Computer Communications 35 (2012) 366–379

Contents lists available at SciVerse ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Selfish routing and wavelength assignment strategies with advance reservation in inter-domain optical networks

Francesco Palmieri [a,*], Ugo Fiore [b], Sergio Ricciardi [c]

[a] Seconda Università degli Studi di Napoli, Dipartimento di Ingegneria dell'Informazione, Via Roma 29, 81031 Aversa (CE), Italy
[b] Università degli Studi di Napoli Federico II, CSI, Complesso Universitario Monte S. Angelo, Via Cinthia, 80126 Napoli, Italy
[c] Universitat Politècnica de Catalunya (UPC), Departament d'Arquitectura de Computadors (DAC), Carrer Jordi Girona 1–3, 08034 Barcelona, Catalunya, Spain

ARTICLE INFO

Article history:
Received 1 February 2011
Received in revised form 28 July 2011
Accepted 13 October 2011
Available online 20 October 2011

Keywords:
Wavelength-routing
RWA
Multi-commodity network congestion games
Nash equilibrium

ABSTRACT

The main challenge in developing large data network in the wide area is in dealing with the scalability of the underlying routing system. Accordingly, in this work we focus on the design of an effective and scalable routing and wavelength assignment (RWA) framework supporting advance reservation services in wavelength-routed WDM networks crossing multiple administrative domains. Our approach is motivated by the observation that traffic in large optical networks spanning several domains is not controlled by a central authority but rather by a large number of independent entities interacting in a distributed manner and aiming at maximizing their own welfare. Due to the selfish strategic behavior of the involved entities, non-cooperative game theory plays an important role in driving our approach. Here the dominant solution concept is the notion of Nash equilibria, which are states of a system in which no participant can gain by deviating unilaterally its strategy. On this concept, we developed a selfish adaptive RWA model supporting advance reservation in large-scale optical wavelength-routed networks and developed a distributed algorithm to compute approximate equilibria in computationally feasible times. We showed how and under which conditions such approach can give rise to a stable state with satisfactory solutions and analyzed its performance and convergence features.

1. Introduction

The large potential bandwidth available in next generation wavelength-division multiplexed (WDM) optical networks makes this technology of crucial importance for satisfying the ever-increasing capacity requirements in communication networks. Such networks will be based on dynamically configurable switching nodes, connected though a mesh of fiber links and operating transparently at the wavelength layer according to several automatic control plane strategies and protocols. These nodes set up and tear down, on a customer's request basis, pure photonic end-to-end communication channels (lightpaths) that can traverse multiple physical links on a common wavelength and essentially create a virtual topology on top of the physical topology. Information sent via a lightpath does not require to be converted from the optical to electrical form when passing through an intermediate node and converted back to the optical domain for retransmission to the next station, greatly reducing delay and latency phenomena and achieving transfer rates in the order of tens of THz. The efficient allocation of lightpaths on the fiber mesh, given a set of requests between pairs of nodes wishing to communicate through a dedicated end-to-end channel, poses several interesting theoretical problems. Given an optical network and a set of end-to-end communication requests, the routing and wavelength assignment (RWA) problem concerns routing each request on the optical transport network, and assigning wavelengths to these routes so that the same wavelength must be assigned along the entire route (*wavelength continuity constraint*), by realizing a lightpath [1]. Obviously, lightpaths that share a common physical link cannot be assigned the same wavelength (*clash constraint*). The objective of the RWA problem, that has been shown to be NP-complete [2], can be usually associated to the optimization of the overall network resources usage together with the minimization of the number of wavelengths used, or the maximization of the number of lightpaths successfully set up subject to a limited number of available wavelengths. However, if the needed wavelength resources are not immediately available at the request time, the connection setup will be blocked and the associated request refused. This may be intolerable for all the network users that require connection services being set up within a specified time frame and for a specified duration, according to a request/booking schema. To provide such services, it is desirable that the network resource control and management logic support advance reservations, i.e. reserving wavelength resources in advance respect to when they are actually

* Corresponding author. Tel./fax: +39 081503 7042.
E-mail addresses: francesco.palmieri@unina.it (F. Palmieri), ugo.fiore@unina.it (U. Fiore), sergio.ricciardi@ac.upc.edu (S. Ricciardi).

0140-3664/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2011.10.010

needed. This is obviously another useful network service model, which cannot only provide guaranteed services to network users but also allow networks to better plan their wavelength allocations. In fact, advance reservations of network resources are especially useful in environments that require reliable synchronized allocations of various resource types at different locations. Such *co-allocations* are necessary in order to assure that all the resources required are available at a given time. Each request specifies an end-to-end connection between two involved nodes, with a specific duration and a scheduling window, i.e. the time period within which the requestor would accept the connection to be set up. The flexibility of network-aware *advance resources reservation* introduces a new temporal dimension into the overall resource allocation problem. To support advance reservations, an RWA algorithm must take into account not only the network's spatial and topological characteristics (links, wavelengths, traffic matrix) but also their temporal characteristics. This would greatly increase the computational complexity of an RWA algorithm. Furthermore, in a real-world large-scale scenario the switching nodes and fiber links are owned and managed by several independent socio-economic organizations often operating in a non-cooperative fashion. According to the distributed nature of the Internet, in fact, these entities typically prefer to take almost unilateral decisions, such as selecting a path to route a connection request from one of their customers, in order to optimize *their own* resource usage and, of course, maximize their revenue. The lack of a central regulation forcing all the nodes to behave according to a common strategy makes network-wide resource optimization very difficult or even impossible. It should also be noted that end-to-end lightpath selection schemes are selfish by nature in that they allow the providers handling the connection request to greedily select the best available routes to optimize their own performance without considering system-wide criteria. Hence, the understanding of the mechanisms behind the selfish behavior of the involved entities in such non-cooperative network systems is of primary importance in resolving large-scale RWA problems where each organization that has to route a set of end-to-end connection request is driven by completely different and even conflicting measures of performance and optimization criteria. A natural framework in which to study such multi-objective optimization problems is the classic game theory. In such a context, our optimization problem can be modeled as a non-cooperative game of independent entities (*players*). These entities do not operate according to a common strategy and act in a purely selfish manner, aiming to maximize their own objective functions. The algorithmic game theory predicts that selfish behavior in such a system can lead to a Nash equilibrium, that is a state of the system in which no player can gain by unilaterally changing its strategy [3]. This approach can be used to optimize global objective functions taking into account the selfish behavior of the participating entities. That is, in such situations where it is difficult or even impossible to impose optimal routing strategies on network traffic, we exploit some less evident interaction dynamics between all the player's choices so that selfish behavior leads to a socially desirable outcome. Players, according to an advance reservation scheme, selfishly choose their private strategies, which in our environment correspond to best paths from their sources to their destinations, apparently without considering the other players' strategies. In doing this, our schema, ensures that, at the beginning of each reservation, specific additional taxes/marginal costs – associated to the conflicts with the other strategies insisting on the same resources – are bounded to the network resources. These costs can implicitly condition the selection process so that the global game may be kept into an equilibrium state. In other words, although each connection request is handled selfishly, it is deterministically assigned on its minimum-latency path (considering the network "congestion" effect due to the other players' impact), from which the corresponding entity/player has no incentive to deviate unilaterally. Extensive simulations have been carried out to evaluate the performance and scalability of the proposed approach in terms of tolerance to large number of simultaneous advance reservation requests as well as in slow connection rejection rate growth in presence of increasing network congestion. Good performance, limited cooperation between nodes and low computational complexity make the proposed model attractive for the future optical wavelength switched Internet.

2. Background

This section briefly introduces some of the concepts that will be useful to better explain the RWA optimization approach, by presenting the existing related literature together with the underlying architectural scenario, the basic assumptions, building blocks and modeling details as long as the theory behind them.

2.1. Related work

The RWA problem in large-scale all-optical networks has been intensely studied. In recent years, many research efforts have targeted the improvement in the efficiency of the management and control layer, following several directions, among which fuzzy ILP [4], and resource-criticality based heuristics aiming to delay as much as possible the utilization of critical resources, reserving them for future lightpath demands [5,6]. Advance reservation in optical networks has also been extensively studied. Zheng et al. [7] present a basic framework for automated provisioning of advance reservation services based on GMPLS protocol suites. In [8], a simulated annealing based algorithm is proposed to find a solution on predetermined k-shortest paths. Lee et al. [9] propose an efficient Lagrangean relaxation approach to resolve advance lightpath reservation for multi-wavelength optical networks. Other works (for example [10]) concentrate on the flexibility in reserving the connections, considering that clients may prefer a moderate delay in the start time of their request rather than having a request blocked. Finally, the exploitation of game theoretic approaches based on the analysis of uncooperative interactions and Nash equilibria in communication networks gave rise to a vast literature [11,12]. The work in [13] analyzed Nash equilibria, by considering their Price of Anarchy (PoA) and Price of Stability (PoS), in selfish routing games on multiple parallel links, where each player desires to minimize his experienced transmission time and seeks to communicate a message by choosing one of the links. In [14] the authors studied atomic routing games on networks, where each player chooses a path to route the traffic from an origin to a destination, with the objective of minimizing the maximum congestion on any edge of the path. Selfish path coloring in single fiber all-optical networks has been studied in [15,16], where the authors investigate the existence and performance of Nash equilibria, considering several information levels of local knowledge that players may have and give bounds for the PoA in chains, rings and trees. Selfish routing games have also been explicitly studied in ring networks [17] by adopting the asymmetric atomic routing model with a load-dependent linear latency on each link. The work in [18] analyzed the existence and complexity properties of pure Nash equilibria and best-response strategies in congestion games with time-dependent costs, in which travel times are fixed but QoS varies with load over time. The complexity of recognizing and computing Nash equilibria under various payment functions has been also studied in [19] where Fanelli et al. analyzed the payment functions in two different settings, both characterized by the objective of minimizing the total number of wavelengths used and minimizing of the number of converters needed. The PoA of selfish routing and path coloring, under payment functions that charge a player only according to his own strategy is discussed in [20,21]. Selfish path multi-coloring games where routing decision are taken in advance and players choose only colors are introduced in [22], providing bounds for the pure price of anarchy and also constant bounds for the PoA in specific topologies.

