

线性传感器网络时间同步协议^{*}

徐朝农¹⁺, 徐勇军², 邓志东¹

¹(清华大学 智能技术与系统国家重点实验室,北京 100084)

²(中国科学院 计算技术研究所,北京 100190)

Timing-Sync Protocol for Linear Sensor Networks

XU Chao-Nong¹⁺, XU Yong-Jun², DENG Zhi-Dong¹

¹(State Key Laboratory of Intelligent Technology and Systems, Tsinghua University, Beijing 100084, China)

²(Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100190, China)

+ Corresponding author: xu_chaonong@ict.ac.cn

Xu CN, Xu YJ, Deng ZD. Timing-Sync protocol for linear sensor networks . *Journal of Software*, 2009, 20(Suppl.):266–277. <http://www.jos.org.cn/1000-9825/09031.htm>

Abstract: Due to the features of high synchronization error and high power consumption, the classic synchronization scheme of two-way packets exchange is unfit for some applications in wireless sensor networks, especially for networks with multi-hop linear topology. This paper proposes a time synchronization protocol named Timing-sync Protocol for Linear Sensor Networks (TPLSN). The synchronization scheme of enhanced two-way packets exchange and the scheme of clock skew compensation are keys to the success of TPLSN. The phenomenon of its synchronization error accumulation over hop count is also investigated. TPLSN is evaluated on a Mica2-compatible test bed. Its synchronization error is less than 20 μ s for the node which is 9 hops away from the time beacon node, the increase ratio of synchronization error to hop count is less than 1 μ s per hop, and the increase ratio of synchronization error to resynchronization cycle is 0.017 μ s per second. Further, to synchronize all nodes in an n-hop linear wireless sensor network, only 2n packets are needed, which is least for any synchronization protocol based on synchronization scheme of two-way packet exchange. Theoretical analysis shows that three factors, including the approximated accuracy, the asymmetry of two-way packets exchange, and the clock skew, have great influences on time offset between two adjacent nodes. Furthermore, the clock frequency order of the linear network is found to be vital to the accumulation of synchronization error over hop count.

Key words: wireless sensor networks; time synchronization; clock synchronization; clock skew; synchronization error accumulation

摘要: 由于存在着高同步误差和高功耗的特点,尤其当网络拓扑为多跳线状时,经典的双向报文交换同步机制不适合于无线传感器网络中的一些应用.提出了 TPLSN 时间同步协议.两个组件,包括:改进的双向报文交换同步机制和时钟漂移补偿机制是 TPLSN 成功之关键.进一步讨论了 TPLSN 同步误差随跳数增长的现象.在一个与 Mica2 兼容的测试床上对 TPLSN 进行了性能评估.距离时间基准节点 9 跳的节点的同步误差小于 20 μ s,同步

* Supported by the National Natural Science Foundation of China under Grant Nos.90820305, 60621062, 60772070 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant Nos.2007AA041702, 2009AA01Z126 (国家高技术研究发展计划(863)); the China Postdoctoral Science Foundation under Grant No.20080430412 (中国博士后科学基金)

Received 2009-05-01; Accepted 2009-07-23

误差随跳数的增长率小于 $1\mu\text{s}/\text{跳}$,同步误差随重同步周期的增长率为 $0.017\mu\text{s}/\text{s}$.此外,同步一个 n 跳的线状网络只需要 $2n$ 个报文,这也是所有基于双向报文交换同步机制的同步协议最小所需的报文数.理论分析表明:近似精度、双向报文交换的不对称性和相对时钟漂移因素对两个相邻节点之间的同步误差有很大的影响.进而,线状网络的时钟序对同步误差的累积至关重要.

关键词: 无线传感器网络;时间同步;时钟同步;时钟漂移;同步误差累积

时间同步^[1]是无线传感器网络的基本中间件,它不仅是实际应用必不可少的组件,也是例如低功耗睡眠、MAC链路调度、路由、数据融合、定位等其他中间件的基础.

相对于单向报文同步机制,经典的双向报文交换同步(简称为双向同步)机制对报文传输延迟不敏感^[2],通用性较强,从而被NTP(network time protocol)^[3]和工业网络标准IEEE1588所采纳.然而,其单跳同步精度有限,并且同步误差随跳数的增加而快速增加.对在实际应用中非常广泛的具有线状拓扑的网络(简称为线性网络)而言,由于网络直径较大,从而造成远离时间基准的节点同步误差很大,限制了经典的双向同步机制的应用.因此,经典的双向同步机制在应用到多跳线性网络时必须要进行改进.

非同步期过长以及在非同步期内发生较大时间的突变是造成经典的双向同步机制同步精度低的重要原因.我们针对这两个问题对其进行了改进.基于这两处改进和一个时钟漂移的最小方差无偏估计器,我们提出了适合于多跳线性无线网络的时间同步协议TPLSN.在GAINS(global accessible intelligent nodes)节点^[4]上的实验表明:(1) 单跳同步精度可达 $10\mu\text{s}$ 左右,同步误差随跳距的增长率约为 $1\mu\text{s}/\text{跳}$.这表明即使远离时间基准节点,节点的同步误差也很小.(2) 同步误差随重同步周期的增长率为 $0.017\mu\text{s}/\text{跳}$,这表明即使长时间不进行重同步操作,同步精度也可满足需要.(3) 对于 n 跳线性网络,同步网络中所有节点的报文个数仅为 $2n$,该值也是全程采用双向同步机制最少需要的同步报文数.以上结果表明TPLSN协议具有高精度和低能耗的优点.

