

Unmarked Point and Adjacency Vertex, Mobility Models for the Generation of Emergency and Rescue Scenarios in Urban Areas

ROMMEL V. TORRES¹, JUAN C. SANCHEZ¹ AND
LUIS MENGUAL GALAN²

¹*Departamento de Ciencias de la Computación y Electrónica,
Universidad Técnica Particular de Loja, 1101608, EC
E-mail: rovtor@utpl.edu.ec, jcsanchez7@utpl.edu.ec*

²*Facultad de Informática, Universidad Politécnica de Madrid, 28660, ES
E-mail: lmengual@fi.upm.es*

Received: April 1, 2012. Accepted: March 13, 2013.

In disaster situations, where ad hoc mobile networks are normally used, the location and quantity of existing obstacles is random. Most existing mobility models have not been developed with consideration of the obstacles that exist in a disaster environment. This paper proposes two methods of mobility that realistically represent movement in an environment with obstacles. Unmarked Point Model (UPM) uses a high granularity strategy and Adjacency Vertex Model (AVM) uses a method that selects the shortest pathway. UPM consumes more resources than AVM to generate node mobility patterns in an ad hoc network, on a real map area of an urban area, where obstacles have been placed to simulate an emergency and rescue scenario. In addition, a comparative analysis of both models in its routing performance is done using AODV [18].

Keywords: Wireless, ad hoc, mobility models, simulation, urban areas, emergency and rescue

1 INTRODUCTION

In emergency and rescue scenarios mobile ad hoc networks (MANET) can create a temporary wireless network from mobile nodes without requiring a preset network infrastructure [6].

MANET networks in a disaster area allows the use of everyday technological devices such as cell phones, PDAs, tablets and other mobile devices [13] to establish communication between rescue teams and survivors. Furthermore, the generation of a temporary network is achieved independently and collaboratively in order to lessen the effects of the disaster.

The use of simulation tools allows us to determine and analyze the behavior of MANET networks without incurring the resource expenditures [13]. A key element in the simulation is the generation and use of a mobility model to determine the behavior of each displacement node [6].

Mobility models must simulate human movement [16], because in emergency and rescue scenarios, especially in urban areas, there are obstacles that prevent the movement of vehicles and limit the people's mobility. [6] Furthermore, each of these obstacles can also lessen signals and even prevents network connections between nodes.

Visual output tools generate node movement in emergency and rescue scenarios in visual form, such as Ad-hockey [1], but they use perfect geometric shapes such as quadrilaterals and simple lines for obstacles, causing the node to interact in an artificial environment.

This paper proposes two models of human mobility: the first is called UPM, inspired by basic deductions of "free space - occupied space" and the second is called AVM and is based on human movement (HUMO) [6] and the use of the shortest path concept (Dijkstra). The implementation of the two proposed models is performed in the SCENGEN [19] scenario generation tool.

To validate the models, we propose a method that generates emergency and rescue scenarios using real maps to simulate the state of the city after a disaster.

The paper is structured as follows: Section II discusses existing mobility models. In Section III and IV, we present the UPM mobility model and the AVM mobility model respectively. Section V shows the process of generating an emergency and rescue scenario and finally, in Section VI, we compare the proposed models by measuring the computational performance and AODV behavior within a simulation in Network Simulator 2 (NS2) [7].

2 RELATED WORK

Mobility models generally fall into two groups: models based on traces that are based on actual movements [7] captured by electronic devices and stored in large databases of information [4, 15, 22], and synthetic models, whose behavior is based on the use of mathematical models [15].

Synthetic models can be further classified according to their similarity to human movement, into realistic synthetic models and unrealistic synthetic models. The realistic synthetic models are divided into models with spatial dependency and models with geographic dependency. Within the unrealistic synthetic models, random models and models with temporal dependency can be found.

Random Models: Are the models where the nodes have free movement in the simulation area. Their direction, speed, and acceleration characteristics, etc. are randomly generated causing abrupt changes in node movement. These models include Random Walk Mobility Model, Random Way-Point Model, Random Direction Model and Boundless Simulation Area Model. Random Walk Mobility Model is the basis of several random patterns and its movement is considered the most unpredictable of all [5, 21]. The Random Way-Point Model is the model used for the simulation of protocols in ad-hoc networks [14]. The Random Direction Model seeks to break the concentration of dots in the center of the area [5]. The Boundless Simulation Area Model uses the relationship between the previous direction and speed with the current one and works within a limitless area so that a node that leaves the area's boundary goes back to its opposite side [5].

Models with Temporal Dependency: These models try to correct the weaknesses of realism of previous models, such as the sudden stop and the drastic change of speed and direction, using a comparison process between the current and next values of speed, acceleration and direction. They are representations of mobility that do not conform to human patterns [5, 9]. Some of these models are: the Gauss-Markov Mobility Model and the Smooth Random Mobility Model. The Gauss-Markov Mobility Model was designed to adapt to different levels of randomness via tuning parameters. A node starts with a speed and an assigned direction, which after a certain period of time is reassigned [5]. The Smooth Random Mobility Model, controls sudden changes of speed to make a change in direction through the use of a slow-down before time runs out [3].

Models with Spatial Dependency: These include the mobility models that handle node groups that are commonly governed by the node movement [6, 10]. Among the models that belong to this group, we can find: Reference Point Group Mobility Model, Column Mobility Model, Pursue Mobility Model and Nomadic Mobility Model. In the Reference Point Group Mobility Model, an arbitrary motion model determines the group movement while its internal movement (each node) is associated with a reference point [8, 10]. In the Column Mobility Model, an arbitrary motion model determines the group movement based on movement along a straight line (grid reference) that moves in a given direction [5]. The Pursue Mobility Model uses the same

strategy used by police to track a thief. Once implemented in the model, the nodes chase after a target [5]. The Nomadic Mobility Model's name evokes the distant, past human behavior when people traveled in wandering groups governed by a leader [5] and functions as such.