2.2. Network congestion games

Rosenthal [23] introduced a class of games, called congestion games, in which each player chooses a particular subset of resources out of a family of allowable subsets for him (its *strategy set*), constructed from a basic set of primary resources for all the players. The cost or delay associated with each primary resource e is a non-decreasing function $c_e(x)$ of the number of players x who choose it, and the total cost received by each player is the sum of the costs associated with the primary resources he chooses. In a *multi-commodity network congestion game*, each player is associated to a traffic flow to be routed throughout a network and its strategy set is represented as a set of origin–destination paths in such a network, whose edges play the role of resources. The flow may be *unsplittable*, in which case each player must choose a single path for its entire flow, or *splittable*, if the opposite is true. Furthermore, in the *atomic* case, there are a finite number of players, each with a specific amount of flow to route whereas, in the non-atomic case, there are an infinite number of players, and each one controls only a negligible fraction of the total flow. In addition, a *weighted congestion game* allows users to have different demands for service and, thus, affects the resource delay functions in a different way, depending on their own weights. In modeling the RWA optimization problem we refer to the atomic unsplittable model, where players have to route their connection demands along a single path (as general case, a demand may be split into n flows, but in the optical domain these streams will appear as n unsplittable optical flows). In such a *multi-commodity network congestion game*the strategy set of each player is represented as a set of origin–destination paths in a network, where the adjacencies between nodes and the associated weights/costs play the role of resources. A game with $n \geqslant 2$ players is defined by a finite set of strategies S_i with $i \in [1,n]$ where S_i denote all the possible strategies of the player i, and n cost functions $f_i : S_1 \times \cdots \times S_n \to R$, one for each player, mapping the set of all the possible strategies for each player to the real number set (some of the works present in the literature focus on payoff functions instead of cost functions; clearly, the difference is only a change in sign). The elements of $S_1 \times \cdots \times S_n$ are called states. The possible strategies for each player are implicated by both the topology of the network and the cost associated to each link. A *pure strategy profile*, or simply strategy profile, is a vector $\overrightarrow{S} = (s_1, \ldots, s_n)$ of deterministically chosen strategies, one for each player. Starting from the strategy profiles for all players and given a set of the strategies unilaterally chosen from each player, we say that the game is in an *equilibrium* if no player can decrease its own cost by changing its choices. This equilibrium concept was first introduced by John Nash [24] and it is known as Nash equilibrium. Such equilibrium defines a fundamental point of stability within the system, because no player can unilaterally perform any action to improve its situation. It is very interesting to explore the existence of *pure* Nash equilibria (PNE) in such games: a strategy profile is a *pure Nash equilibrium* if for each player i it holds that:

$$f_i(s_1, \ldots, s_i, \ldots, s_n) \leqslant f_i(s_1, \ldots, s'_i, \ldots, s_n) \tag{1}$$

for any strategy $s'_i \in S_i$.

Although Nash showed that each non-cooperative game can converge to a Nash equilibrium, the existence of a PNE is an open question for many games. Moreover, due to the selfish behavior of the players, such a pure equilibrium does not necessarily optimize a global goal. Such a goal is also known as the *social cost* of a strategy profile $\overrightarrow{S}$, defined as:

$$sc(\overrightarrow{S}) = \max_{i \in [1,n]} f_i(\overrightarrow{S}). \tag{2}$$

Depending on the involved cost function, the players' selfish behaviors might not optimize the social cost. It is also well known that a Nash equilibrium does not necessarily need to minimize the social cost. At the other end, the network management objective is minimizing the *social cost* measured by the total cost incurred by all players. The global performance of Nash equilibria is measured by the so-called *Price of Anarchy* (*PoA*) or *coordination ratio* which is defined as the ratio of the social cost of the worst Nash equilibrium over the optimal solution [25], and reflects the loss in the global performance due to lack of coordination between players:

$$PoA = \frac{\max_{\overrightarrow{S} \text{ is a NE}} sc(\overrightarrow{S})}{opt} \tag{3}$$

where

$$opt = \min_{\overrightarrow{S} \in (S_1 \times \cdots \times S_n)} f_i(\overrightarrow{S}) \tag{4}$$

denotes the optimum social cost for a game.

Clearly, a game with a low Price of Anarchy can be reasonably left almost unconditioned, since the involved selfish players — by virtue of being selfish — are guaranteed to achieve an acceptable performance. On the other hand, in presence of a large Price of Anarchy, it is necessary to introduce some *social control* and *coordination mechanisms* (such as taxes, costs or incentives, etc.) that implicitly force players to collaborate more efficiently. Some congestion games admit a potential function defined over the set of pure strategy profiles, with the property that the gain of a player unilaterally shifting to a new strategy is equal to the corresponding increment in the potential. It has been shown [26] that the existence of such a potential function implies that at least one Nash equilibrium exists. Formally, a real function $\Phi(\overrightarrow{S}) : S_1 \times \cdots \times S_n \to R$ is a β-potential function if it has the property that:

$$\begin{aligned} &f_i(s_1, \ldots, s_i, \ldots, s_n) - f_i(s_1, \ldots, s'_i, \ldots, s_n) \\ &\quad = \beta_i \cdot (\Phi(s_1, \ldots, s_i, \ldots, s_n) - \Phi(s_1, \ldots, s'_i, \ldots, s_n)), \end{aligned} \tag{5}$$

where the β_i are the real-valued components of a vector β. The effect on the cost function f_i of a strategy change by player i will then be the projection, weighted through the vector β, of the variation in potential associated to the change, so that local minima of the potential function will correspond to Nash equilibria. Such equilibria exist, and can be computed in pseudo-polynomial time, in games with linear cost functions $c_e(x)$ associated to the individual resource e [3], where each $c(x)$ function can be defined as:

$$c_e(x) = a_e x + b_e, \tag{6}$$

where $a_e, b_e \geqslant 0$ are constant values conditioning the cost function trend.

2.3. Marginal costs in congestion games

As already sketched in the previous section, to mitigate the performance degradation due to the players' non-cooperative and selfish behavior, we can introduce some incentives that influence the players' selfish choices and hopefully induce an optimal network configuration. These incentives can be naturally modeled by non-negative per-unit-of-traffic *taxes* (or prices) assigned to the resources. Such taxes become an additional cost factor, which the players should take into account. Simply stated, a player's cost for adopting a strategy should be calculated by adding such marginal cost associated to choosing a specific resource to the latency due to the resource's congestion. Although these additional costs increase the players' individual cost, they do not affect the social cost because they are payments inside the system and can be feasibly ''refunded'' to the players. The goal is to find a set of moderate and efficiently computable *optimal marginal costs*, which make the Nash equilibria of the modified game coincide with the optimal solution. Designing optimal taxes is a central topic in game theory. In general, any traffic equilibrium reached by the selfish players who are conscious of both the resource usage latencies and the taxes will minimize the social cost, that is, will minimize the total latency [27]. According to [28], we can formally define the marginal cost associated to a resource e by:

$$c_e^*(x) = c_e(x) + x \cdot c_e'(x), \tag{7}$$

where $c_e'(x)$ denotes the derivative $\frac{d}{dx}c_e(x)$.

Observe that the function $c_e^*(x)$ describing the marginal cost of increasing traffic on the resource e is composed by a first term capturing the per-unit latency incurred by the additional traffic introduced by the other players' choosing e and a second one accounting for the increased congestion experienced by the traffic already using the resource. Essentially, the only difference between an optimal route assignment and an assignment in the context of a Nash equilibrium is that the former accounts for this ''conscious'' second term while the latter disregards it.

3. The reference model

We will model our approach to the RWA problem with a game theoretic formulation by working in an atomic unsplittable weighted multi-commodity scenario where the communication resources are booked and allocated according to a time-slotted advance reservation paradigm. This means coping with a ''scheduled'' traffic model where the setup and teardown times of the demands are known in advance.

It is common in this setting to view the wavelength routed network as a connected graph with its nodes being the optical switching nodes and its edges being the available wavelengths (i.e. different channels) on the optical fibers that provide the actual communication. Since each fiber link can support several WDM channels, there is typically more than one edge connecting the same pair of nodes. The resulting structure is a multi-graph and its construction process is sketched in Fig. 1 below.

To keep the formulation as general as possible, we make no specific assumption on the number of wavelengths per fiber and the number of fiber on each link. All these parameters are fully and independently configurable at the network topology definition time. Nevertheless, we require that all the network nodes operate under a unique control-plane providing a common link-state routing protocol and a signaling facility to handle resource reservations (such as those provided in a multi-domain GMPLS-like framework [29]). Furthermore, we assume that every connection is bidirectional and consists in a single atomic traffic flow that cannot be split between multiple paths (as we have seen, such assumption does not cause any loss of generality). Each connection request, viewed as an independent player in our game-theoretic approach, can be satisfied by establishing a single lightpath between its source and destination nodes. Notice that, to enforce the continuity constraint, this path can only be built on edges associated to the same wavelength. We are given a network (graph) $G = (V,E)$ and a set of end-to-end connection requests $R = (r_1, r_2, \ldots, r_{|R|})$ arriving as an ordered sequence according to a Poisson process with exponentially distributed call-holding time. In our advance reservation model, the time is slotted with a slot size equal to t', where this length depends on the minimum duration of an advance reservation (Fig. 2). Each advance reservation request r_i can only start at the beginning of a timeslot and is described by a 5-tuple, (u,v,d,s,e) where u and v are the nodes in G that are the connection's ingress and egress points, d is the reservation duration expressed in time slots, and s and e are the starting and ending time of the *scheduling window*. The scheduling window defines the acceptable set-up time range of the connection request, so that if the needed connection cannot be established within such time period, the request will be withdrawn. The window size may be fixed if the start and end times of the connection cannot be altered or flexible when those time limits can slide within a larger window. Several integer linear program formulations and algorithms have been proposed to solve these problems [30,31]. In our work we will consider dynamic end-to-end connection requests that belong to a fixed scheduling window.

