

## 基于冗余节点选择模型的水下传感器网络拓扑修复

景彦凯, 张 钢, 刘春风

(天津大学 计算机科学与技术学院 天津市先进网络重点实验室, 天津 300350)

**摘 要:** 现有水下传感器网络的拓扑修复算法大多只是完成网络连通性修复, 未考虑节点能耗过快造成网络寿命缩短的问题。为此, 提出一种基于冗余节点选择模型的拓扑修复算法。该算法在网络部署完成后利用分布式的方法选择关键节点并对其进行监控。当节点失效时, 使用冗余节点选择模型选择冗余节点, 通过移动冗余节点对失效节点进行修复, 同时对冗余节点采取睡眠唤醒策略以延长网络寿命。实验结果表明, 与区域移动修复算法相比, 该算法在节点移动总距离、网络寿命、失效节点首次出现时间、投递率等方面性能均有所提高。

**关键词:** 水下传感器网络; 拓扑修复; 冗余节点; 节点移动; 网络寿命

**中文引用格式:** 景彦凯, 张 钢, 刘春风. 基于冗余节点选择模型的水下传感器网络拓扑修复[J]. 计算机工程, 2017, 43(11): 90-96, 101.

**英文引用格式:** JING Yankai, ZHANG Gang, LIU Chunfeng. Topology Repair in Underwater Sensor Network Based on Redundancy Node Selection Model[J]. Computer Engineering, 2017, 43(11): 90-96, 101.

### Topology Repair in Underwater Sensor Network Based on Redundancy Node Selection Model

JING Yankai, ZHANG Gang, LIU Chunfeng

(Tianjin Key Laboratory of Advanced Networking, School of Computer Science and Technology, Tianjin University, Tianjin 300350, China)

**[Abstract]** In underwater sensor network, most existing repair algorithms of network topology just consider to repair the connectivity of the network, without taking into account the network lifetime. To deal with the above problems, an algorithm based on redundancy node selection model is proposed. In this algorithm, a distributed manner is adopted to select critical nodes after network deployed, in which the redundancy nodes are selected to complete the topology repair by moving to the position when the critical nodes are failure. Meanwhile, the sleep/wake strategy is adopted to prolong the network lifetime. Experimental result shows that, compared with Block Movement Repair (BMR) algorithm, the proposed algorithm can greatly reduce the node's moving distance, prolong the network lifetime and guarantee delivery ratio.

**[Key words]** underwater sensor network; topology repair; redundancy node; node movement; network lifetime

**DOI:** 10.3969/j.issn.1000-3428.2017.11.015

## 0 概述

水下传感器网络是指将能耗很低、具有较短通信距离的水下传感器节点部署到指定海域中, 利用节点的自组织能力自动建立网络<sup>[1]</sup>。水下传感器网络由固定在海底或悬浮在水中的传感器节点组成。节点搭载的传感器负责采集温度、深度、海洋生物、突发事件信息等参数, 这些数据通过声学通信方式传递给漂浮在水面的基站或船舶。水面设施采用无线电方式传给卫星或互联网, 最终到达用户手中。水下传感器网络在海洋环境研究、海底资源勘探、国

土防卫等方面被广泛应用<sup>[2]</sup>。当前, 国内外在水下传感器网络领域的研究工作主要围绕水下通信技术、媒介接入技术、路由、同步、定位等方面展开, 在网络拓扑方面的研究较少。

网络拓扑在网络能量消耗、网络容量和可靠性方面起着决定性的作用<sup>[3]</sup>。在水下传感器网络中, 网络的连通性是进行数据转发、数据交换的基础。但是由于水下环境的特殊性, 传感器节点存在易失效、不易恢复的可能性, 特别是当起桥梁作用的节点失效后, 会将网络分割成几个不连通的部分。在这种情况下, 也无法通过选择新的路径将数据发送到

**基金项目:** 国家自然科学基金(61402327); 天津大学自主创新基金(2014XZ-0119)。

**作者简介:** 景彦凯(1992—), 男, 硕士, 主研方向为水下传感器网络; 张 钢, 教授、博士; 刘春风(通信作者), 副教授、博士。

**收稿日期:** 2016-10-12 **修回日期:** 2016-11-29 **E-mail:** cfliu@tju.edu.cn

Sink 节点。水下传感器网络必须能够及时、高效地进行拓扑修复,保证网络的连通性。为此,本文在冗余节点选择模型的基础上,提出一种水下传感器网络拓扑修复算法。

## 1 相关工作

目前,国内外研究在水下传感器网络拓扑修复方面均取得了一定的进展。文献[4]提出增强失效节点的邻居节点通信功率,从而增大邻居节点的通信半径来重新建立连接,完成拓扑修复。在水下传感器网络中,节点的能量是有限的,因此,这种方式将会增加失效节点邻居节点的能耗,可能会很快出现新的失效节点,最终缩短网络寿命。文献[5]提出基于水下自组织航行器(Autonomous Underwater Vehicle, AUV)节点,通过 Steiner 树问题的拓扑愈合算法来进行拓扑修复。但这种修复算法没有考虑到水下环境的特殊性,而且在失效节点的发现和 AUV 节点如何移动方面没有给出具体的说明。文献[6]通过鱼群启发式算法来部署 AUV 节点。这种修复算法需要借助 AUV 来执行拓扑修复,并不具有普遍性。文献[7]提出通过调整节点的深度来完成拓扑修复,并给出了 2 种拓扑修复算法,分别从移动大量节点和移动一个节点方面考虑。在区域移动修复(Block Movement Repair, BMR)算法<sup>[7]</sup>中,只是单纯地将失效节点下方的所有节点向上移动,完成了网络连通性的修复,但是由于大量节点移动造成网络的能耗加快,网络寿命缩短。分布式水下修复算法(Distributed Underwater Repair Algorithm, DURA)<sup>[7]</sup>在选择单个节点进行移动时,没有考虑到此节点的能量,可能存在修复完成不久后再次出现断裂的情况,而且网络修复所需时间也较长。