为了进一步提高 TPLSN 协议在多跳线性网络下的同步精度,我们研究了其同步点处同步误差随跳数累积现象.理论分析表明:近似精度、双向报文交换的不对称性和相对时钟漂移对两个相邻节点间的同步误差有很大的影响.进而,线性网络的时钟序对同步误差随跳数的累积至关重要.从而我们认为:针对网络的实际特点进行合理的节点位置摆放能提高 TPLSN 协议的同步精度.

第 1 节定义本文所涉及的一些基本术语.作为第 3 节的基础,第 2 节简要介绍传统双向同步机制以及时钟漂移估计和时钟漂移补偿技术.第 3 节详细介绍 TPLSN 协议,包括 TPLSN 协议的框架、对经典双向同步机制的改进和对全局时钟漂移的估计.第 4 节分析 TPLSN 协议同步误差随跳数累积现象.第 5 节为实验与分析.第 6 节为相关工作回顾和讨论.最后为本文总结.

1 术 语

考虑这样的场景:全网中只有一个时间基准节点,只有该节点拥有获得真实时间(或实时时间)的能力.本文中的同步是指节点必须尽可能准确地从时间基准节点处获得真实时间.

时间(time)、频率(frequency):每个节点拥有一个由晶体振荡器驱动的计时设备.当实时时间为 t 时其晶体振荡器的频率记为 $f(t)$.而对应的本地时间记为 $c(t)$,也称为节点的本地时间.显然的: $f(t) = dc(t)/dt$.对于具体的某个节点,例如节点 A ,为表示方便,其晶体振荡器的频率记为 $f_A(t)$,本地时间记为 $c_A(t)$.

时间偏移(time offset)、同步误差(synchronization error)、全局时间偏移(global time offset)与全局同步误差(global synchronization error):节点间的本地时间之差称为时间偏移.例如:节点 B 相对于节点 A 的时间偏移定义为 $c_A(t) - c_B(t)$.与文献[5]一致,节点 B 和节点 A 之间的同步误差定义为 $abs(c_A(t) - c_B(t))$.如果节点 A 为时间基准节点,则称 $c_A(t) - c_B(t)$ 为全局时间偏移. $abs(c_A(t) - c_B(t))$ 为全局同步误差,这里 abs 为取绝对值操作.

跳数(hop count):连接节点和时间基准节点之间的最短路径的跳段数.

时钟漂移(clock skew):节点 B 相对于节点 A 的时钟漂移定义为 $(f_A(t) - f_B(t))/f_B(t)$,记为 $k_{B,A}$.

2 经典双向同步机制、时钟飘移估计及补偿技术

2.1 经典双向同步机制

图 1 表示了经典双向同步机制^[6].考虑两个节点:节点A和节点B.在第*i-1* 轮同步周期时,节点B需要和节点A 同步.此时,节点B向节点A发送一个同步请求报文.当节点A接收到该报文时,随即发出一个同步应答报文.节点 B 获得 $c_B(t1^{(i-1)})$, $c_A(t2^{(i-1)})$, $c_A(t3^{(i-1)})$ 和 $c_B(t4^{(i-1)})$ 这 4 个时刻信息后,可根据(1)式计算出其相对于节点 A 的时间偏移.

$$\xi(i-1) = \frac{(c_A(t2^{(i-1)}) - c_B(t1^{(i-1)})) - (c_B(t4^{(i-1)}) - c_A(t3^{(i-1)}))}{2} \quad (1)$$

随后,节点 B 在其本地时间上加上 $\xi(i-1)$ 后,就达到了和节点 A 之间的瞬时同步.

与双向同步相关的一些术语定义如下:

同步点(synchronous moment):节点 B 计算出时间偏移后,修改自己的本地时间达到和节点 A 之间的瞬时同步,此时刻称为节点 B 的同步点.

同步期(synchronized period)、非同步期(non-synchronized period):如图 1 所示,一轮同步由同步期和非同步期组成.其中,双向报文交换开始直至同步点的时间段称为非同步期,从本轮同步点后直至下轮双向报文交换开始的时间段称为同步期.

重同步周期(resynchronization cycle):由于节点间存在时钟飘移,因此即使在达到瞬时同步后,一段时间后又将出现失步,因此需要再次进行同步操作.相邻轮同步操作之间的时间段长度称为重同步周期.很明显,重同步周期是同步期和非同步期之和.

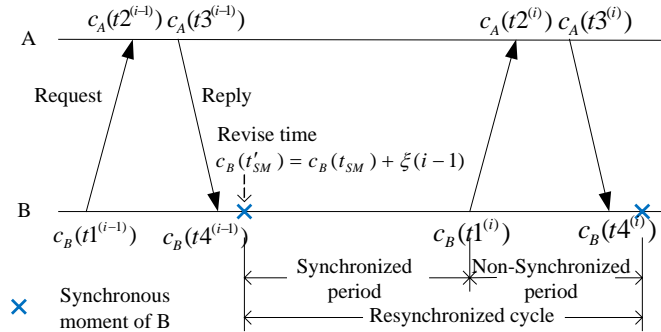


Fig.1 Synchronization scheme of classic two-way packets exchange

图 1 经典双向同步机制

2.2 时钟飘移估计与补偿

线性时间模型^[7,8]是目前最为广泛的时钟飘移模型.线性时间模型假设节点间的相对时钟飘移在一段时间内保持恒定.基于线性时间模型,我们提出了一个时钟飘移估计器.

