OPEN ACCESS

University of BRISTOL

Song, T. M. K., Kaleshi, D., Zhou, R., Boudeville, O., Jing-Xuan, M., Pelletier, A., & Haddadi, I. (2011). Performance evaluation of integrated smart energy solutions through large-scale simulations. In IEEE International Conference on Smart Grid Communications (SmartGridComm), 2011. (pp. 37 - 42). Institute of Electrical and Electronics Engineers (IEEE). 10.1109/SmartGridComm.2011.6102351

Link to published version (if available):
10.1109/SmartGridComm.2011.6102351

Link to publication record in Explore Bristol Research
PDF-document

Performance Evaluation of Integrated Smart Energy Solutions Through Large-Scale Simulations

Terence Song[†], Dritan Kaleshi[†], Ran Zhou[†], Olivier Boudeville[‡], Jing-Xuan Ma[‡], Aude Pelletier[§] and Idir Haddadi[§]

[†]Centre for Communications Research, University of Bristol, Bristol, United Kingdom
[‡]Département SINETICS, EDF R&D, Clamart, France
[§]Smart Metering Programme, EDF Energy / EDF R&D UK Centre, London, United Kingdom

Abstract—**The information and communication technology (ICT) infrastructure that will empower the Smart Grid with real-time management of power flows, bidirectional metering, and more effective integration of renewable energy sources is both large and complex, involving many different sensing, measurement, and control devices. The analysis of such large-scale distributed systems together with the possible communications network technologies can be extremely difficult, not least because many modeling techniques tend to analyze individual components, rather than the relationships and interactions between components, and their impact on the system. In this paper, we describe the CLEVER simulator—a discrete-event simulator developed specifically for evaluating the performance of the ICT infrastructure of smart energy solutions at very large scales. This paper also presents results from simulation experiments comparing the impact of various access technologies on the performance of a smart metering infrastructure. Finally, an outline of ongoing work is provided, and directions for future work are identified.**

Keywords-large-scale simulation; performance evaluation; smart metering

I. INTRODUCTION

Information and communication technologies, which are themselves major consumers of electricity [1], will play a central role in the creation of a smarter power grid to help realize a more efficient and sustainable energy future [2]. A "Smart Grid" is created by overlaying an electrical grid with an information and communication system made up of sensing, measurement, and control devices, which include smart meters. This will allow the electricity production, transmission, distribution and consumption parts of the grid to communicate with each other, thus making it possible for the grid to dynamically respond to changes (e.g., faults, generation fluctuations, demand-response, etc.). The information and communication technology (ICT) infrastructure that will empower the Smart Grid with real-time management of power flows, bidirectional metering data, and more effective integration of renewable energy sources is both large and complex. There are many choices in terms of technology solutions for providing a well-performing end-to-end system, as well as different strategies for deployment and service delivery. Furthermore, the data communication requirements for smart energy systems are appreciably different from more general communication networks: characterized by typically low data rate and small payload transactions, from many millions of devices, in a potentially coordinated manner (e.g. when executing demand response applications). A smart metering infrastructure (SMI) is an example of such a system. Due to the significant costs associated with deploying and operating an SMI, one of the most pressing needs for energy suppliers and communications providers is to reduce the significant risks inherent in commissioning large-scale ICT infrastructures, especially one where there is little prior experience. To mitigate these risks, it is necessary to investigate the impact of these choices on the non-functional requirements of the ICT infrastructure such as performance, reliability, and availability, under both normal and exceptional conditions. As such, an evaluation of architectural and technological alternatives for integrated energy management solutions based on both cost and performance is critical.

In this paper, we describe the CLEVER simulator, a scalable discrete-event simulator developed specifically for evaluating the performance of the ICT infrastructure of smart energy solutions, allowing for different architectures, communications options, and service strategies to be compared on the same platform and under a common set of assumptions. The simulator integrates models of communications link technologies with processing models of communications nodes running distributed concurrent applications, and provides an extremely versatile platform for "what-if" analysis of smart metering and smart grid systems in real scale (i.e., representing hundreds of thousands of homes, using millions of simulation actors). In the following sections, we will set out the modeling requirements, and then describe how the main system elements of an SMI are represented in the simulator. Results from simulation experiments, comparing the impact of various access technologies on the performance of the SMI, are presented. Finally, an outline of ongoing work is provided, and directions for future work are identified.

