

# 一种基于簇树结构的 ZigBee 网络最大生存期方法研究

陈 宁, 徐显秋

(重庆科技学院, 重庆 401331)

**摘 要:**提出了一种基于簇树结构的 ZigBee 网络最大化生存时间的方法,该簇树主要针对实时传感应用,且每个数据流应确保端到端的生存期。该方法将全网范围内的优化问题分解成每个簇的优化问题来解决,找到了一种低复杂度的优选方案。

**关键词:**最大生存期;簇树;ZigBee 网络

**中图分类号:**TP393

**文献标志码:**A

## 引 言

ZigBee 是一种短距离、低速率的无线传感器网络的技术标准,其 PHY 层和 MAC 层协议为 IEEE802.15.4 协议标准<sup>[1]</sup>。主要特性是低速率、近距离、低功耗、低复杂度和低成本。

在家用和工业方面的实时传感控制领域,ZigBee 簇树结构网络是一种兼顾成本和效率的选择。在本文中,主要针对基于簇树结构的 ZigBee 网络,在确保端到端传感控制报文的生命周期前提下,如何最大化网络的生存时间。

一个簇树网络包括多个簇,每个簇有簇头、边界节点和普通节点。簇头每隔一定时间通过广播信标的方式来同步簇中的各节点。信标间歇期间:为了节省能耗,只有超帧阶段,簇中节点和簇头进行通信,其余时间处于休眠状态。作为边界节点,在连接的各个簇中均要处于激活状态。一旦给定簇树结构的网络,数据流的传输路径就被该树的结构所确定下来。至于如何优化簇树结构的问题不在本文探讨的范围之内,这里仅以一个预先定义好的簇树结构网络,来研究给定簇树网络中 ZigBee 参数的优化配置,主要包含如下内容:

(1) 通过对 ZigBee 参数的优化,在保证端到端实时

流数据的生命周期前提下,使整个网络的生存时间最大化。

(2) 对优化问题建模,尽可能考虑了节点的各种特性,如电能、节点类型等。

(3) 通过将全局性的优化问题分解为基于簇的一系列本地问题,提出了一种启发式算法,以低复杂度优化配置 ZigBee 参数。

## 1 问题描述

IEEE802.15.4 发布的簇  $C_k$  中  $BI_k$  和  $SD_k$  的结构由如下公式定义:

$$\left. \begin{aligned} BI_k &= aBSFD \cdot 2^{BO_k} \\ SD_k &= aBSFD \cdot 2^{SO_k} \\ 0 \leq SO_k \leq BO_k \leq 14 \end{aligned} \right\} \quad (1)$$

在  $SD$  中,对于非实时数据,竞争访问期(CAP)以基于竞争的方式被访问,对于实时数据而言,自由竞争期(CFP)被分派给保证时间间隙(CTSs)节点。为了给非实时数据报文提供服务,给 CAP 预留载荷时,GTSs 被分配在  $SD_k$  之外。

簇实际上是一个逻辑节点组,不同簇中两个节点有可能相互干扰<sup>[2]</sup>。为了避免这种簇间冲突,多个簇中的  $BI$ s 和  $SD$ s 要很好的安排以免交叠。这种安排在一些文

收稿日期:2013-03-17

基金项目:重庆市石油与天然气学会资助项目(KJ2011004)

作者简介:陈宁(1973-),男,重庆人,工程师,硕士,主要从事无线传感网方面的研究,(E-mail)ningchen660@sina.com

献<sup>[3-6]</sup>中有涉及。然而,在 ZigBee 簇树网络中,满足实时性的前提下,怎样优化配置  $BI_s$  和  $SD_s$  仍然是个难题。那么我们要解决的问题就是对于一个给定的含有  $K$  簇的簇树网络和数据流集  $F$ ,通过优化各簇中的  $BI_k$  和  $SD_k$  以满足端到端的生存期条件。同时,在考虑能量储备和每个节点类型情况下,使网络的生存时间最大化。

