Internet Draft   Eugenia Nikolouzou, NTUA
<draft-nikolouzou-bgrpp-sim-00.txt>   Peter Sampatakos, NTUA
                                                         Lila Dimopoulou, NTUA
July 2002   Iakovos S. Venieris, NTUA
Expiration: January 2003   Martin Winter, Siemens AG
                                                        Bert F. Koch, Siemens AG
                                                        Thomas Engel, Siemens AG
                                                        Stefano Salsano, CoRiTeL
                                                       Vincenzo Genova, CoRiTeL
                                                        Fabio Ricciato, CoRiTeL
                                          Gerald Eichler, T-Systems Nova GmbH

BGRPP: Performance evaluation of the proposed Quiet Grafting mechanisms

Status of this Memo

Copyright Notice

Abstract

End-to-end provisioning of quality of service guarantees is nowadays one of the most basic and crucial issues. Towards this end, the inter-domain

resource control architecture for DiffServ networks described in [1],
tries to address this issue by applying a scalable and efficient
signaling approach. Based on this architecture, this paper tries to
provide an analysis and a performance study of the introduced quiet
grafting mechanisms.

Table of Contents

1. Introduction

Quality of Service provision is one of the main points of focus for the
next generation Internet architectures. The existence of various QoS
technologies necessitates the need to endow this diverse environment
with individual inter-domain mechanisms for the provision of end-to-end
QoS to users.

However, the scalability problem introduced by the number of resource
reservation requests is one of the main problems in an inter-domain
scenario. A distributed approach is required for inter-domain resource
reservations that can scale in terms of message processing load, state
storage and control message load. Current proposals, such as RSVP, lack
efficiency and scalability.

Toward the confrontation of this need, the BGRP Plus (BGRPP)
architecture described in [1], proposes a solution to the scaling
problem, by providing sink tree based aggregation for resource
reservations over a network of DiffServ domains. In fact, since routers
only maintain the sink tree information, the total number of reservation
states at each router scales, in the worst case, linearly with the
number of domains in the Internet.

In addition, with the deployment of the quiet grafting mechanisms, the
number of signaling messages is reduced, since each signaling message
has not to travel all the way from the source to the destination domain.
As shown in [2], the lack of quiet grafting mechanisms may lead to
scalability problems, especially for short-lived small bandwidth flows,
such as VoIP. Transit domains are mainly affected by this behavior.

In [1], a first perspective to quiet grafting mechanisms is given, while
in this draft these mechanisms are further analyzed and verified through

extended simulation scenarios. Therefore, a clear and solid approach for
reducing the signaling overhead of the aforementioned BGRP framework is
introduced while the applicability of the results, especially with
respect to real world networks and scenarios is stated.

2. Quiet grafting

2.1. Requirements

As described in [1], the fundamental concept of BGRPP is the aggregation
it performs on destination-based inter-domain reservations and thus
resulting to the formation of sink trees. It is therefore evident that
pure BGRP addresses the problem of scalability to a certain extent, i.e.
it significantly reduces the amount of control state information kept at
routers in conjunction with the volume of REFRESH messages needed for
the maintenance of the corresponding state. Of interest to us, however,
is enhancing BGRP with additional mechanisms so as to achieve a
reduction in the number of PROBE and GRAFT messages as well.

As stated in [1], in order a resource reservation request to be
successfully established in the network, the PROBE message should be
forwarded between BGRPP agents hop-by-hop along the BGP route until it

reaches the destination domain. Each BGRPP agent forming the end-to-end
path should check the resource availability. The last BGRPP agent, which
corresponds to the root of the sink tree, can assign a sink tree
identifier to the reservation, which uniquely identifies the sink tree
the reservation belongs to. Then, a GRAFT message is generated containing
this identifier.

Consequently, signaling messages still have to travel the full path from
the source to destination increasing the signaling overhead and
utilizing a significant amount of bandwidth. Aiming at reducing the
signaling overhead, the quiet grafting mechanisms are introduced.
The quiet grafting mechanisms should provide an intermediate BGRPP
agent with the necessary functionality in order to successfully answer
a PROBE message, before the latter arrives at the destination domain.

