



(51) International Patent Classification:
H04L 29/06 (2006.01)

(21) International Application Number:
PCT/US2013/048047

(22) International Filing Date:
27 June 2013 (27.06.2013)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
13/536,713 28 June 2012 (28.06.2012) US

(71) Applicant: **ALCATEL LUCENT** [FR/FR]; 3, avenue
Octave Gréard, F-75007 Paris (FR).

(72) Inventors: **CAI, Yigang**; 1960 Lucent Lane, Naperville,
IL 60563-1594 (US). **HUA, Suzann**; 2000 Lucent Lane,
Naperville, IL 60563-1443 (US).

(74) Agent: **LA BRUNO, David, M.**; Alcatel-Lucent Usa Inc.,
Attention: Docket Administrator - Room 3B-212F, 600-
700 Mountain Avenue, Murray Hill, NJ 07974-0636 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY,
BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM,
DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KN, KP, KR,
KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME,
MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ,
OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC,
SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

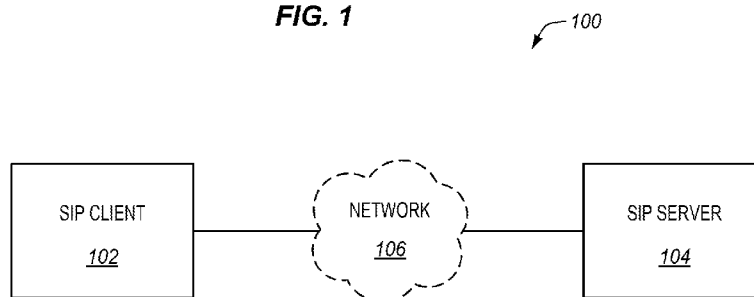
(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ,
UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ,
TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK,
EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,
MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,
TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: SESSION INITIATION PROTOCOL (SIP) FOR MESSAGE THROTTLING

FIG. 1



(57) Abstract: Systems and methods that use SIP for message throttling. A system includes a SIP client that communicates with a SIP server. When in operation, the SIP client sends a SIP request to the SIP server, such as a SIP INVITE. The SIP client then receives a SIP response from the SIP server answering the request. The SIP client parses the SIP response to identify an overload indicator which indicates that an overload condition exists in the SIP server, and limits additional SIP requests toward the SIP server based on the overload indicator.



SESSION INITIATION PROTOCOL (SIP) FOR MESSAGE THROTTLING

Field of the Invention

5 The invention is related to the field of communication systems and, in particular, to network elements that communicate through Session Initiation Protocol (SIP).

Background

Session Initiation Protocol (SIP) is a signaling protocol defined by the Internet Engineering Task Force (IETF)) for controlling communication sessions over an Internet Protocol (IP) network. For example, SIP may be used to set up, modify, or tear down a voice call, a video call, an IP TV session, an online gaming session, etc. One particular type of IP-based network that uses SIP is an IP Multimedia Subsystem (IMS) network.

The exchange of SIP messages between entities of a network to manage a SIP session is referred to as a SIP transaction. A SIP transaction includes a SIP client (also referred to as a user agent client) that sends SIP requests. The transaction also includes a SIP server (also referred to as a user agent server) that receives SIP requests and returns SIP responses.

20 **Summary**

Embodiments described herein provide for handling overload conditions in a SIP server. A SIP server may experience congestion, hardware or software failures, or some other issue that can overload the server. Presently, when a SIP server experiences an overload condition, the SIP server is still able to respond to a SIP client. Therefore, the client keeps sending requests to the server because it does not know that the server is overloaded. As presently defined, SIP doesn't specify any type of throttling between a SIP client and a SIP server to relieve the overload condition. If a server is experiencing an overload condition and a client continues to send requests as usual, then the overload condition will get worse and may eventually disable the server all together.

30 The embodiments described herein add message throttling to SIP so that overload conditions are managed between a SIP client and a SIP server. If a SIP server receives a SIP request from a SIP client, then the server determines whether an overload condition exists. If so, the server is able to notify the client of the overload condition through a SIP

response. The client then reduces additional SIP requests toward the server that is overloaded. This reduces the burden on the server so that it can recover from the overload condition. When the server does recover, it is able to notify the client of the recovery so that the client can send additional SIP requests toward the server at a normal rate.

5 One embodiment comprises a SIP client that communicates with a SIP server. The SIP client is configured to send a SIP request to the SIP server, and to receive a SIP response from the SIP server answering the SIP request. The SIP client is configured to parse the SIP response to identify an overload indicator which indicates that an overload condition exists in the SIP server, and to limit additional SIP requests toward the SIP server
10 based on the overload indicator.

 In another embodiment, the SIP response includes a new SIP header defined for the overload indicator.

 In another embodiment, the overload indicator indicates a severity level of the overload condition in the SIP server.

15 In another embodiment, the SIP client is configured to limit additional SIP requests toward the SIP server by a percentage based on the severity level of the overload condition.

 In another embodiment, the SIP client is configured to send another SIP request to the SIP server, and to receive another SIP response from the SIP server. The SIP client is configured to parse the other SIP response to identify the overload indicator which indicates
20 that the overload condition no longer exists in the SIP server, and to resume transmission of additional SIP requests toward the SIP server at a normal rate based on the overload indicator.

 In another embodiment, the SIP client is configured to wait a time period before resuming transmission of the additional SIP requests toward the SIP server at the normal
25 rate.

 In another embodiment, the SIP client is configured to gradually increase transmission of the additional SIP requests toward the SIP server until the transmission reaches the normal rate.

 In another embodiment, the SIP server is configured to receive the SIP request from
30 the SIP client, and to generate the SIP response as an answer to the SIP request. The SIP server is further configured to detect the overload condition, to insert the overload indicator in the SIP response that the overload condition exists in the SIP server, and to transmit the SIP response to the SIP client.

Another embodiment comprises a method of notifying a SIP client of an overload condition in a SIP server. The method includes sending a SIP request from the SIP client to the SIP server, and receiving a SIP response in the SIP client from the SIP server answering the SIP request. The method further includes parsing the SIP response in the SIP client to
5 identify an overload indicator which indicates that an overload condition exists in the SIP server, and limiting additional SIP requests toward the SIP server based on the overload indicator.

