Latency for Real-Time Machine-to-Machine Communication in LTE-Based System Architecture

Navid Nikaein
Mobile Communication Department, Eurecom
06904, Sophia Antipolis, France
navid.nikaein@eurecom.fr

Srdjan Krco
Ericsson d.o.o.
Belgrade, Serbia
srdjan.krco@ericsson.com

Abstract— **Machine-to-machine communication has attracted a lot of interest in the mobile communication industry and is under standardization process in 3GPP. Of particular interest is LTE-Advanced support for various M2M service requirements and efficient management and handling of a huge number of machines as mobile subscribers. In addition to the higher throughput, one of the main advantages of LTE/LTE-A in comparison with the previous cellular networks is the reduced transmission latency, which makes this type of networks very attractive for real-time mobile M2M communication scenarios. This paper presents a M2M system architecture based on LTE/LTE-A and highlights the delays associated with each part of the system. Three real-time M2M applications are analyzed and the main latency bottlenecks are identified. Proposals on how the latency can be further reduced are described.**

Keywords- Latency, LTE, LTE-A, M2M Communication Scenario, Real-Time Application, System Architecture.

I. INTRODUCTION

As the number of human users of mobile networks is coming to saturation (in many countries penetration is higher than 100%), the M2M domain has become the main focus of many mobile and IT operators and vendors as a new revenue opportunity. In general, the estimates agree that the number of M2M connections and connected devices will steadily grow in the coming years thus increasing the number of mobile network users and creating a new multi-billion market [1]. According to [2], the number of sensors and machines (intelligent, connected devices) being connected to the Internet in 2010 will reach 10% of the volume of IT and telephony devices and will grow at three times the pace of traditional IT and telephony systems over the next several years.

Mobile operators like Telenor, Vodafone and Telefonica have created dedicated units or even companies to focus on M2M business opportunity. Similarly, mobile vendors are creating their own visions, programs and initiatives, such as Ericsson's "50 billion connected devices", to drive development of M2M portfolio. According to this vision, Ericsson expects that in 2020 there will be 50 billion devices connected and available to be used in various existing and new applications. Large IT vendors like IBM and HP also have ambitious plans to connect and exploit information generated by trillions of sensors.

At the present time, the most interesting applications from the commercial point of view are related to smart electricity, automatic water and gas meters reading. However, the M2M application space is vast and includes security, health monitoring, remote management and control, tracking and tracing, intelligent transport systems, distributed/mobile computing and gaming, industrial wireless automation, and ambient assisted living etc.

Machine-to-machine is more than just connected wireless devices sharing data; it's also about collecting and distributing the meaningful data efficiently, often in real-time with a desired latency, managing connected devices, providing back-end connectivity anywhere anytime and enabling third party services to utilize machine generated information when and where applicable and according to the business, security and application rules. To achieve this, a number of components and systems, comprising a M2M ecosystem, have to work in harmony.

Of course, there are a number of problems that have to be solved before such an ecosystem is created: efficient deployment and management of such a huge number of devices, addressing, security, business models as well as providing standardized APIs and functions for interaction with M2M terminals are some of the challenges. The communication network plays an important part of the ecosystem and its ability to support M2M services and traffic requirements, which differ in both the way of working and the requirements from those designed for human-to-human communications [3], will be crucial for the success of such a distributed setup.

LTE is a mobile communication system designed for data services that will eventually replace the existing 3G (WCDMA) systems. It is expected that it will be one of the most important communication technologies for M2M services in the coming years due to its wide adoption as well as its characteristics.

In this paper, we focus on real-time interactive M2M communication scenarios based on LTE/LTE-Advanced with the tight end-to-end delay requirements with the following objectives:

- Draw a M2M system architecture based on LTE/LTE-A;
- Analyze real-time communication scenarios and traffic requirements in which latency is a key issue;
- Calculate latency budgets for each communication scenario using the M2M system architecture;
- Highlight possible improvements to reduce the latency

In the following subsections, first, a LTE/LTE-A based M2M ecosystem and communication scenarios are described and different M2M domains and their associated latencies presented. Then, three real-time M2M application scenarios are analyzed, the main latency bottlenecks identified and finally some possible guidelines to reduce the latency are proposed.

