

无线传感器网络蚁群仿生路由的设计与仿真研究<sup>\*</sup>)

张石 张哲 朱吉昌

(东北大学信息科学与工程学院 沈阳 110004)

**摘要** 本文设计了一种基于蚁群算法的无线传感器网络路由协议。该协议利用了蚁群算法的高效寻径能力,实现了网络节点的高效、节能路由。同时,本文还对该协议进一步给出了分步式实现方案,并对其进行了深入的仿真测试。

**关键词** 无线传感器网络, 蚁群算法, 路由, 网络仿真

## Design and Simulation of an Ant Routing Protocol in Wireless Sensor Networks

ZHANG Shi ZHANG Zhe Zhu Ji-chang

(Information and Engineering Institution, Northeastern University, Shenyang 110004, China)

**Abstract** A new wireless sensor networks routing protocol based on ant-system bionics algorithms is proposed in this paper. The protocol is associated with the high-efficiency path searching ability of ant-system, and achieves a fast, low-energy routing. At the same time, the distributed implement method of the routing is also presented in this paper. At last, a particular simulation and analysis of the routing is done to show its performance.

**Keywords** Wireless sensor networks, Ant-system algorithms, Routing, Network simulation

针对不同的传感器网络应用,研究人员提出了不同的路由协议。但到目前为止,仍缺乏一个具有通用性,并满足无线传感器网络中能量高效、可扩展、鲁棒性好且快速收敛的路由协议规范。本文提出了一个基于蚁群算法的智能无线传感器网络路由协议,该协议具有节能、快速等特点,同时该协议有一个通用性较强的主体框架,可以方便地针对特定的应用进行扩充。

## 1 蚁群节能路由算法设计

为了简化无线路由问题,可以将整个网络看成一个无向加权连通图,用  $G(N, E, C)$  来表示,其中  $N$  表示网络节点集合,  $E$  表示网络链路集合(假定在该类网络中一对节点之间最多只有一条链路),  $C$  表示各链路上的代价集合。举例来说,链路  $e \in E$  的代价函数为  $C(e)$ ,  $C \in R^+$ 。在传感器网络中,典型的代价函数为能量消耗。依据以上简化模型,无线传感器网络中节能路由问题的实质就是在  $G$  中基于能耗代价寻找连接源节点 Sink 和目的节点集  $D$  ( $Sink \notin D$ ) 的最小连接树。

另外,在无线传感器网络路由协议设计中,能量负载在网络中的均衡性也是一个重要的问题。为了防止数据流量集中于能耗最低的路径上,我们在寻找能耗代价的最小连接树时,也要将能量负载均衡性考虑到其中。

总之,无线传感器网络仿生路由算法设计的任务就是利用仿生算法生成各个节点到 Sink 节点的最小能耗数据通路,同时,也尽可能地整个网络的能量负载均衡化。

蚁群算法是一种模拟进化算法<sup>[1]</sup>,其本质上是一个复杂的智能系统。它具有较强的鲁棒性、优良的分布式计算机制、易于与其他方法结合等优点。蚁群算法模拟真实蚂蚁的觅食寻径过程,利用蚂蚁在所经路径上留下信息素的变化来选择路径。本文基于蚁群算法设计了一种无线传感器网络节能路

由协议,在该协议中假定每个传感器节点上有  $n$  只蚂蚁,它们分别出发寻找 Sink 节点,并在寻径过程中根据路由的能耗对路径上的信息素进行局部和全局更新,以达到算法进化的目的。在蚁群算法中,状态转移规则、全局和局部信息素更新规则详细设计如下。

## 1.1 状态转移规则

在节点  $r$  上的第  $i$  只蚂蚁依据下述规则来选择节点  $s$ : 如果  $q \leq q_0$ ,

$$\rho_i(r, s) = \begin{cases} 1 & \max(PH_i(r, s)), s \in Nb_i(r) \\ 0 & \text{否则} \end{cases} \quad (1)$$

否则,

$$\rho_i(r, s) = \begin{cases} \frac{PH_i(r, s)}{\sum_{u \in Nb_i(r)} PH_i(r, u)}, & s \in Nb_i(r) \\ 0 & \text{否则} \end{cases} \quad (2)$$

