

移动自组网中基于稳定邻居的数据复制算法^①

常秋菊 杨金民 (湖南大学 软件学院 湖南 长沙 410082)

摘要: 移动自组网中由于节点的移动和能量的有限性,节点间的链路不稳定,使得数据的访问成功率较低,节点间能量消耗不平衡。针对此问题,提出基于节点稳定邻居的复制算法,对节点的数据项访问频度进行加权处理,综合考虑节点的稳定邻居数和剩余能量,以此确定数据副本的放置位置。模拟实验结果表明,该算法能有效提高数据访问成功率,平衡节点间能量消耗。

关键词: 移动自组网;数据复制;稳定邻居;剩余能量;数据访问率

Data Replication Algorithm Based on Stable Neighbors in Mobile Ad Hoc

CHANG Qiu-Ju, YANG Jin-Min (School of Software, Hunan University, Changsha 410082, China)

Abstract: In mobile ad hoc networks, the links among nodes are unstable due to the characteristics of node movement and limited energy, which leads to the poor rate of successful data access and the imbalance in the energy consumption among nodes. The paper discusses the stable neighbors of a node, the access frequency of a node's weight, and the node's surplus energy used in allocating the replicas. Simulation experimental results show that the algorithm improves the rate of successful data access and balances the power consumption among nodes.

Keyword: mobile ad hoc; data replication; stable neighbors; surplus energy; data accessibility

1 引言

移动自组网是一种临时组建的无线移动通信网络。它由一系列自治节点组成,每个移动节点既是客户机,又是路由器。移动自组网不需要固定基础设施的支持,就能够快速部署一个临时网络,满足网络通信的需求。它具有无中心、自组织、组网灵活、抗毁性强等特征,可广泛应用于军事领域、灾难救助和传感器网络等不同领域中。

移动自组网中,节点的移动性导致节点间通信链路不稳定,节点能量耗尽导致节点失效,都会引起网络分割现象。网络分割成多个互不连通的子网后,节点不能访问其它子网中节点存储的重要数据资源,致使访问请求失败。因此,移动自组网中如何提高数据访问成功率是一个重要研究课题。数据复制是解决该问题的有效途径。将数据从一个节点复制到多个节点上后,即使发生网络分割,只要分割后的子网中有数据项的副本,即可保证数据访问成功。

文献[1]提出的 DCG(Dynamic Connectivity based Grouping)指在全局网络拓扑中,基于双连接子图将节点划为不同组,计算组中节点对各个数据项的总访问频度并降序排列。数据副本放置在该数据项访问频度最高的节点上。这样,子网中能够存放更多不同数据项的副本。其中双连接子图是基于网络当前拓扑的静态分组。文献[5]提出的 WEAB(Weighted Expected Access and Battery)指每当节点访问本机未存的新数据时,收集邻居对该节点已存数据的访问频度、已有副本数等信息。对于新数据与该节点已存数据的成功访问次数差值,分别计算该节点和邻居的加权和。选择具有最大正和的数据项,用新数据项替换。这样即保证了节点存储本机经常访问的数据,又使邻居能够在较近的节点中访问所需数据。

上述算法中,没有考虑节点的移动性及节点与邻居间的稳定性。由于连接度高的网络中数据访问更高效[6]。因此,本文提出一种新的复制算法,该复制算

^① 收稿时间:2010-02-21;收到修改稿时间:2010-03-19

法预测节点与邻居在一段持续时间后的连接概率,以节点的稳定邻居集为分组,对数据的访问频度加权处理,在副本放置过程中综合考虑节点的访问频度、稳定邻居数和剩余能量。

2 基于稳定邻居的数据复制算法

移动自组网中,由于节点间的无线链路和节点的移动,节点和邻居不能持续通信。为了获得高访问成功率和能量消耗的平衡,本文中提出基于稳定邻居的副本放置。主要内容包括获取节点的稳定邻居集、数据访问频度的加权和基于稳定邻居的副本放置。

2.1 系统模型

在数据访问的过程中,节点请求某个数据项,若节点本地存储了请求的数据项或数据项副本,则访问成功;若该节点的邻居节点中存储了该数据项或数据项副本,访问成功;若该节点的多跳邻居节点存储了该数据项或数据项副本,访问成功;若节点本地、邻居节点和多跳邻居节点中没有该数据项或数据项副本,则称访问失败。

