

# Generating Realistic Node Mobility and Placement for Wireless Multi-Hop Network Simulation

Bratislav MILIC<sup>(†a)</sup> and Miroslaw MALEK<sup>(†b)</sup>, *Nonmembers*

**SUMMARY** There exists a considerable number of node placement models and algorithms for simulation of wireless multihop networks. However, the topologies created with the existing algorithms do not have properties of real networks. We have developed NPART (Node Placement Algorithm for Realistic Topologies) in order to resolve this fundamental issue in simulation methodology. We compare topologies generated by NPART with open wireless multihop network in Berlin. The NPART generated topologies have almost identical node degree distribution, number of cut-edges and vertices as the real network. Unlike them, topologies generated with the common node placement models have their own characteristics which are considerably different both from NPART and from reality. NPART algorithm has been developed into a tool. We propose a method and present a tool for integration of NPART with various realistic node mobility algorithms and tools, such as Citymob [1] and MOVE [2]. This integrated tool allows easy and time-efficient generation of highly complex, realistic simulation scenarios. We use the tool to evaluate effects of integration between existing open community wireless multi-hop networks and vehicular ad-hoc networks (VANETs). The evaluation shows that despite partial coverage and peculiar topological properties of open networks, they offer high levels of performance and network availability to the mobile end users, virtually identical to performance and availability of planned, dedicatedly deployed networks. Our results indicate that the integration of these networks may bring considerable benefits to all parties involved.

**key words:** *wireless multi-hop networks, simulation, simulation models, node placement, node mobility*

## 1. Introduction

Wireless multi-hop networks (WMN) are composed of autonomous processing nodes that use wireless network adapters for communication and share a common set of communication protocols. In case that two nodes are unable to communicate directly, a subset of network nodes is responsible for message relaying, hence they are called multi-hop. They form a distributed system that is primarily used for communication, although other distributed or centralized applications and services may be deployed in it.

At the time of their introduction, WMNs were envisioned as general purpose networks, applicable in wide class of scenarios. However, some of the proposed application scenarios, such as communication in a general purpose mobile ad-hoc network, did not match actual user needs [3], and have not been used in practice despite numerous developed protocols. Research community has realized the need for application-oriented protocols. The current trend is to focus

on realistic scenarios and innovative applications, such as the ubiquitous Internet access, sensing, vehicular, and logistics applications.

Different use-scenarios may require dedicated communication protocols in order to offer sufficient quality of service (QoS) to end users. Development and verification of WMN protocols is usually performed by simulators due to their low operating cost and fast setup. The quality of simulation directly depends on used simulation model. The simulation model of a WMN is complex and it consists of six sub-models: *node model* describes hardware of a node; *deployment model* provides node positions (network topology); *mobility models* define node movement patterns in mobile networks; *radio model* describes the characteristics of the radio used by the node; *wireless signal propagation model* deals with characteristics of wireless channel and effects of the environment on the channel; *traffic models* define traffic patterns in a network.

Some simulation sub-models are based on real data measurements, such as wireless signal propagation [4], traffic models [5], mobility models [1]. However, there exists a notable lack of topological measurements in real-world WMNs and researchers are forced to use artificial node placement models in simulation. Node deployment belongs to crucial simulation sub-models and have high impact to the outcome and quality of simulation since topology of a simulated network directly depends on node positions.

In our previous work [6], [7] we have analysed wireless mesh networks in Berlin and Leipzig and shown that they are substantially different than topologies generated by the well-known node placement models from the literature. To resolve that issue, in [8] we have proposed NPART algorithm that generates realistic topologies for WMN simulation and we describe it briefly in Sect. 3. In Sect. 4 we extend NPART by describing a methodology and corresponding tool that integrates the NPART algorithm with the existing realistic mobility generators for VANETs (Vehicular Ad-Hoc Networks). This extension allows the comprehensive modelling of the realistic urban scenarios for evaluation of WMNs, VANETs or their combination.

In Sect. 5, we apply the extended NPART tool to evaluate the benefits of integration between existing community networks (such as the Freifunk networks [9]) and VANETs. We observe different QoS milestones (throughput, probability of a successful connection, connection duration) and show that this integration is crucial for the success of VANETs in their early stages of adoption, when VANETs

Manuscript received February 1, 2012.

Manuscript revised March 30, 2012.

<sup>†</sup>The authors are with the Humboldt University in Berlin, Germany.

a) E-mail: milic@informatik.hu-berlin.de

b) E-mail: malek@informatik.hu-berlin.de

DOI: 10.1587/transcom.E95.B.2682

on their own are not sufficient to provide meaningful network performance and availability to the end users.

## 2. Node Placement and Mobility in WMN Simulation

There exists a number of different deterministic and random node placement and mobility models that shape the outcome of WMN simulations. In this section we briefly introduce some of them and evaluate how do they compare to real wireless multi-hop networks.

### 2.1 Node Placement Models

In WMN research, the most frequent node placement models are grid and uniform. The grid placement model puts nodes at intersections of a rectangular grid. It is highly organized so its use is justified only for simulation of networks that will actually have such structure (i.e., planned networks). In the uniform placement model, a placement area (rectangular or circular) of size  $|A|$  is chosen and  $n$  nodes are placed inside of it with the uniform probability  $p_{uniform} = \frac{n}{|A|}$ . It is artificial and additionally, in order to have a connected topology obtained from the uniform node placement, node density has to be high. In order to obtain a network that is connected with a high probability, the average degree of nodes in it should be between 10.8 [10] and 13.8 [11]. Such dense networks have little in common with the measurements we made in the reality, where average degree of nodes is considerably lower and the density is not uniformly distributed in the network.