其中,  $q$  是在  $[0, 1]$  之间均匀分布的随机数,  $q_0$  是满足  $0 \leq q_0 \leq 1$  的常数,  $\rho_i(r, s)$  代表在节点  $r$  的第  $i$  只蚂蚁选择下一跳节点  $s$  的概率,  $PH_i(r, s)$  是第  $i$  只蚂蚁释放在节点  $r$  和  $s$  间的链路上的信息素总量,  $Nb_i(r)$  是与节点  $r$  相邻且未被该蚂蚁在此次路由过程中经过的节点集合。采用此定义来实现蚂蚁状态的转移可保证寻找优化路径时避免陷入局部优化。

## 1.2 信息素的局部更新规则

对第  $i$  只蚂蚁,如果节点  $r, s$  是它选择的路径上的两个相邻节点,则信息素  $PH_i(r, s)$  用公式(3)来调节,否则不调节。

$$PH_i(r, s) \leftarrow (1 - \alpha_0) \cdot PH_i(r, s) + \Delta PH \quad (3)$$

其中,  $\alpha_0$  表示信息素局部更新挥发系数 ( $0 < \alpha_0 < 1$ );  $\Delta PH$  为常数,表示第  $i$  只蚂蚁本次寻路过程中残留在  $(r, s)$  上的信息量。信息素局部更新的目的在于避免后来的蚂蚁只在当前所走过的优化路径附近寻路。

## 1.3 信息素的全局更新规则

对第  $i$  只蚂蚁,如果节点  $r, s$  是它选择的路径上的两个相

<sup>\*</sup>) 基金项目: 国家自然科学基金(60434030); 国家自然科学基金(60374072)。张石 教授, 博士, 主要从事智能网络、信号处理研究。

邻节点,则信息素  $PH_i(r,s)$  按公式(4)来调节,否则按公式(5)来调节。

$$PH_i(r,s) \leftarrow (1-\alpha_1) \cdot PH_i(r,s) + F \quad (4)$$

$$PH_i(r,s) \leftarrow (1-\alpha_1) \cdot PH_i(r,s) \quad (5)$$

其中,  $\alpha_1$  表示信息素全局更新挥发系数( $0 < \alpha_1 < 1$ )。此规则的目的是搜寻全局最优解。式中,限制函数  $F$  定义如下:

$$F = M_{EC} - EC_i \quad (6)$$

其中,  $M_{EC}$  为常数,其值为蚂蚁路由费用的最大值;  $EC_i$  为蚂蚁  $i$  寻径的总费用,其值可通过下式计算获得,

$$EC_i = \sum_{r=1}^v \sum_{s=1}^v C_{rs} P_{rs}^i \quad (7)$$

其中,  $P_{rs}^i = \begin{cases} 1 & (r,s) \text{ 是蚂蚁 } i \text{ 所选路由的边} \\ 0 & \text{否则} \end{cases}$ ,  $v$  是网络中节点的数目。边  $(r,s)$  上的费用计算公式如下:

$$C_{rs} = e_{rs} / R_s \quad (8)$$

这里,  $e_{rs}$  表示节点  $r$  和  $s$  直接通信的能量消耗,  $R_s$  表示节点  $s$  的剩余能量(与节点初始能量相比后的归一化的值)。这个度量标准综合考虑了节点的能量消耗以及其剩余能量。通过上述定义可见,如果所选路由的总费用最小,那么最优蚂蚁进行全局信息素更新后所选路由各链路上的信息素应该增加更多。

#### 1.4 蚁群节能路由的算法步骤

设全网共有  $v$  个节点,在每个节点上放置  $n$  只蚂蚁,蚁群算法迭代次数为  $N_c$ 。基于以上三点规则,具体的蚁群簇间路由建立过程如下:

(1) 初始化网络拓扑中各边的相应信息素  $PH_0$ , 初始信息素的计算公式如下:

$$PH_0(r,s) = \eta / d_{rs} \quad (9)$$

其中,  $\eta$  为常数,  $d_{rs}$  为边  $(r,s)$  两端点中到 Sink 节点较近端到 Sink 的距离;

