Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005



# A Neural Network Solution to QoS-IP Team-Optimal Dynamic Routing

M. Baglietto, G. Battistelli, R. Bolla, R. Bruschi, F. Davoli, and R. Zoppoli

*Abstract*— **Dynamic-routing in a packet-switched telecommunication network with Quality of Service (QoS) capabilities is addressed. The problem is posed in an informationally decentralized (team) setting, where routing and scheduling decisions are combined. Such decisions are taken at the network nodes, on the basis of local information and possibly of some data received from the neighboring nodes, with the common goal to minimize the expected total delay, spent by packets in traversing the network. Stationarity of the control strategies over an infinite optimization horizon (in the presence of no changes in the traffic parameters and network topology) is achieved by considering an approximation based on a receding-horizon approach. Optimal strategies in this setting are in turn approximated by means of feed-forward neural networks. The problem is posed and a computationally decentralized algorithm for its numerical solution is described. A specific numerical example is also considered, where the neural approximators are tuned, and then used in constructing dynamically varying routing tables and scheduling coefficients in a network simulation based on ns-2, where the gain of the dynamic strategies is evaluated, over an adaptive routing approach based on the measurement of aggregate traffic parameters.**

## I. Introduction

In common practice, the term “dynamic routing“ is often used as a synonym of “adaptive routing“ [1], [2], especially as regards packet networks. In more precise terms, however, adaptivity refers to the capability of adjusting the routing tables when changes in traffic parameters or network topology occur, and most current routing strategies are adaptive in this way; nevertheless, they are based on the knowledge of aggregate variables (e.g., average offered load, average flows) that are slowly varying and whose dynamics is most often not explicitly modelled. On the other hand, truly dynamic routing strategies should consist of closed loop control functions, mapping real-time information that regards “instantaneous“ values of random variables, whose time evolution determines the system’s behavior (e.g., the number of connections traversing a given link or the buffer contents in terms of packets queued for a given link). In this sense, dynamic routing strategies have been widely investigated and are indeed applied in networks where routing decisions have to be taken at the “call“ or “flow“ level. Such is the case for circuit-switched telephone networks (see, e.g., [3], [4], [5], [6], [7], [8], [9], [10] and [11]), ATM networks [3], [12], [13], [14], [15] and [16], optical networks (e.g., [17], [18] and [19]), and Quality of Service (QoS) IP networks, possibly with MPLS [20]. In essence, the relatively slow (with respect to the packet level) time scales, the discrete nature of the state space, the applicability of Markovian queueing models and possible reductions in the search space (e.g., alternate routing on double-hop paths if the direct route is congested), characterize this specific context, and allow to pose tractable dynamic optimal control problems. In another respect, [21], [22], [23], [24], [25] and [26] consider optimization problems in routing based on flow-level information, possibly with QoS constraints, even in the framework of game theory, but without a dynamic control formulation. There are indeed relatively few formulations of the dynamic routing problem in this setting, where optimal dynamic routing strategies are sought, based on the instantaneous state of the network queues (i.e., using queue dynamic equations, rather than stationary distributions), in the presence of centralized or even decentralized information, [27], [28], [29], [30], [31], [32], [33] and [34]. Computationally distributed routing algorithms were considered by Sarachik and Özgüner [34] (their algorithm, however, is valid only for a single destination in the network) and by Iftar and Davison [31], who presented a routing controller that guarantees the clearing of the queues at the nodes in the absence of external inputs, and keeps the lengths of the queues limited as the external message arrival rates are bounded by certain quantities. Owing to the formidable difficulties that arise in this kind of problems, we adopt the methodology introduced in [35], for a generic traffic network to be cleared over a finite horizon, and further specialized to the routing case in telecommunication networks in [36], where non-linear approximating functions are used in the control laws, with the advantage of reducing the problem to a parametric optimization one, without loosing the wealth of information implied by a functional approach. The powerful approximating capability of such mechanisms allows us also to pose the problem in an informationally decentralized setting, in which the nodes act as “the cooperating decision makers of a team“. This means that each node makes its decisions on the basis of a “personal“ information set (i.e., the lengths of the local queues and, possibly, some data coming from other nodes, typically the neighboring ones), and that it aims at minimizing a cost function that is common to all the decision makers of the team (see [37] for the fundamentals of a team organization).

The communication network is modeled as a graph, in which a set of nodes are connected through a set of links. The links cannot be overloaded by traffic beyond their finite capacities. The routing problem consists of directing packets from the nodes, where they originate, to their destinations, in such a way as to minimize a given cost function. Whenever the flows of packets entering the communication network

The authors are with the Department of Communications, Computer and System Sciences, DIST–University of Genoa, Via Opera Pia 13, 16145 Genova, Italy, {mbaglietto, bats, lelus, roberto.bruschi, franco, rzop}@dist.unige.it

vary over time, the nodes may be requested to modify the amount of information to be sent to their neighbors in real time. In this case, a dynamic routing problem arises, which is here addressed by using a receding horizon (RH) control scheme: a discrete-time model is dealt with and, at any time instant, a finite horizon (FH) cost is minimized. Only the control actions relative to the current time instant are applied. The solution of a sequence of FH optimal control problems ensures the routing of the traffic flows on the links over an infinite horizon (IH).

A further aspect to be considered is that of QoS. In particular, for each class of packets corresponding to different service requirements, we may want to specify service rates at the nodes (to be used by packet schedulers), and to weigh a specific route differently from others in the cost functional. Both aspects will be introduced in our model.

Various fixed-structure control strategies can be used (i.e., linear combinations of algebraic or polynomial basis functions, nonlinear approximators like feed-forward neural networks, radial basis functions, linear combinations of sinusoidal functions with variable frequencies and phases, etc.). How to choose a nonlinear approximator (which benefits in general from better approximation capabilities than those of traditional linear ones) for solving a given functional optimization problem is a most important but still open issue. We have chosen feed-forward neural networks and optimized their parameters by a stochastic approximation algorithm. Such a choice has been greatly motivated by successful results obtained in solving highly nonlinear optimal control problems [38], [39], [40] and [41].

## II. MODELLING THE COMMUNICATION NETWORK

In [35] a discrete-time model for generic "traffic networks" was proposed, generalizing the continuous-time one proposed by Segall in [30] and used in subsequent works. The specialization of that model for the case of IP communication networks was presented in [36]. The aim of this section, is to modify such a model in order to account for the presence of different classes of services, thus accounting for QoS capabilities of the network.

The proposed model for a communication network consists in a connected directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, consisting of a set $\mathcal{N}$ of $N$ nodes and a set $\mathcal{L}$ of oriented links. At each node $i \in \mathcal{N}$, an input flow may enter the network. Each message has a destination node $d \in \mathcal{N}$.

