Providing QoS in IP Micromobility Networks

Gábor Zsolt Bilek, István Dudás
Sándor Szabó, dr. Sándor Imre*
Budapest University of Technology and Economics
Department of Telecommunications
Mobile Communications and Computing Laboratory
Magyar Tudósok krt.2, H-1117 Budapest, HUNGARY
E-mail: {bilekg, dudasi}@mcl.hu

***Abstract*-In this document our aim is to give a new integrated solution that provides QoS in the micromobility area. The provisioning of QoS has four main components that are in the scope of this paper. These are the followings: a Call Admission Control (CAC) algorithm, a method to map the users QoS demand with the help of Service Level Specification (SLS), a procedure to shorten the time needed for a handover by unified signalling and IP packet header compression.**

I. Introduction

The recent years have proved that the growing number of mobile users raised demand on using real-time applications. The guaranteed serving of these demands is not correctly possible over the currently employed IP protocols (IPv4, IPv6) due to their best effort model. This problem gets more complex if we consider the tasks added by mobility criteria. These tasks are mainly the parts of the handover process. Handover happens when a mobile host lefts a cell and goes to another. In this case, the mobile needs to get a new address to use in the new cell, policy control and call admission control must be executed, not to mention the demanded service class for which resources should be reserved. First of all, these tasks must be carried out in a short time not to result in an interruption for the call, which can be intolerable for a real-time application. Moreover, these tasks should be efficient enough to avoid congestion.

Providing QoS is based on resource management. An appropriate solution for this problem can be the Bandwidth Broker architecture. This architecture presumes the DiffServ (Differentiated Service) structure [3, 5]. This model handles the flows in groups, in contrast with the IntServ (Integrated Service) model, where each flow is handled separately, according to its special demands. The intricacy of this model comes to the front mainly in a backbone network, where thousands of flows should be managed, which cannot be expected from a router. Therefore, the DiffServ model, as a solution, decreases the complexity in the backbone with extruding the per flow management to the end points of the network. Table 1 shows a short comparison of the two models. The mentioned DiffServ flow groups form the traffic classes. A Bandwidth Broker [2] is responsible for a micromobility area, executes call admission control for the cells, handles the SLS of users and the resources of the network with the help of its database.

TABLE I

Comparison of IntServ and DiffServ

	IntServ	**DiffServ**
Complex routers	Every router	End routers
QoS management	Per flow	With classes
Scalability	Low	High
QoS resolution	Small scale	Large scale
Costs	Expensive	Cheap

This paper is organised as follows. At first, we introduce a call admission control algorithm that consists of four conditions, uniting the advantages of existing methods. Then we show a technique how SLS can be applied. Next, utilizing these elements, we introduce a unified handover message and handover process. At last we show how IP packet header compression can be utilized in micromobility environment.

II. Call Admission Control

Admission Problems

Call Admission Control (CAC) algorithms play significant role in the quality that a mobile network can provide. Their main task is to set and maintain the balance between reaching the highest utilization possible and to guarantee priority for the handoff calls arriving to the cell when needed. Therefore when a mobile host indicates a new call in the cell, a decision must be taken whether to allow it to start the call, or the system needs the unused capacity to leave for a possible handoff call, coming from one of the neighbouring cells. Specifically if the algorithm lets the new call enter, there is a chance that the handoff call will be interrupted, which means large degradation in terms of quality.

Lots of existing approaches can be found to solve admission questions. There are models using “guard channels”, reserving a predetermined piece of bandwidth for handoff calls, which cannot be allocated for new calls. Special algorithms can change the size of this guard channel adaptively, according to the traffic in the cell [1, 7]. Other methods take the state of neighbouring or other farther cells into consideration. For example, there are algorithms that calculate the average usage of an area, possibly with the use

* This project is supported by ETIK, OTKA F042590 and COST 279

of weighting [7]. Finally there are algorithms that estimate the future state of a cell by calculating and interpolating the moving of each mobile in it, using handoff probability matrices and residence time probability density function matrices [6].

It can be stated that the algorithm needs to estimate future traffic to reach higher efficiency. This can be done by concluding it from previous traffic. In an ideal case, the algorithm should know about the past of each mobile host as much as possible, which is a rather complicated exercise in itself. Instead most CAC algorithms try to follow the aggregated resource demand and its variation. Applying the Bandwidth Broker architecture which was mentioned earlier, the elements required for estimations like the capacity of the cells, the current usage, the current demands, etc. are available. Therefore the Broker will take the decision for its local scenario centrally, but from the aspect of the whole network, the task is distributed to micromobility areas.