Towards this end, the BGRPP agent must be able to identify the sink
tree, to which the reservation belongs and in addition must have
pre-reserved resources for this sink tree, so that it can guarantee that
resources are available on the path from the current point to the
destination domain.

2.2 BGRPP Enhancements

Based on the above requirements, it is essential that a new reservation

can be classified earlier to a sink tree and moreover that resources are available for that reservation. The most adequate mechanisms for achieving these two goals are the network layer reachability information (NLRI) labeling of the sink tree and the introduction of an efficient resource management algorithm.

The first one will enable an early sink tree identification, i.e. it will classify a new reservation to its corresponding tree. This is mainly due to the fact that the nodes of a sink tree can be aware of the NLRI information of the destination domain, which serves the root of the tree. Therefore, when a new reservation request arrives (PROBE message), a node can verify if its destination is pertained to the domains that correspond to the NLRI label of a sink tree. However, it is not our intention to delve into the mechanisms that distribute this kind of information since they have already been described in [1]. On the contrary, the effectiveness of the distribution of the NLRI information associated to the root of the tree will be studied.

Moreover, the resource management algorithm will perform hysteresis with respect to delayed bandwidth release so as to maintain available resources while achieving high resource utilization. The concepts of this algorithm will be examined in detail and its performance will be studied when applied to a real network topology.

2.2.1 Hysteresis for the creation of Resource Cushions

Irrespective of the identification of a sink tree, the quiet grafting of a new reservation will not be feasible if there are not enough resources so as to accommodate the new request. Therefore, it is required that BGRPP nodes are assigned with more bandwidth than

currently reserved. To accomplish that, over-reservation, quantization and hysteresis techniques are likely to be employed.

The first two approaches are likely to introduce some limitations with respect to their impact on the Quiet Grafting probability and can inadvertently produce the opposite effects to the ones envisioned. Over-reservation, when performed without control, can lead to a reduction of the Quiet Grafting probability. This is due to the fact that future requests, coming from BGRPP nodes other than the ones with over-reserved resources, are likely not to be grafted onto the tree or not to be accommodated at all. Another impact of over-reservation requests can be PROBE messages that travel unnecessarily towards the root of the tree even if the requested resources are available, operating as before at the expense of Quiet Grafting.

Quantization of the requested resources, i.e. rounding them up to a

multiple of a chosen quantum, can lead to undesirable results as well. This is particularly true in the case of sink trees that consist of many leaf nodes (N) and are moreover incapable, due to lack of future reservation requests, of using the remaining amount of the reserved bandwidth.

As a result, the root of the sink tree will end up having reserved at least N*quantum resources, far surpassing the actual needs. Thus, the adoption of the quantization mechanism can only be justified for sink trees with a limited amount of leaves and a much greater amount of reservation requests.

In essence, given the aforementioned concerns coupled with the complexity induced by the adoption of the corresponding techniques, it is proposed that the BGRPP nodes do not perform over-reservation or quantization upon receiving a new reservation request. Instead, hysteresis on the release of resources is more appropriate for the formation of resource cushions at the BGRPP nodes. The resource cushion mechanism that will be employed bases its resource release policy on the existence of the release period and thresholds. In the next section the corresponding algorithm is described in detail.

2.2. Resource Cushion Mechanism

When a BGRPP agent receives a REFRESH message indicating a smaller amount of resources than currently reserved and decides not to further forward this message downstream towards the root of the sink tree, then it allocates resources downstream, which are not in use upstream. These resources, which are reserved downstream but not upstream of a BGRPP agent due to retained REFRESH messages, are called resource cushions. A resource cushion for a specific BGRPP agent is tied to a sink tree. A BGRPP agent may build resource cushions for all of its sink trees. Building resource cushions has as an impact the reduction of the signaling load, since retained REFRESH messages reduce the signaling load of downstream domains. The use of resource cushions for arriving reservation requests further reduces the signaling load of downstream domains, when reservation requests are not forwarded but served from resource cushion immediately.

