A Proposal for Performance Analysis and Dimensioning of IoT Networks

J.R. Emiliano Leite, Edson L. Ursini (IEEE member), Paulo S. Martins
School of Technology
University of Campinas (UNICAMP)
Limeira, Brazil
j750465@dac.unicamp.br, {ursini, paulo}@ft.unicamp.br

***Abstract*—One key element in the Internet of Things is the mediator. In order to successfully integrate this device, it is necessary the planning and dimensioning of its capacity. In this work, we tackle such challenge by using a discrete event simulation model that is validated with an analytical Jackson networks queue model. The resulting model and simulations attempt to capture several features that are the typical of the IoT systems such as RFID tags, AdHoc networks, the Internet, gateways and clusters with internal node mobility and their applications. Through a number of case studies, we showed that the model allows the estimation of the capacity of the mediator, as well as other critical parameters such as CPU utilization and mean network delay.**

***Index Terms*—mediator, dimensioning, Internet of Things, RFID, smart city, AdHoc, sensor**

I. Introduction

The Internet of Things (IoT) adopts simple and open technologies that have been consolidated in the telecommunications arena, such as the Ethernet, WIFI, AdHoc, ZIGBEE, Bluetooth, RFID (Radio-Frequency Identification), Sensors, WSS (Wireless Signal Solutions), Wireless Sensor Networks (WSN), IPV6, 6LowPAN, Wired, GSM and LTE among others.

It has targeted applications with direct impact in the world economy and society, such as Environmental, Energy (Smart Grid), Transportation, Healthcare, Retail, Supply Chain, People Tracking and Surveillance (Smart home, building, and city), Industrial and logistics among others.

Traditional Internet is also used through components such as the mediator, as well as technologies such as the Cloud and Fog. The most common way to access the databases is through cell phones. The use of RFID tags is also relevant for the identification of medical equipment, patients, caregivers and medicine, among others.

A key element in any IoT network is the so-called mediator, which is the device that concentrates the IoT traffic. It also accomplishes the pre-processing of data (i.e. filtering, classification, grouping, aggregation etc..) before it is transmitted to the application. It integrates the events with the business semantic. In essence, it performs the bridging of the physical world of RFID's and sensors with the business applications. The mediator allows the seamless interconnection of heterogeneous devices across distinct platforms, models and manufactures.

Dimensioning the realm of wireless technologies can take two forms: coverage and (node and link) capacity. This works concerns with the dimensioning of node capacity, and specifically the mediator, since it is the bottleneck of the network.

The goal of this paper is to investigate through a case study the mean queueing time and the CPU utilization for each node for a given probability of connectivity of the nodes resulting from the mobility. We also aim to estimate the incoming and outgoing traffic for each cluster, considering its connectivity, for RFID and sensor networks, including the mediator. It is possible to study the traffic in the mediator equipment and the throughput in the four applications. Specifically, we aim at dimensioning the capacity of an IoT network, which concentrates most data flows in the network. Thus, the traffic under consideration in this work converges to the IoT mediator.

The discrete-event network simulation model allows an approximate placement of the sensor nodes and RFID in a cluster, as well as their approximate traffic load, but it does not express their mobility. The mobility of nodes within a cluster is simulated through the Random Way Point (RWP) algorithm. A degradation in connectivity for some nodes may cause the overall reduction of the traffic in the upper layers. It may also increase the traffic in the surrounding nodes. In the latter case, the possibility of using emergency gateway nodes is crucial for many types of applications, as a measure to counteract the performance degradation in the affected nodes.

The remainder of this paper is organized as follows: In Section II we review previous work. The network model is discussed in Section III. A case study illustrating the application of the model is shown in Section IV. In Section V, the results are discussed. We summarize and present our conclusions in Section VI.

II. Related Work

The work by Karnouskos et al. [1] introduces the simulation of a Smart Grid city using software agents, including home appliances, an electric car and a power generating plant. It was possible to create a dynamic infrastructure that simulates a future smart grid city.

Capone et al. [2] presents the modeling and simulation for improvement of the power efficiency of the IEEE 802.15.4e DSME standard, targeting wireless sensor networks. The implementation of the simulation model is carried out in OPNET.

The authors analyze the power consumption and they propose a set of improvements for low-power applications, which allows significant power reduction.

