

# The Research of High Efficient and Real Time Data Aggregation Method Applied in WSNs

Kaiqiang Liu, Tao Du, Shouning Qu

School of Information Science and Engineering, University of Jinan, Jinan Shandong  
Email: [kq0902@163.com](mailto:kq0902@163.com), [du\\_t@163.com](mailto:du_t@163.com), [gsn@ujn.edu.cn](mailto:gsn@ujn.edu.cn)

Received: Jun. 19<sup>th</sup>, 2015; accepted: Jul. 7<sup>th</sup>, 2015; published: Jul. 14<sup>th</sup>, 2015

Copyright © 2015 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

In this paper, the advantages and detects of routing and data fusion algorithm are analyzed, and a new method based on n-average value of split is proposed. By using data viscosity, this algorithm could not only sense the changing of environment, but also find the abnormal change. Using the advantages of semi-static clustering in managing sensor nodes, the efficiency of routing is obviously improved, and the energy consumption is more balanced. At last, a series of experiments are designed to prove the performance of this algorithm, and based on the experiment results, the efficiency in data filtering and prolonging networks' lifetime is analyzed.

## Keywords

WSNs, Data Aggregation, Data Fusion, Routing by Semi-Static Clustering, Weight Moving Average

---

# 高效实时的无线传感器网络数据融合算法研究

刘凯强, 杜 韬, 曲守宁

济南大学信息科学与工程学院, 山东 济南  
Email: [kq0902@163.com](mailto:kq0902@163.com), [du\\_t@163.com](mailto:du_t@163.com), [gsn@ujn.edu.cn](mailto:gsn@ujn.edu.cn)

收稿日期: 2015年6月19日; 录用日期: 2015年7月7日; 发布日期: 2015年7月14日

## 摘要

本文分析了现有的路由及数据融合算法的优势与缺陷，提出了采用n阶移动平均的数据融合算法，充分利用节点间的数据黏性，使算法不仅可以跟踪环境变化，而且能够迅速捕捉到不符合趋势的环境变动。算法借鉴了半静态聚簇路由机制在节点管理机制的优点，使得WSN可以以较高的效率对簇内节点进行管理，使能耗更加均衡。最后设计了多个模拟场景的实验方案，将本文提出的算法进行各方面的可行性分析，阐释了本文方案在过滤冗余信息、延长网络寿命上的有效性。

## 关键词

无线传感器网络，数据融合，半静态聚簇路由，加权移动平均

## 1. 引言

无线传感器网络 WSNs (Wireless Sensor Networks) 是一项新型的专用网络，结合无线通信与传感器技术，具有成本低、自组织、容错性好、抗毁性强等特点，在气象检测、火灾报警、地质灾害预警、医疗监控等领域都有广阔的发展前景 [1] [2]。该网络由部署在监测区域内的大量传感器以自组织和多跳的方式构成，以协作方式感知、采集、传输和处理网络覆盖区域内监测对象信息。无线传感器网络无需固定拓扑、连接及节点的数量大，结合多种类传感器，可以对目标区域的感知参数(如声音、光线、温度、湿度、烟雾、震动、压力、红外及磁场等)作综合判断，可以深入到人不能够进入的恶劣环境中，做到无人值守的无缝监控网络覆盖区域的情况。传感器节点的能量供应(通常为自身携带的电池)并不是无限的，其节点的大数量及地理环境上的限制可能会使能源的补充变的不现实，这就决定了无线传感网要有较高的能源效率，因此要避免大量冗余数据的传输，在保持侦测数据准确度的同时减少网络中的数据流量，从而节约能量延长网络寿命 [3] [4]。

数据融合算法将来自不同数据源的相同或近似的数据按照一定规则进行合并的处理过程。由于无线传感网中有大量的感知节点，如图 1 所示，ID 分别为 1、2 和 3 的这三个节点都将侦测到这一事件的发

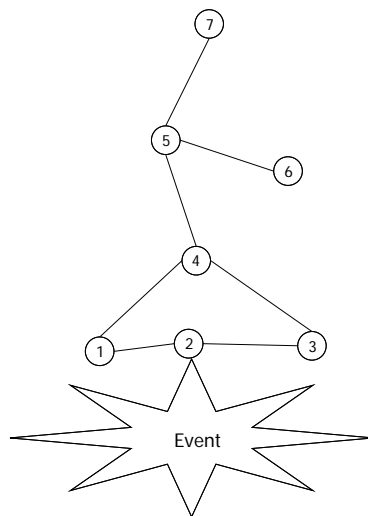


Figure 1. Diagram of multi-node data sense  
图 1. 多节点数据感知示意图

生。如果每个感知到事件的节点都向汇聚节点报告，网络中会充斥着大量冗余无效的数据，不仅浪费能量而且可能会造成网络的堵塞，造成有用的数据不能及时传递至汇聚节点，导致网络失效。数据融合的最终目的是减少在网络中流动的数据包。相关研究表明，无线节点在发送与接收状态的能耗要占到节点总能耗的 70%左右 [5]，因此高效的数据融合算法不仅能够很好的控制网络拥塞问题，更能够提高能量的使用效率，延后网络失效时间。同时，数据融合带来的另一个优势在于可以避免单一节点的出错导致的误判，提高系统的准确性。综上所述，设计一个能够减少数据源端发送、接收次数但同时不会影响网络监控的实效性、准确性的数据融合机制是必要的。

## 2. 系统模型和问题描述

### 2.1. 无线传感器网络的结构

无线传感器网络由大量传感节点(sensor)和唯一的汇聚节点(sink)组成，所有的传感节点被随机地、均匀地散布在一个广阔的区域，汇聚节点负责收集数据并通过卫星通信或互联网将信息汇集到监控中心加以分析。用一个连通的图  $G = (V, E)$  来表示无线传感网，其中集合  $V$  表示这  $N$  个传感节点，每个节点享有唯一的编号， $V = \{v_1, v_2, \dots, v_N\}; |V| = n$ ，集合  $E$  表示节点之间的信息链路，每个传感节点的通信半径均为  $r_c$ ，若满足  $|v_i - v_j| \leq r_c$ ，则有  $e_{ij} \in E$ 。所有传感节点的覆盖区域构成了无线传感网的网络范围，汇聚节点大致位于此网络的中心位置，节点间以多跳、协作的方式将数据传输至汇聚节点，汇聚节点可以通过卫星或者埋设的线路发送至位于互联网的数据中心(Data Center)，网络管理员可以通过互联网实时查看网络区域的情况。

