This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

Location-Free Topology Control Protocol in Wireless Ad Hoc Networks

Jang-Ping Sheu, Shin-Chih Tu, and Chi-Hung Hsu

Dept. of Computer Science and Information Engineering

National Central University, Taiwan

Abstract- **Topology control not only achieves the objective of power saving but also increases the system throughput by increasing the spatial reuse of communication channels. However, there exists a hidden terminal problem due to asymmetric transmission radii among nodes after topology control. In this paper, we propose a distributed protocol that deals with topology control at network layer and hidden terminal problem at MAC layer. Each node in the networks determines its power for data transmission and control packets transmission according to the received beacon messages from its neighbors. The proposed protocol works without location information and uses little control packet overhead to prevent the potential collisions due to the hidden terminals. Simulations show that our protocol significantly decreases total power consumption in the networks and has a better network throughput compared to previous work.**

I. INTRODUCTION

Mobile ad hoc networks (MANETs) have become popular research topics recently. Due to the mobile manner of wireless nodes, the construction of an ad hoc network, called topology, is usually temporary or changes dynamically. In a MANET, besides the distribution of wireless nodes, the topology is mainly decided by their transmission power or radius. Conventionally, in a MANET, the transmission radii of nodes are fixed and all nodes use the maximum power to transmit packets. However, the overall performance on the end-to-end delay, channel utilization, and lifetime of a MANET can be enhanced if the transmission power of each node is properly adjusted to a reduced level [1][2][3]. The primary object of topology control is to design an energy-efficient protocol that optimizes the transmission power of each node, while the resulting topology retains its property of connectivity. In general, higher network throughput can be achieved due to the following two benefits. First, the interference is reduced by varying the transmission radii of the nodes to a nearer scope [1][4]. Second, more data transmissions are able to simultaneously occur in the neighborhood of a node and thus, increase spatial channel re-use[2].

Several papers were proposed to address topology control in the literature. We can classify them into centralized and distributed computing methods. The centralized topology control methods [5][6][7], such as the minimum spanning tree based algorithm [5], assume that a central entity (e.g., sink or access point) knows the locations of each node, and is capable of determining the optimum transmission power of each node through the collected global information. Although this centralized method is simple, it is not scalable. Moreover, such a central entity is against the nature of ad hoc networks in which it normally lacks infrastructure. On the other hand, the distributed methods [8][9][10] have the advantage of scalability and adaptation to mobility of nodes whereby each node makes a local decision of the suitable transmission power based on the gathered information from nearby neighbors.

This work was supported by the National Science Council of Republic of China under grant NSC94-2213-E-008- 024

However, most of the proposed approaches for topology control hold the assumption that each node knows its own location information by means of a global positioning system, triangulation-based positioning protocols and other positioning methods. However, to be equipped with a positioning device not only increases the cost of hardware deployment but also brings about several other disadvantages. First, it confines the location-aware based protocols to work in the outdoor environment. Besides, the acquisition of location information will introduce computation delay, extra message overhead and energy consumption at each node. For this reason, the authors in [11] presented the XTC algorithm which is one of the few topology control protocols which are location-free. The XTC algorithm consists of three steps. In the first step, each node broadcasts once at the maximum power and then ranks all its neighbors according to its link quality to them (from high to low). Each node transmits its ranking results to neighboring nodes during the second step. In the final step, each node locally examines all of its neighbors in the order of their ranking and decides which one needs to be directly linked. The XTC algorithm features the basic properties of topology control such as symmetry and connectivity while running faster than most previous algorithms.

Although many efforts have targeted the topology control issue, most of them [5][6][7][8][9][10][11] focus on the network layer and fail to consider the interference or collision problems that can occur at the medium access control (MAC) layer. The potential collision, which is the well-known hidden terminal problem, occurs when each node does not have the same transmission power in the resulting topology after power control. For example, in Fig. 1, assume that the resulting topology after the power control protocol is constructed by five nodes and links (solid edges) between them. If node *A* intends to transmit data packets to *B*, it initially sends a RTS packet at a determined power (the dotted circle centered at *A*). Node *B* replies with a CTS at its determined power (the dotted circle centered at *B*) to ensure that all its one-hop neighbors (e.g., *A* and *E*) can overhear. Then, node *A* starts to transmit the data packets to *B*. Nodes *C* and *D*, which lie outside the reserved floors of nodes *A* and *B*, will not receive the RTS/CTS message exchanged by nodes *A* and *B*. As *C* uses its determined power (the dotted circle centered at *C*) to send packets to the direct linking neighbor *D*, it causes a collision at node *B*. The hidden terminal problem is due to the asymmetric link between nodes *B* and *C*.

