Construction of low weighted and fault-tolerant topology for wireless ad hoc and sensor network

Hong Zeng, Jianhui Zhang* and Guojun Dai

School of Computer Science and Technology,
Hangzhou Dianzi University,
Hangzhou, Zhejiang 310018, China
Fax: +86 571 86878537
E-mail: jivon@hdu.edu.cn
E-mail: jhzhang.zju@gmail.com
E-mail: daigj@hdu.edu.cn
*Corresponding author

Abstract: To preserve network connectivity is an important issue especially in wireless ad hoc and sensor networks (WASN), where wireless links are easy to be disturbed and tiny sensors are very easy to fail accidently. Therefore, it is necessary to design a fault-tolerant network. A feasible method is to construct a k-connected (k-vertex connected) topology. In this paper, we consider k-connectivity of wireless network and propose a simple global algorithm (GAFT$_k$) which preserves the network k-connectivity and reduces the maximal transmission power (TP). The average degree expectation of the topology generated by GAFT$_k$ is $O\left((k+3)^2\right)$ and the expected weight is $O\left(\frac{(k+3)^2}{12\sqrt{\pi n n_e}}\right)$. Based on GAFT$_k$, we further propose an efficient localised algorithm (LAFT$_k$) which preserves k-vertex connectivity while maintaining bi-directionality of the network. In addition, both algorithms achieve significant reductions in energy consumption. Our simulation results show that GAFT/LAFT have better performance than some other current fault-tolerant protocols.

Keywords: wireless ad hoc and sensor networks; network connectivity; k-connected topology; fault-tolerant topology.

Reference to this paper should be made as follows: Zeng, H., Zhang, J. and Dai, G. (2013) 'Construction of low weighted and fault-tolerant topology for wireless ad hoc and sensor network', *Int. J. Sensor Networks*, Vol. 14, No. 4, pp.197–210.

Biographical notes: Hong Zeng received the BE in Computer Science and Technology from Zhejiang University and ME in computer application technology from Hangzhou Dianzi University, Hangzhou,Zhejiang,China, in 1998 and 2005, respectively. He is an Assistant Professor at Hangzhou Dianzi University, Hangzhou, China. His research interests include wireless sensor network, circuit and system, embedded system, control theory and algorithms. He is a member of IEICE.

Jianhui Zhang received the PhD and the BS from Zhejiang University and Northwestern Polytechnical University, China in 2008 and 2000, respectively. He is an Associate Professor of Computer Science at the Hangzhou Dianzi University now. His research interests include algorithm design and analysis, distributive systems, wireless and sensor networks. He is a member of IEEE/ACM.

Guojun Dai received the PhD and the BS from Zhejiang University, China in 1998 and 1990, respectively. He is a Professor of Computer Science at Hangzhou Dianzi University now. His research interests include distributive systems, wireless and sensor networks and embeded system. He is a senior member of IEEE/ACM.

This paper is a revised and expanded version of the conference paper 'A simple algorithm for fault-tolerant topology control in wireless sensor network' presented at *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, French Riviera, France, 14–18 September, 2008.

1 Introduction

Fault tolerant topology control (TC) is an important issue in WASN, since in such network, wireless nodes and links are easy to fail and each sensor device only has limited power and computational resource. Many topology control algorithms have been designed to maintain network connectivity, to reduce energy consumption and to improve network capacity (Santi, 2005). However, only a few of them have considered the fault-tolerance, such as CBTC$_k(\alpha)$ (Bahramgiri et al., 2002) and FGSS/FLSS (Li and Hou, 2004).

The CBTC$_k(\alpha)$ provides fault tolerant topology, which is an extension of CBTC (Li et al., 2001). In CBTC$_k(\alpha)$, every node increases its transmission power (TP) until either the

maximum angle between its two consecutive neighbours is at most α (a constant depended on k) or its maximal power is reached. Therefore, the transmission power would be adjusted to a excessive value. Furthermore, a cone-based antenna may not obtain exact cone information because the radio is easily disturbed by the surrounding.

Fault tolerant local spanning sub-graph (FLSS$_k$) can preserve k-connectivity and maintain bi-directionality (Li and Hou, 2004). It is based on a min-max optimal centralised algorithm, Fault tolerant Global Spanning Sub-graph (FGSS$_k$). The main idea of FGSS$_k$ is that it arranges all edges in the network in ascending order of weight, and for each edge (N_u, N_v), it checks whether node N_u is k-connected to node N_v. If not, it adds the edge (N_u, N_v). Whether FGSS$_k$ is k-connected needs to be tested by using network flow techniques (NFT) (Even and Tarjan, 1975), which increases the complexity and cost of the algorithm. It is inefficient to test the k-connectivity by using NFT, since the time and space complexities of both FGSS/FLSS then will be very high. Note that they check the vertex-connectivity every time when they need to decide whether an edge can be put into their topologies. In a k-connected topology with totally n nodes, there is at least $\frac{kn}{2}$ edges, which means that FGSS need to check the vertex-connectivity at least $\frac{kn}{2}$ times. If we adopt Esfahanian's algorithm (Esfahanian and Hakimi, 1984) whose complexity is lower than that of Even's Even and Tarjan (1975), FGSS will require $\frac{kn}{2}[n-\delta-1+\frac{1}{2}k(2\delta-k-1)]$ times calls of MFA (max flow algorithm). Here, δ is the minimum vertex degree of the network. In fact, the time complexity is much greater than these lower bounds. Notice that FLSS requires more times calls of MFA because many edges are repeatedly checked by different nodes. So these algorithms are not suitable for WASN. Furthermore, redundant edges would be inserted into the topology because some key edges for k-connectivity can be very long and locate in the rearward of the ascending weighted order.

There have been several research efforts recently on studying the necessary theoretical condition for k-connected topology (Bettstetter, 2002; Penrose, 1999) and devising approximation algorithms to construct such topologies (Bahramgiri et al., 2002; Hajiaghayi et al., 2003; Li et al., 2003). All of above fault-tolerant topology control algorithms either use unrealistic assumptions (such as, the locations of all nodes and the exact distance among them are known) or involve complex calculation. In WASN, the error of the location information obtained by wireless radio could be very big. Or to obtain the exact location information costs additional energy and time even by the range-free localisation scheme (He et al., 2003; Martins et al., 2011). It is necessary to design a TC algorithm under the requirements that the network is a non-complete graph and the calculation is simple.

In this paper, we propose two low weighted and energy efficient TC algorithms GAFT$_k$ and LAFT$_k$ (GAFT and LAFT in short) with simple calculation to construct a low-weighted and k-connectivity topology. The two algorithms need not the location information of nodes but the 'distance' among nodes. The 'distance', which does not refer to the Euclidean distance, is simply represented by the received signal strength indication (RSSI) difference among nodes. We call it as RD[1] in short. Therefore, the time and space complexity of these two algorithms are decreased. The main contributions of this paper include that:

- the topologies constructed under GAFT$_k$ and LAFT$_k$ preserve k-connectivity of the network
- the time complexity of both GAFT$_k$ and LAFT$_k$ is low, so that sensors with limited computational and power capacity can still afford the TC protocol efficiently
- the resulting topology is energy efficient and bidirectional (after the addition of unidirectional links).

The rest of the paper is organised as follows. Section 2 introduces our models and assumptions. Section 3 presents GAFT and its performance analysis, while Section 4 gives its localised implementation LAFT. Simulation studies on both GAFT and LAFT are provided in Section 5. Section 6 concludes the paper.

2 Models and assumptions

Assume that the network is uniformly randomly deployed in a $c \times c$ area. Each node has an omni-direction antenna, which can adjust its TP discretely. Its transmission radius is denoted as r and the maximal transmission radius is denoted as $r_{\max}$. An undirected simple graph $G(V, E)$ is applied to analyse our TC algorithms, where V is the set of all nodes in the network and $E = \{(N_u, N_v) | d(N_u, N_v) \leq r_{\max}, N_u, N_v \in V\}$ is the edge set of G. There are totally n nodes in V. $d(N_u, N_v)$ denotes the RD among nodes N_u and N_v. A unique address *ID* is assigned to each node. ID assignment is researched in other papers (Nesargi and Prakash, 2002) and out of the scope of this paper. The one-hop neighbourhood of a node N_u, denoted as $H^1_{N_u}$, is the set of neighbours that N_u can directly reach by the maximal TP. We also assume that $G(V, E)$ is k-connected. Li et al. (2003) proved that, for $k > 0$ and n sufficiently larger, the probability that the given $G(V, E)$ is $(k+1)$-connected is at least $\mathrm{e}^{-\mathrm{e}^{-a}}$ when $n\pi r^2_{\max} \geq \ln n + (2k-1)\ln\ln n - 2\ln k! + 2a$ where a is a constant.

As to radio propagation model, there is no prefect model which can exactly describe the actual radio propagation till now. Since the radio propagation has inherent complexity (Zhou et al., 2004). Here, we adopt the *shadowing model* (Rapport, 2001) as an example. The following equation is given by the shadowing model.

$$P_r(d)[\mathrm{dBm}] = P_t[\mathrm{dBm}] - \overline{PL_{d_0}} - 10\alpha \log\left(\frac{d}{d_0}\right) - X_{dB}. \quad (1)$$

Here P_t is the TP level, P_r is the received power level, $\overline{PL_{d_0}}$ is the average path loss on the distance d_0 and X_{dB} is a Gaussian random variable with zero mean and standard deviation σ_{dB}, which is called the shadowing deviation and is obtained by measurement. α ($\alpha \geq 2$) is called the path loss exponent and is usually empirically determined by field measurement. d_0

is the close-in reference distance, which is determined from measurements close to the transmitter.

