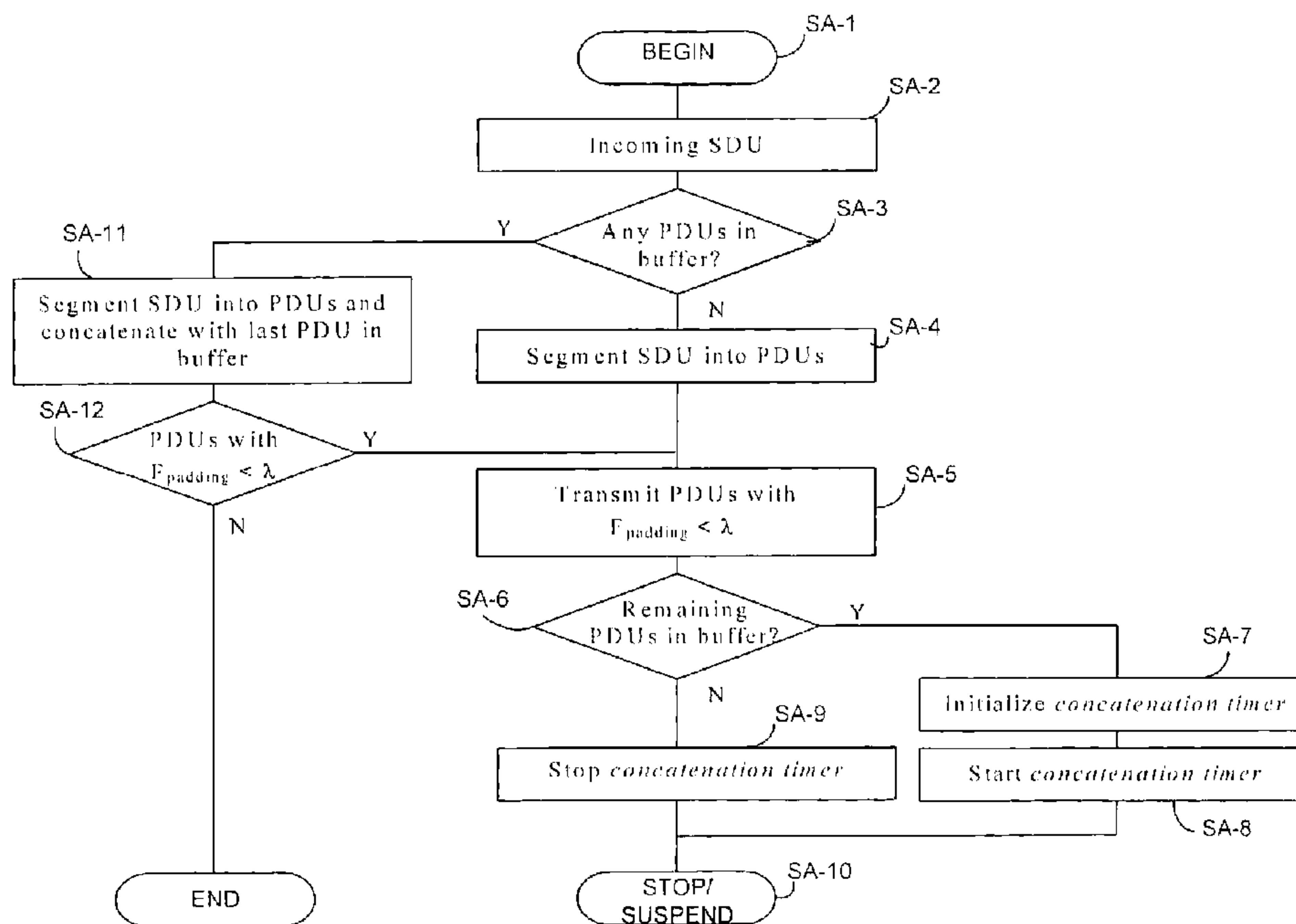




(86) Date de dépôt PCT/PCT Filing Date: 2006/06/27
 (87) Date publication PCT/PCT Publication Date: 2007/01/25
 (45) Date de délivrance/Issue Date: 2015/12/01
 (85) Entrée phase nationale/National Entry: 2008/01/02
 (86) N° demande PCT/PCT Application No.: SE 2006/050220
 (87) N° publication PCT/PCT Publication No.: 2007/011298
 (30) Priorité/Priority: 2005/07/19 (US60/700,327)

(51) Cl.Int./Int.Cl. *H04W 88/08* (2009.01),
H04L 12/951 (2013.01), *H04L 12/955* (2013.01),
H04W 80/02 (2009.01), *H04W 88/16* (2009.01)
 (72) Inventeurs/Inventors:
 SAGFORS, MATS, FI;
 TORSNER, JOHAN, FI;
 WAGER, STEFAN, FI
 (73) Propriétaire/Owner:
 UNWIRED PLANET, LLC, US
 (74) Agent: GILBERT'S LLP

(54) Titre : METHODE POUR REDUIRE LE REMPLISSAGE POUR UN TRAFIC DE TYPE SYSTEME VOCAL SUR INTERNET PAR UNE COMMANDE DE LIAISON RADIO
 (54) Title: MINIMIZING PADDING FOR VOICE OVER INTERNET PROTOCOL-TYPE TRAFFIC OVER RADIO LINK CONTROL



(57) **Abrégé/Abstract:**

A radio access network node (24) comprises protocol data unit (PDU) formation logic (36); a PDU buffer (38); a concatenation timer (40); and a buffer readout mechanism (39). The protocol data unit (PDU) formation logic (36) serves. e.g.. for segmenting incoming service data units (SDUs) to form protocol data unit (PDUs). The PDU buffer (38) stores one or more PDUs. The buffer readout mechanism (39) controls readout of contents of the PDU buffer (38). For example, when contents of a PDU in the PDU buffer (38) has not reach a predetermined fill level, the buffer readout mechanism (39) uses the concatenation timer for determining a delay for readout of the PDU from the PDU buffer (38). The delay provides opportunity for at least a portion of a yet-arrived SDU to be included in the PDU prior to readout of the PDU from the PDU buffer (38), and thereby reduce padding in an outgoing PDU.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau(43) International Publication Date
25 January 2007 (25.01.2007)