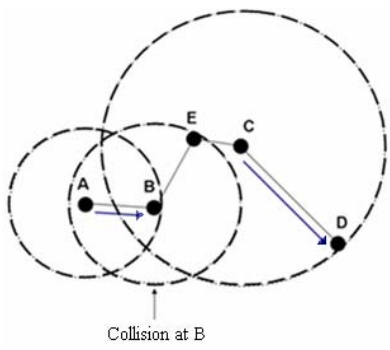


Figure 1: Hidden terminal problem

Unfortunately, the problem cannot be overcome by using the standard RTS/CTS control packets mechanism which control the packets transmission power is the same way as the data transmission power. Some MAC layer power control protocols have been proposed to solve the hidden terminal problem [12]. In their protocols, each node transmits control packets using the common maximum power to avoid the hidden terminal problem. Nevertheless, the corresponding cost is the reduction of spatial channel reuse. This is because a node will restrain all nodes in the neighborhood from transmitting if it is in the communication period, even though the neighboring nodes' transmissions would not interfere with its correct message receipt.

In our paper, we propose a distributed protocol, named *Location-Free Topology Control* (LFTC) protocol, to deal with the topology control issue at the network layer and the hidden terminal problem at the MAC layer. The LFTC protocol is not only simple but it also has a low messages exchange overhead. In addition, the LFTC allows multiple communications to be carried out concurrently in the neighborhood. Finally, the connectivity property is preserved in the protocol.

The rest of the present paper is organized as follows. Section 2 presents our protocol. Section 3 evaluates the performance of our protocol in simulations. Section 4 concludes the paper.

II. Location-Free Topology Control Protocol

In this section, we will present a protocol that constructs a power-efficient network topology while avoiding the potential collision due to the hidden terminal problem. We have the following assumptions for our protocol. Given a set V of n nodes deployed in a two-dimensional area, the network topology is denoted as a graph $G = (V, E)$ in the plane which is constructed by having each node use the common maximum transmission power $P_{\max}$. E is the edges (links) set in which an edge $(u, v) \in E$ if nodes u and v can directly communicate using the power $P_{\max}$. Every node has a unique *ID* and could communicate to other nodes through an omni-directional antenna. None of the nodes are aware of their exact coordinates and relative distance to their neighbors in the area. However, the signal from the other nodes could be accurately received and the received signal power could be exactly measured with the help of the radio interface in each node.

The initial topology $G = (V, E)$ before topology control must be connected, i.e., there exists at least one path between any pair of nodes in G. The minimum power for a node u to directly communicate with a node v is denoted by P_{uv}. Here we take the model presented in [12] for node u to determine the power P_{uv} when u receives a message from v and that v's transmission power is known by u. The energy cost for u to send one packet to v is denoted by $C(P_{uv})$ which can be obtained by the power P_{uv}. Transmission medium is symmetric in our assumptions. Therefore, $P_{uv} = P_{vu}$ and $C(P_{uv}) = C(P_{vu})$. The common maximum transmission power of each node is denoted by $P_{\max}$. All nodes in the network are capable of changing their transmission power below the value of $P_{\max}$. In addition, we assume that there is an underlying MAC layer to solve the wireless contention problem.

Our protocol consists of two phases: the first phase is *link determination phase*, and the second one is *interference announcement phase*. In the *link determination phase*, each node, say u, independently selects a set of its next-hop nodes from all of its neighbors according to a power-efficient strategy. In the *interference announcement phase*, some nodes will actively inform their neighbors in advance if they consider themselves as potential interferers who can affect the ongoing communications of their neighbors.

A. Link Determination Phase

In the first phase, each node independently decides its *direct communication set* (DCS) whenever it receives a "*hello*" message from a neighbor. A node u will directly communicate with its neighboring node v if there is no *common neighbor* node i (denoted by CN_i) of u and v such that messages sent from u to v via i ($u \rightarrow i \rightarrow v$) has a lower total energy than the energy required from u directly to v ($u \rightarrow v$). Each node would randomly broadcast a "*hello*" message once using the maximum power $P_{\max}$ at any moment during the first phase. While a node u hears a "*hello*" message from a neighbor v, it immediately computes for P_{uv} and $C(P_{uv})$ since the transmission power of a "*hello*" message is a constant, $P_{\max}$. Every "*hello*" message contains the sender *ID* and a specific data structure of the sender which is referred to as the *vicinity table*.