We shall denote by $\mathcal{S}(i)$ the set of nodes (whose cardinality is $|\mathcal{S}(i)|$) that are downstream neighbors of node $i$, i.e., the set of nodes $j$ for which a directed link $(i,j)$ exists. (Similarly, we shall denote by $\mathcal{P}(i)$ the set of nodes that are upstream neighbors of node $i$.) At each node $i \in \mathcal{N}$, $|\mathcal{S}(i)|$ output links are present. On each output link there are $C$ buffers (one for each class of service) of length

$$b^{c}_{ij}(t) = \sum_{d \in \mathcal{N}} b^{cd}_{ij}(t)\,, \quad (i,j) \in \mathcal{L},\, c \in \mathcal{C},\, t = 0, 1, \ldots \quad (1)$$

in which messages are stored once they are routed to a node $j \in \mathcal{S}(i)$. (We denote by $\mathcal{C} \triangleq \{1, \ldots, C\}$ the set of the classes of service.)

We assume the network to be "connected", i.e., each node of the network to be reachable from each other node. (This simplifies the model presented later on; some minor notational changes allow the number of destination nodes to be smaller than $N$.)

The control variables for our dynamic system turn out to be the routing tables and the scheduling decisions on the nodes. We assume that these quantities are updated (synchronously throughout the network), at discrete periodic instants $0, 1, \ldots$ . Moreover, we shall assume the quantities of packets on the buffer to be large enough to be represented by real numbers instead of integers. This leads us to a *discrete-time fluid model*.

In order to account for the various possible delays (besides the queueing ones) that may occur in a real communication network [31], we denote by $p_{ij}$ the total delay in transmitting a message at node $i$ (i.e., the time between starting and ending the transmission of a message), in propagating it on the link joining node $i$ to node $j$, and in processing the message at node $j$ (i.e., in identifying its destination, inserting it in the queue of messages with destination $d$, and performing the routing computations). We assume that $p_{ij}$ can be rounded off to an integer, that is, to a multiple of the sample period. Then, we have the following network model:

$$\begin{aligned} &b^{cd}_{ij}(t+1) = \quad (2)\\ &b^{cd}_{ij}(t) + \left[ r^{cd}_{i}(t) + \textstyle\sum_{k \in \mathcal{P}(i)} f^{cd}_{ki}(t - p_{ki}) \right] u^{cd}_{ij}(t) - f^{cd}_{ij}(t),\\ &\qquad (i,j) \in \mathcal{L},\, d \in \mathcal{N}^{i}, c \in \mathcal{C},\ t = 0, 1, \ldots \end{aligned}$$

subject to the following constraints:

$$u^{cd}_{ij}(t) \geq 0 \quad (3)$$

$$\sum_{j \in \mathcal{S}(i)} u^{cd}_{ij}(t) = 1 \quad (4)$$

where $(i,j) \in \mathcal{L},\, d \in \mathcal{N}^{i}, c \in \mathcal{C}, \quad t = 0, 1, \ldots$.

The quantities $u^{cd}_{ij}(t)$ represent the routing control variables (routing tables), and are used to determine the portion of the traffic belonging to class $c$, with destination $d$, arriving at node $i$ in the time interval $[t, t+1]$ that is routed from node $i$ to node $j \in \mathcal{S}(i)$. The quantities $r^{cd}_{i}(t)$ represent the stochastic external input flow entering node $i$ in the time interval $[t, t+1]$ belonging to class $c$ with destination $d$.

The traffic flows belonging to each classe of service $c \in \mathcal{C}$ sent in the time interval $[t, t+1]$ on link $(i,j)$ are denoted by $f^{c}_{ij}(t)$. The proportions of the latter quantities belonging to different classes of service are determined by suitable schedulers:

$$f^{c}_{ij}(t) = b^{c}_{ij}(t)\, v^{c}_{ij}(t)\,, \quad (i,j) \in \mathcal{L}, c \in \mathcal{C},\, t = 0, 1, \ldots \quad (5)$$

The vectors $\underline{v}_{ij} \triangleq \mathrm{col}\left[ v^{c}_{ij},\, c \in \mathcal{C} \right]$ represent the scheduling actions, used by the decision makers on the routing nodes to determine the portions of traffic belonging to different classes starting from node $i$ on link $(i,j)$ in the time interval $[t,t+1]$. Their components have to satisfy the following constraint:

$$0 \leq v_{ij}^{c}(t) \leq 1\,, \quad (i,j) \in \mathcal{L},\, c \in \mathcal{C},\, t=0,1,\ldots \tag{6}$$

The overall traffic flow on link $(i,j)$ starting from node $i$ in the time interval $[t,t+1]$ is given by

$$f_{ij}(t) = \sum_{c \in \mathcal{C}} f_{ij}^{c}(t)\,, \quad (i,j) \in \mathcal{L}\,, t=0,1,\ldots \tag{7}$$

subject to the capacity constraints:

$$f_{ij}(t) \leq C_{ij}\,, \quad (i,j) \in \mathcal{L}\,, t=0,1,\ldots. \tag{8}$$

Though each buffer is served by a FIFO policy, in order to obtain a suitable fluid model we shall assume the packets belonging to different classes of services to be homogeneously distributed in each queue, thus obtaining (recall (1) and (7))

$$f_{ij}^{cd}(t) = f_{ij}^{c}(t)\,\frac{b_{ij}^{cd}(t)}{\displaystyle\sum_{d' \in \mathcal{N}^i} b_{ij}^{cd'}(t)} \;=\; v_{ij}^{c}\, b_{ij}^{cd}(t),$$
$$(i,j) \in \mathcal{L},\, d \in \mathcal{N}^i, c \in \mathcal{C},\, t=0,1,\ldots \tag{9}$$

Actually, in the present formulation we have chosen to forward suitable portions of the buffer contents at time $t$ in the interval $[t,t+1]$. If, on the one hand, we are not supposing that the buffers are served at the maximum allowable rate, on the other hand this allows the routers to suitably react to congestion situations, hopefully avoiding packets loss.

The model stated above corresponds to a datagram network with multicommodity flows and bifurcated routing (as happens in minimum average delay routing problems [42]) where packets belonging to the same traffic flow may be spread over multiple paths toward the destination. This possibility, which would be desirable to achieve minimum delay and maximum throughput (at least in an open queueing network, disregarding flow control), is most often avoided, in order to preserve the order of packets in the flow. As a matter of fact, in the presence of best effort TCP traffic, splitting the flow increases the reordering burden at the destination and, in the presence of large differences in delay jitter over multiple paths, might give rise to retransmissions; at any rate, it should be certainly avoided in the case of QoS routing of real-time flows (RTP/UDP). If we want to enforce the requirement that all packets to the same destination and QoS class follow the same route, we can do so in our model by having constraint (4) be satisfied by a single $u_{ij}^{d}$ equal to 1 for each given destination *d*. Formally, this can be obtained through the enforcement of the additional constraint

$$u_{ij}^{cd}(t)\; u_{il}^{cd}(t) = 0, \qquad i \in \mathcal{N};\, j,\, l \in \mathcal{S}(i);\, j \neq l;\, c \in \mathcal{C};$$
$$d \in \mathcal{N}^i;\, t=0,1,\ldots \tag{10}$$

In order to set the routing problem in the framework of an optimal control one, in the lines of [36] we define the objective of the overall system to be the minimization of an Infinite Horizon (IH) weighted traffic cost

$$J_{IH} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{c \in \mathcal{C}} \sum_{(i,j)\in\mathcal{L}} \Bigg[ \sum_{d \in \mathcal{N}^i} \alpha_{ij}^{cd}\, b_{ij}^{cd}(t) + \sum_{d \in \mathcal{N}^i} \sum_{\tau=\max(0,t-p_{ij})}^{t-1} \beta_{ij}^{cd}\, f_{ij}^{cd}(t-\tau) \Bigg] \tag{11}$$

where $\alpha_{ij}^{cd}$ and $\beta_{ij}^{cd}$, $(i,j) \in \mathcal{L}$, $c \in \mathcal{C}$, $d \in \mathcal{N}^i$, are positive weight constants.

