

Intelligent Clustering Algorithm of WSN Based on DPSO*

FAN Xinggang*, HOU Jiabin, JIE Jing, WANG Wanliang, WANG Yi

(College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: This paper presents a new intelligent clustering routing algorithm of WSN (Wireless Sensor Network), DPSO—CR (Discrete Particle Swarm Optimization—Clustering Routing), to solve three major problems that can consume energy heavily in LEACH (Low Energy Adaptive Clustering Hierarchy), a classical clustering protocol in WSN. To solve the problem of uneven clustering in LEACH, this paper proposes to calculate the positions of global best cluster heads to make the clustering structure even through modified DPSO. To solve the heavy energy consumption of one-hop communication between cluster head and base station in LEACH, this paper proposes to calculate multi-hop minimum energy consumption path between each cluster head and base station. To solve the heavy energy consumption of global cluster heads re-election mechanism, this paper proposes local cluster heads update mechanism to save the energy consumption further. The experiment shows in comparison with LEACH, DPSO—CR not only has a reasonable and even cluster, the lifetime of network is also prolonged greatly.

Key words: LEACH; DPSO—CR; multi-hop minimum energy consumption path; global best cluster heads; local cluster heads update mechanism

EEACC: 6150P

doi: 10.3969/j.issn.1004-1699.2011.04.024

基于 DPSO 的智能 WSN 分簇路由算法*

范兴刚*, 侯佳斌, 介靖, 王万良, 王翊

(浙江工业大学计算机科学与技术学院 杭州 310023)

摘要: 主要针对无线传感网络经典分簇协议 LEACH (Low Energy Adaptive Clustering Hierarchy) 中存在的三个使能量消耗过大的问题, 提出了一种新的智能无线传感网络分簇路由算法 DPSO—CR (Discrete Particle Swarm Optimization—Clustering Routing)。针对 LEACH 中分簇不均匀的问题, 提出用改造后的离散粒子群优化 DPSO 直接计算全局最优簇首的位置, 使簇结构分布均匀合理; 针对 LEACH 簇首与基站单跳通信能量消耗过大的问题, 根据能量传输代价作为权值计算每个簇首到基站的多跳最小能耗路径; 针对 LEACH 按轮进行全局簇首重选机制带来能耗过大的问题, 提出了局部簇首更新机制来进一步节省能量消耗。实验结果表明, 与 LEACH 协议相比, DPSO—CR 算法不但分簇结构合理均匀, 网络的生命周期也大大延长。

关键词: LEACH; DPSO—CR; 多跳最小能耗路径; 全局最优簇首; 局部簇首更新

中图分类号: TN915

文献标识码: A

文章编号: 1004-1699(2011)04-0593-08

无线传感器网络 WSN (Wireless Sensor Network) 是 21 世纪新兴的信息技术之一, 并且被美国商业周刊列为 21 世纪改变世界的十大技术之一。WSN 是由许多廉价的具有信息采集功能的传感器节点组成的网络, 用来感知和采集各种信息, 并将采集到的信息传输给基站 BS (Base Station) 对信息做进一步处理。传感器节点由电池提供能量, 因此能量非常有限, 而且每个节点的计算存储能力也很弱^[1]。

由于 WSN 能量有限, 设计能量有效的 WSN 路由协议显得尤为重要。目前 WSN 路由协议分为平面路由协议和分层路由协议两大类。LEACH 协议是一种有代表性的低功耗分层路由协议, LEACH 相比其他平面路由协议生命周期提高 15% 以上^[2]。然而 LEACH 依照概率随机选择簇头的机制有很大的随机性不保证产生均匀的簇结构, 不均匀的簇结构会大大缩短 WSN 的生命周期。而且 LEACH 中

簇首和 BS 直接通信的机制也会消耗大量的能量。

PSO 是一种群智能算法,由 Kennedy 和 Eberhart 首先提出^[3]。PSO 已经被证明是一种解决工程领域 NP 难问题的有效的解决方法^[4-6]。为解决 LEACH 协议分簇不均匀的问题,本文把基本粒子群优化 (PSO) 算法改造成适合分簇问题的 DPSO。并且直接由基站运行改造后的 DPSO 计算出最佳簇首的位置。在计算过程中,本文使用 PSO 惯性权重调整和变异策略来避免 PSO 的容易陷入局部最优的现象并且通过针对本问题的启发算法来帮助计算最佳簇首的位置。计算出最佳簇首位置后,本文根据能量传输代价为权值计算每个簇首经由其他簇首到基站的最短路径。另外本文提出让簇首感应自己的剩余能量,当能量小于一定阈值时,算法会选择当前簇内节点剩余能量最多的簇内节点当选新的簇首,从而避免 LEACH 中隔一定时间全部重新选择簇首带来的能量消耗。

