

无线传感器网络(k, m)-容错连通支配集的分布式构建

马晨明¹ 王万良² 洪 榛³

(浙江工业大学信息工程学院 杭州 310023)¹ (浙江工业大学计算机科学与技术学院 杭州 310023)²

(浙江理工大学机械与自动控制学院 杭州 310018)³

摘 要 采用连通支配集作为虚拟骨干可以延长无线传感器网络的生命时间,但是考虑到节点容易失效,虚拟骨干还需要具有一定的容错性。对此,针对任意 k 和 m 取值,提出了一种完全分布式的 k -连通 m -支配集构建算法,其中 k -连通保证了网络中支配节点之间的容错性, m -支配则保证了普通节点与支配节点之间的容错性。该算法可以在异构网络中进行扩展,首先构建连通支配集,然后采用最大独立集和贪心的思想将普通节点进行 m -支配,最后在局部拓扑中通过公共邻居节点将连通支配集扩展为 k -连通。仿真实验证实,该算法可以通过较低的通信开销获得规模较优的 k -连通 m -支配集。

关键词 无线传感器网络,容错, k -连通 m -支配集,异构,分布式算法

中图分类号 TP393 文献标识码 A DOI 10.11896/j.issn.1002-137X.2016.1.029

Distributed Construction for (k, m)-Fault Tolerant Connected Dominating Set in Wireless Sensor Network

MA Chen-ming¹ WANG Wan-liang² HONG Zhen³

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)¹

(College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)²

(Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou 310018, China)³

Abstract Virtual backbone based on connected dominating set can prolong the lifetime of wireless sensor network. However, considering nodes are prone to failure, virtual backbone also needs to have a certain degree of fault tolerance. In this regard, a fully distributed algorithm was proposed for fault tolerance constructing k -connected m -dominated set for arbitrary k and m values. The algorithm can be extended in the heterogeneous network. It constructs connected dominating set firstly, then makes all common nodes m -dominating with the idea of maximum independent set and greedy, and finally extends the connected dominated set k -connectivity by the common neighbor nodes in the local topology. Simulation experiments confirm that the algorithm can obtain better size k -connected m -dominating set with low message overhead.

Keywords Wireless sensor network, Fault tolerance, k -connected m -dominated set, Heterogeneity, Distributed algorithm

1 引言

随着物联网的快速发展,无线传感器网络已经受到国内外学者的广泛关注。减少能耗是无线传感器网络研究的基础性问题^[1],由于网络没有类似蜂窝通信中的固定骨干,节点在通信中需要采用泛洪,这样不但会造成信息的冗余和碰撞,还加剧了网络能耗。采用连通支配集构建虚拟骨干使网络层次化,可以有效限制路由搜索空间,降低了通信拥塞和干扰发生的概率。然而,网络中可能会出现节点失效或链路错误等问题,如果这些失效位于虚拟骨干上,那么网络会因为中断而影响服务质量。因此,在节省能量的同时还必须考虑容错性,构建容错连通支配集,使部分节点失效时仍能保证网络正常运行,是无线传感器网络中的重要研究课题^[2]。

本文研究 k -连通 m -支配集的构建问题, k -连通保证了网络中的支配节点之间至少存在 k 条不相交路径, m -支配则保证了普通节点的容错性,我们将该问题记作 (k, m) -CDS。任意 k, m 取值的 (k, m) -CDS 已经被证明是 NP-Hard 问题。

2 相关工作

鉴于虚拟骨干的优越性,现有学者已经对最小连通支配集进行了广泛的研究。研究主要集中在单位圆盘图上构造 $(1, 1)$ -CDS,提出了基于数学建模^[3]、最大独立集^[4]、最小生成树^[5]、剪枝^[6]和贪心^[7]等算法。然而,随着网络应用变得越来越复杂, $(1, 1)$ -CDS 逐渐不能满足用户的需求,部分学者开

到稿日期:2015-03-30 返修日期:2015-05-12 本文受国家自然科学基金(61402414, 61304256, 61379123),“十二五”国家科技支撑计划(2012BAD10B01),浙江省自然科学基金(LQ14F020005, LQ13F030013),浙江省教育厅项目(Y201431815, Y201327006)资助。

马晨明(1983—),男,博士生,主要研究方向为无线传感器网络拓扑控制, E-mail: michael_zjut@163.com; 王万良(1957—),男,教授,博士生导师,主要研究方向为自动化、无线网络; 洪 榛(1983—),男,博士,主要研究方向为无线传感器网络拓扑控制、智能交通。

始研究 (k, m) -CDS 以提高网络的可靠性。

Dai 等^[8]较早提出了多连通多支配集在移动自主网络中的构造问题,分别提出了 3 种不同的算法。但是,他主要针对 $k=m$ 的特殊情况进行研究,同时算法还依赖于网络尺寸、节点的密度等参数,而这在实际应用中很难获得。