The presence of the weight coefficients $\alpha_{ij}^{cd}$ and $\beta_{ij}^{cd}$ allows the cost (11) to take into account a wide variety of practical situations. If such coefficients are set equal to one, the cost functions give the total time spent by the messages at the nodes and on the links of the communication network. By the superscripts $c$ in $\alpha_{i}^{cd}$ and $\beta_{ij}^{cd}$, different priorities can be set for different classes of services; the superscripts $d$ may set different priorities on the messages sent to the various destinations. Finally, the pairs $(i,j)$ in the coefficients $\beta_{ij}^{cd}$ enable one to associate possible costs that have to be paid to convey messages through the link joining node $i$ to node $j$.

## III. Statement of the dynamic combined QoS-routing/scheduling problem and neural approximations

We shall suppose [35] the control quantities (routing and scheduling quantities) at each node to be set in a decentralized way. In order to formalize this, we shall assume to have different Decision Makers (DMs), one on each routing node, making their routing and scheduling decisions on the basis of a *personal information set* $\underline{I}_i(t)$. Such vectors can include the lengths of the local queues and (possibly) some information received from the other DMs, typically the neighboring ones. As an example, in the lines of [35], let us consider an information structure in which each node receives the lengths of the queues of the downstream neighbors with one step of delay. In this case, the personal information sets take on the form

$$\underline{I}_i(t) = \mathrm{col}\big[\, b_{ij}^{cd}(t), d \in \mathcal{N}^i,\, c \in \mathcal{C},\, j \in \mathcal{S}(i); \tag{12}$$
$$b_{jk}^{cd}(t-1), j \in \mathcal{S}(i), k \in \mathcal{S}(j), c \in \mathcal{C}, d \in \mathcal{N}^j \big]\,,$$

for $i \in \mathcal{N}, t = 0,1,\ldots$. The routing and the scheduling functions take on the form, respectively,

$$u_{ij}^{cd}(t) = \gamma_{ijt}^{1cd}\left[\underline{I}_i(t)\right], \tag{13}$$

$$v_{ij}^{c}(t) = \gamma_{ijt}^{2c}\left[\underline{I}_i(t)\right], \tag{14}$$

for $i \in \mathcal{N}$ $j \in \mathcal{S}(i)$, $c \in \mathcal{C}$, $t = 0,1,\ldots$.

For the sake of compactness, let $\underline{r}_\infty \triangleq \mathrm{col}\,[\underline{r}_i^{cd}(t),\, i \in \mathcal{N},\, c \in \mathcal{C},\, d \in \mathcal{N}^i,\, t=0,1,\ldots]$. Moreover, let $\underline{x}(t) \triangleq \mathrm{col}\Big[b_{ij}^{cd}(t),\,(i,j) \in \mathcal{L},\, d \in \mathcal{N}^i,\, c \in \mathcal{C};\, f_{ki}^{cd}(t-\tau),\, i \in \mathcal{N},\, k \in \mathcal{P}(i),\, d \neq i,\, k \neq d,\, c \in \mathcal{C},\, \tau = 1,\ldots,p_{ki}\Big]$. Such quantities play the role of state vectors in our model. An infinite–horizon decentralized optimal control problem can now be defined.

**Problem RS$_{IH}$.** *Find the routing strategies (13) and the scheduling strategies (14) that minimize the expected cost* $\underset{\underline{x}(0),\underline{r}_{\infty}}{\mathrm{E}} (J_{IH})$.

$\square$

To face Problem RS$_{IH}$, following the lines of [36], we adopt a receding horizon technique. To do this, we first have to fix a finite control horizon $T$. Let $\underline{r}_i(t) \stackrel{\triangle}{=} \mathrm{col}\left[r_i^{cd}(t+s),\, s=0,\ldots,T-1,\, c\in\mathcal{C},\, d\in\mathcal{N}^i\right]$, and $\underline{r}(t) \stackrel{\triangle}{=} \mathrm{col}\left[\underline{r}_i(t), i\in\mathcal{N}\right]$. All stochastic variables are characterized by a given probability density function, $p\left[\underline{x}(t), \underline{r}(t)\right]$. Let us define the following finite horizon (FH) cost function

$$
\begin{aligned}
J_{FH}(t) \;=\; & \sum_{(i,j)\in\mathcal{L}} \sum_{c\in\mathcal{C}} \sum_{d\in\mathcal{N}^i} \Big[ \sum_{s=1}^{T} \alpha_{ijs}^{cd}\, b_{ij}^{cd}(t+s) \;+ \\
& \sum_{s=0}^{T-1} \tilde{\beta}_{ijs}^{cd}\, f_{ij}^{cd}(t+s) \Big], \quad t=0,1,\ldots \qquad (15)
\end{aligned}
$$

where

$$
\tilde{\beta}_{ijs}^{cd} \stackrel{\triangle}{=} \sum_{\tau=s+1}^{\min(s+p_{ij},T)} \beta_{ij\tau}^{cd}.
$$

Here, $\alpha_{ijs}^{cd} = \alpha_{ij}^{cd}$ and $\beta_{ijs}^{cd} = \beta_{ij}^{cd}$ for $s=0,\ldots,T-1$; $\alpha_{ijT}^{cd} > \alpha_{ij}^{cd}$ and $\beta_{ijT}^{cd} > \beta_{ij}^{cd}$ give rise to a suitable "final cost". The presence of this term in the formulation of the FH optimal control problem results to be particularly useful as we want to use a RH control scheme.

For every time instant $t=0,1,\ldots$ we can now state the following finite horizon optimization problem

**Problem RS$_{FH}(t)$.** *Find the optimal control strategies*

$$
\begin{aligned}
u_{ij}^{cd*}(t+s) &= \gamma_{ijs}^{1cd*}[\underline{I}_i(t+s),\, t], \qquad (16)\\
v_{ij}^{c*}(t+s) &= \gamma_{ijs}^{2c*}[\underline{I}_i(t+s),\, t], \qquad (17)
\end{aligned}
$$

*(for* $i\in\mathcal{N}$, $j\in\mathcal{S}(i)$, $c\in\mathcal{C}$, $d\in\mathcal{N}^i$, $s=0,\ldots,T-1$*) that minimize the expected cost* $\underset{\underline{x}(t),\underline{r}(t)}{\mathrm{E}}[J_{FH}(t)]$.

$\square$

As it has been stated, our problem is in the framework of the *team theory* [37]. In fact the decision makers DM$_i$ generate their routing and scheduling control variables on the basis of personal information sets $\underline{I}_i(t)$, but they cooperate on the minimization of the same cost. As is well known, team optimal control problems can be solved analytically in very few cases, (e.g., when the problem is LQG and the information structure is *partially nested*). Problem RS$_{FH}$ is neither LQG nor, in general, characterized by partially nested information structure, and there is no hope to solve it analytically.