Models with Geographic Dependency: The models with geographic restriction emphasize the solution of stage limitations, which force an efficient mobility of nodes. There are two variants of geographic dependency: the Pathway Models and the Obstacle Models. Pathway type models are developed for scenarios that are based on obtaining pathways for the movement of nodes to reach their target. These pathways can be representations of city streets, roads or highways, which depending on the flexibility of the established scenario, must abide by road restrictions on the roads such as speed limits and traffic lights, among others [7, 15]. The Obstacle Models are models based on solutions obtained from the defined obstacles in a scenario. Generally, it achieves the implementation of solutions based on movements in its vertex, such as the Human Obstacle Mobility Model (HUMO), using mathematical models or the use of graphic modeling [15].

Exist some studies in the mobility models area, by example, Karamshuk *et al.* [11] shows a survey for human mobility models but does not take into account their relationship with emergency and rescue scenarios. Papageorgiou *et al.* [17] shows a mobility model from the rescue team perspective only, our research take both perspectives together rescue team and victim.

The two proposed models, UPM and AVM, belong to the group of realistic synthetic mobility models with geographic restrictions. Figure 1 shows the relationships between our proposed models and the studied models.

3 UNMARKED POINT MODEL (UPM)

The UPM qualifies as an Obstacle Model because it handles the analysis of "free - busy" space to make its movement and thus generates its pathway. The UPM is different from the rest of the models for two reasons. First, because it can work on scenarios containing obstacles with non-geometrical or amorphous shapes, because its movement is based on the low-level treatment of the image, that is, in scenarios where obstacle images present great granularity or pixelation, as shown in Figure 2.

The UPM was developed by taking into consideration the analysis performed by a person in a scenario crowded with obstacles similar to those in an emergency and rescue scenario. Indeed, an individual must analyze a range of possibilities before making his next move, e.g. before crossing a

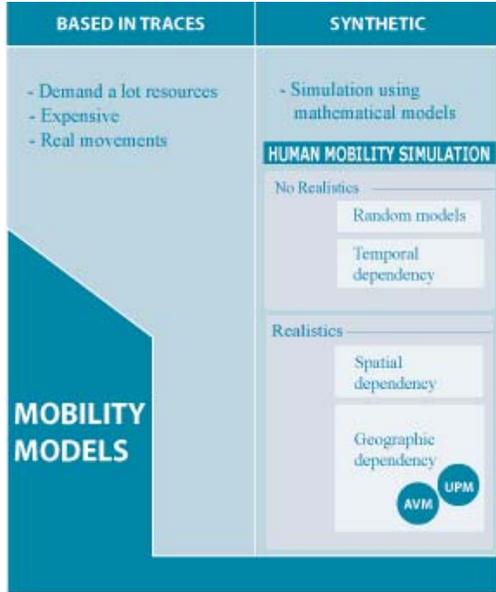


FIGURE 1
Mobility models

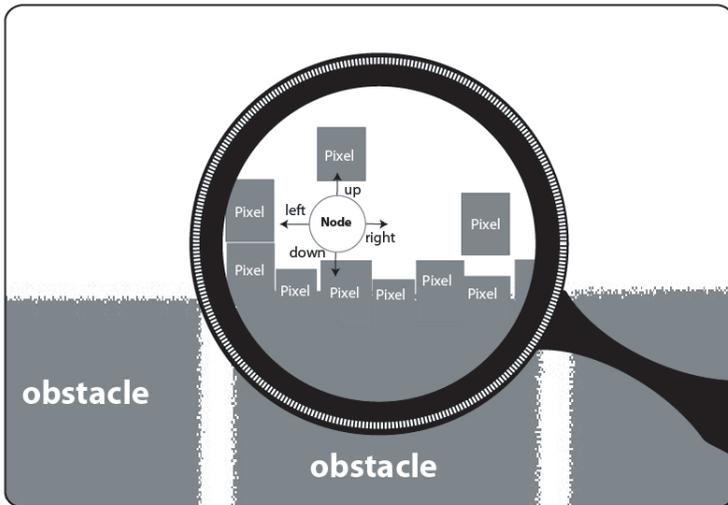


FIGURE 2
Pixelated surfaces

river. Node movement is done through verification and use of open spaces adjacent to the node's position in any of the four cardinal points.

Given a node's current position, the following elements are used to define its next position:

Obstacles

The obstacles are randomly distributed in a binary matrix generated by a map of a real city, to simulate a disaster area (see Section V, which describes the generation of a disaster scenario). Occupied spaces are represented with a 1 and the free spaces are represented with a 0. This permits it to be possible to distinguish the presence of an obstacle or a point that can become part of the pathway to the target point.

Movement

The node starts from a randomly selected position obtained through the application of a uniform distribution that is responsible for finding a point with a value of 0 in the matrix. It then evaluates the four closest values to the current position, always estimating the approaching pathway to a defined target point, which in the case of an emergency and rescue scenario, could be an evacuation or aid point. The target point is the destination to be reached. The movements performed are made in a row or a column. The process is shown in Figure 4.

Pathway

UPM proposes an exhaustive way to move as it works in areas with many obstacles. Unlike models such as HUMO or OAM, it is not based on the vertex of the polygon, but on the state of the next free position. Algorithm 1 shows the selection process of the near free position point to evacuation point using a matrix of neighboring free/occupied locations.

In the evacuation point of this model, if a node is in a position (x_a, y_b) to approaching its target point in (x_n, y_n) , it has to choose the free space of its next positions $(x_a \pm \kappa, y_b)$ or $(x_a, y_b \pm \kappa)$; which is convenient to reach the position (x_n, y_n) as shown in Figure 3.

We can represent UPM model based in the General Random Walk Model. The major difference between them is κ that in UPM is equal to a length measurement unit. If the node is located in the (x_a, y_b) position then each movement is represented by:

$$S = (l, \theta, \Delta t_f, \Delta t_p) | (x_a \pm \kappa, y_b \pm \kappa) \in B_{3\kappa \times 3\kappa} \quad (1)$$

Where

$b_{i,j} = 0$ next free position

$l = \kappa$ length of step

Algorithm 1 UPM node movement.

Require: A binary matrix where it is detailed free and occupied spaces.