II. MODELLING REQUIREMENTS

The basic elements[1] of a typical SMI are shown in Fig. 1. The SMI allows, on one side, for enhanced energy usage monitoring through the remote and ongoing collection, processing, and management of consumption data from very

---

[1] To keep this figure uncluttered, only a minimum number of the main elements and their relationships are shown. In reality, there may be hundreds or thousands of data concentrators, with potentially hundreds or thousands of homes connected to each data concentrator. Other system elements have been abstracted and/or combined.
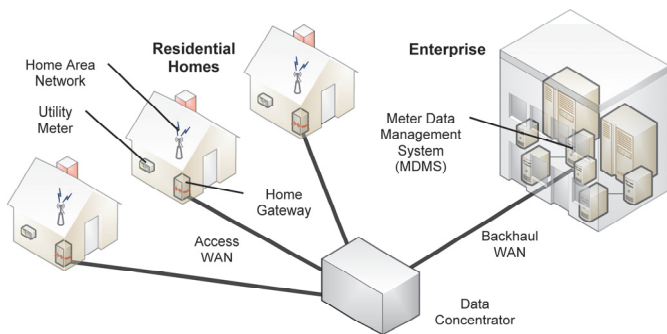
Figure 1.   The basic elements in a typical smart metering infrastructure.

large numbers (millions) of smart meters, and on the other hand for the provisioning of energy management services, such as demand-side management and load control, using a communications network.

The functional requirements of the ICT infrastructure for smart metering, which describe the intended behavior in terms of functions and services the system is required to perform, have already been outlined by various organizations (e.g., the Energy Retail Association's Supplier Requirements for Smart Metering Project [3]) and international standardization efforts (e.g., the EU CENELEC Smart Metering Coordination Group, IEC SG3, IEEE P2030, EU Joint Working Group on Smart Grids, and ITU-T TSAG Focus Group on Smart Grid), which are actively working to ensure implementations are consistent and interoperable.

However, the non-functional requirements of ICT infrastructures for smart energy, which describe how well some behavioral or structural aspect of the system should be accomplished, and the impact of technological and architectural choices on performance, reliability, availability, and security are significantly less well understood. For instance, the SMI shown in Fig. 1 has a logical tree topology, where the Meter Data Management System (MDMS) is the root and home devices are the leaves. Considering the pattern of application data-flow between the enterprise and house-holds, it can be easily surmised that data aggregation points, such as the access WAN, the data concentrator, backhaul WAN, and the MDMS, can all be potential performance bottlenecks if they are not properly dimensioned. Their normal mode of operation, under certain load conditions, can also cause performance bottlenecks further along in the system.

The SMI's ICT infrastructure can only be properly dimensioned if its performance, under both normal as well as exceptional conditions, can be studied end-to-end. Assessing the performance of the system end-to-end, and the performance of its constituent elements, provides essential information that can be used not only to identify resource bottlenecks but also for algorithm tuning, capacity planning, deployment, administration, cost estimation, and long-term planning of the evolution of the system (i.e., to predict performance in the presence of traffic growth, technological advances, as well as introduction of new services onto the system together with possible market and regulatory changes). However, the analysis of large-scale distributed systems and networks can be extremely difficult, not least because many modeling techniques tend to analyze individual components (local performance), rather than the relationships and interactions between components, and their impact on the system as a whole.

Discrete-event simulation (DES) [4] is a well-established technique for performance analysis of communications systems and networks that have many different time-varying parameters, and has become a valuable tool in developing, testing, and evaluating network protocols and architectures. It provides a practical methodology, for understanding network system behavior that is either too complex for mathematical analysis (using analytical models), or too expensive to be investigated through measurements or prototyping on real equipment. Simulation provides a means to study, simultaneously, the interactions of many system variables (i.e., their relationships and how their state or behavior affect others) to provide valuable insight into the consequences of multiple design options, thus allowing alternative designs to be compared relatively quickly and cheaply under like-for-like conditions. It also allows for the potential to capture emergent or unexpected behavior, provided the simulation system is driven by the "right" conditions.