CDSA^[9]利用深度优先计算 $(1, 1)$ -CDS 中叶子块的个数,如果当前叶子块大于 1 则选择最优路径上的节点对叶子块进行合并,这样每次迭代后叶子块的数量将减少 1,最终所有支配节点组成的集合不存在割点,但是算法只能处理 $k=2$ 且 $m=1$ 的情况。

DSBA^[10]采用扩充的思想构造 $(2, m)$ -CDS。算法首先构建一个 m -支配集,然后再进行 2-连通。第一阶段,邻域内优先级最小的节点将邻居优先级最大的节点作为候选支配节点;第二阶段,将连通分支中所有节点当作一个 Block,消息在 Block 中进行共享,然后由 Block 选择可以最大化减小连通分支的节点。算法采用同步协商的方式选举支配节点,因而能够得到较小的支配集规模,但是通信的回合数和发送的消息数过多。

贪心算法 CGA^[11]由一个中心节点搜集全网信息,然后优先选择性能较好的节点构建 (k, m) -CDS,最后将冗余节点删除以最小化支配节点数。该算法能够得到较优的支配集规模,但是时间和消息复杂度较高,不适合密集部署的网络。

Kim 等提出了常数近似比率的 $(3, m)$ -CDS 算法。FT-CDS-CA^[12]先构造一个 $(2, 3)$ -CDS 的子集 Y ,然后从 Y 中得到所有非 3-连通的节点集 X ;接着,选择剩余节点将 X 中的所有节点都转换为 3-连通;最后,调整节点的邻居支配节点数,最多选择 $m-3$ 个普通节点成为支配节点。Wang 等在文献^[13]中进一步对该算法进行了完善,并给出了仿真实验。

Blackbone2^[14]将节点分为黑色和白色,白色节点执行构建算法,判断两跳中是否存在 m 个黑色节点支配当前节点,如果存在,继续检查网络是否 k -连通,如果连通则保持白色,否则改变为黑色并成为支配节点;黑色节点执行剪枝算法,首先判断当前节点的邻居白色节点是否都已经被 m -支配,如果满足,继续判断邻域黑色节点的子图是否 k -连通,如果连通则改变状态为白色。上述过程不断循环,直到所有节点颜色达到稳定,但是算法的构建时间较长。

k -CDS^[15]将节点的参数 Domi 和 Total 分别初始化为 k 和所有邻居节点 Domi 之和。算法通过消息交互获得邻居节点的 Total, Total 最大的节点成为支配节点并通知邻居节点;邻居节点收到消息后将 Domi 减 1,然后通知其他节点对 Total 更新;如果节点的 Domi 等于 0,表示它的邻域内已经存在 k 个支配节点,其将成为普通节点。反复执行上述过程,直到所有节点 Domi 都为 0,这样形成的节点集是 k -支配的。然后,利用最小生成树和块-割点图将 m -支配集进行 2-连通,但是算法产生的消息开销和支配集过大。

DDA^[16]迭代构建 $(1, m)$ -CDS,然后构建一个初始 k -连通分支,分支上的节点广播 KC 消息,邻居 Black 节点收到消息后发现如果至少存在 k 条可行路径,则广播一个 AC 消息通知其他节点;否则,发送 RC 请求 k -连通的路径。当白色节点收到 RC 时,对 k 条连通路径进行确认,如果存在则先回复一个 CS 消息,然后广播 KC 成为 Black 节点;否则,将 ID 添

加到请求列表中并向邻居节点转发 RC 请求。算法的时间和消息复杂度较高,在最坏情况下需要收集整个网络的拓扑消息。

现有研究中,块割点的集中式算法虽然能够生成规模较小的支配集,但是操作都需要由中心节点控制,当网络规模较大时如何有效获取节点信息仍有待于研究;大多分布式算法只能构建特殊 k 和 m 取值的连通支配集,而且算法的消息开销也过大。此外,以上算法均没有考虑节点的异构性和能量的影响。针对以上问题,本文在文献^[5]的基础上,首先优化了构建 $(1, 1)$ -CDS 的通信开销,然后提出了完全分布式的容错拓扑控制算法 DFTA。所提算法不但可以适用于异构网络,而且可以对任意 k 和 m 取值的容错网络进行分布式的构建,提高了扩展性。

3 网络模型与定义

3.1 网络模型

用图 $G(V, E)$ 表示该网络模型,其中 V 表示节点集合, E 是由 V 中节点组成的通信链路集合。给出如下假设:

- (1) N 个节点随机部署在 $M \times M$ 区域中,由 ID 进行唯一标识,节点的能量和通信半径可以异构;
- (2) Sink 节点位于中心 $(M/2, M/2)$,且计算、存储和能量均不受限制;
- (3) 节点可以通过接收信息的信号强度 RSSI 获得与邻居节点之间的距离 d ;
- (4) 假设节点的通信半径 $R \in [R_{\min}, R_{\max}]$,节点 v_i 和 v_j 之间存在通信链路当且仅当 v_i 和 v_j 相互处于对方的通信半径内。

