

NPART - Node Placement Algorithm for Realistic Topologies in Wireless Multihop Network Simulation

Bratislav Milic and Miroslaw Malek
Institute for Informatics
Humboldt University, Berlin
{milic, malek}@informatik.hu-berlin.de

ABSTRACT

Despite a considerable number of topology generation algorithms for simulation of wireless multihop networks it is difficult to find one with output similar to real networks.

We propose NPART – a Node Placement Algorithm for Realistic Topologies whose output topologies resemble networks encountered in reality. The algorithm is flexible since it is sufficient to provide it with different input data to obtain different topologies. To demonstrate its quality, we compare topologies generated by NPART algorithm with our measurements from open wireless multihop networks in Berlin and Leipzig. Compared with real topologies, the generated topologies have almost identical node degree distribution, similar number of cut-edges and vertices, and distribution of component sizes after bridge removal.

The importance of node placement algorithm is demonstrated by comparing ns-2 simulation results for grid and uniform node placement with NPART generated topologies. Simulation results show that node placement model plays as important role in simulation outcome as the accuracy of wireless signal propagation model.

Categories and Subject Descriptors

C.2.1 [Network Architecture and Design]: Wireless communication; C.2.1 [Network Architecture and Design]: Network topology

General Terms

Algorithms, measurements

Keywords

Node placement, topology generation, simulation, wireless multihop networks

1. INTRODUCTION

Wireless multihop networks (WMNs) are used for various purposes such as Internet access, sensing and logistics appli-

cations. Each of usage scenarios requires a dedicated protocol in order to offer sufficient quality and performance to end users. Verification of a developed protocol is necessary and, usually, it is performed by simulators due to their low operating cost and fast setup. Quality of simulation directly depends on simulation model. The model of WMNs is complex and it consists of five sub-models: node model describes hardware and software of a node; deployment and mobility models provide node positions and their movement patterns; radio model describes the characteristics of the radio used by the node; wireless signal propagation model deals with attenuation and characteristics of wireless channel; traffic models define traffic patterns in a network.

Some of sub-models are based on real data measurements (e.g., wireless signal propagation [1], traffic models [21]). However, some of them are synthetic and somewhat arbitrary like the topology generators/node placement models.

In [10], [11] and [13] we have presented results of measurements from wireless mesh networks in Berlin and Leipzig. The analysis based on 1500 topological samples from each of networks showed that properties of artificial topologies are substantially different from properties observed in reality. Table 1 illustrates the differences between sample simulation setups and characteristics of their topologies and characteristics captured through measurement. The predominant uniform placement model has issues in reproducing reality: in order to produce connected topologies, node density must be increased. This may create higher channel contention in simulation studies than it is to expect in protocol deployments. Higher node density also reduces the number of bridges and articulation points in a network. They are significant since their failure disconnects the network.

The main approach in natural sciences such as physics is to observe reality and create a model that reflects it. Most of the existing node placement models do not follow this approach: they have not been inspired by reality nor verified by measurements. In order to correct this fundamental issue, we propose algorithm that produces realistic topologies.

The paper is organized as follows: Section 2 provides definitions, Section 3 reviews existing topology generators for WMN simulations. In Section 4, we present our node placement algorithm. Section 5 shows that the developed algorithm generates topologies whose properties reflect the characteristics of real networks.

2. DEFINITIONS

We use undirected graphs to model communication in WMNs. In a communication graph, network nodes are represented as vertices. If a node p is able to communicate with

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIMUTools 2009, Rome, Italy

Copyright 2009 ICST 978-963-9799-45-5.

	n	Area	R	\bar{d}	\bar{b}	\bar{ap}	\bar{c}
Berlin	315	/	/	4.02	93.59	75.93	1
Leipzig	586	/	/	4.35	101.39	93.32	1
Ngai [16]	100	200x200	40	10.34	0.24	0.43	1.09
Zhu [24]	100	1500	250	7.54	1.54	2.38	1.39
	150	x1500		11.19	0.35	0.49	1.08
Wu [23]	100	100x100	25	15.49	0.07	0.09	1
	300			46.67	0	0	1

Table 1: Comparison of sample simulation setups from literature and networks from Berlin and Leipzig. R is communication radius, n number of nodes, \bar{d} is the average node degree, \bar{b} and \bar{ap} are average number of bridges and articulation points. The average number of network partitions is \bar{c} .

a node q there exists an edge pq in the graph. The following terms and definitions are used in the paper [22].

The degree of a vertex v is the number of edges incident on v . A pendant vertex is a vertex of degree 1. Maximally-connected subgraph is a graph such that there exists a path between any pair of nodes (p, q) . Components of a graph G are its maximally-connected subgraphs. A bridge in a graph (cut-edge) is an edge whose deletion increases the number of components. An articulation point in a graph (cut-vertex) is a vertex whose deletion increases the number of components in the graph. Given an undirected graph, a degree sequence is a monotonic nonincreasing sequence of degrees of its vertices. A degree set is a set of integers that make up a degree sequence. The frequency of an event i is the number n_i of times the event occurred in the 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.

3. RELATED WORK

In WMN research, the most frequent node placement models are chain, grid and uniform. The chain placement model places nodes equidistantly on a line. In the grid placement model (also known as the mesh placement), nodes are located at intersections of a rectangular grid. Number of edges in grid depends on node communication radius, shape and size of cell. In the literature these parameters are typically chosen so that all nodes that are not on the grid border have degree of four. 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 $p_{uniform} = \frac{n}{|A|}$. If the placement area is rectangular $((0, x_{max}), (0, y_{max}))$, this is typically achieved by sampling the x coordinate of a vertex from $U(0, x_{max})$ and y from $U(0, y_{max})$.

It is particularly difficult to create connected low-density topologies with the existing random placement models. To ensure connectivity of a simulated network, node density is increased. Bettstetter shows in [2] that for the uniform placement model, nodes must have average degree of 10.8 to obtain a network that is connected with probability of 0.99. Li et al. [8] provide even higher estimation – they claim that obtaining of the same connectivity probability requires 13.78 neighbor nodes on the average.

