Available online at www.sciencedirect.com
SCIENCE DIRECT
Computer Communications 28 (2005) 947–955
computer communications
www.elsevier.com/locate/comcom

An intelligent GGSN dispatching mechanism for UMTS

Shin-Ming Cheng[a], Phone Lin[a,*,1], Guan-Hua Tu[a], Li-Chen Fu[a], Ching-Feng Liang[b]

[a]*Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, ROC*
[b]*Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu, Taiwan, ROC*

Received 11 February 2004; revised 20 November 2004; accepted 23 November 2004
Available online 9 December 2004

Abstract

3GPP proposed the *Universal Mobile Telecommunication System* (UMTS) to provide high-speed wireless transmission services to mobile users. To efficiently route the packets, the PDP contexts are maintained in the UE, SGSN, and GGSN in the UMTS network. Before beginning a session, the PDP Context Activation procedure is exercised to create the PDP contexts, where the SGSN selects one GGSN to serve the UE based on an IP list (for all GGSNs that can serve the UE). If the selected GGSN does not accept the request due to exception events, the SGSN chooses another GGSN until the list is exhausted, which may cause unbalance-loaded GGSNs, redundant traffic load to the network, introduce delay for the PDP context activation procedure, and thus decrease QoS for the UMTS network. To resolve the above issues, this paper proposes an intelligent GGSN dispatch mechanism 'IGD' with different sorting algorithms. The proposed mechanism is considered practical and easily installed in the existing UMTS network. We also conduct simulation experiments to investigate the performance of the IGD mechanism.

Keywords: APN; GGSN; PDP context; UMTS

1. Introduction

Universal Mobile Telecommunication System (UMTS) [5,15,19] is one of the third generation mobile communication systems, which is evolved from *General Packet Radio Service* (GPRS) The UMTS provides wireless high-speed transmission services to mobile users. Fig. 1 illustrates the UMTS architecture. UMTS consists of the *UMTS Terrestrial Radio Access Network* (UTRAN) and the core network. In UMTS, a *user equipment* (UE) communicates with the UTRAN through the air interface Un [1]. The core network consists of two service domains, the *Circuit Switched* (CS) domain and the *Packet Switched* (PS) domain. On the core network, the UMTS release 5 [10] proposes the *IP Multimedia Subsystem* (IMS) for multimedia applications. The *Mobile Switching Center* (MSC) provides the CS services for mobile users, which is connected to the *Public Switched Telephone Network* (PSTN). The mobility databases HLR and VLR maintain the location information for mobile users. In the PS domain, the UE connects to the external *Packet Data Network* (PDN) through the *Serving GPRS Support Node* (SGSN) and the *Gateway GPRS Support Node* (GGSN). The SGSN delivers the packets between UEs and their counter-parts in the external PDN. The GGSN acts as a gateway between UMTS and external PDN, which is connected with SGSNs via an IP-based GPRS backbone network. In this paper, we focus on the PS domain. In the domain, four traffic classes are identified, which are *conversational*, *streaming*, *interactive*, and *background*. The traffics for the conversational or streaming class have the fairly constant and realtime characteristics, which require constant and high bandwidth transmission services. Typical applications for these two classes include VoIP and video streaming applications. The traffics for the interactive and background classes have the non-realtime characteristics, which can be served by

* Corresponding author. Address: Department of Computer Science and Information Engineering, National Taiwan University, No.1 Sec. 4 Roosevelt Rd., Taipei, Taiwan, ROC. Tel.: +886 2 23625336; fax: +886 2 23628167.
E-mail addresses: shimi@pcs.csie.ntu.edu.tw (S.-M. Cheng), plin@csie.ntu.edu.tw (P. Lin).
[1] P. Lin's work was supported in part by the National Science Council (NSC), R.O.C., under Contract NSC93-2213-E-002-095, in part by Computer and Communications Researches Lab/Industrial Technology Research Institute (CCL/ITRL), and in part by Microsoft Inc.

0140-3664/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2004.11.007

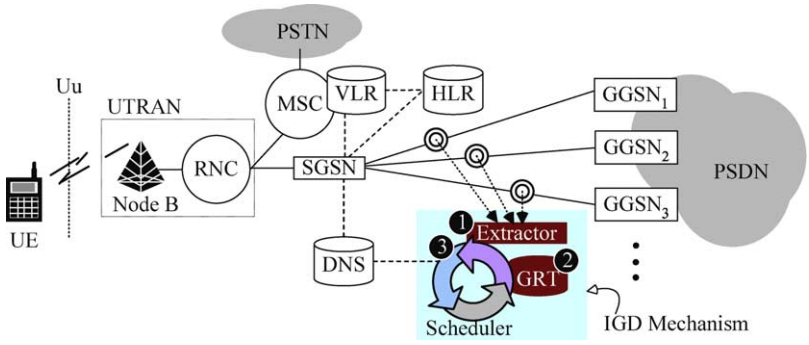

Fig. 1. Network architecture of UMTS.

the best-effort delivery services. Typical applications for these two classes are web-browsing and email applications.