3.2 相关定义

定义 1(连通支配集) 假设 DS 为节点集 V 的子集,对于任意非 DS 中的节点 $V \setminus DS$,如果在其邻域至少存在一个 DS 中的节点,同时保证 DS 中任意节点可达,则 DS 为连通支配集。

定义 2(不相交路径) 假设节点 u 和 v 之间的任意两条路径分别为 $p_1 = (u, w_{11}, \dots, w_{1m}, v)$ 和 $p_2 = (u, w_{21}, \dots, w_{2n}, v)$,如果链路 p_1 和 p_2 中的节点除了端点 u 和 v 外,其他中间节点均不相同,则称 p_1 和 p_2 是不相交路径。

定义 3(k 连通性) $G(V, E)$ 是 k 连通的当且仅当任意支配节点之间均存在 k 条不相交路径。这样,当 $k-1$ 个节点失效时,支配节点仍然是连通的。

定义 4(m 支配) 假设图 $G(V, E)$ 导出的连通支配集为子图 $G'(V', E')$, $V' \subseteq V, E' \subseteq E$ 。对于任意节点 $u \in V \setminus V'$,至少存在 m 个支配节点 $v_i \in V'$ 使 $(u, v_i) \in E', i=1, 2, \dots, m$,则 G' 是 m -支配的。这样,当任意 $m-1$ 个节点失效时,该节点仍然能够和支配节点进行通信。

定义 5(k -连通 m -支配问题) 对于任意给定的正整数 k 和 m ,求解 $G(V, E)$ 子图 $G'(V', E')$,使其不但是 k 连通的而且是 m -支配的,即为 k -连通 m -支配问题。

4 算法描述

4.1 连通支配集的构建

本节算法在文献^[5]中拓扑构建算法的基础上进行改进,

在节点通信半径存在异构的网络环境中进一步优化了消息开销。A3M^[5]在初始时采用了两轮消息交换以获得邻居节点的信息,这里只需要一次信息交换。算法的主要思想是通过比较节点之间的距离与通信半径,假设节点 u 收到了节点 v 发送的 Hello 消息,节点通过 RSSI 可以获得 u, v 之间的距离 $d(u, v)$,然后将其与通信半径 R_u 进行比较,如果 $R_u \geq d(u, v)$ 则确定节点之间可以通信,这样在不影响其他性能的基础上减少了网络的通信开销。

4.2 m -支配集的生成

本节将(1,1)-CDS 扩展成 m -支配集。下面对相关属性和机制给出描述:

- (1)黑色 1:4.1 节算法完成后的活跃节点颜色;
- 灰色 1:4.1 节算法完成后的普通节点颜色;
- 黑色 2: m -支配后的扩充活跃节点颜色;
- 灰色 2: m -支配后的普通节点颜色。
- (2) N_1 :节点的邻居节点集合。
- (3) N_g :节点的邻居灰色 1 节点集。
- (4) Dom :节点 N_1 中的黑色 1 和黑色 2 节点的数量。
- (5) E :节点的剩余能量。
- (6) R :节点的通信半径。

(7) w_1 :如果节点的邻居灰色 1 节点数越多,那么节点成为支配节点后即能覆盖更多的节点。同时还需要考虑该节点在所有灰色 1 邻居节点中的相对能量比重和通信半径,这是因为当前节点的能量越大,可以有效延长网络的生命时间,而节点的通信半径越大,可以减少产生的活跃节点数。使用式(1)对权值进行计算:

$$w_1(u) = \omega_1 * \frac{|N_g(u)|}{|N_1(u)|} + \omega_2 * \frac{E_u}{\sum_{i \in N_g(u)} E_i} + \omega_3 * \frac{R_u}{R_{\max}} \quad (1)$$

(8) T : m -支配集生成的总运行时间。由于该过程至多进行 $m-1$ 轮,总时间计算如式(2)所示,其中 T_1 表示每一轮中 m -支配时间。

$$T = (m-1) * T_1 \quad (2)$$

(9)时间同步机制:在每一轮中,各个灰色 1 节点将进行一次 m -支配,节点根据式(3)计算等待的超时。由于式(1)中的 3 个部分都在 $[0, 1]$ 之间,因此权值累加后的 w_1 同样处于 $[0, 1]$ 之间,这就确保了每一轮竞争在 T_1 内结束,从而可以忽略节点数量对算法总时间的影响。

$$timer1 = (1 - w_1) * T_1 \quad (3)$$

(10)Active 消息:邻域权值最大的灰色节点竞争成为 m -支配节点。

该部分的思想源于最大独立集和贪心算法的创新, m -支配构建可以由灰色 1 节点在上一过程结束后设置一个超时开始。如果当前灰色 1 节点的 $Dom \leq m$,需要等待与 w_1 成反比的时间 $timer1$ 竞争成为支配节点,权值越大,等待的时间越短。如果在 $timer1$ 内节点没有收到邻居节点发送的 Active 消息,当前节点就开始广播 Active 消息并成为黑色 2 节点;反之,如果当前节点在超时还没结束时收到了邻居节点广播的 Active 消息,就说明节点竞争失败,如果当前 Dom 还没有达到 m ,那么它将在下一轮中继续竞争。灰色 1 节点收到 Active 消息后将 Dom 加 1,并修改列表中对记录的状态,该过程反复运行,直到所有灰色 1 节点的 Dom 都等于 m 。

