

文章编号: 1001-0920(2007)09-0967-05

无线传感器网络 TDMA 调度的能量-时延 Pareto 优化

毛剑琳^{1,2}, 吴智铭¹

(1. 上海交通大学 自动化系, 上海 200030; 2. 昆明理工大学 自动化系, 昆明 650093)

摘要: 针对多到一数据传输模式的无线传感器网络, 提出了多目标 TDMA(时分多址)调度优化模型, 考虑了数据包的时延和节点状态切换导致的能量消耗, 合理地建立了 TDMA 调度问题和进化搜索算法间的映射关系, 并设计了基于微粒群的 Pareto 优化算法. 仿真实验表明, 该算法可以有效地找到一组能量和时延目标的 Pareto 优化解, 其结果优于图着色算法.

关键词: 无线传感器网络; 时分多址; 微粒群优化算法; 多目标优化; Pareto 优化

中图分类号: TP393.1; TP18 **文献标识码:** A

Energy-delay Pareto optimization in TDMA scheduling for wireless sensor networks

MAO Jiarlin^{1,2}, WU Zhiming¹

(1. Department of Automation, Shanghai Jiaotong University, Shanghai 200030, China; 2. Department of Automation, Kunming University of Science and Technology, Kunming 650093, China. Correspondent: MAO Jiarlin, E-mail: km_mjl@yahoo.com.cn)

Abstract: In wireless sensor networks with many-to-one transmission mode, a multi-objective time division multiple access (TDMA) scheduling model is presented, which concerns about the packet delay and the energy consumed on node state transition, and a mapping between the scheduling problem and evolutionary algorithm is reasonably set up. A Pareto optimization algorithm based on particle swarm optimization (PSO-Pareto) is proposed. The computational results demonstrate that PSO-Pareto algorithm can effectively find the delay and energy Pareto solutions, and is superior to other techniques in the literature.

Key words: Wireless sensor networks; Time division multiple access; Particle swarm optimization; Multi-objective optimization; Pareto optimization

1 引言

由于在国防、商业和科学研究等方面具有巨大的应用潜力,无线传感器网络(WSN)已成为研究领域的热点^[1]. 在这类网络中,节点具有感知环境、计算和小范围通讯的能力,可以自动地收集信息并将结果发送到基站节点. 由于在大多数应用中,传感器网络节点使用电池供电,这令网络的生存时间受到限制. 因此,如何降低网络的能量消耗成为这个领域研究的主要目标之一^[2,3].

在实时数据采集的传感器网络中,时间是不可忽略的另一个性能指标. 针对能量和时间这两方面的性能要求,网络介质访问控制(MAC)层的时分多

址(TDMA)是一种很好的解决方案. 尤其在节点无移动且网络部署情况已知的场合,采用 TDMA 方式进行通信可以带来以下好处^[4]: 1) 可避免信道冲突以及冲突引起的丢包和能量损耗; 2) 保证数据传输的实时性和可靠性; 3) 能令节点在不工作期间进入睡眠状态,以保存能量. 在节能型 TDMA 算法中,为了避免耗能较大的空闲侦听状态,节点在无任务状态下应关闭无线收发器并进入睡眠状态,等到工作时隙到来时才打开无线收发器. 然而,过于频繁的状态切换也会导致不必要的能量消耗^[2]. 因此,与其他网络不同,传感器网络中的 TDMA 调度应该考虑让节点的工作状态尽量连续,以节省切换能量.

在目前的数据收集网络的 TDMA 研究中,多

收稿日期: 2006-05-25; 修回日期: 2006-09-06.

基金项目: 国家自然科学基金项目(60074011, 60574049).

作者简介: 毛剑琳(1976—),女,云南宣威人,讲师,博士生,从事无线传感器网络、网络控制系统的研究; 吴智铭(1936—),男,江苏苏州人,教授,博士生导师,从事离散事件系统、混合动态系统等研究.

