

HELSINKI UNIVERSITY OF TECHNOLOGY

Department of Electrical and Communications Engineering

Henri Koskinen

Connectivity and Reliability in Ad Hoc Networks

Master's thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology

Supervisor: Professor Samuli Aalto (pro tem)

Instructor: Professor Jorma Virtamo

Espoo, 5th February 2003

Author:	Henri Koskinen		
Title:	Connectivity and Reliability in Ad Hoc Networks		
Date:	5th February 2003	Number of pages:	48
Department:	Department of Electrical and Communications Engineering		
Professorship:	S-38 Teletraffic Theory		
Supervisor:	Professor Samuli Aalto (pro tem)		
Instructor:	Professor Jorma Virtamo		
<p>An ad hoc network is a research concept that has gained increasing attention lately. It is defined as a wireless multihop network independent of any fixed network infrastructure, formed by mobile terminal devices. There are numerous potential applications for such networks: conferencing, military networks, and networks formed in emergency and rescue operations, just to name a few.</p> <p>Despite a long history of research, the first ad hoc network is yet to be implemented: the nature of ad hoc networks poses several challenging problems. One of them is that of connectivity, namely, the requirement that the network connect every pair of network nodes. This property depends on the pairwise node distances and the range of communication of the nodes.</p> <p>This study addresses the connectivity problem by modelling the network as a geometric random graph. Assuming a common communication range for the network nodes (or, more generally, limit for the range), the threshold value of this range for connectivity is defined as a random variable. Utilizing a graph algorithm that finds this threshold range from a given set of nodes, extensive simulations are carried out assuming uniform spatial distribution of the nodes. Analysis of the simulation data results in statistical models that bind together the required number of nodes and/or communication range and the allowed area spanned by the network so that a random network is connected with a high probability.</p> <p>The scope of the study is extended to the more general concept of k-connectivity. A k-connected network retains connectivity after the removal of any $k - 1$ nodes, which bears significance in terms of network reliability. The definition of the threshold range can readily be generalized to k-connectivity. Algorithms for finding the threshold ranges for 2- and 3-connectivity are developed, and statistical models are fitted to simulation data in the same way as in the case of simple connectivity. In addition, some comparative analysis between the communication ranges required for the different degrees of connectivity is made.</p>			
Keywords:	Ad hoc networks, connectivity, reliability, random graphs, graph algorithms, statistical models		

Tekijä:	Henri Koskinen		
Otsikko:	Ad hoc -verkkojen yhteydessisyys ja luotettavuus		
Päivämäärä:	5. helmikuuta 2003	Sivumäärä:	48
Osasto:	Sähkö- ja tietoliikennetekniikan osasto		
Professuuri:	S-38 Teleliikenneteoria		
Työn valvoja:	Ma. professori Samuli Aalto		
Työn ohjaaja:	Professori Jorma Virtamo		
<p>Ad hoc -verkko on viime aikoina kasvavaa kiinnostusta herättänyt tutkimuskonsepti. Sil- lä tarkoitetaan langattomien päätelaitteiden keskenään muodostamaa verkkoa, joka on täysin riippumaton kiinteästä verkkoinfrastruktuurista. Tällaisille verkoille on lukuisia sovellusmahdollisuuksia, joista erilaiset istunnot, sotilasverkot ja pelastusviranomaisten häätätilanteissa muodostamat verkot ovat vain muutamia esimerkkejä.</p> <p>Vaikka ad hoc -verkkoja on tutkittu jo kauan, ensimmäinen käytännön toteutus antaa vielä odottaa itseään: ad hoc -verkkoihin liittyy monta haastavaa ongelmaa. Eräs niistä on verkon yhteydessisyys eli vaatimus siitä, että verkon kaikki solmut saavat yhteyden toisiinsa. Tämä riippuu solmujen keskinäisistä etäisyyksistä ja suoran viestinnän kan- tamasta.</p> <p>Tässä työssä lähestytään yhteydessisyysongelmaa mallintamalla verkko satunnaisgraafi- na. Olettamalla kaikille verkon solmuille yhtä suuri kantama (tai yleisemmin suurin saavutettava kantama) määritellään verkon yhteydessisyyden rajakantama satunnais- muuttujaksi. Tämän käyttäytymistä tutkitaan simuloimalla olettaen, että solmujen si- jainnit noudattavat tasajakaumaa, ja käyttäen apuna graafialgoritmia, joka määrittää rajakantaman annetusta solmujoukosta. Simulointidatan analyysin tuloksena saadaan tilastollisia malleja, jotka kytkevät toisiinsa tarvittavan solmujen lukumäärän ja/tai kan- taman sekä sallitun pinta-alan, jolle solmut ovat hajaantuneet, siten että satunnainen verkko on yhteydessinen (graafiteorian termein yhtenäinen) suurella todennäköisyydellä.</p> <p>Tarkastelua laajennetaan yleisempään k-yhtenäisyyden käsitteeseen. Verkko joka on k- yhtenäinen pysyy yhtenäisenä, kun siitä poistetaan mitkä tahansa $k - 1$ solmua, millä on tärkeä merkitys verkon luotettavuuden kannalta. Rajakantaman määritelmä voidaan suoraan yleistää k-yhtenäisyyteen. Työssä kehitetään 2- ja 3-yhtenäisyyden rajakanta- mat etsivät algoritmit, ja simulointidataan sovitetaan tilastolliset mallit samaan tapaan kuin yksinkertaisen yhtenäisyyden tapauksessa. Lisäksi tehdään vertailevaa analyysiä eri yhtenäisyyden asteisiin tarvittavien kantamien välillä.</p>			
Avainsanat:	Ad hoc -verkot, yhteydessisyys, yhtenäisyys, luotettavuus, satunnaisgraafit, graafialgoritmit, tilastolliset mallit		

Acknowledgements

This Master's thesis was written in the Networking Laboratory of Helsinki University of Technology for the AHRAS project funded by the Finnish Defence Forces Technical Research Centre.

I would like to thank the supervisor of the thesis, Professor Samuli Aalto, for his insightful and tireless guidance, and the instructor, Professor Jorma Virtamo, for his invaluable tutoring. It has truly been a privilege to work under the unparalleled instruction of two professors.

I also thank all the other people in the lab for making it such an inspirational working environment. In particular I want to thank my office mates (I had three of them along the way - I hope I'm not the reason!) for showing me the ropes. Daniel Lopez, Eemeli Kuumola and Aleksi Penttinen were all very much fun to work with.

Finally, I would like to express my utmost gratitude to my dear family and friends for all their support over the past years. I owe them a debt for a great many things.

Espoo, 5th February 2003

Henri Koskinen

Contents

1	Introduction	1
1.1	Ad hoc networks	1
1.2	Contributions of this study	2
1.3	Structure of the thesis	2
2	The connectivity problem	4
2.1	Problem statement	4
2.2	Previous work	5
2.3	Used approach	7
3	Simple connectivity	10
3.1	Finding the critical transmission range for connectivity	10
3.2	Collection and analysis of statistics	11
3.2.1	Modelling the distribution	11
3.2.2	Modelling individual quantiles	17
4	Biconnectivity	20
4.1	Fault tolerance and route diversity through range redundancy	20
4.2	Finding the transmission range required for biconnectivity	21
4.2.1	Depth-first search	21
4.2.2	Incrementing the range gradually	23
4.2.3	Summary of the algorithm	25
4.3	Statistical analysis	25
5	Triconnectivity	29
5.1	Introduction	29
5.2	Finding the transmission range required for triconnectivity	31
5.2.1	Storing the DFS tree	31
5.2.2	Finding articulation pairs	32
5.2.3	Efficiency improvement	34

5.2.4	Summary of the algorithm	35
5.3	Statistical analysis	36
6	Discussion and conclusions	39
6.1	Application of the models	39
6.2	Asymptotic examination	40
6.3	Diagnostics	41
6.4	Conclusions	42
6.5	Further work	42
7	Summary	44
	References	45
A	Algorithms	46

Chapter 1

Introduction

1.1 Ad hoc networks

The concept of an ad hoc network is defined as a network formed by mobile, wireless terminal devices without the aid of any fixed infrastructure. The network operates in a decentralized manner, utilizing multihop connections to carry traffic. This means that the devices are in the role of both terminals and routers. The topology of the network can change dynamically because of the mobility of the devices.

Ad hoc networks have been a topic of research from as early as the 1960's, only at that time the concept was referred to as packet radio networks. Several projects have been engaged in the field since then, but no implementation satisfying the criteria of a pure ad hoc network has been introduced. This is due to several problematic issues inherent to such networks. The concept has thus remained an intellectual challenge for the research community.

What keeps up the interest for such networks are the alluring potential applications for them. Conferencing using laptop computers or hand-held devices is one obvious example, military and emergency networks are others. Dedicated sensor networks could provide measurement and monitoring information about an area; vehicles could be connected to each other with ad hoc networks. This, along with recent leaps in the development of various communication technologies, is why ad hoc networks have become a "hot topic" lately. In 1999, IETF created the MANET working group (short for Mobile Ad Hoc Networks) as a framework for the study of this field. Extensive overviews of potential applications as well as emerged ideas, algorithms and protocols for ad hoc networks are given in [1] and [2].

As already mentioned, ad hoc networks have several characteristics unique to them that pose problems in their design. Such issues have been widely discussed in [3]. Some of the topmost questions are summarized in the following.

Routing is traditionally based on maintaining topology information of the network at each party involved in the routing process. The dynamic topology of ad hoc networks resulting from the mobility of the network nodes makes this challenging: attempting to keep the information up to date in the presence of a high degree of mobility will easily lead to unsustainable signalling loads. As an alternative to the conventional proactive routing principles, reactive methods have been proposed in which routes only for connections to be established are found on demand.

Another issue is scalability. It has been shown that with a uniform traffic pattern,

the capacity per node decreases when the number of nodes increases, as each node becomes more and more occupied in relaying traffic. Large-scale networks have therefore been deemed infeasible. The depletion of capacity is amplified particularly in routing bottlenecks, i.e. nodes used for relaying by several connections. Routing bottlenecks can be reduced by route diversity.

Since the devices making up the network are most likely to be battery-operated, energy efficiency becomes important for maximizing their functioning time. Transmission power adjustment, minimum-energy routing and routing policies are ways to achieve this goal.

The security of ad hoc networks is a difficult matter. This problem has already been witnessed in the existing wireless LAN networks. Measures need to be taken in order to prevent the intrusion of outsiders into and the leakage of information out of the network.

Finally, there are some fundamental questions that arise when Quality of Service in ad hoc networks is considered. The reliability of connections is one of them. Node mobility makes connections unstable; they can also fail as a result of the exhaustion of energy in relay nodes. These problems can be alleviated by enabling several independent routes for a given connection. However, due to the nature of the wireless links, the establishment of a connection between any two nodes may not always even be possible. This brings us to the problem of network connectivity which is the topic of this study. Network connectivity is closely related to route diversity, which makes it an important property in terms of network capacity as well as reliability.

1.2 Contributions of this study

The aim of this study is to find a model that binds together three quantities, namely, the number of nodes in an ad hoc network, the maximum distance over which two nodes can communicate, and the area over which the nodes are scattered, in such a way that the resulting network is connected (or further biconnected or triconnected) with a high probability when the nodes are assumed to be spatially uniformly distributed.

The approach is to regard the network as a geometric random graph. Some basic results regarding the connectivity of such graphs are shown, and algorithms are developed that determine from a given set of nodes the threshold communication range for 1-, 2- and 3-connectivity. These algorithms are utilized to carry out extensive simulations in order to obtain data for statistical analysis. Finally, models are fitted to the data to capture the behavior of the threshold ranges and thereby to predict connectivity in ad hoc networks.

1.3 Structure of the thesis

Chapter 2 introduces the connectivity problem and gives an overview of previous work carried out on the topic. In addition, the perspective taken on the problem in this study is described.

Chapter 3 focuses on simple connectivity, i.e. the requirement that every node is

connected to every other node by the network. Chapters 4 and 5 broaden the scope to biconnectivity and triconnectivity, respectively. These properties mean that the network retains simple connectivity even at the failure of any one or two nodes, respectively. Algorithms play a central role in the latter two chapters.

Discussion and diagnostics of the findings as well as conclusions and reflections on future work are presented in chapter 6. Chapter 7 concludes the study with a brief summary.

Chapter 2

The connectivity problem

This chapter presents the problem of connectivity in ad hoc networks. It gives an overview of previous studies on the matter. This is followed by a brief preview on the approach taken in this study, the methods used and developed, and the findings made.

2.1 Problem statement

A widely studied problem in the field of ad hoc networks is that of connectivity: every node in the network should be able to communicate with every other node. This fundamental property of the network boils down to how the largest distance over which two nodes can communicate directly relates to the pairwise node distances in the network. Figure 2.1 illustrates this relation.

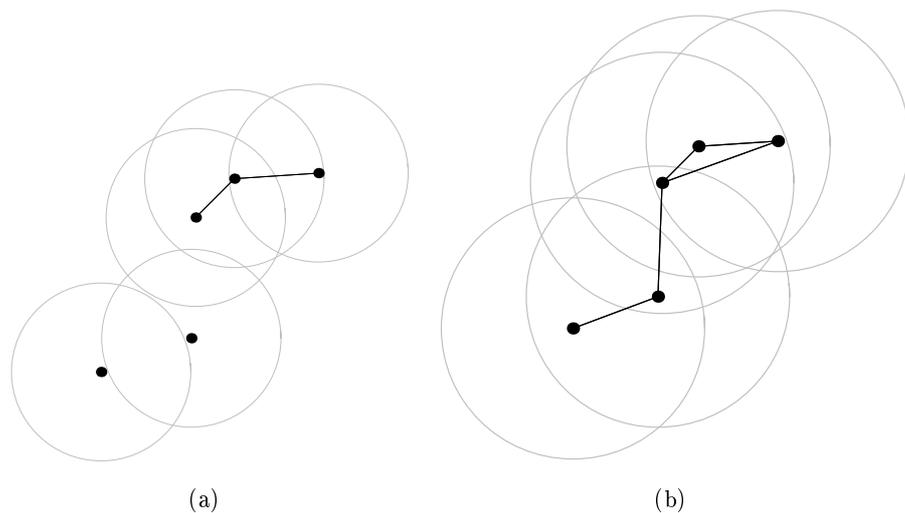


Figure 2.1: Example of how the network topology depends on the maximum communication range

2.2 Previous work

This problem has been tackled by choosing appropriate models to describe the network and deriving results analytically from these models. Typically, and as in Figure 2.1, the maximum transmission range r has been assumed to be constant, implying that all nodes have equal transmission and reception capabilities and that the radio environment determining the attenuation of transmissions is homogenous throughout the network. As for the spatial distribution of the nodes, by far the most popular model is the two-dimensional homogenous Poisson point process, characterized by the intensity λ . This process has the following properties:

1. The number of points in a region of area A in the plane is Poisson-distributed with mean λA .
2. The point numbers in any two non-intersecting regions are independent.

In particular, λ is the average number of points per unit area and can therefore be interpreted as the density of points. Also, given the number of points in a certain region, the points in that region are uniformly distributed.

This model is partly motivated by continuum percolation [4] where a theorem closely related to the problem of connectivity has been proven in the following setting. Disks with equal radii are generated on an infinite plane according to a Poisson process. As the intensity of the process increases, disks overlap and form clusters. The theorem states that there exists a finite critical intensity above which a unique unbounded cluster exists almost surely. It should be noted that a cluster of disks is equivalent to a connected network when the maximum transmission range is twice the disk radius. This simple relation is demonstrated by Figure 2.2.

In one dimension, the unbounded cluster corresponds directly to a connected network with maximum transmission range twice the disk radius. However, in a recent study Dousse et al. [5] showed that the appearance of an unbounded cluster at a finite intensity, i.e. the occurrence of percolation, requires the domain to be infinite in two dimensions. Then again in two dimensions, an infinite cluster forming is a weaker condition than full connectivity. Indeed, Philips et al. showed in [6] that with the model of a two-dimensional Poisson point process with node density λ and

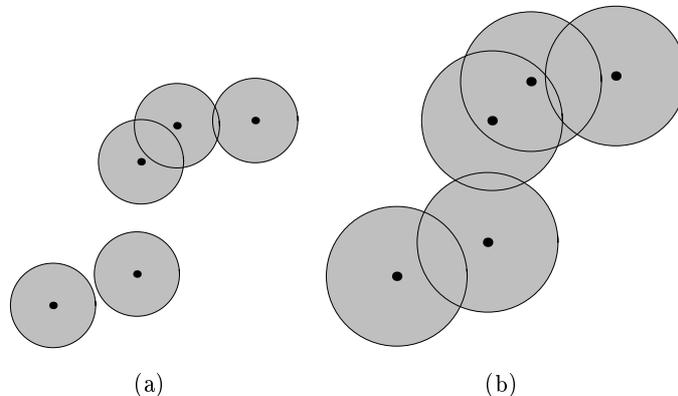


Figure 2.2: Disks with same locations as in Figure 2.1 but with radii equal to half the ranges shown therein

transmission range r , the expected number of direct neighbors of a node, $\lambda\pi r^2$, must grow logarithmically with the network area in order to ensure connectivity. In other words, a larger area always requires either a longer transmission range or a greater node density. The intuitive explanation is that the greater the area, the more likely a region with sparsely located nodes. This renders the connectivity of random networks in the infinite plane with limited transmission range impossible.