Notice that GAFT/LAFT do not depend on a certain radio propagation model to obtain the 'distance' information. Here the 'distance' is not the Euclidean distance but the RD. GAFT/LAFT obtain RSSI information as following. Each node N_u broadcasts a ***hello*** message with the maximal TP level P_{t0} and includes P_{t0} value in the message. If a neighbour N_v receives the message with the received power P_{r0}, it can know the TP value and ID of the transmitter. The receiving power can be calculated from RSSI tested at the receive antenna. So the RD between N_u and N_v is equivalent to $P_{t0} - P_{r0}$, which represents the 'distance' between the nodes mentioned above. Furthermore the minimal threshold power level P_{rth} for a successful receiving can be knew according to signal-to-noise ratio (SNR) and the lowest antenna receive sensitivity, the minimal threshold transmission power level P_{tth} can be obtained by the following equation.

$$\begin{cases} P_{r0} &= P_{t0} - \overline{PL_{d_0}} - 10\alpha \log\left(\frac{d_{N_u N_v}}{d_0}\right) - X_{dB} \\ P_{rth} &= P_{tth} - \overline{PL_{d_0}} - 10\alpha \log\left(\frac{d_{N_u N_v}}{d_0}\right) - X_{dB} \\ & \Rightarrow P_{tth} = P_{rth} + P_{t0} - P_{r0} \end{cases} \quad (2)$$

According to equation (2), additional distance information is not needed. In fact, we can still derive equation (2) if other radio propagation models are used. The equation is derived to adjust the TP for each node. The reason to use the equation (2) is to make GAFT/LAFT independent of a certain radio propagation model and eliminate the impact of radio irregularity (Zhou et al., 2006). It also simplifies the complexity of these two algorithms.

3 Global algorithm for fault tolerant: GAFT

We first provide a low weighted and low complexity centralised algorithm GAFT to construct a k-connected topology for a non-complete weighted graph $G(V, E)$. GAFT is based on Menger's theorem (West, 2001) that a graph is k-connected *iff* there is no cut-set $V_c (V_c \subseteq V)$ with $|V_c| \leq k - 1$. Here $|*|$ denotes the size of the set $*$.

The basic idea of GAFT is that it grows the k-connected subgraph from a single node step by step and in each step, a node inside the subgraph invites new nodes joining the subgraph. There are two criterias for the selection of new nodes (or links):

- adding the new nodes (or links) can improve the connectivity of the subgraph to satisfy the k-connected requirement
- if the subgraph already satisfies connectivity requirement or multiple links can improve the connectivity, select the shortest links.

In GAFT, V_k is the set of nodes currently in the subgraph, while V_f is the set of those who already connect to at least k neighbours in the subgraph. Assume that GAFT begins from the node N_0. Initially, $V_k = N_0$ and $V_f = \varnothing$. Then N_0 connects with its k closest neighbours. If defining N_i and its closest connected neighbours as a set S_{N_i}, we can obtain S_{N_0}. Then we merge S_{N_0} into V_k, i.e., $V_k := \cup S_{N_0}$. Since N_0 already has k neighbours being in V_k, we then include N_0 in V_f. If all nodes are in V_f, i.e., $V = V_f$, GAFT terminates. Next, node N_i ($N_i \in V_k \cap \overline{V_f}$) tries to connect with its k closest neighbours. N_i selects the neighbours who can improve the connectivity of the subgraph (nodes in V_k) with higher priorities. If $S_{N_i} \subseteq V_k$, GAFT has to find another node, N_j ($N_j \notin V_k$), which is the closest to the nodes in V_f. This can guarantee that the subgraph (V_k) is growing. $V_k := \cup\{N_j\}$ and $V_f := \cup\{N_i\}$. If $S_{N_i} \nsubseteq V_k$, N_i only selects those neighbours in V_k (thus $S_{N_i} = S_{N_i} \cap V_k$). Then GAFT selects the remaining $k - |S_{N_i}|$ from those nodes who are not in V_k and are closest to the nodes in V_f. Assume that these nodes are N_k. GAFT lets $V_k := \cup\{N_k\}$. In this case, N_i would not be included in V_f, since it still need $k - |S_{N_i}|$ connections. This whole process will terminate until all nodes are in V_f. The detailed algorithm is given in Algorithm 1. Theorem 1 guarantees that the G_k obtained by GAFT is k-connected.

Algorithm 1: GAFT

```
   Input  : G(V, E), a simple connected graph;
1  ;
   Output: G_k(V, E_k), a k-connected spanning subgraph of
           G;
2  ;
3  begin
4      E_k := ∅; V_k := S_{N_0}; V_f := ∅;;
5      while N_i ∈ V_k & N_i ∉ V_f do
6          N_i selects its k "closest" neighbors, i.e., S_{N_i};;
           /* Here, N_i selects the closest neighbors
              among who can improve the k-connectivity
              in V_k first. If connectivity is
              guaranteed, then it selects the closest
              neighbors;                                */
7          ;
8          if S_{N_i} ⊆ V_k then
9              Find a node N_j ∉ V_k, which are closest to a
               node in V_f;;
10             V_k := ∪{N_j};;
11             V_k := ∪S_{N_i};;
12             E_k := ∪{e(N_i, N_l)}, N_l ∈ S_{N_i};;
13             ;
14             V_f := ∪{N_i};;
15             Continue;
16         end
17         else
18             S_{N_i} = S_{N_i} ∩ V_k;;
19             Connect to k − |S_{N_i}| neighbors closest to
               nodes in V_k;
20         end
21     end
22     Adjust all nodes' TP according to Algorithm 2;;
23 end
```

Theorem 1: *Let $G = (V, E)$ be a graph on $V = \{N_0, N_1, \ldots, N_{n-1}\}$ with $n \geq k + 1$. $G_k(V, E_k)$ is obtained by GAFT. If $G(V, E)$ is k-connected, then G_k is also k-connected.*

Proof: If $n = k + 1$, both of G and G_k are complete graphs. When $n \geq k + 2$, suppose that there is a minimal cut set $S_{\min} \subset V$ with $|S_{\min}| \leq k - 1$ so that $G_k - S_{\min}$

is disconnected. Let V_1 and V_2 be two distinct connected components of $G_k - S_{\min}$. Without loss of generality, suppose that nodes in V_1 join V_k earlier than those in V_2.

Before any node in V_2 joins V_k, we have $V_k \subseteq V_1 \cup S_{\min}$. Since GAFT always invites new node into V_k, there must be one moment at which a node $N_i \in V_2$ is invited into V_k. If $S_{N_i} \subseteq V_k$, at least one neighbour of N_i is in V_1 since $|S_{\min}| < k$. N_i must connect to the neighbour and then V_1 and V_2 are connected. The theorem is proved. Therefore there must be $S_{N_i} \nsubseteq V_k$. Then GAFT runs $S_{N_i} = S_{N_i} \cap V_k$ and $S_{N_i} \subseteq S_{\min}$ (otherwise V_1 and V_2 are connected). Next GAFT selects the remaining $k - |S_{N_i}|$ from those nodes who are not in V_k. Since these $k - |S_{N_i}|$ neighbours respectively connect to nodes in V_k, there are still k links between V_k and V_2. Because $|S_{\min}| < k$, there must be at least one link between N_i and V_2. V_1 and V_2 are connected.

If $S_{\min}$ separates G_k into more than two disjoint components, the above proof can be applied to every pair of components and then the k-connectivity of G_k can be obtained. □

GAFT only needs RD between nodes and does not need extra distance or location information. In fact, G may not be a complete graph in a realistic network. If the average number of nodes in the maximal cover range of a node is denoted as $n_e \triangleq n\pi\left(\frac{r_{\max}}{c}\right)^2$, each node only needs to directly communicate with n_e ($n_e < n$) neighbours. It reduces much calculation. Here if the graph is a complete graph, $n_e = n$.

To adjust the TP, several heuristic algorithms have been proposed, such as PID (Zhang et al., 2007), LINT and LILT (Ramanathan and Hain, 2000). They are more suitable for mobile networks than static ones because they can adjust the TP frequently and quickly. Nevertheless they create redundant adjustment in static networks. In terms of the immovability, a TP adjustment algorithm (TPA) is proposed to reduce communication messages as shown in Algorithm 2. Each node can determine the distance $d()$ with its neighbours from the messages, such as ***hello*** messages. According to these estimated distance, each node arranges its neighbours into a non-decreasing order and then calculates requisite TP level according to Algorithm 2 since the minimal threshold power level (denoted as P_{rth}) for a successful receive can be knew beforehand by SNR and the lowest antenna receive sensitivity. An ***affirm*** message is ultimately broadcasted to inform its neighbours of the final TP level. Each node totally broadcasts two messages (***hello*** and ***affirm***), which decreases power consumption and interference.

Theorem 2: *After running TPA, G_k is bi-directionally connected and k-connected only if G is bi-directionally connected and k-connected.*

In fact, TPA only adds some edges on G_k as shown in Algorithm 2 and makes G_k bi-directionally connected. So TPA does not decrease the connectivity of G_k.

Before giving out Theorem 3, we show a fact. In n nodes, node N_u connects with m ($m \geq k$) of them, the probability (named as $P(m, l)$) that N_v connects with l neighbours in the m nodes is $P(m, l) = \binom{m}{l}/\binom{n}{l}$.

Algorithm 2: TP adjustment–TPA

```
Input  : A set V containing n nodes;
;
Output : G(V, E), a bi-direcationally connected graph G;
;
begin
    for N_i, i ← 0 to n − 1 do
        Broadcast a Hello message with maximum TP
        level P_tmax.
    end
    if N_i receives Hello messages then
        Detect P_rN_j of these messages from
        N_j, j ∈ H^1_Ni & j ≠ i and calculates d(N_i, N_j);
        Sort its neighbors in non decreasing order;
    if N_i need communicates with the first N_j nodes then
        Calculate: P_tN_i = P_rth + P_tmax − P_rN_j ;;
        Broadcast P_tN_i by an Affirm message;
    end
    Receive the Affirm messages from its neighbors and
    know the P_tN_j of its neighbors;;
    Set its adjusted TP level as max{P_tN_i, P_tN_j};;
end
```

Theorem 3: *After running GAFT, the average degree expectation of G(V, E) is $O\left(\frac{(k+3)^2}{4}\right)$.*

Proof: Suppose that N_0 firstly starts GAFT, so $V_k = S_{N_0}$ and there are $k+1$ nodes in V_k. When N_i connects with its k nearest neighbours, it may add extra neighbours to other nodes in V_k so the 'extra degree' is created. If we consider the 'extra degree' to be in the degree of N_i, the total weight of G_k would not change. Let δ_i denote the degree of N_i. The degree expectation of N_i is $E(\delta_i) = k + \sum_{j=1}^{k} j \cdot P(|V_k|, j)$. The first node N_0 does not add extra neighbours because V_k is empty when it joins. So the average degree expectation is

$$
\begin{aligned}
E(\delta) &= \frac{1}{n}\sum_{i=1}^{n} E(\delta_i) = \frac{1}{n}\left[k + \sum_{i=2}^{n} E(\delta_i)\right] \\
&\leq \frac{1}{n}[nk + (k-1)\sum_{j=1}^{k} P(k+1, 1)j \\
&\quad + \sum_{l=k+1}^{n}\sum_{j=1}^{k} P(l, 1)j] \\
&= \frac{k^2+5k}{4} + \frac{k^2+n-k-2}{n^2} \\
&\leq \frac{k^2+5k}{4} + \frac{k^2+n}{n^2} \\
&\leq \frac{k^2+5k}{4} + \frac{(n-1)^2+n}{n^2} \leq O\left(\frac{(k+3)^2}{4}\right). \quad (3)
\end{aligned}
$$

Note that V_k grows from $k+1$ to n when nodes join it one by one. □

Theorem 3 says that each node needs $O\left(\frac{(k+3)^2}{4}\right)$ neighbours on average. The result is close to the Penrose's proof that the node degree is much close to the connectivity when the number of nodes is larger enough (Penrose, 1999). It indicates that a k-connected subgraph can be obtained when each node uses its local information to link its neighbours properly.