A rigorous study of issues such as system scalability, survivability, rare event failures, and emergent behavior due to the interaction of large numbers of traffic sources with different traffic generation characteristics, would require realistic models to be simulated for long time scales. The capabilities of conventional simulation techniques (executed on a single computing host) are not sufficient to address such simulation scenarios. As the size of the simulation or the level-of-detail in its models increase, so does the amount of computing resources required to execute the models. Large-scale simulations of communications networks and systems at this complexity require large amounts of memory and computing power that is usually available on supercomputers and high-performance computing clusters. Consequently, concurrent, parallel, and distributed simulation techniques must be employed [5].

III.   SIMULATION PLATFORM

The CLEVER simulator is a scalable discrete-event simulator developed specifically for evaluating the impact of different design choices on the end-to-end system performance of SMIs at very large scales. It is implemented entirely in Erlang [6]—a functional programming language with built-in support for concurrency, distribution, and fault tolerance, commonly used for building massively scalable systems. The simulator consists of two distinct parts:

- The simulation platform, which consists of the Erlang runtime; a distributed trace system; an object-oriented framework for mapping Erlang functional programming constructs to object-oriented programming (OOP) concepts (WOOPER) [7][8]; and the Sim-Diasca simulation engine [9], which provides the basic abstractions for defining distributed actor-based simulation models, as well as the instrumentation necessary for deploying, initializing, launching, and executing actor models on concurrent, distributed, and parallel computing platforms. Sim-

Diasca also provides the necessary facilities for measuring simulation data, collecting results, and generating reports.

- The CLEVER simulation models, which represent the system under study as a collection of dynamic stochastic processes that act and interact with each other in simulated time. The modeling approach is described in more detail in the following section.

IV. MODELLING APPROACH

A. Modelling the SMI

The CLEVER simulator models the SMI depicted in Fig. 1 as a collection of simulation actors that act and interact with each other through the exchange of logical (application) messages (illustrated in Fig. 2). The concept of "actors" is used here to mean autonomous decision-making entities that execute, under the control of the simulation engine, behavior appropriate for the system or entity they represent. More specifically, each actor individually assesses its situation and makes decisions on the basis of a set of rules

The simulator defines two basic types of actors for representing the SMI, or indeed any communications network or distributed system, as a collection of communications node actors connected by communications link actors. Communications node actors model application behavior,
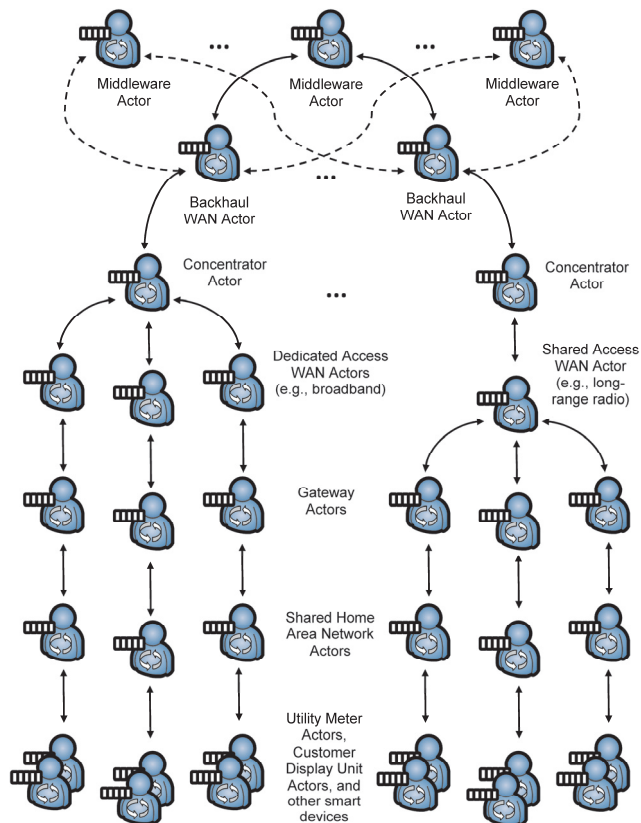

Figure 2. The main elements of the SMI represented as simulation actors, and the relationships between them.

which produce and consume logical messages, and communications link actors model communications behavior, which deliver logical messages from one communications node to another. In the following sections, we describe how the communications node and communications link models represent the following three basic elements:

- The demands on network services and resources, which are generated by communication node actors and form a stream of requests and resource holding times.

- The mechanisms the system uses in processing those demands or requests for service. These are different for communication node actors and communications link actors.

- The instrumentation for collecting statistics to form performance predictions, and analysis of the output data.