IoT applications cannot be developed and tested without the simulation models and access to the online historic data. The work by Novak et al. presents a simulation of an integrated approach for a SCADA [3]. It supports the access to online complex data and the reuse of simulation code. The key idea is to integrate s imulation, data sources, optimization and a SCADA system into one framework. The authors provide the requirements for integration within the context of industrial automation. The solution was implemented and tested within the software prototype layer.

Chunxiao et al presents a Middleware (mediator) solution for RFID tags and ZIGBEE Sensor networks [4]. The mediator implements concentration, protocol conversion, communication across several applications, filtering, aggregation and transmission to application systems. It offers API interfaces so that the applications do not need to deal with the sensors level of the network.

Lu et al. present a K-coverage Fuzzy optimization procedure using an algorithm for plant growth simulation [5]. The optimization of the network planning is critical to the increase of the RFID network performance. The K-coverage model is conceived as a multidimensional optimization process with limiting conditions. The simulation results show the effectiveness of the proposed algorithm in comparison with other competing approaches. However, none of these works covered in the literature tackle the traffic aspects of the network such as partial and global delays and CPU utilization.

The work by Samaniego et al. [6] concerns with the management of heterogeneous IoT resources. It uses the model of virtual resources mapped onto physical resources, and the Constrained Application Protocol (CoAP, RFC 7252) for the communication of the virtual resource and the programming language. The architecture was developed in three layers: 1) View Abstraction Layer (VAL), 2) Hardware Abstraction Layer (HAL), and 3) Physical Layer (PL : Sensors). The experimental analysis adopted the emulation of clients. Similarly, their work does not address the traffic aspects of the network as accomplished in our paper.

Leite et al [7] address a bibliographic research on IoT and RFID, including the mediator, in applications such as environment/healthcare/smart city, and addressing simulation, security, Cloud and Fog, and so on. Leite et al [8] [9] simulate an AdHoc Network using a Discrete Event Simulation with ARENA and a MATLAB Mobility Model. The traffic of this network is used in the IoT Network of this paper, as Input to the mediator. Unlike the work in [8], in this work we tackle the performance of the queues, CPUs and traffic in an IoT network, in addition to the clusters of sensors in the AdHoc network.

III. Network Model

A TCP/IP packet is modeled as an entity that arrives to the system and crosses several internal queues in a cluster before

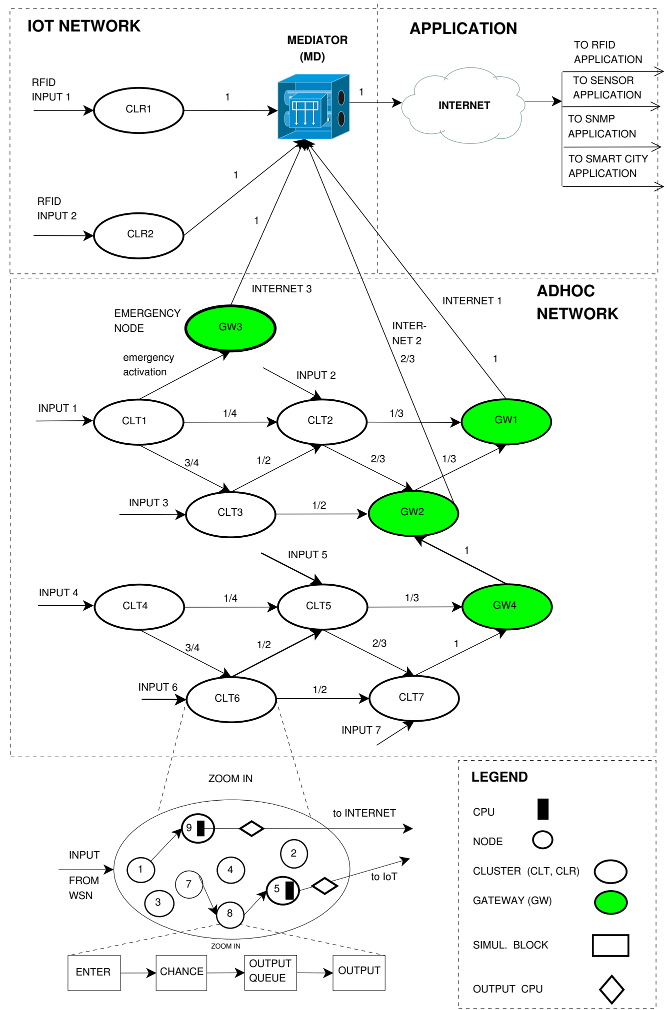

Fig. 1. IOT network model

its departure (i.e. before it is consumed by an application). The network model is a hierarchy consisting of clusters which contain nodes, which in turn have multiple CPUs, thus allowing several parallel connections. Inherent to each queue is the waiting delay before a packet can be processed by a server. Clearly, both queueing and processing times are subject to statistical distributions. Therefore, a network cluster may be regarded as a set of internal queues (each one associated with an outbound link).

