

基于多路径蚁群算法的无线传感器网络的路由

任秀丽 梁红伟 汪宇

(辽宁大学信息学院 沈阳 110036)

摘要 针对能量控制在无线传感器网络路由上的特殊要求,为了促使网络节点能量消耗相对均衡,将基本蚁群算法(ACS)应用于无线传感器网络的路由,提出一种基于多路径蚁群算法的无线传感器网络的路由(MACS)。该算法利用蚁群的自组织、自适应和动态寻优能力,通过蚂蚁并行地寻找从源节点到达目的节点的最优路径和次优路径,使得网络中的节点不需要维护全局信息,形成多条传输路径,延长了整个网络的生命期。仿真结果表明,该算法和定向扩散路由(DD)、基本蚁群算法及极大-极小蚁群算法(MMAS)相比,在路由代价和节能方面效果显著。

关键词 无线传感器网络,路由,蚁群算法,多路径

中图分类号 TP393 **文献标识码** A

Multipath Routing of Ant Colony System in Wireless Sensor Networks

REN Xiu-li LIANG Hong-wei WANG Yu

(Information Department, Liaoning University, Shenyang 110036, China)

Abstract Due to the special demand of energy control and balancing the energy of nodes in wireless sensor networks, multipath routing based on ant colony system(MACS) was proposed. The algorithm utilizes the self-organization, self-adaptability and dynamic optimization capabilities of the ant colony to find the optimal routing and suboptimal routing from Sources to the Sink. The experimental results show that the algorithm is more effective and available than DD Routing, ACS and MMAS in routing cost and energy dissipation.

Keywords Wireless sensor network, Routing, Ant colony system, Multipath

无线传感器网络是由许多能量有限、感知能力有限、数据处理能力有限的节点组成的,节点间通过协同工作来完成复杂的任务。但是无线传感器网络面临着节省能耗和延长网络寿命的问题,这就对路由协议有更高的要求。

蚁群算法是一种基于生物习性的启发式算法,用于解决复杂组合优化问题。目前主要有蚁群算法、极大-极小蚁群算法、多蚁群算法等,这些算法在路径规划、网络路由等方面有着广泛的应用。尽管单个蚂蚁本身能力有限,但是通过蚁群的协同工作就可以执行复杂的任务。典型的基于蚁群的路由算法有两类:一类是 Ant-Net 算法^[1],通过前向蚂蚁和后向蚂蚁来实现路由优化,即前向蚂蚁收集节点信息,后向蚂蚁利用这些信息更新路由表;另一类是 ant based control 算法^[2],以概率的方式选择和更新路径,该算法只有一种从源节点出发的蚂蚁,这些蚂蚁到达目的节点后就死亡。这样,当蚂蚁到达节点时,节点的路由表就被更新了。这两种算法对网络的变化有很强的适应能力,能够快速建立优化路径,但是都利用了网络延时信息更新路径,忽略节点当前状态,导致部分节点能量迅速耗尽,引起网络瘫痪。而且,为了优化路径,需要更多的蚂蚁协同工作,导致大量的广播通信而消耗网络节点能量。