B. Discrete-Event Queue Model

Fundamental to the evaluation of system performance, and the resources needed to service requests, is the study and analysis of waiting lines or queues [10], which form when there is insufficient capacity to service requests. As communications node and communications link actors interact with each other, through the sending and receiving of logical messages, they generate requests for service on each other. In order to allow communications node and communications link actors to manage requests for service, the simulator defines a generic discrete-event service queue model with built-in support for measuring various queue performance indicators, such as queue event rates, waiting and service times, queue utilization, etc. The discrete-event service queue allows for any queuing situation to be represented. The implementation supports various queuing disciplines (e.g., FIFO, priority), queues with a single server or multiple servers, as well as queues with shared server capacity (i.e., multiple-queue, single-server).

From a queuing point-of-view, the simulation actors that represent the SMI form a collection of interactive queuing systems (i.e., a queuing network), where the departure of some queues feed into other queues. Queuing networks are a more realistic model for a system with many resources interacting with each other. However, the interaction between queues makes the analysis of a queuing network much more complicated. As such, queuing networks must be examined as a whole [10].

C. The Communications Node Model

Communications node actors model application behavior (e.g., the Meter Data Management System, data concentrator, home metering gateway, utility meter, etc.) and the computational resources needed to execute that behavior. The behavior of communications node actors is defined in terms of tasks, which are executed by the communications node actors as part of domain use cases, such as scheduling a meter reading, or generally as part of their spontaneous behavior, such as updating the consumption database. A task is an action

or work—any behavior appropriate for the system they represent—that requires the use of computational resources. The task processing model can be parameterized with different (preemptive and non-preemptive) task scheduling schemes and different levels of processing capacity.

Use cases define the business contexts in which events (execution of tasks, and the transmission of messages), which represent the demands on computing and network resources, are generated by communications node actors. Each use case is characterized by a process flow map, which defines the flow of business processes from one communications node to another, and an associated data flow map, which defines the application messages exchanged between communications nodes to trigger business processes. Communications node actors do not pass logical messages to each other directly; instead, they rely on communications link actors to deliver their logical messages.

D. The Communications Link Model

The communications infrastructure is a major component of the SMI. In order to assess end-to-end system performance, predict the impact of change, or otherwise optimize technology decision-making (e.g., to minimize cost and complexity), properties of the underlying network, whether existing or planned, must be considered. In the CLEVER simulator, the route from the meter gateway to the MDMS is made up of a series of different communications links. The communications route is represented in the simulator as a series of communications link actors, each representing one or more steps of the communications route, as illustrated in Fig. 2.

Depending on the desired performance metrics, the modeling of network traffic can be on different levels or time scales. We considered carefully the tradeoff in computational resources required a detailed model of the link behavior, possibly at protocol level, and a more behavioral model represented by a queuing node characterized by throughput, loss, delay and jitter parameterized closely to link behavior known from other communications network studies for specific communications link technologies. We have chosen the latter in this first stage of the CLEVER simulator.

Communications link actors are abstract representations of the communications channels that allow communications node actors to communicate with each other. The communications link model is not tied to a specific communications technology; it is purposefully made to be generic enough to represent both wireless and wired link technologies. It models the behavior of application traffic by incorporating network attributes such as available bandwidth, latency, degree of packet loss (when there is insufficient capacity or resources), duplication of messages (e.g., due to retransmissions), and reordering of messages. The communications link model is made up of the following abstract components that together define its logical topology (how data moves from one endpoint to another):

- Link endpoints, to which communications node actors connect to send and receive logical messages. Endpoints are sources/targets of link services.

- Link services, which define the flow of logical messages from source endpoints to target endpoints (i.e., the logical topology of the link).

- Link service queues, with configurable size and management policy (e.g., FIFO, Priority, etc.).

- Link delay table, in which messages are held for a certain time once they have been served (i.e., serialized), to model deterministic propagation delay.

It may be easier to think of the communications link actor as a network switch device, where logical messages received by a communications link actor (through its endpoints) are multiplexed onto and de-multiplexed from the service queues managed by the communications link actor, the interconnection of which forms an end-to-end path between actors.

E. Data Collection

The simulator uses probes to collect statistics on messaging (e.g., event count, traffic count, message delay, response time), queuing (e.g., event count, waiting and service times, queue utilization), transmission (e.g., link utilization), and processing-related events (e.g., processing capacity utilization). The probes are fully configurable, allowing the simulator user to balance between the granularity of statistics collection and the computational resources and time required to run simulation experiments.