To efficiently route the packets in the UMTS network, the SGSN, GGSN and UE maintain the *packet data protocol* (PDP) contexts (i.e. the routing information in UMTS). Before a UE starts a session to the external PDN, the PDP Context Activation procedure is invoked to establish the session from the UE to the server through SGSN and GGSN. At this moment, the PDP contexts are created in GGSN, SGSN, and UE for the session, respectively. According to the PDP contexts, the packets are routed through the UMTS to the external PDN. In the PS domain in UMTS Release 99, the PDP context activation procedure is invoked once for establishing the tunnel to deliver the packet data for a PS domain traffic. In the IMS in UMTS release 5 [10], the PDP context activation procedure is executed twice during the session establishment for a IMS session. One is for establishing the tunnel to deliver the *Signaling Initial Protocol* (SIP) [16] signaling message, and the other is for establishing the tunnel to deliver the data of the IMS-based session. If the QoS requirement for the activated session is changed, the PDP Context Modification procedure is exercised to modify the PDP contexts. The session is closed by executing the PDP Context Deactivation procedure. For the details of the three procedures, readers may refer to [5].

In the PDP context activation procedure, depending on the application invoked by the UE (e.g. web browsing, video streaming, email, or VoIP), the SGSN may select different GGSNs to serve it. In UMTS, each GGSN is associated with an *access point name* (APN[2]). The details of APN can be found in 3GPP TS 23.003 [4]. There may be several GGSNs serving for an APN. When the SGSN receives the APN sent from the UE, it uses the APN to query the *Domain Name Server* (DNS) to obtain a list containing the IP addresses of the GGSNs that can serve for it. Then the SGSN selects one of GGSNs in the list, and sends a request to it to establish a tunnel between SGSN and GGSN. If the selected GGSN does not accept the request due to some exception events (e.g. not enough memory or bandwidth resource), the SGSN chooses another GGSN until the list is exhausted. For each request of the tunnel establishment, signaling message exchange is required, which causes redundant traffic load to the network, and also introduces delay for the PDP context procedure. The worst case is that each GGSN in the list cannot serve the UE, and the SGSN knows such a condition until it tries all GGSNs in the list. Besides, the first GGSN in the GGSN IP list may be always fully loaded, but the last GGSN may be idle in most of time (i.e. load unbalancing). This may cause the heavily loaded GGSN is more likely to crash, and the QoS of the UMTS network decreases significantly.

For the above issue, we propose an intelligent GGSN dispatch mechanism called IGD to reduce the number of signaling messages exchanged and thus shortens the time for the PDP context activation. The IGD mechanism also accommodates the load balancing functionality for GGSNs. Our mechanism can be installed in the UMTS network with minor cost. We construct simulation experiments to evaluate the performance of the IGD mechanism. The rest of the paper is organized as follows. Section 2 describes the PDP context activation procedure. Section 3 details our proposed IGD mechanism. Section 4 investigates the performance of the IGD mechanism. Finally, Section 5 concludes this study.

2. The PDP context activation procedure

Before detailing the IGD mechanism, this section illustrates the PDP context activation procedure [5] in PS domain session in UMTS release 99. For the details of the procedure to activate the PDP context for the IMS-based session, readers may refer to Appendix A. We focus on how the APN information is exchanged through this procedure. Fig. 2 shows the message flow. Six steps are executed in this procedure.

Step 1. The UE sends an Activate_PDP_Context_Request message to the SGSN, which contains the APN information and QoS profile. The APN field is filled with a string, e.g. ‘ibm.com.mnc789.mcc88.gprs’, or left as a blank.

[2] The APN is the logic name of a GGSN and is used as a reference point name of external PDN that provides different kinds of application for the UE.

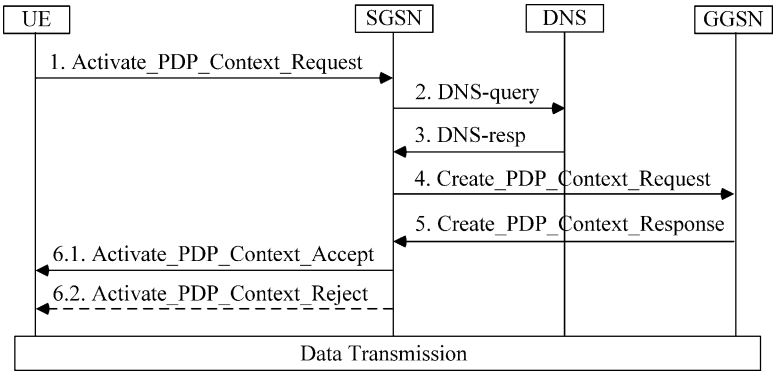

Fig. 2. The message flow for the PDP context activation procedure.

Step 2. Upon receipt of the Activate_PDP_Context_Request message, the SGSN checks the APN field. If it is blank, the SGSN uses the default APN (obtained from the HLR) to serve the UE. Then the SGSN uses the APN to query the DNS to get the IP address of the GGSN serving for this APN by sending a DNS-query message.

Note that the APN is either carried in the Create_PDP_Context_Request message (to be elaborated in Step 4) or is a default APN.

Step 3. In DNS, a table is maintained to keep a list of the GGSN IP addresses associated with different APN labels. When receiving a DNS-query message, the DNS looks up this table, and then returns a list containing all GSSN IP addresses associated with the APN.

Step 4. The SGSN checks the received GGSN IP list. If the list is not empty, then the SGSN selects the first GGSN in the list, and then sends a Create_PDP_Context_Request message to the selected GGSN to create a tunnel between the SGSN and GGSN. Otherwise (i.e. the list is empty, and no GGSN can serve for the UE), this procedure exits by performing Step 6.2.

