

METHOD, DEVICE AND SYSTEM FOR TRANSMITTING DATA

This application claims priority to Chinese Patent Application No. 201010253583.0, filed on August 12, 2010 and entitled "METHOD, DEVICE AND SYSTEM FOR TRANSMITTING DATA", which is incorporated herein by reference in its entirety.

FIELD OF THE INVENTION

The present invention relates to the field of communication techniques, and more particular, to a method, device and system for transmitting data.

BACKGROUND OF THE RELATED ART

Machine to Machine (hereinafter, M2M for short) is a type of network, in which all objects are connected to the Internet through information sensing devices to realize intelligent recognition, locating, tracing, monitoring and managing. Currently, information sensing devices applied in M2M comprise RF (Radio frequency) recognition devices, infrared sensors, Global Positioning Systems (GPS), laser scanners, etc.

With the development of M2M, the number of Machine Type Communication (hereinafter, MTC for short) devices has become very huge. In M2M applications, MTC devices are tightly linked to wireless communication techniques: MTC devices receive data from a control center through wireless connections, or report data to the control center through wireless connections. In various particular M2M applications, such as meter reading of wireless electricity meters and condition monitoring of vending machines or POS machines, wireless electricity meters, vending machines or POS machines may have a large number of random or periodic data to report to a control center. In these applications, the number of MTC devices reporting data (such

as, wireless electricity meters, vending machines or POS machines) is huge, however each MTC device only has a relatively smaller amount of data to be reported at a time, which may be tens to hundreds of bytes, even less than the overhead of an IP header in a data packet, leading to low transmission efficiency. For example, in common MTC services, taking wireless meter reading (meter reading of wireless electricity meters) as an example, the amount of data in the application layer to be reported at a time may be 50 bytes, and a TCP/IP header in an current system may be 40 bytes, the byte amount of the TCP/IP header takes up 80 percentage of the application layer data; when there are a large amount of MTC devices reporting data, the transmission efficiency of the wireless communication system may be significantly degraded.

In light of the problem of significantly degraded transmission efficiency, a prior solution is to employ convergers. A converger corresponds to a small sub-network which is constituted by multiple MTC devices through a Zigbee or wired network. The converger converges and re-packs packets reported by the multiple MTC devices, and then adds a header compliant with a wireless air interface network protocol to the packet before forwarding it. Thus, the header overhead can be effectively reduced and system transmission efficiency can be improved.

However, the use of convergers may cause increased initial investment of users. Further, prior small sub-networks corresponding to a converger employ other formats or wired transmission, making its upper network invisible to each MTC device. This configuration affects the control of the upper network on the MTC device, and prevents the one-to-one control of the upper network on the MTC device.

SUMMARY OF THE INVENTION

A method, device and system for transmitting data, capable of improving system transmission efficiency without affecting the control of the network side on each MTC device, are provided according to various aspects of this invention.

According to an aspect of this invention, there is provided a method for transmitting data, comprising: receiving N first data packets which are transmitted from M terminals to a same destination address by an access network entity, wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than 1; combining the first data payloads of the N first data packets to a single second data payload, and adding a second data packet header into a front part of the second data payload to form a second data packet, wherein the second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet are the same as those in at least one of the first data packet headers; and sending the second data packet to a server corresponding to the destination address.

According to another aspect of this invention, there is provided an access network entity, comprising: a receiving module configured to receive N first data packets which are transmitted from M terminals to a same destination address by an access network entity, wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than 1; a creating module configured to combine first data payloads in the N first data packets to a single second data payload, and add a second data packet header into a front part of the second data payload to form a second data packet, wherein the second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet are the same as those in at least one of the first data packet headers; and a sending module configured to send the second data packet to a server corresponding to the destination address.

According to another aspect of this invention, there is provided a system for transmitting data, comprising any access network entity provided in the embodiment of this invention.

According to the method, device and system for transmitting data of embodiments of this invention, the access network entity combines first data payloads in N first data packets transmitted from M terminals, making the N first data payloads share a common data packet header, as such packet

header overheads can be effectively reduced, and system transmission efficiency can be improved. Further, because packets are combined at the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more explicit description of the technical solutions of embodiments of this invention, a brief introduction of accompanying drawings to be used in the description of these embodiments is given below. Obviously, accompanying drawings described below are merely some embodiments of this invention, for those skilled in the art, other accompanying drawings can be derived from these ones without any creative efforts.

Fig.1 is a schematic flowchart of a data transmission process method according to an embodiment of this invention;

Fig.2 is a schematic flowchart of a data reception process method according to another embodiment of this invention;

Fig.3 is a schematic flowchart of a data transmission process method according to another embodiment of this invention;

Fig.4 is a schematic diagram of data convergence in a data transmission process method according to another embodiment of this invention;

Fig.5 is a schematic flowchart of a data transmission process method according to another embodiment of this invention;

Fig.6 is a schematic diagram of data convergence in a data transmission process method according to another embodiment of this invention;

Fig.7 is a schematic flowchart of a data transmission process method according to another embodiment of this invention;

Fig.8 is a schematic diagram of data convergence in a data transmission process method according to another embodiment of this invention;

Fig.9 is a schematic structure diagram of an access network entity according

to another embodiment of this invention;

Fig.10 is another schematic structure diagram of an access network entity according to another embodiment of this invention.

