

Faculty of Informatics

Faculty of Informatics - Papers

University of Wollongong

Year 2004

Topology control in heterogeneous ad-hoc
networks

G. Srivastava* P. Boustead†
J. F. Chicharo‡

*University of Wollongong, gauravs@uow.edu.au

†University of Wollongong, boustead@uow.edu.au

‡University of Wollongong, chicharo@uow.edu.au

This paper originally appeared as: Srivastava, G, Boustead, P & Chicharo, JF, Topology control in heterogeneous ad-hoc networks, Proceedings. 12th IEEE International Conference on Networks, 16-19 November 2004, vol 2, 665-670. Copyright IEEE 2004.

This paper is posted at Research Online.

<http://ro.uow.edu.au/infopapers/127>

TOPOLOGY CONTROL IN HETEROGENEOUS AD-HOC NETWORKS

Gaurav Srivastava, Paul Boustead, Joe F.Chicharo
School of Electrical, Computer and Telecommunication Engineering,
University of Wollongong, NSW, Australia.

Email: gaurav@titr.uow.edu.au, paul@titr.uow.edu.au, Joe_chicharo@elec.uow.edu.au

Abstract—Topology control in an ad-hoc network can provide better spatial reuse of the wireless channel and conserve power. Topology construction and maintenance is a challenging issue. A number of distributed topology control algorithms have been proposed to remove the need of a centralised controller. Distributed algorithms such as Location Information No Topology (LINT), Location Information Link State Topology (LILT) and Mobile-Grid (MG), aim to achieve overall network connectivity and low transmission range by maintaining a minimum node degree value. In the case of a non-uniform node distribution, maintaining a minimum node degree can unnecessarily partition a network. In this paper we propose a distributed algorithm that utilises one hop neighbours and their location information to maintain a number of critical links required to keep a connected network. Such critical links are included along with the links required to meet the node degree criterion. A distributed mechanism to construct and maintain a network topology is proposed, which can be integrated as part of the neighbour discovery protocol. Furthermore, nodes collaborate to remove unidirectional links. A simulation based analysis of the proposed algorithm is provided for a number of node degree values. Simulations indicate that the proposed algorithm is able to achieve higher connectivity for different node distributions.

I. INTRODUCTION

Wireless networking has led to a large growth in portable computing. Mobile Ad-hoc Networks (MANETs) are a group of mobile wireless nodes working together without any fixed infrastructure. Each node can act as a wireless router and forward traffic of other mobile nodes. Ad-hoc networks are characterised with low bandwidth, limited transmission range, power constraints and limited processing capabilities.

The topology of an Ad-hoc network can have a considerable impact on its performance. In a high density network, nodes may experience interference from many other nodes which can reduce the available bandwidth considerably [1] [2]. In the case of a sparse network, low transmission power of network devices can limit the network connectivity and unnecessarily partition a network. The topology of a mobile wireless network is more susceptible to changes as compared to a wired network.

A number of topology control algorithms have been proposed for Ad-hoc networks that optimise the network connectivity and minimise the power usage. The algorithms can be classified into centralised and distributed in nature. Distributed topology control algorithms such as LINT [3], LILT [3] and MG [4] adjust the transmission power to achieve a particular node degree value. A node degree of a node is defined as the number of nodes within its transmission range. A node can increase or decrease its transmission range to alter its node degree value. A minimum node degree of six was initially proposed in [5] and $5.1774\log(N)$ by Kumar et. al. in [6], for a uniform node distribution. In the

case of heterogeneous node distributions, node degree can vary significantly depending on the location and movement pattern of nodes [7]. A fixed node degree value in such distributions may result in a number of disconnected network components or clusters.

In LILT, LILT and MG, each network node executes power control independently without any collaboration or message exchange among the network nodes. The lack of collaboration can result in a number of unidirectional links, as power control by one network node may not necessarily add a bidirectional link with its neighbouring nodes. Each independent power adaptation decision by a network node, can generate a number of link-state updates and disrupt the node degree value of other nodes within its transmission range. The neighbouring nodes may need to re-execute topology control to maintain a fixed node degree value. This process of independent topology control, can increase the convergence time of the algorithm and introduce additional communication delay. Furthermore, a fixed power is used for all network communications at a particular node, until the next topology control decision. Using a fixed power for all communications also limits the power saving benefits that can be achieved by using the location information of the neighbouring nodes and adjusting the transmission power accordingly.

