

Configurable Node Density Generation with Application to Hotspots Modelling*

Daniel Calabuig^{a†}, Jose F. Monserrat^a, and Narcís Cardona^a

^aUniversidad Politecnica de Valencia,
Institute of Telecommunications and Multimedia Applications,
Camino de Vera s/n, 46022 Valencia, Spain

Mobility models are very relevant mainly when studying the performance of wireless systems by means of computer simulations. The main problem arises when deciding the best mobility model for a particular application. In some cases, it is very important to emulate hotspots or, in general, zones with different user (or node) densities. Current models do not allow complete control over hotspots, or in other words, they do not allow any general node density to be defined in the simulation area. Usually, when hotspots are modelled, closed zones are created with different numbers of users in each area, thus ensuring a fixed node density in each area. However, this approach results in an unfair comparison among users since they cannot move across zones.

This paper proposes a new mechanism to solve these drawbacks. Using this mechanism, any general node density can be emulated allowing nodes to move around the entire simulation area. Any mobility model can be applied together with this density control mechanism, provided that the mobility model ensures a uniform node distribution.

1. INTRODUCTION

Mobility is one of the key characteristics of wireless systems. It produces most of the effects that make quality fluctuate, like signal fading or handovers. Thus, it is very important to correctly emulate user mobility and to know the characteristics of the mobility model used to perform system simulations.

Mobility is not only important in wireless systems but also in other areas like transport, study of migratory birds or even hurricanes [1]. This fact has motivated the creation of a huge number of different mobility models for different ap-

plications. In all models, the mobile entities are usually referred to as mobile nodes or just nodes.

1.1. Mobility models

The mobility models used in simulations of wireless networks can be roughly classified into independent or group-based models. Independent models characterize the movement of each node independently of the rest of nodes. On the other hand, group models generate some dependence between the movement of certain nodes.

Some independent mobility models are:

- Random walk: Each node moves from its current location to a new location by randomly choosing an arbitrary direction and speed from a given range. Such a move is performed either for a constant time or travelled distance. Then, a new speed and direction are chosen. At the boundaries, nodes bounce off like billiard balls on a pool table. A more detailed description of this model can be found in [2]. This model is the simplest and many other variants have been proposed to better emulate node mobility.

*NOTICE: this is the authors version of a work that was accepted for publication in Mathematical and Computer Modelling. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Mathematical and Computer Modelling, vol. 53, no. 11-12, pp. 2229-2237, 2011, DOI:10.1016/j.mcm.2010.08.028.

†This work has been funded by the Spanish Ministry of Science and Innovation under the project TEC2008-06817-C02-01/TEC.

- Random waypoint [3]: In this model, nodes wait for some random time and then choose a new destination moving towards it with random speed. When the destination has been reached, the process starts again.
- Boundless simulation area mobility model [4]: The model exchanges the planar rectangular simulation area with a boundless torus. In this way, nodes that reach one side of the simulation area continue travelling and reappear on the opposite side.
- Smooth random mobility model [5]: This model is basically an extension of the simple random walk model. Here, two independent stochastic processes are used to trigger direction and speed changes. The new speeds, for example, are chosen from a weighted distribution of preferred speeds. At the trigger, the speed —or direction— changes are determined by a Poisson process.
- Random Gauss-Markov mobility model [6]: This model enhances the smooth random mobility model. A node's next location is predicted —or generated— from its past location and velocity. Depending on the parametrization, this generates an entire spectrum of models from random walk to fluid flow.

Group mobility models are usually an extension of the above models. An exception to this is the fluid flow mobility model. This model represents the behaviour of all nodes at the same time using flow equations. This approach can only be used if the movement of individual nodes is not relevant. The behaviour of the generated traffic is similar to a fluid flowing through a pipe. As a result, the fluid flow mobility model represents traffic on highways very well [1]. Some other group models are [1]:

- Exponentially correlated random mobility model: A motion function generates group behaviour.
- Column mobility model: The set of mobile nodes form a line and move forward in a particular direction.

