

Modelling Mobile User Behaviour for Future Wireless Networks Using OPNET Modeller

D. R. Bageet¹, P. Dugenie³, A. Munro¹, D. Kaleshi¹, J. Irvine²

¹ Networks & Protocol Group, CCR,
Department of E & E Eng., University of
Bristol, UK

² Department of E & E Eng., University of
Strathclyde, Glasgow, UK

³ NexWave Solutions SA, 34967
MONTPELLIER Cedex 2, FRANCE

Email: D.R.Bageet@bristol.ac.uk, Telephone +44 117 915 1279, Fax: +44 (0) 117 954 5206

Abstract

This paper presents a new mathematical framework of a Scalable Mobility Model (SMM) to characterise user movement behaviour, which can be applied to a wide number of network scenarios ranging from Local Area Networks (LAN) to Wide Area Networks (WAN). The core technique to describe the spatial and temporal behaviour of mobile users is the introduction of a novel concept referred to as *Pole of Gravity*. We use OPNET Modeller and its object-oriented modelling approach to construct this stochastic framework and show how the mobility traces generated from OPNET simulator has been used to estimate the teletraffic demand for City Area of Bristol, UK.

Keywords: *Mobility model, teletraffic model, and multimedia*

1. Introduction

Mobile services have been experiencing accelerated growth. The increase of both mobile related traffic and mobility related signalling load, together with the limited number of radio resources and increasing competition in service provision, has meant that mobile operators are continuously designing and refining their planning tools to provide optimal and economic network configurations. To allow efficient resource allocation, good models of the underlying movement of mobile users are necessary. As mobile systems develop, the requirements on mobility models become more demanding. As cell sizes decrease to allow greater capacity, local accuracy becomes more important, but the requirement to model wide scale behaviour remains.

A considerable amount of research efforts has been used to characterise user density and mobility, and calling behaviour and their performance impacts on wireless networks. In this context, different mobility models have been used to tackle specific issues, fluid and gravity models [1] describe aggregate population movement while the Markovian and Brownian models characterise mobile user displacement on a per mobile basis.

The need for different mobility models arise due to the fact that no single model provides sufficient accuracy over all topologies for both Mobility Management and Radio Resource Management scenarios. In order to refine those models, variations have been made by exploiting transport traffic data so that they can be applied to the specific topological area under investigation. While these enhanced models [2, 3] are rich and expressive within the context of transportation modelling, they are unnecessarily over-detailed at a topographical level as a basis for creating event patterns that allow teletraffic analysis of a mobile network for the purposes outlined above. Furthermore, users with mobile terminals are less constrained by the transport network, which, in fact, is absent in many short-range built-environment scenarios. What is required is a scaleable algorithmic mobility model that can be applied to a large number of possible multi-layer network configurations.

The rest of this paper is organised as follows. Section 2 gives the description of the Pole of Gravity concept. Section 3 provides a description of the OPNET Models used to model the Pole of Gravity. In Section 4 we show the simulation results by mapping mobile user behaviour into terminal traffic demand. Conclusions are provided in Section 5.

2. Pole of Gravity

Conventional mobile network planning tools such as PLANET [4] are driven by area coverage considerations rather than customer demand which has been the planning approach used in first and second generation of wireless networks. However, the success of future wireless systems will largely depend on the Quality of Service (QoS) that they will provide. As such the main aim of future wireless planning is no longer area coverage, but to supply as many mobile users with their traffic demand as possible with a minimum of infrastructure. The key concept to achieve this is the introduction of *Pole of Gravity* (PG).