PCT

(10) International Publication Number
WO 2007/011298 A1

(51) International Patent Classification:

H04L 12/56 (2006.01)

(21) International Application Number:

PCT/SE2006/050220

(22) International Filing Date: 27 June 2006 (27.06.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/700,327 19 July 2005 (19.07.2005) US

(71) Applicant (for all designated States except US): TELEFONAKTIEBOLAGET LM ERICSSON (PUBL) [SE/SE]; S-164 83 Stockholm (SE).

(72) Inventors; and

(75) Inventors/Applicants (for US only): SÅGFORS, Mats [FI/FI]; Ravalsvägen 8 C13, FIN-02400 Kyrkslätt (FI). TORSNER, Johan [SE/FI]; Skogstorp svägen 2 C9, FIN-02430 Masaby (FI). WAGER, Stefan [SE/FI]; Gårdsgränden 8 H, FIN-02360 Espoo (FI).

(74) Agent: HASSELGREN, Joakim; Ericsson AB, Patent Unit LTE, S-164 80 Stockholm (SE).

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

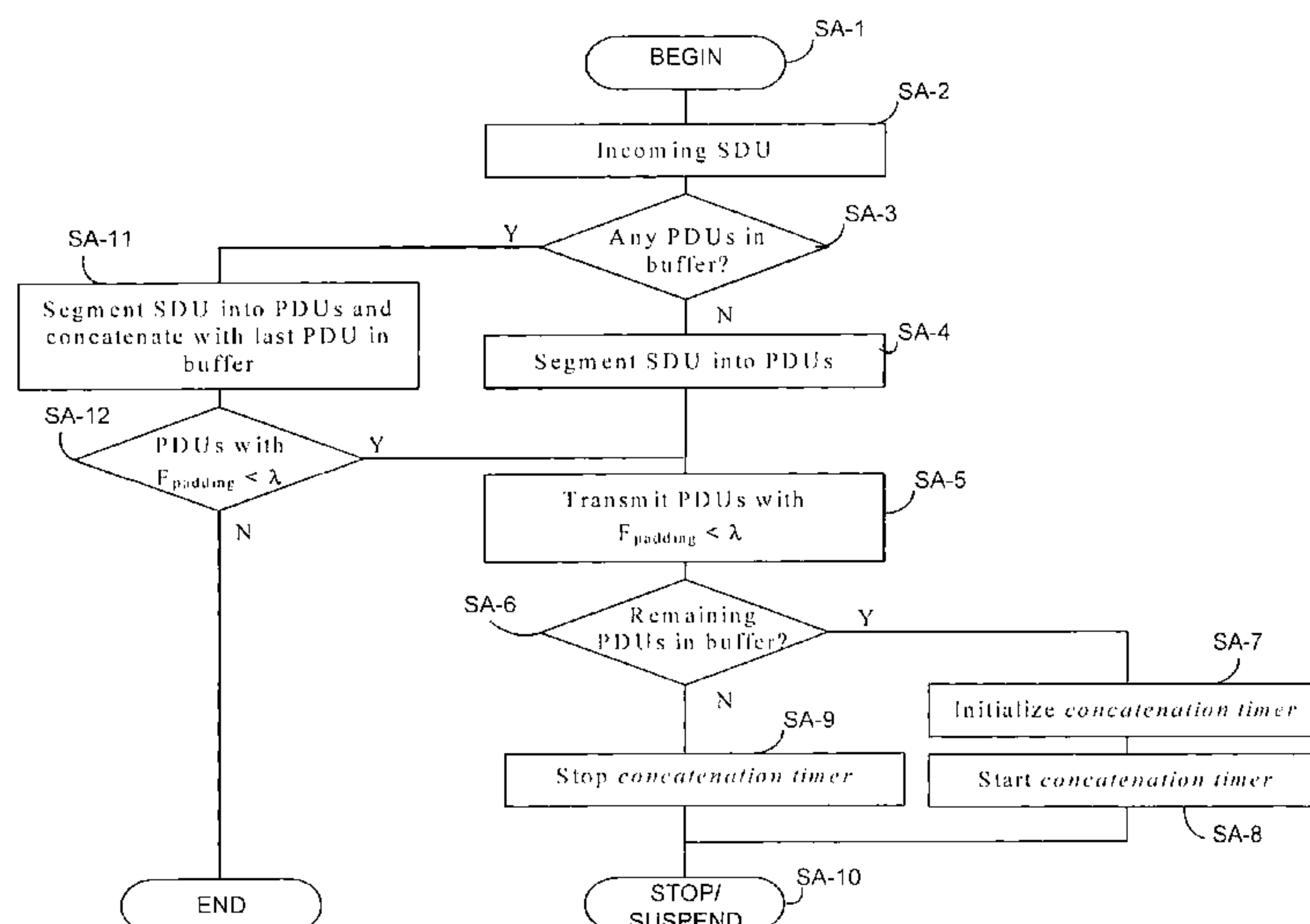
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

- with international search report
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: MINIMIZING PADDING FOR VOICE OVER INTERNET PROTOCOL-TYPE TRAFFIC OVER RADIO LINK CONTROL



(57) Abstract: A radio access network node (24) comprises protocol data unit (PDU) formation logic (36); a PDU buffer (38); a concatenation timer (40); and a buffer readout mechanism (39). The protocol data unit (PDU) formation logic (36) serves. e.g.. for segmenting incoming service data units (SDUs) to form protocol data unit (PDUs). The PDU buffer (38) stores one or more PDUs. The buffer readout mechanism (39) controls readout of contents of the PDU buffer (38). For example, when contents of a PDU in the PDU buffer (38) has not reach a predetermined fill level, the buffer readout mechanism (39) uses the concatenation timer for determining a delay for readout of the PDU from the PDU buffer (38). The delay provides opportunity for at least a portion of a yet-arrived SDU to be included in the PDU prior to readout of the PDU from the PDU buffer (38), and thereby reduce padding in an outgoing PDU.