A BGRPP agent uses two parameters to control the size of its resource cushions. These parameters are:

RBS: Release block size
RP: Retain period

Both parameters together define a virtual release rate RR = RBS/RP.

Whenever a resource cushion exceeds the RBS, a release timer is
activated which runs down during the duration of a single RP. If a
resource cushion shrinks to a size below RBS during a running release
timer, then the release timer is cancelled. In the case where the
release timer runs out without being cancelled, then the corresponding
resource cushion is decreased by RBS and a REFRESH message releasing
this amount of resources is generated and sent downstream to the next
BGRPP agent on the sink tree. Thus, resource cushions are reduced with
the release rate RR unless they are used to serve arriving resource
requests. RBS and RP are the parameters that control the release rate.
In this way released resources are not immediately forwarded towards
the sink of the tree, but are used to form resource cushions.

Additionally, those retained resources are released step-wise improving
in this way the performance of the network.

[3](). Evaluation of the Quiet Grafting Mechanisms

[3.1](). Simulation goals

In this section, the gain of introducing the quiet grafting mechanism
in a network will be evaluated, in terms of average hop number
reduction that a PROBE (GRAFT) message has to traverse to reach its
destination. Therefore, two simulation scenarios, aiming at a dynamic
analysis, will be specified, whereas the topology, the traffic model and
the parameters of the resource cushion mechanism will be defined.

Quiet grafting aspires to reduce the average number of traversed hops
for a signaling message, and consequently the total number of messages
in the network. As a result, the performance of the network with respect
to CPU usage and bandwidth used for signaling messages will also be
improved.

It is also worth mentioning that there is a trade-off between delayed-
release and resource utilization. We are seeking a relation between the
gain in terms of reduction of the average number hop traversed, and the
utilization of the reserved resources.

Both simulation scenarios aim at studying and evaluating the efficiency
of the quiet grafting mechanism. The first scenario serves as a proof of
concept for the early tree identification mechanism evaluating the gain
on the message path length reduction.

The second scenario validates the applicability of the resource cushion
algorithm and evaluates the performance of the proposed approach.

. Scalability Issues

Our main goal is to evaluate the effectiveness of the Quiet Grafting mechanism in terms of the number of hops that a reservation request has to traverse in order to be accommodated into the corresponding sink tree. It is assumed that resources are always available, and therefore the effectiveness of the NLRI information distribution will be solely examined.

Assumptions for sink tree creation

The topology for carrying out the simulations has been defined after taking into consideration the actual topology of the Internet. More precisely, the number of the AS domains that an inter-domain reservation can possibly traverse will not exceed the 9. In addition, for each transit AS domain, there will be at least two border routers that will forward the reservation request to the border router of the adjacent AS domain, upstream and downstream. Given the fact that there will be one BGRPP node corresponding to a border router of the AS domain, we can deduce that the path being traversed by a reservation request will consist of a maximum number of 18 BGRPP nodes.

Hence, the simulations performed are based on sink tree topologies that vary in depth (4 to 14) in order that they reflect the actual topology of the Internet. Moreover, in regard to the spreading of the sink trees, additional simulations targeted for the analysis of topologies varying in width (3 to 4 children for each node) have not produced a divergence on the results. Therefore, binary trees have been solely examined since more complex topologies are not considered to be of significant added value with respect to the goal of the simulations.

For identifying the effectiveness of the NLRI information distribution to the nodes of a sink tree, it is essential to examine how the number of populated nodes can contribute to the reduction of the path length. With the term "populated nodes" we denote the BGRPP elements that are aware of the NLRI information of the destination domain (root of the tree) and therefore can identify the destination (sink tree) of the impending reservation request. These nodes are assigned the task of intercepting a reservation request before reaching the root of the tree and silently grafting it onto the sink tree. It would be ideal if every node of a sink tree was populated but obviously, the "number" of populated nodes performing this identification introduces a scalability problem. If this were true, then we would actually bring into scene again the problem that has been tackled by the BGP through the aggregation it performs on routes.