Thanks to the time invariance of the dynamics, from here on, we shall drop the index $t$ from equation (16) and (17) and look for sequences of strategies $\gamma_{ijs}^{1cd}$, $s=0,\ldots,T-1$ and $\gamma_{ijs}^{2c}$, $s=0,\ldots,T-1$ inside the finite horizon $[0,T]$. We shall consider $t=0$ as a generic time instant to obtain the strategies (16) and (17), and remove the index $t$ from $J_{FH}(t)$ and from "Problem RS$_{FH}(t)$": we shall simply write $J_{FH}$ and "Problem RS$_{FH}$."

In our RH framework, for each DM, the control strategies corresponding to the first stage of FH optimal control problem, will be used as time-invariant startegies , i.e.,

$$
u_{ij}^{cd}(t) = \gamma_{ij0}^{1cd*}[\, I_i(t)\,] \qquad (18)
$$

and

$$
v_{ij}^{c}(t) = \gamma_{ij0}^{2c*}[\,\underline{I}_i(t)\,] \qquad (19)
$$

for $i\in\mathcal{N}$, $j\in\mathcal{S}(i)$, $c\in\mathcal{C}$, $d\in\mathcal{N}^i$, $s=0,\ldots,T-1$.

In order to face Problem RS$_{FH}$, we shall resort to the approximate technique proposed in [35] and assign a given structure to the control strategies. This is done not to obtain a simplified suboptimal solution, but just because we are unable to derive the optimal solution in analytical form (see [43]). Let us aggregate the functions (16) in the following vectorial form

$$
\underline{u}_i(s) = \underline{\gamma}_{is}^{1}\left[\underline{I}_i(s)\right], \quad i\in\mathcal{N},\, s=0,\ldots,T-1
$$

where $\underline{\gamma}_{is}^{1} \stackrel{\triangle}{=} \mathrm{col}\left[\gamma_{ijs}^{1cd},\, j\in\mathcal{S}(i),\, c\in\mathcal{C},\, d\in\mathcal{N}^i\right]$, and similar definitions hold for $\underline{u}_i(t)$ and, in the following, for $\hat{\underline{\gamma}}_i$ and $\hat{\underline{u}}_i(t)$. Let us introduce similar notations for the scheduling decision:

$$
\underline{v}_{ij}(s) = \underline{\gamma}_{ijs}^{2}\left[\underline{I}_i(s)\right], \quad i\in\mathcal{N},\, s=0,\ldots,T-1
$$

where $\underline{\gamma}_{ijs}^{2} \stackrel{\triangle}{=} \mathrm{col}\left[\gamma_{ijs}^{2c},\, c\in\mathcal{C},\right]$. Similar definitions hold for $\underline{v}_{ij}(t)$ and, in the following, for $\hat{\underline{\gamma}}_{ij}$ and $\hat{\underline{u}}_{ij}(t)$.

We search for approximate strategies of the form

$$
\hat{\underline{u}}_i(s) = \hat{\underline{\gamma}}_{is}^{1}\left[\underline{I}_i(s), \underline{w}_i^1(s)\right], \quad i\in\mathcal{N},\, s=0,\ldots,T-1 \qquad (20)
$$

and

$$
\hat{\underline{v}}_{ij}(s) = \hat{\underline{\gamma}}_{ijs}^{2}\left[\underline{I}_i(s), \underline{w}_i^2(s)\right], \qquad (21)
$$
$$
(i,j)\in\mathcal{L},\, s=0,\ldots,T-1
$$

where the mappings $\hat{\underline{\gamma}}_{is}^{1}$ and $\hat{\underline{\gamma}}_{ijs}^{2}$ take on fixed structures, and $\underline{w}_i^1(s)$ and $\underline{w}_i^2(s)$ are finite-dimension vectors of parameters to be determined so as to minimize the expected value of the cost (15).

As approximate routing functions, we shall use multilayer feedforward neural networks with sigmoidal activation functions.

In order the simplify the following presentation, let us suppose that, at each routing node, the routing and scheduling decisions are taken by a single neural control function:

$$
\mathrm{col}\left[\underline{u}_i(s);\, \underline{v}_{ij}(s), j\in\mathcal{S}(i)\right] = \hat{\underline{\gamma}}_{is}[I_i(s),\, \underline{w}_i(s)] \qquad (22)
$$

where $\underline{w}_i(s) \stackrel{\triangle}{=} \mathrm{col}[\underline{w}_i^1(s); \underline{w}_{ij}^2(s), j\in\mathcal{S}(i)]$, $i\in\mathcal{N}$, $s=0,\ldots,T-1$.

Though one could choose to actually implement such an aggregate function, he could consider more convenient to separate the neural control functions (20) and (21) instead of dealing with a single one (22). In the former case different input vectors could be given to the different control functions performing routing and scheduling.

Let us consider the $i$-th neural network at stage $s$. Clearly, the input variables are the components of the personal information vector $\underline{I}_i(s)$. Let us denote by $\bar{u}_{ij}^{d}(s)$, $j \in \mathcal{S}(i)$, $c \in \mathcal{C}$, $d \in \mathcal{N}^i$, the components of the output vector corresponding to the routing control variables $\hat{u}_{ij}^{cd}(s)$. The latter are obtained by suitable normalization blocks (see [35] for details) that enable us to remove the constraints (4).

The scheduling control variables $\hat{v}_{ij}^{cd}(s)$, $j \in \mathcal{S}(i)$, $c \in \mathcal{C}$, $d \in \mathcal{N}^i$. are obtained directly as outputs of the neural approximator. For notational consistency, let also us denote such outputs as $\bar{v}_{ij}^{cd}(s)$.

The use of the sigmoidal functions ensures the fulfilment of the non-negativity constraints (3), and (6).

In order to remove the constraints (10), we add to the cost (15) penalty functions of the form

$$\rho_i^{cd}(s) = \frac{1}{\sum\limits_{j\in\mathcal{S}(i)} [\hat{u}_{ij}^{cd}(s)]^K} - 1 \,, \tag{23}$$

for $i \in \mathcal{N}$, $d \in \mathcal{N}^i$, $s = 0, \ldots, T-1$, $K \in R_0^+$.

Finally, another penalty function is used to ensure the fulfilment of constraint (8):

$$\theta_{ij}(s) \, = \, \max\Big\{ f_{ij}(s) - C_{ij}, 0 \Big\}^K \,, \tag{24}$$

for $(i,j) \in \mathcal{L}$, $s = 0, \ldots, T-1$.