II. M2M System Architecture Based on LTE

Figure 1 gives a high level overview of a M2M ecosystem encompassing the functionality required to support a heterogeneous and distributed system like M2M. This ecosystem includes the following domains:

- M2M capillary networks incorporating smart devices and their gateways using a number of short or wide range communication technologies, reusable across a number of application domains;
- M2M access with adequate support for M2M services;
- M2M core providing interconnectivity and extendable by relevant M2M services (registry, request analyzer, control), 3rd party services (location, charging, processing of data, etc.) and LTE services (AAA, IMS, etc.);
- M2M applications including domain specific processing and visualization of information, and the end user applications interacting with the smart devices through a common platform.

All services and applications are offered and designed to be operated on top of the IP connectivity layer, which expands from the M2M capillary to M2M core.

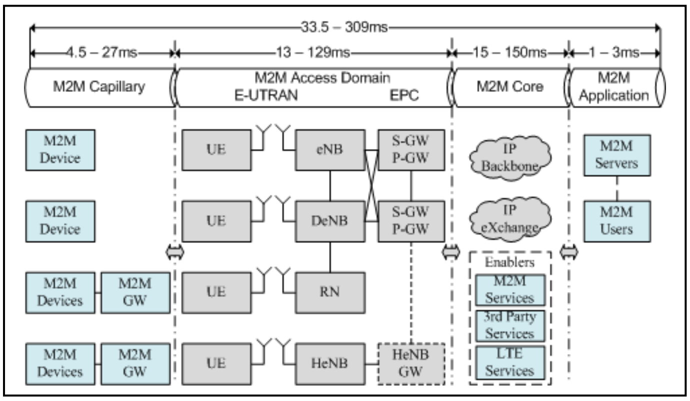

Figure 1 M2M System Architecture

As shown in the figure, a M2M system consists of M2M devices connected to an UE in the evolved UTRAN (E-UTRAN), either directly or via M2M gateways (M2M GW). The M2M GWs act as an access network for the M2M devices. The evolved NodeBs (eNB) in E-UTRAN are connected to the evolved packet core (EPC) via serving gateways (S-GW). The packet data network gateway (P-GW) acts as the gateway to the core network and provides connectivity to the IP eXchange (IPX) network and the IP backbone. The IPX and IP backbone provide connectivity among M2M devices, servers, and users. The evolved packet system (EPS) including E-UTRAN and EPC form the M2M access network, whereas IPX, IP backbone and service enablers form the M2M core network.

In LTE-A, access in E-UTRAN may also be provided through a relay node (RN), which forwards data between UEs and a donor eNB (DeNB). Another possibility is to access the EPC through a home eNB (HeNB). The HeNB may be directly connected to the S-GW or through a HeNB GW.

This figure also shows different M2M domains and their associated latency budgets. The latency induced by service enablers in M2M core and by the handover and UE wakeup delay in discontinuous reception (DRX) mode in M2M access is not shown in the figure as they are application dependent. While the M2M core latency is dominant, the actual latency bottleneck depends on the applications.

The following subsections present the latency concept for both control-plane (c-plane) and user-plane (u-plane), and an in-depth analysis of the latency budget for each domain and component in the M2M ecosystem.

A. Notions of Latency

One of the important design objectives of LTE/LTE-A (and to some extent HSPA) has been to reduce the network latency which consists of both c-plane and u-plane latency.

The c-plane latency can be defined as the time taken by the first packet to successfully reach the receiver reference point. In the LTE/LTE-A, the c-plane latency is defined as a transition time between two states, IDLE or DRX to ACTIVE.[1] Typically, in the LTE/LTE-A the transition time from the IDLE to the ACTIVE state should be less than 100ms, and from the DRX to the ACTIVE state depends on the DRX cycle [3].

