基于生成树的无线传感器网络时钟同步算法*

胡 雯†, 郑 霖, 仇洪冰, 覃秀英
(桂林电子科技大学 信息与通信学院, 广西 桂林 541004)

摘 要: 考虑到无线传感器网络时钟同步多跳误差累积问题,提出一种基于动态生成树的全网时钟同步算法,只需由根节点开始沿树边广播一次同步消息,全网待同步节点即可采用时钟偏差补偿,并结合贝叶斯后验估计算法对时钟进行准确估计。一个同步轮次内每个节点至多广播两次消息即可完成全网同步。使用 OMNeT ++ 软件进行仿真实验表明算法有效降低了多跳累积误差,提高了同步精度,且具有较快的收敛速度和较低的开销。

关键词: 无线传感器网络; 时钟同步; 贝叶斯估计; 多跳累积误差
中图分类号: TP391 **文献标志码**: A **文章编号**: 1001-3695(2012)10-3876-03
doi:10.3969/j.issn.1001-3695.2012.10.072

Ad hoc tree structured time synchronization algorithm for wireless sensor networks

HU Wen†, ZHENG Lin, QIU Hong-bing, QIN Xiu-ying
(*School of Information & Communication, Guilin University of Electronic Technology, Guilin Guangxi* 541004, *China*)

Abstract: For wireless sensor networks spanned many hops and the synchronization error accumulated while the path length grows, this paper proposed an Ad hoc tree structured synchronization algorithm for WSNs. As a root of time synchronization path, root node broadcasted only one message along the tree edges, could synchronizate all nodes through the clock offset compensation and Bayesian posterior estimation. Each node was able to achieve network-wide time synchronization by broadcasting at most two messages. Simulation on OMNeT ++ shows the proposed algorithm can effectively reduce the synchronization error. It is a fast, resource-efficient algorithm with high synchronization precision and low overhead.

Key words: wireless sensor network(WSNs); clock synchronization; Bayesian estimation; accumulated error

0 引言

目前针对无线传感器网络时间同步协议[1]已经有了较多的研究。早期研究主要集中在同步时间误差来源,研究[2]表明误差来源可看做两部分:a)由节点时钟频率偏移引起的时间偏差;b)消息传递过程中产生的延迟。由此,许多通过估计和减少这些误差来提高同步精度的算法被相继提出。具有代表性的算法有 DMTS(delay measurement time synchronization)算法[3],通过估算消息传递过程中所有延迟完成全网同步,以降低同步精度为代价降低能耗;RBS(reference broadcast synchronization)算法[4],通过利用无线信道广播特性消除发送端时延影响,但同步过程需要交换信息次数较多;TPSN(timing-sync protocol)算法[2],采用成对同步机制,通过估算传输时延和相对时钟偏移来同步以减少同步过程中通信量,但多跳误差累积对同步精度影响较大。近几年来研究者倾向于通过一种合适的拓扑结构[5]来进行同步,其中轻量级的 LTS(light-weight time synchronization)算法[6]首先建立树型网络拓扑,再利用成对同步机制沿着树边进行同步;TSTP(tree structured time synchronization)算法[7]在网络中动态建立树型拓扑,再利用成对同步机制沿树边同步。

任何算法都做不到绝对同步,若单跳误差独立同分布 $\varepsilon \sim N(0,\sigma^2)$,经过 n 跳后的累积误差均方差为$\sqrt{n}\sigma$。目前的同步算法大多是采用提高单跳同步精度或选取最短路径同步来减少跳数,以降低多跳误差累积,却没有考虑到充分利用周围节点的时钟信息以降低误差随跳数累积的速度。目前采用成对同步机制的算法如 TPSN、LTS、TSTP 等属于拉(pull)式算法,在实现中时间信息是由高层次节点向低层次节点索取的,不能快速扩展至全网,收敛速度较慢;拓扑结构建立后不再改变,全网同步效果极易受到节点失效等特殊状况的影响。传统使用贝叶斯估计的算法在实现中节点间需要多次交换同步消息,高层节点才能获得足够的先验时钟信息进行贝叶斯算法估计,在逐层次扩展到全网的过程中,对整体的资源消耗非常大,收敛时间很长。

