

# Simulation of AdHoc Networks Including Clustering and Mobility

J.R. Emiliano Leite, Edson L. Ursini, and Paulo S. Martins<sup>(✉)</sup>

School of Technology, University of Campinas (UNICAMP),  
R. Paschoal Marmo, 1888, Limeira, Sao Paulo 13484, Brazil  
`paulo@ft.unicamp.br`

**Abstract.** We introduce a discrete-event simulation model of an AdHoc network considering the presence of clusters and node mobility. The main goal is to study the volume of traffic in relatively large networks of sensor systems and Internet of Things considering fading and network connectivity. We also evaluate relevant parameters such as the mean CPU utilization and the mean queueing time in each node. The model is relatively general in that it combines the traffic from small devices such as sensors as well as more complex intermediate systems such as gateways and Internet nodes. It is also extensible to other types of scenarios and it allows the evaluation of the network under other performance criteria or evaluation metrics. The results show that the model yields simulation values that could be analytically validated by Jackson networks.

**Keywords:** Emergency services · MANET · VANET · WSN · IoT · Traffic planning · Network dimensioning · Connectivity · Clustering

## 1 Introduction

Node mobility is a critical element in the design of stable, scalable and adaptive clusters in mobile AdHoc networks (MANETs). It has a significant impact on network performance and it must be accounted for in the analysis of such networks. Another key architectural component for the self-organization of an AdHoc network is the presence of network clusters. Both features (i.e. mobility and clusterization) are thus essential when planning and dimensioning the next generation AdHoc networks.

In this work, we provide a simulation model based on discrete event (for network traffic) and Random Way Point (RWP) (for mobility) simulation for an AdHoc network that accommodates clusters and the effects of node mobility. The model captures a number of features of complex systems, including gateways, emergency nodes, Internet and IoT traffic. The goal 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.

Despite the existence of a body of work on AdHoc networks, none of the studies cited and surveyed in the literature (Sect. 2) tackles the features of the AdHoc under consideration in our work, e.g. the issue of functions to increase network robustness (emergency clusters). Additionally, they do not address a more comprehensive simulation model for traffic performance evaluation - with the adoption of discrete event simulation (with Arena simulation software), dynamic mobility model (with MATLAB), and the extension of traffic analysis - allowing the inclusion of clusters.

The remainder of this paper is organized as follows: In Sect. 2 we review previous work. The proposed approach is introduced in Sect. 3. A case study illustrating the application of the model is shown in Sect. 4. In Sect. 5 the results are discussed. We summarize and present our conclusions in Sect. 6.

## 2 Related Work

Amis *et al.* arrange the network in clusters using the Max-min algorithm [1]. In our work, clusters are formed according to the application under consideration, i.e. nodes are grouped according to their functional similarity. Furthermore, our major goal is to study the traffic load flowing through the clusters for the purpose of dimensioning these devices and their links. Pramanik *et al.* [7] use MATLAB to evaluate the connectivity of an AdHoc network by means of the Random Way Point algorithm. However, the authors neither make use of discrete event simulation of network traffic nor arrange the network in clusters. Nassef [5] shows the influence of fading and mobility in network node connectivity. This is incorporated in our work through the adopted mobility model.

Although clustering has traditionally been used for increasing the scalability of networks, the work by Phanish and Coyle also opts for multi-level clustering large-scale AdHoc and sensor networks as a means to support energy-efficient strategies for gathering data [6]. Two design variables are influential to the optimality of a multi-level network: (1) The number of levels, and (2) The number of nodes operating at each level. These variables are characterized within a multi-hop, multi-level hierarchical network of variable size that gathers and aggregates data at each level. Unlike the work by Phanish and Coyle [6], our model considers a single level of clustering. As mentioned earlier, our goal is the dimensioning of network traffic.

In the work by Cai *et al.* [2], the authors present a clustering algorithm based on the group mobility and a revised group mobility metric obtained from the speed and direction of nodes. The authors also account for the residual energy of nodes and the number of neighbor nodes in their model. Unlike our work, Cai *et al.* [2] uses the Gauss Markov instead of the Random Way Point mobility model, and their goal is to compose more stable clusters including highly dynamic environments. Our model has more stable clusters due to the sensor networks, and it includes mobility and connectivity to evaluate the total volume of traffic.

