Simulating and Measuring Burnout Robustness of Damaged Mesh Spatial Networks

Mitchell G. B. Johnson and Ken A. Hawick
Computer Science, Massey University, North Shore 102-904, Auckland, New Zealand
email: {m.g.b.johnson, k.a.hawick}@massey.ac.nz
Tel: +64 9 414 0800 Fax: +64 9 441 8181

Abstract
Mesh networks arise in several application areas including social nets; power systems networks, communications systems, and especially in peer-peer networks of highly mobile devices such as sensors or future generation personal devices such as phones or tablet computers. We study the resilience and failure modes of simulated mesh networks as highly central nodes “burnout” and are removed. We use centrality metrics to quantitatively assess network degradation as critical nodes are progressively removed. Network damage is then parameterised in terms of the number of most critical nodes removed. We report experiments realised over many generated networks to extract the systemic trends in Dijkstra distance and node between-ness. We discuss the range within which mesh networks can be safely operated while still maintaining connectivity and integrity.

Key Words
mesh network; damaged network; load burnout; between-ness.

1 Introduction
Mesh networks are becoming widely used in: urban scenarios [25]; mining scenarios [17] with health and safety implications; security and sensor applications including intruder detection networks [6]; as well as in communication networks where availability is also the key concern [4]. Mesh networks typically arise in the context of wireless devices deployed at spatial locations that must each receive and transmit their own data as well as act as a relay for other devices in the structure.

A degree of interactive collaboration between all nodes is therefore necessary, and the usual enforced hierarchical relationships between hubs, router, server nodes and so forth that exist in a fixed conventional network do not necessarily apply. A mesh network can be further complicated if the nodes are mobile. A mobile ad-hoc networks (MANET) can be implemented as a mesh network where the traffic propagation routes are recomputed as individual nodes move around.

It is possible to study wireless mesh networks in specific detail related to the frequencies, band-width allocation, handover latencies and so forth. It is also useful to consider more general ideas related to the graph structural properties of a network. Network capacity, load dynamics and reliability [2] can be studied using a range of theoretical techniques including game theory [5, 9] and tensor algebra [28] as well as graph network theory [29]. In this present paper we focus on practical and experimentally computable graph metrics and use these to study network resilience.

We explore the robustness and reliability [26] of mesh networks by simulating the consequences of nodes failing. We consider various centrality measures [11] including the node degree and the between-ness centrality to rank individual nodes according to their importance and criticality to the network as a whole [12].

We investigate how the bulk properties of a mesh network changes as the most highly ranked nodes are progressively “burned out” or removed. This is a fairly realistic scenario as the most heavily critical nodes might well be the most overloaded in a realistic mesh net, and therefore the most likely to fail.

The between-ness centrality metric [20] is of especial interest and is discussed in Section 2 below. It identifies the most central or important node in a given network - and therefore the node whose failure would have the greatest
impact upon the whole system. Recent work has explored this and other centrality metrics [18] for applications including: social [8, 22] networks; power systems [15]; and communications networks [27].

In this present paper we focus on spatial patterns of mesh nodes that might arise from a peer-to-peer distribution of sensors or of future generation individual mobile phones or other highly portable devices. Figure 1 shows a typical mobile mesh network with a spatial pattern of nodes ranked according to their between-ness. We focus on a peer-oriented network where all nodes have similar localised communication abilities such as range and power. Other work has considered the implications of adding a few long range nodes – such as satellite up-links – and how these “small-world” links change the properties of the network [16].

We restrict ourselves here to nodes that might model wireless sensors or future generation personal mobile devices that are all commodity devices and which have of necessity to form a peer-oriented mesh network. Various approaches are generally used to design wireless networks and in particular techniques to vary the network geometry or topology to achieve best coverage [19]. In the absence of statistically likely spatial location patterns for mobile user/devices, we simply generate uniform random patterns of nodes to experiment with.

Our article is structured as follows: In Section 2 we review some background and terminology for mesh networks. In Section 3 we describe our numerical experiments and simulation method. We present some quantitative results arising from removing highly central network nodes in Section 4. We give a discussion of the implications for various centrality heuristics and their use in studying mesh networks in Section 5 and offer some conclusions and areas for further study in Section 6.

2 Network Notation

The commonly used notation for a mesh network that can be modelled as a graph is to refer to Graph $G$ with a set $V$ of vertices or nodes. There are then $N_V$ nodes and also a set of $E$ connecting edges of which there are $N_E$ individual peer-peer connections.