### 2.2. 传感器节点的能耗模型

从图 2 的能量消耗分布图可以看出，节点在生命周期中的能耗集中在通信模块上，其中发送与接收的功耗占节点总功耗的 60%~70% [5]。传感器节点只要还在工作则在数据的感知与基础处理的能量消耗是不可避免的。如果数据融合机制可以有效减少冗余数据的发送与接收次数，就可以大幅减少能量损耗，从而延长网络生命周期，这也是本方案的主要研究目的。

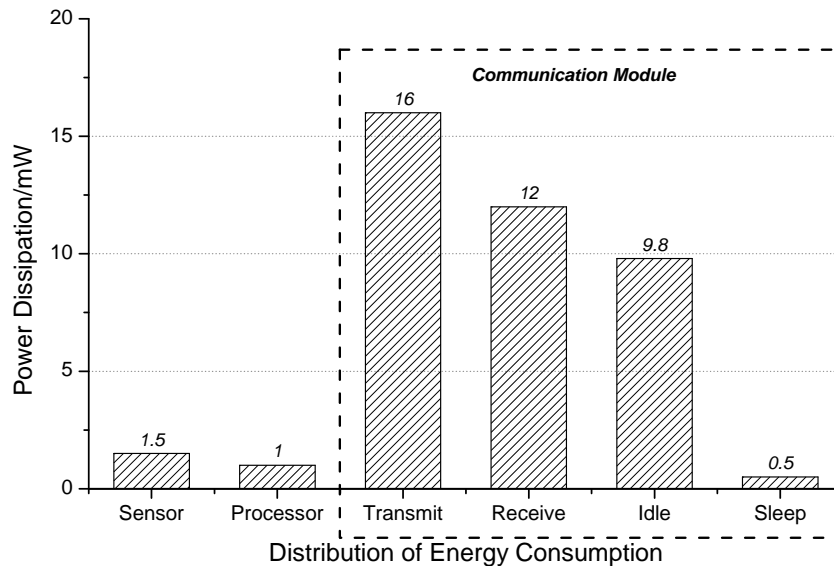


Figure 2. Distribution of energy consumption in WSN nodes

图 2. 传感器节点能量消耗分布

本文参考经典的信息传输能耗模型来计算节点间的通信能耗:

$$E_{TX}(k, d) = \begin{cases} L * E_{elec} + L * \epsilon_{amp} * d^2, & d < d_0 \\ L * E_{elec} + L * \epsilon_{amp} * d^4, & d \geq d_0 \end{cases} \quad (1)$$

$$E_{RX}(k) = L * E_{elec} \quad (2)$$

公式(1)表示节点发送数据的能量消耗,公式(2)表示节点接收数据的能量消耗。 $L$ 表示信息的长度, $d$ 是传输距离, $d_0$ 是传输距离阈值。 $E_{elec}$ 是硬件电路驱动发送机或者接收机运行的参数,它的值受数字编码、调制等因素影响。 $E_{amp}$ 表示信号放大所需要的功率, $E_{elec}$ 和 $\epsilon_{amp}$ 都是硬件相关的传输参数,不受算法的约束。通过公式可知,发送长度为 $L$ 的数据包,消耗的能量只与传输距离 $d$ 有关。当传输距离小于距离阈值 $d_0$ 时,消耗的能量随传输距离的平方增长;当传输距离大于距离阈值 $d_0$ 时,射频能量消耗则随传输距离的4次方增长。公式(1)中第二个等式的高指数能耗公式只有在当网络中的节点失效比较多的后期,单跳距离足够远时才会适用。本文认为网络的最初布设是足够密集的随机布设,网络生命周期的绝大部分时间适用第一个公式。

### 3. 相关工作

#### 无线传感网数据融合算法研究现状与进展

无线传感网处在研究的最前沿,如网络热点问题、能量空洞、数据融合过滤、多路径路由以及自组织结构形成方式等诸多问题 [6] [7]目前尚没有形成国际公认的解决方案。近些年的研究已经提出了不少路由协议和数据融合方案,其中很有代表性的一类数据融合算法是从网络层进行数据过滤与数据融合。这类算法可以分为基于查询 [8]的和基于事件响应 [9]的两种。基于查询的数据融合包括分簇、兴趣扩散以及数据传输三个步骤 [10] [11],数据流量小且能量效率高,但相对于响应式的融合机制有时延较大等缺点。基于响应的数据融合算法因其迅速响应以及较高的冗余数据过滤率,是目前的科研工作的主要方向。

响应式数据融合算法从网络组织结构可分为平面融合结构、树状融合结构与分簇融合结构等 [12]。

第一种基于平面结构的数据融合算法,因为其简单、有效,鲁棒性高等特点赢得广大学者的青睐。该结构需避免“内爆”的冗余数据,同时也存在着不利于设置过滤点,较难形成有效的数据过滤机制的缺点,不适用与大规模的传感网。

第二种基于树结构的数据融合算法一般会包括树结构的生成、维护和数据传送三个阶段 [13]。由于树形结构的单向性限制,导致路由过程不灵活,而且所有的算法对网络能量有效性的调节都是基于路由线路能耗的,容易引起热点问题,而且数据融合点都设置在汇聚节点的周围,易于加剧能量洞问题,导致这些算法的能量有效性较差。

第三种基于分簇的融合算法是一种典型的层次结构的数据融合算法。算法首先要对所有传感节点进行聚簇,在每一个簇中,会选举出一个簇头来负责发送、转发数据包,而簇内成员只需要将数据包交给簇头节点即可。簇头因为额外负责了数据的转发、路由工作,其能耗会相对其他的节点高,因此簇头的选择与拓扑的周期维护则是基于簇结构的数据融合路径的关键所在。这类数据融合算法包括 LEACH、LEACH-C 以及在此基础上改进的 HEED、TEED 和 PEGASIS 等。在 [14]中,低功耗自适应聚类路由算法(LEACH)定义了“轮”的概念,一轮流程包括初始化和稳定工作两个阶段。在初始化阶段节点随机产生一个随机数值,并与算法设置的一个阈值做比较,大于阈值的节点则当选簇头,担任通信中继、数据融合的任务。簇头周围的节点就近加入簇中,完成初始化阶段后进入了稳定工作阶段,一段时间后重新选举簇头以均衡能量消耗。LEACH 算法没有考虑节点剩余能量对簇头的选举的影响,尚有改进的余地。