A PG is a discrete stochastic node describing the spatial and temporal movement in a predefined

topological area under consideration. This stochastic system extracts the features of the head-of-the-line proportional processor sharing (HLPPS) fluid models [5] in order to populate mobile users in different classes of mobility based on their mobility behaviour. As such it provides a mathematical framework for predicting and tracking mobile user behaviour [6]. The main features of a PG are:

- **Classes of Mobility** – Mobile users having the same mobility behaviour are grouped into a specific class of mobility (CM). The corresponding CM also reflects mobile users’ calling behaviour and use of services. For our model, three-CM have been considered: *Business, Residential* and *Others*.
- **Area Zones** – The geographical area under consideration is divided into regions having specific mobility related characteristics. Four environments have been considered: *Working, Domestic, Shopping* and *Streets*.
- **Attractivity Points (APs)** – These represent locations that attract users with a specific CM and at which mobiles spend a certain amount of time. The attractivity of an AP is a function of time and varies for each CM based on the environment in which the mobile user is located. Each CM is assigned one of the following attraction weights, $w \in [0, 1]$: for example, dominant (0.9), normal (0.5), or null (0.1) based on zone’s attractivity during a specific time of the day.
- **Time Periods** – In the context of a wireless network, the *rush hour* (RH) and the *busy hour* (BH) can be identified. Both of these times periods are taken into consideration allowing the analysis of mobility-related signalling and proper planning and dimensioning issues.

3. OPNET Models and Network Configuration

In this section, we describe the network configuration and the OPNET simulation model. Figure 1 shows the OPNET network model illustrating the PG concept, represented by a fixed communication node for the City Area of Bristol, UK having an extension of 8 km \times 8 km.

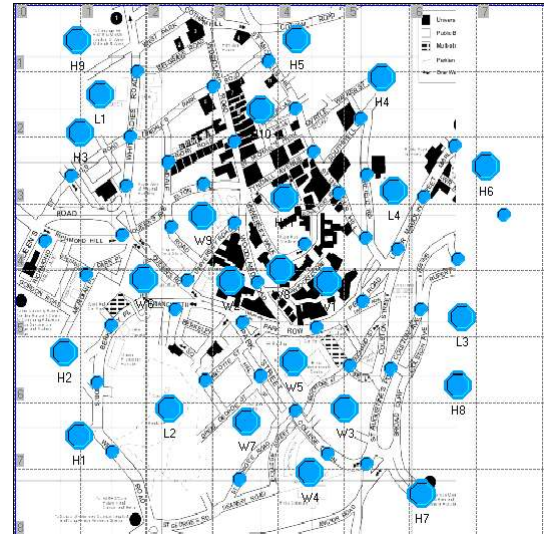


Figure 1: OPNET Network Model - Distribution of PGs in City Area of Bristol

PGs are distributed over the geographical region and are interconnected through the logical connectivities defined through the *angular connectivity*, R , which represents the most probable path that MUs will take during simulation. The *Physical Sub-Model* of the Mobility Model defines the distribution of the PGs together with the geographical region under consideration. Since the model is based on queuing theory, it implicitly introduces the concept of stability, which is characterised by the *Gravity Sub-Model*. This Sub-Model constantly calculates the *attractivity* of PGs in the system and determines the time spent by each mobile in a PG. This time is Markovian at times and this process is an enhancement of the HLPPS algorithm. The *Fluid Sub-Model* fixes the law of circulation of each mobile unit between the PGs taking into consideration the *attractivity* defined by the gravity law and *distance of separation* determined by the physical sub-model. All these three sub-models conceptualise a stochastic system which has been modelled in OPNET. The resulting platform simulates user movement behaviour dictated by the three sub-models which has been encapsulated in a single network node detailed in the next section.

3.1 The Platform Node Model

Each node within the network is uniquely identified by its name and object ID and is composed of two queues and four processors objects as shown in Figure 2. The main functions of each queue and processor is described below:

Buffer Queue Module: This module is the main module of the node model. It initialises and configures the model based on the attributes which are defined at both the node and simulation level. It generates the number of packets corresponding to

the number of MUs for each mobility class which is considered to be a queue of infinite capacity. Each generated packet based on its mobility class is inserted into specific queues and schedules the time of departure through the self-interrupt for each class based on the HLPPS algorithm.