Ensure: Next node movement point (x,y).

```

1: if     $((pX < pY) \wedge (p\_target\_X <> pX)) \vee (p\_target\_Y == pY)$ 
   then
2:   if  $(p\_target\_X < pX)$  then
3:     if  $(left\_free)$  then
4:        $(node\_left\_move)$ 
5:     else
6:        $(node\_down\_move) \vee (node\_up\_move) \vee (node\_right\_move)$ 
7:     end if
8:   else
9:     if  $(right\_free)$  then
10:       $(node\_right\_move)$ 
11:    else
12:       $(node\_down\_move) \vee (node\_up\_move) \vee (node\_left\_move)$ 
13:    end if
14:  end if
15: else
16:  if  $(p\_target\_Y < pY)$  then
17:    if  $(down\_free)$  then
18:       $(node\_down\_move)$ 
19:    else
20:       $(node\_left\_move) \vee (node\_up\_move) \vee (node\_right\_move)$ 
21:    end if
22:  end if
23: end if

```

$$\theta = [0, 360]$$

Δt_f = time of movement

Δt_p = time of pause

Each node moves a distance κ , in the UPM model the time of movement is in function of κ . Each pause is used by the node to decide the next step.

4 ADJACENCY VERTEX MODEL(AVM)

The AVM uses the Pathway Model and Obstacle Model strategies because its movement is defined by the existence of points given to the obstacles and also because the nodes have to make a quantitative comparison of distances to

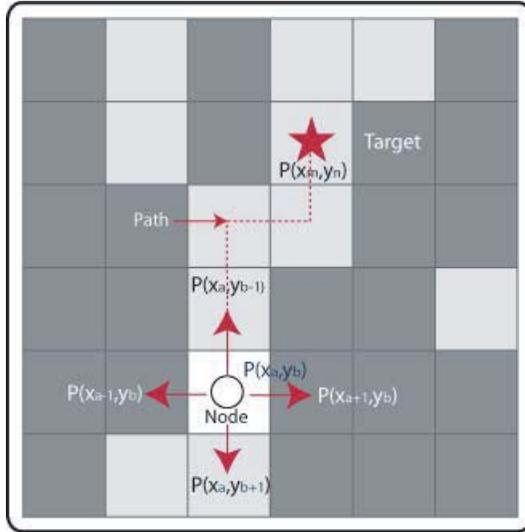


FIGURE 3
UPM node movement

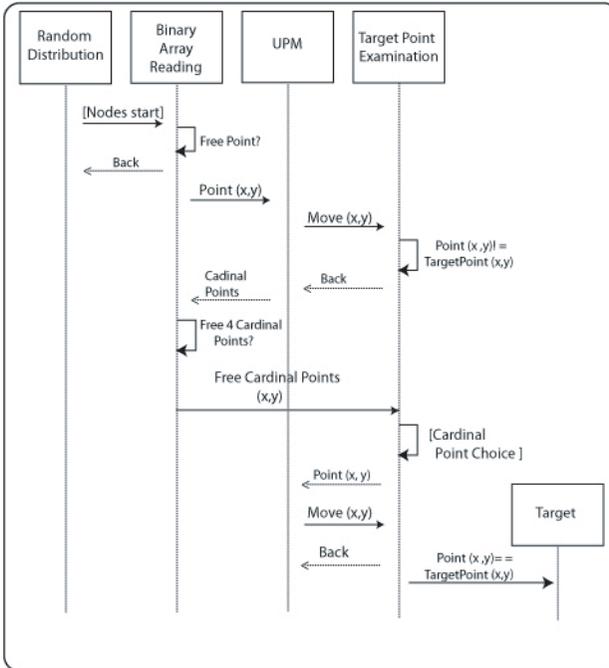


FIGURE 4
UPM node movement diagram

make their next move. This movement is similar to HUMO movements [6,16] and its improved variant, Obstacle Avoidance Mobility (OAM) [6]. AVM is also Pathway model because nodes move according to previously established pathways on an adjacency matrix [19].

The AVM is similar to the OAM model as it is based on the human mobility model (HUMO); it also applies the concept of the shortest path to reach the target. However, the model is built around two operating strategies that make it a hybrid model, i.e. a model simultaneously based on obstacles and pathways. The first strategy is used for the generation of the obstacles and their pathways, while the second strategy is to generate the movement of randomly distributed nodes after a disaster.

The AVM differs from the other models because it uses an adjacency matrix of vertex, which enables the formation of areas with very simple or complex obstacles, through modifying adjacency points or eliminating vertex.

Given a node's current position, the following elements are used to define the next position:

Obstacles

These are randomly distributed obstacles in a file that contains a binary matrix that represents a segment of a real city map, simulating a disaster area. The vertexes of each polygon are marked with a digit greater than 1. Each vertex contains a number of adjacent vertexes that are close to the visible points which can be reached directly. This information is stored in an adjacency matrix.

Movement

The node starts from a position selected randomly according to a uniform distribution; it immediately calculates the nearest vertex position by quantitative comparison of the results of the distance between the vertex position and the target or evacuation point. It does this using the Euclidean distance formula between two points, as indicated by in Algorithm 2.

Pathway

The pathway generated from a starting point to another target is determined through the shortest jump (referred as the shortest distance in the algorithm). Adjacency matrix is used for getting the distances and marking the vertices already visited. As indicated in the Figure 5 the pathway uses the distance for cost calculation, the neighboring free obstacle vertex with lower distance is selected. Figure 6 shows the interactions of the components in a use case diagram.

