

**DESCRIPTION****Title**

End-to-end Session Initiation Protocol overload control in IP Multimedia subsystem

**Field of the invention**

The present invention generally relates to wired and wireless communication networks, and more specifically relates to a method, apparatus and computer program product for enabling end-to-end Session Initiation Protocol SIP overload control in IP Multimedia subsystem IMS.

**Background**

Mobile data transmission and data services are constantly making progress, wherein such services provide various communication services, such as voice, video, packet data, messaging, broadcast, etc. In recent years, Long Term Evolution LTE™ has been specified, which uses the Evolved Universal Terrestrial Radio Access Network E-UTRAN as radio communication architecture according to 3GPP specification.

Since LTE™ is basically designed as pure packet switched system, solutions have been proposed so as to also enable legacy circuit switched services in E-UTRAN environment, such as voice calls and short message service.

In particular, in contrast to legacy cellular telecommunications standards including GSM, LTE does not provide dedicated channels for circuit switched telephony. Instead LTE is an all-IP system providing an end-to-end IP connection from the mobile equipment to the core network and out again. Therefore, in order to provide voice connections over standard LTE bearers, a specific form of Voice over IP may be used.

Voice over LTE is currently rolled out across the globe, and it is based on Internet Protocol Multimedia Subsystem IMS Session Initiation Protocol SIP call flows. First deployments show

that robust and efficient overload control logic is required to guarantee a stable system and a recovery from overload even after complete site failures.

Hence, in view of the above drawbacks, there is a need for improving an overload control logic in an IP Multimedia subsystem.

### **Summary of the Invention**

Therefore, in order to overcome the drawbacks of the prior art, it is an object underlying the present invention to provide an end-to-end SIP overload control in IMS.

In particular, it is an object of the present invention to provide a method, apparatus and computer program product for enabling an end-to-end Session Initiation Protocol overload control in an IP Multimedia subsystem.

According to a first aspect of the present invention, there is provided a method, comprising determining usage of a resource at session setup with a signaling protocol, counting the determined used resources during the session setup, increasing a value of a progress counter for overload control based on the counted used resources, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging, and prohibiting, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value.

According to a second aspect of the present invention, there is provided an apparatus, comprising a determination unit configured to determine usage of a resource at session setup with a signaling protocol, a counting unit configured to count the determined used resources during the session setup, a processing unit configured to increase a value of a progress counter for overload control based on the counted used resources, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging, and a prohibiting unit configured to prohibit, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value.

According to a third aspect of the present invention, there is provided a computer program product comprising computer-executable components which, when the program is run, are configured to carry out the method according to the first aspect.

Advantageous further developments or modifications of the aforementioned exemplary aspects of the present invention are set out in the dependent claims.

According to certain embodiments of the present invention, the threshold is set dependent on an overload level of the occurring overload.

Further, according to certain embodiments of the present invention, usage of resources of a single network element is counted.

Further, according to certain embodiments of the present invention, usage of resources in the network is counted, and the progress counter is increased by an application server.

Further, according to certain embodiments of the present invention, in case an application server does not support the progress counter, the received progress counter is sent transparently through the application server, whereas the originating and the terminating Call Session Control Functions add their resource use.

Further, according to certain embodiments of the present invention, the progress counter is increased by one by each entity handling the session setup request.

According to certain embodiments of the present invention, the progress counter is increased by each Session Initiating Protocol resource with a value dependent on the logic it applies.

Further, according to certain embodiments of the present invention, the value a resource adds depends on its overload level.

Further, according to certain embodiments of the present invention, the progress counter is used and added by network elements only in case of an overload.

Further, according to certain embodiments of the present invention, the progress counter is a dedicated Session Initiating Protocol header field and/or carried as an Uniform Resource Identifier parameter in existing header fields.

Further, according to certain embodiments of the present invention, the Session Initiation Protocol messaging method is one of 'INVITE', 'SUBSCRIBE', 'UPDATE' and 'PRACK'.

Still further, according to certain embodiments of the present invention, the progress counter is removed when the Session Initiation Protocol leaves a validity area.

### **Brief description of drawings**

— For a more complete understanding of example embodiments of the present invention, reference is now made to the following descriptions taken in connection with the accompanying drawings in which:

Fig. 1 schematically shows a general Voice over LTE architecture as a non-limiting use case of the present invention;

— Fig. 2 illustrates a method according to certain embodiments of the invention;

Fig. 3 depicts a general structure of an apparatus according to certain embodiments of the invention;

Fig. 4 shows a basic call flow according to certain embodiments of the invention;

Fig. 5 shows a basic call flow according to further certain embodiments of the invention;

Fig. 6 shows a basic call flow of a variant of Fig. 4 according to further certain embodiments of the invention;

Fig. 7 shows a further basic call flow of a variant of Fig. 4 according to certain embodiments of the invention;

Fig. 8 shows a rejection logic based on pcoc and Overload level with respect to the situation of Fig. 6.

**Description of exemplary embodiments**

Exemplary aspects of the present invention will be described herein below. More specifically, exemplary aspects of the present invention are described hereinafter with reference to particular non-limiting examples and to what are presently considered to be conceivable embodiments of the present invention. A person skilled in the art will appreciate that the invention is by no means limited to these examples, and may be more broadly applied.

It is to be noted that the following description of the present invention and its embodiments mainly refers to specifications being used as non-limiting examples for certain exemplary network configurations and deployments. Namely, the present invention and its embodiments are mainly described in relation to 3GPP specifications being used as non-limiting examples for certain exemplary network configurations and deployments. As such, the description of exemplary embodiments given herein specifically refers to terminology which is directly related thereto. Such terminology is only used in the context of the presented non-limiting examples, and does naturally not limit the invention in any way. Rather, any other network configuration or system deployment, etc. may also be utilized as long as compliant with the features described herein.

Some example versions of the disclosure and embodiments are described with reference to the drawings. In the following, different exemplifying examples will be described using, as an example of a communication network, a cellular wireless communication network, such as an LTE or LTE-Advanced based system. However, it is to be noted that the present invention is not limited to an application using such types of communication system, but is also applicable in other types of communication systems, be it wireless systems, wired systems or systems using a combination thereof.

Hereinafter, various embodiments and implementations of the present invention and its aspects or embodiments are described using several alternatives. It is generally noted that, according to certain needs and constraints, all of the described alternatives may be provided alone or in any conceivable combination (also including combinations of individual features of the various alternatives).