DETAILED DESCRIPTION OF THE EMBODIMENTS

A clear and complete description of technical solutions of embodiments of this invention is given with reference to the accompanying drawings of the embodiments of this invention. Obviously, embodiments described herein are merely some embodiments of this invention, but not all of them. Based on those embodiments of this invention, other embodiments can occur to those skilled in the art without any creative efforts, all of which fall within the scope of this invention.

Fig.1 is a schematic diagram of a method for transmitting data according to an embodiment of this invention. As shown in Fig.1, the method comprises:

101. An access network entity receives N first data packets which are transmitted from M terminals to a same destination address. The first data packet comprises a first data packet header and a first data payload, and N is equal to or greater than 1.

The present embodiment can be applied in a M2M-type communication network, wherein the terminals in this embodiment may be, for example, MTC devices, and the server may be, for example, a control center server, such as a MTC server. Below, a description is given taking M2M as an example. In M2M, there are multiple MTC devices which report data to a MTC server. For instance, during a process of wireless meter reading, multiple electricity meters reports their own data to a MTC server through an access network and a core network.

Each MTC device sends a data packet, wherein M MTC devices may send N data packets, M can be equal or not equal to N . For example, a MTC device may send a single data packet, or several independent data packets. A MTC device sends several independent data packets may be described that, for example, a same electricity meter sends two data packets, wherein one data

packet represents a total amount of electric power consumed and another data packet represents a time period in which most power is consumed. In this embodiment, data packets sent by the MTC devices are referred to as first data packets, and the N first data packets have the same destination address, that is, the N data packets are sent to the same MTC server.

A process of sending the first data packets to the MTC server by the MTC devices may comprise: the MTC device sends the first data packets to an access network entity, and the access network sends these packets to the MTC server through a core network after these packets is processed by the access network entity.

In a Long term Evolution (LTE) system, the access network entity may be for example an e Node B. In a Universal Mobile Telecommunications System (UMTS), the access network entity may be, for example, a NODE B or a Radio Network Controller (RNC).

102. The access network entity combines first data payloads in the N first data packets to a single second data payload, and adds a second data packet header into a front part of the second data payload to form a second data packet. The second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet are the same as those in at least one of the first data packet headers.

After the access network entity receives the N first data packets to be sent to the same destination address, in order to reduce packet header overheads in these data packets, the access network entity converges the N first data packets. The first data packet comprises a first data packet header and a first data payload. The first data packet header may be a TCP/IP header, or an UDP/IP header, comprising a source port number, a destination port number, a source IP address, a destination IP address. The source port number and the source IP address may be called as a source address, and the destination port number and the destination IP address may be called as a destination address. The N first data packets converged by the access network entity have the same destination port number and the same destination IP address.

However, the source port numbers and source IP addresses may or may not be different.

[0001] A particular process of converging the N first data packets by the access network entity may comprise concatenating first data payloads in the N first data packets by the access network entity to form a single data payload, that is, a single second data payload; adding a second data packet header into a front part of the second data payload. Wherein, a structure of the second data packet header is defined according to a pre-negotiation between the access network entity and the MTC server, as well as the type of the N first data packets. Specifically, firstly, when the N first data payloads have the same length and there is a convergence amount pre-negotiated between the access network entity and the MTC server, the second data packet header comprises a source address, a destination address and length information of the data packet; Secondly, when the N first data payloads have the same length and there is not a convergence amount pre-negotiated between the access network entity and the MTC server, the second data packet header further comprises an amount identifier representing the amount of the first data packets being converged, in addition to a source address, a destination address and length information of the data packet; Thirdly, when the N first data payloads have different lengths and there is not a convergence amount pre-negotiated between the access network entity and the MTC server, the second data packet header further comprises an amount identifier representing the amount of the first data packets being converged and length identifiers representing the length of each first data payload, in addition to a source address, a destination address and length information of the data packet.

In the N first data packets, all the first data packet header may or may not have the same source port number and the same source IP address. If all the first data packet headers have the same source port number and the same source IP address, the source address, the destination address in the second data packet header are the same as those in the first data packet headers; if not all the first data packet headers have the same source address, the source address, the destination address in the second data packet header are the

same as those in any one of the N first data packet headers.

103. The access network entity sends the second data packet to a server corresponding to the destination address.

After converging the N first data packets into a second data packet, the access network entity sends the second data packet to the server corresponding to the destination address through a core network, according to the destination address in the first data packet headers. Wherein, the server may be for example a MTC server.

Note that, in this embodiment, the first data packets from the plurality of MTC devices are converged by the access network entity, instead of being converged by convergers and then forwarded to the access network entity, data may be directly sent to the access network entity by each MTC device, so that each MTC device is visible for the access network entity, allowing the access network entity performs one-to-one control on each MTC device.