Also in [5], the number of alternate link-disjoint paths between node pairs was examined under the two-dimensional percolation model in order to study network reliability and routing bottlenecks. This is equal to the size of the minimal cutset in the network between the pair of nodes, i.e. the smallest set of links whose removal would disconnect the two nodes. It was proven that the number of alternate paths between nodes A and B is not less than $\min\{N_A^\infty, N_B^\infty\}$, where N_K^∞ is the size of the minimal cutset separating node K from nodes located arbitrarily far away from K . In fact, this was shown to be equal to the size of the minimal cutset separating node K from nodes at a finite distance L_K for each node K . It was found by simulation that for a fixed range r , the mean of L_K tends to zero when the density λ increases. In other words, the number of link-disjoint paths between two nodes is determined by the number of nodes adjacent to each end node when the node density is high. With this observation, the authors in effect discovered the validity of a theorem regarding the connectivity of geometric random graphs, which had recently been proven by Penrose in [7] and which will be presented and utilized later in this study.

Dousse et al. further observed that when the density is just above the critical percolation density, the network consists of "islands", clusters of nodes well connected to each other, which in turn are interconnected by few links only, acting as the bottlenecks of the network. These bottlenecks correspond directly to the notion of articulation points or sets observed in this study.

In [8], Bettstetter derived an analytical expression in attempt to bind together the transmission range and the node density needed to obtain almost surely a connected (or, as an extension, k -connected; a concept that will be introduced later and is also observed in this study) network. It is worth noting that this approach does not take into account the dependence on the system area shown by Philips et al. The expression was derived starting from the probability

$$e^{-\lambda\pi r^2}$$

of a random node, with transmission range r and generated from a Poisson process with density λ , being out of range from all other nodes. With the simplifying assumption of this event being independent for all n nodes in the network, the probability of no node being isolated was then estimated to be

$$(1 - e^{-\lambda\pi r^2})^n. \tag{2.1}$$

Based on the theorem by Penrose, this was further approximated to give the probability that the network is connected. (The theorem, in a narrow form, states that as the node density tends to infinity, the transmission range needed to connect every node to at least one other node equals the range needed for connectivity with a probability that tends to one.)

Problems were encountered when trying to verify this expression with simulations of 500 nodes scattered on a square-shaped area: with a fixed range, the proportion of connected realizations turned out to be significantly less than predicted. This

was explained to result from the fact that the simulations were carried out on a bounded area whereas the analytical expression was derived for an infinite area. In the bounded square, nodes located near the edges of the area are likely to have less connections than those located in the middle. This *border effect* was eliminated by using a toroidal distance metric, i.e. assuming that nodes at the edge were considered close to nodes at the opposite edge. Using this distance metric provided more satisfactory agreement with the analytical expression, although the probability of connectivity was still overestimated.

All in all, the applicability of the derived analytical expression remained ambiguous. It may agree to some extent with simulations using the toroidal distance metric, but that does not mean it applies in the infinite plane: the probability of connectivity in the infinite plane is always zero with finite transmission range and density of randomly placed nodes, as already shown by Philips et al. In fact, this applies with the above expressions as well: if n is assumed finite along with r , then $\lambda = n/A$ becomes arbitrarily small, making every node isolated with a probability arbitrarily close to one. A positive λ requires n to be infinite, which results in the estimated connectivity probability (2.1) becoming arbitrarily small with finite r .

Moreover, it can be argued that the domain of any real-life ad hoc network is bound to have edges anyway; a counterexample would have to be as far-fetched as a globe-scale network scattered over oceans as well as continents. (Of course, this would come into question in the case of a network constituted by satellites.) It is therefore necessary to take the implications of this fact into account.

The border effect was taken into consideration in a later study by Bettstetter and Zangl [9]. The approach was to divide the observed region into a central area void of border effects, consisting of the points separated from all the boundaries by at least the transmission range, and several border zones. Applying geometrical analysis, the worst-case coverage area of nodes located in these border zones was calculated. Using the same logic as above, this was then used in deriving a fairly complicated expression for the probability of no node being isolated which in turn served as an approximation for the probability of connectivity. Although several corners were cut in deriving this expression, it turned out to be in considerably better agreement with simulations than (2.1). In addition to the circular domain considered, the authors also reported having successfully predicted simulation results in a square region with the same method.

Despite the encouraging results, even the latter analytical method has some shortcomings. In particular, via utilizing Penrose's theorem, it applies only when $n \gg 1$. Indeed, agreement with simulation results was only demonstrated with $n \geq 100$. Also, applying the analysis to more complex-shaped domains becomes increasingly difficult. Finally, solving the desired parameter - either the range in relative units or the number of nodes required - from the derived analytical expression is not straightforward.

2.3 Used approach

In this study, the problem of connectivity is approached from the following probabilistic angle. Each realization of n randomly placed nodes has its threshold transmission range which is required for connectivity. This threshold range, or the *critical transmission range* $R_{\text{crit}}(n)$, is therefore a random variable with a certain distribution for

each n , which depends on the spatial distribution of the nodes. In several earlier studies on connectivity, for example [3] and [8], the effect of the transmission range has been studied by fixing it to a chosen value, generating a number of random realizations, and determining the proportion of realizations found to be connected with the chosen range. Repeating this with several fixed ranges and plotting the determined proportions against them, an interpolated curve making a transition from 0% to 100% has finally been obtained. The connection between this curve and the critical transmission range is obvious: the curve represents the cumulative distribution function of the random variable.

Motivated by the apparent difficulty of the analytical approach to the connectivity problem, the statistical behavior of the critical transmission range is studied through extensive simulations. This is done using the typical modelling assumptions: a chosen number of nodes uniformly distributed in a square region. In the analysis of the statistics, it is found that the behavior of the critical range can be described extremely well with models of simple, analytical form.

The study is extended to the stronger requirements of biconnectivity and triconnectivity: a k -connected network is one that remains connected after the removal of any $k - 1$ nodes or, equivalently, has at least k node-disjoint paths between every pair of nodes. The degree of connectivity is therefore an important property from the viewpoint of network reliability and load balancing. Biconnectivity and triconnectivity were also included in [8]; the number of link-disjoint paths was also studied in [5]. As in the case of connectivity, the threshold transmission range for bi- and triconnectivity can be similarly defined. The analysis of simulation data shows that the same distributions and models can be used to describe the statistical properties of these quantities as in the case of simple connectivity; only the parameters of the models differ.

As the task of determining the threshold range for a given degree of connectivity from a given set of nodes is more laborious than that of testing whether the nodes form a k -connected network with a given range, the development of algorithms for finding the threshold ranges is an integral part of this study. It turns out that in the case of simple connectivity, the threshold range is equal to the longest link distance in the minimum spanning tree of the nodes, as proposed by Sánchez, Manzoni and Haas in [10]. There are several known algorithms for finding this tree. In this context, the result shown in [6] means that this longest link distance tends to infinity with the network area when the node density is constant. In the cases of bi- and triconnectivity, the range is found incrementally, by first finding a lower bound for it. At this point, the theorem by Penrose is an important motivation. The range for biconnectivity can be found relatively easily by utilizing a well-known graph traversal algorithm. However, the case of triconnectivity is notably more difficult. In general, the problem of finding the threshold range for k -connectivity can always be decomposed into n problems concerned with $k - 1$ -connectivity. Because of the inefficiency of this brute-force method with any substantial number of nodes, more efficient methods are needed. Even though there exists an efficient data structure called the SPQR tree for decomposing a biconnected network into triconnected components, the difficulty of implementation suggested by the theory behind it discourages the use of this structure. Instead, an algorithm based on extensions to that used with biconnectivity is developed for the purposes of this study.

In order to verify the statistical models obtained for the threshold range for simple connectivity, their predictions are compared to simulation results presented in [8] and

used in the verification of the expression derived therein. This is justified because the simulation scenario in question involved up to ten times as many nodes as in the simulations carried out in this study. The model applied in the comparison is therefore not based on a replication of the original simulations.

Finally, it is shown that in reality the threshold range cannot obey the analytical form assumed by the model. This results from the fact that this model, too, predicts a finite threshold range in an infinite area. Indeed, further diagnostics show that the form of the model requires adjustment.

Chapter 3

Simple connectivity

This chapter deals with simple network connectivity as defined in the problem statement in the previous chapter. It first presents a method of finding the critical transmission range for connectivity for a given set of nodes. It then examines the statistical behavior of this quantity in two ways: by modelling its whole distribution, and by modelling individual quantiles of its distribution.

3.1 Finding the critical transmission range for connectivity

To begin, an important result given in [10] without proof is presented with the proof:

Theorem 3.1 *The critical transmission range for connectivity R_{crit} is equal to the longest link distance in the minimum spanning tree of the nodes.*

The minimum spanning tree, or MST, is a connected graph that contains all the nodes and minimizes the sum of the link distances.

Proof: Consider an arbitrary set of nodes and assume that their MST is known.

Assume first that R_{crit} is shorter than the MST's longest link. By the definition of the MST, its longest link (as well as every other link) is the shortest possible way to connect the two subsets of nodes separated by the link. (Otherwise the link sum of the tree could be made even smaller by changing the link to a shorter one.) The assumption made thus implies that R_{crit} is too short to connect the two subsets separated by the MST's longest link, which contradicts the definition of R_{crit} .

The assumption that R_{crit} is longer than the MST's longest link is trivially wrong, since in this case all the nodes have been connected using distances shorter than R_{crit} by the MST. This completes the proof. \square

Several algorithms exist for finding the MST; in this study, the Prim algorithm was used [11]: starting with any single node, new nodes are added to the tree one by one, so that at each step the node closest to the nodes included so far is added. One realization with 15 nodes as well as their MST is depicted in Figure 3.1.

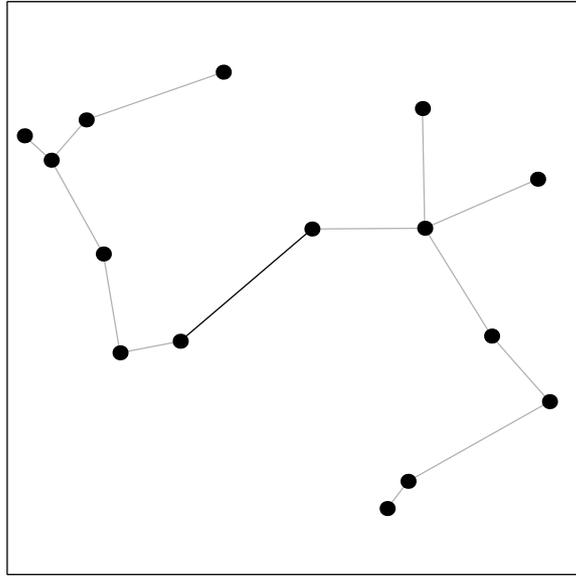


Figure 3.1: A sample set of 15 nodes and their MST. The longest link is shown with a dark line.

3.2 Collection and analysis of statistics

To gain information about the distributions of $R_{\text{crit}}(n)$, statistics were gathered. Each sample was obtained by placing n nodes in a unit square with uniform probability distribution and determining their MST. Data sets consisting of 5000 samples each were collected for various n . Figure 3.2 shows histograms of two such data sets, one with $n = 5$ and the other with $n = 350$.

3.2.1 Modelling the distribution

The first step in the analysis of the statistics was to find out how the expected value of R_{crit} depends on n . Figure 3.3(a) shows the means of the data sets plotted against n and Figure 3.3(b) their squared inverses against n . As can be seen, the plotted values in the latter seem to form a straight line. This encouraged fitting a model of the form $1/E[R_{\text{crit}}(n)]^2 = kn + c$ to the data using linear regression. However, this

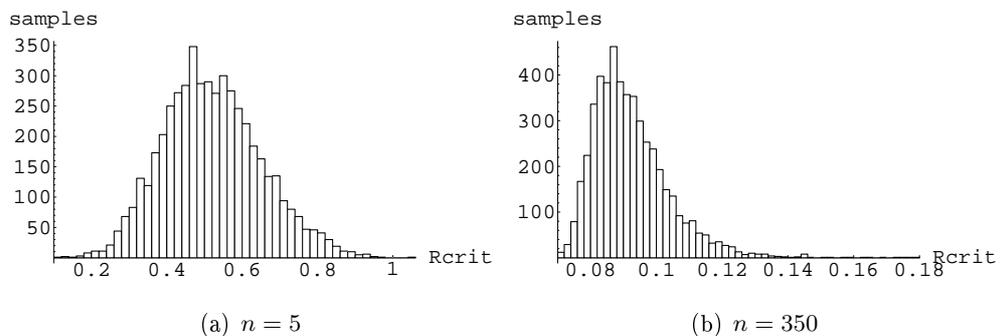
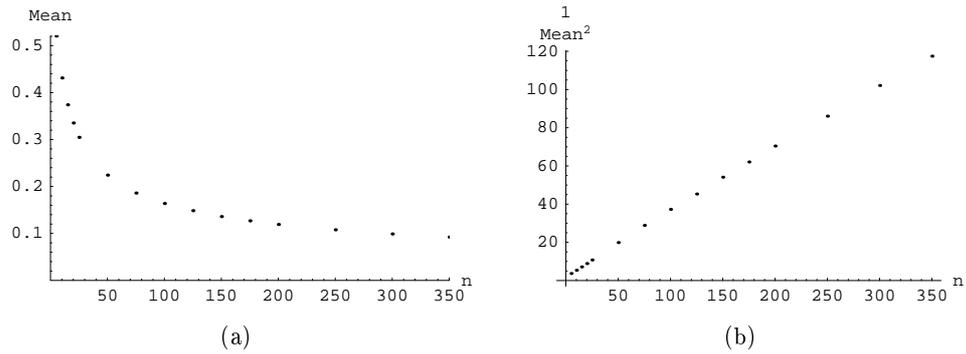


Figure 3.2: Histograms of data sets of R_{crit} samples

Figure 3.3: The dependence of the means of R_{crit} sample data sets on n

was done by minimizing a weighted sum of squares $\sum_i w_i e_i^2$ to compensate for the transformation done on the mean (denote this by y): without weights, the model would minimize the sum $\sum_i e_i^2 = \sum_i (kn_i + c - 1/y_i^2)^2$, whereas the ultimate goal is to minimize the sum of squared residuals of the model after the inverse transformation, $\sum_i (1/\sqrt{kn_i + c} - y_i)^2$, which is not the same. In light of the hypothesis of the model, $E[y] = 1/\sqrt{kn + c}$, the latter can be approximated as

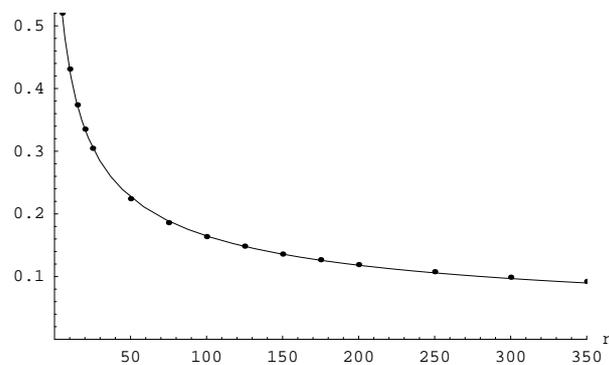
$$\begin{aligned} \sum_i \left(\frac{1}{\sqrt{kn_i + c}} - y_i \right)^2 &= \sum_i \left(\frac{kn_i + c}{\sqrt{kn_i + c}^3} - \frac{y_i^3}{y_i^2} \right)^2 \\ &\approx \sum_i \left[y_i^3 \left(kn_i + c - \frac{1}{y_i^2} \right) \right]^2 = \sum_i y_i^6 \left(kn_i + c - \frac{1}{y_i^2} \right)^2 \triangleq \sum_i w_i e_i^2, \end{aligned}$$

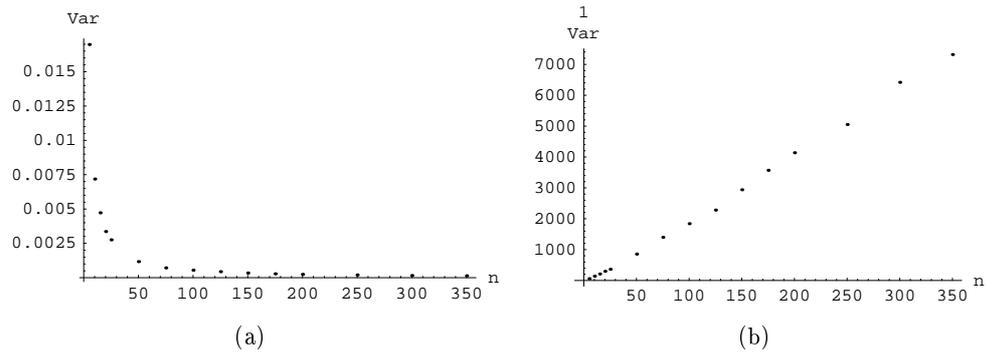
i.e. the sixth powers of the means were used as the weights.