Centrality metrics rank the mesh network nodes in an order according to which is the most highly connected or critical to the network as a whole. The simplest centrality metric is the in or out degree of a particular node. This is just the number of other peer nodes that it connects to or from. Simple static degrees like this do not necessarily give insights into the wider implications of a particular node failing.

The “between-ness” centrality metric is defined in terms of the node through which the highest number of pathways connecting any two other nodes pass. The between-ness is computed using the shortest path distance between each pair of nodes $(s, t)$; $s \in V, t \in V$. The fraction of the shortest paths that pass through each vertex $v$ is computed and summed over all possible pairs of vertices $(s, t)$. This can then be written as:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}}$$  \hspace{1cm} (1)

where $\sigma_{s,t}$ is total number of shortest paths from $s$ to $t$ and $\sigma_{s,t}(v)$ is the number that pass through $v$. This can be normalised by dividing by number of node pairs not including $v$. This factor is $(n-1)(n-2)$.

Computing shortest-paths for a network is a long-standing problem and there are several algorithms available [7, 13, 24]. Often the choice for networks that are not too large is dominated by the ease of integration with the data structures and the other software tools deployed. The computational complexity for the shortest-paths is $O(N^3)$ using the Floyd-Warshall algorithm [10]. There are other and newer algorithms such as Brandes’ algorithm, which takes $O(NV + NE)$ [3]. In this present work the calculations were done repeatedly as networks were progressively allowed to fail and it was sufficient and easiest to use implementations of the Floyd-Warshall algorithm in our own software.

Mesh networks as we model them here have a spatial location for each node. We present a two-dimensional and three-dimensional spatial map points and for simplicity each node is located at $(x, y, z)$ within a scaled unit cube. This can be treated as a realistic map so that nodes near the edge will inevitably be less connected as “leaf” nodes, than those in the heart of the space. Another artificial construct is to imagine period wrap around boundaries on the unit cube so that topologically the space forms the surface of a torus. This approach – commonly used in physics and statistical mechanical models – allows us to impose a degree of spatial locality and symmetry so that locally each node has much the same (average) environment and connectivity. Comparisons between these two geometries allows us to investigate the role of leaf nodes and ask to what extent this property relates to their between-ness.

The shortest-path distance sometimes known as the Dijkstra distance is an average over all pairs of nodes in a graph. It characterises in a bulk sense the connectedness of a graph in a single positive distance-like quantity. It is possible to weight the distance by actual spatial distances, but it is also common to simply use number of hops or node-node traversals in its unweighted form. We can write this for our graph or network of vertices $v \in V$ so that for each pair of vertices $s \in V, t \in V$ we find the shortest path $p_{s,t}$ that minimises the (weighted or unweighted) path length over all possible paths $P_{s,t}$ for all $s, t \in V$ between vertices $s, t$. We can normalise appropriately so we arrive at a Dijkstra distance $D$ so that:

$$D = \frac{1}{N(N-1)} \sum_{s,t} p_{s,t}$$  \hspace{1cm} (2)

where $N = |V|$ is the number of vertices. Usually we do not count self connections and we must also take
account of the case when the graph fragments and there is no longer a connecting path between a particular pair of nodes. This is normally done by ignoring disconnected pairs, and effectively restricting the sum to pairs within connected components. The transition from a fully connected single component network to a fragmented one thus shows up appropriately as a discontinuity or sudden change in the Dijkstra distance.

3 Simulation Techniques

The networks we experiment with in this present paper are simulated systems, generated according to a model with a single parameter. To simulate the effects of targeted node elimination in the irregular network we randomly generate $N$ points on either a two-dimensional unit square or a three-dimensional unit cube. These points are each stored in an eight byte C++ double and passed to a Node class that stores the position of the Node, its between-ness, radius and a vector of links to other nodes within the radius of it. The Declaration of this class is shown in listing 1

```cpp
class Node{
    double x;
    double y;
    double z;
    double radius;
    double betweenness;
    vector<Link> links;
}
```

Listing 1. The class definition for the Node class

We create a one-dimensional array of Nodes of size $N$ then each points position is determined randomly in space. Once all points are allocated we iterate over them all and calculate which points are connected to other points and store these links in the C++ STL vector type of Links. The simple Link class is defined in Listing 2 contains pointer to the connected Node and the distance to that node forming an undirected graph.

```cpp
class Link{
    Node* dest;
    double distance;
}
```

Listing 2. The class definition for the Link class

When assigning positions in a unit square/cube with a static radius ($R$), higher values of $N$ causes the number of connections per node to increase. This is representative of real world wireless systems where the transmission power of each node is static and the number of connections per node can increase rapidly. We explore the effect of different radii on the between-ness in Figure 8. Once the system is initialised the we run the Floyd-Warshall algorithm to find the paths from each node to any other node and store the paths.