4.3 k -连通网络的扩展

本节将 m -支配集扩展成 k -连通。下面对过程中用到的

属性和机制给出描述。

(1)黑色:已经 k -连通的黑色 1 和黑色 2 节点。

(2) N_b :节点的邻居黑色节点集。

(3) w_2 :当前灰色 2 节点的 N_b 节点数越多,那么节点成为支配节点后即可减少添加的 k -连通节点,其他需要考虑的因素与 w_1 相同,具体如式(4)所示:

$$w_2(u) = \omega_1 * \frac{E_u}{\sum_{i \in N_b(u)} E_i} + \omega_2 * \frac{|N_b(u)|}{|N_1(u)|} + \omega_3 * \frac{R_u}{R_{\max}} \quad (4)$$

(4)Connect 消息:节点通过广播 Connect 消息开始 k -连通过程。消息中包括两个列表,其中 List1 表示邻居黑色节点开始 k -连通的顺序,而 List2 表示当前节点 k -连通所需要扩充的活跃节点集合。

(5)时间同步机制:黑色节点收到 Connect 消息后,会根据 ID 在 List1 中的位置 index 等待相应的超时开始 k -连通过程以保证最小化新增的活跃节点数。该等待的超时如式(5)所示:

$$timer2(i) = index(i) * T_2 \quad (5)$$

(6) T_2 :每个黑色节点需要等待的最长时间。这里主要包括 3 个方面,首先是本地构建 k -连通拓扑的计算时间,其次是单跳邻域最长的通信时间,此外还需要考虑一定的通信时延。具体计算如式(6)所示:

$$T_2 = T_{comp} + T_{comm} + T_{delay} \quad (6)$$

算法从 Sink 节点开始广播一个 Connect 消息,孩子黑色节点收到消息后等待 List1 中的超时,然后开始 k -连通过程。节点 u 以自己 and 每一个邻居黑色节点 v 作为局部拓扑,检查 $N_b(u)$ 中是否至少存在 $k-1$ 个节点同时也是 $N_b(v)$,如果存在则表示 u 与 v 之间 k -连通;否则,从 N_1 中选择状态为灰色 2 并且满足 $v \in N_b(u)$ 的权值 w_3 最大的节点 w 作为黑色节点加入到局部拓扑,直到该拓扑成为 k -连通,并将新扩充的节点作为 List2。然后,在 List2 中进行剪枝操作,删除冗余节点(即删除该节点后局部拓扑仍然为 k -连通),再将列表随 Connect 消息继续向下发送。黑色节点收到该消息后,首先根据 List2 更新 N_1 中节点的状态,如果发现自己 ID 在 List2 中,还需要变成黑色节点。

下面通过一个实例对 m -支配和 k -连通过程进行描述,这里构建(2,2)-CDS,初始拓扑如图 1(a)所示。此时,节点 D 因为同时处于节点 A, B 的通信半径内,已经是 2-支配的,将成为灰色 2 节点;节点 F, H 因为在邻域内存在较多灰色 1 节点而优先成为支配节点,那么节点在邻域内开始广播 Active 消息通知邻居节点当前节点已成为支配节点,如图 1(b)所示;最终,节点 C, E, I, G 发现已经是 2-支配的,得到如图 1(c)的(1,2)-CDS。

图 1(d)~图 1(f)显示了 k -连通过程,节点 A 首先构建本地 k -连通拓扑,添加新增节点 D 到 List2 中,然后将列表 List1 = {B, F} 和 List2 = {D} 封装到 Connect 消息并广播出去,如图 1(d)所示;邻居节点 D 收到消息后成为黑色节点,而节点 B 和 F 按照在 List1 中的顺序先后开始 k -连通。此时,节点 B 发现与 H 之间的不相交路径少于 2,于是在灰色 2 节点集中选择权值最大的节点 I 与 H 连通,如图 1(e)所示。最终,节点 F 和 H 在超时结束后发现已经没有需要继续 k -连通的节点,如图 1(f)所示。