A Dist-RNG [8] graph is constructed in one power alteration per node. However, a Dist-RNG graph maintains a low link redundancy, which can lead to low fault tolerance [8]. As a result, a mobile network has more chance of being disconnected or partitioned. A Dist-RNG based graph may also have a larger hop diameter than a LINT, LILT and MG based graph. Due to a large hop diameter, packets are routed over larger number of hops, thereby increasing the end-to-end packet delivery delay.

In this paper we introduce a distributed topology control algorithm, Collaborative Algorithm with Probable Critical Links (CA-PCL), that extends the node degree heuristic proposed in LINT, LILT and MG to include a number of probable critical links along with the links specified by the target node degree value. A topology construction and maintenance strategy is proposed in CA-PCL that can be integrated as part of a neighbour discovery mechanism. Neighbour discovery mechanisms are used by routing protocols to keep track of the links to the neighbouring nodes. An example of the neighbour discovery mechanism is a "Hello" message [9]. In CA-PCL, unidirectional links are identified and converted to bidirectional links by overriding the local node degree value. The proposed algorithm provides a

better connected and reliable network topology for various node distributions.

The remainder of the paper is organised as follows. Section II outlines the node degree based topology control techniques proposed in literature. Section III describes the proposed algorithm CA-PCL. Section IV provides a simulation based analysis of this approach. Section V concludes the paper.

II. TOPOLOGY CONTROL ALGORITHMS

Since the advent of DARPA packet radio networks, numerous topology control algorithms have been proposed for wireless networks. Topology control algorithms can be classified into centralised and distributed in nature. Centralised topology control assumes a network node with global link state information to compute a minimum power connected topology. Examples of the centralised topology control algorithms in literature include RNG [10], MST [11], Connect [3], NTC [12], minR [13] and Biconn-Augment [3]. In distributed topology control, all topology control decisions are made on the local link-state information of a node and its neighbouring nodes. The distributed nature of the algorithm reduces the bottleneck of a centralised controller and the overheads of disseminating the link-state information to the coordinating node. Examples of the distributed topology control algorithms in literature include LINT [3], LILT [3], Dist-RNG [8], Dist-NTC [12] and MG [4].

LINT proposed by *Ramanathan et al.* [3] uses locally available neighbour information collected by a neighbour discovery mechanism to keep the degree of neighbours bound. A node becomes a ‘neighbour’ if it is within the transmission range of another node. Neighbour information is updated when a node moves in or out of the communication range of each other. A node maintains a list, comprising of its one hop neighbours. A bi-directional link is established when two nodes are within each others transmission range. The aim of maintaining a particular node degree is to provide power saving in a dense network where a maximum power topology may no longer be necessary to provide a connected graph. Nodes can achieve further power saving by transmitting data over smaller distance and relying on intermediate nodes to forward their traffic. In [3] all network nodes are configured with three parameters, the desired node degree nd_d , a high threshold of the node degree nd_h and a low threshold of node degree nd_l . A node periodically checks the number of its active neighbours. If the degree is greater than nd_d , the node reduces its operational power. If the degree is less than nd_d the node increases its operational power. If neither is true then no action is taken. The increase or decrease in transmission power is bounded by maximum and minimum power settings of the radio and the range of the node degree requirements. Maintaining a particular node degree can work well in a uniform node distribution, however in the case of a clustered node distribution the network can become unnecessarily partitioned. In [3] the desired transmission range (r_d) is calculated by using Equation 1, where nd_c is the current node degree and r_c is the current transmission range.

$$r_d = r_c \sqrt{\left(\frac{nd_d}{nd_c}\right)} \quad (1)$$

This procedure of altering the transmission power without any information exchange can result in a number of iterative topology control decisions before the algorithm converges to the desired node degree.