We then compute the between-ness using the stored paths and determine the maximum between-ness which we store to be averaged and plotted for results.

Once this process is completed we remove the node with the maximum between-ness, recompute the links between all of the nodes and run the algorithm again up to $N/4$ times. We arrived at the $N/4$ value through trialling various configurations. This complete simulation in then run two hundred times to get an average for the final results.

Algorithm 1 Algorithm for the network generation and simulation

```plaintext
for all runs in totalNumberOfRuns do
    declare vector< Node > nodes
    Randomly assign positions for each node
    for all nodes in totalNumberOfNodesToToRemove do
        Calculate links for each node
        Compute paths > each node to every other node
        Compute betweenness for each node and assign
        Store the maximum betweenness in int array
    end for
    Write betweenness array to file
end for
```

Algorithm 3 shows the pseudo-code for the generation and simulation of the mesh in both two and three dimensions. Importantly we see that the links are re-calculated and paths to each node are re-computed after each node is removed. This is to make sure that the paths for all nodes are still the shortest as by removing the node with the maximum between-ness it should affect the most nodes by definition of having the highest between-ness.
Figure 3. Visualisation of a two-dimensional, $N = 100$, non-periodic system where $R = 0.1$. Showing Connections between linked nodes and fragmentation into separate component clusters when 149 nodes have been removed.

The visualisation elements of this simulation are performed in a separate program and reads in dump-files produced in each simulation step and produce renderings of the two-dimensional or three-dimensional mesh using OpenGL. The output of the visualisation is shown in Figure 2. This shows a two-dimensional system with $N = 600$ and $R = 0.1$ no nodes removed and the links between the nodes shown in grey. Figure 3 shows the same system after 150 of the nodes with the highest between-ness have been removed. We see that the network is no longer contiguous and has been separated into eight clusters. This is the expected result as the densest clusters are provide the most routes and thus have the smallest between-ness. Creating a visual output for the user is necessary to understand the microscopic behaviour of individual nodes and validate the code creating the links.

4 Experimental Results

We present plots of the average Dijkstra distance for various generated networks, and also plots of the experimentally measured between-ness. These two metrics are shown as the number of nodes removed from the network is progressively removed.

The data shown has been averaged over many differently seeded randomly generated network node patterns. This allows us to extract the general statistical trend independent of the particular accidental pathologies of a particular pattern of generated nodes. In addition an estimate of the relative fluctuations is shown, based on the standard deviations from the averages across different network patterns.

In all off the plots presented below the error bars have been scaled by $\frac{1}{10}$. This is to allow us to compare the relative errors both within each data set and between different sets. The standard deviation cannot be reduced by increasing the number of tests performed because of the nature of the random system. The maximum between-ness that is produced is so dependent on the network topology that for each different random configuration the maximum can vary by a factor of two or more.

Figure 4 shows the systematic variation of the average Dijkstra distance of simulated networks for various values of radial connectivity. When the radius is small the Dijkstra distance simply tails off as nodes are removed. However at a critical value of radius the connectivity is such that a peak forms. The peak shows where there is a reversal of the trend. Providing the nodes are sufficiently well connected – with radius above 0.075, there is an initial growth in Dijkstra distance as nodes are removed, followed by a downward trend when the network breaks up into separate fragments.

Figure 5 shows the effect on the between-ness of having different densities of nodes in the system. Since the area of the unit box is fixed, varying $N$ varies the density. At low density there is no peak in the curve and the between-ness tails off monotonically with removed nodes.

As density is increased ($N$ of 400 and 600) a peak and trough region appears along with a regime of rising maximum between-ness. This means that at low density no node is dominant. At higher densities, nodes can play a very important connectivity role which holds the network together until the transition occurs and it tails off into fragments.

The absolute values of between-ness vary for different densities too. This emphasises that a dense system can have nodes that are relatively speaking much more connected than some. They can act as hub nodes that are therefore of much higher importance to overall network properties than others.