最后本文使用 MATLAB 对本文提出的算法进行仿真验证。实验比较了 DPSO—CR 和 LEACH 的分簇效果,还比较了传输数据所消耗的能量情况。实验结果表明无论从分簇的均匀性还是网络生命周期来看,DPSO—CR 相对 LEACH 协议都有大幅提高。

1 LEACH 协议及其改进

1.1 LEACH 协议描述和分析

下面简要描述 LEACH 协议的原理并给出分析。

LEACH 协议分为两个阶段。第一个阶段是产生簇的阶段。这个阶段中每个传感器节点产生一个 0-1 的随机数,如果该随机数小于一个阈值公式的值这个节点就成为簇首。这个阈值公式表达式如下:

$$T(n) = \begin{cases} \frac{K}{N - K[r \bmod (N/K)]} & n \in Gr \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

其中 n 表示某个传感器节点的编号, N 是传感器节点的个数, K 是期望产生的簇头个数, r 是选举簇首的轮数, Gr 是最近的 N/K 轮中还没当选过簇首的节点集合。

节点当选簇首后会广播其成为簇首的消息,其余普通节点选择收到信息的信号强度也就是距离自己最近的簇首加入。普通节点发送给簇首加入它的消息,簇就建立了。

第二个阶段是稳定的数据传输阶段,普通节点

按 TDMA 时间片在规定时间内给簇首,簇首对数据经过一定的融合后将数据一跳传给基站。数据传输阶段进行一个时间段后(远长于第一阶段)进行下一轮簇首的选择^[2]。

LEACH 的随机选择簇首机制可能导致非常差的分簇效果。首先导致簇首个数不确定,可能出现簇首个数远小于期望簇首数 K 的情况;簇首可能分布很不均匀,比如集中某一区域,这样普通节点可能距离最近的簇首仍然非常远,这样在通信中必然耗费大量的能量,簇首分布不均匀还导致簇与簇之间能量消耗不均衡;在簇首选择的时候也没有考虑剩余能量因素,剩余能量少的节点担任簇首将导致节点能量迅速耗尽而死亡。再者簇首直接将数据一跳传输给基站也不合理,长距离的传输对节点能量的消耗是非常大的;最后 LEACH 协议按轮重新全局选择簇首的机制由于当选簇首的广播消息也耗费大量的能量^[10,15]。

1.2 对 LEACH 协议的改进

针对 LEACH 协议存在的问题,国内外学者也提出了不少 LEACH 的改进策略。

M. J. Handy 等人对 LEACH 协议簇首选择公式进行改进,考虑了剩余能量的因素,改进后的公式如下^[7]:

$$T(n) = \frac{P}{1 - p \left(r \bmod \frac{1}{P} \right)} \left[\frac{E_{\text{cur}}}{E_{\text{max}}} + \left(r_s \text{div} \frac{1}{P} \right) \left(1 - \frac{E_{\text{cur}}}{E_{\text{max}}} \right) \right] \quad (2)$$

其中 P 是簇首所占所有节点百分比, E_{cur} 是现在剩余能量, E_{max} 是初始能量, r_s 是该节点到目前为止还没当选簇首的轮数。通过该改进一定程度上避免了低剩余能量的节点当选簇首。但是概率性机制没有改变,仍然可能选择低剩余能量成为簇首,而且分簇不均匀这个大问题也没有解决。

王国芳等人利用 Prim 算法生成包括基站在内的所有簇首节点的最小生成树,实现了簇首和基站之间的多跳通信,找到了簇首间短距离通信的最优路径^[8]。

梁英等人采用先使用 LEACH 算法进行分簇得到候补簇首,然后用粒子群算法进行优化在已经产生的簇内确定最后簇首^[13]。虽然先用 LEACH 算法进行分簇,再在簇内用粒子群选择最优簇首的方法能在一定程度上解决簇首位置不合理的情况,但是从全局来看先用 LEACH 进行分簇可能已经产生不合理的簇划分,再在已经划分的簇内优化只是一种局部优化。而本文提出利用粒子群算法直接寻找全

局最优的簇首位置,从而使簇结构在整个监测区域来看都是分布均匀合理的。

2 本文的协议描述

本文算法假设节点能量有限且能感知剩余能量,并且可以根据发送距离长短调整发射的功率。另外本文假设传感器节点拥有 GPS 定位功能能获取自己相对基站的位置信息。基站能量无限,计算最佳簇首位置和簇首到基站的多跳路径都由基站来计算。