We also study the total weight of the topology given by GAFT. The probability that a certain node N_i lies on a particular distance d_i is proportional to the area of the cycle with radius d_i. If the deployment scenario is normalised by dividing with the size c of the deployed area, then the distance d_i becomes $R_i \triangleq \frac{d_i}{c}$ and the maximal transmission radius $r_{\max}$ becomes $R_{\max} \triangleq \frac{r_{\max}}{c}$. Now the deployment scenario becomes a unit square $[0, 1]^2$.

Theorem 4: *After normalising the deployment scenario by dividing with c, the approximation guarantee of the weight expectation is* $O\left(\frac{(k+3)^2}{12\sqrt{\pi n n_e}}\right)$.

Proof: Consider an arbitrary node N_i, the maximal transmission is $R_{\max}$ and the degree expectation of N_i is n_e. Let the maximal transmission radius be equivalently divided into n_e cycles centred at N_i with the radius R_i. That is $R_i - R_{i-1} = \frac{R_{\max}}{n_e}$ and $R_i = \frac{i \cdot R_{\max}}{n_e}$. We define the nodes in the circle with the radius R_i as the set S_i and $NM_i = S_i - S_{i-1}$. $S_1 \geq 1$ because there is at least one node N_i. If there is a neighbour N_v in NM_i, then the weight of the edge (N_i, N_v) is $R_{i-1} \leq \omega(N_i, N_v) \leq R_i$ and it can be approximately considered as $\omega(N_i, N_v) \approx (R_i + R_{i-1})/2$. Denote Ev_i as the event that N_i connects with a neighbour in NM_i, so we have the probability $P(Ev_i)) == (R_i^2 - R_{i-1}^2)/R_{\max}^2$. The weight of the edge between N_i and its neighbour N_v can be obtained by the equation: $E(\omega(N_i, N_v)) = P(Ev_1)\frac{R_1}{2} + \sum_{i=2}^{n_e} P(Ev_i)\frac{R_i + R_{i-1}}{2}$. In GAFT, the weight of each node contains two parts caused by k and possible extra neighbours. According to the proof in Theorem 3, $E(\delta_i) = k + \sum_{j=1}^{k} j \cdot P(|V_k|, j)$. So the weight of the edges between N_i and its neighbours is

$$\begin{aligned}
E(\omega(N_i)) &= \frac{E(\delta_i)}{n_e}\left[P(Ev_1)\frac{R_1}{2} + \sum_{i=2}^{n_e} P(Ev_i)\frac{R_i + R_{i-1}}{2}\right] \\
&= \frac{E(\delta_i)}{2n_e R_{\max}^2}\left[R_1^3 + R_{n_e}^3 - R_1^3 + \sum_{i=2}^{n_e} R_{i-1}R_i(R_i - R_{i-1})\right] \\
&= \frac{E(\delta_i)}{2n_e R_{\max}^2}\left[R_{\max}^3 + \sum_{i=2}^{n_e}\frac{(i-1)\cdot i \cdot R_{\max}^3}{n_e^3}\right] \\
&= \frac{E(\delta_i)R_{\max}}{6n_e^3}(4n_e^2 - 1). \qquad (4)
\end{aligned}$$

Notice that the weight of each edge is shared by two nodes. According to equation (4), the weight expectation of G_k is

$$\begin{aligned}
E(\omega(G_k)) &= \frac{1}{2}\sum_{i=1}^{n} E(\omega(N_i)) \\
&= \frac{R_{\max}}{12n_e^3}(4n_e^2 - 1)\sum_{i=1}^{n} E(\delta_i) \\
&\leq \frac{R_{\max}}{3n_e}\sum_{i=1}^{n} E(\delta_i) \leq O\left(\frac{(k+3)^2}{12\sqrt{\pi n n_e}}\right)
\end{aligned}$$

where $n_e = \pi r_{\max}^2 \frac{n}{c^2} = \pi n R_{\max}^2$, so $R_{\max} = \sqrt{\frac{n_e}{n\pi}}$. Here we do not give the detailed derivation because $E(\delta_i)$ is knew from equation (3). □

Note that the n_e is determined by the maximal transmission radius $r_{\max}$ of a node, therefore $n_e \leq n$. When $n_e = n$, the topology becomes a complete graph, the average weight of which becomes $O(k + \frac{n}{3})$. It is well-known that the weight of an MST in the above graph model is $\Theta(\sqrt{n})$ (Steele, 1992). The weight of the optimal k-connected graph is not less than $\frac{k}{2}\omega(MST) = \Theta(k\sqrt{n})$. Therefore we have an approximation ratio of $O\left(\frac{(k+3)^2}{12kn\sqrt{\pi n_e}}\right)$ under this special case.

Another remarkable property of GAFT is its time complexity. If we use binary search algorithm (BSA) to check whether the $m(m \leq k)$ neighbours of a node N_i in V_k is also in V_f, then it takes at most $mO(\log(|V_f| + 1))$ steps. If the m nodes are in V_f, N_i has to find $k + 1 - m$ node closest to V_f in the remainder $n - |V_k|$ of $V - V_k$, which costs $(k + 1 - m)O(\log(n - |V_k| + 1))$ steps. This kind of check happens with the probability $p(n - |V_k|, k)$ according to the fact in Theorem 2. So we can work out the steps in which N_i finds the $k + 1 - m$ node is $\sum_{i=n-|V_k|}^{n} P(i, k)(k + 1 - m)O(\log(n - |V_k| + 1)) < (n - |V_k|)(k + 1 - m)O(\log(n - |V_k| + 1))$. After all, we have $O(m\log(|V_f| + 1)) + (n - |V_k| - k)(k + 1 - m)O(\log(n - |V_k| + 1)) = O(m\log(|V_f| + 1)) + (n - 1 - k)(k + 1 - m)O(\log n)$ because $n - |V_k| \leq n - 1$. In other words, it takes GAFT at most $O(m\log(|V_f| + 1)) + (n - 1 - k)(k + 1 - m)O(\log n)$ steps to insert N_i into V_k. Since $|V_f| \leq n$, the complexity of GAFT is $O((n - k - 1)(k + 1)\log n)$. Li and Hou (2004) use NFT to construct topologies by the FGSS$_k$ and FLSS$_k$ algorithms. The time complexity of FGSS$_k$ is $O(m^2\sqrt{n})$, where m is the number of edges. For a k-connected topology, $m \geq \frac{kn}{2}$. It is obvious that it take makes FGSS$_k$ and FLSS$_k$ much more time on NFT. As mentioned in Section 1, it is much complex to calculate the maximal flow of the network especially when the network scale is large.

4 Local algorithm for fault tolerant: LAFT

In this section, a localised algorithm (LAFT) is given out, which implements the GAFT locally. LAFT takes $O(n_e \log n_e)$ steps and totally $O(n)$ messages to construct a k-connected topology.

In Algorithm 3, the two-hop neighbourhood of a node N_i, denoted as $H_{N_i}^2$, is the set of nodes which can be directly reach from nodes in $H_{N_i}^1$ via the maximum TP. Each node N_i has two sets $V_k^{N_i}$ and $V_f^{N_i}$. The former is the set of nodes currently in the subgraph, while the later is the set of nodes who connect to at least k neighbours in the subgraph. Each node uses two messages (*hello* and *reply*) to obtain the information about one-hop and two-hop neighbours. These messages contain the IDs and TP values among nodes. Each node collects the

information about one-hop neighbours by the *hello* messages broadcast by its one-hop neighbours. Then it broadcasts a *reply* message to tell its neighbours the information about its one-hop neighbours. Notice that TPA also need two messages: *hello* and *affirm*. The two messages are same with *hello* and *reply* used by LAFT, they contained the same information.

Algorithm 3: LAFT

Input : $G(V, E)$, a simple connected graph;
;
Output: $L_k(V, E_k)$, a k-connected spanning subgraph of G;
;
begin
  $E_k := \varnothing$; Each node N_i sets $V_k^{N_i} = S_{N_i}$ and $V_f^{N_i} = \varnothing$;;
  Each node N_i collects the information about $H_{N_i}^1$ and $H_{N_i}^2$;;
  Finds the k closest neighbors, i.e. S_{N_i};;
  while $N_i \in V_k^{N_i}$ & $N_i \notin V_f^{N_i}$ **do**
    if $S_{N_i} \subseteq V_k^{N_i}$ **then**
      Find another neighbor N_j ($N_j \notin V_k^{N_i}$ and $N_j \in H_{N_i}^1 \cup H_{N_i}^1$) closest to $V_k^{N_i}$;;
      $V_k^{N_i} := \cup\{N_j\}$; $V_f^{N_i} := \cup\{N_i\}$;
    end
    ;
    else
      $S_{N_i} := S_{N_i} \cap V_k^{N_i}$;;
      Connect with $k - |S_{N_i}|$ neighbors closest to $V_k^{N_i}$; Notice that these neighbors are in $H_{N_i}^1$ and $H_{N_i}^2$;
    end
    $E_k := \cup\{e(N_i, N_l)\}$, $N_l \in S_{N_i}$;
  end
  Adjust all nodes' TP according to **algorithm** 2;;
end

Subsequently each node runs LAFT to determine which neighbours to connect with. After that, nodes run TPA to adjust its TP level. Whereafter each node informs their neighbours of its TP level by broadcasting an *affirm* message. We can easily find that each node need broadcast only three messages (*hello*, *reply* and *affirm*) so the message complexity of LAFT for the whole network is $O(n)$.

