

文章编号: 2095-4980(2021)03-0506-05

## WSNs 中基于数据匹配的分布式故障节点检测算法

李 星

(内蒙古师范大学 青年政治学院, 内蒙古 呼和浩特 010000)

**摘 要:** 针对无线传感网络的节点故障问题, 提出一种新的分布式故障节点检测算法(DFDA)。DFDA 算法利用节点度信息估计节点对网络的重要性, 并尽可能将节点度高的节点保存到网络中。通过比较节点间感测的数据, 检测故障节点。为了增强检测的准确性, 采用双重测定策略。仿真结果表明, 相比于同类算法, DFDA 算法提高了检测故障节点的精确度, 并降低了虚警率。

**关键词:** 无线传感网络; 故障节点; 节点度; 数据匹配; 检测

中图分类号: TPT393

文献标志码: A

doi: 10.11805/TKYDA2020259

## Matching-based distributed fault node detection algorithm in WSNs

LI Xing

(Youth College of Political Science, Inner Mongolia Normal University, Hohhot Neimonggol 010000, China)

**Abstract:** A new Distributed Fault Detection Algorithm(DFDA) is proposed to solve the node fault problem in wireless sensor networks. The DFDA algorithm uses the node degree information to estimate the importance of the node to the network, and saves the node with high node degree to the network as much as possible. The fault nodes are detected by comparing the sensing data among nodes. In order to enhance the accuracy of the detection, a double determination strategy is adopted. The simulation results show that the DFDA algorithm improves the accuracy of detecting fault nodes and reduces the false alarm rate compared with similar algorithms.

**Keywords:** Wireless Sensor Networks; fault node; node degree; data matching; detection

无线传感网络(WSNs)是由微型、低成本的传感节点组成的自组织网络<sup>[1]</sup>。传感节点感测监测区域数据, 再将数据传输至控制中心, 进而实现对监测区域内异常事件的管控。

WSNs 网络内的传感节点容易出现故障, 本文将出现故障的节点称为故障节点。导致节点故障主要有 2 个原因: 1) 节点硬件本身的问题; 2) 外界环境的毁坏。通常, WSNs 常部署于野外恶劣环境, 这对传感节点的性能要求提出了苛刻要求。质量不高的传感节点容易在野外环境出现故障。一旦传感节点出现故障, 会降低网络可靠性和 WSNs 服务质量, 失去对监测区域的管控<sup>[2]</sup>。

研究人员常通过错误避免和容错策略解决 WSNs 中节点的故障问题。错误避免策略是通过强化节点的设计过程<sup>[3]</sup>, 利用高质量的元器件材料制造节点, 降低节点发生故障的概率。但该策略增加了部署 WSNs 的成本。

容错策略是使网络能够容忍一定量故障节点的技术, 即网络对一定数量的故障节点具有抵抗性。网络内即使发生一定数量的故障节点, 仍不影响网络的可靠性和 WSNs 服务质量<sup>[4]</sup>。

容错策略是通过引用冗余理念增强网络对故障节点的抵抗性, 并没有将故障节点排除在网络外。该策略并没有彻底地排除这些故障节点对网络性能的影响。

通过检测故障节点、故障节点诊断和网络修复机制<sup>[5]</sup>的故障管理技术, 可以彻底解决因故障节点所带来的问题。目前多数检测机制视节点的重要性雷同, 并没有考虑节点间的差异性。

网络内的各节点在维持网络连通和网络覆盖率方面的贡献并不相同, 可依据节点对网络的贡献将它们划分为重要节点(High-Importance Node, HIN)和不重要节点(Low-Importance Node, LIN)。如果将 HINs 与 WSNs 网络隔离可能极大降低服务质量, 反之, 在网络内排除 LINs 可能并不影响网络的服务质量。

收稿日期: 2020-06-02; 修回日期: 2020-06-22

基金项目: 内蒙古自治区高等教育科学研究“十三五”规划课题资助项目(NGJGH2016201)

作者简介: 李 星(1981-), 男, 硕士, 主要研究方向为计算机网络。email:lixing198112@sina.com

本文提出分布式故障节点检测算法(Distributed Fault Detection Algorithm, DFDA)。DFDA 算法先利用节点的节点度值, 构建邻居节点信息矩阵; 再利用节点间所感测数据的相似性, 构建数据匹配矩阵; 最后, 基于双重测定原则判断节点的状态, 并将故障节点与网络隔离。仿真结果表明, 提出的 DFDA 算法提高了检测故障节点的概率, 并控制了虚警率。