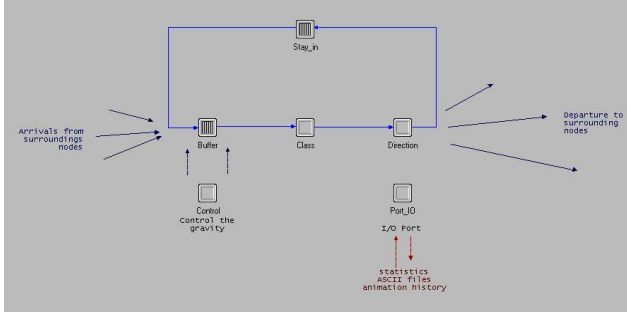


Figure 2: Node Model for PG

Class Processor Module: The Class module receives packet from the Buffer module and assigns the mobile a mobility class which may or may not change from its previous one based on the attractivity function controlled by the Gravity Law and forwards the packet to the Direction module.

Direction Processor Module: This module receives the packet from the Class module and based on the attractivity and connectivity of the surrounding nodes, the packet is forwarded accordingly to a particular node.

Stay in Queue Module: Ensures that packets is delivered to the desired node and are sent to the immediate process module (Buffer).

Control Processor Module: This module controls the reactivity of the model when a change of attractivity defined through the 'Attraction Function' occurs at a particular time of simulation.

Port IO Processor Module: The processor creates a file to store the mobility traces generated by all the mobiles at the end of the simulation time period.

The attributes defined at the node and simulation levels are defined below in Table 1:

Attribute Name at Node Level	Data Type
Angular connectivity	Integer
Mean Sojourn Time in a PG	Double
Spatial Distribution of MUs over PG	Double
Density and Initial Attraction of MUs in a PG	Compound
Attractivity Function	Compound
Attribute Name at Simulation Level	Data Type
Number of MUs	Integer
Simulation Time	Double
Output File	String

Table 1: Attributes used to configure a simulation at Node and Simulation Levels

3.2 Process Models of the Processors

Each processor is associated to a finite state machine (FSM); each state in the FSM corresponds to a special function. In this section, we provide a description of the working of the different process models for the Buffer, Class Direction, Control and Port_IO modules.

3.2.1 Process Model of Buffer Module

The FSM of the buffer_queue process model is shown in Figure 3. The principle of buffer_queue consists of two steps: *Initialisation*, where the process is configured before it enters the Idle state and the second step which describes the transitions from an idle state to three forced states: T_Actif, Arrival and Control and back to the Idle state. The transitions to the respective states are triggered through the interrupts TIMER (*self interrupt*), ARRIVAL (*stream interrupt*) and CONTROL (*remote interrupt*). These two steps are detailed below.

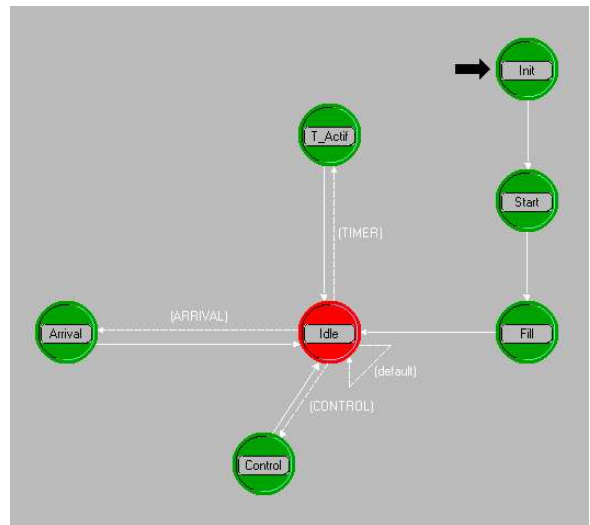


Figure 3: FSM of buffer_queue process model

First Step - Initialisation:

Init state: This state consists of the initialisation and is the first state entered by the process model. The initial interrupt is a begin simulation interrupt. User defined attributes at the node and simulation levels are loaded and the distribution variables are initialised for the spatial spreading of the MUs over a PG.

Start state: This determines the maximum number of mobiles in each class of mobility for the specific PG based on the initial density and attraction defined at the node level. The number of mobiles in

each specific class is determined through the gravity law.