In this embodiment, first data payloads in N first data packets sent by M terminals that are received by an access network entity are combined to share a common data packet header by the N first data payloads. The packet header overheads can be effectively reduced and transmission efficiency can be improved. In addition, because data packets are combined by the access network entity, the access network entity may perform a single-to-one control on each MTC device.

Fig.2 is a schematic diagram of a flowchart of a method for data packet reception process according to another embodiment of this invention.

201. A server receives a second data packet sent by an access network entity.

Corresponding to the embodiment shown in Fig.1, this embodiment is a process for a server to receive a converged data packet sent from an access network entity. This embodiment can be applied in a M2M type communication network, the server may be for example a MTC server, which can collect and process data reported by MTC devices.

Referring to the embodiment shown in Fig.1, after the access network entity converges the N first data packets into a second data packet, the second data

packet is sent to the server through a core network.

202. The server obtains first data payloads of the N first data packets sent from M terminals from a second data payload in the second data packet; N is larger than 1.

After receiving the second data packet, the server obtains N first data payloads from the second data packet, according to a pre-negotiation with the access network entity and the second data packet header. Each first data payloads is data sent by a MTC device.

[0002] The process of the server obtaining N first data payloads from the second data packet may be as follows: 1. when the N first data payloads have the same length, and there is a convergence amount N pre-negotiated and determined between the server and the access network entity, the server removes the second data packet header after receiving the second data packet, and then evenly divides the second data payload into N segments, each segment being a first data payload, i.e., data sent by a MTC device; 2. when the N first data payloads have the same length and there is not a convergence amount pre-negotiated between the server and the access network entity, after receiving the second data packet, the server removes the second data packet header, obtains an amount identifier N representing the amount of the converged first data packets from the second data packet header, and evenly divides the second data payload into N segments, each segment being a first data payload, i.e., data sent by a MTC device; 3. when the N first data payloads have different lengths, and there is not a convergence amount pre-negotiated between the server and the access network entity, after receiving the second data packet, the server removes the second data packet header, obtains an amount identifier N representing the amount of the converged first data packets and length identifiers representing the length of each first data payload from the second data packet header, and divides the second data payload into N segments according to the length of each first data payloads, each segment being a first data payload, i.e., data sent by a MTC device;

[0003] In this embodiment, the server receives the second data packet sent

and converged by the access network entity from N first data packets that are sent from M terminals, and obtains N first data payloads from the second data payload of the second data packet to obtain data sent from each terminal. Because the N first data payloads are made to share a common data packet header by the access network entity, packet header overheads can be effectively reduced and transmission efficiency can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

Fig.3 is a schematic diagram of a method for data packet transmission process according to another embodiment of this invention. As shown in Fig.3, in connection with the embodiments in Figs.1 and 2, the method comprises the following content.

301. An access network entity negotiates the amount of first data packets to be combined with a server. In this embodiment, N first data payloads in N first data packets sent by M MTC devices have the same length. The first data packets sent by the MTC devices may be application layer packets, a description is given in which the first data packet headers are TCP/IP headers as an example. The server may be a MTC server.

The access network entity negotiates in advance with a MTC server to converge and transmit a constant amount of data sent from MTC devices, that is, negotiates in advance the amount of first data packets to be converged. Wherein, the action of negotiation can be initiated by the access network entity or the MTC server. A particular negotiation process may comprise a and b, or a' and b':

a. the access network entity sends a negotiation request message for negotiating the amount of first data packets to be combined to the MTC server, wherein "to be combined" means "to be converged";

b. the MTC server receives the negotiation request message sent from the access network entity.

a'. the MTC server sends a negotiation request message for negotiating the

amount of first data packets to be combined to the access network entity;

b'. the access network entity receives the negotiation request message sent from the MTC server.

After the negotiation with the MTC server, the access network entity may converge the negotiated amount of first data packets, and the MTC server may decompress the converged data to obtain each of the first data packets.

Note that the access network entity can negotiate with the MTC before each convergence of first data packets, or after a convergence amount has been negotiated between the access network entity and the MTC server, the access network entity may converge a negotiated amount of first data packets each time.

[0004] 302. The access network entity concatenates first data payloads in the N first data packets to form a second data payload.

Upon the convergence amount N is negotiated and determined, the access network entity converges N first data packets sent from multiple MTC devices. Fig.4 is a schematic diagram of data convergence according to the method for data packet transmission process in the embodiment 1 of this invention. As shown in Fig.4, MTC device 1, MTC device 2, ..., MTC device M send N first data packets, each comprising a TCP/IP header and a first data payload.

The access network entity concatenates a first data payload 1, a first data payload 2, ..., a first data payload N contained in the N first data packets sent from MTC device 1, MTC device 2, ..., MTC device M, to form a second data payload.

303. The access network entity adds a second data packet header into a front part of the second data payload. The second data packet header has the same source port number, destination port number, source address and destination address as those in at least one of the first data packet headers.

After the access network entity concatenates the N first data payloads to a single second data payload, a second data packet header is added into a front part of the second data payload (see Fig.4). Wherein, when the N first data packets have the same TCP/IP header, the source address and the destination

address in the TCP/IP header of the second data packet header are the same as those in the TCP/IP headers of the first data packet headers; when not all the N first data packets have the same TCP/IP header, the TCP/IP header of the second data packet header may have the same source address and destination address as those in the TCP/IP headers of any one of the first data packet headers.