There are four fields in the *vicinity table* as shown in Fig. 2. The first field, *neighbor_ID,* records the node's *ID* if a node, say u, overheard the "*hello*" message sent from a node v. The field of *direct_comm_cost* stores the $C(P_{uv})$ which is the required cost when u directly communicates with v. The *min_comm_cost* records the minimum communication energy cost from node u to node v. The value in this field would be dynamically updated whenever node u learns of a less-energy path for it to communicate with node v. Note that, the communicating path between nodes could be direct (one-hop) or indirect (multi-hop). The last field *link_type* indicates whether the neighbor v belongs to the DCS(u) or not. If marked as "*d*", node v is a next-hop node of node u (u can directly link to v); otherwise, v is an indirectly communicating neighbor of u and is marked as "*i*" in the *link_type* field.

neighbor_ID	*direct_comm_cost*	*min_comm_cost*	*link_type*
v	*5*	*5*	*d*

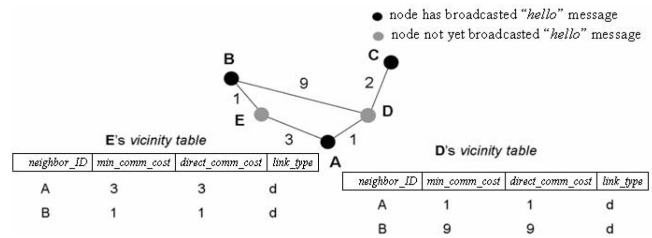
Figure 2: The *vicinity table* of a node *u*.

The content in the *vicinity table* in each node is empty at the beginning. Upon overhearing a "*hello*" message from any node, a receiver inserts a new record and updates the fields of its table according to the collected information in the "*hello*" message. Assuming that a node u hears a "*hello*" message sent from node v, it will insert a record of v into its *vicinity table* and act as follows. If there is no CN between nodes u and v , node u records node v as one of its next-hop neighbors ($v \in$ DCS(u)). The *link_type* of v in u's *vicinity table* (written as $link\_type_u(v)$ for simplicity) is marked as '*d*'. Obviously, the value in *min_comm_cost* of v in u's *vicinity table* (written as $min\_comm\_cost_u(v)$) is the same as the value of *direct_comm_cost* of v (written as $direct\_comm\_cost_u(v) = C(P_{uv})$), which represents the temporary minimum energy consumption of communication cost from u to v by direct transmission.

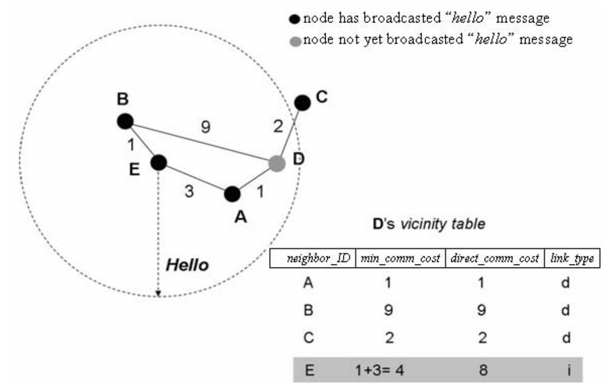
If there exist some CNs with node v, node u will check the type of each CN i and update the *vicinity table* accordingly. There are three types of CNs in our protocol. If the summation of $min\_comm\_cost_u(i)$ in u's table and $min\_comm\_cost_v(i)$ in v's table is smaller than the $direct\_comm\_cost_u(v)$, node i is a relay CN and node u has a power-efficient path to node v via node i. In this case, node u would exclude v from its next-hop neighbors ($v \notin$ DCS(u)) by marking the $link\_type_u(v)$ as "*i*". Thus, the minimum communication cost from u to v ($min\_comm\_cost_u(v)$) is replaced by the summation of $min\_comm\_cost_u(i)$ and $min\_comm\_cost_v(i)$. If node i is not a relay CN, node u would compute the summation of $min\_comm\_cost_u(v)$ and $min\_comm\_cost_v(i)$, which is equal to the minimum communication cost from u to i via v ($u \rightarrow \ldots \rightarrow v \rightarrow \ldots \rightarrow i$). If the summation is smaller than the $min\_comm\_cost_u(i)$, node i is a benefited CN and node u has a power-efficient path to node i via node v. Therefore, the $min\_comm\_cost_u(i)$ is updated to the summation of $min\_comm\_cost_u(v)$ and $min\_comm\_cost_v(i)$ and node i does not belong to u's next-hop nodes ($j \notin$ DCS(u)). If the CNs do not belong to the above-mentioned types, they are called the irrelevant CNs. The irrelevant CNs will not cause any effect on the communication cost of node u with its neighbors.