WO 2007/011298 A1

**MINIMIZING PADDING FOR VOICE OVER
INTERNET PROTOCOL-TYPE TRAFFIC OVER
RADIO LINK CONTROL**

5

BACKGROUND

I. TECHNICAL FIELD

[0001]

10 [0002] This invention presents technology for minimizing the amount of padding when transmitting VoIP or similar traffic over a radio link employing segmentation and concatenation, e.g. the 3G link layer RLC. Reducing padding results in increased network capacity.

15

II. RELATED ART AND OTHER CONSIDERATIONS

[0003] The Wideband Code Division Multiple Access (WCDMA) radio link control (RLC) protocol includes functions for segmentation and concatenation of higher layer "packets" into RLC protocol data units (PDUs). Typically, the RLC is operated
20 with a discrete number of allowed RLC PDU sizes. Segmentation involves a large higher-layer packet being divided or "segmented" into smaller RLC PDUs. Concatenation facilitates the possibility of sending (parts of) several higher layer packets in one RLC PDU. Concatenation reduces the need of padding if several higher-layer PDUs are queued for transmission, in case each packet does not fit perfectly into the
25 RLC PDUs. Segmentation and concatenation are possible in both unacknowledged mode (UM) and acknowledge mode (AM) WCDMA RLC.

[0004] Improved layer two (L2) solutions for voice over internet protocol (VoIP) arc currently being discussed in 3GPP. See, for example. R2-0509693. "L2
30 Optimizations for VoIP" (Qualcomm. 3GPP TSG-RAN WG2 meeting 46bis. 4-8 April 2005) and 1-12-041645. "L2 Considerations for VoIP Support" (Qualcomm. RAN2#43).

[0005] Consider the case of a VoIP service being mapped over a link-layer protocol like WCDMA RLC, which deploys concatenation. In current Radio Link Control protocol (RLC) implementations, the radio link control typically segments incoming Service Data Units (SDUs) into equally sized so-called Protocol Data Units (PDU), which commonly are configured to carry a 40 byte payload.

[0006] Assume that ROHC header compression is used on the application level to minimize the IP overhead transmitted over the link, and that the speech codec transmits voice frames of size 159 bits in 20ms intervals (7.95kbps AMR). This will produce RLC SDUs of a size around 24 bytes. The exact size may fluctuate a bit depending on the ROHC implementation.

[0007] A typical feature of VoIP-like traffic is a fairly well predictable inter-arrival time of packets. In the example above, RLC SDUs will arrive to the RLC with an (average) inter-arrival time of 20 ms. Unless there is any queuing at the RLC layer, the RLC SDUs will be segmented into RLC PDUs directly upon arrival. The twenty four byte SDU will fill only part of the RLC PDU and the rest will be filled with a one byte length indicator to indicate the end of the SDU, and then fifteen bytes of padding. In this scenario, the overall padding added to the traffic stream is around thirty five percent. Thus, the present practice is very inefficient, because thirty five percent of the transmission capacity will be spent on transmitting obsolete bits (e.g., padding). This is particularly problematic for HSDPA-like shared transport channel realizations, where the queue-buildup takes place in the radio base station (RBS) at times of congestion. This means that a large fraction of the resources are also wasted at times when the radio resources are scarce, since a large fraction of the resources are spent on padding.

[0008] The existing RLC protocol supports concatenation, e.g., part(s) of the following SDU may be concatenated into the RLC PDU carrying the end of the last SDU, thus avoiding the use of padding. However, this concatenation requires that the next SDU be available in the radio network controller (RNC) node. But realistically, in the case of a VoIP service with 20ms (or more) between SDUs, the RNC buffer will be empty most of the time, assuming low or moderate load on the air interface. This is especially the case when using the HS-DSCH channel, for which the scheduling buffer resides in the NodeB (e.g., radio base station or RBS), and each RLC SDU is typically forwarded to the Node B directly upon arrival.

[0009] What is needed, therefore, and an object of the present invention, are apparatus, methods, and techniques for reducing the amount of padding and thereby increase network capacity and/or efficiency.

BRIEF SUMMARY

5 [0010] A radio access network node comprises protocol data unit (PDU) formation logic; a PDU buffer; a concatenation timer; and a buffer readout mechanism. The protocol data unit (PDU) formation logic serves, e.g., for segmenting incoming service data units (SDUs) to form protocol data unit (PDUs). The PDU buffer stores one or more PDUs. The buffer readout mechanism controls readout of contents of the PDU
10 buffer. For example, when contents of a PDU in the PDU buffer has not reach a predetermined fill level, the buffer readout mechanism uses the concatenation timer for determining a delay for readout of the PDU from the PDU buffer. The delay provides opportunity for at least a portion of a yet-arrived SDU to be included in the PDU prior to readout of the PDU from the PDU buffer, and thereby reduce padding in an outgoing
15 PDU.

[0011] The buffer readout mechanism delays readout of a PDU from the PDU buffer until either (1) the contents of the PDU in the buffer has reached the predetermined fill level, or (2) a predetermined time interval (maintained by the concatenation timer) has expired.