In particular, the following examples versions and embodiments are to be understood only as illustrative examples. Although the specification may refer to "an", "one", or "some" example version(s) or embodiment(s) in several locations, this does not necessarily mean that each such reference is to the same example version(s) or embodiment(s), or that the feature only applies to a single example version or embodiment. Single features of different embodiments may also be combined to provide other embodiments. Furthermore, words "comprising" and "including" should be understood as not limiting the described embodiments to consist of only those features that have been mentioned and such example versions and embodiments may also contain also features, structures, units, modules etc. that have not been specifically mentioned.

In general, a telecommunication network comprises plural network elements, such as base stations BS, evolved NodeB's (eNB; i.e. base station in LTE environment), user equipments UE (e.g. mobile phone, smart phone, Computer, etc.), controllers, interfaces, etc, and in particular any equipment used in the provision of a telecommunications service.

A basic system architecture of a communication system where example versions and embodiments are applicable may comprise a commonly known architecture of one or more communication networks comprising a wired or wireless access network subsystem and a core network. Such an architecture may comprise one or more communication network control elements, access network elements, radio access network elements, access service network gateways or base transceiver stations, such as a base station, an access point or an eNB, which control a respective coverage area or cell (macro cell, small cell) and with which one or more communication elements or terminal devices such as a UE or another device having a similar function, such as a modem chipset, a chip, a module etc., which can also be part of a UE or attached as a separate element to a UE, or the like, are capable to communicate via one or more channels for transmitting several types of data. With most relevance for the present invention, core network elements such as Call State Control Functions (CSCFs), Application Servers and gateway network elements, policy and charging control network elements, mobility management entities, operation and maintenance elements, and the like may be comprised.

The general functions and interconnections of the described elements, which also depend on the actual network type, are known to those skilled in the art and described in corresponding specifications, so that a detailed description thereof is omitted herein. However, it is to be

noted that several additional network elements and signaling links may be employed for a communication to or from a base station and a communication network besides those described in detail herein below.

Fig. 1 shows a general Voice over LTE VoLTE architecture as an example use case in which the present invention may be applied. However, it is to be noted that the present invention is not limited thereto, but may be applied to other cases of signaling protocols in which access to independent IMS is performed, and can be used for all types of VoIP networks.

According to the exemplary use case illustrated in Fig. 1, user equipments UE with VoLTE-ability are each connected to a base station eNB (evolved NodeB), and the base station in turn is controlled by a core network/IMS.

Basically, VoIP, such as VoLTE as one use case, is based on IP Multimedia subsystem IMS Session Initiation Protocol SIP call flows, wherein such IMS SIP call flows for VoLTE are characterized by the following properties.

Firstly, the same network element NE may handle the same call in different roles, e.g. as Proxy Call Session Control Function P-CSCF and Serving Call Session Control Function S-CSCF. Secondly, the same S-CSCF may be in the call path several times, before and after invocation of one or more Application Servers AS at the IMS Service Control ISC interface. As examples, Telephony Application Server TAS, Service Centralization and Continuity Application Server SCC-AS for VoLTE may be invoked together or independently. Thirdly, the call handling for originating and terminating side may be independent, i.e. the same network element and same role may be in the call path both on originating and terminating side.

As an example, an end-to-end call flow with a single AS only and both sides served by same Network Element NE in all roles may be considered. Thereby, a call will run through the call control logic in the network element NE in originating P-CSCF, originating S-CSCF twice, terminating Interrogating Call Session Control Function I-CSCF, terminating S-CSCF twice and terminating P-CSCF, which is in total 7 times.

However, if the network element NE or the whole network is in overload, usually quota based rejection mechanisms may be applied, e.g. every second message is rejected. In the

example above this mechanism could be applied in total seven times during the call flow, which means that only one out of 128 (= 2 to the order of 7) end-to-end calls will be able to traverse a CSCF rejecting "every second message". As the other 127 will mostly try again and add additional load to the node and network, this mechanism is inefficient.

However, there is no standard and default call flow in the IMS by now. For example, click-to-dial calls may be initiated by the Application Server AS. Also, CSCF roles may not always be located on the same node.

Therefore, according to certain embodiments of the present invention, the basic issue is to prioritize messages, which are more "advanced" in the call flow.

This means that, according to certain embodiments of the invention, the more system resources were already spent on one call set-up attempt, the less likely it should be rejected because of overload. As an essential component of the present invention, the "resources already spent" are counted or accumulated during the call set-up, and the value is carried in SIP signaling.

Thereby, the counter may also be referred to as a "progress counter", which is also abbreviated as "pcoc" (progress counter for overload control) in the present application.

Messages with a progress counter higher than a certain threshold, dependent on the overload level, may not be rejected or will be less likely to be rejected.

Fig. 2 shows a method according to some example versions of the disclosure, which may be performed by a network element included in e.g. a core network of a cell-based communication network.

In Step S21, The usage of a resource at session setup with a signaling protocol is determined.

Then, in Step S22, the determined used resources during the session setup is counted.



Further, in Step S23, a value of a progress counter for overload control based on the counted used resources is increased, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging.

Still further, in Step 24, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value is prohibited.

In Fig. 3, a diagram illustrating a configuration of an element comprised in a (tele-) communication network element of a communication network for VoIP according to some example versions of the disclosure is shown, which is configured to implement end-to-end SIP overload control in IMS described in connection with some of the example versions of the disclosure. The embodiment may be carried out in or by a network element. It is to be noted that the network element may comprise elements or functions, such as a chipset, a chip, a module etc., which can also be part of a network element or attached as a separate element to a network element, or the like. It should be understood that each block and any combination thereof may be implemented by various means or their combinations, such as hardware, software, firmware, one or more processors and/or circuitry.

The network element 30 shown in Fig. 3 may comprise a processing function, control unit or processor 31, such as a CPU or the like, which is suitable for executing instructions given by programs or the like related to the network element control procedure.

The processor 31 is configured to execute processing related to the above described end-to-end SIP overload control in IMS. In particular, the processor 31 comprises a sub-portion 310 as a determination unit configured to determine usage of a resource at session setup with a signaling protocol. The portion 310 may be configured to perform processing according to S21 of Fig. 2. Furthermore, the processor 31 comprises a sub-portion 311 usable as a counting unit configured to count the determined used resources during the session setup. The portion 311 may be configured to perform processing according to S22 of Fig. 2. Furthermore, the processor 31 comprises a sub-portion 312 usable as a processing unit configured to increase a value of a progress counter for overload control based on the counted used resources, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging. The portion 312 may be configured to perform processing according to S23 of Fig. 2. Furthermore, the processor 31 comprises a sub-

portion 313 usable as a prohibiting unit configured to prohibit, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value. The portion 313 may be configured to perform processing according to S24 of Fig. 2.

