

# 基于平滑高斯半马尔可夫模型的移动无线传感器网络生成树算法

田杰<sup>1,2</sup>, 王子豪<sup>2</sup>, 魏玉宏<sup>2</sup>

(1. 长安大学 电控学院, 西安 710018; 2. 武警工程大学 通信工程系, 西安 710086)

**摘要:** 针对以往移动无线传感器网络研究只是单纯地对移动群体进行分簇而没有充分利用组群移动的内部稳定性的问题, 结合组群模型中节点运动的规律和内聚性原理, 采用平滑高斯半马尔可夫移动模型刻画组内单个节点移动特征, 构建了一种适合移动网络的稳定生成树算法 (GM-base stable spanning tree algorithm, GSST); 实验证明, 该算法从单个节点运动变化入手, 在预测未来节点运动情况, 选择稳定的链路构建网络结构方面, 提高了移动网络的稳定性; 同时, 利用树的分层特征, 简化移动网络的组网过程, 并实现网络重组局部化; 该算法有效延长节点存活率, 均衡数据传输量。

**关键词:** 移动无线传感器网络; 平滑高斯半马尔可夫移动模型; 内聚性

## A Smooth Gauss-Semi-Markov Model Based Spanning Tree Algorithm for Mobile Wireless Sensor Networks

Tian Jie<sup>1,2</sup>, Wang Zihao<sup>2</sup>, Wei Yuhong<sup>2</sup>

(1. College of Electronic Control, University of Chang'an, Xi'an 710018, China;

2. Department of Information Engineering, Engineering University of PAP, Xi'an 710086, China)

**Abstract:** Aimed at the previous research on wireless sensor nodes in a mobile group, the mobile population were only clustered and didn't make full use of internal stability, combined with the cohesion principle and the law of the nodes' motion in a group mobility model, using a smooth Gauss-Semi-Markov model to describe a single node's moving characteristics within a mobile group, construct a stable spanning tree algorithm for mobile networks (GM-base stable spanning tree algorithm, referred to as GSST). Experiments show that the algorithm, in the light of a single node movement variety, predicts the future movement, selects stable links to construct the network structure, and improves the mobile networks stability. At the same time, which use tree's hierarchical feature, simplify the networking process, and realize the network reorganization localization, which can effectively prolong the nodes' survival rate, balance the amount of data transmission.

**Keywords:** mobile wireless sensor networks; smooth Gauss-semi-Markov mobility model; cohesion

## 0 引言

随着无线传感器网络应用的不断扩大, 单纯的静止网络已经不能满足现代化发展的需求, 移动无线传感器网络以其独特的移动性和可控性的特点, 受到越来越多的关注。移动无线传感器网络是一种特殊的无线传感器网络, 节点的移动性是其突出的特点。在某些特定的应用环境 (例如: 癌细胞跟踪监测、野生动物迁徙、单兵作战系统等) 中, 传感器是需要依附于某些移动物体, 具有移动的盲目性。而且由这些传感器组成的网络, 其拓扑结构随时发生着不可预估的变化, 如果不能合理调整网络结构, 就会导致多次重复组网, 消耗大量能量, 严重影响整个网络生存周期。

针对上述问题, 一些专家学者提出了应用于不同场景的移动模型, 并且依据生物群运动的内聚性<sup>[1]</sup>, 提出了组移动模型的概念。但是, 目前的研究还未涉及组群内部组网方式, 忽略了组群内部节点的连接关系, 导致多次重复分组, 网络稳定性

下降。本文结合平滑高斯半马尔可夫移动模型<sup>[2]</sup>和组移动模型<sup>[3]</sup>, 将向同一目标前进的节点看作一个整体, 只关心群体内部之间的相对速度和距离, 同时采用内聚性原理, 明确网络发展趋势, 大大简化了网络结构, 实现网络重组的局部化, 保证网络的持续稳定性, 有效延长网络生命周期。