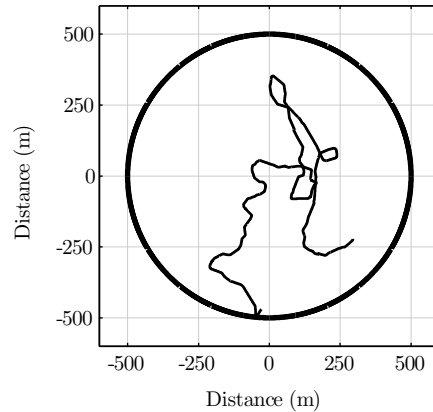


Figure 1. Typical path followed by a node. The thick line shows the border of the simulation area.

- Nomadic community mobility model: A group mobility model where a set of nodes moves together from one location to another.
- Pursuit mobility model: For each group all members follow a target node moving around the simulation area.
- Reference point group mobility model: The group movement is based on the path travelled by a logical centre. The logical centre moves according to an independent mobility model.

The mobility model that is finally selected has great impact on the results obtained from simulations. This fact can make the same scenario have very different performance depending on the selected mobility model. For this reason, standardization bodies propose their own models for testing the performance of their technologies. Then, different institutions can make reasonably impartial studies that can be compared. These models are usually a mixture of the previous models and attempt to take advantage of the main benefit of each model.

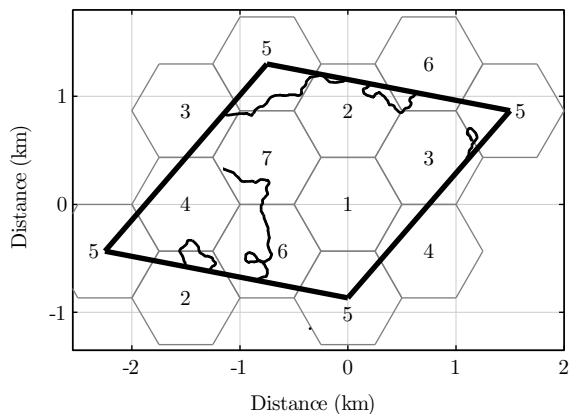


Figure 2. Typical path followed by a node in a scenario with seven cells. This figure shows how cells are replicated following the idea of the boundless simulation area mobility model of [4].

For example, the European Telecommunications Standards Institute (ETSI) proposes a model in [7] that was also used in the MORANS activity of COST 273 [8]. The model is similar to a random walk. The movement of nodes is generated independently. In the initial state, nodes are homogeneously distributed around the simulation area and select a random direction between 0 and 2π and a random speed between minimum and maximum speeds. After this initial state, the nodes start moving in the selected direction. In every simulation iteration, all nodes have a certain probability of changing their direction. This probability is calculated so that the mean time spent in the same direction is 5 s. When a node changes its direction, a new one is selected from $-\pi/4$ to $\pi/4$ with respect to the current direction. Figure 1 shows an example of the path followed by a node in this model within a cell simulation area of radius 500 m. In this example, nodes bounce back when they reach the border. Nevertheless, it is also possible to use this model in a boundless simulation area like that shown in Figure 2.

1.2. Hotspot modelling

The question of how to model hotspots has arisen previously. Among all the aforementioned mobility models, only group-based models can somehow emulate hotspots. Nevertheless, most of them do not provide a mechanism for controlling the size of the hotspot or even the node density. Another important drawback is that if a node is attached to some group, it remains attached to it until the end of the simulation. This means that nodes cannot exit the hotspot or return to it. It is important to use a model that allows nodes to behave in this way, otherwise if the hotspot area is a zone of good quality, nodes within it would always perceive good quality and would never go to other areas with worse conditions. Moreover, nodes moving throughout the simulation area would always perceive a bad quality. This feature will produce a large difference in the quality perceived by different users, which may have severe effects on the results and hence encourage misleading conclusions.