本文在蚁群算法的基础上,结合极大-极小蚁群算法、Ant-Net 算法,提出一种基于多路径蚁群算法的无线传感器

网络的路由。该算法并行地寻找最优传输路径,保留次优传输路径,以达到多路径传输的目的,使传感器节点的能量均衡消耗,延长了传感器网络的生命期。

1 蚁群算法

蚁群算法^[3]是人们真实的蚁群觅食行为进行研究后而产生一种自然算法,基本原理可大致描述如下:蚂蚁属于群居昆虫,个体行为极其简单,而群体行为却相当复杂。相互协作的一群蚂蚁很容易找到从蚁穴到食物源的最短路径,而单个蚂蚁则不能。人们通过大量的研究发现,蚂蚁之所以可以做到这一点,是因为蚂蚁个体之间是通过在其所经过的路径上留下一种可称之为信息素的物质来进行信息传递。蚂蚁可以嗅到这种信息素,而且可以根据信息素的浓度来指导自己对前进方向的选择。同时,该信息素会随着时间的推移逐渐挥发掉,所以路径的长短及该路径上通过蚂蚁的多少就对残余信息素的强度产生影响。反过来,信息素的强弱又指导着其它蚂蚁的行动方向。因此,某一路径上走过的蚂蚁越多,后来者选择该路径的概率就越大。这就构成了蚂蚁群体行为表现出的一种信息正反馈现象。蚂蚁个体之间就是通过这种信息交流快速找到食物源或蚁穴的。

将基本蚁群算法应用到无线传感器网络中寻找从源节点

到稿日期:2008-05-23 本文受辽宁省教育厅 2008 年度高等学校科研项目资助。

任秀丽(1965-),女,博士后,教授,硕士生导师,主要研究方向为无线网络与通信,E-mail:rxl@lnu.edu.cn;梁红伟(1983-),男,硕士生,主要研究方向为无线传感器网络;汪宇(1979-),女,硕士生,主要研究方向为无线传感器网络。

到目的节点的最优路径时会出现两个问题：一是寻找路径会陷入局部最优，就是人工蚂蚁搜索到一定程度，会出现停滞现象^[4,5]。人工蚂蚁发现的路径是一致的，不能对路径进一步搜索，从而导致无法找到全局最优路径。二是找到全局最优路径时，数据就会全部沿着最优路径传输，致使此路径上传感器节点能量消耗过快，容易造成节点“死亡”，对于整个传感器网络的生命周期十分不利。

2 多路径蚁群算法的路由选择

2.1 算法设计思想

针对无线传感器网络路由的特性，本文对基本蚁群算法进行改进，赋予蚂蚁新的特性和路径搜索方式。

采用 Ant-Net 算法将蚂蚁分为前向蚂蚁和后向蚂蚁。定义前向蚂蚁具有生命周期，避免在较差路径上搜索，消耗能量。在生命周期内收集节点信息，并在节点间进行局部信息素更新，将信息带给后向蚂蚁。后向蚂蚁按原始路径返回，根据前向蚂蚁的信息对最优路径进行全局更新。蚂蚁具有记忆功能，用以保存经过的传感器节点信息。

本文改进的蚁群算法采用蚂蚁并行搜索路径，即在源节点的 m 只前向蚂蚁同时独立地进行路径搜索，可以加快搜索最优路径的速度。另外，对于本文使用改进算法搜索得到的次优路径，采用一种新的全局信息素规则进行更新。多路径蚁群算法的实现流程如图 1 所示。整个过程如下：

第 1 步 路径搜索初始时，将前向蚂蚁 m 只集中在源节点 S 处。

第 2 步 m 只前向蚂蚁从源节点 S 处出发，各自依据转移概率，从相邻的下一跳节点集合中选择一个节点。

第 3 步 到达中间节点 V_i 时，首先检查自己是否来过该节点。如果来过，则终止蚂蚁。否则，检查该节点 V_i 是否比前一节点 V_j 距离目的节点更近，且距离源节点更远。若是，则更新节点间的信息素浓度(局部更新)。在建立源节点到目的节点路径的过程中，每个被访问的传感器节点标识符等信息由前向蚂蚁携带。