By adding the terms (23) and (24) and by substituting the parameterized routing and scheduling functions in the expression of the cost $J$, we have:

$$\begin{aligned} & J_{FH}\,[\underline{w}, \underline{x}(0), \underline{r}] = \\ & \sum_{(i,j)\in\mathcal{L}} \sum_{c\in\mathcal{C}} \sum_{d\in\mathcal{N}^i} \left[ \sum_{s=1}^{T} \alpha_{ijs}^{cd}\, b_{ij}^{cd}(s) \,+\, \sum_{s=0}^{T-1} \tilde{\beta}_{ijs}^{cd}\, f_{ij}^{cd}(s) \right] \\ & +\; K_L \sum_{s=0}^{T-1} \sum_{i\in\mathcal{N}} \left[ \sum_{c\in\mathcal{C}} \sum_{d\in\mathcal{N}^i} \rho_i^{cd}(s) \,+\, \sum_{j\in\mathcal{S}(i)} \theta_{ij}(s) \right] \end{aligned} \tag{25}$$

where $\underline{r} \triangleq \underline{r}(0)$, $K_L$ is a positive constant, $\underline{w} \triangleq \mathrm{col}\,[\underline{w}_i\,(s),\, i \in \mathcal{N}, s = 0, \ldots, T-1]$; $\underline{w}_i\,(s)$ is the vector whose components are given by all the weight and bias coefficients of the neural network of the decision maker $DM_i(t)$ (see the control strategies (22)). Thus the functional optimization Problem $\mathrm{RS}_{FH}$ has been reduced to the unconstrained nonlinear programming

**Problem $\mathrm{RS}'_{FH}$.** *Find the vector* $\underline{w}^*$ *that minimizes the expected cost* $\mathop{\mathrm{E}}\limits_{\underline{x}\,(0),\underline{r}} \{J_{FH}\,[\underline{w}, \underline{x}\,(0), \underline{r}]\}$. □

As to the capabilities of neural networks (followed by the normalization blocks) to approximate the optimal control functions, the reader is referred to [35].

## IV. Off-line optimization of the neural routing strategies

In order to address Problem $\mathrm{RS}'_{FH}$, let consider the gradient algorithm

$$\underline{w}^{k+1} = \underline{w}^k - \eta_k \, \nabla_{\underline{w}} \mathop{\mathrm{E}}\limits_{\underline{x}(0),\underline{r}} J_{FH}\,\left[ \underline{w}^k, \underline{x}\,(0), \underline{r} \right] \,, \qquad k = 0, 1, \ldots \tag{26}$$

where $\eta_k$, $k = 0, 1, \ldots$, are stepsizes a priori known to all the DMs. This would allow to assign each $DM_i$ a personal processor with the task of updating the components of the "local" parameter vector $\underline{w}_i$, thus implementing a computationally distributed optimization procedure.

As proposed in [43], in order to avoid the explicit computation of the expected cost and of its gradient as expressed in (26) we compute the "realization" $\nabla_{\underline{w}} J_{FH} \left[ \underline{w}^k, \underline{x}\,(0)^k, \underline{r}^k \right]$ instead of the gradient $\nabla_{\underline{w}} \mathop{\mathrm{E}}\limits_{\underline{x}\,(0),\underline{r}} J_{FH}\,\left[ \underline{w}^k, \underline{x}\,(0), \underline{r} \right]$. Then, we consider the "stochastic gradient" algorithm

$$\underline{w}^{k+1} = \underline{w}^k - \eta_k \, \nabla_{\underline{w}} J_{FH}\,\left[ \underline{w}^k, \underline{x}\,(0)^k, \underline{r}^k \right] \,, \qquad k = 0, 1, \ldots \tag{27}$$

where the index $k$ now denotes both the steps of the iterative procedure and the discrete-time instants at which the vectors $\underline{x}(0)^k, \underline{r}^k$ are generated randomly on the basis of their probability density function $p[\underline{x}(0), \underline{r}]$ (See, for instance, [44] for a description of this method as well for its convergence properties.) To ensure (hopefully) the convergence, we take $\eta_k = c_1/(c_2 + k)$, $c1, c2 > 0$,

In order to apply algorithm (27), the components of the gradient $\nabla_{\underline{w}} J_{FH} \left[ \underline{w}^k, \underline{x}\,(0)^k, \underline{r}^k \right]$, i.e., of the partial derivatives $\partial\, J_{FH} \left[ \underline{w}^k, \underline{x}\,(0)^k, \underline{r}^k \right] / \partial \underline{w}_i(s)$ must be computed. To avoid complicating the equations excessively, we shall not consider the terms (23) and (24). Moreover, we shall drop the index $k$ and simply write $J_{FH}$ instead of $J_{FH} \left[ \underline{w}^k, \underline{x}(0)^k, \underline{r}^k \right]$.

Let us define the following variables: $\lambda_{ij}^{cd}(s) \triangleq \dfrac{\partial J_{FH}}{\partial b_{ij}^{cd}\,(s)}$, $\quad i \in \mathcal{N}$, $d \in \mathcal{N}^i$, $c \in \mathcal{C}$, $s = 0, \ldots, T-1$.

Moreover, let us denote by $\bar{y}_{iji}^{cd}(t)$ the input to $\hat{\underline{\gamma}}_{is}$ corresponding to $b_{ij}^{cd}(t)$, $j \in \mathcal{S}(i)$. Similarly, if an input to $\hat{\underline{\gamma}}_{is}$ corresponds to a state variable $b_{kl}^{cd}(t - p_{jk})$, $k \in \mathcal{S}(i)$, $l \in \mathcal{S}(k)$, we shall redefine it as $\bar{y}_{kli}^{cd}(t)$. Moreover, $\underline{\bar{y}}_i(s) \triangleq \mathrm{col}[\bar{y}_{iji}^{cd}(s),\ j \in S(i);\ \bar{y}_{kli}^{cd}(s), k \in \mathcal{S}(i),\ l \in \mathcal{S}(k); c \in \mathcal{C},\ d \in \mathcal{N}^i]$.

The partial derivatives $\partial\, J_{FH}/\partial \underline{w}_i(s)$ are obtained by a classical *backpropagation procedure*, which, at stage $s$ and for node $i$, allows the computation of $\partial J_{FH}/\partial \underline{\bar{y}}_i(s)$. The backpropagation is initialized by the following procedure (whose derivation is omitted for the sake of brevity).

Note that a similar equations were obtained in [35] and [36], where different models were considered. The following procedure is specific for the case treated here, as it accounts for the presence of the scheduling functions to deal with different classes of service.

**Backpropagation initialization procedure.**

*We have*

$$\frac{\partial J_{FH}}{\partial \bar{v}^{c}_{ij}(s)} = \frac{\partial J_{FH}}{\partial \hat{v}^{c}_{ij}(s)} = \sum_{d\in\mathcal{N}^i} b^{cd}_{ij}(t)\,\frac{\partial J_{FH}}{\partial f^{cd}_{ij}(s)},$$

$$\frac{\partial J_{FH}}{\partial \bar{u}^{cd}_{ij}(s)} = \sum_{k\in\mathcal{S}(i)} \frac{\partial J_{FH}}{\partial \hat{u}^{cd}_{ik}(s)}\frac{\partial \hat{u}^{cd}_{ik}(s)}{\partial \bar{u}^{cd}_{ij}(s)},$$