The user plane latency, also known as transport delay, is defined as the one-way transit time between a packet being available at the IP layer of the sender and the availability of this packet at the IP layer of the receiver. In the LTE/LTE-A, this latency is defined between the UE and EPC edge nodes. The LTE/LTE-A specifications target the user-plane latency of less than 5ms in unloaded condition (a single user with a single data stream) for a small IP packet with no payload.

B. Latency Budget

1) M2M Capillary Domain

TABLE I shows latency in the M2M capillary domain induced by the processing, transferring and gateway/UE access delays. The application delay accounts for the delay introduced by a client application that resides in a M2M device and communicates with the M2M application server. The gateway formatting and transferring delay increases with the number of devices attached to it as data may have to be aggregated before being forwarded to the M2M application server. The UE and M2M gateway access delays are estimated for an USB and a ZigBee interfaces in this analysis [5], which add additional delays.

[1] ECM (EPS Connection Management) IDLE and CONNECTED states describing connectivity between the UE and the EPC can also be used to represent LTE IDLE and ACTIVE states.

TABLE I LATENCY BUDGET FOR M2M CAPILLARY

Latency Estimates	Description
1-3ms	Application processing and collecting delays - M2M gateway (1-3ms) - M2M device (1ms)
1-3 ms	M2M device/gateway formatting and transferring delays
1.5-20ms	M2M gateway access delay(e.g. Zigbee)
1ms	UE terminal access delay (e.g. USB)

2) M2M Access Domain

TABLE II highlights the components in E-UTRAN and EPC contributing to the total M2M access latency assuming the LTE/LTE-A FDD frame structure [3][10].

The c-plane and u-plane establishment delays depend on the actual state of the UE: LTE IDLE, LTE ACTIVE, RRC (Radio Resource Control) IDLE, and RRC CONNECTED. This delay is the highest during the UE power-up when a UE transits from the LTE IDLE state to the LTE ACTIVE state and is zero in the LTE ACTIVE state. In the RRC IDLE or CONNECTED states, discontinuous reception mode (DRX) can be enabled. This mode is introduced in LTE to improve UE battery life time by reducing the transceiver duty cycle in the active operation. The DRX offers significant improvement with respect to the resource utilization, particularly for applications characterized by the ON-OFF periods or extended OFF periods. However, the cost associated with enabling the DRX mode is that there will be extended latency when an UE needs to wake up and transmits/receives data (see TABLE III). The result in [7] shows that the packet delay increases exponentially with the UE power savings and when the DRX cycle is greater than 80 sub-frames (80ms).

The u-plane latency depends mainly on scheduling policy, buffering and processing, TTI and frame alignment, number of retransmissions, and IP access delays. The delay associated with scheduling can be improved if there are pre-allocated resources (semi-persistent scheduling). The processing delay is assumed to be the same for UE/(H/D)eNB/RN/(S/G)-GW and includes buffering, header compression, ciphering, segmentation, and RLC/MAC processing. The retransmission takes at the best 5ms (HARQ RRT time), and here we assumed the transmission error rate to be varied from 30% to 50% to estimate the retransmission latency for fixed and mobile machines. The latency for the (H/D)eNB and RN increases as the number of UEs increases in the coverage cell.

In the 3GPP LTE-A, relaying at RN can be classified according to which layer is used to forward the data, namely L0, L1, L2, and L3. The L0 relaying is the over-the-air amplify and forward the received signal and does not induce any delay. The L1 relaying is a digital buffer and forward the received signal inducing minimal buffering and processing delay compared to the L0 relay. The L2 relaying requires additional processing to decode and forward the received frame inducing at least one sub-frame delay and possibly scheduling delay. The L3 relaying, also known as self-backhauling, induces an additional delay as the RN terminates the layer 3 (RRC) for the UE interface before forwarding the packet.

The EPC IP access to the M2M core network delay through the S/P-GW is calculated for a propagation speed in copper cables of 200000 km/s, and distance between 2 nodes of 200 – 400 km.