Given a particular sink tree, an "initial state" (number) of populated nodes needs to be created in relation to which, the mean path length of a potential reservation request is computed. This state is produced after performing a certain number of reservations towards the root of the tree. The nodes of the tree (not necessarily leaves) dedicated for

initiating those reservations are randomly chosen. All the nodes that
are traversed while the reservations make their way up to the root are
transformed to populated nodes.

In this way, an initial state of populated nodes can be produced whose
topology significantly mitigates the potential degree of homogeneity
introduced by the binary tree.

Our aim is to compute the impact of this state (number of populated
nodes) on the average number of nodes that a reservation request has to
traverse before striking a populated node. To that end, a certain amount
of additional reservation requests needs to take place.

As before, randomly selected nodes initiate those reservations, which
do not modify the initial state of populated nodes, i.e. they do not
produce additional populated nodes, as is the case with the creation of
the "initial state". Therefore, a mean value of the number of nodes
traversed before striking a populated one can be computed for a
particular populated state.

3.2.2 Simulation Results

The simulations were carried out for a variety of binary sink trees and
for a variety of "initial states" (number of populated nodes) for each
tree.

In Table 1, it can be seen how a particular sink tree of depth 10,
(which has a total number of 2047 nodes) behaves to an increase of the
percentage of populated nodes. The specific binary tree has been chosen
since it is considered as the most representative for an inter-domain
reservation (4 transit AS domains). In the first column, the number of
populated nodes is depicted whereas in the second one, the percentage
of nodes that comprise the populated ones is shown. Finally, the third
column illustrates the mean hop count of a reservation for a certain
number of populated nodes whereas in the fourth column, the percentage
of the achieved path length reduction is presented.

 Table 1: The Mean path length in relation to the number of populated
 nodes for a tree with depth=10

populated nodes	populated nodes %	Mean path length	path length reduction%
82	4	5,214	47,8
123	6	4,358	56,4
164	8	3,952	60,5
204	10	3,410	65,9

```
|   245   |   12    |   3,204     |   67,9    |
|   307   |   15    |   2,793     |   72,1    |
|   350   |   17    |   2,569     |   74,3    |
|   410   |   20    |   2,267     |   77,3    |
+---------+---------+-------------+-----------+
```

It can be seen from Table 1 how the number of populated nodes, which comprises a relatively low percentage (4%-20%) of the total number of nodes (2047), affects the average path length of a reservation request.

Having in mind that for a non-populated tree (Quiet Grafting is not activated) the number of traversed nodes is equal to its depth (10), it can be deduced that the percentage corresponding to the reduction of the actual path length attains the values of around 47,8% to 77,3% for remarkably small percentages of populated nodes.

However, it can be seen from Table 2, that in the case of sink trees of small depth (1 to 3 transit domains), the percentage reduction of the path length does not attain the same values. In fact, a comparison between 5 sink trees with depth values of 4, 6, 8, 10 and 12 is presented in terms of the path reduction for the same percentage of populated nodes. It is evident that the sink trees of small depth do not exhibit the same effectiveness on the reduction of the path length for a certain percentage of populated nodes. For example, a sink tree of depth 6 with 20% populated nodes will attain a 63,7% reduction of the path length whereas a sink tree of depth 12, for the same percentage of populated nodes, will achieve an 80,6% reduction of the path length.

Table 2: The percentage path length reduction in relation to the percentage of populated nodes for trees with depth 4, 6, 8, 10 and 12

Populated nodes %

depth	6%	10%	15%	20%
4	32,3	39,5	45,0	49,5
6	38,3	49,3	56,7	63,7
8	40,8	55,9	65,3	69,5
10	56,4	65,9	72,1	77,3
12	63,3	71,3	76,7	80,6

3.3. Performance Evaluation of the Resource Cushion Algorithm

3.3.1 Assumptions for topology

In order to evaluate the proposed quiet grafting mechanism, the resource
cushion algorithm technique and the ability of early sink tree
identification are used in conjunction. It is assumed that every BGRPP
agent of the sink tree can identify the destination domain, and
therefore can execute the following step; the BGRPP agent will check,
whether it has enough resources to accommodate a new reservation
request. If there are sufficient resources, the PROBE message will be
terminated and a GRAFT message towards the corresponding source will be
generated.