Such dense networks have strong impact on simulation results since network diameter is significantly reduced, numerous independent paths between each pair of nodes exist, failure of individual nodes and losses on individual commu-

nication links do not impact the connectivity nor the functionality of the network.

The need for improved node placement model in WMNs has been already noticed and several non-homogeneous models have been proposed. Bettstetter et al. [3] place nodes in accordance with the uniform process and then apply thinning to it. The thinning operation removes nodes from a network that have less than k neighbors within radius t_r . Parameters k and t_r are specified by the user and they control the level of inhomogeneity of the topology. However, authors of [3] discuss only the node placement, ignoring the properties of topologies that are obtained from it.

Liu and Haenggi [9] propose two quasiregular placement models. In the first, vertex coordinates are Gaussian distributed with the mean given by regular grid points. The second 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. In [5], a two-dimensional Gaussian distribution is used to determine location of sensor nodes. This idea can be extended so that there are multiple vertex focal points, each of the focal points having a non-uniform distribution attached to it.

Onat and Stojmenovic [17] propose considerably different approach. They have developed several algorithms that create connected topologies with high probability and allow user to choose the average node degree. The shape of topologies primarily depends on the selected algorithm. The algorithms do not guarantee connectivity of their output - if the end result is not connected, algorithm is restarted. Their analysis focuses on algorithm complexity and probability that created graph is connected. The probability density function for node degree for each of algorithms is presented and the differences among placement topologies created by different algorithms is informally (visually) demonstrated.

4. NPART – A NODE PLACEMENT ALGORITHM FOR REALISTIC TOPOLOGIES

Our goal is to develop a node placement / topology generating algorithm that is:

- **Flexible** – it is capable to create easily more than one node distribution type.
- **Realistic** – if algorithm receives input based on measurements from a real network, the algorithm-produced topologies should have properties similar to the original network properties.
- **Random** – the algorithm does not merely re-create a sampled topology from measured node locations, wireless device parameters (power, receiving threshold), signal to noise ratio. It is capable to create new, random topologies while preserving the properties of flexibility and realism.

The starting point for NPART development is taken from [11]. In [11], the following sociological and technological reasons that shape topologies of real networks were identified:

- It is more likely that new participants join the network in areas where connectivity is already good.
- A participant in the network expects to have at least a single communication link to the remainder of the network.

```

place nodes(nodes  $n$ , communication radius  $R$ ,
number of candidates to evaluate per iteration  $retries$ ):
   $placedNodes$  = place first node arbitrarily at  $(x,y)$ 
   $minX = maxX = x$ 
   $minY = maxY = y$ 
  repeat
     $minMetric = \infty$ 
    repeat
      repeat
         $x\text{-coordinate} = U(minX - R, maxX + R)$ 
         $y\text{-coordinate} = U(minY - R, maxY + R)$ 
        create node  $candidateN$  from coordinates
      until ( $candidateN \cup placedNodes$  is connected)
       $m = \text{apply metric on } placedNodes \cup candidateN$ 
      if ( $m < minMetric$ )
         $bestCandidate = candidateN$ 
         $minMetric = m$ 
      endif
    until ( $retries$  different candidates are evaluated)
    if required, update  $minX, maxX, minY, maxY$ 
    based on  $bestCandidate$  location
     $placedNodes = placedNodes \cup bestCandidate$ 
  until (all  $n$  nodes are placed)

```

Figure 1: NPART pseudo code description.

- A pendant node may become a seed for a new, larger and well-connected subnetwork.
- It is the network that specifies the area it occupies, not the other way around. Instead of defining the node placement area like in most of the existing placement algorithms, the network should be allowed to grow.

4.1 Algorithm description

The algorithm is presented in Figure 1. As input parameters, algorithm accepts n , the number of vertices to be placed and communication radius R . The algorithm calculates topologies based on the path-loss model in which nodes are able to communicate if their euclidean distance is less or equal to the communication radius R . The user should specify the additional propagation models in the simulation (shadowing and Rayleigh fading [1]) to create realistic simulation results.

In the first iteration of the algorithm the first vertex is placed at an arbitrary point (x,y) in two-dimensional space. The variables $minX$ and $maxX$ are initialized to x , $minY$ and $maxY$ to y . Values of these variables from iteration I_k are used to determine the placement area of nodes in the next iteration: in iteration I_{k+1} , x coordinate of candidate nodes is uniformly sampled from $(minX - R, maxX + R)$, y coordinate is chosen from $(minY - R, maxY + R)$. This enables the network to grow, without need to predefine its geographical size.

It is possible that a vertex placed in rectangle $((minX - R, minY - R), (maxX + R, maxY + R))$ in iteration I_k is not connected to topology from iteration I_{k-1} . For example, let the nodes A, B, and C be placed in the starting three iterations (Figure 2). The candidate vertex 0 in iteration I_4 is disconnected from nodes placed in I_3 . Such vertices are ignored and a new candidate vertex is generated. This condition ensures connectivity of produced topology (functionality is implemented in the innermost loop of the algorithm in Figure 1).

If the candidate vertex is connected with the topology produced in previous algorithm iteration (such are the candidate vertices 1, 2 and 3), a user-defined metric is applied to it. Section 4.2 describes four metrics that we have implemented and tested. If the candidate vertex has lower metric

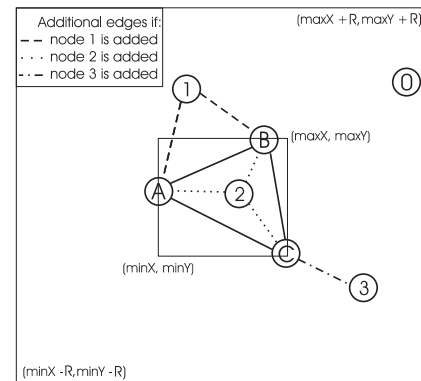


Figure 2: Placement area and candidate nodes.

than previous candidate vertices, it is considered as the best candidate and its data is preserved (variable $bestCandidate$) and the minimal metric value is updated (variable $minMetric$).