Another possibility is to create a small simulation area for the hotspot of some predefined size and to generate as many nodes as necessary to have the desired node density. This procedure is widely used —see [8] as an example— but has the same problem explained before, i.e. nodes inside the hotspot will never exit it. Other mobility models, like [9] and [10], generate the movement of nodes independently and emulate hotspots. In these models, nodes move away from the hotspot and come back independently of the other nodes. Nevertheless, hotspot size and node density cannot be controlled. Probably, the most complete model is the one proposed by Hyytiä *et al.* [11]. This model is based on a random waypoint. They proposed increasing the node density inside the hotspot by decreasing the node speed when they enter the hotspot or by increasing the waiting time of waypoints in the hotspot. This mechanism allows nodes to enter and exit the hotspot. Moreover, it is possible to compute the speed reduction or waiting time increase needed to give a certain node density in the hotspot area. The main problem with this model is that it inherits all the drawbacks of the random waypoint model, most of all that nodes tend to concentrate in the

centre of the simulation area [10]. This fact also affects the results and hence can also encourage misleading conclusions [12].

Another approach used for Mobile Ad hoc Networks (MANETs) is based on identifying the hotspots in a certain area and then assuming that nodes only move from one hotspot to another. Kim *et al.* [13] defined a procedure for obtaining the hotspot locations from real measurements. Moreover, they also computed the hotspot node density, although this was uniform. This procedure could be extended to calculate the node density over the whole area in order to emulate a real scenario. A modification of this approach was proposed by Lee *et al.* [14]. Basically, they changed the way that nodes move from one hotspot to another based on four basic rules. Although both approaches control what is happening in each hotspot, it is not possible to maintain a controlled node density outside the hotspots when the nodes move to their new destination.

1.3. Motivation for this paper

For MANETs, it is very important to precisely emulate the movement of each node, since their performance depends not only on node mobility but also on the interaction of nodes. Nevertheless, to evaluate the performance of cellular systems, this complex modelling of user movement is not required. Rather, when assessing cellular systems it is quite important to be able to control the shape and load of the traffic. Stress testing is a clear example of the requirement for accurate modelling of node density. A stress test is often used to determine the maximum load a system can handle before an unacceptable degradation in quality of service occurs. In these studies, instead of using complex mobility models for MANETs, it would be better to use other more simple models like the random walk, random waypoint or even the model proposed by ETSI, as long as the hotspots could be perfectly controlled. For example, in heterogeneous systems, where multiple technologies are located in the same area, the proper definition of hotspots is of paramount importance to correctly plan for hotspot-specific technologies, like IEEE 802.11.

To sum up, so far the hotspot models proposed in the literature do not simultaneously satisfy the two requirements highlighted here, namely that nodes should be able to enter and leave the hotspot and that node density should be completely configurable. This paper proposes a new mechanism to control the node density over the entire simulation area. Thus, hotspots can be emulated by defining areas with higher node density. This mechanism is used over any other mobility model and allows nodes to move around the simulation area—and hence to move outside hotspots—while maintaining the desired node density.

The rest of the paper is arranged as follows. Section 2 presents the mechanism with a simple example. Section 3 generalizes the concepts of Section 2 to any general scenario. The performance of this mechanism is studied in Section 4 and conclusions are drawn in Section 5.

2. MECHANISM DESCRIPTION

This new approach is based on flow theory, although it is completely different from the fluid flow mobility model. In short, the idea is to make normal use of mobility models but assuming that nodes may suddenly bounce off *imaginary* bounds. This forced bouncing prevents nodes from moving outside certain zones and, hence, can create different node densities. The underlying mobility models combined with this idea must create a homogeneous node density when they operate normally. This is the case for the random walk—with or without a boundless simulation area—but not for the random waypoint. Thus, the mobility models proposed by ETSI in [7] and by COST 273 in [8] are the preferred options due to their wide usage. The rest of this paper describes in detail this idea and how to compute the probability of bouncing and, if so, with what imaginary boundary.