(2) 从目标节点放出  $n$  只蚂蚁去寻找 Sink 节点。每只蚂蚁通过重复应用状态转移规则来选择各自的路径。当一只蚂蚁成功地完成由目标节点到 Sink 节点的路由选择后,这只蚂蚁将依据局部更新规则对所选择路由上各路径的信息素进行局部更新;

(3) 对所有的蚂蚁重复(2)直到  $n$  只蚂蚁都完成了寻路;

(4) 记录全局最优蚂蚁(即由目标节点到 Sink 节点路由能耗最低的蚂蚁),然后使用全局更新规则对全网各路径上的信息素进行全局更新;

(5) 对所有的目标簇首节点重复(2)-(4),直到完成了  $v$  个节点到 Sink 节点的路由;

(6) 重复(2)-(5)  $N_c$  次,以获得全局最优解。

## 2 蚁群节能路由的模拟实现

蚁群节点路由可以集中式或分布式实现。所谓集中式实现就是网络内的所有节点把其位置信息、能量信息通过泛洪的方式汇聚到 Sink 节点,由 Sink 节点进行集中的最优路径选择,然后,再由 Sink 节点发起泛洪,把计算出的全网的最优路由表发送到网内的每个节点;分布式实现就是指每个节点只保存其邻居节点的位置和能量信息,同时只记录其上一跳和下一跳节点的 ID 为路由表,在蚁群寻路时由每个节点分别计算出其最优的下一跳,进而形成一条完整的最优路由。本文中采用分布式方法对蚁群路由进行实现。相对于集中式实现,分布式实现的优点如下:

(1) 集中式蚁群路由要进行海量的数据泛洪,这样做在网络内会引起“内爆”和“重叠”,Sink 节点获取到的网内节点数据会出现缺失或出错;分布式蚁群路由节点只需获取其邻居节点的数据且无需转发,减小了网络间的数据量;

(2) 集中式蚁群路由要求每个节点保存全网的路由表;分布式蚁群路由中节点只需保存其上一跳及下一跳节点的信息,符合无线传感器网络节点存储空间有限这一特点;

(3) 集中式蚁群路由的最优路由是由 Sink 节点集中式计算完成的,这对 Sink 节点的计算能力及相应的能量储备提出了要求;分布式蚁群路由的寻路过程通过每个节点的共同计算完成,对计算能力、能量储备的要求很低,同时,也在寻路效率上获得了提升。

### 2.1 协议分组包头的定义

分布式实现的蚁群路由的通信分组头定义如下。

(1) Hello 分组头及邻居节点列表

Hello 分组头是蚁群节能路由由协议过程中发送的第一种包头,该包头在协议中用来进行邻居节点的发现。该分组头的数据结构中包括分组头的类型标识、节点的 ID、节点的相对位置和节点当前的能量情况,如表 1 所示。

表 1 Hello 分组头的数据结构

Hello 分组头			
分组头类型	节点 ID	节点坐标	节点能量
(8 bits)	(32 bits)	(96 bits)	(8 bits)

当基于蚁群节能路由的无线传感器网络系统的路由发起过程开始后,每个节点以一个随机的时间延迟发送携带 Hello 分组头的数据包。当节点  $r$  收到节点  $s$  的 Hello 数据包时,表明节点  $r$  在节点  $s$  的单跳通信范围之内,即节点  $s$  为节点  $r$  的邻居节点。蚁群节能路由规定每个节点保存一个邻居节点列表,该列表以双向链表的形式实现,数据域中包括邻居节点的 ID、邻居节点的位置、邻居节点的能量情况及当前节点到邻居节点路径上的信息量值,如表 2 所示。

表 2 邻居节点列表的数据结构

邻居节点列表的数据域			
节点 ID	节点坐标	节点能量	信息量
(32 bits)	(96 bits)	(8 bits)	(32 bits)
0x000000A0	(32,57,0)	50%	0.9
...	...	...	...