Before proofing the topology constructed by LAFT is k-connected, we firstly proof Theorem 5. In $G = (V, E)$, suppose there are a node N_u and its two-hop neighbours N_v, $N_v \in H_{N_u}^2$. By defining a node set $V_{N_u} = \{N_u\} \cup H_{N_u}^2$ and a edge set E_{N_u}, which contains all edges connected with the nodes in V_{N_u}, we can obtain a local graph $G_{N_u} = (V_{N_u}, E_{N_u})$. After running LAFT, a k-connected local subgraph is obtained. The k-connected local subgraph is defined as $L_{N_u} = (V_{N_u}, E_{N_u}^k)$.

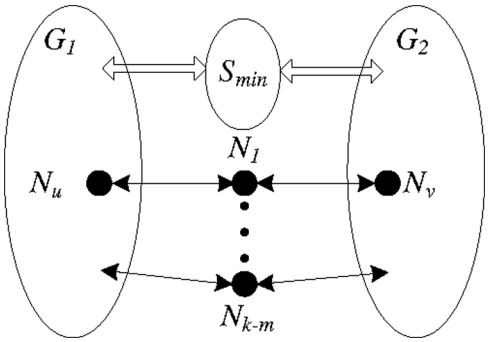
Theorem 5: *If the connectivity of G_{N_u} is m, then the connectivity of L_{N_u} is k when $m > k$ and is m when $m \leq k$.*

Proof: When LAFT runs in the local graph G_{N_u}, it becomes GAFT. So LAFT can surely construct a m-connected subgraph according to Theorem 1 when the connectivity $m \leq k$. LAFT can construct a k-connected topology when $m = k$. $m > k$ is a more sufficient condition. Therefore LAFT can surely construct a k-connected topology when the connectivity $m > k$. □

Theorem 6: *Let $G = (V, E)$ be a graph on V = {N_0, N_1, . . ., N_{n-1}} with n ≥ k + 1. L_k(V, E_k) is obtained by LAFT. If G(V, E) is k-connected, then L_k(V, E_k) is also k-connected.*

Proof: We proof the theorem by contradiction. Suppose the connectivity of $L_k(V, E_k)$ is m and $m < k$. There must be a minimal 'cut set' (West, 2001), named as $S_{\min}$, which cut the $L_k(V, E_k)$ into at least two disconnected parts. And the size of $S_{\min}$ (the number of nodes contained in $S_{\min}$) must equal to m. So the graph L_k breaks into at least two parts by removing all the nodes in $S_{\min}$ (see Figure 1). Without loss of generality, L_k breaks into only two parts: G_1 and G_2 shown in the figure. Since the origin graph G is k-connected, its minimal cut set must contain other $k - m$ nodes: $N_1, \ldots, N_{k-m}$. These nodes must connect G_1 with G_2 through other nodes in G. As shown in Figure 1, N_1 connects G_1 with G_2 through N_u and N_v in G. If the connectivity among N_1, N_u and N_v is no more than k, N_1 must connect with N_u and N_v after running LAFT according to Theorem 5. Therefore $S_{\min}$ must also contain N_1. It contradicts the assumption that $S_{\min}$ is a minimal cut set.

Figure 1 A demonstration that the topology is k-connected



If the connectivity among N_1, N_u and N_v is more than k, the links between N_1 and N_u, N_1 and N_v must be deleted after running LAFT. Otherwise $S_{\min}$ must contain N_1, which contradicts the assumption. When the links between N_1 and N_u, N_1 and N_v are deleted, there must another k disjointed pathes among N_1, N_u and N_v. Not all of these k pathes go through $S_{\min}$ because its size is less than k. So there must be at least one path, which does not go through $S_{\min}$ but go through other node, named as N_{k-m}. Here $S_{\min}$ must contain N_{k-m} additionally. It also contradicts the assumption that $S_{\min}$ is a minimal cut set.

If $S_{\min}$ separates $G(V, E)$ into more than two disjoint components, the above proof can be applied to every pair of components and then the k-connectivity of L_k can be obtained. □

Theorem 7: *$L_k(V, E_k)$ is bi-directionally connected.*

It is very similar to Theorem 2 because LAFT also runs TPA, which adds some edges for those unidirectional edges.

LAFT can also use the quick sort algorithm (QSA) to sort the neighbours before broadcasting the ***affirm*** message to adjust the power. And the average complexity of QSA is $O(n_e \log\ n_e)$. If we use BSA to check whether any node in S_{N_i} is also in S_{N_u}, then it takes at most $kO(\log(|S_{N_u}|+1))$ steps to check whether the former k nodes belong to S_{N_i}. If $S_{N_i} \subseteq V_k^{N_u}$, N_i has to find the $(k+1)^{th}$ node in the remainder $n_e - k$ nodes in $H_{N_i}^1$, which takes $O(\log(|S_{N_u}|+1))$ steps to check whether any node in the remainder $n_e - k$ nodes in $H_{N_i}^1$ is also in S_{N_u}. This kind of check would happen with the probability $p(|S_{N_u}|, k)$ according to the fact in Theorem 2. So we can figure out the steps that N_i finds the $(k+1)^{th}$ node is at most $O(\log(|S_{N_u}|+1)) \sum_{i=k+1}^{n_e} P(i,k) < (n_e - k)O(\log(|S_{N_u}|+1))$. After all, we have $O(n_e \log n_e) + kO(\log(|S_{N_u}|+1)) + (N_e - k)O(\log(|S_{N_u}|+1)) = O(n_e \log n_e)$ because $|S_{N_u}| \leq n_e$. It takes each node to finish the calculation of LAFT in at most $O(n_e \log\ n_e)$ steps.

As analysed in previous section, the time complexity of FGSS_k is $O(m^2\sqrt{n})$. So we can simply deem the complexity of FLSS_k is $O(m_e^2\sqrt{n_e})$, where m_e is the number of edges linking the n_e nodes. Notice that $m_e \geq kn_e/2$.

5 Simulation

Simulations for the performance evaluation of GAFT/LAFT are conducted with the Omnet++ simulation tool (Omnetpp, 2008). Comparison with other fault-tolerant protocols (such as CBTC (Li et al., 2001) and FGSS/FLSS (Li and Hou, 2004)) or without TC (NONE) will be given. In the following context, NONE indicates that no TC protocol is implemented and all nodes adjust their TP levels to the maximum.

The simulations consider the influence of not only medium access control (MAC) layer but network layer on the topologies established by GAFT/LAFT. Here MAC layer and network layer respectively implements the 802.11 protocol (MAC802.11, 1999) and the GEAR (geographical and energy aware routing) protocol (Yu et al., 2001). The receive sensitivity of the radio is at least –98 dBm. The radio can be adjusted for a range of output power levels from –20 dBm to 5 dBm in steps of 1 dBm. The maximal transmission radius is $r_{\max} = 261.195$ m. The detailed values of the relative parameters are provided in MICA2 Datasheet (2010) and MPR/MIB Node Hardware Users Manual (2010). All nodes are deployed in a $1000 \times 1000\ \text{m}^2$ area. Variable numbers of nodes from 50 or to 500 are deployed in the area in steps of 10 or 20. Each data point is the average of 50 simulation samples. For the purpose of analysis convenience, the variance, average or maximum of some data are given in some figures.

The energy consumption contains many aspects but the simulation mainly counts the transmitting and receiving power consumption. The power consumption by centre processing unit (CPU) and other hardware components is very different between different topology control protocols, for example FGSS/FLSS can consume huge energy and requires relative big memory space because they need call many times of the MFA (Even and Tarjan, 1975).

In the simulation, we set half of total nodes as source nodes and other half as destination nodes. A source node starts the routing and transmits packets to a destination node, which is randomly chosen by the source node.

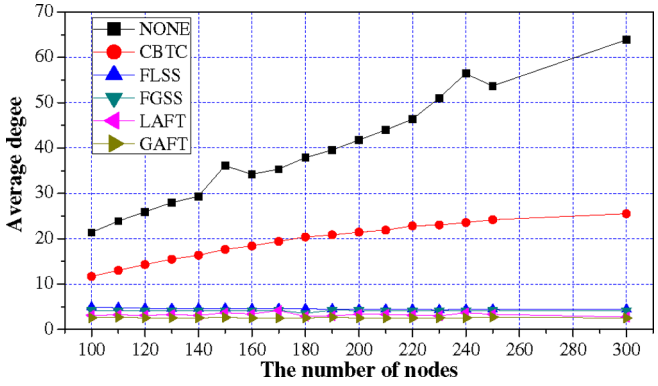
A major task in the simulations is to check the connectivity of the result topologies. Akyildiz (2007) describe the problem in detail. Esfahanian proposals an algorithm to calculate the vertex connectivity, which requires $[n - \delta - 1 + \frac{1}{2}k(2\delta - k - 3)]$ times calls of the MFA (Esfahanian and Hakimi, 1984). To check the connectivity is required by FGSS/FLSS but is only an additional task of the simulation programs to affirm whether the topologies constructed by GAFT/LAFT and $\text{CBTC}(\frac{2\pi}{3k})$ are k-connected.

5.1 Logical neighbours: node degree

Logical neighbours (degree) refer to those with whom one node has direct linkages. Although bigger degree means more accesses to other nodes with bigger transmission radius, it incurs more interference. The network capacity is the tradeoff between the link access and the interference. Therefore it is suitable to decrease the transmission power as low as possible while to keep a certain degree.