2.1 计算最佳簇首位置

本文提出算法的第一步就是通过改造的离散粒子群算法计算最佳簇首的位置。下面详细描述。

2.1.1 粒子群算法的适应度函数

运用粒子群算法的第一步就是要确定适应度函数。本文的问题是从 N 个节点找出一组处于最佳位置的节点担当簇首的角色使分成的簇均匀。当一个簇内的节点能紧密地围绕着簇首节点时可以被认为是个均匀的簇。当簇内所有节点到簇首的距离总和最小且它们到簇首距离都差不多大小时(也就是簇内所有节点到簇首的距离方差最小时)可以刻画出紧密围绕的特征。因此从所有节点来看,当所有簇的簇内节点到各自簇首距离总和的平均值与所有簇的簇内节点到各自簇首的距离方差的平均值同时达到最小时,所有的簇的结构应该是均匀合理的。适应度函数定义如下:

$$F = \min \left[\text{mean} \left(\sum_{j=1}^n \left(\sum_{i=1}^m D_{ji-jCH} \right) \right) * \text{mean} \left(\sum_{j=1}^n \left(\text{var} \left(\sum_{i=1}^m D_{ji-jCH} \right) \right) \right) \right] \quad (3)$$

其中 D_{ji-jCH} 代表第 j 个簇中第 i 个簇内节点到簇首的距离, i 从簇内第一个节点到第 m 个节点, j 从第一个簇到第 n 个簇。mean 是求平均值的函数, var 是求方差的函数。

这里还有一点要说明,根据文献[9],当簇首节点所占百分比为所有节点 5% 时,能量最有效。本文中运用了这个研究成果,簇首个数就设定为所有节点的 5%,簇内节点个数都相等。

2.1.2 分簇问题的离散粒子群改造

要解决本文提出的计算最佳簇首位置的问题,必须对基本 PSO 进行相应的离散改造。国内外学者对解决 TSP 问题(Travelling Salesman Problem)的粒子群离散改造的研究成果比较多,比较著名的是 Clerc 提出的基于交换子的离散改造方案^[11]。但分簇问题并非经典的 TSP 问题,本文在借鉴 Clerc 方

法的基础上,提出了针对分簇问题的 DPSO 模型。

首先本文定义了针对分簇问题的粒子结构:

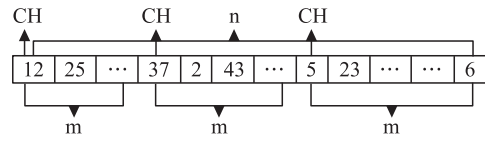


图 1 本路由算法粒子结构示意图

如图 1 所示,粒子的维数 n 代表有 n 个传感器节点,其中每一维里的数字代表每个节点的编号。 m 代表每个簇内都有相等的 m 个节点,这里 m 应为 $20(1/5\% = 20)$ 。CH 代表簇首(Cluster Head)假设 i 是 n 维粒子的坐标,如果 i 对 20 取余结果为 1,则 i 代表的是簇首的坐标,否则为簇内节点坐标。

标准 PSO 按照以下两个公式更新粒子的速度和位置:

$$v_{in}(t+1) = wv_{in}(t) + c_1 \cdot (P_{in}(t) - x_{in}(t)) + c_2 \cdot (P_{gn}(t) - x_{in}(t)) \quad (4)$$

$$x_{in}(t+1) = x_{in}(t) + v_{in}(t+1) \quad (5)$$

其中 $x_{in} = (x_{i1}, x_{i2}, \dots, x_{in})$ 代表粒子 i 的位置也就是上述粒子的结构,每个位置就是问题的一个解。 $v_{in} = (v_{i1}, v_{i2}, \dots, v_{in})$ 代表粒子 i 的速度。 $p_{in} = (p_{i1}, \dots, p_{in})$ 代表粒子 i 的最好位置, $p_{gn} = (p_{g1}, \dots, p_{gn})$ 代表所有粒子的最好位置。 C_1 是粒子个体学习因子, C_2 是群体学习因子, w 是惯性权重因子。

根据文献[11],本文也将粒子速度定义为 $v_i = \text{swap}(j, k)$,即代表将粒子位置坐标 j 的节点编号与坐标 k 的节点编号进行交换,一个速度可能是若干个这样的交换序列的集合。同时也更改了公式(4)和式(5)中的“+”,“—”和“*”的定义。

(位置+速度):旧位置与速度“+”的结果为新位置。让速度中的所有交换序列依次改变旧位置的节点编号。假设旧位置为(1,4,5,3,2),速度为((1,2),(3,5)),则新位置为(4,1,2,3,5)。

(位置—位置):两个位置的“—”的结果为一个速度。假设 $p1$ 为第一个位置, $p2$ 为第二个位置, $p1-p2$ 产生的 v 应满足 $p2+v=p1$ 。如上 $p1$ 为(4,1,2,3,5), $p2$ 为(1,4,5,3,2),则“—”的结果应为((1,2),(3,5))。

(速度+速度):两个速度的“+”的结果为新速度。即简单将两个速度的交换序列合并。设一个速度为((1,2),(3,5)),另一个为((3,4)),新速度为((1,2),(3,5),(3,4))。