如图 1 所示,令: $\overline{T_A(i)} = c_A(t3^{(i)}) - c_A(t3^{(i-1)})$, $\overline{T_B(i)} = c_B(t4^{(i)}) - \xi(i-1) - c_B(t4^{(i-1)})$, 以及 $e(i) = \overline{T_A(i)} - \overline{T_B(i)}$.显然, $e(i)$ 是在 $\overline{T_B(i)}$ 时间段内节点 B 落后节点 A 的时间. 根据线性时间模型, $e(i)$ 可被建模为 $e(i) = k_{B,A} \times \overline{T_B(i)} + w(i)$. 其中, $w(i)$ 为高斯白噪声, $w(i) \sim N(0, \sigma^2)$.

以下方法可估计出 $k_{B,A}$. 令: $E = (e(1), e(2), \dots, e(n))^T$, $H = (\overline{T_B(1)}, \overline{T_B(2)}, \dots, \overline{T_B(n)})^T$. 这里的 n 为缓冲区的尺寸,其值设置过小,采样数不足,导致估计值抖动剧烈;设置过大,估计值不能反映晶体振荡器的短期动态性.文献[9]经过在Mica2 平台^[10]上的测量后认为 8 较合适.第 5 节的实验也采纳了该值.

根据线性模型的估计理论^[11],可得:

$$\hat{k}_{B,A} = (H^T H)^{-1} H^T E \quad (2)$$

是 $k_{B,A}$ 的最小方差无偏估计.

在获得了时钟漂移后,时钟漂移补偿则易于进行.以图 1 进行说明.设第 $i-1$ 轮同步点时刻之前节点 B 的时间为 $c_B(t_{SM})$,同步点时刻之后节点 B 的时间为 $c_B(t'_{SM})$,补偿的时间点在第 $i-1$ 轮同步点之后到第 i 轮同步点之前,若节点 B 的本地时间为 $c_B(t)$,即 $t'_{SM} < t < t4^{(i)}$.则时钟漂移补偿过程可用下式表示:

$$c_B(t) \leftarrow c_B(t) + \hat{k}_{B,A}((c_B(t) - c_B(t'_{SM}) + (c_B(t_{SM}) - c_B(t4^{(i-1)}))) = c_B(t) + \hat{k}_{B,A}((c_B(t) - \xi(i-1) - c_B(t4^{(i-1)}))) \quad (3)$$

3 TPLSN 协议

为达到与时间基准节点之间的同步,节点需要获知其全局时钟漂移信息.在介绍了 TPLSN 协议的概要和对双向同步机制的两处改进后,第 3.3 节介绍了如何根据局部时钟漂移计算出全局时钟漂移.

3.1 TPLSN协议概要

图 2 描述了 TPLSN 协议的概貌.节点 N 需要和时间基准节点(节点 0)同步时,按照双向同步机制,它发出一个同步请求报文.其同步路径上的父节点在接收到该报文后,缓存报文中携带的时间信息,并继续向其父节点发送新的同步请求报文.该过程持续进行,直至时间基准节点接收到同步请求报文.随后时间基准节点立即返回一个同步应答报文.其子节点接收到该报文后,获取其中携带的时间信息,计算出全局时间偏移,修改自己的时间以达到和时间基准节点同步,并随即向其子节点发送一个同步应答报文,该报文中携带了与该同步应答报文相对应的同步请求报文中的时间信息.该过程继续进行直到节点 N 接收到同步应答报文时为止.明显的,实现 n 跳线性网络上所有节点的时间同步只需 $2n$ 个同步报文,低能耗的优点显而易见.

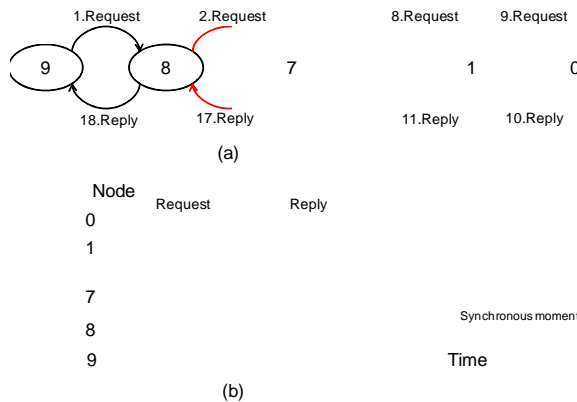


Fig.2 (a) Synchronization procedure of TPLSN; (b) Space-Time diagram of TPLSN

图 2 (a) TPLSN 协议的同步过程;(b) TPLSN 协议的时空图

3.2 对双向同步机制的两处改进

非同步期过长以及在非同步期内可能发生较大时间突变,是造成经典的双向同步机制精度低的重要原因.尤其对于 TPLSN 协议而言,直接采用经典的双向同步机制时问题会更加突出.从图 2 中可以看出:(1) 对于处于同步路径始端的节点(例如节点 N)来说,同步请求和同步应答报文之间的时延过长,时钟漂移在非同步期内带来的误差不可忽略,并且路径越长,误差效应越明显^[12].(2) 节点与其父节点之间同步操作使得节点时间发生突变,而该突变时刻(即其同步点时刻)发生在其子节点的非同步期内,这严重影响了子节点的同步精度.这两点正是经典双向同步机制力图避免的情况,因此 TPLSN 协议必须要克服这两个弊端.