现有的拓扑修复算法或是单一地考虑对网络连通性进行修复,没有考虑到网络能耗和寿命;或是没有考虑水下传感器网络的特殊性。当起桥梁作用的关键节点失效造成网络被分割成不连通部分时,可以利用水下传感器网络中存在的冗余节点来对网络拓扑进行修复。在进行网络连通性修复的同时,也能够均衡网络能耗,延长网络寿命。在关键节点选择方面,文献[8]提出了一种分布式的方法来进行关键节点选择,这种方法比较简便且不需要知道整个网络的拓扑结构,更符合水下传感器网络的实际情况。但是仅依靠邻居节点间的距离来进行关键节点选择往往会造成关键节点数目远超过实际情况,而造成非必要的拓扑修复次数大幅增加,对整个水下传感器网络的寿命造成影响。在冗余节点选择方面,文献[9]讨论了一种通过概率模型来进行冗余节点选择的方法,并提出了期望覆盖质量(Expected Deployment Quality, EDQ)的概念。节点数目越多,EDQ 的值也越大。该方法通过判断节点冗余度的

期望与 EDQ 之间的关系,来判断节点是否为冗余节点。这种方法简化了冗余节点的选择过程。由于是通过邻居节点的数目来进行冗余节点的选择,因此选择的顺序是十分重要的,但是文献[9]中并没有对选择顺序进行说明,同时也没有考虑到节点的能量情况,只是单纯地以邻居节点数目作为标准进行选择,没有考虑到水下传感器网络的实际情况,因此,不能简单地将其应用到拓扑修复算法中。

本文以冗余节点选择模型<sup>[9]</sup>为基础,通过对一定范围内的冗余节点的深度进行调整来完成拓扑修复,同时对其他冗余节点采取睡眠唤醒策略<sup>[10]</sup>进一步提高网络的性能。本文提出 RNMR(Redundancy Node Movement Repair)算法,并将该算法在 NS2 上进行仿真实验,验证其对水下传感器网络进行拓扑修复的可行性。

## 2 基于冗余节点选择模型的拓扑修复算法

在水下传感器网络中,网络连通的重要性不言而喻。当网络中一些关键节点失效后,对网络进行拓扑修复是很有必要的。在水下传感器网络中进行拓扑修复的主要技术难点有:1)关键节点失效造成网络被分割成几个不连通的部分,无法使用路由协议进行修复;2)水下传感器节点能够进行深度的调整,但水平方向的调整困难;3)进行拓扑修复时要考虑到能耗对网络寿命的影响<sup>[7]</sup>。

在进行拓扑修复时,需要对传感器节点进行重定位,并且节点需要在尽可能短的时间内移动到指定位置,来完成网络拓扑的修复。由于水下环境的特殊性,传感器节点在进行水平方向上的移动时需要额外的动力,并且会消耗大量的能量,但是传感器节点在进行竖直方向上的移动所需要的能量相对来说较少<sup>[11]</sup>,因此通过传感器节点竖直方向上的移动来完成网络拓扑的修复是可行的。

由于水下传感器网络中的节点很难通过太阳能进行充电,对节点的电池进行替换也很困难<sup>[12]</sup>,因此在进行传感器网络的拓扑修复时,还要考虑到能耗对网络寿命的影响。单纯地通过增强通信功率、扩大通信半径的方法将会加大失效节点邻居节点的能量消耗。在邻居节点能量耗尽后,需要进行新的修复,加速了网络的能量消耗,大幅缩短了网络的寿命。在通过移动节点对拓扑进行修复时,也需要考虑到节点的能量问题,否则将会出现节点在移动过程中就将能量耗尽,从而无法完成修复的现象。

在 RNMR 拓扑修复算法中,水下传感器网络采取随机部署的方法。在部署完成后节点通过关键节点选择模型判断自身是否为关键节点,同时根据能量、距离等条件选择出失效处理节点对其状态进行监测。当失效处理节点检测到关键节点失效时,此时认为网络处于不连通状态,发送冗余节点选择包,

通过冗余节点选择模型进行冗余节点选择,在冗余节点选择完成后,通过冗余节点调度模型对冗余节点进行调度,完成拓扑修复。

### 2.1 关键节点选择模型

当水下传感器网络中关键节点失效时,网络将会被分割成为几个不连通的部分,但是如果是普通节点失效,对于网络的连通性并没有影响,此时不需要进行拓扑修复。因此,在网络初始化完成后,需要对网络中的关键节点进行选择。传统的 DFS (Depth First Search) 拓扑修复算法<sup>[13]</sup>需要掌握整个网络的拓扑后才能够进行关键节点选择,不适合水下传感器网络。本文采用一种基于分布式的方法来选择网络中的关键节点。当节点部署完成后,每个节点计算邻居节点间的距离,如果存在必须通过邻居节点才能进行通信的情况,则计算邻居表中是否有其他节点能够与这 2 个节点进行通信,如果没有则认为该节点是关键节点。