For that purpose, a simulation scenario is specified, which is based on
the realization of a sink tree reflecting a real network that is
consisted of a number of inter-connected DiffServ domains. In fact, a
simple binary and complete tree is proposed, since the properties of
symmetry simplify the interpretation of the results. Each node of the
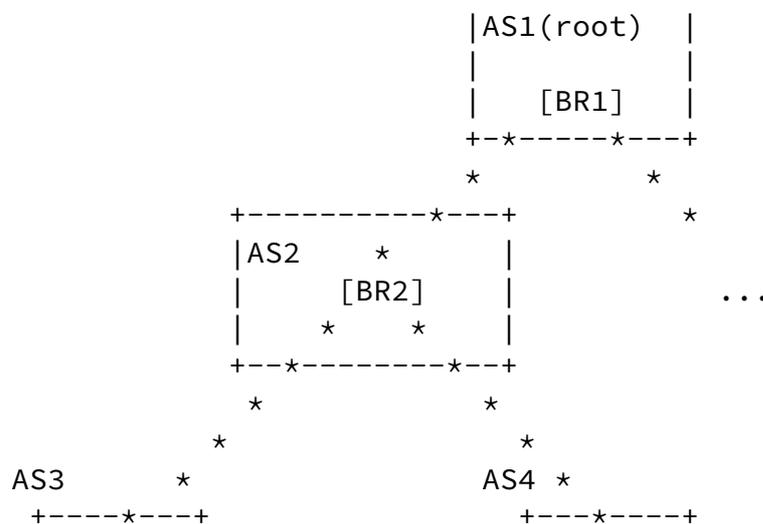sink tree will be a BGRPP agent capable of performing quiet grafting.

In order to have a relatively large number of nodes, while keeping at
the same time complexity at a low level, the depth of the tree for this
simulation scenario was chosen to be equal to 4. That results in a total
number of N = 31 nodes. This scenario could represent a real network
topology, consisting of a number of source domains, the destination
domain and the three transit domains in order to form the binary tree as
depicted in Figure 1. We have also made the assumption that traffic is
injected into the network from the leaves of the tree, which correspond
to the source domains. Therefore, reservation requests are only
generated at the edges of the network topology. Moreover, as it can be
observed in Figure 1, there is only one border router in each AS
belonging to the sink tree formed with destination the AS1.

```
                                |AS1(root)  |
                                |           |
                                |   [BR1]   |
                                +-*-----*---+
                                  *         *
                  +---------*---+           *
                  |AS2     *    |
                  |      [BR2]  |            ...
                  |    *     *  |
                  +--*--------*--+
                    *           *
                  *               *
        AS3        *           AS4 *
          +----*---+             +---*----+
```

```
                  |  [BR3] |                      |  [BR6] |
              +--*--*--+                       +--*--*--+
                  *       *                        *        *              ...
            +---*-+     +--*--+              +--*--+       +*---+
         AS5  |[BR4]| AS6 |[BR5]|        AS7 |[BR7]|   AS8 |[BR8]|
            +*--*-+     +-*--*+              +*--*-+       +-*--*+
              *       *     *      *          *      *       *      *
                *       *      *      *        *      *       *      *
            +----+ +----+ +----+ +----+    +----+ +----+ +----+ +----+
            |[BR]| |[BR]| |[BR]| |[BR]|    |[BR]| |[BR]| |[BR]| |[BR]|
            |    | |    | |    | |    |    |    | |    | |    | |    |   ...
            |AS9 | |AS10| |AS11| |AS12|    |AS13| |AS14| |AS15| |AS16|
```
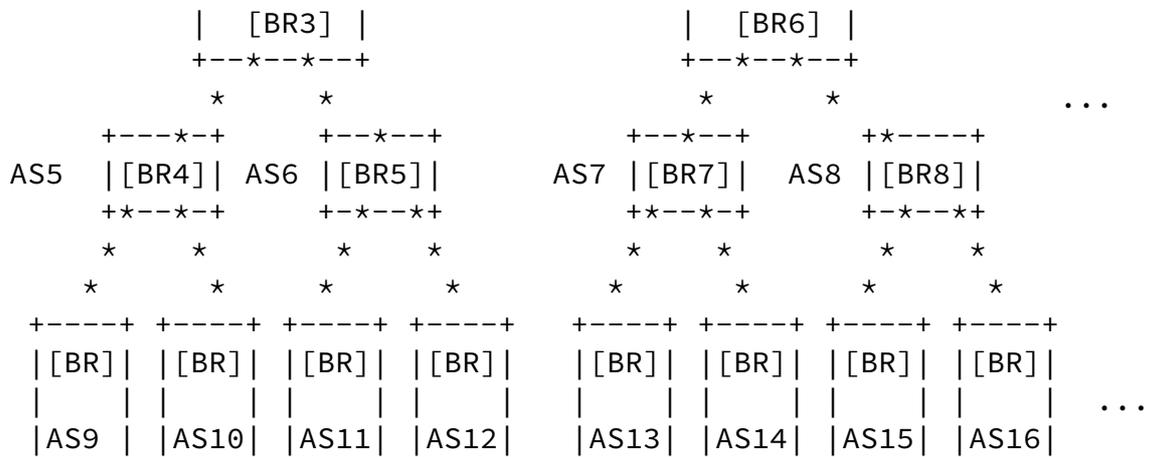
   Figure 1: The created sink tree