## 1 移动模型

现有的移动模型可大致分为实体移动模型和组移动模型。其中, 实体移动模型主要针对单个移动实体的运动规律 (如加减速时速率和角度变化等), 应用于随机无组织移动节点形成的网络。常见的模型有: Random Walk 移动模型、Random Waypoint 移动模型、Random Direction 移动模型<sup>[4]</sup>。该 3 种模型中, 节点的运动方式均为随机的“布朗运动”, 区别在于是否存在停止时间。因此, 这些模型中的节点的速度存在突发性, 不能很好地描述实际节点的运动。而平滑高斯半马尔可夫移动模型则认为节点在加减速的过程中是平滑的, 前一时刻的运动状态会对下一时刻的运动状态产生影响, 没有突发性的改变, 这样的过程符合牛顿定律, 能有效反应节点实际运动规律。

而组移动模型主要针对存在内部组织关系的组群移动规律, 以运动规律的相似度为分组原理, 越相近网络越稳定<sup>[5]</sup>。

收稿日期:2015-11-10; 修回日期:2015-12-07。

基金项目:中央高校基金(2014G1321034)。

作者简介:田杰(1970-),女,陕西西安人,副教授,硕士研究生导师,主要从事无线传感器网络、Ad hoc 网络方向的研究。

徐华等人利用上述特点,提出了基于链路稳定性的传感器网络分簇算法,用已存活时间来预估未来存活时间,将存活时间作为分簇的原则<sup>[6]</sup>。朱珺青等人则是以同一运动轨迹为划分标准,以簇为单位进行数据收集,而后用边缘节点转发<sup>[7]</sup>。

以上的分簇算法只将节点单纯的分组,并没有考虑组群内部密切的组织关系。而本文将两种模型融合在一起,从外部的整体特点入手,简化内部节点运动规律,同时,以存在固定目标为前提,具有明确的指向性,利用树结构的等级性能更好的突出组群内部的运动相关性,大大降低了内部节点之间运动的复杂性。

## 2 算法描述

### 2.1 网络模型

由于存在固定的移动目标,利用内聚性原理,移动节点会以组群的形式向固定目标前进,即节点运动的方向明确并一致。因此,本文只研究节点沿以 sink 节点为起点,以移动目标为终点的方向上的速率(与后文提及的速度相同)。此时,对网络做如下假设:

1) sink 节点位于网络中心,其能量无穷大,能时时与基站保持联系;

2) 各个节点的最大通信半径一致且均为  $R_c$ ,各个节点能根据通信范围自动调节发射频率;

3) 单个节点的移动规律符合平滑高斯半马尔可夫移动模型要求,节点速度  $v_i \in U[v_{\min}, v_{\max}]$  ( $i \in [1, 2, 3, \dots, n]$ ),  $v_i$  表示节点  $i$  的速度,  $U[.,.]$  表示服从  $[.,.]$  均匀分布。

根据网络模型可知,每个节点度符合平滑高斯半马尔可夫移动模型,该模型中节点在加速过程中可以调整加速时隙来控制加速度的范围,即  $a_i \in U[a_{\min}, a_{\max}]$ 。但是 GSST 算法是适用于负载型移动网络组群,各个节点的运动具有较强的自主性,无法精确获得加速时隙、因此引入随机数  $\theta$ ,刻画加速度的变化。

### 2.2 GSST 算法

在网络的初始阶段,所有节点在监测范围内随机选择初始地点  $d_i$ ,并在  $U[v_{\min}, v_{\max}]$  和  $U[a_{\min}, a_{\max}]$  中随意选择初始速度  $v_i$  和初始加速度  $a_i$  (均已知)。为了能清晰地描述 GSST 算法,将具体算法分为初始形成、局部调整两个阶段。