Ren *et al.* [8] proposed a new mobility and stability-based clustering algorithm (MSCA) for urban scenarios. The performance of the proposed algorithm

was assessed by changing the maximum lane speed and the traffic flow rate. As mentioned earlier, our model considers more stable clusters, as it is formed by sensor networks, targeted to IoT applications, which are not meant to be as mobile as e.g. VANETs, and it takes into consideration mobility and connectivity of the internal nodes for the evaluation of the total traffic volume. This volume is estimated by means of a combination of discrete-event simulation with the Random Way Point mobility model.

Celes *et al.* [3], using discrete-event simulation with NS-3, point out that five widespread vehicular mobility traces present gaps, potentially leading to error prone operation. They present a framework allowing more fine-grained traces, which leads to more trustworthy simulation results. The authors employ clustering algorithms to fill the gaps of real-world traces. They also compare the communication graph of the original and the calibrated traces using network metrics. The results reveal that the gaps lead to unrealistic network topologies. Like our model, their work approaches simulation with a previous trace of node mobility. Our approach also employed simulation with clustering and mobility, and since we also model sensor network nodes, we have a previous knowledge of the approximate location of clusters. Nevertheless, our goal differs from theirs since we aim at estimating the traffic for the purpose of dimensioning and planning network capacity.

### 3 Network Model and Proposed Approach

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

Figure 1 shows the network model with its inputs (packets) and outputs (packets) to each cluster. The details are as follows:

- 7 clusters ( $CLT_1 \dots CLT_7$ ); 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;
- 4 gateways or Internet nodes ( $GW_1 \dots GW_4$ ); 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;  $GW_4$  and also  $GW_2$  are protocol converters, i.e. they are used to integrate two subnets;
- 7 Inputs: the inputs model data packets generated by IoT sensors;
- 3 Internet outputs: they model the flow of IP packets outbound;
- Node mobility: the Distributed Dynamic Routing algorithm for mobile AdHoc networks is employed;

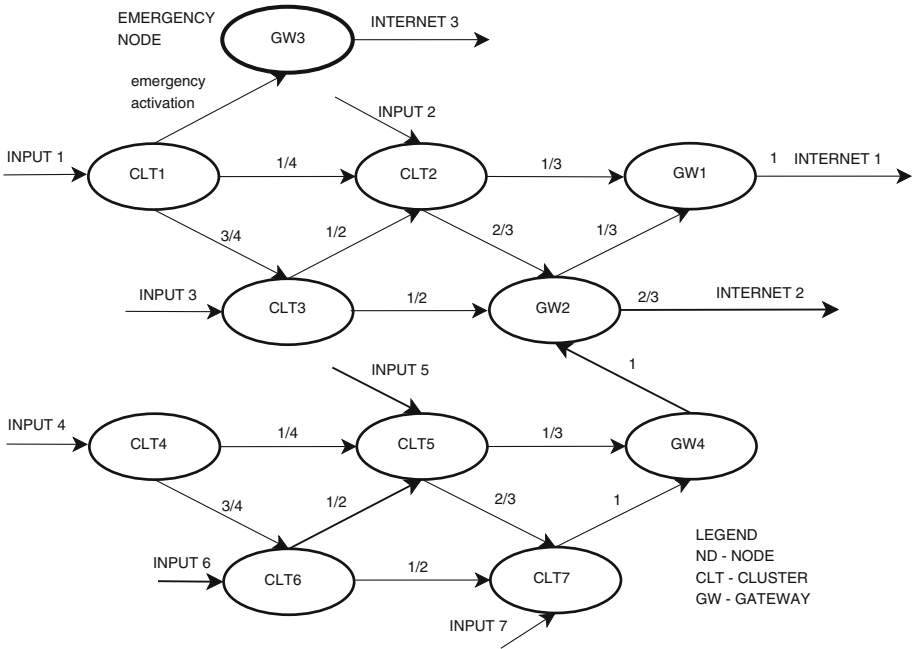


Fig. 1. Mobile AdHoc network model

- 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 (Table 2));
- 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. 2) 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 to that, each cluster has one or more output CPUs which are used for its output channels/links (Table 1). 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 4 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;

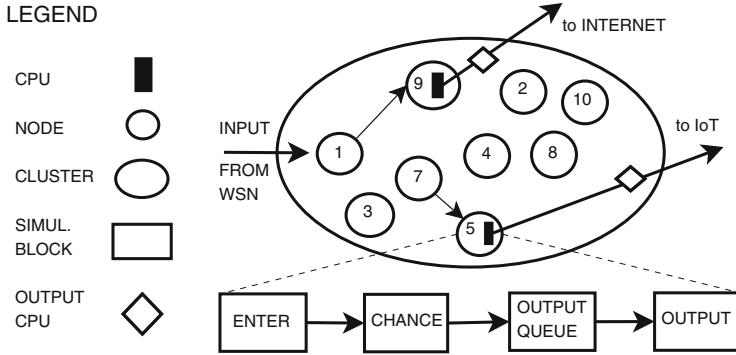


Fig. 2. Cluster organization model (snapshot)

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 (Sect. 4). 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 1 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 AdHoc network 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.

