

(19)



(11)

EP 2 517 419 B1

(12)

EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention of the grant of the patent:
25.05.2016 Bulletin 2016/21

(51) Int Cl.:
H04L 12/801 (2013.01) *H04L 12/875* (2013.01)
H04L 12/823 (2013.01) *H04L 12/841* (2013.01)

(21) Application number: **09802113.2**

(86) International application number:
PCT/EP2009/009266

(22) Date of filing: **24.12.2009**

(87) International publication number:
WO 2011/076239 (30.06.2011 Gazette 2011/26)

(54) **A METHOD OF SCHEDULING TRANSMISSION IN A COMMUNICATION NETWORK, CORRESPONDING COMMUNICATION NODE AND COMPUTER PROGRAM PRODUCT**

VERFAHREN ZUR ÜBERTRAGUNGSPLANUNG IN EINEM KOMMUNIKATIONSNETZ, ENTSPRECHENDER KOMMUNIKATIONSKNOTEN UND COMPUTERPROGRAMMPRODUKT

PROCÉDÉ PERMETTANT D'ORDONNANCER UNE TRANSMISSION DANS UN RÉSEAU DE COMMUNICATION, NOEUD DE COMMUNICATION ET PRODUIT-PROGRAMME D'ORDINATEUR CORRESPONDANTS

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

(43) Date of publication of application:
31.10.2012 Bulletin 2012/44

(73) Proprietor: **Telecom Italia S.p.A.**
20123 Milano (IT)

(72) Inventors:
• **SABELLA, Dario**
I-10148 Torino (IT)
• **CARETTI, Marco**
I-10148 Torino (IT)
• **ANDREOZZI, Matteo, Maria**
I-56126 Pisa (IT)
• **STEA, Giovanni**
I-56126 Pisa (IT)

(74) Representative: **Maccalli, Marco et al**
Maccalli & Pezzoli S.r.l.
Via Settembrini, 40
20124 Milano (IT)

(56) References cited:
WO-A1-2008/059000 WO-A1-2009/077852
US-A1- 2008 008 096

- **MATTEO MARIA ANDREOZZI ET AL:** "Flexible scheduling for Real-Time services in High-Speed Packet Access cellular networks" WIRELESS CONFERENCE, 2009. EW 2009. EUROPEAN, IEEE, PISCATAWAY, NJ, USA, 17 May 2009 (2009-05-17), pages 189-193, XP031583298 ISBN: 978-1-4244-5935-3
- **MORTON EDITOR ATOEB_AMPAMP A C ET AL:** "Revised text of G.1020 adding RTCP-XR parameters and other clarifications; D 168" ITU-T DRAFT STUDY PERIOD 2005-2008, INTERNATIONAL TELECOMMUNICATION UNION, GENEVA ; CH, vol. STUDY GROUP 12, 5 June 2006 (2006-06-05), pages 1-33, XP017406229
- **LUIGI ATZORI ET AL:** "Playout buffering in ip telephony: a survey discussing problems and approaches" IEEE COMMUNICATIONS SURVEYS, IEEE, NEW YORK, NY, US, vol. 8, no. 3, 1 July 2006 (2006-07-01), pages 36-46, XP011144707 ISSN: 1553-877X
- **A. Bacioccola, C. cicconetti, G. Stea:** "User-level performance evaluation of VoIP using ns-2" Proceedings of the 2nd international conference on Performance evaluation methodologies and tools , NSTools 07 vol. ValueTools no 321, 20, 22 October 2007 (2007-10-22), XP002578866 Nantes, France ISBN: 978-963-9799-00-4 Retrieved from the Internet:
URL :<http://portal.acm.org/citation.cfm?id=1345289> [retrieved on 2010-04-20]

Note: Within nine months of the publication of the mention of the grant of the European patent in the European Patent Bulletin, any person may give notice to the European Patent Office of opposition to that patent, in accordance with the Implementing Regulations. Notice of opposition shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

EP 2 517 419 B1

DescriptionField of the invention

- 5 **[0001]** This disclosure relates to techniques for scheduling data transmission in communication networks.
[0002] This disclosure has been developed with attention paid to its possible use in scheduling data packets of a Voice over IP (VoIP) data stream in a point-to-multipoint communication node.

Description of the related art

- 10 **[0003]** The Internet has originally been designed having in mind reliable packet transfer. In fact, all the lower layers of the protocol stack include mechanisms for guaranteeing reliable communication, such as sequence numbers, timeouts, checksums (such as Cyclic Redundancy Check), Forward Error Correction (FEC), Automatic Repeat-reQuest (ARQ), etc. This is due to the Internet being initially conceived as a computer data network, wherein correctness and integrity
15 of computer data are key attributes.
- [0004]** Nowadays, the Internet is well established as a global, multi-service infrastructure, providing an ever-growing support for real-time services. Packetized voice, such as Voice over IP (VoIP), is among the most important emerging real-time applications for the Internet, acting as a major driver for the deployment of advanced Quality of Service (QoS) solutions.
- 20 **[0005]** Despite the massive presence of real-time traffic, for which timeliness is as important as integrity, few Quality-of-Service (QoS) solutions actually use time as a parameter for taking decisions, except at an application level. This is probably due to the historically well-known problem of clock synchronization across the Internet, which makes it difficult to have globally synchronized clocks at two remote points.
- [0006]** The lack of reliable clock synchronization is often a problem for performance measurement, where long measurement intervals (days to weeks) and high precision are required (thus making even small clock skews extremely harmful).
- [0007]** Progresses in clock synchronization are currently being made, which enable to achieve sub-millisecond accuracy in such a long term. However, voice conversations last for minutes to a few hours, and synchronization errors in the order of few milliseconds hardly generate any noticeable effect in human perception.
- 30 **[0008]** The accuracy required for keeping clocks synchronized for a couple of hours with up to few milliseconds of absolute difference is already achievable at little cost with available techniques, such as from Global Position System (GPS) chips, which are inexpensive and already mounted in most portable devices, the same result being achieved also by proprietary solutions available in the Universal Mobile Telecommunications System (UMTS) network (where much tighter synchronization is indeed a major requirement) or the well-known Network Time Protocol.
- 35 **[0009]** Figure 1 schematically represents a typical VoIP scenario in which two users U1 and U2 communicate via two user terminals UE1 and UE2 which are connected to a common network N (e.g. the internet).
- [0010]** At the sender side (i.e. UE1), the first component is an encoder 100, which periodically samples the voice signal of the first user U1 (i.e. the speaker). A wide variety of encoders are available, representing different trade-off points in the balance between quality and bandwidth consumption. Typically, encoders are either sample-based or frame-based.
40 The former encoders (e.g., based on the G.711 codec) encode individual speech samples periodically, while the latter encoders (e.g., based on the G.729 codec) group a certain number of samples within a time window (i.e., a frame) of some milliseconds. For that reason, frame-based encoders often achieve higher compression rates and smaller data rates, though their encoding/decoding complexity is usually higher.
- [0011]** The generation of speech frames can either occur at periodic intervals, or, more commonly, be modulated by Voice Activity Detection (VAD). VAD capitalizes on the natural alternation of talkspurt and silence periods in a single (unidirectional) stream of a bidirectional conversation. During silence periods, either no speech frames are produced at all, or they are produced at a reduced rate and/or using a reduced number of bits, so as to convey some comfort noise to the listener. Comfort noise permits a conversation party to feel the liveliness of the other one, and can be generated also locally. For instance, the G.711 codec usually generates 172-byte packets during talkspurts and does not send any
50 information during silence periods. On the other hand, the Global System for Mobile Communications (GSM) Adaptive Multi-Rate (AMR) codec normally generates 31-byte payload packets during talkspurts, and either 0-byte synchronization information or 5-byte comfort noise packets during silence periods.
- [0012]** A number of speech frames may also be multiplexed into the same packet payload by means of a multiplexer 102, so as to reduce the overhead of transport, network and Media Access Control (MAC) headers, though at the expense of increasing the transmission delay. This can also be done to increase the level of redundancy, so as to conceal losses. The level of speech multiplexing may also vary over time, according to network conditions.
- 55 **[0013]** Finally, the VoIP payload is encapsulated into data packets, such as RTP/UDP/IP packets, i.e. packets being communicated at the application layer via the Real-time Transport Protocol (RTP), at the transport layer via the User

Datagram Protocol (UDP), and at the internet layer via the Internet Protocol (IP).