## 1 系统模型

假定  $N$  个传感节点随机分布于  $l \times l$  二维区域内, 这些节点构成节点集  $S = \{s_1, s_2, \dots, s_N\}$ 。所有节点传输半径相同, 即  $\forall s_i, R_i = R, i = 1, 2, \dots, N$ , 其中  $R_i$  表示节点  $s_i$  的传输半径。

故障节点所产生的数据与周围邻居节点所产生的数据可能不相同。因此, 通过比较邻居节点所感测的数据(相似或者相异), 能够判断节点的状态(错误节点或正常节点)<sup>[6]</sup>。令  $X_i$  和  $X_j$  分别表示 2 个邻居节点  $s_i$  和  $s_j$  所感测的数据。如果满足  $|X_i - X_j| \leq \delta$ , 则认为它们没有发生错误, 其中  $\delta$  为阈值, 其值取决于应用类型。

令  $D$  表示  $X_i$  与  $X_j$  间差异, 其是离散—时间函数  $D(t)$ 。如果  $D(t)$  超过  $D_p$ , 则  $X_i$  与  $X_j$  间存在差异, 即不匹配。其中  $D_p$  表示阈值。用  $q(D)$ 、 $P(D)$  分别表示匹配、不匹配时的  $D(t)$  概率分布函数(Probability Distribution Function, PDF)。

## 2 DFDA 算法

### 2.1 邻居节点信息矩阵

在部署了节点后, 每个节点识别它的一跳邻居节点, 并构建一跳邻居节点集。令  $N(s_i)$  表示节点  $s_i$  的一跳邻居节点集。节点依据一跳邻居节点集, 计算自己的节点度。节点度等于节点的一跳邻居节点个数。节点  $s_i$  依式(1)计算自己的节点度  $r_i$ :

$$r_i = |N(s_i)| \quad (1)$$

式中  $|N(s_i)|$  为集  $N(s_i)$  的个数。

为了识别邻居节点, 每个节点广播 Hello 消息, 其包含自己的节点度和自己感测的数据<sup>[7]</sup>。据此, 每个节点构建邻居节点信息矩阵。令  $M_i$  表示节点  $s_i$  的邻居节点信息矩阵, 其定义如式(2)所示:

$$M_i = \begin{bmatrix} s_1 & r_1 & X_1 \\ s_2 & r_2 & X_2 \\ \vdots & \vdots & \vdots \\ s_m & r_m & X_m \end{bmatrix}, \quad s_k \in N(s_i), k = 1, 2, \dots, m \quad (2)$$

式中  $m = |N(s_i)|$ 。矩阵  $M_i$  中第 1 列表示节点 ID 号, 第 2 列对应的元素表示其节点度。

构建了邻居节点信息矩阵后, 节点  $s_i$  再计算它的邻居节点的节点度  $r_i^{\text{avg}}$ :

$$r_i^{\text{avg}} = \left[ \frac{1}{m} \sum_{k=1}^m r_k \right], \quad k = 1, 2, \dots, m \quad (3)$$

式中  $r_k$  为节点  $s_i$  的邻居节点  $s_k$  的节点度。

节点  $s_i$  再利用  $r_i^{\text{avg}}$  将其邻居节点进行二类划分: 节点度高的集  $N_H(s_i)$  和节点度低的集  $N_L(s_i)$ 。为了表述简单, 将在  $N_H(s_i)$  内的节点称为热节点,  $N_L(s_i)$  内的节点称为非热节点。将节点度大于平均节点度的节点纳入  $N_H(s_i)$ , 反之, 将节点纳入  $N_L(s_i)$ , 如式(4)所示:

$$\begin{cases} N_H(s_i) \leftarrow s_k, \text{ if } r_k \geq r_i^{\text{avg}} \\ N_L(s_i) \leftarrow s_k, \text{ if } r_k < r_i^{\text{avg}} \end{cases} \quad (4)$$