Table 3.1 shows the results of the weighted linear regression which mean that the model obtained is roughly $E[R_{\text{crit}}(n)] = 1/\sqrt{n/3} + 2$. In the table, SE stands for the standard error of the parameter estimate, TStat is a t -distributed statistic for testing

Table 3.1: Linear regression results for the model of the expected value

	Estimate	SE	TStat	PValue
1	1.93794	0.0286534	67.634	0.
x	0.348727	0.00242932	143.549	0.

Figure 3.4: Model obtained for $E[R_{\text{crit}}(n)]$

Figure 3.5: The dependence of the sample variances of R_{crit} sample data sets on n

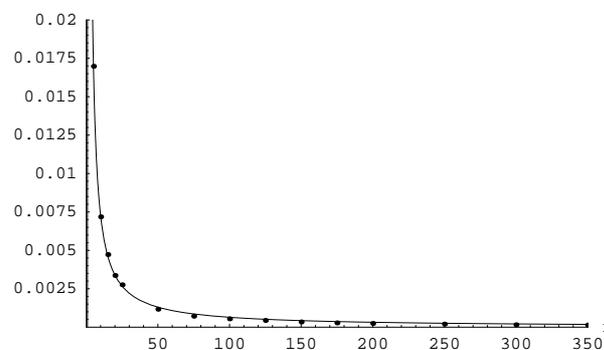
whether the parameter could be zero, and PValue is the probability of obtaining this statistic given the null hypothesis that the parameter is zero. It can be seen that with both parameters, this probability is equal to zero within the used accuracy, so the null hypothesis is extremely unlikely.

The coefficient of determination R^2 for this model was 99.98%. For comparison, that for the model obtained without using weights was 98.08%. Figure 3.4 shows the curve of the model together with the means of the data sets.

The next step was to examine the behavior of the variance of R_{crit} as a function of n . Figure 3.5(a) shows the sample variances of the data sets plotted against n and Figure 3.5(b) their inverses against n . The latter implies a linear dependence, so the model to be fitted was $1/\text{Var}[R_{\text{crit}}(n)] = sn + t$. Using the same logic as above, the fourth powers of the sample variances were used as weights in the regression. The results are shown in Table 3.2, making the model roughly $\text{Var}[R_{\text{crit}}(n)] = 1/(16n - 20)$. It should be noted that only values of n equal to or greater than 2 are sensible, so the model predicts a positive and limited variance for all cases that come into question (this was not the case with the model obtained without using weights). The curve

Table 3.2: Linear regression results for the model of the variance

	Estimate	SE	TStat	PValue
1	-20.0476	1.05272	-19.0436	0.
x	15.7905	0.193768	81.4917	0.

Figure 3.6: Model obtained for $\text{Var}[R_{\text{crit}}(n)]$

of the model together with the sample variances is shown in Figure 3.6.

The final goal was to fit a known, analytical probability distribution to each data set. Certain criteria were to be met by these distributions. First, they should be limited to a closed interval, as R_{crit} is limited to $[0, \sqrt{2}]$. Second, as Figure 3.2 shows, the distributions should become more peaked and develop a tail to the right as n increases.

The Beta(α, β) distribution seemed appropriate for the purpose. It is defined by

$$f_X(x) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1}, \quad 0 < x < 1, \quad \alpha, \beta > 0 \quad (3.1)$$

$$E[X] = \frac{\alpha}{\alpha + \beta}, \quad \text{Var}[X] = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}.$$

Two Beta distributions are shown in Figure 3.7. Since the distribution as such is defined for the interval $[0,1]$, it has to be scaled to cover $[0, \sqrt{2}]$ to suit our purpose. The resulting distribution is found by making the substitution $y = \sqrt{2}x$ in the relation between the probability density function (PDF) and the cumulative distribution function (CDF) as follows:

$$\int_0^x f_X(x) dx = F_X(x) = \int_0^y f_X\left(\frac{y}{\sqrt{2}}\right) \frac{dy}{\sqrt{2}} = F_X\left(\frac{y}{\sqrt{2}}\right).$$

The scaled distribution is therefore characterized by

$$f_Y(y) = \frac{f_X\left(\frac{y}{\sqrt{2}}\right)}{\sqrt{2}}, \quad F_Y(y) = F_X\left(\frac{y}{\sqrt{2}}\right), \quad 0 < y < \sqrt{2}, \quad \alpha, \beta > 0 \quad (3.2)$$

$$E[Y] = E[\sqrt{2}X] = \frac{\sqrt{2}\alpha}{\alpha + \beta}, \quad \text{Var}[Y] = \text{Var}[\sqrt{2}X] = \frac{2\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}.$$

With models describing both the expected value and the variance of R_{crit} as a function of n at hand, we can now make the expressions for $E[Y]$ and $\text{Var}[Y]$ above match the models to solve the parameters α and β for each n . First, we set

$$E[Y] = \frac{\sqrt{2}\alpha}{\alpha + \beta} = \frac{1}{\sqrt{kn + c}} \quad \Leftrightarrow \quad \beta = \alpha(\sqrt{2(kn + c)} - 1). \quad (3.3)$$

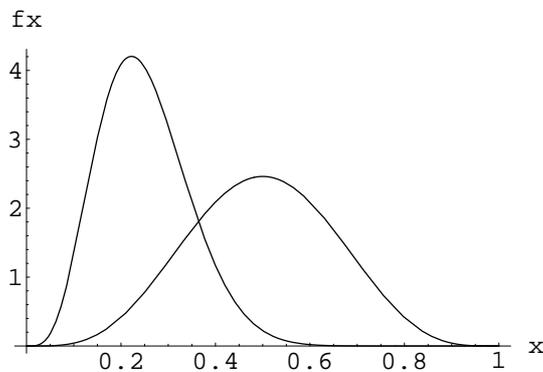


Figure 3.7: Beta(5,10) and Beta(5,5) distribution

Substituting this in the condition for the variance, we get

$$\begin{aligned} \text{Var}[Y] &= \frac{2\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} = \frac{2\alpha^2(\sqrt{2(kn + c)} - 1)}{2\alpha^2(kn + c)(\alpha\sqrt{2(kn + c)} + 1)} \\ &= \frac{\sqrt{2(kn + c)} - 1}{(kn + c)(\alpha\sqrt{2(kn + c)} + 1)} = \frac{1}{sn + t} \\ \Leftrightarrow \quad \alpha &= (sn + t) \left(\frac{1}{kn + c} - \frac{1}{\sqrt{2(kn + c)^{3/2}}} \right) - \frac{1}{\sqrt{2(kn + c)}}. \end{aligned} \quad (3.4)$$

Figure 3.8 shows PDF's of the form (3.2) obtained using (3.4) and then (3.3) plotted together with corresponding data set histograms that have been scaled so that their total integral becomes 1, for different n . It can be seen that for $n = 5$ the PDF is in excellent agreement with the statistics, but as n becomes larger, the fitted distribution does not keep up with the growing peakedness and asymmetry of the data. In the cases $n = 150$ and $n = 350$, this trend is exaggerated by the fact that the variance of the data is overestimated by the model obtained for it, as shown by Figure 3.6. However, it can be seen in the same figure that the variance is in fact underestimated by the model when $n = 25$, which confirms the trend.

Nevertheless, in all cases the PDF follows the tail of the histogram very closely, implying that these distributions could be used to determine how long a transmission range provides a high probability of connectivity for each n . This is supported by Figure 3.9 where the empirical cumulative distributions of the same data sets have been plotted along with the CDF's of the fitted distributions. It can be seen that for quantiles of around 90% and above, the two are very close to each other.

In Table 3.3, transmission ranges corresponding to certain quantiles of the fitted distributions have been given as arguments to the corresponding empirical cumulative distribution functions (ECDF's) to demonstrate how well these distributions serve

Table 3.3: Results of using fitted distributions for transmission range determination

n	Desired quantile (%)						
	50.0	75.0	90.0	95.0	99.0	99.5	99.9
	Obtained empirical quantile						
5	51.1	76.24	89.72	94.48	98.76	99.42	99.92
10	51.3	75.52	89.44	94.22	98.48	99.02	99.78
15	53.72	76.94	89.34	93.54	98.1	98.76	99.48
20	54.16	77.42	89.62	94.2	97.8	98.48	99.4
25	56.02	79.08	89.84	94.2	97.82	98.48	99.3
50	59.64	81.58	91.66	95.14	98.26	98.88	99.56
75	59.58	82.8	92.18	95.62	98.24	98.7	99.62
100	58.08	81.64	91.46	95.06	98.18	98.86	99.54
125	56.66	79.64	90.64	94.5	98.06	98.74	99.64
150	55.0	80.26	91.64	95.14	98.46	99.06	99.6
175	53.18	79.84	91.72	95.26	98.12	98.9	99.58
200	52.8	79.66	91.58	95.02	98.36	98.84	99.42
250	50.02	77.58	90.46	94.56	98.26	98.84	99.48
300	48.44	78.14	90.32	94.4	98.42	98.9	99.42
350	46.48	76.2	90.48	94.6	98.42	98.84	99.46

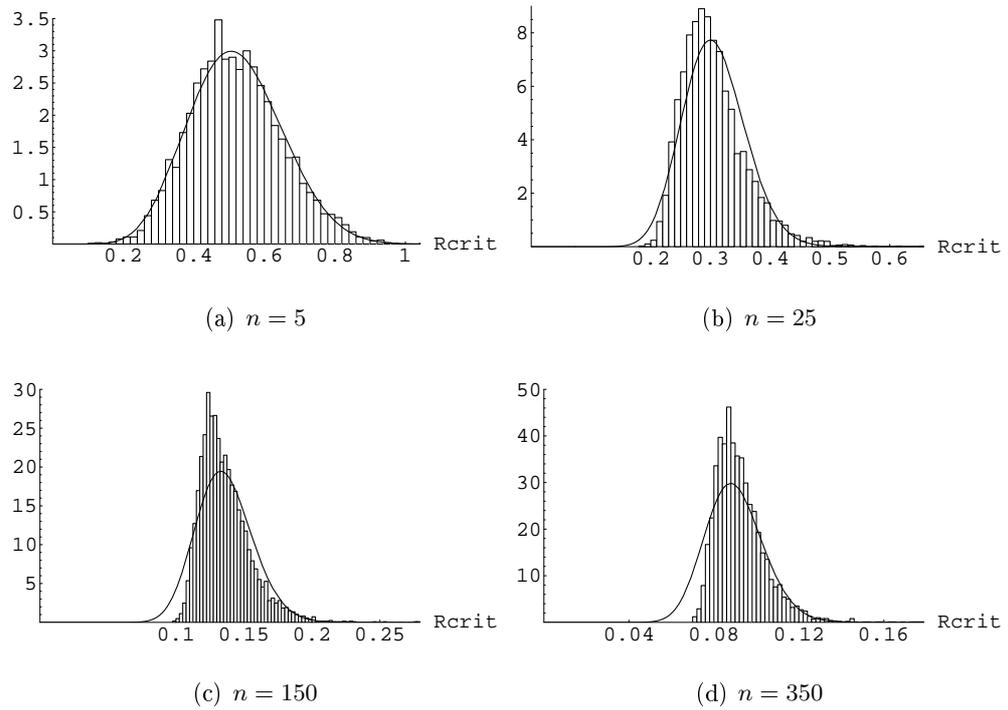


Figure 3.8: Histograms of R_{crit} samples and fitted Beta distribution PDF's

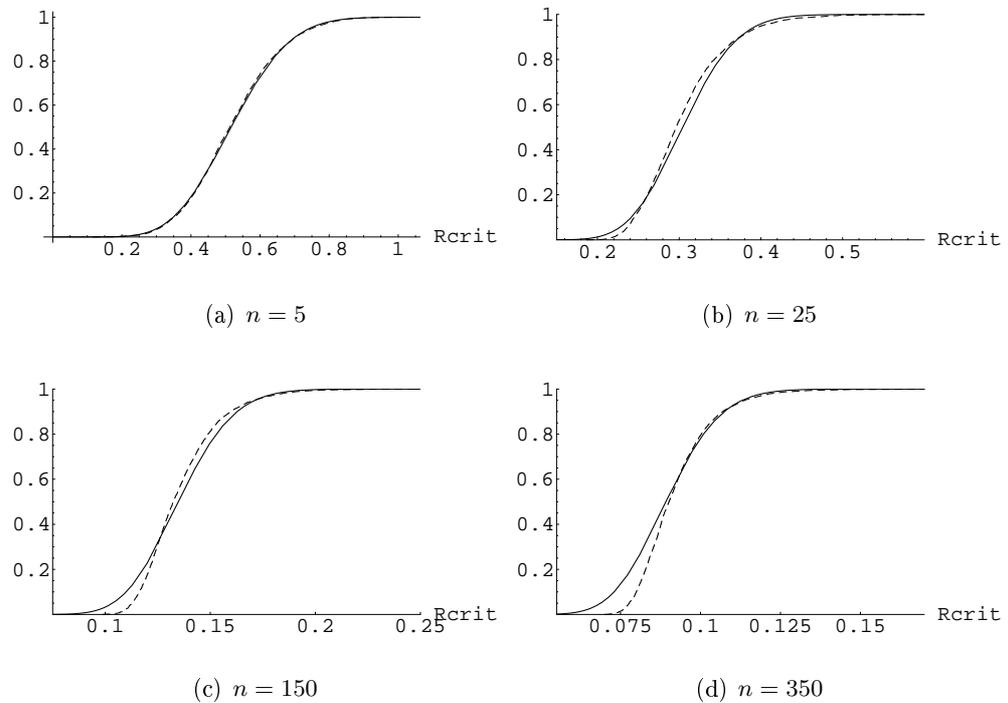


Figure 3.9: Empirical (dashed lines) and fitted Beta distribution CDF's

their purpose. For example, it can be seen that in light of the statistics at hand, the transmission range that according to the fitted distribution provides full connectivity with 50%-probability in the case of 350 randomly placed nodes actually does so with a probability of only 46.48%. This kind of inaccuracy in the lower quantiles was anticipated due to the observations made above. However, the results show that the difference between the desired and the achieved probability of full connectivity decreases as the desired probability increases: for instance, for the quantiles of 95% and above, the difference in probability in all cases is less than 1.5%. In practice, when estimating appropriate transmission ranges, the inaccuracies of the model could be compensated for by using sufficient margins in order to ensure the desired probability: Table 3.4 shows how a margin of 1.5% has been enough to achieve 95% probability of connectivity. Also, it can be seen in Table 3.3 that 99% is achieved using the requirement of 99.9%.

Table 3.4: Example of achieving 95% probability of connectivity using proper margin

n	5	10	15	20	25
ECDF _n (CDF _n ⁻¹ (96.5%)) (%)	95.96	95.8	95.12	95.52	95.4
	50	75	100	125	150
	96.16	96.56	96.16	95.76	96.14
	175	200	250	300	350
	96.26	96.08	95.56	96.08	95.84

It deserves to be pointed out that the fact that no ECDF value in the tables reaches 100% indicates that the fitted distribution does not provide grossly oversized transmission ranges, as $\text{ECDF}_n(x) = 1.0$ would mean that the critical transmission ranges found in all realizations with n nodes have been less than x .