TABLE II LATENCT BUDGET FOR M2M ACCESS DOMAIN

Latency Estimates	Latency Element
0 – 77.5ms	C-plane establishment delay - LTE idle to LTE active (47.5ms +2Ts1c*) - RRC idle to LTE active (37.5ms+2Ts1c*) - RRC connected to LTE active (13.5ms) - LTE active (0ms) *Ts1c (2-15ms) is the delay on S1 c-plane interface
0 – 28.5 ms	U-plane establishment delay - LTE idle to LTE active (13.5ms+Ts1u*) - RRC idle to LTE active (3.5ms+Ts1u*) - RRC connected to LTE active (3.5ms) - LTE active (0ms) *Ts1u (1-15ms) is the delay on S1 u-plane interface
6ms	U-plane Scheduling delay (request and grant)
1-4ms	U-plane UE processing delay
1-4ms	U-plane (H/D)eNB/RN processing delay
1.5ms	U-plane TTI and Frame alignments
1.5-2.5ms	U-plane Retransmission 30%-50% for 5ms HARQ RRT
1-4ms	U-Plane S/P-GW processing delay
1-2ms	U-plane M2M core IP access delay (S/P-GW)

As TABLE II indicates, the c-plane establishment delay is dominant when comparing to the u-plane latency as the 1ms frame size in the LTE reduces significantly the transmission latency.

TABLE III shows the handover related latency for both data forwarding and radio processing for contention-free access [3]. The estimate may vary depending on the procedures. Six handover scenarios are possible among HeNBs, (D)eNBs, and RNs. Data forwarding for the (D)eNB/RN handover scenarios is done through the X2 interface inducing very low latency, while for the HeNB/eNB handover scenarios is done through the Internet, thus adding variable high latency. Please note that in a typical HeNB deployment, the HeNB is connected with the EPC via HeNB GW or directly over a fixed-line broadband access and the Internet.

TABLE III LATENCY BUDGET FOR DRX AND HANDOVER

Latency Estimates	Description
10-512ms	Length of DRX cycle - Short DRX cycle (10-320ms) - Long DRX cycle (10-512ms)

5-150ms	U-plane data forwarding latency for handover from source to target eNB: - (D)eNB/RN over the X2 interface (5ms) - HeNB/eNB over the Internet (15-150ms)
12ms	C-plane radio layer processing (DL sync., UL resource request/grant, timing advance)

3) M2M Core and Application Domains

Interconnection between the M2M devices/gateways and servers/users may be done either through the IP backbone or the IP Packet eXchange (IPX). The delay for the IP backbone (Internet) depends on the region as well as the number of nodes in the network, and processing delays in the nodes. For example, in Europe it could vary from 15ms up to 150ms [5]. This is also true for the IPX; in Europe the delay for interactive traffic class is in the range of 42 – 122ms [8]. The delay for a M2M service enabler represents only one transaction between a M2M device and the service enabler. It depends on the service publishing and lookup, and increases with the number of devices registered in the system. The application server processing delays are in the order of a few milliseconds and may increase with the number of M2M devices and users connected to the application server.

However, this delay should be upper-bounded for the real-time applications. TABLE IV shows the latency estimates for the M2M core and application domains.

TABLE IV LATENCY BUDGET FOR M2M CORE AND APPLICATION DOMAIN

Latency Estimates	**Description**
15-150ms	Network access delay through* - Internet (15-150ms) - IP eXchange (42-122ms) *Also applicable to M2M users accessing M2M application servers
300-500ms	Service enablers delay - Service publishing (300ms) - Service lookup (300-500ms)
1-3ms	Application access/processing delay

III. Application Scenarios

Although a large variety of M2M application scenarios with heterogeneous requirements and features exists, they can be classified into two main M2M communication scenarios as defined in [3], communication of M2M devices with M2M servers/users and communication between the M2M devices. In the following subsections, three real-time M2M application scenarios with very low latency requirements are described and evaluated.