Reference signs 32 and 33 denote transceiver or input/output (I/O) units (interfaces) connected to the processor 31. The I/O units 32 may be used for communicating with the network element. The I/O units 33 may be used for communicating with a management application. Reference sign 34 denotes a memory usable, for example, for storing data and programs to be executed by the processor 31 and/or as a working storage of the processor 31.

In one implementation of the present invention according to certain embodiments thereof, the progress counter counts resources used within a single element. While it is possible to use such an indication only in the communication within a node (e.g. from P-CSCF to S-CSCF), it is important that as one component of the invention it must be possible to carry the indication across other node, e.g. from S-CSCF to S-CSCF through an AS in a parameter in the Route header of the S-CSCF.

In a further implementation of the present invention according to certain embodiments thereof, the progress counter counts resources used within the network. In this case, for example, the indication of the progress counter may not be sent transparently from S-CSCF to S-CSCF through the AS like in the above exemplary implementation, but the AS may actively increase the counter.

In particular, as one exemplary use case of the present invention, the progress counter counts the resources used in a single network element. As a second exemplary use case, the progress counter counts the resources used in the network.

In practice, combinations are also possible. For example, all CSCFs in a network may support the progress counter, but the Application Servers may not. Then it may be sent transparently through the AS, but the originating and terminating CSCF roles would add their resource usage.

According to certain embodiments of the present invention, the progress counter may be a simple counter, which is just increased by one by each entity handling the request.

Alternatively, according to certain embodiments, each SIP role in the call may increase the progress counter dependent on the logic it applies, such as the S-CSCF increases the counter by a higher value after the ISC interface was used, reflecting the fact that the external AS interface was used. As an example therefore, a stateless I-CSCF, which will not stay in the call path after 200 OK, might not increase the value. As further example, an "INVITE"-message for Multimedia Telephony MMtel may be considered more important and resource intensive than a call for best effort VoIP. As still further example, the P-CSCF may increase the value more with an associated Rx Diameter or Iq H.248 interaction than without.

As a variant of the previous item, the value a node adds to the progress counter may depend on its overload level. That is, for example, the higher the overload the more the progress counter is increased. As a special case the progress counter need not be present in every message during normal operation, but is only used and added by network elements in overload.

According to further embodiments of the present invention, the progress counter may be a dedicated SIP header field or carried as URI parameter in existing header fields. Further, Combinations are also possible, such as sending the 'pcoc' through an AS as part of the "ODI" (3GPP original dialogue identifier in 3GPP TS 24.229) and in a dedicated header between CSCFs.

The "SIP INVITE"-message is the most important use case, but it is to be noted that the present invention is not restricted thereto, and may also cover other SIP methods, such as "SUBSCRIBE", "UPDATE", and "PRACK" as well. Distinctions and prioritization between the methods may also be done according to an overload table.

According to certain embodiments of the invention, the progress counter may be removed when the SIP request leaves the validity area, e.g. when the message is sent to a user equipment UE.

In the following, examples for implementing the present invention according to certain embodiments are described. It is to be noted that the following examples are merely intended to illustrate the present invention, but the subject-matter thereof is not limited thereto.

According to a non-limiting exemplary implementation of certain embodiments of the present invention, the "INVITE"-message of the basic IMS call flow is shown. It is to be noted that only the SIP INVITE is explicitly shown, whereas Diameter and other interactions are ignored, and responses are also omitted. All call flows are shown end-to-end, that is the call flow in question is not rejected based on or despite a possible overload.

Fig. 4 shows a basic call flow according to certain embodiments of the invention.

In Fig. 4, all CSCF roles on originating and terminating side are within a single network element NE1. Only NE1 is assumed to support the mechanism, this is why the "pcoc"-counter is either stored in the S-CSCF with the dialogue information or passed transparently through the AS as indicated by the brackets in Fig. 4. As a consequence, the pcoc is not increased by the AS.

Fig. 5 shows a basic call flow in a case, in which all functional entities (e.g. S/P-CSCF, TAS) in the core network support the mechanism of the invention. The pcoc is increased in each step. The example shows the functional entities distributed across three network elements NE1, NE2 and NE3.

Fig. 6 shows a variant of the call flow of Fig. 5 with the same distribution of roles across NEs. However, in this example the pcoc counter is not increased by the I-CSCF, because it is stateless and will not be part of subsequent requests. This it is considered of lower "weight" and a less important "resource".

Finally, also Fig. 7 shows a variant of the call flow of Fig. 5 with the same distribution of roles across NEs in the upper call flow, and all roles on NE3 in the lower call flow.

However, in this example the pcoc counter is only increased by network elements in overload. In the upper call flow, the CSCF NE1 on the originating side is not in overload, nor is the TAS (NE2). The counter is increased only on the terminating side. However, for the

lower call flow all CSCF roles are on NE3 and thus the counter is increased also on the originating side.

Fig. 8 shows a rejection logic based on pcoc and Overload level with respect to the situation of Fig. 6. For example, if overload level 3 is observed, then N2 out of 6 initial "INVITE"-messages with pcoc=1, and N3 out of 6 messages with pcoc=2, will be rejected.

Beneath others, the present invention provides the following advantages. The main advantage is that rejection of calls in overload will prefer calls in early phase. As a result, the number of successful e2e calls can be significantly increased compared to arbitrary random message rejection.

Furthermore, the solution according to the present invention may apply to both network element overload and network wide overload.

Still further, the solution is independent of the call flow. This is in contrast to overload control which distinguishes interface types.

Moreover, the present invention may also work with co-located roles and with roles implemented on dedicated servers.

Additionally, the solution may start simple with simple counters and be refined as explained in the above.

To summarize, the present invention provides a prioritization of messages, which are more 'advanced' in the call flow. This means the more system resources were already spent on one session (e.g. call) set-up attempt, the less likely it should be rejected because of overload. According to certain embodiments of this invention, the 'resources already spent' are counted or accumulated during the session (call) set-up, and the value is carried in SIP signaling. Such counter is defined as a progress counter for overload control (pcoc). Messages with a progress counter value higher than a certain threshold, in dependency of the overload level, will not be rejected or will be less likely to be rejected.

It is to be noted that embodiments of the present invention may be implemented as circuitry, in software, hardware, application logic or a combination of software, hardware and application logic. In an example embodiment, the application logic, software or an instruction

set is maintained on any one of various conventional computer-readable media. In the context of this document, a "computer-readable medium" may be any media or means that can contain, store, communicate, propagate or transport the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer or smart phone, or user equipment.