We can represent AVM model based in the General Random Walk Model where the node is located in the (x_a, y_b) position. A node can reach the target

Algorithm 2 AVM node movement.**Require:** An adjacency matrix**Ensure:** Next nodes movement point (x,y).

```

1: for (var2  $\leftarrow$  0, ....., num_max_nodes) do
2:   if ((pX == max[var2][0])  $\wedge$  (pY == max[var2][1])) then
3:     start_node2  $\leftarrow$  var2
4:     var2  $\leftarrow$  num_max_nodos
5:   end if
6:   if (pX  $\neq$  max[target_node][0])  $\wedge$  (pX  $\neq$  max[target_node][1])
7:     then
8:       for (var32  $\leftarrow$  0, ....., num_max_colum[2]) do
9:         a = max[start_node2][var3] - max[target_node][0]
10:        b = max[start_node2][var3 + 1] - max[target_node][1]
11:        cost  $\leftarrow$   $\sqrt{[(a)^2 + (b)^2]}$ 
12:        if (var3 == 2) then
13:          mincost  $\leftarrow$  cost
14:          pX  $\leftarrow$  max[start_node2][var3]
15:          pY  $\leftarrow$  max[start_node2][var3 + 1]
16:        else
17:          if (cost < min_cost) then
18:            min_cost = cost
19:            pX  $\leftarrow$  max[start_node2][var3]
20:            pY  $\leftarrow$  max[start_node2][var3 + 1]
21:          end if
22:        end if
23:      end for
24:    end if

```

point (x_n, y_m) across N a set of neighboring intermediates free obstacles vertex (x_i, y_j) . Then each movement is represented by:

$$S = (l, \theta, \Delta t_f, \Delta t_p) | (x_i, y_j) \in N \quad (2)$$

Where

$$N = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_m)\}$$

$$l = \sqrt{(x_i - x_a)^2 + (y_j - y_b)^2}$$

$$\theta = [0, 360]$$

Δt_f = time of movement

Δt_p = time of pause

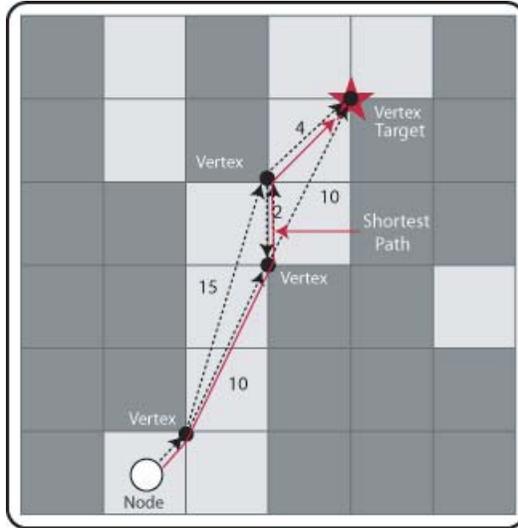


FIGURE 5
AVM node movement

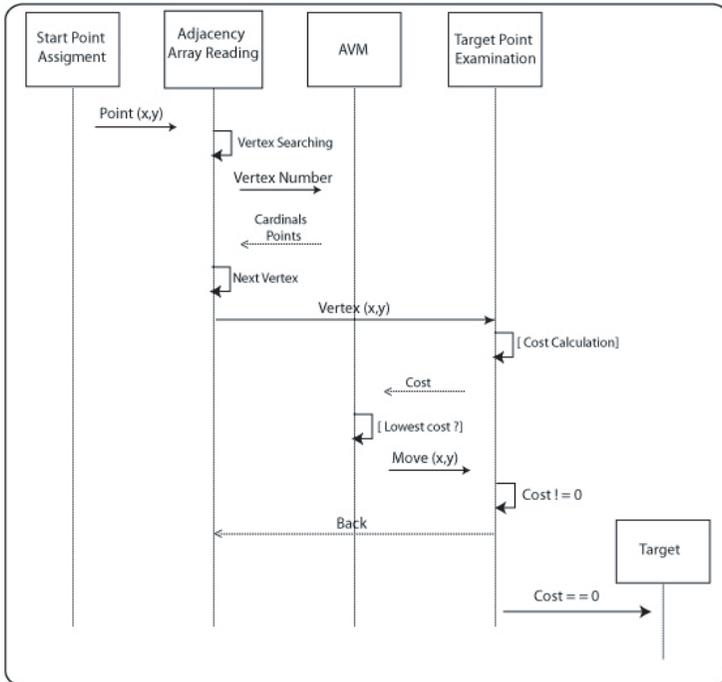


FIGURE 6
AVM node movement diagram

5 GENERATION OF DISASTER SCENARIOS BASED ON REAL MAPS

To validate the UPM and AVM models, we will use a segment plan of a real city, 2 km x 1 km, located in Loja city, Ecuador in South America, a panorama of the city is shown in Figure 7.

Based on a real map of the city, we will proceed to select the appropriate area and start to edit the image, removing unnecessary architectural details in the resource graph, only preserving blocks of the structures. The map was

