Network Performance of Multi-Service Circuit Switched Networks Simulational comparison of variants of DAR and RTNR

Luísa Jorge (ljorge@inescc.pt)
ESTIG – IPB, Campus de St^a Apolónia, 5301-857 Bragança, Portugal;
INESC-Coimbra, Rua Antero de Quental 199, 3000-033 Coimbra, Portugal

José Craveirinha (jcrav@inescc.pt)
DEEC – FCTUC, Pólo 2 da Univ. Coimbra, 3030-290 Coimbra, Portugal;
INESC-Coimbra, Rua Antero de Quental 199, 3000-033 Coimbra, Portugal

Teresa Gomes (teresa@inescc.pt)
DEEC – FCTUC, Pólo 2 da Univ. Coimbra, 3030-290 Coimbra, Portugal;
INESC-Coimbra, Rua Antero de Quental 199, 3000-033 Coimbra, Portugal

Abstract

The use of dynamic routing methods has an important impact on the performance of multi-service networks namely in overload and failure conditions. The paper describes and compares the performance of variants of reference dynamic routing methods (DAR and RTNR).

Two variants of DAR resulting from the extension to multi-service networks of the original formulation of the method are presented. Also a simplification of the trunk reservation mechanism per traffic class of RTNR is considered.

The major results of extensive experimentation with these variants of the routing methods using a discrete event simulation platform are discussed. The simulation model includes Poisson and Engset type multi-class traffic flows. Conclusions concerning the relative performance of the methods at network level and per traffic class are put forward.

1. Introduction

Dynamic routing was firstly applyed to the core of circuit switched telecommunication networks, to improve network performance namely in overload and failure situations. In multi-service networks dynamic routing is used to increase carried traffic and also to provide differentiated grade of service according to the needs of the supported traffic flows, therefore leading to better performance and reduced network cost.

A simplified version of Real Time Network Routing (RTNR) and an extended version of Dynamic Alternate Routing (DAR) will be analysed and compared through a discrete-event simulator (previously described in [6]).

DAR is a routing method know for its simplicity, which was developed for single-service networks. Two DAR variants are discussed: the first one is a simple extension of the original DAR to support simultaneously several traffic classes, using fixed trunk reservation to protect traffic in the direct path and using no other service protection mechanism (DAR0). The second variant besides supporting several traffic classes, uses dynamic trunk reservation, both for protection of traffic in the direct path and for service protection (DAR1) of each traffic class by using a mechanism similar to RTNR. The considered RTNR variant (RTNR1) results from the simplification of the calculation of the number of reserved virtual channels, for each traffic class.

For comparison purposes, a fully meshed network topology with 6 nodes (see [8]) widely used in dynamic routing studies, was used. The network was dimensioned for two separate set of experiments. In the first set of experiments only Poissonian traffic with single channel and two channel calls was considered. In the second set of experiments besides the previous types of traffic flows, Engset type traffic flows (not previously considered in the literature concerning dynamic routing simulation studies) with 6 and 24 channels per call were considered.

Finally it should be pointed out that although these methods were developed in the context of circuit switched networks, similar methods are likely to be adopted in the context of ATM (Asynchronous Transfer Mode) or IP/MPLS

Proceedings of the The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'04)
1526-7539/04 $20.00 © 2004 **IEEE**

(IP/Multiprotocol Label Switching), using the concept of equivalent bandwidth (see [7]).

This paper is divided in three parts. First, the basic characteristics of the routing methods RTNR and DAR are presented. Second the proposed variants of RTNR and DAR are described. Then the simulation conditions including the description of the considered types of traffic flows, are described. Also a detailed analysis of the relative performance of the implemented variants of the methods is presented. This comparison will focus not only on average blocking probabilities but also on the blocking probability per traffic class. Finally some concluding remarks are made.

2. Review of the Dynamic Routing Methods

In this study the performance of two dynamic routing methods, DAR and RTNR, will be analyzed through simulation.

2.1. RTNR

The description of this method will use, in general, the notation in [2].

When a new call arrives the Originating Switch (named OS_j) tries initially to establish the call in the trunk that connects directly to the Termination Switch (named TS_k). If that is not possible, it tries to find a suitable two-trunk path. If several are available, the call is established in the path with greater availability (at that time), as defined below. Thus, it is the originating switch which makes the routing decisions, based on updated trunk state information.

On call arrival, when OS_j tries to find an available two-trunk path, it needs to receive from TS_k information regarding all trunks connected to it. When OS_j receives this information from TS_k, it compares this with its own information, in order to find the least loaded two-link path to route the call. In this method, routing decisions are based on aggregate link states. Rather than detailed information, a rough approximation of the link states, in terms of "bit maps", is used. This allows the switches to have a simple way of representing and exchanging trunk state information.