后续的算法中, PEGASIS 算法针对了节点剩余能量进行改进, 使各个节点能量消耗更均衡 [15]; TEED 算法则设置了软、硬两个阈值实时判断当前数据是否有足量变动, 低于阈值的数据则被抑制不予发送, 减少了数据的发送次数, 但是还是遗传了 LEACH 算法周期性维护结构等的缺点 [10]; 张静在 [16]中提出了基于权值的极小支配集算法(WMDS), 用图论中的独立支配集构成虚拟骨干网, 在此基础上选举支配节点作为簇头, 簇头节点负责将数据融合并路由至汇聚节点, 各节点可以轮流当选支配节点, 以达到节点能量分布均衡的目的, 算法采用的极小支配集策略有效的应对了网络拓扑的变化, 如果网络有局部的变动, 只需要有个别相关节点更新它们的状态, 同时也极大的减少了数据发送的次数, 有效地延长了网络的寿命;许建等学者在 [17]中提出的基于二次独立集的数据融合调度算法采用时分复用思想, 通过构造 2 次最大独立集实现对加权数据的无冲突调度, 为允许通信的链路分配传输时隙, 并根据能量消耗预测调整数据融合路径, 该算法降低了融合的时延, 加权的公平性保证以及延长了网络生命周期。

## 4. 基于 N 阶加权移动平均的数据融合算法

### 4.1. 改进的半静态聚簇路由算法

路由结构作为本算法的次要方面和前提, 在杜韬提出的基于划分的半静态聚簇路由算法 EESSC [18]基础上加以改进。因为此算法将节点剩余能量与信息传输代价引入到聚簇算法中, 较好的保证了网络能量均衡性和聚簇合理性, 并且定制了特殊的簇头轮换机制以及再聚簇机制, 使得能耗更加均衡。

在无线传感网中, 空间距离是衡量能量消耗的最重要的因素欧拉公式是计算空间中任意节点间距离的最常用的方法, 计算公式如公式(3):

$$d_{ab} = \sqrt{(X_a - X_b)^2 + (Y_a - Y_b)^2} \quad (3)$$

在公式(3)中,  $X_a$  和  $Y_a$  分别表示节点  $a$  的  $x$  轴与  $y$  轴坐标,  $X_b$  和  $Y_b$  分别表示节点  $b$  的  $x$  轴与  $y$  轴坐标。计算结果保存在一个  $n$  维三角矩阵中, 我们记作距离矩阵:

$$d[n][n] = \begin{bmatrix} 0 & d_{12} & d_{13} & d_{14} & d_{15} & \cdots & d_{1n} \\ & 0 & d_{23} & d_{24} & d_{25} & \cdots & d_{2n} \\ & & 0 & d_{34} & d_{35} & \cdots & d_{3n} \\ & & & 0 & d_{45} & \cdots & \vdots \\ & & & & 0 & \vdots & \vdots \\ & & & & & \ddots & d_{n-1n} \\ & & & & & & 0 \end{bmatrix} \quad (4)$$

其中,  $d_{ij}$  表示第  $i$  个节点到第  $j$  个节点的距离, 显然此距离矩阵的主对角线元素为 0, 本身的距离度量为零; 如果  $i$  节点与  $j$  节点间不存在有效的数据链路, 则  $d_{ij}$  设置为无穷大。

半静态聚簇路由算法主要分为 4 步 [18]: 首先是网络初始化阶段, 由汇聚发布路由指令, 通知所有节点开始聚簇, 如 图 3; 然后是聚簇阶段, 将 WSNs 的节点按 HAC 方法聚簇, 并且生成簇内节点链接表, 与半静态聚簇路由算法 EESSC 不同, 改进的半静态聚簇路由算法同时需要在簇间形成链接表, 设置簇间转发节点路由信息(见 图 4); 第三步是簇头轮换, 聚簇完成后, 节点按照其在簇内节点链接表内的顺序轮流担任簇头; 最后是重新聚簇, 如果簇内节点链接表中所有的节点都无法满足簇头要求时, 向汇聚节点发送重新聚簇的要求, 由汇聚对节点重新聚簇。

借鉴半静态聚簇路由算法, 形成数据融合的层次结构:

改进的半静态聚簇路由算法使网络在最初阶段每一个节点都是一个簇, 如 图 5 所示, 在结束时得到如 图 4 的拓扑及簇内、簇间的链接关系。节点 3、7、13、16 和节点 17 维护一份簇内的链接关系表, 而