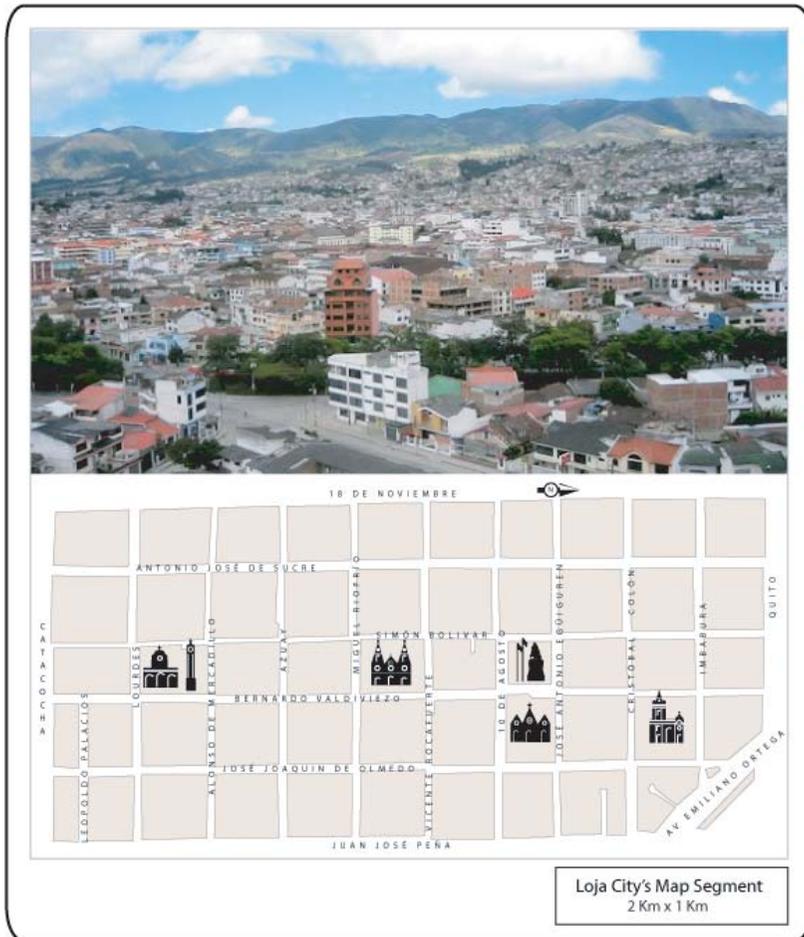


FIGURE 7
Abstraction of one segment of the map of Loja city-Ecuador

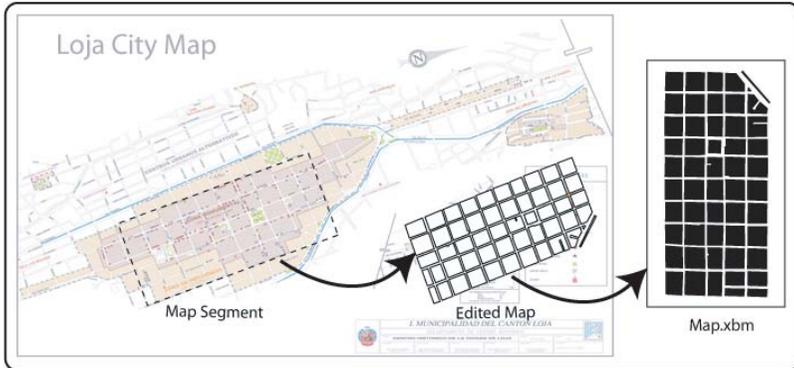


FIGURE 8
Obtaining XBM image file

then converted to a bitmap format called Exchange monochrome BitMAP (XBM). This format was chosen because its internal composition facilitates the construction of the binary matrix representing the scenario. This procedure is summarized in Figure 8.

The internal code of the image is transformed using a program developed in C++; the resulting binary data that will help to control the mobility of the UPM. Figure 9 summarizes the expressed idea.

Manipulating the binary file obtained above and using a program developed in C++, we randomly generate obstacles in the image. This result is indicated in Figure 10.

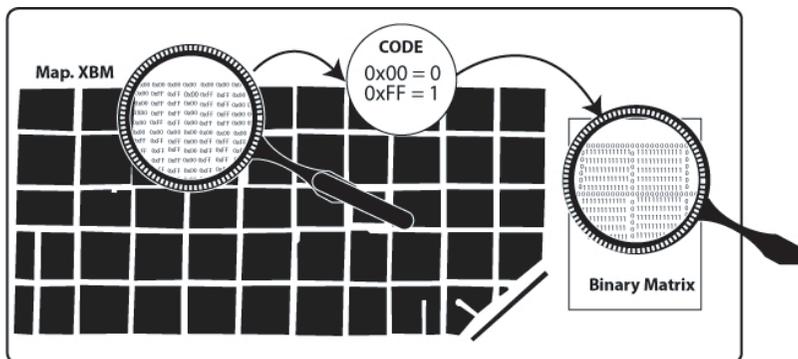


FIGURE 9
Binary representation of the XBM image file

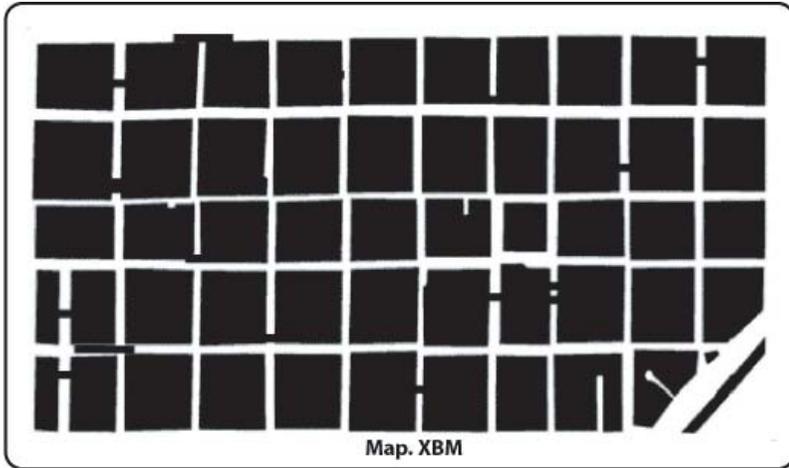


FIGURE 10
Disaster area with obstacles generated (Map.XBM)

Figure 11 shows the way we obtain the adjacency points of one vertex. After that, we determinate the node's adjacency vertex coordinates and we put them within one preformatted structure.

UPM and AVM have been implemented in the SCENGEN tool. The software developed in C++ generates scenarios for the NS2 tool. SCENGEN implements 6 mobility models, none of the belongs to the geographic restriction type: the data are generated randomly using uniform distribution or the

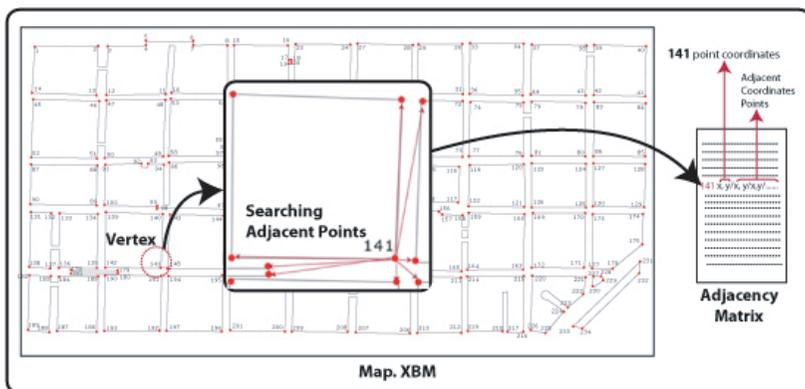


FIGURE 11
Obtaining of adjacency points matrix file

Gauss-Markov distribution. There are many simulation tools. Among them are:

- Scengen is an application developed in C++ to generate mobility scenarios in a fast and simple way.
- NS2 environment was developed in C++ and TCL. It's widely used in the academic research community for creating and evaluating protocol, simulating communication models and more [20].
- NS3 is a modern tool eliminates complexity produced by the use of both C++ and TCL. It has fewer mobility models implemented than NS2. [20].
- BonnMotion is a program made in JAVA program aimed at the creation and analysis of mobility scenarios [2].
- Mobi-Real is a tool for realistic simulation of human and vehicle movement with variation of behavior depending on the context of the application [12].

The SCENGEN tool was chosen for the following features:

- It is an application that creates mobility scenarios for the NS2 simulation tool.
- Its code is open, light, and available online and fully developed in C++.
- It allows new patterns of mobility.
- It facilitates the visualization of generated movements.

6 EXPERIMENT RESULTS

6.1 Computational Performance

The computational performance is the evaluation of the efficiency of resource use hardware and software resources for a particular task. We used three metrics: the scenario generation time, processor usage and the number of generated lines. Table 1 and Table 2 show the computational performance for UPM and AVM by node density.

In Figure 12, we can see a marked difference in runtime between the two models. The high time consumption for each UPM runtime tends to increase linearly in relation to the increasing number of nodes. In comparison, the AVM runtimes are minimal.

Figure 13 indicates the consumption of processor resources. UPM uses all the processor resources, while AVM, the use of processor resources is directly related to the number of nodes.

Finally, Figure 14 shows that the UPM generates a greater amount of movements than the AVM. The amount of movement affects directly the simulations performed in the NS2 simulator.

Node density	Scenario generation time (s)	Processor Usage (%)	Number of generated lines
20	11.942	100	457
30	13.781	100	674
40	18.243	100	893
50	29.769	100	1124
60	39.153	100	1309
70	50.514	100	1589
80	42.267	100	1786
90	53.371	100	2036
100	89.322	100	2248

TABLE 1
Data obtained with UPM

Node density	Scenario generation time(s)	Processor Usage(%)	Number of generated lines
20	0.341	25	206
30	0.137	32	322
40	0.351	41	428
50	0.513	53	539
60	0.529	62	629
70	0.859	71	758
80	0.738	80	847
90	1.347	89	972
100	1.689	94	1084

TABLE 2
Data obtained with the AVM

In conclusion, it can be determined that AVM exhibits a more efficient use of resources than UPM, essentially because UPM has very detailed movements that are resource demanding.

6.2 Relationship with the ad hoc routing protocol

Since mobility models have a direct influence on the efficiency of a protocol in a network, it is necessary to measure their behavior by analyzing a controlled simulation. Table 3 shows the parameters used in the simulation. The selection of the AODV routing protocol is discretionary, and it does not affect the results, since it is common to all mobility models.

The simulation is develop with NS2 using scenarios previously created and 20 TCP connections for evaluating the AODV protocol performance.

Evaluation of each viewpoint is done according to the following metrics: rate of received packets, delay average, application layer efficiency, routing layer efficiency and dropped packets.

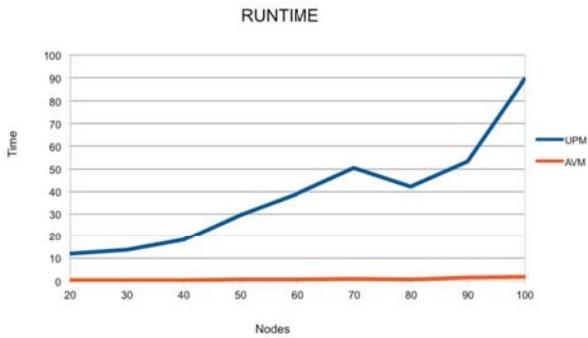


FIGURE 12
Generation time of a scenario

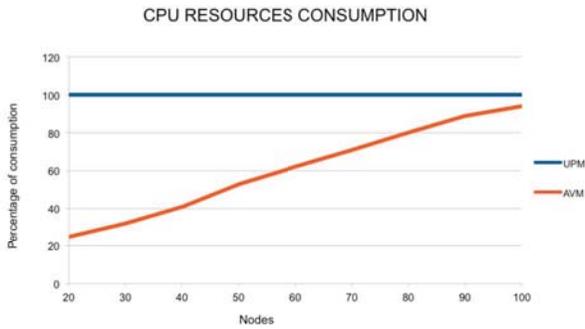


FIGURE 13
CPU utilization

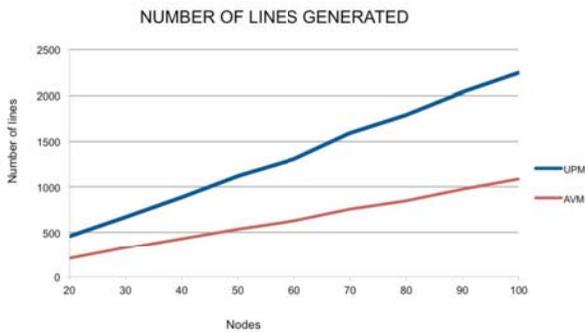


FIGURE 14
Number of generated lines

Item	Detail
Routing protocol	AODV
Nodes density	20, 30, 40, 50, 60, 70, 80, 90, 100
Mobility models	UPM, AVM
Evacuation point	Static in the map
Number of connections	20 connections (All to Node 0)
Dynamic viewpoint	Node 0 moves
Static viewpoint	Node 0 doesn't move

TABLE 3
Simulation parameters

According to Dynamic viewpoint, all nodes are moving simulating the behavior of the survivors in a disaster area, they communicate each other for mutual aid to reach the destination, which is an evacuation point. In the static viewpoint, node 0 is located at the evacuation point without moving anywhere, staying connected to all nodes and pretending to be emergency or rescue equipment. The following section is related to an analysis of the two mobility models according to the two points of view previously indicated.