Several non-homogeneous node placement models have been proposed as well. Bettstetter et al. [12] place nodes in accordance with the uniform process and then apply thinning to it. Onat and Stojmenovic [13] developed several algorithms that create connected topologies with high probability and allow user to choose the average node degree. Unfortunately, the properties of topologies generated according to these models were not analysed in these papers nor compared with real networks.

Liu and Haenggi [14] propose quasi-grid placement model. It selects vertices from a realization of the uniform placement model such that every selected vertex is closest to a regular grid point. The obtained topologies resemble the grid structure but they are not as regular as grids.

### 2.2 Node Mobility Models

A bit surprising, node mobility models are much more realistic than the node placement models. Only the Random Waypoint Model (RWM) can be considered artificial, while other models in use are either reality-inspired or they truly reflect the reality in detail.

In RWM model user defines the minimum ( $v_{min}$ ) and maximum ( $v_{max}$ ) allowed speeds of nodes, and the pause time between two movements. A node chooses a random point in the placement area and heads towards it with a speed selected from  $U(v_{min}, v_{max})$ . Once the destination is

reached, the node waits for pause time and then repeats the process. It was believed that RWM preserves uniform distribution of speed and that the average speed of nodes is arithmetical mean of minimum and maximum speed. It was proven in [15] that these assumptions are not true. The average speed asymptotically approaches the minimum speed, which particularly affects simulations that set the minimum node speed to zero—instead of envisioned highly mobile network, an almost static network is simulated.

Community model [16] defines placement area that is divided into  $s$  sub-areas, so called “communities”, a “gathering place” and “home”. Each node has one home community. A node is more likely to visit its home community than other places. The authors argue that this model captures human mobility in village communities.

Simulation of Urban MObility (SUMO [17]) is a road-traffic simulation package. It supports simulation of vehicle movement, multi-lane streets, different vehicle types, different junction types. SUMO is a pure vehicular simulator, so it has to be coupled with dedicated tools that can produce scripts for simulation of wireless networks. Several tools support this, such as the MObility model generator for Vehicular networks (MOVE) [2]. It runs SUMO, processes its output, and transforms it in ns-2 or GloMoSim scripts.

CityMob [1] generates ns2 scenarios for VANET simulation. It arranges streets in a rectangular grid, with equidistant placement of streets. Vehicles move on streets with a random speed, within a user-defined range of min-max values, from one random point to another. It provides several mobility models out of which the most interesting and realistic is the Downtown Model. In the downtown model, vehicles may stop at traffic lights, break down, and there is an area of interest (downtown) where vehicles move with lower speed.

## 3. Node Placement Algorithm for Realistic Topologies

The analysis of the existing node placement algorithms has revealed that they are highly artificial. The topologies produced by them are considerably different than what we observed and measured in reality and thus inadequate for realistic simulation of WMNs.

This discrepancy between used node placement models and real networks opens a question on validity of simulation results obtained by using such flawed models. Node Placement Algorithm for Realistic Topologies (NPART) eliminates this issue in the simulation of WMNs by creating realistic topologies.

### 3.1 Background and Assumptions

In order to create realistic topologies, input to the topology generator should originate from real networks. Simultaneously, in order to provide an extensible topology generator, its input should be something that is sampled rather easily in real networks. For instance, information such as node location in a real network is not a good choice for the topology

generator since it may be impossible to obtain in real networks where users do not want to disclose their locations.

We use network topology as the initial input, since it can be sampled from user-initiated networks with ease. In some cases this data is even available from web presentations of such networks [9]. The input data to the placement algorithm should not originate from a single instance of a real topology, but it should be an aggregate topological property that represents overall characteristics of a real network. Of various possible topological properties we use the degree distribution as the input parameter of the NPART.

The degree  $d(v)$  of a node  $v$  in a graph is the number of edges incident on  $v$ . The frequency of an event  $i$  is the number  $n_i$  of times the event occurred in an experiment. The frequency can be absolute, when the counts  $n_i$  are given, and relative when counts are normalized by the total number of events. Degree distribution is the frequency of occurrence of node degrees in a graph. It can also be absolute or relative.

We have chosen the node degree distribution since it provides a compromise between input detail level, feasibility of sampling in real networks, data anonymity, and captured realism.

### 3.2 Algorithm Description

The NPART algorithm is presented in Fig. 1. Input parameters of the algorithm are the number of nodes to be placed  $n$ , communication radius of nodes  $R$ , and the desired absolute node degree distribution  $target$  of the network to be produced. The target degree distributions in the algorithm implementation (described in Sect.4) were sampled from Berlin and Leipzig networks, but other degree distributions can be used as well.

NPART adds nodes to the produced topology one by one. In an iteration  $I_k$  there are already  $k$  placed nodes that form an intermediate topology and NPART places the  $(k + 1)$ th node. The set of already placed nodes is the variable  $placedNodes$ .

The area occupied by the network (ordered pair  $networkArea$ , gray area in Fig. 2) is not pre-specified by user as it is common in other WMN topology generators, but it may change in iterations, depending on the locations of the placed nodes (line 16). Network area is defined as a rectangle that includes all already placed nodes and it is defined as ordered pair of coordinates  $((minX, minY), (maxX, maxY))$ . In Fig. 2, it is marked by the grey rectangle. The first node is placed at  $(0, 0)$ , so the initial network area is set to  $((0, 0), (0, 0))$  (line 2).