Fill state: Based on the number of mobiles for each class of mobility, packets having a default size of zero are created representing mobile users. At the same time, the Interface Control Information pointer (ICI) is used in order to create an association between the packets and their respective ICI pointers. The following information are kept in the ICI pointer: *packet id* (refers to the mobile ID), last and new CM (i.e. Business, Residential or Others), x & y co-ordinates, velocity and the environment (Working, Residential, Streets, Leisure) of the Mobile. A self interrupt is scheduled for each class of mobility to initiate the departure of the mobile from a particular node to the surrounding nodes based on the HLPPS algorithm.

Idle state: This is the state in which the process waits for the next interrupt. The three interrupts: TIMER, ARRIVAL or CONTROL will initiate a transition to any of the three forced states: T_Actif, Arrival or Control states respectively.

Second Step:

T_Actif state: This forced state occurs when the timer of a CM expires. The corresponding CM queue is checked and if the queue is not empty, a packet is chosen randomly from the queue and forwarded to the *Class Processor Module* and a self scheduled interrupt for the corresponding class of mobility is generated for future events.

Arrival state: This state is called when a packet arrives at the input stream of the module from the *Stay_In* module. The process retrieves the corresponding attributes of the packet from the ICI pointer, calculates the new x & y co-ordinates and sets new attributes of the packet to the ICI pointer. A single *RESULTS* data structure is used to store the information of the packet and the list structure is shown below:

```
typedef struct
{
    double simtime; /* Simulation Time */
    int mobID; /* Mobile ID */
    int lastenvironment; /* Last Environment */
    double lastx; /* Last X Co-ordinate */
    double lasty; /* Last Y Co-ordinate */
    int lastclass ; /* Last Class of Mobility */
    int newenvironment; /* New Environment */
    int newx; /* New X Co-ordinate */
    int newy; /* New Y Co-ordinate */
}
```

```
int newclass; /* New Class of Mobility */
double velocity; /* Velocity of Mobile */
} RESULTS;
```

Control state: This forced state is executed when a remote interrupt occurs due to a change in attraction in the PG for a specific CM. This states re-calculates the steady-state conditions of the model defined by the Gravity Sub-Model and jumps to the 'Idle state'.

3.2.2 Process Model of Class Module

The Class process model, shown in Figure 4, a moving mobile will change its corresponding CM based on its attractivity weight as compared to PG attractivity defined through the '*attractivity function*' at a specific time of the simulation.

The state 'Init' initialises the state variables and loads the corresponding CM together with their respective initial attractivity weights for each PG. Then the Class FSM transits to the 'Idle' state. Once a packet arrives, on the input stream line of the Class module, the FSM jumps to the 'Arrival' state. The ICI pointer is used to retrieve the CM of the packet and its attractivity weight is compared with all the attractivity weights of each CM at the PG and based on this comparison, the MU is either assigned a new CM or retains its previous CM. The fields of the ICI pointer are updated and the packet is forwarded to the Direction Module through the output stream. The FSM transits to the 'Control state' only if a change in 'attractivity' defined by the '*attractivity function*' occurs during the simulation process updating the state variables of the corresponding CM before transiting to the 'Idle state'.

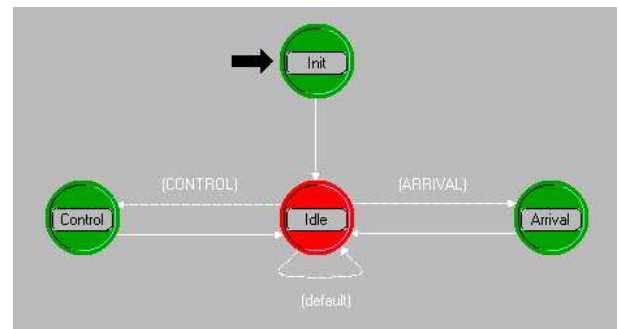


Figure 4: FSM of Class process model

3.2.3 Process Model of Direction Module

Figure 5 shows the process model implemented in the Direction Module. This process model implements the functioning of the Fluid Sub-Model by constantly calculating the 'transitional probability matrix' and forwarding the packet to the desired PG. It consists of four main states -

'Init' state, 'Prob' state, 'Forward' state and 'idle' state.