In order to explain the model, let us start with a simpler case before moving on to the general case. Therefore, let us focus on the simulation area presented in Figure 3. This area is divided into two square zones with one side in common. Zone A has more nodes than zone B. Nodes are

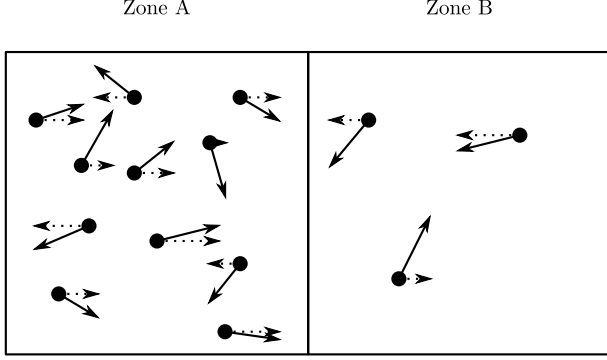


Figure 3. Simulation area with two zones of different node densities.

moving with random direction and speed. The number of nodes that will move from zone A to B in a certain time interval Δt depends directly on the velocity component that is perpendicular to the border between the zones. Figure 3 shows this component with dotted lines. Let us assume that the average speed in zone A is v_A , then the average perpendicular component is:

$$v_A^\perp = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} v_A \cos(\gamma) d\gamma = \frac{2v_A}{\pi}. \quad (1)$$

Thus, on average, the number of nodes that will move from zone A to B in time interval Δt is at most half the number of nodes at a distance $2v_A\Delta t/\pi$ from the border. The other half are moving in the opposite direction. Note that nodes follow random directions. This quantity is:

$$\Delta n_A = \frac{v_A \Delta t \delta_A L}{\pi}, \quad (2)$$

where δ_A is the node density in zone A and L is the length of one side of the square zone. Analogously, the number of nodes moving from zone B to zone A is:

$$\Delta n_B = \frac{v_B \Delta t \delta_B L}{\pi}. \quad (3)$$

If $\Delta n_A = \Delta n_B$, the scenario of Figure 3 is in equilibrium. Nodes may change their zone but

the node densities will not vary, i.e. the number of nodes that move from A to B is exactly the same as the number of nodes coming back from B to A. The only parameter available to force the equilibrium is velocity. Thus, nodes should have different speeds in each zone, depending on the desired densities. If velocity and density must be independent, then, a new degree of freedom must be introduced in these equations. Let $\rho_{A \rightarrow B}$ and $\rho_{B \rightarrow A}$ be the probabilities that any node of zone A and B respectively does not bounce off the border between zones. Then, the previous expressions must be rewritten as:

$$\Delta n_A = \frac{v_A \Delta t \delta_A L \rho_{A \rightarrow B}}{\pi}, \quad (4)$$

$$\Delta n_B = \frac{v_B \Delta t \delta_B L \rho_{B \rightarrow A}}{\pi}. \quad (5)$$

Hence, the system is in equilibrium if:

$$\frac{\rho_{A \rightarrow B}}{\rho_{B \rightarrow A}} = \frac{v_B \delta_B}{v_A \delta_A}. \quad (6)$$

If the previous expression is greater than 1, $\rho_{A \rightarrow B}$ can be set to 1 and $\rho_{B \rightarrow A}$ can be computed from (6). On the other hand, if the expression is lower than 1, then $\rho_{B \rightarrow A}$ is set to 1 and $\rho_{A \rightarrow B}$ is computed from (6). This simple example is useful for introducing the idea behind the model proposed in this paper. An heterogeneous node distribution is achieved by forcing some nodes to bounce, thus maintaining a high density of nodes in certain zones. It is worth noting that this is not a mobility model by itself. This approach just sets some probabilities for bouncing at points where the product of velocity and node density changes. Now, imagine that on the right of zone B there is an additional zone C. The probability that a node in zone A reaches zone C is:

$$\rho_{A \rightarrow C} = \rho_{A \rightarrow B} \rho_{B \rightarrow C}. \quad (7)$$