### 2.2 数据匹配矩阵

节点获取邻居节点所感测的数据后, 与自己所感测的数据进行比较, 进而判断所感测的数据是否匹配<sup>[8]</sup>。令  $\mathcal{E}_i$  表示节点  $s_i$  与其邻居节点的数据匹配矩阵。 $\mathcal{E}_i$  内的任一元素  $C_{ij}$  的定义如式(5)所示:

$$C_{ij} = \begin{cases} 0, & \text{if } |X_i - X_j| \leq \delta \\ 1, & \text{if } |X_i - X_j| > \delta \end{cases}, \quad 1 \leq j \leq m \quad (5)$$

从式(5)可知, 节点  $s_i$  将自己感测的数据  $X_i$  与邻居节点  $s_j$  所感测的数据进行比较。如果它们的差异小于阈值  $\delta$ , 则为零, 反之为 1。

### 2.3 与热节点间数据的匹配

为了提高对数据匹配的精确度, 节点将自己感测的数据与热节点数据进行多次比较。具体而言, 节点  $s_i$  与热节点  $s_h \in N_H(s_i)$  的数据进行比较(依据式(5)), 并比较  $q$  次。因此, 便可形成  $m \times q$  维矩阵  $\Theta_i$ :

$$\Theta_i = \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1q} \\ C_{21} & C_{22} & \cdots & C_{2q} \\ \vdots & \vdots & \cdots & \vdots \\ C_{I1} & C_{I2} & \cdots & C_{Iq} \end{bmatrix} \quad (6)$$

式中:  $\Theta_i$  表示节点  $s_i$  的数据矩阵;  $I = |N_H(s_i)|$  表示集  $N_H(s_i)$  内的元素的个数。

由于观察了  $q$  次, DFDA 算法能够容忍  $\left\lfloor \frac{q-1}{2} \right\rfloor$  次传输错误。显然, 容错能力随  $q$  值的增加而上升。 $q$  值的增加也增加了通信开销和能量消耗。为此, 本文依据网络环境, 谨慎地选择  $q$  值。

### 2.4 故障节点的检测

DFDA 算法采用双重测定策略检测故障节点, 只有双重测定决议一致, 才进行最终的判决(故障节点或非故障节点)。首先, 依据数据匹配矩阵信息  $\mathcal{E}_i$ , 初步判断节点的状态, 称为一级状态判定。令  $\gamma_i^1$  表示节点  $s_i$  的一级状态判定结果, 如式(7)所示:

$$\gamma_i^1 = \begin{cases} \text{LF}, & \text{if } \|\mathcal{E}_i\| > \sigma \\ \text{LG}, & \text{if } \|\mathcal{E}_i\| \leq \sigma \end{cases} \quad (7)$$

式中:  $\|\mathcal{E}_i\|$  表示矩阵  $\mathcal{E}_i$  所有列元素之和, 即  $\|\mathcal{E}_i\| = \sum_{j=1}^m C_{ij}$ 。将  $\|\mathcal{E}_i\|$  值与阈值进行比较, 如果  $\|\mathcal{E}_i\|$  大于阈值  $\sigma$ , 则将节点  $s_i$  判定为故障嫌疑(Likely Faulty, LF)节点, 否则判定为暂时正常(Likely Good, LG)节点。阈值  $\sigma$  定义为  $\sigma = \mu m$ , 其中  $m$  为节点  $s_i$  的一跳邻居节点数,  $\mu$  为控制参数, 且  $\mu \in (0,1)$ , 可通过节点密度调整  $\mu$  值。

从式(6)可知, 一级状态判定只粗略地考虑了一跳邻居节点所感测的数据信息, 并没有仔细地考虑  $N_H(s_i)$  内节点所感测的数据信息。为此, 进一步利用矩阵  $\Theta_i$  的信息对节点状态进行判定, 即二级状态判定。

计算  $\Theta_i$  内所有元素值之和, 即  $\Psi_i = \sum_{k=1}^I \sum_{j=1}^q C_{kj}$ , 再将其平均值  $\Psi_i / (I \times q)$  与 1 进行比较。如果  $\Psi_i / (I \times q)$  大于 1, 则将节点  $s_i$  判定为 LF, 否则判定为 LG, 如式(8)所示:

$$\gamma_i^2 = \begin{cases} \text{LF}, & \text{if } \Psi_i / (I \times q) > 1 \\ \text{LG}, & \text{if } \Psi_i / (I \times q) < 1 \end{cases} \quad (8)$$