3.3.1.1 Traffic Model

Concerning the traffic model used for the simulation scenario, a traffic
generator that generates reservation requests with exponential
distributed inter-arrival times and exponential distributed holding
times is attached to nodes belonging to source domains. As regards the
size of each reservation request, for the sake of simplicity, a
homogeneous scenario is assumed where all reservations have the same
fixed size. It is assumed, without loss of generality that each request
asks for one unit of bandwidth, since an infinitive bandwidth capacity
is presumed for every node of the tree.

Two different scenarios are considered with different traffic loads, in
order to examine the effect of load on the performance of the quiet
grafting, with 20 and 100 flows average accordingly. The simulation
scenarios imply that in an initially "reservation free" tree, traffic
flows will be generated at the leaves of the tree contemporarily,
following the traffic pattern previously described.

During the initial phase, each reservation request (PROBE message) has
to travel all the way up to the root of the tree in order that resources
are reserved. After the initial phase, sequential requests can be
potentially accommodated from the already reserved resources, due to the
resource cushion algorithm given the fact that the sink tree has been
successfully early identified.

3.3.1.2 Performance measures

The following measures are used to rate quiet grafting mechanism
performance:

Ptot : the overall sink tree average utilization
Stot : the overall sink tree average signaling overhead

The Ptot is defined as the average utilization of the nodes, where
Ptot = sum_of_active_reservations_of_nodes/total_reserved_bw.

Additionally, the signaling overhead is defined as the percentage of
the arriving PROBE messages, which are forwarded to the next BGRPP
agent.

The objective of the simulations is to present the effectiveness of the
introduced resource cushion mechanism on reducing the number of messages
processed by each node, limiting in this way the number of signaling
messages, and on improving the accomplished performance.

Based on the realized resource cushion mechanism, there are two
parameters, which need to be appropriately tuned for achieving the
desired network performance: the RBS and the RP. The guidelines for
setting these parameters compromise a trade-off between achieved
resource utilization and signaling overhead. A great value of RBS may
result in a low network performance, but will provoke a smaller number
of exchanged signaling messages. On the other hand, a small value of RP
results in frequent checks on the current resource status, and therefore
improves the network utilization, but burdens the routers' processing.

In the next section, the impact of those parameters is examined.

3.3.2 Simulation results

3.3.2.1 Overall sink tree performance

In Table 3, the setting of the parameters is given as well as the
corresponding scenarios. The results concerning the overall utilization
and signaling overhead of the sink tree are shown in Table 4 and 5.
[All simulation scenarios have lasted 10 hours].