Since the second data packet is sent to the MTC server by the access network entity through a core network, the second data packet header may further comprise a Core Network (CN) header.

304. The access network entity sends the second data packet to a server corresponding to the destination address.

According to the destination address in the second data packet header, the access network entity sends the converged second data packet to the MTC server through the core network.

305. The server divides the second data payload into N segments according to the negotiation request message at 301, each segment corresponds to one first data payload.

After receiving the second data packet, the MTC server removes the second data packet header, and evenly divides the second data payload into N segments according to the pre-negotiated convergence amount N. The N segments are the N first data payloads, and each segment is application layer data sent from one MTC device.

Note that when not all the N first data packet have the same TCP/IP header, a TCP/IP header is arbitrarily selected from the N first data packets, and the source address in that TCP/IP header is used as the source address in the second data packet header; in such a case, a re-transmission mechanism should be established on the access network entity for causing the access network entity to retransmit data when there is an error occurred in the data received by the MTC server. Specifically, the retransmission process may be as follows.

When an error in the received second data packet is detected by the MTC

server or a timer is timeout, the MTC server may send a retransmission instruction message to the access network entity.

After receiving the retransmission instruction message, the access network entity retransmits to the MTC server a second data packet corresponding to a second data packet header, according to second data packet header information contained in the retransmission instruction message.

The TCP/IP layers on both the access network side and the MTC server side do not particularly identify which MTC device has error data in the second data packet, instead, the second data packet as a whole is considered as a complete TCP/IP packet as default.

In various embodiments of this invention, the convergence function provided for the access network entity can be arranged in any layer of the wireless access network. For example, it can be realized in the Packet Data Convergence Protocol (PDCP) layer, the Radio Link Control (RLC) layer or the Media Access Control (MAC) layer.

1. A process implemented at the PDCP layer: after the N first data packets are decompressed at PDCP and to be forwarded to an upper layer, the PDCP may converge the N first data packets according to the above method. Wherein each PDCP Service Data Unit (SDU) forwarded is a complete piece of data of a MTC device. The N SDUs are concatenated with IP headers thereof removed, and a common IP header and a CN header are added before be transmitted to a Serving GPRS Support Node (SGSN). This process does not have any effect on the processes at RLC layer and MAC layer.

2. A process implemented at the RLC layer: a reference to the above description can be made for its specific convergence process. Note that when the convergence is performed at the RLC layer, transmissions at the PDCP layers on the MTC device side and the access network side are preferably transparent transmissions, that is, data is directly forwarded at the PDCP layer without any process. Otherwise, a common and compressed IP header must be added into a front part of the second data payload at the RLC layer.

3. A process implemented at the MAC layer: a reference to the above

description can be made for its particular convergence process. Note that when the convergence is performed at the MAC layer, transmissions at the RLC layer and the PDCP layer have to be transparent transmissions.

In this embodiment, first data payloads in N first data packets sent by M terminals that are received by an access network entity are combined, by the access network entity, to share a common data packet header, so that packet header overheads can be effectively reduced and transmission efficiency can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

Fig.5 is a schematic diagram of a method for data packet transmission process according to another embodiment of this invention. As shown in Fig.5, in connection with the embodiments in Figs.1 and 2, the method comprises the following steps.

501. An access network entity concatenates first data payloads in N first data packets to form a single second data payload.

In this embodiment, first data payloads in N first data packets sent by M MTC devices have the same length. The first data packets sent by the MTC devices may be application layer packets, a description is given in which the first data packet headers are TCP/IP headers as an example. The server may be a MTC server.

The access network entity converges data sent from the M MTC devices. Fig.6 is a schematic diagram of data convergence according to the method for data packet transmission process in the embodiment 2 of this invention. As shown in Fig.6, MTC device 1, MTC device 2, ..., MTC device M send N first data packets, each comprising a TCP/IP header and a first data payload.

The access network entity concatenates a first data payload 1, a first data payload 2, ..., a first data payload N contained in the N first data packets sent from MTC device 1, MTC device 2, ..., MTC device M, to form a single second data payload.

502. The access network entity adds a second data packet header into a front

part of the second data payload. The second data packet header further comprises an amount identifier for indicating the amount of the first data packets that are concatenated.

After concatenating the N first data payloads to a single second data payload, the access network entity adds a second data packet header into a front part of the second data payload (see Fig.6). The second data packet header comprises a TCP/IP header and an amount identifier N, wherein a reference can be made to the description in the various embodiments above for the TCP/IP header, which not be described in detail herein; the amount identifier N is used to indicate the amount of the converged first data packets.

Since the second data packet is sent to the MTC server by the access network entity through a core network, the second data packet header may further comprise a CN header.

503. The access network entity sends the second data packet to a server corresponding to the destination address.

According to the destination address in the second data packet header, the access network entity sends the converged second data packet to the MTC server through the core network.

504. The server divides the second data payload into N segments according to the amount identifier in the second data packet header, each segment corresponding to one first data payload.