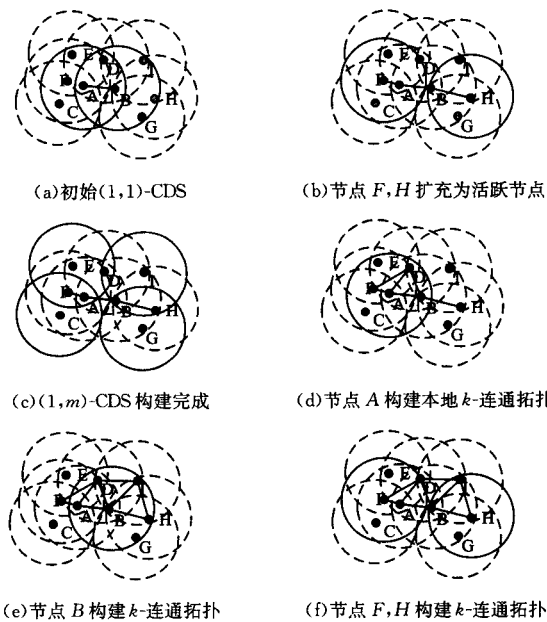


图1 m -支配和 k -连通过程的运行实例

5 仿真实验及分析

5.1 实验环境及参数设置

通过仿真实验对 DFTA 算法的有效性进行验证。实验采用基于网络事件驱动的 Atarraya^[17] 软件进行评估。在网络中随机生成拓扑,拓扑中节点的邻居节点个数需要至少多于 $\max\{k, m\}$, 否则丢弃该拓扑重新生成。为了更容易获得满足要求的网络拓扑,将传感器节点部署在 100×100 的二维平面中。实验结果取随机生成的 20 个拓扑分别运行 20 次后的平均值,具体参数如表 1 所列。

表 1 实验参数设置

参数	数值
区域(m^2)	100×100
部署方式	随机均匀部署
拓扑实例	20
节点数量	100~300
传输半径(m)	[15, 40]
初始能量(J)	[0.5, 1]
k, m 的取值	2, 4
权值比重	$\omega_1=1/3, \omega_2=1/3, \omega_3=1/3$

5.2 实验结果分析

实验 1 和实验 2 用来评估构建 $(1, 1)$ -CDS 的通信开销, 将 DFTA 与 A3M 进行比较。实验 1 将 100 个节点部署到网络中, 观察传输半径对发送消息数的影响。图 2 显示随着节点传输半径的增大, 两个算法所发送的消息数都在减少, 这是因为单个活跃节点可以覆盖的节点数逐渐增多, 当处于休眠的节点被唤醒时, 如果发现当前它的邻居节点都已经被覆盖, 那么就不再继续发送消息而直接进入休眠状态。同时, 我们还发现 DFTA 发送的消息数相比 A3M 在不同通信半径下都减少了 100, 这是因为 DFTA 节省的消息开销等于节点的数量。

实验 2 将传输半径设为 30, 通过改变节点数量观察发送消息数的变化情况。通过图 3 发现随着节点数量的增加, 需要发送的消息数量不断增加; 同时我们还发现两个算法的差距越来越大, 当节点数为 200 时 A3M 需要发送 624.0 个消息, 而 DFTA 只需要发送 424.0 个消息, 相当于节省了 32.1% 的通信开销, 显然 DFTA 的扩展性更好。

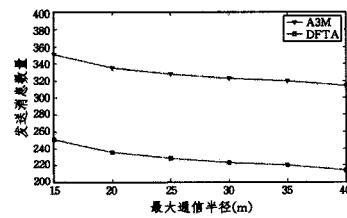


图 2 实验 1 传输半径变化时算法发送的消息数

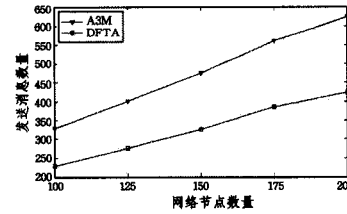


图 3 实验 2 节点数变化时算法发送的消息数

实验 3 和 4 主要对不同 k, m 取值时 DFTA 产生的活跃节点性能进行评估, 实验 3 将 200 个节点部署到网络, 观察传输半径对活跃节点数的影响。图 4 显示随着节点传输半径的增大, 所需要的支配节点数都在不断减少; 而且还注意到当通信半径变大时活跃节点下降开始变得平缓, 这是因为传输半径增加使得单位活跃节点可覆盖的节点数增多, 所以所需要的活跃节点数自然就少了。当传输半径为 20 时构建 $(3, 4)$ -CDS 需要 70.3 个活跃节点, 这一情况在半径为 25 时为 59.1, 相当于减少了 11.2 个活跃节点, 而这一数据从 35 变化到 40 时仅为 4.8。

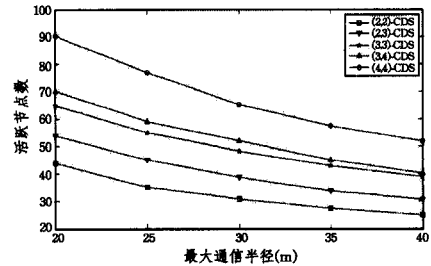


图 4 实验 3 传输半径变化时不同 (k, m) -CDS 的活跃节点数

实验 4 将传输半径设为 25, 通过改变节点数来观察活跃节点数的变化情况。图 5 显示随着网络节点数的增加, 所需要的活跃节点数逐渐增多。当节点数为 250 时, 构建 $(3, 3)$ -CDS 需要 58.8 个支配节点, 而构建 $(3, 4)$ -CDS 也只需要大约 61.7 个支配节点, 可见此时只需要添加少量节点就能增加 m -支配性; 当构建 $(4, 4)$ -CDS 时, 这一数据上升到了 78.6, 相比 $(3, 4)$ -CDS, 平均需要增加 16.9 个活跃节点, 也就是说为了维护更强的容错性, 算法的代价变大。同时, 还发现当节点数从 250 变化到 300 时, 支配集的规模出现振荡, 即当节点密度达到一定程度时所需要增加的支配节点数已经越来越少。