  Table 3: Setting of parameters RP and RBS

     Scenarios   Release   Release   Load
                     Period (sec)   Block Size
    +------------+-----------------+-----------+-----------+
    | Scenario 1 |10 - 200(step 20) | 1 unit | 20 flows |
    | Scenario 2 |10 - 200(step 20) | 1 unit | 100 flows |
    | Scenario 3 |10 | 1-20 units | 20 flows |
    | Scenario 4 |10 | 1-20 units | 100 flows |
    +------------+-----------------+-----------+-----------+

```
  Table 4: Results for different RP    Table 5: Results for different RBS

   RP       Scenario 1       Scenario 2   RBS       Scenario 3    Scenario 4
  (sec)                                   (units)
  +-----+-----+-----+-----+-----+        +----+-----+-----+-----+-----+
  |     | Ptot | Stot%| Ptot | Stot%|    |    | Ptot | Stot%| Ptot |Stot% |
  +-----+-----+-----+-----+-----+        +----+-----+-----+-----+-----+
  | 10  |0,9245|34,450|0.9418|12,990|    | 1  |0,9259|34,143|0,9409|12,998|
  | 20  |0.9157|22,723|0.9261|7,9394|    | 2  |0,9391|40,212|0,9554|19,313|
  | 40  |0.8761|14,721|0.9083|4,9259|    | 4  |0,9427|39,050|0,9583|25,005|
  | 60  |0.8668|10,497|0.8989|3,9021|    | 6  |0,9303|35,522|0,9679|25,159|
  | 80  |0.8479|8,9639|0.8934|3,3312|    | 8  |0,9237|31,444|0,9663|25,340|
  | 100 |0.8444|7,7516|0.8896|3,0331|    | 10 |0,9116|28,899|0,9659|22,763|
  | 120 |0.8307|6,7801|0.8887|2,6848|    | 12 |0,9042|24,921|0,9627|23,037|
  | 140 |0.8287|6,0672|0.8831|2,5475|    | 14 |0,8948|22,997|0,9612|20,362|
  | 160 |0.8294|5,5970|0.8847|2,4444|    | 16 |0,8831|17,293|0,9583|20,843|
  | 180 |0.8225|5,1285|0.8776|2,3528|    | 18 |0,8744|17,177|0,9560|18,648|
  | 200 |0.8262|4,8331|0.8765|2,2171|    | 20 |0,8585|13,695|0,9535|17,188|
  +-----+-----+-----+-----+-----+        +----+-----+-----+-----+-----+
```

Table 4 presents the effect of the release period RP on the performance
of the resource cushion algorithm. As expected, the overall signaling
overhead percentage decreases, respectively to the increase of the
release period. By increasing the RP parameter (which represents the
time during which the free resources should exceed the RBS), resources
are released less frequently. Longer retained resources may accommodate
future requests, limiting in this way the number of requests forwarded
to the next nodes. On the other hand, the increase of the RP has a
negative impact on the average utilization of the sink tree, since the
resource status is checked less frequently, resulting in longer
intervals between subsequent releases of resources. Notice that under
heavy load conditions the signaling overhead is reduced to 2,2171%.

We have to stress here that since each simulation scenario lasted 10
hours, the values of the release period RP were chosen to be
comparatively small to the simulation duration.

In Table 5, we can observe the impact of RBS parameter on the
performance of the quiet grafting mechanism. As it can be seen, the
effect of the RBS is not similar to the one of RP.

We can observe that there is a maximum of request forwarding activity
between small and large values of RBS.

Moreover, the randomly changing but stationary load, provokes a
re-allocation of the released bandwidth, which results in request
forwarding. Notice that there is an optimum RBS that follows changing

demand best. We have to stress here that:
-lower RBSs do not decrease allocated bandwidth fast enough
-larger RBSs cannot follow small temporary demand changes

With a very small RBS, resources are not released fast enough to follow the changing bandwidth demand. Moreover, more requests are forwarded with higher RBS values, because releases follow demand closer. Nevertheless, large RBS values do not allow to follow small demand changes.