After receiving the second data packet, the MTC server removes the second data packet header, and evenly divides the second data payload into N segment, according to the amount identifier N in the second data packet header. The acquired the N first data payloads, each segment is a piece of application layer data sent from a MTC device.

Wherein, in various embodiments of this invention, each first data payload may comprise identification information of a MTC device corresponding to this first data payload. The MTC server can be aware of the MTC device corresponding to each data after obtaining this first data payload.

Further, each first data payload may not comprise identification information of

a MTC device corresponding to the first data payload. In such cases, 501 can be replaced with 501': an access network entity concatenates first data payloads in N first data packets according to a sequence and an amount negotiated with a server. There may be several negotiated sequences, for example, the sequence may be based on source IP address or source port numbers in the N first data packet headers of the N first data packets, in which case the source address in the second data packet header is the same as the source address in the first one of the first data packet headers of the N first data packets being concatenated. The sequence may also be based on terminal identification (identification ID) of each MTC devices, wherein the identification ID may be for example the serial number of each MTC device or user information corresponding to each MTC device (for example, house number). The negotiated amount may be the amount of first data packets sent by each of the M MTC devices, that is, the access network entity and the MTC server also pre-negotiates about the amount of the first data packets sent by each MTC device. Specifically, the process of 501' may be as follows.

The access network entity and the MTC server negotiates, in advance, about the sequence of concatenating N first data payloads and the amount of first data payloads to be concatenated that are sent from each MTC device. The sequence may be the ascending/descending order of the source IP address, the source port numbers or MTC device identifications IDs in the N first data packet headers. After the negotiation, the access network entity concatenates the N data payloads according to the negotiated sequence, and setting a source address in the first one of the first data packet headers in the second data packet header.

When replacing 501 with 501', the following step can be further comprised after 504: the MTC server, according to the concatenation sequence and concatenation amount negotiated with the access network entity, adds a corresponding source address preceding each of the N first data payloads. Specifically, the MTC server stores source port numbers and/or source IP addresses of all associated MTC devices, as well as the correspondence between those source port numbers and/or source IP addresses and MTC device identifications IDs. After receiving the second data packet, the MTC

server obtains the source port number or source IP address or identification ID of a MTC device corresponding to the first one of the concatenated first data payloads from the second data packet header. According to the negotiated concatenation sequence and concatenation amount, the MTC server can derive the source port numbers or source IP addresses of MTC devices corresponding to other ones of the first data payloads. Then the MTC server adds a corresponding source port number and/or source IP address preceding each first data payload for normal TCP/UDP/IP layer operations.

In the method employing 501', if one first data payload in the second data packet has an error, the MTC server is enabled to derive which source address the data with an error comes from, and thus to directly request a MTC device at the corresponding source address for retransmission, without the need of providing a retransmission mechanism in the access network entity.

In this embodiment, first data payloads in N first data packets sent by M terminals that are received by an access network entity are combined, by the access network entity, to share a common data packet header. Thus, packet header overheads can be effectively reduced and transmission efficiency can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

Fig.7 is a schematic diagram of a method for data packet transmission process according to another embodiment of this invention. As shown in Fig.7, in connection with the embodiments in Figs.1 and 2, the method comprises the following steps.

701. An access network entity concatenates first data payloads in N first data packets to form a single second data payload.

In this embodiment, not all the N first data payloads in the N first data packets sent by M MTC devices have the same length, and all the N first data payloads may have different lengths, or some may have the same length. The first data packets sent by the MTC devices may be application layer packets, a description is given in which the first data packet headers are TCP/IP headers as an example. The server may be a MTC server.

The access network entity converges data sent from the M MTC devices. Fig.8 is a schematic diagram of data convergence according to the method for data packet transmission process in the embodiment 3 of this invention. As shown in Fig.8, MTC device 1, MTC device 2, ..., MTC device M send N first data packets, each comprising a TCP/IP header and a first data payload.

The access network entity concatenates a first data payload 1, a first data payload 2, ..., a first data payload N contained in the N first data packets sent from MTC device 1, MTC device 2, ..., MTC device M, to form a single second data payload.

702. The access network entity adds a second data packet header into a front part of the second data payload. The second data packet header further comprises an amount identifier for representing the amount of the concatenated first data packets and length identifiers for representing the length of each first data payload.

After concatenating the N first data payloads to a single second data payload, the access network entity adds a second data packet header into a front part of the second data payload (see Fig.8). The second data packet header comprises a TCP/IP header, an amount identifier N, and length identifiers L1-Ln. A reference can be made to the description in the various embodiments above for the TCP/IP header, which not be described in detail herein. The amount identifier N is used to indicate the amount of the converged first data packets, the length identifiers L1-Ln are used to indicate the lengths of the N first data payloads respectively.

Since the second data packet is sent to the MTC server by the access network entity through a core network, the second data packet header may further comprise a CN header.

703. The access network entity sends the second data packet to a server corresponding to the destination address.

According to the destination address in the second data packet header, the access network entity sends the converged second data packet to the MTC server through the core network.