图 3 把节点与其父节点之间的同步操作单独分离出来进行研究(不再明确地表达出同步轮数标记,因为所有时间都在同一轮内).设节点 A 是节点 B 同步路径上的父节点.节点 A 在其同步期内时间发生了突变,即时间增加了 λ .假设节点 B 的非同步期的时间很长,时钟漂移在此期间的误差不可忽略.在这种情况下,节点 B 的时间修改量则不能简单地采用式(1)来进行计算.

我们来推导合理的修改量的表达式. $S_A, P_{A \rightarrow B}$ 和 R_B 分别代表报文发送延迟, 传播延迟与接收延迟. $D_{t1}^{B \rightarrow A}$ 代表在实时时间 $t1$ 时, 节点 B 相对于节点 A 的时间偏移.

由图 3 可得:

$$c_A(t2) = c_B(t1) + S_B + P_{B \rightarrow A} + R_A + D_{t1}^{B \rightarrow A} \tag{4}$$

$$c_B(t4) = c_A(t3) + S_A + P_{A \rightarrow B} + R_B - D_{t3}^{B \rightarrow A} \tag{5}$$

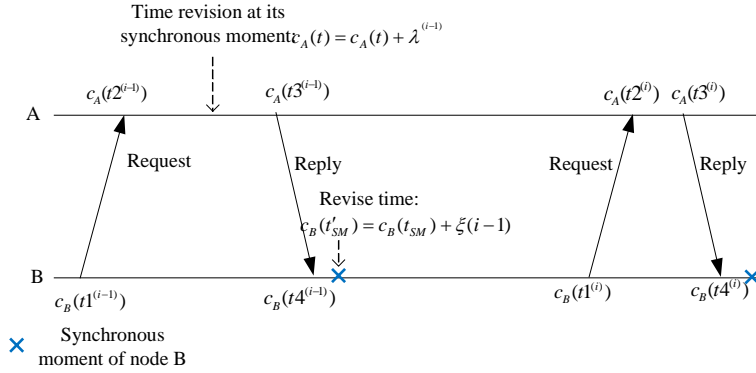


Fig.3 Synchronization procedure of TPLSN between neighbor nodes

图 3 TPLSN 中相邻节点间的同步过程

由图 3, 在节点 B 的非同步期内, 当节点 A 的本地时间增加 λ 后, $D_{t1}^{B \rightarrow A}$ 则可分解为

$$D_{t1}^{B \rightarrow A} = RD_{t1 \rightarrow t3}^{B \rightarrow A} + D_{t3}^{B \rightarrow A} - \lambda \tag{6}$$

其中, $RD_{t1 \rightarrow t3}^{B \rightarrow A}$ 代表在 $t1 \sim t3$ 时间段内, 由时钟漂移所导致的时间偏移变化量.

式(4)–式(5), 可得:

$$\frac{(c_A(t2) - c_B(t1)) - (c_B(t4) - c_A(t3))}{2} = \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + \frac{D_{t3}^{B \rightarrow A} + D_{t1}^{B \rightarrow A}}{2} \tag{7}$$

其中, $S^{UC} = S_B - S_A, P^{UC} = P_{B \rightarrow A} - P_{A \rightarrow B}$ 和 $R^{UC} = R_A - R_B$.

由于 $D_{t3}^{B \rightarrow A} \approx D_{t4}^{B \rightarrow A} = -D_{t4}^{A \rightarrow B} = T_{t4}^A - T_{t4}^B, RD_{t1 \rightarrow t3}^{B \rightarrow A} \approx RD_{t1 \rightarrow t4}^{B \rightarrow A}$. 结合式(6), 可得:

$$\frac{(c_A(t2) - c_B(t1)) - (c_B(t4) - c_A(t3))}{2} = \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + \frac{RD_{t1 \rightarrow t4}^{B \rightarrow A}}{2} - \frac{\lambda}{2} + D_{t4}^{B \rightarrow A}.$$

注意: $D_{t4}^{B \rightarrow A}$ 就是在 $t4$ 时刻节点 B 相对于节点 A 的时间偏移, 也是我们希望获得的目标. 若令

$$\xi = \frac{(c_A(t2) - c_B(t1)) - (c_B(t4) - c_A(t3))}{2} - \frac{RD_{t1 \rightarrow t4}^{B \rightarrow A}}{2} + \frac{\lambda}{2} \tag{8}$$

为节点 B 在其同步点时刻的时间修改量, 则在同步点时刻的瞬时同步误差为

$$Error = \xi - D_{t4}^{B \rightarrow A} = \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} \tag{9}$$

而 $RD_{t1 \rightarrow t4}^{B \rightarrow A}$ 可如下计算:

$$\begin{aligned} RD_{t1 \rightarrow t4}^{B \rightarrow A} &= D_{t1}^{B \rightarrow A} - D_{t4}^{B \rightarrow A} + \lambda = c_A(t1) - c_B(t1) - c_A(t4) + c_B(t4) + \lambda \\ &= c_B(t4) - c_B(t1) - (c_A(t4) - c_A(t1)) - \lambda \\ &= (c_B(t4) - c_B(t1)) \left(1 - \frac{f_A}{f_B} \right) = -(c_B(t4) - c_B(t1)) \hat{k}_{B,A}. \end{aligned}$$

与式(1)相比, 式(8)中增加了两项. 其中 $\frac{RD_{t1 \rightarrow t4}^{B \rightarrow A}}{2}$ 项解决了同步期过长的问题, $\frac{\lambda}{2}$ 项解决了父节点在同步期内的时间突变问题.