Another embodiment comprises a SIP server that communicates with a SIP client. The SIP server is configured to receive a SIP request from the SIP client, and to generate a
10 SIP response that answers the SIP request. The SIP server is configured to detect an overload condition, to insert an overload indicator in the SIP response that the overload condition exists, and to transmit the SIP response to the SIP client.

Other exemplary embodiments may be described below.

15 **Description of the Drawings**

Some embodiments of the present invention are now described, by way of example only, and with reference to the accompanying drawings. The same reference number represents the same element or the same type of element on all drawings.

FIG. 1 illustrates a communication network in an exemplary embodiment.

20 FIGS. 2-3 are flow charts illustrating a method of exchanging overload handling capabilities via SIP in an exemplary embodiment.

FIG. 4 is a flow chart illustrating a method 400 of notifying a SIP client of overload conditions in an exemplary embodiment.

25 FIG. 5 is a flow chart illustrating a method 500 of limiting the number of SIP requests that is sent to a SIP server in an exemplary embodiment.

FIG. 6 illustrates communication network in another exemplary embodiment.

FIG. 7 is a message diagram illustrating SIP messaging used to provide overload handling in an exemplary embodiment.

Description of Embodiments

The figures and the following description illustrate specific exemplary embodiments of the invention. It will thus be appreciated that those skilled in the art will be able to devise various arrangements that, although not explicitly described or shown herein, embody the principles of the invention and are included within the scope of the invention. Furthermore, any examples described herein are intended to aid in understanding the principles of the invention, and are to be construed as being without limitation to such specifically recited examples and conditions. As a result, the invention is not limited to the specific embodiments or examples described below, but by the claims and their equivalents.

FIG. 1 illustrates a communication network 100 in an exemplary embodiment. Network 100 represents a packet-switched network that uses SIP protocol, such as an IP Multimedia Subsystem (IMS) network. In this embodiment, network 100 includes a SIP client 102 connected to a SIP server 104 by a network 106. Client 102 and server 104 may represent network elements of a packet-switched network. For example, client 102 may represent a Serving-Call Session Control Function (S-CSCF) of an IMS network, while server 104 may represent an application server of an IMS network. Network 100 may include many other SIP user agents that are not shown for the sake of clarity.

In the embodiments described below, client 102 is able to throttle SIP requests toward server 104 if server 104 becomes overloaded. Before client 102 is able to throttle SIP requests, client 102 can notify server 104 that it supports overload handling and vice-versa, which is further described in FIGS. 2-3.

FIGS. 2-3 are flow charts illustrating a method 200 of exchanging overload handling capabilities via SIP in an exemplary embodiment. The steps of method 200 will be described with reference to network 100 in FIG. 1, but those skilled in the art will appreciate that method 200 may be performed in other networks and systems. The steps of the flow charts described herein are not all inclusive and may include other steps not shown. The steps may also be performed in an alternative order.

In step 202, client 102 generates a SIP request that is intended for server 104, such as a SIP INVITE, a SIP MESSAGE, a SIP OPTIONS, etc. One assumption is that client 102 supports overload handling. In other words, client 102 is able to limit or throttle the number of SIP requests that are sent to a server if the server is overloaded. Therefore, client 102 inserts an indicator (referred to as a capability indicator) in the SIP request that it supports overload handling in step 204. The indicator may comprise a Boolean value, an

integer value, a string, etc. Client 102 then transmits the SIP request to server 104 in step 206. The SIP request therefore notifies server 104 that client 102 supports overload handling.

5 In order to allow for the notification as described above, a new value may be defined in SIP for the capability indicator. The new value may be added to the existing Accept header of SIP messages. The new value may be named "Overload-Notification" value or something similar.

10 In FIG. 3, server 104 receives the SIP request from client 102 in step 208. As an answer to the SIP request, server 104 generates a SIP response in step 210, such as a SIP 200 OK. Server 104 also parses the SIP request to identify the capability indicator in step 212. If server 104 determines that client 102 supports overload handling (based on the capability indicator), then server 104 determines if it also supports overload handling. If server 104 supports overload handling, then server 104 inserts a corresponding capability indicator in the SIP response (in step 214) that it supports overload handling. As with the
15 SIP request, the capability indicator may be a new value added to the existing Accept header of the SIP response. Server 104 then transmits the SIP response to client 102 in step 216. The SIP response therefore notifies client 102 that server 104 supports overload handling.

20 In order to support the SIP sessions in network 100, client 102 will send multiple SIP requests to server 104. For example, client 102 may send SIP INVITEs, SIP MESSAGEs, SIP OPTIONS, etc., to server 104. In the embodiments described herein, SIP is further enhanced so that server 104 can notify client 102 of overload conditions, which is further described in FIG. 4.

25 FIG. 4 is a flow chart illustrating a method 400 of notifying a SIP client of overload conditions in an exemplary embodiment. The steps of method 400 will be described with reference to network 100 in FIG. 1, but those skilled in the art will appreciate that method 400 may be performed in other networks and systems.

30 In step 402, server 104 receives a SIP request from client 102. As an answer to the SIP request, server 104 generates a SIP response in step 404, such as a 200 OK. If client 102 supports overload handling (based on the prior notification), then server 104 determines whether an overload condition exists in step 406. An overload condition comprises any condition or processing state where a server operates above its capacity. For example, assume that a server has the capacity to handle 100,000 requests during a time frame. If the

server receives 120,000 requests during that time frame, then the server is operating above its capacity and is considered overloaded.

In addition to determining whether an overload condition exists, server 104 may also determine a severity level of the overload condition. The severity level may be a percentage
5 indicating how overloaded the server 104 is. For example, the severity level may be 10%, 50%, 100%, etc.