如图 1 所示,假设节点  $u$  的 2 个邻居节点  $a$  和  $b$  的距离为  $d_{ab}$ ,可表示为:

$$d_{ab} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2} \quad (1)$$

如果邻居节点间的距离  $d_{ab}$  大于 2 倍通信半径,也就意味着节点与节点  $a$  和  $b$  可能位于 2 个不同的区域,2 个节点无法实现直接通信,接下来要判断节点  $u$  的邻居节点中是否存在能够与节点  $a$  和  $b$  同时通信的节点,如果存在节点  $v$  使得  $d_{av} < r$  同时  $d_{bv} < r$  ( $r$  为通信半径),也就说明节点  $a$  和  $b$  能够通过节点  $v$  进行通信,此时,不能够认为其是关键节点。反之,则认为节点  $a$  和  $b$  必须要通过节点  $u$  才能够实现通信,节点  $u$  承担着桥梁的作用,如果节点  $u$  失效,可能会造成节点  $a$  所在区域与节点  $b$  所在区域无法实现通信,网络被分割,因此,节点  $u$  可以认为是关键节点。

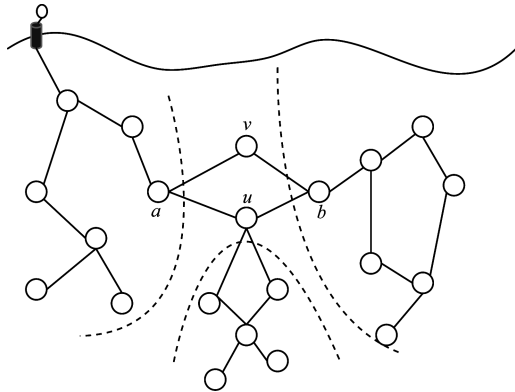


图 1 关键节点选择模型示意图

### 2.2 冗余节点选择模型

在冗余节点的选择方面,不仅要考虑到节点冗余度的期望值与 EDQ 之间的关系,而且还要考虑到

冗余节点的选择顺序以及节点的能量等方面。节点的冗余度可以表示为:

$$\xi(i) = \frac{(\cup_{j \in \Psi_i} SD_j) \cap SD_i}{SD_i} \quad (2)$$

其中,  $SD_i$  表示节点  $i$  的传感范围,  $\Psi_i$  表示节点  $i$  的邻节点集合。节点自身的传感范围与邻居节点传感范围重合的部分占自身传感范围的比例就是节点的冗余度,即  $\xi(i)$ 。

所有的传感器节点随机部署在监测区域  $\Omega$  中,一个节点位于  $\Omega$  区域任意一点的概率为  $1/\|\Omega\|$  ( $\|\Omega\|$  表示监测区域的体积)。为了更加贴近实际情况,将传感器节点的传感半径设定为服从  $N(r_0, \delta^2)$  正态分布。对于传感半径分别为  $r_i$  和  $r_j$  的 2 个传感器节点  $S_i$  和  $S_j$ ,如果存在重合的传感区域,  $S_j$  需要在以  $S_i$  为球心、 $r_i + r_j$  为半径的球体传感范围内,而重叠区域的点一定在  $S_j$  的传感范围内。使用  $p_{ij}$  来表示  $S_i$  和  $S_j$  存在重叠区域的概率,则  $S_i$  和  $S_j$  存在重叠区域的概率可以表示为:

$$p_{ij} = \frac{\left(\frac{4}{3}\right)\pi r_j^3}{\|\Omega\|} = \frac{r_j^3}{\left(\frac{4}{3}\right)\pi (r_i + r_j)^3} \quad (3)$$

令  $E(p_{ij})$  为  $p_{ij}$  的数学期望,根据  $p_{ij}$  的计算公式得到  $E(p_{ij})$ :

$$\begin{aligned} E(p_{ij}) &= \frac{1}{2\pi\delta^2} \int_0^{2r_0} \frac{r_j^3}{(r_i + r_j)^3} e^{-\frac{(r_i - r_j)^2 + (r_j - r_0)^2}{2\delta^2}} dr_i dr_j \\ &= \frac{1}{4} - \frac{e^{-r_0}}{4} \end{aligned} \quad (4)$$

用  $\overline{p_{ij}}$  来表示  $S_i$  和  $S_j$  节点不存在重叠区域的概率,  $\overline{p_{ij}} = 1 - p_{ij}$ 。如果节点  $S_i$  有  $k$  个邻居,  $S_i$  节点与至少一个邻居节点存在重叠区域的概率为  $p_{ik}$ ,与任何邻居节点均没有重叠的传感区域的概率为  $\overline{p_{ik}}$ ,则  $\overline{p_{ik}}$  可表示为:

$$\overline{p_{ik}} = (\overline{p_{ij}})^k = (1 - p_{ij})^k \quad (5)$$

由式(5)可将  $p_{ik}$  表示为:

$$p_{ik} = 1 - \overline{p_{ik}} = 1 - (1 - p_{ij})^k \quad (6)$$

如果节点冗余度的期望值大于 EDQ,即  $S_i$  节点的冗余度  $\xi(i)$  的期望满足式(7),也就说明  $S_i$  与其邻居节点重叠的比例大于 EDQ 值,即使让  $S_i$  节点进入睡眠状态覆盖率也会达到 EDQ 值。

$$E(\xi(i)) \geq E_{EDQ} \quad (7)$$

其中,  $E_{EDQ}$  为 EDQ 值。

基于以上分析,由式(2)、式(4)、式(6)、式(7)可以得到式(8),即如果节点  $S_i$  的邻居节点数目满足

式(8),则可以认为  $S_i$  符合成为冗余节点的基本条件。

$$k \geq \frac{\ln(1 - E_{EDQ})}{\ln\left(\frac{3}{4} + \left(\frac{1}{4}\right)e^{-r_0}\right)} \quad (8)$$