节点  $r$  收到来自  $s$  节点的 Hello 数据包后,首先在其邻居节点列表中查找  $s$  节点的 ID,如果不存在该节点 ID 的表项,则依次把 Hello 分组头中对应的内容填入邻居节点表项的数据结构中;如果该节点的表项已经存在,则更新该表项的能量数据域。

为了确保节点能可靠地获取其邻居节点信息,以备最优路由选择,每个节点在每次路由建立过程的开始阶段,会以随机的时间间隔,发送两次 Hello 含分组头的数据包。经仿真数据分析,由于在此阶段,网络内的数据负荷比较轻,这样的机制已能确保邻居信息的获取正确率接近 100%。

(2) Ant 分组头及路由表

Ant 分组头是蚁群节能路由中为查找最优路径寻找定义的一种分组头。当网内邻居节点查找过程结束之后,节点开始进行蚁群选路,发送包含 Ant 分组头的数据包。Ant 分组头的数据结构中包括分组头的类型标识、源节点 ID、蚂蚁寻

路包的序列号及蚂蚁所走路径上的代价总量,如表 3 所示。

表 3 Ant 分组头的数据结构

Ant 分组头			
分组头类型	源节点 ID	序列号	所走路径上的代
(8 bits)	(32 bits)	(8 bits)	价总量(32 bits)

当网络进入到蚁群寻路阶段后,每个节点分别按一定的时间延迟发送含 Ant 分组头的寻路数据包,此延迟时间与该节点到 Sink 节点的距离成反比(这样定义的好处是当距离 Sink 节点较远的节点找到其到 Sink 的最优路径时,该路径上其他节点到 Sink 节点的最优路径也相应确定了,这样其路径上的节点就无需再发送自己的 Ant 寻路数据包,可以减少网络上的数据负荷)。在每次寻路过程中,每个节点要发送四次蚂蚁寻路数据包,每次发送时,数据包的序列号相应加 1。当节点  $r$  收到源节点  $s$  发出(或转发)的序列号为  $i$  的 Ant 寻路数据包后,根据源节点的 ID 号和序列号进行路由表的查找,蚁群节能路由协议的路由表也是一个双向链表,它的数据域包括源节点的 ID 号、序列号、上一跳节点的 ID 号及下一跳节点的 ID 号,如表 4 所示。当路由表中没有从源节点  $s$  发出的第  $i$  号路由表项时,添加该表项,并从网络模拟器为每个数据包添加的 IP 头中获取发送(或转发)该数据包的节点 ID 号,同时,根据式(1)和式(2)中的蚁群算法状态转移规则选择要转发的下一跳节点,填入路由表的相应项中;如果路由表中已经存在从源节点  $s$  发出的第  $i$  号路由表项时,说明蚂蚁所走路径中出现环路,这时丢弃该数据包。除非 Ant 寻路数据包被丢弃,否则该数据包将一直根据状态转移规则转发,在每次转发过程中,Ant 分组头内的路径代价总量数据项将被逐次累加,直到该数据包到达 Sink 节点。

表 4 Ant 节能路由节点路由表

Ant 路由表			
源节点 ID	序列号	上跳节点 ID	下跳节点 ID
(32 bits)	(8 bits)	(32 bits)	(32 bits)
0x000000C0	1	0x0000000B	0x0000000D
...	...	...	...

(3) 全局更新分组头及 Sink 蚁群路由数据表

全局更新分组头是蚁群节能路由中为进行最优路径上的信息素更新而定义的。其数据结构中包括分组头的类型标识、发出蚁群寻路数据包的源节点 ID、数据包序列号及信息素更新值,如表 5 所示。同时,蚁群节能路由规定在 Sink 节点上建立一个蚁群路由数据表。数据表是以数组和双向链表混合实现的,其中数组的数据项包括收到的蚁群数据包的源节点 ID、蚁群数据包的序列号、上跳节点的 ID 及蚁群数据包所走路径的总代价值。由于在每一轮寻路过程中,每个节点要发送 4 次蚂蚁寻路数据包,因此蚁群路由数据表是由上述 4 个数组元素为单元构成的双向链表,其中每个单元用来存储来自同一个源节点的 4 次寻路数据包内容,如表 5 所示。

表 5 全局更新分组头的数据结构

全局更新分组头			
分组头类型	源节点 ID	序列号	信息素更新值
(8 bits)	(32 bits)	(8 bits)	(32 bits)

当 Sink 节点收到来自  $s$  节点的第  $i$  号蚂蚁寻路数据包

后,分别将  $s$  节点的 ID、序列号  $i$  和 Ant 分组头内的记录的总代价值存入蚁群路由数据表,并从 IP 头中获取发送(转发)该数据包的上一跳节点 ID,存入数据表,如表 6 所示。