One of the main characteristics of RTNR is its capability for handling multi-class traffic, that is traffic classes with different bandwidth and holding time requirements. In the model, routing controls of different traffic classes are independent. To do so, it uses *virtual networks*, which route exclusively traffic belonging to the corresponding traffic class. Bandwidth is reserved to a traffic class only when its performance objectives aren't being met. Another important characteristic of the method is that some parameters, like trunk reservation, can be changed on a call-by-call basis.

Two trunk reservation mechanisms are used simultaneously by RTNR, namely:

i *Protection of traffic in the direct path* – On each particular trunk, some channels may be reserved to be used only on single-trunk paths for each service traffic class. This may help in a congestion situation, although as a side effect, a call may be prevented from being established in a two-trunk path, although there are free channels on both trunks.

ii *Protection per traffic class* – To guarantee traffic class performance objectives, some channels may be conditionally reserved and be used only for traffic belonging to a certain traffic class. This prevents traffic belonging to other classes from using those channels, if the performance objectives of the protected class aren't being met.

In a multi-service network, calls of a particular class-of-service, i, are assumed to consume an average bandwidth equal to r^i, using a single unit of capacity denoted as one virtual trunk (VT).

Some details about the method are needed to fully understand it. Namely, we will focus on:

1. How the trunks load states (needed for the creation of bit maps) are computed;
2. What is done to ensure greater fairness in sharing the network between the different service traffic classes;
3. How the switch selects the two-trunk paths.

2.1.1. Computing trunk load states As previously mentioned, when selecting the two-trunk path of a call, it is essential to know each trunk load state, which is determined in terms of its idle bandwidth. Each trunk may be placed in one of 6 aggregation states: Busy (B), Reserved (R), and 4 load states, ordered by increasing occupation level – very lightly loaded (LL1), lightly loaded (LL2), medium loaded (LL3), and heavily loaded (HL).

The thresholds of the load states are defined per trunk, and are based on 4 quantities for each service class. Each Originating Switch OS_j maintains, for each traffic class i, and for each destination TS_k: the current number of calls-in-progress (CIP_k^i), the current switch-to-switch blocking level (NN_k^i), the estimate of the offered traffic load (TL_k^i) and the estimate of the number of virtual trunks required to meet the blocking objective ($VTtraf_k^i$).

CIP_k^i is defined as the number of active calls from OS_j to TS_k (either on the direct path or in two-trunk paths). The switch-to-switch blocking level, NN_k^i, is given by

$$NN_k^i = OV_k^i / PC_k^i \tag{1}$$

where OV_k^i and PC_k^i are counters of blocked calls and offered switch-to-switch calls, over the last periodic update interval for virtual network i, respectively.

The offered traffic load in period n is estimated by

$$TL_n^i = 0.5 \times TL_{n-1}^i + 0.5 \times \left(CIP_n^i + OV^i\right) \qquad (2)$$

The number of virtual channels needed to meet the blocking objectives for the current offered traffic load, TL_k^i, from OS_j to TS_k, is estimated by,

$$VTtraf_k^i = 1.1 \times TL_k^i \qquad (3)$$

The various reservation levels, R_k^i, computed in each trunk for each service traffic class, are presented in Table 1, according to [2].

NN_k^i - Blocking Level	Reservation Level	$R_k^i\,(VTs)$
$[0,.01]$	0	0
$(.01,.05]$	1	$.05 \times VTtraf_{k^i}$
$(.05,.15]$	2	$.1 \times VTtraf_{k^i}$
$(.15,.5]$	3	$.15 \times VTtraf_{k^i}$
$(.5,1]$	4	$.2 \times VTtraf_{k^i}$

Table 1. Trunk Reservation Levels

The number, $Rtraf_k^i$, of reserved virtual channels, for traffic of service class i, is given by

$$Rtraf_k^i = \min\left[R_k^i, \max\left(0, VTtraf_k^i - CIP_k^i\right)\right] \qquad (4)$$

From equation (4) it can be seen that channels are reserved only when there is blocking and the number of calls in progress for virtual network i is below $VTtraf_k^i$

For N traffic classes, the total reserved bandwidth, $RBWtraf_k$, is given by

$$RBWtraf_k = \sum_{i=1}^{N} Rtraf_k^i \times r^i \qquad (5)$$

A trunk is busy (on load state B) when its idle bandwidth is below r^i. If it's greater or equal than r^i but lower or equal to the total reserved trunk bandwidth (equation 5), the trunk is on reserved (R) load state. For the remaining load states, boundary thresholds are proportional to the estimated offered-traffic-load level. Table 2 presents, according to [1], a summary of how the load states for the trunk from OS_j to TS_k can be computed using the value of the trunk idle bandwidth, $ILBW_k$.

From Table 2 it is visible that the number of reserved channels increases or decreases directly with the variation of estimated offered traffic load. The same thing happens to the number of free channels on a trunk needed for it to be labeled "lightly loaded".

2.1.2. Fair(er) Network Capacity Sharing

To understand how the traffic classes share available bandwidth, we will see how a call is allocated to the direct path. In RTNR, during the network engineering process, the bandwidth for trunk OS_j to TS_k is divided in N slices (where N is the number of traffic classes that share this trunk bandwidth). The slice corresponding to service class i is named $VTeng_k^i$.

