

基于节点控制的空间信息网拓扑重构算法

刘 军¹, 于 耕², 张慧鹏¹

(1. 东北大学信息科学与工程学院, 辽宁沈阳 110819; 2. 沈阳航空航天大学, 辽宁沈阳 110136)

摘 要: 空间信息网是一种融合陆海空天信息系统的新型自组织网络, 成为研究热点. 针对网络特点和应用需求, 提出一种预防和恢复相结合的拓扑重构策略, 通过检测拓扑关键点触发预防性重构, 通过拓扑故障触发恢复性重构, 重构时在一定范围内选择冗余节点, 该节点在虚拟力的导向下自主地移动到待维护区域, 并以修复区域的局部拓扑通信代价最小为目标, 进行拓扑联合优化. 仿真结果表明, 所提出的重构策略能够优化网络性能, 提高网络的抗毁性.

关键词: 空间信息网; 拓扑重构; 节点控制; 虚拟力

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2011) 08-1837-08

A Topology Reconfiguration Algorithm Through Node Position Control for Space Information Networks

LIU Jun¹, YU Geng², ZHANG Hui-peng¹

(1. College of Information Science and Engineering, Northeastern University, Shenyang, Liaoning 110819, China;
2. Shenyang Aerospace University, Shenyang, Liaoning 110136, China)

Abstract: Space information networks, which have become a popular research focus, are a new type of Ad Hoc networks constituted by information systems of land, sea, air and space. This paper proposed a topology reconfiguration scheme combing protection and recovery strategies according to the features and application demands of space information networks. Detection of key nodes in topology and topology failures both trigger reconfiguration process. In reconfiguration process, selected redundant nodes move under the guidance of virtual forces to reconfigure topology. Redundant nodes jointly optimize topology next with targets of minimizing communication cost of designated repairing areas. Simulation results show the proposed topology reconfiguration scheme can improve network performances as well as network invulnerability.

Key words: space information networks; topology control; node position control; virtual force

1 引言

随着航天技术的快速发展, 天空地一体化的空间信息网将逐步形成^[1], 如图 1 所示, 该网络是由空间、临近空间、空中和地面节点, 按照资源的最大有效利用原则, 通过通信链路的互通互联, 形成的综合信息系统. 由于具有通信距离远, 覆盖面积大, 不受地理条件限制等诸多优点, 在情报收集、侦察监视、通信保障、导航定位等领域具有重要的应用价值.

空间信息网所处环境复杂多变, 通信链路和系统成员暴露于空间, 极易受到干扰、威胁和破坏^[2]; 且节点的移动性、无线通信的脆弱性以及设备按需的状态变迁等, 都势必造成网络拓扑结构不断发生改变.

空间信息网的上述特点使其网络性能随拓扑结构呈现动态变化, 甚至会由于节点移动或故障, 造成拓扑分割, 网络失效. 为了保证网络运行的稳定性、可靠性和抗毁性, 赋予网络极强的自我防护和修复能力, 应对网

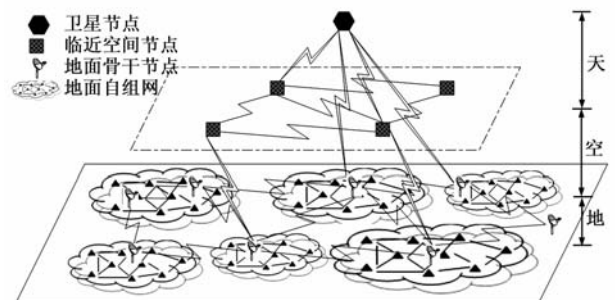


图1 空间信息网络模型

络拓扑的优化和重构技术深入研究,而空间信息网结构的局部不对称性、节点的可移动性以及大量冗余节点和链路的存在也为网络拓扑的优化、重构提供了条件.

2 相关研究

由于空间信息网的研究仅仅处在理论探索阶段,目前针对拓扑重构的相关研究成果极少,研究中借鉴现有自组网的研究成果.

网络重构可以分为预防式重构和恢复式重构.预防式重构的基本思想是预先规划分配网络资源,防止未来预期可能出现的网络故障,这种方法的优点是能保证通信的实时性,缺点是每次通信需要维护较多的信息,在切换到备用资源时涉及到较多技术问题,同时要以牺牲部分网络资源为代价,预防式重构方法比较适合对确定性的故障进行修复.恢复式重构的基本思想是网络发生故障后,为受故障影响的业务动态搜索可用备用资源,对受损业务实施恢复,恢复式重构方法的优点是对资源利用效率较高,但由于需要首先发现故障才能触发重构,因此时效性相对较低.

目前在自组织网络中对于网络层路由重构的研究较多.预防式路由重构^[3,4]在路由初始化时,为防止由于链路或节点的失效引起通信中断,预留多条物理链路或预设多条路由.恢复式路由重构常与路由协议的维护机制相结合,尤其对于 Ad Hoc 网络中应用最广的 AODV 和 DSR 路由协议已经提出了大量的路由故障修复策略^[5,6];此外还提出了许多基于选播策略的路由恢复方法^[7,8],当传输路由破裂后,将原路由上的节点按照断裂链路的位置划分成源节点组和目的节点组,利用选播协议的特点重新建立传输路径.