Step 5. The GGSN decides whether it accepts or rejects the PDP context activation request based on some rules (e.g. whether the GGSN is overloading). If the GGSN accepts the request, a corresponding PDP context is created. The GGSN allocates the IP address for the UE and create a tunnel to the destinated external PDN. Otherwise (i.e. the GGSN rejects the request), the GGSN returns a Create_PDP_Context_Response message with a Cause value (that specifies the acceptance of the request or the reasons for rejecting the request) to the SGSN.

Step 6. When the SGSN receives the Create_PDP_Context_Response message, it checks the Cause value. Two cases are considered:

Case 1. If the Cause value is positive (i.e. the request has been accepted by the GGSN), the SGSN sends an Activate_PDP_Context_Accept message to the UE and starts to serve for the UE.

Case 2. If the Cause value is negative, the GGSN is removed from the GGSN IP list (obtained in Step 3). Then, SGSN tries the next GGSN in the list by performing Step 4. If the list is empty, the SGSN sends an Activate_PDP_Context_Reject message to the UE.

Note that in Steps 3 and 4, for the PDP context activation requests with the same APN, the DNS returns the same GGSN IP list. The first GGSN in the list may suffer from heavy loading. Furthermore, as the first GGSN cannot serve any connections, the SGSN will try more than one GGSN in this procedure. This leads to more signaling message exchanges for PDP context activation. To reduce the signaling traffic for the PDP context, in Section 3, we propose the IGD mechanism.

3. Intelligent GGSN dispatch mechanism

This section describes the Intelligent GGSN Dispatch (IGD) mechanism to reduce signaling cost for the PDP context activation procedure and to distribute the loads for GGSNs Note that the PDP Context Activation, PDP Context Deactivation, and the PDP Context Modification procedures may change the loading of a GGSN. Thus, the IGD mechanism monitors the PDU for these procedures. As shown in Fig. 1, in the IGD mechanism, we add three components in the existing UMTS network, which are the Extractor, GGSN Resource Table, and Scheduler. The functionalities of these components are illustrated as follows.

Extractor. See Fig. 1 (1). This component monitors the PDUs of the Create_PDP_Context_Response, Delete_PDP_Context_Response and Update_PDP_Context_Response messages sent from GGSN to SGSN to estimate the loading of each GGSN. To simplify our description,

Table 1
The usage of information elements in message **A**, **D**, and **U**

Information element	Message	Usage
Cause	**A**	Determine if a PDP context activation request has been granted or rejected by a specific GGSN
TEID control plane	**A**	Determine for which activated PDP context the D or U message is
Qos profile	**A**	Know how much QoS has been reserved for a granted session
Cause	**D**	Check if a PDP context has been deactivated successfully
Cause	**U**	Check if a PDP context has been modified successfully
Qos profile	**U**	Know how much QoS has been modified for a granted session

we let **A**, **D**, and **U** denote the Create_PDP_Context_Response message, the Delete_PDP_Context_Response message, and the Update_PDP_Context_Response message, respectively. For the details of the parameters in these three messages, readers may refer to [2]. Table 1 lists the parameters referenced in our mechanism.

Note that there are existing products for Extractor, e.g. HP E4250 ACCESS7. This solution can be easily deployed on the existing UMTS network without introducing any modification and thus with minor cost.

GGSN Resource Table. See Fig. 1 (2). This table stores the current usage of the resource in a GGSN, G_i. The details of this table are shown in Fig. 3. This table contains the GGSN IP address (see (a)), the network bandwidth (denoted as B_i) reserved by the GGSN for the activated PDP contexts (see (b)), and the number (denoted as N_i) of successfully activated PDP contexts in this GGSN (see (c)). We keep a list (named 'Tunnel List'; see (d)) to record the bandwidth reserved for each GTP tunnel created in the GGSN, where $b_{i,j}$ denotes the bandwidth reserved for the tunnel GT_j in the GGSN G_i.

Scheduler. See Fig. 1 (3). The Scheduler references the GGSN Resource Table to generate a GGSN IP list based on the GGSNs' loading for the DNS.

The mechanism works as follows. As the Extractor is powered on, it monitors the **A**, **D**, and **U** messages, and updates the GGSN Resource Table. Three cases are considered:

Case 1. The Extractor detects an **A** message. If the Cause value in this message is positive (i.e. a PDP context activation request has been accepted), the Extractor extracts three information carried in this message, which are the GGSN address, TEID, and the guaranteed bit rate in the QoS profile. The Extractor uses the GGSN address to find the corresponding GGSN Resource Table, adds one to the number N_i of activated PDP contexts, and increases the consumed bandwidth B_i by the guaranteed bit rate information. The Extractor creates an entry in the Tunnel List for the accepted PDP context request. The TEID GT_j and $b_{i,j}$ are filled with the GTP tunnel ID for the request and the guaranteed bit rate, respectively. Note the Extractor still works as usual for the IMS-based session establishment procedure since there are two **A** messages in the procedure.

Case 2. The Extractor detects a **D** message from GGSN to SGSN. If the Cause value in this message is positive (i.e. a PDP context has been deactivated in the GGSN), the Extractor uses the source IP address header in the message to identify the IP address of the GGSN that has deactivated the PDP context, and modify the corresponding GGSN Resource Table. The Extractor identifies which GTP tunnel has been closed by using the TEID information carried in the GTP header. The corresponding entry in the Tunnel List for the deactivated GTP tunnel is deleted, and N_i and B_i are decreased by one and $b_{i,j}$, respectively.