[0014] At the receiver side (i.e. in the terminal UE2), speech frames are de-multiplexed by means of a demultiplexer 104 and fed into a playout buffer 106. The playout buffer 106 causes speech frames to be decoded at the same interval at which they were generated by the encoder. For that reason, it may be necessary to re-order, delay or even drop speech frames if they arrive after their expected playback time.

[0015] Playout buffers may be fixed or adaptive. Fixed buffers assume the network delay to be constant during a conversation, and therefore delay the first packet of a talkspurt of a fixed amount of time. Adaptive buffers, endeavour to dynamically adapt the playback point to the changing network conditions, normally on a per-talkspurt basis (although some may also vary the playout delay within a talkspurt, at the risk of generating spurious frequency effects).

[0016] The playout buffer delivers speech frames to a decoder 108, which actually playbacks them to the user U2 (i.e. the listener). Some decoders may implement Packet Loss Concealment (PLC) techniques, which allow missing speech frames to be somewhat reconstructed by interpolating surrounding frames that were received correctly. While PLC techniques can mask only a limited number of losses, they effectively reduce the impairment due to loss as perceived by a listener.

[0017] The evaluation of VoIP conversational quality has been the subject of extensive investigation. The most widely used evaluation framework for VoIP is the so-called E-model, standardized by Telecommunication Standardization Sector (ITU-T) of the International Telecommunication Union (ITU). This evaluation framework involves computing a predictive estimation of the subjective quality of the packetized voice from transmission parameters.

[0018] The output of an E-model computation is a scalar number, called the "R factor", computed as a function of delays, packet loss, equipment impairment factors, and user quality call expectation:

$$R = R_0 - I_s - I_d - I_{e,eff} + A \quad (1)$$

wherein R_0 is the basic signal-to-noise ratio (received speech level relative to circuit and acoustic noise), I_s accounts for the impairments which occur with the voice signal, I_d sums all impairments due to delay and echo effects, $I_{e,eff}$ is the effective equipment impairment factor, taking into account the codec and its tolerance to random packet losses.

[0019] Furthermore, A is a "bonus" factor that models the user expectation of the technology employed. For instance, the value of A is greater in satellite networks than in classical circuit-switched networks, because user expectations in satellite networks are lower than those in wired networks. The typical range for the A factor is [0, 20] and exemplary values as proposed by the ITU are reported in the following table:

| Communication System | A factor |
|--|----------|
| Wired phone | 0 |
| Cellular in building | 5 |
| Cellular in moving vehicle | 10 |
| Access to hard-to-reach geographical zones (many satellite hops) | 20 |

[0020] Finally, $I_{e,eff}$ can be computed as:

$$I_{e,eff} = I_e + (95 - I_e) \cdot P_{p1} / (P_{p1} + B_{p1}) \quad (2)$$

where I_e is an equipment impairment factor, which is used to characterize the behaviors of the codec with a low bit rate, P_{p1} is the packet loss probability, and B_{p1} is the codec packet loss robustness factor.

[0021] Once an R factor has been obtained, it may be mapped directly to an estimated Mean Opinion Score (MOS). For example, the specification ITU-T G107 (ITU-T Recommendation G.107 (04-2009), "The E-model, a computational model for use in transmission planning") describes a possible relationship between the R factor and the estimated MOS. The Mean Opinion Score may thus be used to improve Quality of Experience (QoE) or QoS solutions. Usually such optimizations are performed at the application level, where QoE may be measured.

[0022] For example, the article from L. Atzori, M.L. Lobina, and M. Corona, "Playout buffering of speech packets based on a quality maximization approach", IEEE Transactions on Multimedia, Vol.8 No.2, pages 420-426, 2006, proposes the use of an optimal playout buffer at the receiver in order to predict the playout delay for an incoming talkspurt. Specifically, the playout instant for the next talkspurt is set to the one that would have maximized the quality of the last N talkspurts.

[0023] Also, the article from A. Bacioccola, C. Cicconetti, and G. Stea, "User level performance evaluation of VoIP using ns-2", Proceedings of STTOOLS'07, First International Workshop on Network Simulation Tools, Nantes (FR), October 22, 2007, describes a possible optimal playback algorithm.

[0024] Z. Qiao, L. Sun, N. Heilemann, and E. Ifeachor, "A new method for VoIP quality of service control use combined adaptive sender rate and priority marking", IEEE ICC 2004, Paris, France, June 20-24, pages 1473-1477, propose a QoS control scheme to adapt the rate of a GSM AMR codec to adapt the sending rate to the current network conditions. The adaptation is based on the MOS measured by the VoIP decoder, which reports feedback information to the VoIP encoder via Real-Time Control Protocol (RTCP) messages.

[0025] Document WO2008/059000 discloses a method for scheduling the transmission of data packets to a receiving device by appropriately assigning priority levels to data packets prior their transmission.

[0026] Document US 2008/0008096 discloses a system for managing playout time in a packet communication network, including a delay estimator module adapted to calculate packet delay at one received packet data.

[0027] M. A. Andreozzi et al., "Flexible Scheduling for Real-Time Services in High-Speed Packet Access Cellular Networks", Wireless Conference 2009, 17 May 2009, pages 189-193 discloses a Channel-Aware ad Real-Time scheduler for UTS High-Speed Packet Access Networks.

Object and summary of the invention

[0028] The inventors have noted that the optimization of QoE at an application level based on feedback information has several drawbacks. For example, such feedback-based techniques make the application unduly complex, and either require synchronization or rely on round-trip times, which are not symmetric and therefore are much less accurate than one-way delays.

[0029] The need is therefore felt for improved solutions that may dispense with such drawbacks, and may permit effective allocation of network resources, e.g. in scheduling downlink voice flows in a centralized (wired or wireless) point-to-multipoint network.

[0030] According to the present invention, that object is achieved by means of a method having the features set forth in the claims that follow. The invention also relates to a corresponding communication node, as well as a computer program product, loadable in the memory of at least one computer and including software code portions for performing the steps of the method of the invention when the product is run on a computer. As used herein, reference to such a computer program product is intended to be equivalent to reference to a computer-readable medium containing instructions for controlling a computer system to coordinate the performance of the method of the invention. Reference to "at least one computer" is intended to highlight the possibility for the present invention to be implemented in a distributed/modular fashion.

[0031] The claims are an integral part of the disclosure of the invention provided herein.

[0032] Various embodiments described herein may be used to schedule the transmission of data streams from a communication node (e.g. a basestation) towards a receiver (e.g. a mobile terminal), wherein the data packets are intended for playout via a reproduction buffer at the receiver.

[0033] In various embodiments, the data packets are arranged in the node in a scheduling queue and dropped from the scheduling queue if their sojourn time in the queue exceeds a given drop deadline.

[0034] In various embodiments, the reproduction buffer of the receiver is emulated at the node in order to determine respective expected playout instants for the data packets. Subsequently, the drop deadlines of the data packets are assigned to the data packets as a function of the respective expected playout instants determined via the reproduction buffer as emulated at the node.

[0035] Various embodiments are applicable to several access technologies, such as to cellular networks including UMTS Long Term Evolution (LTE) or High-Speed Downlink Packet Access (HSDPA).

[0036] Various embodiments rely on the recognition that in these cellular network technologies, absolute time is usually known via out-of-band network synchronization procedures.

[0037] Various embodiments are also applicable in broadband wireless networks with centralized control at the MAC level, such as 802.11e Hybrid Coordination Function (HCF) Centralized Control Access (HCCA), 802.16, or TDMA systems (e.g., 802.11 WLANs with tweaked TDMA drivers), when an absolute time reference is added.

[0038] Various embodiments are also applicable to wired networks, where a central switch sends data packets to clients.

[0039] Various embodiments rely on the recognition that the Mean Opinion Score (MOS) may be used directly at a communication network level.

[0040] Various embodiments are applied to downlink flows, which rely on a prediction of the (loss and delay) behavior of the downstream segment of the end-to-end path (therein including the application part).