A. Autopilot (Intelligent Transport System)

This scenario includes both vehicle collision detection and avoidance (especially on highways) and how the urgency actions are taken in case of an accident. It is based on a M2M device equipped with sensors embedded in the cars and surrounding environment and used in automatic driving systems. These M2M devices (cars, road sign units, highway cameras) send information to a backend collision avoidance system. The backend system distributes notifications to all vehicles in the vicinity of the location of the collision, together with information required for potential actuation of relevant controls in the affected cars. In all receiving cars, the automatic driving systems based on the received information take over the control fully or partially (brakes activated, driving direction changed, seating belts tightened, passengers alerted etc). If there is no such system in a car, the driver is notified and instructed. Also, depending on the proximity of the accident, different commands are sent to the cars, i.e. the cars which are closer to the place of the possible collision are getting immediate commands for the actuators, while the cars which are further away from this place get driver notifications only.

Two main traffic patterns can be identified in this scenario:

- Periodic, low data rate keep-alive messages (GPS, speed, time) from the M2M devices to the backend system (once per minute, in the order of 500B per message);
- Event-driven, short bursts emergency signals from the M2M backend to the M2M devices including warning and actuation commands (each burst in the order of 1-2kB).

The first traffic pattern can be modeled as short sleep periods with (very) short bursts as in a classical ON-OFF traffic model [9], while the second could be modeled with very long OFF periods. For the first case, a short DRX cycle may be used at the UE side possibly with pre-allocated resources.

In this scenario, it is assumed that before a user initiates a registration process with the autopilot application server, the terminal is in the LTE ACTIVE state and is synchronized to the network. Access to the autopilot application server is done through the IP backbone. It is also assumed that the M2M devices containing the sensors and applications in the cars are connected and synchronized with the UE. The transition from the RRC CONNECTED to the LTE ACTIVE is relevant only, as the UE can only enter a short DRX mode.

As potentially there will be a high number of autopilot users, the serving cell capacity could be an issue (the number of simultaneous users that can be served by a cell). The actual total throughput is not critical as the amount of traffic generated by each user will not be too high. The influence of handovers must also be taken into account as the car will move through several cells with different speeds. Network density is also an issue in this scenario, but assuming the road is very well covered with LTE eNodeB cells there is no need for inter-LTE handovers (handovers between the LTE and other networks, e.g. HSPA), i.e. only intra-LTE handovers (handover inside the LTE network and between eNB/RN) exist. It is also assumed that M2M devices are connected to an UE via M2M gateways and have the same performance.

The total one-way latency estimate for each component in the M2M ecosystem listed in the tables above includes the following[2]:

- M2M capillary: 4.5-25ms (1,1-3,1.5-20,1);
- M2M access: 57-378ms (13.5,3.5,6,1-4,1-4,1.5,1.5-2.5,1-4,1-2) + (10-320,5,12);
- M2M core: 15-150ms;
- M2M application: 1-3ms.

Taking into account all delay components listed above, the total one–way end-to-end delay can vary from 77.5 – 556ms.

B. *Virtual Race (Gaming Machine)*

One example of the many possible M2M games is the virtual race (e.g. virtual bicycle race using real bicycles). The opponents are on different locations, possibly many kilometers away. At the beginning, the corresponding length of a race is agreed (i.e. 10 km or 20 min) between the peers. The measurements are taken by sensors (GPS, temperature, humidity, speed, terrain configuration etc.) and are exchanged between the opponents. They are used by the application to calculate the equivalent positions of the participants and to show them the corresponding state of the race (e.g. "you are leading by 10 m"). The number of competitors may be more than two, and all competitors must mutually exchange information, and the applications must present all participants the state of other competitors. For a large number of competitors (hundreds or more), a corresponding application server must be used. During the race they are informed about the place and the distances from each other (e.g. "you are the 3rd behind the 2nd by 10 m and leading before the 4th by 15 m").

The packets containing GPS and sensors data are on the order of 1 kb. Taking into account the typical speeds (of bicycles) in this scenario (rarely higher than 50 km/h = 13.9 m/s), the packets should be exchanged approximately every 100 ms, which corresponds to a resolution of 1.4 m. The application should be aware of the positions of all competitors with respect to the end of the race, and, when the competitors are close to the finish, packets should be sent every 70 ms which corresponds to a resolution of 1 m (GPS accuracy). Data rates are normally not higher than 10 kb/s (about 15 kb/s at the final stage of the competition).