704. The server divides the second data payload into N segments according to the amount identifier and the length identifiers in the second data packet header, with each segment corresponding to one first data payload.

After receiving the second data packet, the MTC server removes the second data packet header, and divides the second data payload into N segments, according to the amount identifier N and the length identifiers in the second data packet header. The N segments are the N first data payloads, with each being a piece of application layer data sent from a MTC device.

In this embodiment, N first data payloads in N first data packets sent by M terminals that are received by an access network entity are combined, by the access network entity, to share a common data packet header, so that packet header overheads can be effectively reduced and transmission efficiency can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

Those skilled in the art may appreciate that some or all steps of the above method embodiments can be realized by hardware related with program instruction. The program aforementioned can be stored in a computer readable storage medium, which when executed causes the hardware or the computer to perform the steps of the above method embodiments, the storage medium aforementioned comprises: ROM, RAM, magnetic disc, optical disc and other mediums capable of storing program code.

Fig.9 is a schematic structure diagram of an access network entity according to an embodiment of this invention. As shown in Fig.9, the access network entity comprises a receiving module 91, a creating module 93 and a sending module 95.

The receiving module 91 is configured to receive N first data packets which are transmitted from M terminals to the same destination address, wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than 1.

The creating module 93 is configured to combine first data payloads of the N

first data packets received by the receiving module 91 to a single second data payload, and add a second data packet header before the second data payload to form a second data packet. The second data packet comprises a second data packet header and the second data payload. The source address and the destination address in the second data packet header are the same as those in at least one of the first data packet headers.

The sending module 95 is configured to send the second data packet that is created by the creating module 93 to a server corresponding to the destination address.

The description of the various method embodiments described above can be referred to for the working flow and principle of each module in this embodiment, which not be described in detail herein.

In this embodiment, first data payloads in N first data packets sent by M terminals that are received by a receiving module are combined to share a common data packet header, so that packet header overheads can be effectively reduced and transmission efficiency of the system can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

Fig.10 is a schematic structure diagram of an access network entity according to another embodiment of this invention. As shown in Fig.10, based on the access network entity of the corresponding embodiment shown in Fig.9, the access network entity further comprises a first negotiation module 97 and/or a second negotiation module 99, and may further comprise a retransmission module 90.

The first negotiation module 97 is configured to send a negotiation request message for negotiating the amount of first data packets to be combined to a server, when each first data payload has the same length. The second negotiation module 99 is configured to receive, from a server, a negotiation request message for negotiating the amount of first data packets to be combined, when each first data payload has the same length,.

The creating module 93 may comprise a first concatenating unit 931 and a first adding unit 933, or the creating module 93 may comprise a first concatenating unit 931 and a second adding unit 935.

The first concatenating unit 931 is configured to concatenate first data payloads in N first data packets to form a single second data payload.

The first adding unit 933 is configured to add a second data packet header into a front part of the second data payload formed by the first concatenating unit, when each first data payloads has the same length. The second data packet header further comprises an amount identifier indicating the amount of the concatenated first data packets.

The second adding unit 935 is configured to add a second data packet header into a front part of the second data payload formed by the first concatenating unit. The second data packet header further comprises an amount identifier for indicating the amount of the concatenated first data packets and length identifiers for indicating the length of each first data payload.

The first concatenating unit 931 is particularly configured to concatenate the first data payloads in the N first data packets according to a sequence and an amount negotiated with the server. The negotiated sequence is based on the source IP addresses or source port numbers or terminal identifications in the N first data packet headers of the N first data packets. The negotiated amount is the amount of first data packets sent from each of the M terminal.

The retransmission module 90 is configured to retransmit a second data packet corresponding to the second data packet header to the server, according to a retransmission instruction message containing the second data packet header sent from the server, when at least two of the first data packet headers have different source addresses.

The description of the various method embodiments described above can be referred to for the working flow and principle of each module and unit in this embodiment, which not be described in detail herein.

In this embodiment, first data payloads in N first data packets sent by M terminals that are received by the receiving module are combined to share a

common data packet header, so that packet header overheads can be effectively reduced and transmission efficiency of the system can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

A system for transmitting data is further provided in another embodiment of this invention, the system may comprise any access network entity provided in the embodiments of this invention.

The system may further comprises a reception process device. The reception process device may correspond to the server in the various embodiments described above.

The data reception process device may comprise a receiving module and an acquisition module.

The receiving module is configured to receive a second data packet sent from an access network entity. The acquisition module is configured to acquire N first data payloads from the second data payload of the second data packet.

The device further comprises a third negotiation module and/or a fourth negotiation module.

The device may further comprise an adding module.

The third negotiation module is configured to receive a negotiation request message for negotiating the amount of first data packets to be combined that is sent from an access network entity, when each first data payload has the same length.

The fourth negotiation module is configured to send a negotiation request message for negotiating the amount of first data packets to be combined to an access network entity, when each first data payload has the same length.

The adding module is configured to add a corresponding source address preceding each first data payload according to a concatenation sequence and a concatenation amount negotiated with the access network entity.

The acquisition module is particularly configured to evenly divide the second

data payload into N segments according to the negotiation request message, when each first data payload has the same length. Each of the N segments corresponds to one first data payload.