The network components are the mediator, gateway and endpoints. The end points can be RFID and sensors for different applications. Figure 1 shows the network model with its inputs (packets) and outputs (packets) to each cluster. The upper part of the model is the IoT network, and its details are as follows:

- Mediator MD, which is a node and contains one or more CPUs;
- Two clusters (CLR_1, CLR_2), which perform the acquisition/input of RFID tags;
- Two RFID inputs, which receive data packets generated by IoT RFID tags (RFID reader);

- Four applications: RFID, Sensors, SNMP management and smart city;
- Internet, which models the traditional Internet;

The lower part of Fig. 1 is an AdHoc Network that generates data traffic which is aggregated by the IoT mediator. It consists of the following elements:

- Seven clusters ($CLT_1....CLT_7$); these are non-mobile and homogenous for the sake of simplicity. However, the model does not restrict the addition of heterogeneous clusters. Each cluster consists of n mobile nodes, where n is a configurable parameter;
- Four gateways or Internet nodes ($GW_1....GW_4$); Both GW_1 and GW_2 are output gateways; GW_3 is an emergency gateway, i.e. it is used as a backup gateway for GW_1, e.g. when the latter overflows its internal buffers; both GW_4 and GW_2 are protocol converters, i.e. they are used to integrate two subnets;
- Seven inputs: model data packets generated by IoT sensors;
- Three Internet outputs: they model the flow of IP packets outbound;
- Node mobility: the Distributed Dynamic Routing algorithm for mobile AdHoc networks is employed;
- Input variables: data arrival and service time distributions in a node;
- Control variables: probability of node connectivity in a cluster. This probability is provided by the Random Way Point algorithm (which depends on a range of variables such as receiver threshold, area size, antenna type, height, and gain, and system loss coefficient among others [8].
- Output variables: mean queue time and mean CPU utilization on each cluster for a given position of the nodes within the cluster.

Each cluster contains several nodes (Fig. 1) which in turn have internally one or more CPUs (only CPUs for nodes 5 and 9 are shown to avoid overcrowding the figure, and because they are connected to the cluster outputs). The model is dynamic and the illustration is only a snapshot representation of an arbitrary instant t in time. For example, at instant $t + 1$ it might be other nodes that engage in transmission outbound. In addition, each cluster has one or more output CPUs which are used for its output channels/links (Table I). These output CPUs are fixed in our model (without sacrificing the quality of the results), although it is possible to configure them to have some limited degree of mobility as well. Nodes share the output CPUs for relaying outbound traffic, provided that they have connectivity, i.e. they are within the power range of either an output CPUs or an intermediate node.

Each node is modeled as four simulation blocks connected in series:

1) *Enter block*: the enter block simulates the arrival of a packet in a cluster. It counts the number of packets entering the cluster;
2) *Chance* is an Arena DECIDE block, and it distributes the packets across a set of outgoing lines, where each line is associated with an outgoing queue; An important parameter in this block is the probability of packet loss, and its value was obtained from the case study (Section IV). The probabilities of a packet being forwarded to an outgoing link are initially configured as shown in Fig. 1 (e.g. 1/4 from cluster 1 to cluster 2 and 3/4 from cluster 1 to cluster 3);
3) *Output queue* represents the queueing time in the outgoing line;
4) *Output cluster* simulates the output (i.e. forwarding) of packets from the cluster. It is also responsible for counting the number of packets leaving the cluster.

TABLE I
NETWORK CONFIGURATION.