Case 3. The Extractor detects a **U** message from GGSN to SGSN. If the Cause value in this message is positive (i.e. a PDP context has been modified in the GGSN), the Extractor uses the source IP address header in the message to identify the IP address of the GGSN (that has modified the PDP context), and modify the corresponding GGSN Resource Table. The Extractor identifies which GTP tunnel has been closed by using the TEID information carried in the GTP header. Assume that the original $b_{i,j}$ value (i.e. the consumed bandwidth by the tunnel) in the Tunnel List is a Mbps. This value is updated to the new guaranteed bit rate, b Mbps, retrieved from QoS profile in

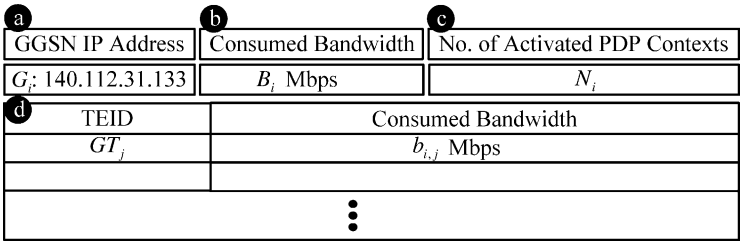

Fig. 3. The GGSN Resource Table.

the **U** message. Then the B_i value (i.e. the consumed bandwidth of the GGSN) is changed to $B_i - a + b$ Mbps. At this moment, the information for the GGSN loading has been changed.

Before the DNS responses, the DNS-resp message (containing GGSN IP list) to the SGSN (i.e. Step 3 in the PDP context activation procedure), the following two steps are performed.

Step I-1. When the DNS receives the DNS-query message (carrying the APN of the GGSN) sent from SGSN (i.e. Step 2 in the PDP context activation procedure), the DNS first checks the IP addresses of GGSNs serving this APN. Then the DNS sends the Reorder-req message (containing these IP address and APN) to the Scheduler to obtain a GGSN IP list where the IP addresses of the GGSNs are sorted based on GGSNs' loading.

Step I-2. Upon the receipt of the Reorder-req message, the Scheduler checks the consumed bandwidth and the number of activated PDP contexts in the GGSN Resource Table to determine the loading of each GGSN. Then according to the GGSNs' loading, the Scheduler sends the sorted GGSN IP list through Reorder-resp message to the DNS.

Note that in Step I-2, there may be different sorting policies. In our study, we propose the following two algorithms named IDG-B and IDG-W.

IGD-B. According to the APN, the scheduler determines the type of the traffic that may be generated from the UE. If the APN is for the best-effort traffic delivery (i.e. the interactive and background traffics), the scheduler sorts the GGSNs in an increasing order of the N_i value (i.e. the number of activated PDP contexts). If the APN is for the real time traffic delivery (i.e. the conversational and streaming traffics), the GGSNs are sorted in an increasing order of the B_i value (i.e. the consumed bandwidth of a GGSN).

IGD-W. This algorithm is similar to IDG-B except that the GGSNs are sorted in a decreasing order according to their N_i and B_i values.

4. Performance evaluation

In this paper, we conduct simulation experiments to compare the performances for the standard PDP context activation procedure and that with the IGD mechanism We adopt the event-driven based simulation technique, which is similar to that used in [17,18], and the details are not presented here. To simplify our description, we use PCAP to denote the standard PDP context activation procedure without the IGD mechanism, and IGD-B and IGD-W to denote the IGD mechanism with the sorting algorithms IGD-B and IGD-W, respectively. Let n be the number of the GGSNs in the UMTS network. In our simulation experiments, we set-up si GGSNs (i.e. $n=6$). These GGSNs are i.i.d. and labeled as G_1, G_2, G_3,…, G_6. Each GGSN has the capability to support the conversational application (e.g. VoIP), streaming application (e.g. video streaming), interactive application (e.g. web browsing), and background application (e.g. email). Assume that each GGSN can serve at most $N_{\max}$ activated PDP contexts. Let X be the number of activated PDP contexts when a PDP context activation request arrives at a GGSN. Define the utilization of the GGSN G_i as

$$U_{\mathrm{p},i} = \frac{E[X]}{N_{\max}} \tag{1}$$

To evaluate the fairness of the computation loading of each GGSN, we observe the sample standard deviation s_{p} of all $U_{\mathrm{p},i}$ values [14], which is calculated by

$$s_{\mathrm{p}} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(U_{\mathrm{p},i} - \frac{1}{n}\sum_{i=1}^{n}U_{\mathrm{p},i}\right)^2} \tag{2}$$

The s_{p} performance can be used to indicate the load balancing of the GGSNs. A smaller s_{p} value implies a system with better load balancing for the GGSNs. Let M be the number of signaling messages exchanged for a PDP context activation. In this paper, we investigate the expected value $E[M]$ of M and the s_{p} performance for the PCAP, IGD-B and IGD-W mechanisms.