先定义每个节点的能耗,节点的能耗被它所属簇的负载循环所确定,即  $SD/BI$ 。由于节点可能属于多个簇,如边界节点,将连接到所有簇的负载循环相加( $= \sum_{C_k \in \text{Cluster}(n)} \frac{SD_k}{BI_k}$ ,见表 1)。考虑到每个节点的电能不同,基于电能对节点的能耗做了规范化处理。我们尽量将节点能耗的最大值变小以延长整个系统的生存时间。如果任一节点能源耗尽,整个系统会受影响。这样,得到度量值  $M_{power}$ :

$$M_{power} = \max_{\forall \text{node } n \in N} \sum_{C_k \in \text{Cluster}(n)} \frac{SD_k}{BI_k} \cdot \frac{1}{power_n} \quad (2)$$

由此,优化问题即转变为最小化  $M_{power}$  的问题。其中约束条件如下:

条件 1:对于每个数据流  $f_i$ ,最坏情况下,端到端时延应当小于等于它的生存周期。

条件 2:为避免簇间冲突,所有  $(BI_k, SD_k)$  对,应按照无  $SD$  交叠来安排。

条件 3:确保非实时数据包的 CAP 服务。

表 1 公式中的符号释义

符号	释 义
$N$	节点集
$\text{Cluster}(n)$	节点 $n$ 所属的簇集
$C_k$	第 $k$ 簇
$GTS_n^k$	在 $C_k$ 中,分派到节点 $n$ 的 GTS 时隙数
$CAP_k$	在簇 $C_k$ 中,CAP 服务的时隙数
$MIN\_CAP_k$	在簇 $C_k$ 中,所求 CAP 的最小时隙数
$U_k^{CAP}$	在簇 $C_k$ 中,CAP 服务的载荷
$U_k^{GTS}$	在 $C_k = \sum_{\forall f_i \in F_k} \frac{b_i/R}{p_i}$ 时的 GTS 载荷
$U_k$	在 $C_k = U_k^{CAP} + U_k^{GTS}$ 时的所有载荷
$Power_n$	节点 $n$ 的电能
$f_i$	第 $i$ 个实时流(由 $src_i, dst_i, p_i, b_i, D_i$ ) 表达。
$d_i^k$	在簇 $C_k$ 中,本地化的每簇生存期
$Route_i$	沿着 $f_i$ 路径的簇集
$F_k$	沿着簇 $C_k$ 传播的流集
$WD_i$	最坏情况下 $f_i$ 的端到端时延
$WD_i^k(BI_k)$	给定 $BI_k$ 值,在 $C_k$ 中 $f_i$ 最坏情况下的一跳时延

## 2 算法分析

对于给定的含  $K$  个簇的簇树结构的网络,遍历所有  $((BI_1, SD_1), \dots, (BI_k, SD_k))$  的可能组合以找到最优方

案,但在  $O(15^K)$  的高复杂度下,造成了可量测性问题。为解决全局无遗漏搜索的问题,本文提出了一种启发式算法,在明显减少复杂度的前提下,同时提供最优解决方案。其基本理念是将全局问题“智能”分解成如下的若干本地问题:

第一步,将属于多个簇的簇内边界节点分解成簇内的虚节点集。

第二步,将端到端的生存期分解成基于簇的单跳生存期集。

第三步,估算每簇最坏情况单跳时延。

第四步,基于上述步骤为每个簇选择  $BI$  和  $SD$ 。

由于本地化,可以优化配置每个簇  $C_k$  的  $BI_k$  和  $SD_k$ 。该启发式算法的复杂度为  $O(15 * K)$ ,与  $O(15^K)$  相比复杂度明显降低。