针对上述问题,本文提出一种基于动态生成树的全网时钟同步算法,基本思想是在一个同步轮次内,每个节点广播一次消息即可在网络中构造一个包含所有节点的生成树,同时估计出自身与子节点间的相对时钟偏差,再由根节点开始沿树边广播一次同步消息,采用时钟偏差补偿结合贝叶斯后验估计算法[8]对全网所有待同步节点时钟进行准确估计,从而快速精确地完成同步。本算法结合贝叶斯估计原理,在同步阶段只需一次广播将同步消息推送至全网,不仅减轻了目前同步算法中普遍存在的多跳误差累积对全网同步效果的影响,更简化了传统贝叶斯算法中节点间需要多次交换信息才能进行同步的过程;动态构建生成树和推送式的信令分发机制也使本算法比采用固定拓扑的拉(pull)式算法更加灵活,收敛速度更快;全网

收稿日期: 2012-01-16; **修回日期**: 2012-02-22 **基金项目**: 国家自然科学基金资助项目(61071088,60962001)

作者简介:胡雯(1987-),女(水族)(通信作者),山东历城人,硕士,主要研究方向为分布式时钟同步算法(snowballbear@yahoo.cn);郑霖(1974-),男,教授,博士,主要研究方向为超宽带通信、无线传感器网络、移动通信、自适应信号处理、扩频通信;仇洪冰(1963-),男,教授,博士,主要研究方向为移动通信、宽带通信网络;覃秀英(1986-),女,硕士,主要研究方向为分布式时钟同步算法.

同步过程中每个节点至多广播两次消息,同步开销很小。

1　消息传输模型

影响时间同步的主要因素是时间同步消息在网络中的传输延迟。总的来说,延迟可分为六类[9]:

a)发送时延(send time)。节点构造消息并递交给 MAC 层准备发送所需的时间,受操作系统和当前的处理器负载影响很严重,最高可达几百 ms。

b)接入时延(access time)。等待信道空闲所需的时间,与网络负载和特定的 MAC 协议有关。

c)传输时延(transmission time)。节点在物理层按比特发送消息所需的时间,取决于消息长度和发送速率,是个确定值。

d)传播时延(propagation time)。承载消息的无线信号在源节点与目的节点之间传播所需要的时间,这个时间依赖于节点间的距离,该时延多在纳秒级,可以忽略。

e)接收时延(reception time)。节点物理层按比特接收消息所需的时间。

f)接收处理时延(receive time)。节点把接收到的比特还原成发送者发送的分组,然后把分组递交给上层所需的时间,其不确定性与发送时延类似。

2　基于动态生成树的全网同步算法

针对目前大部分的传感器网络应用场景,本算法基于以下假设:根节点作为全网的时间基准节点,时间精度最高,其余节点最终要达到与根节点的时钟同步;全网节点同构,即节点的硬件设施相同;在一个同步周期内,晶振漂移有限。