The CAC Algorithm

The algorithm examines admission by a four-step condition sequence. The sequence starts from the simplest condition, continuing with more complex and comprehensive conditions. Therefore, in a very congestive situation, the rejection can be done soon, without checking all the conditions.

In the algorithm C_i denotes the capacity of cell i, W_i denotes the capacity used in cell i, and can be calculated as follows:

$$W_i = \sum_J N_{ij} \cdot w_j \quad (1)$$

where J is the set of traffic classes, N_{ij} is the number of hosts using the traffic class j in cell i, and w_j is the bandwidth demand of traffic class j. Then w_{new} means bandwidth demanded by the new call, and W_{end} marks bandwidth that will probably be released in τ time by finished calls. Finally $H_a[b]$ is the handoff vector, that shows the probability of directions how user demands go on from cell a to cell b. The four phases are the followings:

1. The simplest condition for admission (2): if there isn't any free bandwidth, rejection is obvious.

$$W_i + w_{new} < C_i \quad (2)$$

2. The second condition (3) examines a future state (τ time later), showing what will probably happen if the new call is admitted. We consider the calls (in their ratio to the whole traffic) coming from a neighbouring cell and the new call with positive sign, and calls that handoff to the neighbourhood or finish with a negative sign. α helps the adjustment of the system.

$$W_i - \sum_K H_i[k] \cdot W_i - W_{end} + \sum_K H_k[i] \cdot W_k + w_{new} < \alpha \cdot C_i \quad (3)$$

3. The third condition (4) examines the impression of the new call to the neighbouring cells k, which has neighbours l. Like in (3), we consider the calls coming from and going to the neighbouring cells. This condition should be checked for those neighbouring cells, which have the highest probabilities that the new call goes on to.

$$W_k + H_i[k] \cdot (W_i + w_{new}) - \sum_L H_k[l] \cdot W_k - W_{end} < \beta \cdot C_k \quad (4)$$

4. Finally, an „overall" condition (5) which examines the average usage of the micromobility area, and where S is the set of the cells of the micromobility area.

$$\frac{1}{|S|} \sum_s W_s < \delta \cdot C \quad (5)$$

It must be analysed what kind of operations the Bandwidth Broker needs to make on what types of data to execute this algorithm. The Bandwidth Broker stores and continuously refreshes the bandwidth in use for each cell. So there is only $H_a[b]$ and W_{end} left to discuss. These values can be set adaptively, that is with a chosen method based on past traffic. For example, we should compose the average of the last values and the values calculated in the past τ time or we should do this with some kind of weighting.

On the right of the equations there are parameters α, β and γ. Normally, the value of them is in the interval from 0 to 1. The role of them is the refinement of the algorithm according to our expectations. A lower value reduces the probability of the case that a call is admitted by the algorithm without convenient conditions, while with a higher value (closer to 1), higher utilization can be reached, but only in a less unsteady environment. Another important question is to determine the value of τ for which it must be assured that a mobile host cannot get out of the sight of the cell (getting into a non-neighbouring cell) without refreshing the probability values. This can be solved by taking the speed limit of the mobiles into consideration (this speed limit is caused by physical attributes).

Using the limits mentioned above, we can give a simple formula (6) for the calculation of τ (r is the radius of the cell).

$$\tau = \frac{2r}{v_{\max}} \quad (6)$$

It can be stated that the algorithm works correctly in case of users moving slower than v_{max}, because a mobile going by the speed limit can get only $2r$ distance farther on during τ time. For example let the cell radius be r=100 meters, and take that the speed limit is v_{max}=50km/h, which is about 15m/s. In this case, we get $\tau \approx$13sec, which is enough in plenty for the algorithm to be executed.

This algorithm has some advantages in contrast to other methods mentioned earlier in this section. It contains the important elements of the main schemes in one algorithm, but

in a less complex way. Besides, the four step structure makes it possible to get a negative decision earlier, without checking all conditions.