*for* $i \in \mathcal{N}$, $j \in \mathcal{S}(i)$, $c \in \mathcal{C}$, $d \in \mathcal{N}^i$, $s = 0,\ldots,T-1$, *where the partial derivatives* $\partial\hat{u}^{cd}_{ik}(s)/\partial\bar{u}^{cd}_{ij}(s)$ *are obtained by differentiating the normalization blocks (see [35]).*

$$\frac{\partial J_{FH}}{\partial \hat{u}^{cd}_{ij}(s)} = \lambda^{cd}_{ij}(s+1)\left\{\Big[\sum_{k\in\mathcal{P}(i)} f^{cd}_{ki}(s-p_{ki}-1)\Big] + r^{cd}_{i}(s)\right\},$$

$$i\in\mathcal{N},\, d\in\mathcal{N}^i,\, j\in\mathcal{S}(i),\, c\in\mathcal{C},\, s=0,\ldots,T-1\,. \quad (28)$$

*The variables* $\lambda^{cd}_{ij}(s)$ *can be computed by means of the following equations* ($i \in \mathcal{N}$, $j \in \mathcal{S}(i)$, $d \in \mathcal{N}^i$, $c \in \mathcal{C}$, $s = 0,\ldots,T-1$)

$$\lambda^{cd}_{ij}(s) = \alpha^{cd}_{ijs} + \lambda^{cd}_{ij}(s+1) + \hat{v}^{c}_{ij}(t)\frac{\partial J_{FH}}{\partial f^{cd}_{ij}(s)} + \\ + \frac{\partial J_{FH}}{\partial \bar{y}^{cd}_{iji}(s)} + step(T-s-2)\sum_{k\in\mathcal{P}(i)}\frac{\partial J_{FH}}{\partial \bar{y}^{cd}_{ijk}(s+1)}$$

*where* $\mathrm{step}\,(a) = 1$, *if* $a \geq 0$ *and* $\mathrm{step}\,(a) = 0$, *if* $a < 0$. *The terms* $\dfrac{\partial J_{FH}}{f^{cd}_{ij}(s)}$ *can be obtained as follows:*

$$\frac{\partial J_{FH}}{\partial f^{cd}_{ij}(s)} = \tilde{\beta}^{cd}_{ijs} - \lambda^{cd}_{ij}(s+1) \;+\; step[T-(s+p_{ij}+1)] \;\cdot \\ \cdot \sum_{l\in\mathcal{S}(i)} \lambda^{cd}_{jl}(s+p_{ij}+1)\,\hat{u}^{cd}_{jl}(s+p_{ij}).$$

*The recursion is initialized by the conditions*

$$\lambda^{cd}_{ij}(T) = \alpha^{cd}_{ijT}\,,\quad i\in\mathcal{N},\, j\in\mathcal{S}(i),\, c\in\mathcal{C},\, d\in\mathcal{N}^i$$

□

The above described stochastic-gradient optimization procedure can be performed completely off line. Once the parameter vectors $w^*_i(s)$, $i \in \mathcal{N}$, $s = 0,\ldots,T-1$, have been computed, (i.e. Problem $\mathrm{RS}'_{FH}$ hes been solved), only $\underline{w}^*_i(0)$, $i\in\mathcal{N}$ will be retained by each $\mathrm{DM}_i$, and used on line to parameterize time-invariant IH routing functions

$$\mathrm{col}\left[\underline{u}_i(t);\, \underline{v}_{ij}(t), j\in\mathcal{S}(i)\right] \;=\; \underline{\hat{\gamma}}_i[I_i(t),\, \underline{w}^*_i(0)]$$

## V. Simulation results

To test the proposed mechanism we have used the ns-2 (Network Simulator [45]) simulation environment, for which we have developed a specific module to implement the proposed neural-routing. The current released simulation does not consider or generate signalling traffic. Moreover, we have used the ns-2 implementation [46] of the OSPF QoS extension proposed in RFC 2676 [47] as a term of comparison. This type of routing is quite different from the neural dynamic one, as it is based on an on-demand Dijkstra algorithm for path computation of QoS traffic flows and it also includes a Flow Admission Control (FAC). A complete description of this alternative mechanism in the form that we have used can be found in [48]. We consider the network topology in Fig.1 and a traffic matrix composed by 21 end-to-end traffic flows between the source-destination source-destination (S-D) pairs: $(1-5), (1-6), (2-5), (2-6), (4-5), (4-6), (6-5)$. Between each pair there are three flows, one for each traffic class. All the network links are characterized by a capacity of 1 Mbps and a propagation delay equal to 30 ms. The rate of neural-routing decision events is 1 event per second, so the routing table can change no faster than once time every second. It must be noted that the original version of the OSPF extension proposal considers only two traffic classes: best effort (class 1) and QoS one (class 3). In this respect we have introduced a third intermediate class by giving to the class 3 (QoS) a priority in the path computation with respect to the intermediate class (class 2). We have used two types of traffic sources, which generate UDP packets. The reason of this choice (UDP instead of TCP) is to avoid the TCP flow control effects, which can complicate the interpretation of the results and does not permit stressing the load of the network for what concerns the internal nodes (in congestion situations TCP traffic would slow down at the sources). Moreover, the usage of UDP is anyway realistic for what concerns the real-time traffic (class 3, and in some cases class 2). The packet size is constant and equal to 550 Bytes. In the first set of tests we have considered CBR (Continuous Bit Rate) packet generation, which gives us the ability to easily evaluate the delay performance (end-to-end delay and jitter). In the second set of tests the UDP packets are generated by using an exponential interarrival time distribution. The

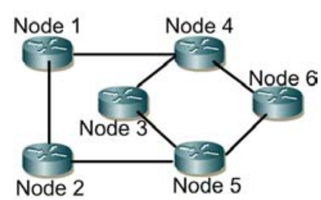


Fig. 1. The reference network topology for the simulation results.

first results come from a validation test of the proposed mechanism, which we have realized to confirm the correct operation of the algorithm and its ns-2 implementation. To this purpose, we have imposed a fixed deterministic rate value of 60 Kbps for all traffic flows except the ones between the pair (2-5); the latter flows have 200Kbps rates for classes 2 and 3, while class 1's rate changes dynamically: it starts at 60 Kbps and, after 5 seconds, it grows to 500Kbps. As one can observe in Fig. 2, which reports the end-to-end delays for each class versus the simulation time, the traffic increase at t=5 along link (2,5) saturates the link; this raises the delay of class 1 packets. At time t=7 s, the routing agent decides to change the path, and the delay lowers to about 0.15 s. It can be noted that the traffic of class 2 and 3 has not been affected by this link saturation and their performance remains unchanged.

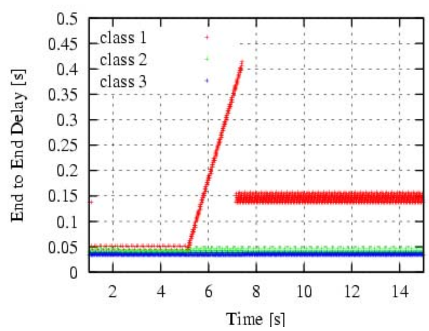


Fig. 2. End-to-end delays of flows of SD pairs (2-5) for each class.