然而,现有对于拓扑重构的研究相对较少.拓扑连通性是路由有效性的前提,网络拓扑抗毁性是业务抗毁性的基础,当网络节点失效后,尽管网络可能仍然连通,但剩余的可用路径用于传输数据时的延迟可能较大,效率不高,影响网络的吞吐量,因此拓扑重构对于保证网络的可靠性而言更为关键.拓扑控制技术^[9,10]可以看作是一种预防式拓扑重构的手段,它是自组织网络中的一个基本问题,与网络的许多协议层密切相关,能够改善物理层的链路质量,提高 MAC 层的带宽使用和空间复用度,优化网络层的可选路由.

现有拓扑重构方法大多从链路控制的角度,通过调整节点发送功率的手段重新构建网络拓扑^[11,12].这种方法在故障节点和链路数目不多时比较有效,而当故障较多,特别是故障发生区域集中时,单纯使用调整功率的方式很难或无法修复拓扑的连通性,且随着故障增多,可能造成网络不同区域形成多个“拓扑空洞”.

无线传感器网络拓扑控制策略不能直接应用的主要原因是:首先,与传感器网络相比,空间网中的节点机动性更强,拓扑结构动态变化;其次,无线传感器网络中的拓扑控制目标集中在节省能量和提高网络覆盖性等方面,而空间网更关注抗毁性和业务的有效性;第三,空间网中的节点之间的距离很远,链路传输时延较长,拓扑控制时需要网络的数据转发机制进行优化以减小数据传输时延.

针对这种情况,本文从控制节点的角度,提出一种通过对节点位置的重新部署来修复网络拓扑的重构策略,作为控制链路重构方法的补充.

3 空间信息网拓扑重构算法

3.1 基本思想

空间信息网的节点虽然具有动态性,但网络节点的运动并不是完全杂乱无章的随机运动,例如卫星节点的运动具有周期性和可预知性,执行相同任务的编队节点运动具有群组特性,临近空间的系留式气球等节点仅在小区域内缓慢运动,由于这些节点间运动的上述特点,在组网过程中可以通过合理划分时间片等方法将他们区分出来,这些在一个时间片内拓扑相对稳定的节点组成的子网称为骨干网,在网络运行过程中发挥着更大的作用.而其他运动频繁,造成网络拓扑不断变化的节点,多数情况在网络中仅作为接入节点,例如在路由过程中通常不作为路径上的转发节点.

因此,骨干网拓扑对空间信息网性能起着至关重要的作用,预防和恢复相结合的重构策略针对于拓扑相对稳定的骨干网.随着网络运行中重构次数的增加,不但能够减少网络瓶颈,修复拓扑故障,恢复网络性能,增强网络的抗毁性和有效性,也有助于使骨干网拓扑结构趋于均衡,优化网络性能.

3.1.1 预防性拓扑重构

网络中存在着拓扑关键点,拓扑中仅由这样的节点连接两个或多个区域,一方面在通信中容易成为网络的瓶颈,造成拥塞,另一方面,一旦故障会造成网络分割,导致不同区域的节点无法通信,预防性重构就是进行拓扑关键点的检测,并采取措施避免这样的关键节点在拓扑中长期存在,主要步骤如下:

(1)节点分类:节点根据自身对拓扑结构状态变化的影响,判定自己的角色,包括普通节点、冗余节点和分点(拓扑关键点).可以用来重构拓扑而不影响网络性能的节点称为冗余节点;对拓扑连通性的破坏起关键作用的节点称为分点;其他节点为普通节点.

(2)拓扑修复:由分点发起拓扑修复过程,在限定的较近范围内选择合适且适量的冗余节点作为拓扑修复节点,修复节点在虚拟力导向下自主地向分点方向

移动.冗余节点位置的重新部署,一方面,修复拓扑故障,恢复网络性能;其次有助于消除拓扑关键点,降低网络分割概率;同时,能减少通信干扰,提高空间复用率,允许更多的节点同时通信.

(3)修复节点联合优化被修复区域拓扑:修复节点以最大化修复节点之间的连通性和最大化修复节点集合对分点邻域内节点的通信覆盖性为目标,联合对被修复区域的拓扑进行优化,以最小通信代价保证被修复区域拓扑的可靠性.

3.1.2 恢复性拓扑重构

当网络分割时,由发现故障的节点充当分点触发恢复性重构,重构步骤与预防性重构类似.当故障区域大时,需由发现故障的多个节点协同动作完成重构,并多次移动多个冗余节点.

重构方案模块框图如图 2 所示.

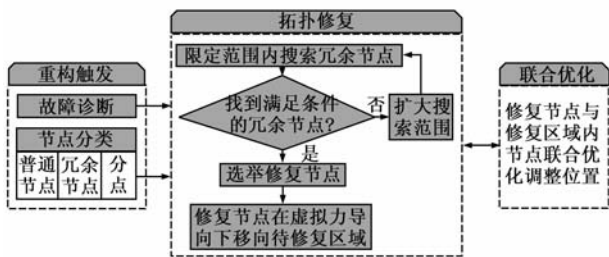


图2 拓扑重构方案

3.2 节点在拓扑中的分类

3.2.1 定义和假设

网络拓扑由图 $G(V, E)$ 表示,其中 $V = \{v_1, v_2, \dots\}$ 和 $E = \{e_1, e_2, \dots\}$ 分别为图 G 的顶点集合和边集合.如果网络中的两个节点可以直接通信,则它们之间存在一条边.假设图 g 中所有的边和顶点都在图 G 中,而且图 g 中每一条边和图 G 中相应的边具有相同的端点,则图 g 是图 G 的子图.

