle traffic uncertainty prove to be sufficient. Reconfiguration of these networks is mostly manual and requires a time duration in the order of hours or days. However, this is not a concern since reconfiguration is required only in case of large and persistent demand deviations, such as the addition of new nodes to the network or network failures. On the other hand, metro access networks serve to a limited number of users. Hence, traffic variability is naturally expected. Since each node of a metro access ring serves a different district of a town, it is possible to see nearly periodic oscillations in the traffic demand [6], [7]. These variations may occur at different time scales. Traffic patterns may change on a daily basis, e.g., in weekdays and weekends different portions of the network may become congested. During working hours, hot spots may shift from residential areas to business districts, corresponding to a traffic variation in the order of hours. At the extreme case, where traffic aggregation is very low, individual flows corresponding to high-speed transactions may cause more frequent fluctuations.

These reasons, together with the cost constraints, necessitate a high level of traffic adaptability in metro access networks. Most of the solutions designed for future metro access networks (e.g., Next Generation SONET (NGS) [8], IP/WDM [9]) already support dynamic reconfiguration. Likewise, reconfigurability is also possible for Ethernet Passive Optical Networks (EPON) that are seen as a promising technology for future access networks [10]. However, development of the methods that can be used for dynamic reconfiguration is still an open research problem.

Dynamic resource allocation has been studied for various problems and under different settings. There are several work in the context of polling systems, where the problem is to find an optimum schedule for a single server to visit several queues. For the case without switching overheads, the optimum solution is the cμ-rule [11], which gives preemptive priority to the queue for which the product of the holding cost (c) and the service rate (μ) is largest. But no general result is available for the case with switching overhead. The problem studied in this paper contains multiple servers (wavelength channels) and preemption is not applicable since it is necessary to preserve connectivity at all times.

A heuristic reconfiguration policy for IP/WDM access networks is proposed in [12]. The main idea is to reconfig-

Manuscript received August 3, 2007; revised February 7, 2008.

The authors are with the Department of Electrical and Electronics Engineering, Bilkent University, Bilkent 06800, Ankara, Turkey (e-mail: {yetginer, ezhan}@ee.bilkent.edu.tr).

Digital Object Identifier 10.1109/JSAC-OCN.2008.031407.

ure wavelengths to balance the load on each wavelength. Simulation results obtained for instant switching case show considerable improvement compared to static allocation. In this paper, we provide an exact formulation for a similar problem considering also the effects of switching delay and using the information about the flow arrival and departure processes. We also propose a heuristic method and compare it with the algorithm presented in [12].

A different approach to the dynamic wavelength allocation (DWA) problem in single-hop broadcast lightwave networks is given in [13], where the reconfiguration problem is decomposed into two subproblems: "How to reconfigure" and "When to reconfigure". Given the traffic demand between each pair of nodes in the network, the first question is answered using the Generalized Longest Processing Time (GLPT) algorithm. The second part of the problem is solved based on the concept of "degree of load balancing (DLB)" which measures the distance of the system to the ideally balanced situation. Assuming that DLB changes over time according to a Markovian process, the problem is modeled as a Markov Decision Process (MDP), the solution of which yields a threshold type policy. Our approach differs from this work in several aspects. First, we try to jointly solve the problems of "How to reconfigure" and "When to reconfigure". Besides, a network where the number of channels is smaller than the number of nodes is studied in [13], hence a single channel is time-shared between multiple nodes. In the DWA problem studied here, each node is assigned one or more channels which are not shared with other nodes. Moreover, in [13], a reconfiguration action may result in multiple switches which take the network into the most balanced state possible. The methods considered in this paper aim to gradually improve the load balance in the network by making a single switch at each decision instant, thus limiting the effect of switching overhead on the network performance. In [13], the cost of reconfiguration is calculated based on a pre-defined function using the number of re-tunings as a parameter, whereas in this paper the reconfiguration cost is modeled as loss of service during the switching delay.

A conceptually similar resource allocation problem is studied in the context of computing grid architectures in [14]. In that work, resources are the servers and competing jobs join separate queues based on their type. The available servers are grouped into clusters to serve different types of jobs. The servers are dynamically switched between clusters in order to minimize the holding cost of jobs in the system. A dynamic programming formulation is developed for the problem and optimal switching policies are obtained. A heuristic method is also proposed and shown to produce efficient resource utilization compared to static allocation.

In this paper, we develop an MDP model for the DWA problem with switching delays. We also propose a new cost function to be used with this model and compare it with cost functions available in the literature. Through simulations, it is shown that this new cost function effectively combines the throughput and fairness objectives. The results for a 3-node network suggest that it is possible to obtain 25% to 35% improvement in throughput with respect to the static allocation and a significant gain in fairness. For larger networks, we introduce a heuristic approach based on the proposed cost function and utilizing first passage probabilities. The heuristic inherently takes into account the switching delays associated with the reconfiguration actions. It is shown that the heuristic method performs close to the optimum policy in terms of throughput. The optimality gap is below 5% for moderate load and it decreases further as the network load increases. Comparisons with other heuristics available in the literature demonstrate the effectiveness of the proposed method for different network loads, average flow sizes and under non-stationary traffic conditions.

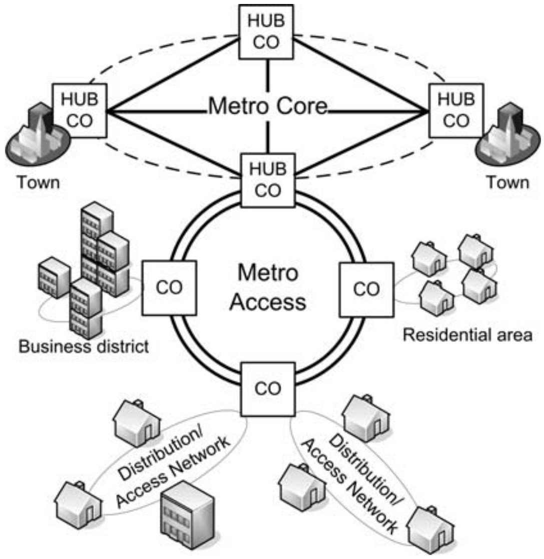


Fig. 1. IP/WDM network architecture

The rest of the paper is organized as follows. In Section II, IP/WDM metro access network architecture and DWA problem are introduced. In Section III, MDP formulation is presented along with the numerical results obtained with different cost functions. The heuristic reconfiguration policy is developed in Section IV and compared with other heuristic approaches in Section V.

II. DWA Framework for IP/WDM Metro Access Networks

Traditional metro networks have been built using a two-level hierarchy comprising metro access and metro core [9], as shown in Fig. 1. Metro access is also called as collector ring or metro edge, and spans a distance of 20-65 km. The traffic from the last mile networks and business is collected via distribution networks and aggregated at the central offices (COs). Metro access network connects these COs to each other and to the metro core through hub COs. Traffic in the metro edge has a hubbed traffic pattern and rings are natural choices of implementation in this part of the network [4]. Metro core, also known as regional network, in turn provides the connectivity between the hub COs and to the long haul backbone.

An IP/WDM metro access network consists of a single feeder ring with up to 10-20 access nodes (AN) each located at a CO. The ring may be implemented using a single fiber or multiple fibers, where each fiber supports tens of wavelengths. Last mile networks and high speed customers are connected to the feeder ring through ANs. The feeder ring itself is connected to the metro core network through a hub node located at the Hub CO. This hub node is responsible for the resource management of the ring. It allocates separate wavelength channel(s) to access nodes. Each AN aggregates traffic from distribution networks and transmits it to the hub node on the wavelengths assigned to itself. Finally, the hub node forwards the traffic to the metro core network. For the downlink case, the traffic follows the reverse path.

Each AN consists of an IP router and an optical add drop multiplexer (OADM). AN is also equipped with tunable receivers and transmitters, so that wavelengths assigned to ANs can be changed dynamically by tuning these transmitters and receivers in order to support DWA.

For the IP/WDM network under consideration, resource management corresponds to the allocation of wavelengths to access nodes. In the simplest case, wavelengths can be assigned to access nodes based on traffic forecasts and are not changed in time, which is called "static allocation". For instance, if all the nodes have the same expected offered load, then the wavelengths should be evenly distributed between nodes. But if the traffic uncertainty or variability is high, static allocation strategy will be inefficient and possibly unfair. In that case, it is a better idea to change the number of wavelengths assigned to each node dynamically ("dynamic allocation") to follow the traffic fluctuations.

Dynamic allocation has an overhead due to signalling requirements, reconfiguration of OADMs and tuning latencies of the transmitters and receivers. During this delay period, called the *switching delay*, and denoted by τ, the wavelength channel being switched becomes unavailable. Hence, there is a tradeoff between the switching costs and the responsiveness of the network to the traffic changes.

In the formulation of the DWA problem, several assumptions and simplifications are made in this paper. First, only the traffic in one direction is considered. This is not a restrictive assumption because traffic to and from the hub node is transmitted on independent set of wavelengths. Hence a similar formulation can be used for the traffic in reverse direction. Secondly, the local traffic between ANs is neglected. This is a realistic assumption since observations within access networks have shown that approximately 90% of data traffic is originated at or destined for points outside the network [12]. It is also assumed that the number of wavelengths is greater than the number of ANs in the feeder ring, which is an operational requirement for wavelength routed IP/WDM networks. Moreover, due to connectivity requirements, each node must be assigned at least one wavelength at any time.

Reconfiguration actions are only allowed at the flow arrival and departure times. For simplicity, at each reconfiguration epoch at most one wavelength switch is permitted. Moreover, if the switching of a wavelength has not been completed, another switch cannot be initiated. Thus, at any time at most one wavelength can be in the switching state.

At each AN, a packet scheduler is used, so that each of the flows at a node uses a fair share of the bandwidth available at that node, and a flow may use the capacity of multiple wavelengths. With this assumption, the total bandwidth allocated to a node can be seen as a single channel with an aggregated capacity of all channels. Moreover, flows are assumed to be elastic, i.e., they do not have any peak rate, so that they can efficiently utilize the available capacity.