The traffic pattern in this scenario is comparable to a periodic CBR data transmission between M2M devices with increasingly shorter periods as the end of the race is getting closer. This traffic pattern can also be modeled as a classical ON-OFF traffic model, where the OFF period is decreasing. With a small and medium number of competitors, the actual throughput is not critical as the amount of traffic generated by a user will be small. With a large number of competitors, e.g. 100, the cell capacity limitations have to be considered.

The assumptions for the UE state, the DRX mode, and handover taken in the autopilot scenario are valid in this scenario as well. Furthermore, in this scenario users can belong to different operators' network, thus further adding to the latency budget. In this analysis, it is assumed that all used mobile networks have the same performance and are interconnected with each other via IPX. In this scenario, M2M devices are directly connected to a UE.

Assuming the two LTE networks have the same performance, all delays except the M2M core delay will be doubled. It also means that any latency reduction will result in a "double" improvement in the total latency. In this scenario the end-to-end delay is actually the delay between the M2M devices of two competitors.

The total one-way latency estimate for each domain is:

- M2M capillary: 6-10ms 2x(1,1-3,0,1);
- M2M access: 112-748ms 2x(13.5,3.5,6,1-4,1-4,1.5,1.5-2.5,1-4,1-2) + (10-320,5,12);
- M2M core: 42-122ms;
- M2M application: not applied in this scenario.

Taking into account all delay components listed above, the total one–way end-to-end delay is between 160ms and 880ms

C. *Smart Environment*

This scenario includes environments in which intelligence is embedded and distributed into a large number of M2M devices to allow monitoring and detecting abnormal situations that could result in damage to the immediate environment. A subset of M2M devices may perform a common real-time task, which requires a tight cooperation and interaction among them to coordinate each subtask. They may send a control command (e.g. actuation) to each other to optimize the overall task objectives (e.g. arrange the movement of a subset of M2M devices to improve connectivity). The information (e.g. environment measurements, commands, data) may be pre-processed and/or aggregated before being exchanged among the M2M devices and/or forwarded to the M2M servers and users in a local or remote (manned) control center. The M2M users may also control the M2M devices. Latency is an important factor since delay may destabilize the distributed control operation.

This scenario includes the following traffic patterns:

- Event driven, low data rate burst of control messages amongst M2M devices and/or from M2M users;
- Event-driven, high data rate burst of data amongst M2M devices/gateways and/or to M2M servers/users.

The resulting traffic patterns can be modeled as no traffic when the system is in sleep mode, with bursts of communication when an event occurs or is triggered. For both cases and depending on the frequency of the events, short and/or long DRX cycle may be used at the UE side.

In this scenario, there is a large number of M2M devices that may be connected to the M2M access via M2M gateways. M2M devices and users may belong to different networks. The influence of handover from a HeNB must be included in the handover scenarios as some M2M devices may monitor home/office areas. Interconnectivity may be provided through both IP backbone and IPX.

[2] For each scenario, the values for each item of M2M domain are shown in the parenthesis.

As in the previous scenario, assuming all the LTE networks have the same performance, the delay between M2M devices will be doubled. In this scenario the end-to-end delay is actually the maximum delay among M2M devices and that of the M2M devices and users.

The total one-way latency estimate for each domain is as follows:

- M2M capillary: 9-54ms 2x(1-3,1-3,1.5-20,1);
- M2M access: 180-1290ms 2x(37.5ms+2Ts1c, 13.5+Ts1u,6,1-4,1-4,1.5,1.5-2.5,1-4,1-2) + (10-512,5,12);
- M2M core: 15-150ms;
- M2M application: 1-3ms.

Taking into account all delay components listed above, the total one–way latency is between 205ms and 1497ms.