If an overload condition is detected, then server 104 inserts an overload indicator in the SIP response in step 408. The overload indicator comprises any data which indicates whether an overload condition exists in a SIP user agent. The overload indicator may
10 comprise a Boolean value, an integer value, a string, etc. For example, the overload indicator may be an integer value in the range of 1-10, 1-100, etc. An integer value greater than zero may indicate that an overload condition exists, and may also indicate the severity level of the overload condition. In this instance, the overload indicator indicates that an overload condition does exist in server 104. Server 104 then transmits the SIP response to
15 client 102 in step 410. Thus, server 104 notifies client 102 of the overload condition through the SIP response.

In order to allow for the overload notification as described above, a new SIP header (or header parameter) may be defined for the overload indicator. The new header may have integer value between 0 and 100, where 0 means no overload condition or an overload
20 condition has recovered. A value greater than 0 may indicate an overload condition. The greater the value, the more severe the overload condition. The new header may be named "Overload-Severity" or something similar.

When client 102 receives a SIP response that indicates an overload condition in server 104, client 102 is able to limit the number of future requests that are sent to server
25 104 while it is overloaded. FIG. 5 is a flow chart illustrating a method 500 of limiting the number of SIP requests that is sent to a SIP server in an exemplary embodiment. The steps of method 500 will be described with reference to network 100 in FIG. 1, but those skilled in the art will appreciate that method 500 may be performed in other networks and systems.

In step 502, client 102 receives the SIP response from server 104. Client 102 parses
30 the SIP response in step 504 to identify the overload indicator inserted by server 104. If the overload indicator indicates that an overload condition exists in server 104, then client 102 limits or throttles additional SIP requests toward server 104 for a time period based on the overload indicator (in step 506). For example, when client 102 determines that server 104 is

overloaded, client 102 may start a timer for a throttling window. The timer may be configurable and dynamically changeable. If the overload indicator received from server 104 is an integer value, then client 102 may limit additional SIP requests by a percentage based on the integer value. If the integer value is 50 (on a scale of 1 to 100), then client 102
5 may limit (or delay) additional SIP requests by 50%. If the integer value is 80 (on a scale of 1 to 100), then client 102 may limit (or delay) additional SIP requests by 80%.

Each time server 104 receives a new SIP request from client 102, server 104 may operate as described in method 400 to determine whether an overload condition exists and/or the severity level of the overload condition. If the overload condition becomes more
10 or less severe, then server 104 notifies client 102 through additional SIP responses. Client 102 continues to limit additional SIP requests toward server 104 while the overload condition exists in server 104. For example, if client 102 receives another SIP response from server 104 indicating that the overload condition still exists, then client 102 may reset the throttling timer and continue to limit new SIP requests toward server 104.

15 Much as server 104 is able to notify client 102 when an overload condition exists, server 104 is also able to notify client 102 when no overload condition exists or when a prior overload condition has recovered, as is further shown in FIG. 4. When server 104 receives a SIP request from client 102 (see step 402), server 104 generates a SIP response (in step 404) and determines whether an overload condition exists (in step 406). If server
20 104 does not detect an overload condition or determines that a prior overload condition has recovered, then server 104 inserts an overload indicator in the SIP response in step 412 which indicates that an overload condition does not exist or that an overload condition has recovered. The overload indicator may be an integer value, such as on a scale of 1-10, 1-100, etc. An integer value of zero indicates that no overload condition exists, or that the severity level of an overload condition is zero. Server 104 then transmits the SIP response
25 to client 102 in step 410. Therefore, server 104 is able to notify client 102 that an overload condition no longer exists through the SIP response.

When client 102 receives the SIP response from server 104 (see step 502 of FIG. 5), client 102 parses the SIP response to identify the overload indicator inserted by server 104
30 (see step 504). In this instance, the overload indicator indicates that no overload condition exists in server 104. Therefore, client 102 terminates throttling and resumes transmission of additional SIP requests toward server 104 at a normal rate in step 508. If client 102 had previously limited SIP requests toward server 104 because it was notified of an overload

condition in server 104, then client 102 could return to normal operation and send additional SIP requests toward server 104 in a normal fashion.

It may not be beneficial to bombard server 104 with SIP requests immediately after it recovers from an overload condition. If client 102 (and other clients) sends a large number of SIP requests to server 104 soon after it recovers from an overload condition, then server 104 may become overloaded again. Therefore, client 102 may continue to throttle SIP requests toward server 104 for a time period after server 104 has recovered. Client 102 may wait a time period before resuming transmission of SIP requests toward the SIP server at a normal rate. Alternatively, client 102 may gradually increase transmission of the SIP requests toward the SIP server until reaching its normal rate of transmission. This should avoid a situation where server 104 becomes overloaded again by a large number of SIP requests.

Example

FIGS. 6-7 illustrate an example of operating an IMS network to provide overload handling through SIP protocol. FIG. 6 illustrates communication network 600 in another exemplary embodiment. Communication network 600 includes an access network 602, a Proxy-Call Session Control Function (P-CSCF) 604, and an IMS network 606. IMS network 606 includes a Serving-Call Session Control Function (S-CSCF) 612, an Interrogate-CSCF (I-CSCF) 614, a Home Subscriber Server (HSS) 616, and an application server (AS) 618. A mobile device 620 connects to IMS network 606 through access network 602.

FIG. 7 is a message diagram illustrating SIP messaging used to provide overload handling in an exemplary embodiment. In this example, assume that S-CSCF 612 communicates with AS 618 using SIP. Further assume that S-CSCF 612 receives a SIP INVITE for a session, and determines that AS 618 should be contacted for the session (such as by processing initial filter criteria (iFC)). Before forwarding the SIP INVITE to AS 618, S-CSCF 612 inserts a capability indicator in the SIP INVITE indicating that S-CSCF 612 supports overload handling. S-CSCF 612 inserts the capability indicator in an existing ACCEPT header of the SIP INVITE. S-CSCF 612 then transmits the INVITE to AS 618. The INVITE therefore notifies AS 618 that S-CSCF 612 supports overload handling.