定义 1(节点邻域) 节点最大发送功率所覆盖的区域称为节点邻域,用 $R_{\max}(v_i) = G(N_{\max}(v_i), E_{\max}(v_i))$ 表示,其中 $N_{\max}(v_i)$ 为落入节点 v_i 邻域内的节点集合, $E_{\max}(v_i)$ 是节点 v_i 邻域拓扑的链路集合.

定义 2(邻接矩阵) 设拓扑图 $G(V, E)$ 中有 n 个节点,如果 $A = [x_{ij}]$ 是 $n \times n$ 阶对称二元矩阵, n 行和 n 列均表示 n 个节点,矩阵元素 $a_{ij} = \begin{cases} 1, & v_i \text{ 和 } v_j \text{ 邻接} \\ 0, & o. w. \end{cases}$, 则称 A 是邻接矩阵.

定义 3(连接度) 一个节点 v 是某一条链路 e 的端点,则称 v 和 e 互为关联;和顶点 v 关联的链路数称为 v 的连接度.

假设:节点在交换消息时能确知彼此相对于自身的方向,设节点 u 知道节点 v 处于 ρ 方向,则节点 v 知

道节点 u 处于 $(\rho + \pi)$ 方向.在没有位置信息的情况下,估算方向的技术是可行的,可以通过使用一个以上的方向天线实现,该技术作为 AOA(Angle-of-Arrival)问题,是正被 IEEE 天线和广播协会探讨的先进技术^[13].

3.2.2 节点分类

节点的角色分为冗余节点、分点和普通节点.冗余节点是指可以移动、不破坏拓扑连通性、移走后不会形成新的分点、且能量充足可参与拓扑修复工作的节点;分点是指其邻域内节点属于 $k(k \geq 2)$ 个不连通子集的节点,有且仅有分点连接上述不连通子集;其他节点属于普通节点.

节点角色的判定步骤如下:

(1)采集拓扑信息,计算邻域拓扑.

(a)广播 HELLO 消息.HELLO 消息中包含节点身份信息 and 发送该消息的功率信息,节点根据接收功率和发送功率可计算相互间的距离.

(b)接收 HELLO 消息.收到 HELLO 消息时,查看此 HELLO 消息的源节点信息,记录该节点身份信息以及通过天线感知到的方向信息,并根据路径损耗模型、发送功率和接收功率计算相对距离,距离作为链路的权值.如果已收到信息相同的 HELLO 消息,转到第(4)步.

(c)计算邻域拓扑.以图 3(a)中的三个节点 v_9, v_{11}, v_{15} 为例,通过 HELLO 消息的交互,节点 v_9 可知链路 $e_{9,11}, e_{9,15}$ 的长度和这两条链路的夹角.根据三角形正弦定理、余弦定理可计算出链路 $e_{9,11}$ 和 $e_{11,15}$ 之间夹角、链路 $e_{11,15}$ 和 $e_{9,15}$ 之间夹角以及链路 $e_{11,15}$ 的长度,进而判断出节点 v_{11} 和节点 v_{15} 是否连通.因此,节点 v_i 通过与邻域内节点交互 HELLO 信息,可以计算得到其邻域拓扑 $R_{\max}(v_i)$.

(d)丢弃 HELLO 消息,释放缓存空间.

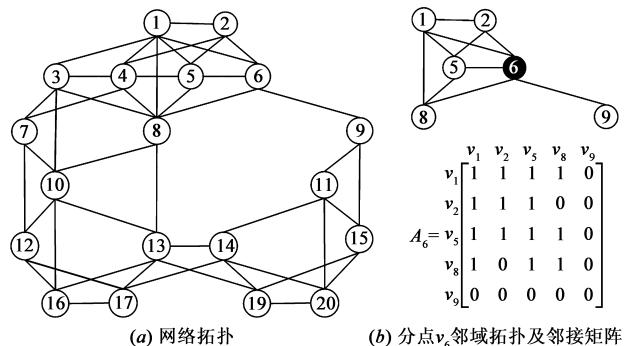


图3 网络拓扑及其邻接矩阵

(2)计算邻节点之间的连通性,判定分点.

邻域拓扑连通是保证节点移走后网络连通的充分条件.节点 v 根据计算得到的邻域拓扑 $R_{\max}(v)$ 建立邻域内邻接矩阵 A ,从而可以了解邻居节点之间在邻域拓扑内的可达关系.设节点 v 邻域内的节点集合 N_{\max}

$(v) = \{u_1, u_2, \dots, u_n \mid d(u_i, v) < \text{radius}(v), i = 1, 2, \dots, n\}$, 建立集合 $X(v) = \{u_1\}$, 将 $a_{ij} = 1$ 对应的节点 u_j 加入集合 $X(v)$, 对于所有属于集合 $X(v)$ 的节点 u_k , 将所有 $a_{kj} = 1$ 对应的节点 u_j 加入集合 $X(v)$. 如果此时 $X(v) = N_{\max}(v)$, 则说明节点 v 的邻域拓扑连通, 节点 v 可以继续冗余节点的判定; 否则节点 v 标记为“分点”。

如图 3(a) 所示的网络拓扑中, 6 号节点的邻域拓扑为 $R_{\max}(v_6) = G(N_{\max}(v_6), E_{\max}(v_6))$, 如图 3(b) 所示, 其邻接矩阵由图 3(b) 中的矩阵 A_6 表示. 邻域内节点集合为 $N_{\max}(v_6) = \{v_1, v_2, v_5, v_8, v_9\}$, 其中节点 v_1, v_2, v_5, v_8 属于同一个连通的节点集合, 9 号节点为孤立节点, 即 6 号节点邻域内节点分属于两个不连通的子集合, 因此节点 v_6 判定自己为“分点”. 6 号节点故障后引起网络分割的概率较大, 对网络性能会产生较大的影响, 因此是拓扑中的关键节点。

(3) 判定冗余节点.