最后, 依据两级判定的结果, 对节点的状态进行裁定, 如式(9)所示:

$$DS = \begin{cases} \text{LF}, & \text{if } \gamma_i^1 = \gamma_i^2 = \text{LF} \\ \text{LG}, & \text{if } \gamma_i^1 = \gamma_i^2 = \text{LG} \\ \text{不确定}, & \text{if } \gamma_i^1 \neq \gamma_i^2 \end{cases} \quad (9)$$

一旦识别了故障节点, 会将这些节点排除网络外。为此, 当检测到故障节点, 就将该节点的重要性降低到零。具体而言, 若节点  $s_i$  为错误节点, 就将节点重要性设置为零, 即  $r_i = 0$ 。

图 1 给出检测节点状态的流程, 该流程以节点  $s_i$  为例。节点  $s_i$  先分别依式(2)构建邻居节点信息矩阵  $M_i$ , 依式(5)构建数据匹配矩阵  $\mathcal{E}_i$  以及依据式(6)计算  $\Theta_i$ 。然后, 再依式(7)和式(8)分别计算  $\gamma_i^1$  和  $\gamma_i^2$  值, 最终通过式(9)判定节点的状态。

### 3 性能分析

#### 3.1 仿真场景

为了更好地分析 DFDA 算法的性能, 利用 MATLAB 软件建立仿真平台。在  $512\text{ m} \times 512\text{ m}$  监测区域内部署 1 024 个节点, 信宿位于监测区域中心。令  $P_n$  表示节点出现故障的概率。仿真参数如表 1 所示。

文献 [9] 提出分布式拜占庭式故障检测算法 (Distributed Byzantine Fault Detection, DBFD); 文献 [10] 提出定位故障检测算法 (Localized Fault Detection Algorithm, LFDA); 文献 [11] 提出基于支持向量机 (Support Vector Machine, SVM) 分类器的故障检测 (Fault Detection through SVM Classifier, FDSC) 算法作为参照, 并与 DFDA 算法进行性能对比分析。主要分析检测准确率 (Detection Accuracy, DA)、虚警率 (False Alarm Rate, FAR), 其定义如式 (10)~式 (11) 所示:

$$DA = \frac{N_{TG} + N_{TF}}{N_{TG} + N_{TF} + N_{NG} + N_{NF}} \quad (10)$$

$$FAR = \frac{N_{FG}}{N_{TF} + N_{FF}} + \frac{N_{FF}}{N_{TF} + N_{FF}} \quad (11)$$

式中:  $N_{TG}$  为正常节点被正确地检测为正常节点的比例;  $N_{TF}$  为故障节点被正确地检测为故障节点的比例;  $N_{FG}$  为正常节点被错误地检测为故障节点的比例;  $N_{FF}$  为故障节点被错误地检测为正常节点的比例。

#### 3.2 检测率及虚警率性能

分析节点的平均邻居节点数对 DFDA 算法检测率和虚警率的影响。为了表述简单, 令  $\bar{n}$  表示节点的平均邻居节点数。

图 2 给出 DFDA 算法的检测率随  $P_n$  的变化情况, 其中  $\bar{n} = 10$ 。从图 2 可知,  $P_n$  值的增加, 使各算法的检测率呈下降趋势。例如, 当  $P_n = 0.05$  时, LFDA 算法的检测率达到 0.99。当  $P_n$  增加到 0.4 时, LFDA 的检测率降至 0.4。与 DBFD, FDSC 和 LFDA 算法相比, 提出的 DFDA 算法的检测率随  $P_n$  的增加而下降, 但下降趋势变缓。即使当  $P_n$  增加到 0.4 时, DFDA 算法的检测率仍达到 0.68。这说明, DFDA 算法能够正确地检测故障节点。

$P_n$  对虚警率 FAR 的影响如图 3 所示。从图 3 可知,  $P_n$  的增加, 使各算法的虚警率上升。与 DBFD, FDSC 和 LFDA 算法相比, DFDA 算法有效地控制虚警率, 并且当  $P_n$  小于 0.25 时, 它的虚警率逼近零。即使当  $P_n$  增加至 0.4 时, 它的虚警率才达到 0.02。