The second test session shows the behaviour in terms of end-to-end average delay, jitter and dropping probability of both the proposed routing mechanism and the one used for comparison in the presence of a deterministic CBR traffic pattern. In particular, the rates utilized for the flows between the pair (6-5) are fixed at 200 Kbps, while the rates of all the other traffic flows are variable from 60 Kbps to 230 Kbps. In Fig. 3 the average end-to-end delay values versus the offered flow rate values obtained with the two mechanisms are shown. It can be note that the proposed mechanism protect effectively the highest priority traffic (class 3) and partially the class 2 traffic and, at the same moment, it tries to avoid and excessive penalization of class 1. The QoS OSPF cannot effectively give an intermediate treatment to class 2: the latter tends to have a behaviour rather similar to that reserved to class 3; moreover the delays of class 1 are always larger than those obtained with the neural-routing.

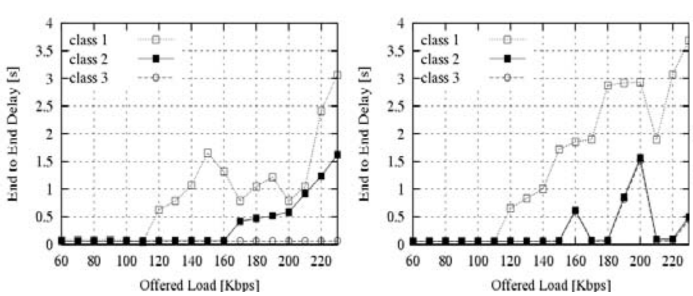


Fig. 3. End to end average delays for all the three traffic classes versus growing flow rates; CBR sources. (Neural-QoS-routing, QoS-OSPF routing.)

In Fig. 4, the average jitter values are reported. For what concerns the jitter performance, again the neural-QoS routing is able to more effectively protect class 3, which has a very low jitter in all the tested cases (lower that those obtained with the QoS OSPF). For what concerns the other classes, class 1 jitter values are always larger with the proposed routing scheme because the mechanism in some sense trades jitter for delay; moreover QoS-OSPF cannot differentiate class 2 from class 3, while the proposed mechanism is able to give intermediate performance to class 2.

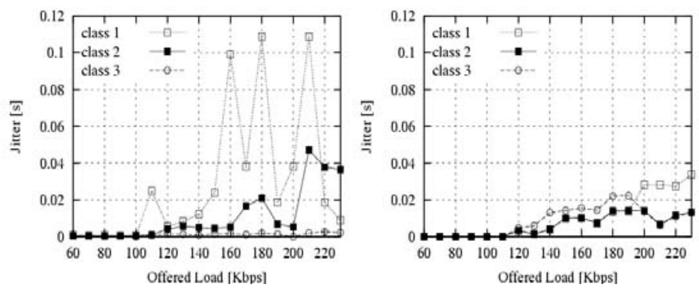


Fig. 4. Average jitter values versus growing flow rates; CBR sources. (Neural-QoS-routing, QoS-OSPF routing.)

The second set of tests has been obtained in the same conditions as the previous group, but by using stochastic packet generation times. More in particular, packet interarrival times are generated by using an exponential distribution. Figs. 5 and 6 are the analogous of Figs. 3 and 4 in the previous case. Similar considerations as before can be done.

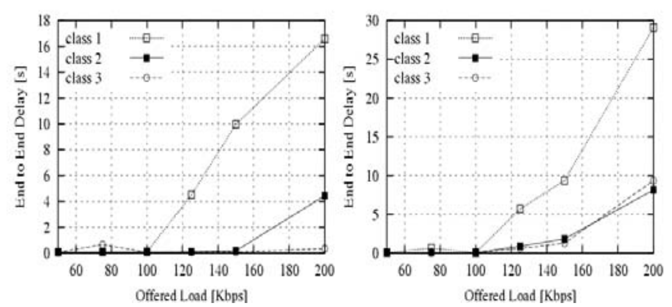


Fig. 5. End to end average delays versus growing flow rates; stochastic sources. (Neural-QoS-routing, QoS-OSPF routing.)

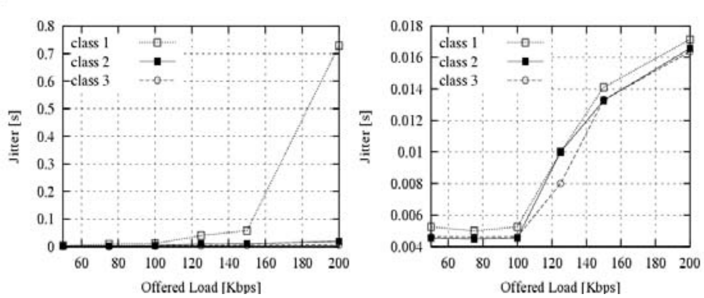


Fig. 6. Average jitter values versus growing flow rates; stochastic sources. (Neural-QoS-routing, QoS-OSPF routing.)

## REFERENCES


[1] http://www.linktionary.com/d/dynamic_routing.html

[2] http://business.cisco.com/glossary/tree.taf-asset_id=92873&word=89309&public_view=true&kbns=2&DefMode=.htm

[3] K. W. Ross, *Multiservice Loss Models for Broadband Telecommunication Networks*, Springer Verlag, Berlin, 1995.

[4] G. R. Ash, R. H. Cardwell and R. P. Murray, "Design and optimization of networks with dynamic routing," *Bell Syst. Tech. J.* , vol. 60, no. 8, pp. 1787-1820, 1981.

[5] G. R. Ash, A. H. Kafker and K. Krishnan, "Servicing and real-time control of networks with dynamic routing," *Bell Syst. Tech. J.*, vol. 60, no. 8, pp. 1821-1845, 1981.

[6] D. Mitra, R. J. Gibbens and B. D. Huang, "State-dependent routing on symmetric loss networks with trunk reservations, I," *IEEE Trans. Commun.*, vol. 41, p. 10 pages, Feb. 1993.

[7] J. Filipiak, *Modelling and Control of Dynamic Flows in Communication Networks*, Springer-Verlag, Berlin/Heidelberg, 1988.

[8] P. Chemouil, J. Filipiak and P. Gauthier, "Performance issues in the design of dynamically controlled circuit-switched networks," *IEEE Commun. Mag.*, vol. 28, no. 10, pp. 90-95, 1990.

[9] A. Girard, *Routing and Dimensioning in Circuit-Switched Networks*, Addison-Wesley, Boston, 1990.

[10] F. P. Kelly, "Routing and capacity allocation in networks with trunk reservation," *Math. Op. Res.*, vol. 15, no. 4, pp. 771-793, 1990.

[11] R. J. Gibbens, F. P. Kelly and S. J. Turner, "Dynamic routing in multiparented networks," *IEEE/ACM Trans. Networking*, vol. 1, no. 2, pp. 261-270, April 1993.

[12] G. Kesidis, J. Walrand and C. J. Chang, "Effective bandwidths for multiclass Markov fluids and other ATM sources," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 424-428, Aug. 1993.

[13] C. A. Courcoubetis, G. Kesidis, A. Ridder, J. Walrand and R. Weber, "Admission control and routing in ATM networks using inferences from measured buffer occupancy," *IEEE Trans. Commun.*, vol. COM-43, no. 2, pp. 1778-1784, 1995.

[14] N. Anerousis and A. A. Lazar, "Virtual Path control of ATM networks with call level quality of service guarantees," *IEEE/ACM Trans. Networking*, vol. 6, no. 2, pp. 222-236, April 1998.