冗余节点的判定需要两跳邻域拓扑信息, 进入冗余节点判定阶段的节点, 如果连接度大于 2 则可判定为“冗余节点”。

(4) 既没有被判定为分点也没有被判定为冗余节点的节点为普通节点。

3.3 拓扑重构

3.3.1 修复节点的选择

修复节点从冗余节点中选取。

(1) 广播 HELP 消息请求进行拓扑重构。

HELP 消息包括分点的 ID 号、分点的位置信息以及搜索半径 R_{\max} . 只有在以分点为圆心 R_{\max} 为半径的空间内的节点能够收到 HELP 消息. 其中 $R_{\max} = k \cdot T_{\max} \cdot v$, k 为调整系数, T_{\max} 为分点能够继续正常工作的时间, v 是网络中可移动节点的速度平均值, R_{\max} 的设定是为了通过消息广播范围的限定提高算法的执行效率并降低开销。

(2) 普通节点处理 HELP 消息。

普通节点 m 接收到 HELP 消息时首先比较自身与分点 c 的距离 $\text{dist}(m, c)$ 与 R_{\max} . 如果 $\text{dist}(m, c) > R_{\max}$, 则丢弃 HELP 消息; 如果 $\text{dist}(m, c) < R_{\max}$, 查看是否收到过相同的 HELP 消息, 是则丢弃, 否则转发该 HELP 消息。

(3) 冗余节点处理 HELP 消息。

不在分点邻域内的冗余节点 r 收到 HELP 消息, 在保证当前自身资源满足预设条件(根据具体应用而设定, 如能量充足等), 同时确保移入分点邻域后不会使拓扑中产生新的分点, 则:

(a) 停止随机运动(目的是防止冗余节点脱离分点

的广播范围)。

(b) 如果 $\text{dist}(r, c) > R_{\max}$, 则丢弃 HELP 消息; 如果 $\text{dist}(r, c) < R_{\max}$, 查看是否收到过相同的 HELP 消息, 是则丢弃, 否则转发该 HELP 消息。

(c) 设定计时器, 建立 HELP 消息列表. 计时器到期, 从列表中选择一个 HELP 消息回复 ACCEPT 消息进行应答, ACCEPT 消息包括冗余节点 ID 号, 冗余节点的位置信息以及分点的 ID 号。

选择的原则是: 根据自己到各个分点的距离和自己的移动速度, 选择自己能够以最短时间到达的分点. 移动到目的地的时间最短, 即该冗余节点完成拓扑重构的时间最短, 重构时间由式(1)表示:

$$T_{\text{reconfigure}} = T_{\text{comput}} \cdot MT + \frac{1}{v} \cdot \text{Dist}_{\text{mobile}} \quad (1)$$

其中, T_{comput} 表示冗余节点每次计算移动方向的时间, MT 表示冗余节点移动中改变方向的次数, \bar{v} 表示节点的移动速度, $\text{Dist}_{\text{mobile}}$ 表示节点总移动路程, $\text{Dist}_{\text{mobile}} = MT \cdot \Delta \bar{w}$, $\Delta \bar{w}$ 表示节点移动步长。

(4) 分点处理 ACCEPT 消息。

分点设定计时器, 建立 ACCEPT 消息列表, 从列表中选择若干冗余节点发送 SELECT 消息进行确认, SELECT 消息包括冗余节点 ID 号和分点的 ID 号. 分点以修复时间、自身能力及冗余度为参数选出修复时间最短的冗余节点发送 SELECT 消息。

如果分点在计时器时间之内都没有接收到任何冗余节点的 ACCEPT 消息, 或者发送 ACCEPT 消息的冗余节点均不满足称为修复节点的条件, 则增大 R_{\max} 的值(即扩大搜索范围), 重新广播 HELP 消息。

(5) 冗余节点处理 SELECT 消息。

冗余节点发送 ACCEPT 消息后同时启动等待 SELECT 消息计时器, 计时器到期前收到 SELECT 消息, 则确定为该分点的修复节点并启动移动过程. 如果在计时器超时后仍然没有接收到该分点的 SELECT 消息, 则在 HELP 消息列表中选择下一个符合条件的分点, 发送 ACCEPT 消息, 并触发等待 SELECT 消息计时器。