本算法分为两个阶段:a)构建动态生成树和时钟偏差估计;b)同步全网节点本地时钟。算法信息流程如图 1 所示。

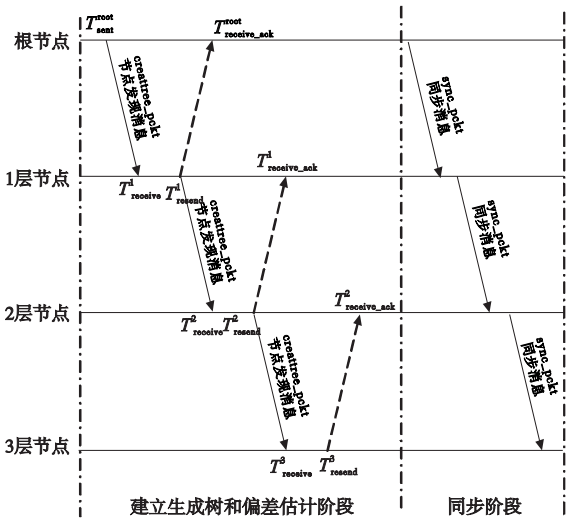

图1　本文算法信息流程

2.1　构建动态生成树和时钟偏差估计阶段

本文算法在构造一个包含所有节点的生成树的同时,即可估计出自身与子节点间的相对时钟偏差,并以列表形式存储在节点上,在同步阶段放入同步消息中广播出去。

2.1.1　具体步骤

a)选取一个根节点 root 作为全网时钟基准节点,其余所有节点最终都要与 root 节点同步;

b)Root 节点首先广播子节点发现消息 creattree_pckt,包括自身 ID、父节点 ID、层次号 levelID,记录下本地广播时间 T_{send};

c)未分配层次的节点接收到 creattree_pckt 消息,记录本地接收时间 $T_{receive}$,将发送节点设为自身的父节点,并更新自身层次号为 levelID + 1,产生新的子节点发现消息 creattree_pckt,并将 $T_{receive}$ 和本地发送时间 T_{resend} 放入消息中广播出去;

d)已分配层次的节点接收到来自自身子节点的 creattree_pckt 消息,记录下本地接收时间 $T_{receive_ack}$,则节点可计算出自身与此子节点间的时钟偏差,在同步阶段将此时钟偏差告知子节点作为时钟修正参考依据;

e)父节点将同步所需时钟偏差量进行缓存,在同步阶段发送给子节点。

为减少信息交换开销,设计如表 1 所示相对时钟偏移量列表 list。

表 1　相对时钟偏移量列表 list

节点 ID	selfID			
子节点 ID	$childID_1$	$childID_2$	…	$childID_N$
父子节点间时钟偏差	$offset_1$	$offset_2$	…	$offset_N$

2.1.2　时钟偏差估计方法

父节点 A 与子节点 B 间时钟偏差 $offset_{AB}$ 可由式(1)估计得出,其原理如图 2 所示。

$$offset_{AB} = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} \tag{1}$$

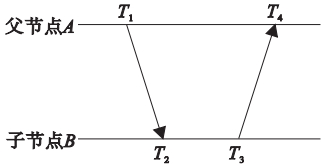

图2　时钟偏差估计方法

2.2　同步阶段

2.2.1　贝叶斯估计方法

假设节点 i 的本地时钟观测值为 T_i,经贝叶斯估计后的值为 T_i',上层节点经过贝叶斯估计估算出的值为 T_{i-1}',且服从方差为 $\sigma_{i-1}'^2$ 的正态分布。

由于节点 i 的时钟观测值存在随机误差,通过信息交换后得出的节点时钟观测值 T_i 总是与未知的准确时间值 t_i 存在误差,这个误差服从 $N(0,\sigma_i^2)$,σ_i^2 从之前的实验记录中取得[7]。由于 $p(T_i|t_i) \sim N(t_i,\sigma_i^2)$,根据上层节点时钟值先验分布服从 $N(T_{i-1}',\sigma_{i-1}'^2)$,由贝叶斯理论可估计出更为准确的本地时钟值 $p(t_i|T_i) \sim N(T_i',\sigma_i'^2)$,其中:

$$\sigma_i'^2 = \frac{\sigma_{i-1}'^2\sigma_i^2}{\sigma_{i-1}'^2 + \sigma_i^2} \tag{2}$$

$$T_i' = \frac{\sigma_i^2}{\sigma_{i-1}'^2 + \sigma_i^2}T_{i-1}' + \frac{\sigma_{i-1}'^2}{\sigma_{i-1}'^2 + \sigma_i^2}T_i \tag{3}$$