#### 2.2.1 初始形成阶段

以 sink 节点选择子根节点为例说明。sink 向周围节点发送连接请求,收到信息的节点向 sink 节点发送自己的位置和运动信息,即  $M(d_i, v_i, a_i)$ 。当接收到邻居节点的信息后, sink 节点会逐一与自身的位置和运动信息进行比较,并扫描结果。若相对速度和相对加速度均为 0,则表示节点  $i$  与 sink 节点之间是相对静止的,能够构建稳定的链路,即节点  $i$  成为一级子根节点,是 sink 节点(根节点)的孩子节点;若相对速度和相对加速度至少有一方不为 0,则表示节点  $i$  与 sink 节点之间存在相对速度,经过一段时间  $t_s$ ,节点  $i$  会脱离 sink 节点的通信范围,即在  $t_s$  内认为节点  $i$  能与 sink 节点建立稳定通信。

为了增强网络的稳定性,选择出链路连接时间较长的节点作为子根节点(或叶子节点),sink 节点需要预估与邻居节点的链路生存时间,并将结果保存到相应的链路生存表中,依次选择适合的稳定节点作为子根节点。在进行具体分析之前,对

一些常用符号加以说明:

$\Delta a_i$  为当前的相对加速度大小(因为存在固定移动目标,加速度的方向固定并一致,所以不考虑其方向);  $\Delta a_i^*$  为预估下一时刻的加速度;  $\Delta v_i$  为当前的相对速度;  $\Delta v_i^*$  为预估下一时刻的速度;  $\Delta d_i$  为当前的相对距离;  $\theta_i$  为加速度随机数,  $\theta \in [0, 1]$ ;  $\tau_i$  为稳定状态下的高斯随机变量,其是均值为 0,方差为  $\sigma_{v_i}$  且与  $v_i$  无关;  $\delta$  为相关记忆参数,  $\delta \in [0, 1]$ ;  $\Delta \bar{v}_i$  为节点  $i$  从初始到上一时刻的平均速度。

针对相对速度和相对加速度至少有一方不为 0 的情况, sink 节点会预估在下一时刻内节点  $i$  的速度和该链路生存时间,具体分 3 种情况对  $t_s$  进行分析:

1)  $\Delta a_i = 0; \Delta v_i \neq 0$ : 此时,节点  $i$  是相对于 sink 节点以  $\Delta v_i$  的速度脱离 sink 节点的通信范围。由于  $\delta$  表示节点运动的关联性,越接近 1 说明节点运动关联性越强,越接近直线运动,因此,为了能更好地突出组群移动的方向性,  $\delta$  值趋向于 1。具体计算如下:

$$\Delta v_i^* = \delta \Delta v_i + (1 - \delta) \Delta \bar{v}_i + \sqrt{1 - \delta^2} \tau_i \quad (1)$$

$$t_s = \frac{\Delta d_i}{\Delta v_i^*} \quad (2)$$

2)  $\Delta a_i \neq 0; \Delta v_i = 0$ : 此时,节点  $i$  是相对于 sink 节点以初速度为 0,  $\Delta a_i$  为加速度脱离通信范围。具体计算如下:

$$\Delta a_i^* = (1 + \theta_i) \Delta a_i \quad (3)$$

$$t_s = \sqrt{\frac{2\Delta d_i}{\Delta a_i^*}} \quad (4)$$

由于节点  $i$  速度  $v_i \leq v_{\max}$ , sink 节点需计算出在节点  $i$  在脱离通信范围时,是否超出最大速度,而且为了简化计算,假设节点一旦达到最大速度,就会一直保持该速度,不受任何影响。如果没有超出,则直接将得到的链路连接时间存储起来,便于节点选择;如果超出,则需要对  $t_s$  进行如下调整。

$$t_s = \frac{v_{\max}}{\Delta a_i^*} + \frac{R_c - d_i - (\frac{v_{\max}^2}{2\Delta a_i^*})}{v_{\max}} \quad (5)$$