Another algorithm/heuristic presented by *Ramanathan et al.* [3] is LILT, which exploits global topology information for recognising and repairing network partitions. There are two main parts of LILT, Neighbour Reduction Protocol (NRP) and Neighbour Addition Protocol (NAP). NRP reduces the transmission power to maintain the node degree around a certain configured value where as NAP increases the transmission power to establish additional links, necessary to keep the network from getting partitioned. When executing NRP or NAP a node relies on the link-state routing protocol to determine whether it is connected or biconnected. If a node does not receive any updates then it may be in a disconnected state. If the topology is biconnected then no action is taken. If the topology is disconnected then the node increases its transmission power to the maximum possible value. If the topology is connected and not biconnected, the node tries to achieve bi-connectivity [3]. This process of increasing the transmission power without any message exchange does not guarantee a link with the neighbouring nodes. Another drawback of the LILT approach is that it relies on global link state information to evaluate bi-connectivity and connectivity. Evaluating the overall network connectivity and bi-connectivity may require a global search, which incurs a computational complexity in the order of $O(N + E)$, where E is the total number of edges in the network.

Liu et al. [4] proposed MG, a distributed topology control algorithm where each node reduces the network interference by maintaining a specific contention index (CI). CI is a product of node density¹ (ρ) and area size (A). In order to maintain global CI, all nodes try to keep the local CI bound to a specific value. The local estimate of CI at a node is evaluated from the number of one hop neighbours. Each node looks up a particular optimisation table to determine whether it is operating around an optimal value of CI. The optimal values of CI are evaluated beforehand through simulations and hard-coded in the network nodes [4]. A node adjusts its transmission range to keep the CI value bound. MG uses CI values between [3,9] to maximise the network capacity [4]. This approach is similar to LINT as it utilises node degree ($CI_{local} = nd - 1$) as a measure of local CI.

RNG graph is a subset of NTC graph, which is computed using Denulay Triangulation (DT). In the case of a greedy RNG algorithm, each link is compared with every other link to ascertain whether a node is a RNG neighbour. The brute force algorithm incurs a processing cost in the order of $O(N^3)$, where N are the number of nodes in the network [8]. A brute force Dist-RNG algorithm will limit the search to the one hop neighbours of a node and thus incurs a processing overhead of $\sum_{i=0}^{N-1} N_i^2$, where N_i are the one hop neighbours of the i^{th} node. A number of techniques have been proposed to compute a Dist-RNG. Such techniques include a Lune approach in [10] and a cone based approach in [8].

¹The number of nodes per unit area.

III. PROPOSED ALGORITHM

The aim of the proposed algorithm, (CA-PCL) is to maintain a minimum node degree and reduce the chance of network partitioning by maintaining a number of probable critical links that may be essential to keep the network connected. In CA-PCL all nodes construct the local topology on the basis of the one hop neighbour information. The neighbour information is evaluated by exchanging “Hello” messages at maximum power [9]. The “Hello” messages are used to evaluate a list of unidirectional neighbouring nodes (N_i). All network nodes use node identification (ID) to develop a neighbour list. The ID is unique number assigned to a node to maintain its identity in a network, for example Internet Protocol address. A node’s ID is used to construct a Topology Control Neighbour (TCN) list comprising of ‘x’ edges per node as required by the node degree value. The TCN list is a subset of the maximum power neighbours and is used to identify links with neighbouring nodes. The details are provided in the *Construct-TCN()* function. All nodes can construct the TCN lists independently of each other. The TCN list is appended to the “Hello” message and broadcasted to the neighbouring nodes.

The node degree parameter ‘x’ is not sufficient to maintain the overall network connectivity and only including the first ‘x’ edges per node may unnecessarily partition the network. The edges that are not included in the TCN list should be tested to evaluate whether they are probable critical links. The edges remaining after including ‘x’ edges per node is denoted by set E_{left} , where $E_{left} = \{N_i - TCN_i\}$. Each element in set E_{left} is tested locally for a probable critical link.