Figure 15 shows the routing protocol efficiency when it interact with both UPM and AVM models. Best values are close to one. Lower values show indicators when higher packet loss exists. If more nodes belong to the network then the efficiency decreases, because the probability unstable routes is greater. Since UPM is a model with shorter movements, its efficiency is higher than AVM.

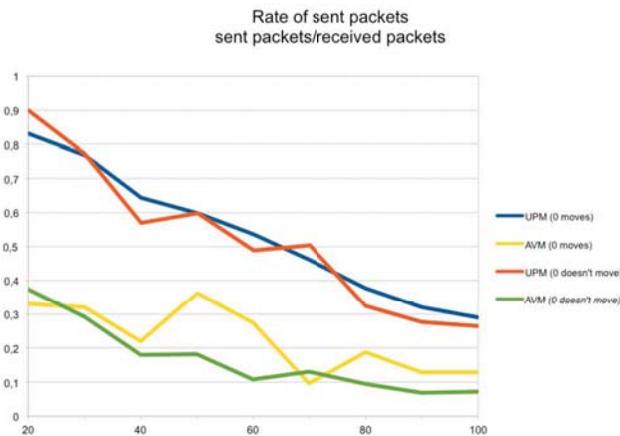


FIGURE 15
Rate of sent packets

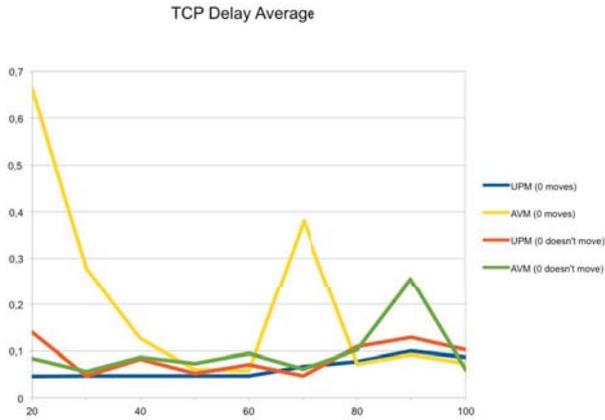


FIGURE 16
TCP delay average

Figure 16 shows that UPM suffers less packet delay. While the number of nodes increases in the network, the delay tends to stabilize for both static and dynamic viewpoints.

According to Figure 17, while more nodes exist in the network, the number of application packets loss decreases since the probability of effective information delivery increases. Logically, from a static viewpoint, there is greater efficiency in the delivery of packets for both models than a dynamic viewpoint. The closer to one, the better it is.

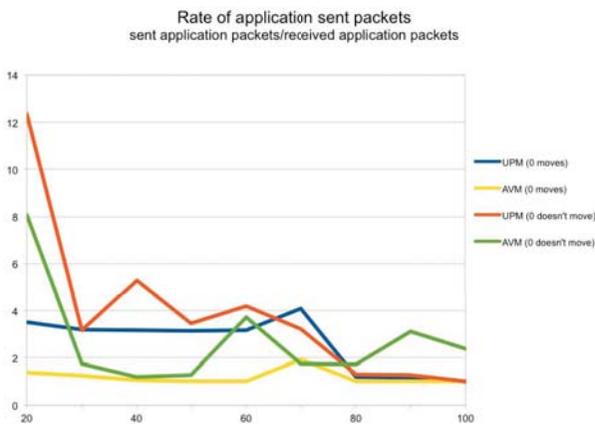


FIGURE 17
Application sent packets

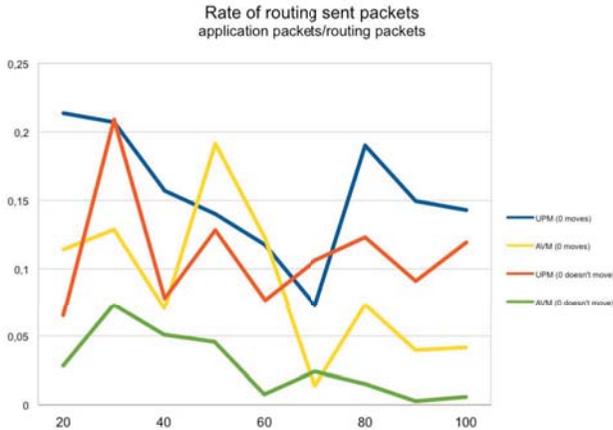


FIGURE 18
Rate of routing sent packets

The rate of sent routing packets, displayed in Figure 18, measures the amount of routing packets required for path generation and consequent application packet delivery. In spite of UPM is a model with shorter movements, it has a better relationship about the efficiency of the routing protocol.

As seen in Figure 19, both models AVM and UPM have similar loss packets for static and dynamic view, respectively. We also appreciate, when node 0

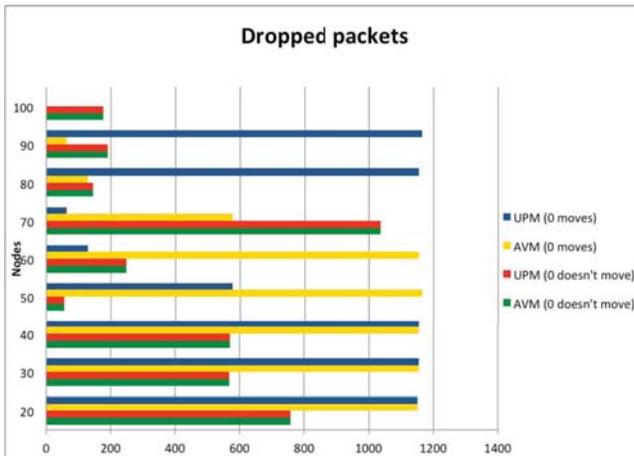


FIGURE 19
Dropped packets

(target point) operates from the dynamic viewpoint, there is a greater amount of loss packets than the static viewpoint.