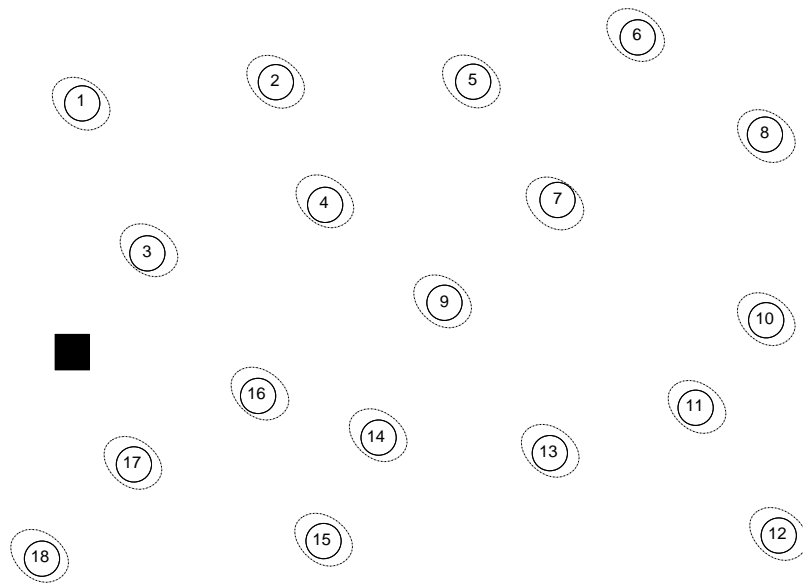


Figure 3. Initialization phase of cluster  
图 3. 聚簇初始阶段

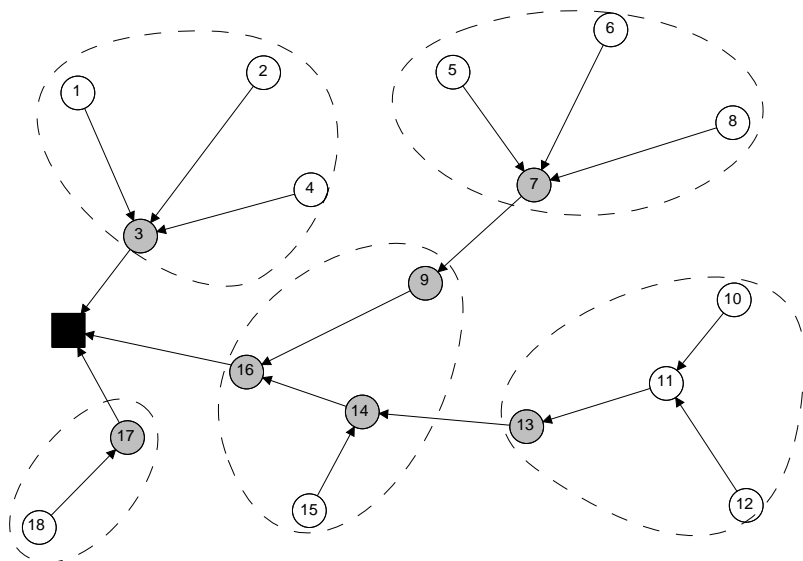


Figure 4. Cluster ending and elect head & deploy forwarding node  
图 4. 聚簇结束、选举簇头和配置转发节点路由

节点 9、节点 14 则维护一份簇间的转发关系表。这样就从宏观上组织起整个网络，作为下一阶段算法正常施行的基础。图 5 所示为网络运行的整体流程，其中的数据融合算法环节将在下一章节阐述，也是本文的核心内容。

#### 4.2. 等长队列的 N 阶加权移动平均融合算法

随着硬件电路的发展，传感节点的硬件本身可以携带的存储空间也在随之增大。注意到传感节点在运算上的能量消耗远小于数据传送的能量消耗，本文提出基于历史数据 n 阶移动平均值的数据融合算法，在末端节点对数据进行充分的冗余过滤，大幅度减少数据包发送。充分利用历史数据对未来事件的预测

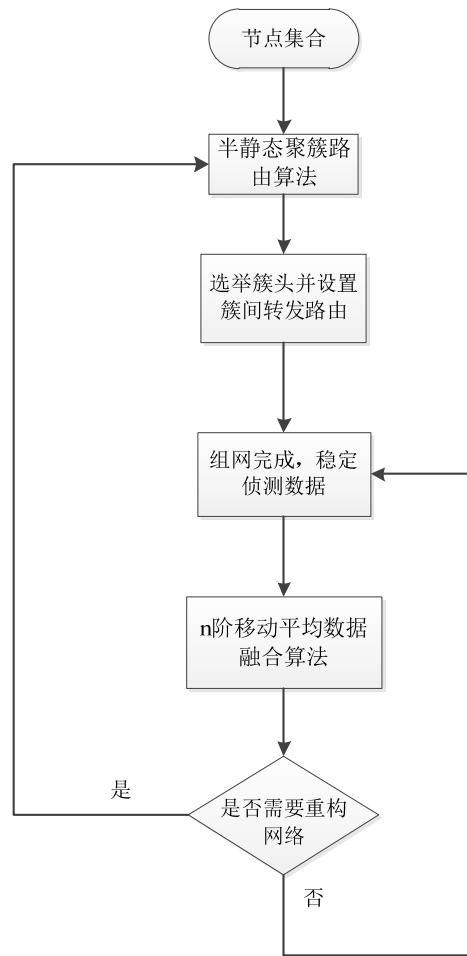


Figure 5. Flow chart of WSN architecture process  
图 5. 传感网整体架构流程

作用，侦测节点上做像素级数据融合，SINK 节点可以综合多组数据进行决策级数据融合，形成响应动作，以最小的数据流量和能量消耗完成消息传递。本文讨论仅有一种传感器的情形，实际中可以扩展为多组传感器，并行运行此算法，在像素级融合后在。

采用移动加权的原因是利用其跟随趋势以减少日志式数据的上报。因为近期数据对未来预测的数据波动的影响最大，因此赋予它较大的权值。常用加权方式分为四种：

- 末日加权移动平均：

$$MA(n) = (C_1 + C_2 + \dots + C_n \times 2) / (n + 1);$$

- 线性加权移动平均线：

$$MA(n) = (C_1 \times 1 + C_2 \times 2 + \dots + C_n \times n) / (1 + 2 + \dots + n);$$

- 梯型加权移动平均：

$MA = [(第1组感测值 + 第2组感测值) \times 1 + (第2组感测值 + 第3组感测值) \times 2 + (第3组感测值 + 第4组感测值) \times 3 + (第4组感测值 + 第5组感测值) \times 4] / (2 \times 1 + 2 \times 2 + 2 \times 3 + 2 \times 4)$  (以5组数据为例)；

- 平方系数加权移动平均：

$MA = [(第1组感测值 \times 1 \times 1) + (第2组感测值 \times 2 \times 2) + (第3组感测值 \times 3 \times 3) + (第4组感测值 \times 4$

$\times 4) + (\text{第 5 组感测值} \times 5 \times 5)] / (1 \times 1 + 2 \times 2 + 3 \times 3 + 4 \times 4 + 5 \times 5)$  (以 5 组数据为例)。

移动平均值法用一组最近的实际数据值来预测未来一期或几期内数据的一种常用的方法。当数据趋势既不快速增长也不快速下降, 且不存在短周期的变化因素时, 移动平均法能有效地消除预测中的随机波动。加权移动平均给固定跨越期限内的每个变量值以相等的权重, 给一个长周期内的各变量以不同的权重, 远离目标期的变量值的影响力相对较低, 应当给予较低的权重, 而最近期的数据最能预示未来的情况, 因而权重应较大。