Thus, from (6):

$$\frac{\rho_{A \rightarrow C}}{\rho_{C \rightarrow A}} = \frac{\rho_{A \rightarrow B}}{\rho_{B \rightarrow A}} \cdot \frac{\rho_{B \rightarrow C}}{\rho_{C \rightarrow B}} = \frac{v_B \delta_B}{v_A \delta_A} \cdot \frac{v_C \delta_C}{v_B \delta_B} = \frac{v_C \delta_C}{v_A \delta_A}. \quad (8)$$

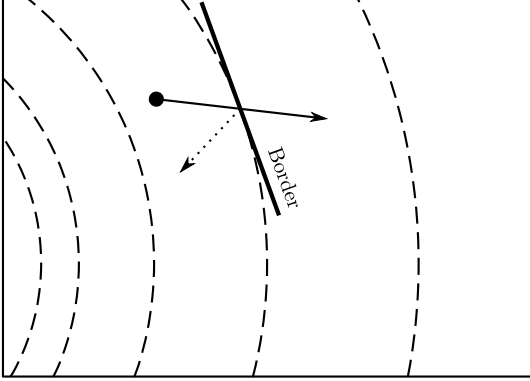


Figure 4. Example of a node bouncing. The striped lines are contour lines of the velocity-density product. The dotted arrow shows the destination after bouncing.

Consequently, the ratio between the probabilities in both directions just depends on the densities and velocities in the zones at the extreme ends of the path followed by the node. Note that this conclusion is only true if no zone along the path has a zero mean velocity or node density. This property can be used to generate a movement of nodes in a simulation area with any density and velocity distributions.

3. GENERAL CASE

Let $\mathcal{A} \in \mathbb{R}^2$ be the simulation area and $v(r)$ and $\delta(r)$ be, respectively, the mean velocity and node density at point $r = (x, y)' \in \mathcal{A}$, where x' is the transpose of x . Then, if a node is at point $r_1 = (x_1, y_1)'$ and, according to the mobility model, the node should move to $r_2 = (x_2, y_2)'$ in the next simulation iteration, the probability of this happening is:

$$\rho_{r_1 \rightarrow r_2} = \min \left\{ \frac{v(r_2)\delta(r_2)}{v(r_1)\delta(r_1)}, 1 \right\}. \quad (9)$$

Note that (9) is equivalent to the conclusions drawn from (8). The main difference is that the zones are now infinitesimally small having, thus, one zone at each point of the simulation area. For

this reason, the mean velocity and density are now represented as functions of node location.

If the node bounces off, then this may happen at any point between r_1 and r_2 . For the sake of simplicity, this paper assumes that bouncing always occurs at the mid-point, i.e. $r_m = (r_1 + r_2)/2$. The border that nodes bounce off is tangential to contour lines of product velocity-density or, in other words, perpendicular to the gradient of this product. Figure 4 depicts how a node bounces in this model.

The gradient can be computed analytically, although it is possible to approximate it using intervals, which simplifies the calculus. Let us define the points r_a and r_b as $r_a = (x_2, y_1)'$ and $r_b = (x_1, y_2)'$. Then, the gradient at r_m can be approximated by the vector:

$$\nabla (v(r_m)\delta(r_m)) \approx \begin{pmatrix} g_x(r_m) \\ g_y(r_m) \end{pmatrix}, \quad (10)$$

where:

$$g_x(r_m) = \frac{v(r_a)\delta(r_a) - v(r_1)\delta(r_1)}{x_2 - x_1}, \quad (11)$$

$$g_y(r_m) = \frac{v(r_b)\delta(r_b) - v(r_1)\delta(r_1)}{y_2 - y_1}. \quad (12)$$

The fastest way of computing the points that nodes reach after bouncing is to apply a rotation. Using the rotation matrix:

$$R = \frac{1}{\sqrt{g_x^2(r_m) + g_y^2(r_m)}} \begin{pmatrix} g_x(r_m) & g_y(r_m) \\ -g_y(r_m) & g_x(r_m) \end{pmatrix}, \quad (13)$$

the gradient will point towards the direction of the abscissa axis. Figure 5 shows an example of such rotations. Afterwards, the node movement after bouncing is equivalent to a movement parallel to the ordinate axis. Then, the rotation can be undone by applying the inverse of R . These three transformations can be concatenated mathematically as the product of the following matrices:

$$T = R^{-1} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} R, \quad (14)$$