Despite losing granularity, the above time slotted model allows the reduction in required processing capacity and increases scalability. When applying it to wavelength routed optical networks we obtain a multi-dimensional resource management scenario with hops, wavelengths and time slots. An online instance of the RWA problem is denoted by (G,R) and is defined as the task of finding an assignment of valid single-wavelength paths, at the granularity of a timeslot and for an integer number of timeslots, to a subset of requests $A \subseteq R$ with different wavelengths for overlapping paths, such that $|A|$ is maximal. This is an online scheduling problem because the requests arrive dynamically and, at each time slot, for each request $r_i \in R$, we check if it is inside its validity time range and, if so, compute a path and check whether a common wavelength on each link of this path can be reserved for its duration d. If such a suitable path is not available, the involved connection request will be deferred to the next time-slot, and this process will be iterated until either the request is satisfied or its time window expires. In order to implement this advance reservation mechanism, the RWA logic needs to maintain a schedule of the valid reservations called the reservation table. Also, the network nodes must work in a synchronized way according to a common reference clock. A strategy s_i for player i is a pair (p_i,d_i) where p_i is a simple path connecting the endpoints of r_i and d_i is its requested duration, implicitly associated to all the edges in p_i. Each player's strategy set consists of k different source–destination paths $(s_1, \ldots, s_k)$, corresponding to the first k available minimum cost path choices. For example, these strategies may include the

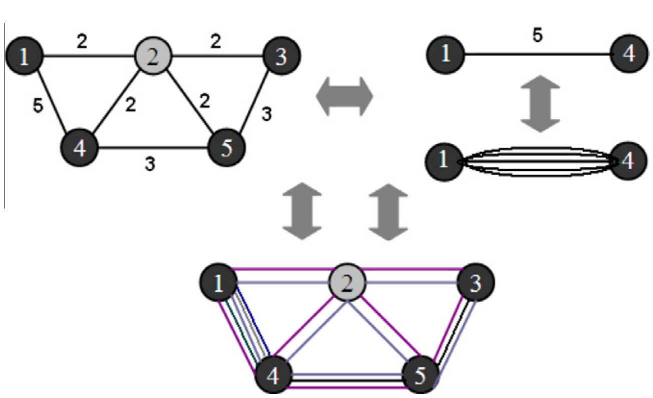

Fig. 1. Generating the working multi-graph.

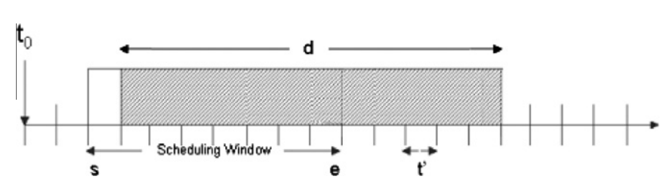

Fig. 2. Connection set-up time range: scheduling window.

first-shortest-path route, the second-shortest-path route, the third-shortest-path route, etc. Hence, the best strategy/path for a connection request can only be chosen from this set. Any feasible path from the source node to the destination node can be a candidate as the actual strategy for satisfying a connection request. The choice of a strategy instead of another one depends on the overall satisfaction of all the players/request and hence on the reachability of an acceptable pure Nash equilibrium status. A routing strategy preferentially choosing the paths with minimal number of hops tends to minimize resources utilization in terms of nodes involved in routing data traffic for the same source–destination pair. On the other hand, paths with low number of hops are expected to be more robust to failures and easy to control/monitor.

We define *social cost* of our problem as the total number of edges needed for routing a given set of requests. Minimizing this quantity is particularly important in cases where fibers are hired or sold *as a whole*. It is straightforward to verify that the social cost of a strategy profile coincides with the maximum player loss of utility in that profile. To quantify the loss in network performance caused by selfish behavior, we investigate the following question: what is the worst-case ratio between the social cost of an uncoordinated outcome and the social cost of the best-coordinated outcome? Hence, the *price of anarchy* of such a game is given by the worst-case number of edges used in a Nash equilibrium (social cost) divided by the optimum achievable social cost, that is, the minimum number of edges that can be used.

4. The two-stage algorithm

In this section we detail our RWA schema, based on a two-stage approach natively conceived to work on large and complex optical transport networks where little or no coordination can be assumed among the participating entities (a common case in presence of multiple independent administrative domains/autonomous systems), properly conceived to cope with the known drawbacks of the state of the art routing algorithms (lack of *global* optimization objectives). Our main goal is to minimize the total blocking probability by optimizing wavelength usage together with the cost and length of designed paths, while keeping the network resource usage fairly balanced, trying to leave on each link sufficient bandwidth to satisfy further requests as much as possible.

While ideally operating in a non-cooperative fashion, all the entities involved in the proposed RWA framework need to be synchronized in some way to share (and manage) a common view of the network topology as long as link resources usage and status. This implies that every node has to run a distributed control-plane providing the necessary link state routing and signaling protocols [29]. An OSPF or ISIS-like protocol can be used to distribute wavelength/label usage and cost information for each link at the optical layer and bandwidth occupation at the IP one. In the case of OSPF, for example, the opaque LSA facility, augmented with new TLVs can support the additional control information to be exchanged among nodes, such as candidate strategies/paths together with marginal costs/allocation-dependent taxes. An extended signaling/reservation protocol, such as RSVP-TE or CR-LDP within the GMPLS framework, can be used to handle all the resource reservation and allocation operations required during the network activity. Also, a common time synchronization is necessary between the network nodes accepting and routing the incoming end-to-end connection requests. Accordingly, a simplified slotted model has been chosen where in each time slot we can distinguish two distinct stages: the reservation and the allocation phase. The reservation phase will start at the beginning of the time slot, with the network state being the result of the allocation phase that happened at the end of the previous time slot. During the reservation phase, each pending connection request within its scheduling window will act as a player. Each player will selfishly choose its own strategy, based on its knowledge of the network state, by looking for the lowest-cost feasible path with a common wavelength. If such a path cannot be found, the connection request will be deferred to the next time slot, if that is still within its scheduling window, otherwise the connection request cannot be honored in the required time range and hence will be discarded. As a consequence of routing connections according to the chosen strategies, players will experience an additional latency (in the game-theoretic sense) caused by the occupation of the available wavelengths on each physical connection between adjacent nodes. This phenomenon can be handled by introducing a marginal cost model properly weighting the proposed strategies by keeping into account the impact of all the proposals and, hence, considering all the available strategy profiles. The principle of marginal cost pricing asserts that on each edge, every player whose strategy is described by a route crossing it, should pay a tax proportional to the additional latency its presence causes for the other players on such edge.

An assignment of edges to paths motivated only by selfish considerations (its associated *Nash equilibria*) does not minimize the total latency; put differently, the result of local optimization by many selfish network users with conflicting interests does not possess any type of global optimality; that is, this lack of regulation carries the cost of decreased network performance. Hence the outcome of selfish behavior can only be improved upon with some form of coordination. The inefficiency of selfish routing (and, more generally, of Nash equilibria) motivates strategies for *coping with selfishness*, that is, introducing methods for ensuring that non-cooperative behavior results in a socially desirable outcome. Accordingly, we have to consider that whenever each player tries to minimize its private cost, expressed in terms of its individual latency, we need a common decision point where each strategy (path) has to be communicated to all other nodes, letting them to build the strategy profile vectors $(s_1, s_2, \ldots, s_{|R|})$ required to construct their final strategies within the congestion game. This information must be made available to all the participating nodes through the aforementioned link state advertisement/update mechanism available at the control plane layer. For each proposed path, the edge costs need to be updated, by computing their associated marginal cost, to account for the candidate reservations that have been proposed within the strategy profiles made available to all the nodes. In simple words, the performance degradation due to the selfish and non-cooperative behavior of the independent players can be mitigated (or even eliminated in the best conditions) by introducing an appropriate set of marginal costs proportionally taxing each connection resource according to the global demand (end hence the degree of conflict on each resource) of all the independent strategies. These marginal costs implicitly charge each network connection/player for the congestion effects caused by its presence. A player, whose ingress node receives a status update, re-computes the next element in its strategy set and uses the strategy/path information obtained by the other players to build an updated strategy profile. The player checks if the costs along the path constituting its original strategy have been updated. If they have not, the player does not change its previous strategy. Otherwise, the player re-adapts the cost to account for the choice it had previously made, by decreasing the costs along its preferred path as if its own reservation had not been made. Note that this step prevents instability: a player would otherwise keep bouncing between its two best choices, if the difference between their total costs were less than the "tax" induced by the reservation. Then, the player computes again a lowest-cost path according to the updated costs, which can be seen as the next choice in its strategy set. If it finds a more satisfactory solution, it makes a strategy change. It computes the necessary adjustments to the costs along the new path and communicates them, along with the updated costs on the old path, to all the participating nodes. The reservation phase terminates at the completion of each time slot; at this point, all the players have the complete knowledge of the network status and of all the proposed strategies. At the end of the reservation phase, each node obtains a strategy profile representing the best desiderata of each player. The use of a common control plane and a link state routing facility implies that all the nodes share a unique synchronized network view and result in the calculation of the same strategy profile. Such profile may be compatible or not with the network resource limitations. In the first case, the strategy profile is a feasible solution of the allocation problems, in the second case it is partially unfeasible and a different solution must be obtained by shifting to the next time-slot the requests that could not be honored because of resource availability conflicts. In detail, players actually trigger resource allocation by issuing a provisional reservation for each resource on the path. If, at the end of this phase, any of the resources in the path is unavailable because it has been requested by other players, the current player will be deferred to the next time slot. The common signaling facility also ensures that all the nodes actually involved in the reservation and allocation of the links/wavelengths resources required in setting up an end-to-end connection (and hence directly involved with a player in our congestion game) have the same view of the resources seizure status independently from their role in the setup process (i.e. if they are originator, destination or transit nodes). The reason for having two phases is that if connection establishment had been allowed as soon as a successful reservation were made, the connection might have needed rerouting many times, since the process of computing a Nash equilibrium involves possibly many strategy changes. While rerouting a connection can be done in a few milliseconds, rerouting of "live" connection carrying user traffic is undesirable, since it is unavoidable, during rerouting, to cause a service disruption that, although momentary, is perceived by the final users.