Function	Probability	Output CPUs
GA_{12}	1	25
GA_{34}	1	28
GA_{57}	1	29
GA_8	1	19
AdHoc submodel	1,1,1,1	30,31,32,33
CLR_1	1	21
CLR_2	1	22
MD	1/2, 1/2	24, discard
GWI	1/3,1/3,1/3	23, 26, 27
GW_1	1	5
GW_2	1/3, 2/3	11,6
GW_3	1	14
GW_4	1	17
CLT_1	1/4, 3/4	1,2,20*
CLT_2	1/3, 2/3	3,4
CLT_3	1/2, 1/2	7,8
CLT_4	1/4, 3/4	12,15
CLT_5	1/3, 2/3	16,13
CLT_6	1/2, 1/2	9,10
CLT_7	1	18

* output-CPU 20 is used only in an emergency

Table I shows the relation of cluster / gateways to output CPUs. The column “Probability” is associated to the column “Output CPUs”. Each probability is used to define the traffic management of each node according to a given application. These values also indicate the probability of a packet being serviced by the indicated output CPU. For example, the probability that cluster CLT_2 sends a packet to output CPU_3 is 1/3, and this probability is 2/3 for output CPU_4.

Each node receives packets at the input link and forwards them to one of the outbound links using UDP over IP (Datagram). Since the arrival of requests for the RFID and AdHoc networks can be modeled as a Poisson process, the traffic volume of each individual node can be extended to the traffic volume of a cluster by the simple sum of the rates of Poissonian arrivals. Thus, we sum the rates of each node to form a cluster of ten nodes.

The model adds four representative applications that process and consume the information leaving the mediator: 1) RFID, 2) Sensor, 3) SNMP management, 4) Smart Cities. The RFID, Sensor and Smart Cities applications receives information and stores in databases. RFID information is all based on traffic generated by RFID tags (96-bit EPC-Global). The SNMP application receives a trap from the mediator, and it sends a request to the same, which replies with a response. In typical IoT, the databases can be reached by mobile or cell phone applications through the secure HTTPS protocol.

IV. Case Study

To evaluate each node independently, a MATLAB routine generates random positions for the ten nodes within the cluster, every one sec (in our case). This case used the Random Waypoint Mobility Model (RWP) to simulate the performance of the network. By changing different parameters, we can either increase or decrease the connectivity. For example, it is possible to increment the connectivity by increasing 1) the number of user nodes, or 2) the number of gateways (interconnection), or 3) transmission power or else 4) by decreasing the simulation area, or a combination of these factors. Mobility determines the location of each node that selects a random destination, and travels towards it in a straight line at a randomly chosen uniform speed. The distance to connect nodes lies within the range from 200 to 500 meters. We used a 1000 $\times$ 1000 m area. The adopted mobility model is the one presented by Leite et al [9]. The probability of no connection P_f was calculated at 4.95%. The P_f value was used in an Arena Chance Block to represent the "Disconnected State" for sensor nodes (AdHoc subnet), or the probability of lost connection.

V. Results and Discussion

The most important result of this work is the analysis and estimation of the total traffic in the network of clusters considering the effect of mobility and fading, i.e. the traffic volume under dynamic conditions. We considered three scenarios, which are described as follows:

- *First Scenario - Overloaded Mediator*: In the first scenario, the inter-arrival times for a cluster are EXPO(0.6), i.e. one packet each 0.6 secs; thus the arrival rate is 1/0.6=1.67 packets/s (10 nodes $\times$ 0.167 packets/s per node). We used the service rate of 10 packets/s (i.e. 1/0.1). Some nodes were unstable since the service rate was less than the arrival rate and their utilization rate was close to 100%. This caused excessive delays and we could see the behavior (i.e. activation) of the emergency node. Thus, in the first scenario, a situation was forced in which the network presented instability, mainly to verify the performance of the emergency node.
- *Second Scenario - Mediator with Extended Capacity:* In the second scenario, the inter-arrival times for a cluster are EXPO(0.6), i.e. one packet each 0.6 secs (the arrival rate is 1/0.6=1.67 packets/s, i.e. 10 nodes $\times$ 0.167 packets/s per node). We used the service rate of 1/0.02, meaning that the service rate is 50, i.e. fifty packets are processed each second (a five-fold speed increase in relation to the first scenario). The increase in the service rate in the mediator (output-CPU-24) was enough to stabilize the system, as shown in Fig. 2. Therefore, the stable network caused the emergency node (output-CPU-20) to remain mostly inactive, i.e. only a few packets flowed through it. Recall also that in the actual system, the processing delay is different for each network element. The results that summarize the dimensioning of the mediator are shown in Table V.
- *Third scenario - Simulation validation - system without mobility*. In this scenario, we validate our simulation model with a Jackson queueing network model. Notice that we removed node mobility from the AdHoc network, thus ensuring 100% connectivity, so that we could validate the model. This is due to the fact that the analytical model adopted does not accommodate packet losses and mobility. Nevertheless, this assumption