As used in this application, the term "circuitry" refers to all of the following: (a) hardware-only circuit implementations (such as implementations in only analog and/or digital circuitry) and (b) to combinations of circuits and software (and/or firmware), such as (as applicable): (i) to a combination of processor(s) or (ii) to portions of processor(s)/software (including digital signal processor(s)), software, and memory(ies) that work together to cause an apparatus, such as a mobile phone or server, to perform various functions) and (c) to circuits, such as a microprocessor(s) or a portion of a microprocessor(s), that require software or firmware for operation, even if the software or firmware is not physically present. This definition of 'circuitry' applies to all uses of this term in this application, including in any claims. As a further example, as used in this application, the term "circuitry" would also cover an implementation of merely a processor (or multiple processors) or portion of a processor and its (or their) accompanying software and/or firmware. The term "circuitry" would also cover, for example and if applicable to the particular claim element, a baseband integrated circuit or applications processor integrated circuit for a mobile phone or a similar integrated circuit in server, a cellular network device, or other network device.

If desired, the different functions discussed herein may be performed in a different order and/or concurrently with each other. Furthermore, if desired, one or more of the above-described functions may be optional or may be combined.

Although various aspects of the invention are set out in the independent claims, other aspects of the invention comprise other combinations of features from the described embodiments and/or the dependent claims with the features of the independent claims, and not solely the combinations explicitly set out in the claims.

It is also noted herein that while the above describes example embodiments of the invention, these descriptions should not be viewed in a limiting sense. Rather, there are several variations and modifications which may be made without departing from the scope of the present invention as defined in the appended claims.

Furthermore, the described elements may be implemented by software, e.g. by a computer program product for a computer, and/or by hardware. In any case, for executing their respective functions, correspondingly used devices, nodes or network elements may comprise several means, modules, units, components, etc. (not shown) which are required for control, processing and/or communication/signaling functionality. Such means, modules, units and components may comprise, for example, one or more processors or processor units including one or more processing portions for executing instructions and/or programs and/or for processing data, storage or memory units or means for storing instructions, programs and/or data, for serving as a work area of the processor or processing portion and the like (e.g. ROM, RAM, EEPROM, and the like), input or interface means for inputting data and instructions by software (e.g. floppy disc, CD-ROM, EEPROM, and the like), a user interface for providing monitor and manipulation possibilities to a user (e.g. a screen, a keyboard and the like), other interface or means for establishing links and/or connections under the control of the processor unit or portion. It is to be noted that in the present specification processing portions should not be only considered to represent physical portions of one or more processors, but may also be considered as a logical division of the referred processing tasks performed by one or more processors.

The following meanings for the abbreviations used in this specification apply:

3GPP	3rd Generation Partnership Project
AS	Application Server
CSCF	Call Session Control Function
eNB	evolved NodeB (base station in LTE)
I-CSCF	Interrogating CSCF
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ISC	IMS Service Control
LTE	Long Term Evolution
NE	Network Element
pcoc	progress counter for overload control
P-CSCF	Proxy CSCF

S-CSCF	serving CSCF
SIP	Session Initiating Protocol
TAS	Telephony Application Server
UE	User Equipment
URI	Uniform Resource Identifier
VoIP	Voice over IP
VoLTE	Voice over LTE



**What is claimed is:**

1. A method, comprising
  - determining usage of a resource at session setup with a signaling protocol;
  - counting the determined used resources during the session setup;
  - increasing a value of a progress counter for overload control based on the counted used resources, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging; and
  - prohibiting, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value.
2. The method according to claim 1, wherein the threshold is set dependent on an overload level of the occurring overload.
3. The method according to claim 1 or 2, wherein usage of resources of a single network element is counted.
4. The method according to claim 1 or 2, wherein usage of resources in the network is counted, and the progress counter is increased by an application server.
5. The method according to claim 1 or 2, wherein, in case an application server does not support the progress counter, the received progress counter is sent transparently through the application server, whereas the originating and the terminating Call Session Control Functions add their resource use.
6. The method according to claim 5, wherein an indication of the progress counter is transmitted across another node through an application server in a parameter in the route header of a Call Session Control Function.
7. The method according to any of claims 1 to 6, wherein the progress counter is increased by one by each entity handling the session setup request.

8. The method according to any of claims 1 to 7, wherein the progress counter is increased by each Session Initiating Protocol resource with a value dependent on the logic it applies.

9. The method according to any of claims 1 to 7, wherein the value a resource adds depends on its overload level.

10. The method according to any of claims 1 to 9, wherein the progress counter is used and added by network elements only in case of an overload.

11. The method according to any of claims 1 to 10, wherein the progress counter is a dedicated Session Initiating Protocol header field and/or carried as an Uniform Resource Identifier parameter in existing header fields.

12. The method according to any of claims 1 to 11, wherein the Session Initiation Protocol messaging method is one of INVITE, SUBSCRIBE, UPDATE and PRACK.

13. The method according to any of claims 1 to 12, wherein the progress counter is removed when the Session Initiation Protocol leaves a validity area.

14. An apparatus, comprising

    a determination unit configured to determine usage of a resource at session setup with a signaling protocol;

    a counting unit configured to count the determined used resources during the session setup;

    a processing unit configured to increase a value of a progress counter for overload control based on the counted used resources, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging; and

    a prohibiting unit configured to prohibit, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value.

15. The apparatus according to claim 14, wherein the threshold is set dependent on an overload level of the occurring overload.

16. The apparatus according to claim 14 or 15, wherein usage of resources of a single network element is counted.

17. The apparatus according to claim 14 or 15, wherein the processing unit is configured to count usage of resources in the network, and the progress counter is increased by an application server.

18. The apparatus according to claim 14 or 15, wherein, in case an application server does not support the progress counter, the received progress counter is sent transparently through the application server, whereas the originating and the terminating Call Session Control Functions add their resource use.

19. The apparatus according to claim 18, wherein an indication of the progress counter is transmitted across another node through an application server in a parameter in the route header of a Call Session Control Function.

20. The apparatus according to any of claims 14 to 19, wherein the progress counter is increased by one by each entity handling the session setup request.

21. The apparatus according to any of claims 14 to 20, wherein the progress counter is increased by each Session Initiating Protocol resource with a value dependent on the logic it applies.

22. The apparatus according to any of claims 14 to 20, wherein the value a resource adds depends on its overload level.

23. The apparatus according to any of claims 14 to 22, wherein the progress counter is used and added by network elements only in case of an overload.

24. The method according to any of claims 14 to 21, wherein the progress counter is a dedicated Session Initiating Protocol header field and/or carried as a Uniform Resource Identifier parameter in existing header fields.