3.2.2 Modelling individual quantiles

It was shown above that with the fitted distributions, the desired probability of connectivity can be achieved more accurately as the probability itself increases. This however does not mean that the higher the quantile, the more accurately it can be predicted: as the tail of the distribution is reached, the rate at which the cumulative probability grows with respect to the variable value decreases. In other words, vertical distances between CDF curves such as those in Figure 3.9 were considered above; how about the horizontal ones? Those in the same cases as in Table 3.3 are given in Table 3.5 where relative deviations of fitted distribution quantiles from empirical ones have been listed. It can be seen that, quite contrary to the accuracy in probability, the accuracy in transmission range leaves plenty of room for improvement in the highest quantiles and especially in them. Although this may result from the sample size used, as with 5000 samples an empirical tail probability of, for instance, 0.1% corresponds to only 5 realizations, this gave reason to investigate whether the behavior of individual quantiles as a function of n could be modelled in the same way as those of the expected value and the variance were before. Figure 3.10 shows that they do indeed obey the same form of functional dependence on n as the expected value. The decay of the slope in the last subfigure is assumed to result from devia-

Table 3.5: Relative errors of fitted distribution quantiles from corresponding empirical quantiles

n	Quantile of fitted distribution (%)						
	50.0	75.0	90.0	95.0	99.0	99.5	99.9
	Relative error from empirical quantile (%)						
5	0.76	0.82	-0.15	-1.00	-0.69	-0.92	0.65
10	0.64	0.36	-0.55	-1.16	-3.39	-4.10	-3.32
15	1.55	1.14	-0.69	-2.17	-4.99	-6.85	-7.77
20	1.76	1.57	-0.38	-1.33	-6.88	-7.34	-7.92
25	2.63	2.48	-0.23	-1.48	-7.02	-8.73	-11.03
50	3.36	3.67	1.96	0.31	-4.05	-6.02	-14.87
75	3.48	3.87	2.94	0.98	-4.19	-5.53	-7.92
100	2.64	3.30	1.79	0.06	-3.80	-5.70	-8.95
125	2.03	2.31	0.78	-0.82	-5.45	-5.46	-9.99
150	1.85	2.47	1.88	0.29	-2.85	-3.61	-9.21
175	0.91	2.19	1.53	0.33	-4.30	-4.83	-6.54
200	0.89	2.26	1.68	0.08	-4.39	-7.31	-8.93
250	0.007	1.18	0.44	-1.14	-4.04	-6.78	-7.24
300	-0.42	1.45	0.33	-0.70	-3.54	-7.58	-11.13
350	-1.07	0.44	0.44	-0.79	-4.20	-7.18	-17.23

tions due to the sample size. Table 3.6 shows the resulting estimates from weighted linear regressions when models for the highest and therefore most interesting quantiles were fitted. The curves of the models are shown in Figure 3.11, together with the empirical quantiles.

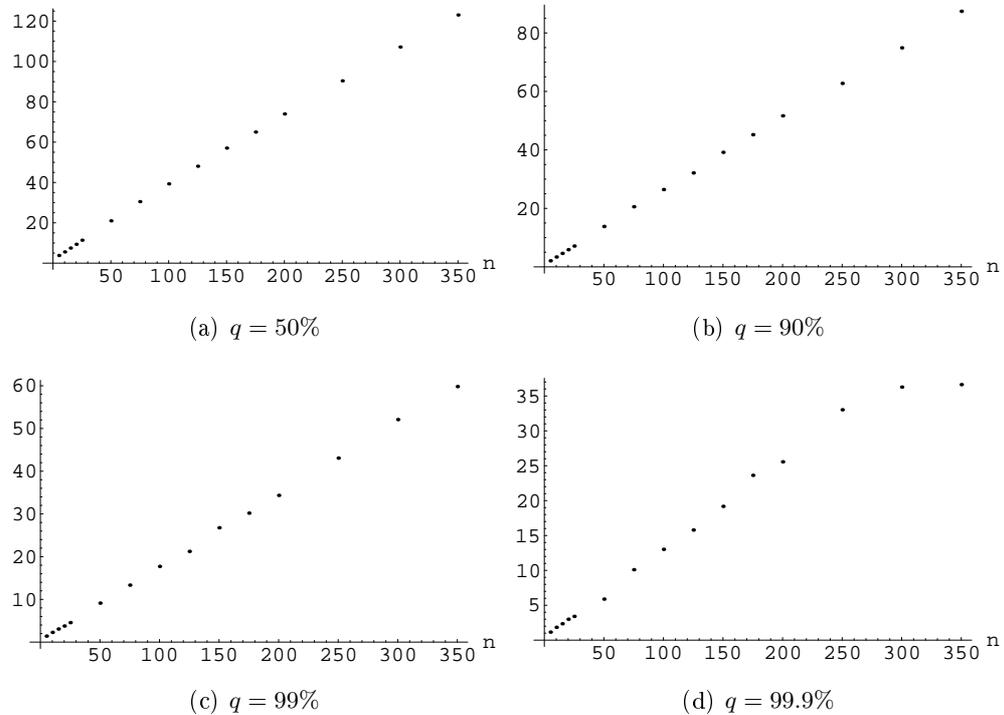
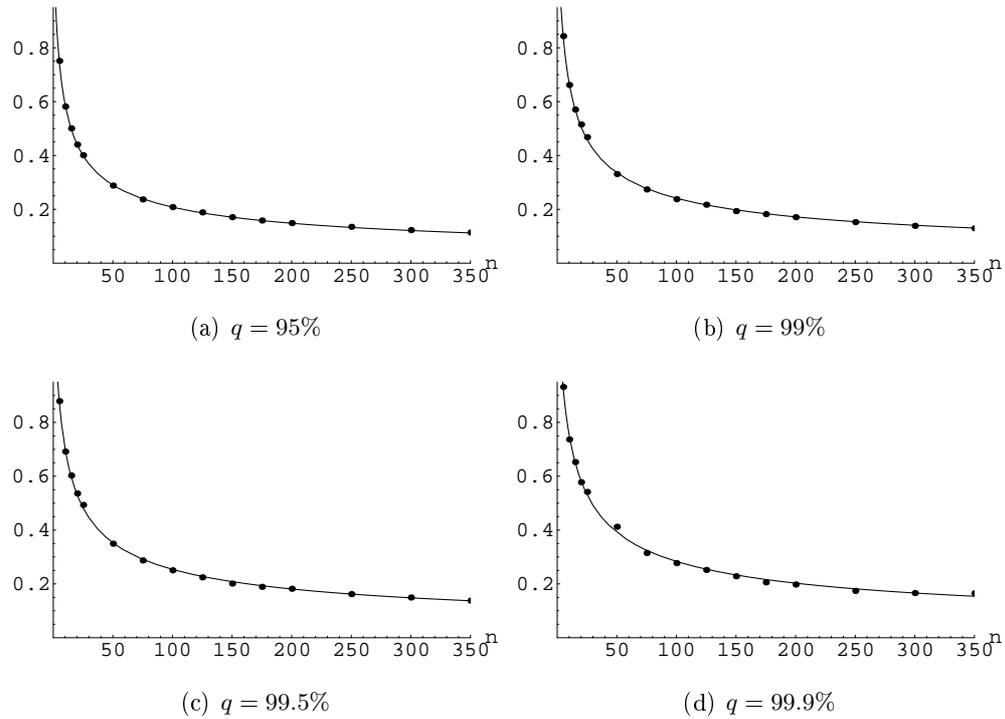
Figure 3.10: Squared inverses of certain empirical quantiles $ECDF_n^{-1}(q)$

Table 3.6: Parameter estimates for models of the form $\text{ECDF}_n^{-1}(q) = 1/\sqrt{An + B}$

	Quantile q (%)			
	95.0	99.0	99.5	99.9
B	0.657175	0.575433	0.545681	0.572809
A	0.224535	0.166313	0.149807	0.118794

Figure 3.11: Models obtained for highest empirical quantiles $\text{ECDF}_n^{-1}(q)$

Needless to say, the figures of Tables 3.3 and 3.5 re-evaluated using the models of the quantiles instead of those of the distributions showed clear improvement, as the former have been explicitly fitted to the desired quantiles themselves. Thus, these direct models make the accuracy of prediction better in terms of both probability and transmission range.

Chapter 4

Biconnectivity

In this chapter, the scope of the study is extended to the stronger requirement of biconnectivity. This is an important property in terms of network reliability and load balancing. A method for finding the threshold transmission range for biconnectivity is first presented. The statistical behavior of this random variable is then analyzed in the same way as that of the critical transmission range was before.

4.1 Fault tolerance and route diversity through range redundancy

After examining the requirement of simple network connectivity, the introduction of the following concepts is in order:

A connected network is *biconnected* if there is no single node whose removal would disconnect the network. In a biconnected network, at least two node-disjoint paths exist between every pair of nodes. In general, a network is *k-connected* if and only if at least k node-disjoint paths exist between every pair of nodes and it therefore remains connected after the removal of any $k - 1$ nodes.

In a connected but not biconnected network, the nodes whose removal would disconnect the network are known as *articulation points*.

Finally, the *degree* of a node is the number of nodes directly connected to it.

Obviously, articulation points are critical components in the network, and a network without such nodes is more fault tolerant than one containing such nodes. It is important to note the following:

Proposition 4.1 *A network with more than two nodes, operating with its critical transmission range R_{crit} always has at least one articulation point, namely, the endpoint(s) of the longest link in the MST whose degree is more than one.*

To be exact, this only holds provided that the MST is unique, which is not the case with e.g. the vertices of a regular polygon. However, when the nodes are randomly placed in continuous space, their MST is always unique because there are no equidistant node pairs.

Of course, nodes other than these endpoints may also be articulation points. This is demonstrated in Figure 4.1 showing the sample node set of Figure 3.1 with a link

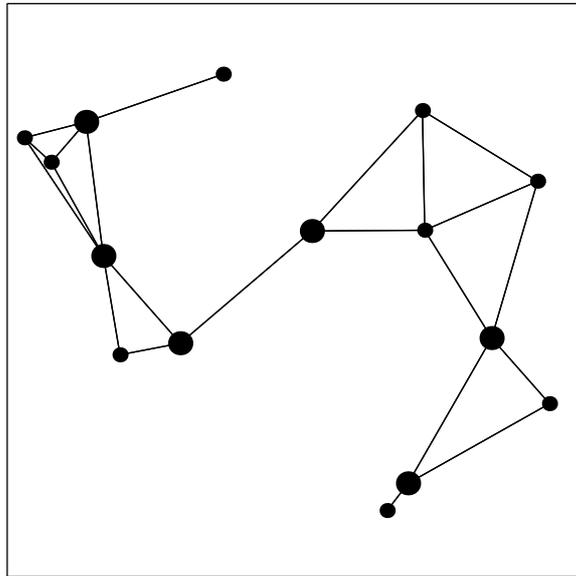


Figure 4.1: The network formed by the sample node set of Figure 3.1 with transmission range R_{crit} . Large dots depict articulation points.

drawn between all node pairs at most R_{crit} apart. It can also be seen here that in the resulting network, several node pairs lack node-disjoint paths between them. This yields another significant feature of articulation points in ad hoc networks: they are prone to becoming relay traffic bottlenecks whose both capacity and energy is exhausted under the constant demand to relay traffic between other node pairs.

4.2 Finding the transmission range required for biconnectivity

As stated in the definition above, a biconnected network is a connected network without articulation points. This yields an elementary method to find the threshold transmission range for biconnectivity - let this be denoted by $R_{\text{bicon}}(n)$: eliminate the possibility of each node being an articulation point by determining R_{crit} for each subset of nodes obtained by removing one of the n nodes; the greatest of these is R_{bicon} . In fact, the threshold range for k -connectivity is the greatest R_{crit} obtained after removing all possible subsets of $k - 1$ nodes in turn. Naturally, the computational complexity of this method is excessive, as n determinations of R_{crit} are needed even for R_{bicon} . In this section, a more efficient method is presented. The idea is to narrow down the set of articulation points to be eliminated by finding a lower-bound estimate for R_{bicon} .

4.2.1 Depth-first search

The articulation points of any connected network can be found with a graph traversal algorithm called the recursive *depth-first search* (DFS; there exists also a non-recursive version of this algorithm). The recursive implementation traverses all the nodes in the network in the following way: starting from any node,

1. Visit the node
2. Recursively visit (invoke DFS for) all the nodes adjacent to the present node that have not yet been visited.

The algorithm results in a *depth-first spanning tree* whose root is the starting node. The links in the original network that resulted in recursive visits correspond to the edges in the tree. Now, a node x is an articulation point if, from any subtree whose root is one of x 's children, there is no link (in the original network) to a node located above x in the tree. As an exception, since no link can lead above the root, it is an articulation point if it has more than one child.

As an example, Figure 4.2(a) shows the graph representing a random five-node network operating with R_{crit} ; the nodes have been named in the order in which their locations have been generated. The DFS tree that is formed when the algorithm is called for node A is depicted in Figure 4.2(b). Using the rules above, the articulation points are found to be nodes A and E , the former because it is the root of the tree and has two children, and the latter because there is no link in the graph leading above E in the tree from the (single-node) subtree rooted by E 's child D (such a link would be illustrated with a dashed line; compare with node B which would also be an articulation point without the link between C and A). Note that it follows from these rules that a leaf (a node with no children) in the DFS tree (C and D in this case) is never an articulation point.

The logic behind the name DFS can also be seen in Figure 4.2: the algorithm first progresses as far as possible in the network and then backs up searching for new branches.

The general rule for finding the articulation points can thus be implemented as follows:

- The nodes of the network are numbered in the order in which they are visited during the search.
- Every instance of the recursive DFS returns either the lowest number of a node

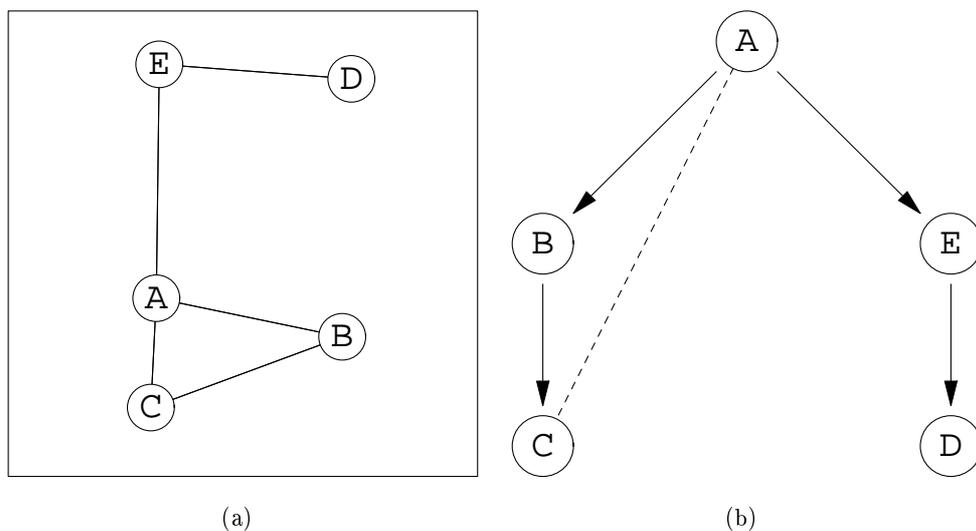


Figure 4.2: The graph (a) and a DFS tree (b) of a five-node network

adjacent to the present node or the smallest return value of the DFS instances invoked from within this instance, whichever is less.

Since every child of a given node in the resulting tree is adjacent to the node, a return value equal to k obtained from a DFS call to any child of node number k indicates that the node is an articulation point.

4.2.2 Incrementing the range gradually

Even though the recursive DFS can be used to find the articulation points from a given set of nodes with a given transmission range, i.e. it tells us whether or not the network is biconnected with the given range, it does not directly determine R_{bicon} for the set of nodes. Of course, the interval containing the right value could be made infinitely small by trial and error, in a fashion similar to several known numerical one-dimensional minimization algorithms, but no matter how many iterations - and how much computing time - used, this would never provide the exact solution.

To find R_{bicon} , a two-phase approach is used. In the first phase, a best-guess lower-bound estimate for the required range is produced by utilizing two necessary but not sufficient conditions for biconnectivity. The articulation points with this estimate are then found. The two conditions are:

1. The network is connected
2. Every node in the network is directly connected to at least two other nodes (or, the network has minimum degree 2).

The latter condition can be justified by noting that if a node is directly connected to only one other node then, under the assumption $n > 2$ without which the whole concept of biconnectivity would be irrelevant, the other node is an articulation point.

As these two conditions are also completely independent, that is to say, connectivity does not guarantee minimum degree of 2 and vice versa, the best-guess estimate used with each realization is the maximum of the threshold transmission ranges for these two conditions.

The strategy employed in finding the best estimate was motivated by the following theorem proved by Penrose in [7]:

Theorem 4.1 *For n points uniformly randomly distributed on the unit cube in d dimensions, with $d > 1$, let ρ_n (respectively σ_n) denote the minimum r at which the graph, obtained by adding an edge between each pair of points distant at most r apart, is k -connected (respectively, has minimum degree k). Then $P[\rho_n = \sigma_n] \rightarrow 1$ as $n \rightarrow \infty$.*

A few remarks about the theorem: minimum degree k is a weaker condition for r than k -connectivity, i.e. the inequality $\sigma_n \leq \rho_n$ always holds with fixed k . Also, ρ_n increases with k (a k -connected network is also $k-1$ -connected), but no relation can be made between σ_n with $k = K$ and ρ_n with $k < K$ (as pointed out above in the case $K = 2$). Of course, the case of interest here is $d = 2$. With these and earlier notations, the threshold transmission ranges for the two conditions mentioned above are $R_{\text{crit}}(n)$ and σ_n with $k = 2$, respectively. Note that also $R_{\text{crit}}(n)$ and $R_{\text{bicon}}(n)$ can be expressed as ρ_n with $k = 1$ and $k = 2$, respectively.