The traffic model applied in our experiments is described as follows. The PDP context activation request arrivals for the conversational, streaming, interactive and background applications are Poisson arrivals with rates λ_{c}, λ_{s}, λ_{i}, and λ_{b}, respectively. The service times for the sessions of the conversational and streaming applications are exponentially distributed with means $1/\mu_{\mathrm{c}}$ and $1/\mu_{\mathrm{s}}$, respectively. We assume that the elapsed times for sessions of the interactive applications and background applications (e.g. WWW and email; which are typical Internet applications) form Pareto distributions with mean $1/\mu_{\mathrm{i}}$ and $1/\mu_{\mathrm{b}}$. The Pareto distribution is widely used to approximate the traffic pattern for the Internet application very well [11,12]. The Pareto distribution has the density function

$$f_{\mathrm{p}}(t) = \left(\frac{\beta}{l}\right)\left(\frac{l}{t}\right)^{\beta+1} \tag{3}$$

and the expected value

$$E[t] = \left(\frac{\beta}{\beta-1}\right)l \tag{4}$$

where β describes the 'heaviness' of the tail of the distribution. If β is between 1 and 2, then the variance for the distribution becomes infinity. Once a suitable value for β is selected to describe the traffic characteristics, then l can be determined by using (4). In this study, we select $\beta=1.21$ as that used in [13]. By substituting $\beta=1.21$ and $1/\mu_{\mathrm{i}}$ and $1/\mu_{\mathrm{b}}$ into (4), we have $l=(21/121\mu_{\mathrm{i}})$ and $(21/121\mu_{\mathrm{b}})$ for

Table 2
Traffic statistics for four kinds of applications

Application name	VoIP	Video	Web browsing	Email
Traffic class	Conversational	Streaming	Interactive	Background
Bandwith requested (kbps)	31	135	Best effort	Best effort
Arrival rate (1/s)	λ_c	λ_s	λ_i	λ_b
Average service time (s)	60	180	180	6

the interactive and the background applications, respectively.

In our study, we adopt the QoS requirements suggested in 3GPP 23.107 [6], 3GPP 26.234 [7], and 3GPP 26.236 [8] for the four kinds of applications. As shown in Table 2, we set $1/\mu_s = 1/\mu_i = 180$ s, $1/\mu_c = 1/(3\mu_i) = 60$ s, and $1/\mu_b = 1/(30\mu_i) = 6$ s. The bandwidth requirements for the conversational and the streaming applications are 31 and 135 kbps, respectively. The bandwidth requirements for the interactive and the background application are best-effort. We normalize the arrival rates λ_c, λ_s, λ_i and λ_b by μ_i, which have the following relationship

$$\lambda_c : \lambda_s : \lambda_i : \lambda_b = \alpha_c : \alpha_s : \alpha_i : \alpha_b \tag{5}$$

where α_c, α_s, α_i and α_b are constant numbers. The traffic load ρ_T for the usage of storage for the PDP contexts can be obtained by

$$\rho_T = \frac{\lambda_c}{\mu_c} + \frac{\lambda_s}{\mu_s} + \frac{\lambda_i}{\mu_i} + \frac{\lambda_b}{\mu_b} = \frac{10\lambda_c + 30\lambda_s + 30\lambda_i + \lambda_b}{30\mu_i} \tag{6}$$

In this study, we change the ρ_T value to investigate the performance of the PCAP, IGD-W, and IGD-B mechanisms. By substituting (5) into (6), we have the set-ups for λ_c, λ_s, λ_i and λ_b as follows

$$\lambda_c = \frac{\alpha_c 30\mu_i\rho_T}{10\alpha_c + 30\alpha_s + 30\alpha_i + \alpha_b};$$

$$\lambda_s = \frac{\alpha_s 30\mu_i\rho_T}{10\alpha_c + 30\alpha_s + 30\alpha_i + \alpha_b};$$

$$\lambda_i = \frac{\alpha_i 30\mu_i\rho_T}{10\alpha_c + 30\alpha_s + 30\alpha_i + \alpha_b};$$

$$\lambda_b = \frac{\alpha_b 30\mu_i\rho_T}{10\alpha_c + 30\alpha_s + 30\alpha_i + \alpha_b}.$$

Similar to [20], in our simulation, we consider four cases for the set-ups of the ratio among λ_c, λ_s, λ_i and λ_b: Case I: λ_c:λ_s:λ_i:λ_b = 1:1:1:1; Case II: λ_c:λ_s:λ_i:λ_b = 4:1:1:4; Case III: λ_c:λ_s:λ_i:λ_b = 1:4:4:1; Case IV: λ_c:λ_s:λ_i:λ_b = 1:1:4:4.

Comparison for the E[M] performances of PCAP, IGD-W, and IGD-B. Fig. 4 compares the average number of signaling messages $E[M]$ for the PCAP, IGD-W and IGD-B mechanisms by considering the four cases. It is obvious that IGD-B and IGD-W significantly outperform PCAP in terms of $E[M]$. With IGD-W and IGD-B, the DNS sends a sorted