The network area from the iteration  $I_k$  is used to determine the placement area of candidate nodes in the iteration  $I_{k+1}$ , by extending it by  $R$  (line 4). A candidate node is placed so that its  $x$  coordinate is uniformly sampled from  $(minX - R, maxX + R)$  and its  $y$  coordinate from  $(minY - R, maxY + R)$  (line 7). If the candidate node is not connected to already placed nodes, a new candidate node is generated until a connected topology is obtained (lines 7 and 8).

In every iteration NPART fits the degree distribution of

```

NPART(nodes  $n$ , communication radius  $R$ ,
desired degree distribution of network  $target$ ):
1   $placedNodes = \{ \text{new node at } (0, 0) \}$ 
2   $networkArea = ( (minX=0, minY=0), (maxX=0, maxY=0) )$ 
3  repeat
4     $placementArea = ( (minX-R, minY-R), (maxX+R, maxY+R) )$ 
5     $minMetricValue = \infty$ 
6    repeat
7      create  $candidateNode$  uniformly random in  $placementArea$ 
8    so that  $candidateNode$  is connected to  $placedNodes$ 
9     $candidateMetric = metric(placedNodes \cup candidateNode)$ 
10   if( $candidateMetric < minMetricValue$ )
11      $bestCandidate = candidateNode$ 
12      $minMetricValue = candidateMetric$ 
13   endif
14   until( $retry$  different node candidates are evaluated)
15    $placedNodes = placedNodes \cup bestCandidate$ 
16   update  $networkArea$  based on  $bestCandidate$  location
17 until(all  $n$  nodes are placed)

```

Fig. 1 NPART pseudo code.

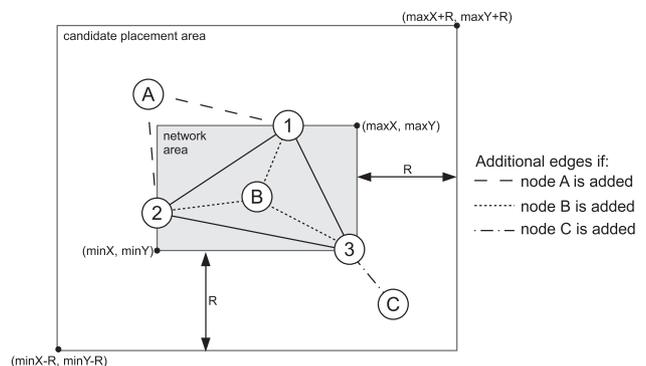


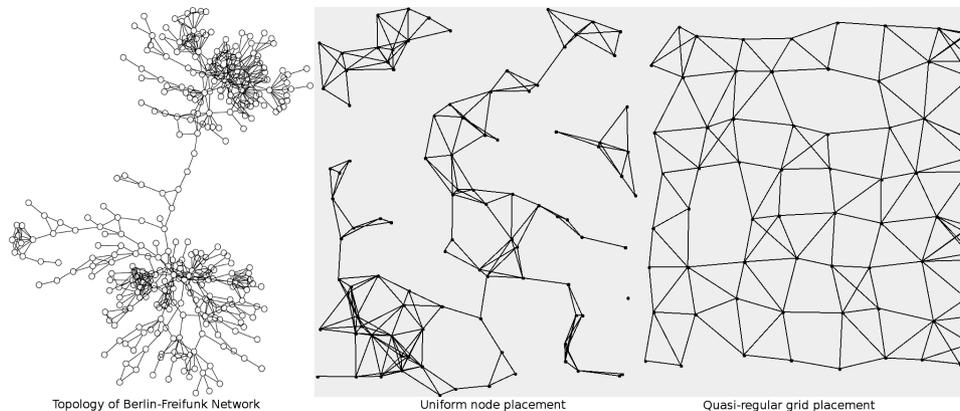
Fig. 2 Node placement area and evaluation of candidate nodes.

the intermediate topology to the desired ( $target$ ) degree distribution. In order to provide good fit between the  $target$  degree distribution and the degree distribution of the end-result topology, before a node is actually added to the topology in an iteration, several candidate nodes are evaluated (lines 6-14). The number of evaluated candidates per iteration is defined by the algorithm parameter  $retry$ .

The NPART-produced topology should have the node degree distribution similar to the  $target$  distribution. For each candidate node that is evaluated in an iteration, we calculate the absolute degree distribution  $candidate$  of the topology that the candidate creates with already placed nodes ( $candidateNode \cup placedNodes$ , line 9). Then, the metric  $M$  (Equation 1) is applied to the  $candidate$  degree frequency in order to estimate how close is it to the  $target$  degree distribution. The candidate node with the best (lowest) metric value in an iteration is selected to be added to the topology in that iteration (lines 10-13 and 15).

### 3.3 Metric Description

The key part of the algorithm is the selection of the best candidate node out of  $retry$  candidates that are evaluated in



**Fig. 3** Visual comparison of a Berlin's WMN, a uniform node placement and a quasi-grid topology.

each iteration of the algorithm. In this section we describe the metric that we use in NPART and provide rationale why did we create such a metric.

Let us observe a simple example where the *target* distribution should have two nodes of degree four, six nodes of degree three and two nodes of degree one. Let us assume that the intermediate topology after three iterations of the algorithm is as shown in Fig. 2. Already placed nodes are 1,2 and 3 and we evaluate three candidates A,B,C in the next iteration.