需要注意的是:此处的 $\hat{k}_{B,A}$ 和第2.2节中的时钟飘移估计器略有差别.考虑了节点A时间的突变后, $\overline{T_A(i)}$ 和 $\overline{T_B(i)}$ 的定义略有变化.即 $\overline{T_A(i)} = c_A(t3^{(i)}) - c_A(t3^{(i-1)}) - \lambda^{(i-1)}$, $\overline{T_B(i)} = c_B(t4^{(i)}) - \xi(i-1) - c_B(t4^{(i-1)})$.其中 $\xi(i-1)$ 也即式(8),只是把同步轮数明确表示出来而已.

3.3 全局时钟飘移的估计

按照式(3)的时钟飘移补偿方法,为获得和时间基准节点之间的同步,必须获得节点的全局时钟飘移.

以图2中的节点I为例来说明如何获得全局时钟飘移.根据全局时钟飘移的定义,有:

$$k_{I,0} = \frac{f_0}{f_I} - 1 = \frac{f_0}{f_{I-1}} \times \frac{f_{I-1}}{f_I} - 1 = (k_{I-1,0} + 1)(k_{I,I-1} + 1) - 1 \quad (10)$$

如果 $\hat{k}_{I-1,0}$ 是 $k_{I-1,0}$ 的无偏估计,且 $\hat{k}_{I,I-1}$ 是 $k_{I,I-1}$ 的无偏估计,则 $\hat{k}_{I,0} = (\hat{k}_{I-1,0} + 1)(\hat{k}_{I,I-1} + 1) - 1$ 是 $k_{I,0}$ 的无偏估计.

该计算过程易于实现,从时间基准节点开始,每个节点只要把其全局时钟飘移捎带在发给子节点的同步应答报文中即可.子节点计算出其局部时钟飘移后,根据式(10)即可计算出其全局时钟飘移.

4 影响同步误差随跳数累积的因素

如前所述,同步误差随跳数累积的现象将会使得远离时间基准节点的节点同步误差较大.因此,本节探讨影响TPLSN协议同步误差随跳数累积的因素,为进一步降低TPLSN协议同步误差提供理论指导.本节首先分析影响相邻节点间时间偏移的因素,基于得到的结论,再分析影响同步误差随跳数累积的因素.

4.1 影响相邻节点间时间偏移的因素

4.1.1 近似精度因素和时钟飘移因素

情况1. $k_{B,A} > 0$.也即:节点A的时钟速率快于节点B.此时,

$$D_{t3}^{B \rightarrow A} = T_{t3}^A - T_{t3}^B < T_{t4}^A - T_{t4}^B = D_{t4}^{B \rightarrow A} \quad (11)$$

类似于式(6),有:

$$D_{t1}^{B \rightarrow A} = RD_{t1 \rightarrow t4}^{B \rightarrow A} + D_{t4}^{B \rightarrow A} - \lambda \quad (12)$$

$$D_{t3}^{B \rightarrow A} = RD_{t3 \rightarrow t4}^{B \rightarrow A} + D_{t4}^{B \rightarrow A} \quad (13)$$

很明显,由于 $k_{B,A} > 0$,因此 $RD_{t3 \rightarrow t4}^{B \rightarrow A} < 0$, $RD_{t1 \rightarrow t4}^{B \rightarrow A} < 0$.

由式(7),结合式(11)~式(13),得:

$$\frac{(c_B(t2) - c_A(t1)) - (c_B(t4) - c_A(t3))}{2} = \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + \frac{RD_{t1 \rightarrow t4}^{B \rightarrow A}}{2} + D_{t4}^{B \rightarrow A} - \frac{\lambda}{2} + \frac{RD_{t3 \rightarrow t4}^{B \rightarrow A}}{2}$$

结合式(8),可得:

$$\xi = \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + D_{t4}^{B \rightarrow A} + \frac{RD_{t3 \rightarrow t4}^{B \rightarrow A}}{2} \quad (14)$$

由于 $RD_{t3 \rightarrow t4}^{B \rightarrow A} < 0$,因此,

$$\xi < \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + D_{t4}^{B \rightarrow A} \quad (15)$$

情况2. $k_{B,A} < 0$.也即:节点A的时钟速率快于节点B.同理可得:

$$\xi > \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + D_{t4}^{B \rightarrow A} \quad (16)$$

注意: $D_{t4}^{B \rightarrow A}$ 是在 $t4$ 时刻,节点B相对于节点A的时间偏移,也是我们真正希望获得的目标.而 ξ 却是TPLSN协议所计算出的时间偏移.可以看出,两者之间并不相同,这个差异量就是TPLSN协议的同步误差.