Note that the order in which players operate in both the reservation and allocation phases plays a crucial role in the outcome of the overall scheme. Players acting later in the reservation phase have a greater probability to achieve their best (original) strategy, because players preceding them could have been forced to abandon their first choice in case of conflicts, and this will decrease the marginal cost of critical resources. Conversely, players acting first in the allocation phase will have an advantage in securing critical resources. Different network management schemas may choose different ordering criteria, depending on their priority objectives. The ordering may:

- be based on an *a priori* weighting of the connection request (maybe for financial reasons or the strategic importance of clients);
- reflect different priorities calculated from the residual request lifetime within the scheduling window, privileging those connection whose setup time range is about to expire, so that the blocking probability will be reduced;
- be conditioned by the connection duration, possibly favoring long-lasting (thus, more remunerative) connections;
- depend on an absolute measure of the impact on network resources, such as the length of the path requested, so that the social cost will tend to be reduced.

4.1. The resource cost and marginal function

To define a reasonable cost function we first have to evidence the required properties and dynamics characterizing such a function. It is intuitive that a *good* cost function should rank each edge proportionally to both the residual and the maximum number of wavelength available on the same pair of nodes. However, the two factors do not need to contribute equally. We have considered the use, for each edge, of the relative load, i.e. the ratio of the number of used wavelength over the total number of available parallel wavelengths. In addition, some provision must be made to appropriately penalize long paths over shorter ones, and to avoid that the cost of an empty link would be zero. Hence, we introduced an additive fixed nonzero cost to each edge. The resulting cost function is therefore the linear function:

$$c_e(x) = a\frac{x}{w_e} + b, \tag{8}$$

where x is the number of used wavelengths on link e, w_e is the total number of wavelength on all edges sharing the same pair of nodes in the multi-graph with edge e, and a and b are adjustable constants ($a > 0,\ b \geqslant 0$), whose value will be tuned by empirical considerations. The ratio between a and b will be determined by the number of hops that an alternative path must have in order to be considered roughly equivalent to the seizing of a single-hop congested link. If, for the sake of simplicity, b is taken to be 1, reasonable values for the number m of hops representing the length of "equivalent" alternative paths yield an estimate of a being near m when the load reaches about 75–90% of the total saturation.

Without introducing any other additional taxation criterion across the edges/resources composing a path, the congestion game players experience only their own traffic delay as their cost. By introducing edge taxation, players are also charged for the right to use edges across a path. This technique has been studied by the traffic community for a long time (e.g. [32] and the references therein), especially in the context of marginal costs [33].

Each selfish player i when using a path p_i will experience a total cost $\Gamma(p_i)$ obtained by combining its initial cost $\gamma(p_i)$ with the influence of the marginal costs $\mu(p_i)$:

$$\Gamma(p_i) = \gamma(p_i) + \tau_i \cdot \mu(p_i), \tag{9}$$

where $\gamma(p_i) = \sum_{e\in p_i} c_e(x_e)$, is the sum of the individual edge costs along the strategy/path p_i, being x_e the occupation of resource e at the beginning of the current time slot. On the other hand, the cumulative marginal cost function $\mu(p_i) = \sum_{e\in p_i} c^*_e(x_e)$ is the sum of the marginal cost taxes $c^*_e(x)$ along the edges of the path p_i. The factor $\tau_i > 0$, denotes the sensitivity of player i to the taxes. In the homogeneous case, all the players can have the same sensitivity to the taxation (i.e. $\tau_i = 1$, for all i), while in the heterogeneous case τ_i can take different positive values for diverse players. Through edge taxation, we aim to force all equilibria on the network to be reached by combining strategy profiles that minimize the social cost. In our approach, we can see the additional marginal cost taxes assigned to every edge as part of the edge latency function itself. Here, instead of taxation, we can speak about artificial delays introduced possibly at the entrance of each edge, in order to minimize the total "congestion" probability due to multiple players that need to traverse the same adjacency between two nodes, and hence the probability to be blocked at the ingress of the edges themselves. Accordingly, we assume that each player's strategy is further charged according to the maximum number of paths that share an edge with it and use the same wavelength. Applying Eq. (7), we can derive the marginal cost function as:

$$c^*_e(x) = 2a\frac{x}{w_e} + b. \tag{10}$$

Marginal cost taxes increase in general the cost for each player, as shown in [34]. The natural question that arises is whether taxes are an efficient mechanism for achieving the desired result. In other words, if the additional "disutility" caused through taxation proportionate to the desired goal, i.e. a routing assignment that minimizes the total latency. Our marginal cost taxes have been conceived as an implicit coordination mechanism obtained through a cost function properly chosen from a family of possible ones, according to a "coordination model" in the sense defined as in [35]. In particular, results presented in [27] suggest that for strictly increasing and differentiable linear latency functions, imposing properly chosen taxes on a selfish routing game not only yields to a game with better coordination ratio, but also that the added disutility for the players is bounded with respect to the original system optimum. That is, with a small decrease in network efficiency, we achieve, at equilibrium, a strategy profile that minimizes the total latency. From Eqs. (8)–(10), we can see that the total cost for an individual resource e still has a linear form in the occupation x. Hence, according to the results presented in [3], at least one pure Nash equilibrium exists and it can be computed in pseudo-polynomial time.

4.2. Determining a Nash equilibrium

The distributed algorithm starts on the network nodes with an initial strategy profile $S = (s_1, s_2, \ldots, s_{|R|})$ built on the selfishly chosen minimum cost paths for each request/player r_i in its valid time range. More precisely, on each time slot every node n selfishly calculates the strategies s_i for all the requests/players r_i in its locally originated requests set $R_n \subset R$, advertises them on the network and simultaneously learns, from the received advertisements (procedure *Advertise_and_Receive_Strategies*, line 6 in Fig. 3), the strategies proposed by the other nodes so that on each iteration it is able to construct a complete strategy profile S containing the strategies associated to all the valid players on the entire network. Then, after a recalculation of the total latencies associated to each path within S, performed by adding the marginal costs introduced by the proposed allocations of other players, it iteratively allows each unsatisfied player to recalculate another path, possibly reducing the associated cost. The algorithm iteratively strives to transform a non-equilibrium configuration into a pure Nash equilibrium, performing a sequence of greedy selfish steps, where each player switches to the path that minimizes latency, given the current strategy profile. Each greedy selfish step consists in a player on a node re-computing its minimum-cost path with respect to

```
procedure RWA(R_n, S)
Input:  local connection request set R_n
        global strategy profile S
1.  S ← ∅;
2.  do
3.      for each player i in R_n do
4.          Selfish_Iteration(S, C, i) // build local strategy set
5.      endfor
6.      Advertise_and_Receive_Strategies (S, C) // build global strategy set
7.  while ∃ i : Γ(s_i^*) ≤ Γ(s_i) OR ¬maxIter // until an equilibrium is found or the maximum number
    of iterations has been reached
8.  for each player i in R_n do  // start the resources allocation
9.      if Allocate (s_i) is successful then
10.         R ← R \ {i} // remove it from the global request set R
11.     endif
12. endfor
13. end procedure RWA
```

Fig. 3. The per-time slot RWA procedure.

```
procedure SELFISH_ITERATION(S, C, i)
Input:  current strategy profile S
        current costs C
        player ID i
Output: new strategy profile S*
        new costs C*
S* ← S \ {s_i}
C* ← UpdateMarginalCosts(S*, C)
s_i* ← MinCostPath(i, S*, C*)
if Γ(s_i*) ≤ Γ(s_i) then // if new strategy improves costs
    S* ← S* ∪ {s_i*} // add it to the strategy profile
    C* ← UpdateMarginalCosts(S*, C*)
    if S* ≠ S then
        Advertise(S*, C*)
        S ← S*
        C ← C*
    endif
endif
end procedure SELFISH_ITERATION
```

Fig. 4. An iteration of the Nashification algorithm.

the choices selfishly made by the other players and possibly changing its best pure strategy and diminishing its latency (cost). In other words, each node dynamically computes, in a stepwise fashion, its local strategy set by indirectly keeping into account (thanks to the marginal costs mechanism) the selfish choices of the players on the other nodes. The process terminates when an equilibrium is reached and no one of the participants is interested in changing its strategies or when a maximum number of iterations (*maxIter*) is reached. However, even without the *maxIter* performance constraint, the linearity of cost functions guarantees [3] the existence of a potential function (Section 2) that, in turn, ensures [26] that a pure Nash equilibrium always exists, so the distributed algorithm will terminate after a finite number of steps into a configuration in which no user has incentive to deviate. When an equilibrium strategy profile is available each node can allocate all the paths associated to its own players (line 8–12 in Fig. 3). Allocation of a path/strategy s_i (procedure *Allocate*, line 9 in Fig. 3) is accomplished by using the traditional two-directions forward provisional resource seizure (i.e. GMPLS RSVP-TE PATH request message) and backward reservation (i.e. RESV message) paradigm. If for a specific request/player this last step is not successful, the associated connection request is shifted (to be served) at the next time slot.