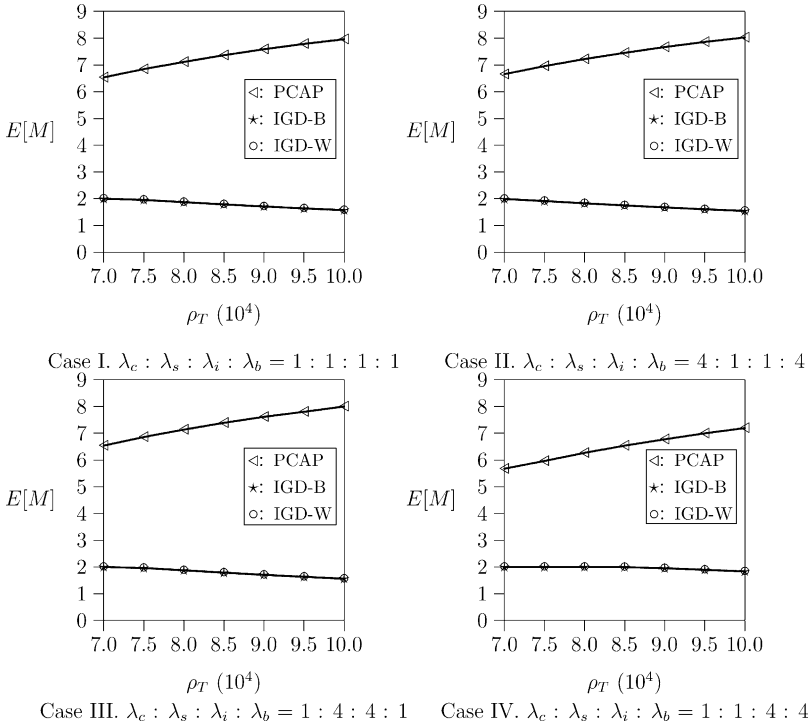

Fig. 4. Comparison for the $E[M]$ performances of PCAP, IGD-W, and IGD-B.

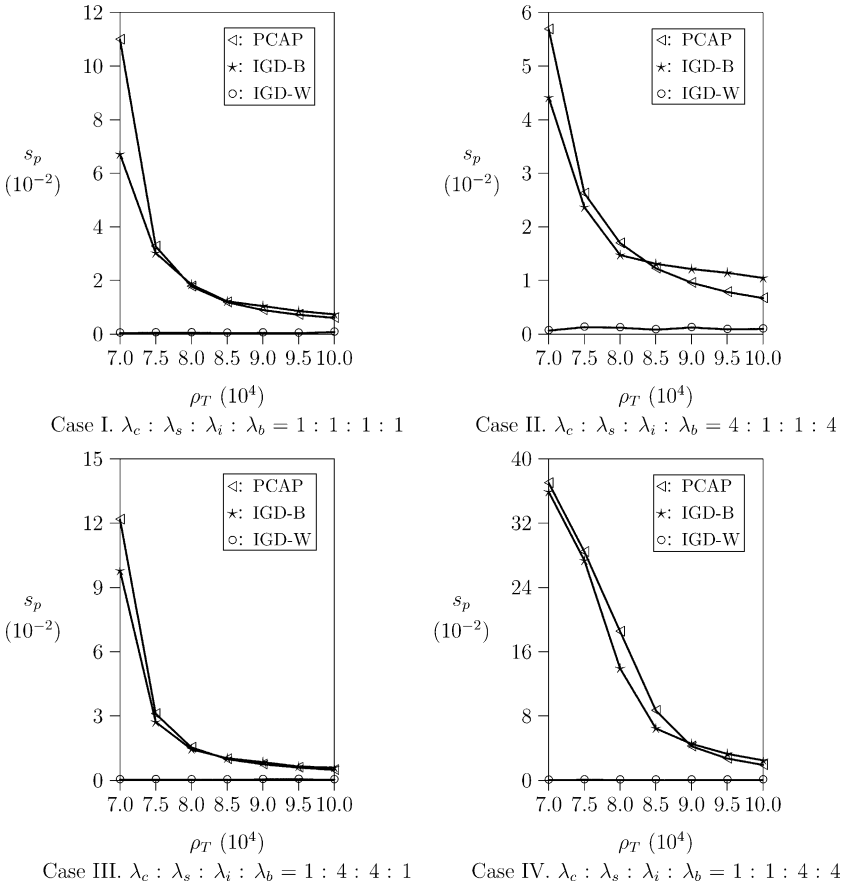

Fig. 5. Comparison for the s_p performances of PCAP, IGD-W and IGD-B.

GGSN list based on the GGSN loading, and at most two signaling messages (i.e. the Create_PDP_Context_Request and Create_PDP_Context_Response messages) are required for a PDP context activation request. As ρ_T increases, the $E[M]$ values for PACP increase slightly. On the other hand, the $E[M]$ values for IGD-W and IGD-B decrease slightly. In IGD-W and IGD-B, as all GGSNs cannot accommodate more PDP context activation requests due to no bandwidth or memory space available, the DNS will directly return SGSN an empty GGSN IP list. Then the SGSN will directly reject this activation request without sending any message to GGSN. To summarize, as system traffic load is larger, IGD-W and IGD-B perform better.

In this figure, we also observe that smaller signaling cost is shown in Case IV than that in Cases I–III. The acceptance for the PDP context activation requests for the conversational and streaming applications depends on the bandwidth and PDP context resources of the GGSN. For the interactive and background applications, whether the PDP context activation requests can be served depends only on the PDP context resources of GGSN. Thus, as the system has higher traffic load for the conversational and streaming applications, the GGSN are more likely overloaded, and larger signaling cost are observed. Since Cases I–III have the same traffic load for the conversational and the streaming applications, and higher than that in Case IV, we observe this phenomenon.

Comparison for the s_p performances of PCAP, IGD-W and IGD-B. Fig. 5 compares the s_p performances for the PCAP, IGD-W and IGD-B mechanisms by considering the four cases. As mentioned previously, a smaller s_p implies a more balancing load for GGSNs. It is obvious that for four cases, IGD-W has smaller s_p values than that of PCAP and IGD-B, which approximates to 0. This phenomenon reflects that the IGD-W mechanism does fairly distribute the traffic loading to different GGSNs. Furthermore, the s_p values for the IGD-W mechanism changes insignificantly as the total traffic load ρ_T increases. This is due to that with IGD-W, the SGSN assigns a GGSN (that has the largest free memory space for PDP contexts) to serve the PDP context activation request.