V. SIMULATION EXPERIMENTS AND RESULTS

One of the major advantages of smart meters over conventional meters is the ability to perform remote and frequent (on demand, or scheduled) collection of data. For example, suppliers can change the rate structure or introduce a time-of use model, where the rates take into account pricing trends in the electricity wholesale market. This type of model is only possible when a detailed view of usage is available, i.e., instead of a single monthly reading, utilities can choose to have automated readings at 15-minute intervals (or even less, to determine total consumption or rate of consumption, for consumption profiling).

Determining when and how data should be collected from potentially millions of devices is not trivial. Traffic flow in the SMI, from very large numbers of sources, is affected by many different variables that interact with each other (e.g., the time of collection, frequency of collection, data sizes, retransmission policies, available bandwidth of communications links, processing capacity of data aggregation points, and other traffic sources). The CLEVER simulator provides a means to study these interactions simultaneously in order to understand traffic distribution in time and topology, and identify areas of congestion so that necessary adjustments can be made to minimize it.

To demonstrate the capabilities of the simulator, we present here results from two selected meter reading simulation experiments, which were run on the University of Bristol's BlueCrystal High-Performance Computing Cluster [11]. The same scenario is run in each of these simulation experiments, the MDMS schedules meter readings by sending a command to
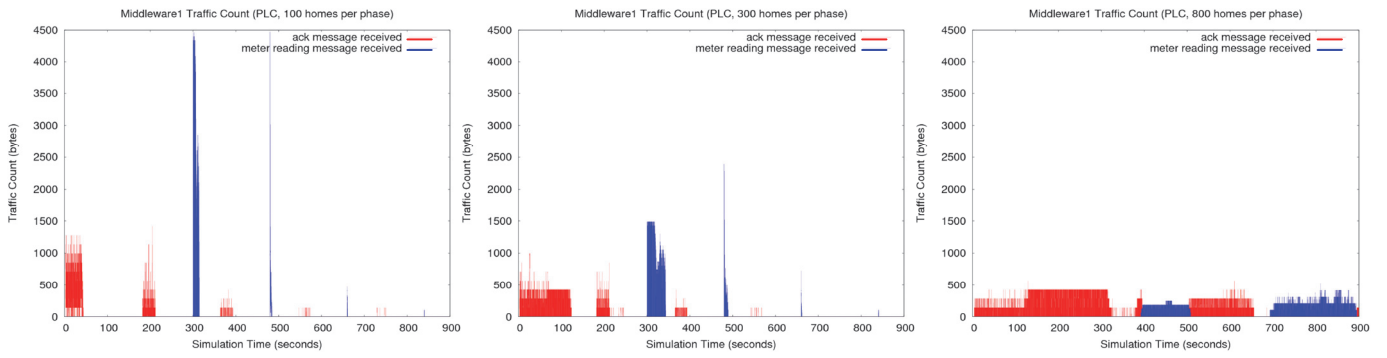
Figure 3. Aggregate traffic received by the MDMS for PLC with 100, 300, and 800 homes per phase.
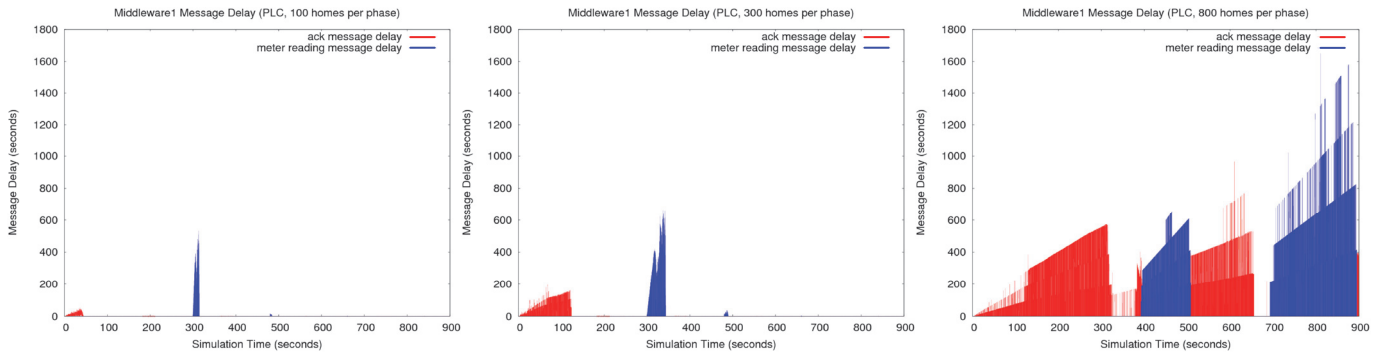