$retries$ vertex candidates are evaluated in each algorithm iteration and only the best candidate is added to the topology at the end of the iteration. If necessary, $minX$, $maxX$, $minY$, $maxY$ variables are updated. For example, if candidate vertex 3 is the best candidate out of three candidate vertices in Figure 2, variables $maxX$ and $minY$ must be updated. Number of evaluated vertex candidates per algorithm iteration $retries$ is parameter of the algorithm, and as the number of evaluated candidates grows, the probability that the produced topology is closer to the predefined goal is increased.

After placement of all n nodes their locations can be translated so that they are in rectangle $((0, 0), (maxX - minX, |maxY - minY|))$. This step is optional and does not influence the functionality of the algorithm.

4.2 Quality metrics

A metric that evaluates quality of candidate vertices is as important as the algorithm itself. Inappropriate metric results in unsatisfactory topologies. Unfortunately, there does not exist a universal metric. User must define a metric and perform tests to check whether the algorithm and the metric produce desired topologies.

Our goal are topologies that have properties observed in real, user-initiated networks. In process of metric input selection, it became obvious from experience with the existing placement models that generic parameters such as the average node degree do not capture sufficient level of detail. Realistic topologies can be produced only with more detailed data that originates from measurements.

Capturing spatial node distribution and link quality metrics (e.g., signal to noise ratio, bit error ratio, packet loss probability) in a real network would be an excellent input for the vertex placement algorithm. However, antennas' gain cannot be automatically collected, signal propagation environment is heterogenous and its impact on link quality cannot be accurately measured with off-the-shelf components that are commonly used in user initiated networks. Additionally, due to privacy concerns not all participants in such a network are willing to disclose their geographical locations, and even if they would, dissemination of such data for scientific purposes would be coupled with serious legal

limitations. In rare cases when it is possible to take samples from user initiated networks, typically only the topological information is available, without node location data.

We have implemented several metrics that use node degree frequencies. We have chosen the degree frequency as algorithm's input because it provides a compromise between detail level, feasibility of sampling, and data anonymity. Degree distribution can be used to efficiently describe networks [15]. It is easily extracted from real networks, regardless of the routing protocol type (proactive or reactive) and it does not contain information of individual users. In proactive protocols, it is trivial to calculate it. In case of reactive routing protocols where no global topology view exists, node degrees can be easily obtained assuming that nodes in the network are cooperative: each node samples its degree and shares it with the central repository. Small errors in sampling (e.g., a node erroneously reports its degree) are hidden by the larger set of correct data.

As input for *distance* and *adaptive* metrics, we calculate relative node degree frequency from degree sequences from all samples that were taken from Berlin's and Leipzig's networks. The relative node degree frequency of real network is multiplied by number of nodes that should be placed by the algorithm, creating absolute vertex degree frequency for target topology *target*. For each candidate vertex that is evaluated, absolute degree frequency *candidate* of topology that it creates with already placed nodes is calculated and compared with the target frequency.

The simplest, *distance* metric is a variation of the Manhattan metric:

$$\sum_{degrees}^d (1_{target_d - candidate_d > 0} \cdot (target_d - candidate_d) + 1_{target_d - candidate_d < 0} \cdot p \cdot (candidate_d - target_d)) \quad (1)$$

where $1_{A(x)}$ is the indicator function, returning one if $x \in A$, zero otherwise. The metric sums difference between proposed and target vertex frequency if difference is positive. If the difference is negative (candidate topology has more nodes of certain degree than the target topology), absolute value of difference is multiplied with penalty factor p . The penalty factor reflects user's tolerance for exceeding of degrees: increased tolerance is modeled by smaller values for the factor p . If $p = 1$, the *distance* metric is identical to the Manhattan metric [7].

The weakness of the *distance* metric is in its impossibility to detect stronger need for creation of vertices with certain degree. Some degrees are more frequent in the target degree frequency so topologies that produce them should be rewarded. For instance, if the algorithm should create 20 vertices with degree two and three vertices with degree four, the metric should give greater reward (smaller metric value) to topologies that increases number of vertices with degree two in early iterations of the algorithm. The *adaptive* metric resolves this issue:

$$\sum_{degrees}^d (1_{target_d - candidate_d > 0} \cdot (target_d - candidate_d) \cdot w_d + 1_{target_d - candidate_d < 0} \cdot p \cdot (candidate_d - target_d)) \quad (2)$$

Degrees	1	2	3	4	5	Dist. metric	Adapt. metric
Abs. Target degree frequency	2	5	3	2	1	0	0
Abs. Placed degree frequency	0	3	0	0	0	/	/
Weights w_d	0.2	0.2	0.3	0.2	0.1	/	/
Candidate 1	0	2	2	0	0	9	1.8
Candidate 2	0	0	4	0	0	15	6.9
Candidate 3	1	2	1	0	0	9	1.9

Table 2: Example of Distance and Adaptive metric calculation for node candidates in Figure 2. Parameter p is set to five, the number of *retries* is three.

and weight $w_d = \frac{|target_d - placed_d|}{\sum_{degrees}^d |target_d - placed_d|}$, where *placed* is the absolute degree frequency of vertices that form the topology from the previous iteration.

Table 2 illustrates the application of *distance* and *adaptive* metrics to the example from Figure 2. Nodes A, B and C are already placed. Topology obtained in I_3 defines the weights w_d for the I_4 . In fourth iteration three candidate vertices are successively evaluated.

For each candidate vertex we calculate metric values by Equations 1 and 2. The *distance* metric calculates equal value for candidate vertices 1 and 3, so either of them can be selected as the best candidate vertex. Adaptive metric correctly chooses candidate vertex 1 as better, since it satisfies greater need to create node of degree three (after I_3 , three more nodes with degree three are required) than to create node of degree one like the candidate vertex 3 (after I_3 two more nodes of degree one are needed to reach the absolute target degree frequency).