Or the acquisition module is particularly configured to evenly divide the second data payload into N segments according to an amount identifier in the second data packet header, when each first data payload has the same length. Each of the N segments corresponds to one first data payload. Each first data payload has the same length, and the second data packet header comprises an amount identifier for indicating the amount of the concatenated first data packets.

Or the acquisition module is particularly configured to, according to an amount identifier and length identifiers in the second data packet header, divide the second data payload into N segments. Each of the N segments corresponds to one first data payload. The second data packet header comprises an amount identifier indicating the amount of the concatenated first data packets and length identifiers indicating the length of each first data payload.

The description of the various method embodiments described above can be referred to for the working flow and principle of each module and unit in this embodiment, which not be described in detail herein.

In this embodiment, first data payloads in N first data packets sent by M terminals that are received by an access network entity are combined to share a common data packet header, so that packet header overheads can be effectively reduced and transmission efficiency of the system can be improved. In addition, because data packets are combined by the access network entity, it is allowed for the access network entity to perform a single-to-one control on each MTC device.

Finally, it should be noted that the above embodiments are merely used to illustrate technical solutions of this invention, but are not limitation thereof. Although this invention has been described in detail with reference to the above embodiments, those skilled in the art may understand that modifications to the technical solutions described in above various embodiments or alternatives to some technical features therein can be made. Such

modifications and alternatives do not make corresponding technical solutions depart from the scope and spirit of the technical solutions of various embodiments of this invention.

Claims

1. A method for transmitting data, characterized in comprising:

receiving N first data packets which are transmitted from M terminals to a same destination address by an access network entity, wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than 1;

combining first data payloads in the N first data packets to a single second data payload, and adding a second data packet header into a front part of the second data payload to form a single second data packet, wherein the second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet header are the same as those in at least one of the first data packet headers;

sending the second data packet to a server corresponding to the destination address.

2. The method for transmitting data according to claim 1, characterized in that each first data payload has a same length, wherein before the combining the first data payloads in the N first data packets to the single second data payload, the method further comprises:

sending, by the access network entity, a negotiation request message for negotiating a amount of first data packets to be combined to the server; or

receiving, by the access network entity, a negotiation request message, which is sent from the server, for negotiating a amount of first data packets to be combined.

3. The method for transmitting data according to claim 1, characterized in that each first data payload has a same length, wherein a procedure of combining the first data payloads in the N first data packets to the single second data payload, and adding the second data packet header into the front part of the second data payload comprises:

concatenating the first data payloads in the N first data packets to form a single second data payload;

adding the second data packet header into the front part of the second data payload, wherein the second data packet header further comprises an amount identifier indicating an amount of the concatenated first data packets.

4. The method for transmitting data according to claim 1, characterized in that a procedure of combining the first data payloads in the N first data packets to the single second data payload, and adding the second data packet header into the front part of the second data payload comprises:

concatenating the first data payloads in the N first data packets to form the second data payload;

adding the second data packet header into the front part of the second data payload, wherein the second data packet header further comprises an amount identifier for indicating an amount of the concatenated first data packets and length identifiers for indicating length of each first data payload.

5. The method for transmitting data according to any of claims 3 to 4, characterized in that a procedure of concatenating the first data payloads in the N first data packets to form the second data payload comprises:

concatenating the first data payloads in the N first data packets according to a sequence and an amount negotiated with the server;

wherein the negotiated sequence is based on a source IP addresses or source port numbers or terminal identifications in the N first data packet headers of the N first data packets; and the negotiated amount is an amount of first data packets sent from each of the M terminals.

6. The method for transmitting data according to any of claims 1 to 4, characterized in that at least two of the first data packet headers have different source addresses, wherein the method further comprises:

retransmitting, by the access network entity, the second data packet corresponding to the second data packet header to the server, according to a retransmission instruction message containing the second data packet header sent from the server.

7. An access network entity, characterized in comprising:

a receiving module configured to receive N first data packets which are transmitted from M terminals to a same destination address, wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than 1;

a creating module configured to combine first data payloads in the N first data packets which are received by the receiving module to a single second data payload, and add a second data packet header into a front part of the second data payload to form a second data packet, wherein the second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet are the same as those in at least one of the first data packet headers;

a sending module configured to send the second data packet which is created by the creating module to a server corresponding to the destination address.

8. The access network entity according to claim 7, characterized in that the access network entity further comprises:

a first negotiation module configured to send a negotiation request message for negotiating an amount of first data packets to be combined to the server, when each first data payload has a same length; and/or

a second negotiation module configured to receive a negotiation request message from the server for negotiating an amount of first data packets to be combined, when each first data payload has a same length.

9. The access network entity according to claim 7, characterized in that the creating module comprises:

a first concatenating unit configured to concatenate the first data payloads in the N first data packets to form the single second data payload;

a first adding unit configured to add the second data packet header into the front part of the second data payload formed by the first concatenating unit, when each first data payload has a same length, wherein the second data packet header further comprises an amount identifier indicating an amount of the concatenated first data packets.