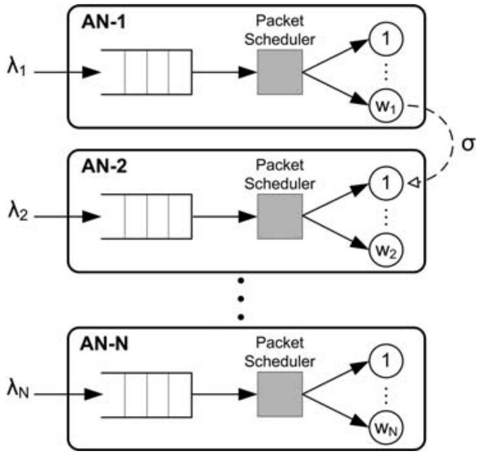


Fig. 2. Logical view of the IP/WDM network

Under these assumptions, the DWA problem can be stated as the maximization of the network efficiency by deciding on a reconfiguration action at each flow arrival or departure event. The action may be to keep the current wavelength allocation intact or to change the allocation by switching a single wavelength between a pair of nodes.

III. Exact Solution of the DWA Problem

Exact solution of the DWA problem can be obtained by modeling the system as a Markov chain whose state transition probabilities depend on the reconfiguration actions taken. The resulting process is called a Markov Decision Process (MDP) [15] and can be solved using numerical techniques, such as value iteration.

A. MDP Model

The abstract view of the network used in MDP model development is illustrated in Fig. 2. The network has N nodes and W wavelengths where W is assumed to be larger than N. To each node i, flows arrive according to an independent Poisson process with rate λ_i. Flow sizes at node i are exponentially distributed with mean B/μ_i, where B is the bandwidth of a single wavelength channel. Total bandwidth allocated to node i is the product of the number of channels assigned, w_i, and B. Hence, service rate of flows at node i is $w_i \times \mu_i$. The switching delay, τ, is exponentially distributed with mean $1/\sigma$.

The state of the network, $s \in S$, can be represented by the triplet $s = (\boldsymbol{f}, \boldsymbol{w}, k)$. $\boldsymbol{f} = [f_i]$ is the flow vector, where f_i is the number of flows at node i, $\boldsymbol{w} = [w_i]$ is the wavelength vector, where w_i is the number of wavelengths allocated to node i, and k indicates the node to which a wavelength is currently being switched. If no switching action is underway, k is 0. Valid states are $s = (\boldsymbol{f}, \boldsymbol{w}, k)$ such that $I_+(k) + \sum_i w_i = W$, where $I_+(k) = 1$ if $k > 0$, $I_+(k) = 0$ otherwise.

A_s is the action space consisting of the valid actions that may be taken at state s. Since at most one switch at a time is allowed, if there is already a wavelength being switched, i.e., if $k > 0$, $A_s = \{a_0\}$, where a_0 corresponds to no-switching.

Otherwise, A_s consists of a_0 and subset of actions a_{lm}, which correspond to switching one wavelength from node l to node m, such that node l has more than one wavelength allocated. That is,

$$A_s = \begin{cases} \{a_0\}, & \text{if } k > 0 \\ \{a_0\} \cup \{a_{lm} \mid w_l > 1\}, & \text{if } k = 0. \end{cases} \tag{1}$$

Transition rates from state $s = (\boldsymbol{f}, \boldsymbol{w}, k)$ to state $s' = (\boldsymbol{f'}, \boldsymbol{w'}, k')$ under action a is denoted as $q_{ss'}(a)$. Transition rates when no switching is performed are given as

$$q_{ss'}(a_0) = \begin{cases} \lambda_i, & \text{if } f' = f + e_i \\ w_i\mu_i, & \text{if } f' = f - e_i \\ \sigma, & \text{if } k > 0 \text{ and } k' = 0 \text{ and } w' = w + e_k \\ 0, & \text{otherwise,} \end{cases}$$

where e_i denotes the unit vector which has 1 in position i and zeros elsewhere. When the action is to switch a wavelength from node l to node m (a_{lm}), there is an instant transition from state $s = (\boldsymbol{f}, \boldsymbol{w}, k)$ to state $s' = (\boldsymbol{f'}, \boldsymbol{w'}, k')$, with $w' = w - e_l$ and $k' = m$.

The cost function is represented as $g(s, a)$, where s is the state and a is the action. It defines the cost per unit time, depending on the state of the system and possibly the action taken. The objective of the MDP is to minimize the infinite horizon total discounted cost defined as:

$$\lim_{n \to \infty} \mathrm{E}\left\{ \int_0^{t_n} e^{-\beta t} g\left(s\left(t\right), a\left(t\right)\right) \mathrm{d}t \right\},$$

where t_n is the occurrence time of the n^{th} state transition and β is the discount rate [16].

In order to solve the continuous time MDP formulation, it is convenient to develop an equivalent discrete time process and use dynamic programming techniques. In the DWA problem, the control actions (wavelength switching) is applied at discrete times (flow arrival or departure instants), but the cost is continuously accumulated. Moreover, the time between successive control choices is variable and depends on the current state and the action taken, resulting in non-uniform transition rates. To develop the discrete time equivalent process, the transition rates should be made uniform regardless of the state and the action. To transform the process into a process with uniform transition rates, the technique of *uniformization* is used [16]. The basic idea of uniformization is to introduce fictitious transitions from a state to itself, so that the transitions that are slow on the average are speeded up with the added transitions.

The uniform transition rate, ν, should be greater than the maximum transition rate of the original process. Hence, for the continuous time MDP at hand, a suitable choice may be

$$\nu = \sum_{i=1}^{N} \lambda_i + W\mu + \sigma$$

where $\mu = \max(\mu_i)$. Next, an equivalent discrete time Markov chain is constructed with the following transition probabilities:

$$p_{ss'}(a) = \begin{cases} q_{ss'}(a)/\nu, & \text{if } s' \neq s \\ 1 - q_s(a)/\nu, & \text{if } s' = s \end{cases}$$

where $q_s(a) = \sum_{s'} q_{ss'}(a)$.

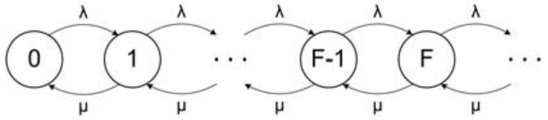


Fig. 3. Infinite Markov chain

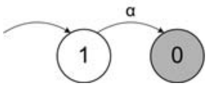


Fig. 4. Exponential distribution

The discount factor for the resulting discrete-time Markov chain is

$$\tilde{\beta} = \frac{\nu}{\beta + \nu}$$

The cost per stage is calculated as

$$\tilde{g}(s, a) = \frac{g(s, a)}{\beta + \nu}$$

Then, Bellman's equation takes the form

$$J(s) = \min_{a \in A_s} \left[\tilde{g}(s, a) + \tilde{\beta} \sum_{s'} p_{ss'}(a) J(s') \right] \tag{2}$$

where $J(s)$ is the cost associated with state s for a given policy [16].

B. Solution of the MDP Model

The solution of the set of linear equations in (2) results in the value of each state and the optimum switching policy, which is the action to be taken at each state. The number of flows at each node is a process described by the Markov chain depicted in Fig. 3, where the service rate μ depends on the number of wavelengths allocated to the node. Since this chain is infinite, the size of the state space, S, and therefore the number of equations in (2) is infinite. In order to use numerical solution techniques, the number of equations should be made finite. This can be accomplished by truncating the number of flows at each node at F. A simple truncation may be inadequate if the probability of states beyond F is not negligible. For this reason, it may be a better idea to match the first moment of the sojourn time in the truncation process.

For the infinite Markov chain with uniform transition rates, sojourn time at the set of states $\geq F$, for any value of F, is exactly same as the busy period in an M/M/1 queue. The first moment (i.e., mean) of this distribution is

$$m_1 = E[T] = \frac{1}{(1-\rho)} \frac{1}{\mu},$$

where $\rho = \lambda/\mu$.

The first moment of the sojourn time distribution can be matched using a simple exponential distribution (Fig.4) with mean $1/\alpha = m_1$ [17]. The resulting chain is shown in Fig. 5, where the state $F+$ corresponds to the set of states with number of flows equal to or greater than F. After truncation, the resulting finite-state discrete-time MDP is solved using the method of value iteration [18].

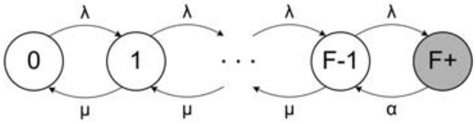


Fig. 5. Truncated Markov chain with first moment matched

C. Cost Function

Cost function is the key component of any optimization problem and it should be designed in accordance with the objectives of the optimization. For the DWA problem at hand, the primary goal is to maximize the throughput which is equivalent to minimizing flow completion times. Meanwhile, it is also desirable that each flow uses a fair share of the capacity available. Following cost functions are considered in this paper.

1) **Flow Sum (FS)**
In resource allocation problems, it is usually assumed that each job waiting or being serviced in the system incurs a *holding cost* per unit time depending on the relative importance of the job type [14]. If each type of job has equal importance, then the holding cost can simply be defined as the sum of the number of jobs
$$g(s,a)=\sum_i f_i. \tag{3}$$
With this definition, the total cost is equal to the sum of flow completion times. Hence, FS aims to minimize the sum of flow completion times which is equivalent to maximizing average throughput.

2) **Normalized Flow Sum (NFS)**
This cost function is derived from the heuristic method of [12]. Although not explicitly stated, this heuristic method can be seen as an approximation to the optimum policy obtained with the cost function
$$g(s,a)=\sum_i \frac{f_i}{w_i}.$$
The motivation behind NFS is to balance the load between wavelength channels to achieve high throughput and fairness.

3) **Normalized Squared Flow Sum (NSFS)**
In this work, we propose the following cost function
$$g(s,a)=\sum_i \frac{f_i^2}{w_i}.$$
The basic idea behind NSFS is to minimize both the flow completion times and load imbalance between wavelength channels in order to obtain better results in terms of throughput and fairness.

These cost functions may be compared based on the following properties, which can be considered useful in order to achieve the objectives of throughput and fairness.

P1. *Cost of a node should be an increasing function of number of flows at the node.* This property is based on the idea that the throughput can be maximized by minimizing the duration of flows at each node. All of the above cost functions satisfy this property.