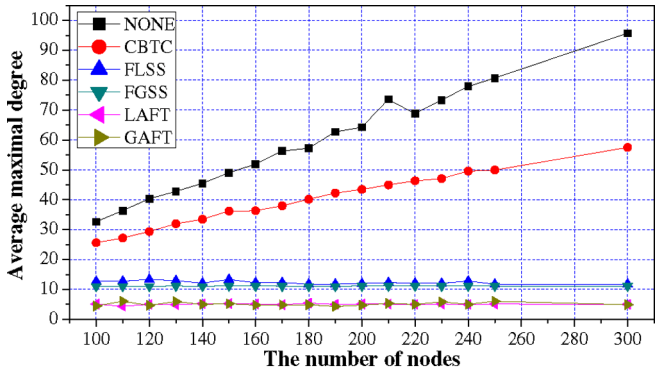
Figure 2 shows the average logical neighbours of the topologies constructed by $\text{CBTC}_k(\frac{\pi}{3})$, FLSS_2, FGSS_2, LAFT_2 and GAFT_2. When NONE is implemented, the average degree increases dramatically with the number of nodes. Those of NONE and $\text{CBTC}_k(\frac{\pi}{3})$ are much higher than those of both FGSS/FLSS and GAFT/LAFT and those of FGSS/FLSS are slightly higher than those of GAFT/LAFT. The average degree under $\text{CBTC}_k(\frac{\pi}{3})$ also increases with the number of nodes. The $\text{CBTC}_k(\frac{2\pi}{3k})$ has the coverage constraints on each individual cone, which makes each node create longer linkages in order to keep at least one neighbour in each cone. Because of the longer linkages, the number of neighbours increases more quickly with the node density increasing. Furthermore, each node has at least $2\pi/\frac{2\pi}{3k} = 3k$ neighbours to keep the whole network k-connectivity. But GAFT/LAFT and FGSS/FLSS only link with visible neighbours. So their average degrees are relatively low.

Figure 2 Comparison of NONE, $\text{CBTC}(\frac{\pi}{3})$, FGSS_2, FLSS_2, GAFT_2 and LAFT_2 with respect to average degree ($k = 2$) (see online version for colours)

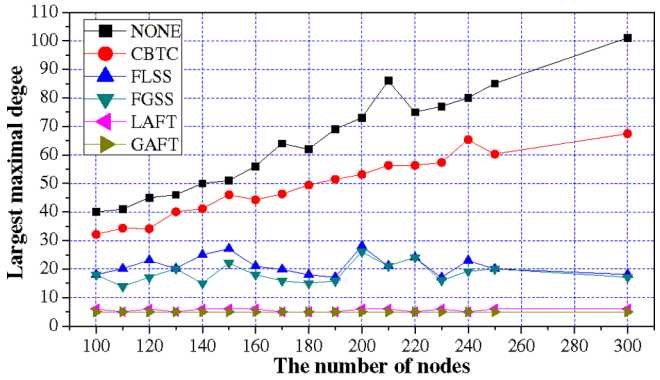


Since each data point is the average of 50 samples in Figure 2, there is a node with the maximal value (the number of logical or physical neighbours) in each sample, which contains n nodes. So we can figure out the average of the 50 maximal value and find the largest one from these values. In the following context, we call the average of the 50 maximal value as *average maximal degree* or *the average maximal number of physical neighbours* and call the largest maximal value as *largest maximal degree*. In Figure 3, the average maximal degree and the largest maximal degree are given out from the topologies derived under CBTC$_k(\frac{\pi}{3})$, FLSS$_2$, FGSS$_2$, GAFT$_2$ and LAFT$_2$. The values under GAFT$_2$/LAFT$_2$ are significantly smaller than those under NONE/CBTC$_k(\frac{\pi}{3})$ and slightly smaller than those under FGSS$_2$/FLSS$_2$. With increasing of the number of nodes, the values under GAFT$_2$/LAFT$_2$ almost change little and their performance improvement shows more remarkable.

Figure 3 Comparison of NONE, CBTC$(\frac{\pi}{3})$, FGSS$_2$, FLSS$_2$, GAFT$_2$ and LAFT$_2$ with respect to the maximal node degree ($k = 2$): (a) average maximal node degree and (b) largest maximal node degree (see online version for colours)



(a)

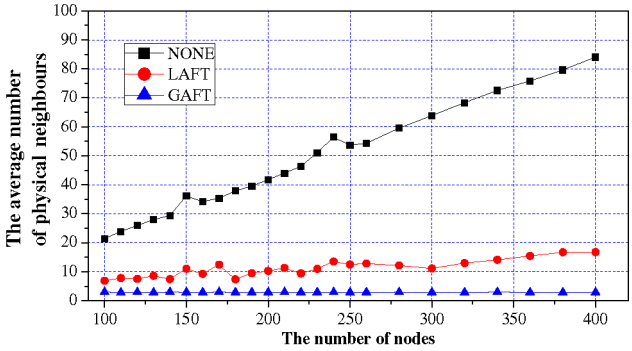


(b)

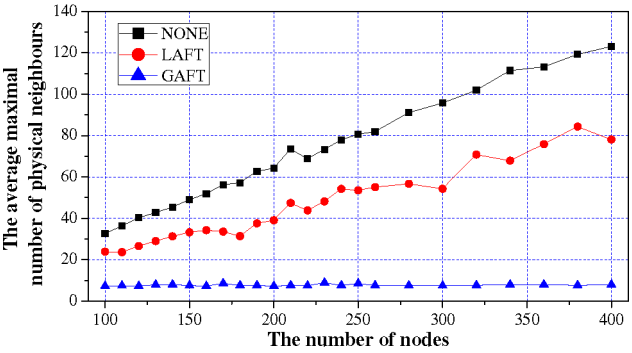
5.2 *The proportion of logical neighbours in physical ones*

In order to link with some neighbours, others is also included in the transmission range. So the inference occurs. All these neighbours are called physical neighbours. The links with logical neighbours are necessary to establish a topology, but interference on other neighbours should be as low as possible. Therefore the ratio between logical neighbours and physical ones is also an important parameters to indicate the interference, as shown in Figure 4. Most of the ratios of GAFT are close to 80% while most of the ratios of LAFT are close to 10%. These results are coincident with those in Figures 5–7. Topology under GAFT has better spatial reuse than that under LAFT.

Figure 4 Comparison of GAFT$_k$ and LAFT$_k$ with respect to the physical node degree ($k = 2$): (a) the average number of physical neighbours and (b) the average maximal number of physical neighbours (see online version for colours)



(a)



(b)

We give out the physical neighbours under 2, 3 and 4-connectivity by GAFT/LAFT in Figures 5–7. The average physical neighbours of LAFT under different connectivity are slightly bigger than those of GAFT. Because GAFT knows the global information but LAFT does not know the information out of two hops, some nodes may link some farer neighbours. Therefore the average maximal physical neighbours increase and the average physical degrees slightly increase as the number of nodes increases in Figures 5–7. Under different k values, the physical neighbours do not change much. GAFT/LAFT are robust under different connectivity.

5.3 *TP level*

A node can save more energy and create less interference when it broadcasts messages with lower TP level. The TP level can reflect the energy saving and the interference. Here we give out the average TP level of GAFT/LAFT under different number of nodes and compare the average radius of NONE, CBTC, FGSS/FLSS and GAFT/LAFT.

Figure 5 Comparison of GAFT_k and LAFT_k with respect to the physical node degree ($k = 3$): (a) the average number of physical neighbours and (b) the average maximal number of physical neighbours (see online version for colours)

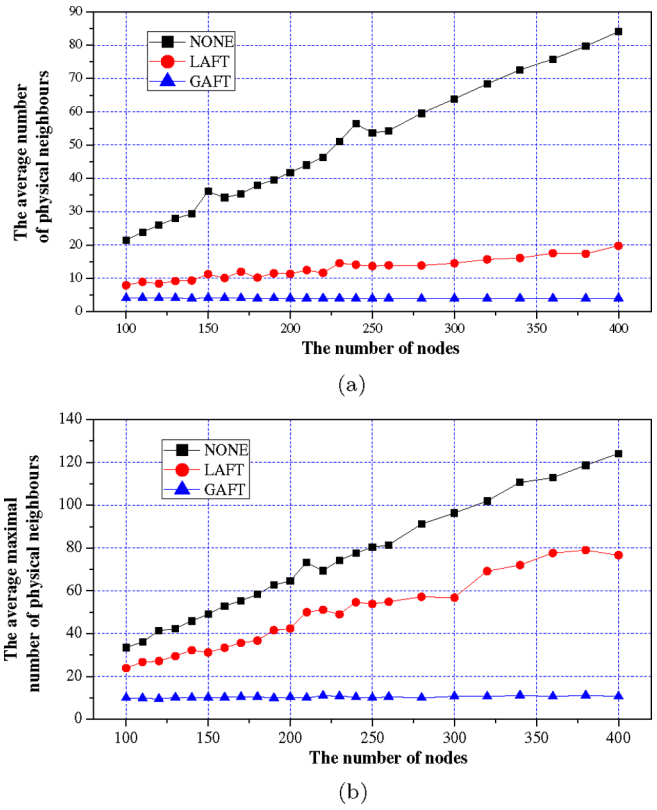
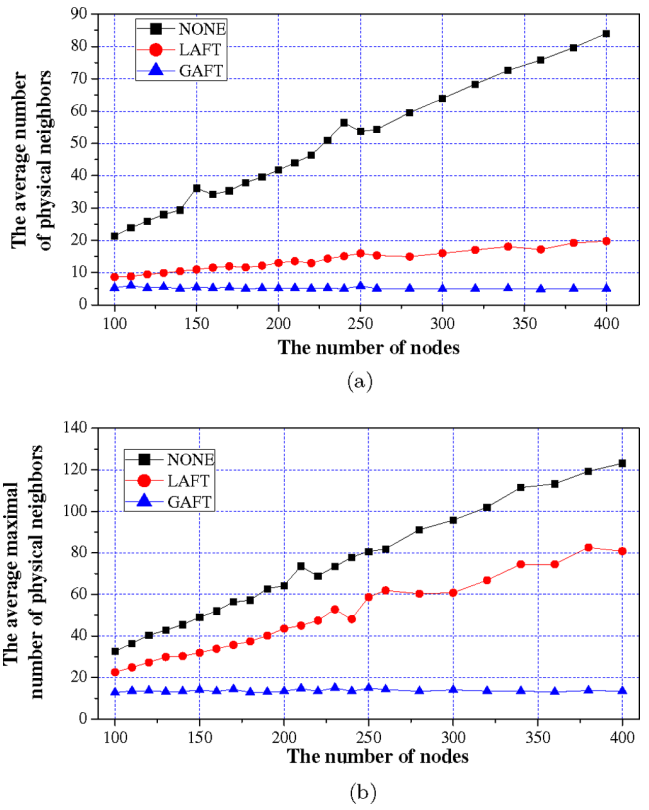
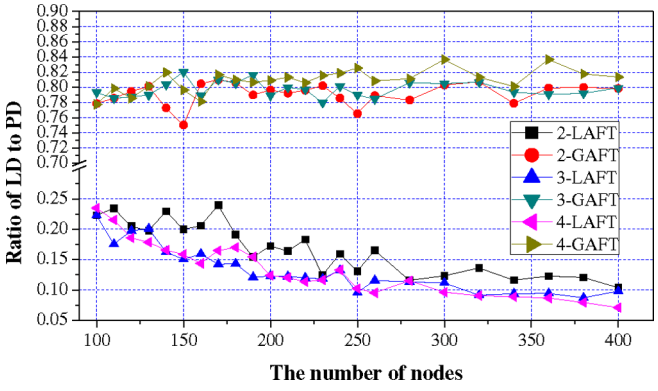