传统使用贝叶斯估计的算法[10]中,节点间需要多次交换同步消息,高层节点才能获得足够的先验时钟信息进行贝叶斯估计以得出准确的时钟,同步后的上层节点再通过多次消息交互的方式同步下层节点。因此两层节点间的同步需要大量的消息交换操作才能实现,在逐层次扩展到全网的过程中,对整体的资源消耗是非常大的,并且收敛时间很长。

本文算法在同步阶段中只需由父节点向子节点广播一次消息,子节点即可对时钟偏差进行补偿,而后使用贝叶斯估计获得更精确的节点时钟。同步过程扩散至全网的过程中,各节

点至多广播一个同步消息。

2.2.2 具体步骤

a)由 root 节点开始,令 $T_1' = T_1, \sigma_1' = \sigma_1$, T_1 为根节点时间值,σ_1 为根节点时钟分辨率。将 T_1、σ_1 和相对时钟偏移量列表 list 放入同步消息 sync_pckt 中广播给自身子节点。

b)作为父节点的节点将自身经贝叶斯估计后的时间值 T_{i-1}'、估计误差的标准差 σ_{i-1}' 和本地时钟偏移量列表 list 放入同步消息 sync_pckt 中广播给子节点。

c)子节点接收到来自父节点的同步消息 sync_pckt 后,根据 list 中的时钟偏差信息计算出本地节点与 root 节点间的时钟关系,更新本地时钟偏移量列表并进行时钟调整。调整后的本地时钟读数 T_i 和从实验记录中取得的误差标准差 α_i,结合 T_{i-1}' 和 σ_{i-1}' 利用式(2)(3)即可计算出更为准确的 T_i' 和 σ_i'。

d)依次按照以上步骤执行,直至全网。

3 仿真实现和性能分析

本文利用 OMNeT ++ 软件作为仿真平台对算法进行仿真模拟。为了更好地进行性能分析,本文使用相同的参数设置对经典 TPSN 算法也进行了仿真。

3.1 同步精度比较

本文是在一个较为常见的网络环境[4]下进行仿真。在 100 m × 100 m 的区域内,均匀随机布置 30 个节点,其中 1 个 root 点和 29 个待同步节点,共 5 跳,单节点无线通信距离为 25 m。根节点 Root 的时钟分辨率为 1 μs ,也就是说根节点产生数据包的时钟偏差位为 1 μs。实验 100 次统计实验结果。

图 3 为随着跳数增加本文算法和 TPSN 算法的同步误差。表 2 为多跳网络同步累积误差。由表 2 中实验数据可知,在这个 5 跳网络中,随着跳数的增加,本文算法引入贝叶斯估计方法后,同步误差累积速度较为缓慢,而 TPSN 算法的误差累积速度相对快得多。完成一次同步后,TPSN 算法的 5 跳同步累积误差达到 8.88 μs,而本文算法得到的累积误差仅为 3.37 μs。

表 2 多跳网络同步累积误差 /μs

算法	跳数				
	1	2	3	4	5
TPSN 算法	4.02	5.52	6.63	7.61	8.88
本文算法	1.05	1.66	2.32	2.89	3.37

3.2 同步开销和收敛速度比较

设网络中有 N 个节点,拓扑结构建立后非叶子节点数为 L ($L < N$)。本文算法完成一次同步所需消息数为($N + L$)个($N + L < 2N$)。若层次结构已知,TPSN 算法完成一次同步所需消息数为($2N - 2$);若层次结构未知,TPSN 算法完成一次同步所需消息数为($3N - 2$)。因此,在网络中的节点数大于 3 个时,本文算法在一个同步周期内的同步开销小于 TPSN 算法。

图 4 为本文算法与 TPSN 算法开销比较。通过图 4 可以看出,本文算法提出的信令分发机制所需要的同步开销比 TPSN 算法小,主要是因为本文算法在一个同步周期内,每个节点在建立生成树和偏差估计阶段只需广播一次消息,而在同步阶段只有非叶子节点需要广播一次消息,从而降低了开销。