III. Service Level Specification

One of the most significant problems of provisioning QoS is drafting quality demands. The user or the application can usually express the required quality demands on a high level of abstraction. Besides the drafting of QoS demands a more formal specification is needed thereby the network components can utilise it. The formalisation is done in multiple steps, therefore a layer hierarchy model is supposed (as shown in Figure 1). As in other layer models (e.g. ISO-OSI reference model) all of the services of the layer should be mapped on the services of the layers beneath.

On the top of the hierarchy the SLA [5] can be found, which is an informal description of the quality demands of the user. The definition of the SLA is usually done manually and depends on the service provider, therefore it is out of scope of this paper. The most important part of the hierarchy is the SLS. The SLS makes it possible to express the QoS parameters numerically such as bandwidth or delay. The next two layers of the hierarchy define the inter- and intranetwork packet forwarding. These methods are defined in the DiffServ specification therefore it is also out of scope of this document. The bottom layer is network equipment specific, it describes how the routers and other network equipment back the QoS provisioning.

The SLS message allows the users to draft its QoS demands. It is supposed that in the network there is a valid SLA between the user and the service provider, and these parameters are mapped to a SLS. This will be needed in the process of authentication. The demanded SLS messages then will be mapped to DSCP (Differentiated Service Code Point) used by the DiffServ framework, which specifies the duties in connection with packet forwarding.

The TOS (Type of Service) field of the IPv4 header and the Traffic Class field of the IPv6 header, that allows these protocols to handle different classes of traffic, are redefined by the DiffServ, and it is called DS field [4]. The first six bits of the DS field form the DSCP (Differentiated Service CodePoint). There are three bits to define the class of the packets, which means eight different classes can be used. Among the eight values zero is used for best effort forwarding; the next four values (1-4) mean the classes of relative QoS (AF – Assured Forwarding). Value five marks the class of absolute QoS (EF – Expedited/Express Forwarding). The last two values (6-7) are reserved for the messages of the routing protocol.

The classification of the different flows in a micromobility domain is a multi level process and the sorting is done by the Bandwidth Broker.

SLA – Service Level Agreement non-technical terms & conditions technical parameters {SLS}-set
SLS – Service Level Specification IP service traffic characteristics offered network QoS guarantees
PDB – Per Domain Behaviour network QoS capabilities DiffServ edge-to-edge aggregates
PHB – Per Hop Behaviour **TCB – Traffic Conditioning Block** generic router QoS capabilities DiffServ edge & core routers
Schedulers (e.g. WFQ, WTP) **Algorithmic Droppers (e.g. RED)** **Markers (e.g. SRTCM, TRTCM)** Implementation, vendor & product specific

Fig. 1. Hierarchy of QoS demand

In Table II the transfer rate is represented in the speed column, in this table only the high and low values are used, but additional refinement is possible. Further research shall be done to find out in case of TCP protocol is used whether the value of loss field can be disregarded, since TCP guarantees the retransmission of packets if needed or the classification of packets can help the TCP protocol to perform better (the number of retransmission decreases). If the UDP protocol is being used the packet loss sensitive flows should be classified as EF to guarantee the arrival of packets.

In Table II some examples are presented how the classification may look like. For example a VoIP call is classified as follows. A VoIP call is a real-time, but not loss sensitive connection, its transfer rate is high. As a result it should be classified as high AF (3-4) or EF (5).

Next, we specify the negotiation of SLS. First the mobile node assembles an SLS message that is forwarded to the network with the connection request. This message is captured by the Bandwidth Broker that decides whether the flow should be authorised or not. On one hand the request should be authenticated whether the user is qualified to have the requested QoS. To do so the SLA should be mapped previously to an SLS so that the Bandwidth Broker can compare the two SLS and decide whether to admit the flow or not. On the other hand the BB has to manage the Call Admission Control, it should estimate whether the QoS demands of the new call can be met besides the ongoing calls.

As the Bandwidth Broker has decided to let the new call in the network it has to assemble the reply that contains mapping of the SLS to the DSCP field. This reply message is forwarded to the mobile node. Before the mobile node receives the reply message the access router stores the value of the DSCP field for further identification.

In case of a handover the Bandwidth Broker sends the same massage to the new access router. The communication goes on as follows: in every IP header the DSCP field is filled with the values received from the Bandwidth Broker.