Figure 6 Comparison of GAFT_k and LAFT_k with respect to the physical node degree ($k = 4$): (a) the average number of physical neighbours and (b) the average maximal number of physical neighbours (see online version for colours)



If no considering the radio irregularity (Zhou et al., 2006), each node can calculates TP value according to equation (2) and adjusts its TP according to Algorithm 2. Figure 8(a) shows that average TP increases with the connectivity. When k becomes bigger, each node need to connect with more neighbours, which leads to a bigger transmission radius. The average TPs of GAFT are less than those counterparts of LAFT, because the physical neighbours of GAFT are less than those of LAFT. With the increasing of the number of nodes, average TPs decrease. The bigger the number of nodes, the higher the deployment density is. Then the same transmission range can cover more neighbours. In order to clearly show the results of our algorithms, we theoretically transform TP into radius distance and compare NONE, $\text{CBTC}_k(\frac{\pi}{3})$, FGSS_2, FLSS_2, GAFT_2 and LAFT_2 when $k = 2$. Notice that it doesn't mean that GAFT/FALT assume that the antenna pattern of a wireless device is a perfect disk although other algorithms do. In Figure 8(b), the radiuses of GAFT_2 are obviously lower than those of NONE, $\text{CBTC}_k(\frac{\pi}{3})$ and LAFT_2 and slightly lower than those of FGSS_2 and FLSS_2. The radiuses of LAFT_2 are obviously lower than those of NONE and $\text{CBTC}_k(\frac{\pi}{3})$ and slightly higher than those of FGSS_2 and FLSS_2. But it does not degrade the performance of LAFT since FGSS/FLSS should examine the network connectivity before each edge is admitted to the subgraph, which would lead to great calculation and memory space cost.

Figure 7 Comparison of GAFT and LAFT with respect to the ratio of LD to PD, where LD and PD refer to degree and the number of physical neighbours (see online version for colours)

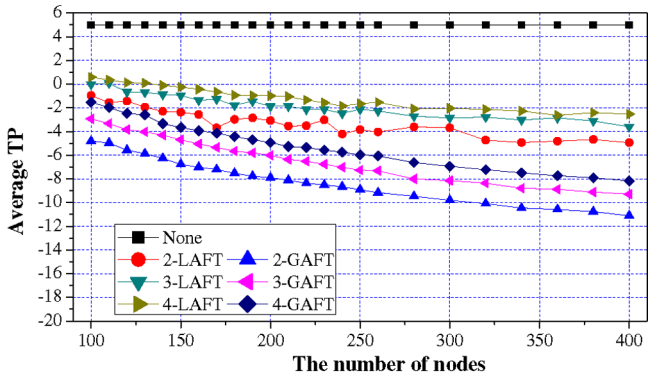


5.4 *Aware of routing and MAC*

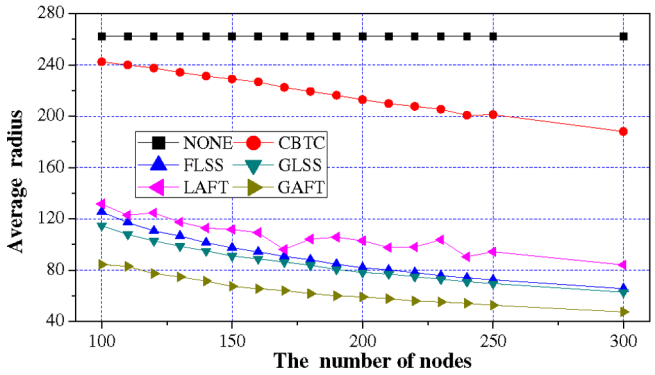
Finally we compare GAFT/LAFT_2, GAFT/LAFT_3 and GAFT/LAFT_4 with respect to the network capacity and the energy efficiency. In the simulation, the MAC protocol is IEEE 802.11 (MAC802.11, 1999) and the routing protocol is the geographical energy aware routing (GEAR) (Yu et al., 2001). In the whole network, half nodes are randomly chosen to be sources and the other half to be destinations. Each simulation lasts for 60 s. In the interval [0, 12 s], the topology and the routing are constructed. And in the interval [12 s, 60 s],

each node creates a packet per second. We consider several indexes: the total amount of data delivered, the total energy consumption and the energy efficiency.

Figure 8 Comparison of TP and radius: (a) comparison of GAFT and LAFT with respect to average transmission power and (b) comparison of NONE, CBTC$_k(\frac{\pi}{3})$, FGSS$_2$, FLSS$_2$, GAFT$_2$ and LAFT$_2$ with respect to the average radius ($k = 2$) (see online version for colours)



(a)



(b)

Data deliver rate (DDR): DDR is defined as the rate between the total number of the delivered data (bytes) to that of the transmitted data. It directly reflects the data rate. Some papers give only the total number of the delivered data, which can not fully illuminate the network capacity and interference. Because many packets are lost when the interference and MAC exist.

Figure 9 shows the DDR of GAFT when the connectivity is 1, 2, 3 and 4 respectively. When $k = 1$, DDR is very closer to but not exactly equivalent to 1. DDR decreases with the number of nodes and the connectivity increasing. The main reason is the medium access conflict. We use an example shown in Figure 14(a) to illustrate the effect of MAC. When N_4 transmits a packet to N_3 with a larger TP level, which leads to a larger transmission range r_1, no others can transmit. Suppose the size of the packet is l bytes and it costs N_4 t time to deliver. So the capacity can be simply indicated as $\frac{l}{t}$. When N_4 transmits the packet with a lower TP level, the range of which is r_3. At the same time, N_1 also can transmit another packet (suppose its size is also l) to N_2 with a small transmission range r_2. Then the capacity is $2\frac{l}{t}$. Therefore a larger transmission incurs more busy medium access. If some nodes can not have a chance to transmit its data for a threshold time, its data will be discarded according to the MAC802.11. Now we can understand the effect of MAC as shown in Figure 8. When the number of nodes increases, the node density increase accordingly. In other words, a same transmission range covers more nodes. When the connectivity increases, many nodes have to adjust their TP level higher. So they cover more neighbour nodes. Both cases incur more congested medium access conflict.

Figure 9 DDR of GAFT under different connectivity (see online version for colours)

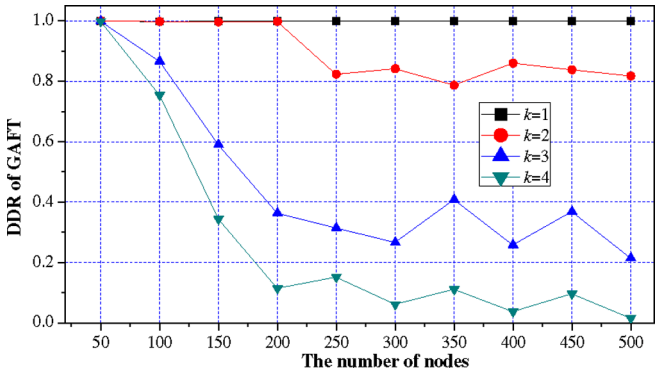


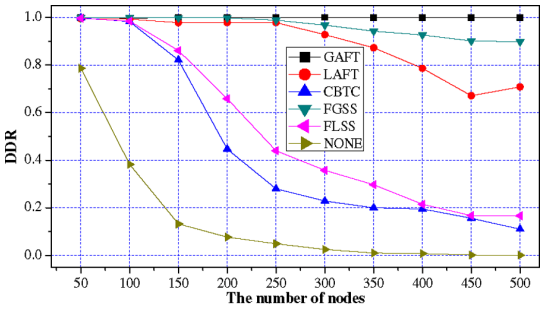
Figure 10 shows the comparison of GAFT/LAFT with CBTC($\frac{2\pi}{3k}$), FGSS/FLSS and NONE with regard to DDR under different connectivity. DDRs of all protocols decrease with the number of nodes and the connectivity increasing, which implies the effect of MAC. NONE has the lowest DDR. GAFT has higher DDR than FGSS and CBTC($\frac{2\pi}{3k}$) and LAFT also has higher DDR than FLSS. The figure illuminates that GAFT/LAFT not only significantly improve the network capacity but also decrease the interference especially when $k = 1$, 2 and 3.

Energy consumption: We estimate the energy consumption by two indexes: the average energy consumption per node and the average energy consumption per byte. In each simulation sample, there is a ratio of the total consumed energy to the total number of nodes. The average energy consumption per node is defined as the average of all the ratios since each data point is the average of 50 simulation samples as previous setting. The average energy consumption per byte is similarly defined.