The initial interrupt is a begin simulation interrupt and the 'Init' state is used to initialise state variables for the simulation and read the angular connectivity 'R' defined at the node level. The PG is divided into R sectors and the closest nodes surrounding it specifies the most probable paths that the packets will take during simulation process thereby constructing a 'transitional probability matrix'. This state also schedules a self-interrupt in order to update the 'transitional probability matrix' through a minimum time of 10 seconds and transits to the 'Idle' state. The update of the probability matrix is performed in the 'Control' state and schedules another self-interrupt based on the 10 second period. Upon the arrival of a packet on its input stream, the FSM jumps to the 'Arrival' state and this process determines whether the packet moves to a new node or stays in the same node.

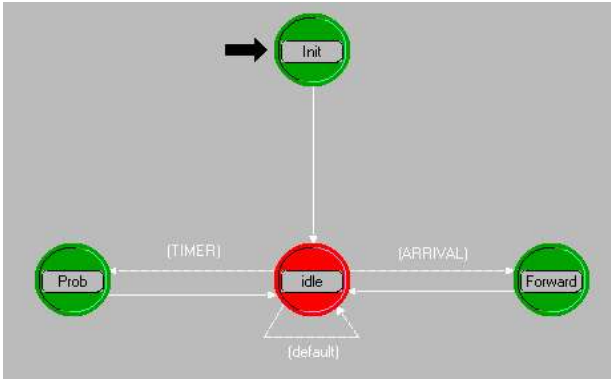


Figure 5: OPNET Process Model of Direction Module

3.2.4 Process Model of Control Module

This process ensures that the attractivity weight for each CM that is affected during the simulation process through the 'attractivity function' is updated accordingly through the Buffer and Class Modules.

In this FSM, the 'Init' state reads the attributes of the 'attractivity function', i.e. the time of change, the CM which is going to be affected and its associated attractivity weight, and schedules a self-interrupt at those intervals of time and transits to the 'idle' state. In this unforced state, the process waits for a self-interrupt and upon its reception, the process proceed to the 'T_Actif' state. When this forced state is entered two remote interrupts are scheduled to the Buffer and Class modules. In the Buffer module, the steady state values of the mobiles is re-calculated while in the Class module, the new attractivity weight for the affected CM is updated. The implementation of the FSM of the Control Module is shown in Figure 6.

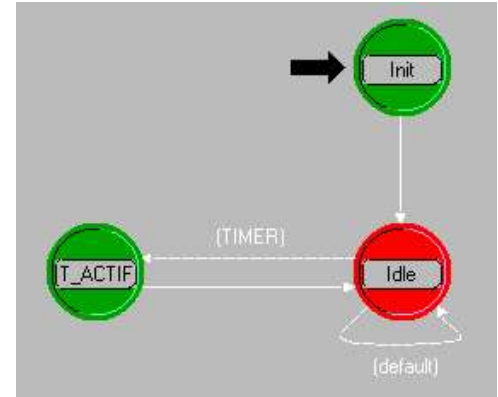


Figure 6: FSM of Control process model

3.2.5 Process Model for Port_IO module

Figure 7 shows the process model of the Port_IO module. This process model is used store the mobility trace generated during simulation for each individual packet.

The principle of this process model is very basic and consists of four FSM. The 'Init' state is entered during the begin simulation interrupt and the process gets the object attributes of the file from the simulation level. In this forced state, the statistic files are opened in order to write the mobility trace produced during the simulation. It schedules a self-interrupt for the dumping of the statistics generated in the file and transits to the 'Idle' state. When a self-interrupt is encountered, the FSM moves to the 'Store' state and writes statistics to the output file and proceeds to the 'Idle' state. When the FSM receives the 'end of simulation' event generated by the simulator, it transits to the 'End' state, closes the output file, releases control back to the simulator and releases the memory.