第 4 步 以前向蚂蚁搜索的路径和其他信息为基础，由后向蚂蚁返回源节点时进行路径的全局信息素更新。

第 5 步 设置前后向蚂蚁循环迭代次数，得到最优路径。

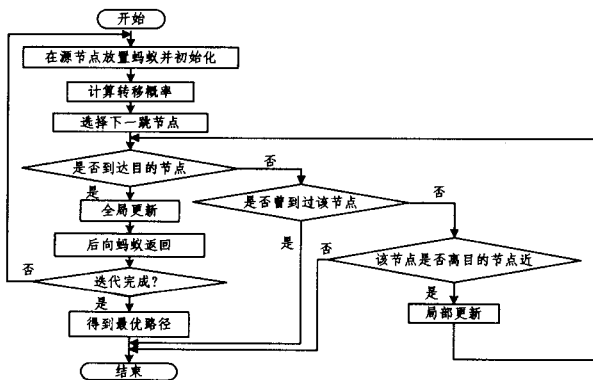


图 1 多路径蚁群算法流程图

2.2 算法的实现

结合基本蚁群算法最优路径^[6]的求解，本文给出了多路径蚁群算法的实现，其具体步骤如下：

Step1 信息量初始化。算法开始运行时，将 m 只前向蚂

蚁均匀地放置于源节点上，并赋予每条路径上相等数量的信息素。令循环次数 $N=0$ ，设置最大循环次数 N_{max} 。

Step2 选择策略。位于节点 J 的前向蚂蚁 $k(k=1, 2, \dots, m)$ 使用如下的状态转移规则选择下一跳节点，选择策略中的参数，如表 1 所列。

$$s_k = \begin{cases} \arg \max_{u \in J_K(r)} \{ [\tau(r, u)] \cdot [\eta(r, u)]^\beta \}, & q \geq q_0 \\ S, & \text{其它} \end{cases} \quad (1)$$

$$P_k(r, s) = \begin{cases} \frac{[\tau(r, s)] \cdot [\eta(r, s)]^\beta}{\sum_{u \in J_K(r)} [\tau(r, u)] \cdot [\eta(r, u)]^\beta}, & s \in J_K(r) \\ 0, & \text{其它} \end{cases} \quad (2)$$

表 1 选择策略中的参数

符号	含义
s_k	序号为 k 的人工蚂蚁所选中的下一跳节点
$J_K(r)$	禁忌表即节点 r 的邻居中未被访问的节点
β	路径长度在选择概率上的作用
η	节点间的距离因子
τ	两节点间支路上的信息量
r, s, u	节点标志符
P_k	前向蚂蚁 k 由一个节点转移到另一节点的概率

q 为一个随机变量，在 $(0, 1)$ 中均匀分布，由此增加了所得路径的数目，在一定程度上削弱了蚁群陷入局部最优的趋势。 q_0 是设定的阈值，参数 q_0 的大小决定了利用先验知识与探索新路径之间的相对重要性。前向蚂蚁在选择下一跳节点前先获得一个随机值 q 。若 $q \leq q_0$ ，则选择节点时按已知信息，选择不在禁忌表 $J_k(r)$ 中而且令表达式 $[\tau(r, u)] \cdot [\eta(r, u)]^\beta$ 值是最大的节点；若 $q > q_0$ ，则按照随机方法进行。S 称为搜索，其概率分布由式(2)计算得到。

Step3 局部更新信息量。在每只前向蚂蚁的寻路过程中，每次位置移动都要对该段支路上的信息量进行修正。采用如下的局部更新规则，局部更新规则中的参数如表 2 所列。

$$\tau(r, s) \leftarrow (1 - \rho) * \tau(r, s) + \rho * \tau_0 \quad (3)$$

$$\tau_0 = (n * L_m)^{-1} \quad (4)$$

表 2 局部更新规则中的参数说明

符号	含义
τ_0	支路上的信息增量
ρ	局部搜索过程的信息素挥发程度
L_{mn}	由启发式近邻算法得到最优路径

τ_0 是一个与问题本身相关联的常量，由经验式(4)确定。前向蚂蚁随着搜索时间加长而接近其生命周期，每经过一个节点，年龄也随之增长。如果前向蚂蚁没有到达目的节点，且未接近生命周期，则跳转到 Step2；否则结束。