In [10], a *Lune* is proposed to evaluate whether a node is located between two network nodes. The location of the neighbouring nodes can be evaluated by using either the ‘signal strength’ of the received broadcast or using Global Positioning System [14]. A *Lune* is defined as the region covered by the intersection of two circles centred at ‘i’ and ‘j’ as shown in Figure 1. The radius of a circle represents the transmission range of a node. The condition that a node ‘k’ exists in the $Lune(i,j)$ is that the $\angle ikj$ of the $\triangle ikj$ is $> 60^\circ$ [10]. In Equation 1, we derive the condition that a node is located in the $Lune(i,j)$ by applying the *Cosine* rule to the $\triangle ikj$.

$$\cos(\angle ikj) = \frac{r_{jk}^2 + r_{ik}^2 - r_{ij}^2}{2r_{jk}r_{ik}} \quad (2)$$

$$\angle ikj > 60^\circ \equiv \cos(\angle ikj) < \frac{1}{2} \quad (3)$$

Equation 4 is derived by using Equation 2 and 3.

$$\frac{r_{jk}^2 + r_{ik}^2 - r_{ij}^2}{2r_{jk}r_{ik}} < \frac{1}{2} \quad (4)$$

Since the sum of the angles in $\triangle ikj \leq \pi$, Equations 5, 6 should hold true. Hence, for node to be in a $Lune(i,j)$, Equations 4, 5 and 6 should be satisfied.

$$r_{ik} \leq r_{ij} \quad (5)$$

$$r_{jk} \leq r_{ij} \quad (6)$$

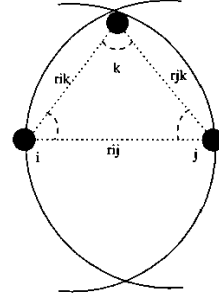


Fig. 1. A lune of i and j is the region between the two arcs

A node executes *Critical-TCN()* function in order to evaluate whether a link is a probable critical link or not. A $Lune(i,j)$ is used to search for a critical link. If a node exists in the $Lune(i,j)$, then link r_{ij} is not considered a probable critical link. If there is no node located in the $Lune(i,j)$, then link r_{ij} is marked as a probable critical link and added to TCN_i .

Nodes receiving the broadcast are able to determine the bidirectional TCNs (BTCNs) by using the *Bidirectional-TCN()* function. BTCNs are evaluated by performing a search between local TCNs and the one hop neighbouring TCNs. Unidirectional TCNs are converted to BTCNs by using the *Convert-TCN()* function, which uses a search to evaluate the unidirectional TCNs and add a corresponding link. The conversion of unidirectional TCNs to bidirectional TCNs does not require any message exchange apart from the “Hello” messages. We assume that all nodes are *Collaborative* and are willing to support such decisions. A node iterates through its neighbour’s TCN list and adds the neighbour’s ID to its local TCN list, if there is a unidirectional link from its neighbour to itself. This procedure is critical as the node degree requirement may not necessarily be met by all network nodes. A node may override its local node degree requirement to establish a bidirectional link.

A *Space Attenuation Model* is proposed in [15] and can be used to evaluate the transmission power required to communicate with a TCN. The maximum power for the data packet is set to reach the furthest local TCN. Adjusting the transmission power on the bases of the link distance can provide further power saving as compared to the fixed power approach. The bottleneck of the algorithm is the process of sorting the one hop neighbour nodes in the order of their distance from the current node.

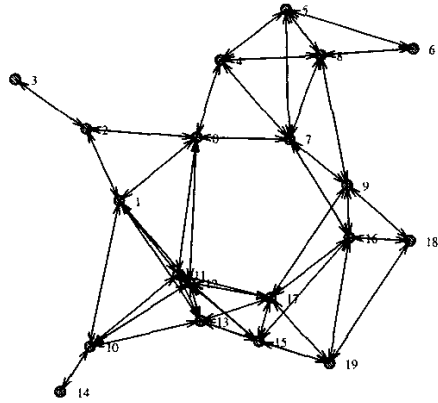
Algorithm *Construct-TCN()*

(* Construct a TCN list *)