(因子*速度):因子与速度的“*”结果仍然为一速度,定义为速度中的每一个交换序列以因子的值为概率保留(因子都为一个小于 1 的实数)。

2.1.3 避免 PSO 的局部最优现象

粒子群算法在迭代过程中容易出现陷入局部最优的现象,本文采用线性惯性权重调整和变异策略来避免这种现象发生。

公式(4)中的 w 就是惯性权重,它是个重要的参数。较大的 w 具有较好的全局收敛能力,而较小的 w 则有较强的局部搜索能力。现在的惯性权重研究一般分为3类:固定惯性权重,线性惯性权重调整,非线性惯性权重调整。其中线性递减的惯性权重调整策略迭代前期具有较好的全局搜索能力,后期收敛性也较好,但缺点是收敛速度慢^[12]。本文采用线性递减的惯性权重调整来避免陷入局部最优,同时结合启发算法来提高收敛速度。调整公式如下:

$$w = w_{\max} - (w_{\max} - w_{\min}) \cdot \frac{\text{iter}}{\text{maxiter}} \quad (6)$$

其中 w_{\max} 是开始迭代的 w 值, w_{\min} 是迭代结束的 w 值,一般取 w_{\max} 为 0.5 ~ 0.7, w_{\min} 为 0.1 ~ 0.2, iter 是当前迭代的轮数, maxiter 是最大的迭代轮数。

本文还提出使用变异的策略来避免陷入局部最优即如果在粒子群算法迭代过程中判断如果每个粒子隔一定轮数个体最优解没有发生变化,并且该粒子位置和全局最优粒子位置的相似度在一定程度以上就使该粒子位置的某些维(即粒子编号)进行交换。本文中具体的做法是如果经过总迭代次数 maxiter 的 10% 轮后,如果某个粒子隔一定轮数个体最优解没有发生变化,并且该粒子位置和全局最优粒子位置相似度在 80% 以上时,则在粒子位置中的每个簇中随机选择一维与当前的簇首所在的维交换即选择本簇内其他节点当簇首。当然本来就是全局最优粒子不需要变异。这种变异策略可以使粒子在迭代过程中产生一定的概率找到比当前全局最优解更好的解,避免陷入局部最优。

2.1.4 启发式算法

本文提出两种启发式算法来提高粒子群迭代计算最优簇首的效率。

(1) 交换簇之间簇内成员的位置

如果在 PSO 迭代过程中发现某粒子位置中某个簇内节点相比自己当前的簇首更靠近另外某个簇首,那这样的簇内节点应该被交换到合适的簇中去。

算法使用一个阈值来决定是否需要某个簇内节点做交换操作。假设传感器节点均匀布撒的区域面积为 S , N 为总的节点个数,那每个簇的理想覆盖面积应该是 $S/(N \cdot 5\%)$ 。把每个簇的覆盖面看成是以簇首为中心的圆面,每个簇的簇半径应为

$\sqrt{S/(N \cdot 5\% \cdot \pi)}$ 。为了提高灵活性和减少误判的可能,算法设定阈值为 $1.5 \cdot \sqrt{S/(N \cdot 5\% \cdot \pi)}$ 。算法检查每个簇的簇内节点,若发现某个簇内节点与当前簇首的距离大于 $1.5 \cdot \sqrt{S/(N \cdot 5\% \cdot \pi)}$ 则启动交换过程。假设发现超过阈值的是 A 簇的簇内节点为 A1,当前簇首为 A-H。首先寻找最接近 A1 的簇首,如果仍为当前簇首 A-H,则终止交换过程。若找到的为其他簇首 B,则在 B 簇中寻找最接近 A-H 的簇内节点 B1,然后判断 B1 到 B 簇簇首 B-H 的距离是否超过阈值,若也超过阈值则交换,否则不交换。

(2) 调整簇首位置

设某个簇中簇首节点与离它最远的簇内节点之间的距离为 Dis-far,与离它最近的簇内节点距离为 Dis-near。若 Dis-far 与 Dis-near 之间的差值过大则认为该簇不均匀即各簇内节点到簇首的距离方差较大,此时需要在当前簇的节点中找出位置更适合的节点当选簇首。本文设定这个差值阈值为 $1.3 \cdot \sqrt{S/(N \cdot 5\% \cdot \pi)}$,当发现某个簇中的 Dis-far 与 Dis-near 之间的差值大于这个差值阈值时,算法将所有簇内节点到当前簇首的距离排序,然后将距离当前簇首中等的那个簇内节点当选为新的候补簇首,继续上述检查过程,直到找到合适的簇首节点成为该簇最终的簇首节点。

使用两种启发式算法的策略是每经过总迭代次数 maxiter 的 5% 轮数后全局最优解,随机选择一半粒子依次使用启发式算法(1)和算法(2)对粒子结构进行指导调整。