10. The access network entity according to claim 7, characterized in that the creating module comprises:

a first concatenating unit configured to concatenate the first data payloads in the N first data packets to form the second data payload;

a second adding unit configured to add the second data packet header into the front part of the second data payload formed by the first concatenating unit, wherein the second data packet header further comprises an amount identifier for indicating an amount of the concatenated first data packets and length identifiers for indicating length of each first data payload.

11. The access network entity according to any of claims 9-10, characterized in that the first concatenating unit is specially configured to concatenate the first data payloads in the N first data packets according to a sequence and an amount negotiated with the server, wherein the negotiated sequence is based on a source IP addresses or source port numbers or terminal identifications in the N first data packet headers of the N first data packets; the negotiated amount is an amount of first data packets sent from each of the M terminals.

12. The access network entity according to any of claims 7-10, characterized in that the access network entity further comprises:

a retransmission module configured to retransmit the second data packet corresponding to the second data packet header to the server, according to a retransmission instruction message containing the second data packet header sent from the server, when at least two of the first data

packet headers have different source addresses.

13. A data transmission processing system, including an access network entity according to any of claims 7-12.

14. The data transmission process system according to claim 13, characterized in further comprising a data receiving processing device which comprises:

a receiving module configured to receive the second data packet sent from the access network entity;

an acquisition module configured to acquire the N first data payloads from the second data payload of the second data packet.

15. A method for processing data transmission, characterized in comprising:

a terminal and other M-1 terminals sending N first data packets with a same destination address to an access network entity, wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than or equal to 1, so that the access network entity combines first data payloads in the N first data packets to a single second data payload, adds a second data packet header into a front part of the second data payload to form a second data packet, and sends the second data packet to a server corresponding to the destination address, wherein the second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet header are the same as those in at least one of the first data packet headers.

16. The method according to claim 15, characterized in that each first data payload has a same length, wherein before a procedure of combining the first data payloads in the N first data packets to the single second data payload, the method further comprises:

sending, by the access network entity, a negotiation request message for negotiating an amount of first data packets to be combined to the server; or

receiving, by the access network entity, a negotiation request message, which is sent from the server, for negotiating an amount of first data packets to be combined.

17. The method according to claim 15, characterized in that each first data payload has a same length, wherein a procedure of combining the first data payloads in the N first data packets to the single second data payload, and adding the second data packet header into the front part of the second data payload comprises:

concatenating the first data payloads in the N first data packets to form the single second data payload;

adding the second data packet header into the front part of the second data payload, wherein the second data packet header further comprises an amount identifier indicating an amount of the concatenated first data packets.

18. The method according to claim 15, characterized in that a procedure of combining the first data payloads in the N first data packets to the single second data payload, and adding the second data packet header into the front part of the second data payload comprises:

concatenating the first data payloads in the N first data packets to form the second data payload;

adding the second data packet header into the front part of the second data payload, wherein the second data packet header further comprises an amount identifier for indicating an amount of the concatenated first data packets and length identifiers for indicating length of each first data payload.

19. A data transmission process method, characterized in comprising:

receiving, by a server, a second data packet which is sent from an access network entity after the access network entity receives N first data packets transmitted to a same destination address from M terminals, wherein the server corresponds to the destination address;

wherein the first data packet comprises a first data packet header and a first data payload, and N is greater than or equal to 1; the second data packet is formed by the access network entity through combining the first data payloads in the N first data packets to a single second data payload and adding a second data packet header into a front part of the second data payload; the second data packet comprises the second data packet header and the second data payload; a source address and a destination address in the second data packet header are the same as those in at least one of the first data packet headers.

20. The data transmission process method according to claim 19, characterized in that each first data payload has a same length, wherein the method further comprises:

receiving by the server a negotiation request message for negotiating an amount of first data packets to be combined sent from the access network entity; or

sending a negotiation request message for negotiating an amount of first data packets to be combined to the access network entity.

21. The data transmission process method according to claim 19, characterized in that each first data payload has a same length, the first data payloads in the N first data packets are concatenated to form the single second data payload; and

the second data packet header is added into the front part of the second data payload, wherein the second data packet header further comprises an amount identifier indicating an amount of the concatenated first data packets.

22. The data transmission process method according to claim 19, characterized in that:

the first data payloads in the N first data packets are concatenated to form the second data payload; and

the second data packet header is added into the front part of the second data payload, wherein the second data packet header further

comprises an amount identifier for indicating an amount of the concatenated the first data packets and length identifiers for indicating length of each first data payload.

23. The data transmission process method according to claim 21 or 22, characterized in that:

the first data payloads in the N first data packets are concatenated according to a sequence and an amount negotiated with the server;

wherein the negotiated sequence is based on a source IP addresses or source port numbers or terminal identifications in the N first data packet headers of the N first data packets;

the negotiated amount is an amount of first data packets sent from each of the M terminals.

24. The data transmission process method according to any of claims 19-23, characterized in that at least two of the first data packet headers have different source addresses, and the method further comprises:

sending a retransmission instruction message containing the second data packet header from the server to the access network entity; and

receiving by the server the second data packet corresponding to the second data packet header retransmitted from the access network entity.