While a traffic class i meets its blocking objective, the other classes are free to share the bandwidth not currently used in this service class slice ($VTeng_k^i$). However, as soon as traffic of class i fails to meet its blocking objective, the other service classes may be forbidden to allocate channels in the slice $VTeng_k^i$ for direct path calls. The number of reserved virtual channels, $Reng_k^i$, in the direct path trunk for traffic class i, is given by

$$Reng_k^i = \min\left[R_k^i, \max\left(0, VTeng_k^i - CIP_k^i\right)\right] \qquad (6)$$

Then the total reserved bandwidth $RBWeng_k$ for N traffic classes is

$$RBWeng_k = \sum_{i=1}^{N} Reng_k^i \times r^i \qquad (7)$$

Thus, if

$$CIP_k^i < VTeng_k^i \qquad (8)$$

then, a call in traffic of class i will always try to use a direct path. However, if

$$CIP_k^i \geq VTeng_k^i \qquad (9)$$

it can only select a trunk on the direct path if

$$ILBW_k \geq r^i + RBWeng_k \qquad (10)$$

where, we recall, $ILBW_k$ is the idle bandwidth in the trunk OS_j to TS_k.

2.1.3. Two-trunk Path Selection

If a call cannot be established in the direct path, a two trunk path will be searched for. When the Originating Switch looks for a two trunk path, it chooses, from the available paths, the one with the lowest occupation, that is the one with the lightest load state. If no two trunk path in a lightly loaded state is available, it can yet be possible to establish the call in a "controlled use path". Controlled use paths are those with at least one trunk in the Heavily Loaded or Reserved state. The controlled use paths, can only be used if certain conditions hold (for details, see [2]). Moreover, only in rare situations can alternative traffic be routed through paths that have at least one trunk in the Reserved state (one such occasion may be when there is no direct path).

Using the previous rules, traffic will be routed preferentially on network areas that have higher availability. For this purpose, each switch OS_j must keep not only information

Trunk Idle Bandwidth	State
$0 \leq ILBW_k < r^i$	Busy
$r^i \leq ILBW_k \leq RBWtraf_k$	Reserved
$RBWtraf_k < ILBW_k \leq .05 \times VTtraf_k^i \times r^i + RBWtraf_k$	HL
$.05 \times VTtraf_k^i \times r^i + RBWtraf_k < ILBW_k \leq 0.1 \times VTtraf_k^i \times r^i + RBWtraf_k$	LL3
$.1 \times VTtraf_k^i \times r^i + RBWtraf_k < ILBW_k \leq .2 \times VTtraf_k^i \times r^i + RBWtraf_k$	LL2
$.2 \times VTtraf_k^i \times r^i + RBWtraf_k < ILBW_k$ and $ILBW_k > r^i$	LL1

Table 2. Trunk Load States by Idle Bandwidth

about the trunks directly connected to it, but also about the state of every trunk connected to every destination switch TS_k.

As RTNR must use aggregate link state information for each trunk, each switch TS_k uses a simple and compact mechanism to store the state information for every incident trunk. A bit map is used with a number of entries equal to the number of switches in the network. Each switch TS_k has several bit maps, one for each load level. For every particular load level, its bit map describes which of the the trunks that connect the other switches to TS_k are in that particular load level or on a lighter load level.

Whenever a switch OS_j requires information from a destination switch TS_k, it sends to TS_k a CCS network request. TS_k answers the request by providing the local bit maps. To find the best two-trunk path, the Originating Switch performs, iteratively, the bitwise AND of one of the bit maps received from TS_k with its bit map, thereby creating a new bit map that describes the two-trunk paths at that load level (or lighter). The process starts with the lighter load state (LL1), follows in monotonically increasing load, and it should end when a two-trunk path is discovered. This is detected very easily by simple observation of the resulting bit map.

To minimize the call setting time, when a two-trunk path is needed, the originating switch, while waiting for information from the destination switch will try to establish the call using the most recent state information received from TS_k. It will use the updated information only if it could not establish the call with the saved information.

In RTNR, some values used by the switches are updated periodically, with an update period usually defined as 3 minutes. Those values are: the switch-to-switch blocking level (NN_k^i), the switch-to-switch reservation level (R_k^i),the estimate of the offered switch-to-switch traffic load (TL_k^i), the estimate of the number of virtual trunks required to meet the blocking objective ($VTtraf_k^i$), and also some values used to check if controlled use paths can be used.

To offer greater opportunities for carrying calls which require higher bandwidths (greater resource needs), the sums of equations (5) and (7) are only executed from the higher bandwidth transport classes down to the traffic class for which the trunk load state is being computed.

2.2. DAR

DAR is one of the simplest dynamic routing methods. In this method [8] each call is offered first to the direct path, and if this is blocked, the call overflows to the currently selected alternate two-trunk path. This means that DAR uses a preferred two-trunk path for traffic that cannot be carried in the direct path. For each switch pair (i, j), there is a preselected switch k that will be used as a transit switch, in the only alternate path allowed. The identity of that current intermediate switch k is stored in the originating switch.