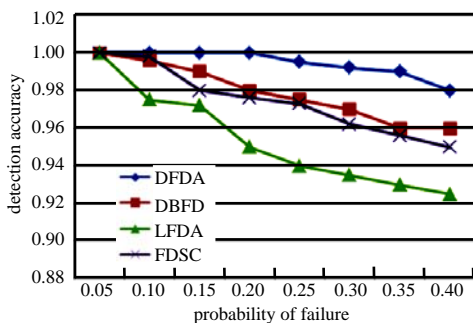


Fig.2 Detection ratio with  $P_n$  ( $\bar{n} = 10$ )  
图 2 检测率随  $P_n$  的变化情况 ( $\bar{n} = 10$ )

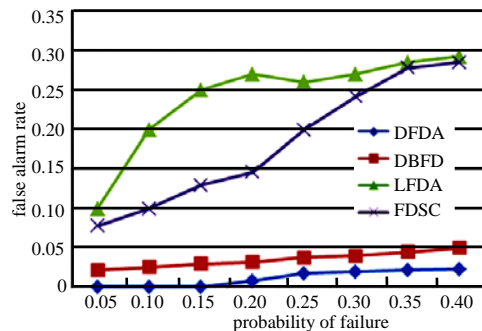


Fig.3 False alarm rate with  $P_n$  ( $\bar{n} = 10$ )  
图 3 虚警率随  $P_n$  的变化情况 ( $\bar{n} = 10$ )

图 4、图 5 分别表示在  $\bar{n} = 25$  的情况下, DBFD, FDSC, LFDA, DFDA 算法的检测率和虚警率。

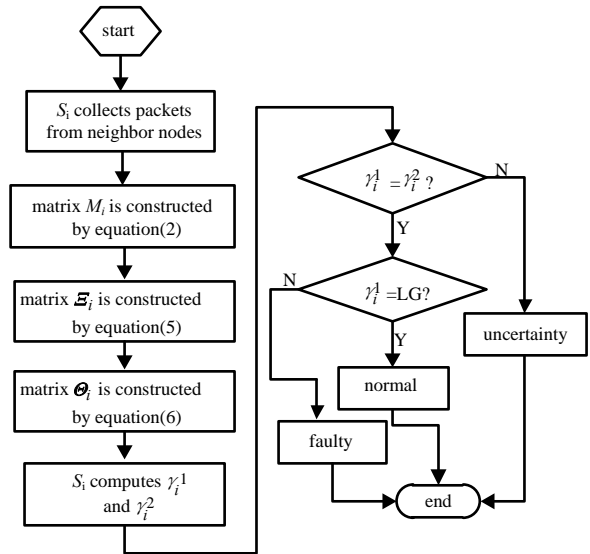


Fig.1 Flow of determining node state  
图 1 判定节点状态流程

parameter	value
simulation area	$512\text{ m} \times 512\text{ m}$
number of nodes	1 024
failure rate of node	$P_n = 0.05 - 0.4$
sensing radius of node	30 m
$q$	20
$\mu$	[0.25, 0.5]

将图 2 与图 4 的数据进行对比可知,  $\bar{n}$  值的增加, 使 DBFD, FDSC 和 LFDA 算法的检测率下降。原因在于:  $\bar{n}$  值越大, 意味着需要检测的节点数量越多, 出现故障节点数量也随之增加, 这就降低了检测率。然而,  $\bar{n}$  值的增加对 DFDA 算法检测率的影响甚少, 其保持高的检测率。

将图 3 与图 5 进行比较发现,  $\bar{n}$  值的增加对虚警率影响并不大。但是,  $\bar{n}$  值的增加降低了 DFDA 算法的虚警率, 特别是当  $P_n$  较大时, 各算法的虚警率也呈增加趋势。但是当  $P_n$  较低时, 例如当  $P_n=0.05$  时, LFDA 算法的虚警率高达 0.19。