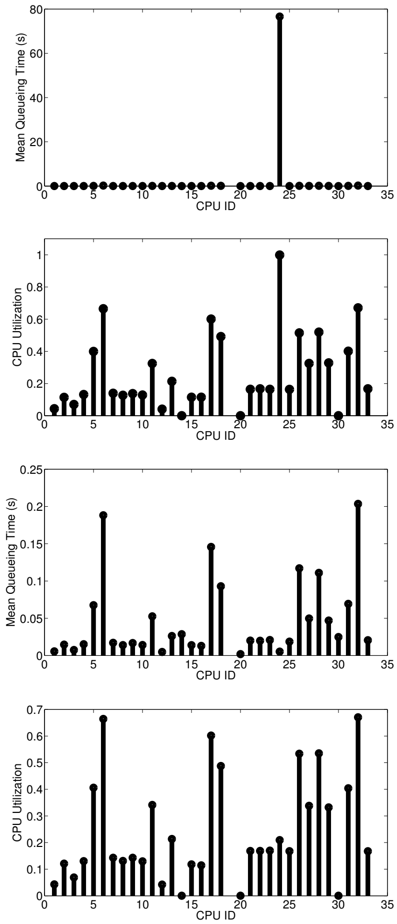

Fig. 2. CPU mean queueing time and utilization: overloaded mediator (top) and extended mediator (bottom)

TABLE V
RESULTS FOR THE MEDIATOR

Parameter	First Scenario (overloaded mediator)	Second Scenario (extended mediator)
arrival rate (packets/s)	20	20
service rate (packets/s)	10	50
mean-queue-time (s)	79	0.0056
CPU Util. (%)	100	20

TABLE II
R MATRIX: PROBABILITY OF TRANSMISSION PER OUTPUT LINK.

——	CLT1	CLT2	CLT3	CLT4	CLT5	CLT6	CLT7	GW1	GW2	GW3	GW4	CLR1	CLR2	MD
CLT1	0	1/4	3/4	0	0	0	0	0	0	0	0	0	0	0
CLT2	0	0	0	0	0	0	0	1/3	2/3	0	0	0	0	0
CLT3	0	1/2	0	0	0	0	0	0	1/2	0	0	0	0	0
CLT4	0	0	0	0	1/2	3/4	0	0	0	0	0	0	0	0
CLT5	0	0	0	0	0	0	2/3	0	0	0	1/3	0	0	0
CLT6	0	0	0	0	1/2	0	1/2	0	0	0	0	0	0	0
CLT7	0	0	0	0	0	0	0	0	0	0	1	0	0	0
GW1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
GW2	0	0	0	0	0	0	0	1/3	0	0	0	0	0	2/3
GW3	0	0	0	0	0	0	0	0	0	0	0	0	0	1
GW4	0	0	0	0	0	0	0	0	1	0	0	0	0	0
CLR1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
CLR2	0	0	0	0	0	0	0	0	0	0	0	0	0	1
MD	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CLT - cluster, GW - gateway, CLR - Cluster RFID, MD - mediator

TABLE III
ARRIVAL (λ) AND SERVICE RATE (Γ) PER NODE (PACKETS/SEC) FOR THE NETWORKS

—	CLT1	CLT2	CLT3	CLT4	CLT5	CLT6	CLT7	GW1	GW2	GW3	GW4	CLR1	CLR2	MD
Γ	1.67	1.67	1.67	1.67	1.67	1.67	1.67	0	0	0	0	1.67	1.67	5.0
λ	1.67	3.54	2.92	1.67	3.54	2.92	5.49	4.68	10.49	0	6.67	1.67	1.67	20

TABLE IV
MEAN QUEUE WAITING TIME PER *NETWORK* CPU: SIMULATION (W_s) VS. ANALYTICAL (W_a) RESULTS (*ms*).