TABLE II

Traffic classes

Application example	Loss	Speed	Protoid	Classification	Real-time
web browsing	0	High/Low	TCP/UDP	Low AF	0
file download	1	High/Low	TCP	AF	0
VoIP	0	Low	TCP/UDP	High AF	1
video conference	0	High	TCP/UDP	High AF, EF	1
VOD	1	High	UDP	EF	1

Having received this message the access router checks whether the value of the flow's DSCP is smaller than the stored value. If it is smaller – this means the user does not employ the traffic class available for him – the access router forwards the packets, all the other routers only have to take a look at the DSCP field of the packet and forward it accordingly. If the packet with a greater DSCP value arrives to the router, the router has various possibilities. It can either throw away the packet or downgrade it into the best-effort class.

IV. Unified Handover Signalling

The problem of QoS provisioning has two main aspects form the point of view of CAC algorithms in micromobility domains: a new connection starts in the domain or a connection that already exists enters the domain as a consequence of a handover.

In case of a new connection the mobile node informs the Bandwidth Broker of what kind of service it needs. Sending the SLS that contains the parameters set up previously in the SLA, does this. By receiving the SLS the Bandwidth Broker starts the authentication, it verifies whether the mobile user is authorised to employ the requested service. The authentication is between the Bandwidth Broker and either the database of the service provider or the Home Agent. As the Broker is certain of the genuineness of the request the proposed CAC algorithm can calculate whether the claim could be fulfilled from the available resources. If the request can be fulfilled the Bandwidth Broker sends the mobile node the appropriate values which should be in the DSCP field of the IPv6 header. After this procedure the new connection can be established with the mobile node and the use of the application with the requested QoS parameters is possible.

In case of an existing connection when a handover is due to take place first it should be ascertained that the bandwidth demand of the application is still genuine. After the authorization is done the Bandwidth Broker responsible for the new cell has to check (with the help of the CAC algorithm) whether the claims set out by the ongoing application in the new cell could be met. At the beginning of the handover process it is assured that the Bandwidth Broker of the previous cell has an authentic copy of the user's SLS parameters. Therefore the mobile node does not need to retransmit its own QoS claim, as consequence the transmission will not burden the radio link unnecessarily. In our proposal in case of the handover goes hand in hand with the alteration of the Bandwidth Broker, and it is certain that the handover occurs, the oAR (Old Access Router) calls upon the Broker – as it has an authentic copy of the SLS – to deliver the copy of the SLS to the Broker managing the new cell. This scheme assures that the new Bandwidth Broker also will possess an authentic copy of the SLS after the handover, since the security system of the Bandwidth Brokers minimize the chance of hostile attacks and interference. When on the contrary handover does not go parallel with the alternation of the Broker, then the Bandwidth Broker forwards the value of the DSCP field to the new access router.

The access router should notify the Bandwidth Broker if a handover is on the way. This is done by a new message sent by the oAR to the Bandwidth Broker. As the Broker receives this message it can decide on the ground of the available data whether the SLS should be transferred to the new Bandwidth Broker or the DSCP field should be forwarded towards the nAR (New Access Router).

With the help of our proposal it is possible that the CAC algorithm is executed faster since there is no need for authentication. As a result the handover process is shortened and the continuous provisioning of QoS parameters for the real-time applications is possible.

In course of a handover our proposal can be very useful, since the SLS negotiation and the CAC algorithm can be executed parallel with the IPv6 address autoconfiguration process [8]. This means that the sequential execution of these tasks is replaced by parallel execution. As an effect while the address configuration goes on (stateless address autoconfiguration, DAD – Duplicate Address Detection; or stateful address autoconfiguration) the SLS authorization and CAC algorithm is executed. As the mobile node has obtained its nCoA (New Care of Address) all the parameters are available to carry on the ongoing connection in the new cell. In case of a handover it is very important because the performance of real-time applications can be improved since delays caused by the handover can be decreased.

V. Header Compression

In this part of the paper we focus on QoS issues from another point of view. When we examine the parameters of the quality of services used by mobile hosts, bandwidth is an important question. Due to the large costs of the straitened capacity of the radio channel, the usage of bandwidth should be efficient as possible. This can be reached by raising the back on this ratio.