加权移动平均法的计算公式如下:

$$F_{t+1} = \frac{\sum_{i=1}^t w_i * h_i}{\sum_{i=1}^t w_i} \quad (5)$$

其中  $\sum_{i=1}^t w_i = 1$ , 即  $F_{t+1} = w_1 h_1 + w_2 h_2 + w_3 h_3 + \dots + w_t h_t$ 。在上式中,  $w_i$  为第  $i$  期历史数据  $h_i$  的权重,  $t$  为传感节点存储历史数据的链表长度,  $F_{t+1}$  为预测的下一期数据。

在无线传感器网络中, 传感器感知到的数据多为湿度、温度、风速等环境变量, 且多具有连续属性。如果这些数据不能够以一种较好策略进行数据包筛选, 仅仅是以一种间隔时间段的策略来作日志式发包, 则充斥着大量冗余数据。加权移动平均能够充分挖掘了数据间的趋势关联关系, 不会局限于环境变量的微量步进, 同时又可以跟踪环境的变化规律, 当有环境因素持续快速变动时可以迅速上报。结合在生成树路径上的上报记忆元件, 过滤掉后期相同时间的数据包的传递。

$$w_0 h[0] + w_1 h[1] + w_2 h[2] + \dots + w_{m-1} h[m-1] = h[m] \quad (6)$$

式中  $w_0 + w_1 + w_2 + \dots + w_{m-1} = 1$ ,  $h[m]$  为依据  $m$  个往期数据作出的下一期数据预测,  $w_0$  为最早一期的数据  $h[0]$  的权重,  $w_{m-1}$  为最近一期数据  $h_{m-1}$  的权重。当预估值  $h[m]$  的与实测值的偏差足够大使 ALERT 超过阈值  $F$  时, 即认为当前环境发生了足够剧烈的变化, 具备了立即上报感测数据的资格。ALERT 的计算方法如下:

$$\text{ALERT} = \frac{|H[m] - h[m]|}{h[m-1]} \times 100\% \quad (7)$$

式中  $H[m]$  为传感节点实时侦测到的当前环境变量数值,  $h[m-1]$  为最近一期关注变量的环境实测数值。当侦测完成后传感节点根据历史数据与当前数据的差异计算当前环境是否发生了足够的变化, 如果变化足够剧烈则立即上报数据并将此次数据压入历史数据记忆栈中, 如果没有足够的变化幅度, 则仅执行压栈操作而不启动数据发送程序。图 6 所示为 N 阶加权移动平均融合算法流程。

## 5. 基于 N 阶加权移动平均的数据融合算法

### 5.1. 平方系数加权移动平均的环境数据跟踪

移动加权平均多用在金融、股票预测领域, 因其对近期的数据加以更大的权重, 可以很好的跟踪趋势和预测下期数据得到很广泛的应用。通过实验对比发现采用平方系数加权移动平均, 相比采用线性加权移动、末日加权移动及梯型加权移动能更好的跟踪环境趋势。每个节点维护一个长度为  $n$  的链表(默认设置为  $n = 20$ ), 这个链表保存了最近的  $n$  组感测数据, 基于此链表数据计算当前的移动加权平均作为预测值, 并对告警比率 ALERT 设置一个阈值  $F$ (默认设置为 15%), 当检测到超过此阈值的变动则发送当前数据。

本文中涉及到的时间序列(Time Tick)单位为“次”, 即感知节点感知数据的次数, 而每次感知数据



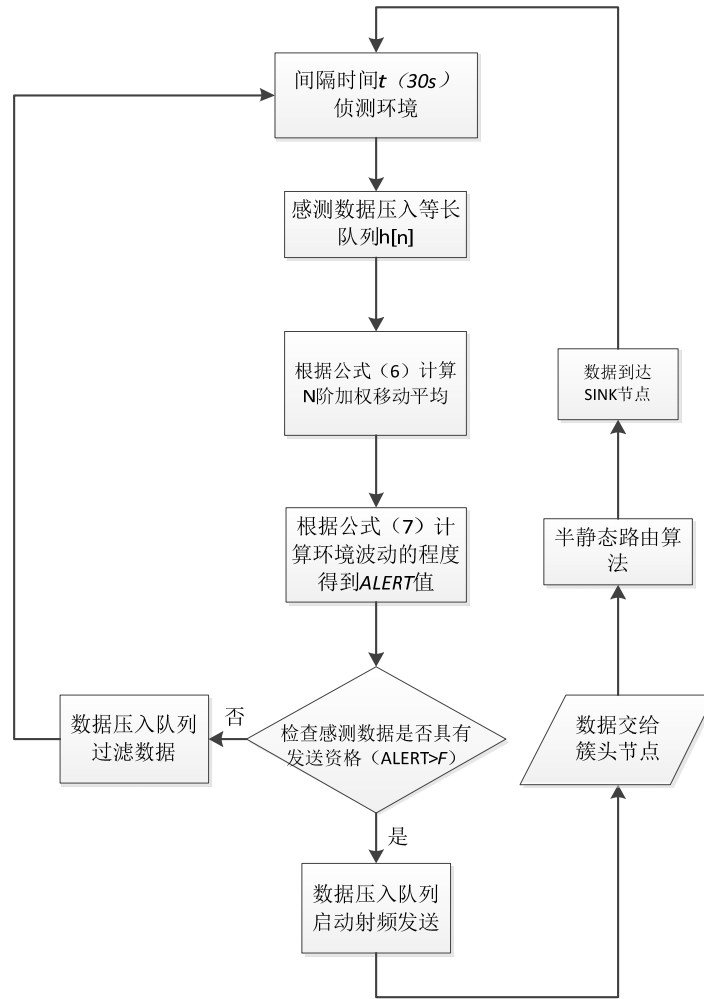


Figure 6. Flow chart of multistage weighting moving average data fusion  
图 6. N 阶加权移动平均数据融合算法流程

的间隔时间(比如 3 秒钟)乘以时间序列次数则为正常时间。其中间隔时间会根据具体环境做不同的设置, 间隔时间越短, 则精确度越高, 网络耗电量将越大。图表中不再单独做标注。实验环境及硬件配置详见附录。