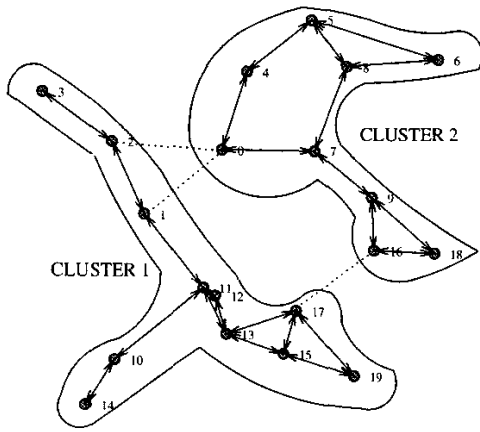
1. $N_i \leftarrow$ 1-hop unidirectional neighbours of node i, sorted in the order of their distance from i
2. $TCN_i \leftarrow$ Local uni-directional TCNs of node i
3. $x \leftarrow$ Target node degree value
4. **if** $N_i \geq x$
5. **then** $TCN_i \leftarrow$ add first x elements of N_i
6. **else** $TCN_i = N_i$
7. **return** TCN_i

Algorithm *Critical-TCN()*

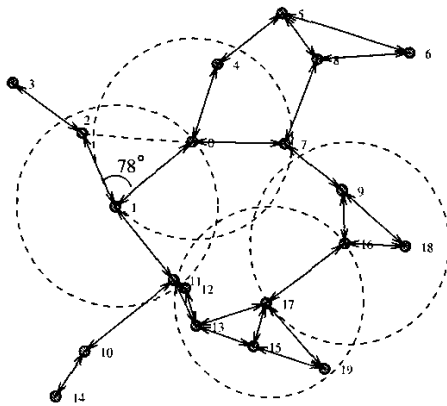
(* Add probable critical links to the TCN list *)



(a) Maximum Power Topology of a 20 node network



(b) LINT(nd=2) based topology of a 20 node network



(c) CA-PCL based topology of a 20 node network

Fig. 2.

1. $N_i \leftarrow$ 1-hop unidirectional neighbours of node i , sorted in the order of their distance from i
2. $TCN_i \leftarrow$ Local uni-directional TCNs of node i
3. $E_{i_{left}} = N_i - TCN_i \leftarrow$ A set of probable critical links to be tested
4. $r_{ij} \leftarrow$ Distance between node i and node j
5. **if** $E_{i_{left}} \neq \emptyset$
6. **for** each node $j \in E_{i_{left}}$
7. **for** each node $k \in N_i$ AND $k \neq j$
8. **if** $r_{ik} \leq r_{ij}$ AND $r_{jk} \leq r_{ij}$ AND $r_{jk}^2 + r_{ik}^2 - r_{ij}^2 < r_{jk}r_{ik}$
9. **then** COND = TRUE
10. **if** COND \neq TRUE
11. **then** Add j to TCN_i
12. **return** TCN_i

Algorithm Bidirectional-TCN()

(* Find Bidirectional TCN list *)

1. $TCN_i \leftarrow$ Local uni-directional TCN list of node i
2. $BTCN_i \leftarrow$ Local bi-directional TCN list of node i
3. **if** $TCN_i \neq \emptyset$
4. **for** each node k in TCN_i
5. $j \leftarrow$ Calculate node k 's TCNs
6. **while** $j \neq \emptyset$
7. **if** $j = i$
8. **then** Add k to $BTCN_i$
9. **return** $BTCN_i$

Algorithm Convert-TCN()

(* Convert unidirectional to bidirectional TCN *)

1. $N_i \leftarrow$ 1-hop unidirectional neighbours of node i
2. $BTCN_i \leftarrow$ Local bi-directional TCN list of node i
3. **if** $N_i \neq \emptyset$
4. **for** each node k in N_i
5. $j \leftarrow$ TCN of node k
6. **if** $i = j$ and $k \notin BTCN_i$
7. **then** Add k to $BTCN_i$
8. **return** $BTCN_i$

One may argue to reverse the process, where we evaluate the critical links before (effectively creating a Dist-RNG) and then add links to satisfy the node degree criteria. Since a Lune is used in order to evaluate the critical links, the processing overheads per node will be in the order of $O(N_i^2)$, where each one hop neighbour's link distance will be compared with every other neighbour node. However, if the critical points are evaluated after satisfying the node degree criteria (CA-PCL), the processing overheads per node will be in the order of $O(N_i(N_i - TCN_i))$, which is lower than N_i^2 for all values of $TCN_i \geq 1$. Low processing overheads can be crucial in battery operated devices with low processing capabilities.