If we add the candidate A to the intermediate topology, the resulting topology has two nodes of degree two (which is undesired since there are no such nodes in the target degree distribution) and two of degree three. The candidate C produces topology with one node of degree 3, two of degree 2 and one of degree 1. Again, if this node is added to topology, the topology will have nodes of degree that do not exist in the target topology, thus the agreement between such intermediate topology and desired topology is not particularly good. The candidate B creates topology with four nodes of degree three out of six that are desired in the target topology, and no nodes with undesired degrees.

We can make two important observations in this example. First, some node degrees are more frequent in the target topology than others. It is beneficial for the final topology and its fit with the *target* degree distribution to select those intermediate topologies that produce these “popular” node degrees. Second, if a candidate node produces a topology that exceeds the targeted degree, such topology should be penalized (metric value increased). Thus, we define the goodness metric of an intermediate topology as:

$$M = \sum_{\text{degrees}}^d (1_{\text{target}_d - \text{cand}_d > 0} \cdot (\text{target}_d - \text{cand}_d) \cdot w_d + 1_{\text{target}_d - \text{cand}_d < 0} \cdot p \cdot (\text{cand}_d - \text{target}_d)) \quad (1)$$

where *target* is the absolute degree distribution of the desired topology, *cand*. (short of *candidate*) is the absolute degree distribution of the topology created by union of already placed nodes and the candidate node. Notation  $\text{target}_d$  and  $\text{cand}_d$  marks the number of nodes of degree  $d$  in the distri-

bution.  $1_A$  is the indicator function, returning one if predicate  $A$  is true, zero if it is false.

The metric sums the difference between proposed and target node frequency for a degree  $d$  over all degrees, if the difference is positive. This difference is weighted by  $w_d = \frac{|\text{target}_d - \text{placed}_d|}{\sum_{\text{degrees}} |\text{target}_i - \text{placed}_i|}$ , where *placed* is the absolute degree frequency from the topology produced in the previous algorithm iteration (this weight favours production of “popular” degrees). If the difference between target degree and candidate degree is negative (candidate topology has more nodes of a certain degree than the target topology), absolute value of difference is multiplied with penalty factor for degree overloading  $p$ .

In [8] we have evaluated four metrics for selection of the best candidate node. Here we have presented and explained in detail only the best identified metric. Other details and refinements of NPART can also be found in [8].

### 3.4 Evaluation of NPART Topologies

This section demonstrates an advantage in terms of realism of properties of topologies produced by the NPART algorithm compared to properties of topologies produced by the common topology generators from literature.

We analyse NPART, the uniform and the quasi-grid placement models. In the analysed topologies, 275 nodes are placed (the average number of nodes in network in Berlin [7]). Parameters of the uniform placement algorithm are chosen to create topologies with the average node degree of six. Denser uniform networks have even greater discrepancy with measurements in reality (low diameter, excellent connectivity) while sparser uniform networks are highly partitioned. The quasi-grid placement algorithm is implemented as described in [14]. The anchor points are placed in 16 by 16 grid. The NPART is run with the following setup: it generates 275 vertices and degree data input is from Berlin's network (NPART/Berlin). The parameter *retry* is set to 150, penalty  $p$  is 5. The data shown in Figs. 4 to 6 is based on 500 executions of the each algorithm.

The properties of the generated topologies (NPART,

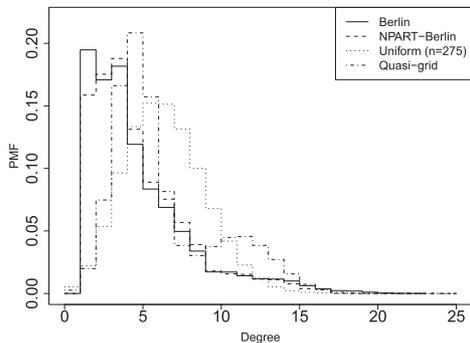


Fig. 4 Comparison of node degree distributions.

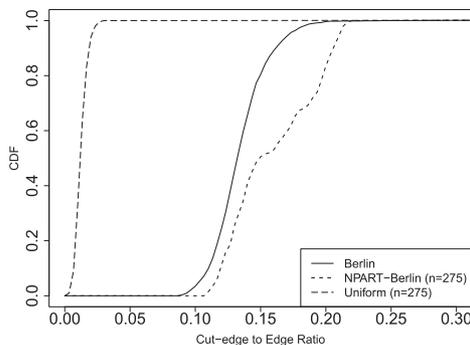


Fig. 5 Comparison of cumulative distributions of cut-edge to edge ratio.

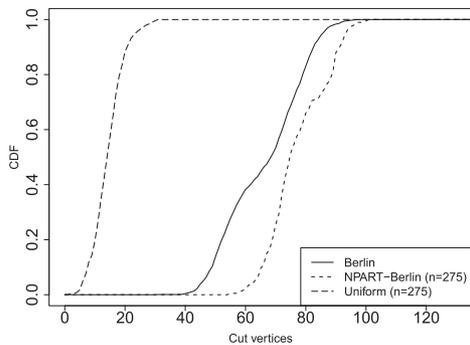


Fig. 6 Comparison of cumulative distributions of cut-vertices.

quasi-grid, and uniform) are compared to properties of real networks. A topology generator is considered better if its topologies resemble the properties observed in reality.

Figure 3 informally illustrates the differences between real topology from Berlin’s network, a topology created by the uniform placement model, and a quasi-grid topology. The uniform-placement model topology has distinguishably different shape than the real topology: there does not exist notable clustering of nodes as in reality. Also, real sample has numerous cut-edges and cut-vertices (edges and vertices whose removal partitions the network), both on network outskirts and in its central parts. Despite its small irregularities, quasi-grid remains very organized and it still closely resembles a grid. There are neither high-density areas in it, nor cut-edges and cut-vertices. An example of NPART topol-

ogy can be see in Fig. 9 and we can notice that it is more similar to the real network than topologies created by the models from literature.