### 2.1 边界节点分解

将簇内的边界节点分解成一个虚节点集,并使每个虚节点仅属于一个簇。原边界节点的能量根据簇的载荷按比例分成了两部分,如公式(3)。每部分分派给新创建的虚节点。

$$power_{v^k} = power_n * \frac{U_k}{\sum_{\forall v^j \in V_n} U_j} \quad (3)$$

公式(3)中,  $V_n$  是边界节点  $n$  的虚节点集,  $v^k$  属于簇  $C_k$  的虚节点集  $V_n$ 。

### 2.2 端到端的生存期分解

对于给定的数据流  $f_i$ ,把端到端的生存期  $D_i$  分解到基于簇的本地生存期中。基本策略是分派较长的本地生存期给一个高载荷、低能量的簇。通过这样处理,可以获得大  $BI_k$ ,该值可以减少“被浪费的时隙部分”,接着减少负载循环,  $\frac{SD_k}{BI_k}$ 。这就是“生存期能量平衡”。在 ZigBee 系统中,分派带时间资源的每个节点如多时隙一样,没有连续的时间周期。这个特性导致了分派的一些时隙被浪费,以致未真正用到传输中,特别是当  $BI$  尺寸很小时<sup>[7]</sup>。生存期划分的详细过程如下:

(1) 计算权重  $w_k$ ,对于每簇中考虑了载荷和能量的  $w_k$  如下:

$$w_k = \alpha \cdot \frac{U_k}{\text{ClusterPower}_k} + \beta \quad (4)$$

式(4)中簇的能量 ( $\text{ClusterPower}_k$ ) 定义成所有节点中(包括虚节点)的最小能量级别。如果  $\alpha \gg \beta$ ,意味着本地生存期完全成比例的将载荷和能量进行了分派。反之,如果  $\alpha \ll \beta$  则不管每簇的载荷和能量,直接进行平均分派。本文中,设定  $\alpha = \beta$  以适应所有策略。

(2) 计算每簇的生存期,  $d_i^k$ , 该值与前面的权重因素  $w_k$  有关:

$$d_i^k = D_i * \frac{w_k}{\sum_{\forall C_j \in Route_i} w_j} \quad (5)$$

### 2.3 最坏时延估算

最大端到端时延, 即数据流  $f_i$  中的  $WD_i$  通过添加最差单跳时延来获取, 每簇中的  $WD_i^k (BI_k)$  沿着路径呈现。由于对簇  $C_k$  而言, 最坏情况下的单跳时延受  $BI_k$  影响, 所以它是关于  $BI_k$  的函数。在计算  $WD_i^k (BI_k)$  时, 考虑这样的场景, 在  $C_k$  中, 数据流  $f_i$  的多个报文没有一次能全部确定下来。在这种情况下, 刚开始的前  $q$  个报文最差情况下的时间  $w_i(q)$  按如下公式计算:

$$w_i(q) = \left\lfloor \frac{q \cdot b_i/R}{S_i} \right\rfloor \cdot BI_k, \text{ 此处 } S_i = \frac{b_i/R}{p_i} BI_k \quad (6)$$

公式(6)含义如下: 传输  $q$  个数据报文的时间是  $q \cdot b_i/R$ , 在每个  $BI_k$  中  $S_i$  被提供给数据流  $f_i$ ,  $\left\lfloor \frac{q \cdot b_i/R}{S_i} \right\rfloor$  就是要求的  $S_i$  的数目。对于有着  $S_i$  数目的数据流  $f_i$  的整个最坏时间是  $\left\lfloor \frac{q \cdot b_i/R}{S_i} \right\rfloor \cdot BI_k$ 。由于第  $q$  个数据报文到达时间是  $(q-1) \cdot p_i$ , 相对于它的到达时间的一跳时延就是  $w_i(q) - (q-1)p_i$ 。在  $q=1, 2, \dots$  所有可能值时,  $WD_i^k (BI_k)$  是  $w_i(q) - (q-1)p_i$  中的最大值。已有文献证明了在  $q=1, 2, \dots, Q$  时, 这个最大值存在。  $Q$  是满足  $w_i(q) \leq Q \cdot p_i$  的第一个整数。  $WD_i^k (BI_k)$  按如下公式计算:

$$WD_i^k (BI_k) = \max_{q=1, 2, \dots, Q} w_i(q) - (q-1)p_i \quad (7)$$

最后, 计算最坏情况下的端到端时延  $WD_i$ , 通过沿着数据流  $f_i$  的路径添加  $WD_i^k (BI_k)$ 。如果  $WD_i \leq D_i$ , 所有  $f_i$  中的数据包可以保证在生存期内到达目的地。