In [12] we have compared the behavior of algorithm, used with the *distance* and *adaptive* metrics, for different combinations of parameters p and *retries*. Input were degree frequencies from Freifunk Berlin and Freifunk Leipzig networks. As quality measure of algorithm we applied the Manhattan metric [7] between targeted and produced absolute degree frequencies. It was shown that the topologies produced with the *adaptive* metric are considerably better than the topologies produced with the *distance* metric. Based on this evaluation, we conclude that *adaptive* metric is better, and topologies produced by it will be analyzed in more detail in Section 5.

It is possible to extract additional degree data from topologies. Let us observe relative degree frequency of neighbors of a node p under condition that degree of p is k . Figure 3 shows that the conditional relative frequencies differ considerably among themselves and to the joint relative frequency.

The metric *secondary - Distribution* uses these differences. First it calculates set of conditional relative degree frequencies for candidate topology and then compares them (using Manhattan metric) with the target conditional relative degree frequency. The *combined* metric is a linear combination of *adaptive* and *secondary* metrics. It is possible to vary the penalty factor p in the *adaptive* metric and weight s for the *secondary - Distribution* metric.

The pseudo code of all described metrics can be found in [12].

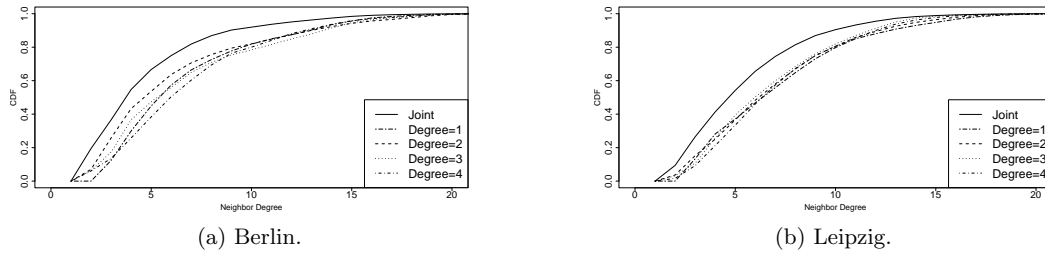


Figure 3: Conditional degree distributions.

5. EVALUATION OF CREATED TOPOLOGIES

This section compares properties of topologies produced by the NPART algorithm and presented metrics with properties of real networks. We demonstrate the quality of the algorithm and test different combinations of algorithm parameters to find the one that produces best results.

The data that is used to provide input degree distributions for the algorithm and for later comparison is taken from Berlin and Leipzig networks. Sampling methodology, detailed analysis and comparison with synthetic placement models are presented in [13]. There is a small change in datasets if compared with [11] and [13]: only the main partition (largest maximally connected subnetwork) is considered for algorithm's degree data input and later result comparison. The main partition in Berlin has 275 and in Leipzig 346 nodes on the average.

To illustrate the improvements brought by the NPART, it is also compared with the uniform placement model (explained in Section 3). Parameters of the uniform placement algorithm are chosen to create topologies with the average node degree of six. The average node degree is substantially lower than proposed in [2] and [8] for networks connected with high probability so it is possible that such a graph is partitioned. Increasing the average node degree above six improves connectivity but creates even greater discrepancy with measurement results (e.g., bridges do not exist in generated topologies), while decreasing it creates highly partitioned graphs. Since the size of Berlin's and Leipzig's main partition differs there are also two uniform placement scenarios, with 275 and 346 nodes. The data for comparison is collected in 500 executions of each scenario.

The NPART is also run with two basic setups: 275 vertices and degree data input from Berlin's network (NPART/Berlin), and 346 vertices and degree data input from Leipzig's network (NPART/Leipzig). The parameter *retry* is set to 150 while parameters for penalty p and secondary metric weight are varied to take values from set $\{0, 1, 5\} \times \{0, 1, 5\}$. Not all results are presented since some parameter combinations do not create reasonable results: as soon as secondary weight s is higher than the penalty p , the algorithm becomes unstable and creates almost fully connected graphs. It is shown later that *secondary* metric is excellent for refinement of the *adaptive* metric but it should not be used on its own.

5.1 Properties of generated topologies

Figure 4 informally illustrates the differences between real topology from Berlin's network, a topology created by the uniform placement model, and an NPART 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 bridges, both on network outskirts and in its central parts. The NPART topology follows these characteristics more accurately.

Although visual representation of topologies in Figure 4 gives a valuable insight in shape and characteristics of generated topologies, such informal comparison is not sufficient. For formal comparison of measured and generated topologies we have selected four topological metrics: degree distribution, bridge and articulation point count and relative component size after bridge removal. These topological metrics directly influence properties of protocols that are simulated. The node degree distribution is correlated to the congestion on the wireless channel and probability of packet loss. Bridges are only communication links between network components. If these components are of considerable size, bridges that connect them tend to get congested, reducing the available throughput per flow and increasing packet latency, thus reducing quality of services deployed in a network. So, in addition to bridge count it is important to capture the size of components that are connected by bridges. Articulation points are gateways between different network components so they may get easily overloaded as well. Also, their removal disconnects a network. Their occurrence is of particular importance in sensor networks, where nodes have limited energy source and processing capacity. Since articulation points tend to route more traffic than other nodes in the network they are more prone to energy exhaustion and critical failure that partitions the sensor network. Due to limited memory and processing power, packet buffers at a node are also more prone to overloading, creating packet drops.

For each metric, we compare distribution of the metric from the measurement with distributions of NPART-created topologies and of the uniform placement topology. We consider that generated topology is better if it closely resembles the properties observed in reality.

Figure 5 shows the vertex degree probability mass function (PMF). As it can be seen, for all parameter combinations, the degree distribution of topologies created by NPART precisely follows the distribution of measurements. NPART adapts with ease to both Berlin and Leipzig distributions. The uniform distribution has its own shape that is considerably different from real distributions. Also, it has zero-degree nodes, indicating existence of partitioned samples.