表 6 蚁群路由数据表

蚁群路由数据表			
源节点 ID	序列号	上跳节点 ID	路径的总代价值
(32 bits)	(8 bits)	(32 bits)	值(32 bits)
0x000000C0	1	0x0000000B	15.5
0x000000C0	2	0x000000400	17.5
0x000000C0	3	0x000000A0	12.5
0x000000C0	4	0x000000D0	13

当 Sink 节点收到来自  $s$  节点的全部 4 个蚂蚁寻路数据包后,Sink 节点会比较 4 个数据包所走的路径的总代价值,选择其中代价值最小的为最优路径。之后,根据式(6)计算出该路径上的全局信息素更新值,并从蚁群路由数据表中取最优路径蚂蚁数据包数据项的源节点 ID 和序列号,组成包含全局更新分组头的数据包,发往蚁群路由数据表中保存的上一跳节点。

当节点  $r$  收到来自 Sink 节点的全局更新数据包后,它在自己的路由表中按源节点的 ID 查找路由表项。如果路由表中存在该源节点的其他序列号表项时,删除这些表项,只保留与全局更新数据包中序列号相同的表项,即保留最优路径路由,然后按全局更新分组头中的信息素更新值对其邻居列表中的对应邻居项的信息量进行更新,最后把全局更新数据包转发给其路由表中的上一跳节点,直到该数据包到达该路径的源节点。

2.2 蚂蚁寻路策略的实现

在蚁群节能路由协议中,蚂蚁寻路策略通过所有节点的分布式计算实现。协议为节点间信息的交互分别定义了 Hello 分组头、Ant 分组头和全局更新分组头。每种分组头尽可能地实现了信息量的最小化,以减小网络内的信息量,节约能耗。同时,协议规定每个节点只保存其邻居节点的位置及能量信息,节点的路由表中只保存节点上一跳及下一跳的路由信息,这样协议在设计时又同时考虑到了节点本地信息的最小化,节约了无线传感器节点有限的存储空间。最后,每个节点完成的计算工作主要为数据表的增加、删除和查找。这些操作的计算量非常小,而且协议在设计时又考虑用上文所述的距 Sink 节点较远的节点优先发送蚂蚁寻路数据包等策略来减少节点的操作次数,因此,本协议在设计实现的过程中也充分适应了无线传感器网络节点计算能力有限的特点。

运行蚁群节能路由协议的网络节点首先发送 Hello 数据包,以获得其邻居节点信息。之后,节点发送 Ant 数据包,根据蚁群算法的状态转移规则进行转发以进行最优路径搜索。其中,蚁群算法中的局部更新规则和全局更新规则实现方法如下。

(1) 蚁群算法的局部更新规则实现

在蚁群节能路由规定每个节点运行一个定时器,每轮定时时间为 0.05 秒。在蚁群寻路阶段中,当节点的定时时间到达时,它将对对其邻居节点列表中所有信息量数据域乘以  $1-\alpha_0$ (见式(3)),即实现信息素随时间在路径上的挥发。同时,在节点每次发送(或转发)Ant 数据包的时候,它都会计算数据包下一跳的节点的 ID,这时,节点将对该 ID 邻居节点的邻居列表中的信息量数据域加  $\Delta PH$ (见式(1)),即实现蚂蚁在所走路径上的信息素释放。

(2) 蚁群算法的全局更新规则实现

蚁群节能路由的全局更新规则主要通过全局更新数据包的发送实现。当 Sink 节点接收到某个源节点的全部 4 次蚂蚁寻路数据包后,它会比较出一条代价最小的路径,并计算出该路径上的全局信息素更新量,沿这条最小路径传递携带信息素更新量的全局更新数据包,以实现该路径上的信息素全局更新。其中,蚁群算法全局更新规则(见式(4)(5))中的  $\alpha_1$  在此设计中与局部更新中的  $\alpha_0$  相等。

2.3 节点意外死亡的处理

由于无线传感器网络是一个拓扑动态变化的网络,经常会出现节点因能量耗尽或其他原因意外死亡的情况,这时原有的路由将失效。为了解决意外节点死亡带来的路由可靠性问题,本文设计蚁群节能路由为一个动态路由,其过程如图 1 所示。