重构过程如图 4 所示, 节点 v_4 和 v_{11} 被选为修复节点, 图 4(b) 是修复过程完成后新生成的网络拓扑, 修复

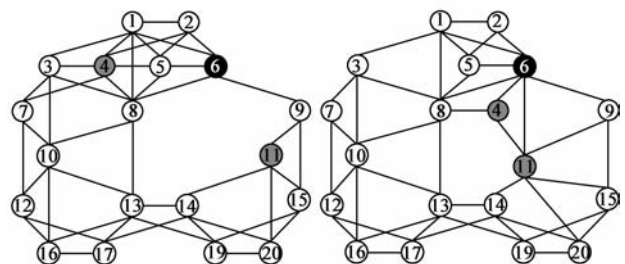


图 4 拓扑重构示意图

后网络拓扑分布更加均衡,避免了容易使拓扑分割的关键节点。

3.3.2 虚拟力导向的修复节点移动

虚拟力最初的研究目的是使机器人能够较好地规避障碍物,后来被用于传感器网络的布局优化^[14,15]。当修复节点接收到 SELECT 消息后,触发移动过程,具体内容如下:

(1) 修复节点初始化节点模型。

修复节点模型用一个七元组表示: $\langle P_i, R_i, \varphi_1, \varphi_2, k_1, k_2, \text{LastDirection} \rangle$ 。其中 P_i 表示修复节点 i 的坐标; R_i 表示修复节点 i 的通信半径; φ_1 表示修复节点 i 的邻居节点集合; φ_2 表示修复节点通信范围内障碍物的集合; k_1 表示修复节点因合力为 0 停止移动的次数,如果 $k_1 = 3$ (修复节点连续三次合力为 0),则要增加一次随机扰动; k_2 表示修复节点因合力方向与 LastDirection(上一次运动方向)相反时停止移动的次数,如果 $k_2 = 3$ (候选修复节点连续三次的合力方向与上一次移动方向相反),则要增加一次随机扰动; LastDirection 记录修复节点上一次的移动方向,由于节点有可能会在局部区域做往复跳动,算法中通过比较节点前一次的移动方向与本次移动方向是否相反,来决定本次节点是否继续移动,若方向相反,本次节点不移动。

(2) 计算虚拟力。

修复节点根据本地信息分布式计算自身所受到的虚拟力。设节点受到四种虚拟作用力:邻居节点斥力、障碍物斥力、分点引力和随机扰动作用力,分别由式(2)~(5)表示。

$$F_r(i, j) = \begin{cases} \frac{\alpha_r m_i m_j}{d(i, j)^{\alpha_r}}, & 0 < d(i, j) < R \\ 0, & d(i, j) \geq R \end{cases} \quad (2)$$

$$F_o(i, j) = \begin{cases} \frac{\alpha_o m_i m_j}{d(i, j)^{\alpha_o}}, & 0 < d(i, j) \leq R \\ 0, & d(i, j) > R \end{cases} \quad (3)$$

$$F_a(i, j) = -\frac{\alpha_a m_i m_j}{d(i, c)^{\alpha_a}} \quad (4)$$

$$\begin{cases} |F_{wander}| = \text{rand}(0, C) \\ \theta_F = \text{rand}[0, 360) \end{cases} \quad (5)$$

其中 $\alpha_r, \alpha_o, \alpha_a$ 是增益系数, m_i 和 m_j 表示节点质量,均设为 1; $d(i, j)$, $d(i, c)$ 分别表示修复节点与邻居节点、障碍物指示节点和分点的距离, R 为修复节点的通信半径。随机扰动作用力的作用是,一旦修复节点陷入运动陷阱,即修复节点在某个局部位置做往复跳动或静止不动而不能

继续向目标位置移动,则在有限的循环次数内,随机产生漫游步长和方向, $\text{rand}()$ 是随机函数, C 是随机扰动幅度, θ_F 是随机方向。

(3) 在虚拟力导向下的移动。

每一步长移动初始,修复节点更新当前分点、邻节点和障碍物指示节点的位置坐标,以适应网络拓扑的动态变化。首先计算邻居节点斥力、障碍物斥力和分点引力的合力,如果合力为 0,修复节点不能移动,需要增加一次随机扰动,使修复节点摆脱局部运动陷阱。如果修复节点的合力不为 0,要首先判断本次的合力方向是否与上一次的移动方向相反,如果不相反,则移动一个单位步长;如果相反,候选修复节点将会运动到上一个位置点,候选修复节点有可能陷入到往复跳动的局部运动陷阱,这时仍需要增加一次随机扰动,使得候选修复节点摆脱局部运动陷阱。候选修复节点反复执行上述流程,直到移动到分点的通信范围之内,完成自身的拓扑重构任务。

综上,修复节点所受合力可由式(6)表示

$$F_i = \begin{cases} F_{wander}, & k_1 = 3 \text{ or } k_2 = 3 \\ \sum_{j \in \varphi_1} F_r(i, j) + \sum_{j \in \varphi_2} F_o(i, j) + \alpha F_a(i, j), & o. w. \end{cases} \quad (6)$$