在 TPSN 算法的实现中,时间信息是由高层次节点向低层次节点索取的,高层节点质听到低层节点开始同步后,随机等待约 10 s,估计低层节点已完成同步后发起同步请求,属于拉式的算法,时间信息不能快速扩展至全网,收敛速度较慢。本文算法在同步阶段开始后,由根节点开始向下推送时钟信息,收敛速度更快。

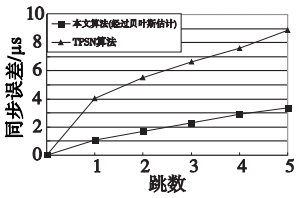

图3 本文算法与TPSN算法5跳同步累积误差比较

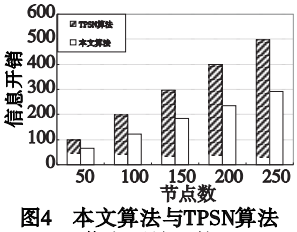

图4 本文算法与TPSN算法信息开销比较

4 结束语

本文提出基于动态生成树的全网时钟同步算法,充分利用了周围节点的时钟信息,结合贝叶斯后验估计进行数据融合,以减小消息交换延迟引起的同步误差。在同步阶段只需一次广播,在将同步消息推送至全网的同时完成全网同步,比传统的需要节点间多次消息交互才能进行贝叶斯估计的算法更加简便。从仿真实验结果即可看出本算法确实降低了多跳误差累积速度,提高了同步精度;动态构建生成树和推送式的信令分发机制使得全网节点只需花费很小的开销即可快速完成同步,算法更加灵活,收敛速度更快;整网同步过程中每个节点最多广播两次消息,同步开销很小。本文算法利用周围节点时钟信息进行数据融合,减小同步累积误差的方式在多跳网络同步研究中具有一定的参考价值。由于实验条件和环境的限制,本文只能在 OMNeT ++ 仿真平台上对算法进行实现,仍需要在实际的硬件平台上对算法进行进一步测试验证。

参考文献:

[1] WU Y C, CHAUDHARI Q, SERPEDIN E. Clock synchronization of wireless sensor networks[J]. IEEE Signal Processing Magazine, 2011, 28(1): 124-138.

[2] GANERIWAL S, KUMAR R, SRIVASTAVA M B. Timing-sync protocol for sensor networks[C]//Proc of the 1st ACM Conference on Embedded Networked Sensor Systems. 2003: 138-149.

[3] SU Ping. Delay measurement time synchronization for wireless sensor networks, IRB-TR-03-013[R]. Berkely, CA: Intel Research, 2003.

[4] ELSON J, GIROD L, ESTRIN D. Fine-grained network time synchronization using reference broadcasts[C]//Proc of the 5th Symposium on Operation System Design and Implementation. New York: ACM Press, 2002: 147-163.

[5] 王伟,林锋,周激流. 无线传感器网络覆盖问题的研究进展[J]. 计算机应用研究, 2010, 27(1): 32-35.

[6] Van GREUNEN J, RABAEY J. Lightweight time synchronization for sensor networks[C]//Proc of the 2nd ACM Internationsl Conference on Wireless Sensor Networks and Applications. New York: ACM Press, 2003.

[7] RAHAMATKAR S, Dr AGARWAL A, SHARMA V, *et al*. Tree structured time synchronization protocol in wireless sensor network [J]. Special Issue on Ubiquitous Computing Security Systems, 2009, 11(4): 712-717.

[8] GAO Q, BLOW K J, HOLDING D J. Simple algorithm for improving time synchronization in wireless sensor networks[J]. Electronics Letters, 2004, 40(14): 889-891.

[9] 汪付强,曾鹏,于海斌. 基于层次结构的工业无线网络时间同步协议[J]. 计算机应用研究, 2010, 27(7): 2688-2691.

[10] 夏小凤,严斌宇,刘方圆,等. 无线传感器网络 TPSN 的优化算法[J]. 智能仪表与传感器, 2010, 18(6): 1465-1467.