3)  $\Delta a_i \neq 0; \Delta v_i \neq 0$ : 此时,节点  $i$  是以  $\Delta v_i$  为初速度,  $\Delta a_i$  为加速度脱离通信范围。先利用公式(1)和公式(3)分别得出下一时刻的速度和加速度,而后用公式(5)求出  $t_s$ 。

$$t_s = \frac{\sqrt{2\Delta a_i^* (R_c - d_i) + (\Delta v_i^*)^2} - \Delta v_i^*}{\Delta a_i^*} \quad (6)$$

与情况 2 一样,需要讨论节点  $i$  何时达到最大速度,并对链路生存时间进行修正,修正公式如下:

$$t_s = \frac{v_{\max} - v_i^*}{\Delta a_i^*} + \frac{R_c - d_i - [\frac{v_{\max}^2 - (v_i^*)^2}{2\Delta a_i^*}]}{v_{\max}} \quad (7)$$

node  $i$  依据  $t_s$  的大小,选择合适的节点作为子根节点,选择节点的个数由现实情况确定,一般不低于两个。而后, sink 节点将选择结果广播,未被选上的节点等待进入下一轮选择。被选中的子根节点重复 sink 节点选择过程,分别选出自己的子根节点,如果有节点除其根节点外,不存在其它邻居节点,则该节点自动成为叶子节点。这些节点自动成为叶子节点,只负责数据采集和传输,不负责数据的融合。整个过程持续到将所有节点遍历一次为止。

#### 2.2.2 局部调整阶段

虽然在组群内的节点相对距离较小,运动规律相似度高,

但是仍会存在脱离根节点的情况。为了较小范围地调整移动网络的结构, 所有根节点选出最不稳定的叶子节点 (即  $t_s$  最小), 在其脱离时间的基础上提前两个通信时隙作为二次选择叶子节点的时刻。此时, 该根节点会重新进入初始形成阶段, 重新选择合适的子根节点。

因为每个分支上链路稳定时间不同, 所以, 可以保证网络重组的局部性。同时, 整个网络成树状结构, 保证数据传输路径的独立性, 因此, 网络局部调整不会影响整个网络的数据收集。

### 3 算法分析

#### 3.1 可行性分析

为了说明算法的可行性, 只需保证节点空间分布均匀和平均速率稳定。

由于平滑高斯半马尔可夫移动模型具有点空间分布均匀和平均速率时间平稳的特征, 并且本组群内所有的节点都符合该移动模型, 所以, 理论上节点空间分布均匀, 且平均速率稳定。实际中, 移动节点会向着某一固定目标移动, 节点相对集中, 能有利于选取合适的邻居节点。同时, 本算法是分层选取子根节点, 属于单向选择, 有效避免集中节点重复选择的情况。

因此, 无论从理论上还是实际中, 该算法都具有可行性。

#### 3.2 能耗分析

GSST 算法适用于移动无线传感器网络, 其节点能耗符合无线传感器网络的规律。参照文献 [8], 可知节点各个子单元以及数据收发单元下的 4 种工作方式的能耗 (见图 1)。

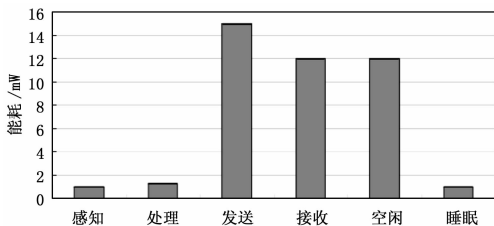


图 1 无线传感器网络能耗示意图

由图 1 可知, 节点能量主要是在数据传输过程中消耗的, 数据处理过程耗能较小。而在网络树形成过程中, 每个节点要进行较为繁琐的计算, 但是其通信次数较少, 因此, 单个节点的能耗较小, 从而整个网络形成所需的能量小。同时, 在 GSST 算法中, 网络初始完成 (即树成形) 后, 由节点移动造成的网络重组可以在局部完成, 进一步减少了整个网络的能耗。