另外假设:

1) 每个移动节点以预先设定的运动模式自由移动,且有唯一标识,移动节点集合设为 $H=\{H_1, H_2, \dots, H_n\}$ 。每个节点产生一个源数据项,节点中不存放本机源数据项的副本。

2) 数据项集设为 $D=\{D_1, D_2, \dots, D_m\}$ 。每个数据项大小相同。为了算法的简便性,不考虑数据项的更新。

3) 每个移动节点可以存储 C 个数据项,除源数据项外。

4) 每个移动节点有相同的、有限的初始电池能量,记为 E 。

2.2 稳定邻居

节点和邻居之间可以直接通信,完成数据项的访问。因此,节点邻居的稳定性分析对一段持续时间内的成功访问有重要影响。

定义 1. 对节点 i , 满足 $d(i, c) < r$ 且稳定率 $\text{pri}(c) > \alpha$ 的一跳邻居节点 c 称为 i 的稳定邻居。所有稳定邻居节点的集合称为节点 i 的稳定邻居集, 记为 SN_i 。稳定率是预测节点和邻居一段时间间隔后仍在相互通信范围的概率。稳定率的计算公式为

$$\text{pri}(c) = \frac{s(i, r) \cap s(c, v_{\max} \Delta t) + s(c, r) \cap s(i, v_{\max} \Delta t)}{s(i, v_{\max} \Delta t) \cup s(c, v_{\max} \Delta t)}$$

S 表示计算面积, d 表示两个节点的距离, r 表示节点的通信范围, Δt 表示时间间隔, v_{\max} 表示节点的最大移动速度。

每个节点 i 上需要维护邻居稳定率表 $S(i)$ 。表 $S(i)$ 记录节点 i 与邻居间的距离、稳定率。稳定邻居集获取过程描述如下:

第 1 步 节点初始化邻居节点表。

第 2 步 对每个邻居 c , 获取移动信息。计算其与节点 i 的距离, 根据稳定邻居的定义计算稳定率 $\text{pri}(c)$, 记录在表 $S(i)$ 中。

第 3 步 将表 $S(i)$ 中每个邻居的稳定率和 α 进行比较, 判断 c 是否为稳定邻居。若是, 加入节点 i 的稳定邻居集 SN_i 中。

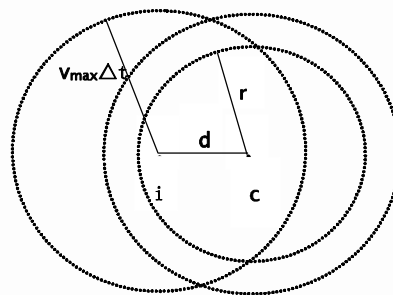


图 1 节点的稳定邻居

2.3 访问频度加权

文中对历史访问频度不清零,与当前访问频度一起加权处理。节点收集的访问频度即加权后的访问频度。每个节点 i 需要维护数据的访问频度表 $P(i)$, $P(i)$ 记录节点对每个数据项的历史访问频度和当前重置间隔的访问频度。

定义 2. 副本的放置算法一定时间间隔后重新执行,间隔的时间称为重置间隔。当前重置间隔之前的访问频度称为历史访问频度。每个重置间隔末,节点将数据项的历史访问频度和当前访问频度加权得到数据项的访问频度 TP_{ij} , 并更新到历史访问频度中。加权函数为

$$\sum_{q=1}^t \frac{P_{ij}}{[q - (-1)^q]^2}$$

P_{ij} 表示节点 H_i 访问数据项 D_j 的频度, 其中 $i < n, j < m$ 。

2.4 基于稳定邻居的数据复制算法

在节点的稳定邻居集中,收集加权后的访问频度,

进行副本的放置,即基于稳定邻居的数据复制算法,简称为 **SNDR**。**SNDR** 中,每个节点先计算自己的稳定邻居数和剩余能量。择优选定一个节点后,在该节点的稳定邻居集中收集并统计数据项的访问频度。对访问频度相似的节点,综合考虑节点的访问频度、稳定邻居数和剩余能量,选择合适的节点放置副本。循环执行算法直到所有节点完成副本放置。