2.1.5 最优簇首计算的流程

整个最优簇首的计算阶段的过程如下:

(1) 随机初始化粒子群中每个粒子的位置和速度。由当前粒子位置根据公式(3)计算每个粒子的适应度值,并根据最优适应度值得出全局最优粒子,个体最好粒子初始化为每个粒子的初始位置。

(2) 接下来开始粒子群算法的迭代过程,更新迭代轮数。每个粒子根据公式(4)和(5)计算新的位置和速度。

(3) 每个粒子根据更新的位置计算各自的适应度函数值,得出最优适应度值。如果新的个体适应度值比个体历史最优适应度值更好,更新个体最优粒子位置。如果新的最优适应度值比历史最优适应度值更好则更新最优粒子位置。

(4) 判断迭代轮数,如果达到 maxiter 的 5% 则随机对一半粒子运用两种启发式算法,然后转第(5)步。否则直接转第(5)步。

(5)判断每个粒子是否连续 \maxiter 的 10% 轮个体最优解未发生变化,若是则对该粒子进行变异,然后转第(6)步,否则直接转第(6)步。

(6)判断是否达到迭代最大轮数,若是则结束迭代,否则更新惯性权值然后转第(2)步继续迭代。

2.2 计算簇首与基站之间多跳路径

LEACH 协议的另外一个严重的缺点就是簇首和基站之间都是单跳的,每个簇首无论距离基站多远,簇首都将收集到的数据直接发给基站。

下面先来看一下无线传感网络的传输能量模型:

$$E_{TX(k,d)} = \begin{cases} kE_{elec} + kE_{fs}d^2 & d \leq d_0 \\ kE_{elec} + kE_{mp}d^4 & d > d_0 \end{cases} \quad (7)$$

$$E_{RX(k,d)} = kE_{elec} \quad (8)$$

$E_{TX(k,d)}$ 表示向距离自身 d 处的节点发送 k 个 bit 数据所消耗的能量, $E_{RX(k,d)}$ 表示从距离自身 d 处的节点接收 k 个 bit 数据所消耗的能量。在发送数据时能量消耗与距离 d^n 成正比,短距离传输即 $d \leq d_0$ 时, $n=2$,这种能量衰减模型称为自由空间模型,在长距离传输即 $d > d_0$ 时, $n=4$,这种模型称为多径衰减模型。 E_{elec} 表示发送或接受 1 bit 数据电路消耗的能量, E_{fs} 是自由空间模型中无线通信放大电路消耗的能量, E_{mp} 是多径衰减模型中无线通信放大电路消耗的能量, d_0 门限值等于 $\sqrt{E_{fs}/E_{mp}}$ [14]。

在文献[8]中,王国芳等人提出建立包括所有簇首和基站在内的最小生成树来减少 LEACH 中簇首与基站单跳通信消耗的能量。但是最小生成树的目的是使图上各点连通且使这棵生成树的权值总和最小,而本文认为计算每个簇首到基站的最小能量代价更符合改进簇首与基站单跳传输能量消耗过大的缺点这个目标。且在文献[14]中,过文亮用理论和实验证明了以能量传输代价为权值的最段路径算法比最小生成树的方法更节省能量。文献[14]将此算法用于 LEACH 协议中簇内各节点到簇首的多跳路径。文献[14]主要考虑 LEACH 的分簇不均匀,有些簇内节点到簇首距离非常远,于是考虑在簇内采用多跳传输,但这样将增加数据传输的时延。本文由于采用了基于群智能算法对选举簇首经过改造后,簇变的很均匀,簇内节点到簇首距离相对比较小,然而仍然存在簇首距离基站过大的情况,因此本文在簇首与基站之间运用此算法建立多跳路径。

本文以下面公式作为最短路径计算时的权值:

$$W_{CHi-CHj(k,d)} = E_{TX(k,d)} + E_{RX(k,d)} = \begin{cases} k(2E_{elec} + E_{fs}d^2) & d \leq d_0 \\ k(2E_{elec} + E_{mp}d^4) & d > d_0 \end{cases} \quad (9)$$

$W_{CHi-CHj(k,d)}$ 表示簇首 i 向距离自身为 d 的簇首 j 发送 k 个 bit 数据的能耗代价。本文以它作为权值对每个簇首为源节点,基站为终点运行 Dijkstra 算法。当然无线传感网络节点应该被看作是无向图上的点。因此本文把传统基于有向图的 Dijkstra 算法改造成基于无向图的 Dijkstra 算法,并使用它确定每个簇首与基站通信所消耗能量最小的多跳路径。