Figure 2(a) is the plot of maximum power topology of a 20 node network. Figure 2(b) is the plot of a topology with a node degree requirements of 2 (LINT(nd=2)). A line represents a link between the two nodes. In Figure 2(b), the network is unnecessarily partitioned into two clusters. Some of the probable critical links are illustrated with a dotted line in Figure 2(b) and include nodes $\{2, 0\}$, $\{1, 0\}$ and $\{17, 16\}$. The topology in Figure 2(c)

is obtained after applying CA-PCL to a 20 node network. The network topology in Figure 2(c) is connected as compared to Figure 2(b). In Figure 2(c) the link between node 2 and 0 is not included as the $\angle 210 (= 78^\circ)$ is greater than 60° and satisfies Equation 5. A link between node 1 and 0 and node 17 and 16 is added as there is no other node present in the Lune(1,0) and Lune(17,16).

IV. SIMULATION AND RESULTS

A. Scenario

A simulation based study of LINT and the proposed algorithm, (CA-PCL) has been conducted. Four cluster distributions are generated in a 600m x 600m grid area and the nodes are varied in number from 10 to 100. All nodes start with a maximum transmission range of 200m (5dBm). An analysis on the performance of LINT and CA-PCL is conducted for a number of node degree (nd) values $\{nd = 1, 2, 3, 4, 5, 6\}$. The results are examined in terms of the average network connectivity, average transmission range and the average number of one hop neighbours. A comparison of the fault tolerance of CA-PCL(nd=6) and Dist-RNG is also conducted.

B. Power model

The transmission power of a node is based on the Space Attenuation Model and is given by $P_{rx} = P_{tx} \left(\frac{c}{4\pi df}\right)^2$, where 'c' is the speed of light ($3.10^8 \frac{m}{sec}$) and the 'f' is the frequency of the spectrum (2.4 GHz), ' P_{rx} ' is the received power, ' P_{tx} ' is the transmitters power, and 'd' is the distance between the transmitter and the receiver [15].

C. Results

The average connectivity is measured by evaluating the mean connectivity of every network node. The mean connectivity of node 'i' is given by $c_i = \frac{x}{N}$, where x is the number of nodes reachable by node 'i' and 'N' is the total number of nodes in the network. The average network connectivity of the entire network is evaluated by summing the mean connectivity of every node and is given by $\frac{1}{N} \sum_{i=0}^{N-1} c_i$. Figure 3(a) is a comparison of the average connectivity against the total number of network nodes for different node degree values. Figure 3(a) illustrates that the average connectivity of LINT for $nd = \{1, 2, 3, 4, 5, 6\}$ is lower as compared CA-PCL. The average connectivity of LINT for $nd = \{1, 2, 3, 4\}$ decreases with an increase in network density as the mean connectivity of each node remains approximately constant. The CA-PCL approach shows a significant improvement in connectivity for low node degree values. Figure 3(a), illustrates that a minimum node degree value in a clustered distribution, is not enough to provide a connected graph and therefore the connectivity of LINT is lower than that of CA-PCL for all node degree values examined.

Figure 3(b) is the plot of the average one hop bidirectional neighbours against the total number of network nodes. The average one hop bidirectional neighbours in the case of LINT are lower than the required number of neighbours. This illustrates the limitation of an independent topology control deci-

sion, where an increase in transmission power by a node, does not necessarily increase its bidirectional neighbour degree. The average one hop neighbours in the case of CA-PCL are larger than LINT for all the node degree values. The difference of approximately 1-3 link/node is observed for all node degree values, but a significant improvement in connectivity is achieved. The extra links are introduced by the using the *Critical-TCN()* and *Convert-TCN()* functions, which aim to maintain a number of probable critical links and convert unidirectional links into bidirectional links.

Figure 3(c) is the plot of the average of the maximum transmission power used for data packets against increasing network density for various node degree values. The maximum transmission power is chosen to analyse the worse case scenario where all communications are routed over the furthest TCN node. Further power saving can be achieved in CA-PCL by adjusting the transmission power based on the link distance of each transmission. The average power decreases with an increase in node density for both the algorithms. The average transmission power in the case of CA-PCL is higher than LINT. This is expected due the *Critical-TCN()* and *Convert-TCN()* function. In the case of $nd = 1$, LINT graph is mainly disconnected. Thus, a significant power difference of approximately 6 dBm is observed between LINT and CA-PCL.