We base the objective comparison on three topological metrics: degree distribution, cut-edge (bridges) and cut-vertex (articulation points) count. We choose these topological metrics because they directly influence properties of protocols that are simulated in analysed topologies. The node degree distribution is correlated to the congestion on the wireless channel and probability of packet loss. Cut-edges and cut-vertices are critical for network connectivity since their failure may partition the network. They are also congestion points for network traffic since they connect larger, well connected subnetworks.

Figure 4 shows the vertex degree probability mass function (PMF). Topologies created by NPART precisely follow the distribution of node degrees in reality. The uniform and quasi-grid distributions have their own distributions that are considerably different from reality.

Similar behaviour can be observed for the cut-edge to edge ratio (Fig. 5) and cut-vertex distributions (Fig. 6) where NPART topologies follow the properties of real networks. The uniform placement model is unable to represent the reality: it produces topologies with less than 1% of cut-edges and only one fourth of cut-vertices that are observed in real networks of same size. The quasi-grid is even less related to reality — due to its highly organized structure the number of cut-edges and cut-vertices in topologies is negligible (in our experience, approximately every third topology had a cut-edge or a cut-vertex).

#### 4. The NPART Tool and Its Integration with Realistic Mobility Generators

The NPART algorithm has been implemented in Java, including easy to use graphic user interface (GUI). Tool’s configuration menu can be seen in Fig. 7. Various NPART parameters can be set and different target degree distributions selected. Beside NPART, the tool supports other placement algorithms (uniform, grid, stationary RWM node distribution) and configuration of their parameters. It exports the node locations in ns-2 and .dot formats. An importer of ns-2 topologies in Jist/SWANS simulator is also available from project’s website. Tool user can generate individual topologies or a whole batch of them.

The NPART and its supporting tool allow the realistic simulation of static networks. However, the emerging wireless technologies and applications (such as the smart grid, smart living, vehicular, and safety networks) are not necessarily static. Such networks are expected to influence all aspects of modern life and the wireless nodes that form them will interact in complex patterns. One of the likely technological developments is integration of the existing, so far independent wireless networks (static and mobile) and technologies in order to provide better support of user needs.

In order to support this technology development trend and simulation of new protocols that will support these ap-

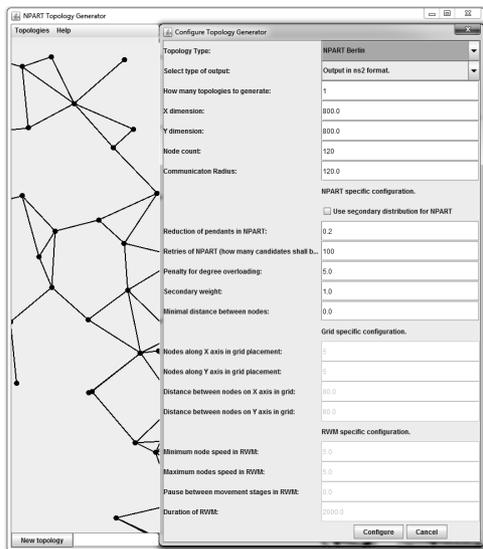


Fig. 7 Screenshot of the NPART tool, in the foreground is the configuration menu, in the background a part of a generated topology.

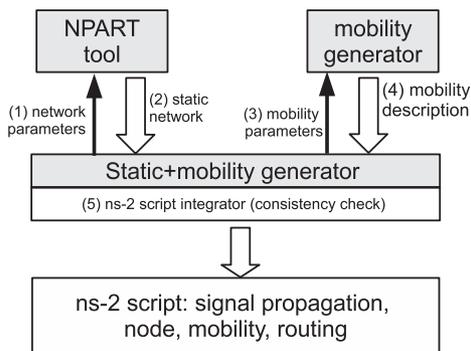


Fig. 8 Integration of NPART and node mobility generators.

plications, we provide a method and tool for integration of NPART topologies with the realistic mobility generators that exist in literature (such as Citymob [1] or MOVE [2]). The proposed approach enables easy and time-efficient generation of realistic simulation scenarios for the next-generation networks and applications.

Figure 8 shows our approach. The common wireless simulation parameters are provided as input to our integration scripts. These parameters include number of static and mobile nodes, wireless signal propagation characteristics, type of nodes, transmission power, characteristics of antenna, etc. They are forwarded by the integrator to NPART and mobility generators to build the common part of simulation scenarios.

The first step in generation of a static-mobile scenario is to execute NPART. Then, the mobility generator is invoked with its set of parameters, plus the size occupied by the static WMN. The geographic area is needed by the mobility generator as limit for node movement. Since NPART does not operate on predefined network area but allows network to grow and define its own deployment area, size has to

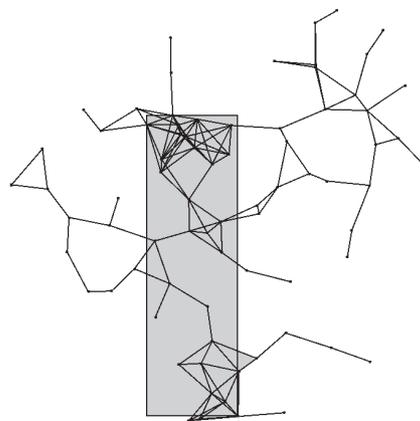


Fig. 9 Downtown example (grey rectangle) in an NPART topology.

be extracted (it is not one of initial simulation parameters).