### 2.4 BI 和 SD 选择

一旦确定了每个数据流的每簇生存期集, 对于每个簇  $C_k$  而言, 就可以通过检验, 获取本地簇中最佳  $BI_k$  值。对于每个簇, 我们考虑了 15 种  $BI$  值的可能性, 并计算了相关的  $SI$  值。得到给出最小  $\frac{SD_k}{BI_k}$  值的最佳  $BI_k$  值, 即当保持最坏情况下单跳时延小于每个流  $f_i$  的每簇本地生存期时, 该流经过了  $C_k$  (例如,  $WD_i^k (BI_k) < d_i^k$ )。如果没有  $BI_k$  值满足这个本地条件,  $BI_k$  就设置为最小值 (如果超过一个簇,  $BO_k$  在公式(1)中应该比 0 大)。

一旦  $BI_k$  确定,  $SD_k$  就能计算出来。首先, 对于属于  $C_k$  的节点  $n$ , 为其服务的 GTS 流量,  $GTS_n^k$  按公式(8)计算。  $GTS_n^k$  是时隙长度的整数倍数, 时隙长度  $SlotLen_k =$

$SD_k/16$  (在  $SD$  中的时隙数是 16, 按照 IEEE 规范<sup>[1]</sup>)。

$$GTS_n^k (SD_k) = \left\lceil \frac{\sum_{\forall j \in travel_n} \frac{b_j/R}{p_i} \cdot BI_k}{SlotLen_k} \right\rceil \quad (8)$$

此外, 对于非实时数据包,  $SD_k$  必须包含  $CAP_k$ 。  $CAP_k$  也是时隙的整数倍, 要求的  $CAP_k$  所需要的时隙最小数按如下公式求出:

$$MIN\_CAP_k (SD_k) = \left\lceil \frac{U_k^{CAP} \cdot BI_k}{SlotLen_k} \right\rceil$$

这个时隙数目,  $T(SD_k)$  在公式(9)中被定义, 其值不大于 16。  $SD_k$  由满足公式(9)的最小  $SD_k$  值确定 (一旦  $SD_k$  被公式(9)确定,  $CAP_k$  的值通过  $16 - \sum_{n \in C_k} GTS_n^k (SD_k)$  计算而确定):

$$T(SD_k) = MIN\_CAP_k (SD_k) + \sum_{n \in C_k} GTS_n^k (SD_k) \leq 16 \quad (9)$$

## 3 测试评估

评价本文提出的启发式算法基于两个参考标准: (1)可行性; (2)负载循环。其详细的评估网络和参数设置描述, 见表 2。

表 2 测试场景: 4 个簇, 11 个节点, 7 个数据流

参数		设置
簇树结构	传输率 R	250kbps
	aBaseSuperframeDuration	15.76msec
	$U_k^{CAP} (\%)$	[0, 10%] 范围随机取值
数据流 $f_i$	报文尺寸 $b_i$	[10, 500] bits 范围随机取值
	周期 $p_i$	[1, 5] sec 范围随机取值
	生存期 $D_i$	[10, 50] sec 范围随机取值

为检测本地化优化方法对可行性的影响, 模拟了基于表 2 的 12 658 个随机测试集, 并针对提出的启发式算法计算了能找到满足时序和生存期条件的解决方案的测试集数目<sup>[8]</sup>。根据整个系统载荷, 将这个集分成两类, 即 LOW\_Load 和 HIGH\_Load。在模拟过程中, 当整个载荷少于 50% 时, 提出的启发式算法 99% 能找到 BI 和 SD 的组合解决方案 (例如, 7024 个测试集中有 6956 个)。另一方面, 可行性在高载荷系统中明显降低, 对于大部分 ZigBee 应用而言这种情况不常见。表 3 和图 1 是部分实验结果。

表 3 方案比较: 可行性

	系统载荷 $\alpha = \sum U_k$	
	低载荷 ( $\alpha < 50\%$ )	高载荷 ( $\alpha > 50\%$ )
整个集数	7024	5634
启发式	6956	3570