The procedure in Fig. 4 details the selfish iteration step described above. The procedure is run on each node for all the requests/players originating in that node, and it is triggered by the reception of an updated strategy profile. This is, in turn, the result of an invocation of the *Advertise*( ) procedure which, as previously asserted, is implemented through the link-state update mechanism of the control plane layer. After all nodes broadcast their updates with their new strategy, each node knows the entire strategy profile S^*. Procedure *UpdateMarginalCosts*( ) computes the new costs C^* associated with a strategy profile S^*, whereas *MinCostPath*( ) finds a minimum-cost path, by using the traditional Dijkstra algorithm, based on these costs. Finally, in line 7, an advertisement is produced if there is a strategy change.

5. Performance considerations

Let's now examine the computational complexity of the above framework for a network (graph) $G = (V,E)$ with $|V| = N$ nodes and $|E| = M$ edges. The *Selfish_Iteration*( ) procedure shown in Fig. 4 is built up by a number of simple sub-procedures whose complexity is analyzed in the following. Line 1 removes the path from the current strategy profile, while line 2 updates the marginal costs: these operations have both a cost of $O(N)$. The Dijkstra's shortest path algorithm is thus calculated in line 3 requiring $O(M + N \log N)$ [36]. Line 4 requires the computation of the path total costs $\Gamma(s_i^*)$ and $\Gamma(s_i)$ and all the links in the path are to be taken into account during this operation in which both delays and marginal costs are considered. In the worst case, the number of links of a simple path (i.e. a path without loops) in a network with N nodes is $N - 1$; since each of the operation involved in the cost calculation has a constant cost $O(1)$, the computation of the Γ factors costs at most $O(2N) = O(N)$. The strategy is added to the strategy profile at line 5 and the marginal costs of the link associated with the newly proposed path are updated at line 6, both operations requiring $O(N)$. If the strategy profile has changed (line 7), a link state update message is sent to other nodes in order to reflect the change (line 8) and the new values of S^* and C^* are stored in S and C respectively in lines 9 and 10; each of these operations has constant costs $O(1)$. For each node, the overall cost of the *Selfish_Iteration ()* procedure is thus given by $O(N) + O(M + N \log N) + O(N) = O(M + N \log N)$.

The *RWA*( ) procedure shown in Fig. 3 is executed by each network node which, after initializing the global strategy profile S at line 1 with a cost of $O(1)$, repeats (lines 3–5), for each player i in the local set R_n, $|R_n| = k$, the *Selfish_Iteration*( ) procedure to construct the local strategy set; at line 6, the node advertises its local strategies to the other nodes and receives their strategies to construct the global strategy set. These operations (lines 2–7) are repeated until there are no improvements or, in the worst case, the maximum number of iterations has been reached. Thus lines 2–7 have a complexity of $maxIter \cdot k \cdot O(M + N \log N)$. Lines 8–12 repeat the allocation phase for each of the k players and, possibly, remove them from the request set R, with a constant cost $k \cdot O(1)$. Therefore, the total cost of the *RWA*( ) procedure is given by $O(1) + maxIter \cdot k \cdot O(M + N \log N) + k \cdot O(1) = maxIter \cdot k \cdot O(M + N \log N)$. In the worst case scenario, before the call setup may be admitted, each network node repeats the *RWA*( ) procedure at each t' slot size, during a maximum time interval given by $(e - s)$ before the assigned time slot expires, as specified by the connection setup request. In the worst case the *RWA*( ) will be executed exactly $w = (e - s)/t'$ times, thus the total computational complexity is given by $w \cdot maxIter \cdot k \cdot O(M + N \log N)$ for each single node, which, as the results on the average delay show, is an affordable complexity. Nevertheless, these computations will be also done at different times; more precisely, each one will be done exactly after t' time units, for at most a time window of size w so that the parameter w must be carefully tuned in order to let each node compute its *Selfish_Iteration*( ) before a new attempt may be performed. In the next section, we study the choice of the parameter w along with the different results in terms of performance and stability.

6. Experimental evaluation and results analysis

In order to evaluate the functionality of the proposed selfish routing and wavelength assignment strategy, we conducted an extensive simulation study on several network topologies (modeled as undirected graphs in which each link has a non-negative capacity). In the following paragraphs we report the simulation details together with the most interesting results and observations emerged during the experiments.

6.1. The simulation environment

The evaluation of the proposed routing framework has been conducted in an optical network simulation environment [37] that allows the creation of heterogeneous network topologies along with the specification of simulation parameters and configuration options. Simulations have been performed on an HP® DL380 Dual Processor (Intel® Xeon® 2.5 GHz) server running FreeBSD® 4.11 operating system and Sun® Java® 1.4.2 Runtime Environment. In all the experiments, we used a dynamic traffic model in which connection requests, defined by a Poisson process, arrive with a parametric rate of λ requests/s and the call-holding time is exponentially distributed. The connections are distributed on the available network nodes according to a random-generated or predefined traffic matrix.

6.2. Results analysis

The results presented are taken from many simulation runs on several network topologies with various parameter and bandwidth unit request values, as summarized in Table 1.

As can be seen from the Table 1, 20 simulations per topology were executed and, to obtain more confidence in the results, each run has been repeated 10 times and the average performance metric values have been calculated. We considered several values for the parameters and measured the blocked connections and the convergence times of the illustrated "Nashification" process. The length of used time window w assumes values from the set

Table 1
Simulations performed and parameters used.

Parameters	Geant2/Internet2
Number of connections	Varying from 0 to 10,000 (step 100)
Random generated bandwidths (OC-unit)	{1,3,12,24,48,192} with different distribution probability
a, b, τ_i	Varying canonical values: $a = 1$, $b = 0$, $\tau_i = 1\ \forall i$
d, s, e	Varying according to Poisson process
w, *maxIter*	Varying in the range {2,3,4,5,6,7,8,9}
Number of simulations	20 simulations run per topology; each simulation repeated 10 times
Measurements	Blocked connections and experienced delays

specified in Table 1 in order to evaluate the algorithm when players have different time intervals during which they have to choose their strategy profiles. In our lambda-switched optical framework, the resources occupied by the routed connections are counted as sum of the ratio between the free and the busy bandwidths along the edges. Resources are thus represented as the sum of the bandwidths on all the network edges, while the traffic volume is represented by the quantity of the utilized bandwidth in a certain time. We tried out different static, predefined [38,39], or randomly generated traffic demand matrices on several network topologies, both randomly generated and well-known, such as Geant2 [40] and Internet2 [41] (Figs. 5 and 6) with the bandwidths for the links ranging from OC-1 to OC-768 bandwidth units. When we used the traffic matrices defined in [38,39] the traffic volumes have been scaled proportionally to the reported traffic distributions.

In our tests, each connection request was characterized by a bandwidth demand ranging from OC-1 to OC-192 (i.e. up to 10 Gbps) units, and the s, e and d reservation parameters for each connection request (Fig. 2) are generated according to a Poisson process (exponentially negative distribution). As the network load grows, that is, the number of busy connection resources increases more and more respect to the free/released ones, we continuously monitor the network efficiency expressed by the rejection ratio/blocking factor. During the simulations, the performance of the algorithm was tested against different values of the parameters of Table 1: the scheduling window size w, the weight factors a and b of Eqs. (8) and (10), and the order of the connection requests. The first simulation is to test the performance of the algorithm with varying time window sizes. The average blocking probability as function of the network load measured in Erlangs is shown in Figs. 7 and 8 (canonical values assumed for $a = 1$, $b = 0$). Results show that the blocking ratios grow quite regularly, but with some differences according to the time window w that has to be chosen. A time window $w = \{4,5\}$ achieves better performances in terms of blocking ratios with respect to too high ($w = \{6,7\}$) or too low ($w = \{2,3\}$) time windows that may drive to sub-optimal results. In fact, in both simulations the best performances have been achieved with parameter $w = \{4,5\}$, meaning that lower blocking may be achieved by giving only *some* chances to the nodes to change their strategy profile. The results obtained with a low value of the time windows w show that a sub-optimal network equilibrium is reached, but the too little available steps avoid the system to reach optimal configurations in most cases. Similarly, the results obtained with a high value of w indicate that a sub-optimal, but sustainable, network equilibrium is reached within too many steps from the initial strategy profile and that margin of optimizations are possible by decreasing the windows size. Thus, tuning the parameter it is possible to obtain a balance between the performances and the computational times (steps) needed to efficiently find good performances. In any case we can observe that blocking ratio grows quite slowly when the network load increases and that the highest, unacceptable values are physiologically reached only when the available network resources are almost totally saturated.

Average time delays experienced by connection requests during the simulations are plotted in Figs. 9 and 10. In particular, we analyzed how fast the average setup delays grow with the number of simultaneous players requesting in each time slot new end-to-end connections. In this scenario, the network load grows physiologically with the number of simultaneous requests but the connection

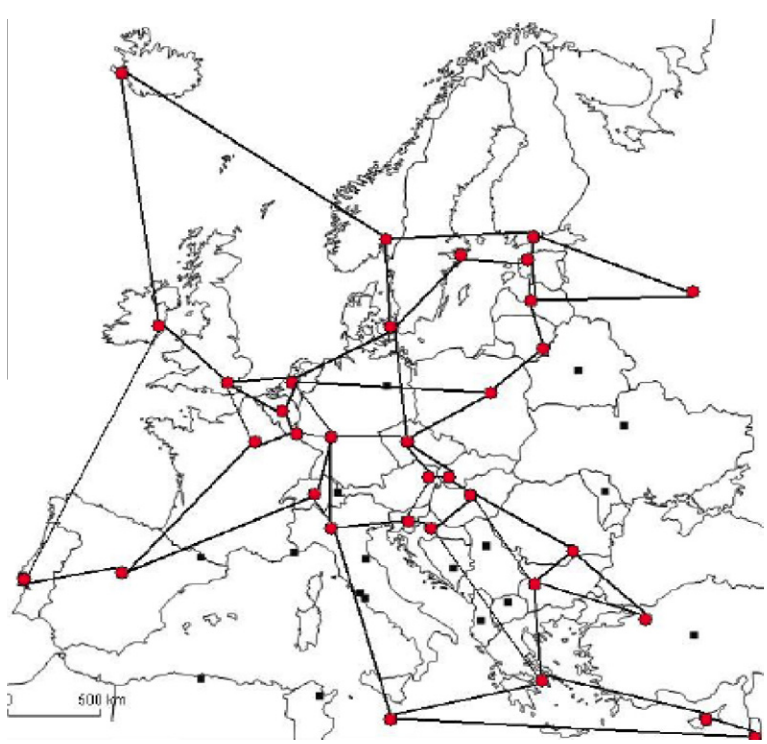

Fig. 5. Geant2.