对算法的描述如下:

全部节点集为 H 。最先选取稳定邻居数与剩余能量之积最大的节点,若积相同,优先选取剩余能量较多的节点,若都相同,则选取下标较小的节点。假定为节点 i ,它的稳定邻居集记为 SN_i 。

第1步 节点 i 给所有稳定邻居发送一个访问信息请求包,收集节点对数据项的访问频度。访问信息请求包中包括节点 i 的标识号。稳定邻居收到请求包后,先返回访问信息回复包给节点 i ,再给除 i 外的稳定邻居发送信息,收到信息的节点在稳定邻居集中减去发送节点。访问信息回复包中包括该节点标识号,节点 i 的标识号,该节点对每个数据项的访问频度。

第2步 节点 i 获得稳定邻居的数据访问频度后,计算节点 i 和 SN_i 对各个数据项的总访问频度。按照总访问频度对数据项排序,排列顺序由高到底。数据项按照排序顺序放置,节点 i 和 SN_i 中节点产生的源数据不放置副本。对每个数据项 j ,取两个访问频度最高的节点,假设为 i 和 k ,判断:

1) 若两个节点对数据项的访问频度相似,即 $|TP_{ij}-TP_{kj}| < \beta$,优先考虑稳定邻居数大和剩余能量多的节点。

若节点 i 满足 $TP_{ij} \times d_i \times e_i - TP_{kj} \times d_k \times e_k > \min(e_i, e_k) \times \beta$,有剩余存储空间,将数据项放置在节点 i 上;否则放置到节点 k 。

2) 若不满足条件 1),在 i 和 k 两者中选取访问频度高的节点放置,如果节点上有剩余存储空间;否则放置到另一个节点上。

若节点 i 和 SN_i 的节点完成所有数据项的放置后,还有剩余存储空间,再次执行算法在剩余空间中放置副本,直至节点 i 和 SN_i 中节点空间存满为止。

第3步 从节点集合 H 中删除节点 i 与 SN_i ,循环算法到节点集合 H 为空。

通过上述描述,完成基于稳定邻居的副本放置过程。**SNDR** 算法能够提高数据的访问成功率,因为算

法分析了一段持续时间内邻居的稳定性,以节点的稳定邻居集为分组放置副本,有效减缓了节点移动和网络分割造成的访问失败。**SNDR** 算法能够平衡节点间的能量消耗,因为算法对访问频度相差不大的节点,将副本优先放置在剩余能量多、稳定邻居数大的节点上,不只是依赖访问频度决定如何放置。

3 模拟结果

复制算法中,用数据访问成功率评估数据访问,用节点存活数评估能量消耗。数据访问成功率指成功的数据访问次数与总的访问次数的比值,存活节点数指目前网络中能量没有耗尽的节点数。实验比较分析了算法的数据访问成功率和存活节点数。

文中采用 **NS2** 模拟实验。假定节点在 $1000 \times 1000(m^2)$ 的空间范围中移动,节点先随机选取一个速度和方向运动 **10s**,暂停 **2** 秒后重新选择一个速度和方向继续运动,移动速度范围为 $0 \sim 10m/sec$ 。网络拓扑生成 **100** 个节点,节点集 $H=\{H_1, H_2, \dots, H_{100}\}$;每个节点产生一个数据项,即 H_i 生成数据项 D_i ,数据项集 $D=\{D_1, D_2, \dots, D_{100}\}$;当节点需要访问其它节点中的数据时,它给所有相连的节点发送一个数据请求包,最后选择具有最短路径的节点访问数据。

每个节点的通信范围为 **100m**;节点中可存储 **10** 数据项,数据项的大小相同,设为 **1MB**;节点的访问函数设为 $0.5(0.01i+0.02j)/s$;节点的初始能量为 **5000J**,接收和传送一个请求消耗 **1J** 能量;路由协议采用 **AODV**;重置间隔设为 **125s**。实验中,稳定邻居中的参数 Δt 设为 **30** 秒, α 设为 **0.24**;参数 β 设为 **0.1**。