D. Latency Bottlenecks and Possible Improvements

In the real-time M2M communication scenarios described in the previous section, surprisingly the main latency bottleneck is identified at the access layer procedures although the latency of the core network is non-negligible. The bottlenecks also vary depending on the applications, in particular DRX delay for the ON-OFF traffic model, handover and HARQ delays for home/mobility based scenarios, and access and processing delays for high density scenarios. The overall delay also depends on the actual system/traffic load, outage probability, and radio propagation conditions [6].

In LTE, based on the knowledge of the uplink and downlink activity requirements for a certain UE, the DRX delay can be significantly reduced through prudent selection of various DRX parameters [7][10]. Depending on the traffic pattern, a certain on-time can be set to keep the UE awake by scheduling it within a certain time window.

Depending on the deployment, the handover delay increases noticeably in scenarios involving HeNB. The worst case is when handover between HeNBs take place. The packet forwarding delay in a handover can be optimized through a tight cooperation of the source and the destination (H)eNBs with the S/P-GW for both uplink and downlink traffic.

The HARQ retransmissions are planned outside of the predefined DRX cycle to allow for a tighter DRX optimization without having to plan for the worst-case retransmissions. There is a DRX retransmission timer defined so that a UE does not have to wait for a full DRX cycle for an expected retransmission that has been lost. The UE wake up time can be considerably improved by sending multiple copies of the paging message to the UE.

Access delay depends on the proximity of the server/service with respect to the M2M capillary domains. Depending on the communication scenarios, it can be improved either through multiple application servers located closer to the client application for instance in the operators' domain or through appropriate and possibly localized/distributed service enablers interconnecting M2M devices. When the M2M access domain, service enablers and application servers belong to the same operator, the access and processing delay can be jointly minimized.

The traffic load and outage probability may drop when sufficiently dense RNs are used to extend the radio coverage and to increase the capacity at the expense of an extra delay for the handover between eNBs and RNs.

IV. CONCLUSION

Latency is becoming a key issue for network operators seeking solutions to support new real-time machine-to-machine applications. A significant latency improvement is possible by careful selection of various parameters and technologies and by providing appropriate services in a M2M ecosystem according to the specific application requirements. Currently, LTE can provide on the order of 10ms latency for the E-UTRAN in the ACTIVE state. The core network adds a significant amount of delay depending on the region and the proximity of the server with respect to the access network serving the device. It should be possible to attain 50ms end-to-end delay in many situations, which would make access network comparable to DSL in terms of latency. The core network should also make some headway in latency reduction.

ACKNOWLEDGMENT

This paper describes work undertaken in the context of the LOLA project - Achieving LOw-LAtency in Wireless Communications (www.ict-lola.eu). The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement n° 248993.

REFERENCES

[1] D. Dicks, J. Blau, and T. Kridle, "M2M on the rise: the technology perspective", Heavy Reading Mobile Networks Insider, July 2010.

[2] Harbor Research Inc., "The Internet of thins meets the Internet of people", White paper, 2010.

[3] 3GPP TS 22.368 V10.0.0, "Service requirements for machine-type-communication (MTC)", March 2010.

[4] 3GPP TR 25.912 V.9.0.0, "Feasibility study for evolved universal terrestrial radio access (UTRA) and universal terrestrial radio access network (UTRAN)", Rel. 9, Dec 2009.

[5] G. Thonet, P. Allard-Jacquin, P. Colle, "ZigBee – WiFi Coexistance", Schneider Electric, White paper and test report, 2008.

[6] LOLA consortium, "D2.1 Target Application Scenarios", FP7 EU LOLA Project, March 2010. Available: http://www.ict-lola.eu

[7] Chandra S. Bontu, Ed Illidge, "DRX mechanism for power saving in LTE", IEEE Communication Magazine, June 2009.

[8] GSMA, "Inter-service provider IP backbone guidelines" PRD IR. 34, version 4.9, March 2010.

[9] A. Adas, "Traffic models in broadband networks", IEEE Communication Magazine, 1997

[10] H. Holma, A. Toskala, "LTE for UMTS – OFDMA and SC-FDMA based radio access", John Willey 2009.

[11] T. Blajic, D. Nogulic, M. Druzijanic, "Latency improvement in 3G LTE", Ericsson Report, 2007.