在水下传感器网络中,节点收集到的数据最终要传递给 Sink 节点,因此,越靠近水面的节点在能量消耗上往往越快。如果靠近水面的冗余节点多一些的话,当非冗余节点能量耗尽后,通过唤醒冗余节点仍能够保证数据的正常传递。在本文算法中,将水下区域按照深度划分成 10 层,处于第  $n$  层的节点深度满足下式:

$$z_n \in [(z/10) \times (n-1), (z/10) \times n] \quad (9)$$

其中,  $z$  表示水下传感器网络所监测区域的深度,  $z_n$  表示第  $n$  层的深度取值范围,  $n \in [1, 10]$ 。自上而下逐层进行冗余节点选择。如果节点  $S_i$  的邻居节点数目满足式(8)且节点  $S_i$  处于第 1 层,那么  $S_i$  判断自身能量是否高于给定的阈值,如果满足,则确定为冗余节点,同时向邻居节点发送状态更改包。如果  $S_i$  位于  $m$  层,  $m \in (1, 10)$ ,则在等待  $t \times m$  的时间或者在收到上层冗余节点的状态更改包后进行冗余节点选择,其中  $t$  为设定的等待时间。如果节点位于第 10 层,则等待时间为  $10 \times t$ 。如图 2 所示,位于第  $m$  层的节点  $a$  在确定自身为冗余节点后,发送状态更改包到邻节点  $b$ 。节点  $b$  判断邻居节点中非冗余节点数目是否满足式(8),如果满足,则进行能量判断,高于阈值的话则发送状态更改包,反之则不进行状态更改。

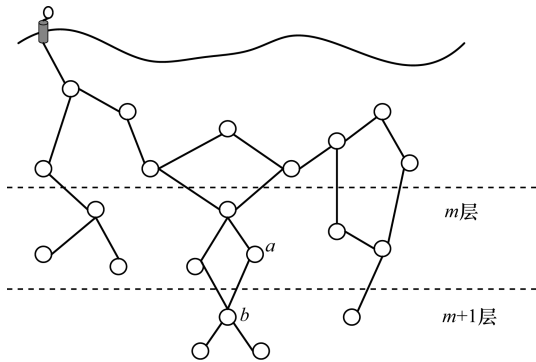


图2 冗余节点层次选择示意图

冗余节点选择模型是基于概率的分层选择模型,能够较为简单地进行冗余节点选择,相比较于传统的冗余节点选择模型,其无需对水下传感器网络进行划分和复杂的冗余度计算,符合水下传感器网络的实际情况,能够较为快速地选择冗余节点并进行水下传感器网络拓扑修复。

### 2.3 冗余节点调度模型

在冗余节点选择完成之后,通过对冗余节点进行调度来完成拓扑的修复以及网络能耗的均衡。

在水下传感器网络中,节点可以通过周期性的唤醒和睡眠机制来有效地降低网络能耗<sup>[14]</sup>。这里将所确定的冗余节点分为 2 个用途,在以失效的关键节点为圆心,以通信距离为半径,下方一倍通信距离为高的圆柱体范围内,通过移动冗余节点来完成对失效的关键节点的修复。可移动节点需根据自身能量情况以及与关键节点间的距离情况选择是否进行移动,如果存在冗余节点可以进行移动,则向失效处理节点发送可移动包 *move*。在这个范围外的冗余节点先进入睡眠状态,降低能量的消耗。失效处理节点首先基于下式来计算完成修复所需要的步数:

$$\text{stepcount} = \lceil \text{Min. } z - u. z \rceil / (r/20) \quad (10)$$

其中, *stepcount* 表示失效关键节点下方的冗余节点到达失效处理节点的深度位置所需要的最大步数, *Min. z* 表示失效的关键节点下方在一倍通信范围内最深的节点的  $z$  坐标,  $u. z$  表示失效关键节点的深度。这里设定节点每次移动的距离为  $r/20$ ,  $r$  表示节点的通信距离。当冗余节点到达相应的深度位置后,获取其邻居节点信息,判断其邻居节点中是否包含失效处理节点。如果包含,失效处理节点通过检测 2 跳邻居节点的信息来确认是否完成修复;反之,则等待其他冗余节点移动到相应的深度位置。当完成修复后失效处理节点发送 *success* 消息,通知其他正在移动的冗余节点停止移动。

由于存在在移动范围内没有冗余节点或者冗余节点所处位置距离失效的关键节点太远,不适合进行移动来完成拓扑修复的可能,此时应放宽冗余节点选择条件,即减少邻居节点所要满足的数目。如果失效处理节点在等待一定时间后没有收到可移动包 *move*,则失效处理节点将冗余节点选择模型所确定的邻居节点数目进行相应的减少,通知邻居节点数目高于这个值的节点进行向上移动,完成拓扑修复。

对于范围之外的节点,先让其进入睡眠状态,数据的转发主要通过其邻居节点进行。当邻居节点的能量低于设定的阈值时,向冗余节点发送唤醒包,对其进行唤醒和数据的转发,以延长网络寿命。