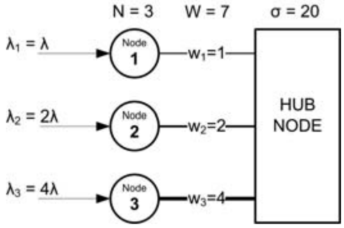


Fig. 6. 3-node test network

P2. *Cost of a node should be a decreasing function of number of wavelengths assigned to the node.* It is clear that, increasing the service rate also increases the throughput. Moreover, this property is useful to account for the costs associated with the unavailability of the reconfigured wavelength during the switching period. It is observed that this property holds for the cost functions NFS and NSFS.

P3. *Total cost should be minimum when the load is balanced among wavelength channels.* A fair service is achieved when the number of wavelengths at each node is proportional to the number of flows at the corresponding node. Hence, it may be desirable that the cost function attains the minimum value at this point, i.e., when w_i is proportional to f_i. This property is satisfied by the cost function NSFS as shown in Appendix I.

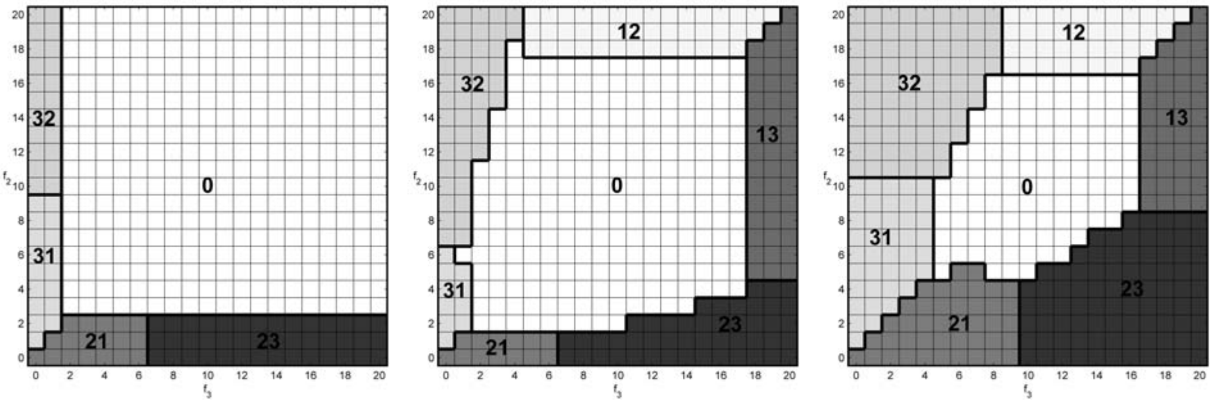
In the following subsection, these cost functions are used in the MDP formulation to obtain optimum reconfiguration policies and the performance of these policies are compared through simulations.

D. Comparison of Cost Functions

The three cost functions are compared on a 3-node network scenario shown in Fig. 6. In this network, there are 7 wavelength channels. Flow arrival rates are λ, 2λ, and 4λ flows/s to nodes 1, 2, and 3, respectively. The bandwidth of a single channel is 10 Gpbs and the average flow size is 1250 MB. Hence, the service rate of a flow by a single channel, μ_i, is 1 flows/s, for all nodes i. Average switching delay, $1/\sigma$, is 50 ms.

Figures 7(a), 7(b) and 7(c) show parts of the optimum policies (corresponding to states with $w=[3,2,2]$ and $f=[15,f_2,f_3]$) obtained using the cost functions considered, for $\lambda=0.7$ and $F=20$. The x-axis corresponds to the number of flows at node 3 and the y-axis is the number of flows at node 2. Each cell in the matrices corresponds to a single state and the value of the cell is the optimum action to be taken at that state. The switching actions, a_{ij} are labeled on the figures as ij meaning that a switch from node i to node j is to be performed. No switching decisions, a_0, are labeled as 0.

FS aims to minimize the total duration of flows in the network and it does not consider load balancing at all. Consistent with this objective, it prefers to switch a wavelength from a node when the number of flows at that node is very low, as can be observed from Fig. 7(a). Since the number of flows at node 1 is large for all states shown, the policy does not make switches from node 1.



(a) Optimum policy for FS

(b) Optimum policy for NFS

(c) Optimum policy for NSFS

Fig. 7. Optimum switching policies for the 3-node test network, for states with $w = [3, 2, 2]$ and $f = [15, f_2, f_3]$.

The policy obtained using NFS is shown in Fig. 7(b). It is observed that this policy makes more switches compared to FS policy. In addition to the actions taken when the number of flows at a node gets small, this policy also makes switches to achieve load balancing between wavelengths. This is evident from the fact that, for high number of flows at nodes 2 or 3, a wavelength is switched from node 1.

Fig. 7(c) plots the policy obtained with NSFS. This policy has a similar structure with the NFS policy, but the area corresponding to no action (a_0) is smaller. So, it may be concluded that this policy makes more switches in order to balance the load at each node. In order to evaluate the performance of each policy, simulations are performed where λ is changed from 0.1 to 0.9. Each simulation is repeated 10 times and the average values are plotted. Throughput and fairness are used as performance metrics. Slowdown is used as a normalized measure of the throughput efficiency [12]. It is defined as the ratio of actual flow duration to the time that would be required if the flow was served by a single dedicated wavelength channel. The average slowdown experienced by all flows is used as the throughput performance metric. Holding cost defined in (3) is also used to measure the throughput performance [14]. To assess the fairness, the “fairness index” proposed in [19] is used, which takes values in the interval (0,1], and it is defined as:

$$f(x) = \frac{\left[\sum_{i=1}^{n} x_i\right]^2}{n \sum_{i=1}^{n} x_i^2}$$

where n is the total number of flows and x_i is the slowdown experienced by the i^{th} flow.

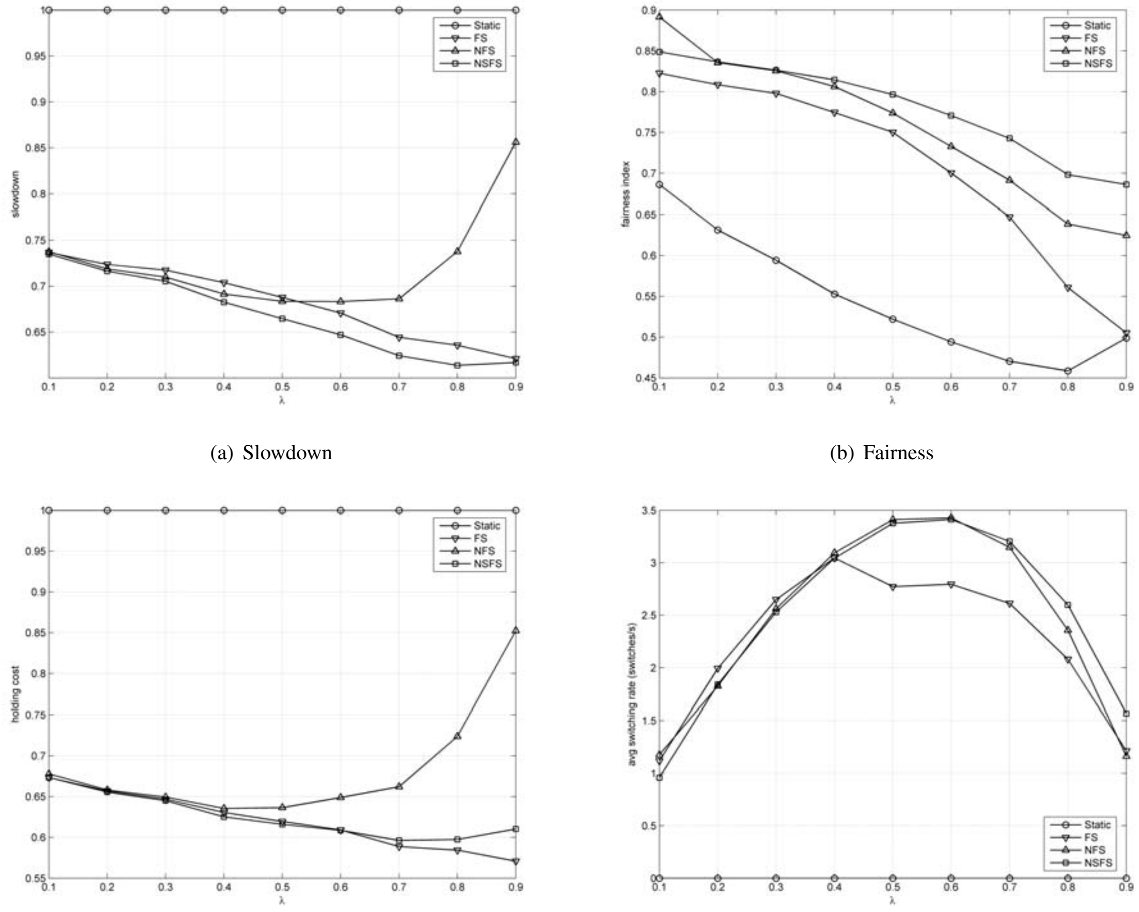
For comparison, we also use the static wavelength allocation, where the channels are assigned based on average traffic demands and they are not reconfigured. For this scenario, the static allocation corresponds to allocating 1, 2, and 4 wavelengths to nodes 1, 2, and 3, respectively. Fig. 8(a) plots the slowdown obtained using each of the switching policies normalized with respect to the slowdown experienced under the static policy. As a first observation, it is seen that the dynamic policies yield significantly better slowdown performance than the static policy. Among the dynamic policies, NSFS achieves the minimum slowdown for all values of network load. The results obtained with FS are close to NSFS. On the other hand, the slowdown obtained with the cost function NFS gets worse as the network load increases.

The performances of policies in terms of fairness as a function of the flow arrival rate are compared in Fig. 8(b). Static policy has a clear disadvantage in terms of fairness. All of the dynamic policies have better fairness at low load levels but as the load increases the fairness begins to drop. Among the dynamic policies, worst performance belongs to FS. This is expected since FS does not consider load balancing. With this policy, fairness drops sharply at high loads to the level obtained by the static policy. NFS is better than FS, but NSFS shows the best performance except at very low load levels.

Fig. 8(c) plots the holding cost obtained with each policy normalized to the holding cost experienced under the static policy. This graph is similar to the slowdown results. For low load levels all of the dynamic policies achieves 30%-35% lower holding costs compared to the static policy. The gains obtained with FS and NSFS increase further with the increasing load while the performance of NSF degrades sharply. It is observed that although NSFS policy is better than FS in terms of slowdown, the difference is negligible in terms of the holding cost. In fact, FS policy optimizes the holding cost and the results suggest that NSFS policy reduces the slowdown and increases fairness without sacrificing the holding cost objective.