For example, assuming nodes D and E have received the "*hello*" messages from some of their neighbors and established their *vicinity tables* as shown in Fig. 3(a). Then in Fig. 3(b), node E broadcasts a "*hello*" message at $P_{\max}$ including the information of its *vicinity table*. Once D receives the "*hello*" message from E, it computes the $C(P_{DE})$ and puts the value into $direct\_comm\_cost_D(E)$. Assume that the value of $C(P_{DE})$ is equal to 8. There exist two CNs, A and B between nodes D and E. Since the summation of $min\_comm\_cost_D(A)$ in D's table and $min\_comm\_cost_E(A)$ in E's table is smaller than the $direct\_comm\_cost_D(E)$, node A is a relay CN and node D has a power-efficient path to node E via node A. The *min_comm_cost* of node D to node E is equal to 4 as shown in Fig. 3(b). Fig. 3(c) illustrates that the communication power consumption from D to B can be further reduced after the $min\_comm\_cost_D(E)$ is determined. The $min\_comm\_cost_D(B)$ is updated to 5 and node B becomes the benefited CN. Accordingly, the field $link\_type_D(B)$ is changed to "*i*".

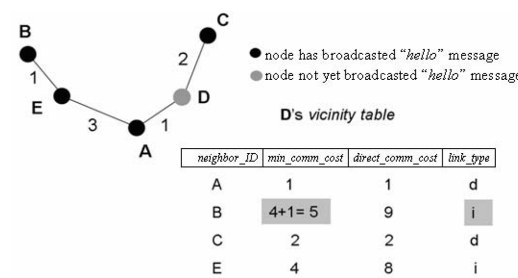
After a node u received the "*hello*" messages from all its neighbors, it could determine the DCS(u) and $P_{\text{data}}(u)$ from its *vicinity table*. Node u determines node v as its next-hop neighbor if the $link\_type_u(v)$ is marked as "*d*". The determined transmission power of node u, $P_{\text{data}}(u)$, is the value that it can directly communicate with all of its next-hop nodes in the DCS(u). Note that the edges (links) constructed in the resulting topology are bi-directional. If node u and v in the resulting topology have an edge between them, the power-efficient way to communicate with each other is through the direct link. However, having no edge between two nodes does not always represent that the transmission power of one node cannot reach to the other node, because it just shows sometimes that an asymmetric link exists between them (for example, node B and node C in Fig. 1).



(a)



(b)



(c)

Figure 3: Example of *link determination phase* (a) *vicinity tables* of nodes D and E (b) a record of new neighbor E is inserted into D's *vicinity table* with *min_comm_cost* = 4 (c) the *min_comm_cost* from node D to node B is updated to 5.

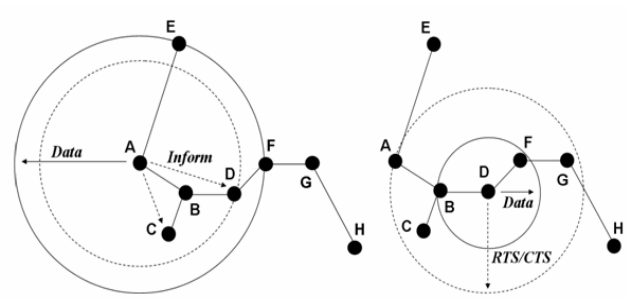
B. *Interference Announcement Phase*

After all the nodes have broadcast the "*hello*" messages and the optimal data transmission power at each node has been determined, each node starts to execute the *interference announcement phase*. This phase avoids data collision resulting from the hidden terminal problem when nodes use the power determined by the previous phase to transmit the data packets. The prevention mechanism proposed here is similar to the 802.11 protocol which takes advantage of the RTS/CTS control packets. However, the difference with this method is that each node can determine a more appropriate power P_{control}, to transmit the control packet. As we mentioned in the section on the previous work, using an excessive control packet power has the adverse effect of hindering data transmission from occurring in the neighborhood and results in a low system throughput.