### 2.4 算法实现

本文算法的实现主要分为关键节点选择模型的实现、冗余节点选择模型的实现、冗余节点调度模型的实现等几个部分。

在关键节点选择算法中,首先要完成邻居表的建立,节点通过发送 *Hello* 数据包来建立邻居表,这个数据包中包含节点位置信息、节点编号、节点类型、能量等相关信息。接下来需要对邻居表中的节点之间的距离进行判断,确认自身是否为关键节点。

关键节点的选择是在网络部署完成后进行的, 优先进行选择, 在其选择完成后会发送节点状态更改包, 改变节点的状态。RNMR 拓扑修复算法中的关键节点选择算法如算法 1 所示。其中, *Distance* 代表 2 倍通信距离; *NodeState* 代表节点的状态, 0 代表普通节点, 1 代表关键节点。

**算法 1** 关键节点选择算法

```
SelectCRnode( )
1. Distance = 2r
2. NodeState = 0
3. Flag = true
4. FOR all neighbor nodes DO
5.   Get node i coordinate (  $x_i, y_i, z_i$  ) from all neighbor nodes
6.   FOR other neighbor nodes DO
7.     Get node j coordinate (  $x_{dj}, y_{dj}, z_{dj}$  ) from other neighbor nodes
8.     Calculate the distance d between the two nodes
9.     IF  $d > \text{Distance}$  THEN
10.      SET backup[i][j] = 1
11.    ENDIF
12.  ENDFOR
13. ENDFOR
14. IF backup[i][j] = 1 THEN
15.  FOR other nodes DO
16.    Get node k coordinate (  $x_k, y_k, z_k$  ) from other nodes
17.    Calculate the distance  $d_1$  between node i and node k
18.    Calculate the distance  $d_2$  between node j and node k
19.    IF  $d_1 < \text{Distance}/2$  and  $d_2 < \text{Distance}/2$  THEN
20.      SET Flag = false
21.      BREAK
22.    ENDIF
23.  ENDFOR
24. ENDIF
25. IF Flag = true THEN
26.  NodeState = 1
27.  insert change state packet to the send queue
28. ENDIF
```

算法 2 与算法 3 为冗余节点选择及调度的算法。当失效处理节点检测到关键节点失效时, 其发送数据包开始进行拓扑修复, 收到数据包的节点根据节点位置来决定开始冗余节点选择时间。根据冗余节点选择模型, 分层进行冗余节点选择, 冗余节点的选择在关键节点失效后进行。对于非冗余节点或者能量小于设定阈值的节点, 如果没有转发过该数据包, 则转发, 否则丢弃, 避免数据包重复转发造成网络拥塞。在冗余节点选择完成后, 依据冗余节点调度模型对冗余节点进行调度, 完成拓扑修复, 其

中; *EnergyThread* 表示满足冗余节点条件的能量阈值; *RedundantNeighbor* 表示满足冗余节点所需的邻居节点数目; *NodeNeighbor* 表示节点邻居节点数目; *MovingEnergy* 表示移动节点需要满足的能量阈值; *SecondMove* 表示是否减少邻居节点数目, 如果为真则表示数据包中存在一个新的设定值 *SecondNeighbor*, 如果邻居节点数目大于这个值, 则进行向上的移动, 其主要是针对范围内非冗余节点来进行。

**算法 2** 冗余节点选择算法

```
Repair Topology( p )
1. IF NodeNeighbor > Redundancy Neighbor
2.  Calculate level according node self coordinate
3.  IF level = 1 and nodeenergy > Energy Thread THEN
4.    SET NodeState = 2
5.    insert change state packet to the send queue
6.  ELSEIF level > 1 and level < 10 and nodeenergy > EnergyThread THEN
7.    Computing wait time
8.  IF( receive packet from upper level or wait time arrival )
9.    SET NodeState = 2
10.   insert change state packet to the send queue
11.  ENDIF
12. ELSEIF level = 10 and nodeenergy > EnergyThread THEN
13.   Computing wait time
14.   Wait until wait time arrival
15.   SET NodeState = 2
16.   insert change state packet to the send queue
17. ELSEIF p has been forwarded THEN
18.  Drop p
19. ELSE
20.  insert p to the send queue
21. ENDIF
```

**算法 3** 冗余节点调度算法

```
RnodeSchedule( p )
1. Get critical node's coordinate (  $x_c, y_c, z_c$  ) from p
2. Calculate range satisfy moving
3. IF NodeState = 2 THEN
4.  IF current node's coordinate in the range and node energy > MovingEnergy THEN
5.    Move to the destination
6.    Send move package to failure handler node
7.  ELSE
8.    Make the node to sleep until it receive wake up package
9.  ENDIF
10. ELSEIF SecondMove = true and current node's coordinate in the range and Node Neighbor > Second Neighbor THEN
11.  Move to the destination
12.  Send move package to failure handler node
13. ENDIF
14. ELSEIF p has been forwarded THEN
```

15. Drop p
16. ELSE
17. insert p to the send queue
18. ENDIF

### 3 算法性能评价

#### 3.1 参数设置

实验将验证在 NS2 仿真平台 Aqua-Sim 模块上调用本文算法对水下传感器网络进行拓扑修复的可行性。Aqua-sim 是专门模拟分析水声传感器网络的仿真软件<sup>[15]</sup>。本文设定在长度为 450 m,宽度为 350 m,深度为 500 m 的水下区域内随机部署节点,传感器节点数目从 400 增加到 600。传感器节点的通信距离为 100 m。每个节点的初始能量为 150 J,节点的通信功率为 2 W,接收功率为 0.75 W,空闲状态的功率为 0.008 W。仿真时间为 3 000 s。发送数据包的节点设为 4 个,发送数据包的时间间隔为 35 s。节点移动的速度为 2.4 m/min,节点每次移动的能耗为 2.4 J。Sink 节点位于水面位置,其坐标的长度、宽度、深度值分别为 225,175,0,用来接收数据包。由于只是进行拓扑修复方面的研究,因此假定节点能够进行准确定位,同时不考虑节点水平方向的移动,节点移动只在竖直方向上进行。