Average rate of switches (switches per second) performed by each of the policies is depicted in Fig. 8(d). The curves corresponding to each policy has a similar pattern. Switching rate increases with increasing load up to 0.6, and then begin to decrease with the increasing load. At moderate loads FS performs the minimum number of switches among all policies. This result is related to the fact that at moderate and high load levels, the probability of having small number of flows at any node decreases and FS policy does not make switches unless the number of flows at a node is very low.

In summary, it can be stated that the NSFS policy has important advantages as a DWA method. It attains minimum slowdown and maximum fairness nearly for all levels of network load.



(a) Slowdown

(b) Fairness

(c) Holding Cost

(d) Average Switching Rate

Fig. 8. Performance of cost functions as a function of network load.

IV. Heuristic Methods

As usual with most of the optimization problems, dynamic programming solution of the MDP suffers from *curse of dimensionality*. The size of the state space for the MDP formulation is $F^N \times W_{max}^N \times (N+1)$, where N is the number of nodes, F is the truncation level for number of flows and W_{max} is the maximum number of wavelengths possible at each node, respectively. Since the size of the state space grows exponentially with N, the problem is solvable only for small networks. For a real-life IP/WDM network with around 10 nodes and 10-100 wavelengths, it is practically impossible to obtain a solution using the MDP approach. For this reason, development of heuristic methods is necessary.

In the following subsections, we introduce three heuristic methods which can be used to determine the switching action to be performed at flow arrival or departure instants. The state of the network prior to any switching action is denoted as $s^* = (\boldsymbol{f^*}, \boldsymbol{w^*}, k^*)$ and A_{s^*} is the valid set of actions at state s^*, as defined in (1).

A. Heuristic Method 1 (HM1)

This heuristic is inspired by the method devised in [14]. Although the context is different in that work, the underlying problem is similar to the DWA problem. HM1 makes switching actions if the action would help to balance the holding costs, taking into account the switching overheads. HM1 can be seen as an approximation to the optimum policy obtained using cost function FS. At each decision instant the following rule is applied:

- Calculate the following for each action $a_{ij} \in A_{s^*}$

$$R = f_j^* + \frac{1}{\sigma}\left(\lambda_j - \mu_j w_j^*\right) - K\left(f_i^* + \frac{1}{\sigma}\left(\lambda_i - \mu_i w_i^*\right)\right)$$

  where K is recommended to be 5 in [14].
- Take the action which yields the maximum R, if it is strictly greater than 0.

Note that the departure rate term in [14] is appropriately modified and the holding costs of flows at each node is taken as 1, to adapt the heuristic to the problem at hand.

B. Heuristic Method 2 (HM2)

This heuristic is proposed in [12]. At each decision epoch, action $a_{ij} \in A_{s^*}$ with $i = \arg\min_x\{f_x^*/w_x^*\}$ and $j = \arg\max_x\{f_x^*/w_x^*\}$, is performed if the following inequality holds

$$\frac{f_j^*}{w_j^*+1} + \frac{f_i^*}{w_i^*-1} < \frac{f_j^*}{w_j^*} + \frac{f_i^*}{w_i^*}$$

The basic idea behind HM2 is to keep all wavelengths in the system evenly loaded. So, a switch will be performed if it is going to improve the load balance in the system. The switching costs are not taken into account and the flow arrival and departure rates are not considered. HM2 may be thought as a first order approximation to the optimum policy obtained using cost function NFS.

C. Heuristic Method 3 (HM3)

It is shown in Section III that the cost function NSFS performs best in terms of slowdown and fairness nearly for all levels of network load. Therefore, it seems reasonable to think that a heuristic method, which aims to minimize the cost function NSFS can be an efficient solution for the DWA problem. Based on this idea, a new algorithm, HM3, is proposed. HM3 applies the following rule to determine the action at each decision instant:

- For each action $a_{ij} \in A_{s^*}$, consider the 2-node subnetwork consisting of the nodes i and j, and calculate the value v_{ij} as
$$v_{ij} = \begin{cases} 0, & \text{if } c \in D \\ 1 - F_{cD}(\tau), & \text{otherwise} \end{cases}$$
where $c = (f_i^*, f_j^*)$ and D is defined as
$$D = \{(f_i, f_j) \mid f_i > m f_j\}, \tag{4}$$
$$m = \sqrt{\frac{w_i^*(w_i^* - 1)}{w_j^*(w_j^* + 1)}}.$$
$F_{cD}(\tau) = \Pr(T_{cD} < \tau)$, where T_{cD} is the time that starting from c, the first transit to a state in D occurs in the two-dimensional birth-death process with arrival rate λ_i, (λ_j), and service rate $(w_i - 1)\mu_i$, $(w_j \mu_j)$, at node i, (node j), respectively. τ is the switching delay, which has an exponential distribution.
- Apply the action with maximum v_{ij}, if $v_{ij} > T$, where $T < 1$ is the switching threshold, used in order to eliminate unnecessary switches.

The basic idea of HM3 can be demonstrated with the help of Fig. 9. In this figure, x- and y-axes correspond to the number of flows at node j and node i, respectively. The wavelength allocation, (w_i^*, w_j^*), defines the line L_W (solid line) with slope w_i^*/w_j^* along which a perfect load balance is achieved. As c (depicted by a star) moves away from L_W the load imbalance and the cost increases. The wavelength allocation of $(w_i^* - 1, w_j^* + 1)$ results in lower NSFS cost if

$$\frac{f_i^*}{f_j^*} < m = \sqrt{\frac{w_i^*(w_i^* - 1)}{w_j^*(w_j^* + 1)}}.$$

This condition is satisfied when the point (f_i^*, f_j^*) is below the line L_T (dashed line), which has a slope of m. The states above L_T, which are shown shaded in Fig. 9, constitute the set D, and the action a_{ij} may be beneficial if $c \notin D$.

At any state, there may be more than one action which potentially decrease the cost. Since at most one action is allowed at each decision step, it is important to select the most beneficial one. For this aim, it is necessary to attach a quantitative value, v_{ij}, to each action, a_{ij}, based on its expected reward. If c is above L_T, i.e., $c \in D$, a_{ij} does not improve the cost, hence v_{ij} is taken as 0 for this case. Otherwise, a_{ij} may decrease the cost.

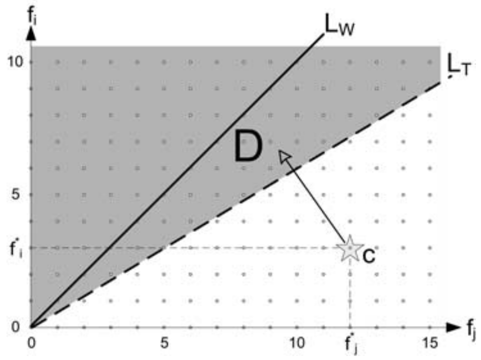


Fig. 9. Geometric interpretation of HM3

When the switching delay is neglected, v_{ij} can simply be defined as the differential cost between the states before and after the switching. However, this approach is not adequate when the switching delay is not negligible. This is due to the fact that, during the reconfiguration period, f_i and f_j may change with the arrival and departure of flows, in which case action may become useless or even detrimental. In order to take this effect into account, v_{ij} is defined as the probability that the intended action will be useful throughout the switching period, which is equal to the probability that the point (f_i, f_j) will always be below L_T until the switching is completed and hence a new switching can be initiated. This can be calculated as the probability that the first passage time starting from c until L_T is hit, is greater than the switching period, τ. With this definition, v_{ij} inherently considers the load imbalance, since it takes larger values as c gets farther away from L_T.

D. Implementation of HM3

HM3 requires the calculation of v_{ij} corresponding to each valid action a_{ij} at each decision epoch. This can be achieved by first truncating f_i and f_j at levels $F_i(> f_i^*)$ and $F_j(> f_j^*)$, respectively. It is observed that truncation with matching the first three moments of the sojourn time, as explained in Appendix II, yields satisfactory results. Then, on this truncated two-dimensional chain, v_{ij} can be calculated using the method given in Appendix III and utilizing Lemma III.1.

However, it is also possible to calculate and record the first passage probabilities for a representative set of states, and reuse this data at each decision epoch to calculate the required v_{ij} values. The method of calculation is illustrated in Fig. 10. f_i and f_j evolve in time with the arrival and departure of flows

$$\begin{aligned} f_i(u) &= f_i(0) + a_i(u) - d_i(u) \\ f_j(u) &= f_j(0) + a_j(u) - d_j(u) \end{aligned}$$

where $a_i(u)$ $(a_j(u))$ and $d_i(u)$ $(d_j(u))$ are the number of arrivals and departures at node i (node j) up to time u.

In the following discussion, $U(r, \epsilon)$ is the function defined

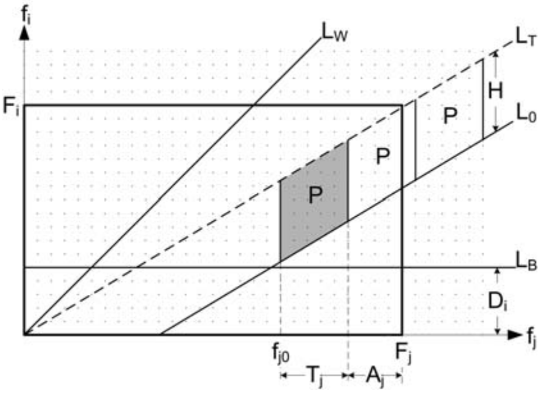


Fig. 10. Calculation of first passage probability data

as

$$U(r,\epsilon)=\arg\min_k\left\{\left(\sum_{i=0}^{k}e^{-r}\frac{r^i}{i!}\right)>1-\epsilon\right\},$$

so that, if γ is a Poisson distributed random variable with parameter r then $\Pr\{\gamma>U(r,\epsilon)\}<\epsilon$.

The proofs of the following lemmas are postponed to Appendix IV for a clear presentation. The existence of the reflecting boundary, $f_i=0$ affects the first passage probabilities because the departure rate from node i is 0 along this line. However, this effect gets smaller with larger f_i^* and can be neglected for states with large enough f_i^*, as discussed in the following lemma.