Step4 全局更新信息量。在所有的前向蚂蚁完成一次搜索之后，将信息带给后向蚂蚁，后向蚂蚁按原始路径返回，根据前向蚂蚁到达目的节点时的信息进行全局信息素更新。对于蚁群算法容易出现的停滞现象，本文提出的算法采用了极大-极小蚁群算法中将各条路径上的信息素浓度限制在 $[\tau_{min}, \tau_{max}]$ ， τ_{min} 可以有效地避免算法停滞； τ_{max} 避免某条路径上的信息素远大于其它路径，使所有的蚂蚁都集中到同一条路径上面，限制算法的扩散。全局更新规则中的参数如表 3 所列。全局更新规则描述如下：

$$\tau(r, s) \leftarrow (1 - \alpha) * \tau(r, s) + \alpha * \Delta\tau(r, s) \quad (5)$$

$$\Delta\tau(r,s) = \frac{1}{L_{gb}} \quad (6)$$

其中 $\alpha \in (0,1)$ 。本文改进的蚁群算法中,利用式(7)对搜索到的次优路径上的信息素进行全局更新:

$$\tau(r,s) \leftarrow (1-\alpha) * \tau(r,s) + \epsilon * \frac{P_{suboptimal}}{P_{optimal}} \quad (7)$$

采用式(7)可以有效地抑制由于最优路径与次优路径上的信息素之间的差距加剧而引起的停滞现象。在搜索到最优路径的同时,保留次优路径,形成数据的多路径传输,使得传感器网络中各节点的能量均衡地消耗,从而延长网络生存周期。

Step5 检查循环迭代终止条件。在设置的循环迭代次数 N_{max} 内,若有 90% 以上的前向蚂蚁选择的最优路径相同,当 $N_c \geq N_{max}$,循环结束,否则 $N_c = N_c + 1$,跳到 Step1。

表 3 全局更新规则中的参数说明

符号	含义
$\Delta\tau(r,s)$	代表支路上信息增量
α	表示全局搜索过程的信息挥发程度
L_{gb}	代表当前的全局最优路径
ϵ	算法中引入的一个参数
$P_{optimal}$	当前搜索中最优路径长度
$P_{suboptimal}$	当前搜索中次优路径长度

3 仿真结果与分析

为了测试本文提出的多路径蚁群算法应用到无线传感器网络中的效果,利用 NS-2 网络仿真器将其性能与定向扩散路由算法、基本蚁群算法和极大-极小蚁群算法进行了比较。场景配置如下:节点均匀分布在 $200m \times 200m$ 的区域中,节点传输最大距离为 30m,节点初始能量为 1000J,令网络中传输的所有(包括探测蚂蚁、兴趣消息)数据帧的大小为 64Byte,仿真总时间为 500s,3 种蚁群算法中参数的设定均为 $\alpha=0.9$, $\beta=2$, $q_0=0.45$, $\rho=0.95$, $m=30$ 。仿真次数为 20 次。网络节点数分别为 50,100,200,300 和 400,实验中根据网络规模缩放网络场景面积以保持节点密度不变。

定向扩散的能量模型,空闲功耗 35mW,接收功耗 395mW,发送功耗 660mW。所有节点都是静止的,一个汇聚点放置在网络场景的右下角,目的节点随机分布于网络场景的左上角。网络流量采用源节点产生 2 事件/s,探测数据每隔 60s 发一次,兴趣每隔 30s 发一次。

性能评价指标主要是传感器网络总的能量消耗和事件平均传输延时。传感器网络总的能量消耗定义为整个网络的传感器节点传输数据量的能量消耗,该指标表明算法的路由代价,反映了总能耗越小,性能越好,网络寿命越长。事件平均传输延时定义为传输有效数据所用的时间与所传输的数据量之比,反映了数据传输路径的质量。