If the call is delivered in the direct path, the transit switch remains selected. If the call is successfully offered to the alternate path, the transit switch also remains constant. Otherwise, if the call is blocked in the alternate path, that is, if it cannot be carried by either the $i-k$ or the $k-j$ trunks, the call is lost, and a new alternate path (and transit switch) is selected at random, from all possible two trunks paths, for the following calls.

Each trunk is given a trunk reservation parameter. A call can only be routed in an alternate path if it leaves, in each trunk on the path, a number of free channels at least equal to the respective trunk reservation parameter.

3. Considered Variants of two Routing Methods

The DAR method was developed only for routing single class traffic. However, to support the growing need for integrated service networks, multiple traffic classes have to be considered in the network. So, we developed an extension of the basic DAR method that can take into consideration several traffic classes (hereafter this variant will be designated as DAR0).

This variant required the addition of the following extensions:

1. Whenever a call arrives, its class must be determined, in order to obtain its characteristics, namely the bandwidth requirements;

2. Each originating switch must store, for each destination switch, as many alternate paths (transit switches) as the existing traffic classes. Each of these alternate paths will be used by the corresponding traffic class when its calls overflows the direct path.

In the RTNR method the number of reserved virtual channels, for service class i, $Rtraf_k^i$ (equation 4), must be checked every time the originating switch obtains a free path for a new call or signals the disconnection of an existent one. In order to simplify this calculation we are going to analyze a new version of RTNR where $Rtraf_k^i$ is always equal to R_k^i (this variant will be designated as RTNR1). In the original version (designated as RTNR0) the value of $Rtraf_k^i$ is always equal or less than R_k^i.

As previously explained, in the RTNR method, the number of reserved trunks, in one trunk, depends on the switch-to-switch blocking detected over a periodic update interval of 3 minutes, among other parameters. Therefore, the trunk reservation mechanism is adaptive. On the other hand, the DAR routing method uses fixed trunk reservation. Having in mind the known very good results achieved by RTNR [3], and assuming they are due, at least partially, to the adaptive trunk reservation method used, we implemented another variant of the DAR method (denoted by DAR1), which uses an adaptive trunk reservation mechanism where the number of reserved channels is computed according to the RNTR simplified version (RTNR1) described earlier. This variant also makes use of the RTNR service class protection mechanism, which operates by controlling the access to the direct path.

To sum up, we propose 2 DAR variants: the first one is a simple extension of original DAR to support simultaneously several traffic classes, using fixed trunk reservation to protect traffic in the direct path and using no service protection mechanism (DAR0); the second one, besides supporting several traffic classes, uses dynamic trunk reservation, of the type used in RTNR, both for protection of traffic in the direct path and for service protection (DAR1). For the RTNR method, besides the original version (RTNR0), a variant where the number of reserved channels for protecting traffic in the direct path is computed in a simplified fashion, was also considered. However, as in the actual simulations results we noticed almost complete coincidence between RTNR0 and RTNR1 performance, only the RTNR0 (original RTNR) results will be presented, and the designation RTNR will be understood as referring to this version.

In the presented simulation study, besides multi-class Poisson traffic flows, we also considered Engset (finite source traffic) traffic flows. While Poisson traffic is usually considered in the simulation of this type of networks, the second type of traffic was also used because it might be adequate to describe traffic inputs of certain service classes (associated with calls with high resource needs). Note that the simulation of this type of traffic is not usually found in performance studies of dynamic routing methods. While the call arrival intensity is constant for Poisson traffic, for Engset (finite source) traffic, it depends on the system state (see, for example, [4]) and is proportional to the current number of free sources.

4. Main Results

For simulation purposes, a fully meshed network topology with 6 nodes, widely used in dynamic routing studies, was considered (see [8]). Some of the results obtained are now presented.

The study was focused on two networks, differing mainly on the number and characteristics of the traffic classes included. For each of these systems, the network mean blocking probability was computed using several simulation runs. Blocking probabilities for each traffic class were also estimated. A comparison of the values obtained by a discrete-event simulator, described in [6] using the variants referred to, is also presented.

The routing methods are compared subject to different load conditions, from nominal load up to 100% overload. For each one, 5 independent replications were performed. The duration of each simulation run was 48 hours of simulated time and the warm up time was 24 hours. Confidence intervals of the estimates of call blocking probabilities were calculated using the method of independent replications.

4.1. System 1 – A system with 2 traffic classes

In the circuit switching network with the previously defined topology, the offered traffic is (in this system) a mixture of two Poisson traffic flows, the network resources being shared by two traffic classes. For one of the flows (class 0) the number of required channels for each call is 1 (traditional telephone traffic) while to the other (class 1) 2 channels are required simultaneously.