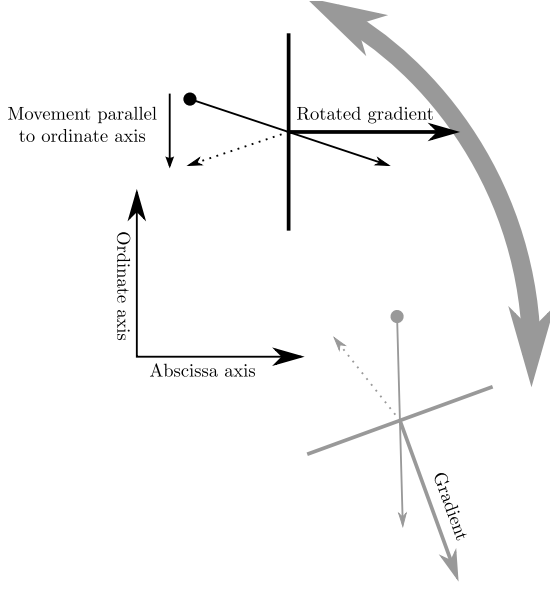


Figure 5. Example of methodology followed to obtain the destination of nodes after bouncing.

$$T = \begin{pmatrix} \frac{g_y^2(r_m)}{g_x^2(r_m) + g_y^2(r_m)} & -\frac{g_x(r_m)g_y(r_m)}{g_x^2(r_m) + g_y^2(r_m)} \\ -\frac{g_x(r_m)g_y(r_m)}{g_x^2(r_m) + g_y^2(r_m)} & \frac{g_x^2(r_m)}{g_x^2(r_m) + g_y^2(r_m)} \end{pmatrix}. \quad (15)$$

Thus, the point that the nodes reach after bouncing is:

$$r_3 = r_1 + T(r_2 - r_1). \quad (16)$$

Finally, note that the direction of movement of nodes changes after bouncing. The new direction is:

$$r_3 - r_m = \left(T - \frac{I}{2}\right)(r_2 - r_1), \quad (17)$$

where I is the identity matrix.

4. MODEL PERFORMANCE

This section demonstrates the performance of the model in a very simple scenario composed of

a cell of 500 m radius with a hotspot of 50 m radius in the centre. Thus, the cell has revolution symmetry around the cell centre. Its simplicity makes it easier to present the results as they only depend on radius. This scenario was implemented in a simulator specifically developed for this paper in Matlab. The hotspot has a smooth border between 40 and 50 m where the node density decreases from a maximum in the hotspot to the value outside the hotspot. Mathematically, the objective node density is:

$$\delta(d) = \begin{cases} U_1, & d \leq d_1, \\ \frac{U_1 - U_2}{2} \cos\left(\frac{\pi(d - d_1)}{d_2 - d_1}\right) + \frac{U_1 + U_2}{2}, & d_1 < d < d_2, \\ U_2, & d \geq d_2, \end{cases} \quad (18)$$

where $d_1 = 40$ m, $d_2 = 50$ m and d is the distance to the cell centre, i.e. $d = \|r\| = \sqrt{x^2 + y^2}$. The quantities U_1 and U_2 are, respectively, the node densities inside the hotspot and in the rest of the cell. For these simulations, the computations were done so that 50% of the nodes were in the hotspot, i.e. $d \leq d_2$, and the rest were outside the hotspot. These quantities were $U_1 = 7.67 \cdot 10^{-5}U$ and $U_2 = 6.43 \cdot 10^{-7}U$, where U is the number of nodes in the simulation area. For the sake of simplicity, the mean node speed was the same over the entire simulation area, 50 km/h. The mobility model underlying this model is the one proposed in COST 273 [8].