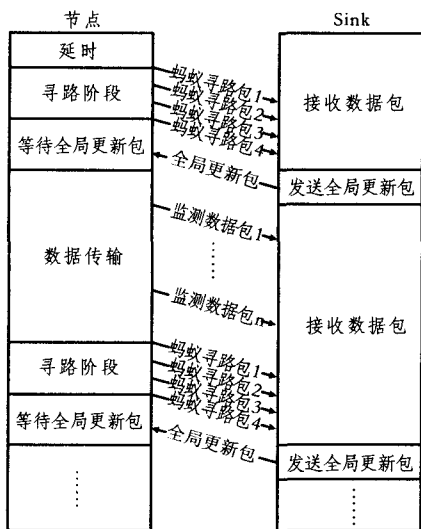


图 1 蚁群节能路由动态过程示意图

当网络开始运行后,节点先经过一个与其到 Sink 节点距离成反比的延时,之后,进入寻路阶段。在寻路阶段过程中,节点以一点的时间间隔向 Sink 节点发送 4 个蚂蚁寻路数据包。Sink 节点在收到第一个节点的第一个蚂蚁寻路数据包之后启动一个定时器。当定时时间结束时,无论 Sink 节点是否接收到全部节点的所有蚂蚁寻路数据包,都进行最优路径的比较。比较完成后沿所选出的最优路径反向发送全局更新数据包,让各个节点确定自己的路由。节点接收到全局更新数据包后进入一定时长的稳定数据传输阶段。如果节点等待全局更新数据包超时,将重新进入寻路阶段,向 Sink 节点发送蚂蚁寻路数据包。当节点的稳定数据包传输阶段完成之后,节点将再次进入寻路阶段。

节点的稳定数据传输阶段时长是根据网络的情况动态可调的。如果网络状况较好,则可选择相对较长的时长,否则相反。这种动态路由设计方法最大程度地保证了协议在无线传感器网络动态拓扑情况下的数据可靠性。同时,也可以根据不同的网络情况在数据可靠性和“网络有效数据利用率(即监测数据与路由形成发送的数据之比)”之间做出相应的调整,确保了该路由协议的最大适用性。

3 路由仿真分析

为了综合验证本文中设计的蚁群节能路由在三维空间中的应用性能,本文将在下面对该路由协议进行全方位的仿真

分析,主要包括:基于在无线传播模型的仿真、关于能量的仿真和关于网络生存时间的仿真。其中基于无线传播模型的仿真是针对该协议在各种不同网络环境、不同应用场合的综合仿真;基于能量的仿真综合检验协议的节能质量;基于时间的仿真通过调整协议中的各种参数,对网络生存时间这一无线传感器网络最重要的性能做出全面的分析。

3.1 不同应用环境的仿真

为了验证协议在不同应用环境中的性能,在仿真监测区域内随机布置若干节点,以 Shadowing 无线传播模型仿真不同的应用环境<sup>[2]</sup>,仿真应用环境参数如表 7 所示。让网络运行 48 小时(实际仿真的时间比例为 12:1)。经过多次仿真,统计结果如图 2 所示。

表 7 Shadowing 偏离  $\sigma_{dB}$  的几种典型值

环境	自由空间	城市	工厂	室内
类型	(宇宙、高空)	(街道、城区)	(工业现场)	(家居)
参数	$\beta=2$	$\beta=4$	$\beta=5$	$\beta=6$
选择	$\sigma_{dB}=10$	$\sigma_{dB}=6$	$\sigma_{dB}=6.5$	$\sigma_{dB}=8$

由仿真结果可以看出,当监测区域内节点数较少时(100 个节点),应用蚁群节能路由的网络丢包率会随着无线信道质量的下降而有所上升,其在自由空间中时,丢包率几乎为零。在信道质量最差的有障碍物的屋内时,其最大丢包率约为 0.0088%;当监测区域内的节点数目增加时(200 个节点),网络内的丢包率明显下降;当监测区域内的节点数目足够多时(300 个节点),网络内丢包率达到最好;另外,为了验证蚁群节能路由在大规模传感器网络内的应用性能,在仿真网络内布置了远大于饱和数目的节点(500 个节点),从仿真结果可以看出,网络内的丢包率并没有明显上升,这说明了该路由协议的分布式计算和最小化本地和通信数据量的特性在大规模网络应用时可以得到较好的体现。而且可以看出,该协议在自由空间中的通信可靠性非常好,在环境科学等应用领域优势突出。