[0041] Various embodiments involve taking scheduling decisions in a cross-layer approach, based on (absolute) timestamps, e.g. in the form of RTP timestamps, which have been introduced in the data-packets at the data source. Such absolute timestamps may be derived with various means, including GPS synchronization, UMTS network syn-

chronization, network protocols and/or time servers.

[0042] Various embodiments exploit the absolute time for scheduling downlink flows in a point-to-multipoint network (e.g., an LTE or UMTS cell, or a TDMA network), i.e. a communication network wherein the network resources are shared among a plurality of users or terminals.

5 **[0043]** The exemplary embodiments described herein primarily refer to downlink voice flows, where:

- there exists a well-established means to estimate the (subjective) quality perceived by the user from (objective) end-to-end network-level measurements, such as loss and delay, i.e. the E-Model or Mean Opinion Score (MOS); and
- the behavior of a voice application at the receiver side (specifically, its playout buffering mechanism) is fairly predictable, so that it is possible to infer the (non negligible) contribution of the voice receiver to loss and delay.

[0044] Those of skill in the art will appreciate that various embodiments may also be applied to other real-time traffic streams, such as video streams. For that purpose a framework similar to the E-Model for video traces can be developed.

15 **[0045]** In various embodiments, each packet is assigned a deadline equal to the presumed or expected playback instant; by so doing, late packets, i.e. those which would be dropped at the receiver, may be dropped directly at the scheduler, thus saving bandwidth and optimizing the network resource allocation.

[0046] In various embodiments, the scheduler may also delay early packets until their actual playback point without affecting user perception, thus increasing the chance that packets with tighter deadlines may be delivered to the receiver in time.

20 **[0047]** In various embodiments, the playback point at the receiver is estimated by emulating for the connection a simplified receiver and/or playout buffer ("reproduction buffer").

[0048] For example, in the case of a VoIP stream, also a simplified adaptive playout buffer may be emulated, i.e. a buffer whose playback point is varied dynamically for each talkspurt. For example, at the end of a talkspurt, computations allow to infer the optimal-a-posteriori playback instant, i.e. the one which would have guaranteed the highest MOS for that talkspurt. The history of those optimal playback instants may then be used to infer the new presumed playback instant for the subsequent talkspurt.

25 **[0049]** In various embodiments, in order to compute the optimal playback instants a posteriori, the E-Model formulas are used. These take into account mouth-to-ear delay, which can be computed if the packet generation and (presumed) playback times may be determined based on clock synchronization. For that reasons, packets may be timestamped at the source applications with absolute times sampled from an accurate clock, and the scheduler is aware of the absolute time when packets are delivered at the receiver, again by reading an accurate absolute clock. In various embodiments, the RTP protocol, in particular the RTP timestamp field, is used for that purpose: for example, the RTP header may be read twice per packet, i.e. at packet arrival, when sequence numbers and packet timestamps may be used to compute the packet deadline, and at packet departure, when the scheduler may estimate the mouth-to-ear delay by comparing the wall clock time (i.e. the absolute time when the packet leaves the queue of the scheduler) to the packet timestamp.

30 **[0050]** The embodiments herein have several advantages, such as:

- they are inexpensive, because the space cost of the reproduction buffer emulation is quite affordable, as the whole implementation framework requires usually less than 100 byte per flow;
- time overhead is negligible, because usually a constant number of operations is required for each flow;
- the most complex computation (i.e., computing the optimal a posteriori playback instant) involves typically just few memory accesses, and may be completed during silence periods, i.e. several hundreds of milliseconds;
- the resulting arrangement is robust: even rather coarse clock errors (in the order of ± 10 ms) are tolerated without any appreciable quality degradation, so that temporary loss of synchronization (even for a long time) hardly affects the scheduling process;
- the resulting arrangement may work in conjunction with any deadline-based scheduler, such as the well-known Earliest Due Date (EDD), or for wireless networks with user-dependent and time-dependent channel conditions with a scheduler that takes into account both the deadlines and channel state;
- the framework may be applied to various communication technologies, including i.a. communication scenarios where the following conditions are met:
 - a) the packet source and the scheduling node are synchronized (e.g., through GPS),
 - b) each packet contains a timestamp,
 - c) the scheduling component is able to read the packet timestamps at arrival and the absolute time at their departure, and
 - d) the downstream segment of the path only includes a playout buffer plus additional delays that can be measured online (e.g., delay due to H-ARQ retransmissions) and/or estimated (e.g., propagation and processing delays).

[0051] Accordingly, various embodiments may be implemented in any access network with a coordinated point-to-multipoint scheduling under the above hypotheses.

Brief description of the annexed representations

[0052] The invention will now be described, by way of example only, with reference to the enclosed representations, wherein:

- Figure 1 has already been described in the foregoing;
- Figure 2 shows an exemplary communication scenario;
- Figure 3 is a block diagram of an embodiment; and
- Figure 4 is a timing diagram of operation of an embodiment.

Detailed description of embodiments

[0053] In the following description, numerous specific details are given to provide a thorough understanding of embodiments. The embodiments can be practiced without one or more of the specific details, or with other methods, components, materials, etc. In other instances, well-known structures, materials, or operations are not shown or described in detail to avoid obscuring aspects of the embodiments.

[0054] Reference throughout this specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. Thus, the appearances of the phrases "in one embodiment" or "in an embodiment" in various places throughout this specification are not necessarily all referring to the same embodiment. Furthermore, the particular features, structures, or characteristics may be combined in any suitable manner in one or more embodiments.

[0055] The headings provided herein are for convenience only and do not interpret the scope or meaning of the embodiments.

[0056] Figure 2 shows an exemplary embodiment, wherein a voice stream organized in data packets is transmitted between two LTE mobile devices UE1 and UE2 by means of the RTP protocol.

[0057] In the embodiment considered, the first mobile device UE1 sends data to a basestation BS (i.e. the eNodeB in a LTE network) with which the mobile device UE2 is associated via a communication network N.

[0058] Each data packet may comprise n voice frames and each packet may be transmitted via m Protocol Data Units (PDU). Moreover, the value of m may be different for different connections, and may change over time, e.g. when the feedback from the H-ARQ process indicates that the number of retransmission has significantly increased/decreased. The value of m may also be fixed and preconfigured at setup.

[0059] Some simplified assumptions will be made at first in order to facilitate understanding the description; further reading the description will however make it clear that such assumptions are in no way mandatory. Specifically, it will be assumed that each RTP packet comprises only one voice frame, and that voice packets are not fragmented at the eNodeB, so that one PDU carries one voice frame.

[0060] In the embodiment considered, packets are timestamped with the frame generation time at creation, wherein $g_{i,k}$ is the generation time of the k^{th} packet in the i^{th} talkspurt, with $i, k \geq 0$, and $a_{i,k}$ is the arrival time at the eNodeB.

[0061] Packets may accumulate a variable delay in the network N, and some packets may also be dropped and/or delivered out of sequence.

[0062] Figure 3 shows a block diagram of a possible implementation of the basestation BS in accordance with the invention.

[0063] In the embodiment considered, when packets arrive at the basestation BS at a time $a_{i,k}$, they are assigned a deadline $d_{i,k}$ at a control module 200.

[0064] The packets, coming out from control module 200 with the assigned deadline $d_{i,k}$, are then scheduled by a scheduler S, which may take into account their deadlines so that scheduler S may drop packets if their sojourn time in the eNodeB queue exceeds their deadline $d_{i,k}$, because these packets are likely to be dropped at the receiver buffer, hence transmitting them would just waste radio resources.

[0065] In the embodiment considered, the scheduler S comprises a plurality of queues SQ where the data packets of the respective data streams are stored. For instance, the queues SQ may be implemented by means of First-In First-Out (FIFO) memories. Subsequently, a scheduling module SM decides which packet will be transmitted or dropped. For example, in the embodiment considered, a packet will be dropped if the scheduled transmission time of the packet $t_{i,k}$ exceeds the deadline $d_{i,k}$ of the packet, i.e. $t_{i,k} > d_{i,k}$.

[0066] In various embodiments, a packet selected for transmission by the scheduler S at time $t_{i,k}$ is taken care of by H-ARQ processes, and, after a measurable delay $d_{i,k}$, it is available for playout at the mobile device UE2.

[0067] In the embodiment considered, the delay $d_{i,k}$ includes possible H-ARQ retransmissions of that packet and an

estimate of the physical delays in the air interface and in the mobile node, which usually are substantially constant.