为检验所提出算法在平衡每个节点的能耗方面的

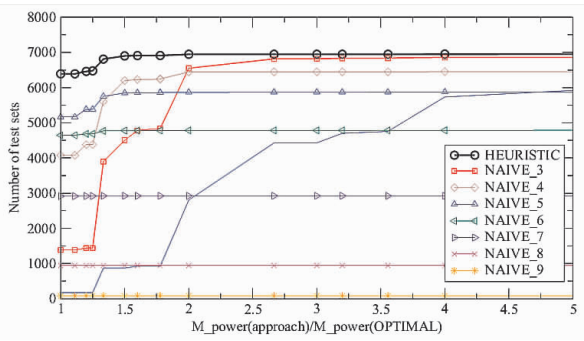


图 1  $M_{power}$  的比较

优势,考虑了每个节点不同电能的情况,图 1 基于  $M_{power}$  对 7024 个低载荷测试集进行了评测。与基准线方法类似,我们提出了图示为 NAIVE\_0, ..., NAIVE\_9 的方法,它们分派了同样的  $BI$  值给所有簇,但未考虑个别流功能和每个节点的电能。例如,NAIVE\_0 将  $BO_k = 0$  分派给所有簇。在图中,  $x$  轴表示  $M_{power} (= \frac{M_{power}(each\_approach)}{M_{power}(OPTIMAL)})$ ,  $y$  轴是测试样本数目。对于 94% 的样本而言,在电能消耗的优化能力方面,启发式与优化解决方案一样,显示出了同样的水准。另一方面,原有方法与启发式方案相比,其能耗较差。

#### 4 结束语

本文提出了一种启发式方案以优化配置 ZigBee 网络簇树的参数,在确保所有实时周期数据流的生存期前提下,达到最大化网络生存时间。实验研究表明该方案在低载荷测试集的情况下,在最大化网络生存时间时,能提供时序化、满足生存期的解决方案。

## Maximum Lifetime Research of ZigBee Networks Based on Cluster-Tree

CHEN Ning, XU Xian-qi

(Chongqing University of Science & Technology, Chongqing 401331, China)

**Abstract:** A method to maximize the lifetime of a ZigBee network based on cluster-tree is proposed, under the constraint that every data flow should deterministically meet its end-to-end deadline for real-time sensing applications. It decomposes the network-wide optimization problem into a set of per-cluster optimization problems, and hence a optimal solution with a low complexity is found.

**Key words:** maximum lifetime; cluster-tree; ZigBee Networks

#### 参考文献:

- [1] ZigBee-Alliance. ZigBee/Specifications [EB/OL]. <http://www.zig-bee.org/>,2012.
- [2] 孙 静.基于 ZigBee 的簇树协议分析[J].网络与信息, 2010,24(11):50.
- [3] 史 品,郑 萍.基于 ZigBee 的 PLC 无线数据采集系统设计[J].西华大学学报:自然科学版,2010,29(6):28-31.
- [4] 乐英高,曹 莉,曾黄麟.基于 ZigBee 和 MSP430 无线温度控制系统设计[J].四川理工学院学报:自然科学版,2012,25(1):52-55.
- [5] Koubaa A, Alves M, Attia M. Collision-free beacon scheduling mechanisms for IEEE 802.15.4/Zigbee cluster-tree wireless sensor networks[C]// Luis M.7th International Workshop on Applications and Services in Wireless Networks (ASWN), Cantabria, Spain, May 24-25,2007:1-8.
- [6] Koubaa A,Alves M,Tovar E.GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks [C]// IEEE. Proc.Of 20th International Parallel and Distributed Processing Symposium, Rhodes Island, Greece, April 25-29,2006:1-8.
- [7] Han J,Choi S,Park T.Maximizing lifetime of cluster-tree Zigbee networks under end-to-end deadline constraints [J].Communications Letters,IEEE,2010,14(3):214-216.
- [8] Casey P R,Tepe K E,Kar N.Design and implementation of Testbed for IEEE 802.15.4 (Zigbee) performance measurements[J].EURASIP Journal on Wireless Communications and Networking,2010(S):160-170.