## 4 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). Table 2 shows the input parameters for the MATLAB algorithm. 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

**Table 1.** Network configuration.

Function	Probability	Output CPUs
$CLT_1$	1/4, 3/4	1, 2, 20 <sup>a</sup>
$CLT_2$	1/3, 2/3	3, 4
$GW_1$	1	5
$GW_2$	1/3, 2/3	11, 6
$CLT_3$	1/2, 1/2	7, 8
$GW_3$	1	14
$CLT_4$	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
$GW_4$	1	17

<sup>a</sup>Output-CPU 20 is used only in an emergency

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 m. We used a  $1000 \times 1000$  m area. The adopted mobility model is the one presented by Pramanik [7].

Two basic propagation models (FS = Free Space and TR = Two-Ray ground propagation model) were considered, which are described by the following equations:

$$P_{t,FS} = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L}; \quad d = \sqrt{\frac{P_t G_t G_r \lambda^2}{(4\pi)^2 P_{t,FS} L}} \tag{1}$$

$$P_{t,TR} = \frac{P_t G_t G_r h_t^2 h_r^2}{d^4 L}; \quad d = \left( \frac{P_t G_t G_r h_t^2 h_r^2}{P_{t,FS} L} \right)^{1/4} \tag{2}$$

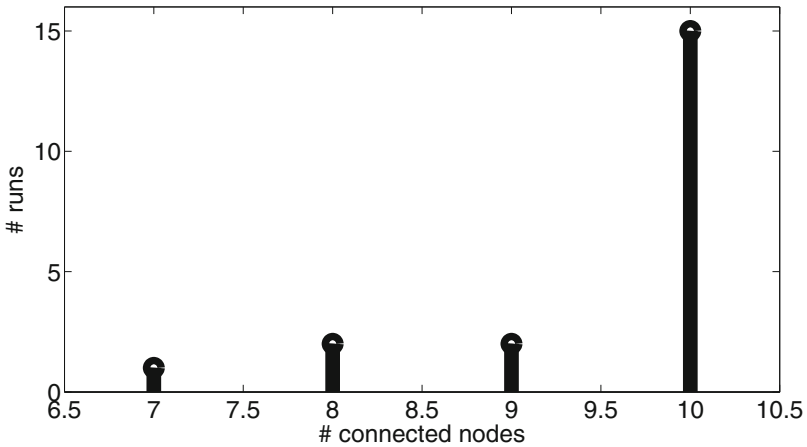
where  $d$  is the minimum distance (in meters) required for connection between a pair of nodes. Using the values from Table 2, we obtain  $d_{FS} = 582$  m and  $d_{TR} = 564$  m. Depending on the scenario (indoor, free space) it is possible to switch from one propagation model to another.

In this case study, since both values obtained are close, we adopted the more conservative value of 500 m. By running the model (MATLAB simulation) 20 times, we observed distances larger than 500 m;

The histogram in Fig. 3 illustrates the number of nodes that remained connected per observation (i.e. simulation run). In 15 simulation runs we had 10 nodes connected, 2 runs yielded 9 nodes, 2 runs ended up with 8, and one execution