如果双向报文交换是对称的,也即: $S^{UC} + P^{UC} + R^{UC} = 0$,则得到:当 $k_{B,A} > 0$ 时, $\xi < D_{t4}^{B \rightarrow A}$.换句话说,节点B欠补偿(补偿不足).同理,当 $k_{B,A} < 0$ 时, $\xi > D_{t4}^{B \rightarrow A}$.换句话说,节点B过补偿(补偿过量).由式(14)~式(16)的比较可以看出:出现这个欠补偿或过补偿的根源在于省略了 $RD_{t3 \rightarrow t4}^{B \rightarrow A} / 2$ 而造成的,也称为近似精度因素.而时钟飘移的性

质则决定了究竟是出现欠补偿还是过补偿.因此,对于 TPLSN 协议来说,近似精度和时钟飘移是导致相邻节点间同步误差的两个因素.

4.1.2 双向报文交换的不对称性因素

双向报文交换的不对称性也会对 TPLSN 的同步精度造成影响.在采取 MAC 时间戳时, $S_A \approx S_B$, $P_{B \rightarrow A} \approx P_{A \rightarrow B}$, 而 R_A 和 R_B 则由处理器的中断响应时间确定.因此,

$$S^{UC} + P^{UC} + R^{UC} \approx R^{UC} = R_A - R_B.$$

当不考虑近似精度因素,即假设 $t_3=t_4$ 时,可得:

$$\xi = \frac{S^{UC}}{2} + \frac{P^{UC}}{2} + \frac{R^{UC}}{2} + D_{t_4}^{B \rightarrow A} = \frac{R_A - R_B}{2} + D_{t_4}^{B \rightarrow A}.$$

因此,如果节点 B 的中断响应时间大于节点 A,就会发生欠补偿现象,否则发生过补偿现象.

表 1 总结了在采用 TPLSN 协议时,上述 3 个因素对两个相邻节点的时间偏移的影响.其中欠补偿用“-”表示,而过补偿用“+”来表示.

Table 1 Influence of three factors on time offset

表1 三个因素对时间偏移的影响

Influence factor	Approximated accuracy	Asymmetry of two-way packets exchange	
		$R_A < R_B$	$R_A > R_B$
Clock skew			
$k_{B,A} > 0$	-	-	+
$k_{B,A} < 0$	+	-	+

4.2 影响同步误差随跳数累积的因素

由于两个相邻节点之间的时钟飘移对它们同步点时刻的时间偏移有很大影响,因此对一个远离时间基准节点的节点来说,其到达时间基准节点的路径上的节点的时钟频率的顺序就对该节点的同步误差有很大影响.这里以图 4 为例进行说明.

假定节点 0 为时间基准节点,节点 J 的时钟频率为 $f_J (J = 0, \dots, n)$.如果由节点 K+1 向节点 K 发出同步请求,则节点 K+1 和节点 K 分别等价与图 1 中的节点 B 和节点 A.

对每个节点对 $[J, J + 1] (J = 0, \dots, N - 1)$,如果时间偏移同向,也即均为欠补偿或均为过补偿,同步误差将会快速累积,造成节点 N 的同步误差很大.相反地,如果节点对的时间偏移反向,则累积速度会降低.具体如图 4 所示.

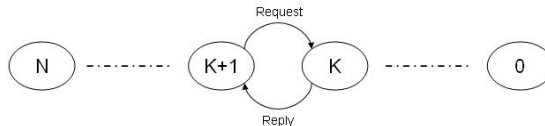


Fig.4 Synchronization error accumulation over hop count

图 4 同步误差随跳数的累积

情况 1. $f_J > f_{J+1} (J = 0, \dots, N - 1)$.也即对每对相邻节点都有 $k_{B,A} > 0$.节点 0 的频率最快,而节点 N 的频率最慢,这种情况称为降序排列.由表 1 可知,同步误差在 $R_A < R_B$ 时将快速累积.相反地,如果 $R_A > R_B$,累积速度将会相对慢些.

情况 2. $f_J < f_{J+1} (J = 0, \dots, N - 1)$.也即对每对相邻节点都有 $k_{B,A} < 0$.此时节点 N 的频率最快,而节点 0 的频率最慢,这种情况称为升序排列.由表 1 可知,同步误差在 $R_A > R_B$ 时将快速累积.相反地,如果 $R_A < R_B$,累积速度将会相对慢些.

5 实验

对于 TPLSN 协议来说,任意时刻的同步误差为同步点处的同步误差与非同步期内由时钟飘移造成的同步误差这二者之和.因此,我们对 TPLSN 协议同步点处的同步误差和非同步期内的由时钟飘移造成的同步误差分别进行实验研究.我们还研究了同步点处同步误差随跳距的累积现象,以了解 TPLSN 协议在多跳线性网络中同

步误差随跳距的增长特性.

5.1 实验平台

我们采用了 GAINS 节点作为实验平台.GAINS 节点硬件兼容 Mica2,根据 Mica2 的开放原理图而设计.所采用的晶体振荡器最大时钟飘移为 70ppm(parts per million).

精确地测量两个节点之间的时间偏移,关键在于如何让节点同时记录下各自的本地时间.我们首创的时间偏移测量方法和节点特性密不可分.GAINS 节点的主处理器 Atmega128L 的外部中断引脚具有“自中断”功能:当外部中断引脚被配置为中断输入时,仍可以由软件强制在该引脚上输出任意电平,并且当输出电平满足中断触发条件时,还会同时触发该外部中断.因此,当把两个节点的中断输入引脚连接在一起,并让其中一个节点主动输出一个触发脉冲,则该触发脉冲将会同时触发这两个节点的外部中断.在中断服务程序中记录下节点的本地时刻,从而可获知节点间的相对时间偏移.