7 CONCLUSIONS AND DISCUSSION

In this paper we present two models of human mobility that can be applied to the simulation of emergency and rescue scenarios. Both handle a large volume of obstacles and pathways. They are perfectly scalable, and depending on the model, they require more or less machine resources. The AVM constitutes the most efficient model, because it uses fewer resources, and it is faster than the UPM. It is therefore the most suitable for the generation of simulated emergency and rescue scenarios.

The UPM is useful in areas with obstacles with no clear geometric shape, such as emergency and rescue scenarios, where the intensity of the disaster may directly affect the shape of the obstacle.

The convenience of AVM shown during the simulation using AODV protocol confirms the effectiveness and efficiency advantages offered by this model against the UPM for low node density. Although as seen, UPM offers advantages enabling it to provide detailed work on the scenario mainly maintaining a good flow of communication with low delay rates and management of greater stability in the network.

Existing mobility models are not focused on the movements of people in disaster situations, where the movements are limited by the visual health of the victim and visibility in your environment. We compared the proposed models in this research because they reflect the conditions in which the victim is after a disaster.

There are some research issues not addressed in this paper. Especially it is interesting to compare our proposed models with others according to characteristics of mobility models: node velocity, node density, length of movements. Another research issue is to improve our models using the group human behavior.

REFERENCES

- [1] The Monarch Project. <http://www.monarch.cs.rice.edu>.
- [2] Nils Aschenbruck, Raphael Ernst, Elmar Gerhards-Padilla, and Matthias Schwamborn. (2010). Bonnmotion: a mobility scenario generation and analysis tool. In *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, page 51. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

- [3] Nils Aschenbruck, Elmar Gerhards-Padilla, and Peter Martini. (2008). A survey on mobility models for performance analysis in tactical mobile networks. *Journal of Telecommunications and Information Technology*, 2:54–61.
- [4] Nils Aschenbruck, Aarti Munjal, and Tracy Camp. (May 2010). Trace-based mobility modeling for multi-hop wireless networks. *Computer Communications*, 34(6):704–714.
- [5] Tracy Camp, Jeff Boleng, and Vanessa Davies. (2002). A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing. Special issue on Mobile Ad Hoc Networking. Research, Trends and Applications*, 2:483–502.
- [6] Yu Chenchen, Li Xiaohong, and Zhang Dafang. (oct. 2010). An obstacle avoidance mobility model. In *Intelligent Computing and Intelligent Systems (ICIS), 2010 IEEE International Conference on*, volume 3, pages 130–134.
- [7] Andrea Clementi, Angelo Monti, and Riccardo Silvestri. (August 2011). Modelling mobility: A discrete revolution. *Ad Hoc Networks*, 9(6):998–1014.
- [8] S. Cristaldi, a. Ferro, R. Giugno, G. Pigola, and a. Pulvirenti. (May 2011). Obstacles constrained group mobility models in event-driven wireless networks with movable base stations. *Ad Hoc Networks*, 9(3):400–417.
- [9] Bhavyesh Divecha, Ajith Abraham, Crina Grosan, and Sugata Sanyal. (2007). Impact of node mobility on manet routing protocols models. *Journal of Digital Information Management*, 5(1):19.
- [10] Yanying Gu, R. Venkatesha Prasad, and Ignas Niemegeers. (December 2009). Mobility Modeling for Personal Networks. *Wireless Personal Communications*, 58(2):169–196.
- [11] Dmytro Karamshuk, Chiara Boldrini, Marco Conti, and Andrea Passarella. (2011). Human mobility models for opportunistic networks. *Communications Magazine, IEEE*, 49(12):157–165.
- [12] K. Konishi, K. Maeda, K. Sato, A. Yamasaki, H. Yamaguchi, K. Yasumoto, and T. Higashino. (sept. 2005). Mobireal simulator-evaluating manet applications in real environments. In *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2005. 13th IEEE International Symposium on*, pages 499–502.
- [13] Franck Legendre, Vincent Borrel, Marcelo Dias de Amorim, and Serge Fdida. (2006). Modeling mobility with behavioral rules: The case of incident and emergency situations. In *AINTEC*, pages 186–205.
- [14] Guolong Lin, Guevara Noubir, and Rajmohan Rajaraman. (2004). Mobility models for ad hoc network simulation. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1. IEEE.
- [15] Mirco Musolesi and Cecilia Mascolo. (2009). Mobility models for systems evaluation. *Middleware for Network Eccentric and Mobile Applications*, pages 1–28.
- [16] C. Papageorgiou, K. Birkos, T. Dagiuklas, and S. Kotsopoulos. (sept. 2009). An obstacle-aware human mobility model for ad hoc networks. In *Modeling, Analysis Simulation of Computer and Telecommunication Systems, 2009. MASCOTS '09. IEEE International Symposium on*, pages 1–9.
- [17] Christos Papageorgiou, Konstantinos Birkos, Tasos Dagiuklas, and Stavros Kotsopoulos. (May 2012). Modeling human mobility in obstacle-constrained ad hoc networks. *Ad Hoc Networks*, 10(3):421–434.
- [18] Charles E Perkins and Elizabeth M Royer. (1999). Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA99. Second IEEE Workshop on*, pages 90–100. IEEE.
- [19] The Scenario Generator. <http://isis.poly.edu/~qiming/scengen/index.html>.

- [20] Razvan Stanica, Emmanuel Chaput, and André-Luc Beylot. (June 2011). Simulation of vehicular ad-hoc networks: Challenges, review of tools and recommendations. *Computer Networks*, 55(14):3179–3188.
- [21] Chen Zhao and Mihail L. Sichitiu. (March 2011). Contact time in random walk and random waypoint: Dichotomy in tail distribution. *Ad Hoc Networks*, 9(2):152–163.
- [22] Qunwei Zheng, Xiaoyan Hong, and Sibabrata Ray. (2004). Recent advances in mobility modeling for mobile ad hoc network research. In *Proceedings of the 42nd annual Southeast regional conference*, ACM-SE 42, pages 70–75, New York, NY, USA. ACM.