**Table 2.** MATLAB input parameters

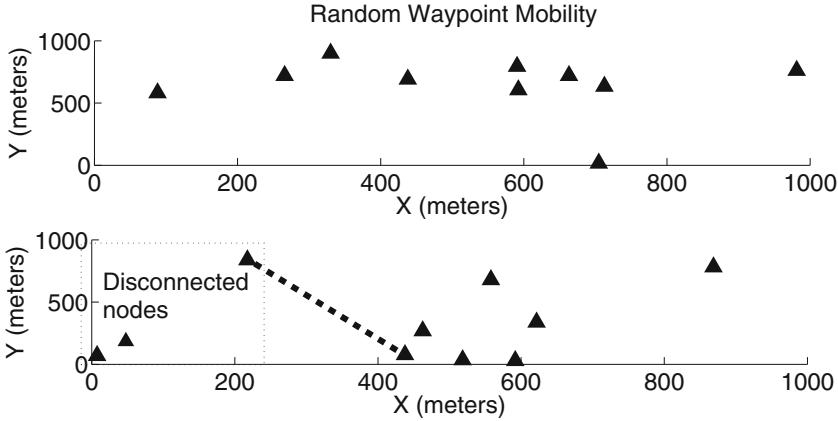
Input parameters	Values
Receiver threshold	-88 dBm
Area size	1000 × 1000 m
Antenna type	Omnidirectional
Antenna height (ht, hr)	1.5 m
Antenna gain ( $G_t, G_r$ )	1.0
System Loss Coefficient (L)	1.0
# Mobile nodes in a cluster	10
Mobility model	Random waypoint
Speed interval	[0.2–2.2] m/s
Pause interval	[0–1] s
Walk interval (walk time)	[2–6] s
Direction interval	[-180 to +180]°
Transmission frequency	5.8 GHz
Transmission power (Pt)	15 dBm
Simulation time	306 s

**Fig. 3.** Number of active nodes in a cluster.

resulted in 7 nodes (in a total of 20 rounds). Thus, the probability of no connection  $P_f$  is given by:

$$P_f = 1 - (15 \times 10 + 2 \times 9 + 2 \times 8 + 1 \times 7) / 200 = 191 / 200 = 0.495 (4.95\%). \quad (3)$$

i.e. we had 9 transmissions without success at 200 possible transmissions ( $20 \times 10$  nodes), or = 0.045 or 4.5%. The  $P_f$  value was used in an Arena Chance Block to represent the “Disconnected State”, or the probability of lost connection.



**Fig. 4.** Node connectivity due to distance.

Figure 4 shows two examples with different node positions. Whereas in the first (top) graph all nodes are connected, in the second (bottom) example three nodes are disconnected since they are 500 m apart from the remaining nodes in the cluster. The importance of these parameters is such that it justifies the use of both Arena and MATLAB simulators. If we use a distance of 250 m (or less) instead of 500 m, the blocking probability (or no connection) could be less than 50%.

## 5 Results and Discussion

This work is part of a major effort to dimensioning the capacity of an IoT network, where the bottleneck lies in the upper levels of the architecture, e.g. at the mediator, which concentrates most data flows in the network. Notice that 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 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 given by the 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 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 have executed two runs, described as follows:

- *First Run:* In the first run, the inter-arrival times for a cluster are EXPO(0.2), i.e. one packet is received each 0.2 s ( $1/0.2 = 5$ ), thus generating an arrival rate of 5 packets/s. We used the service rate of  $1/0.33$ , meaning that the service rate is 3, i.e. three packets are processed each second. 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 run a situation was forced in which the network presented instability, mainly to verify the performance of the emergency node. By running the complete model (1.000 s) in the simulation model, it was possible to extract the following traffic load at the Gateways:
  - (1) Traffic at the Internet Gateway ( $GW_1$ ): 5.2550 packets/s;
  - (2) Traffic at the Internet Gateway ( $GW_2$ ): 2.2350 packets/s;
  - (3) Traffic at the Emergency Gateway ( $GW_3$ ): 0.3910 packets/s.
- *Second Run:* In the second run, the inter-arrival times for a cluster are EXPO(0.6), i.e. one packet each 0.6 s (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  $1/0.1$ , meaning that the service rate is 10, i.e. ten packets are processed each second (a threefold speed increase in relation to the first run). The increase in the service rate was enough to stabilize the system, as shown in Fig. 5. 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.

By running the complete model (5.000 s) for the second run, it was possible to extract the following traffic load at the Gateways: (1) Traffic at the Internet Gateway ( $GW_1$ ): 4.3828 packets/s; (2) Traffic at the Internet Gateway ( $GW_2$ ): 6.3736 packets/s, and (3) Traffic at the Emergency Gateway ( $GW_3$ ): 0.0030 packets/s.

In the second run, we also used the Jackson's network analytical model [4], 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.