Experiments in the unit square showed that when $k = 1$, the probability in question remained low with reasonable n . However, in the case $k = 2$ the implied effect was much more visible: with the range σ_n , the network was very seldom disconnected and usually had clearly fewer - if any - articulation points than with the range $R_{\text{crit}}(n)$. Figure 4.3(a) shows how this tendency is obeyed also by the sample node set observed earlier.

What is encouraging is that σ_n is very easy to find: given k , i.e. the desired minimum degree, and the distance matrix of the points, one must only find the $k+1$ -th-smallest element of each row in the matrix (assuming the zero on the diagonal is included); the greatest of these is σ_n .

The above observations are why σ_n is always determined first from every realization. Whether or not this is greater than $R_{\text{crit}}(n)$ is then implicitly found out after running the recursive DFS to locate the articulation points with the range σ_n : if not every node is visited during the search, the network is not connected with σ_n , which leads to determining $R_{\text{crit}}(n)$ and the corresponding articulation points. But, as mentioned, this is very rare.

In the second phase, it remains to eliminate each articulation point found with the estimate for R_{bicon} . This could be done in the way described in the beginning of this section, but again a more efficient method is used: the subnetworks separated by the articulation point are first found. (This can be done by, for example, invoking the recursive DFS with the articulation point removed until all nodes have been visited. The resulting DFS trees are the subnetworks.) Next, the shortest range needed to connect these subnetworks is determined by, in effect, treating the subnetworks as single nodes in the Prim algorithm: starting with any subnetwork, merge it with the closest other subnetwork until all the subnetworks have been connected, recording the longest link distance needed in the process. The longest of these, in turn, when the same is repeated for every articulation point, is finally R_{bicon} . Figure 4.3(b) shows the sample node set with transmission range R_{bicon} .

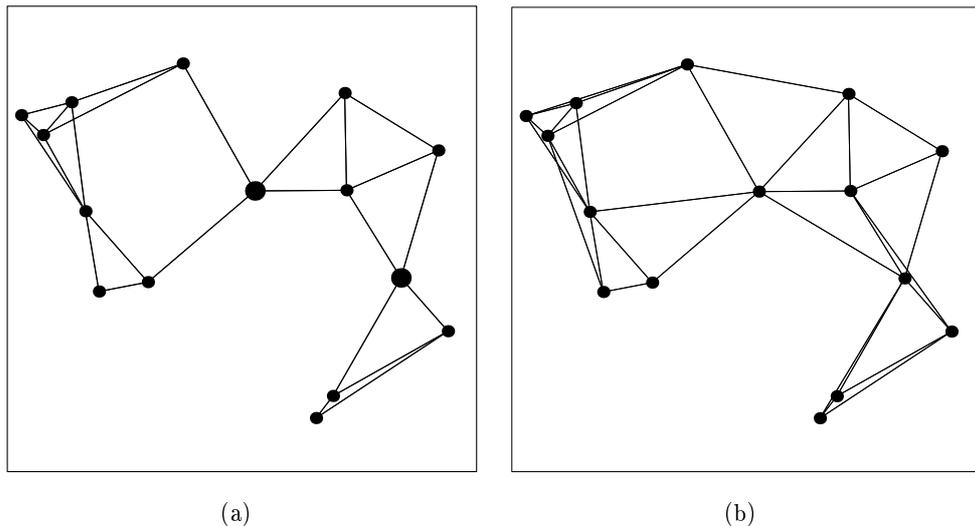


Figure 4.3: The sample node set with transmission range (a): σ_n ($k = 2$), (b): $R_{\text{bicon}}(n)$. Note in the former the number of articulation points in comparison to Figure 4.1.

4.2.3 Summary of the algorithm

To summarize, below are the steps taken to find R_{bicon} from a given set of nodes:

1. Find the threshold range for minimum degree 2.
2. Find the articulation points with this range by using recursive DFS.
 - If the network is found to be disconnected with this range, find R_{crit} and the articulation points with that range.
 - If the network is connected and no articulation points exist, the range found in step 1 is R_{bicon} .
3. For each articulation point, find the threshold range for connectivity with the articulation point removed. The longest of these is R_{bicon} .

It is worth mentioning that Penrose's theorem can also be utilized in finding $R_{\text{crit}}(n)$: first determine the threshold range for minimum degree 1 and the resulting connected subnetworks, then treat them as single nodes in the Prim algorithm as described above. In fact, this proved to be much faster than employing the Prim algorithm all the way. The reason is that this method enables us to, in effect, disregard the shorter links in the MST. The pseudocode representations of this algorithm as well as the one summarized above are given in Appendix A.

4.3 Statistical analysis

Data sets of the same size and for the same n as with $R_{\text{crit}}(n)$ were gathered from $R_{\text{bicon}}(n)$ samples. Figure 4.4(a) shows the squared inverses of the means and Figure 4.4(b) the inverses of the sample variances of these data sets. It is evident that they show similar dependence on n as those of $R_{\text{crit}}(n)$. This is easy to accept by intuition: after all, R_{bicon} is just R_{crit} for a certain subset of $n - 1$ nodes. It can therefore be thought of as some kind of a "conditional" R_{crit} , which explains the similar behavior.

The parameter estimates from weighted linear regressions for the means and sample variances are shown in Table 4.1, making the models approximately $E[R_{\text{bicon}}(n)] = 1/\sqrt{n/4 + 1}$ and $\text{Var}[R_{\text{bicon}}(n)] = 1/(12.8n - 13.5)$. Again, for $n > 1$, the latter

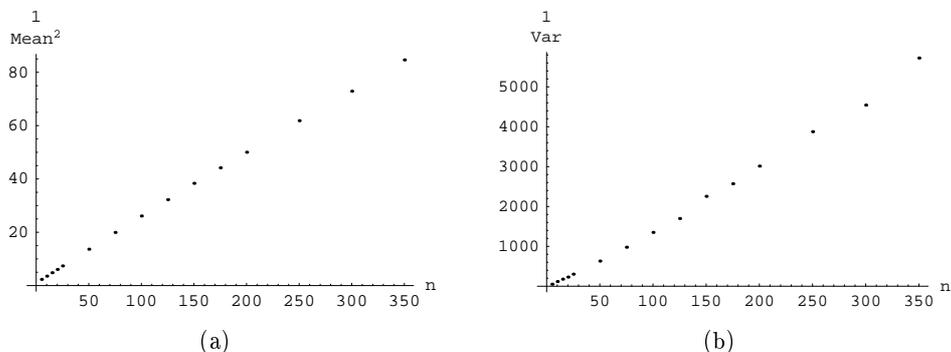


Figure 4.4: The dependence of (a): the means and (b): the sample variances of R_{bicon} data sets on n

obtains positive and finite values. The curves of the models are shown in Figure 4.5.

Table 4.1: Parameter estimates for the models of $E[R_{\text{bicon}}(n)]$ and $\text{Var}[R_{\text{bicon}}(n)]$

	$1/E[R_{\text{bicon}}(n)]^2$	$1/\text{Var}[R_{\text{bicon}}(n)]$
1	1.07933	-13.4706
x	0.24866	12.7928

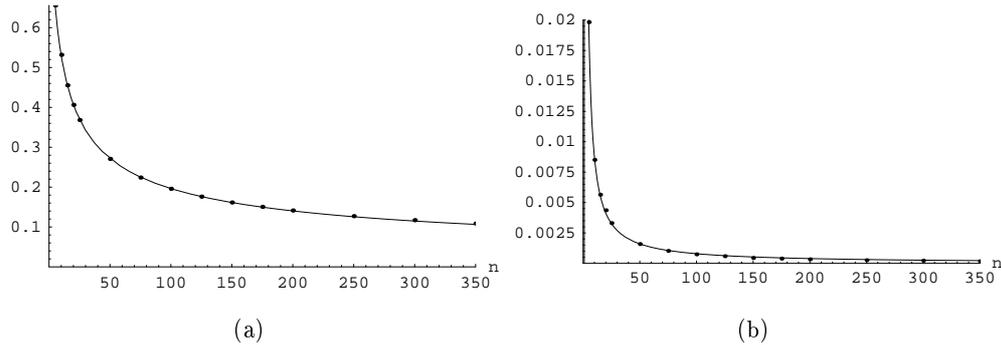


Figure 4.5: Models obtained for (a): $E[R_{\text{bicon}}(n)]$ and (b): $\text{Var}[R_{\text{bicon}}(n)]$

Because the models differ from those obtained for $R_{\text{crit}}(n)$ only in parameters, not in forms, the rules (3.4) and (3.3) for fitting Beta distributions of the form (3.2) to the data are applicable here as well. Figure 4.6 shows that the trend in how well the distribution fits is the same as with $R_{\text{crit}}(n)$. In particular, although the fitting is done by matching only the expected value and the variance, the distribution is in agreement with the data when $n = 5$ even in terms of the reversed skewness. This is the benefit of using a limited-interval distribution for the approximation.

As shown by Figure 4.8, also the empirical quantiles behave as before. The results of the weighted regression are presented in Table 4.2 and the curves of the models in Figure 4.7.

Table 4.2: Parameter estimates for modelling $1/\text{ECDF}_n^{-1}(q)^2$

	Quantile q (%)			
	95.0	99.0	99.5	99.9
1	0.495801	0.469256	0.47827	0.431728
x	0.157316	0.121966	0.110161	0.0944855

Table 4.3 shows the relative differences between the highest empirical quantiles of $R_{\text{bicon}}(n)$ and $R_{\text{crit}}(n)$. It can be deduced that when the requirement of biconnectivity is added on top of that of simple connectivity, the transmission range must be increased by roughly 15-20% in order to satisfy that requirement with the same probability. Again, the highest quantile should be regarded with caution due to sample size effects.

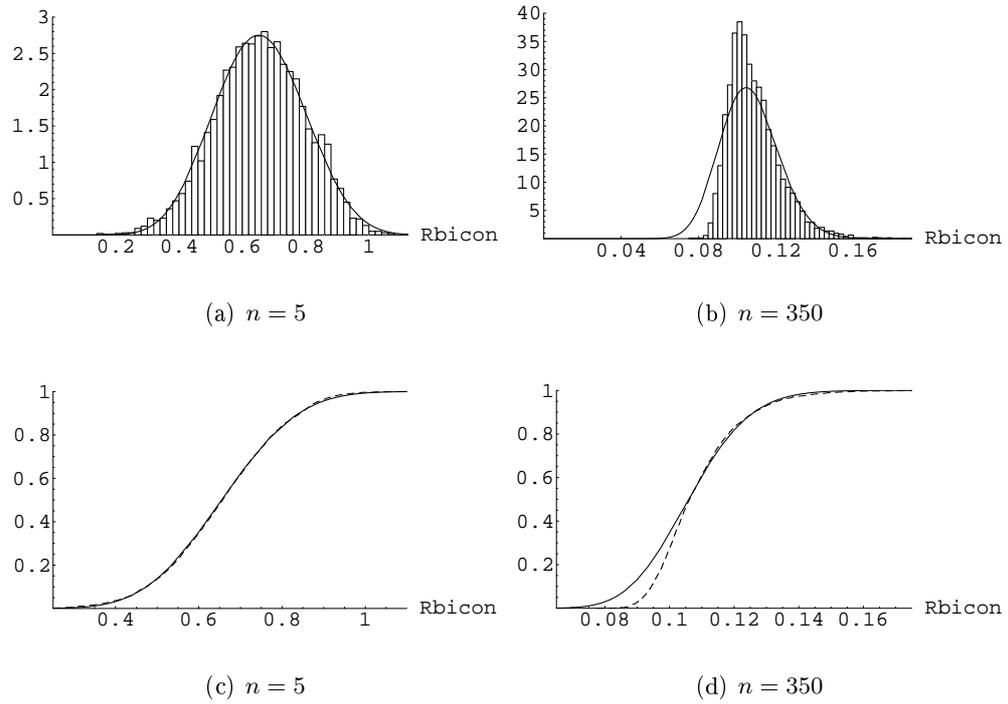


Figure 4.6: Empirical distributions of $R_{\text{bicon}}(n)$ and fitted Beta distributions (solid lines)

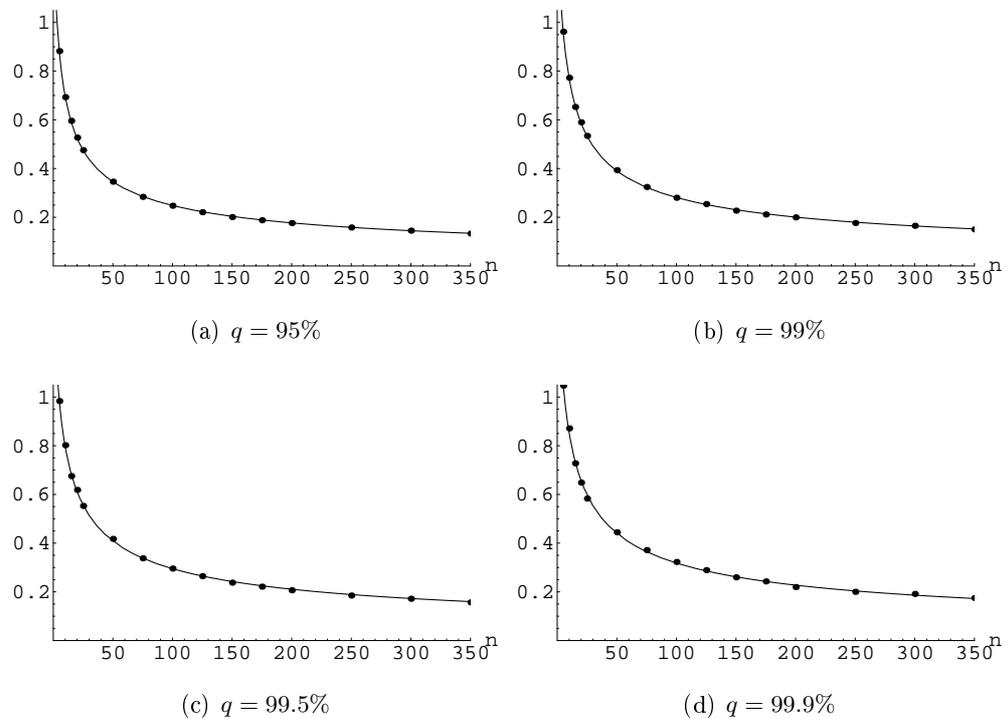
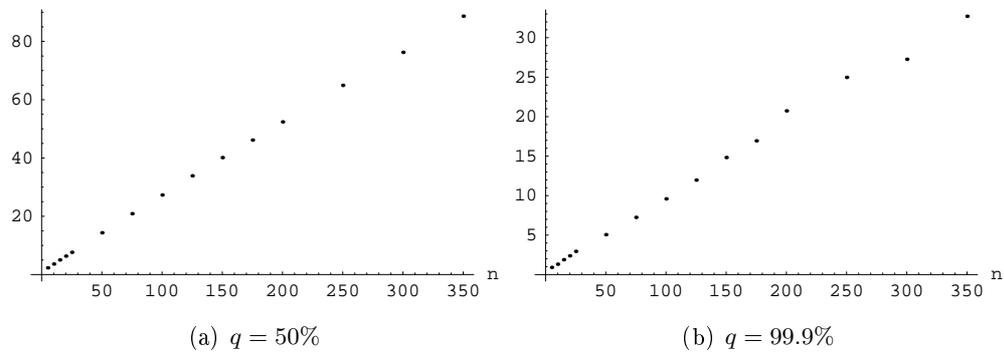


Figure 4.7: Models obtained for highest empirical quantiles $\text{ECDF}_n^{-1}(q)$

Figure 4.8: Squared inverses of $\text{ECDF}_n^{-1}(q)$ plotted against n Table 4.3: Relative differences between empirical quantiles of $R_{\text{bicon}}(n)$ and $R_{\text{crit}}(n)$

n	Empirical quantile (%)			
	95.0	99.0	99.5	99.9
	Relative difference (%)			
5	17.4	14.1	11.9	12.3
10	19.2	16.7	16.0	18.3
15	19.0	14.5	12.1	11.5
20	19.7	14.5	15.4	12.3
25	18.6	14.1	12.1	7.7
50	20.1	18.8	19.5	7.8
75	19.9	18.4	17.7	18.1
100	18.8	17.9	18.0	16.5
125	17.0	17.1	17.9	14.9
150	17.6	17.5	18.1	13.7
175	18.6	16.2	17.1	18.1
200	18.5	16.9	13.6	11.0
250	17.1	15.7	14.2	15.0
300	18.1	19.0	15.0	15.3
350	17.0	16.3	13.7	5.8

Chapter 5

Triconnectivity

This chapter broadens the view further on triconnectivity. It first shows that for random networks, every increment in the degree of connectivity always requires a greater transmission range. It then proceeds like the previous chapters, by presenting first an algorithm for finding the threshold range for triconnectivity and then statistical analysis of simulation results.