[0068] In various embodiments, the estimated delay dI is set as follows:

$$dI = T_{proc} + T_{OW} + (m - 1) \cdot T_{H-ARQ} \quad (3)$$

where T_{proc} represents an estimate of the processing delay at the physical and MAC layers at the receiver, T_{OW} is the time to send a PDU through the downlink segment (i.e., the time it takes from an H-ARQ process to deliver the packet to the receiver, therein excluding the ACK/NACK generation and reporting), T_{H-ARQ} is the time for an H-ARQ transmission cycle and m is the number of transmissions taken into account for a packet.

[0069] In addition to the delay $dI_{i,k}$, the only further delay that the packet undergoes before the voice sample is played out at the time $p_{i,k}$ is the one of the playout buffer at the receiver.

[0070] Therefore, the arrival time $q_{i,k}$ at the receiver playout buffer is:

$$q_{i,k} = t_{i,k} + dI_{i,k} \quad (4)$$

and the overall network delay of the frame i may be defined as:

$$\delta_{i,k} = q_{i,k} - g_{i,k} \quad (5)$$

[0071] In various embodiments, the sequence of $\delta_{i,k}$ is fed as an input to an emulated playout buffer B which allows to determine the deadline $d_{i,k}$. Specifically, the emulated optimal playout buffer B emulates the playout buffer of the mobile device UE2, and its purpose is to identify a posteriori (i.e., once talkspurt i is over) what the optimal playout instant po_i should have been, i.e. the one that would have warranted the highest MOS for that talkspurt.

[0072] In the embodiment considered, the deadline $d_{i,k}$ is computed as follows:

$$d_{i,k} = po_i - (a_{i,k} - g_{i,k}) - dI \quad (6)$$

where po_i is the estimated playout delay at the receiver for the i^{th} talkspurt, wherein the playout delays is defined as the interval between the generation and playout time of any successfully played packet in the talkspurt, and dI is the estimate of the delay due to the radio interface and processing at the receiver (other than that generated by its playout buffer).

[0073] The history of past optimal playout delays po_j , with $0 \leq j \leq i$, is then used to infer the new playout delay po_{i+1} for the incoming talkspurt.

[0074] In various embodiments, a non-causal optimal playback algorithm is used. More specifically, in various embodiments, the playout buffer waits for the whole set of frames of a given talkspurt to arrive at the scheduler and be transmitted (or dropped), and then selects the playback delay - based on the pattern of arrivals, transmissions and drops - according to which the best possible speech quality would have been achieved. The purpose of including such an optimal playout buffer is to predict what a clever adaptive playout buffering algorithm would do. For example, this permits to delay early packets and/or to drop packets that have a low chance of being played out.

[0075] In various embodiments, the control module 200 and the emulated buffer B are implemented by means of software code portions, which are executed by a processor in the basestation BS. However, the modules may also be implemented in the form of a dedicated digital circuit.

[0076] Figure 4 shows an exemplary scenario including a transmission phase 400, a buffering phase 402 and a playout phase 404 of a talkspurt with five VoIP frames being generated with equal time difference P at the time instants $g_{i,1}$, $g_{i,2}$, $g_{i,3}$, $g_{i,4}$, and $g_{i,5}$.

[0077] In the embodiment considered, the k -th frame of talkspurt i is (virtually) passed to the decoder at the time $p_{i,k}$, i.e. played out, and the playout delay po_i for all the packets in a talkspurt which are actually played are assumed to be constant, i.e., $p_{i,k} - P_{i,j} = g_{i,k} - g_{i,j}$, for each k, j that are actually played.

[0078] Without loss of generality, it may be assumed that $q_{i,k} = \infty$ for each frame i that is lost due to the network. Similarly, it may be assumed that the playout buffer of the receiver discards directly all frames that are received too late, i.e. packets with $\delta_{i,k} > po_i$ which implies that all frames with $\delta_{i,k} = q_{i,k} = \infty$ are discarded, regardless of the value of po_i . Frames that are discarded by the playout buffer contribute to the loss rate L_i which is defined as the ratio between the number of discarded frames and the number of frames in the talkspurt.

[0079] In the example of Figure 4, $L_i = 1/5$ because all frames but the first one are received "in time".

[0080] Therefore, a degree of freedom of the playout buffer B is the playout delay po_i .

[0081] In various embodiments, an optimal playout delay $po_i^{opt} \in \{\delta_{i,k}\}$ is determined based on a MOS function which considers QoS and/or QoE criteria. For example, in the embodiment considered, the value of po_i^{opt} is calculated as the value of possible playout delays that maximizes the R factor. However, those of skill in the art will appreciate that the actual formula used to compute the optimal playout delay po_i^{opt} may depend on the specific codec used.

[0082] In various embodiments, the value po_i^{opt} is calculated by merging equations (1) and (2) described herein, and using for the impairments I_d equation (2) of the article from Atzori et al.:

$$R = R_0 - I_s - [0.024 \cdot po_i + 0.11 \cdot (po_i - 177.3) \cdot \mathbf{1}_{\{po_i \geq 177.3\}}] - I_e - (95 - I_e) \cdot P_{p1} / (P_{p1} + B_{p1}) + A \quad (7)$$

[0083] In various embodiments, the packet loss probability P_{p1} in equation (7) is substitute with:

$$P_{p1} = \frac{\sum_{k=1}^{N_i} \mathbf{1}_{\{\delta_{k,i} > po_i\}}}{N_i} \quad (8)$$

wherein N_i is the number of packets in the i -th talkspurt (which may be measured, e.g., by inspecting the sequence numbers), and the notation $\mathbf{1}_{(x)}$ means that the result is 1 if x is true, and otherwise 0. Accordingly, the summation provides the count of packets in the i -th talkspurt that have a delay $\delta_{k,i}$ being larger than po_i , i.e. those that were dropped either within the network (i.e. $\delta_{k,i} = +\infty$) or at the playout buffer itself (i.e. $\delta_{k,i} > po_i$).

[0084] Accordingly, equation (7) may be written as:

$$R = R_0 - I_s - [0.024 \cdot po_i + 0.11 \cdot (po_i - 177.3) \cdot \mathbf{1}_{\{po_i \geq 177.3\}}] - I_e - (95 - I_e) \cdot \left(\frac{\sum_{k=1}^{N_i} \mathbf{1}_{\{\delta_{k,i} > po_i\}}}{N_i} \right) / \left(\frac{\sum_{k=1}^{N_i} \mathbf{1}_{\{\delta_{k,i} > po_i\}}}{N_i} + B_{p1} \right) + A \quad (9)$$

[0085] Those of skill in the art will appreciate that the above equations are just an exemplary implementation, and that other calculations of the R factor may be used. In any case, as already mentioned in the foregoing, po_i^{opt} may be calculated as the value of possible playout delays that maximizes the R factor:

$$\delta_{i,k} > po_i^{opt} = \arg \max_{po_i} \{R\} \quad (10)$$

[0086] Usually, the MOS is obtained via a non-increasing function of po_i and L_i , and L_i itself is a non-increasing function of po_i . Accordingly, there will exist an optimal value po_i^{opt} that permits to maximize the R factor for the talkspurt. For example, such an optimal value po_i^{opt} may be computed through a search in the set of possible network delays in a talkspurt $\{\delta_{i,k}\}$.

[0087] Accordingly, the playout buffer is adaptive, since it sets the playback point on a per-talkspurt basis, while it is non-causal, because the delay of all speech frames have to be known before selecting the playback point.

[0088] In various embodiments, the optimal playout buffer is used as follows:

- after talkspurt i , the optimal playout delay po_i^{opt} is computed,
- then, an exponential average is used to infer a likely playout delay for the next talkspurt ($i+1$):

5

$$po_{i+1} = \alpha \cdot po_i + (1 - \alpha) \cdot po_i^{opt}, \text{ with } 0 < \alpha < 1. \quad (11)$$

10

- po_{i+1} is then used to set the deadlines for the packets in the next talkspurt according to equation (6).