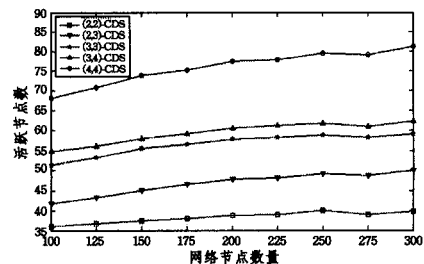


图 5 实验 4 节点数变化时的不同 (k, m) -CDS 的活跃节点数

实验 5 和实验 6 将传输半径设为 25, 比较不同算法产生的容错拓扑的性能, 这里选择两个较新的算法 DDA 和 k -CDS 作为参照。通过图 6 发现, 当网络比较稀疏时, DDA 的性能要优于相同规模下的其他算法, 比如在 125 个节点组成的网络中, 构建 $(2, 2)$ -CDS 时 DDA、DFTA、 k -CDS 3 个算法分别需要 39.0、40.2 和 62.8 个活跃节点; 而构建 $(4, 4)$ -CDS 时, DDA 和 DFTA 这一数据对应为 68.2 和 70.1, 这里没有 k -CDS 是因为 k -CDS 不能构建 4-连通网络。随着节点数量的增加, DFTA 与 DDA 的性能差距逐渐缩小。当在 275 个网络节点的环境下, DFTA 所需要的活跃节点数要少于 DDA, 算法此时只需要 80.8 个活跃节点, 而 DDA 需要 82.0 个活跃节点, 这说明 DFTA 算法更适合节点规模比较密集的网络环境。此外, 在构建 2-连通网络时, k -CDS 算法产生的活跃节点数一直处于增长中, 这说明算法直到节点数为 300 时仍然没有达到收敛。

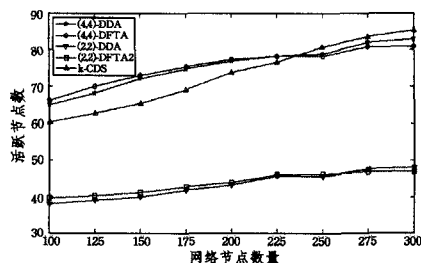


图 6 实验 5 不同算法的活跃节点数比较

实验 6 是在实验 5 的基础上对算法在构建过程中需要发送消息数的比较。图 7 显示在构建相同 (k, m) -CDS 时, DFTA 需要发送的消息数少于另外两个算法。当节点数量为 150 时, k -CDS 和 DDA 构建 $(2, 2)$ -CDS 分别需要发送 2036.8 和 3569.3 个消息, 而 DFTA 只需要 588.1 个消息, 相当于节省了 71.1% 和 83.5% 的通信开销, 这一情况在构建 $(4, 4)$ -CDS 时更加明显, 说明 DFTA 算法的扩展性较好。随着节点数量的增加, 各个算法之间的差距逐渐变大, 从曲线的变化趋势发现 DDA 发送的消息数随着节点数量的增加呈指数变化, 这将严重影响网络的生命时间。

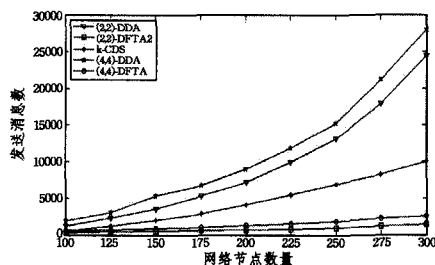


图 7 实验 6 中不同算法的发送消息数比较

结束语 考虑实际网络环境中节点可能出现故障而失效的问题, 本文在研究节点异构和能效的基础上, 针对任意 k 和 m 取值的应用需求, 提出了完全分布式的 (k, m) -容错连通支配集构建算法 DFTA。算法首先改进了 A3M 在构建 $(1, 1)$ -CDS 时通信开销过大的问题, 然后不依赖于全局的拓扑信息, 分布式地将 $(1, 1)$ -CDS 扩展为 m -支配和 k -连通, 从而减少了计算和通信开销。仿真实验结果证实, DFTA 比 A3M、 k -CDS 和 DDA 等算法更加有效, 特别是在节点比较密集的环境中。但是, 该算法产生的活跃节点数相比 $(1, 1)$ -CDS 仍然过多, 特别是当 k, m 取值较大时, 在保证网络容错的基础上继续减少活跃节点数将是下一步研究的重点工作。