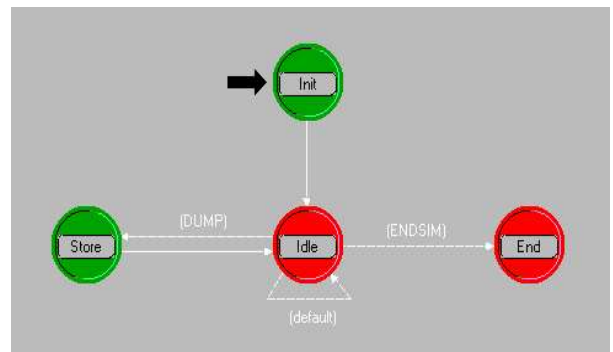


Figure 7: FSM of Control process model

4. Simulation: Mobility Model Validation

Based on the modelling approach in OPNET detailed in section 3, the architecture of the simulator is shown in Figure 8. The mobility simulator was used in an OPNET simulation to test and verify the

working of the model. The goal of the simulation was in twofold: Firstly, confirm that the stochastic system modelled in OPNET centred around the HLPPS queuing theory characterised mobile user behaviour and secondly use the mobility trace generated to show the importance of user mobility while examining traffic patterns for 3G EDGE systems.

The simulation was conducted in the City Area of Bristol, shown in Figure 1 for 1700 mobile users distributed in three CMs, *Business*, *Residential* and *Others* and over four environments, *Working*, *Residential*, *Shopping* and *Streets*. The simulation time ranges from 7.00 a.m. to 10.00 a.m., thus including mobile user displacement before, during and after morning rush hours (7.00 a.m. – 9.00 a.m.).

Figures 9 (a) and (d) illustrate the working of the model indicating population movement in the different environments with time. From the above, we note that user displacement moves from the residential environments during the off-peak hours and converge towards the working and shopping environments and in the process, some mobile users change their class of mobility irrespective of their past history. This movement is initiated through the 'attractivity function' which changes for each CM as time proceeds.

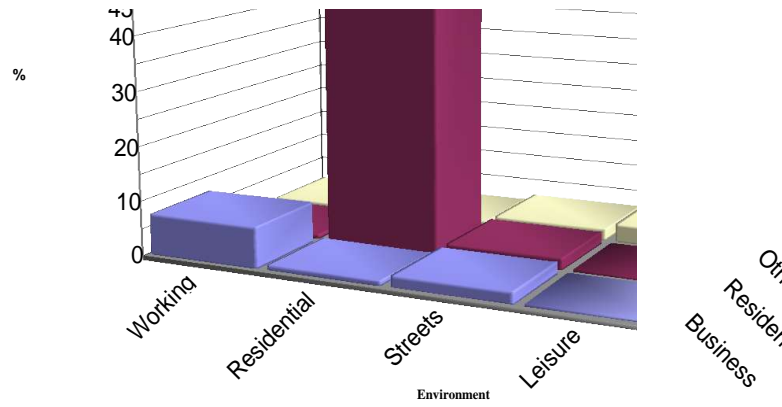


Figure 9 (a): Spatial Distribution of mobiles at 7.00 a.m.

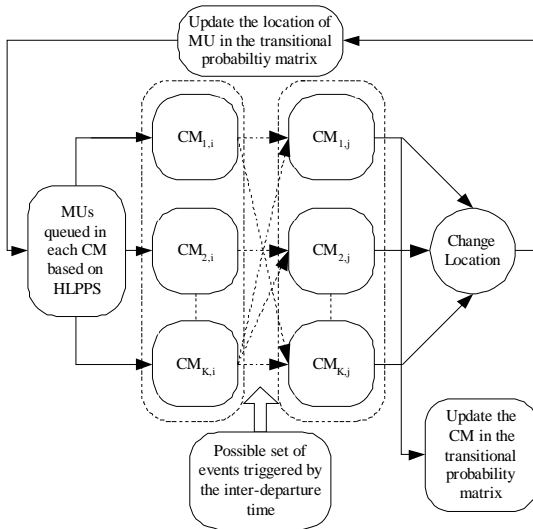


Figure 8: Mobility Simulator Architecture

Figure 9 (b): Spatial Distribution of mobiles at 10.00 a.m.

4.1 Mapping Terminal Mobility into Traffic Demand

In this section, we evaluate the performance of a 3G EDGE system, by integrating mobility trace generated by the event driven simulator OPNET into a discrete event object orientated simulator coded in Java.