另一个难点是对时钟飘移的估计精度问题.时钟飘移的估计精度直接影响着实验结果的正确性.在我们的实验中,为了尽可能地提高计时精度,计时脉冲的来源设置为时钟节拍(在 GAINS 节点上,每个时钟节拍约为 $1/7.3728\mu\text{s}$),从而时钟值是一个大整数(在我们的实验中被设置为 48 位整数,其最大值为 $2^{48}-1$).相反地,时钟飘移值却非常小,典型值为 $1\sim 100\text{ppm}(10^{-6}\sim 10^{-4})$.明显地,单精度浮点数不能满足它们之间的运算的精度要求.为了解决这个问题,我们在 Atmega128L 平台下把 Softfloat 移植到 TinyOS 上.Softfloat 是符合 IEC/IEEE 浮点数标准的开源软件,可支持 64 位和 80 位浮点数的运算.文献[13]也介绍了一种简化的方法,但其浮点精度却不如 Softfloat 高,也不能满足我们对实验精度的需要.

5.2 实验

5.2.1 实验 1:同步点处同步误差

我们随机选择了 10 个 GAINS 节点,节点编号为 0~9.其中,节点 0 被指定为时间基准节点.这 10 个节点随机线状排列.即节点 i 只能和节点 $i+1$ 、节点 $i-1$ 直接通信.每次重同步操作都由最后一个节点即 9 号节点触发.重同步周期固定为 13s(该值随机选定).每一次同步触发都使得所有节点得到一次同步.

与两个节点的实验不同,根据 TPLSN 协议的时空图,对于除节点 0 外的其余 9 个节点来说,它们的同步点在时间上并不重合.因此我们分别在节点 $i(i=1,\dots,9)$ 的同步点处测量其与节点 0 之间的时间偏移(即全局时间偏移).整个实验持续进行了 5 个小时.全局时间偏移和全局同步误差的均值和均方差统计参数在表 2 中也同时列出.

从表 2 中可以看出,同步点处同步误差基本上随跳距的增长而增长.可以看出:单跳同步误差均值为 $10.34\mu\text{s}$,节点 9 的同步误差均值小于 $20\mu\text{s}$,并且同步误差均值随跳距的增长率小于 $1\mu\text{s}/\text{跳}$.也即,同步点处同步误差随节点跳距增长的速度缓慢.从表 2 最后一列负时间偏移样本数目与样本总个数的比值可以看出:时间偏移有正有负,因此同步点处同步误差随跳距的增长不会呈线性增长.

Table 2 Synchronization error and time offset in random order of clock frequency

表2 在随机序排列下的同步误差与时间偏移

Hop count	Node index	Local clock skew (ppm)	Global clock skew (ppm)	Average global synchronization error (μs)	Standard deviation of global synchronization error (μs)	Average global time offset (μs)	Standard deviation of global time offset (μs)	Δ (%)
0	0	0	0	0	0	0	0	
1	1	-51	-51	10.42	6.69	10.34	6.79	3.22
2	2	-11	-62	9.33	5.31	6.01	8.99	28.2
3	3	2	-60	9.22	7.01	1.17	11.63	37.1
4	4	54	-6	10.40	8.26	-2.96	12.93	66.7
5	5	-45	-51	9.57	9.82	-4.77	12.95	60
6	6	-4	-56	13.78	11.95	-0.61	18.34	52.2
7	7	50	-5	13.64	11.68	-8.99	16.36	18.99
8	8	-46	-51	18.58	16	-8.21	23.29	70.59
9	9	69	17	19.24	14.30	-8.49	22.58	66.7

Δ =样本中全局时间偏移为负数的样本个数/样本总个数(与表 4 和表 5 相同).

为了和 TPSN 协议相比较,我们在同样的环境下实现了 TPSN 协议.测得单跳平均同步误差为 14.6 μ s,节点 9 的同步误差均值为 78.5 μ s,并且同步误差均值随跳数的增长率约为 8 μ s/跳.这说明,对非同步期内的时钟飘移补偿方法即使对非同步期非常短的双向同步机制也起到了大幅提高同步精度的效果.

5.2.2 实验 2:同步误差与重同步周期的关系

现在我们来研究同步期内同步误差与重同步周期的关系.无论是在单跳还是在多跳网络下,同步期内同步误差与重同步周期的关系完全相同,因此只要对两个节点进行实验就足够了.

我们任意选择了两个 GAINS 节点,事先测得它们之间的相对时钟飘移为 26ppm.3 次实验的重同步周期分别为 13,26 和 52s,每次实验均持续了 5 个小时.在同步期内随机采样节点间的时间偏移.具体为:若重同步周期为 \tilde{T} ,则在节点到达同步点后的 $[1, \tilde{T} - 1]$ 秒内随机地进行采样.之所以这样做是为了避开处理器的繁忙期,以获得更加稳定的数据.为了看出时钟飘移补偿的效果,我们还在这两个节点上实现了未作时钟飘移补偿的 TPLSN 协议,以便更直观地说明时钟飘移补偿的必要性和补偿后的效果.表 3 为实验的结果.每个表格单元包含两个数据,其中前者为未作时钟飘移补偿的 TPLSN 协议的同步误差测量结果(以 TPLSN_N 来代表),后者为 TPLSN 协议的同步误差测量结果(以 TPLSN 来代表).