Figures 6, 7 and 8 show the performance of this model. The results were obtained for 10^5 nodes moving during 100 simulation seconds (1000 simulation iterations). In the initial state, half of the nodes were placed in the hotspot—from 0 to 50 m—and the other half outside the hotspot—from 50 to 500 m. Figure 6 shows how this initial distribution varies as the simulation progresses. Thanks to the node density control mechanism presented here, the hotspot was perfectly modelled during the entire simulation. On the other hand, if nodes move freely along a random walk, the node density tends to be homogeneous, which makes the initial distribution vary. Fig-

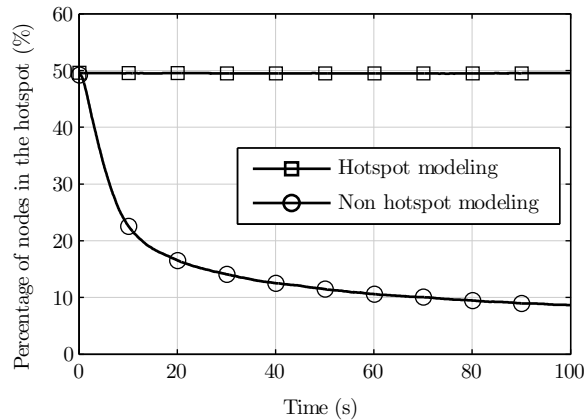


Figure 6. Percentage of users in hotspot as simulation progresses.

Figure 7 shows the node density averaged over 100 seconds of the simulation. Moreover, this figure shows the objective node density too. Obviously, in the initial state, nodes are placed according to the objective. It is worth noting that with the proposed mechanism the objective function is perfectly matched. Figure 8 shows in detail the variation of node density around the hotspot border. As can be seen, this mechanism is capable of modelling smooth borders between zones. Obviously, any other node density function could be implemented as objective. For instance, Figure 9 shows a node density in the form of a cross. Moreover, it would also be possible to define a mean node speed that depends on node location, as occurs in simulation areas with streets, highways and malls. In this latter case, a node could exit the mall, enter the streets of a city and take the highway, increasing its speed during the process.

5. CONCLUSION

This paper has presented a mechanism capable of creating any general node density in a simulation area. The mechanism is based on sudden bounces that depend on the desired node density

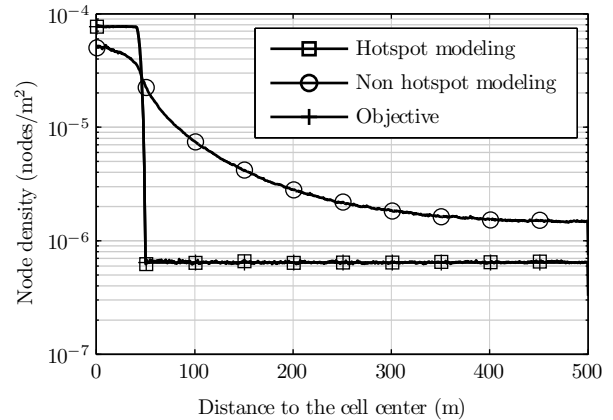


Figure 7. Node density normalized by number of users U as a function of distance to cell centre.

and average node speed. Moreover, this paper has presented an easy way of obtaining the destination and movement direction of nodes after bouncing.

Results have shown the accuracy of the mechanism in perfectly matching the desired node density. Even smooth borders are precisely emulated, allowing the definition of complex densities in the simulation area.

Moreover, the mechanism allows an impartial study of any wireless system, since all nodes will have the same movement pattern and will not be stuck inside specific zones.