20 [0012] In an example implementation, non-limiting implementation, the SDUs are twenty four byte voice over Internet Protocol (VoIP) packets which arrive with twenty millisecond inter-arrival time; forty octets of payload are allowed in each PDU; and the predetermined time delay is set between twenty milliseconds and forty milliseconds

[0013] In terms of predetermined fill level, in an example embodiment the buffer
25 readout mechanism delays read out of the PDU from the PDU buffer by the predetermined time interval if an extent of padding that would occur upon transmission of current contents of the PDU in the PDU buffer is not less than a threshold. In one example implementation, the threshold is ten percent of a length of the PDU size.

[0014] Another aspect of the technology concerns a method of operating a radio access network node. The method includes the basic, example steps of segmenting incoming service data units (SDUs) to form protocol data unit (PDUs); storing one or more PDUs in a PDU buffer; and delaying readout of a PDU of the

5 PDU buffer when contents of the PDU in the PDU buffer has not reached a predetermined fill level. The delay provides opportunity for at least a portion of a yet-arrived SDU to be included in the PDU prior to readout of the PDU from the PDU buffer, and thereby reduce padding in an outgoing PDU. The method can comprise delaying the readout of the PDU from the PDU buffer until either (1) the

10 contents of the PDU in the buffer has reached the predetermined fill level, or (2) a predetermined time interval has expired.

According to an aspect of the present invention, there is provided a radio access network node comprising:

15 protocol data unit formation logic for segmenting incoming service data units to form protocol data units;

a protocol data unit buffer for storing one or more protocol data units;

a concatenation timer; and

a buffer readout mechanism for controlling readout of a protocol data unit in

20 the protocol data unit buffer, the buffer readout mechanism being arranged to:

read out the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is less than a threshold; and

25 delay readout of the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is not less than the threshold,

whereby when the extent of padding is not less than the threshold, the buffer readout mechanism is arranged to use the concatenation timer for determining a

30 delay for readout of the protocol data unit from the protocol data unit buffer for providing an opportunity for at least a portion of a not-yet-arrived service data unit to be included in the protocol data unit prior to readout of the protocol data unit from

4a

the protocol data unit buffer, and thereby reduce padding in an outgoing protocol data unit.

According to another aspect of the present invention, there is provided a method of operating a radio access network node, the method comprising:
5 segmenting incoming service data units to form protocol data units;
storing one or more protocol data units in a protocol data unit buffer; and
controlling readout of a protocol data unit in the protocol data unit buffer,
wherein said controlling readout comprises:

10 reading out the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is less than a threshold; and

delaying readout of the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit
15 is not less than the threshold, thereby providing opportunity for at least a portion of a not-yet-arrived service data unit to be included in the protocol data unit prior to readout of the protocol data unit from the protocol data unit buffer, and thereby reduce padding in an outgoing protocol data unit.

20 BRIEF DESCRIPTION OF THE DRAWINGS

[0015] The foregoing and other objects, features, and advantages of the invention will be apparent from the following more particular description of preferred embodiments as illustrated in the accompanying drawings in which
25 reference characters refer to the same parts throughout the various views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention.

[0016] Fig. 1 is a schematic view of a control node which performs RLC
30 protocol concatenation.

4b

[0017] Fig. 2A and Fig. 2B are flowcharts showing example steps of an example embodiment.

[0018] Fig. 3A - Fig. 3C are diagrammatic views showing a sequence of
5 three packets arriving at a node which implements RLC protocol concatenation.

DETAILED DESCRIPTION

[0001] In the following description, for purposes of explanation and not limitation,
10 specific details are set forth such as particular architectures, interfaces, techniques,
etc. in order to provide a thorough understanding of the present invention. However,
it will be apparent to those skilled in the art that the present invention may be
practiced in other embodiments that depart from these specific details. That is, those
skilled in the

art will be able to devise various arrangements which, although not explicitly described or shown herein, embody the principles of the invention and are included within its spirit and scope. In some instances, detailed descriptions of well-known devices, circuits, and methods are omitted so as not to obscure the description of the present invention with unnecessary detail. All statements herein reciting principles, aspects, and embodiments of the invention, as well as specific examples thereof, are intended to encompass both structural and functional equivalents thereof. Additionally, it is intended that such equivalents include both currently known equivalents as well as equivalents developed in the future, i.e., any elements developed that perform the same function, regardless of structure.

[0002] Thus, for example, it will be appreciated by those skilled in the art that block diagrams herein can represent conceptual views of illustrative circuitry embodying the principles of the technology. Similarly, it will be appreciated that any flow charts, state transition diagrams, pseudocode, and the like represent various processes which may be substantially represented in computer readable medium and so executed by a computer or processor, whether or not such computer or processor is explicitly shown.

[0019] The functions of the various elements including functional blocks labeled as "processors" or "controllers" may be provided through the use of dedicated hardware as well as hardware capable of executing software in association with appropriate software. 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 or distributed. Moreover, explicit use of the term "processor" or "controller" should not be construed to refer exclusively to hardware capable of executing software, and may include, without limitation, digital signal processor (DSP) hardware, read only memory (ROM) for storing software, random access memory (RAM), and non-volatile storage.

[0020] According to one example mode, the last PDU in the RNC segmentation buffer is delayed for a certain amount of time, guarded by a timer, hereafter referred to (only for sake of convenience) as the concatenation timer 40. By delaying the last PDU in the RNC until the next SDU arrives, the PDU can be filled up with parts of this next SDU instead of padding. This eliminates the use of padding on this link and thus increases the capacity of the system. The tradeoff is, of course, an increased delay. A

threshold λ is used to set the acceptable percentage of padding in outgoing PDUs. PDUs exceeding this level are stored until the concatenation timer expires.

[0021] As an example, non-limiting embodiment of implementation, Fig. 1 illustrates a control node, such as a radio network controller (RNC) 24, which (as indicated by arrow 32) receives SDUs, e.g., from a core network. The RNC 24 comprises, among numerous other unillustrated constituent elements, an (optional) SDU reception buffer 34; PDU formation logic 36; PDU buffer 38; PDU buffer readout mechanism 39; and concatenation timer 40.