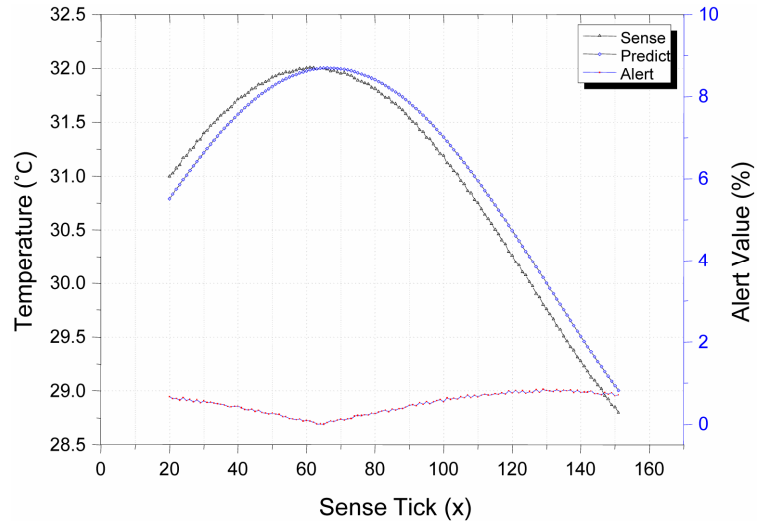
设置  $n = 20$ , 根据公式(7)权重系数  $w_i$  设置如表 1 所示。

图 7 为算法跟踪正常环境缓慢变化的分析图,  $x$  轴为传感节点侦测环境的次数, 每次间隔时间为 5 秒钟(模拟实验时间间隔), 左侧  $y$  轴为环境的温度值, 右侧  $y$  轴为 ALERT 告警比率。Sense 线表示传感节点侦测到的环境值, Predict 线表示算法根据历史数据做出的跟踪预测值, ALERT 线为算法根据公式(7)得到的告警比率。很显然像温度、湿度等这些环境变量正常情况下是在持续的缓慢变化中, 从图中我们可以看到在一个比较长的时间段内, 温度从  $31^{\circ}\text{C}$  升至  $32^{\circ}\text{C}$  后降至  $29^{\circ}\text{C}$ , 算法对环境的跟踪良好, ALERT 值保持在 2% 以下, 不触发信息发送。

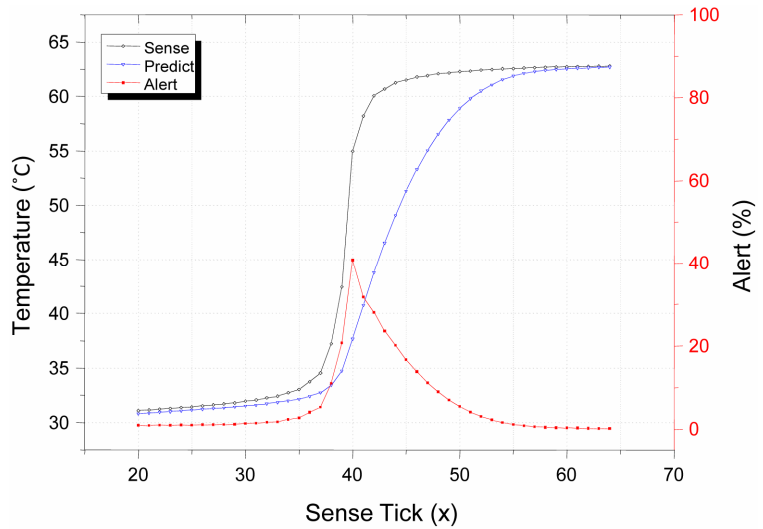
图 8 所示模拟环境失火的情况, 环境温度迅速上升到  $60^{\circ}\text{C}$  以上, 传感节点在一分钟内(约 12 次侦测环境)侦测的快速变化的数据使 Predict 线明显滞后于 Sense 线的变化趋势, 造成 ALERT 比率急剧升高至 40%, 这样的高比率足以触发我们预设的 15%~20% 的告警阈值。在警报发出后, Predict 线会逐渐跟上, ALERT 回落不会再持续的报警, 可以节约传感节点的能量。

**Table 1.** Weight coefficient result  
**表 1.** 权重计算结果

$w_1$	$w_2$	$w_3$	...	$w_{18}$	$w_{19}$	$w_{20}$
1	4	9	...	324	361	400

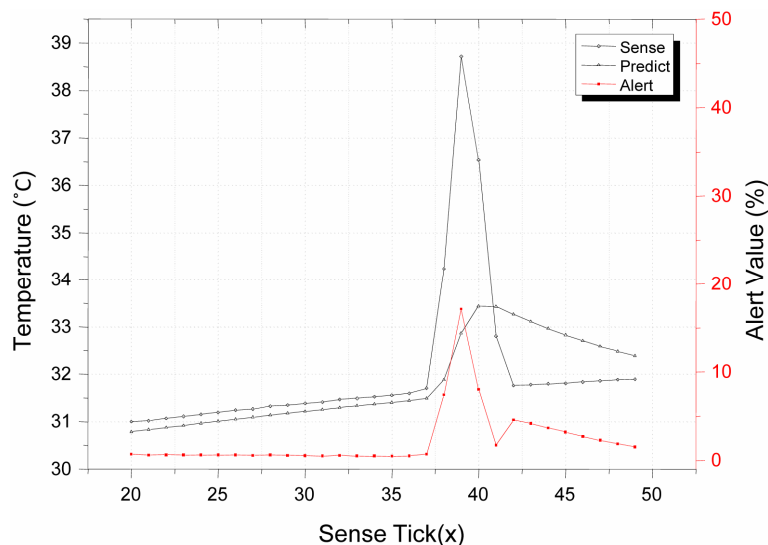


**Figure 7.** Algorithm tracking environments temperature  
**图 7.** 算法跟踪环境温度值的缓慢变化



**Figure 8.** Algorithm responds to temperature dramatic rise  
**图 8.** 算法对环境温度有剧烈跃升时的响应

图 9 所示情景缩小环境温度粒度，适用于需要对温度精确控制的环境。首先要下调告警阈值 ALERT 的值为 10%，可以看到环境只有四个数据(第 38~41 次)明显偏离趋势，迅速从 31.5°C 左右升至近 39°C，则引起了 16%左右的告警值。突破阈值及时并发出消息。通过这个模拟环境说明本算法可以根据环境做一定的自定义精度，当环境要检测诸如室内火情时，适当的上调告警阈值而不会将比如室内空调的开启造成的温度变化误报；当环境需要较高的精度时，下调告警阈值会更及时的报告环境的变化，但相应的会发送更多数据包消耗更多能量。