On the other hand, for the PCAP and IGD-B mechanisms, the s_p performances are almost the same and much larger than that of IGD-W. Without scheduling (i.e. PCAP), the SGSN always assigns the GGSN to the PDP context request following the same GGSN list, which causes unbalancing loads to GGSNs. With IGD-B, the SGSN allocates the GGSN (that has smallest free memory space) to serve the PDP context request. Therefore, an unbalancing load of GGSNs is incurred in IGD-B. As ρ_T increases, the s_p values of PCAP

and IGD-B decrease significantly. When ρ_T increases, the system is more likely overloaded (i.e. all GGSNs cannot accommodate any new PDP context request), and the loads for different GGSNs become the same.

Furthermore, in this figure, we observe that for the four traffic cases, the s_p trend for these mechanisms are similar. The reason is the same as that mentioned in the $E[M]$ performance comparison.

5. Conclusion

This paper proposed an intelligent GGSN dispatch mechanism with two sorting algorithms, named 'IGD-B' and 'IGD-W', respectively, to reduce traffic load for the PDP context activation procedure and balance the loading for the GGSNs in the UMTS network. In the IGD mechanism, a sorted GGSN IP list is generated from the scheduler and sent to the SGSN. By using the sorted GGSN list, the SGSN can choose the most suitable GGSN to serve the PDP context request without sending any unnecessary signaling messages. We also conducted simulation experiments to investigate the performances of the IGD-B and IGD-W mechanisms. In our simulation experiments, we apply to traffic model reported by 3GPP. Our study indicated that IGD-B and IGD-W can significantly reduce the number of signaling messages. Furthermore, the IGD-W performed very well for the load balancing for GGSNs.

Acknowledgements

The authors would like to thank the anonymous reviewer. His comments have significantly improved the quality of this paper.

Appendix A. The PDP context activation procedure for IMS-based traffic

This section describes the procedure for the PDP context activation procedure for IMS-based traffic. When a UE starts the packet transmission for an IMS-based session, an end-to-end QoS negotiation is required. The message exchanges for this negotiation is based on the SIP protocol. In the UMTS release 5 network, besides SGSN and GGSN, two elements, Call State Control Function (CSCF) and Policy Decision Function (PDF), are involved for the IMS-based session. The CSCF behaves like a SIP Proxy that accepts SIP messages and routes them to other CSCFs. The PDF retrieves QoS profile (from CSCF) and appropriate policy rules to make the decision whether the requested UE can be authorized for use of the resource. The message flow for establishing an IMS-based session is shown in Fig. A1, and its corresponding description is shown below.

Step 0. The UE executes the PDP context activation procedure to establish a tunnel to GGSN. This tunnel is used to deliver the SIP signaling for the IMS-based session establishment procedure [10].

Step 1. The UE sends a SIP invite message with SDP parameters to P-CSCF to request a IMS-based session establishment.

Steps 2 and 3. When P-CSCF receives the SIP invite message, it routes the message to the peer node along the SIP signaling path. The peer node responds P-CSCF a SIP 100 trying message. Upon receipt of this message, P-CSCF checks the negotiated SDP parameters

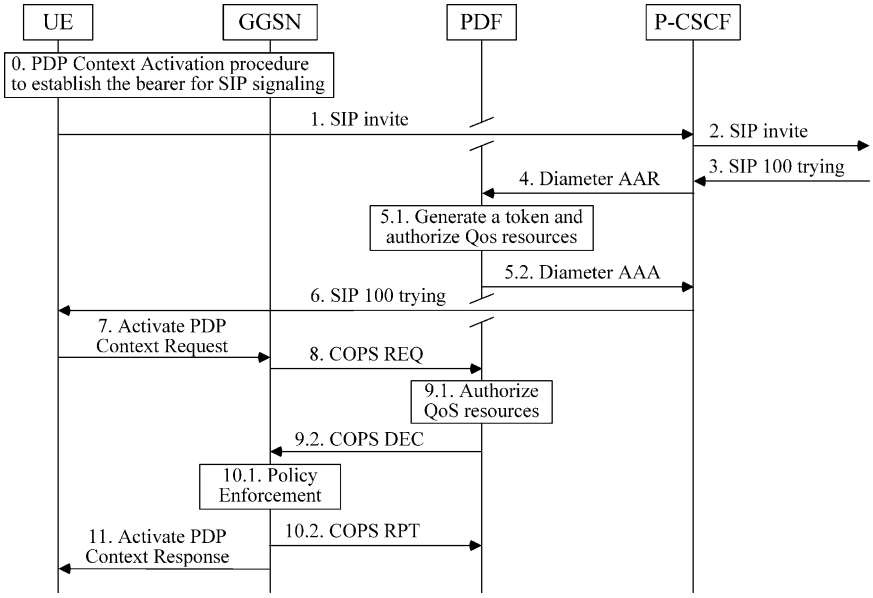

Fig. A1. The message flow for IMS-based session establishment procedure.

to determine the QoS for the session.

Step 4. The P-CSCF sends Diameter AAR message to the PDF for authorization of the session request.