### 4 算法仿真

目前, 针对组群移动的无线传感器网络生成树的研究较少, 而且, 本文主要是基于链路稳定性的, 因此, 为了仿真数据更具说服力, 采用徐华提出的基于链路稳定性的分簇算法 (LSBC)。LSBC 是利用链路已存活时间, 预测该链路未来存活时间, 每间隔时间  $T$ , 会重新进行分簇。

本文采用 MATLAB 仿真, 取 30 个移动节点随机分布在  $1\ 000\ \text{m} \times 800\ \text{m}$  区域内, 固定目标位于该区域右下角 (900, 50), 最大速度为  $0.2\ \text{m/s}$ , 最小速度  $0.1\ \text{m/s}$ , 最大加速度为  $0.05\ \text{m/s}^2$ , 最小加速度为  $0.01\ \text{m/s}^2$ 。每个节点的能量为  $1\ \text{J}$ , 时间间隔  $T$  为  $150\ \text{s}$ , 整体网络运行时间设为  $1\ 000\ \text{s}$ 。仿真结

果图如下:

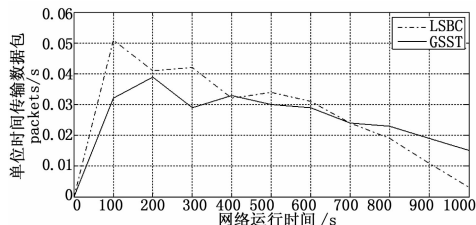


图 2 单位时间传输数据包量比较图

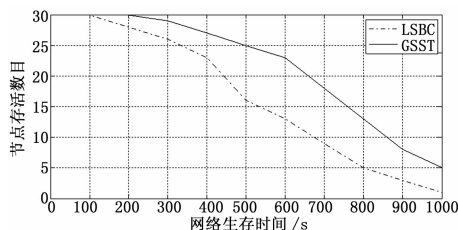


图 3 节点存活数目比较

根据图 2 可得, 在网络运行初期, LSBC 算法需要节点间不断地通信, 以确认节点目前运动变化, 从而得到链路的期望存活时间, 而 GSST 算法在网络形成阶段, 节点间只需通信两次, 并且分层通信。所以, 在初期, LSBC 的数据量大于 GSST。当网络运行在  $100\ \text{s}$  至  $700\ \text{s}$  时, LSBC 算法经历了四次分簇过程, 除了传输正常的的数据以外, 还要附加分簇信息, 因此, 此阶段其数据传输量仍高于 GSST 算法。换言之, 在整个网络运行的前  $700\ \text{s}$  内, LSBC 传输的数据量多于 GSST, 由于节点能量主要消耗在通信模块上, 传输的数据量越大耗能越多, 所以, LSBC 的能耗高于 GSST。当网络运行到最后  $300\ \text{s}$  内, LSBC 能量消耗较大, 导致多数节点死亡 (见图 3), 所以, 数据量低于能耗较小的 GSST。

图 3 是执行两个算法时节点存活数目的对比图。由于 GSST 算法一旦生成树后, 只需局部调整子树结构, 损耗网络能量少, 而且, 在调整过程中, 节点间通信次数和数据量均小于 LSBC 算法, 能有效延长节点生存时间, 增加网络的生命周期。

### 5 总结

GSST 算法是一种结合节点移动模型和组群移动模型的新型生成树算法, 是全面综合的考虑移动无线传感器网络特征, 符合牛顿定律的自组网方式。该算法利用生物内聚性原理, 分层构建子树, 突出组群内部节点的连接关系, 实现网络重组的局部化。通过 MATLAB 仿真, 该算法虽然造成一定的网络时延, 但是对整个网络的数据采集影响不大, 而且明显网络能耗, 提高网络生存周期。但是, 本算法只适用于有固定目标的情况, 对于随机选取目标的移动网络不适用, 需要进一步研究扩宽算法应用范围。

#### 参考文献:

[1] 徐 华, 涂亚庆, 肖 玮. 组移动模型中一种基于种群特性的传感器网络分簇方法 [J]. 传感技术学报, 2009, 22 (4): 543-547.