**Figure 9.** Tracking environment and alarm changes under high-accuracy  
**图 9.** 高精度的环境变化跟踪与告警

在这个情景中，如 图 10，缩小环境的时间粒度，来看节点侦测出错时的处理。当节点在很短的时间有一个或一两个数据出现错误飘移，如图中 Sense 本应是一个平滑的增长，但是因为某种因素导致第 34 次的数据出现错误，而后迅速恢复正常水平。ALERT 线增至 7.5%左右，Predict 线只有较小升幅，算法的预测功能消除了这种随机错误，降低了错误对 ALERT 线的瞬时影响，最终没有引发告警。此情景展示了算法具有一定的容错能力。

## 5.2. 加权移动平均的冗余数据融合与过滤

当环境在一个较短时间出现了如 图 11 的波动,我们可以看到当下环境中明显的三次波动,则视为此节点在这段时间内需要发送三个消息包,即算法在理想状态下感知到此环境的节点仅会作出三次数据上报动作。

图 12 显示了算法对具有三次波动的环境做出了及时的跟踪及告警。图 13 展示当算法设置不同的 ALERT 值会对冗余数据的过滤产生影响。ALERT 值设置的越高,则过滤的数据则越多,造成数据漏报的几率就会越大;ALERT 值设置的越低,则算法的告警精确度会越高,数据漏报的几率就越小。在这个实例中,当 ALERT 值大于 50%时会造成数据漏报,ALERT 的值在 10%~25%之间都是合适的。ALERT 在某种程度上指示了环境波动的剧烈性,在这个取值以内不同剧烈程度的波动均可以保持在极低的数据漏报率,因此也可以认为是算法对环境异常具有一定的鲁棒性。

以上分析主要针对的是单个节点对数据的融合、过滤,当同一个聚簇中的多个节点同时感知到此波动时,可以在簇头节点设置过滤点,过滤掉空间距离较近的多节点的冗余数据,在此不作为重点考虑。

## 6. 结论

本文分析了现有的路由及数据融合算法的优势与缺陷,提出了采用  $n$  阶移动平均的数据融合算法,在基于划分的半静态聚簇路由由算法的基础上,设计了一套能量节约的数据融合算法,保证了聚簇结果的合理性,充分利用节点间的数据黏性,使算法不会陷于步进跟踪环境变化,而且仍能够迅速捕捉到不符合趋势的环境变动。根据实验数据的分析,在正常的环境下,算法能够密切跟踪当前的趋势,能够过滤掉几乎所有的不必要的正常发包;而在事件发生时,如果阈值设置合理,则能够过滤 95%以上的冗余数

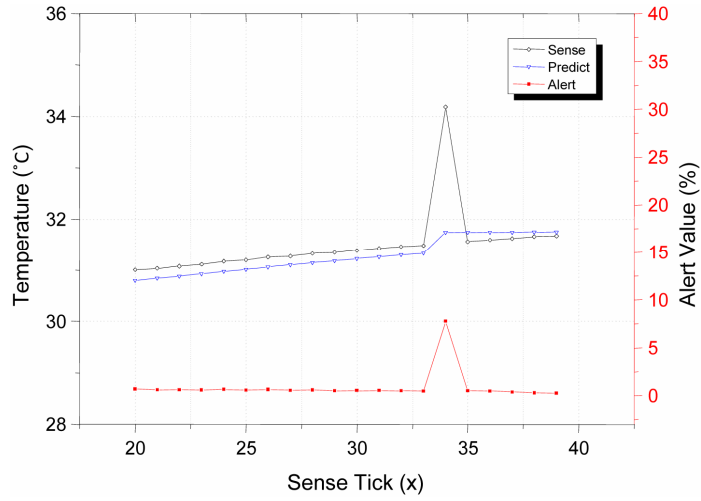


Figure 10. Robustness to individual errors  
图 10. 节点对个别误报数据的容错

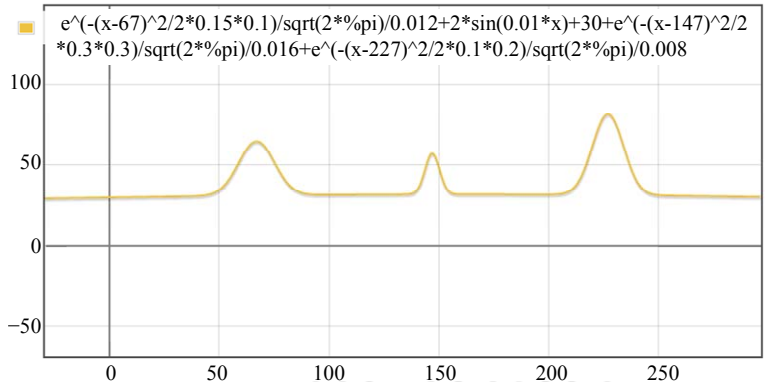


Figure 11. Three dramatic jump in this environment  
图 11. 具有明显的三次波动的环境

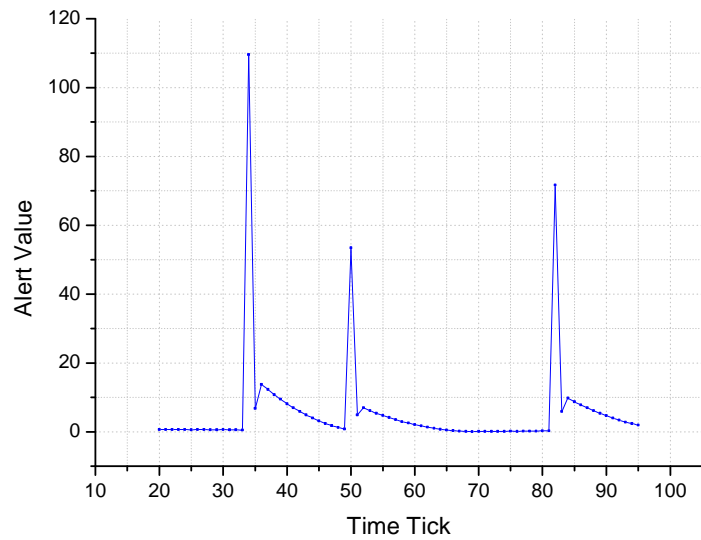


Figure 12. Alert changes along with complex environment  
图 12. 对复杂环境变化的 ALERT 值跟踪