#### 3.2 结果分析

在本文所提出的 RNMR 拓扑修复算法中,通过失效处理节点是否能通过移动上来的冗余节点与 2 跳邻居节点建立通信来判断修复是否完成,即网络是否恢复连通。如果修复失败,则由失效处理节点向 Sink 节点发送修复失败的消息包。水下传感器网络的寿命取决于节点能量,在本文中节点能量的消耗主要在接收、转发数据包以及垂直方向的移动 2 个方面。在网络中,如果节点的能量变为 0,则意味着节点死亡,如果在网络中有超过 10% 的节点死亡,则意味着网络瘫痪。这里针对不同的节点数目,主要分析了 4 个性能指标:完成修复时节点的移动总距离,水下传感器网络的网络寿命,投递率以及非关键节点的失效节点第一次出现的时间。在相同节点数目下进行 10 次仿真,结果取平均值,并将本文所提出的拓扑修复算法与文献[7]中的 BMR 算法进行比较。在 RNMR 算法仿真中 Sink 节点均未收到修复失败的消息包。

本文所进行的仿真均在网络处于全连通的状态下进行,即 Sink 节点能够接收到 4 个源节点所发送的数据包。由于水下传感器网络采用声波通信,为了使源节点的数据能够顺利被 Sink 节点接收,网络中存在的节点数目较多。由于节点数目较多,在网络中存在着较多的冗余节点。

节点移动总距离随节点数目的变化趋势如图 3

所示,从中可以看出完成修复时节点移动总距离随节点数目的变化趋势。在本仿真中,节点移动总距离是指在本次仿真时间内进行拓扑修复后移动节点的总距离,即将所有移动节点的距离之和。节点数目从 400 增大到 600,RNMR 算法完成修复时,节点移动距离从 446 m 增加到 618 m,而 BMR 算法则由 1 451 m 增加到 2 450 m。当关键节点失效时,BMR 算法移动了其下方所有节点来完成修复,节点数目越多,移动距离越长。由于 RNMR 算法对关键节点选择方面进行了优化,同时利用冗余节点选择模型和冗余节点调度模型选择部分冗余节点进行移动,因此能够较大幅度地减少节点移动总距离。

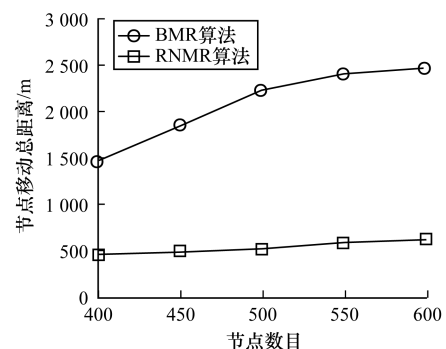


图 3 节点移动总距离随节点数目的变化趋势

图 4 描述了采用不同修复算法对水下传感器网络进行拓扑修复时,网络寿命随节点数目的变化情况。在最初始状态,即节点数目为 400 时,RNMR 拓扑修复算法同 BMR 拓扑修复算法均能使其在仿真时间内存活。当节点数目增多时,RNMR 拓扑修复算法使得网络寿命的缩减速度变得缓慢,而 BMR 拓扑修复算法则造成网络寿命较为明显的缩短。在对关键节点进行修复的过程中,BMR 拓扑修复算法移动了大量节点,而且在移动的节点中不对节点的能量情况进行判断,造成某些节点能量消耗加快,从而降低网络寿命。随着节点数目的增多,所需移动的节点数目也随之增加。同时,由于节点的大量移动可能会造成某些区域节点过于密集,大量数据被重复转发,也会造成网络能耗的增加,缩短网络寿命。当节点数目达到 600 时,采用 BMR 算法进行拓扑修复的水下传感器网络的网络寿命减少到 2 090 s。而 RNMR 在选择冗余节点时,就考虑到节点的剩余能量问题,进行移动修复的节点能量较高,在完成修复的同时,能够较长时间地完成数据转发工作,同时进入睡眠状态的冗余节点的能量均较高,在其被唤醒后,能够优先承担能量较低的邻居节点的转发任务,均衡了网络能耗,避免了以缩短网络寿命为代价的拓扑修复。