The bridge to edge ratio (Figure 6) and articulation point count (Figure 7) show that NPART topologies follow the

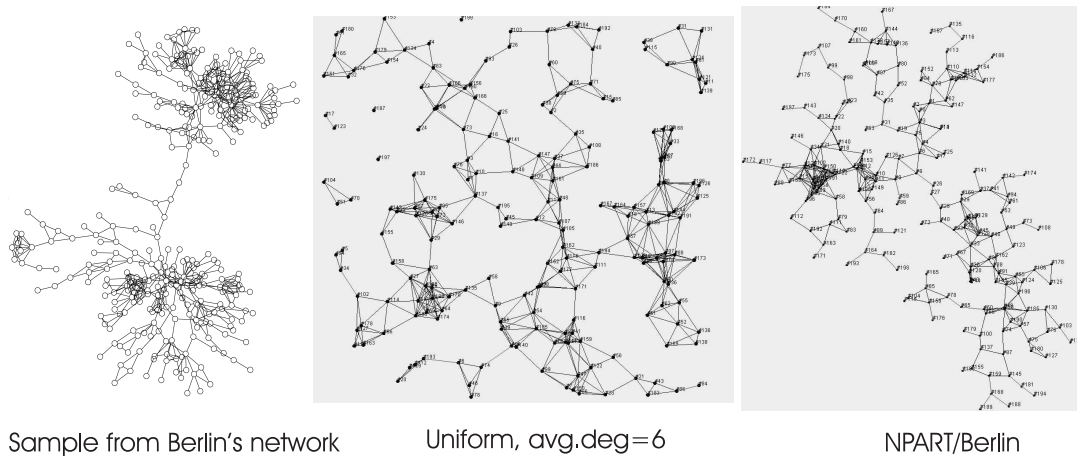


Figure 4: Visual comparison of a measured topology, a topology created by uniform node placement model and a NPART-produced topology.

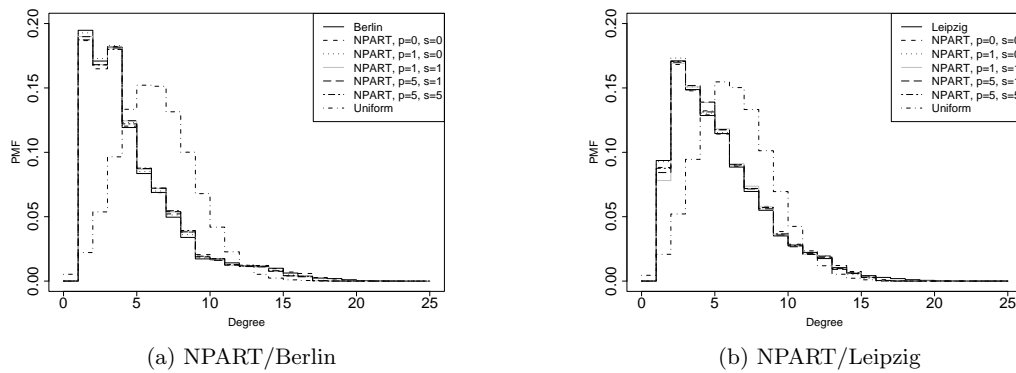


Figure 5: Comparison of node degree distributions.

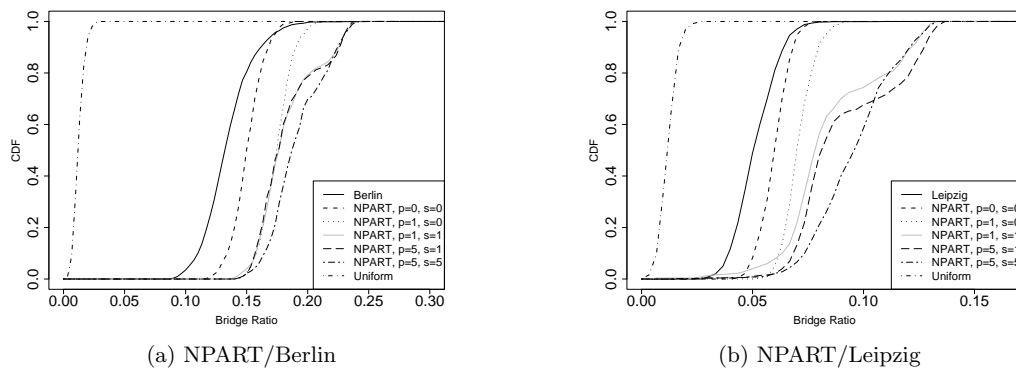


Figure 6: Cumulative distributions of bridge to edge ratio for real samples and created topologies.

properties of real networks. However, the proposed algorithm creates slightly more bridges and articulation points than it should. The uniform placement model is unable to

adapt nor to represent the reality: its topologies have less than 1% of bridges and few articulation points.

Figure 8 shows the cumulative distribution of relative com-

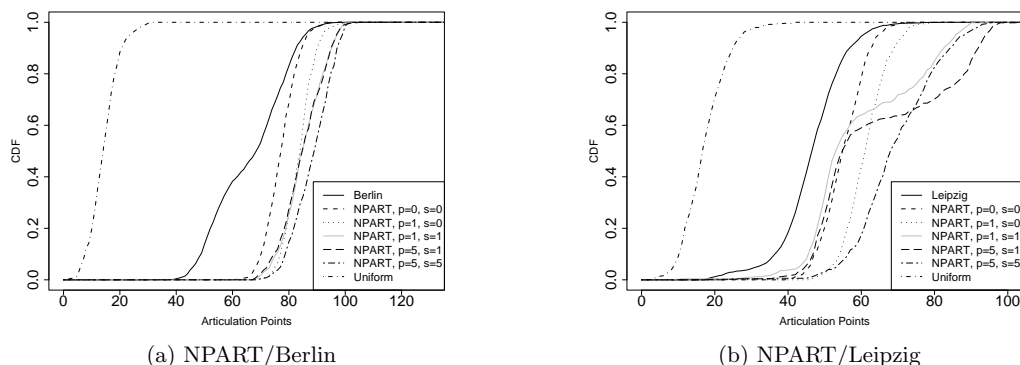


Figure 7: Cumulative distributions of articulation point count for real samples and created topologies.