This decreases the number of forwarded requests to lower levels again beyond a certain RBS value.

This is justifiable if we consider that while the RBS is increased (but still gets small values), a greater amount of resources is released concluding in a higher level of utilization. Moreover, since the release resources procedure is more frequently performed, a smaller amount of resources is retained for accommodating future requests resulting in a higher signaling overhead. Nevertheless, as the RBS raises up to really great values, the possibility of accumulating that amount of free resources declines. This significantly impacts the utilization level, particularly under low load conditions. Accordingly, it is anticipated that the signaling overhead will be reduced.

Concluding, it is obvious from Table 3 and 4 that the algorithm's performance is much more sensitive to the RP than to the RBS parameter. Moreover, the performance of the quiet grafting mechanisms is considerably improved under heavy load conditions.

4. Applicability of Results

The applicability of the presented simulation results and conclusions in a real network are strongly dependent on the assumptions that have been made for the simulations and the degree of their resemblance to a real network topology.

The pattern of the sink tree formation chosen for the simulations is closely following the concept of sink tree formation in real topologies. As aforementioned, a pair of nodes in the sink tree represents an AS. The results are depicting that while the number of domains forming a sink tree is increasing, a greater control message reduction is succeeded, keeping though the same percentage of populated nodes. Thus, even if in real networks the sink tree is formed by more ASs, the expected results are at least as satisfactory as the ones presented.

Moreover, the number of "children" of each node, which reflects the number of ASs that are interconnected through a given AS, has been assumed that is equal in all levels. In real networks this figure is different, since a more random model may apply. In any case, the increase of this number produces an increase on the probability of a

node to become "populated" resulting in the inflation of the Quiet
Grafting effectiveness.

It is also worth mentioning that the increase in the injected traffic in
the examined network, results in a significant performance improvement
of the quiet grafting mechanism. This can be more easily observed in
real networks in the transit domains, where a great number of resource
requests are accumulated. Therefore, the need for CPU and memory for
each router that runs the reservation protocol is minimized.

5. Acknowledgements

6. Author Information

Eugenia Nikolouzou, NTUA
NTUA – National Technical University of Athens
Heroon Polytechniou 9, 157 73, Athens GREECE
e-mail: enik@telecom.ntua.gr

Petros D. Sampatakos, NTUA
e-mail: psampa@telecom.ntua.gr

Lila Dimopoulou
e-mail: lila@telecom.ntua.gr

Iakovos S. Venieris, NTUA
e-mail: ivenieri@cc.ece.ntua.gr

Martin Winter,
Siemens AG, ICN WN CC EK A19
81359 Munich – Germany
e-mail: martin.winter@icn.siemens.de

Bert F. Koch, Siemens AG
e-mail: bert.koch@icn.siemens.de

Thomas Engel,
Siemens AG, CT IC 2

Munich – Germany
e-mail: Thomas.Ex.Engel@mchp.siemens.de

Stefano Salsano
CoRiTeL – Consorzio di Ricerca sulle Telecomunicazioni
Via Anagnina, 203
00040 Morena Roma – ITALY
e-mail: salsano@coritel.it

Vincenzo Genova, CoRiTeL
e-mail: genova@coritel.it

Fabio Ricciato, CoRiTeL
e-mail: ricciato@coritel.it

Gerald Eichler, T-Systems Nova GmbH
e-mail: Gerald.Eichler@t-systems.com

7. References

[1]   S. Salsano, V. Genova, F. Ricciato, M. Winter, B. F. Koch,
L. Dimopoulou, E. Nikolouzou, P. Sampatakos, I.S. Venieris,
G. Eichler, "Inter-domain QoS Signaling: the BGRP Plus
Architecture", draft-salsano-bgrpp-arch-00.txt, Internet Draft,
May 2002.

[2]   P. Pan, E. Hahne, H. Schulzrinne: "BGRP: Sink-Tree-Based
Aggregation for Inter-Domain Reservations", Journal of
Communications and Networks, Vol. 2, No. 2, June 2000, pp.
157-167, http://www.cs.columbia.edu/~pingpan/papers/bgrp.pdf.