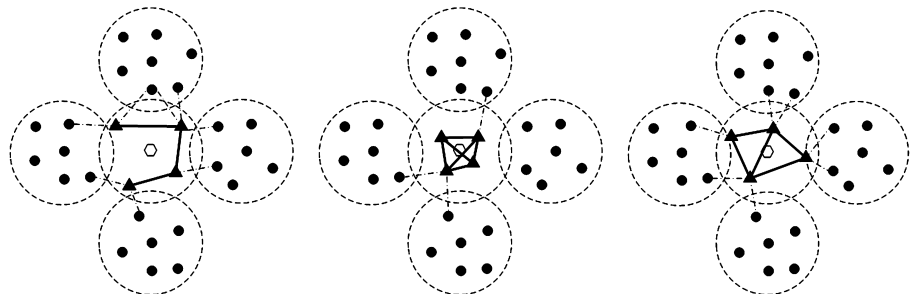
修复节点的运动方程如式(7):

$$M_i(t + \Delta t) = \frac{F_i}{|F_i|} \Delta \bar{w} + M_i(t) \quad (7)$$

其中, $M_i(t)$ 为修复节点 t 时刻的坐标, $M_i(t + \Delta t)$ 为修复节点经过 Δt 时间后的坐标, $F_i/|F_i|$ 为修复节点移动方向的归一化向量, $\Delta \bar{w}$ 为修复节点移动的单位步长,方案中取值为使移动次数和移动路程同时接近最小的长度。

3.4 拓扑优化

拓扑重构过程中,修复节点在本地计算的虚拟力作用下向目的区域移动,移动到目的区域后可能出现图 5(a)和图 5(b)所示的情况。在图 5(a)中修复节点对分点邻域内的非连通子集实现了全覆盖,但修复节点之间的连通性较差;图 5(b)中修复节点之间的连通性



(a) 修复节点间连通性差

(b) 修复节点通信覆盖性差

(c) 联合优化后的局部拓扑

图5 拓扑优化示意图

很强,但是对非连通子集的通信覆盖较差.因此,当所有修复节点进入分点通信范围后,要采集修复区域拓扑信息联合对分点的邻域拓扑进行优化,优化内容包括修复节点之间的连通性和修复节点对分点邻域内非连通子集的通信覆盖性,两个优化内容是相互竞争的关系.优化的目标拓扑如图 5(c),修复节点之间既有较强的连通性又实现了对非连通子集的覆盖.图 5 中,三角形节点表示修复节点,六边形节点表示分点,圆形节点表示分点邻域内的节点.

将拓扑优化问题转化为最小化通信代价问题求解,设分点邻域范围内网络的通信代价为 C ,拓扑优化问题可由式(8)表示:

$$\min C(b_{ij}) = \eta C_r + \lambda C_{rc} \quad (8)$$

其中 $C_r = \sum_{i=1}^N \sum_{j=1}^N b_{ij} c(L_i, L_j)$, 代表修复节点之间的总通信代价; $C_{rc} = \sum_{k=1}^M c(L_{h(k)}, L_k)$, 代表修复节点与非连通子集内代表节点(代表节点指负责所在连通子集与外界连通的节点)的总通信代价. C_r 越大,修复节点间拓扑的连通性就需要越大的通信代价来维护,也说明了修复节点拓扑的连通性很差;同理, C_{rc} 越大,说明修复节点与代表节点的连通性很差.为兼顾修复节点之间的连通性和修复节点对代表节点的覆盖性,应使总的通信代价 C 最小.

L_i, L_j 是修复节点 i, j 的位置, L_k 是代表节点 k 的位置, N 表示修复节点的数目, M 是代表节点数目, $h(k)$ 是覆盖代表节点 k 的修复节点.

b_{ij} 是整数变量, $b_{ij} = \begin{cases} 1, & \text{if } (i, j) \in T \\ 0, & \text{o. w.} \end{cases}$, T 表示邻域拓扑,若修复节点 i, j 之间存在链路,则 $b_{ij} = 1$, 否则 $b_{ij} = 0$.

$c(L_i, L_j)$ 是链路代价函数,代表了链路 (i, j) 的通信代价,表示特定误码率条件下节点 i 向节点 j 发送信息在单位时间内所损耗的通信代价,综合考虑了三种不同的无线通信技术:全向 RF,定向 RF,FSO.由式(9)表示:

$$c_{ij} = k_{ij} (e^{\alpha_{ij} |L_i - L_j|}) (|L_i - L_j|^2) \quad (9)$$

其中, $k_{ij} = P_R(j) \frac{4\pi}{D_T(j) A_R(j)}$, 是常数, $P_R(j)$ 是在保证特定误码率下节点 j 的最小接收功率, $D_T(j)$ 代表发射机的方向, $A_R(j)$ 代表有效接受区域. α_{ij} 是从节点 i 到节点 j 电磁辐射衰减,单位为 dB/km. $|L_i - L_j|$ 代表链路 (i, j) 长度.

修复节点采用基于虚拟力导向的控制方法自主地实现局部拓扑优化.定义两种虚拟力:修复节点之间的引力、修复节点与代表节点之间的引力.由于 c_{ij} 是凸函

数,其负梯度方向即为其数值最快速减小方向,因此将修复节点所受虚拟力定义为 c_{ij} 的负梯度,修复节点沿梯度方向移动使 c_{ij} 的快速达到最小,修复节点停止移动,系统进入平衡状态.