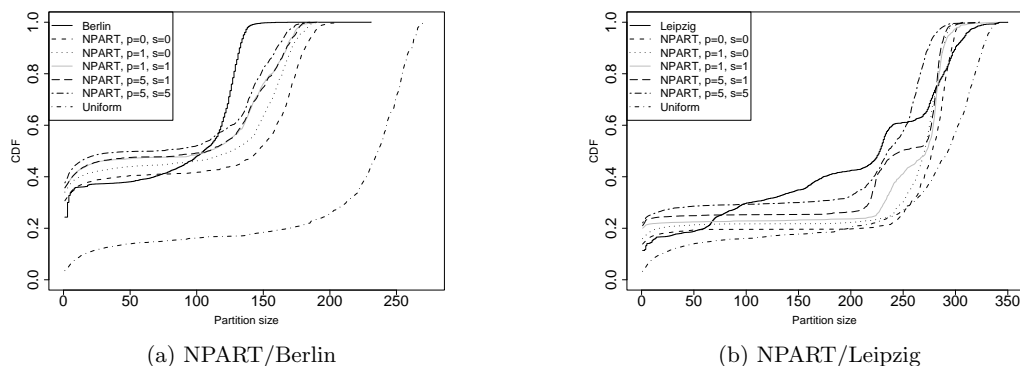


Figure 8: Cumulative distribution of weighted network components obtained by removing of all bridges.

ponent size obtained by removal of bridges. The majority of components are small (less than five nodes) and distribution directly obtained from component size would not provide much information – almost whole distribution weight would be concentrated between one and five. To offset this effect, each component is weighted by its size, relatively to the network size: $C_{rel} = \frac{C_{count} \cdot |C|}{n}$. For instance, in a graph with 100 nodes, after all bridges are removed, ten components of three vertices exist. The relative component size is: $\frac{3 \cdot 10}{100} = 0.3$: 30% of vertices are in three-vertex components.

NPART is again considerably better than the uniform placement model, in particular for the Berlin’s network. Topologies produced with the assistance of *secondary* metric have distributions more aligned with real measurements than samples produced by the *adaptive* metric, both for Leipzig and Berlin distributions.

5.2 Issues of simplified propagation modeling

Although considerably better than existing topology generators, the algorithm can be further improved since it creates more bridges and articulation points than it should.

We have carefully investigated the input data and original topology samples. It was revealed that some nodes with high degree have several pendant nodes attached to them. In reality it is possible due to the phenomenon of correlated shadowing. The correlated shadowing on wireless

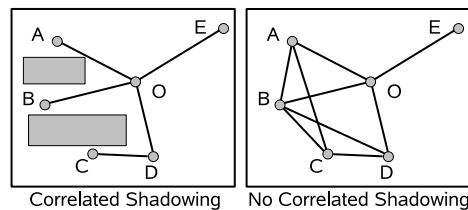
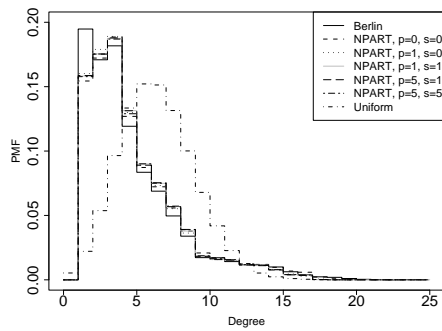


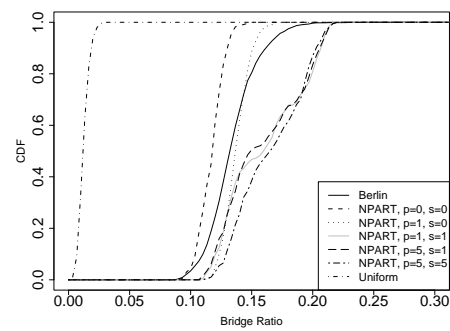
Figure 9: Impact of correlated shadowing.

channel may cause that nodes which are physically close cannot communicate (as shown in the left part of Figure 9, nodes A and B cannot communicate despite rather small distance between them). Correlated shadowing model exists for single-wireless-hop analysis [6] but it is not supported in discrete event simulators for multihop networks. The propagation models supported by simulators create links if internode distance is small (as shown in right part of the Figure 9). As the result, there are less pendant nodes in simulator than in reality for identical placement of nodes.

If high number of pendants is requested from placement generator, it cannot fulfil this request without affecting other characteristics of generated topology: it creates requested

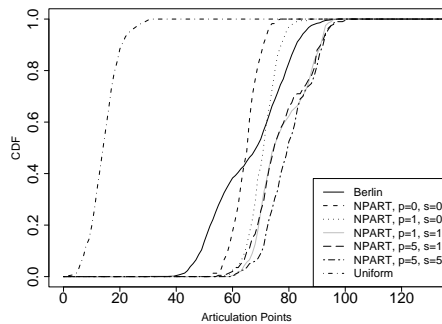


(a) Node degree distribution (NPART/Berlin)

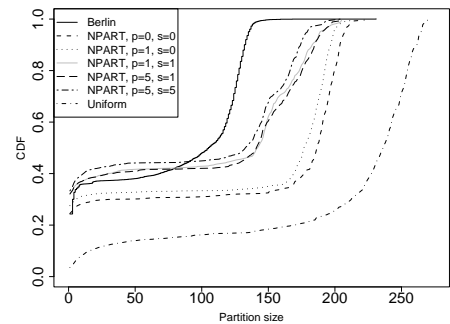


(b) Cumulative distribution of bridge to edge ratio (NPART/Berlin)

Figure 10: Node degree and bridge to edge ratio distributions after reduction of pendant node count by 20%.