5.1 Introduction

Analogous to the term biconnectivity, 3-connected networks are commonly referred to as *triconnected*. In this context, the notion of articulation points is extended to *articulation pairs*: the removal of any such pair of nodes from a biconnected network will disconnect the network. Also, the load imposed by relay traffic on an articulation pair is likely to be high.

It was pointed out in the context of biconnectivity that under the assumption of randomly placed nodes, a network operating with its threshold transmission range for connectivity always has at least one articulation point. This can be generalized to all degrees of connectivity:

Proposition 5.1 *A network of at least $k + 2$ nodes randomly placed in continuous space operating with its threshold transmission range for k -connectivity always has at least one articulation set of k nodes.*

Proof: Consider the process of increasing the transmission range among the set of nodes: as the range increases, new links between node pairs are enabled. At some threshold range, the network becomes k -connected as a result of some critical link forming, i.e. there is no longer an articulation set of $k - 1$ nodes in the network whose removal would disconnect the network into two connected components, as the critical link connects these components. (Because of the random node locations, this critical link is unique.)

Since the number of nodes in the network is assumed to be at least $k + 2$, thus making an articulation set of k nodes a sensible concept, at least one of the aforementioned components would consist of more than one node. Assume now that a critical link endpoint representing such a component is removed. The critical link then ceases to exist, thereby bringing the articulation set of $k - 1$ nodes back into effect: the

removal of this set would again disconnect the network, only now with the removed endpoint missing from one of the connected components. Therefore, this endpoint together with the articulation set of $k - 1$ nodes constitutes an articulation set of k nodes at the threshold range for k -connectivity. \square

It is easy to see that in the case $k = 1$, the critical link is the longest link in the MST and, as stated earlier, at least one of its endpoints is an articulation point at the threshold range for connectivity. An example of the case $k = 2$ is illustrated in Figure 5.1 showing the critical link and the articulation point it eliminates (or, the "last" articulation point) in the sample node set used earlier, as well as the four articulation pairs at the critical range for biconnectivity. It can be seen that in this case both the endpoints of the critical link have the property inferred in the above proof, i.e. they both make up an articulation pair with the last articulation point. However, it is notable that there are also articulation pairs completely independent of the critical link and the last articulation point. The conclusion is that the combination of the last articulation set of $k - 1$ nodes and at least one of the critical link endpoints is a sufficient but not necessary condition for an articulation set of k nodes at the critical range.

The consequence of the result proved is that in random networks, no two degrees of connectivity are achieved with the same threshold range. Note that the assumption of distinct pairwise node distances is essential here: consider as a counterexample the vertices of a regular pentagon. This network achieves connectivity and biconnectivity with the same transmission range (when the links forming the sides of the pentagon are enabled), as well as triconnectivity and 4-connectivity (when the links between the rest of the node pairs are enabled).

Every increment in the required level of network reliability thus implies a need for a greater transmission range.

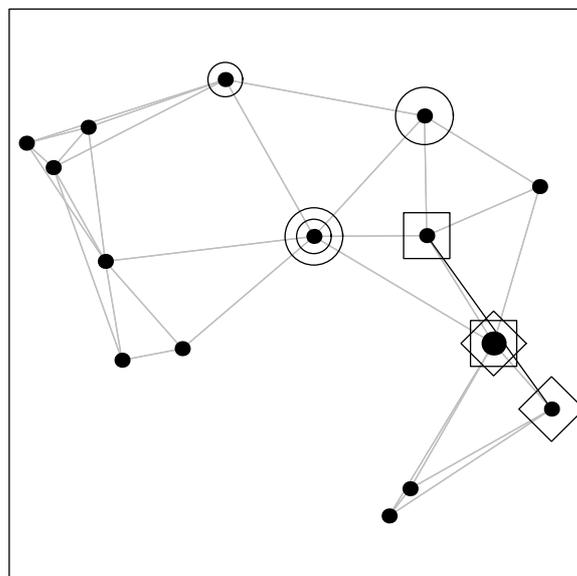


Figure 5.1: The sample node set with transmission range $R_{\text{bicon}}(n)$. The critical link is shown with a dark line and the articulation point eliminated by that link with a large dot. Each articulation pair is marked with a distinct symbol.

5.2 Finding the transmission range required for triconnectivity

The threshold transmission range for triconnectivity, or $R_{\text{tricon}}(n)$, can be found with the same strategy as in the case of biconnectivity: find a lower-bound estimate for the range (in this case, the greater one of the threshold ranges for minimum degree 3 and R_{bicon}), determine all articulation pairs with that range, and finally eliminate the articulation pairs.

The demanding task here is to efficiently find all the articulation pairs. As shown by Figure 5.1, not all of them are associated with the critical link or the last articulation point, so all node pairs have to be taken into account. The straightforward method of finding out which of the $n(n-1)/2$ node pair removals disconnect the network is again too inefficient. Another way is to run the DFS with each of the n nodes of the biconnected network removed in turn: an articulation point found in a DFS tree forms an articulation pair with the node removed for that tree. (To be exact, the last run is redundant, so only $n-1$ runs are needed.)

There is a known data structure in graph theory called the SPQR-tree which allows the decomposition of a biconnected graph into its triconnected components in linear time [12]. Its definition is however quite complex, which makes its implementation even more so. This conception is supported by the fact that a brief search yielded only two instances that had reported to have implemented the SPQR-tree, both for the purposes of graph drawing applications, and both of which claiming to have made the only implementation known so far. As, in addition, neither implementation was in unrestricted distribution, another algorithm based on the DFS was developed specifically for the purposes of this study.

The method used here is simple and, apart from the SPQR-tree, more efficient than the ones mentioned above. It is based on storing the DFS tree of the biconnected network (obtained with the lower-bound range estimate). At each of the $n-1$ node removals, the unaffected part of the tree is preserved and only the part altered by the node removal is reformed. Consider as a simple example the removal of node C from the DFS tree of Figure 5.2(b). It can be seen that the rest of the tree remains unaffected, so no part of the tree remains to be rebuilt. Furthermore, the remaining tree yields no articulation points, meaning that node C is not part of any articulation pair. (This can easily be verified by looking at the graph of the network in Figure 5.2(a).)

5.2.1 Storing the DFS tree

The implementation of this method requires that the DFS tree of the biconnected network is stored in sufficient detail. The nodes of the tree are stored in a list in the order in which they are visited during the DFS. In addition to this, for each node the shortest paths (in terms of hop count) starting with each child of the node and leading above the node in the tree are stored. In the case of multiple paths with equal length, the one whose endpoint is located lowest in the tree is selected, for reasons becoming evident shortly. (Recall that such paths serve as the justifications for the node not being an articulation point.) No paths are however stored for the root, it being the topmost node in the tree, or for the leaves, for they have no children. Table 5.1 shows how the tree of Figure 5.2(b) is stored. It can be seen that even though

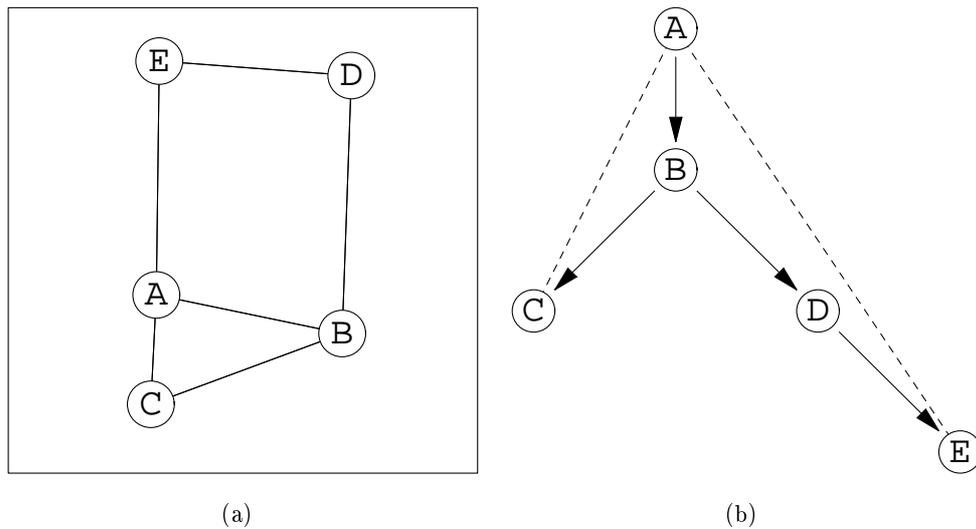


Figure 5.2: The graph (a) and a DFS tree (b) of the five nodes of Figure 4.2 operating at R_{bicon}

Table 5.1: The data structures used to store the DFS tree of Figure 5.2(b)

<code>tree</code>	{ A , B , C , D , E }
<code>paths</code>	{ {}, {{C,A}}, {{D,E,A}}, {}, {{E,A}}, {} }

the mere list of nodes in the tree is insufficient to fully describe the tree, the paths are enough to remedy this: for example, the two together show that node B has two subtrees, one consisting of node C only and the other one of nodes D and E ; these are the separate sequences in the list `tree` that start with node B 's children. Also, nodes C and E are known to be leaves because no paths are stored for them, and node A , being the first node in the list, is the root with node B , the second node, as its only child (because the network is biconnected). It should be emphasized that even though in this case all the links in the network (all the dashed lines in Figure 5.2(b)) can be inferred from the stored data, this is not usually the case: only the shortest paths are stored, as described above.

5.2.2 Finding articulation pairs

Given the node removed from the network, the DFS tree is preserved from the top down to the first node affected by the removal. Such nodes are the ones with the removed node specified as other than the first node - the child - in the paths stored for them, because we no longer know whether a path leading above such a node in the tree exists via every child of that node. Also, the endpoints of the paths stored for the removed node are affected, since they are the nodes located above the removed node that are known to be connected to the subtrees under the removed node. This justifies the criteria for the selection of the paths to be stored: the aim is to reduce the chances of regarding nodes located high up in the tree as affected by the removal of a node located far down.

When the first affected node is found, it is added to the tree (marked as visited) together with all, if any, subtrees rooted by its children that do not contain the

removed node (note that the subtrees rooted by the children of a given node are not connected below the node and are therefore completely independent). Finally, DFS is called for the affected node to complete the DFS tree. The main idea is that any articulation points will be found by the DFS because the preserved tree (excluding the first affected node) is known not to contain any.

There are two special cases of the removed node as to where the reconstruction of the tree must be initiated. The first one, the root of the tree, results in calling DFS for its child and, in effect, rebuilding the whole tree. (In fact, the same can be said when the child of the root is removed, but here the first affected node - the root - is found as the endpoint of the paths stored for the child, so the general rule given above applies.) The second case is a leaf that is only specified in the paths stored for its parent: in this case, the rest of the tree is preserved completely yielding no articulation points, as was the case with node C in Figure 5.2(b).

Figure 5.3 illustrates two examples of this method. When node E is removed from the sample network (a), the highest-located node affected by the removal is node B because, as shown by Table 5.1, E has been recorded in one of its paths. The tree can therefore be preserved from the root down to node B , with the addition of the unaffected subtree consisting of node C (b). The tree is completed by initiating the DFS from node B ; in the new tree (c), node B is found to be an articulation point. Therefore, nodes B and E form an articulation pair in the network.

In the second example, node D is removed (d). This time the root of the tree, node A , is the endpoint of the path stored for D , so no part of the tree other than the root is preserved. When the tree is built anew (e), the root obtains a second child and is

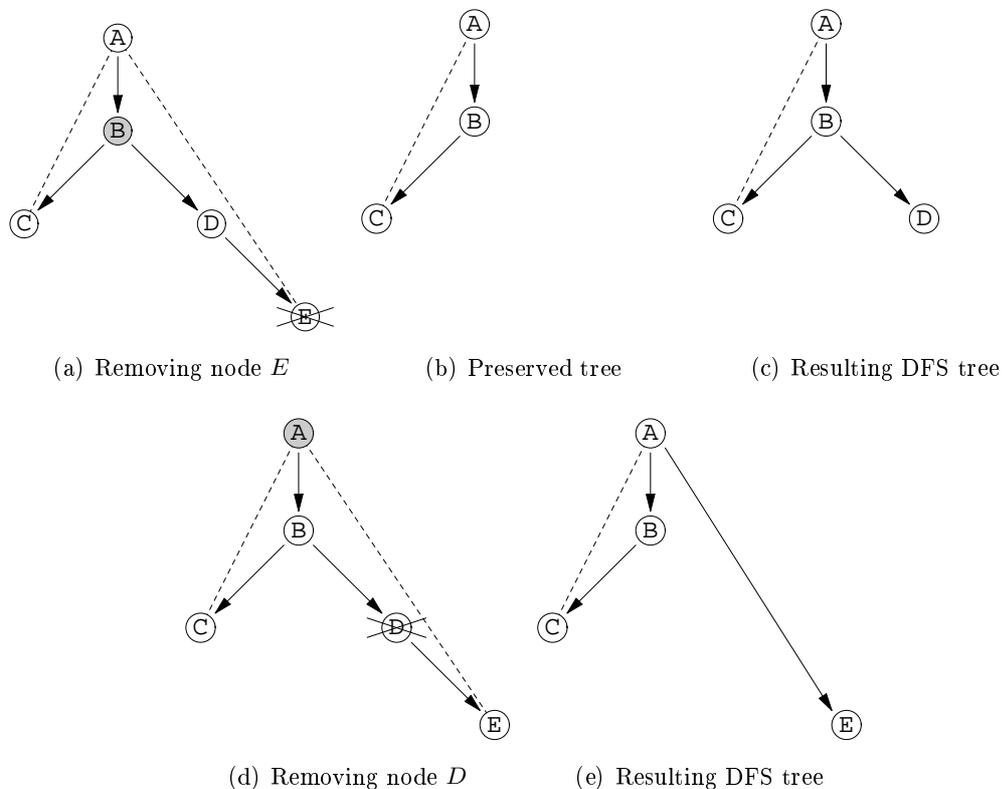


Figure 5.3: Examples of finding articulation pairs from the network of Figure 5.2(a)

therefore an articulation point. Thus, nodes A and D are another articulation pair. Both of these results can be verified with the aid of Figure 5.2(a). The remaining articulation pair consisting of nodes A and B could be found by removing either one of them and rebuilding the DFS tree. In fact, as was the case in the first example above, the tree resulting from the removal of node A is obtained from the original tree simply by erasing the removed node and all its connections. In the remaining tree, the root B has two children, making it an articulation point as expected.

5.2.3 Efficiency improvement

It could be observed above that the stored DFS tree is of little or no use when the removed node is close to the root in the tree. This can be alleviated by utilizing the fact that the DFS tree obtained from a given network is not unique. For instance, any node of the network can be chosen as the root: Figure 5.4(a) shows the resulting tree when the DFS is called for node E of the sample network. Now, consider the removal of node D from this tree. The highest-located affected node is node A , so two nodes of this tree can be preserved instead of only one which was the case with the tree used before.

Nevertheless, this tree turns out to be very similar to that in Figure 5.2(b): in effect, every node has been shifted down one level, with the exception of the new root. Changing the root has therefore had little impact on the distances of nodes from the root in the tree. Fortunately, there is another way to vary the outcome of the DFS, namely, the order in which the adjacency list of each node is processed by the algorithm. As an example, Figure 5.4(b) shows the tree obtained when the adjacency lists are processed in the reverse alphabetical order, i.e. of all unvisited nodes in the list, the DFS is recursively called for the node standing last in the alphabet. It can be seen that this, combined with changing the root, results in a tree totally different

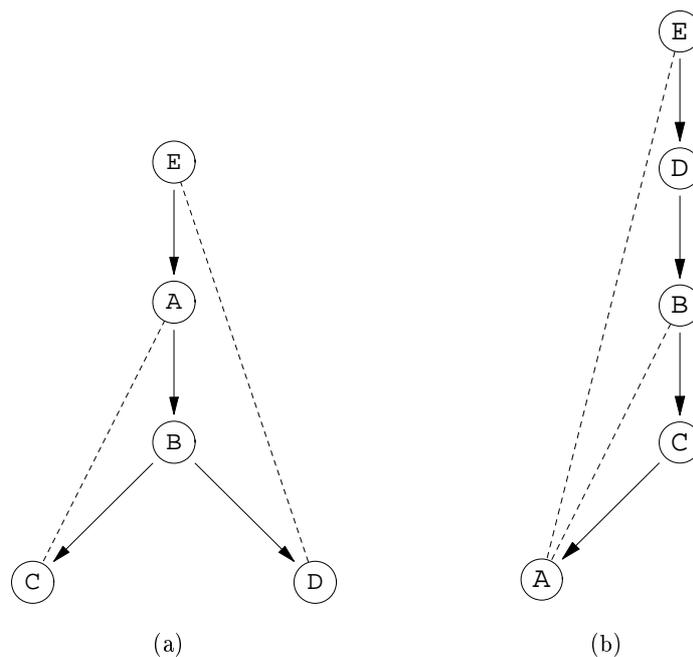


Figure 5.4: Different DFS trees obtained from the graph of Figure 5.2(a)

from the one in Figure 5.2(b). In particular, nodes close to the root in one tree are located further from the root in the other.