2.3 局部簇首更新机制

和 LEACH 协议多次进行簇首选举不同,本文提出只由基站进行一次簇首选举出最好位置的簇首,在后面轮次中不再进行全局簇首选举,而通过节点自感知剩余能量局部更新簇首从而进一步节省能量。具体过程如下:在传输数据给簇首时,簇内节点都会捎带一个剩余能量信息。簇首会将其簇内节点当前剩余能量记录下来,同时簇首检查自己的剩余能量,当剩余能量低于初始能量的 10% 时,簇首选出本簇内剩余能量最多的簇内节点,让它成为新的簇首。在本次数据传输给基站时将这个新节点 ID 报告给基站。然后由基站来通知该簇内节点新的簇首信息以及向多跳簇首间的相关簇首(指老簇首的上跳和下跳簇首)报告新簇首信息。可以看到基于能量感应机制的局部簇首更新机制基本不需要耗费额外的能量,大部分广播新簇首相关信息耗费的能量都由基站来完成,这样相比 LEACH 的全局更新簇首机制可以大大节省能量。当然付出的代价是需要记录簇内节点的剩余能量信息。

2.4 路由算法的整体描述

DPSO—CR 和 LEACH 协议一样,也分为簇首选举阶段和数据传输阶段。

在簇首选举阶段首先由所有传感器节点向基站报告位置信息,由基站根据所有节点的位置信息来运行改造的 DPSO 算法计算出全局最优簇首。这个过程前面已经详细说明,这里不再赘述。然后基站再根据簇首运行以能量代价为权值计算簇首到基站的多跳路径。然后基站将广播它的计算结果,包括簇首 ID 及位置(每个节点都有自己预先设置的 ID),以及簇首通信的上一跳和下一跳信息。每个节点收到自己的身份,簇内节点记录各自的簇首 ID。簇首节点记录它的簇成员 ID 以及上下跳簇首 ID 以及位置,然后就可以进入数据传输阶段。

在数据传输阶段,也和 LEACH 协议一样,簇内节点将数据发送给各自的簇首,簇首收集所有簇内节点的数据后将数据经过簇首间的多跳路径传送给基站。当全部簇内节点都死亡只剩簇首节点时,簇首直接感知数据并将采集的数据传给簇首。

3 仿真实验分析

本文以 MATLAB 为仿真工具,对 LEACH 协议和 DPSO—CR 算法从分簇效果和生命周期两方面进行比较。

首先介绍实验的场景:100 个传感器节点随机地分布在一个 200 m 为边长的正方形区域。基站位于正方形的正中心且能量无限。

3.1 分簇效果的比较

这里比较 LEACH 协议和 DPSO—CR 的分簇效果。LEACH 协议和 DPSO—CR 使用的节点位置都一样。改造的 DPSO 的参数如下:最大迭代次数 \maxiter 为 1000 轮,初始惯性权重 $w_{\max}=0.7$,迭代结束的惯性权重 $w_{\min}=0.1$,个体学习因子 $C_1=0.2$,群体学习因子 $C_2=0.2$,在以后的迭代过程中惯性权重 w 线性递减,且始终保持 $C_1=C_2$ 并且 $C_1+C_2+w=1$ 。而 LEACH 协议中期望簇首概率 P (即 K/N)=0.05。

图 2 显示了运行 LEACH 协议的分簇效果。相同图形代表同个簇,某个簇当中下三角形代表簇首节点,其他围绕簇首的相同图形代表簇内节点。可以看出,虽然期望簇头个数为 5 个,但由于 LEACH 依照概率选簇头的机制导致簇头个数可能少于 5 个;簇内成员个数严重不均衡,簇首和簇内各节点的距离可能很大;簇首位置可能非常不合理。

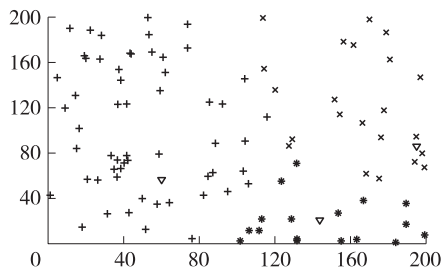


图 2 LEACH 分簇效果(期望簇首个数为 5)

图 3 显示同样位置的节点分布运行 DPSO—CR 的分簇效果,可以看出 DPSO—CR 的分簇效果明显要比 LEACH 协议好。簇首均匀地分布在各个区域,且簇内节点个数都一样,除了节点随机布撒比较稀疏的区域外簇内节点都比较紧密地围绕在簇首周

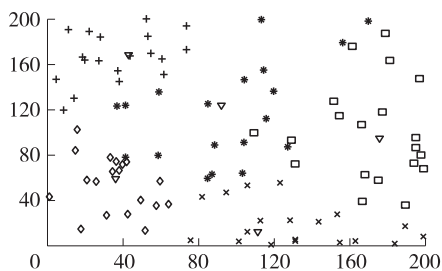


图 3 本文路由算法分簇效果

围,没有出现簇首与簇内节点距离过大的情况。

这里比较 LEACH 协议和 DPSO—CR 算法的适应度函数(公式 3)的值,可以对两种算法的分簇效果有更直观的认识。公式 3 分成 f_1 和 f_2 ,其中

$$f_1 = \text{mean} \left(\sum_{j=1}^n \left(\sum_{i=1}^m D_{ji-jch} \right) \right)$$

$$f_2 = \text{mean} \left(\sum_{j=1}^n \left(\text{var} \left(\sum_{i=1}^m D_{ji-jch} \right) \right) \right)$$