25. The apparatus according to any of claims 14 to 24, wherein the Session Initiation Protocol messaging method is one of INVITE, SUBSCRIBE, UPDATE and PRACK.

26. The apparatus according to any of claims 1 to 25, wherein the progress counter is removed when the Session Initiation Protocol leaves a validity area.

27. A computer program product for a computer, comprising software code portions for performing the steps of any of claims 1 to 13 when said product is run on the computer.

28. The computer program product according to claim 27, wherein

the computer program product comprises a computer-readable medium on which said software code portions are stored, and/or

the computer program product is directly loadable into the internal memory of the computer and/or transmittable via a network by means of at least one of upload, download and push procedures.

**Abstract****End-to-end Session Initiation Protocol overload control in IP Multimedia subsystem**

The present invention addresses method, apparatus and computer program product for an improved end-to-end Session Initiation Protocol overload control in IP Multimedia subsystem. The usage of a resource at session setup with a signaling protocol is determined, the determined used resources during the session setup is counted, a value of a progress counter for overload control based on the counted used resources is increased, wherein the progress counter for overload control is carried in a Session Initiation Protocol messaging, and, in case of an overload, rejection of messages in the Session Initiation Protocol messaging with a progress counter value higher than a set threshold value is prohibited.

Fig. 4

Fig. 1

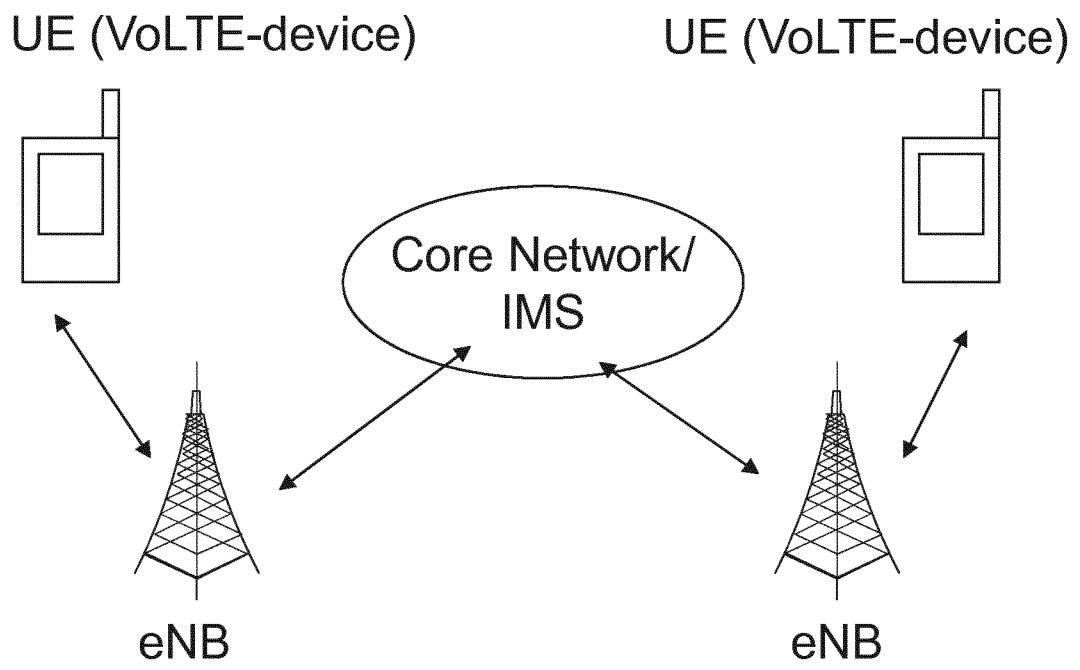