Fig. 6. Internet2.

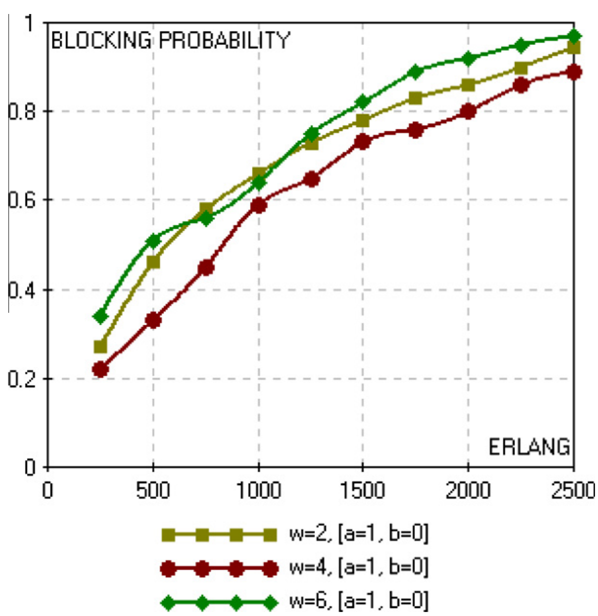

Fig. 7. Geant2 blocking probability, varying window size.

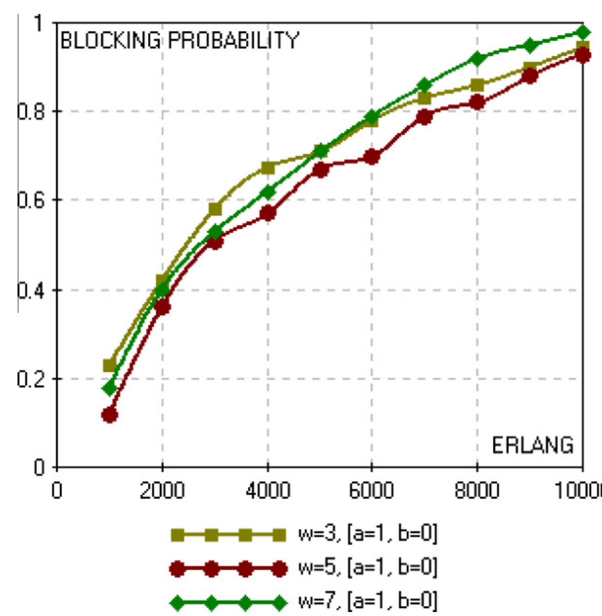

Fig. 8. Internet2 blocking probability, varying window size.

requests and release rates are kept balanced by progressively reducing the connection lifetime. As expected, the greater the time window w is, the higher the delays are. Lower delays have been reported with smaller values of the time window ($w = \{2,3\}$) whilst higher delays have been experienced with greater values of the time window ($w = \{6,7\}$). Results show also that the delays grow faster as w increases and grow at slower rates with low w values. This difference is particularly marked with longer links/lightpaths (higher distances between nodes) as in the case of the Internet2 network (which spreads, in fact, along longer distances). We also observe that, with all the chosen time window values, the delay grows almost linearly with the number of simultaneous players. Also with an high number of simultaneous players/connections, the observed delays always remain under the 1000 ms threshold, which is an affordable time delay for a network [22], thus demonstrating the scalability of the presented approach also in presence of significantly high connection loads.

Now we focus on the behavior of the algorithm when varying the values of the parameters a and b; recall from Eqs. (8) and (10) that the parameter a weights the relative load of links whilst b is a fixed cost for traversing the link. In Figs. 11 and 12 we show the network blocking probability when a is either predominant or negligible with respect to b (medium values of the window size w are assumed). Results for both networks show the same behavior: when choosing values for a greater than b, the cost functions of Eqs. (8) and (10) forces the connections to be spread over the network to avoid paying high costs for traversing loaded links, thus better balancing the load over the available resources, resulting in notable lower blocking probabilities. Anyway, from the Figs. 13 and 14, in which the average experienced time delays are

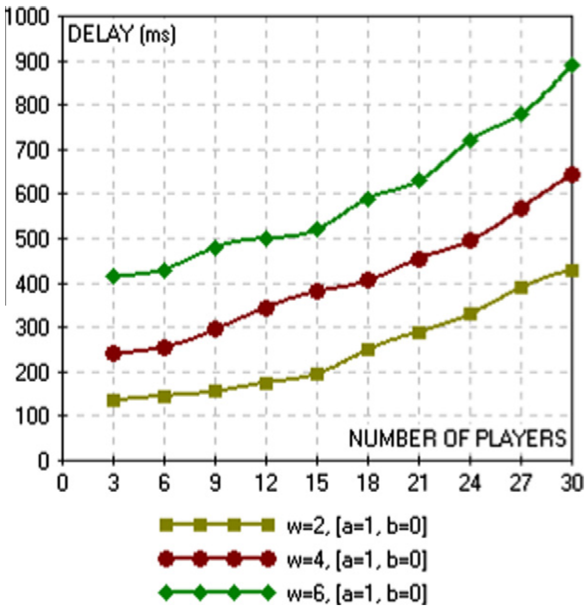

Fig. 9. Geant2 delays, varying window size.

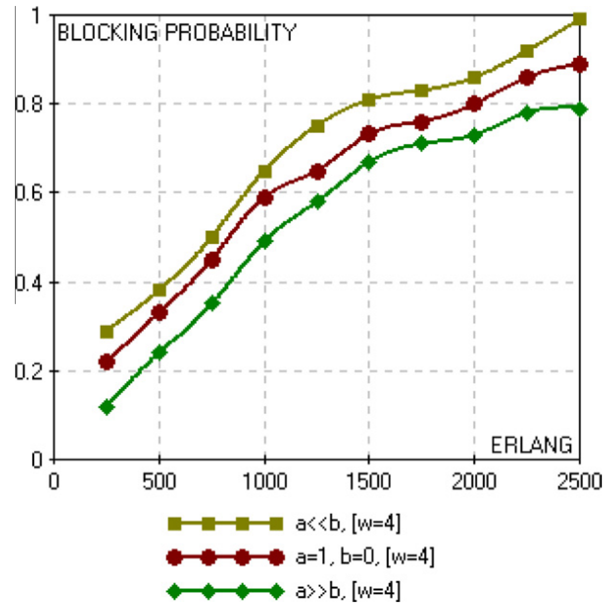

Fig. 11. Geant2 blocking probability, varying parameters *a*, *b*.

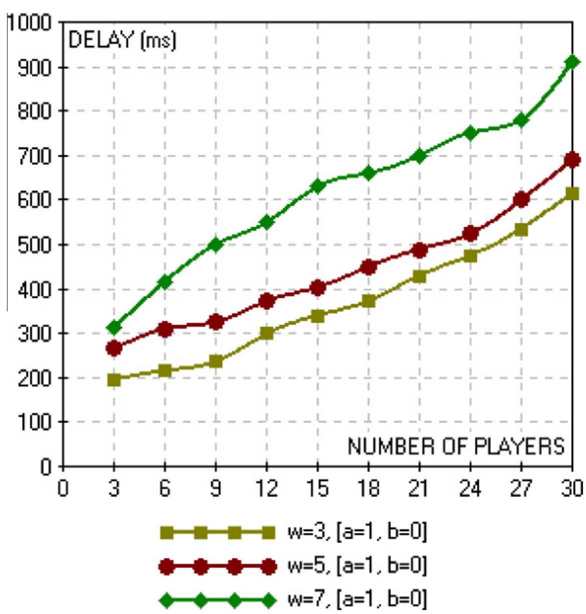

Fig. 10. Internet2 delays, varying window size.

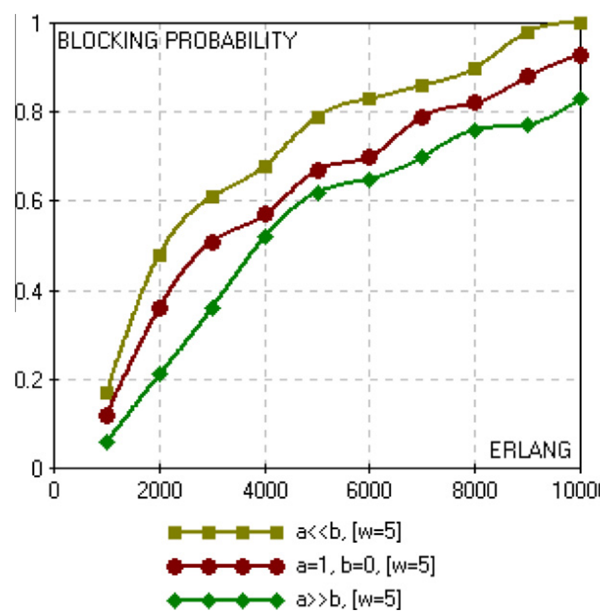

Fig. 12. Internet2 blocking probability, varying parameters *a*, *b*.

shown, we can observe that the better load balancing leads to increased delays, due to the longer paths that will be generally preferred to the shortest ones. Lowest delays have been in fact obtained for values of a lower than b, since they force shorter paths to be cheaper and, thus, to be chosen more frequently. Anyway, shortest paths mean also greater blocking probability, due to the congestion of critical network links. Therefore, a tradeoff exists between load balancing and delay; if the objective is to maximize the number of served connections, a high value of the ratio a/b should be preferred, whereas if the objective is to minimize the average delay, low values of a/b should be chosen.