表 1 LEACH 和 DPSO—CR 的适应度函数值比较

	f_1	f_2	$F=f_1 \cdot f_2$
LEACH	2543.1	1291.5	3 284 400
DPSO—CR	715.6	227.5	162 810

上表数据是相同分布的传感器节点连续各运行 3 次 LEACH 协议和 DPSO—CR 适应度函数的平均值。从表中可以看出,无论是 f_1 的值还是 f_2 的值,LEACH 协议的值都比 DPSO—CR 大好几倍,总适应度函数值 LEACH 协议是 DPSO—CR 的 20 倍左右。这说明 LEACH 协议中簇首和簇内节点不但距离总和过大而且距离大小非常不均匀,这将大大缩短网络生命周期,这可以从生命周期比较中看出。

3.2 网络生命周期和能量消耗比较

这里通过模拟传感器节点周期性的数据传输来比较 LEACH 协议和 DPSO—CR 的生命周期和能量消耗情况。每个传感器节点初始能量均为 0.5 J,按公式(7)和公式(8)的能量传输模型计算能量消耗,其中参数为 $E_{\text{elec}}=50 \text{ nJ/bit}$, $E_{\text{fs}}=10 \text{ pJ}/(\text{bit} \cdot \text{m}^2)$, $E_{\text{mp}}=0.0013 \text{ pJ}/(\text{bit} \cdot \text{m}^4)$, $d_0=87 \text{ m}$ 。实验比较两个协议中从分簇到数据传输的能量消耗情况。实验中分两种控制信息包和数据包两种,控制信息包为分簇阶段信息传递的数据包,大小为 100 bit,数据包为传输数据所用的数据包,大小为 4 000 bit。数据传输阶段按轮进行,两个协议中都规定每轮中每个簇内节点采集一个数据包,并发送给它的簇首,簇首收集本簇内所有簇内节点的数据包,并且按照两种协议中单跳或多跳方式传给基站。

其中横轴是实验中规定的轮数,纵轴是仍然存活的节点数量。下面表 2 对图 4 中的关键数据进行了描述。

表 2 LEACH 和 DPSO—CR 的网络生命周期数据比较

	第一个节点死亡轮数	最后一个节点死亡轮数	网络每轮平均能耗
LEACH	80	605	$8.26 \times 10^{-2} \text{ J}$
DPSO—CR	243	1008	$4.96 \times 10^{-2} \text{ J}$

从图 4 和表 2 中可以看到,LEACH 协议第一个节点死亡的轮数是 80 轮,而 DPSO—CR 第一个节点

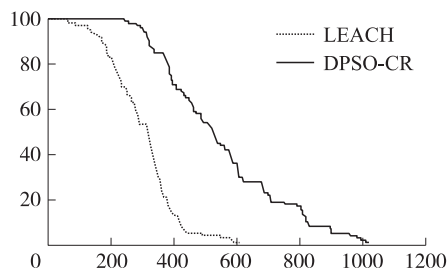


图4 网络生命周期比较

死亡轮数是 243 轮, DPSO—CR 比 LEACH 协议晚了两倍左右的时间。LEACH 协议中最后一个节点死亡的轮数 605 轮, 而 DPSO—CR 最后一个节点死亡的轮数是 1008 轮。网络每轮平均能耗也是个重要的指标, 它反映了路由算法的能量有效性, 可以看到 LEACH 算法每个节点每轮的能耗是 DPSO—CR 的 1.7 倍, 这正是本文提出的三种节省能量机制的效果体现。

图 5 中显示了整个网络总能耗情况。纵轴代表网络总能量 50 J (0.5 J×100), 横轴代表实验轮数。可见, 从整个网络能量消耗来看, DPSO—CR 与 LEACH 协议相比大大节省了能量。文献 [13] 先 LEACH 分簇然后通过粒子群局部优化簇首的方法在相近的实验条件下只比 LEACH 延长 20% 左右的生命周期, 可见本文提出的路由算法要优于文献 [13] 的方法。