The node-to-node traffic offered for service class 0 calls is the one presented in [8]. The traffic of the service class 1 calls is one fifth of this value. The capacity of each network link was changed as compared to the original network [8], so that the expected blocking values in the redimensioned network (with additional traffic) would be similar to blocking probability in the original network (with only class 0 traffic).

4.1.1. Analysis of Network Mean Blocking The DAR routing method presents significant performance variations, depending on the used trunk reservation scheme. For low overload factors (ρ below 1.4), DAR using RTNR-like trunk reservation (DAR1) presents better results than with fixed trunk reservation (DAR0). For higher overload factors, the opposite is true.

RTNR is the most advantageous method for low overload factors (below 1.4) while for higher overload factors, DAR0 performs better (see Figure 1).

In general, regarding global blocking probability, DAR1 presents a behavior similar to the RTNR method, but with a greater or equal blocking, as can also be seen in Figure 1.

It should be noticed that mean blocking probability alone is not a sufficiently good network performance measure. In fact, it gives the value corresponding to the blocking probability averaged over all traffic flows, where the blocking probability of the traffic flows with higher mean have a predominant weight. This entails that "smaller" traffic flows can experience excessive blocking probability (usually designated as marginal blocking probability). This can be particularly inconvenient for traffic flows with higher bandwidth requirements and moderate or low intensity, which have inherently a higher call congestion (marginal blocking probability).

Note that the mean blocking probability is given by:

$$B = \frac{\sum_f A(f)B(f)}{\sum_f A(f)} = \frac{\sum_k \sum_{f^{(k)}} A^{(k)}(f^{(k)})B^{(k)}(f^{(k)})}{\sum_k \sum_{f^{(k)}} A^{(k)}(f^{(k)})} \quad (11)$$

where $f^{(k)}$ represents end-to-end traffic flow of class k, $A^{(k)}(f^{(k)})$ is the mean offered traffic for $f^{(k)}$ (belonging to class k), $B^{(k)}(f^{(k)})$ is the mean end-to-end blocking probability for flow $f^{(k)}$, and B represents the probability of an arbitrary call offered to the network being blocked.

A possible form of obtaining a single, more evenly balanced, network blocking parameter is to define *a channel average* blocking probability:

$$B_c = \frac{\sum_k \sum_{f^{(k)}} C^{(k)}A^{(k)}(f^{(k)})B^{(k)}(f^{(k)})}{\sum_k \sum_{f^{(k)}} C^{(k)}A^{(k)}(f^{(k)})} \quad (12)$$

where $C^{(k)}$ is the number of channels needed for each k class call.

This is equivalent to replacing the values of the traffic offered $A^{(k)}(f^{(k)})$ (in Erlang) by the corresponding "equivalent" single channel traffic offered, obtained by multiplying $A^{(k)}(f^{(k)})$ by the number of channels required by calls of traffic flow $f^{(k)}$.

According to this measure B_c, the performance of RTNR, DAR1 and DAR0 is quite similar, as can be seen in Figure 2. DAR0 has a slightly lower blocking than RTNR for low overload factors (below 1.5). For higher overload factors, DAR0 performs better for the overload factors between 1.5 and 1.9.

4.1.2. Analysis of Blocking Probability per Traffic Class It can be seen from Figure 3 that DAR0 method gives higher call blocking probability for traffic class 1 for all the overload factors. This should be expected since only a fixed service protection for direct traffic is used.

On the other hand, using RTNR, for overload factors ranging from 1.4 to 1.8, the blocking probability is higher for the traffic class with lower resource needs. In this case, the service protection mechanism used by RTNR seems to be overprotecting the class with higher bandwidth requirements at the cost of higher blocking for traffic with lower resource needs. This seems to be the cause for the marked increase in the network mean call blocking probability experienced by RTNR (Figure 1) for those overload factors. DAR1, using RTNR-like trunk reservation, has shown a performance quite similar to RTNR concerning this measure.