(a) Cumulative distribution of articulation point count (NPART/Berlin)



(b) Cumulative distribution of weighted network components (NPART/Berlin)

Figure 11: Cumulative distributions of articulation point count and weighted network components after reduction of pendant node count by 20%.

number pendant nodes, but also increases number of bridges and articulation points more than it should.

To overcome this issue, we provide option in our tool to a user, so that he/she may reduce the number of generated pendant nodes. The user has freedom to decide which topology characteristic is more important for him/her: accurate degree distribution, or better fit with bridge and articulation point distributions. We have performed numerous tests and empirically found that the 20% reduction ratio of pendant node count provides good results.

Due to space limitations, we present results only for NPART/Berlin samples. NPART/Leipzig exhibits similar behavior [12]. The degree distribution (Figure 10(a)) follows closely the real distribution, except of course for nodes of degree 1. Bridge share and articulation point count are reduced as it can be seen in Figures 10(b) and 11(a). It can be seen that reduction in pendant node count results in NPART topologies that have characteristics which are closer to real distributions than topologies from Figures 6 and 7.

The relative component size distribution in Figure 11(b) retains good fit with reality as for the original distribution. It also demonstrates the importance of *secondary* metric: in Figure 11(b) the topologies created without the *secondary*

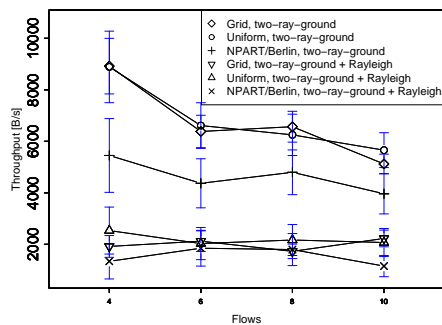
metric have worse alignment with reality than in Figure 8(a), while topologies that have used the secondary metric remain as good as they were.

5.3 ns-2 Simulation results

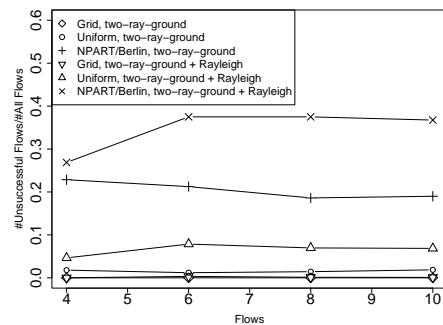
This section demonstrates that choice of node placement algorithm considerably impacts simulation results. For this purpose, we have used ns2 simulator [4], version 2.29 with Rayleigh-Ricean fading extension [19]. Nominal communication range of nodes is set to 250m.

There are six distinct simulation setups:

- Grid node placement and two-ray-ground propagation
- Grid node placement and Rayleigh propagation
- Uniform node placement and two-ray-ground propagation
- Uniform node placement and Rayleigh propagation
- NPART/Berlin node placement and two-ray-ground propagation model
- NPART/Berlin node placement and Rayleigh propagation model



(a) The average throughput with 95% confidence intervals.



(b) Ratio of unsuccessful flows to total number of flows.

Figure 12: ns-2 simulation results for different topology and signal propagation models.

Grid consists of 272 nodes, put in 16 rows and 17 columns. Internode distance is 200m. For uniform node placement, 275 nodes are placed in 2700x2700m area, producing average node degree of 7.4. Such parameter selection creates network that is not too dense but connected with rather high probability. NPART algorithm generates 275-node topologies, using data input from Berlin's network and combined metric ($s=1$, $p=5$). Routing protocol is AODV [18].

The number of TCP flows is varied (4,6,8,10) in each of simulation setups to test protocol's behavior under different network loads. We observe throughput and number of flows that were unable to start (unsuccessful flows). In order to avoid counting of unsuccessful flows that are caused by partitioned network, uniform placement topologies are tested for connectivity before they are accepted for simulation. Simulations are performed on 50 different topologies except for grid where all topologies are identical. TCP flows are created between randomly selected pairs of nodes.

Warm up phase is set to 30s and simulation is executed for 250s. Warm up phase of 30s is sufficient because AODV is a reactive routing protocol with aggressive purging of inactive routes. Prolonging the warm up time cannot change results: nodes that are required to maintain the local connectivity have enough time to execute neighbor detection process (Section 6.10 in [18]) because of frequent heartbeats. Inactive routes are deleted after 15s (Section 10 in [18]) so even if we prolong the warm up phase, the inactive routes will be removed from routing tables. Runtime of 250s is sufficient for TCP (that is used as transport protocol for FTP traffic) to leave initial slow-start phase: the total simulation time is much longer than the packet round-trip times in networks that were simulated. Increasing this value to larger values does not bring considerable changes: we have performed a test, increasing simulation run time to 500s for uniform placement scenario with 10 flows under two-ray-ground propagation model. Average throughput with 95% confidence intervals was 5708.2 [4932.3, 6484.0] which is very close to 5657.1429 [4980.2, 6334.1] obtained for 250s runs.

As expected and already shown in research studies [14] [20], there exists a considerable difference in simulation results between over-simplistic, over-optimistic two-ray-ground propagation model and realistic Rayleigh model: obtained throughput is considerably higher and number of unsuccessful flows is considerably lower for the two-ray-ground model.

It is important to observe that for the same propagation model there also exists considerable difference in simulation results between grid, uniform node placement and NPART produced topologies: in NPART topologies throughput is lower than in both synthetic placements. This is particularly notable if two-ray-ground model is exclusively used. Stochastic variations of the channel introduced by the Rayleigh fading reduce the difference between placement models, but NPART topologies still have smaller throughput.

Simulation of NPART/Berlin topologies with realistic wireless signal propagation model results in particularly high ratio of unsuccessful flows – although a communication path between pair of nodes exists (NPART guarantees connected topologies), AODV is unable to find it. In [10] we have predicted such behavior of AODV with a different methodology. The uniform placement model creates fewer unsuccessful flows than NPART/Berlin while in grid structured network only one of all simulated flows was unsuccessful, even with Rayleigh fading on the wireless channel.