Lemma IV.1. *Let P_B be the probability of the event of hitting the reflecting boundary at $f_i=0$ before the first passage to D occurs starting from c. Then for a given $\epsilon>0$, $P_B<\epsilon$ if $f_i^*>D_i=U(\tau\mu_i(w_i-1),\epsilon)$.*

The following lemma shows that, if $f_i^*>D_i$ the first passage probability becomes approximately a function of the distance between the point c and the line L_T.

Lemma IV.2. *If $f_i^*>D_i$, then the first passage probability starting from c is approximately a function of*

$$h((f_i^*,f_j^*))=mf_j^*-f_i^* \tag{5}$$

where $h((f_i^*,f_j^*))$ corresponds to the vertical distance between c and the line L_T, given by the equation $f_i=mf_j$.

It is also possible to neglect the first passage probability starting from a state with sufficiently large distance to L_T, as stated in the following lemma.

Lemma IV.3. *If $h((f_i^*,f_j^*))>H$, then the first passage probability starting from c is smaller than 2ϵ, where*

$$H=mD_j+A_i, \tag{6}$$

$A_i=U(\tau\lambda_i,\epsilon)$, and $D_j=U(\tau\mu_jw_j,\epsilon)$.

With this assumption, first passage probabilities for the states below the line L_0 can be approximated as 0 and first passage probabilities starting only from states between L_T and L_0 in Fig. 10 are non-zero.

If the slope of L_T is approximated as

$$m\approx(w_i-0.5)/(w_j+0.5), \tag{7}$$

then it suffices to calculate the first passage probabilities starting from states in the shaded region, P. Because, P is repeating itself along the strip between the lines L_T and L_0, hence all of the states in this strip can be mapped to a state in P as stated by the following lemma.

Lemma IV.4. *Let $T_j=(2w_i-1)/\gcd(2w_i-1,2w_j+1)$ and $T_i=mT_j$. First passage probability starting from state $(f_i^*nT_i,f_j^*+nT_j)$ for any integer $n>0$ is equal to the first passage probability starting from state (f_i^*,f_j^*) if $f_i^*>D_i$.*

Therefore, for a given wavelength allocation w_i and w_j, the first passage probability starting from any (f_i,f_j) can be obtained by considering only the states $0\le f_j\le F_j$ and $0\le f_i\le F_i$. F_j can be calculated as

$$F_j=f_{j0}+T_j+A_j \tag{8}$$

where

$$f_{j0}=\lceil(D_i+H)/m\rceil \tag{9}$$

and $A_j=U(\tau\lambda_j,\epsilon)$ is the margin added so that the effects of the boundary on the right hand side can be ignored. F_i can be taken as

$$F_i=\lceil mF_j\rceil \tag{10}$$

To sum up, HM3 calculates and saves the first passage probability data corresponding to states (f_i,f_j), for $f_i=0,\ldots,F_i$ and $f_j=0,\ldots,F_j$, for each possible node pair (i,j) and wavelength allocation (w_i,w_j). The outline of this calculation is shown in Algorithm 1. During the simulation, this data is used by the dynamic part of the HM3 algorithm for calculating the switching actions, as given in Algorithm 2. First passage probability for state $c=(f_i^*,f_j^*)$ is taken to be 0, if $c\in D$ or $h(f_i^*,f_j^*)>H$. Otherwise, it is read from the saved data directly if $f_i^*<F_i$ and $f_j^*<F_j-A_j$, or after a mapping operation done according to Lemma IV.4, if the condition is not satisfied.

The number of iterations for calculating the first passage probability data in Algorithm 1 is $O(N^2\times W^2)$. On the other hand, the memory requirement for Algorithm 1 grows proportionally with $N^2\times W^2$. Algorithm 2, which is used to determine the switching actions compares at most $N\times(N-1)$ values read from the data generated by Algorithm 1, i.e., the algorithm has an $O(N^2)$ complexity.

V. Numerical Results

For the network given in Fig. 6, reconfiguration policies are calculated using the heuristic methods discussed in Section IV. Parts of the policies corresponding to a sample set of states are shown in Figs. 11(a), 11(b) and 11(c) for comparison purposes. the switching decisions are close to the axes and it does not perform switches to balance the load on each wavelength. HM2 resulted in a symmetric matrix, since it does not consider the arrival rates. Due to this symmetric nature, there are states at which more than one action have the same cost. These states are shown unlabeled in the figure. It is also observed that HM2 makes switches from node 1 when the number of flows at other nodes gets large. So it may be concluded that HM2 makes switches to balance the load. Compared with the NFS policy, HM2 performs switches at more states. Finally,

Algorithm 1 Calculation of First Passage Probabilities

```
Input: $N$, $W$, $\tau$, $\epsilon$, and $\lambda_i$, $\mu_i$, $i = 1, \ldots, N$
Output: $M_F, M_{T_i}, M_{T_j}, M_{f_{j0}}$
for all node pairs $(i, j)$ do
    for $w_i = 2$ to $W_{max}$ do
        for $w_j = 1$ to $(W_{max} + 1 - w_i)$ do
            STEP-1: Set $M_{T_i}[i, j, w_i, w_j] = T_i$, $M_{T_j}[i, j, w_i, w_j] = T_j$, where $T_i$ and $T_j$ is calculated as in Lemma IV.4.
            STEP-2: Set $M_{f_{j0}}[i, j, w_i, w_j] = f_{j0}$, where $f_{j0}$ is calculated using (9).
            STEP-3: Calculate $F_j$ and $F_i$ using (8) and (10), respectively.
            STEP-4: Construct the infinitesimal matrix, $Q$, for the two-dimensional Markov chain with states $(f_i, f_j)$, where $f_i$, $f_j$ are truncated at $F_i$ and $F_j$, respectively, as explained in Appendix II.
            STEP-5: For each $c = (f_i, f_j)$, set $M_F[i, j, w_i, w_j, f_i, f_j] = F_{cD}(\tau)$, where $D$ is obtained using (4) with the assumption in (7), and $F_{cD}(\tau)$ is calculated using Lemma III.1.
        end for
    end for
end for
```

Algorithm 2 Calculation of Switching Actions

```
Input: $s^* = (f^*, w^*, k^*), M_F, M_{T_i}, M_{T_j}, M_{f_{j0}}$
Output: $a^*$
for all $a_{ij} \in A_{s^*}$ do
    if $(f_i^*/f_j^*) > m$, where $m$ is given in (7) then
        $v_{ij} \leftarrow 0$
    else if $h(f_i^*, f_j^*) > H$, where $h(f_i^*, f_j^*)$ and $H$ are defined by (5) and (6), then
        $v_{ij} \leftarrow 0$
    else
        $T_i \leftarrow M_{T_i}[i, j, w_i^*, w_j^*]$
        $T_j \leftarrow M_{T_j}[i, j, w_i^*, w_j^*]$
        $f_{j0} \leftarrow M_{f_{j0}}[i, j, w_i^*, w_j^*]$
        $c \leftarrow \max(0, \lfloor (f_j^* - f_{j0})/T_j \rfloor)$
        $\tilde{f}_i \leftarrow f_i^* - cT_i$
        $\tilde{f}_j \leftarrow f_j^* - cT_j$
        $F_{cD}(\tau) \leftarrow M_F[i, j, w_i^*, w_j^*, \tilde{f}_i, \tilde{f}_j]$
        $v_{ij} \leftarrow 1 - F_{cD}(\tau)$
    end if
end for
$(u, v) \leftarrow \arg\max_{ij}\{v_{ij}\}$
if $v_{uv} > T$ then
    $a^* \leftarrow a_{uv}$
else
    $a^* \leftarrow a_0$
end if
```

TABLE I
TIME VARYING ARRIVAL RATES.

time (s)	λ_1	λ_2	λ_3	λ_4	λ_5
0– 500	1	2	3	4	5
500– 900	1	2	3	4	5
1300–1700	2	3	4	5	1
1700–2100	3	4	5	1	2
2100–2500	4	5	1	2	3
2500–2750	5	1	2	3	4

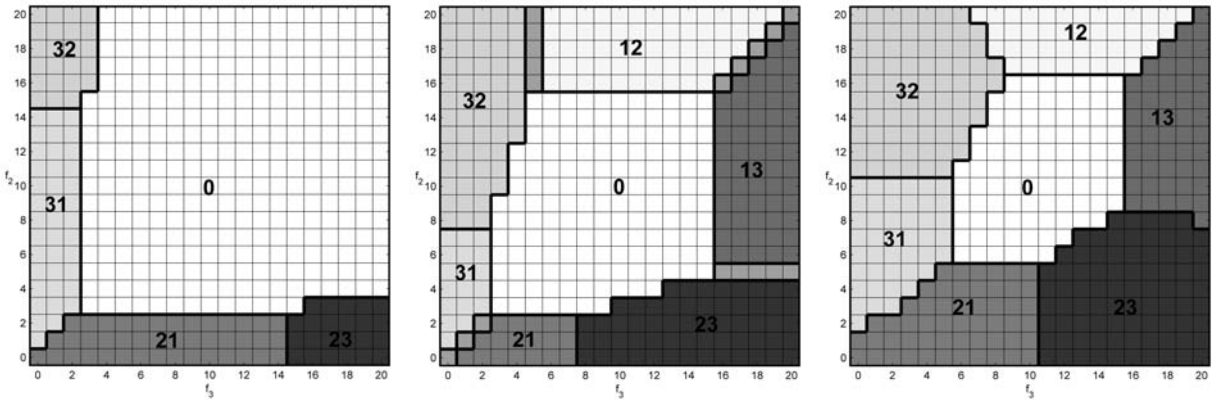
Fig. 11(c) plots the policy obtained with HM3 using $T = 0.9$. Similar to HM2, switches from node 1 are performed when the number of flows at node 2 or node 3 gets large. The area corresponding to no-switching is narrower for HM3 compared to HM1 and HM2. Hence, HM3 performs switches at a larger number of states and the resulting policy closely resembles the NSFS policy.