数研究关注了 TDMA 调度的时间性能^[4-7]. 文献[4, 5]证明数据收集网络的时隙调度是一个 NP 问题. 文献[4]在关注数据包传输次序的基础上, 基于图着色算法, 提出了 3 种以最短任务完成时间为目标的算法. 文献[6]针对线型和树型的网络拓扑, 采用流水线法提出了数据分发/收集的集中调度方案. 文献[7]则提出了数据收集的分布式时隙调度方案. 在能量性能的研究方面^[3,8], 文献[3]采用的节能方法是令节点在空闲时关闭无线收发器. 文献[8]在簇型网络中, 讨论了节点的发送/接收能量和状态切换能量, 在簇首节点使用禁忌搜索方法来达到最优能量调度的目的, 该方法没有考虑数据包的传输时延.

针对目前的研究现状, 本文提出一种新的多目标 TDMA 调度方案, 可以把调度的时间性能和能量性能结合在一起讨论, 并采用微粒群优化算法进行多目标优化, 最终得到时延和能量目标的 Pareto 优化解集.

2 调度问题描述

在 TDMA 协议中, 时间被等分为时隙, 无线通信在时间上的资源分配以时隙为基准. 在数据收集传感器网络中, 所有数据包都经路由节点送到基站节点处, 对于这种多到一 (many to one) 的传输模式^[4], 其 TDMA 时隙调度模型可描述如下: 有一批由传感器节点产生的数据包需经路由节点送至基站. 其中, 一个传感数据包经多跳传输到达基站的过程称为一个任务. 在一个任务中, 每一跳的传输称为一个子任务, 一个子任务对应一个执行节点, 并占用一个时隙. TDMA 的调度任务就是为这些子任务分配时隙, 使得所有的执行节点在不发生信道冲突的情况下, 完成这批数据包的传输任务, 并使数据包的平均时延和节点的平均能耗最小.

对上述模型中的时延和能量目标分别以 f_1 和 f_2 表达, 如下所示:

$$f_1 = \text{AvgDelay},$$

$$f_2 = \frac{1}{N} \sum_{i=1}^N [P^{\text{tx}} (t_i^{\text{s-tx}} + t_i^{\text{rx}}) + P^{\text{rx}} (t_i^{\text{s-rx}} + t_i^{\text{rx}})].$$

其中: f_1 是这批任务的平均时延, f_2 是节点的平均能耗, N 是节点的数目. 根据文献[2], P^{tx} 和 P^{rx} 分别表示节点的发送功率和接收功率, $t_i^{\text{s-tx}}$ 和 $t_i^{\text{s-rx}}$ 分别是节点 i 在发送/接收状态与睡眠状态之间的切换时间的总合, t_i^{tx} 和 t_i^{rx} 则分别是节点 i 完成这批任务所需要的总的发送时间和总的接收时间.

对于此 TDMA 调度问题, 本文进行集中式调度方法的研究, 即: 网络拓扑和节点的任务量已知, 由基站进行网络的时隙调度, 并将时隙安排的结果以广播的形式发布到各传感器节点.

3 基于微粒群的能量-时延 Pareto 优化算法

3.1 微粒群算法原理

微粒群优化是由 Kennedy 和 Eberhart^[9] 在 1995 年提出的一种基于种群的进化计算方法. 其思想如下: 根据编码方式, 随机产生一个初始种群, 该种群中每个个体被称为微粒. 微粒在已定义的搜索空间中以一定的速度飞行, 在飞行过程中, 微粒记录所经历过的最优位置, 并与同伴分享全局最优信息, 以此作为调整自身飞行速度的依据. 整个过程的迭代方程如下:

$$V_{id} = W * V_{id} + C_1 * \text{rand}() * (P_{gd} - X_{id}) + C_2 * \text{rand}() * (P_{id} - X_{id}), \quad (1)$$

$$X_{id} = X_{id} + V_{id}. \quad (2)$$

其中: V_{id} 和 X_{id} 分别是微粒 i 的速度和位置; P_{id} 为微粒 i 所经历的最好位置 (p_best), P_{gd} 为所有微粒所经历的最好位置 (g_best); W 为惯性权重, 它使微粒保持运动惯性, 使其具有扩展搜索空间的趋势, 从而有能力探索新的区域; C_1 和 C_2 为加速度常数, 表示把微粒拉向 p_best 和 g_best 的随机项 (见式 (1) 后两项) 权值, $\text{rand}()$ 返回一个在 $[0, 1]$ 范围内变化的随机数.