Some generators, such as the MOVE, require loading of a real street city map. In this case, a map (that approximately fits the NPART network area size) is selected from the existing maps and passed to the mobility generator. Other mobility models require definition of the area within a network where nodes spend more time than in remainder of network (“home area” in [16], “downtown area” in [1]).

Static and mobile generators provide two separate and possibly overlapping simulation scripts. Since we use mobility generators as-is, some of them include simulation header in their output, which is also provided in scripts generated by NPART tool. The task of the *script integrator* is to merge the static and mobile simulation scripts, to remove double definitions of simulation parameters, to provide consistent node identification (each generator uses its own numbering). Integration process cannot be fully automatized since there is no standard for simulation-script generation. As the consequence, the script integrator requires a certain degree of customization for each mobility generator. On the positive side, once the script integrator is customized, it is fully reusable.

We have implemented the integration scripts in Perl programming language for integration of NPART topologies with Citymob generator but the process is similar for other mobility generators. In addition to the listed common script integration tasks, it is also necessary to determine the downtown area in NPART topology and to forward it to the Citymob.

In our approach, the downtown area is defined as the area with high node density. To determine it, we load ns-2 script in our Perl script, recreate generated topology, and calculate node degrees in it. The nodes are sorted by their degrees in descending order. Twenty percent of nodes with highest degrees are chosen as the nodes belonging to the downtown. A rectangle that includes them is the downtown area. We introduce another constraint on the downtown area — it cannot be larger than one quarter of the whole map. There is no technical reason for this constraint, but a logical one: it is to expect that downtown of a city is considerably smaller than the whole city. If the downtown area is too

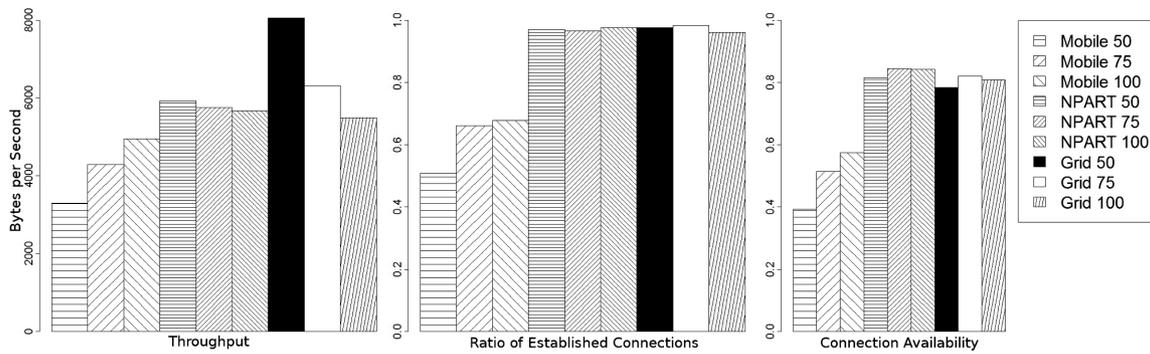


Fig. 10 Simulation results.

large, the node with the least degree is removed from the list of downtown nodes and new downtown area is calculated. This process is repeated until satisfying area is determined. This limitation occurs rarely, since NPART topologies have a small dense core, as it can be seen in Fig. 9 where the downtown area is grazed out.

##### 5. Integrating VANETs and Open Community Networks for Performance and Availability Improvement

We demonstrate the use of the developed toolkit to evaluate the possibility of integration of the existing urban wireless networks and the envisioned VANETs. Certain QoS metric (such as network performance and availability) have to be met already in the early stages of VANET adoption, otherwise the early adopters of the technology may experience dissatisfaction and disappointment leading to rejection of the technology by users due to its initial low performance/inadequate user experience.

The goal of the evaluation is to compare QoS metrics of a VANET if it is supported by a planned network and in scenario where it uses the existing, open community WMNs such as those in Berlin, Leipzig, or Prague [9]. As QoS metrics of interest we identify the ratio of successful connections, the throughput established by the connections, and the connection availability. We define connection availability as the ratio between time in which a connection was functional and time in which user wanted to use it (i.e., since it is opened by an application, until its closure).

We use NPART tool and its extension described in Sect. 4 for creation of integral, static-mobile simulation scenarios. Citymob2 was used as the mobility generator in the developed framework. The scenarios are evaluated in ns-2 simulator [18] extended with the Rayleigh fading plug-in [19] for wireless signal propagation. Rayleigh fading is the dominant factor in signal propagation in mobile networks [4] so it is crucial to incorporate it in simulation, otherwise the simulation would not reflect the reality [20], [21].

The movement patterns and initial positions of mobile nodes are determined by the Citymob [1]. The downtown area needed for Citymob is determined according to the algorithm we presented in the previous section. Maximum

speed of nodes is 100 km/h outside of downtown area and 50 km/h in the downtown area. Minimum node speed is 25 km/h. There may be up to three node breakdowns.

The routing protocol in the network is AODV [22]. We select a reactive routing protocol because the high mobility of vehicular nodes eliminates the benefits of proactive routing protocols. After warm-up phase that lasts for 50 seconds, five TCP flows are started between random pairs of mobile nodes. They start at random times within the 50 seconds after the warm-up phase and remain open until the end of simulation. The simulation lasts for 1200 seconds.