[15] R. Bolla, A. Dalal'ah, F. Davoli, M. Marchese and M. S. Obaidat, "Adaptive distributed admission control and routing in ATM networks," *Internat. J. Commun. Syst.*, vol. 10, pp. 215-225, 1997.

[16] R. Bolla, F. Davoli, M. Marchese and M. Perrando, "QoS-aware routing in ATM and IP-over-ATM," *Computer Commun.*, Special Issue on Performance Evaluation of Telecommunication Systems: Models, Issues and Applications, vol. 24, no. 9, pp. 811-821, May 2001.

[17] M. Dueser, A. Zapata and P. Bayvel, "Investigation of the scalability of dynamic wavelength-routed optical networks," *J. of Optical Networking* 3, 667-686, 2004.

[18] M. Kodialam and T.V. Lakshman, ' 'Dynamic routing of restorable bandwidth-guaranteed tunnels using aggregated network resource usage information," *IEEE/ACM Trans. Networking*, Vol. 11 , No. 3, pp. 399-410, June 2003.

[19] A.E. Ozdaglar and D.P. Bertsekas, "Routing and wavelength assignment in optical networks," *IEEE/ACM Trans. Networking*, Vol. 11 , No. 2 , April 2003, pp. 259-272.

[20] J.L. Marzo, E. Calle, C. Scoglio, T. Anjah, "QoS online routing and MPLS multilevel protection: a survey," *IEEE Commun. Mag.*, Vol. 41 , No. 10 , Oct. 2003, pp. 126-132.

[21] Y.A. Korilis, A. A. Lazar and A. Orda, "Achieving network optima using Stackelberg routing strategies," *IEEE/ACM Trans. Networking*, vol. 5, no. 1, pp. 161-173, Feb. 1997.

[22] Y.A. Korilis, A. A. Lazar and A. Orda, "Capacity allocation under noncooperative routing," *IEEE Trans. Automat. Contr.*, vol. 42, no. 3, pp. 309-325, March 1997.

[23] Y.A. Korilis, A. A. Lazar and A. Orda, "Architecting noncooperative networks," *IEEE J. Select. Areas Commun.*, vol. 13, no. 7, pp. 1241 1251, Sept. 1995.

[24] R.A. Guérin and A. Orda, "QoS routing in networks with inaccurate information: theory and algorithms," *IEEE/ACM Trans. Networking*, vol. 7, no. 3, pp. 350-364, June 1999.

[25] A. Orda, "Routing with end-to-end QoS guarantees in broadband networks," *IEEE/ACM Trans. Networking*, vol. 7, no. 3, pp. 365-374, June 1999.

[26] G. Fodor, P. Barta, Á. Marquetant, A. Rácz and T. Henk, "Modelling and analysis of routing and resource allocation techniques in multi-service networks," *Internat. J. Commun. Syst.*, vol. 12, no. 2, pp. 103-123, March 1999.

[27] A. Ephremides, P. Varaiya and J. Walrand, "A simple dynamic routing problem," *IEEE Trans. Automat. Contr.*, vol. AC-25, no. 4, pp. 690-693, 1980.

[28] L. Tassiulas and A. Ephremides, "Throughput properties of a queueing network with distributed dynamic routing and flow control," *Adv. Appl. Prob.*, vol. 28, pp. 285-307, 1996.

[29] A. Orda ,R. Rom and M. Sidi , "Minimum delay routing in stochastic networks," *IEEE/ACM Trans. Networking*, vol. 1, no. 2, pp. 187-198, April 1993.

[30] A. Segall, "The modeling of adaptive routing in data–communication networks," *IEEE Trans. Commun.*, vol. 25, pp. 85–95, 1977.

[31] A. Iftar and E. J. Davison, "A decentralized discrete–time controller for dynamic routing," *Internat. J. of Control*, vol. 69, pp. 599–632, 1998.

[32] F. H. Moss and A. Segall, "An optimal control approach to dynamic routing in networks," *IEEE Trans. Automat. Contr.*, vol. 17, pp. 329–339, 1982.

[33] G. J. Stassinopoulos and P. Konstantopoulos, "Optimal congestion control in single destination networks," *IEEE Trans. Commun.*, vol. 33, pp. 792–800, 1985.

[34] P. E. Sarachik and Ü. Özgüner, "On decentralized dynamic routing for congested traffic networks," *IEEE Trans. Automat. Contr.*, vol. 27, pp. 1233–1238, 1982.

[35] M. Baglietto, T. Parisini, R. Zoppoli, "Distributed-information neural control: the case of dynamic routing in traffic networks," *IEEE Trans. Neural Networks*, vol. 12, no. 3, pp. 485-502, May 2001

[36] M. Baglietto, R. Bolla, R. Bruschi, F, Davoli, and R. Zoppoli, "Neural Approximation of Team Optimal Dynamic Routing in IP Networks," *Proc. 43th Conference on Decision and Control* , Paradise Island, Bahamas, pp. 5140–5145, 2004.

[37] J. Marschak and R. Radner, *Economic Theory of Teams*. Yale University Press, 1972.

[38] T. Parisini and R. Zoppoli, "A receding–horizon regulator for nonlinear systems and a neural approximation," *Automatica*, vol. 31, n. 10, pp. 1443–1451, 1995.

[39] T. Parisini and R. Zoppoli, "Neural approximation for multistage optimal control of nonlinear stochastic systems," *IEEE Trans. Autom. Contr.*, vol. 41, pp. 889–895, 1996.

[40] T. Parisini, M. Sanguineti and R. Zoppoli, "Nonlinear stabilization by receding–horizon neural regulators," *Internat. J. of Control*, vol. 70, n. 3, pp. 341–362, 1998.

[41] T. Parisini and R. Zoppoli, "Neural approximations for infinite–horizon optimal control of nonlinear stochastic systems," *IEEE Trans. on Neural Networks*, vol. 9, n. 6, pp. 1388-1408, 1998.

[42] D. Bertsekas and R. Gallager, *Data Networks*, $2^{nd}$ Ed., Prentice-Hall, Englewood Cliffs, NJ, 1992.

[43] R. Zoppoli, M. Sanguineti, T. Parisini, "Approximating Networks and Extended Ritz Method for Solution of Functional Optimization Problems," *J. of Optimization Theory and Applic.*, vol. 112, no. 2, pp. 403-439, February 2002.

[44] B. T. Polyak and Ya. Z. Tsypkin, "Pseudogradient adaptation and training algorithms," *Automation and Remote Control*, vol. 12, pp. 377–397, 1973.

[45] "The Network Simulator 2," http://www.isi.edu/nsnam/ns/.

[46] "QoS Routing n NS2," http://keskus.hut.fi/tutkimus/ironet/ns2/qosrns-0.2-pre2-ns2.1b8a.tar.gz.

[47] G. Apostolopoulos, D. Williams, S. Kamat, R. Guerin, A. Orda, and T. Przygienda, "QoS Routing Mechanisms and OSPF Extensions," RFC 2676, August 1999.

[48] A. Singh, G. Mittal and S. K. Nandi, "QoS and Traffic Engineering: MPLS, DiffServ and Constraint Based Routing," May 2000, http://keskus.hut.fi/tutkimus/ironet/ns2/qosr.pdf.