In the beginning, the control power of each node u, denoted by $P_{control}(u)$, is set to the same value as its determined data transmission power $P_{data}(u)$. In the meantime, u has to judge if it will cause interference with an ongoing transmission in its indirect neighbors in the future. It is not necessary for node u to consider that it would interfere with the data transmission of its direct link neighbors because the RTS/CTS packets sent by these neighbors before sending their data can be overheard by u. However, it probably causes a hidden terminal problem since node u does not receive the control packet in advance from the indirect nodes, and starts its transmission. This case has been described in the Fig. 1, whereby the interferer, node C, causes collision to its indirect neighbor B. To determine whether it is a potential interferer to the indirect linking neighbors or not, node u needs only to observe its resulting *vicinity table* formed in the first phase. If it finds that any node $v \notin$ DCS(u) and the direct transmission power P_{uv} is less than the determined transmission power $P_{control}(u)$, node u realizes that it is a potential interferer to node v.

If a node u is aware that it is a potential interferer to some indirect linking neighbor v, it will notify the interfered node v in advance. The notification performed by u is to broadcast an "*Inform*" message including the sender *ID*. When broadcasting the "*Inform*" message, it is unnecessary for node u to use P_{max} to notify all of its neighbors. Instead, the transmission power of the "*Inform*" message should be just high enough to reach to the farthest interfered node. When a node v overhears an "*Inform*" message from a node u, node v would check whether the sender u is its direct neighbor or an indirect one. If the "*Inform*" message is sent from the direct neighbor $u \in$ DCS(v), v just ignores the message. Otherwise, if $u \notin$ DCS(v), v compares its current $P_{control}(v)$ to P_{vu}. If $P_{control}(v) < P_{vu}$, $P_{control}(v)$ is set to P_{vu} in order to ensure that the RTS/CTS control packets sent out by v can be correctly heard by node u.

An example of the *interference announcement phase* is illustrated in Fig. 4. Fig. 4(a) shows that node A realizes that its transmission (the range of data transmission power is indicated by solid circle) may affect the ongoing transmission of nodes C and D. Therefore, it broadcasts an "*Inform*" message to notify them in advance (the range of the "*Inform*" message is indicated by the dashed circle). In Fig. 4(b), assuming node D received the "*Inform*" messages from nodes A and G. Hence, D would magnify the RTS/CTS control packet power, $P_{control}(D)$ to prevent all its potential interferers such as A and G from starting their transmissions.



(a) (b)

Figure 4: (a) *A* broadcasts an "*Inform*" message to the interfered neighbors *C* and *D* (b) *D* broadcast a control packet to inform node *A* before data transmission.

A summary of our protocol is presented as follows:

The algorithm of LFTC

Link Determination Phase:

Each node broadcasts a "*hello*" message with its *vicinity table*.
If a node u receives the "*hello*" message from its neighbor v;
  If u has a power-efficient path to node v via node i
    Insert (v, $direct\_comm\_cost_u(v)$, $min\_comm\_cost_u(i)$ + $min\_comm\_cost_v(i)$, ***i***) into u's *vicinity table*;
  Else
    Insert (v, $C(P_{uv})$, $C(P_{uv})$, ***d***) into u's *vicinity table*;
  end If
If u has a power-efficient path to node i via node v
  Update the record of i in its *vicinity table* with (i, $direct\_comm\_cost_u(i)$, $min\_comm\_cost_u(v)$ + $min\_comm\_cost_v(i)$, ***i***);
end If
If node u received the "*hello*" messages from all its neighbors in the link determination phase, it will adjust the transmission power $P_{data}(u)$ such that u can directly communicate with all of its neighbors in its *direct communication set* DCS(u).

Interference Announcement Phase:

Set $P_{control}(u) = P_{data}(u)$;
If u is a potential interferer to the indirect linking neighbors v
  u will broadcast an "*Inform*" message to inform the interfered nodes;
End if
If node u overhears an "*Inform*" message from node v
  If v is not a direct neighbor of u
    If $P_{control}(u) < P_{uv}$
      Set $P_{control}(u) = P_{vu}$ to ensure u's RTS/CTS control packets can be correctly heard by v;
      If u is a potential interferer to an indirect linking neighbor i
        u will broadcast an "*Inform*" message to inform the interfered node i;
      End if
    End if
  End if
End if

III. SIMULATION RESULTS

To evaluate the performance of the proposed LFTC protocol, we utilized a simulator in GloMoSim[14]. Ad Hoc nodes are randomly distributed in a square region of 1000 m x 1000 m. The maximum transmission power of each node is 15 dBW, while the receive threshold is -85dBW. The raw transmission bandwidth is assumed to be 2 Mbps. In the simulation environment, the adopted path loss model is $1/d^2$, and the maximum communication range of each node is up to 250 m. For comparison purposes, we take the CSMA/CA MAC protocol with the RTS/CTS mechanism. The AODV routing protocol [13] is also used and slightly modified to find the minimum energy consumption paths instead of the shortest paths between two end nodes. The XTC topology control algorithm [11] is chosen as our comparison candidate since it has the same advantage as our protocol which works without the aid of either directional or location information. Since the XTC algorithm only decides on one power, then the transmission power for data and RTS/CTS control packets at each node is assumed to be the same in our simulation.