Thus, finding the articulation points after removing a certain node can be much more efficient when using one tree than when using another. (Actually, the sample network observed here demonstrates this rather poorly because it consists of only five nodes, but with a large number of nodes this becomes increasingly evident.) This is why several DFS trees are computed and stored from each network before starting to consider any node removals. As nodes are identified with an index range from 1 to n in the implementation of the algorithm, the different trees are obtained by applying the reverse order of the one-dimensional distance of the node index from a given number k in the DFS and starting the DFS from the node with the largest distance from k (that is, from either 1 or n), with several k distributed evenly between 1 and n . The intention of this is to have for each node a tree where the node is located far from the root. Finally, given the node removed from the network, it is first checked whether in any of the stored trees the removed node is a leaf only specified in the paths stored for its parent; in this case, the node is known not to belong to any articulation pair, as pointed out before. Otherwise, the tree in which the highest affected node is located lowest (or, to be exact, appears latest in the list of nodes, which does not necessarily indicate the node's distance from the root) is chosen.

Computationally, this proves to be well worth the effort. For example, with 350-node networks it was still advantageous to increase the number of trees used to nine, which resulted in the time needed to find all the articulation pairs decreasing by over 35%, compared to using one tree only. Overall, utilizing the stored DFS trees reduced the time required to find the articulation pairs by over 70% with 350-node networks, in comparison to building all $n - 1$ trees from scratch.

5.2.4 Summary of the algorithm

In a nutshell, the algorithm used to find R_{tricon} is as follows:

1. Find the threshold range for minimum degree 3.
2. Build the DFS tree using this range. If the network is found to be disconnected or to contain any articulation points, follow the steps needed to find R_{bicon} and rebuild the tree.
3. Store the tree, along with a number of other trees built using the current range estimate.
4. Find all articulation pairs with the current range estimate by removing $n - 1$ nodes from the network in turn and finding the resulting articulation points using the most suitable tree stored.
5. If at this point no articulation pairs are found, the threshold range for minimum degree 3 has turned out to be R_{tricon} . Otherwise, eliminate the articulation pairs by determining the threshold range for connectivity with each articulation pair removed in turn. The longest of these is R_{tricon} .

Figure 5.5 shows the sample node set observed earlier with the threshold range for minimum degree 3 and $R_{\text{tricon}}(n)$. Note that with this particular node set, the former range does not even provide biconnectivity.

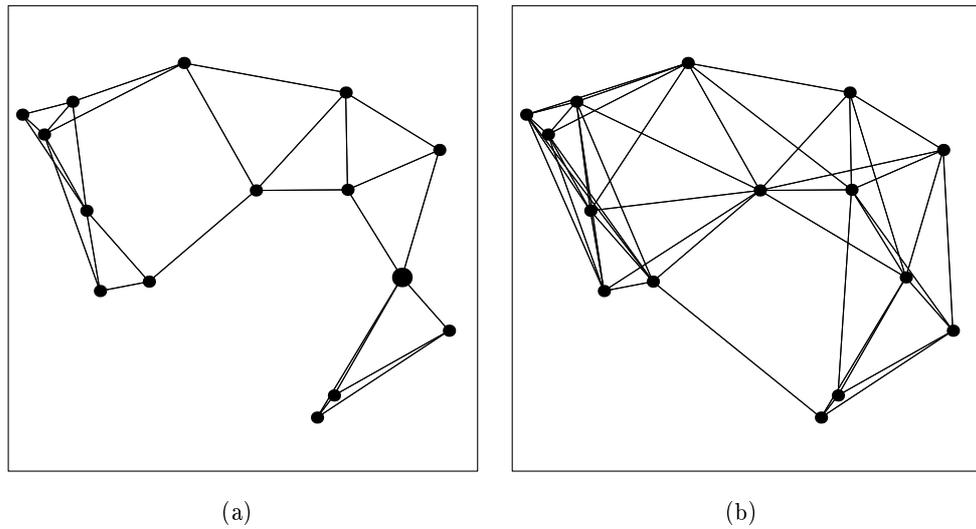


Figure 5.5: The sample node set with threshold range for minimum degree 3 (a) and triconnectivity (b)

5.3 Statistical analysis

Table 5.2 shows the weighted linear regression results for the means and sample variances of $R_{\text{tricon}}(n)$. The models obtained are approximately $E[R_{\text{tricon}}(n)] = 1/\sqrt{n/5 + 3/4}$ and $\text{Var}[R_{\text{tricon}}(n)] = 1/(12.0n - 14.6)$. The agreement of the models with the data is demonstrated in Figure 5.6.

Table 5.2: Parameter estimates for the models of $E[R_{\text{tricon}}(n)]$ and $\text{Var}[R_{\text{tricon}}(n)]$

	$1/E[R_{\text{tricon}}(n)]^2$	$1/\text{Var}[R_{\text{tricon}}(n)]$
1	0.7356	-14.6367
x	0.192681	11.9986

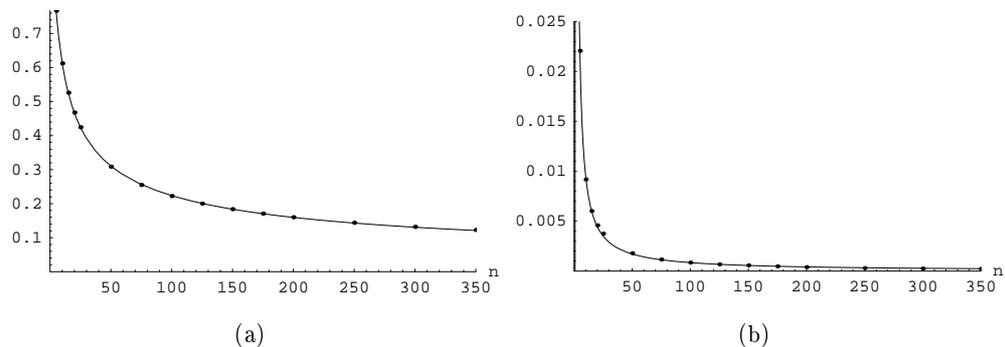


Figure 5.6: Models obtained for (a): $E[R_{\text{tricon}}(n)]$ and (b): $\text{Var}[R_{\text{tricon}}(n)]$

Beta distributions fitted using these models are depicted in Figure 5.7. The new observation to be made is the slightly visible deviation of the data from the fitted distribution even in the case $n = 5$.

The parameter estimates for the models of the highest empirical quantiles are pre-

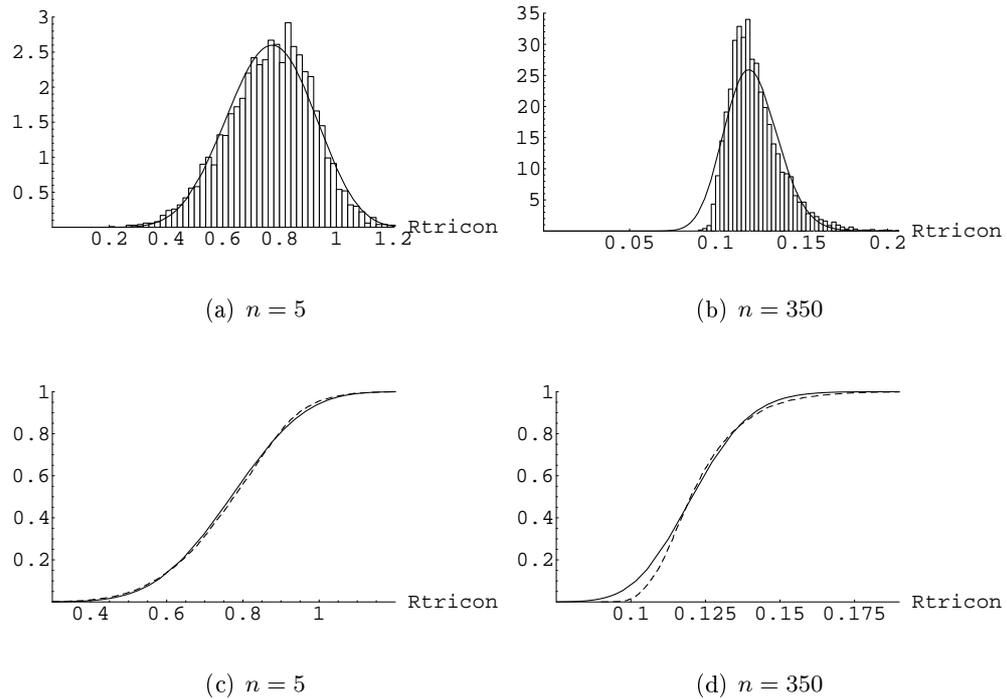


Figure 5.7: Empirical and fitted Beta distributions (solid lines) for $R_{\text{tricon}}(n)$

sented in Table 5.3, and the fit of the models is shown in Figure 5.8. It should be noted that even though the value predicted by the last model exceeds $\sqrt{2}$ when $n = 2$, this is not the case with $n > 3$, i.e. with node numbers relevant in the context of triconnectivity.

Table 5.4 shows that the required increase in the transmission range is roughly 30% from the corresponding quantile of $R_{\text{crit}}(n)$ when triconnectivity is to be achieved. In Table 5.5, the increase has been expressed in proportion to that needed for biconnectivity. The fact that the majority of these ratios is less than two suggests that the "cost", in terms of transmission range, per increment in the required degree of connectivity decreases with the degree, i.e. the cost function is concave. This is consistent with the fact that the threshold transmission range for k -connectivity when k (and the number of nodes) tends to infinity is limited by the dimensions of the area - $\sqrt{2}$ in this case - and cannot therefore exceed that limit.

On the other hand, based on the observations from Tables 4.3 and 5.4, it could be argued that the increase in the squared range, which can be interpreted to represent the coverage area of a node or, in some cases, the required transmission power, is close to constant in the gradual transitions to 2- and 3-connectivity.

Table 5.3: Parameter estimates for modelling $1/\text{ECDF}_n^{-1}(q)^2$

	Quantile q (%)			
	95.0	99.0	99.5	99.9
1	0.389088	0.352953	0.342611	0.34069
x	0.124881	0.100613	0.0920022	0.0777993

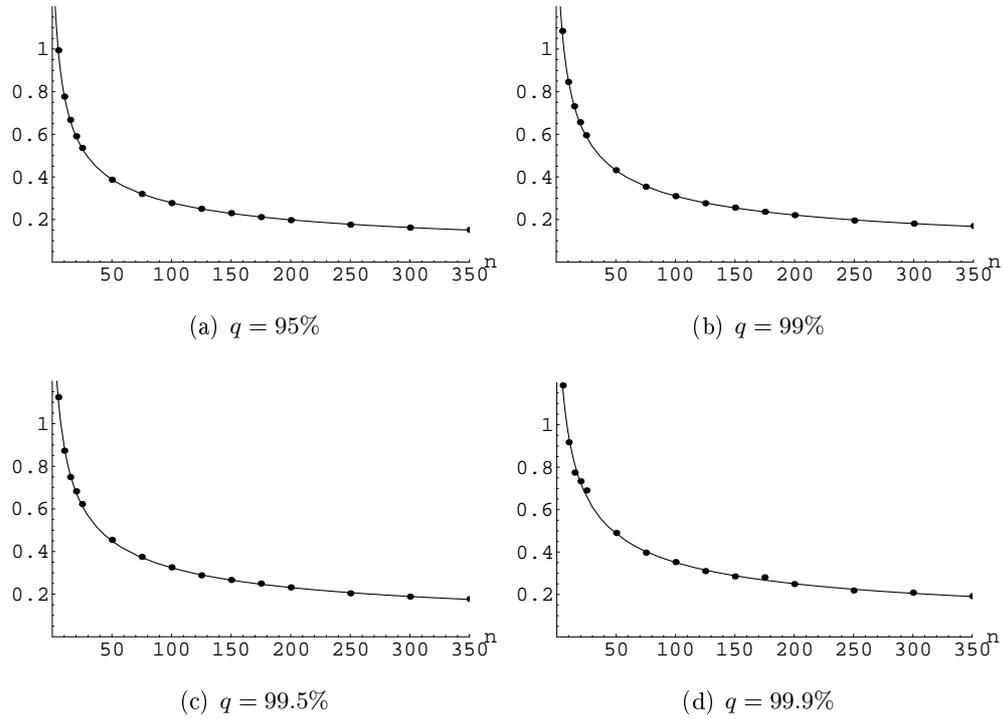


Figure 5.8: Models obtained for highest empirical quantiles $ECDF_n^{-1}(q)$

Table 5.4: Relative differences between empirical quantiles of $R_{tricon}(n)$ and $R_{crit}(n)$

n	Empirical quantile (%)			
	95.0	99.0	99.5	99.9
	Relative difference (%)			
5	32.3	28.6	27.9	27.2
10	33.6	27.7	26.3	24.6
15	33.4	28.2	24.3	18.7
20	34.1	27.5	27.4	27.0
25	33.7	27.3	26.2	27.5
50	34.2	30.4	30.1	18.9
75	35.2	29.3	30.6	26.2
100	33.1	30.3	30.1	27.3
125	32.7	27.3	28.5	23.3
150	34.7	32.4	32.4	24.9
175	33.6	29.9	32.1	36.5
200	32.6	29.2	27.4	26.0
250	30.3	28.1	25.7	25.7
300	31.9	30.6	26.0	25.7
350	33.0	31.3	28.5	16.1

Table 5.5: Ratios of the figures in Table 5.4 to those in Table 4.3

n	Empirical quantile (%)			
	95.0	99.0	99.5	99.9
	Ratio			
5	1.85	2.04	2.35	2.20
10	1.75	1.66	1.64	1.34
15	1.75	1.95	2.02	1.62
20	1.74	1.89	1.78	2.20
25	1.81	1.93	2.18	3.60
50	1.70	1.61	1.54	2.41
75	1.77	1.59	1.73	1.45
100	1.76	1.70	1.67	1.66
125	1.92	1.60	1.59	1.57
150	1.97	1.85	1.79	1.82
175	1.81	1.85	1.88	2.02
200	1.76	1.72	2.02	2.36
250	1.77	1.79	1.82	1.72
300	1.76	1.60	1.74	1.68
350	1.95	1.92	2.07	2.77

Chapter 6

Discussion and conclusions

In this chapter, the statistical models obtained in the previous chapters are verified against findings from previous studies presented in chapter 2. This is done with the aid of both simulation data and analytical results. Diagnostics are then carried out which show that the form of the models requires adjustment. Finally, some conclusions are presented.

6.1 Application of the models

In the previous sections, simulations have been performed with a varying number of nodes randomly distributed in the unit square. The required ranges have been seen to fit models of the form $r = 1/\sqrt{cn + d}$ where c and d are the parameters of the model. Because the unit in these models is the side of the square, they can be generalized as $r/\sqrt{A} = 1/\sqrt{cn + d}$, or

$$r = \sqrt{\frac{A}{cn + d}} \quad (6.1)$$

where A is the area of the square region of interest. The model therefore binds together three quantities: the largest area allowed, the required number of nodes, and the required transmission range. Given any two of these, the third can be predicted.

To show how this can be done, the results for simple connectivity are applied to an example scenario simulated and used for verification in [8]. The problem setting is quoted below:

"Example (Design of a large-scale wireless sensor network): A wireless sensor network should cover an area of size $A = 500 \times 500$ m². Since all sensors exchange information, e.g. for environmental monitoring, the network should be connected. The sensors are equipped with transceivers that transmit a range of $r_0 = 20$ m in free space and do not perform power control. How many sensors do we need to distribute over the area?"

Figure 6.1 is an excerpt from [8] depicting the simulation results as well as the analytical approximation (2.1) derived therein. It shows that according to (2.1), 2500 sensors would provide a connected network with a probability of roughly 99%. However, it can be seen that only about 82% of the simulated random topologies with 2500 nodes were connected.