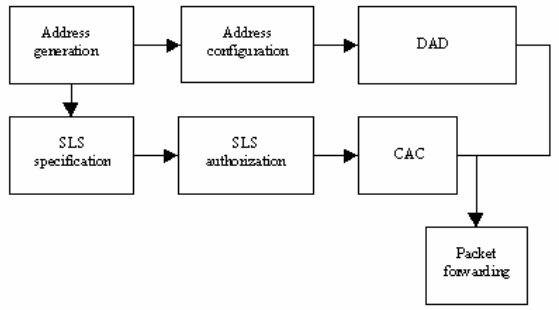

Fig. 2. The handover process

In IPv4 the packet header is not too long (minimum 20 bytes – without optional parts, maximum 60 bytes), but during a TCP session for example, it changes rarely and partly, so it is unnecessary to resend it all the time. And when we take a look at the same part of the header in case of IPv6, it is obvious that we need a method to solve this problem, because an IPv6 header can be 100 bytes long as well (minimum 40 bytes, there is no maximum) with its 16 byte source and the same destination addresses and other optional parts. Taking into consideration that an average radio channel has a BER of 10^{-3}, the probability of damaged packets will grow which also leads to lower efficiency.

The solution is header compression [9]. This scheme is known in wired systems. In this paper we examine the possibilities of its usage in micromobility networks. The principle of header compression is very simple, it uses the slow start mechanism. At the beginning of the session the whole header is sent, as time passes it happens less and less frequently. In case of the change of the header the process starts again. When the whole header is not sent, there is only a compressed header containing a CID (Context Identifier) which refers to the context of the session where the packet belongs to. This context is the last sent full header.

Our aim is to utilize this method in a mobile network. The difference is that users are mobile, they move from one cell to another. At a new cell, their address changes, causing that the header of the packets must be modified, so the slow start mechanism of header compression must restart. To avoid this, a little modification is needed on the algorithm. In our proposal the source address or the destination address shall not be compressed according to the direction of the communication. As a result in case of uplink communication the source address, in case of downlink communication the destination address will not be compressed rather it will be sent unchanged as the part of the compressed header.

Provided that IPv4 is being used mostly in today's networks our proposal increases the size of the compressed header by 4 bytes. However, if IPv6 is being used the size of the compressed header would increase by an additional 16 bytes which should be avoided. As a result not the complete address is sent without compression but only a part of it. Since the address of the mobile equipment is composed of a network prefix and a unique identifier of the mobile user, it is possible to send a part of the network prefix and the unique identifier uncompressed and to compress the rest of the network prefix. The size of the uncompressed part sent should be communicated to the partner at the beginning of the connection in the context so that the address of the mobile equipment can be composed at any time.

VI. Summary

In this paper a new QoS provisioning method has been proposed for IP micro-mobility architectures. In the first part in micromobility domain a novel Call Admission Control algorithm was introduced that adapts to mobile environment better. In the second part the QoS demand was mapped formally; as a result a new SLS message was introduced. Then the way of SLS negotiation was described and this negotiation process was combined with the address autoconfiguration. Finally we show how header compression methods can be used in mobile environment. As a consequence the provisioning QoS guarantees with a unified method in a mobile system relying on the micromobility architecture is possible.

References

[1] Yi Zhang, Derong Liu: „An Adaptive Algorithm for Call Admission Control in Wireless Networks*", IEEE Global Communications Conference*, San Antonio, Nov. 2001

[2] Günther Stattenberger, Torsten Braun: „QoS Provisioning for Mobile IP Users", *1st International Workshop on Services & Applications in the Wireless Public Infrastructure*, Evry (Paris), France, July 25-27, 2001

[3] D. Grossman: „New Terminology and Clarifications for DiffServ", IETF RFC 3260, April 2002

[4] K. Nichols, S. Blake, F. Baker, D. Black: „Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", IETF RFC 2474, December 1998

[5] Weibin Zhao, David Olshefski, Henning Schulzrinne: „Internet Quality of Service: an Overview", *Columbia Technical Report*, February 2000

[6] David A. Levine, Ian F. Akyildiz, Mahmoud Naghshineh: „A Resource Estimation and Call Admission Algorithm for Wireless Multimedia Networks Using the Shadow Cluster Concept", *IEEE/ACM Transactions on networking*, February, 1997

[7] Jon M. Peha, Arak Sutivong: „Admission Control Algorithms for Cellular Systems", *ACM/Baltzer Wireless Networks Volume 7, Number 2*, March 2001

[8] S. Thomson, T. Narten: „IPv6 Stateless Address Autoconfiguration", IETF RFC 2462, December 1998

[9] M. Degermark, B. Nordgren, S. Pink: „IP Header Compression", IETF RFC 2507, February 1999