Before proceeding to compare and analyze the results of the XTC algorithm and our proposed LFTC protocol based on several performance metrics, we would like to understand more the characteristics of the resulting topology of both protocols. Fig. 5(a) shows the original network topology of 100 nodes, wherein the nodes use the maximum power to communicate with the others. Fig. 5(b) and 5(c) represent the resulting topologies generated by the XTC algorithm and our LFTC protocol, respectively. Based on the average node's degree in these topologies, we discuss their pros and cons. The average node's degree should not be too large nor too small. A large degree implies an increase of interference and collision, as well as an unnecessary energy waste such as the topology shown in Fig 5(a). A small degree also tends to increase the overall network power consumption because longer paths have to be taken from end to end. Note that the topology of our protocol has a higher average degree than the one in the XTC. This is because each node in our protocol considers the best power-efficient links to every neighboring nodes, while the nodes in the XTC only allows a minimum number of links to connect to the closer neighbors.

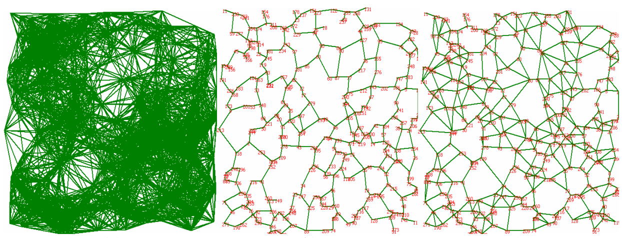


Figure 5: Network topologies of 100 nodes constructed by (a) without topology control (b) XTC protocol (c) LFTC protocol.

For any pair of nodes in the resulting topology, we use the modified AODV protocol to find out the route which has the minimum total power consumption from among all the paths between them. For simplicity, we referred to this route as the least cost path. The energy cost in the least cost path is the summation of the data transmission power of all transmitting nodes which participate in the data dissemination along the path. The power depletion of the RTS/CTS control packets is not included in the energy cost of the path. The following two figures show the network density impacts on the hop counts and energy cost in the least cost path. The number of ad hoc nodes varies from 50 to 300 in the 1000 m x 1000 m region.

Fig. 6 compares the average hop counts in our protocol with the one in the XTC approach. While the hop counts in both protocols increase as the network density becomes denser, the hop counts in our protocol are obviously less than that in the XTC. It is significant to note that the difference in the average hop counts of the two algorithms grows from about 2 hops in the 50-node network environment to around 5 hops in the 300-node environment. It reveals that the denser the network size, the better the performance of the resulting topology generated by our protocol when one node sends packets to another node. This is because having fewer hop counts means less delay time in the multi-hop wireless transmission.

Fig. 7 shows the average energy cost of the least cost paths between all node pairs in the network. It can be observed that our protocol has similar simulation results to that performed by the XTC algorithm. Since the nodes in our protocol have slightly higher average degrees than those of the XTC, it implies that the decided data transmission power of nodes in our protocol could be slightly more than that in the XTC. However, our protocol has the benefit of having fewer hop counts in the least cost path. Overall, the LFTC performs just as good as the XTC algorithm as regards the average energy cost in the least cost path.

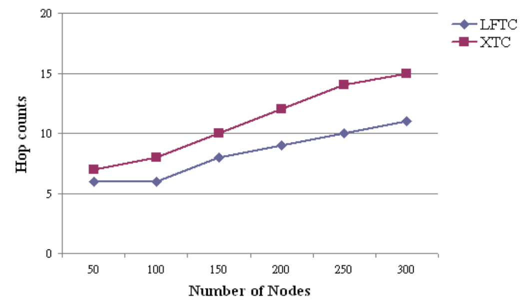


Figure 6: Average hop counts in various network densities.