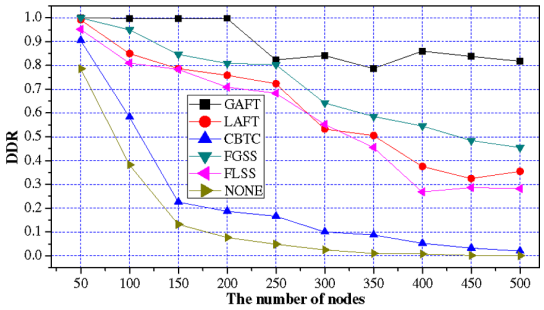
Figure 11 gives the average energy consumption per node and byte of GAFT under different connectivity. Both figures are very similar. Notice that the letters, p, μ, m and n, on the ordinates in the following figures denote the order of magnitude. When $k = 1$, the average energy consumption per node reaches the lowest value. And its corresponding variance is also very low as shown in Figure 12(a). In other words, all nodes consume their closely, which is expected. But the fault-tolerance is most bad when $k = 1$. With the connectivity increasing, the average energy consumption per node also increases fleetly. With higher connectivity, each node transmits packets by higher TP level even it transmits to a very near neighbour. In the case, the energy is wasted in a way. High

connectivity implies high fault-tolerance but it is at the cost of energy. Furthermore, higher connectivity incurs lower DDR as shown in Figures 9 and 10. In other words, some data must be retransmitted before they are discarded.But the data is not successfully transmitted ultimately. So the extra energy is cost on the redundant retransmission.

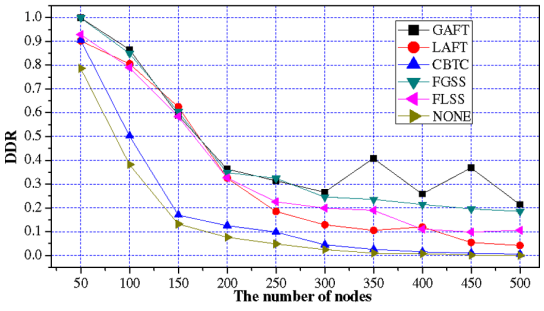
Figure 10 Comparison of GAFT/LAFT, CBTC($\frac{2\pi}{3k}$), FGSS/FLSS and NONE with regard to DDR: (a) $k = 1$; (b) $k = 2$; (c) $k = 3$ and $k = 4$ (see online version for colours)



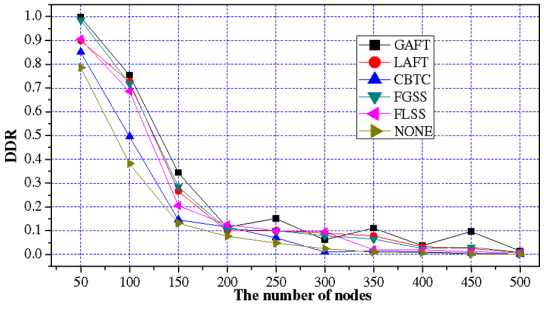
(a)



(b)



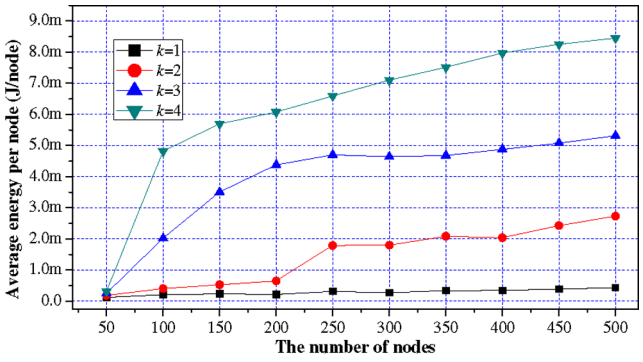
(c)



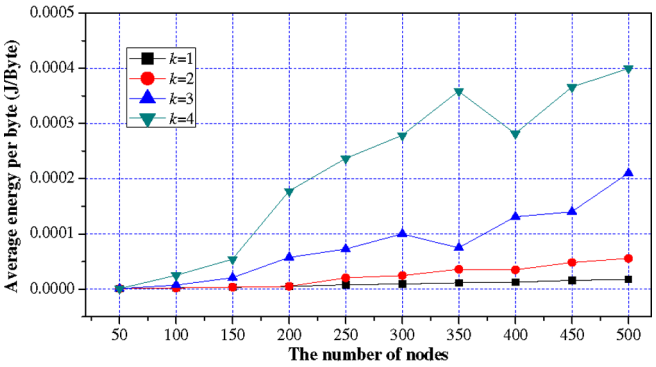
(d)

The variances of the average energy consumption per node and byte are shown in Figure 12. Under higher connectivity, the variance is also higher. But the variance begins decreasing when the number of nodes is 400 and the connectivity is 2, 3 or 4. In Figure 13, we give the comparison of GAFT/LAFT with NONE, CBTC($\frac{2\pi}{3k}$), FGSS/FLSS with regard to the energy consumption per byte under different connectivity.

Figure 11 Energy consumption under GAFT: (a) average energy consumption per node and (b) average energy consumption per byte (see online version for colours)



(a)

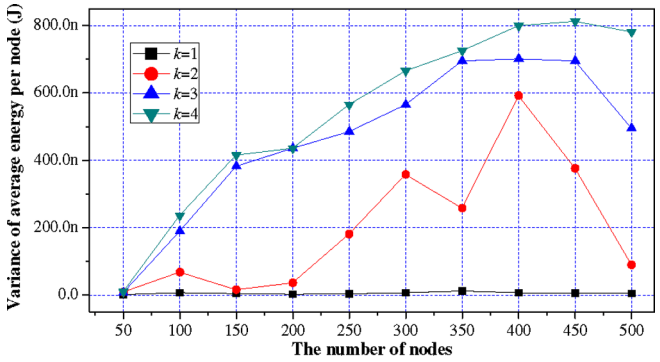


(b)

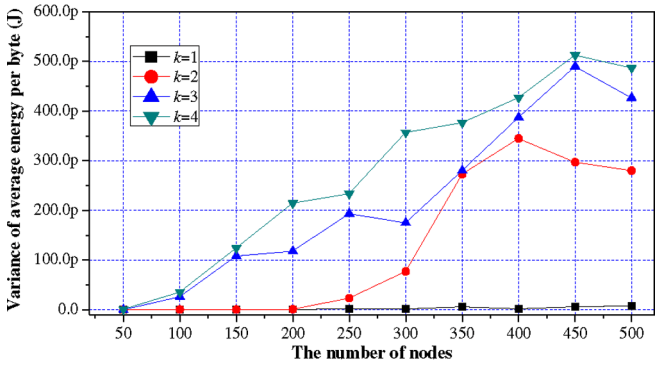
All above figures can not directly illustrate the influence of the routing. We use the example shown in Figure 14 to discuss the influence. The main purpose of a routing protocol is to find and maintain one or more paths from one or more sources to destinations while to consume energy and increase the network capacity as low as possible. When it establishes pathes, the routing protocol has their constrained conditions. For example, the EAR routing (Shah and Rabaey, 2002) requires that the next-hop node should be more close to the target than the previous-hop node when it establishes pathes from the source to the target. But the routing can not be feasible on the topology shown in Figure 14(a). In Figure 14(a), suppose that N_4 is the source node and N_1 is the target node. When N_4 tries to establish a path toward N_1 through some medium nodes, such as N_3, the process can not go on because the distance $d(N_1, N_3) > d(N_4, N_3)$. The case is called as 'routing avoid' by Yu et al. in the paper (Yu et al., 2001), in which Yu et al. proposed GEAR to deal with 'routing avoid'. But some additional calculation and communication are created. If we insert a edge (N_2, N_4) in Figure 14(a), N_4 can find a path through N_2 shown in Figure 14(b). A strictly constructed topology can keep the network connectivity and save energy, but it also deletes some optional linkages and affects the routing running on it. The effect between the topology and the

routing is mutual so the consideration on both sides should be put into the relative research.

Figure 12 The variance of energy consumption under GAFT: (a) the variance of average energy consumption per node and (b) the variance of average energy consumption per byte (see online version for colours)



(a)



(b)

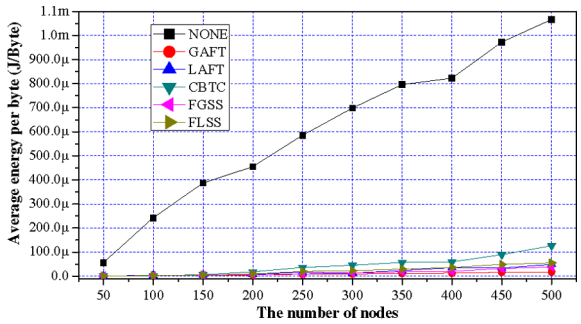
But TC is necessary because the great interference would create and the network capacity would decrease dramatically when NONE is implemented as shown in Figures 2 and 3 and so on. In other words, TC can influence MAC.

6 Conclusion and future work

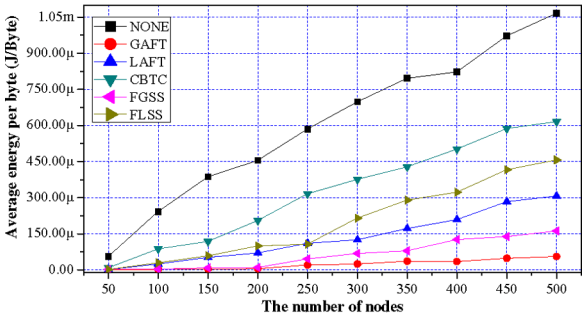
In this paper, a low weighted and energy efficient k-connected TC algorithm GAFT is proposed to preserve the network k-fault tolerance. Then a localised implementation LAFT of GAGT was presented to adapt to some self-organised wireless network. These two algorithms need simple calculation and result in k-connected topology.

We relaxed the assumption that the antenna pattern is a perfect disk and simplified the calculation of TC and the message transmission. In realistic scenarios, the transmission range is irregular and variable (Zhou et al., 2006) so UDG model is not feasible. GAFT/LAFT use the RD to determine the linkage relation among nodes. The real distance does not effect the two algorithms. GAFT/LAFT obtain performance over other algorithms on node degree and average transmission radius. In the simulation, we estimate the influence of the topologies on the energy consumption, the network capacity and the interference. we also estimate and analyse the influence of MAC, routing and topology on each other.