3.2 编码

将进化算法用于 TDMA 分配, 一个重要的问题是在个体表达和问题的解之间找到合适的映射关系. 这样才能把个体的搜索空间限定在问题的解空间内, 并通过在此空间内进行最优解的搜索来达到问题求解的目的.

根据第 2 节对问题的描述, 一批数据包的总传输任务可以看成是所有子任务的一个序列. 由此, 标记一个子任务为 (任务编号, 序列号), 其中: 任务编号指该子任务隶属于哪个任务, 序列号则指这个子任务在该任务中执行的顺序号. 例如, 子任务 (3, 2) 是指第 3 个任务的第 2 跳传输. 在编码中, 一个任务编号出现的次数可以暗示其对应的子任务序列号, 因此可取出子任务序列中的任务编号作为一个个体的编码序列. 为了说明此编码原理, 图 1 给出一个示例.

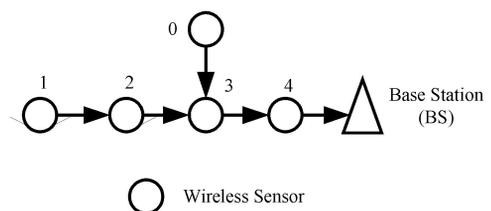


图 1 示例网络

图 1 中有两个传输任务, 分别从节点 0 和 1 各传输一个数据包到基站, 其中前者需要 3 跳传输, 后者

需要 4 跳传输才能到达基站, 即所有的子任务为 (0, 1), (0, 2), (0, 3), (1, 1), (1, 2), (1, 3) 和 (1, 4), 分别对应的执行节点的 ID 是: 0, 3, 4, 1, 2, 3, 4. 若现有一子任务序列为 (0, 1) (1, 1) (1, 2) (0, 2) (0, 3) (1, 3) (1, 4), 则其编码为 0110011.

综上所述, 一个个体是由任务编号组成的序列, 个体编码的总长度为 $\sum_{i=0}^{N-1} M_i$, N 为任务总数, M_i 为任务 i 所需的传输跳数, 编码中出现的任务编号 i 的次数为 M_i . 根据这个编码规则, 一个个体就可以随机方式生成.

根据上述编码机制可知, 个体序列中代码的取值区间为 $[0, N - 1]$, 由此, 解的搜索空间得以限定. 此外, 个体编码到问题解的映射过程为: 对个体所对应的子任务序列顺序地进行时隙分配, 并在不冲突的情况下, 将可并行执行的子任务安排在同一时隙, 最终可以得到一个可行的时隙分配解. 至此, TDMA 调度问题和搜索算法间的映射关系便已完全建立.

3.3 多目标微粒群算法解的评价机制

在多目标优化问题中, 解的评价机制可为微粒群在搜索空间中的飞行提供方向. 一般来说, 多目标问题解的评价方法有两种: 一种是多个目标的加权和; 一种是 Pareto 优化方法.

在微粒群优化算法中, 有个体和群体的概念. 在算法进行迭代时, 微粒的飞行方向是通过自身最优解和全局最优解的拉动作用来共同提供的. 由此, 需要对微粒自身的最优解和全局的最优解进行定义和评价. 与多目标解的两种评价方法相结合, 对不同层次的解采用不同的方法进行评价, 可以得到多目标微粒群优化算法的评价机制如下:

(1) 局部最优解的评价

对于微粒群中个体的局部最优解, 采用加权和的方法进行评价. 对于一个最小化 M 个目标的问题, 引入权重因子 $w_i \in [0, 1]$, 将评价函数取为

$$fit(s) = \sum_{i=1}^M w_i * f_i, \quad (3)$$

其中 $\sum_{i=1}^M w_i = 1$. 在此方法中, 每个微粒只需维护一个局部最优解, 对应该微粒的最优位置 P_{id} . 具体的评价方法为: 当微粒飞行到一个新位置 X_{id} 时, 采用式 (3) 计算该位置的评价函数值, 若该评价函数值优于微粒在 P_{id} 的评价函数值时, 则更新其最优位置为当前位置 X_{id} . 此外, 对于一个微粒而言, 各目标权重因子 w_i 的确定方法为: 在初始化时, 以 $\sum_{i=1}^M w_i = 1$

为条件, 采用随机方式生成 w_i . 这种方法对支持全局 Pareto 解的多样性是有帮助的.