图 2 为 **SNDR** 算法和 **WEAB(h=1)** 算法的数据访问成功率比较图。如图所示,在模拟初期, **SNDR** 的数据访问成功率在开始一段时间内比 **WEAB** 略高。这是因为模拟刚开始时,网络中节点数多, **SNDR** 以稳定邻居集为分组存储数据项,先放置组中未存的数据项,可以存放更多不同的数据项。随着时间的推移, **SNDR** 的数据访问成功率在一段时间内比 **WEAB** 更高一些。一是因为 **SNDR** 中节点可以在稳定邻居中持续访问成功;二是因为 **SNDR** 放置时优先考虑稳定连接度大的节点,使数据可访问性更高。在模拟结束时, **SNDR** 的数据访问成功率呈下降趋势,这是由于网络中存活节点数减少,稳定邻居数也随之减少。

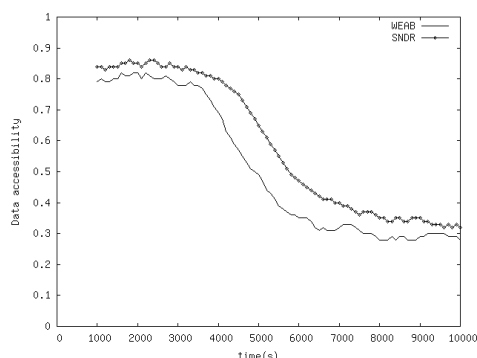


图2 数据访问成功率

图3为SNDR算法和WEAB(h=1)算法的存活节点数比较图。模拟初期,SNDR的存活节点数比WEAB差别不大。这是因为节点的初始能量相同,副本放置在访问频度较高的节点。经过一段模拟时间后,SNDR的存活节点数比WEAB有所增加。一是因为SNDR中节点可以持续访问附近的稳定邻居;二是因为若干个重置间隔后,不同节点间的剩余能量开始出现差异,SNDR算法放置副本时优先考虑剩余能量多的节点。因而平衡了节点间的能量消耗。

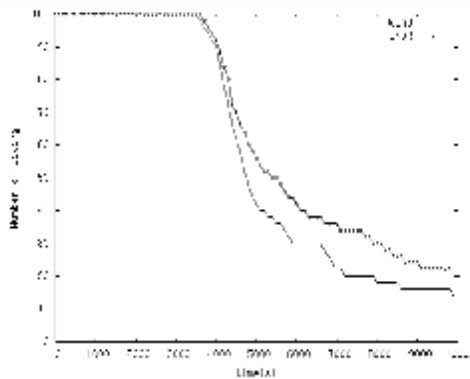


图3 存活节点数

4 结语

移动自组网中,节点的移动性是网络链接不稳定的根本原因。文中分析节点邻居的稳定性,以节点的

稳定邻居集为分组,以分组放置副本,对分组中没有存储的数据项优先放置,有效减缓了网络分割对数据访问的影响,提高了数据可用性。在副本的放置过程中,对访问频度相似节点,优先考虑稳定连接度大、剩余能量多的节点,使节点间能量消耗平衡。

参考文献

- 1 Hara T. Effective replica allocation in ad hoc networks for improving data accessibility. Proc. of the 20th annual joint conference of the IEEE computer and communications societies, April 2001.1568 – 1576.
- 2 Wang K, Li B. Efficient and guaranteed service coverage in partitionable mobile ad-hoc networks. IEEE Joint Conference of Computer and Communication Societies, 2002.
- 3 Ishihara S, Okino T, Watanab T, Mizunot T. Evaluation of the Behavior of Replication of Data Associated with Locations in Ad hoc Networks. Proc. Of SCI2004, 2004.
- 4 Tenzekhti F, Day K, Ould-Khaoua M. Replication algorithms for the World-Wide Web. 16th Annual International Symposium on High Performance Computing Systems and Applications, 2004.
- 5 Shinohara M, Hara T, Nishio S. Replica Allocation Considering Power Consumption in Mobile Ad Hoc Networks. 4th Annual IEEE International Conference on Pervasive Computing and Communications Workshops, 2006.
- 6 Shi K, Haas ZJ. Quantitative Analysis of Partition Statistics and their Impact on Data Replication in MANETs. Sixth International Symposium on Parallel and Distributed Computing, 2007.
- 7 Hara T, Madria SK. Consistency Management Strategies for Data Replication in Mobile Ad Hoc Networks. IEEE Transactions on Mobile Computing, 2009.