A comparison of the fault tolerance of CA-PCL(nd=6) and Dist-RNG is illustrated in Figure 3(d). Figure 3(d) is the plot of the average connectivity of CA-PCL(nd=6) and Dist-RNG against the number of network nodes for different values of node failure rate (f). It is evident from Figure 3(d) that CA-PCL maintains larger connectivity as compared to Dist-RNG for similar node failure rate. This was expected as the total number of links in a Dist-RNG topology is lower than CA-PCL(nd=6). Therefore, CA-PCL(nd=6) has a larger ability to cope up with node failure as compared to Dist-RNG.

In summary, the CA-PCL approach shows a significant improvement over the LINT approach for low node degree values $\{1, 2, 3, 4, 5, 6\}$. If the required node degree value is increased, the utility of the maintaining critical points decreases as a network topology is better connected. However, there might be distributions where high node degree is not sufficient and maintaining critical points is crucial to keep a connected graph. A high node degree value also increase the transmission range, leading to higher power consumption for the entire network. The proposed algorithm, CA-PCL(nd=6), also has a larger fault tolerance as compare to Dist-RNG.

V. CONCLUSION

In this paper we have extended the heuristics in [3] and [4] and proposed a distributed topology control algorithm CA-PCL. A simulations based analysis of CA-PCL, LINT and Dist-RNG has been provided.

REFERENCES

- [1] Gupta P. and Kumar P.R., "The capacity of Wireless Networks," in *Transactions and information theory*, Mar. 2000.
- [2] P.R. Kumar, P. Gupta, and Robert Gray, "An Experimental Scaling Law