The performance of the proposed algorithm is compared with three other well-known RWA schemas and the average blocking probabilities are measured and plotted in Figs. 15 and 16. We evaluated our approach against the canonical shortest paths (minimum hop algorithm, MHA [42]), the shortest widest path algorithm (SWP) [43] and the minimum interference routing algorithm (MIRA) [44] transposed into the optical domain [45]. The Dijkstra-based algorithms (MHA and SWP) tend to congest critical links, which results in higher blocking probabilities, more visible in the Internet2 network topology, which is less meshed than Geant2. The proposed algorithm has achieved better performance almost at every load, with MIRA being quite close in terms of rejection ratio. Anyway, even if MIRA performs sometimes better than our algorithm (in some points present at high, medium and low loads), unbalanced network utilization of MIRA and its difficulties on estimating bottlenecks on critical links for cluster nodes make our approach preferable for its more linear behavior achieved in both networks.

Finally, we show the sensitiveness of the algorithm to the order of the connection requests. As we have seen in Section 4, many different ordering criteria are applicable when selecting connection

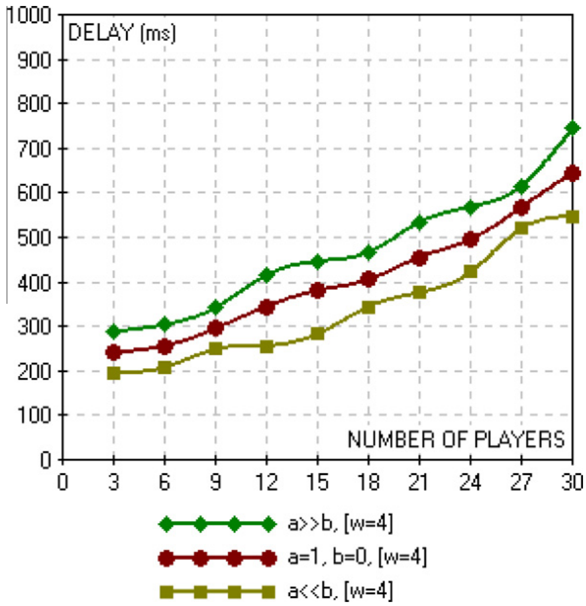

Fig. 13. Geant2 delays, varying parameters *a*, *b*.

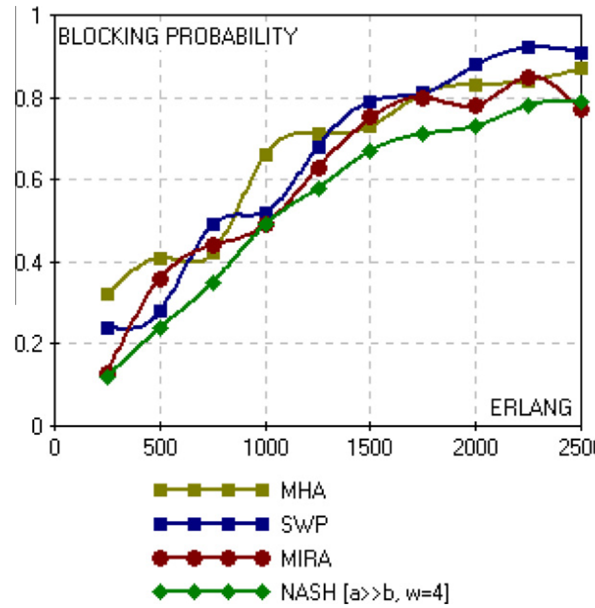

Fig. 15. Geant2 blocking probability comparison with other RWA algorithms.

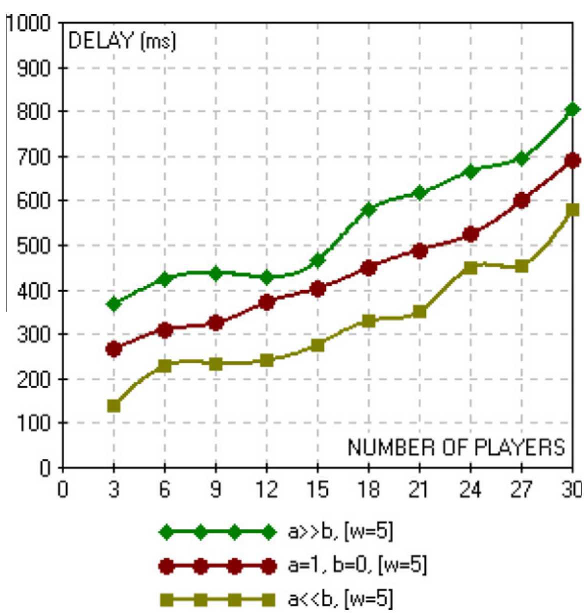

Fig. 14. Internet2 delays, varying parameters *a*, *b*.

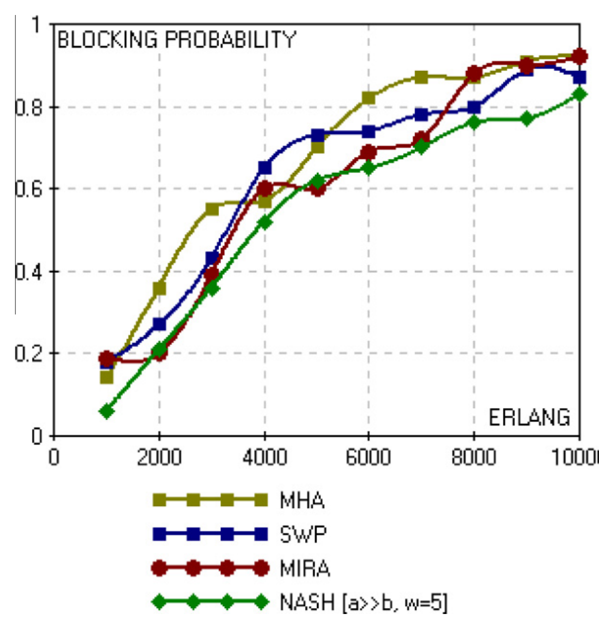

Fig. 16. Internet2 blocking probability comparison with other RWA algorithms.

requests to be served, with the best one depending essentially on the operating scenario and optimization objectives. In order to perform a fair comparison whatever the chosen prioritization is, and to keep the generality of the results, we differentiate between high priority and best effort connection requests. The general framework in which prioritized and best effort connection requests operate is the following. Players (connections requests) choose their strategies by selfishly competing during the reservation phase, so that the Nash equilibrium is preserved, and then, only during the allocation phase, high priority connections are allowed to allocate their resources first. Main results for a medium loaded Geant2 network ($w = 4$, $a >> b$) are shown in Fig. 17 as cumulative distribution function (CDF) of the time in which a given set of connection requests are accepted at or below a given time slot t' in the time window. More than 60% of the prioritized connections are accepted during the first time slot, i.e. their allocation requests have been satisfied and the corresponding resources have been assigned to them, with the acceptance rating growing up to about 95% within the end of the time windows. Connection requests that have not been satisfied at the current time slot move farther to the next time slot, up to the end of the scheduling window. Best effort connections are in general much more delayed toward the end of the time window, with a greater probability of being blocked. Such an high acceptance ratio of the prioritized connections indicate that privileging the connections during the allocation phase is able to differentiate high priority traffic from the best effort one, while keeping intact the properties of the Nashification process.

In conclusions, the results have shown that it is possible to achieve good performances and affordable delays with low/medium values of the time window w and by tuning the values for the parameters a and b in function of the desired optimization criteria (load balancing *vs* delay). The proposed algorithm has often

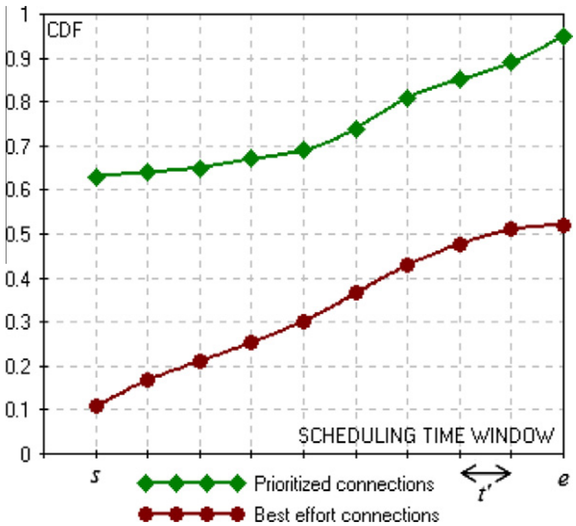

Fig. 17. CDF of the connection requests in the scheduling window.

reached lower blocking ratios with respect to the existing RWA schema with which it has been compared, even in presence of a discrete growing number of simultaneous players and in real network topologies where the geographical distances may be quite long. Finally, we showed that the ordering in which the connections are served in the allocation phase is decisive for privileging high priority requests with respect to best effort traffic, while preserving the presented Nash equilibrium-driven strategy and thus its benefits, especially in presence of highly competitive scenarios with minimum collaboration such as in interconnections of multiple independent autonomous systems networks.