修复节点 i 所受的合力为:

$$F_i = \eta \sum_{j=1}^N b_{ij} f_{ij} + \lambda (h_{(k)} = i) \sum_{k=1}^M h_{f(k)k} f_{ik} \quad (10)$$

其中 $f_{ij} = -\nabla^i c_{ij} = \left[-\frac{\partial c_{ij}}{\partial X_i} \quad -\frac{\partial c_{ij}}{\partial Y_i} \quad -\frac{\partial c_{ij}}{\partial Z_i} \right]^T$, 经计算得

$$f_{ij} = k_{ij} (2e^{\alpha |L_i - L_j|} + \alpha |L_i - L_j| e^{\alpha |L_i - L_j|}) L_{ij} \quad (11)$$

L_{ij} 是从节点 i 的位置 L_i 指向节点 j 的位置 L_j 的位移向量.从式(11)中可以发现,修复节点所受合力仅使用本地信息就可以计算,基于虚拟力导向的节点移动方法具有分布式的本质属性,为动态场景下拓扑优化提供了可扩展和自组织的控制能力.用循环迭代的方法,求出修复节点的新位置, $L_i^{n+1} = L_i^n + \delta F_i$, 其中 R_i^n 表示修复节点 i 在第 n 次循环迭代的位置, F_i 是作用在节点 i 上的虚拟力, δ 是步长系数,通常取大于 0 的数,终止条件为 $|f_{ij}| < \epsilon$.

4 仿真分析

使用 NS-2 网络模拟软件对拓扑重构算法性能进行仿真测试,仿真环境为 1000 个节点分布在 $1000\text{km} \times 1000\text{km} \times 1000\text{km}$ 的范围内,应用层采用 CBR 数据流,每个报文 512 字节,通过设置不同的节点运动场景,采用多次仿真求平均值的方式,对比了在拓扑重构前后,网络性能的变化情况.采用分组的成功投递率、网络的控制开销两个指标作为分析的依据.分组的成功投递率定义为目的节点应用层接收到的分组数与源节点应用层发送的分组数之比,用来反映网络处理和传输数据的能力.控制开销定义为网络中发送的控制分组数与发送的所有分组数之比,用来反映网络带宽的利用率.

4.1 有效性分析

网络的有效性是指网络满足通信业务传输性能要求的程度,例如,随网络负载增加时网络处理数据的能力,是一种基于网络业务性能的可靠性指标.仿真中设置 300 个数据流,对比重构前后网络性能随负载的变化情况.

从图 6(a)和图 6(b)中可以看出,当节点发包速率较慢时,重构前后的分组成功投递率都较高,重构后的网络开销大于重构前.这是由于网络中负载较小时,拓扑中的关键点处不易形成瓶颈而造成拥塞,而重构过程中的节点分类、分点避免和拓扑优化机制会引起网络开销的增加.

随着网络负载的增加,与重构前相比,重构后网络的分组成功投递率始终保持在较高水平,开销减小速度更快.网络负载较重时,大量拓扑中关键点成为了数据传输业务的瓶颈,使网络发生拥塞,大量分组无法成功地到达目的节点,投递率快速减少;同时,网络中充斥大量路由由错误报文以及重路由引发的寻路请求报文、应答报文,使网络的控制开销急剧增加,网络传输效率低下.而对网络进行重构不仅减少了网络瓶颈,修复了拓扑故障;而且,冗余节点的移动有助于抑制通信干扰,提高网络容量.网络拓扑结构在重构算法的作用下逐渐趋于均衡,为路由的优化选择和负载均衡分配提供较好的基础,提高了通信效率,优化了网络性能.

4.2 抗毁性分析

网络的抗毁性指网络经历失效时维持业务传输性能或从失效中快速恢复业务传输性能的能力,反映了破坏网络性能的困难程度.仿真中设置 300 个数据流,发包速率为 1 个数据包/秒,对比重构前后网络性能随故障数的变化情况.

从图 7(a) 和图 7(b)可以看出,当网络中的故障数较少时,重构前后的分组成功投递率都较高,重构后的网络开销略多于重构前.这是由于网络中存在冗余的拓资源,少量故障对拓扑连通性破坏不明显,关键点发生故障的概率也较小;而重构过程会带来一部分网络开销.

随网络遭遇故障数的增加,不采用拓扑重构时分组成功投递率急剧减少,开销快速增加;而重构后的分组成功投递率始终接近 100%,且网络控制开销增加不明显.这是由于网络中出现大量故障时,关键节点的故障概率也随之增大,对拓扑连通性破坏严重,使网络中可用路由减少,造成生存下来的路径传输质量和效率下降,导致成功到达目的节点的分组数急剧减少;同时,路由错误和重路由产生大量控制分组,增加了网络的控制开销.拓扑重构策略通过冗余节点的移动修复拓扑故障并对拓扑结构进行优化,为上层协议的高效运行提供了保障,使网络性能得到了有效的恢复,增强了网络的抗毁性.

5 结论

空间信息网与传统的地面自组织网络相比在网络结构、成网方式和应用环境等方面存在较大差异,对拓扑优化和重构的研究存在迫切需求.本文针对于空间信息网中拓扑相对稳定的骨干网提出一种预防和恢复相结合的拓扑重构策略,预防性重构就是进行拓扑关