15

[0089] Thus, playout instants are selected based on the past history. This leaves the problem of selecting the playout delay for the first talkspurt, since no estimate is available at that time. To overcome this problem, a default value may be selected for the initial value, wherein the initial value may also be based on information regarding the call endpoints. For example, a typical default value for the initial playout delay may be 200 ms. Moreover, also the weight parameter α may be optimized heuristically.

[0090] As mentioned in the foregoing, a RTP packet may also include n voice frames, with $n \geq 1$.

[0091] Various embodiments assume that the RTP timestamp refers to the generation time of either the first (or the last voice frame). In that case, the generation time of each frame may be computed e.g. based on the codec period.

20

[0092] In various embodiments, the packet deadline and discard time (which were assumed to be equal so far) are considered separately. For example, the deadline may be computed based on the generation instant of the first voice frame, while the discard time of the packet may be set to the presumed playout instant of the last voice frame. This means that the packet may be transmitted as long as it still carries some useful content.

25

[0093] Accordingly, when inserting frames in the playout buffer, each frame may be associated to the correct generation time. Furthermore, depending on how large RTP packets are and on how the PDU length is set in the LTE cell, one packet may end up being fragmented into several PDUs at the eNodeB. In this case, all the PDUs may be associated to the same deadline and discard time, and the network delay of the packet may be set based on the transmission time of the last PDUs of a packet.

[0094] Due to the fact that the network may drop packets and is not assumed to preserve their sequence, it might be possible that the first packet of a talkspurt i , which arrives at time a_{i,k_0} at the basestation BS is in fact the k_0^{th} , with $k_0 \geq 0$.

30

[0095] Accordingly, various embodiments provide for packets to carry sequence numbers. In that way, the scheduler may determine that k_0 other packets are missing, independently from their generation time.

[0096] In various embodiments, the scheduler computes thus a lower bound on the network delay of the previous packets directly upon arrival of packet k_0 . These packets will thus inevitably have an earlier deadline than packets already queued in the system.

35

[0097] In various embodiments, the scheduler tries to realign the PDUs. For example, if a packet $k < k_0$ arrives, the associated PDU(s) may be placed in the buffer before later packets.

[0098] As this may not be feasible for schedulers using FIFO memories as buffers, the associated PDU(s) may be placed in FIFO order, however with their correct deadline and discard time, computed according to equation (6). In that case, deadlines are non-monotonically increasing for the same flow. However, this will not affect scheduling, and, if later packets are transmitted well before their deadline, there may still be room for transmitting the earlier ones.

40

[0099] The exemplary solution described in the foregoing distinguishes talkspurts. For most - if not all - codecs, the packet generation period is constant, which makes such a task easy. In fact, RTP packets carry both timestamps and sequence numbers. Thus, for those codecs, which do not generate and send any information during silence periods, it is possible to observe that any two packets which arrive consecutively at the basestation BS, carrying sequence numbers k_1, k_2 , and with generation timestamps such that talkspurts ts with $ts(k_2) > ts(k_1) + P \cdot (k_2 - k_1)$ certainly belong to different talkspurts. In practical cases, a safety margin (e.g., one period) may be added to the right-hand side of the above inequality to account for jittery sources.

45

[0100] Conversely, silence periods should be estimated only a posteriori, i.e. when a new talkspurt begins, because a burst of consecutive losses may be mistaken for a silence period.

50

[0101] For those codecs which do send reduced information during silence periods, silence period may be detected immediately by examining the packet size.

[0102] Computation of the optimal playout delay may be performed by sorting the network delays, and computing the R factor for each possible network delay. Assuming M packets in a talkspurt, this would lead to a time overhead $O(M)$. Although M may not be expected to be a very large number in practical cases (the average talkspurt length being in the order of 1s), the time overhead may also be limited without relying on this assumption.

55

[0103] For example, the network delays may be quantized at a rather coarse resolution, e.g., 10ms, which usually does not introduce appreciable quality degradation.

[0104] Moreover, instead of storing all possible network delays, a limited number of $B + 1$ integer delay counters may be used.

[0105] For example, in various embodiments, the counter i , with $0 \leq i \leq B-1$, is related to a network delay range being equal to $[i \cdot Q + C, (i + 1) \cdot Q + C[$, wherein Q is the quantization interval and C is a delay offset which may be set to the fixed delay component along the path if the latter is known (otherwise C may also be set to zero as a safe estimate).

[0106] Conversely, the last counter, i.e. the B^{th} counter, may be related to delay range $[B \cdot Q + C, \infty[$.

[0107] In that way, whenever a packet arrives at the optimal playout buffer, the related counter may be increased in a constant time, and the cost is reduced to $O(B)$. In fact, when the delay is divided into Q buckets (e.g. buckets having a delay range from 0 to 10 ms, from 11 to 20 ms, from 21 to 30 ms, etc.), an arrival of a packet with a given delay may immediately be associated to the respective bucket. Accordingly, it is not necessary to scan an array of possible delays, but merely a single operation is required. Accordingly, the time required for executing this operation is constant for all possible delays.

[0108] For example, in case the quantization interval is set to 10ms, and the maximum delay is 500ms, B would be 50, which is rather affordable.

[0109] Generally, the bucket delay ranges need not be of the same size, and/or may also vary over time. In fact, progressive delay ranges may be a viable solution, especially for long-range calls.

[0110] Furthermore, optimal values of Q and C for a flow may be dynamically estimated by examining the delay distribution as the connection progresses. This may further decrease the amount of operations required to compute the optimal playout delay.

[0111] As already stated, the solution described herein may be employed with any scheduler that takes into account deadlines in order to select which PDUs to transmit. For instance, the well-known Earliest Due Date (EDD) algorithm may be used.

[0112] In cellular networks, however, where mobility is a key requisite and channel conditions vary fast, advanced schedulers also take into account other information such as a channel quality feedback from the mobile users. For instance, the Hybrid Channel-Aware and Real-Time (HY-CART) scheduler, devised for UMTS High Speed Packet Access, sorts users according to their priority, which is computed by mixing channel and deadline information. In such cases, the role of the solution described herein may provide meaningful deadlines, which also permit to take better decisions directly in the scheduler.

[0113] Without prejudice to the underlying principles of the invention, the details and the embodiments may vary, even appreciably, with respect to what has been described by way of example only, without departing from the scope of the invention as defined by the annexed claims.

Claims

1. A method of scheduling transmission of data streams from a node (BS) towards a receiver (UE2) in a communication network, wherein said data streams include data packets for playout via a receiver reproduction buffer (106) at the receiver (UE2), the method including:

- arranging said data packets in at least one scheduling queue (SQ) at said node (BS);
- dropping data packets from said scheduling queue (SQ) if their sojourn time in said scheduling queue (SQ) exceeds a given drop deadline, d ;
- determining respective playout values, p_0 , for said data packets indicative of expected playout instants, p , for said data packets by said receiver reproduction buffer (106), and
- assigning (200) said drop deadlines, d , to said data packets as a function of said respective playout values, p_0 ,

characterised in that:

- said determining respective playout values, p_0 , for said data packets indicative of expected playout instants, p , for said data packets by said receiver reproduction buffer (106) comprises emulating, at said node (BS), the receiver reproduction buffer (106) for said data packets with an emulated reproduction buffer (B), wherein said emulating the receiver reproduction buffer (106) comprises emulating an optimal playout buffering algorithm,

wherein said optimal playout buffering algorithm determines said respective playout values, p_0 , as a function of a Mean Opinion Score, MOS, which considers QoS and/or QoE criteria.

2. The method of Claim 1, wherein said data packets include timestamps, g , indicative of the generation time of said data packets.