[0022] Fig. 2A and Fig. 2B are flowcharts showing representative, non-limiting, example steps of an example embodiment. The steps of Fig. 2A and Fig. 2B can be executed instantaneously or substantially instantaneously. 2A particularly shows example steps for handing incoming SDUs, while Fig. 2B shows example steps performed upon expiration of the concatenation timer. The two events that trigger operation are (1) an incoming SDU (step SA-1) or (2) expiration of the concatenation timer (step SB-2). Step SA-2 reflects receipt and processing of the incoming SDU. As step SA-3, a check is made (e.g., by PDU formation logic 36) whether there are any PDUs already in PDU buffer 38. If not, as step SA-4 the incoming SDU is segmented (e.g., by PDU formation logic 36) into PDU(s), and the PDU(s) are stored in PDU buffer 38. If a PDU is full, or if the extent of padding that would result if the current contents (of a PDU in PDU buffer 38) were now transmitted is less than the threshold λ , then as step SA-5 that PDU is readout of buffer 38 (e.g., by PDU buffer readout mechanism 39), e.g., to the NodeB. Then, as step SA-6, a check is made whether there are any remaining PDUs in PDU buffer 38 which have a padding ratio higher than the threshold λ . If the check result is affirmative, as step SA-7 the concatenation timer 40 is initialized to zero and then started as step SA-8. After starting of concatenation timer 40 at step SA-8, the SDU processing routine of Fig. 2A is temporarily ended or suspended (step SA-10). If the check result from step SA7 was negative, the concatenation timer is stopped at step SA9, and then the SDU processing routine of Fig. 2A is temporarily ended or suspended (step SA-10)

[0023] If at step SA-4 it is determined, upon receipt of an incoming SDU, that PDUs already reside in PDU buffer 38, then as step SA-11 the incoming SDU is segmented into PDUs and concatenated with the last PDU already in PDU buffer 38.

As step SA-12 a check is made whether PDU buffer 38 contained any PDUs with an extent of padding less than the threshold λ . If the determination at step SA-12 is positive, step SA-5 and ensuing steps are performed (e.g., step SA-6, step SA-7, step and SA-8. or (as appropriate) step SA-9). If the determination at step SA-12 is
5 negative, then the SDU processing routine of Fig. 2A is temporarily ended or suspended (step SA-13).

[0024] As reflected by step SB-1, the concatenation timer 40 is appropriately monitored. Should the concatenation timer 40 expire, an interrupt or other indication is received (step SB-2). In view of expiration of concatenation timer 40, as step SB-3 the
10 last PDU in PDU buffer 38 is readout, and the timer monitoring routine of Fig. 2B is temporarily concluded.

[0025] Thus, when contents of a PDU in the PDU buffer has not reach a predetermined fill level, the buffer readout mechanism uses the concatenation timer for determining a delay for readout of the PDU from the PDU buffer. The delay provides
15 opportunity for at least a portion of a yet-arrived SDU to be included in the PDU prior to readout of the PDU from the PDU buffer, and thereby reduce padding in an outgoing PDU. The buffer readout mechanism delays readout of a PDU from the PDU buffer until either (1) the contents of the PDU in the buffer has reached the predetermined fill level (step SA-5), or (2) a predetermined time interval (maintained by the concatenation
20 timer) has expired (step SB-3).

[0026] To cause as little extra delay as possible, the concatenation timer (e.g., concatenation timer 40) may be adapted to the inter-arrival time of incoming SDUs. This may be done either by continuous filtered measurements on incoming traffic or by reading QoS attributes like the guaranteed bit rate and frame size, if available. The
25 concatenation timer can be set slightly larger than the inter-arrival time of incoming SDUs, to cover possible jitter. This jitter may also be measured.

[0027] Consider an example case illustrated in the time sequence of Fig. 3A – Fig. 3C. In the example case, VoIP packets of 24 bytes arrive at PDU buffer 38 with 20 ms inter-arrival time to the RLC, where the RLC PDU allows for 40 octets of payload in
30 each PDU (excluding the length indicator). Fig. 3A shows a first packet (Packet #1) arriving (step 2A-2) and being stored in PDU buffer 38. It is assumed, for sake of this

example. that the acceptable padding percentage is set to 10%. Since Packet #1 is the only packet in PDU buffer 38 and thus PDU buffer 38 has only 24 bytes, for Packet #1 it is determined at step SA-5 that the percent or ratio of padding to entire PDU length exceeds the threshold λ . Consequently, Packet #1 remains in PDU buffer 38 (step SA-5 6) and the concatenation timer 40 is both initialized (step SA-7) and started (step SA-8). Preferably, the concatenation timer is initialized/set to a value larger than 20 ms. but less than 40 ms in this example.

[0028] Fig. 3B shows arrival of a second packet (Packet #2) Upon arrival of the second packet (Packet #2), it is determined at step SA-3 that a PDU already resides in PDU buffer 38. i.e., Packet #1 is already in the first PDU of buffer 38. Accordingly, as 10 step SA-11 the newly arriving packet (Packet #2) is segmented into two RLC PDUs in 38. As such, a first PDU of PDU buffer 38 includes the first VoIP packet, a one byte length indicator (LI) and fifteen bytes of the second VoIP packet. But nine bytes of the second packet did not fit into the first RLC PDU of PDU buffer 38, and consequently 15 are stored in a second PDU position of PDU buffer 38 as illustrated in Fig. 3B.

[0029] The first PDU of PDU buffer 38 is readout in accordance with step SA-5, as shown in Fig. 3B. However, since it is determined at step SA-6 that the percentage of padding in the second PDU of PDU buffer 38 (as shown in Fig. 3B) is above the threshold λ . both step SA-7 and step SA-8 are performed. At step SA-7, the 20 concatenation timer 40 is re-initialized and at step SA-8 the concatenation timer 40 is started again.