对应第 2 节描述的 TDMA 调度问题, s 对应一个调度方案, $f_i (i = 1, 2)$ 分别对应这个调度方案下的平均时延和节点平均能耗.

(2) 全局最优解的评价和 Pareto 解集的维护

由于 TDMA 调度中的时延和能量目标存在冲突, 即追求切换能量消耗最小必然要求节点的工作连续性好, 这就意味着父节点应收集完所有子节点的数据包之后才统一进行下跳传输. 显然这样会损害到数据包包的时延性能, 反之亦然. 对于这类不能同时达到最优的多目标问题, 加权和的方法难以充分评价解的优劣性, 因此对全局解的评价引入 Pareto 优化概念.

在 Pareto 优化中, 为了比较解的优劣性, 引入支配概念^[10], 其严格定义如下: 对于一个最小化 M 个目标的问题 $\min F = [f_1, f_2, \dots, f_M]$, 存在任意两个解 x_1 和 x_2 , 若 x_1 支配 x_2 , 则必须同时满足如下条件:

- 1) x_1 在所有目标上不都比 x_2 大, 即 $f_j(x_1) \leq f_j(x_2), \forall j (1, 2, \dots, M)$;
- 2) x_1 至少在一个目标上是严格比 x_2 小的, 即 $f_j(x_1) < f_j(x_2)$.

若上述条件中的任何一条不满足, 则 x_1 不支配 x_2 . 所有互不支配的解构成非支配解集, 当解的比较空间是全局空间时, 该非支配解集称为 Pareto 最优解集^[10]. 以一个最小化两目标问题的示意图为例 (见图 2), 图中的解 B 在两个目标上都比解 C 小, 因而解 B 是支配解 C 的. 而解 B 在两个目标上都不同时比解 A 小, 反之亦然. 因此, 解 A 和解 B 称为非支配解, 在全局意义的比较下, 解 A 和解 B 称为 Pareto 优化解, 所有的 Pareto 优化解可构成 Pareto 最优前沿.

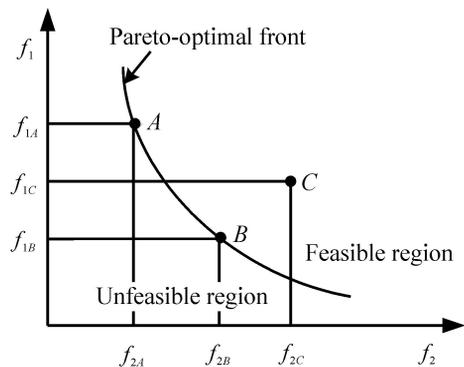


图 2 Pareto 优化概念的图解

3.4 PSO-Pareto 优化算法

PSO-Pareto 优化算法流程如下:

1) 初始化 PSO 算法参数, 包括群大小, 最大迭代次数 \max_gen_PSO , W , $C1$, $C2$, i 等.

2) 初始化 Pareto 解集. 对于所有的初始解, 依次计算它们的目标值 f_1, f_2, \dots, f_i ; 然后依 Pareto 优化概念, 逐一判断它们之间的支配关系, 并形成一个初始的 Pareto 解集.

3) 当迭代次数 $generation$ 未达到 \max_gen_PSO 时,

```
{ generation = generation + 1;
  用式(1)和式(2)产生下一代微粒群;
  根据3.3节的评价机制评价新的微粒群,并更新所有个体的最优位置和全局的最优位置集合(即 Pareto 解集);
}
```

4) 空间飞行结束, 输出 Pareto 优化结果.

4 仿真结果

仿真试验中, 81 个节点被放置在 640 m^2 的网格区域中, 随机生成的网络拓扑如图 3 所示. 其中: 最小节点度数为 2, 最大节点度数为 8, 平均节点度数为 5.62. 基站节点是位于区域正中间的节点. 自由空间传播的信道容量为 500 kbps, 一个数据包的大小是 1 kb, 数据包的路由采用 Dijkstra 最短路径法. TDMA 时隙的大小取为一个数据包所需的传输时间和可能的时钟偏移时间之和, 以消除时钟偏移对 TDMA 算法的影响. 节点活动状态和睡眠状态之间的切换时间是 $470\ \mu\text{s}$, 发送和接收一个数据包所需的功率是 81 mw 和 180 mw^[2].