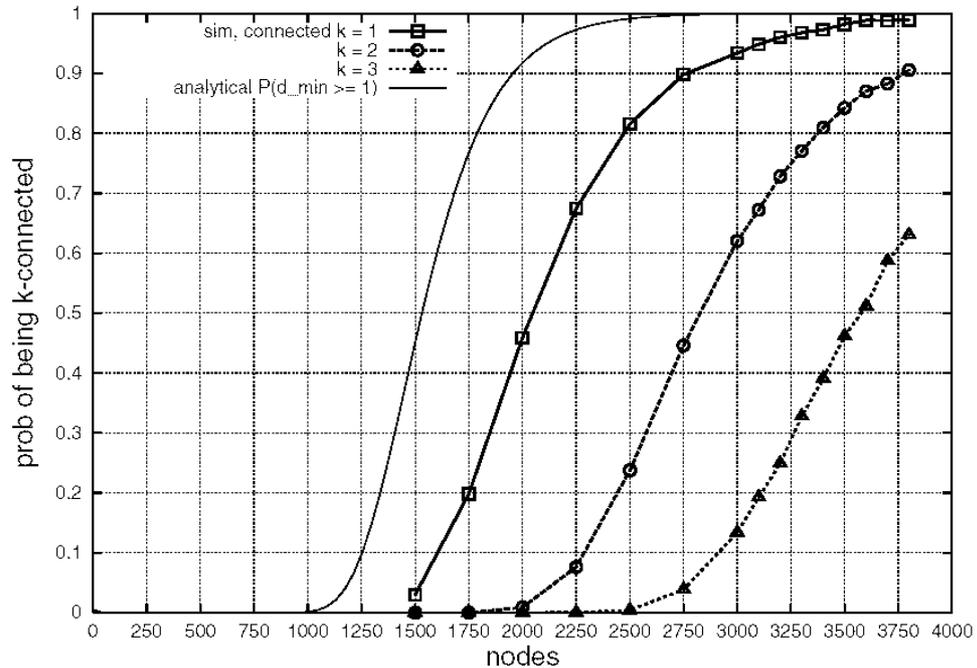


Figure 6.1: Simulation results for n nodes with $r_0 = 20\text{m}$ uniformly distributed on $A = 500 \times 500\text{m}^2$, 3000 random topologies [8]

Solving for n in (6.1) and applying the parameters obtained for the 99% quantile in Table 3.6 yields $n = 3755$. Similarly, the prediction obtained for the 95% quantile is 2781. Looking at Figure 6.1, the former seems to be a very good estimate, but according to the simulation results, the latter would have resulted in a connectivity probability of only about 90%. It can further be seen that approximately 3100 nodes were needed to achieve the probability of 95%. The error may partly result from the bold extrapolation in the prediction: simulation results involving no more than 350 nodes were used to fit the model, and the prediction is almost eight times that much. Bearing this in mind, the accuracy of the prediction for the 99% quantile is in fact quite surprising.

6.2 Asymptotic examination

Increasing the number of nodes in the simulations is intuitively interpreted as increasing the node density in a constant region. It can also be interpreted as increasing the area while keeping the node density constant. For instance, in comparison to the simulations with only five nodes in the area, having 350 nodes can represent not only 70 times the nodes in that area, but also the same density as in the five-node case, applied to 70 times the original area. The effect of the latter can be seen by writing (6.1) in the form $r = 1/\sqrt{cn/A + d/A} = 1/\sqrt{c\lambda + d/A}$, where λ denotes the node density. This shows that with a fixed node density, the required range increases with the area of the network, which is in agreement with [6].

However, the latter form also shows that when the area tends to infinity while the node density is constant, the required transmission range predicted by the model

tends to a finite limit. This is inconsistent with the result shown in [6]. Looking at (6.1), if n and A are increased while keeping their ratio constant, then in order that the required range has no finite limit, the denominator in the expression must grow slower than linearly with n . In other words, the inverse of the square of the required range cannot grow linearly with n in a given area, contrary to the conclusion made in the analysis of the simulation data.

6.3 Diagnostics

The models fitted to the data therefore call for a closer look. Figure 6.2 shows residual-versus-dependent-variable plots of linear models fitted (without using weights) to the squared inverses of different quantiles and means of the critical transmission range for connectivity. If the assumed form of dependence on an independent variable in a model is correct, such a plot should show a "constellation" evenly distributed around zero with no curving trend. The plot in the case of the 99% quantile reveals no such trend, nor can anything decisive be said in the 95% case. However, the plot corresponding to the means shows a distinctive trend with negative curvature, indicating that the rate of increase of the data points with n decreases in comparison to that of the model. This confirms the conjecture made above.

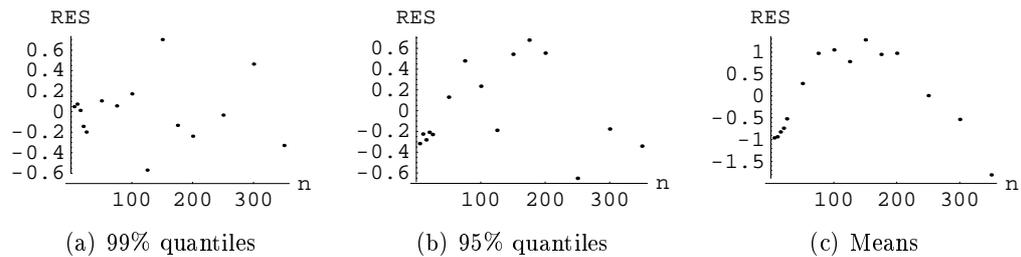


Figure 6.2: Residuals of linear models fitted to the squared inverses of empirical quantiles and means of $R_{\text{crit}}(n)$, plotted against n

The fact that this phenomenon could not be observed in the first two plots probably results from the estimates for the quantiles at the tail of the distribution having excessive variance with the used sample size of 5000. However, the peak-to-peak variation of the residuals in these plots can be seen to be less than in the last plot. This gives reason to speculate whether the behavior of the highest quantiles is still better - although not perfectly - described by models of the form (6.1) than that of the expected value. In fact, such a trend would also explain why the prediction of the higher quantile was better than that of the lower one in the beginning of this chapter.

Now that it has been confirmed that the slope of the squared inverse of the required range as a function of n actually decreases with n , a simple way to improve predictions involving large (that is, larger than in the data at hand) n immediately emerges. Recall that the models in this study were fitted using weighted linear regression, with the conventional aim of minimizing the sum of squares of the eventual residuals. This resulted in more emphasis being put on - and the slope of the model in effect being dictated by - the data points with small n . More accurate predictions in settings involving large n could obviously be achieved with models fitted using an opposite weighting policy, i.e. one that puts more emphasis on the data points with large n .

6.4 Conclusions

This work consisted mainly of two entities: the development of algorithms for finding the threshold transmission ranges for 1-, 2- and 3-connectivity for a given set of nodes, and the statistical analysis of simulation data obtained by utilizing these algorithms and using typical network modelling assumptions. It was found that the statistical behavior of the critical ranges within the settings simulated can be well described with models of simple analytical form. While these models are able to predict connectivity in ad hoc networks rather accurately in the vicinity of their scope, i.e. when the number of network nodes is moderate, their accuracy deteriorates as this number grows. This is due to the fact that the form of the model is inconsistent with theory.

The models obtained in this study are of course bound to the assumed square shape of the network area. Perhaps more important is however the methodology that has been shown for predicting network connectivity with the aid of statistical models; this methodology can be utilized irrespective of the region in question.

6.5 Further work

Although it has been shown that a statistical model of the form (6.1) performs well in predicting connectivity in a wide range of settings, finding the correct form for the model remains for future work.

It seems logical to assume that the critical range for connectivity exhibits similar statistical behavior in bounded domains of other shapes (it should also be in order to consider only convex domains), with the parameters dictated by the border effect intensity inherent to each shape. Some indication of that intensity could be the way the circumference of the region relates to its area. One possible "shape index" for measuring this could be the circumference of the region of the particular shape with unit area. For example, for a circle this index is $2\sqrt{\pi} \approx 3.5$ (and it is known from the classical problem of variational analysis that the circle minimizes the circumference of a given area). For a rectangle whose longer side is a , it is $2(a+1/a)$ which equals to 4 in the case of a square and increases infinitely with a . This implies that the wider the rectangular region, the stronger the border effect, i.e. the longer transmission range or number of nodes would be required. This makes sense, as randomly placed nodes would be spread over a larger distance. This is another potential topic of further study.

The generalization of the model (6.1) to the cases of bi- and triconnectivity has an interesting implication: k -connectivity with higher k could be predicted in a similar way using linear models. The growing complexity of the algorithms to find the critical range for k -connectivity could be compensated for by restricting simulations to realizations containing only few nodes but using large sample sizes to improve model accuracy. This would not necessarily even require new, elaborate algorithms, but the brute-force method of decomposing a problem of finding the threshold range for k -connectivity into n problems concerned with $k-1$ -connectivity could be sufficient.

Relating to Penrose's theorem, it would be worth while to study more closely how the probability that the threshold range for minimum degree k equals to that for k -connectivity grows with n with different k .

Finally, this study is restricted to static networks only. As such, it has direct applicability in, for instance, sensor networks. Considering the effects of node mobility to network connectivity however adds a significant additional dimension. This involves the important choice of mobility model, several of which have been proposed in literature. An issue with great impact is that the assumption of uniform spatial distribution of nodes over time is not valid with some of the most intuitive mobility models.

Chapter 7

Summary

This study focused on the problem of connectivity in ad hoc networks. With uniform spatial distribution of the network nodes, square-shaped network area and a common limit for the communication range as the underlying modelling assumptions, statistical models were fitted to extensive simulation data in order to ultimately determine how the probability that such a network is connected depends on three quantities: the number of nodes in the network, the area over which the nodes are scattered, and the transmission range limit.

Simple network connectivity is a weak property in the context of the general notion of k -connectivity. A natural extension for the scope of this study was to examine bi-connectivity and triconnectivity which are stronger requirements in terms of network reliability and route diversity.

The key definition that allowed precise, quantitative study of these properties was that of the threshold transmission range for k -connectivity which in fact is the distance separating a certain pair of nodes in every network realization. This node pair is unique in the case of random networks. An important part of this study was developing algorithms for finding the critical range efficiently in the cases $k = 1, 2, 3$.

The essential finding in the statistical analysis of simulation data was that the statistical behavior of the threshold ranges can be bound very closely to the three quantities mentioned above using models of simple, analytical form.

References

- [1] Charles E. Perkins. *Ad hoc networking*. Addison-Wesley, Boston, 2001.
- [2] C.-K. Toh. *Ad hoc mobile wireless networks: protocols and systems*. Prentice-Hall, Upper Saddle River (NJ), 2002.
- [3] Juan Francisco Redondo Antón. Ad Hoc Networks - design and performance issues. Master's thesis, Networking Laboratory, Helsinki University of Technology, 2002.
- [4] R. Meester and R. Roy. *Continuum percolation*. Cambridge University Press, Cambridge, 1996.
- [5] O. Dousse, P. Thiran, and M. Hasler. Connectivity in ad-hoc and hybrid networks. In *Proceedings of Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002)*, volume 2, pages 1079–1088, 2002.
- [6] T. K. Philips, S. S. Panwar, and A. N. Tantawi. Connectivity properties of a packet radio network model. *IEEE Transactions on Information Theory*, 35(5), September 1989.
- [7] M. D. Penrose. On k-connectivity for a geometric random graph. *Random Structures and Algorithms*, 15(2):145–164, 1999.
- [8] C. Bettstetter. On the minimum node degree and connectivity of a wireless multihop network. In *Proc. 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02)*, pages 80–91, June 2002.
- [9] C. Bettstetter and J. Zangl. How to achieve a connected ad hoc network with homogeneous range assignment: an analytical study with consideration of border effects. In *Proc. 4th IEEE International Conference on Mobile and Wireless Communication Networks (MWCN'02)*, pages 125–129, September 2002.
- [10] M. Sánchez, P. Manzoni, and Z. J. Haas. Determination of critical transmission range in Ad-Hoc Networks. In *Proceedings of Multiaccess Mobility and Teletraffic for Wireless Communications 1999 Workshop (MMT'99)*, October 1999.
- [11] R. C. Prim. Shortest connection networks and some generalizations. *Bell Systems Technology Journal*, 36:1389–1401, 1957.
- [12] G. Di Battista and R. Tamassia. On-line Maintenance of Triconnected Components with SPQR-trees. *Algorithmica*, 15:302–318, 1996.

Appendix A

Algorithms

Algorithm 1 Find $R_{\text{crit}}(X)$

Require: $X = \{x_1, \dots, x_n\}, n > 1, x_i \in \mathbb{R}^2 \forall i$

Ensure: $R = R_{\text{crit}}(X)$

Calculate $\mathbf{D} \in \mathbb{R}^{n \times n} : d_{ij} = \|x_j - x_i\|_2$ {the distance matrix}

$R \leftarrow \max_i \{\min_j \{d_{ij} | d_{ij} > 0\}\}$ {the threshold range for minimum degree 1}

$S \leftarrow \emptyset$ {the set of connected subnetworks}

$P \leftarrow \{1, 2, \dots, n\}$ {the set of nodes not yet included}

while $P \neq \emptyset$ **do** {Note that P cannot contain only 1 element}

$C \leftarrow \{P(1)\}$ {the first element in P }

$P \leftarrow P \setminus \{P(1)\}$

$i \leftarrow 0$

repeat

$i \leftarrow i + 1$

$N \leftarrow \{j | j \in P, d_{C(i)j} \leq R\}$ {nodes within range R from $C(i)$ }

$C \leftarrow C \cup N$ {Append N at end of C }

$P \leftarrow P \setminus N$

until $P = \emptyset \vee i = \text{card}(C)$

$S \leftarrow S \cup \{C\}$ {Maintain C as a set}

end while

$N_S \leftarrow \text{card}(S)$

if $N_S > 1$ **then** {run the Prim algorithm for the connected subnetworks:}

 Calculate $\mathbf{M} \in \mathbb{R}^{N_S \times N_S} : m_{ij} = \min\{d_{kl} | k \in S(i), l \in S(j)\}$

 { \mathbf{M} is the distance matrix for the connected subnetworks}

$C \leftarrow \{1\}$

$P \leftarrow \{2, \dots, N_S\}$ {the subnetworks not yet included}

while $P \neq \emptyset$ **do**

$s \leftarrow \text{argmin}_{j \in P} \{m_{ij} | i \in C\}$ {the closest subnetwork not yet included}

$r \leftarrow \min\{m_{ij} | i \in C, j \in P\}$ {and its distance from the included subnetworks}

$C \leftarrow C \cup \{s\}$

$P \leftarrow P \setminus \{s\}$

$R \leftarrow \max\{R, r\}$

end while

end if

Algorithm 2 Find $R_{\text{bicon}}(X)$

Require: $X = \{x_1, \dots, x_n\}, n > 2, x_i \in \mathbb{R}^2 \forall i$
Ensure: $R = R_{\text{bicon}}(X)$

 Calculate $\mathbf{D} \in \mathbb{R}^{n \times n} : d_{ij} = \|x_j - x_i\|_2$ {the distance matrix}

 $R \leftarrow \max_i \{\min_j \{d_{ij} | \exists d_{ik} > 0 : d_{ik} < d_{ij}\}\}$ {the threshold range for minimum degree 2}

 $V \leftarrow \emptyset$ {the set of nodes visited}

 $A \leftarrow \emptyset$ {the set of articulation points}

DFS(1) {Call Algorithm 3 for the first node. The return value of this top instance will be 1.}

 {The recursive depth-first search will complete the sets V and A .}

if $\text{card}(V) \neq n$ **then** {the network is not connected with the current range}

 $R \leftarrow R_{\text{crit}}(X)$ {Run Algorithm 1 (Use R as the initial range; the distance matrix is already known)}

 $V \leftarrow \emptyset$
 $A \leftarrow \emptyset$ {Re-initialize the sets}

DFS(1)

end if
 $N \leftarrow \text{card}(A)$
for $i = 1$ to N **do** {If $A = \emptyset$ do nothing}

 $Y \leftarrow X \setminus \{x_{A(i)}\}$
 $\mathbf{E} \in \mathbb{R}^{(n-1) \times (n-1)} \leftarrow \{d_{jk} | j, k \neq A(i)\}$ {the distance matrix \mathbf{D} with the row and column $A(i)$ deleted}

 $r \leftarrow R_{\text{crit}}(Y)$ {Use \mathbf{E} as the distance matrix and R as the initial range}

 $R \leftarrow \max\{R, r\}$
end for

Algorithm 3 on the following page is the recursive depth-first search customized to maintain the set of articulation points. It is defined here as a sub-algorithm within Algorithm 2. The range R and the sets V and A are treated as global variables defined in Algorithm 2.

Algorithm 3 DFS(i)

Require: $i \in \{1 \dots n\}$ {the node that DFS is called for: "this node"}**Ensure:** v is the first node in V referred to under this DFS call $V \leftarrow V \cup \{i\}$ {Append i at the end of $V!$ } $v \leftarrow s \leftarrow \text{card}(V)$ $a \leftarrow \text{FALSE}$ {The articulation point condition for nodes other than the root} $c \leftarrow 0$ {The number of children of this node in the DFS tree}**for all** $1 \leq j \leq n$ such that $0 < d_{ij} \leq R$ **do** **if** $\exists k : V(k) = j$ **then** {this adjacent node has already been visited} $v \leftarrow \min\{v, k\}$ **else** $c \leftarrow c + 1$ $w \leftarrow \text{DFS}(j)$ **if** $w = s$ **then** $a \leftarrow \text{TRUE}$ **else** $v \leftarrow \min\{v, w\}$ **end if** **end if****end for****if** $(s = 1 \wedge c > 1) \vee (s \neq 1 \wedge a = \text{TRUE})$ **then** $A \leftarrow A \cup \{i\}$ **end if**Return v