The performance of HM1, HM2 and HM3 are evaluated through simulations and the results are plotted in Fig. 12. The results of the optimum policies corresponding to this scenario were shown in Fig. 8. It is observed that the performance of HM1 is worse than the optimum policy FS for all network loads. HM1 is more conservative in switching, and results in higher slowdown and holding costs. It has also the worst fairness performance among the heuristic methods. HM2 behaves similar to optimum policy NFS as the load increases. Its slowdown and holding cost performance are good at low network loads. But as the load is increased, HM2 becomes the worst method. HM3 gives the best results in terms of slowdown and fairness. Its behavior is close to the optimum policy NSFS, especially for load levels greater than 0.5.

For a heuristic method to be robust, it should work satisfactorily under different conditions. It has been already shown that HM3 performs well for all levels of network load for a given average flow size. Thus, another dimension of interest is the performance of the heuristic methods for different flow length distributions. It is clear that for a given load level, uncertainty increases with decreasing flow size (increasing arrival rate). At the extreme case, as the average flow size goes to zero, current state of the network carry no information about the future states and there is no point in dynamic reconfiguration. Therefore, the switching policy should make less and less switches and converge to the static policy as the flow sizes decrease.

To evaluate the effects of average flow size on the performance of heuristic methods, simulations are performed on the same network. The load is kept fixed at 0.5 and the average flow size and flow arrival rate are changed accordingly. The results are plotted in Fig. 13. It has been observed that only heuristic method HM3 succeeds to adapt to changes in the flow size. For small flow sizes, performances of HM1 and HM2 deteriorate rapidly, even below the performance of the static policy. On the other hand, HM3 adjusts itself appropriately and converges to the static policy as the average flow size decreases. Moreover, HM3 shows the best fairness performance for all values of average flow size.

As discussed in Section I, metro access networks have

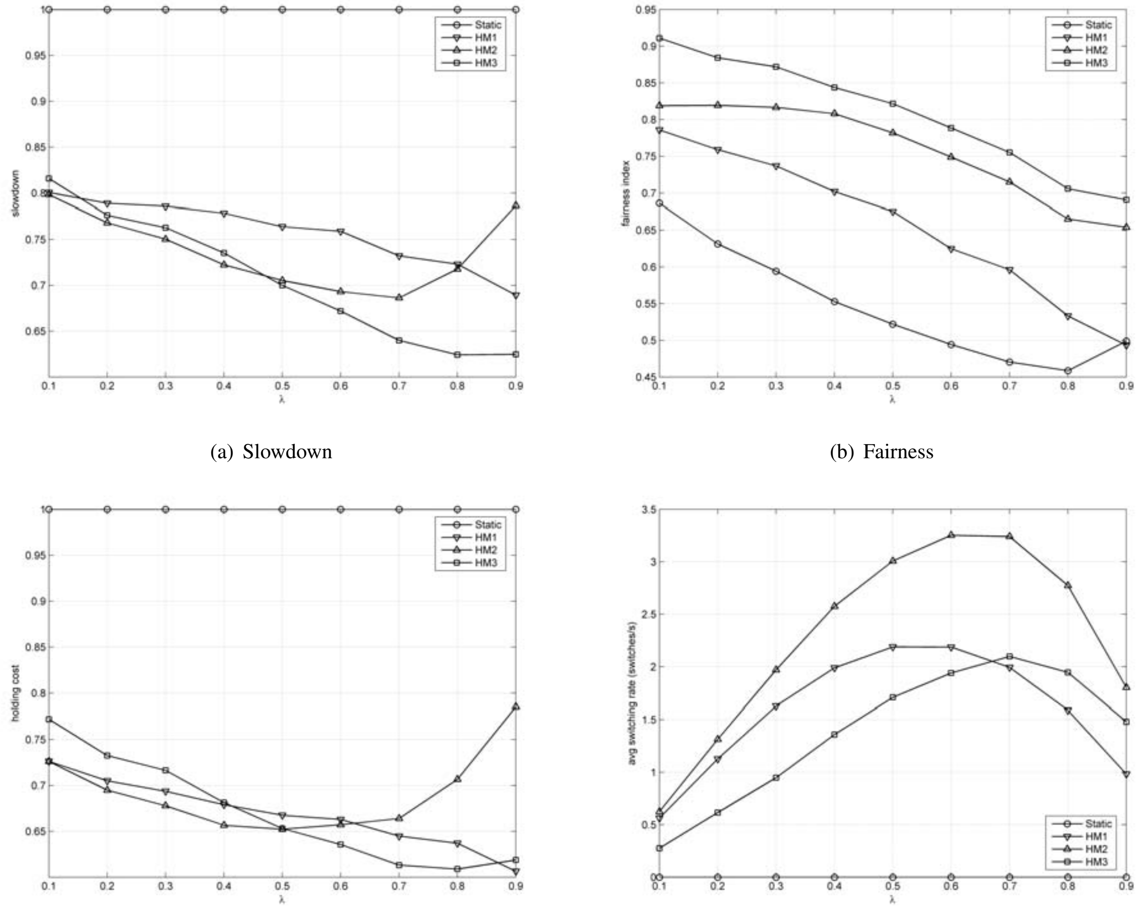


(a) Heuristic policy HM1

(b) Heuristic policy HM2

(c) Heuristic policy HM3

Fig. 11. Heuristic switching policies for the 3-node test network, for states with $w = [3, 2, 2]$ and $f = [15, f_2, f_3]$.



(a) Slowdown

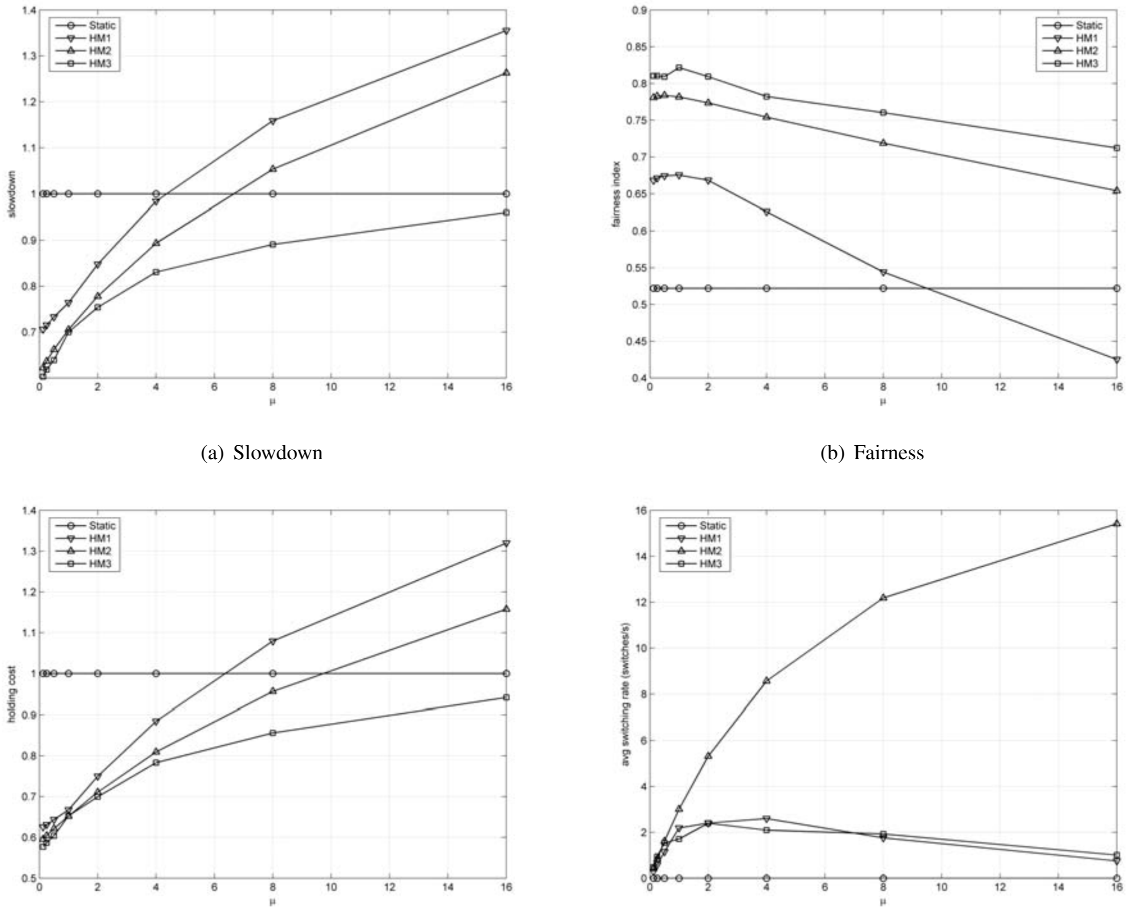
(b) Fairness

(c) Holding Cost

(d) Average Switching Rate

Fig. 12. Performance of heuristic methods as a function of network load.

traffic patterns changing in time. Hence, the adaptability of the DWA methods to time varying traffic characteristics is an important property. To test this case, simulations are performed using the network shown in Fig. 14 which has 5 nodes and 30 wavelengths. Flow arrival rates are changed with time according to Table I. Since, the average flow arrival rate to each node is equal, for the static policy each node is allocated 6 wavelength channels. Performance metrics are calculated for the time interval [500-2500] sec and tabulated in Table II. HM1 succeeds to decrease the holding cost to nearly 60% of the static policy. But the improvement in terms of slowdown is just 30%. Interestingly, the fairness with HM1 is below the static policy. HM2 performs the highest number of switches and attains a good performance in terms of slowdown and holding costs. It achieves 49% lower slowdown with respect to the static policy and a fairness index of 0.6842. HM3, on the



(a) Slowdown

(b) Fairness

(c) Holding Cost

(d) Average Switching Rate

Fig. 13. Comparison of heuristic methods for different average flow sizes.

TABLE II
COMPARISON OF HEURISTIC POLICIES UNDER DYNAMIC TRAFFIC CONDITIONS.

Method	# Switch	Slowdown	Fairness	H. Cost
Static	0	0.5786	0.4631	17309.0
HM1	21697	0.4146	0.4982	10049.0
HM2	23249	0.2949	0.6842	7789.6
HM3	14654	0.2832	0.7765	7794.3

other hand attains better results by making much less switches. Holding cost of HM3 is close to HM2, but the slowdown is 4% better and there is significant improvement in fairness compared to HM2.