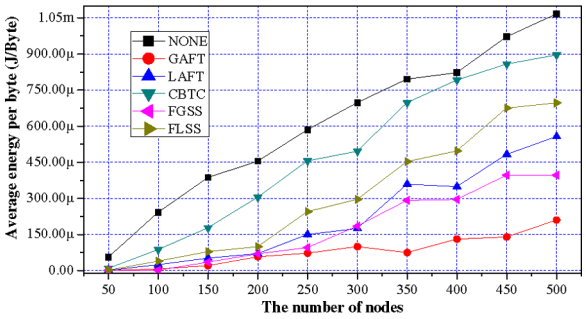
Figure 13 Comparison of NONE, GAFT/LAFT, CBTC($\frac{2\pi}{3k}$), FGSS/FLSS with regard to the energy consumption per byte: (a) $k = 1$; (b) $k = 2$; (c) $k = 3$ and (d) $k = 4$ (see online version for colours)



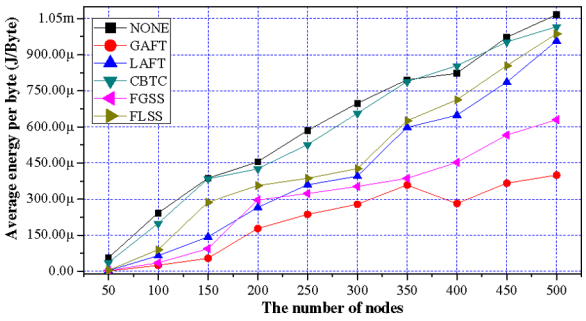
(a)



(b)



(c)

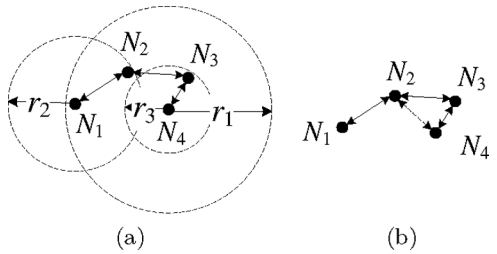


(d)

Some works are still needed to be studied. Although the point graph model is proved to be correct under the ideal case, the more realistic models should be developed to obtain the accurate quantitative information needed by the network designer. For example, the probability link model can be introduced. More realistic node distribution also should be investigated urgently. The uniform random deployment is

the most general one of many cases. In fact, nodes can not be artificially deployed under a strict uniform random distribution. Recently Yi et al. (2006) studied the connectivity probability of the random geometric graph with the Bernoulli node fault model since some nodes may become inactive. It is natural that some nodes come into the sleeping state or become unavailable due to internal breakdown. More realistic radio propagation model (Zhou et al., 2004) shall be considered when to design a protocol since the radio propagation can make great effect on a protocol stack in the sensor node. Furthermore, the effect between layers in the protocol stack should be deeply researched. For example, too long SIF in MAC802.11 causes unnecessary packet-loss and debases the network efficiency. And too short SIF can not prevent the network from transmission collision and results in more noisy interference, which also increase the probability of packet-loss. Therefore the cross-layer research is a urgent and complex issue (Gong et al., 2007).

Figure 14 A example shows the routing and the topology influence each other: (a) a given topology and (b) a modified topology (see online version for colours)



A most important problem is that any protocols in WSN can not be implemented synchronistically because the timer in all sensors are not synchronous and the delay time may be very long when packets are delivered from source nodes to destination nodes. For example, some nodes have begun a routing protocol while others begin a topology control protocol. The desynchronisation decreases the anticipant performance of the routing or even baffles the routing process.

Although theoretical analysis shows that a k-connected topology can be fault-tolerant. There is still problems how many the connectivity k should be and whether to construct a k-connected topology is the best method to make a network fault-tolerance. When $k = 4$ shown in Figure 10(d), DDRs of GAFT/LAFT are a little better than those of CBTC ($\frac{2\pi}{3k}$) and FGSS/FLSS but they are all very close to that of NONE. When no nodes or a few of nodes fail in a network, the cost to construct a k-connected topology may not be worth the decreasing of DDR especially when $k \geq 2$. The similar case happens in Figure 13(d). So the node failure pattern and its effect on the network should be researched or a new kind of fault-tolerant TC protocol should be developed.

Topology control is necessary to reduce the energy consumption and to improve the data deliver rate. Multi-connectivity TC protocols can benefit for the network fault tolerance. Efficient TC protocols are under demanded and the cross-layer effect should be considered to better the network performance.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 61190113 and 61003298), the program for Zhejiang Provincial Key Innovative Research Team on Sensor Networks(No. 2009R50046-6, No. 2009R50046-7, No. 2009R50046-4)

References

Akyildiz, A.H. (2007) *On the Evolution of Connectivity Algorithms, Invited, Submitted, Selected Topics in Graph Theory*, Edited by Robin Wilson and Lowell Beineke, Cambridge University Press, Cambridge.

Bahramgiri, M., Hajiaghayi, M. and Mirrokni, V.S. (2002) 'Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks', *Proceedings of IEEE Eleventh International Conference on Computer Communications and Networks (ICCCN)*, Vol. 8, pp.392–397.

Bettstetter, C. (2002) 'On the minimum node degree and connectivity of a wireless multihop network', *The Fifth ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MOBIHOC)*, Lausanne, Switzerland, Vol. 6, pp.80–91.

Esfahanian, A.H. and Hakimi, S.L. (1984) 'On computing the connectivities of graphs and digraphs', *Networks*, pp.355–366.

Even, S. and Tarjan, R.E. (1975) 'Network flow and testing graph connectivity', *SIAM Journal on Computing*, Vol. 4, No. 4, pp.507–518.

Gong, M.X., Midkiff, S.F. and Mao, S. (2007) 'A cross-layer approach to channel assignment', *Mobile Netw. Appl.*, Vol. 12, pp.43–56.

Hajiaghayi, M., Immorlica, N. and Mirrokni, V.S. (2003) 'Power optimization in fault-tolerant topology control algorithms for wireless multi-hop networks', *Proceedings of ACM 9th Annual International Conference on Mobile Computing and Networking (MobiCom)*, San Diego, CA, USA, pp.300–312.

He, T., Huang, C., Blum, B.M., Stankovic, J.A. and Abdelzaher, T. (2003) 'Range-free localization schemes for large scale sensor networks', *Proceedings of ACM 9th Annual International Conference on Mobile Computing and Networking (MobiCom), San Diego, CA*, pp.81–95.

Li, N. and Hou, J.C. (2004) 'FLSS: a fault-tolerant topology control algorithm for wireless networks', *Proceedings of the 10th ACM/IEEE Annual International Conference on Mobile Computing and Networking (MobiCom)*, New York, pp.275–286.

Li, L., Halpern, J.Y., Bahl, P., Wang, Y.M. and Wattenhofer, R. (2001) 'Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks', *Proceeding of ACM Symposium on Principles of Distributed Computing (PODC)*, Newport, RI, USA, pp.264–273.

Li, X.Y., Wan, P.J., Wang, Y. and Yi, C.W. (2003) 'Fault tolerant deployment and topology control in wireless networks', *Proc. Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp.117–128.

Omnetpp (2008) http://www.omnetpp.org/

MAC802.11 (1999) *Information Technology-Telecommunication and Information Exchange between Systems-Local and Metropolitan Area Networks-Specific Requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard, Technical Report.

Martins, F.V.C., Carrano, E.G., Wanner, E.F., Takahashi, R.H.C. and Mateus, G.R. (2011) 'A hybrid multiobjective evolutionary approach for improving the performance of wireless sensor networks', *IEEE Sensors Journal*, Vol. 11, No. 3, pp.545–554.

MICA2 Datasheet (2010) http://www.xbow.com/

MPR/MIB Node Hardware Users Manual (2010) http://www.xbow.com/

Nesargi, S. and Prakash, R. (2002) 'Configuration of hosts in a mobile ad hoc network', *Proceedings of IEEE INFOCOM*, New York, pp.1587–1596.

Penrose, M.D. (1999) 'On k-connectivity for a geometric random graph', *Communications of the ACM*, Vol. 15, No. 2, pp.145–164.

Ramanathan, R. and Hain, R.R. (2000) 'Topology control of multihop wireless networks using transmit power adjustment', *Proceedings of the IEEE Conference on Computer Communications, Tel Aviv, Israel*, Vol. 2, pp.404–413.

Rapport, T.S. (2001) *Wireless Communications: Principles and Practice*, Prentice-Hall, Upper Saddle River, NJ.

Santi, P. (2005) 'Topology control in wireless ad hoc and sensor networks', *ACM Comp Surveys*, Vol. 37, No. 2, pp.164–194.

Shah, R.C. and Rabaey, J.M. (2002) 'Energy aware routing for low energy ad hoc sensor networks', *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, Orlando, FL, pp.17–21.

Steele, M. (1992) 'Asymptotics for euclidian minimal spanning trees on random points', *Probability Theory and Related Fields*, Vol. 92, pp.247–258.

West, D.B. (2001) *Introduction to Graph Theory*, 2nd ed., Prentice-Hall Inc., Upper Saddle River, NJ.

Yi, C.W., Wan, P.J., Li, X.Y. and Frieder, O. (2006) 'Asymptotic distribution of the number of isolated nodes in wireless ad hoc network with bernoulli nodes', *IEEE Transaction on Communications*, Vol. 54, No. 3, pp.510–516.

Yu, Y., Govindan, R. and Estrin, D. (2001) *Geographical and Energy Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks*, UCLA Computer Science Department/CSD-TR-01-0023.

Zhang, J.H., Shen, X.F., Chen, J.M. and Sun, Y.X. (2007) 'Research on PID based transmission power control in wireless sensor network', *Chinese Journal of Sensors and Actuarots*, Vol. 20, No. 1, pp.177–182.

Zhou, G., He, T., Krishnamurthy, S. and Stankovic, J.A. (2004) 'Impact of radio irregularity on wireless sensor networks', *Proceedings of ACM International Conference on Mobile Systems, Applications, and Services (MobiSYS)*, pp.125–138.

Zhou, G., He, T., Krishnamurthy, S. and Stankovic, J.A. (2006) 'Models and solutions for radio irregularity in wireless sensor networks', *ACM Transactions on Sensor Networks (TOSN)*, Vol. 2, No. 2, pp.221–262.