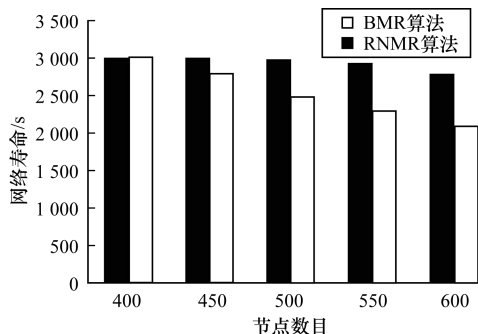


图 4 网络寿命随节点数目的变化趋势

对不同节点数目下, RNMR 与 BMR 拓扑修复算法的失效节点首次出现时间进行比较, 结果如图 5 所示。可以看出, 在本文仿真中, 当非关键节点能量为 0 时, 将其定义为失效节点, 失效节点的出现时间能够反映网络能耗的均衡性。如果在仿真过程中没有出现失效节点, 则将其出现时间设定为仿真时长。随着节点数目的增加, 在采取 BMR 拓扑修复算法的水下传感器网络中, 失效节点出现的时间越来越早, 而采取 RNMR 拓扑修复算法的水下传感器网络中失效节点出现时间要远远晚于 BMR 拓扑修复算法。这主要是由于 BMR 拓扑修复算法在进行拓扑修复时, 移动了下方所有的节点来进行拓扑修复, 而没有进行能量阈值的划分, 节点大量移动对所有节点的能量都进行消耗。移动上来的节点很可能能量比较低, 而关键节点起着连通子网络的功能, 承担着较多的数据转发, 因此会加速能量消耗, 造成节点死亡时间提前。在 RNMR 拓扑修复算法中, 选择进行移动的冗余节点在能量方面往往比较充足, 能够较长时间地承担数据转发任务, 同时随着节点数目的增多, 冗余节点也随之增多, 能够进行睡眠唤醒调度的冗余节点也增多, 避免了某些节点能量消耗过快的情形, 延缓了节点的能量消耗。

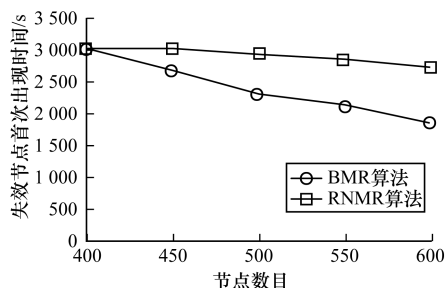


图 5 失效节点首次出现时间随节点数目的变化趋势

对不同节点数目下, RNMR 与 BMR 拓扑修复算法的投递率进行比较, 结果如图 6 所示。可以看出, RNMR 的投递率高于 BMR 的投递率。RNMR 移动的只是符合条件的冗余节点, 避免了某一区域节点过于密集造成网络拥塞, 同时分层进行冗余节点选择能够较大程度上保证数据包转发到 Sink 节点。

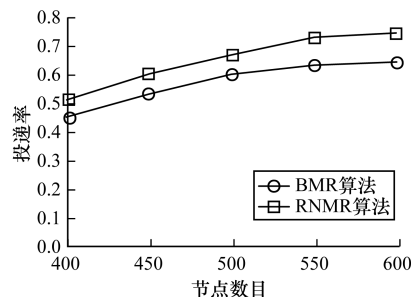


图 6 投递率随节点数目的变化趋势

## 4 结束语

水下传感器网络与普通传感器网络的主要区别是水下环境会引起节点能量的严重损耗, 同时节点也容易损坏, 如果处在关键位置的节点能量耗尽或者损坏, 将会造成整个网络的瘫痪。因此, 需要设计拓扑修复算法, 恢复网络的连通性。本文以一种冗余节点选择模型为基础, 利用关键节点选择模型和冗余节点调度模型完成拓扑的修复, 同时给出详细的推导过程和计算方法。通过 NS2 平台进行模型实现, 仿真分析算法对水下传感器网络进行拓扑修复过程中节点的移动距离以及网络寿命, 并与 BMR 拓扑修复算法进行对比, 验证算法的可行性。下一步将研究拓扑修复算法对投递率的影响以提升投递率, 同时对如何降低网络能耗进行研究, 完善修复算法。由于水下传感器网络的研究还处于起步阶段, 完善的理论体系仍有待建立, 因此还需要进行大量的仿真实验。

## 参考文献

- [1] AKYILDIZ I F, POMPIL D, MELODIA T. State of the Art in Protocol Research for Underwater Acoustic Sensor Networks [J]. ACM SIGMOBILE Mobile Computing and Communications Review, 2007, 11(4): 11-22.
- [2] HEIDEMANN J, SROJANOVIC M, ZORZI M. Underwater Sensor Networks: Applications, Advances and Challenges [J]. Philosophical Transactions of the Royal Society: Mathematical Physical and Engineering Sciences, 2012, 370(1958): 158-175.
- [3] 吕超, 王硕, 谭民. 水下移动无线传感器网络研究综述[J]. 控制与决策, 2009, 24(6): 801-807.
- [4] WANG Y, LI F, DAHLBERG T A. Energy-efficient Topology Control for Three-dimensional Sensor Networks [J]. International Journal of Sensor Networks, 2008, 4(1/2): 68-78.
- [5] LIU L F, LIU Y. Study of Topology Recovery Algorithm Based on Full Steiner Minimum Tree Problem in Underwater Wireless Sensor Networks [J]. Journal on Communications, 2010, 31(9): 30-35.
- [6] 何明, 梁文辉, 陈秋丽, 等. 水下移动无线传感器网络拓扑愈合与优化 [J]. 控制与决策, 2015, 30(2): 251-255.