In response to receiving the INVITE, AS 618 generates a response such as a SIP 200 OK. Because the INVITE includes a capability indicator which shows that S-CSCF 612

supports overload handling, AS 618 determines whether AS 618 also supports overload handling. If it does, then AS 618 inserts a corresponding capability indicator in the 200 OK indicating that AS 618 supports overload handling. AS 618 inserts the capability indicator in an existing ACCEPT header of the 200 OK. AS 618 then transmits the 200 OK to S-CSCF 612. The 200 OK therefore notifies S-CSCF 612 that AS 618 supports overload handling.

During the SIP session or other SIP sessions, S-CSCF 612 may send multiple SIP requests toward AS 618, such as new INVITEs, Re-INVITEs, etc. When the SIP requests are received, AS 618 generates the appropriate responses, such as 200 OKs. Because both AS 618 and S-CSCF 612 support overload handling based on the prior notifications, then AS 618 determines whether an overload condition exists and a severity level of the overload condition.

If an overload condition is not detected (as with the first two INVITEs), then AS 618 inserts an overload indicator (OVERLOAD IND) in the 200 OK that no overload condition exists. More particularly, AS 618 inserts the overload indicator in a new SIP header defined for SIP responses. In this example, the overload indicator comprises an integer value in the range of 0-100. An integer value of zero indicates that an overload condition does not exist or no longer exists. AS 618 then sends the 200 OK to S-CSCF 612.

If an overload condition is detected (as with the third INVITE), then AS 618 inserts an overload indicator in the 200 OK that an overload condition does exist. An integer value greater than zero (e.g., 50 in FIG. 7) indicates that an overload condition exists, and also indicates the severity level of the overload condition. Because an overload condition was detected, the overload indicator is a value between 1-100 depending on the severity of the overload condition. AS 618 then transmits the 200 OK to S-CSCF 612.

When S-CSCF 612 receives the 200 OK that indicates an overload condition in AS 618, S-CSCF 612 is able to limit the number of future SIP requests that are sent to AS 618 while it is overloaded. For example, S-CSCF 612 may set a throttling timer during which time S-CSCF 612 will throttle SIP requests toward AS 618. S-CSCF 612 may throttle additional SIP requests toward AS 618 by 50% based on the integer value of the overload indicator (which is 50). By limiting future SIP requests toward AS 618, the overload condition may be resolved more quickly or easily.

The above example illustrates that overload handling can be implemented effectively through SIP. The addition of new SIP headers allows a SIP server to notify a

SIP client of overload conditions so that the client can reduce the number of SIP requests that are sent to the server while the server attempts to recover from an overload condition.

Any of the various elements shown in the figures or described herein may be implemented as hardware, software, firmware, or some combination of these. For example,
5 an element may be implemented as dedicated hardware. Dedicated hardware elements may be referred to as “processors”, “controllers”, or some similar terminology. When provided by a processor, the functions may be provided by a single dedicated processor, by a single shared processor, or by a plurality of individual processors, some of which may be shared. Moreover, explicit use of the term “processor” or “controller” should not be construed to
10 refer exclusively to hardware capable of executing software, and may implicitly include, without limitation, digital signal processor (DSP) hardware, a network processor, application specific integrated circuit (ASIC) or other circuitry, field programmable gate array (FPGA), read only memory (ROM) for storing software, random access memory (RAM), non volatile storage, logic, or some other physical hardware component or module.

Also, an element may be implemented as instructions executable by a processor or a
15 computer to perform the functions of the element. Some examples of instructions are software, program code, and firmware. The instructions are operational when executed by the processor to direct the processor to perform the functions of the element. The instructions may be stored on storage devices that are readable by the processor. Some
20 examples of the storage devices are digital or solid-state memories, magnetic storage media such as a magnetic disks and magnetic tapes, hard drives, or optically readable digital data storage media.

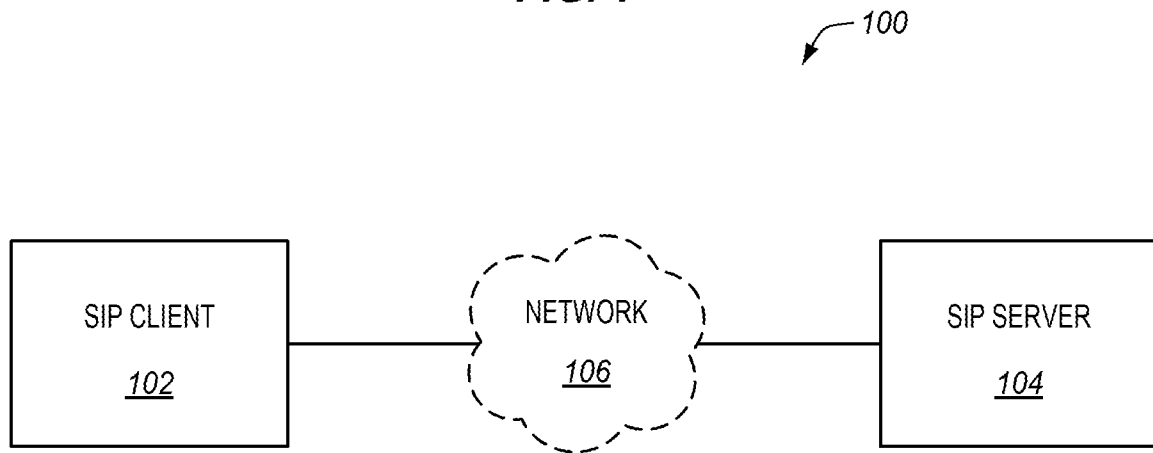
Although specific embodiments were described herein, the scope of the invention is not limited to those specific embodiments. The scope of the invention is defined by the
25 following claims and any equivalents thereof.