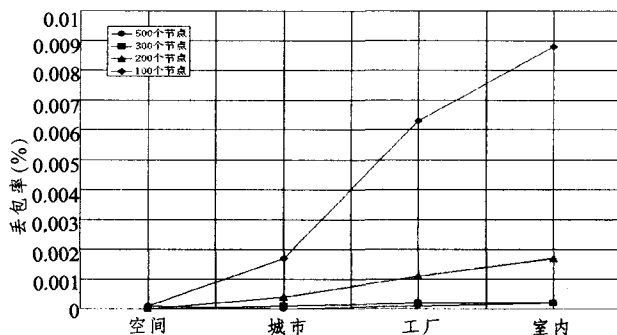


图 2 无线传播模型仿真结果

3.2 关于能量的仿真

在无线传感器网络中,能量是最关心的问题之一。评价一个路由协议的优劣在很大程度上取决于其对能量的利用效率。在本小节中,采用蚁群节能路由与无线传感器网络中的定向扩散<sup>[4]</sup>路由进行仿真对比,以验证该路由协议的节能特性。

本文设定一个  $100 \times 100 \times 1m^3$  的仿真监测区域,其中随机放置 125 个传感器节点,节点的通信半径为 30 米,Sink 节点位于目标区域边缘。仿真区域俯视图如图 3 所示。设定该网络每半小时进行一次路由重新选择。在数据传输过程中,平均每 10 秒钟随机地有一个节点向 Sink 节点发送数据,节

(下转第 118 页)

ing, 2006 (in Chinese with English abstract)

[18] Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 1988 24(5): 513-523

[19] MacQueen J. Some methods for classification and analysis of multivariate observation//*Proc. of the 5<sup>th</sup> Berkeley SympMath Statist and Prob 1*. California: University of California Press, 1967:281-297

[20] Wu Chen, Zhang Quan. An Information Retrieval Method Based on Language Concept Space Using Clustering Method. *Computer Engineering*, 2006 (in Chinese with English abstract)

[21] Wu Chen, Zhang Quan. Introduce Linguistics and Semantics into the Probabilistic Indexing Method for Chinese IR: Using Concept Theory. *Proceedings of ICTAI-2006*

[22] Wu Chen, Zhang Quan. Content matching: a concept-based approach for information retrieval. *Journal of Southeast university (English edition)*, 2006, 12(5)

[23] Jing Yaohong, Miao Chuanjiang. An Algorithm of Extracting Text Character Based on a Model of Context Framework. *Journal of Computer Research and Development*, 2004, 41(4)

[24] Jing Yaohong. An information retrieval method based on language concept space using clustering method. *Computer Engineering and Applications*, 2003, 39(13)

(上接第 72 页)

点发送的监测信息包大小为 256 bits。

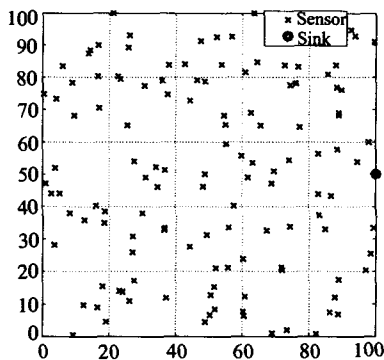


图 3 仿真环境俯视图

设定每一个节点的初始能量相同,均为 3J,并采用自由空间传输信道模型进行仿真<sup>[3]</sup>。在该模型中,节点发送和接受数据时器件的能耗  $E_{dec} = 20\text{nJ/bit}$  ( $1\text{nJ} = 10^{-9}\text{J}$ ),发送功率放大器的能耗为  $E_{amp} = 100\text{pJ/bit/m}^2$  ( $1\text{pJ} = 10^{-12}\text{J}$ )。鉴于以上设定,传感器节点发送消息时消耗的能量  $E_{run} = E_{dec} + E_{amp} = 20\text{nJ/bit} + 100\text{pJ/bit/m}^2$ ;节点接收数据时消耗的能量  $E_{recv} = E_{dec} = 20\text{nJ/bit}$ 。