REFERENCES

1. C. Schindelhauer, Mobility in wireless networks, in: International Conference on Current Trends in Theory and Practice of Computer Science, Merin, Czech Republic, 2006.
2. T. Camp, J. Boleng, V. Davies, A survey of mobility models for ad hoc network research, *Wireless Communications and Mobile Computing* 2 (5) (2002) 483–502.
3. J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, J. Jetcheva, A performance comparison of multi-hop wireless ad hoc network routing

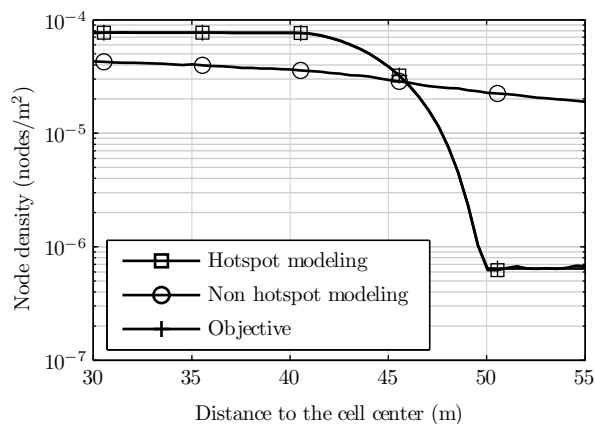


Figure 8. Node density normalized by number of users U as a function of distance to cell centre. Detail showing smooth border.

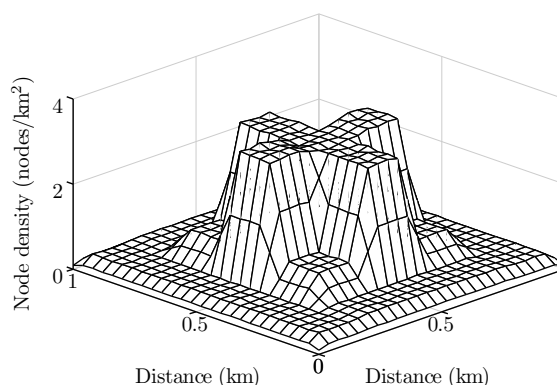


Figure 9. Node density normalized by number of users U in the form of a cross in a square simulation area.

- protocols, in: ACM/IEEE International Conference on Mobile Computing and Networking, Dallas, USA, 1998.
4. Z. J. Haas, A new routing protocol for reconfigurable wireless networks, in: IEEE International Conference on Universal Personal Communications, San Diego, USA, 1997.
 5. C. Bettstetter, Smooth is better than sharp: a random mobility model for simulation of wireless networks, in: ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM), Rome, Italy, 2001.
 6. B. Liang, Z. J. Haas, Predictive distance-based mobility management for pcs networks, in: IEEE Annual Joint Conference Computer and Communications Societies (INFOCOM), New York, USA, 1999.
 7. ETSI, Universal mobile telecommunications system (UMTS): Selection procedures for the choice of radio transmission technologies of the UMTS, TR 101.112, ETSI (1998).
 8. J. Monserrat, L. M. González, R. Fraile, N. Cardona, Morans layer 2: Traffic information synthetic scenario, TD (04)071, COST 273 (2004).
 9. V. Borrel, M. D. de Amorim, S. Fdida, A preferential attachment gathering mobility model, IEEE Communications Letters (2005) 900–902.
 10. S. Lim, C. Yu, C. R. Das, Clustered mobility model for scale-free wireless networks, in: IEEE Conference on Local Computer Networks (LCN), Tampa, USA, 2006.
 11. E. Hyttiä, P. Lassila, J. Virtamo, A Markovian waypoint mobility model with application to hotspot modeling, in: IEEE International Conference on Communications (ICC), Istanbul, Turkey, 2006.
 12. C. Bettstetter, C. Wagner, The spatial node distribution of the random waypoint mobility model, in: German Workshop on Mobile Ad Hoc Networks (WMAN), Ulm, Germany, 2002.
 13. M. Kim, D. Kotz, S. Kim, Extracting a mobility model from real user traces, in: IEEE Conference on Computer Communications (INFOCOM), Barcelona, Spain, 2006.
 14. K. Lee, S. Hong, S. J. Kim, I. Rhee, S. Chong, Slaw: A mobility model for human walks, in: IEEE Conference on Computer Communica-

tions (INFOCOM), Rio de Janeiro, Brazil, 2009.