We claim:

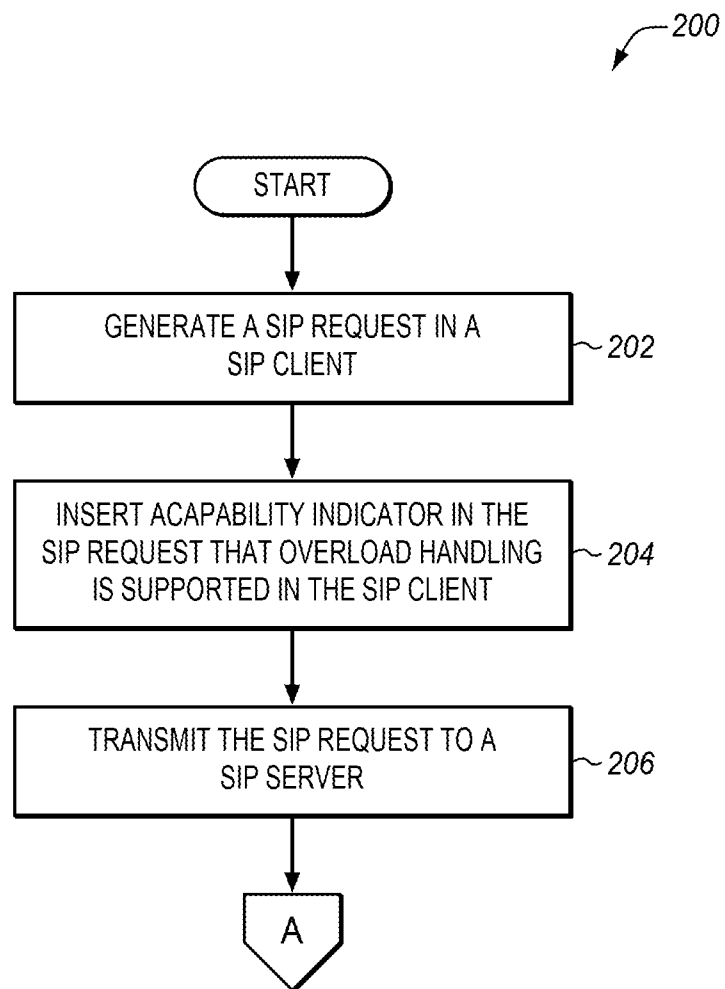
1. A system comprising:
 - a Session Initiation Protocol (SIP) client configured to send a SIP request to a SIP
 - 5 server, and to receive a SIP response from the SIP server;
 - wherein the SIP client is configured to parse the SIP response to identify an overload indicator which indicates that an overload condition exists in the SIP server, and to limit additional SIP requests toward the SIP server based on the overload indicator.
- 10 2. The system of claim 1 wherein:
 - the SIP response includes a new SIP header defined for the overload indicator.
3. The system of claim 2 wherein:
 - the overload indicator indicates a severity level of the overload condition; and
 - 15 the SIP client is configured to limit additional SIP requests toward the SIP server by a percentage based on the severity level of the overload condition.
4. The system of claim 1 wherein:
 - the SIP client is configured to send another SIP request to the SIP server, and to
 - 20 receive another SIP response from the SIP server;
 - the SIP client is configured to parse the other SIP response to identify the overload indicator which indicates that the overload condition no longer exists in the SIP server, and to resume transmission of additional SIP requests toward the SIP server at a normal rate based on the overload indicator.
- 25 5. The system of claim 1 wherein:
 - the SIP server is configured to receive the SIP request from the SIP client, and to generate the SIP response as an answer to the SIP request;
 - the SIP server is configured to detect the overload condition, to insert the overload
 - 30 indicator in the SIP response that the overload condition exists in the SIP server, and to transmit the SIP response to the SIP client.

6. A method comprising
- sending a Session Initiation Protocol (SIP) request from a SIP client to a SIP server;
 - receiving a SIP response in the SIP client from the SIP server;
 - parsing the SIP response in the SIP client to identify an overload indicator which
- 5 indicates that an overload condition exists in the SIP server; and
- limiting additional SIP requests toward the SIP server based on the overload indicator.
7. The method of claim 6 wherein:
- 10 the SIP response includes a new SIP header defined for the overload indicator.
8. The method of claim 7 wherein:
- the overload indicator indicates a severity level of the overload condition; and
 - limiting additional SIP requests toward the SIP server based on the overload
- 15 indicator comprises limiting the additional SIP requests toward the SIP server by a percentage based on the severity level of the overload condition.
9. The method of claim 6 further comprising:
- sending another SIP request from the SIP client to the SIP server;
- 20 receiving another SIP response from the SIP server;
- parsing the other SIP response to identify the overload indicator which indicates that the overload condition no longer exists in the SIP server; and
 - resuming transmission of additional SIP requests toward the SIP server at a normal rate based on the overload indicator.
- 25
10. The method of claim 6 further comprising:
- receiving the SIP request in the SIP server from the SIP client;
 - generating the SIP response in the SIP server as an answer to the SIP request;
 - detecting the overload condition in the SIP server;
- 30 inserting the overload indicator in the SIP response that the overload condition exists in the SIP server; and
- transmitting the SIP response from the SIP server to the SIP client.