3. The method of Claim 1 or Claim 2, wherein said determining respective playout values, p_o , for said data packets indicative of expected playout instants, p , for said data packets by said receiver reproduction buffer (106) comprising emulating, at said node (BS), the receiver reproduction buffer (106) for said data with an emulated reproduction buffer (B), wherein said emulating the receiver reproduction buffer (106) comprises emulating an optimal playout buffering algorithm, includes determining said respective playout values, p_o , as a function of an estimated transmission delay, δ , between the generation time, g , of said packets at a sender (UE1) in said communication network and the arrival time, q , of said packets at the receiver (UE2).
- 5
4. The method of Claim 3; wherein said estimated transmission delay, δ , is determined as a function of:
- 10
- a first value indicative of the time, g , said data packets are generated at said sender (UE1),
 - a second value indicative of the time, a , said data packets are received at said node (BS),
 - a third value indicative of the expected time said data packets are transmitted from said node (BS) to said receiver (UE2), and
 - 15 - a fourth value indicative of the estimated delay for transmitting said data packets from said communication node (BS) to said receiver (UE2).
5. The method of any of Claim 3 or Claim 4, including emulating said receiver reproduction buffer (106) at said node (BS) as an adaptive playout buffer.
- 20
6. The method of any one of the preceding claims, wherein said data stream is a Voice over IP data stream, and wherein said Mean Opinion Score is calculated by means of the E-Model.
7. The method of any of any one of the preceding claims, wherein said respective playout values, p_o , are determined a posteriori, by determining an optimal playout delay, p_o , as a function of previous transmission delays, δ .
- 25
8. The method of Claim 7, wherein said data stream is a Voice over IP data stream, and wherein said respective playout values, p_o , are determined as indicative for the expected playout instant, p , of all data packets of a given talkspurt by determining the optimal playout delay, p_o , of the previous talkspurt.
- 30
9. The method of Claim 7 or Claim 8, wherein said respective playout values, p_o , are determined by means of an exponential average taking into account also previous values, p_o , for said respective playout values.
10. A communication node (BS) for the scheduled transmission of data streams towards a receiver (UE2) in a communication network, wherein said data streams include data packets for playout via a receiver reproduction buffer (106) at the receiver (UE2), the communication node (BS) comprising:
- 35
- at least one scheduling queue (SQ) where said data packets of said data streams are stored, and
 - a control module (200) configured to perform all the steps of the method of any of Claims 1 to 9.
- 40
11. The communication node (BS) of Claim 10, wherein said communication node (BS) is a Universal Mobile Telecommunications System Long Term Evolution and/or High-Speed Downlink Packet Access communication node.
12. A computer program product, loadable in the memory of at least one computer and including software code portions which, when executed on a computer, causes the computer to execute all the steps of a method according to any of Claims 1 to 9.
- 45

Patentansprüche

- 50
1. Verfahren zur Ablaufplanung von Datenströmen von einem Knoten (BS) zu einem Empfänger (UE2) in einem Kommunikationsnetz, wobei die Datenströme Datenpakete zum Umsetzen über einen Empfängerwiedergabepuffer (106) am Empfänger (UE2) beinhalten, wobei das Verfahren Folgendes beinhaltet:
- 55
- Anordnen der Datenpakete in mindestens einem Ablaufplanungswartefeld (SQ) am Knoten (BS);
 - Auslassen von Datenpaketen aus dem Ablaufplanungswartefeld (SQ), wenn deren Verweilzeit im Ablaufplanungswartefeld (SQ) eine gegebene Auslassfrist, d , überschreitet;
 - Bestimmen jeweiliger Umsetzwerte, p_o , für die Datenpakete, die für erwartete Umsetzzeitpunkte, p , für die

Datenpaket durch den Empfängerwiedergabepuffer (106) bezeichnend sind, und
 - Zuweisen (200) der Auslassfristen, d , an die Datenpakete als Funktion der jeweiligen Umsetzwerte, po ,

dadurch gekennzeichnet, dass

- das Bestimmen jeweiliger Umsetzwerte, po , für die Datenpakete, die für erwartete Umsetzzeitpunkte, p , für die Datenpakete durch den Empfängerwiedergabepuffer (106) bezeichnend sind, Emulieren des Empfängerwiedergabepuffers (106) für die Datenpakete mit einem emulierten Wiedergabepuffer (B) am Knoten (BS) umfasst, wobei das Emulieren des Empfängerwiedergabepuffers (106) Emulieren eines optimalen Umsetzpufferalgorithmus umfasst,

wobei der optimale Umsetzpufferalgorithmus die jeweiligen Umsetzwerte, po , als Funktion eines Mean Opinion Score, MOS, bestimmt, der Dienstgüte- und/oder Erfahrungsgütekriterien in Betracht zieht.

2. Verfahren nach Anspruch 1, wobei die Datenpakete Zeitmarken, g , beinhalten, die für die Erzeugungszeit der Datenpakete bezeichnend sind.
3. Verfahren nach Anspruch 1 oder Anspruch 2, wobei das Bestimmen jeweiliger Umsetzwerte, po , für die Datenpakete, die für erwartete Umsetzzeitpunkte, p , für die Datenpakete durch den Empfängerwiedergabepuffer (106) bezeichnend sind, Emulieren des Empfängerwiedergabepuffers (106) für die Daten mit einem emulierten Wiedergabepuffer (B) am Knoten (BS) umfasst, wobei das Emulieren des Empfängerwiedergabepuffers (106) Emulieren eines optimalen Umsetzpufferalgorithmus umfasst, Bestimmen der jeweiligen Umsetzwerte, po , als Funktion einer abgeschätzten Übertragungsverzögerung, δ , zwischen der Erzeugungszeit, g , der Pakete an einem Sender (UE1) im Kommunikationsnetz und der Ankunftszeit, q der Pakete am Empfänger (UE2) beinhaltet.
4. Verfahren nach Anspruch 3, wobei die abgeschätzte Übertragungsverzögerung, δ , als Funktion von Folgendem bestimmt wird:
 - einem ersten Wert, der für die Zeit, g , bezeichnend ist, zu der die Datenpakete am Sender (UE1) erzeugt worden sind,
 - einem zweiten Wert, der für die Zeit, a , bezeichnend ist, zu der die Datenpakete am Knoten (BS) empfangen worden sind,
 - einem dritten Wert, der für die erwartete Zeit bezeichnend ist, zu der die Datenpakete vom Knoten (BS) zum Empfänger (UE2) gesendet worden sind, und
 - einem vierten Wert, der für die abgeschätzte Verzögerung zum Senden der Datenpakete vom Kommunikations-Netz-knoten (BS) zum Empfänger (UE2) bezeichnend ist.
5. Verfahren nach irgendeinem von Anspruch 3 oder Anspruch 4, das Emulieren des Empfängerwiedergabepuffers (106) am Knoten (BS) als adaptiver Umsetzpuffer beinhaltet.
6. Verfahren nach irgendeinem der vorhergehenden Ansprüche, wobei der Datenstrom ein Voice-over-IP-Datenstrom ist und wobei der Mean Opinion Score mittels des E-Modelles berechnet wird.
7. Verfahren nach irgendeinem der vorhergehenden Ansprüche, wobei die jeweiligen Umsetzwerte, po , a posteriori durch Bestimmen einer optimalen Umsetzverzögerung, po , als Funktion vorheriger Übertragungsverzögerungen, δ , bestimmt werden.
8. Verfahren nach Anspruch 7, wobei der Datenstrom ein Voiceover-IP-Datenstrom ist und wobei die jeweiligen Umsetzwerte, po , als bezeichnend für den erwarteten Umsetzzeitpunkt, p , aller Datenpakete eines gegebenen Sprachblockes durch Bestimmen der optimalen Umsetzverzögerung, po , des vorherigen Sprachblockes bestimmt werden.
9. Verfahren nach Anspruch 7 oder Anspruch 8, wobei die jeweiligen Umsetzwerte, po , mittels eines exponentiellen Mittelwertes bestimmt werden, der auch vorherige Werte, po , für die jeweiligen Umsetzwerte berücksichtigt.
10. Kommunikations-Netz-knoten (BS) für die geplante Sendung von Datenströmen zu einem Empfänger (UE2) in einem Kommunikationsnetz, wobei die Datenströme Datenpakete zum Umsetzen über einen Empfängerwiedergabepuffer (106) am Empfänger (UE2) beinhalten, wobei der Kommunikations-Netz-knoten (BS) Folgendes umfasst:

mindestens ein Ablaufplanungswartefeld (SQ), wo die Datenpakete der Datenströme gespeichert sind, und ein Steuermodul (200), das konfiguriert ist, alle Schritte des Verfahrens irgendeines der Ansprüche 1 bis 9 durchzuführen.