(下转第 101 页)

淆层构造的结果,比 16 个仿射等价类代表元作为混淆层同等情况下构造得到的结果要好。但是由于本文方法构造的两轮 SP 结构的分组长度、密钥长度等都只有 16 bit,在实际构造分组密码体制时用现有的计算水平较容易被穷举攻击。寻找合适的 S 盒作为混淆层以及较好的线性层来设计两轮 SPN 结构,将是下一步的研究工作。

### 参考文献

- [1] BIHAM E, SHAMIR A. Differential Cryptanalysis of DES-like Cryptosystems[C]//Proceedings of Advances in Cryptology-CRYPTO'90. Berlin, Germany: Springer-Verlag, 1990:2-21.
- [2] BIHAM E, SHAMIR A. Differential Cryptanalysis of Snefru, Khafre, REDOC-II, LOKI and Lucifer [C]//Proceedings of Advanced in Cryptology-CRYPTO'91. Berlin, Germany: Springer-Verlag, 1991:156-171.
- [3] BIHAM E. On Matsui's Linear Cryptanalysis [C]//Proceedings of Advanced in Cryptology-EUROCRYPT'94. Berlin, Germany: Springer-Verlag, 1994:341-355.
- [4] MATSUI M. Linear Cryptoanalysis Method for DES Cipher[C]//Proceedings of Cryptology-EUROCRYPT'93. Berlin, Germany: Springer-Verlag, 1993:386-397.
- [5] MATSUI M. The First Experimental Cryptanalysis of DES [C]//Proceedings of Advanced in Cryptology-CRYPTO'94. Berlin, Germany: Springer-Verlag, 1994:1-11.
- [6] LEANDER G, POSCHMANN A. On the Classification of 4 Bit S-boxes [C]//Proceedings of WAIFI'07. Berlin, Germany: Springer-Verlag, 2007:159-176.
- [7] SAARINEN M J O. Cryptographic Analysis of all  $4 \times 4$ -bit Sboxes [C]//Proceedings of SAC'11. Berlin, Germany: Springer-Verlag, 2011:118.
- [8] HONG S, LEE S, LIM J, et al. Provable Security Against Differential and Linear Cryptanalysis for the SPN Structure [C]//Proceedings of FSE'00. Berlin, Germany: Springer-Verlag, 2001:273-283.
- [9] PARK S, SUNG S H, LEE S, et al. Improving Upper Bound on the Maximum Differential and the Maximum Linear Hull Probability for SPN Structures and AES [C]//Proceedings of FSE'03. Berlin, Germany: Springer-Verlag, 2003:247-260.
- [10] CANTEAUT A, ROUE J. On the Behaviors of Affine Equivalent Sboxes Regarding Differential and Linear Attacks [C]//Proceedings of Advances in Cryptology-Eurocrypt'15. Berlin, Germany: Springer-Verlag, 2015:45-74.
- [11] KEON D, KIM J, PARK S, et al. New Block Cipher: ARIA [C]//Proceedings of ICISC'03. Berlin, Germany: Springer-Verlag, 2003:432-445.
- [12] ANDERSON R, BIHAM E, KNUDSEN L. Serpent: A Proposal for the Advanced Encryption Standard [C]//Proceedings of AES'98. Berlin, Germany: Springer-Verlag, 1998:1-23.
- [13] DAEMEN J, RIJMEN V. The Design of Rijndael: AES-the Advanced Encryption Standard [M]. [S. l.]: Springer Science & Business Media, 2013.
- [14] 屈龙江, 付绍静, 李超. 密码函数安全性指标的研究进展[J]. 密码学报, 2014, 1(6):578-588.
- [15] 李超, 屈龙江, 周悦. 密码函数的安全性指标分析[M]. 北京: 科学出版社, 2011.

编辑 索书志

(上接第 96 页)

- [7] ERKAY U, FATIH S, KEMAL A, et al. Distributed Connectivity Restoration in Underwater Acoustic Sensor Networks via Depth Adjustment [C]//Proceedings of 2015 IEEE International Conference on Communications. Washington D. C., USA: IEEE Press, 2015:6357-6362.
- [8] JORGIĆ M, STOJIMENOVÉ I, HAUSPIE M, et al. Localized Algorithms for Detection of Critical Nodes and Links for Connectivity in Ad Hoc Networks [C]//Proceedings of Mediterranean Ad Hoc Networking Workshop. Washington D. C., USA: IEEE Press, 2004:12.
- [9] HUANG J, SUN L, WEI X et al. Redundancy Model and Boundary Effects Based Coverage-enhancing Algorithm for 3D Underwater Sensor Networks [J]. International Journal of Distributed Sensor Networks, 2014, 2014(1):1-12.
- [10] 蒋鹏, 陈峰. 基于冗余节点休眠和分阶段唤醒策略的传感器网络三维覆盖控制方法[J]. 电子与信息学报, 2009, 31(12):2807-2812.
- [11] CARRICK D, MAREK D, IULIU V, et al. Autonomous Depth Adjustment for Underwater Sensor Networks: Design and Applications [J]. IEEE/ASME Transactions on Mechatronics, 2012, 17(1):16-24.
- [12] OVALIADIS K, SAVAGE N, KANAKARIS V. Energy Efficiency in Underwater Sensor Networks: A Research Review [J]. Journal of Engineering Science and Technology Review, 2010, 3(1):151-156.
- [13] GOYAL D, CAFFERY J. Partitioning Avoidance in Mobile Ad Hoc Networks Using Network Survivability Concepts [C]//Proceedings of the 7th International Symposium on Computers and Communications. Washington D. C., USA: IEEE Press, 2002:553-558.
- [14] SU R, VENKATESAN R, LI C. An Energy-efficient Asynchronous Wake-up Scheme for Underwater Acoustic Sensor Networks [J]. Wireless Communications and Mobile Computing, 2016, 16(9):1158-1172.
- [15] 李莉, 杨丽娟, 李钢, 等. 基于 Aqua-Sim 的水声传感器网络 MAC 层协议分析[J]. 传感器与微系统, 2015, 34(5):52-54.

编辑 金胡考