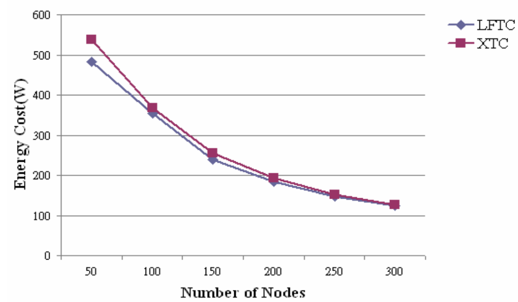


Figure 7: Average energy cost of the least cost paths between all node pairs in various network densities.

The prevention mechanism for collision avoidance (*interference announcement phase*) is integrated with the topology control algorithm (*link determination phase*) to obtain the LFTC protocol. Then we compared the ratio of collision of the LFTC with the XTC in an environment wherein 100 nodes are deployed in a 1000 m x 1000 m region. Note that in the XTC, it is assumed that the nodes use their determined power to transmit both the data and the control packets. In Fig. 8, the x-axis represents the number of end-to-end paths (or number of source nodes) in which data packets are disseminating simultaneously. The y-axis is the ratio of total collisions of data packets. In the simulation, source nodes are randomly chosen and they send a data packet once. Retransmission mechanism is not implemented when collisions occur. Our protocol performs close to 100 percent in terms of successful data receiving ratio regardless of how many concurrent data transmission paths exist in the network. On the contrary, the collision ratio in the XTC increases smoothly in the beginning as the number of paths increases. Then the ratio boosts as the number of paths continues to rise. When there are 50 concurrent transmission paths in the network, the ratio of collisions in the XTC reaches up to 75%, which means that only 1/4 of the total transmitted data packets can be received successfully. It proves that our protocol effectively avoids the problem of collision and enhances the overall performance.

The throughput of the LFTC and the 802.11 standard in a 100-node network is shown in Fig. 9. In the figure, we vary the number of source nodes which generate data packets, wherein each data packet is assumed to be 256 bytes. It can be observed that the throughput of our LFTC is, on the average, about 1.7 times higher than that of the 802.11 standard. The increase in channel utilization is due to the fact that multiple transmissions can proceed simultaneously in the LFTC protocol. However, in the case of the 802.11 standard, only one transmission at a time proceeds in the transmitter and receiver's communication range since all their neighbors are within the carrier-sense range of each other.

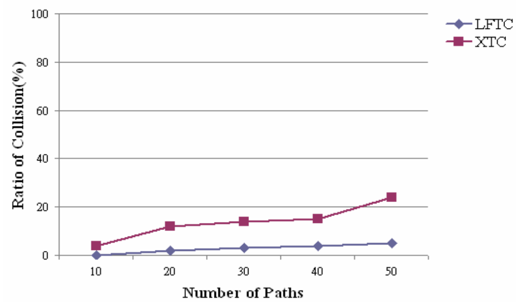


Figure 8: Ratio of collisions in 100 nodes network.

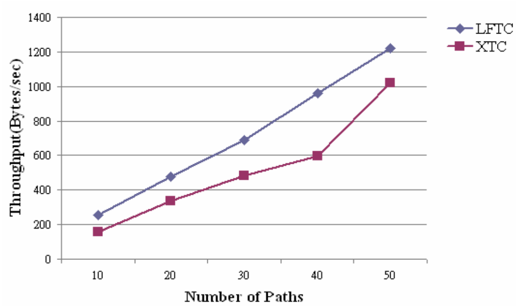


Figure 9: Throughput performance of LFTC and 802.11 protocols.

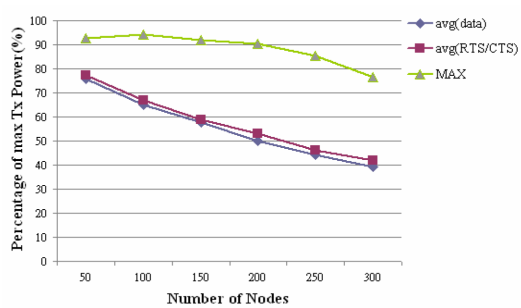


Figure 10: Percentage of transmission power of data and control packets in the LFTC protocol.

It is assumed that the percentage of transmission power is 100% if a node uses the maximum power $P_{\max}$ for communication. In our LFTC protocol, Fig. 10 illustrates the percentage of the average and highest transmission power of data/control packets for all nodes over different network densities. It can be observed that the percentage of the average transmission power for either data or control packets is less than 60% of $P_{\max}$ if our protocol is used. When the network is denser, the performance of energy conservation is more prominent. It is also noticeable that the average RTS/CTS transmission power is a little higher than the average data transmission power. It reveals that only a slightly higher power for control packets in some nodes is enough to block the potential interferers in the neighborhood from beginning their transmissions. Instead of using the maximum power to transmit the RTS/CTS control packets, our protocol can significantly conserve more energy in most of the nodes.