- 5 11. Kommunikations-Netzknotten (BS) nach Anspruch 10, wobei der Kommunikations-Netzknotten (BS) ein Universal-Mobile-Telecommunications-System-Long-Term-Evolution- und/oder High-Speed-Downlink-Packet-Access-Kommunikations-Netzknotten ist.
- 10 12. Computerprogrammprodukt, das in den Speicher mindestens eines Computers ladbar ist und Softwarecodeabschnitte beinhaltet, die, wenn sie auf einem Computer ausgeführt werden, den Computer veranlassen, alle Schritte eines Verfahrens gemäß irgendeinem der Ansprüche 1 bis 9 auszuführen.

Revendications

- 15 1. Procédé de planification de la transmission de flux de données à partir d'un noeud (BS) vers un récepteur (UE2) dans un réseau de communication, dans lequel lesdits flux de données comprennent des paquets de données pour lecture, par l'intermédiaire d'une mémoire tampon de reproduction (106), au niveau du récepteur (UE2), le procédé comprenant :

- 20 - l'agencement desdits paquets de données dans au moins une file d'attente de planification (SQ) au niveau dudit noeud (BS) ;
- la sortie desdits paquets de données de la file d'attente de planification (SQ) si leur temps de séjour (t) dans ladite file d'attente de planification (SQ) dépasse une date de sortie donnée (d) ;
- 25 - la détermination de valeurs de lecture respectives (po) pour lesdits paquets de données qui sont indicatives des moments de lecture attendus (p) desdits paquets de données par ladite mémoire tampon de reproduction (106) au niveau du récepteur, et
- l'attribution (200) desdites dates de sortie (d) auxdits paquets de données en fonction desdites valeurs de lecture respectives (po),

30 **caractérisé en ce que :**

- ladite détermination des valeurs de lecture respectives (po) pour lesdits paquets de données indicatives des moments de lecture attendus (p), pour lesdits paquets de données par ladite mémoire tampon de reproduction (106) au niveau du récepteur comprend l'émulation, au niveau dudit noeud (BS) de la mémoire tampon de reproduction (106) au niveau du récepteur pour lesdits paquets de données avec une mémoire tampon de reproduction émulée (B), dans lequel ladite émulation de la mémoire tampon de reproduction (106) au niveau du récepteur comprend l'émulation d'un algorithme tampon de lecture optimal,
- 35 - dans lequel ledit algorithme tampon de lecture optimal détermine lesdites valeurs de lecture respectives (po) en fonction d'une Note Moyenne d'Opinion (MOS), qui prend en compte les critères QoS (Qualité de Service) et/ou QoE (Qualité d'Expérience).

- 40 2. Procédé selon la revendication 1, dans lequel lesdits paquets de données comprennent des horodatages (g), indicatifs de l'heure de génération desdits paquets de données.

- 45 3. Procédé selon la revendication 1 ou la revendication 2, dans lequel ladite détermination des valeurs de lecture respectives (po), pour lesdits paquets de données indicatives des moments de lecture attendus (p) pour lesdits paquets de données par ladite mémoire tampon de reproduction (106) au niveau du récepteur, comprend l'émulation, au niveau dudit noeud (BS), de la mémoire tampon de reproduction (106) au niveau du récepteur pour lesdites données avec une mémoire tampon de reproduction (B) émulée, dans lequel ladite émulation de la mémoire tampon de reproduction (106) au niveau du récepteur comprend l'émulation d'un algorithme tampon de lecture optimal, comprend la détermination desdites valeurs de lecture respectives (po) en fonction d'un délai de transmission estimé (δ) entre l'heure de génération (g) desdits paquets au niveau d'un expéditeur (UE1) dans ledit réseau de communication, et l'heure d'arrivée (q) desdits paquets au niveau dudit récepteur (UE2).

- 50 4. Procédé selon la revendication 3, dans lequel ledit délai de transmission estimé (δ) est déterminé en fonction de :
- 55 - une première valeur indicative de l'heure (g) à laquelle lesdits paquets de données sont générés au niveau

EP 2 517 419 B1

dudit expéditeur (UE1),

- une deuxième valeur indicative de l'heure (a) à laquelle lesdits paquets de données sont reçus au niveau dudit noeud (BS),

- une troisième valeur indicative de l'heure prévue à laquelle lesdits paquets de données sont transmis entre ledit noeud (BS) et ledit récepteur (UE2), et

- une quatrième valeur indicative du délai estimé pour la transmission desdits paquets de données entre ledit noeud de communication (BS) et ledit récepteur (UE2).

5

10

5. Procédé selon l'une quelconque des revendications 3 ou 4, comprenant l'émulation de ladite mémoire tampon de reproduction (106) au niveau du récepteur, au niveau dudit noeud (BS) comme tampon de lecture adaptatif.

6. Procédé selon l'une quelconque des revendications précédentes, dans lequel ledit flux de données est un flux de données de Voix sur IP, et dans lequel ladite Note Moyenne d'Opinion est calculée à l'aide du modèle électronique.

15

7. Procédé selon l'une quelconque des revendications précédentes, dans lequel lesdites valeurs de lecture respectives (po) sont déterminées à posteriori, en déterminant un délai de lecture optimal (po), en fonction des délais de transmission (δ) précédents.

20

8. Procédé selon la revendication 7, dans lequel ledit flux de données est un flux de données de Voix sur IP, et dans lequel lesdites valeurs de lecture respectives (po) sont déterminées comme étant indicatives du moment de lecture attendu (p) de tous les paquets de données d'une impulsion vocale donnée, en déterminant le délai de lecture optimal (po) de l'impulsion vocale précédente.

25

9. Procédé selon la revendication 7 ou la revendication 8, dans lequel lesdites valeurs de lecture respectives (po) sont déterminées au moyen d'une moyenne exponentielle prenant en compte également les valeurs (po) précédentes desdites valeurs de lecture respectives.

30

10. Noeud de communication (BS) pour la transmission planifiée de flux de données vers un récepteur (UE2) dans un réseau de communication, dans lequel lesdits flux de données comprennent des paquets de données pour lecture, par l'intermédiaire d'une mémoire tampon de reproduction (106), au niveau du récepteur (UE2), le noeud de communication (BS) comprenant :

- au moins une file d'attente de planification (SQ) dans laquelle lesdits paquets de données desdits flux de données sont stockés, et

- un module de commande (200) configuré pour effectuer toutes les étapes du procédé selon l'une quelconque des revendications 1 à 9.

35

40

11. Noeud de communication (BS) selon la revendication 10, dans lequel ledit noeud de communication (BS) est un noeud de communication d'un Système Universel de Télécommunications Mobiles à Evolution à Long Terme et/ou d'Accès par Paquets en Liaison Descendante Haut Débit.

45

12. Produit-programme informatique, pouvant être chargé dans la mémoire d'au moins un ordinateur et comprenant des parties de code logiciel qui, lorsqu'il est exécuté sur un ordinateur, provoque l'exécution, par l'ordinateur, de toutes les étapes d'un procédé selon l'une quelconque des revendications 1 à 9.