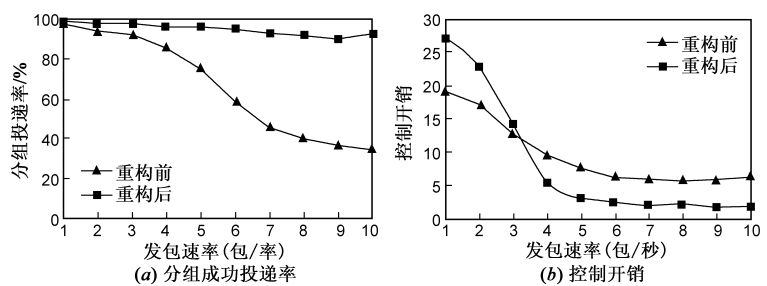


图6 网络有效性

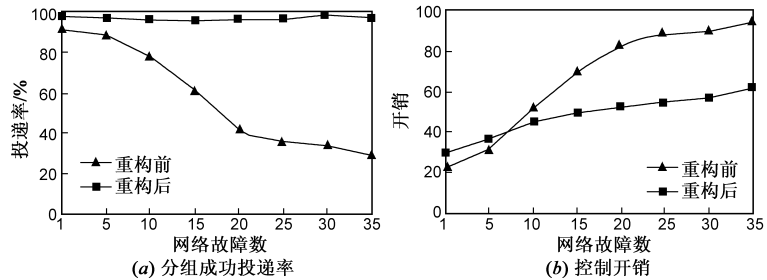


图7 网络抗毁性

键点的检测,并采取措施避免这样的关键节点在拓扑中长期存在;当网络分割时,由发现故障的节点充当分点触发恢复性重构.重构时在一定范围内选择冗余节点,该节点在虚拟力的导向下自主地移动到待修复区域,并以修复区域的局部拓扑通信代价最小为目标,进行拓扑联合优化.使用 NS2 网络模拟软件验证了所提出的空间信息网拓扑重构算法,对比了重构前后网络的有效性和抗毁性,仿真结果表明,所提出的重构策略能够优化网络性能,提高网络的抗毁性.

参考文献

- [1] Joab Jackson. The Interplanetary Internet: NASA researchers quarrel over how to network outer space[J]. IEEE Spectrum Magazine, 2005, 43(4): 31 - 35.
- [2] Stuart D M, Jaime L, Christopher C D. Autonomous reconfiguration and control in directional mobile ad hoc networks[J]. IEEE Circuits and Systems Magazine, 2009, 9(2): 10 - 26.
- [3] Tuan L, Nadeem A, Sanjay J. Location-free repair in hybrid sensor networks[A]. InterSense'06 Proceedings of the First International Conference on Integrated Internet Ad Hoc and Sensor Networks[C]. ACM, New York, NY, USA: 2006.
- [4] Chang B H, Liang Y H, Lin Y M. Distributed route repair for increasing reliability and reducing control overhead for multicasting in wireless MANET[J]. Information Science, 2009, 179(11): 1705 - 1723.
- [5] Yu C W, Wu T K, Cheng R H. A low overhead dynamic route repairing mechanism for mobile ad hoc networks[J]. Computer Communications, 2007, 30(5): 1152 - 1163.
- [6] 李 ■, 刘军. 基于 AODV 协议的自组网络安全机制的研究[J]. 电子学报, 2006, 34(2): 272 - 276.

- Li Zhe, Liu Jun. The research on security mechanism based on AODV routing protocol in mobile Ad Hoc network[J]. Acta Electronica Sinica, 2006, 34(2): 272 - 276. (in Chinese)
- [7] Tuan L, Nadeem A, Nandan P, Sanjay J. Fault repair framework for mobile sensor networks[A]. First International Conference on Communication System Software and Middleware (IEEE Comsware 2006)[C]. New Delhi, India: IEEE, 2006.
- [8] Yigal B, Pramod V K. Optimal construction of redundant multi-cast trees in directed graphs[A]. Proceedings of the 28th Conference on Computer Communications (IEEE INFOCOM 2009)[C]. Rio de Janeiro, Brazil: IEEE, 2009.
- [9] Rajmohan R. Topology control and routing in Ad Hoc networks: a survey[J]. ACM Computing Surveys, 2005, 33(2): 60 - 73.
- [10] Paolo S. Topology control in wireless Ad Hoc and sensor networks[J]. ACM Computing Surveys, 2005, 37(2): 164 - 194.
- [11] Ganeriwal S, Kansal A, Srivastava M B. Self aware actuation for fault repair in sensor networks[A]. IEEE International Conference on Robotics and Automation (ICRA)[C]. Piscataway NJ, USA: IEEE, 2004.
- [12] Saha I, Sambasivan L K, Patro R K, Ghosh S K. Distributed fault-tolerant topology control in static and mobile wireless sensor networks[A]. 2nd International Conference on Communication Systems Software and Middleware 2007 (IEEE Comsware 2007)[C]. Bangalore, India: IEEE, 2007.
- [13] Huang Z C, Zhang Z S, Ryu B. Power control for directional antenna-based mobile Ad Hoc networks[A]. International Conference on Communications and Mobile Computing[C]. Proceedings of the International Conference on Communications and Mobile Computing[C]. ACM, New York, USA, 2006.
- [14] Howard A, Mataric M J, Sukhatme G S. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem[A]. The 6th International Sympon Distributed Autonomous Robotics Systems (DARS)[C]. Fukuoka, Japan: DARS, 2002.
- [15] Zou Y, Chakrabarty K. Sensor deployment and target localization based on virtual forces[A]. The 22nd Conference of the IEEE Computer and Communications Society (INFOCOM)[C]. San Francisco, California: INFOCOM, 2003.

作者简介



刘 军 男, 1969 年生于辽宁锦州. 东北大学信息科学与工程学院副教授. 研究方向为空间网络、自组织网络.

E-mail: liujun@ise.neu.edu.cn

于 耕 男, 1973 年生于陕西西安. 沈阳航空航天大学民航学院, 研究方向为航空机载电子信息系统.