Figure 10 (a) shows the user traffic for 1700 active voice services users moving through the city centre scenario. The plot shows the average active users per cell. While the base stations are laid out in a regular grid with a cluster size of 4, the effect of shadowing means that the cells themselves vary slightly in coverage and therefore size. The occurrence of hot spots of traffic can clearly be seen. Figure 10 (b) shows the actual carried traffic for a three-carrier EDGE system with the same base sites and a packet speech service. It can be seen that the carried traffic has been shaped by the available capacity, although it is still far from uniform. While additional cells in the hot spot areas would improve deployment, the example shows the requirement to consider the network as well as mobility when examining traffic patterns.

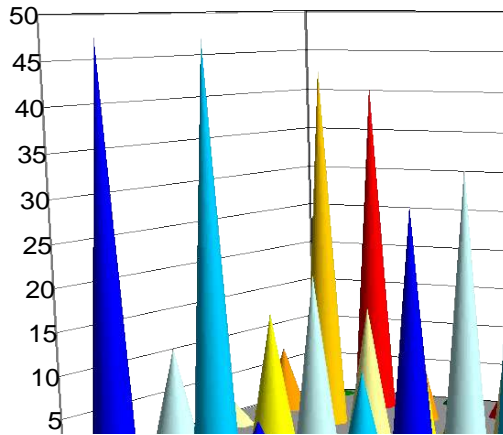


Figure 10 (a): Spatial Distribution of Traffic by Base Station Area (i.e. ideal offered traffic)

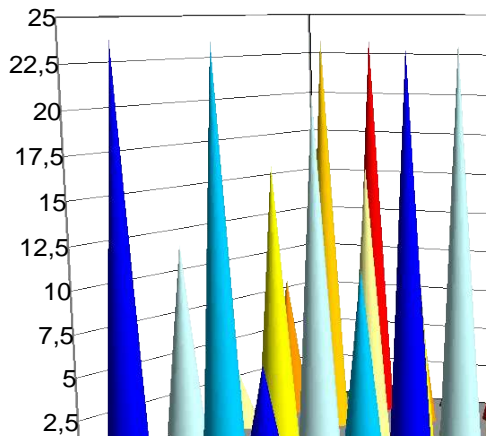


Figure 10 (b): Spatial Distribution of Carried Traffic by Base Stations

5. Concluding Remarks

In this paper, we have introduced a mobility model which allows the tracking and location of mobile users on a per mobile basis efficiently over a number of different scales. The model is based on queuing theory and we have used the results of Head-of-the-Line Proportional Processor Sharing algorithm to model user displacement in the geographical region of Bristol. We have taken advantage of the capabilities of the discrete event-driven simulation tool OPNET to model the novel concept of PG. As a first step, for further investigation related to teletraffic issues in the context of a wireless network, we have also provided a mapping between user displacements into traffic demand.

Acknowledgements

The work reported in this paper has formed part of the Work Area 2 of the Core 2 Research Programme of the Virtual Centre of Excellence in Mobile & Personal Communications, Mobile VCE, www.mobilevce.com whose funding support, is gratefully acknowledged. More detailed technical reports on this research are available to Industrial Members of the Mobile VCE.

References

1. Lam D., Cox D. C., Widom J., "Teletraffic Modelling for Personal Communications Services," *IEEE Communications Magazine*, Vol. 35, No. 2, 1997, pp.79- 87
2. Markoulidakis J., Lyberopoulos G., Tsirkas D., Sykas E., "Mobility Modeling in Third-Generation Mobile Telecommunications Systems," *IEEE Personal Communi s*, pp. 41-56, August 1997
3. Dahlberg T. A., Jung J., "Teletraffic Modelling for Mobile Communications", *Proc. of the International Conference on Communications (ICC98)*, June 1998.
4. PLANET, Mobile System International Ltd.
5. Bramson, M., "Convergence to equilibria for fluid models of certain FIFO and processor sharing queuing networks," *In F. P. Kelly, S. Zachary and I. Ziedins editors, Stochastic Networks, Theory and Applications*, Oxford Science Publications, pp. 1 – 17, 1996.
6. Bageet D. R., Dugenie P., Munro A., Kaleshi D., Irvine J., "Mobility Modelling for Future Wireless Communications," *submitted to MobiCom2002*