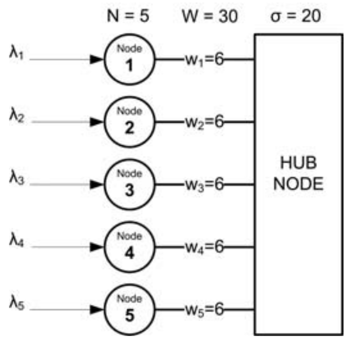


Fig. 14. 5-node test network

VI. CONCLUSION

In the evolution of metro access networks, dynamic traffic adaptability is a major requirement as a result of the inherently variable nature of the traffic demand. In this work, benefits and tradeoffs related to dynamic wavelength allocation is investigated for an IP/WDM metro access network. The problem is formulated as an MDP and a new cost function is proposed. It is demonstrated that the optimum policy obtained using the proposed cost function achieves superior performance in terms of slowdown and fairness. Also, a heuristic method based on the proposed cost function and first passage probabilities is developed and compared with similar heuristics in the literature. Through simulations it is demonstrated that the proposed heuristic generates near-optimum solutions and results in significant improvements in throughput efficiency and fairness for a wide range of network load and average flow size conditions. It is also verified that the heuristic method adapts to dynamically changing network load appropriately.

ACKNOWLEDGMENT

The work described in this paper was carried out with the support of the BONE-project ("Building the Future Optical Network in Europe"), a Network of Excellence funded by the European Commission through the 7th ICT-Framework Programme, and by the Scientific and Technological Research Council of Turkey (TUBITAK) under project 104E047.

APPENDIX I
MINIMA OF COST FUNCTION NSFS

Lemma I.1. *The function,* $g(\boldsymbol{f}, \boldsymbol{w}) = \sum_i f_i^2/w_i$*, is convex cup, and it is minimized when* $\boldsymbol{w}$ *is proportional to* $\boldsymbol{f}$.

Proof: Let R be the region consisting of $\boldsymbol{w}$ vectors defined by

$$\sum_{i=1}^{N} w_i = W$$

For any vector $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ in R, the vector $\theta\boldsymbol{\alpha} + (1-\theta)\boldsymbol{\beta}$ is in R for $0 \le \theta \le 1$, because

$$\sum_{i=1}^{N} (\theta\alpha_i + (1-\theta)\,\beta_i) = \theta W + (1-\theta)W = W$$

So, R is a convex region. For all $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ in R and $0 \le \theta \le 1$,

$$\begin{aligned}\theta g(\boldsymbol{f},\boldsymbol{\alpha}) + (1-\theta)g(\boldsymbol{f},\boldsymbol{\beta}) - g(\boldsymbol{f},\theta\boldsymbol{\alpha} + (1-\theta)\boldsymbol{\beta}) &= \\ \sum_{i=1}^{N} f_i^2 \frac{(\alpha_i - \beta_i)^2}{\alpha_i\beta_i(\theta\alpha_i + (1-\theta)\beta_i)} &\ge 0\end{aligned}$$

Hence, g is convex cup ($\cup$) over R and therefore it has a minima which can be found using the method of Lagrange Multipliers:

$$\frac{\partial}{\partial w_i}\left(\sum_{i=1}^{N}\frac{f_i^2}{w_i} + \lambda\left(\sum_{i=1}^{N} w_i - W\right)\right) = -\frac{f_i^2}{w_i^2} + \lambda = 0$$

$$\Longrightarrow \frac{f_i^2}{w_i^2} = \lambda \Longrightarrow \frac{f_i}{w_i} = \sqrt{\lambda}$$

APPENDIX II
TRUNCATION OF MARKOV CHAINS WITH THREE MOMENT MATCHING

For an infinite Markov chain with uniform transition rates (Fig. 3), sojourn time, T, at states $s \ge K$ for any K is exactly the same as the busy period in an M/M/1 queue. First three moments of T are: $m_1 = E[T] = \frac{1}{(1-\rho)}\frac{1}{\mu}$, $m_2 = E[T^2] = \frac{2}{(1-\rho)^3}\frac{1}{\mu^2}$, and $m_3 = E[T^3] = \frac{6(1+\rho)}{(1-\rho)^5}\frac{1}{\mu^3}$, where $\rho = \lambda/\mu$.

In order to match the first three moments of the sojourn time, two-phase $Coxian^{+}PH$ distribution, shown in Fig. 15, can be used [17].

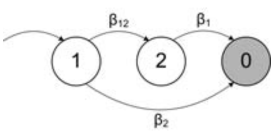


Fig. 15. $Coxian^{+}PH$ distribution

The parameters of $Coxian^{+}PH$ distribution are

$$\beta_1 = (1-p_x)\lambda_{x1} \qquad \beta_{12} = p_x\lambda_{x1} \qquad \beta_2 = \lambda_{x2}$$

$$\lambda_{x1} = \frac{u + \sqrt{u^2 - 4v}}{2\mu_1} \qquad \lambda_{x2} = \frac{u - \sqrt{u^2 - 4v}}{2\mu_1}$$

$$p_x = \frac{\lambda_{x2}(\lambda_{x1}\mu_1) - 1}{\lambda_{x1}}$$

$$u = \frac{6 - 2m_3}{3m_2 - 2m_3} \qquad v = \frac{12 - 6m_2}{m_2(3m_2 - 2m_3)}$$

The resulting truncated chain is shown in Fig. 16.

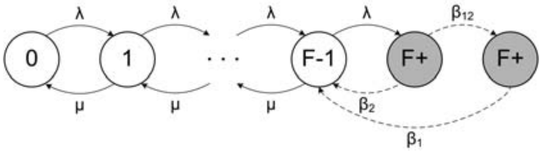


Fig. 16. Truncated Markov chain with first three moments matched

APPENDIX III
FIRST PASSAGE PROBABILITIES

For a finite, irreducible, continuous time Markov chain (CTMC) with n states and generator matrix Q, the first passage time from a source state c into a non-empty set of target states D is defined as

$$T_{cD}(t) = inf\{u > 0 : X(t+u) \in D \mid X(t) = c\}$$

where $X(t)$ denotes the state of CTMC at time $t \ge 0$ [20].

When the CTMC is stationary and time-homogeneous, T_{cD} is independent of t:

$$T_{cD} = inf\{u > 0 : X(u) \in D \mid X(0) = c\}$$

Let $f_{cD}(t)$ be the probability density function of T_{cD}, then

$$\Pr(a < T_{cD} < b) = \int_a^b f_{cD}(t)\mathrm{d}t \qquad 0 \le a < b$$

Using a first step analysis, the Laplace transform of f_{cD} can be written as

$$L_{cD}(s) = \sum_{k \notin D} p_{ck}\left(\frac{-q_{cc}}{s - q_{cc}}\right)L_{kD}(s) + \sum_{k \in D} p_{ck}\left(\frac{-q_{cc}}{s - q_{cc}}\right)$$

The first term denotes the event that the system first transits to a non-target state k then to a target state in D. The second term is for the case where the system transits from state c directly to a state in D. Using the relation $p_{ck} = -q_{ck}/q_{cc}$, this expression can be rewritten as,

$$(s - q_{cc})L_{cD}(s) = \sum_{k \notin D} q_{ck}L_{kD}(s) + \sum_{k \in D} q_{ck}$$

The set of equations can also be expressed in matrix-vector form. For example, when $D = \{1\}$,

$$\begin{bmatrix} s - q_{11} & -q_{12} & \cdots & -q_{1n} \\ 0 & s - q_{22} & \cdots & -q_{2n} \\ 0 & -q_{32} & \cdots & -q_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & -q_{2n} & \cdots & s - q_{nn} \end{bmatrix} \begin{bmatrix} L_{1D}(s) \\ L_{2D}(s) \\ L_{3D}(s) \\ \vdots \\ L_{nD}(s) \end{bmatrix} = \begin{bmatrix} 0 \\ q_{21} \\ q_{31} \\ \vdots \\ q_{n1} \end{bmatrix}$$

The value of $L_{cD}(s)$ can be obtained by solving this set of n linear equations. The value of $f_{cD}(t)$ is then calculated by using one of several methods for numerical transform inversion, such as Euler and Post-Widder algorithms [21].

The Laplace transform L_{cD} has also a direct probabilistic interpretation as stated by the following lemma.

Lemma III.1.

$$L_{cD}(\sigma) = \Pr(T_{cD} < \tau) = F_{cD}(\tau)$$

where τ is an exponential random variable with rate σ.

Proof:

$$\begin{aligned}
\Pr(T_{cD} \le \tau) &= \int_0^\infty \Pr(T_{cD} \le t)\sigma e^{-t\sigma}\mathrm{d}t \\
&= \int_0^\infty \int_0^t f_{cD}(s)\mathrm{d}s\sigma e^{-t\sigma}\mathrm{d}t \\
&= \int_0^\infty f_{cD}(s)\int_s^\infty \sigma e^{-t\sigma}\mathrm{d}t\mathrm{d}s \\
&= \int_0^\infty f_{cD}(s)e^{-s\sigma}\mathrm{d}s \\
&= L_{cD}(\sigma)
\end{aligned}$$

Appendix IV
Proofs of Lemmas IV.1-4

A. Lemma IV.1

Proof: Let B denote the event that the boundary $f_i = 0$ is hit in time interval $0 \le u < \tau$ and B' its complementary event. The probability of event B can be written as

$$P_B = \Pr\{B\} = 1 - \Pr\{f_i(u) > 0, 0 \le u < \tau\}. \qquad (11)$$

P_B can be bounded as follows

$$\begin{aligned}
P_B &= 1 - \Pr\{f_i^* + a_i(u) - d_i(u) > 0, 0 \le u < \tau\} \\
&< 1 - \Pr\{f_i^* - d_i(u) > 0, 0 \le u < \tau\} \\
&= 1 - \Pr\{d_i(\tau) < f_i^*\} \\
&< 1 - \Pr\left\{\hat{d}_i(\tau) < f_i^*\right\}
\end{aligned}$$

where $\hat{d}_i$ is a Poisson random variable with parameter $\tau\mu_i(w_i - 1)$. Hence, P_B converges to 0 as f_i^* increases. If $f_i^* > D_i$, then

$$P_B < 1 - \Pr\left\{\hat{d}_i(\tau) < D_i\right\} < 1 - (1 - \epsilon) = \epsilon$$

B. Lemma IV.2

Proof: The first passage probability can be decomposed into two terms conditioned on B:

$$\begin{aligned}
F_{cD}(\tau) &= \Pr\{T_{cD} < \tau\} \\
&= P_B \Pr\{T_{cD} < \tau | B\} + (1 - P_B)\Pr\{T_{cD} < \tau | B'\}
\end{aligned}$$

If $f_i^* > D_i$ then due to Lemma IV.1 $F_{cD}(\tau)$ can be approximated as:

$$\begin{aligned}
F_{cD}(\tau) &\approx \Pr\{T_{cD} < \tau \mid B'\} \\
&= Pr\{\inf\{u > 0 : h(f_i(u), f_j(u)) < 0\}\}
\end{aligned}$$

where

$$h(f_i(u), f_j(u)) = m(f_j^* + a_j(u) - d_j(u)) - (f_i^* + a_i(u) - d_i(u))$$

$$\begin{aligned}
&= (mf_j^* - f_i^*) - m(d_j(u) - a_j(u)) + (d_i(u) - a_i(u)) \\
&= h(f_i^*, f_j^*) - m(d_j(u) - a_j(u)) + (d_i(u) - a_i(u))
\end{aligned}$$

Since, a_i, d_i, a_j, and d_j are independent Poisson processes with state independent rates, $F_{cD}(\tau)$ is a function of $h(f_i^*, f_j^*)$.