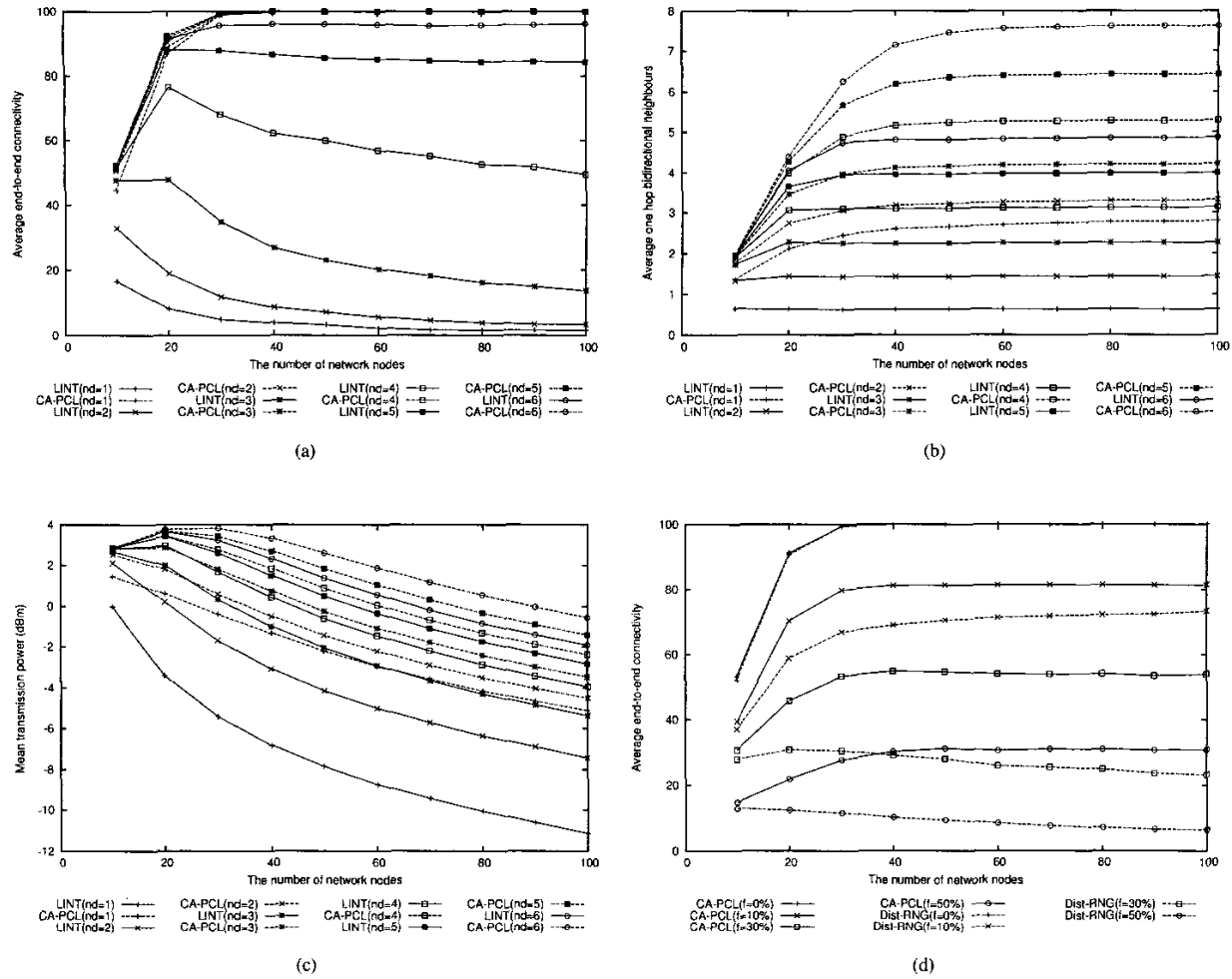


Fig. 3.

- for Ad Hoc Networks,” in *The Convergence of Information Technologies and Communication*, Oct. 2001.
- [3] R. Ramanathan and R. Rosaes-Hain, “Topology control of multihop wireless networks using transmit power adjustment,” in *Proceedings of IEEE INFOCOM*, 2000.
 - [4] Jilei Liu and Baochun Li, “Mobilegrid: Capacity-aware topology control in mobile ad hoc networks,” in *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference on*, Vol., Pages: 570-574, 2002.
 - [5] L. Klienrock and J. Silvester, “Optimum transmission radii for packet radio networks or why six is a magic number,” in *Proceedings of the IEEE National Telecommunication Conference*, pages 4.3.1-4.3.5 Brimingham, Dec. 1978.
 - [6] Feng Xue and P.R. Kumar, “The number of neighbours needed for connectivity of wireless networks,” in *To appear in Wireless Networks*.
 - [7] E.M. Royer, P.M. Melliar-Smith, and L.E. Moser, “An analysis of the optimum node density for ad hoc mobile networks,” in *Proceedings of the IEEE International Conference on Communications, Helsinki, Finland*, June 2001.
 - [8] Chi-Fu Huang, Yu-Chee Tseng, Shih-Lin Wu, and Jang-Ping Sheu, “Distributed topology control algorithm for multihop wireless networks,” in *Proceedings of the 2002 International Joint Conference on Neural Networks, Vol. 1*, Pages: 355-360, 2002.
 - [9] C.E Perkins and E.M. Bending-Royer, “Ad hoc On-Demand Distance Vector Routing Protocol for Mobile Ad Hoc Networks,” in *draft-ietf-manet-aodv-11.txt*, June 2002.
 - [10] Kenneth J. Supowit, “The relative neighbourhood graph with application to minimum spanning trees,” in *Journal of the ACM*, vol. 30, no. 3, pp 428-448, 1983.
 - [11] Ning Li, Jennifer C. Hou, and Lui Sha, “Design and Analysis of an MST-Based Topology Control Algorithm,” in *The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 2003*.
 - [12] L. Hu, “Topology control for multihop packet radio networks,” in *IEEE Transactions on Communications*, Vol.41, Iss.10, Pages: 1474- 1481, 1993.
 - [13] Esther Jennings and Clayton Okino, “Topology control for efficient information dissemination in ad-hoc networks,” in *International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS 2002*, 2002.
 - [14] Gopal D. and Raj J, “Potential Networking Applications of Global Positioning Systems,” in *OSU Technical report TR-24*, April 1996.
 - [15] Paul H. Young, *Electronic Communication Techniques*, Macmillian Publishing Company, 1994.