—	CPU ID											
Time	1	2	3	4	5	6	7	8	9	10	11	12
W_a	4.4	14.3	13.4	30.9	88.0	233.0	17.1	17.1	17.1	17.1	53.8	4.4
W_s	4.9	14.8	6.4	18.7	75.3	266.0	17.6	16.3	16.7	18.3	58.3	4.5
——	CPU ID											
Time	13	14	15	16	17	18	19	20	21	22	23	24
W_a	30.9	–	14.3	14.3	200.3	121.7	–	–	20.0	20.0	–	5.0
W_s	32.2	–	15.7	13.9	209.6	131.1	–	–	23.2	20.6	–	5.6

CPU 19 - RESERVED (for future extensions).
CPUs 14 and 20: Emergency (not activated in this case study).
CPUs 23,25-33: Application CPUs, not contemplated in this example.

does not compromise the validity of the results. In this scenario, we also used the Jackson's network analytical model [10], which was calculated as a Markov chain to validate the simulation model under the exponential distribution for both arrival and service distributions. Therefore, note that the simulation model is not limited to the use of the Poissonian distribution initially assumed in this work. We have adopted this type of distribution since it allowed the validation of this model. However, once validated, it was possible to evaluate other conditions not allowed by the analytical model, such as different distributions other than the exponential, the inclusion of the loss of connectivity, and the probability distributions regarding the traffic between clusters. Moreover, the proposed model is flexible in that it is not restricted to the use of Random Waypoint mobility model, and other types of mobility models may be used and compared.

Since the initial simulation model has both exponential arrival and service distributions, it may be validated against Jackson's open queueing network model [10]. The solution is obtained from a Markov chain. The packet arrival rate is 1/0.6 = 1.67 packets/sec. The first seven arrivals, each generated by a cluster (gateways do not generate traffic), yield 1.67 packets/sec (the remaining four are gateway inputs), therefore: Γ = [1.67, 1.67, 1.67, 1.67, 1.67, 1.67, 1.67, 0, 0, 0, 0,1.67,1.67, 5.0]. We also need the 14 x 14 matrix R (Table II), which describes the probabilities shown in Fig. 1 . The total arrival rates in each cluster or gateway is given by the vector: $\lambda = \Gamma\ [I - R]^{-1}$, λ = [1.67, 3.54, 2.92, 1.67, 3.54, 2.92, 5.49, 4.68, 10.49, 0, 6.67,1.67,1.67, 20]. From the rates obtained (Table III), it is possible to calculate the mean queue delays for each CPU (Table IV) (W_i, [i=1....24]) by means of the equation (1), which gives the delay in an M/M/1 queue:

$$W_i = \frac{\lambda_i/\mu_i}{\mu_i - \lambda_i}, \qquad \mu_i = \frac{1}{0.1} = 10 \tag{1}$$

packets/s, where λ_i and μ_i are the rates for each CPU. Since all the delay values obtained from the simulation model matched the ones from the analytical model, the

ISSN 2447-8326. V.1. © 2017 BTSYM

simulation model may be deemed validated. This validation is a crucial step since it allows further extensions to this model, i.e. the inclusion of other model features such as new types of distributions.

We may observe in Fig. 2 that the mediator (output CPU-24) was overloaded (100% utilization), which caused the entire network to delay packet processing. By increasing its service rate up to five times, the utilization dropped down to 20%, and the overall latencies also (globally) reduced .

The mean network delay is given by equation (2) as follows:

$$\overline{W} = \frac{\sum \lambda_i}{\sum \Gamma_i} W_i \tag{2}$$

For the analytical model, we obtain $\overline{W_a}$ = 233.2 ms, and for the simulation model $\overline{W_s}$ = 247.3 ms.

The initial distribution adopted for the arrival and service rate was the exponential. This distribution is suitable since 1) it allows the validation of the model with an analytical model; 2) it is the one that stresses the network (the worst case when there is no bursts). Erlang (1) is the actual exponential. If the exponential distribution does not match the reality, it is possible to combine exponential distributions to form Erlang(k) distributions, which may better reflect and the actual traffic model in the network. The infinite summation of Erlang(k) distributions leads to a constant distribution. Otherwise, if there are bursts in network, the Pareto or Hyper-exponential distributions may be employed, depending upon the application. Once the model is validated by incremental evolution [11], other types of extensions may be studied.