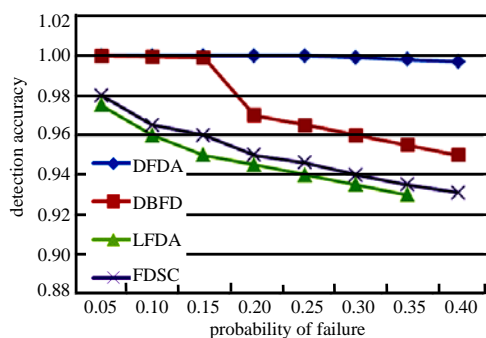


Fig.4 Detection ratio with  $P_n$  ( $\bar{n}=25$ )  
图 4 检测率随  $P_n$  的变化情况 ( $\bar{n}=25$ )

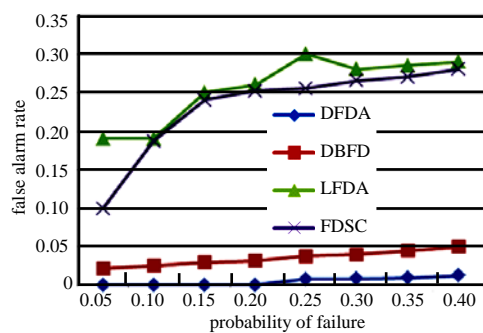


Fig.5 False alarm rate with  $P_n$  ( $\bar{n}=25$ )  
图 5 虚警率随  $P_n$  的变化情况 ( $\bar{n}=25$ )

## 4 结论

本文提出一种新的分布式故障节点检测算法 DFDA。DFDA 算法结合了节点度和节点间感测数据的差异性判断节点状态。仿真结果表明, 提出的 DFDA 算法提高了对故障节点的检测率。后期, 将研究节点故障频率对 DFDA 算法的性能影响。

### 参考文献:

- [1] 尚亚丽. WSNs 中基于能效感知的任播路由[J]. 太赫兹科学与电子信息学报, 2019,17(6):1012-1016. (SHANG Yali. An energy-efficient anycast routing in wireless sensor networks[J]. Journal of Terahertz Science and Electronic Information Technology, 2019,17(6):1012-1016.)
- [2] BABAIE S, KHOSROHOSSEINI A, KHADEM Z A. A new self-diagnosing approach based on petri nets and correlation graphs for fault management in wireless sensor networks[J]. Journal System Architecture, 2013,59(8):582-600.
- [3] CHERAGHLOU M N, BABAIE S, SAMADI M. LRC: novel fault tolerant local re-clustering protocol for wireless sensor network[J]. Journal Computer, 2012,4(8):99-104.
- [4] ZHANG Z, SHU L, MEHMOOD A. A short survey on fault diagnosis in wireless sensor networks[C]// International Wireless Internet Conference, 2016:21-26.
- [5] TANG P, CHOW W S. Wireless sensor-networks conditions monitoring and fault diagnosis using neighborhood hidden conditional random field[J]. IEEE Transactions on Industrial Informatics, 2016,12(3):933-940.
- [6] 邢小东. 基于改进蛙跳算法的无线传感网络节点故障诊断技术研究[J]. 机床与液压, 2019,47(12):97-102. (XING Xiaodong. Research on fault diagnosis technology of wireless sensor network based on improved frog leaping algorithm[J]. Machine Tool and Hydraulics, 2019,47(12):97-102.)
- [7] 管毅, 罗文兴. 一种基于支持向量机的无线传感器故障检测的研究[J]. 科技通报, 2018(4):90-97. (GUAN Yi, LUO Wenxing. Wireless sensor failure detection based on supporting vector machines[J]. Bulletin of Science and Technology, 2018(4):90-97.)
- [8] YAN Jingwen. Spatial error concealment algorithm research[D]. Wuhan, China: Wuhan University, 2018.
- [9] PANDA M, KHILAR P M. Distributed Byzantine fault detection technique in wireless sensor networks based on hypothesis testing[J]. Computers Electrical Engineering, 2015,48(6):270-285.
- [10] CHEN J, KHER S, SOMANI A. Distributed fault detection of wireless sensor networks[C]// Proceedings of the 2006 Workshop on Dependability Issues in Wireless AD HOC Networks and Sensor Network. Los Angeles: IEEE, 2006:65-72.
- [11] ZIDI S, MOULALI T, ALAYA B. Fault detection in wireless sensor networks through SVM classifier[J]. IEEE Sensor Journal, 2018,18(1):340-347.