Fig. 2

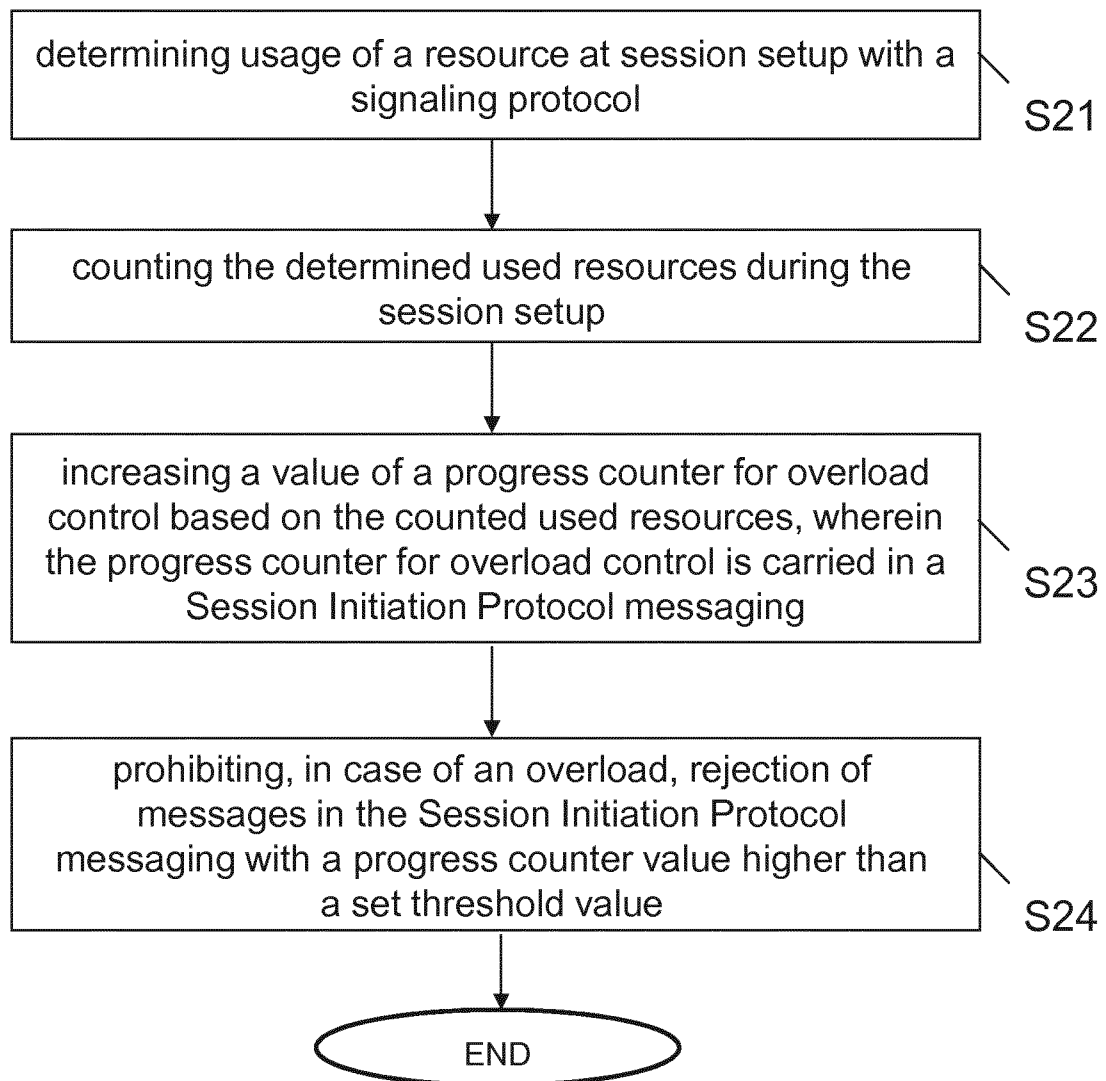


Fig. 3

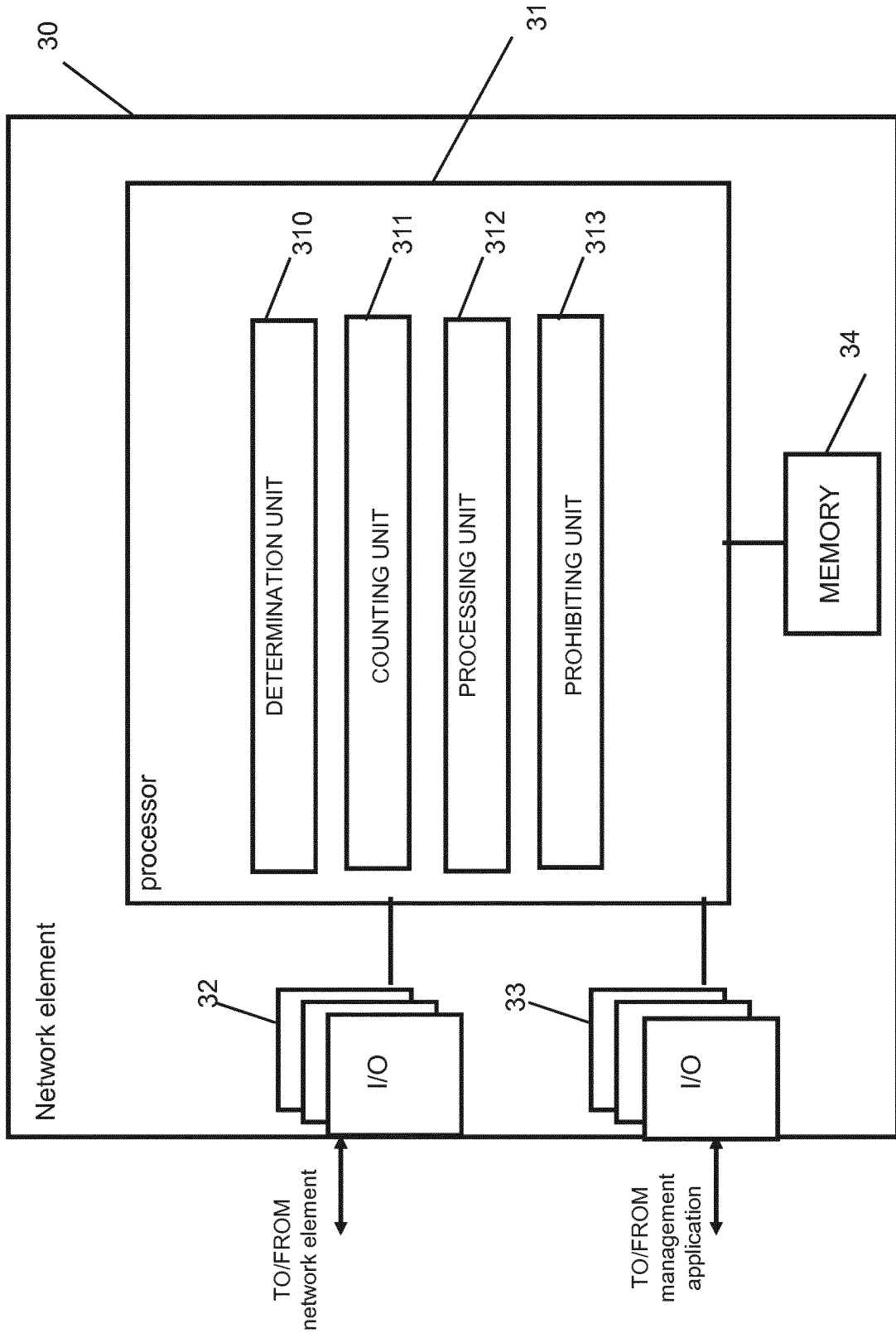




Fig. 4

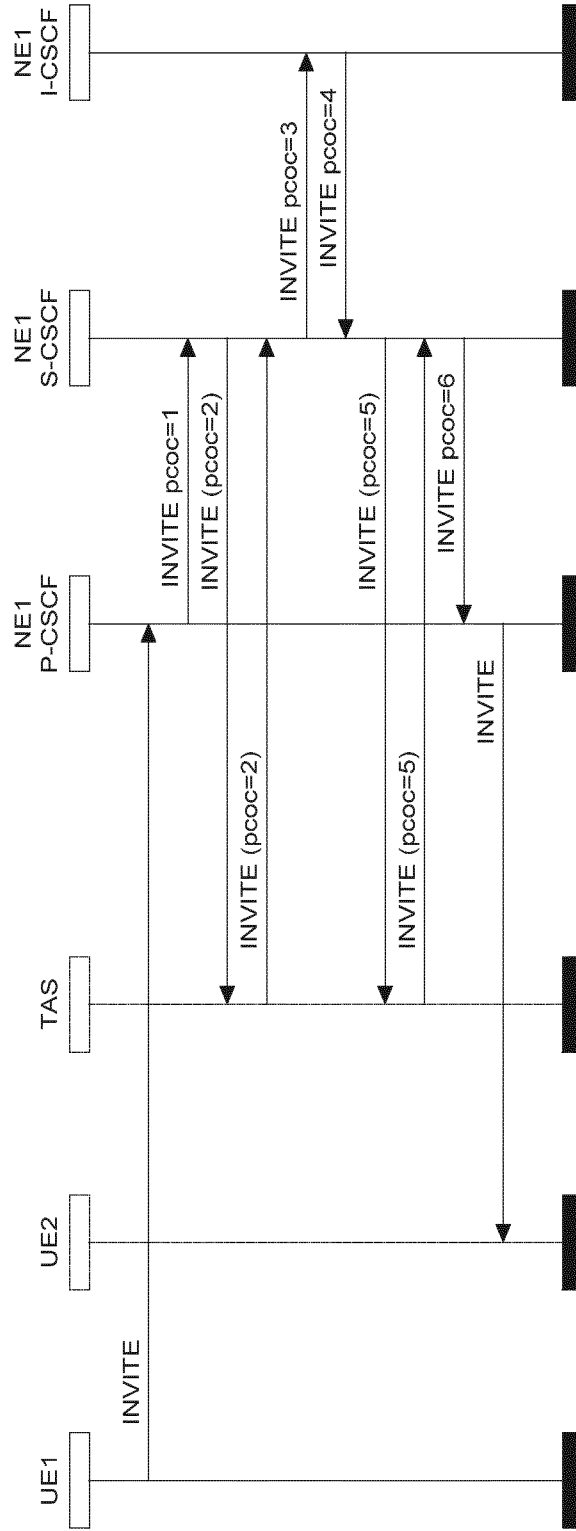


Fig. 5

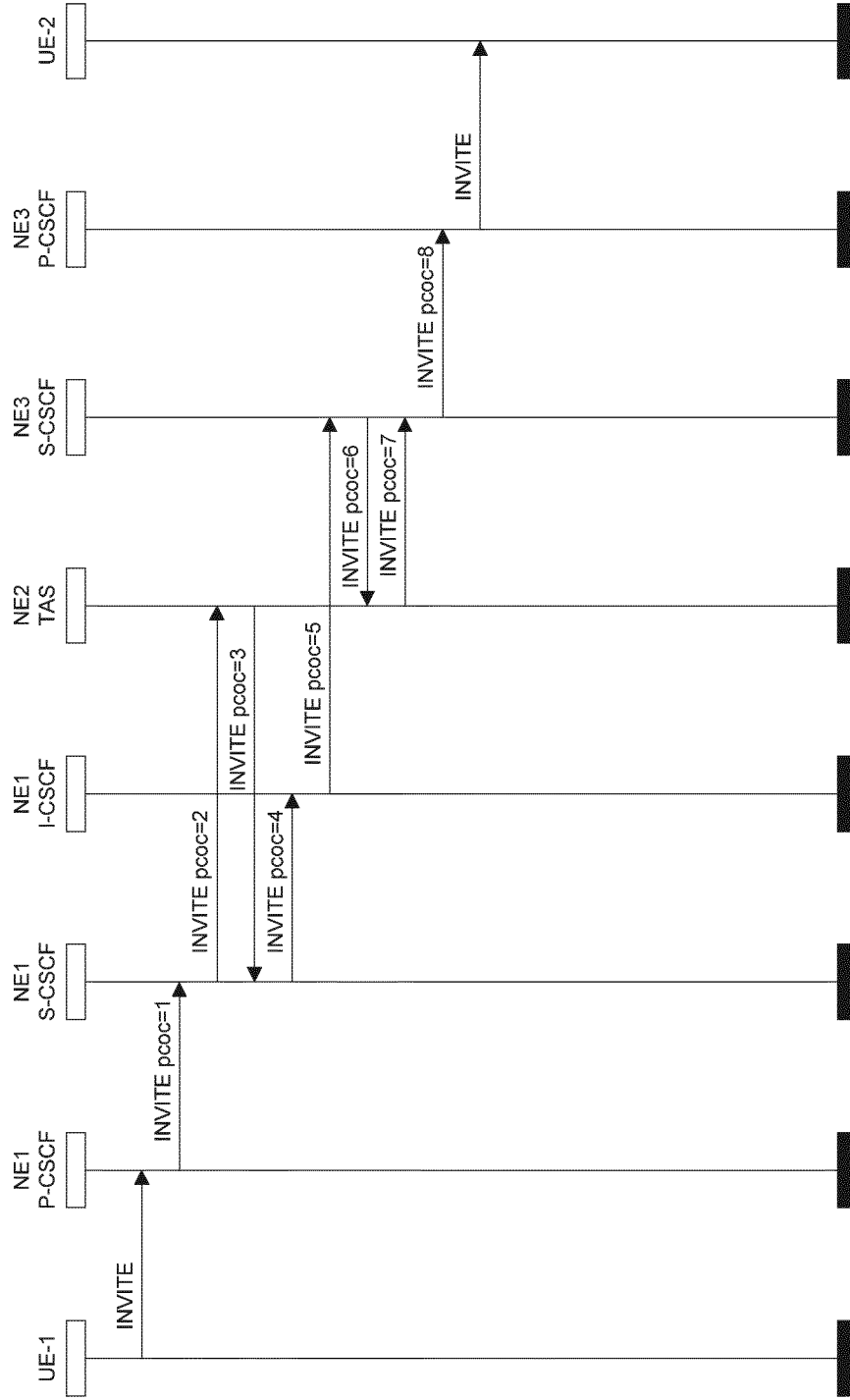


Fig. 6

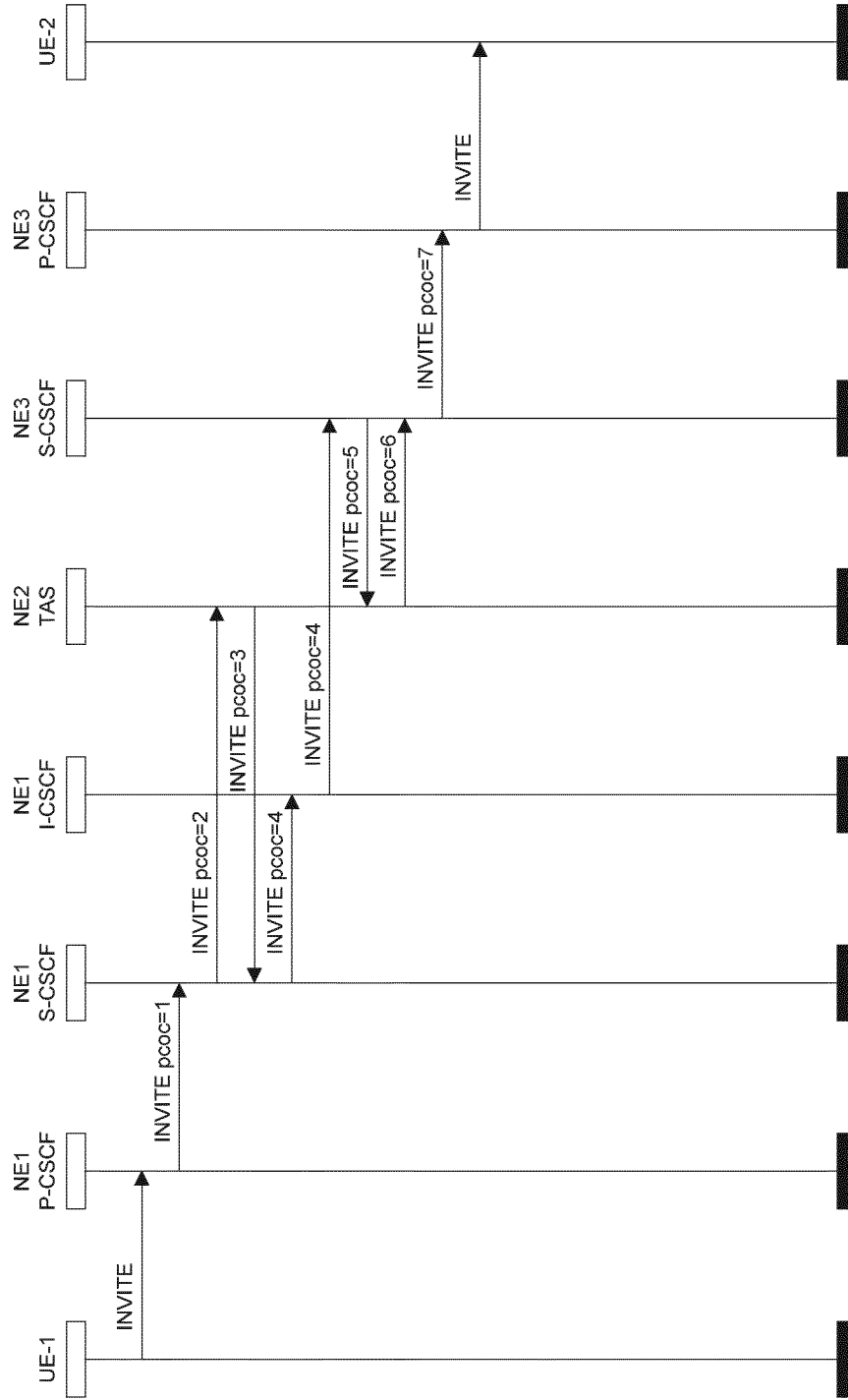


Fig. 7

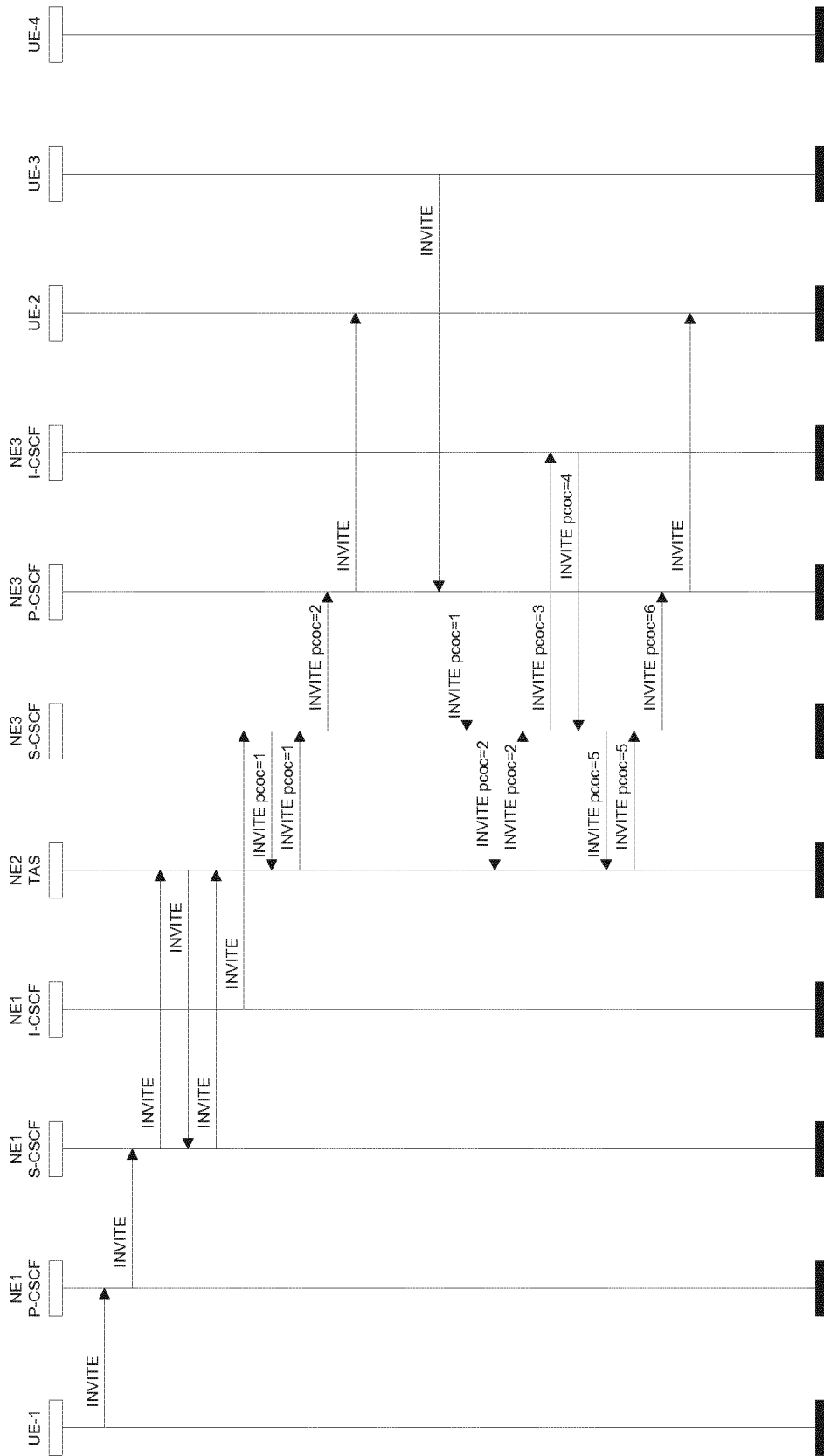


Fig. 8

Overload level	Number of Initial INVITE messages rejected
0	0
1	0
2	$\frac{N1[\text{pcoc}=1]}{6}$
3	$\frac{N2[\text{pcoc}=1] + N3[\text{pcoc}=2]}{6}$
4	$\frac{N4[\text{pcoc}=1] + N5[\text{pcoc}=2] + N6[\text{pcoc}=3]}{6}$
5	...
6	...
7	...
8	1