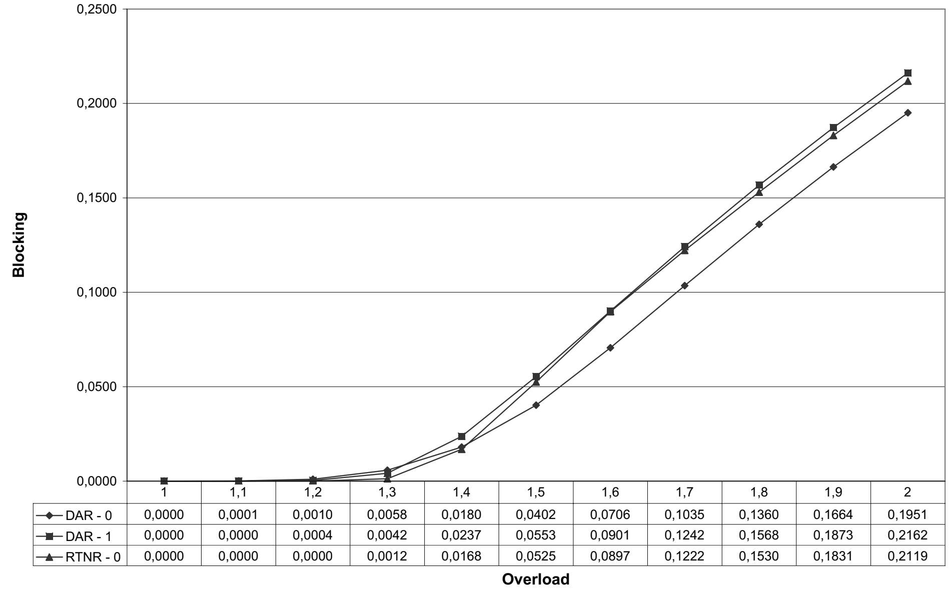
From the comparison between RTNR and DAR0, in Figure 3 and for overload factors below 1.4, blocking probability using RTNR is very small and similar for both traffic classes, and RTNR behaves better than DAR0 for lower overload factors. With RTNR, as load increases, the blocking probability increases more in the traffic class with lower resource requirements. Note that the blocking probability shows a steep increase for service class 1 after the 1.6 overload factor, and is higher than the blocking for service class 0 for overload factors greater than 1.8. On the other hand, DAR0 shows softer variations: service class 1 has always blocking levels higher than service class 0 and this difference increases with the overload factors.

For a summary analysis of the System 1, Table 3 presents the global behavior of this system, showing the relative performance of the methods RTNR and DAR0 in terms of the overload factors. The inflection points were estimated by linear interpolation.

Table 3 makes visible an apparent advantage for RTNR in low-medium loads (for an overload factor up to about 1.45), while DAR has better performance for higher overloads. It should be noticed, however, that simulation results allow the conclusion that this advantage is really strong only in terms of network mean call blocking, and is not so relevant for channel average blocking probability, namely for higher overload factors, up to an overload factor equal to 2. Note that for this range of the overload factor and taking into account the overlapping of the confidence intervals (results not shown in the figures) it is not possible to distinguish the performance of the two methods. It should also be noticed that RTNR provides better performance for service class 1 for all overload factors.

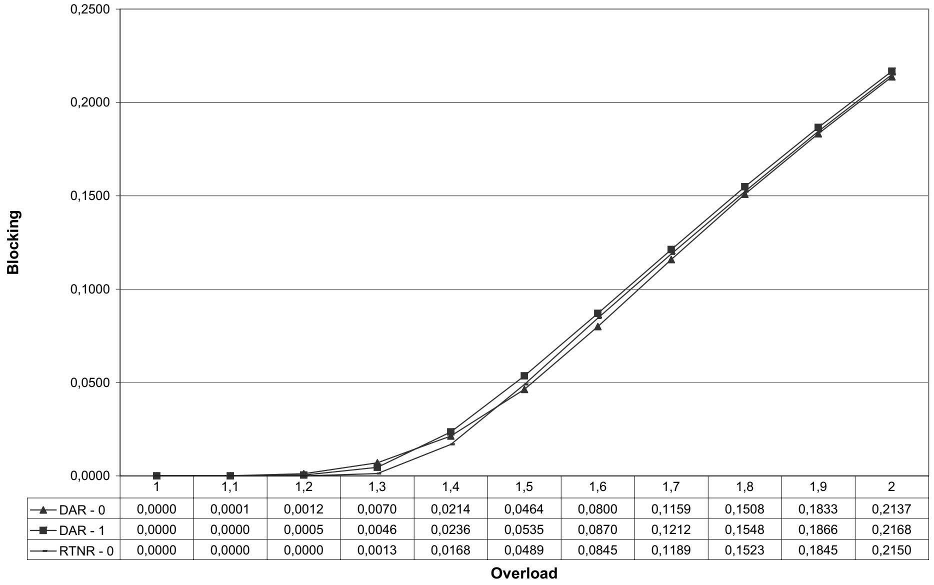
4.2. System 2 – A system with 4 traffic classes

In this system there are 4 traffic classes. The traffic flows are Poissonian for two classes and Engset-type (finite-source) for the other two. One of the Poisson traffic flows (class 0) requires 1 channel per call while the other (class 1) requires 2. Concerning the Engsetian traffic flows, the chan-



	1	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2
◆ DAR - 0	0,0000	0,0001	0,0010	0,0058	0,0180	0,0402	0,0706	0,1035	0,1360	0,1664	0,1951
■ DAR - 1	0,0000	0,0000	0,0004	0,0042	0,0237	0,0553	0,0901	0,1242	0,1568	0,1873	0,2162
▲ RTNR - 0	0,0000	0,0000	0,0000	0,0012	0,0168	0,0525	0,0897	0,1222	0,1530	0,1831	0,2119