6. CONCLUSIONS

We have proposed, developed and evaluated NPART - Node Placement Algorithm for Realistic Topologies. The algorithm provides input to simulation of static wireless multihop networks. It is flexible since it is sufficient that user defines a metric in accordance with his/her needs or provides different input data and algorithm creates topologies with different properties.

The algorithm guarantees the connectivity of generated topologies. In wireless mesh or sensor networks, this is a common property. In some application scenarios, such as Disruption Tolerant Networks, complete network connectivity is rarely encountered. For such scenarios, NPART may be used to produce connected subnetworks of a globally disconnected network. We plan to extend NPART so that it can fully support generation of disconnected topologies.

We have evaluated the algorithm and shown that with appropriate metric and parameter selection, NPART-created topologies resemble those from practice. Stochastic analysis is used to compare NPART-produced topologies with ubiquitous uniform placement algorithm and real networks in Berlin and Leipzig. The properties of interest are node degree distribution, bridge to edge ratio, articulation point count and size of graph components after bridge removal.

The properties of topologies produced by our algorithm fit closely the measurements of real networks, while uniform placement model has its own properties that are far from reality.

The importance of accurate signal propagation models is known and shown in numerous publications but the impact of topology generators on simulation outcome is often overlooked. We have compared the average throughput and the number of unsuccessful flows in topologies produced by the grid, uniform placement model and our proposed NPART algorithm under simplistic two-ray-ground and realistic Rayleigh signal propagation models. Simulation results show that node placement model plays as important role in simulation outcome as the wireless signal propagation model.

A tool based on NPART is developed and it can be downloaded from project's webpage¹. User can specify type of input data (Berlin or Leipzig data sets are built into the tool), number of retries, metric parameters, and output format. The tool currently supports ns2 and .dot output formats. We have also developed an importer of ns2 topologies for Jist/SWANS simulator. We do hope that our results will encourage the research community to use realistic modeling in all segments of simulation setup thus increasing the simulation quality and narrowing the gap between simulation and reality.

7. REFERENCES

- [1] A. Aguiar and J. Gross. Wireless Channel Models. Technical Report TKN-03-007, TU Berlin, 2003.
- [2] C. Bettstetter. On the minimum node degree and connectivity of a wireless multihop network. In *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Lausanne, Switzerland, 2002.
- [3] C. Bettstetter, M. Gyarmati, and U. Schilcher. An inhomogeneous spatial node distribution and its stochastic properties. In *Proceedings of 10th ACM-IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Chania, Greece, 2007.
- [4] K. Fall and K. Varadhan. The ns2 manual. September, 2008. <http://www.isi.edu/nsnam/ns/ns-documentation.html>.
- [5] L. Fang, W. Du, and P. Ning. A beacon-less location discovery scheme for wireless sensor networks. In *In Proceedings of 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, 2005.
- [6] M. Gudmundson. Correlation model for shadow fading in mobile radio systems. *Electronic Letters*, 27:2145–2146, 1991.
- [7] E. Krause. *Taxicab Geometry: An Adventure in Non-Euclidean Geometry*. Dover, New York, 1986.
- [8] X. Li, P. Wan, Y. Wang, and C. Yi. Fault tolerant deployment and topology control in wireless networks. In *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Maryland, USA, 2003.
- [9] X. Liu and M. Haenggi. Toward quasisregular sensor networks: topology control algorithms for improved energy efficiency. *IEEE Transactions on Parallel and Distributed Systems*, 17:975–986, 2006.
- [10] B. Milic and M. Malek. Adaptation of breadth first search algorithm for cut-edge detection in wireless multihop networks. In *Proceedings of 10th ACM-IEEE Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Chania, Greece, 2007.
- [11] B. Milic and M. Malek. Analyzing large scale real-world wireless multihop network. *IEEE Communication Letters*, 11:580–582, 2007.
- [12] B. Milic and M. Malek. NPART - Node Placement Algorithm for Realistic Topologies in Wireless Multihop Network Simulation. Technical Report 224, HU Berlin, 2008.
- [13] B. Milic and M. Malek. *to appear in : Handbook of Wireless Ad Hoc and Sensor Networks*, chapter Properties of Wireless Multihop Networks in Theory and Practice. SpringerVerlag, 2009.
- [14] J. Mullen and H. Huang. Impact of multipath fading in wireless ad hoc networks. In *Proceedings of The 2nd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, 2005.
- [15] M. Newman, S. Strogatz, and D. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64, 2001.
- [16] E. C. H. Ngai, Y. Zhou, M. R. Lyu, and J. Liu. Reliable reporting of delay-sensitive events in wireless sensor-actuator networks. In *Proceedings of The Third IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Vancouver, Canada, 2006.
- [17] F. A. Onat and I. Stojmenovic. Generating random graphs for wireless actuator networks. In *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM*, 2007.
- [18] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing (RFC 3561). July 2003.
- [19] R. J. Punnoose, P. V. Nikitin, and D. D. Stancil. Efficient simulation of ricean fading within a packet simulator. In *Proceedings of the Vehicular Technology Conference*, 2000.
- [20] M. Takai, J. Martin, and R. Bagrodia. Effects of wireless physical layer modeling in mobile ad hoc networks. In *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, 2001.
- [21] N. Vicari. Models of WWW-Traffic: a Comparison of Pareto and Logarithmic Histogram Models. Technical Report 198, Institute of Computer Science, University of Wuerzburg, 1998.
- [22] D. B. West. *Introduction to Graph Theory*. Prentice Hall, 1996.
- [23] Y. Wu and Y. Li. Construction algorithms for k-connected m-dominating sets in wireless sensor networks. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, 2008.
- [24] S. Zhu, W. Wang, and C. V. Ravishankar. A new power-efficient scheme to deliver time-sensitive data in sensor networks. In *Proceedings of The Third IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Vancouver, Canada, 2006.

¹<http://www.rok.informatik.hu-berlin.de/NPART>