[0030] At the reception of the "third" packet (Packet #3) shown in Fig. 3C. as step SA-11 the third packet is concatenated with the stored nine bytes of Packet #2 into the sole occupied PDU (PDU #2) of PDU buffer 38. In this case, the occupancy (in octets) 25 of PDU #2 is: $9+24 = 33$ octets plus 2 length indicators, and thus is less than the RLC PDU payload size. Depending on the level of "optimization" (delay versus capacity), PDU #2 can now either be transmitted with seven octets of padding, or stored for concatenation with yet another VoIP packet. With a threshold of 10%, it is determined at step SA-12 that PDU #2 should still remain in PDU buffer 38. Note that in this case, 30 concatenation timer 40 is not restarted. since no PDU was transmitted. The concatenation timer 40 is always PDU specific. With a higher threshold, like e.g. 20%. PDU #2 of Fig. 3C would be transmitted, causing more padding, but smaller delay.

[0031] For the downlink, the proposed functionality can be implemented in the RNC without changes to the standard. The uplink does require a standardized solution.

[0032] The techniques described herein provide a mechanism for tradeoff between delay performance and capacity for VoIP-like services. When active, the techniques
5 reduce the amount of padding on the radio link, which will increase capacity in the system.

[0033] Readout of the packets from PDU buffer 38 and from the node which hosts PDU buffer 38 can be to any suitable device or transmission line. Furthermore, the concept of "readout" can include moving PDUs from PDU buffer 38 to a suitable
10 transmission unit for conveying the PDUs out of the node or device which hosts the PDU buffer 38.

[0034] Although various embodiments have been shown and described in detail, the claims are not limited to any particular embodiment or example. None of the above description should be read as implying that any particular element, step, range, or
15 function is essential such that it must be included in the claims scope. The scope of patented subject matter is defined only by the claims. The extent of legal protection is defined by the words recited in the allowed claims and their equivalents. It is to be understood that the invention is not to be limited to the disclosed embodiment, but on the contrary, is intended to cover various modifications and equivalent arrangements.

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. A radio access network node comprising:
 - protocol data unit formation logic for segmenting incoming service data units to form protocol data units;
 - a protocol data unit buffer for storing one or more protocol data units;
 - a concatenation timer; and
 - a buffer readout mechanism for controlling readout of a protocol data unit in the protocol data unit buffer, the buffer readout mechanism being arranged to:
 - read out the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is less than a threshold; and
 - delay readout of the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is not less than the threshold,whereby when the extent of padding is not less than the threshold, the buffer readout mechanism is arranged to use the concatenation timer for determining a delay for readout of the protocol data unit from the protocol data unit buffer for providing an opportunity for at least a portion of a not-yet-arrived service data unit to be included in the protocol data unit prior to readout of the protocol data unit from the protocol data unit buffer, and thereby reduce padding in an outgoing protocol data unit.
2. The node of claim 1, wherein the buffer readout mechanism is arranged to delay readout of a protocol data unit from the protocol data unit buffer until either the current contents of the protocol data unit in the protocol data unit buffer have reached a predetermined fill level, or a predetermined time interval has expired.
3. The node of claim 2, wherein the predetermined time interval is adjusted for inter-arrival time of incoming service data units.

4. The node of claim 3, wherein the predetermined time interval is adjusted for inter-arrival times of incoming service data units by continuously filtered measurements on incoming traffic or by reading quality of service attributes.
5. The node of claim 2, wherein the service data units are twenty four byte voice over Internet Protocol, VoIP, packets which arrive with twenty millisecond inter-arrival times, wherein forty octets of payload are allowed in each protocol data unit, and wherein the predetermined time interval is set to between twenty milliseconds and forty milliseconds.
6. The node of any of claims 1 to 5, wherein the threshold is ten percent of a length of the protocol data unit.
7. The node of any of claims 1 to 6, wherein the concatenation timer is reset upon readout of a protocol data unit from the protocol data unit buffer.
8. A method of operating a radio access network node, the method comprising:
 - segmenting incoming service data units to form protocol data units;
 - storing one or more protocol data units in a protocol data unit buffer; and
 - controlling readout of a protocol data unit in the protocol data unit buffer, wherein said controlling readout comprises:
 - reading out the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is less than a threshold; and
 - delaying readout of the protocol data unit if an extent of padding that would occur upon transmission of current contents of the protocol data unit is not less than the threshold, thereby providing opportunity for at least a portion of a not-yet-arrived service data unit to be included in the protocol data unit prior to readout of the protocol data unit from the protocol data unit buffer, and thereby reduce padding in an outgoing protocol data unit.

9. The method of claim 8, further comprising delaying the readout of the protocol data unit from the protocol data unit buffer until either the contents of the protocol data unit in the buffer has reached a predetermined fill level, or a predetermined time interval has expired.
10. The method of claim 9, further comprising adjusting the predetermined time interval for inter-arrival time of incoming service data units.
11. The method of claim 9, further comprising adjusting the predetermined time interval for inter-arrival times of incoming service data units by continuously filtered measurements on incoming traffic or by reading quality of service attributes.
12. The method of claim 11, wherein the service data units are twenty four byte voice over Internet Protocol, VoIP, packets which arrive with twenty millisecond inter-arrival times, the method further comprising:
allowing forty octets of payload in each protocol data unit; and
setting the predetermined time interval to between twenty milliseconds and forty milliseconds.
13. The method of any of claims 8 to 12, wherein the threshold is ten percent of a length of the protocol data unit.
14. The method of any of claims 8 to 13, further comprising:
using a concatenation timer for determining the delay for readout of the protocol data unit from the protocol data unit buffer if the extent of padding is not less than the threshold; and
resetting the concatenation timer upon readout of the protocol data unit from the protocol data unit buffer.