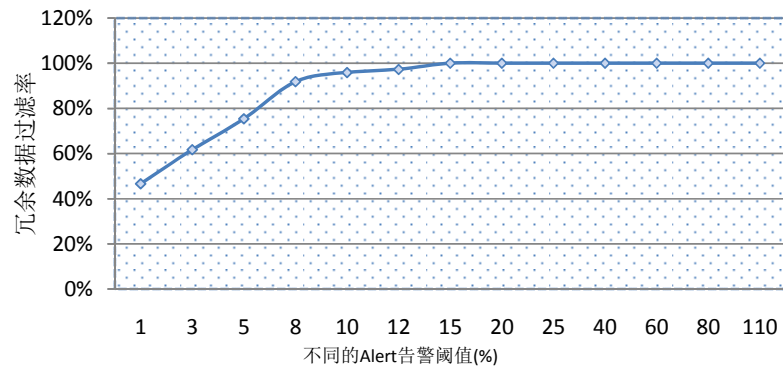


Figure 13. Different alert threshold have different redundant data filtering ratio  
图 13. 不同 ALERT 阈值的选取对冗余数据的过滤的影响

据。算法还具有一定的抑制偶然错误的机制，由于利用了历史数据的平滑作用，得以在一定程度上能够抑制由于传感器在随机错误造成的误判。

## 致 谢

论文在撰写期间始终得到导师曲守宁教授和杜韬博士在学术上的认真指导，使得本课题的研究能够顺利进行并取得了阶段性成果。在论文的研究和撰写过程中，导师倾注了大量的心血，特别是导师严谨的治学态度、对学术孜孜不倦的探索精神、鼓励创新的开拓风范，以及他长者的博大胸襟都将使作者终生受益。借此论文成稿之际，谨向导师曲守宁教授和杜韬博士致以崇高的敬意和衷心地感谢！

## 参考文献 (References)

- [1] 陈正宇, 杨庚, 陈蕾, 许建 (2011) 无线传感器网络数据融合研究综述. *计算机应用研究*, **5**, 1601-1604+1613.
- [2] 黄春燕 (2013) 无线传感器网络下的数据融合研究. 硕士论文, 湖南工业大学, 株洲.
- [3] 王金萍 (2013) 传感异构数据采集及数据融合技术的研究与实现. 硕士论文, 南京邮电大学, 南京.
- [4] 傅海波 (2013) 无线传感网中数据融合算法的研究. 硕士论文, 南京邮电大学, 南京.
- [5] 郭新 (2013) 无线传感器网络路由协议及数据融合技术研究. 博士论文, 华南理工大学, 广州.
- [6] Rehman, A., et al. (2011) A review of wireless sensors and networks' applications in agriculture. *Computer Standards & Interfaces*.
- [7] Villas, L.A., et al. (2011) A spatial correlation aware algorithm to perform efficient data collection in wireless sensor networks. *Ad Hoc Network*, **12**, 10-30.
- [8] Villas, L.A., et al. (2013) An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks. *Computer Communications*, **36**, 1054-1066. <http://dx.doi.org/10.1016/j.comcom.2012.04.007>
- [9] Ozdemire, S. (2008) Functional reputation based reliable data aggregation and transmission for wireless sensor networks. *Computer Communications*, **31**, 3941-3953. <http://dx.doi.org/10.1016/j.comcom.2008.07.017>
- [10] J.N. Al-Karaki, Ul-Mustafa, R. and Kamal, A.E. (2009) Data aggregation and routing in wireless sensor networks: Optimal and heuristic algorithms. *Computer Networks*, **53**, 945-960. <http://dx.doi.org/10.1016/j.comnet.2008.12.001>
- [11] Bagaa, M., et al. (2012) Efficient data aggregation with in-network integrity control for WSN. *Journal of Parallel and Distributed Computing*, **72**, 1157-1170. <http://dx.doi.org/10.1016/j.jpdc.2012.06.006>
- [12] Fateh, B. and Govindarasu, M. (2013) Energy minimization by exploiting data redundancy in real-time wireless sensor networks. *Ad Hoc Network*, **11**, 1715-1731. <http://dx.doi.org/10.1016/j.adhoc.2013.03.009>
- [13] Yu, X., Wu, P., Han, W. and Zhang, Z. (2013) A survey on wireless sensor network infrastructure for agriculture. *Computer Standards & Interfaces*, **35**, 59-64. <http://dx.doi.org/10.1016/j.csi.2012.05.001>
- [14] Heinzelman, W.R., Chandrakasan, A. and Balakrishnan, H. (2000) Energy efficient communication protocol for wireless microsensor networks. 2000 *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*,

- 4-7 January 2000. <http://dx.doi.org/10.1109/hicss.2000.926982>
- [15] Lindsey, S. and Raghavendra, C.S. (2002) PEGASIS: Power efficient gathering in sensor information systems. 2002 *IEEE Aerospace Conference Proceedings*, **3**, 1125-1130. <http://dx.doi.org/10.1109/aero.2002.1035242>
- [16] Zhang, J. and Jia, C.-F. (2009) A routing algorithm based on minimal dominating set with weight for wireless sensor networks. *Chinese Journal of Sensors and Actuators*, **22**, 1784-1788.
- [17] 许建, 杨庚, 陈正宇, 王海勇, 杨震 (2014) 基于二次独立集的数据融合调度算法. *通信学报*, **1**, 62-71.
- [18] 杜韬 (2014) 能量有效的无线传感器网络路由算法研究. 硕士学位论文, 山东师范大学, 济南.

## 附 录

**Table A.** Lab's condition parameter table  
**表 A.** 实验环境一览表

项目	配置描述
操作系统	Windows 7ultimate--64bit
eclipse 版本	Luna Release (4.4.0)
java 虚拟机	jre 1.7.0_60
硬件配置	AMD Phenom II, 4G 内存, 500G 存储空间
数据分析	linux version 2.6.32-504.1.3.el6.i686 gcc version 4.4.7 (Red Hat 4.4.7-11) NS 3.21