- [1] Aziz A A, Sekercioglu Y A, Fitzpatrick P, et al. A Survey on Distributed Topology Control Techniques for Extending the Lifetime of Battery Powered Wireless Sensor Networks [J]. IEEE Communications Surveys & Tutorials, 2013, 15(1): 121-144
- [2] Lee S, Mohamed Y. Recovery from multiple simultaneous failures in wireless sensor networks using minimum Steiner tree [J]. Parallel and Distributed Computing, 2010, 70(5): 525-536
- [3] Hong Zhen, Yu Li, Zhang Gui-jun, et al. Topology Construction Based on Minimum Connected Dominating Set for Wireless Sensor Networks [J]. Journal of Electronics & Information Technology, 2012, 34(8): 2000-2006 (in Chinese)
洪榛, 俞立, 张贵军, 等. 基于最小连通支配集的无线传感网拓扑构建研究 [J]. 电子与信息学报, 2012, 34(8): 2000-2006
- [4] Kui Xiao-yan, Du Hua-kun, Liang Jun-bin. An energy-balanced connected dominating sets for data gathering in wireless sensor networks [J]. Acta Electronica Sinica, 2013, 41(8): 1521-1528 (in Chinese)
奎晓燕, 杜华坤, 梁俊斌. 无线传感器网络中一种能量均衡的基于连通支配集的数据收集算法 [J]. 电子学报, 2013, 41(8): 1521-1528
- [5] Ma Chen-ming, Wang Wan-liang, Hong Zhen. A Topology Control Method based on CDS Tree in Heterogeneous Wireless Sensor Network [J]. Chinese Journal of Sensors and Actuators, 2014, 27(6): 814-820 (in Chinese)
马晨明, 王万良, 洪榛. 异构无线传感器网络中基于 CDS 树的拓扑控制方法 [J]. 传感技术学报, 2014, 27(6): 814-820
- [6] Das A, Mandal C, Reade C, et al. An improved greedy construction of minimum connected dominating sets in wireless networks [C] // Proc. of IEEE Wireless Communications and Networking Conference, Cancun, Mexico, 2011: 790-795
- [7] He Jing, Ji Shou-ling, Fan Ping-zhi, et al. Constructing a load-balanced virtual backbone in wireless sensor networks [C] // Proc. of International Conference on Computing Networking and Communications, Maui, Hawaii, USA, 2012: 959-963
- [8] Dai Fei, Wu Jie. On Constructing k -Connected k -Dominating Set in Wireless Networks [C] // Proc. of the 19th IEEE International Parallel and Distributed Processing Symposium, Denver, Colorado, 2005: 81-90
- [9] Wang Feng, Thai M T, Du Ding-zhu. On the construction of 2-connected virtual backbone in wireless networks [J]. IEEE Transactions on Wireless Communications, 2009, 8(3): 1230-1237
- [10] Gao Xiao-feng, Xu Bo-sheng, Li Jun. A distributed design for minimum 2-Connected m -Dominating Set in bidirectional wireless ad-hoc networks [J]. Tsinghua Science and Technology, 2012, 17(5): 553-566
- [11] Wu Yi-wei, Wang Feng, Thai M T, et al. Constructing k -Connected m -Dominating Sets in Wireless Sensor Networks [C] // Proc. of Military Communications Conference, Orlando, Florida, USA, 2007: 29-31
- [12] Kim D, Wang Wei, Li Xian-yue, et al. A new constant factor approximation for computing 3-connected m -dominating sets in homogeneous wireless networks [C] // Proc. of the 29th IEEE International Conference on Computer Communications, San Diego, CA, USA, 2010: 1-9

表5 实验结果

网络	节点数	边数	可靠度	BDD 层数	MDD 层数	MDD 节点数
1	4	5	0.76008	9	4	8
2	4	6	0.80559	10	4	15
3	6	8	0.78082	14	6	56
4	6	15	0.80979	21	6	295
5	8	12	0.79270	20	8	339
6	10	15	0.78902	25	10	1521

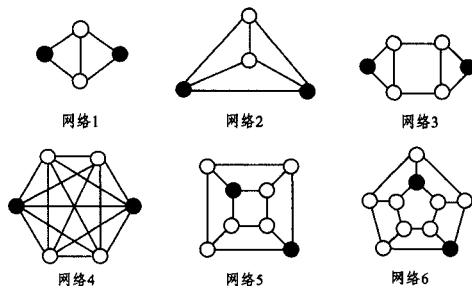


图9 实验基准网络

结束语 本文引入 MDD 计算节点和边不可靠网络的可靠度,构造了节点和边不可靠网络的形式化模型,给出了分析节点和边不可靠网络可靠度的 NEF_MDD 算法。由于每个 MDD 变量被用来描述单个节点和其未访问的邻接边,因此,NEF_MDD 算法构建的决策图层数等于网络节点数目。实例分析及实验结果表明,NEF_MDD 算法构建的 MDD 层数和节点规模小于 BDD 算法构建的决策图。