图 2 反映了在 4 种不同的算法下网络总的能量消耗随节点数目增多的变化关系。从图中可以看到,MACS 的能量消耗要小于 DD,ACS 和 MMAS 的能量消耗,因为在开始阶段,都要进行消息的扩散传播,但是由于 DD 中所有的节点参与兴趣和探测数据的泛洪,随着时间的增加,消耗更多的能量,ACS 和 MMAS 利用网络延时信息更新路径,以及为了优化路径需要较多的蚂蚁协同工作,这样将消耗过多的传感器节

点能量,而本文提出的 MACS 由于维持多路径和蚂蚁动态寻优的特性,避免了周期性的蚂蚁扩散,节省了网络的能量消耗。

当网络节点数目为 200 时,测试 4 种算法的平均传输延时。图 3 反映了 4 种算法网络传输有效数据的平均传输延时随时间的变化关系。由图可知,MACS 的平均传输延时要小于 DD,ACS 和 MMAS 的平均传输延迟,主要是因为 DD 中,最优路径的建立是通过泛洪方式,需要的时间长,从而增加了网络延时。ACS 和 MMAS 中,它们都利用了网络延时信息更新路径,从而减缓了找到最优路径的速度,因此平均传输延时比较大。而本文提出的 MACS,具有动态寻优特性,加快了算法的收敛速度,在较短的时间内找到最优路径,减少了网络延时。

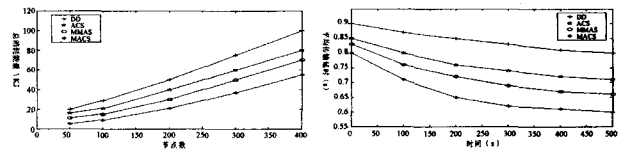


图 2 总能量消耗

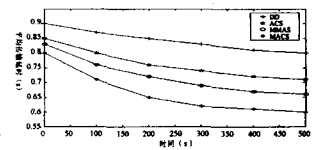


图 3 平均传输延时

通过 MACS 与 DD,ACS 和 MMAS 在总能量消耗和平均传输延时两个方面的比较,可以看到 MACS 在路由质量上优于另外 3 种算法,并且由于维持多路径和蚂蚁动态寻优的特性,避免了周期性的蚂蚁扩散,换取路由代价的降低,达到了节能的目的。

结束语 蚁群算法作为一种新的生物进化算法,具有分布计算、信息正反馈和启发式搜索等特点。本文对蚁群算法的改进避免了原蚁群算法陷入局部最优解,产生过早收敛的问题。同时将其应用在无线传感器网络中进行路由选择,很好地权衡了节点的能量消耗,对于提高无线传感器网络的网络效率,延长网络的生存周期具有很高的应用价值。

参考文献

- [1] Kassabaldsi, El-Sharkaw I M A, Marks R J. Swarm intelligence for routing in communication networks[J]. Global Telecommunications, 2001, 6(6): 3613-3617
- [2] Schoonderwoerd R, Holland O, Brut EN J, et al. Ants for load balancing in telecommunication networks[R]. Bristol Hewlett Packard Lab, 1996
- [3] Gunes M, Sorges U, Bouazizi. IARA-the-ant-colony based routing algorithm for MANETs[C]// International Conference on Parallel Processing Workshops (ICPPW' 02). 2002: 79-85
- [4] Baras J S, Mehta H A. Probabilistic emergent routing algorithm for mobile ad hoc networks[C]// Wiopt 03: Mobilizing and Optimization in Mobile, Ad Hoc and Wireless Network. Sophia-Antipolice, France, March 2003: 20-125
- [5] Shnyder V, Hempstead M, Chen B. Simulating the power consumption of large-scale sensor network application[C]// Proc. of SEN-SYS'04. Baltimore, 2004: 188-200
- [6] Dorigo M, Gambardella L M. Ant Colony System: Cooperative Learning Approach to the Traveling Salesman Problem[J]. IEEE Transaction Evolutionary Computation, 1997, 1(1): 53-66