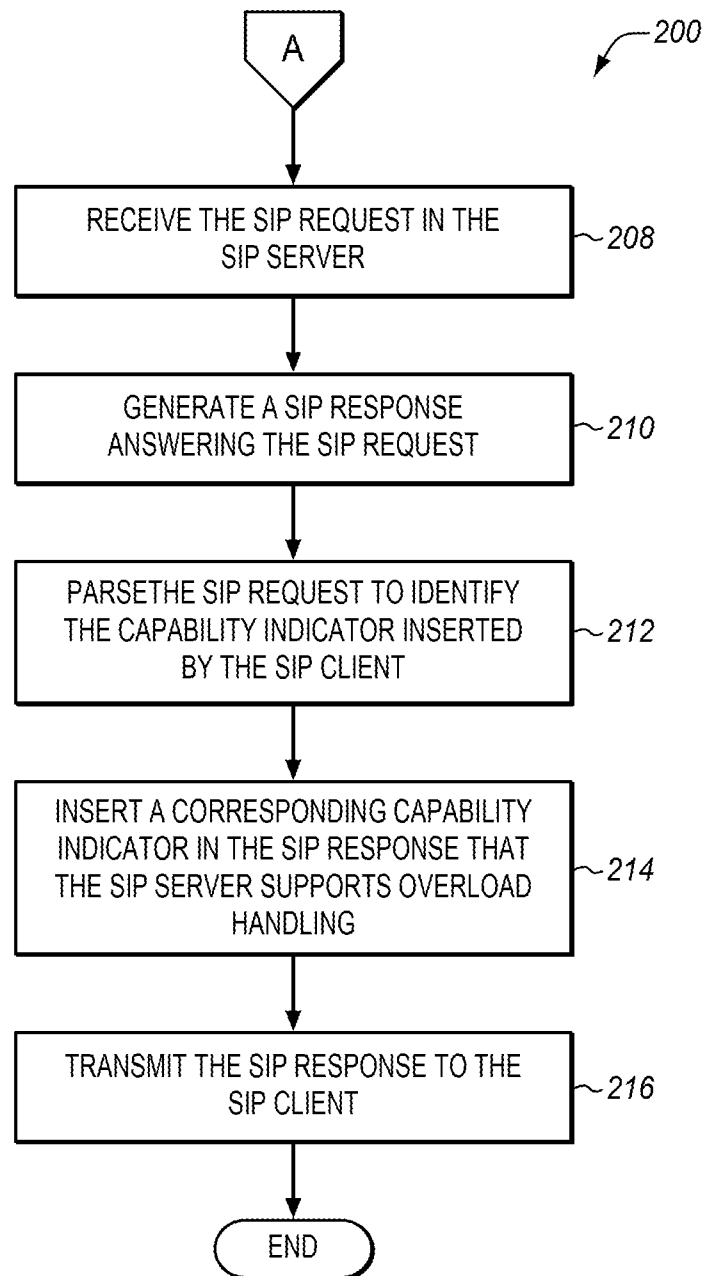
1/7

FIG. 1

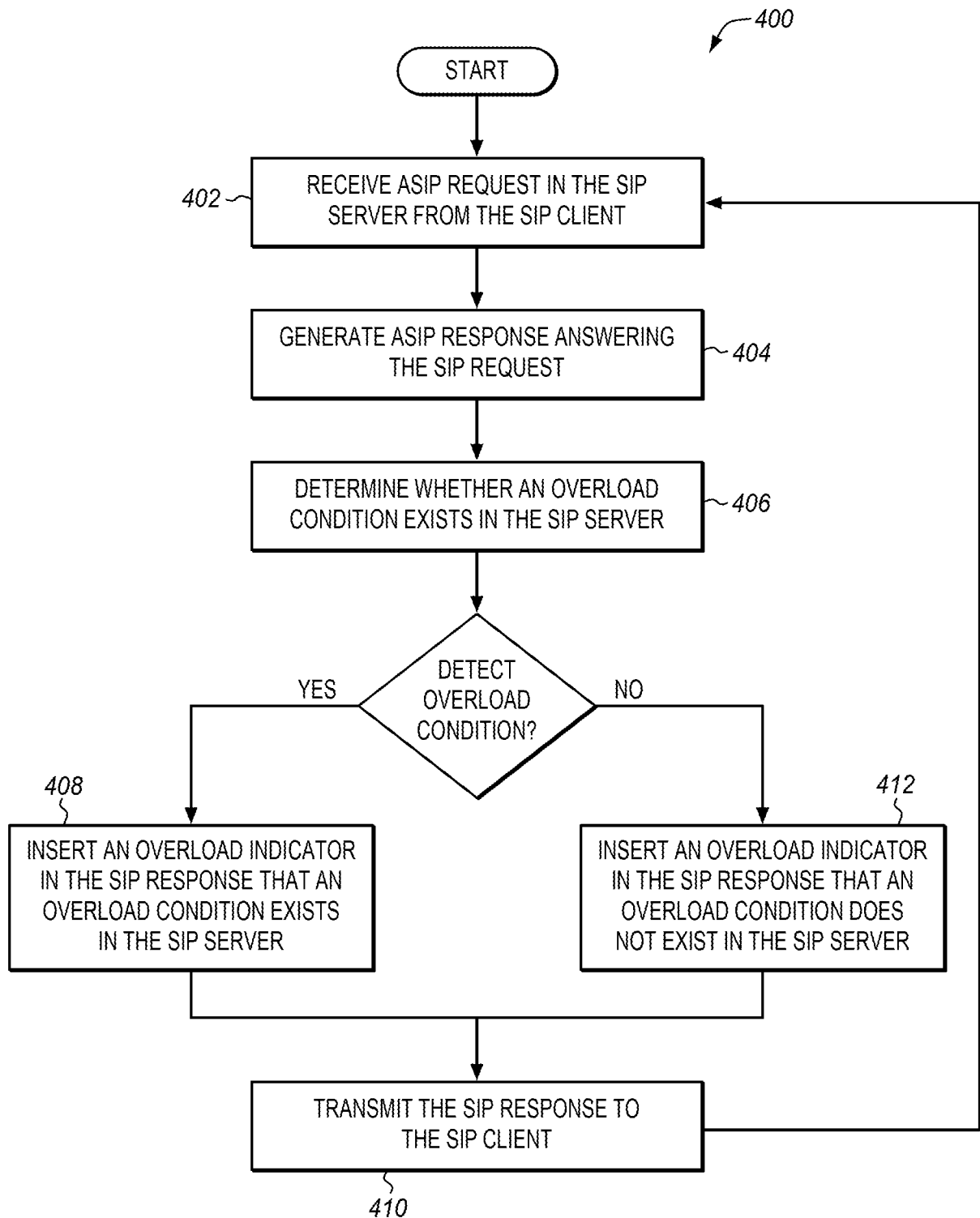
2/7

FIG. 2

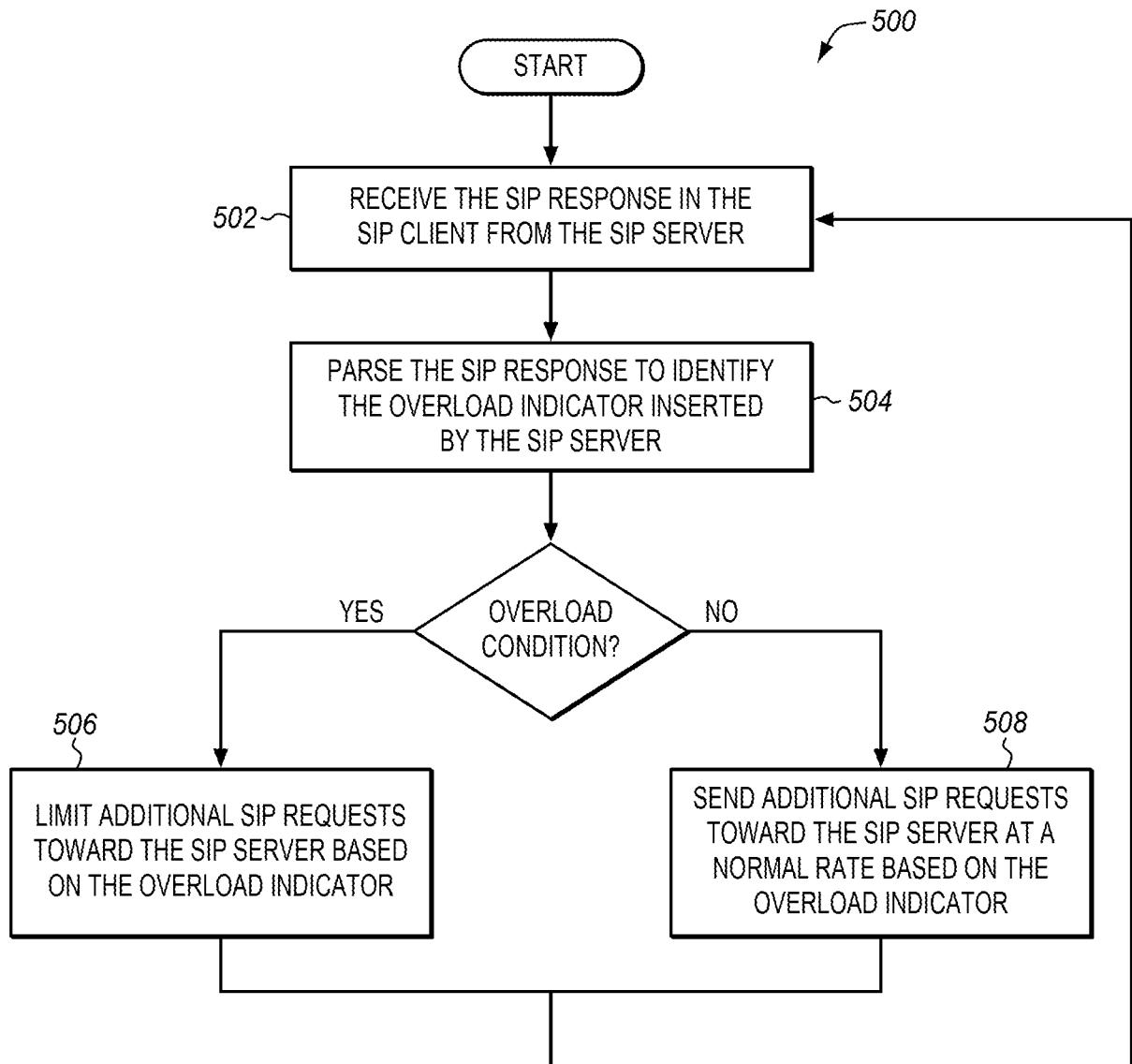
3/7

FIG. 3

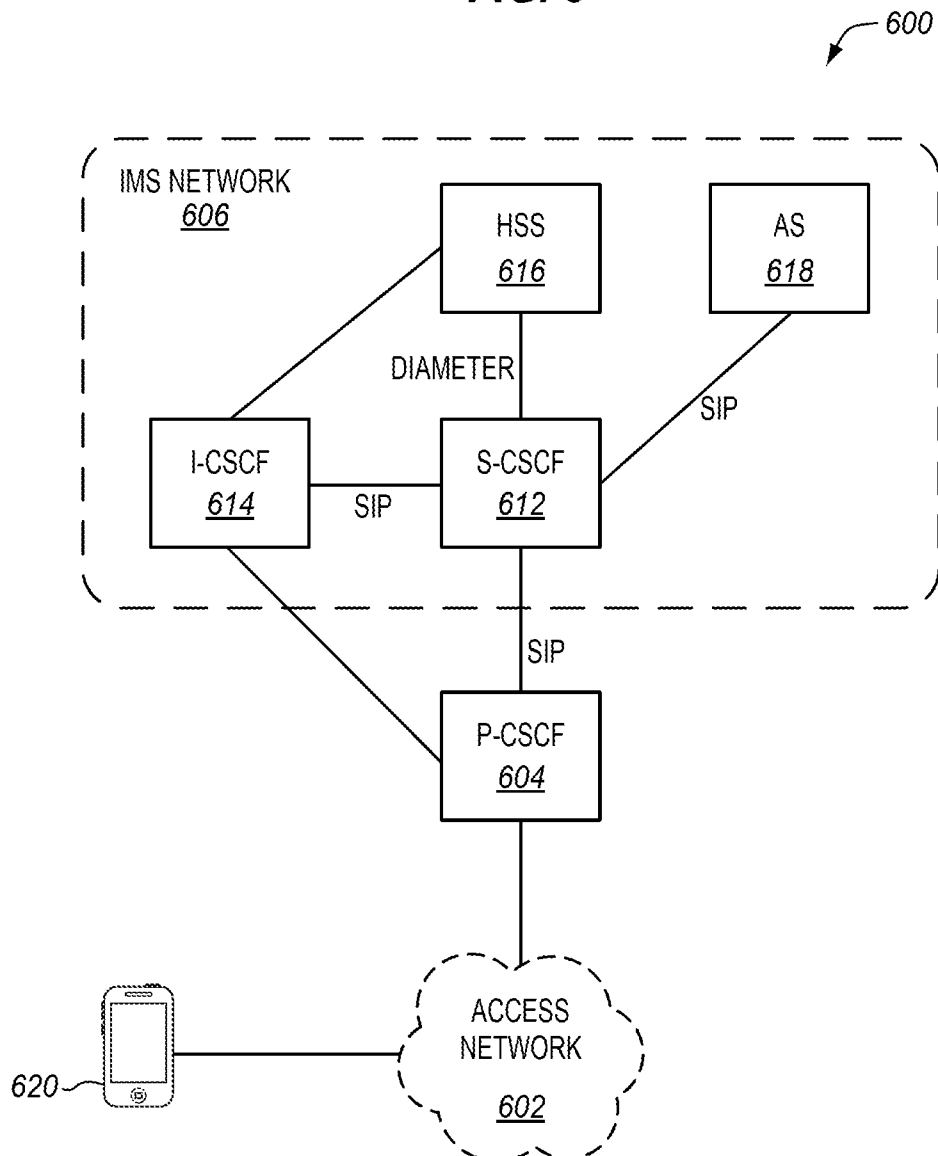
4/7

FIG. 4

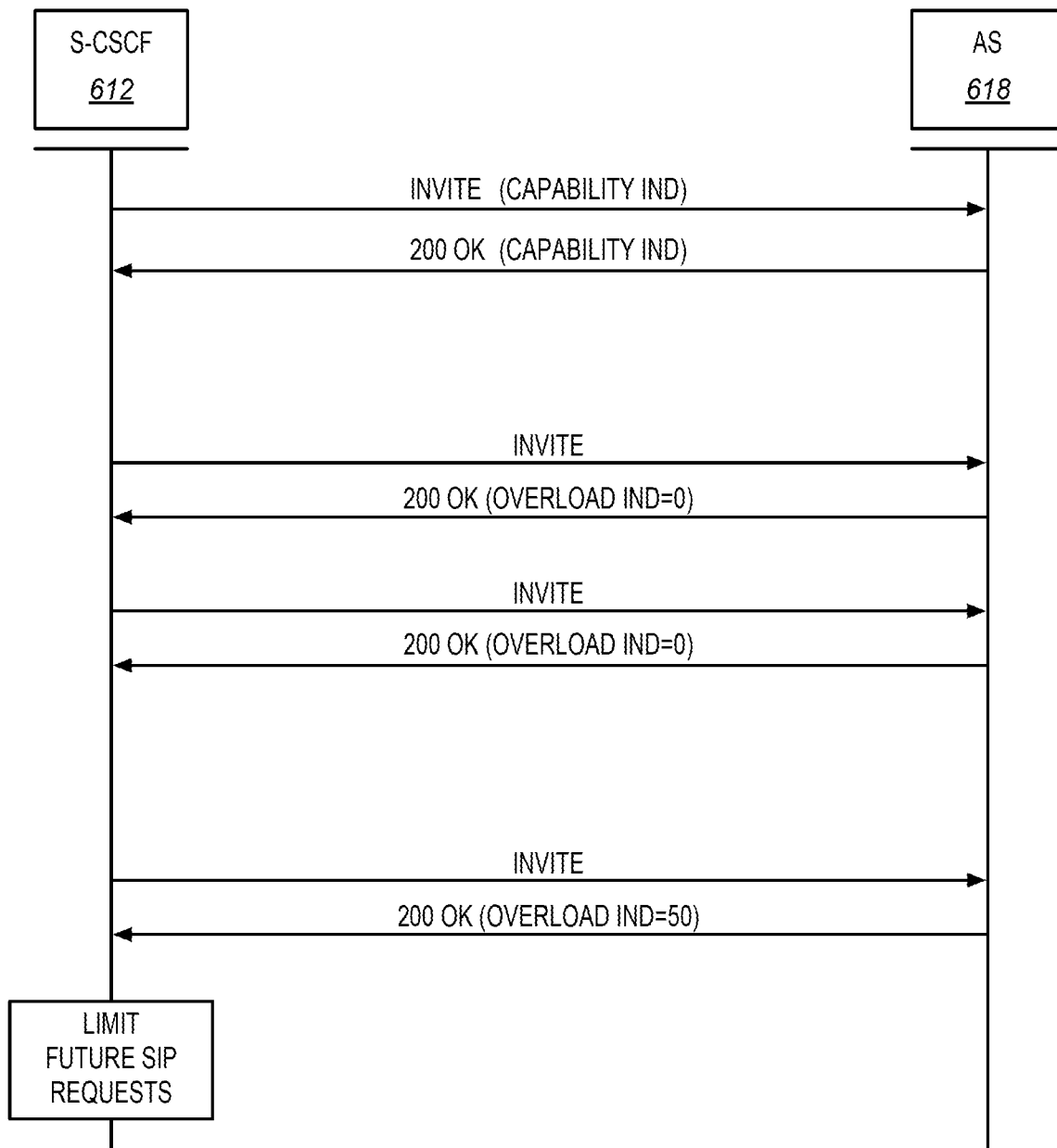
5/7

FIG. 5

6/7

FIG. 6

7/7

FIG. 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2013/048047

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, INSPEC, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	GURBANI V ET AL: "Session Initiation Protocol (SIP) Overload Control; draft-ietf-soc-overload-control-08.txt", SESSION INITIATION PROTOCOL (SIP) OVERLOAD CONTROL; DRAFT-IETF-SOC-OVERLOAD-CONTROL-08.TXT, INTERNET ENGINEERING TASK FORCE, IETF; STANDARDWORKINGDRAFT, INTERNET SOCIETY (ISOC) 4, RUE DES FALAISES CH- 1205 GENEVA, SWITZERLAND, 12 March 2012 (2012-03-12), pages 1-35, XP015081886, [retrieved on 2012-03-12] page 6, paragraph 4. - page 11, paragraph 5.4 page 13, paragraph 5.7 page 16, paragraph 6 - page 17, paragraph 6.2 -----	1-10

☐

Further documents are listed in the continuation of Box C.

☐

See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

13 September 2013

Date of mailing of the international search report

20/09/2013

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Karavassilis, Nick