Notice that the validation of the network was carried out in the third scenario. This is due to the fact that our first scenario the network was found to be congested, which precludes validation with the Jackson analytical model, which assumes a balanced network. The second scenario corrected this condition and the third scenario validated the model, by adding the constraints of null mobility and full connectivity. The numerical results of the simulation can be made more precise (i.e. closer to the values from the analytical model) by adjusting a number of factors, e.g. both the number of simulation runs and also the size of the simulation duration.

The proposed model is general and it can be instantiated for specific applications. For example, the probabilities of transmission for outgoing links can be measured in a real application and replaced in the model. The arrival and service distributions considered may also be replaced by actual measurements and/or other types distributions.

VI. Conclusions

One of the most critical components of an IoT network is the network mediator, as it is a traffic concentrator. Therefore, the growth of such networks is largely dependent upon on the capacity of such device. In order to establish such limits, one must resort to simulation and analysis of the traffic generated by the plethora of technologies that are covered by the IoT.

In this work, we have tackled the issue of dimensioning the capacity of the mediator, by considering distinct types of traffic and the diversity of devices that may compose the system. The simulation model presented was based on both discrete event and Random-Way Point simulation for an AdHoc network that accommodates clusters and the effects of node mobility. It captured a number of features of complex systems, including the mediator, gateways, emergency nodes, Internet and IoT traffic, and most important the traffic generated by RFID tags and sensors. We analyzed through a case study the mean queueing time and the output-CPU utilization for each node for a given probability of connectivity of the nodes resulting from the mobility. Through the model it was possible to estimate the incoming and outgoing traffic for each cluster and the IoT mediator, which was the focal point and the initial goal of this work.

In future work, we would like to enhance the model with a traffic growth prediction model, which will allow a more precise estimation of the spare capacity that should be considered in the design of the mediator. Furthermore, the mobility model used in this work assumed for illustration purposes a lower mobility of the devices. We also contemplate exploring this issue by increasing the mobility of the nodes in the network to a point that it models a VANET.

The model that was presented is scalable, meaning that it accommodates the inclusion of other types and number of components, by using the concept of clusters. Therefore, we argue that it may provide a useful tool for dimensioning devices within other applications, including the context of smart cities.

References

[1] S. Karnouskos and T. N. d. Holanda, "Simulation of a smart grid city with software agents," in *2009 Third UKSim European Symposium on Computer Modeling and Simulation*, Nov 2009, pp. 424–429.

[2] S. Capone, R. Brama, F. Ricciato, G. Boggia, and A. Malvasi, "Modeling and simulation of energy efficient enhancements for ieee 802.15.4e dsme," in *2014 Wireless Telecommunications Symposium*, April 2014, pp. 1–6.

[3] P. Novk, R. indel, and R. Mordinyi, "Integration framework for simulations and scada systems," *Simulation Modelling Practice and Theory*, vol. 47, pp. 121 – 140, 2014.

[4] C. Fan, Z. Wen, F. Wang, and Y. Wu, "A middleware of internet of things(iot) based on zigbee and rfid," in *IET International Conference on Communication Technology and Application (ICCTA 2011)*, Oct 2011, pp. 732–736.

[5] S. Lu and S. Yu, "A fuzzy k-coverage approach for rfid network planning using plant growth simulation algorithm," *Journal of Network and Computer Applications*, vol. 39, pp. 280 – 291, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S108480451300177X

[6] M. Samaniego and R. Deters, "Management and internet of things," *Procedia Computer Science*, vol. 94, pp. 137–143, 2016.

[7] J. R. E. Leite, E. L. Ursini, and P. S. Martins, "Integration of iot with rfid and sensor networks: A literature review (in portuguese)," in *Brazilian Technology Symposium (BTSym 2016)*, December 2017.

[8] ——, "Simulation of adhoc networks including clustering and mobility," in *AdHocNow-2017*, September 2017.

[9] J. R. E. Leite, E. L. Ursini, and P. S.Martins, "Performance analysis of a multi-mode adhoc wireless network via hybrid simulation," in *Brazilian Telecommunication Symposium (SBrT 2017)*, 2017.

[10] J. R. Jackson, "Networks of waiting lines," *Operations Research*, vol. 5, no. 4, pp. 518–521, 1957.

[11] E. L. Ursini, V. S. Timoteo, V. F. Santos, and P. S. Martins, "An ip vpn network design and dimensioning approach using analytical-simulation models with incremental validation," in *Proceedings of the Winter Simulation Conference (WSC-14)*, December 2014, pp. 4095–4096.