Warm up phase of 50 s is sufficient because AODV is a reactive routing protocol with aggressive purging of inactive routes. Prolonging the warm up time cannot change simulation results: nodes that are required to maintain the local connectivity have enough time to execute neighbour detection process (Sect. 6.10 in [22]). AODV purges the inactive routes from routing tables after 15 s of inactivity (Sect. 10 in [22]) so this also fits within our warm up phase duration.

There are three simulation setups of interest:

- Only mobile nodes are present (50, 75, and 100 nodes)
- There exist a hundred-node static grid and mobile nodes (50, 75, and 100 mobile nodes).
- There exist a hundred-node community WMN (topology is generated by NPART/Berlin) and mobile nodes (50, 75, and 100 mobile nodes).

The dedicated VANET-support network is simulated by placing 100 nodes in a 10 by 10 grid, in the placement area of  $2700 \times 2700$  meters (this size corresponds to the average area of 100-node NPART topologies). Inter-node distance in the grid is 250 m. We compare the use-case of a planned VANET-support network to a case where VANET is supported by an existing community WMN, also consisting of 100 nodes, placed in accordance with NPART/Berlin model.

Simulations are performed on 10 different NPART topologies. For node mobility, for each of the topologies there exist 20 different mobility scenarios. Since all grid topologies are identical, and there is no support network for mobile-only network, we simulate them with 200 different mobility scenarios.

Figure 10 shows the simulation results. The results of simulation of a pure mobile network are marked by Mobile,

followed by number of mobile nodes. The results of simulation of a mobile network that is supported by a planned network are marked by Grid followed by the number of mobile nodes. The results of the simulation of a mobile network that is supported by a community WMN are marked by NPART followed by the number of mobile nodes. So, for instance NPART 75 simulation scenario is the use case where communication of 75 mobile nodes is supported by a 100 node community network.

If we observe the throughput (which we measure only for established connections, during the time in which they existed) in Fig. 10(a), there is a considerable improvement in network performance introduced by the support of static networks.

If we observe the ratio of successfully established connections in Fig. 10(b), it can be seen the weakness of low-density VANETs—only 50 to 70% of connections are established, which indicates the clear need for their support in early phases of their use. The connectivity is sporadically provided in small sections of network where nodes are either in direct reach, or within 1-hop distance providing brief and bursty communication, so the acceptable throughput results are misleading—connections do not exist network-wide.

Figure 10(c) shows the connection availability. Pure low-density VANETs have very low connection availability, which inevitably leads to user dissatisfaction [23]. If VANET is supported by a community WMN, the connection availability considerably increases (in some scenarios up to twofold), providing more stability to a network and to the network QoS that leads to higher user satisfaction. Despite the irregular placement of community networks, they provide the same availability as the planned networks, but it their longer routes degrade the throughput a bit.

If we compare the results of the dedicated network and community WMN, it can be seen that planned network has some small advantages in performance, but for the availability the results are almost identical. In practice these small performance advantages may not be sufficient to justify deployment of a dedicated network. A static network that supports a VANET is crucial only in the early phases of the VANET use, while its market penetration is still low. As VANET use and density of VANET nodes increase, the gains obtained by the static support networks diminishes. The operators of VANETs (i.e., car manufacturers and their partners that will be working on vehicular networking) may be more interested in cooperation with the existing networks (setup and deployment costs are almost non-existent, maintenance costs are low due to community involvement) than in development and deployment of dedicated networks (with moderate to high setup and deployment costs, high maintenance costs) for a small performance increase.

## 6. Conclusions

We have proposed, developed and evaluated NPART - Node Placement Algorithm for Realistic Topologies. The algorithm provides realistic topological input to simulation

of static wireless multi-hop networks. We stochastically analyse various topological properties of NPART-produced topologies and compare them with topologies produced by the common node placement models (such as the uniform and quasi-grid) as well as with real networks in Berlin and Leipzig. The stochastic analysis shows that NPART produces topologies with properties of the real networks, while other artificial node placement models have their own properties that are far from reality. A GUI-enabled NPART tool has been developed and it is available for download at NPART webpage<sup>†</sup>. The tool currently supports ns-2 and Jist/SWANS simulators. It can also generate other node placements: uniform, grid and quasi-grid.

NPART allows evaluation of realistic static topologies. However, emerging wireless technologies, protocols and applications require more complex evaluation scenarios. It is to expect in the near future that existing, currently independent wireless technologies will be cooperating in order to support the growing user needs and expectations. To support this expectation trend and allow easier development of such technology, we also provide a method and a tool for integration of NPART topologies with the realistic mobility generators that exist in literature (such as Citymob [1] or MOVE [2]). The resulting toolset allows easy and time-efficient generation of realistic simulation scenarios for networks of various sizes, topologies and mobility patterns.

We demonstrate the use of the developed toolset to evaluate the benefits of the integration of static, user initiated networks that already exist in large cities (such as the Freifunk Networks in Berlin and Leipzig [9]) and of the developing VANET networks.

We first show that low density VANETs cannot deliver sufficiently good performance, network coverage and availability to the end users. In initial phases of VANET adoption it will be necessary to complement them with supporting, static WMNs that provide connectivity and resource redundancy. Then, we show that the existing community initiated WMNs, despite their irregular shape and partial city coverage, offer performance and connection availability that are very comparable with network performance and availability of expensive planned static networks (they are planned to provide complete coverage of the city area) that are dedicated to support VANETs.