网络平均能耗仿真结果如图 4 所示。其中,图 4(a)和图 4(b)分别为运行定向扩散路由和运行蚁群节能路由的传感器网络在 10 个小时内的能量消耗平面图。图 4(a)中网络的平均能量消耗为 214.3mJ,图 4(b)中网络的平均能量消耗为 143.75mJ,减少了 32.9%。相对定向扩散路由,蚁群路由由节约能耗的主要原因是:定向扩散路由是一种基于查询方式的路由协议,在路由建立阶段要进行兴趣消息的泛洪,这样增加了建立路由的代价,而蚁群节能路由通过节点的分布式计算来有选择地发送路由建立消息(在无线传感器网络中,节点的计算能量代价远远小于通信能量代价),不但减少了通信数据量,而且在建立路由时就把“节能”作为选取标准,因此路由能耗得以大幅降低。与此同时,由于蚁群节能路由在建立过程中综合考虑了节点能量及节点间通信能耗,网络中的能量负载也更加均衡

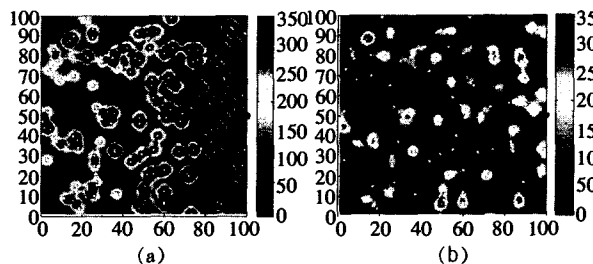


图 4 路由协议能量仿真结果平面图

了,这可以从图 4(a)与图 4(b)的对比中反映出来。

### 3.3 关于网络生存时间的仿真

在网络生命周期仿真中,仿真环境与上小节设定相同,网络持续运行直到有节点能量耗尽。基于定向扩散路由的网络共计运行 57 小时 22 分,基于蚁群节能路由的网络共计运行 84 小时 04 分,这意味着基于遗传蚁群路由的网络生命周期比基于定向扩散路由的网络延长了 46.5%。基于蚁群节能路由的网络生命周期得到大幅延长的原因有二,一是分布式路由优化计算使得节点节约了大量的通信能耗;另外,蚁群节能路由在建立过程中考虑了能量负载在网内的均衡性,避免了由此局部节点能量耗尽而引起的网络出现盲点,这也是网络生命周期延长的主要原因。

**结束语** 本章中针对无线传感器网络的特性,设计了一种蚁群节能路由协议。该协议具有分布式计算、分布式信息存储、路由通信代价少及本地存储信息量小、低能耗、能量负荷均匀等特点。另外,该路由协议还具备适应性好,可扩展等特点。

为了具有较好的通用性,在上面所述的路由协议为平面路由,即无线传感器节点之间在路由上没有层次之分。如果针对其他应用环境,该协议可以方便地扩充为层次路由协议,即网络可以根据位置信息化分成动态变化的若干簇,簇内通过能量状况及所处位置选举产生一个簇头节点。簇内的簇头节点在其任期内保持侦听状态,其他节点在无数据时保持休眠,当有数据时才唤醒并将数据直接发送给簇头,各个簇头节点再通过蚁群路由形成二级最优通路与 Sink 节点通信。通过这种简单的扩充蚁群节能路由就可以在不同的应用环境中获得更加优越的性能。

### 参考文献

[1] Marco D, Vittorio M, Alberto C. The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Trans. on Systems, Man, and Cybernetics-Part B*, 1996, 26(1): 1-13

[2] Rappaport T S. *Wireless communications, principles and practice*. Prentice Hall, 1996

[3] Friis H T. A note on a simple transmission formula//*Proc. IRE*. 1946, 34

[4] Intanagonwivat C, Govindan R, Estrin D. Directed diffusion: A scalable and robust communication paradigm for sensor network//*Proc. 6th Annual Int'l Conf. on Mobile Computing and Networks (MobiCOM 2000)*. Boston, MA, August 2000