Figure 7 shows the system with periodic boundary
Figure 5. Graph of the maximum between-ness for $N = 400$ and $N = 600$ in a non-periodic system two dimensions where $R = 0.1$

Figure 6. Graph of the distribution of points for single values of nodes removed

Figure 6 shows a histogram of the distribution of values for a single value of the number of removed node $n^*_r$. As can be seen the distribution is significantly skewed, with a tail towards higher between-ness values. We have employed a representative average value for the plots along with uncertainty estimates based on approximate widths of the distribution. This approach appears to give consistent behaviour. This skewness is likely a manifestation of finite-size effects.

conditions. This means the system connectivity remains higher and the system is less prone to fragmentation. The result is that all three curves are shifted into the same region as the low density non-periodic system and there is no peak in the between-ness.

Figure 8 shows us that the effect of different radii on the maximum between-ness vs the number of culled nodes in a non periodic, two dimensional system where $N = 600$. The first radius tested was $r = 0.05$ and this network failed to maintain any stability as the number of connections was too small to maintain a contiguous network. For radius $r = 0.75$ the effect of removing the nodes with the highest between-ness shows a high peak followed by steep degradation of the network. We see the between-ness drop from the initial configuration to a local minimum then rise sharply to a peak. This local minimum represents the most desirable network configuration because the maximum reliance on a single node is minimised. This effect is visible from $r = 0.75$ through $r = 0.2$. For $r = 0.5$ and $0.8$ we see a gradual degradation of that network rather than a specific peak. This is due to the radius being so large that each point is connected to most of the other points as thus removing reliance on any single node. The average number of connections for each node in the non-periodic, two dimensional system where $N = 600$ and $r = 0.1$, is 14. We believe that this is representative of real world systems shown in [15] and [14].

Figure 9 shows the effect of different radii on the maximum between-ness in a three dimensional mesh simulation. We first note that the transition from completely unstable to gradual degradation is much sharper than the two-dimensional simulation case. Secondly, the three-dimensional network degrades similarly to the two-dimensional system with a initial local minima which rises to a peak then quickly degrades to equilibrium.

5 Discussion

We see from Figure 7 that introducing periodic boundary conditions into the system significantly increases the robustness of the network. Periodicity is a useful property to introduce in some models, to preserve local symmetry. In the systems discussed in this paper however periodicity is somewhat artificial and unrealistic for social, and other spatial model network systems.

We can explore systematic variations in the network transition points by plotting (approximate) peak points for the different radii values. Figure 10 shows there is mono-
It appears that the Dijkstra distance and the betweenness centrality are useful metrics to observe network robustness and fragmentation. The presence and position of the peak in the maximum between-ness plotted against removed number of nodes gives a strong indication of if and when a network will fail.

In this paper we have been able to examine these trends systematically and with averaged curved over many different realised individual network patterns. This is useful to build up a picture of the systemic behaviour. Studying one-off individual networks in areas such as water networks, power systems or other phenomena is often difficult.
Figure 9. Plot of the maximum between-ness with different radii for N = 600 in a non-periodic three-dimensional system

6 Conclusion

We have shown that the between-ness metric can be usefully applied as a metric to study the properties of a damaged mesh network, which might arise from mobile sensors, or from some other ad hoc arrangement of devices or individuals deployed in a spatial system. We developed the model of N randomly placed devices or individuals with the single model parameter of spatial proximity radius r. The number of individuals placed in a unit box effectively controls the system density and the radius controls the connectivity. In effect it appears that it is the ratio between these that governs behaviour of the model.

We have seen that system behaviour changes quite dramatically depending upon whether the model system is periodic or not. This seems to be largely due to the more complete connectivity for a given radius, in a periodic system. In a non-periodic system the nodes are classified as interior nodes with a straightforward average connectivity, or boundary nodes which will have a reduced connectivity. We also see measurable differences in the change in the between-ness for two and three dimensional systems and the resulting differing average connectivities. There is scope to explore this effect further with four-dimensional or higher dimensional simulated systems. It might also be usefully investigated in other mesh patterns or geometries.

The experimental procedure of systematically remov-
ing the most connected node in rank order gives interesting insights into the systemic changes in connectivity and ultimately into the fragmentation point of the networks. This paper has combined this metric along with a systemic network generation model to show definite trends in network properties that can be characterised independently of fluctuations arising from accidental properties of individual random network patterns.

We believe an understanding of these trends and systems changes in fragmentation point will be useful in further study of actual networks and other network generation models. In particular, for studying the systemic and progressive effects of damage in real and simulated networks.

We have been able to investigate relatively large individual networks and to sample over statistically representative number of different networks. There is scope to apply parallel computing techniques to computing for example betweenness so that even larger networks can be analysed in a feasible amount of compute time.

References