IV. Conclusions

In this paper, we proposed a two-phase LFTC protocol which deals with topology control and provides the mechanism to prevent the hidden terminal problem after the topology is constructed. Above all, our protocol is a location-free protocol. Each node is able to determine two optimal powers: one for data transmission and another for control packets transmission. To ensure that every transmitted data packet would be received intact without any interference, we used the RTS/CTS mechanism in IEEE 802.11 but with a slight modification. The node does not necessarily use the maximum power in sending the RTS/CTS in order to defer all of its neighbors' transmissions.

Simulation results proved that our LFTC protocol has fewer hop counts, lower control packet overhead and smaller ratio of collisions when compared to another location-free protocol such as the XTC. In addition, it also surpasses the 802.11 standard by about 1.7 times in throughput. The percentage of average transmission power in all the nodes in our protocol is varies from 60 percent to 20 percent of the maximum power when the number of nodes in the network varies from 50 to 300. Thus, our LFTC protocol has a good energy conservation performance.

References

[1] T.-C. Hou and V. O. K. Li, "Transmission range control in multiple packet radio networks," *IEEE Trans. Communications*, Vol. 34, No. 1, pp.38-44, Jan. 1986.

[2] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Information Theory.*, Vol. 46, No. 2, pp. 388-404, Mar. 2000.

[3] S. Narayanaswamy, V. Kawadia, R. S. Sreenivas, and P. R. Kumar, "Power control in ad-hoc networks: Theory, architecture, algorithm and implementation of the COMPOW protocol," in *Proc. of the European Wireless Conf.*, Florence, Italy, pp. 156-162, Feb. 2002.

[4] M. Burkhart, P.v. Rickenbach, R. Wattenhofer, and A. Zollinger, "Does topology control reduce interference?," in *Proc. of the ACM Int'l Symp. on Mobile Ad-hoc Networking and Computing*, pp. 9-19, Tokyo, Japan, May 2004.

[5] R. Ramanathan and R. Rosales-Hain, "Topology control of multihop wireless networks using transmit power adjustment," in *Proc. of the 19th INFOCOM,* pp. 404-413, Tel Aviv, Israel, Mar. 2000.

[6] E.L. Lloyd, R. Liu, M.V. Marathe, R. Ramanathan, and S.S. Ravi, "Algorithmic aspects of topology control problems for ad hoc networks," *Mobile Networks and Applications*, Vol. 10, No. 1-2, pp. 19-34, Feb. 2005.

[7] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc, "Power consumption in packet radio networks," in *Proc. of the 14th Symp. on Theoretical Aspects of Computer Science*, pp. 363-374, Feb. 1997.

[8] N. Li, and J. C. Hou, "Localized fault-tolerant topology control in wireless ad hoc Networks," *IEEE Trans. Parallel and Distributed Systems*, Vol. 17, No. 4, pp.307-320, Apr. 2006.

[9] N. Li, J.C. Hou, and L. Sha, "Design and analysis of an MST-based topology control algorithm," in *Proc. of the IEEE Conf. on Computer Communications*, pp. 1702–1712, San Francisco, CA, USA, Apr. 2003.

[10] C.-C. Shen, C. Srisathapornphat, R. Liu, Z.Huang, C. Jaikaeo, and E.L. Lloyd, "CLTC: A cluster-based topology control Framework for ad hoc networks," *IEEE Trans. Mobile Computing*, Vol. 3, No. 1, pp. 18-32, Jan.-Mar. 2004.

[11] R. Wattenhofer and A. Zollinger, "XTC: A practical topology control algorithm for ad-hoc networks," in *Proc. of the 4th Int'l Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks*, pp. 26-30, Santa Fe, New Mexico, USA , Apr. 2004.

[12] S.-L. Wu, Y.-C. Tseng, and J.-P. Sheu, "Intelligent medium access for mobile ad hoc networks with busy tones and power control," *IEEE J. Selected Areas in Comm.*, Vol.18, No. 9, pp. 1647-57, Sep. 2000

[13] C. E. Perkins, and E. Royer, "Ad-hoc on-demand distance vector routing," IETF RFC3561, 2003.

[14] X. Zeng, R. Bagrodia, and M. Gerla, "Glomosim: A library for parallel simulation of large-scale wireless networks," in *Workshop on Parallel and Distributed Simulation*, pp.154–161, Alberta, Canada, May1998.