Figure 4. Aggregate delay for traffic received by the MDMS for PLC with 100, 300, and 800 homes per phase.

the meters, which activate scheduled readings to be sent to the MDMS at the specified start date, time, and interval. Meters verify that the requested reading schedule has been set up by sending an acknowledgement to the MDMS. Table I shows the parameters that were used for the various link technologies.

In the first experiment, we compared the aggregate traffic (and associated delay) of messages received by the MDMS for scheduled meter readings using power line carrier (PLC) with different numbers of homes on the phase (see Table II), with the total number of homes maintained at 7200 in all cases. All meters were scheduled to send readings at half-hourly intervals, starting at 5 minutes past midnight. Fig. 3 shows the aggregate traffic received by the MDMS in each case, and Fig. 4 shows the associated aggregate delay. We can observe from these two figures that as the number of homes per phase increases, the arrival rate at the MDMS decreases (and conversely, traffic delay increases). This is mainly due to congestion at the access links (the concentrators and backhaul links were set to a very high capacity). With a small number of homes per phase, all meter readings arrive at the MDMS over a short period of time (hence the spike in meter reading traffic in the plot left of Fig. 3) and they experience the shortest delay (among the three cases). For larger numbers of homes per phase, the access link becomes a bottleneck, causing meter readings (and other traffic, such as acknowledgements from recently scheduled meters in response to 'schedule readings' commands) to experience long delays at the access network (see the plot right of Fig. 4). As the access links are pushed to the limit, the arrival of traffic at the MDMS is spread more evenly over time

(no sudden increases in traffic, as shown by the plot right of Fig. 3). Although this may be good for the MDMS (no sudden spikes in the demand for service), service-level agreements may not be met due to the large delays experienced.

In the second experiment, we compared the impact of different access link technologies on meter reading performance. The scenario parameters used for different access link technologies are shown in Table II. In this experiment, meter readings (with 300-second interval), compete for access link bandwidth with other supplementary meter reading traffic (with 50-second and 100-second intervals), with all readings starting at approximately the same time (and with a 60-second spread). Fig. 5 shows the cumulative average delay for meter readings arriving at the MDMS using different access link technologies. The initial jump in delay is due to meter readings competing with not only supplementary readings, but also acknowledgements in response to schedule readings commands (sent at the beginning of the scenario). Following the initial jump, the cumulative average delay decreases at each interval, as the rate of arrival at the access link is less than the rate of departure. This is the case for all scenarios simulated. As can be seen in Fig. 5, the cumulative average delay of meter readings arriving at the middleware for power-line with 100 homes per phase is similar to that of long-range radio with 6680 homes per sector.

In these experiments, we used the simulator to identify bottleneck points in the end-to-end system, allowing for parameters to be adjusted, or different options to be considered

(e.g., using a different access technology with higher capacity for dense urban areas, or employing a different scheduling scheme for meter reading or other application traffic to avoid congestion at the access network).

TABLE I.   LINK PARAMETERS FOR EXPERIMENTS

Link Technology	Link Parameters	
	Available Data Rate (Up, Down)	*Delay*
Backhaul[a]	15Gbit/second, 15Gbit/second	1ms/100km
GPRS[b]	37kbit/second, 37kbit/second	90ms
Long-Range Radio	133kbit/second, 133kbit/second	90ms
PLC	3.4kbit/second, 6.0kbit/second	0ms

a. Data rates for the backhaul have been purposely set to a very large value in order to extract traffic statistics for these links.

b. The current implementation does not take into account the maximum number of sessions available.