Steps 5.1 and 5.2. Upon receipt of the Diameter AAR message, the PDF generates a token that is used to authorize the resource request. If the authorization is successful, the PDF replies the P-CSCF a Diameter AAA message. This token is also carried in Steps 6–8. Here, we consider the successful case. For the unsuccessful case, the readers may refer to [3,9].

Step 6. The P-CSCF replies the UE a SIP 100 trying message, where the authorization token is carried in this message.

Step 7. The UE sends an Activate_PDP_Context_Request message to the GGSN, which contains the authorization token and negotiated QoS profile. Note that the PDP context is for the tunnel establishment between the UE and GGSN, which is used to carry the IMS-based packets.

Step 8. Upon receipt of the Activate_PDP_Context_Request message, the GGSN sends a COPS_REQ message to the PDF for resource authorization.

Steps 9.1 and 9.2. Upon receipt of the token carried in the Activate_PDP_Context_Request message, the PDF authorizes the requested resource, and replies the GGSN a COPS DEC message to establish the tunnel between the UE and the GGSN.

Steps 10.1 and 10.2. Based on the received authorization information from the PDF, the GGSN enforces the PDF policy decision to reserve the resource for the tunnel. Then the GGSN responses a COPS_RPT message to the PDF to report the successful resource reservation.

Step 11. The GGSN accepts the PDP context activation request (at Step 7), and returns the corresponding Activate _PDP_Context_Response message to the UE.

References

[1] 3GPP, Third Generation Partnership Project, Radio interface protocol architecture, Technical Report Technical Specification 3G TS 25.301 version 5.2.0 (2002–2009), 2002.

[2] 3GPP, Third Generation Partnership Project, Technical Specification Group Core Net-work, General Packet Radio Service (GPRS), GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface, Technical Report Technical Specification 3G TS 29.060 version 6.1.0 (2003–2006), 2003.

[3] 3GPP, Third Generation Partnership Project, Technical Specification Group Core Net-work, General Packet Radio Service (GPRS), End-to-end Quality of Service (QoS) signalling flow, Technical Report Technical Specification 3G TS 29.208 version 6.1.0 (2004–2009), 2003.

[4] 3GPP, Third Generation Partnership Project, Technical Specification Group Core Net-work, Numbering, addressing and identification, Technical Report Technical Specification 3G TS 23.003 version 5.6.0 (2003–2006), 2003.

[5] 3GPP, Third Generation Partnership Project, Technical Specification Group Services and Systems Aspects, General Packet Radio Service (GPRS), Service Description, Stage 2, Technical Report Technical Specification 3G TS 23.060 version 6.1.0 (2003–2006), 2003.

[6] 3GPP, Third Generation Partnership Project, Technical Specification Group Services and System Aspects, Quality of Service (QoS) concept and architecture, Technical Report Technical Specification 3G TS 23.107 version 5.9.0 (2003–2006), 2003.

[7] 3GPP, Third Generation Partnership Project, Technical Specification Group Services and System Aspects, Transparent end-to-end Packet-switched Streaming Service (PSS), Protocols and codecs, Technical Report Technical Specification 3G TS 26.234 version 5.5.0 (2003–2006), 2003.

[8] 3GPP, Third Generation Partnership Project, Technical Specification Group Services and System Aspects, Packet switched conversational multimedia applications, Transport protocols, Technical Report Technical Specification 3G TS 26.236 version 5.3.0 (2003–2006), 2003.

[9] 3GPP, Third Generation Partnership Project, Technical Specification Group Services and System Aspects, End-to-end Quality of Service (QoS) concept and architecture, Technical Report Technical Specification 3G TS 23.207 version 5.8.0 (2003–2006), 2003.

[10] 3GPP, Third Generation Partnership Project, Technical Specification Group Services and System Aspects, IP Multimedia Subsystem (IMS), Technical Report Technical Specification 3G TS 23.228 version 6.7.0 (2004–2009), 2004.

[11] M. Cheng, L.-F. Chang, Wireless dynamic channel assignment performance under packet data traffic, IEEE Journal on Selected Areas in Communications 17 (7) (1999) 1257–1269.

[12] ETSI, Universal Mobile Telecommunications System (UMTS), Selection Procedures for the Choice of Radio Transmission Technologies of the UMTS, Version 3.2.0, Technical Report TR 101 112, ETSI, 1998.

[13] C.E. Fossa Jr., N.J. Davis IV, A dynamic code assignment algorithm for quality of service in 3G wireless networks, Proceedings of IEEE WCNC2002 1 (March) (2002) 1–6.

[14] R.V. Hogg, E.A. Tanis, Probability and Statistical Inference, sixth ed., Prentice Hall, Englewood Cliffs, NJ, 2001.

[15] H. Holma, A. Toskala, WCDMA for UMTS, second ed., Wiley, New York, 2000.

[16] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, RFC 3261-SIP: Session Initiation Protocol, June 2002.

[17] P. Lin, Channel allocation for GPRS with buffering mechansims, ACM/Baltzer Wire-less Networks 9 (5) (2003) 431–441.

[18] P. Lin, Y.-B. Lin, Channel allocation for GPRS, IEEE Transactions on Vehicular Technology 50 (2) (2001) 375–387.

[19] Y.-B. Lin, I. Chlamtac, Wireless and Mobile Network Architectures, Wiley, New York, 2001.

[20] T. Minn, K.-Y. Siu, Dynamic assignment of orthogonal variable-spreading-factor codes in W-CDMA, IEEE Journal on Selected Areas in Communications 18 (8) (2000) 1429–1439.