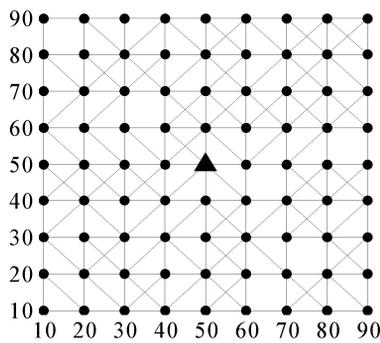


图 3 网络拓扑结构

仿真实验比较了 3 种算法: 最大度数优先的 2-Distance 节点着色算法 (MDFCA), 文献 [7] 提出的基于节点的调度算法 (NBSA), 以及本文提出的 PSO-Pareto 算法. 在 PSO-Pareto 中, 种群的大小取为 50, 最大进化代数取为 200. 仿真实验采用的任务是每个节点产生一个传感数据包, 并将之送至 BS. 3 种算法均采用节能方式, 即无收发任务时, 节点进入睡眠状态.

表 1 给出了 3 种调度算法的平均时延、平均能

耗和总任务完成时间结果. 在 3 种算法中, MDFCA 的结果在 3 个性能上都是最差. 这是因为在 MDFCA 中, 有很多时隙被分配给了无传输任务的节点, 从而导致了大量时隙的浪费, 降低了传输的时间性能. 此外, 在 MDFCA 中, 节点在每个颜色周期只能工作一个时隙, 这使得节点每次传输都需要切换其状态, 能量消耗比较大. 可见, 基本图着色算法不适用于数据收集传感器网络.

表 1 3 种算法的结果列表

算 法	任务平均时延/ slot	节点平均能耗/ mJ	总任务完成时间/ slot
MDFCA	92.80	2.965	250
NBSA	42.54	2.662	94
PSO-Pareto(1)	39.2	2.671	96
PSO-Pareto(2)	40.05	2.650	99
PSO-Pareto(3)	40.32	2.648	91
PSO-Pareto(4)	40.58	2.594	91
PSO-Pareto(5)	42.21	2.578	96
PSO-Pareto(6)	43.26	2.574	91
PSO-Pareto(7)	44.49	2.562	95

NBSA 算法在 MDFCA 的基础上作了改进, 以节点任务为依据, 排除了无传输任务的空闲时隙, 在时间性能上比 MDFCA 有较大的改善, 尤其是总任务完成时间被大大缩减. 在能量方面, 由于 NBSA 在排除空闲时隙以后, 增加了节点工作状态的连续性, 能量消耗也比 MDFCA 小. 需说明的是, 这两种算法都不具备调节网络能量消耗的能力.

依 Pareto 优化概念对各算法的结果进行分析, 由表 1 显见, NBSA 和 PSO-Pareto 的结果都对 MDFCA 的结果构成支配关系. 为清晰起见, NBSA 和 PSO-Pareto 之间的比较由图 4 给出. 图 4 显示 NBSA 的解被 PSO-Pareto 解集中的解 (2) ~ (5) 所支配, 可见, PSO-Pareto 算法能得到比 NBSA 更好的调度结果. 此外, PSO-Pareto 在对能量和时延进行折衷的同时, 并没有损坏总任务完成时间性能. 从

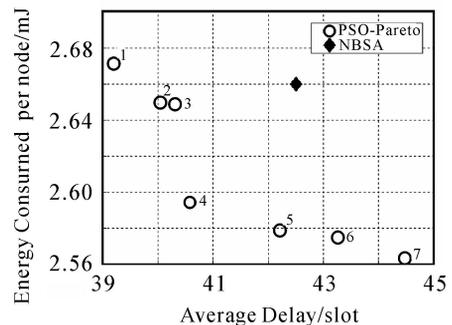


图 4 能量-时延 Pareto 结果图

表 1 可以看到, PSO-Pareto 的总任务完成时间是和 NBSA 算法相当的, 其中解 (3), (4), (6) 比 NBSA 算法的总任务完成时间要小.

对上述结果进行分析和总结, PSO-Pareto 优于另两种算法的原因在于:

1) 有效解决问题的思路. 本文的 TDMA 时隙分配思路是对所有子任务依次安排时隙, 这种解决思路不会产生空闲时隙, 可以保证较好的时间性能, 而且子任务时隙能灵活调整, 可令节点工作的连续性好, 从而降低能量消耗.