Figure 1. Network Mean Call Blocking Probability

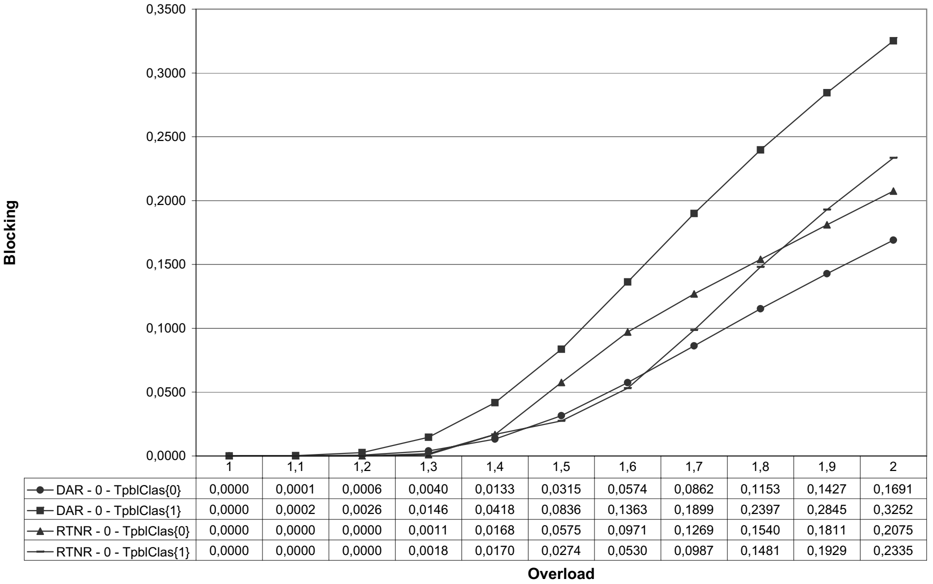


	1	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2
▲ DAR - 0	0,0000	0,0001	0,0012	0,0070	0,0214	0,0464	0,0800	0,1159	0,1508	0,1833	0,2137
■ DAR - 1	0,0000	0,0000	0,0005	0,0046	0,0236	0,0535	0,0870	0,1212	0,1548	0,1866	0,2168
— RTNR - 0	0,0000	0,0000	0,0000	0,0013	0,0168	0,0489	0,0845	0,1189	0,1523	0,1845	0,2150

Figure 2. Channel Average Blocking Probability

nels requirements are 6 and 24, for classes 2 and 3 respectively.

The node-to-node traffic offered by classes 0 and 1 is the same as for the identical classes in System 1. As we had previously done for class 1, the traffic for classes 2 and 3 was engineered as a fraction of traffic of class 0 (though a much smaller fraction). Unlike classes 0 and 1, we have generated traffic of classes 2 and 3 only between selected pairs of switches. The number of sources, the call rate per free source and the percentage of traffic from each source to each particular destination were specified for each traffic flow of classes 2 and 3. These parameters were estimated from the network capacity and traffic matrix using a procedure described in [5]. As in System 1, the number of channels in each network trunk was redimensioned to accommodate the additional traffic.



	1	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2
● DAR - 0 - TpblClas{0}	0,0000	0,0001	0,0006	0,0040	0,0133	0,0315	0,0574	0,0862	0,1153	0,1427	0,1691
■ DAR - 0 - TpblClas{1}	0,0000	0,0002	0,0026	0,0146	0,0418	0,0836	0,1363	0,1899	0,2397	0,2845	0,3252
▲ RTNR - 0 - TpblClas{0}	0,0000	0,0000	0,0000	0,0011	0,0168	0,0575	0,0971	0,1269	0,1540	0,1811	0,2075
— RTNR - 0 - TpblClas{1}	0,0000	0,0000	0,0000	0,0018	0,0170	0,0274	0,0530	0,0987	0,1481	0,1929	0,2335

Figure 3. Blocking Probability per Traffic Class

Performance Measure	ρ low	Inflection point	ρ medium - high
Mean Call Blocking Probability	RTNR	1.42	DAR
Channel Average Blocking Probability	RTNR	1.46	DAR
Traffic Class 0 Blocking Probability	RTNR	1.42	DAR
Traffic Class 1 Blocking Probability	RTNR	-	RTNR

Table 3. System 1 – Method with Better Blocking Performance – RTNR vs. DAR0

4.2.1. Analysis of Network Mean Blocking For lack of space we do not present the curves with results for this system. The major conclusions from the analysis of results are the following.

The DAR routing method presents significant performance differences depending on the used trunk reservation mechanism. For lower overload factors (below 1.4), DAR1 performs better than DAR0, while for higher overload factors the opposite is true. It can be seen that RTNR is slightly superior to both DAR variants for lesser loads (below overload factor 1.4), while for load factors 1.5 and above, DAR0 performed better.

These results are similar to those of System 1, and DAR1 behaviour is similar to RTNR, though always slightly worse.

It is also important to consider the network channel average blocking probability B_c, since in this system traffic from classes with greater channel requirements has higher impact in terms of link occupation than in System 1. In System 2 the performances of DAR0, DAR1 and RTNR are very similar with respect to B_c. RTNR slightly outperformed both DAR variants, for smaller loads (overload factors below 1.5) while for higher loads DAR0 performed slightly better (or similarly) to RTNR.