C. Lemma IV.3

Proof: $F_{cD}(\tau)$ can be partitioned conditioning on the number of arrivals and departures at nodes i and j during the time interval τ. Using $P_x(y)$ as a shorthand notation for $\Pr\{x = y\}$,

$$\begin{aligned}
F_{cD}(\tau) = &\sum_{i^+=0}^{\infty}\sum_{i^-=0}^{\infty}\sum_{j^+=0}^{\infty}\sum_{j^-=0}^{\infty} P_{a_i}(i^+)P_{d_i}(i^-)P_{a_j}(j^+)P_{d_j}(j^-) \\
&\Pr\left\{T_{cD} < \tau \mid a_i = i^+, d_i = i^-, a_j = j^+, d_j = j^-\right\}
\end{aligned}$$

$$\begin{aligned}
&< \sum_{i^+=0}^{\infty}\sum_{j^-=0}^{\infty} P_{a_i}(i^+)P_{n_j}(j^-)\Pr\left\{T_{cD} < \tau \mid a_i = i^+, d_j = j^-\right\} \\
&= \left(\sum_{i^+=0}^{A_i}\sum_{j^-=0}^{D_j} + \sum_{i^+=0}^{\infty}\sum_{j^-=D_j}^{\infty} + \sum_{i^+=A_i}^{\infty}\sum_{j^-=0}^{\infty} - \sum_{i^+=A_i}^{\infty}\sum_{j^-=D_j}^{\infty}\right) \\
&\quad P_{a_i}(i^+)P_{n_j}(j^-)\Pr\left\{T_{cD} < \tau \mid a_i = i^+, d_j = j^-\right\} \\
&< \sum_{i^+=0}^{A_i}\sum_{j^-=0}^{D_j} P_{a_i}(i^+)P_{n_j}(j^-)\Pr\left\{T_{cD} < \tau \mid a_i = i^+, d_j = j^-\right\} + \\
&\quad \epsilon + \epsilon - \epsilon^2
\end{aligned}$$

Observe that if $m(f_j^* - D_j) - (f_i^* + A_i) > 0$ then $M = 0$, and $F_{cD}(\tau) < (2\epsilon - \epsilon^2) < 2\epsilon$.

D. Lemma IV.4

Proof: T_j is an integer by the definition of gcd. Since w_i and w_j are integers $\gcd(2w_i - 1, 2w_j + 1)$ is an integer. Therefore,

$$\begin{aligned}
T_i &= mT_j \\
&= \frac{w_i - 0.5}{w_j + 0.5}\frac{2w_j + 1}{\gcd(2w_i - 1, 2w_j + 1)} \\
&= \frac{2w_i - 1}{\gcd(2w_i - 1, 2w_j + 1)}
\end{aligned}$$

is also an integer. Then,

$$\begin{aligned}
h(f_i + nT_i, f_j + nT_j) &= m(f_j + nT_j) - (f_i + nT_i) \\
&= mf_j - f_i + mnT_j - nT_i \\
&= h(f_i, f_j)
\end{aligned}$$

If $f_i^* > F_i$ then due to Lemma IV.2 F_{cD} can be approximated as a function of $h(f_i^*, f_j^*)$. Since, $h(f_i + nT_i, f_j + nT_j) = h(f_i^*, f_j^*)$, first passage probabilities starting from these states are equal.

REFERENCES

[1] A. A. M. Saleh and J. M. Simmons, "Evolution toward the next generation core optical network," *J. Lightw. Technol.*, vol. 24, no. 9, pp. 3303-3321, May 2006.

[2] T. Koonen, "Fiber to the home/fiber to the premises: what, where, and when?" *Proc. IEEE*, vol. 94, no. 5, pp. 911-934, May 2006.

[3] D. Cavendish, "Evolution of optical transport technologies: from SONET/SDH to WDM," *IEEE Commun. Mag.*, vol. 38, no. 6, pp. 164-172, June 2000.

[4] N. Ghani, "Regional-metro optical networks," in *Emerging Optical Network Technologies: Architectures, Protocols and Performance*, K. Sivalingam and S. Subramaniam, eds. Springer, ch. 4.

[5] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang, "Experience in measuring Internet backbone traffic variability: models, metrics, measurements and meaning," in *Proc. International Teletraffic Congress (ITC-18)*, Berlin, Germany, Aug. 2003, pp. 221-230.

[6] K. Fukuda, K. Cho, and H. Esaki, "The impact of residential broadband traffic on Japanese ISP backbones," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 1, pp. 15-22, Jan. 2005.

[7] C. Kattirtzis, E. Varvarigos, K. Vlachos, G. Stathakopoulos, and M. Paraskevas, "Analyzing traffic across the Greek school network," in *Proc. 14th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN05)*, Crete, Greece, Sept. 2005.

[8] D. Cavendish, K. Murakami, S.-H. Yun, O. Matsuda, and M. Nishihara, "New transport services for next-generation SONET/SDH systems," *IEEE Commun. Mag.*, vol. 40, no. 5, pp. 80-87, May 2002.

[9] A. A. M. Saleh and J. M. Simmons, "Architectural principles of optical regional and metropolitan access networks," *J. Lightw. Technol.*, vol. 17, no. 2, pp. 2431-2448, Dec. 1999.

[10] G. Kramer and G. Pesavento, "Ethernet passive optical network (EPON): building a next-generation optical access network," *IEEE Commun. Mag.*, vol. 40, no. 2, pp. 66-73, Feb. 2002.

[11] C. Buyukkoc, P. Varaiya, and J. Walrand, "The c^1-rule revisited," *Advances in Applied Probability*, vol. 17, no. 1, pp. 237-238, Mar. 1985.

[12] J. Yates and A. Greenberg, "Reconfiguration in IP over WDM access networks," in *Proc. Optical Fiber Communication Conference (OFC00)*, p. 165.

[13] I. Baldine and G. N. Rouskas, "Traffic adaptive WDM networks: a study of reconfiguration issues," *J. Lightw. Technol.*, vol. 19, no. 4, pp. 433-455, Apr. 2001.

[14] M. Fisher, C. Kubicek, P. McKee, I. Mitrani, J. Palmer, and R. Smith, "Dynamic allocation of servers in a grid hosting environment," in *Proc. 5th IEEE/ACM International Workshop on Grid Computing GRID04*, Pitssburgh, PA, Nov. 2004, pp. 421-426.

[15] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 2005.

[16] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2000.

[17] M. H.-B. T. Osogami, "A closed form solution for mapping general distributions to minimal PH distributions," in *Proc. International Conference on Performance Tools (TOOLS03)*, Urbana, IL, Sept. 2003, pp. 200-217.

[18] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2000.

[19] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," Digital Equipment Corporation, MA, tech. rep. DEC-TR-301, Sept. 1984.

[20] P. G. Harrison and W. J. Knottenbelt, "Passage-time distributions in large Markov chains," in *Proc. ACM SIGMETRICS*, Marina Del Rey, CA, June 2002, pp. 77-85.

[21] J. Abate and W. Whitt, "Numerical inversion of Laplace transforms of probability distributions," *ORSA J. Computing*, vol. 7, no. 1, pp. 36-43, 1995.

Emre Yetginer received his B.S. and M.S. degrees from Bilkent University, Turkey, both in electrical and electronics engineering, in 1999 and 2002, respectively. During 1999-2000, he was with the Bilkent University Center for Communications and Spectrum Management. Since 2000, he has been a research scientist at the Scientific and Technical Research Council of Turkey - National Research Institute of Electronics and Cryptology (TUBITAK-UEKAE). He is currently working towards a PhD degree in electrical and electronics engineering at Bilkent University.

Ezhan Karasan received B.S. degree from Middle East Technical University, Ankara, Turkey, M.S. degree from Bilkent University, Ankara, Turkey, and Ph.D. degree from Rutgers University, Piscataway, New Jersey, USA, all in electrical engineering, in 1987, 1990, and 1995, respectively. During 1995-1996, he was a post-doctorate researcher at Bell Labs, Holmdel, New Jersey, USA. From 1996 to 1998, he was a Senior Technical Staff Member in the Lightwave Networks Research Department at AT&T Labs-Research, Red Bank, New Jersey, USA. He has been with the Department of Electrical and Electronics Engineering at Bilkent University since 1998, where he is currently an associate professor. Dr. Karasan is a member of the Editorial Board of Optical Switching and Networking journal. He is the recipient of 2004 Young Scientist Award from Turkish Scientific and Technical Research Council (TUBITAK), 2005 Young Scientist Award from Mustafa Parlar Foundation and Career Grant from TUBITAK in 2004. Dr. Karasan received a fellowship from NATO Science Scholarship Program for overseas studies in 1991-94. Dr. Karasan is currently the Bilkent team leader of the FP6-IST Network of Excellence (NoE) e-Photon/ONe+ and FP7-IST NoE BONE projects. His current research interests are in the application of optimization and performance analysis tools for the design, engineering and analysis of optical networks and wireless ad hoc/sensor networks.