MDD 的构建过程中会生成同构子网,同构子网会增加算法的时间和空间开销。设计一个较优的同构子网处理方法是未来的研究方向。

参考文献

- [1] Ball M O. Complexity of network reliability computations[J]. Networks, 1980, 10(2): 153-165
- [2] Lin Y K. A simple algorithm for reliability evaluation of a stochastic-flow network with node failure[J]. Computers & Operations Research, 2001, 28(13): 1277-1285
- [3] Lin Y K, Huang C F. Assessing reliability within error rate and time constraint for a stochastic node-imperfect computer network[J]. Proceedings of the Institution of Mechanical Engineers Part O—Journal of Risk and Reliability, 2013, 227: 80-85
- [4] Yan Zong-shuai, Nie Chen-hua, Dong Rong-sheng, et al. A Novel OBDD-Based Reliability Evaluation Algorithm for Wireless Sensor Networks on the Multicast Model[J]. Mathematical Problems in Engineering, 2015, 2015: 1-14

- [5] Kuo S Y, Yeh F M, Lin H Y. Efficient and exact reliability evaluation for networks with imperfect vertices[J]. IEEE Trans. Reliability, 2007, 56(2): 288-300
- [6] Jiang Yi-nan, Li Rui-ying, Huang Ning, et al. Survey on network reliability evaluation methods[J]. Computer Science, 2012, 39(5): 9-13 (in Chinese)
江逸楠, 李瑞莹, 黄宁, 等. 网络可靠性评估方法综述[J]. 计算机科学, 2012, 39(5): 9-13
- [7] Theologou O R, Carlier J G. Factoring and reductions for networks with imperfect vertices [J]. IEEE Trans. Reliability, 1991, 40(2): 210-217
- [8] Sun Yan-rui, Cui Li-yan, Zhang De-xiang. A factoring algorithm for reliability evaluation of distributed networks with imperfect nodes[J]. Computer Science, 2002, 29(4): 111-113 (in Chinese)
孙艳蕊, 崔立彦, 张祥德. 计算具有不可靠结点分布式网络可靠度的一个因子分解算法[J]. 计算机科学, 2002, 29(4): 111-113
- [9] Xiao Yu-feng. Reliability analysis of two terminals network based on discrete probability model [D]. Beijing: Beijing University of Posts and Telecommunications, 2009 (in Chinese)
肖宇峰. 基于离散概率模型的二端网络可靠性分析[D]. 北京: 北京邮电大学, 2009
- [10] Yeh W C. A simple heuristic algorithm for generating all minimal paths[J]. IEEE Transactions on Reliability, 2007, 56(3): 488-494
- [11] Sun Yan-rui. Reliability evaluation of stochastic-flow network under both time and cost constraints[J]. Journal of Northeastern University (Natural Science), 2013, 11(34): 1537-1541 (in Chinese)
孙艳蕊. 带时间和成本约束的随机流网络可靠度的计算[J]. 东北大学学报(自然科学版), 2013, 11(34): 1537-1541
- [12] Nagayama S, Sasao T. On the optimization of heterogeneous MDDs[J]. IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 2005, 24(11): 1645-1659
- [13] Xing L, Dai Y. A new decision-diagram-based method for efficient analysis on multistate systems[J]. IEEE Transactions on Dependable and Secure Computing, 2009, 6(3): 161-174
- [14] Herrmann J U, Soh S, West G, et al. Using Multi-valued Decision Diagrams to Solve the Expected Hop Count Problem[C]// The IEEE 23rd International Conference on Advanced Information Networking and Applications (AINA 2009). Bradford, United Kingdom, 2009: 419-424
- [15] Bollig B, Wegener I. Improving the variable ordering of OBDDs is NP-complete[J]. IEEE Trans. Computers, 1996, 45: 993-1002

(上接第 132 页)

- [13] Wang Wei, Kim D, Kyung A M, et al. On Construction of Quality Fault-Tolerant Virtual Backbone in Wireless Networks[J]. IEEE/ACM Transactions on Networking, 2013, 21(5): 1499-1510
- [14] Schleich J, Danoy G, Bouvry P, et al. Backbone2, an efficient deterministic algorithm for creating 2-connected m-dominating set-based backbones in ad hoc networks[C]// Proc. of the 7th ACM International Symposium on Mobility Management and Wireless Access. New York, USA, 2009: 91-98
- [15] Zheng Chan, Yin Ling, Sun Shi-xin. Constructing 2-connected k-dominating sets for fault-tolerant backbone in wireless sensor

networks[J]. Control and Decision, 2013, 28(5): 650-656 (in Chinese)

郑婵, 尹令, 孙世新. 无线传感器网络中 2-连通 k-支配的容错连通支配集构造[J]. 控制与决策, 2013, 28(5): 650-656

- [16] Li Ying-shu, Wu Yi-wei, Ai Chun-yun, et al. On the construction of k-connected m-dominating sets in wireless networks[J]. Combinatorial Optimization, 2012, 23(1): 118-139
- [17] Wightman P M, Labrador M A. Atarraya: a simulation tool to teach and research topology control algorithms for wireless sensor networks[C]// Proc. of 2nd International Conference on Simulation Tools and Techniques. Rome, Italy, 2009: 26-35