50

55

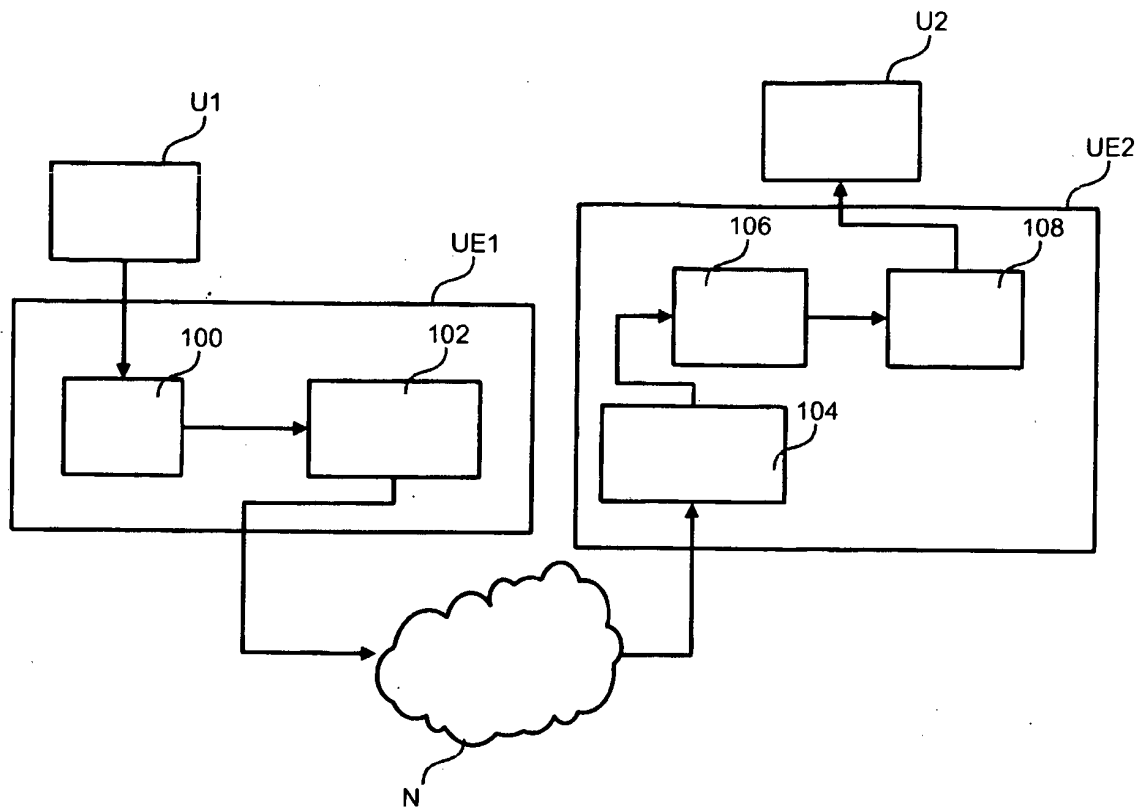


Fig. 1

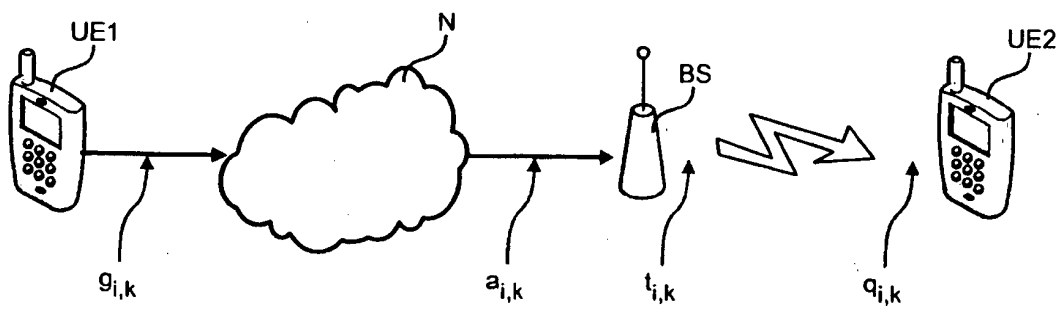


Fig. 2

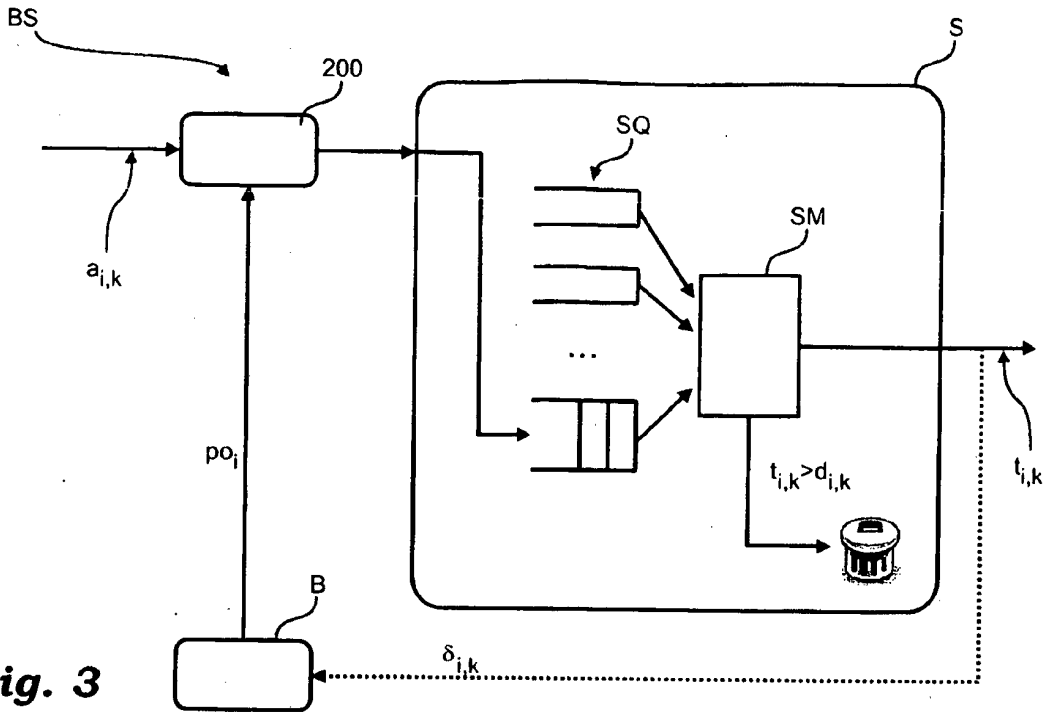


Fig. 3

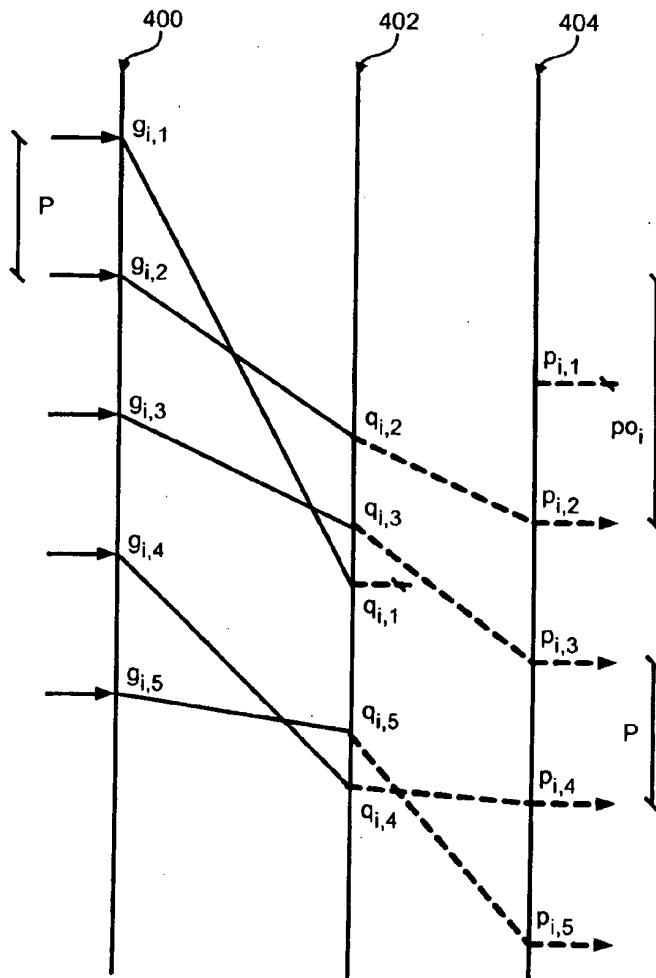


Fig. 4

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- WO 2008059000 A [0025]
- US 20080008096 A [0026]

Non-patent literature cited in the description

- **L. ATZORI ; M.L. LOBINA ; M. CORONA.** Playout buffering of speech packets based on a quality maximization approach. *IEEE Transactions on Multimedia*, 2006, vol. 8 (2), 420-426 [0022]
- **A. BACIOCCOLA ; C. CICCONE ; G. STEA.** User level performance evaluation of VoIP using ns-2. *Proceedings of STOOLS'07, First International Workshop on Network Simulation Tools*, 22 October 2007 [0023]
- **Z. QIAO ; L. SUN ; N. HEILEMANN ; E. IFEACHOR.** A new method for VoIP quality of service control use combined adaptive sender rate and priority marking. *IEEE ICC 2004*, 1473-1477 [0024]
- **M. A. ANDREOZZI et al.** Flexible Scheduling for Real-Time Services in High-Speed Packet Access Cellular Networks. *Wireless Conference 2009*, 17 May 2009, 189-193 [0027]