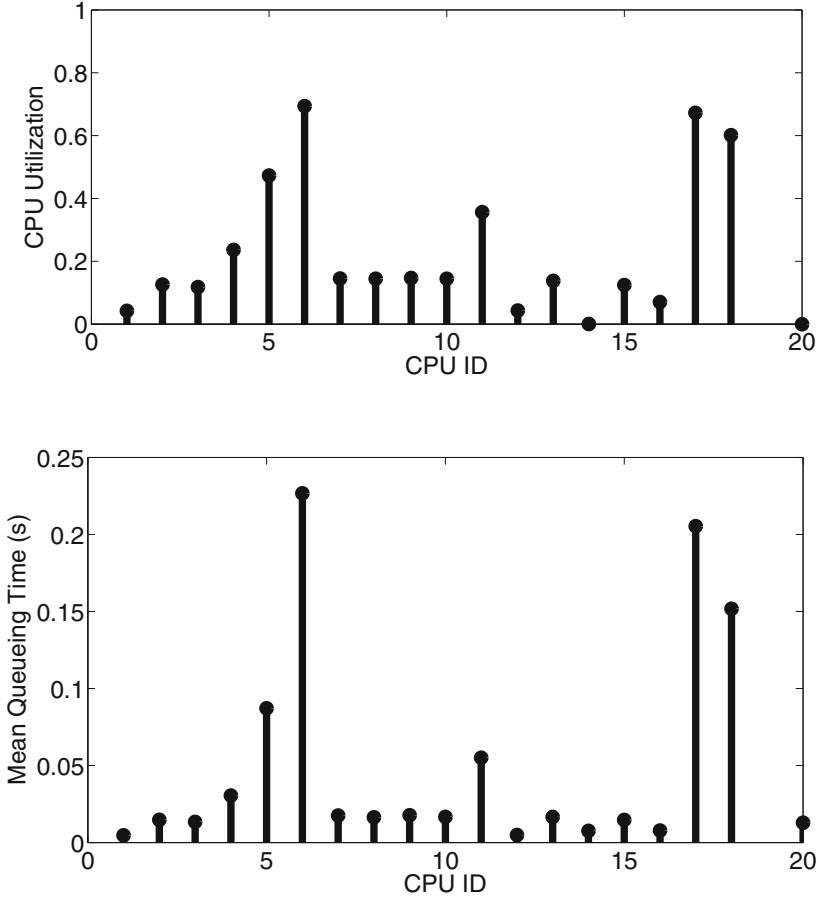


Fig. 5. Mean output-CPU utilization (upper) and mean queueing time (lower) (for second run)

## 6 Conclusion

In this work, we presented a simulation model based on both discrete event and Random-Way Point simulation for an AdHoc network that accommodates clusters and the effects of node mobility. The model captures a number of features of complex systems, including gateways, emergency nodes, Internet and IoT traffic. 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, considering its connectivity.

Ongoing work attempts to include fuzzy logic for decision making regarding emergency cases. This fuzzification would allow a fully dynamic model, which

reacts by triggering the emergency node based on the actual traffic, instead of using the current procedure that is based on the probability distribution of traffic volume. We intend to use this model for dimensioning and planning the IoT network mediator. Future work may also contemplate specific applications as well as configurations regarding clustering and mobility/connectivity. Another possibility is to apply prediction methods to evaluate network traffic in order to improve future dimensioning.

## References

1. Amis, A.D., Prakash, R.T., Vuong, H., Huynh, D.: Max-min cluster formation in wireless ad hoc networks. In: Proceedings of Ad Hoc Infocom 2000 (2000)
2. Cai, M., Rui, L., Liu, D., Huang, H., Qiu, X.: Group mobility based clustering algorithm for mobile ad hoc networks. In: 2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS), pp. 340–343, August 2015
3. Celes, C., Silva, F., Boukerche, A., Andrade, R., Loureiro, A.: Improving VANET simulation with calibrated vehicular mobility traces. *IEEE Trans. Mob. Comput.* **PP**(99), 1 (2017)
4. Jackson, J.R.: Networks of waiting lines. *Oper. Res.* **5**(4), 518–521 (1957)
5. Nassef, L.: On the effects of fading and mobility in on-demand routing protocols. *Egypt. Inform. J.* **11**, 67–74 (2010)
6. Phanish, D., Coyle, E.J.: Application-based optimization of multi-level clustering in ad hoc and sensor networks. *IEEE Trans. Wireless Commun.* **PP**(99), 1 (2017)
7. Pramanik, A., Choudhury, B., Choudhury, T.S., Arif, W., Mehedi, J.: Simulative study of random waypoint mobility model for mobile ad hoc networks. In: IEEE Proceedings Global Conference on Communication Technologies (GCCT 2015), pp. 112–116 (2015)
8. Ren, M., Khoukhi, L., Labiod, H., Zhang, J., Veque, V.: A new mobility-based clustering algorithm for vehicular ad hoc networks (VANETs). In: 2016 IEEE/IFIP Network Operations and Management Symposium, NOMS 2016, pp. 1203–1208, April 2016