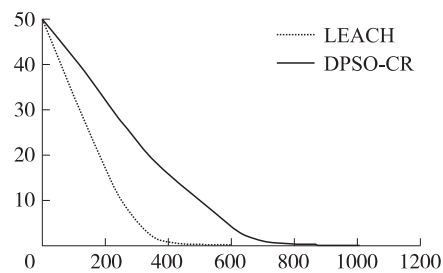


图5 网络总能耗比较

4 总结和展望

本文针对 LEACH 协议中存在的三个使能量消耗过大的机制, 提出了基于离散粒子群算法的智能无线传感网络分簇路由算法 DPSO—CR。该算法主要包含三种新的机制来分别改进 LEACH 协议中存在的三个问题。其中基于离散粒子群算法计算全局最优位置的簇首是本文的核心和主要创新点, 这个机制能大大改进 LEACH 协议分簇不均匀的缺点。在簇首间基于能耗代价的最短路径算法也改善了 LEACH 协议簇首和基站单跳通信能耗过大的缺点。基于能量自感应的局部簇首更新机制通过将广播消息所消耗的能量转嫁到基站上的方法又进一步延长

了网络的生命周期。

当然 DPSO—CR 算法是有一定代价和前提条件的。首先计算全局最优簇首需要节点的位置信息, 这要求传感器节点具有 GPS 定位功能或者先通过无线传感网络定位算法计算出每个节点相对于基站的位置。局部簇首更新机制也需要付出一定的节点存储代价。但节能是无线传感网络的首要考虑因素, 通过一定的代价来延长网络的生命周期还是很有必要的。

接下来的工作是要进一步提高全局最优簇首的精度, 因为粒子群算法不保证能找到全局最优解, 这还需要结合智能算法研究领域来进一步深入研究。另外可以结合其他新型智能算法来研究簇首间的多跳路径, 比如蚁群算法。

参考文献:

- [1] 孙利民, 李建中, 陈渝等. 无线传感器网络 [M]. 北京: 清华大学出版社, 2005.
- [2] 胡钢, 谢冬梅, 吴元忠. 无线传感器网络路由协议 LEACH 的研究与改进 [J]. 传感技术学报, 2007, 20(6): 1391-1396.
- [3] Kennedy J, Eberhart R. Particle Swarm Optimization [C]//Proceeding of The IEEE International Conference on Neural Networks. Piscataway, NJ: IEEE Service Center, pp 1942-1948, 1995.
- [4] Coello C A C, Pulido G T, Lechuga M S M. Handling multiple objectives with particle swarm optimization [C]//IEEE Transactions on Evolutionary Computation, 2004, 8(3): 256-279.
- [5] Wachowiak, M. P., Smolikova, R., Zheng, Y. F., Zurada, J. M., and Elmaghraby A. S.: An approach to multimodal biomedical image registration utilizing particle swarm optimization [J]. IEEE Trans. Evol. Comput., 2004, 8(3): 289-301.
- [6] Fukuyama, Y., Yoshida, H.: A Particle Swarm Optimization for Reactive Power and Voltage Control in Electric Power Systems [C]//Proc. Congress on Evolutionary Computation, Seoul, Korea. Piscataway, NJ: IEEE Service Center, 2001: 87-93.
- [7] Handy M J, Haase M, Timmermann D. Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-Head Selection. 2002.
- [8] 王国芳, 李腊元, 李春林, 刘会静. 无线传感器网络中基于能量约束的簇首多跳算法 [J]. 传感技术学报, 2009, 22(7): 997-1001.
- [9] Wendi B Heinzelman, Anant ha P Chandrakasan, Hari Bal2 akrishnan. An Application-Specific Protocol Architecture for Wireless Microsensor Networks [J]. IEEE Transactions on Wireless Communications, 2002, 1(4): 660-670.
- [10] 张伟华, 李腊元, 张留敏, 王选政. 无线传感器网络 LEACH 协议能耗均衡改进 [J]. 传感技术学报, 2008, 21(11): 1918-1922.
- [11] M. Clerc, in: Discrete Particle Swarm Optimization, illustrated by the Traveling Salesman Problem New Optimization Techniques in Engineering [M]. Springer, 2004: 219-239.
- [12] 田雨波, 朱人杰, 薛权祥. 粒子群优化算法中惯性权重的研究进展 [J]. 计算机工程与应用, 2008, 44(23): 39-41.
- [13] 梁英, 于海斌, 曾鹏. 应用 PSO 优化基于分簇的无线传感器网

络路由协议[J]. 控制与决策, 2006, 21(4): 453-457.

- [14] 过文亮. 无线传感器网络节能路由协议研究[D], 上海大学硕士学位论文, 2009.



范兴刚(1974-), 男, 浙江工业大学计算机科学与技术学院副教授, 工学博士, 研究领域为物联网, 网络通信、实时通信, 分布式实时系统的设计等。
xgfan@zjut.edu.cn;

- [15] Wairagu G. Richard, Extending Leach Routing Algorithm for Wireless Sensor Networks [D], in Msc. Thesis, Makerere University, March, 2009.



侯佳斌(1985-), 男, 浙江工业大学计算机科学与技术学院研究生, 研究领域为无线传感网络路由协议; houjiabin.student@sina.com。