2) 好的搜索算法. PSO 算法能以 Pareto 解集为指导充分搜索解空间, 提高时隙安排的灵活性, 从而能找到更好的能量和时延性能的折衷点.

5 结 语

本文针对无线传感器网络多到一的数据传输特点, 结合 Pareto 优化方法和搜索能力强的 PSO 算法, 对 TDMA 调度问题提出了 PSO-Pareto 算法. 该算法具有如下优势: 1) 在进化搜索算法的问题求解框架之下, 易于处理多个目标寻优的问题, 建立多目标优化的模型; 2) 可充分发挥进化搜索算法对 NP 问题的求解能力; 3) Pareto 优化概念的引入使多目标解的评价更加合理, 能为决策者在网络性能的取舍上提供更多的选择.

在无线传感器网络中, 多目标优化问题普遍存在, 而且多数目标之间存在牵制关系. 对此, 本文提出的多目标优化算法可以为这类问题提供一个很好的解决思路.

参考文献(References)

[1] Kahn J M, Katz R H, Pister K S J. Next century challenges: Mobile networking for smart dust [C]. Mobicom 99. Seattle Washington, 1999: 271-278.

[2] Shih E, Cho S H, Ickes N, et al. Energy-efficient link layer for wireless microsensor networks[C]. Proc of the Workshop on VLSI 2001. Orlando, 2001: 16-21.

[3] Pei Guangyu, Charles Chien. Low power TDMA in large wireless sensor networks[C]. Proc of MILCOM. 2001, 1: 347-351.

[4] Sinem Coleri Ergen, Pravin Varaiya. TDMA scheduling algorithms for sensor networks [R]. Berkeley: Department of Electrical Engineering and Computer Sciences, University of California, 2005.

[5] Choi H, Wang Ju, Esther A Hughes. Scheduling on sensor hybrid Network[C]. IEEE ICCCN. San Diego, 2005: 503-508.

[6] Florens C, McEliece R. Packet distribution algorithms for sensor networks[C]. IEEE INFOCOM. San Diego, 2003: 1063-1072.

[7] Shashidhar Gandham, Zhang Ying, Huang Qing-feng. Distributed minimal time convergecast scheduling in wireless sensor networks [C]. The 26th Int Conf Distributed Computing Systems (ICDCS06). Lisboa, 2006.

[8] Gaurav Jolly, Mohamed Younis. An energy-efficient, scalable and collision-free MAC layer protocol for wireless sensor networks[J]. Wireless Communications and Mobile Computing, 2005, 5(3): 285-304.

[9] Eberhart R, Shi Y. Comparison between genetic algorithms and particle swarm optimization[C]. Volume 1447 of Lecture Notes in Computer Science. Springer, 1998: 611-616.

[10] Deb K. Evolutionary algorithms for multi-criterion optimization in engineering Design [C]. Proc of Evolutionary Algorithms in Engineering and Computer Science (EUROGEN-99). Chichester: John Wiley & Sons, 1999: 135-161.

(上接第 966 页)

[36] Zuo X Q, Li S Y. Design of a fuzzy logical controller by immune algorithm with application to an inverted pendulum system[C]. Proc of the 2003 Congress on Evolutionary Computation. Honolulu, 2003, 1: 100-106.

[37] Ding Y S, Ren L H, Shao S H. Automatic design of takagi-sugeno fuzzy controllers by a new DNA-based evolutionary algorithm [J]. Acta Automatica Sinica, 2001, 27(4): 510-520.

[38] Larrañaga P, Lozano J A. Estimation of distribution algorithms: A new tool for evolutionary computation

[M]. Dordrecht: Kluwer Academic Publishers, 2001.

[39] Stewart P, Stone D A, Fleming P J. Design of robust fuzzy-logic control systems by multi-objective evolutionary methods with hardware in the loop [J]. Engineering Applications of Artificial Intelligence, 2004, 17(3): 275-284.

[40] Amaral J F M, Tanscheit R, Pacheco M A C, et al. Evolutionary fuzzy system design and implementation [C]. Proc of the 9th Int Conf on Neural Information Processing. Singapore, 2002, 4: 1872-1876.