7. Conclusions

In large-scale communication networks, like the Internet, it is usually unfeasible to *globally* manage network traffic. Accordingly, when modeling the traffic behavior in absence of global control, it is typically assumed that network users follow the most rational approach, that is, they behave selfishly to optimize their own individual welfare. Such a consideration motivates our RWA approach based on models from the Game Theory, in which each player is aware of the situation facing all other players and tries to minimize his own cost. We re-formulated the RWA problem in modern connection-oriented all-optical network architectures by considering solution strategies from distributed multi-commodity network congestion games, which are solved by multiple agents operating in a non-cooperative but coordinated manner. The simulation results show that our approach may be particularly attractive for its scalability features and hence useful in large optical networks where many nodes, belonging to different administrative domains, operate selfishly by exchanging only a small amount of information needed for the coordination among them.

Acknowledgments

This work was supported in part by the COST Action IC0804 on Energy Efficiency in Large Scale Distributed Systems, the Spanish Ministry of Science and Innovation under the DOMINO project (TEC2010-18522), the Catalan Government under the contract SGR 1140 and the DIUE/ESF under the Grant FI-201000740.

References

[1] R. Ramaswami, K.N. Sivarajan, Routing and wavelength assignment in all-optical networks, IEEE/ACM Transactions on Networking 3 (5) (1995) 489–500.
[2] I. Chlamtac, A. Ganz, G. Karmi, An approach to high-bandwidth optical wans, IEEE Transactions on Communications 40 (7) (1992) 1171–1182.
[3] D. Fotakis, S. Kontogiannis, P. Spirakis, Selfish unsplittable flows, in: Theoretical Computer Science (TCS) 348, vols. 2–3, pp. 129–366.; D. Fotakis, S. Kontogiannis, P. Spirakis, Preliminary version in the 31st International Colloquium on Automata, Languages and Programming (ICALP'04), 2004, pp. 593–605.
[4] B. Meliàn, J.L. Verdegay, Fuzzy optimization models for the design of WDM, Networks IEEE Transactions on Fuzzy Systems 16 (2) (2008) 466–478.
[5] T. Matsumoto, T. Takenaka, Overlap degree aware routing in all-optical routing networks, IEICE Transactions on Communications 91 (1) (2008) 212–220.
[6] F. Palmieri, U. Fiore, S. Ricciardi, A minimum cut interference-based integrated RWA algorithm for multiconstrained optical transport networks, Journal of Network and Systems Management 16 (4) (2008) 421–448.
[7] J. Zheng, B. Zhang, H.T. Mouftah, Toward automated provisioning of advance reservation service in next-generation optical Internet, IEEE Communications Magazine 44 (12) (2006) 68–74.
[8] T.D. Wallace, A. Shami, Advanced lightpath reservation in WDM networks, in: Proceedings of IEEE INFOCOM, 2006.
[9] S.S.W. Lee, A. Chen, M.C. Yuang, A Lagrangean relaxation based near-optimal algorithm for advance lightpath reservation in WDM networks, Photonic Network Communications (2009) 1–7, doi:10.1007/s11107-009-0215-9.
[10] E. He, X. Wang, V. Vishwanath, J. Leigh, AR-PIN/PDC: flexible advance reservation of intradomain and interdomain lighpaths, in: Proceedings of Globecom 2006, San Francisco, CA, 2006.
[11] M. Mavronicolas, P. Spirakis, The price of selfish routing, in: Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing (STOC), 2001, pp. 510–519
[12] T. Roughgarden, E. Tardos, How bad is selfish routing?, Journal of ACM 49 (2) (2002) 236–259
[13] M. Hoefer, A. Souza, Tradeoffs and average-case equilibria in selfish routing, ACM Trans. Comput. Theory 2, 1, Article 2 (November 2010), 25 pages. doi:10.1145/1867719.1867721, 2010.
[14] C. Busch, M. Magdon-Ismail, Atomic routing games on maximum congestion, in: S.-W. Cheng, C.K. Poon (Eds.), AAIM 2006, LNCS, vol. 4041, Springer, Heidelberg, 2006, pp. 79–91.
[15] G.F. Georgakopoulos, D.J. Kavvadias, L.G. Sioutis, Nash equilibria in all-optical networks, Discrete Mathematics 309 (13) (2009) 4332–4342.
[16] V. Bilò, M. Flammini, L. Moscardelli, On Nash equilibria in non-cooperative all optical networks, in: V. Diekert, B. Durand (Eds.), STACS 2005, LNCS, vol. 3404, Springer, Heidelberg, 2005, pp. 448–459.
[17] X. Chen, X. Xiaodong Hu, W. Weidong Ma, Reducing the maximum latency of selfish ring routing via pairwise cooperations, Lecture Notes in Computer Science, vol. 6509, Springer, 2010. pp. 31–45.
[18] M. Hoefer, V.S. Mirrokni, H. Roglin, S.-H. Teng, Competitive routing over time, Theoretical Computer Science, in press, corrected proof, Available online 13 June 2011, doi:10.1016/j.tcs.2011.05.055.
[19] A. Fanelli, M. Flammini, G. Melideo, L. Moscardelli, A. Navarra, Game theoretical issues in optical networks, in: Proceedings of the 8th International Conference on Transparent Optical Networks (ICTON), IEE, Nottingham, United Kingdom, 2006.
[20] I. Milis, A. Pagourtzis, K. Potika, Selfish routing and path coloring in all-optical networks, Lecture Notes in Computer Science, vol. 4852, Springer, 2007. pp. 71–84.
[21] E. Bampas, A. Pagourtzis, G. Pierrakos, K. Potika, On a non-cooperative model for wavelength assignment in multifiber optical networks, in: Proceedings of the 19th International Symposium on Algorithms and Computation, Springer, 2008, pp. 159–170.
[22] G. Mohan, C. Siva Ram Murthy, A time optimal wavelength rerouting algorithm for dynamic traffic in WDM networks, Journal of Lightwave Technology 17 (3) (1999).
[23] R.W. Rosenthal, A class of games possessing pure-strategy Nash equilibria, International Journal of Game Theory 2 (1973) 65–67.
[24] J.F. Nash, Equilibrium points in n-person games, Proceedings of the National Academy of Sciences of the United States of America 36 (1) (1950) 48–49.
[25] E. Koutsoupias, C.H. Papadimitriou, Worst-case equilibria, in: C. Meinel, S. Tison (Eds.), STACS 1999, LNCS, vol. 1563, Springer, Heidelberg, 1999, pp. 404–413.
[26] D. Monderer, L. Shapley, Potential games, Games and Economic Behavior 14 (1996) 124–143.
[27] G. Karakostas, S.G. Kolliopoulos, The efficiency of optimal taxes, in: Combinatorial and Algorithmic Aspects of Networking, First Workshop on Combinatorial and Algorithmic Aspects of Networking, CAAN, 2004
[28] T. Roughgarden, E. Tardos, How bad is selfish routing?, Journal of the ACM 49 (2) (2002) 236–259
[29] J. Comellas, R. Martinez, J. Prat, V. Sales, G. Junyent, Integrated IP/WDM routing in GMPLS-based optical networks, IEEE Network 2003 (2003) 22–27.
[30] J. Kuri, N. Puech, M. Gagnaire, E. Dotaro, R. Douville, Routing and wavelength assignment of scheduled lightpath demands, IEEE JSAC 21 (8) (2003) 1231–1240.
[31] A. Jaekel, Lightpath scheduling and allocation under a flexible scheduled traffic model, in: Proceedings of IEEE GLOBECOM 2006, 2006.
[32] R. Cole, Y. Dodis, T. Roughgarden, Pricing network edges for heterogeneous selfish users, in: Proceedings of the 35th Annual ACM Symposium on Theory of Computing, 2003, pp. 521–530
[33] S. Dafermos, F.T. Sparrow, The traffic assignment problem for a general network, Journal of Research of the National Bureau of Standards, Series B 73B (1969) 91–118.

[34] Cole, Y. Dodis, T. Roughgarden, How much can taxes help selfish routing?, in: Proceedings of the 4th ACM Conference on Electronic Commerce, 2003, pp. 98–107.
[35] G. Christodoulou, E. Koutsoupias, A. Nanavati, Coordination mechanisms, in: Proceedings of 31st ICALP, 2004.
[36] H.R. Lewis, L. Denenberg, Data Structures and Their Algorithms, HarperCollins, New York, 1991.
[37] F. Palmieri, U. Fiore, S. Ricciardi, SimulNet: a wavelength-routed optical network simulation framework, in: Proceedings of ISCC 2009, 2009, pp. 281–286.
[38] R. Ramaswami, K.N. Sivarajan, Design of logical topologies for wavelength-routed optical networks, IEEE Journal on Selected Areas in Communications 14 (1996) 840–851.
[39] S. Uhlig, B. Quoitin, I. Lepropre, S. Balon, Providing public intradomain traffic matrices to the research community, ACM SIGCOMM Computer Communication Review 36 (1) (2006) 83–86.
[40] Geant2, <http://www.geant2.net/server/show/nav.00d007009> (accessed 12.09).
[41] Internet2, <http://www.internet2.edu/network/> (accessed 12.09).
[42] D. Awduche, L. Berger, D. Gain, T. Li, V. Srinivasan, G. Swallow, Extensions to RSVP for LSP tunnels, Internet Draft <draft-ietf-mplsrsvp-lsp-tunnel-04.txt>, 1999.
[43] R. Guerin, D. Williams, A. Orda, QoS routing mechanisms and OSPF extensions, in: Proc. IEEE GLOBECOM, 1997.
[44] K. Kar, M. Kodialam, T.V. Lakshman, Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications, IEEE Journal on Selected Areas in Communications 18 (12) (2000) 921–940.
[45] K. Kar, M. Kodialam, T.V. Lakshman, Integrated dynamic IP and wavelength routing in IP over WDM networks, in: Proc. IEEE INFOCOM, 2001.