TABLE II.   SCENARIO PARAMETERS FOR EXPERIMENT II

Access Technology	Scenario Parameters	
	Home Groups[c] : Access Link Actors : Concentrator Actors	*Total Homes : Total Actors*
Broadband	1000:1000:1	21000:84043
GPRS	3201:3:1	22407:67256
Long-Range Radio	6680:1:1	20040:60130
	9030:1:1	18060:54187
PLC	300:3:1	21000:63351

c. Each home group consists of a Gateway Actor, a HAN Actor, and a Meter Actor.
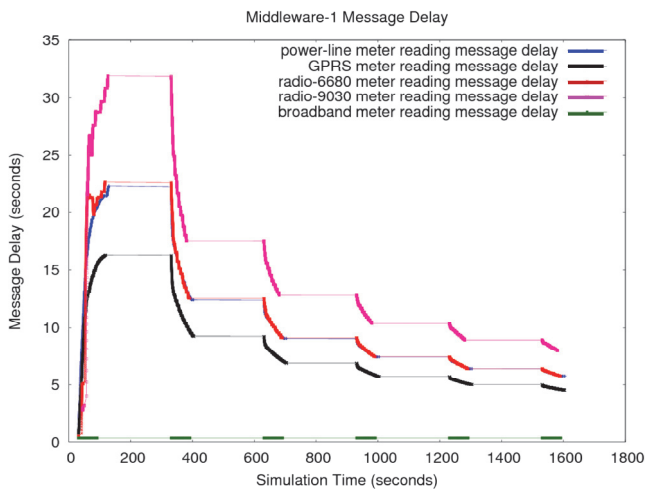

Figure 5.   Cumulative average delay at the MDMS for different access technologies.

VI.   CONCLUSIONS AND FUTURE WORK

The ICT infrastructure envisaged for smart energy solutions is both large and complex. In this paper, we described the CLEVER discrete-event simulator, which has been developed specifically for the purpose of evaluating the impact of architectural and technological choices, and service strategies, on the performance of smart metering infrastructures. The main requirements for the CLEVER simulator were to be able to integrate in one platform communications infrastructure models with smart metering applications and provide the capability to compare, on the same platform and in like-for-like conditions, the performance of the system end-to-end against a very wide range of parameters in real scale. Results from running simulation experiments demonstrate the capabilities of the simulator.

The model we have adopted is generic and flexible enough to be used for both the evaluation of basic functionality scenarios, which define the business requirements related to the management of metering assets, as well as scenarios with more advanced functionality such as load shedding and demand response. Work is currently ongoing to introduce additional simulation actors and to enrich the simulation models with more complex behavior to simulate smart grid applications and scenarios in real scale. Furthermore, we intend to evaluate other non-functional aspects such as availability and security as part of future work.

REFERENCES

[1] M. Pickavet, W. Vereecken, S. Demeyer, P. Audenaert, B. Vermeulen, C. Develder, D. Colle, B. Dhoedt, and P. Demeester, "Worldwide energy needs for ICT: the rise of power-aware networking," Proc.2nd Int. Symp. on Advanced Networks and Telecommunication Systems (ANTS), Bombay (India), December 2008.

[2] C. Develder, W. Haerick, K. Mets, and F. De Turck, "Smart Grids and the role of ICT," Proc. IEEE Smart Grid Comms Workshop, IEEE Int. Conf. on Communications (ICC), 2010.

[3] S. Harrison, A. Manson, and J. Brogden, "Smart Metering Operational Framework Proposals and Options", version 1, Energy Retail Association, August 2007.

[4] Averill M. Law, "Simulation Modeling and Analysis", 4th Edition, McGraw-Hill, August, 2006.

[5] R. M. Fujimoto, K. Perumalla, A. Park, H. Wu, M. H. Ammar, and G. F. Riley, "Large-scale network simulation: How big? How fast?", Proc. 11th IEEE/ACM Int. Symp. On Modeling, Analysis and Simulation of Computer Telecommunications Systems (MASCOTS), 2003.

[6] The Erlang Programming Language, http://www.erlang.org/.

[7] Meiler Page-Jones, *Fundamentals of Object-Oriented Design in UML*, Addison-Wesley, 2000.

[8] Wrapper for Object-Oriented Programming in Erlang (WOOPER), http://ceylan.sourceforge.net/main/documentation/wooper/.

[9] Sim-Diasca, http://www.sim-diasca.org/.

[10] Chee Hock Ng, Boon-Hee Soong, *Queueing Modelling Fundamentals: With Applications in Communication Networks*, 2nd Edition, Wiley, April, 2008.

[11] BlueCrystal High-Performance Computing Cluster, University of Bristol, http://www.acrc.bris.ac.uk/acrc/hpc.htm.