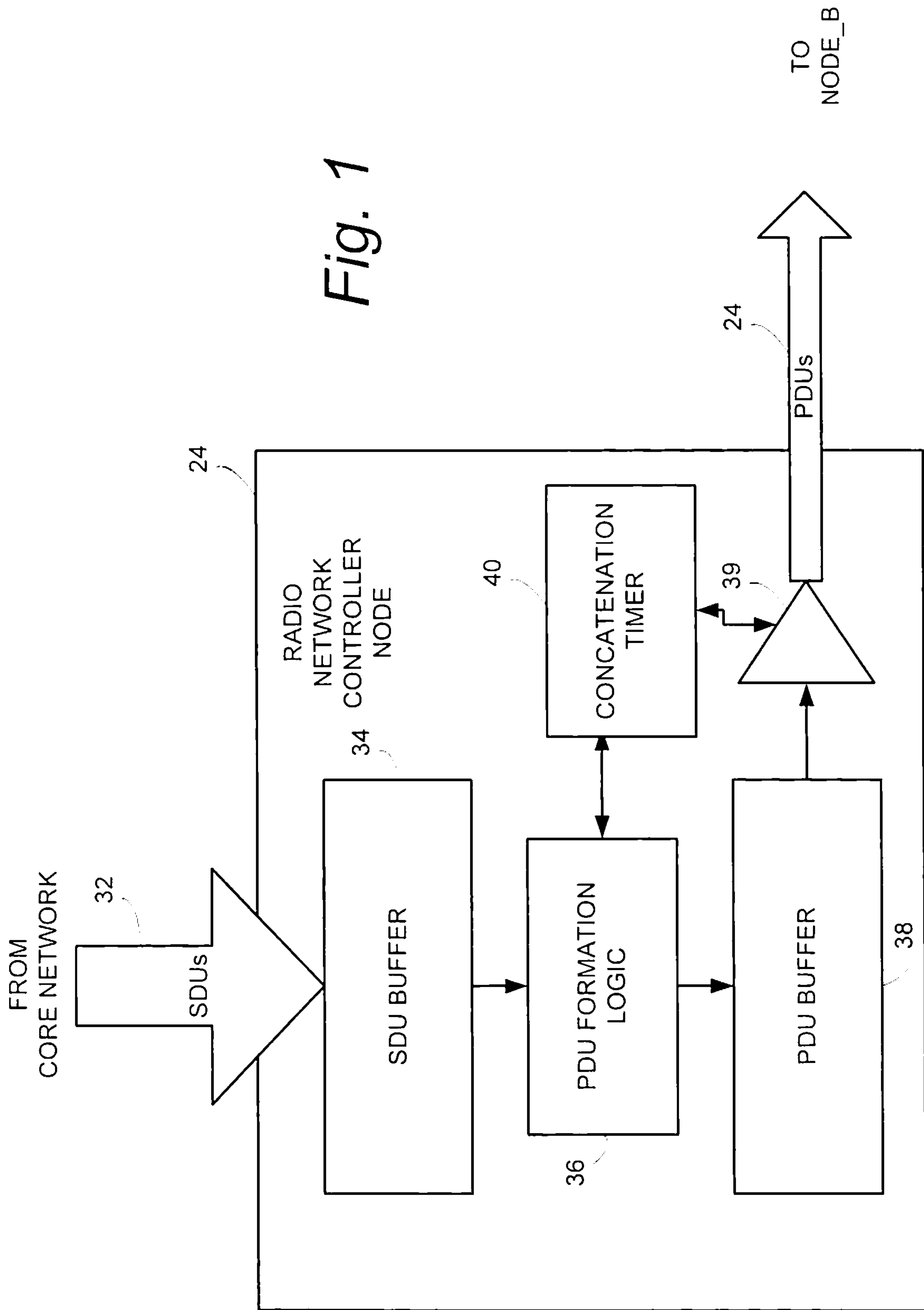


Fig. 1

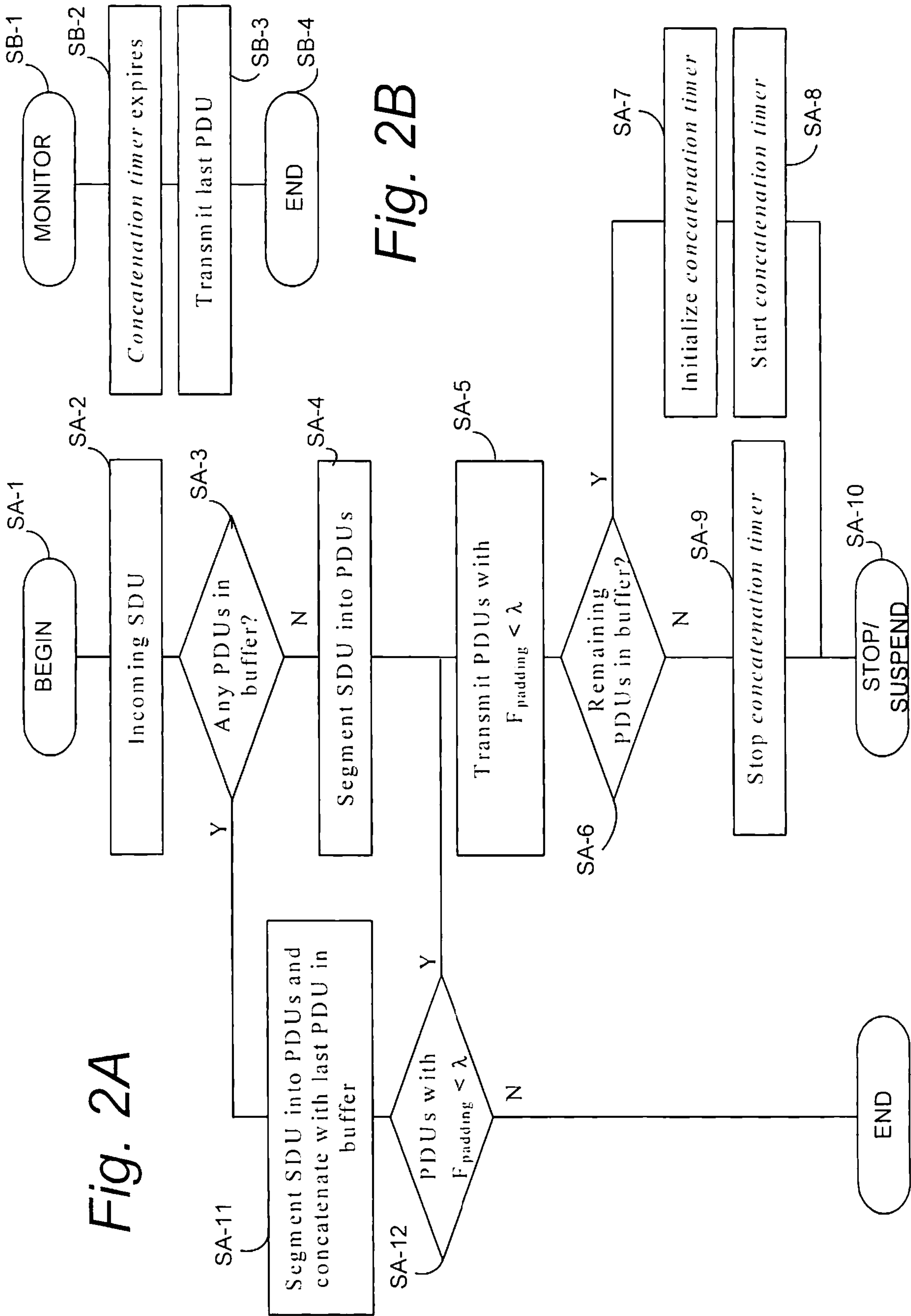


Fig. 2A

Fig. 2B