4.2.2. Analysis of Blocking Probability per Traffic Class Using DAR0, blocking probability is greater for traffic classes with greater resource needs. As for the other methods, the same pattern could be observed in RTNR, but in this case we also observed an inversion in the performance of class 0 as compared to class 1, similar to System 1. Also, DAR1 presents a behavior quite similar to RTNR. In particular, RTNR outperforms DAR0 for every traffic class for overload factors less than 1.4. However, for other load factors, in the medium-higher range, DAR0 performed better than RTNR for most traffic classes – this can be seen in Table 4. This table presents a global comparison of DAR0 and RTNR by showing the method with better performance in each traffic class for different overload factors, and signals the performance inflection points.

From Table 4 it is visible that for System 2, (likewise to System 1), RTNR seems to have a better performance

Performance Measure	low ρ	Inflection	medium ρ	Inflection	high ρ
Mean Call Blocking Probability	RTNR	1.46	DAR	-	DAR
Channel Average Blocking Probability	RTNR	1.46	DAR	-	DAR
Class 0 Blocking Probability	RTNR	1.45	DAR	-	DAR
Class 1 Blocking Probability	RTNR	-	RTNR	-	RTNR
Class 2 Blocking Probability	RTNR	1.45	DAR	1.75	RTNR
Class 3 Blocking Probability	RTNR	1.35	DAR	1.65	RTNR

Table 4. System 2 – Method with Better Blocking Performance – RTNR vs. DAR0

for low and medium loads (up to 1.45 overload factor), but for higher overloads DAR0 tends to perform better. However, this difference is again more visible in mean blocking probability than in channel blocking probability. As an exception to this trend, RTNR is the best method for class 1 through all overload factors, and is better for the higher resource needs classes (2 and 3) both for lower and higher loads.

5. Conclusions

Two variants of the DAR method for multi-service networks and a simplified version of RTNR, were presented. These implementations of the dynamic routing methods were compared, in terms of global network performance and performance per traffic class, by means of a discrete event simulation platform. Beyond the classical Poisson type traffic, Engset (finite source) traffic flows were included in the simulation models.

In the test network where all traffic flows are Poissonian the analysis of results enabled the conclusion that RTNR has better performance for low-medium overloads while DAR performs better for higher overloads. However these differences are less apparent for channel average blocking probability (defined in the fourth section) than for the network mean call blocking probability. Concerning the marginal blocking probabilities of the different traffic classes RTNR provided better performance for class 1 (two channels per call) for all overload factors. The relative performance of the method variants has no stable pattern for single-channel traffic, depending on the overload factor range. Concerning the comparison of the two DAR variants, DAR0 (with fixed trunk reservation) and DAR1 (with dynamic trunk reservation per traffic class of RTNR type), it can be said that DAR1 performes better for low overloads and the situation is the opposite for high overloads, following a trend similar to RTNR.

The experimentation with the network where the traffic offered is a mixture of Poisson and Engset type traffic flows revealed similar trends in terms of relative performance of the variants of the dynamic routing methods.

A major conclusion of this study is the importance of the adopted service protection mechanisms. In most cases the relative performance of a dynamic routing method depends more on the used protection mechanism than on the basic routing algorithm of the method. This is particularly critical when one analyses the performance per traffic class.

Acknowledgment

Work partially supported by programme POSI of the IIIrd EU programme co-sponsored by FEDER and national funds.

References

[1] G. R. Ash. *Dynamic Routing in Telecommunications Networks*. Mc Graw Hill, New York, 1998.

[2] G. R. Ash, J. S. Chen, A. E. Frey, and B. D. Huang. Real-time network routing in a dynamic class-of-service network. In *13th International Teletraffic Congress*, pages 187–194, 1991.

[3] G. R. Ash and B. D. Huang. An analytical model for adaptive routing networks. *IEEE Transactions on Communications*, 41 (11):1748–1759, November 1993.

[4] R. B. Cooper. *Introdution to Queueing Theory*. North Holland, New York, Oxford, 1981.

[5] L. Jorge. Um estudo simulacional de redes inter-centrais com encaminhamento dinâmico - incluindo redes com integração de serviços. Master's thesis, Departamento de Engenharia Electrotécnica e de Computadores, Universidade de Coimbra, 2001.

[6] L. Jorge, J. Craveirinha, and T. Gomes. A simulator for multiclass traffic networks with dyamic routing methods. In *4th Conference on Telecommunications*, pages 363–366, Aveiro, Portugal, June 2003.

[7] F. P. Kelly. *Stochastic networks, theory and applications*, chapter Notes on effective bandwidths, pages 141–179. Claredon Press, Oxford, 1996.

[8] D. Mitra and J. B. Seery. Comparative evaluations of randomized and dynamic routing strategies for circuit-switched networks. *IEEE Transactions on Communications*, 39(1):102–116, January 1991.