This result opens possibility of integration of user-initiated WMNs and VANETs that can benefit both communities. Open WMNs may get benefits in form of the discounted/free Internet access provided by the car manufacturers which have the most interest in VANET adoption and reliable operation. In return, VANETs will reach the targeted QoS levels much easier than if depending solely on mobile networks. If legal hurdles are overcome, the biggest advantage for car manufacturers is ability to provide low cost, dependable, mobile communication by using the WMN's support for high throughput and fault tolerance, while minimizing the maintenance and deployment costs.

<sup>†</sup><http://www.rok.informatik.hu-berlin.de/NPART>

## References

- [1] F. Martinez, J. Cano, C. Calafate, and P. Manzoni, "CityMob: A mobility model pattern generator for VANETs," Proc. IEEE International Conference on Communications, pp.370–374, 2008.
- [2] K. Karnadi, Z. Mo, and K. Lan, "Rapid Generation of Realistic Mobility Models for VANET," Proc. Wireless Communications and Networking Conference, WCNC, 2007.
- [3] M. Conti, "Multi-hop ad hoc networking: From theory to reality," Proc. 10th International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems, 2007.
- [4] T. Rappaport, Wireless Communications, Principles and Practice, Prentice Hall, 2008.
- [5] N. Vicari, "Models of WWW-Traffic: A Comparison of Pareto and Logarithmic Histogram Models," Tech. Rep. 198, Institute of Computer Science, University of Wuerzburg, 1998.
- [6] B. Milic and M. Malek, "Analyzing large scale real-world wireless multihop network," IEEE Commun. Lett., vol.11, no.7, pp.580–582, 2007.
- [7] B. Milic and M. Malek, Guide to Wireless Ad Hoc Networks, ch. Properties of wireless multihop networks in theory and practice, Springer Verlag, 2009.
- [8] B. Milic and M. Malek, "NPART — Node Placement Algorithm for Realistic Topologies in Wireless Multihop Network Simulation," Second International Conference on Simulation Tools and Techniques (SIMUTools), 2009.
- [9] "Berliner Freifunk-Community," <http://olsrexperiment.de/>
- [10] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network," Proc. 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp.80–91, 2002.
- [11] X. Li, P. Wan, Y. Wang, and C. Yi, "Fault tolerant deployment and topology control in wireless networks," Proc. 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing, Maryland, USA, 2003.
- [12] C. Bettstetter, M. Gyarmati, and U. Schilcher, "An inhomogeneous spatial node distribution and its stochastic properties," Proc. 10th ACM-IEEE MSWIM 2007, pp.400–404, Chania, Greece, 2007.
- [13] F.A. Onat and I. Stojmenovic, "Generating random graphs for wireless actuator networks," 8th IEEE Symposium on a World of Wireless, Mobile and Multimedia Networks WoWMoM, 2007.
- [14] X. Liu and M. Haenggi, "Toward quasiregular sensor networks: Topology control algorithms for improved energy efficiency," IEEE Trans. Parallel Distrib. Syst., vol.17, no.9, pp.975–986, 2006.
- [15] J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," Proc. IEEE INFOCOM, pp.1312–1321, 2003.
- [16] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," Proc. Workshop on Service Assurance with Partial and Intermittent Resources, pp.239–254, 2004.
- [17] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wagner, "SUMO — Simulation of Urban MObility," Proc. 4th Middle East Symposium on Simulation and Modelling, pp.183–187, 2002.
- [18] K. Fall and K. Varadhan, "The ns2 manual," 2008.
- [19] R.J. Punnoose, P.V. Nikitin, and D.D. Stancil, "Efficient simulation of rician fading within a packet simulator," Proc. Vehicular Technology Conference, pp.764–767, 2000.
- [20] J. Mullen and H. Huang, "Impact of multipath fading in wireless ad hoc networks," Proc. Second ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, 2005.
- [21] M. Takai, J. Martin, and R. Bagrodia, "Effects of wireless physical layer modeling in mobile ad hoc networks," Proc. Second ACM International Symposium on Mobile Ad Hoc Networking and Computing, 2001.
- [22] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing (RFC 3561)," July 2003.
- [23] "'AT&T is now the worst carrier' — Consumer Reports," [money.cnn.com/2010/12/06/technology/consumer\\_reports\\_att/index.htm](http://money.cnn.com/2010/12/06/technology/consumer_reports_att/index.htm)



**Bratislav Milic** is a senior researcher at Humboldt University in Berlin. He finished his undergraduate studies at School of Electrical Engineering at Belgrade University, Serbia. He received his Ph.D. in Computer Science at Humboldt University in Berlin in 2009. His research interest is in availability modelling, dependable computing, mobile computing and communication, and wireless multi-hop networks. He is involved in several industrial projects with partners from logistics and manufacturing sectors.

He has published numerous journal and conference publications, and authored four book chapters.



**Miroslaw Malek** is professor and holder of Chair in Computer Architecture and Communication at the Department of Computer Science at Humboldt University in Berlin. His research interests focus on dependable, embedded and distributed systems including modelling, failure prediction, dependable architectures and service availability. He authored and co-authored over 200 publications including six books, has supervised almost 30 Ph.D. dissertations (ten of his students are professors) and founded, organized

and coorganized numerous workshops and conferences. He serves on editorial boards of several journals and is consultant to government and companies on technical and strategic issues in information technology. Malek received his PhD in Computer Science from the Technical University of Wroclaw in Poland, spent 17 years as professor at the University of Texas at Austin and was also, among others, visiting professor at Stanford, Università di Roma "La Sapienza", Politecnico di Milano, Keio University, Technical University in Vienna, New York University, Chinese University of Hong Kong, and guest researcher at Bell Laboratories and IBM T.J. Watson Research Center.