Fig. 3A

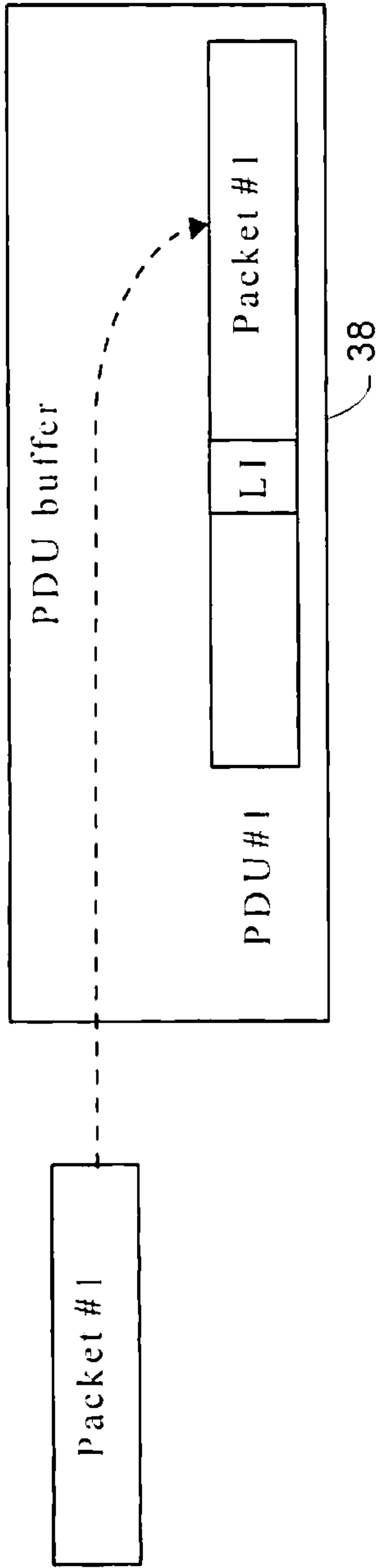


Fig. 3B

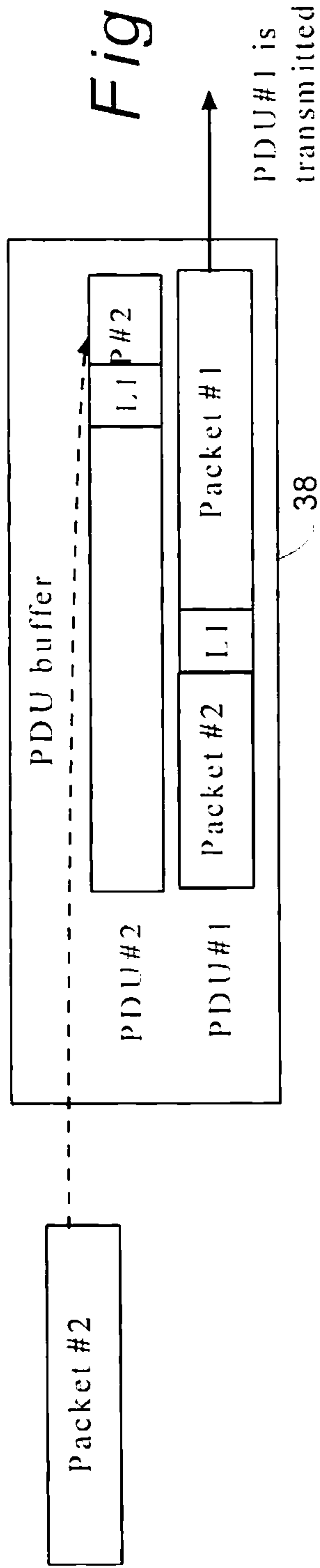


Fig. 3C