由表 3 可见:在不同的重同步周期下,TPLSN 协议的同步误差均值随重同步周期的增长略有增长.具体为:在未作时钟飘移补偿时,TPLSN 协议的同步误差随重同步周期的增长率为 25.968 μ s/s,而进行了时钟飘移补偿后,相应值则为 0.017 μ s/s.换句话说,经过时钟飘移补偿后,在同步期内任意点的同步误差受重同步周期的影响很小.因此在达到同步后,可以在很长时间内无须再次进行同步操作也能满足绝大多数应用的需求.从而 TPLSN 协议具有极低的同步能耗.

Table 3 Relationship between synchronization error at non-synchronized period and resynchronization cycle

表3 同步期内同步误差与重同步周期的关系

Resynchronization cycle (s)	Average synchronization error (μ s) TPLSN_N/TPLSN	Standard deviation of synchronization error	Minimal synchronization error	Maximal synchronization error
		(μ s) TPLSN_N/TPLSN	(μ s) TPLSN_N/TPLSN	(μ s) TPLSN_N/TPLSN
13	366.79/10.25	150.50/7.01	79.62/0.27	614.01/29.03
26	717.23/10.98	340.32/6.33	68.22/1.22	1292.45/24.55
52	1382.78/11.02	695.62/6.53	70.42/0.0	2159.40/35.67

5.2.3 实验 3:同步误差的累积

根据第 4.2 节的结论,同步误差的累积现象和网络节点的时钟频率的顺序密切相关.因此,与实验 1 采用相同的平台,我们分别在升序和降序排列下,测量了每个节点的全局时钟偏移.实验结果分别见表 4 和表 5.

Table 4 Synchronization error and time offset in ascending order of clock frequency

表4 在升序排列下的同步误差与时间偏移

Hop count	Node index	Local clock skew (ppm)	Global clock skew (ppm)	Average global synchronization error (μ s)	Standard deviation of global synchronization error (μ s)	Average global time offset (μ s)	Standard deviation of global time offset (μ s)	Δ (%)
0	9	0	0	0	0	0	0	
1	0	-16	-16	7.67	5.12	6.33	6.54	15.22
2	7	-2	-22	8.57	6.83	6.47	10.81	25.18
3	4	-1	-23	10.06	8.11	8.93	10.87	13.33
4	1	-44	-68	12.39	10.82	10.44	13.33	16.91
5	5	-1	-68	10.68	10.51	3.34	15.33	37.5
6	8	1	-68	12.14	9.03	2.04	17.29	43.05
7	6	-4	-73	15.30	11.39	-4.03	19.08	61.7
8	3	-4	-77	16.36	13.02	-7.53	21.33	70.37
9	2	-2	-79	18.33	14.98	-12.77	21.20	77.15

Table 5 Synchronization error and time offset in descending order of clock frequency**表5** 在升序排列下的同步误差与时间偏移

Hop count	Node index	Local clock skew (ppm)	Global clock skew (ppm)	Average global synchronization error (μs)	Standard deviation of global synchronization error (μs)	Average global time offset (μs)	Standard deviation of global time offset (μs)	Δ (%)
0	2	0	0	0	0	0	0	
1	3	2	2	11.07	6.40	10.65	7.54	6.85
2	6	4	7	15.57	9.46	14.02	10.92	7.62
3	8	4	11	24.22	9.65	23.81	11.40	5
4	5	0.5	11	27.70	13.13	25.58	14.39	6.56
5	1	1	12	29.08	13.06	28.76	13.82	2.38
6	4	44	56	35.03	16.86	34.42	17.02	5.71
7	7	2	58	45.53	19.09	45.53	19.09	0
8	0	6	63	55.12	20.18	55.12	20.18	0
9	9	17	80	73.76	16.77	73.76	16.77	0

可以看出:相对于降序排列,节点在升序排列下可以获得更低的同步误差.实验结果解释如下:由于每个非时间基准节点都需要做非常耗时的时钟飘移估计操作,因此它们的处理器负载相对时间基准节点来说要重的多,从而导致了非时间基准节点的中断响应时间远大于时间基准节点,也即: $R_A < R_B$. 根据表 2 的结论,在降序排列方式下,双向报文交换的不对称性因素导致了欠补偿,近似精度也导致出现欠补偿.因此,几乎所有节点的全局时间偏移都是负值,并且导致同步误差随跳数的迅速累积.相反地,在升序模式下,这两个因素恰好具有相反的作用.因此,由表 5 可以看出,节点的全局时间偏移有正有负,并导致同步误差随跳数累积相对缓慢.至于实验 1,由于是随机排列,因此其同步误差累积也必定比在降序排列下要慢得多.

6 相关工作与讨论

在多跳RBS协议中,两个节点在需要进行时间同步时,源节点首先建立一条与目的节点之间的最短路径.随后,路径上的相邻节点之间使用单跳RBS协议进行同步.一个携带时间戳信息的报文在沿着路径传播的时候,逐跳地以事后同步方式^[14]进行时间戳转换.在报文最终到达目的节点后,目的节点根据报文中的时间戳获知其与源节点间的时间偏移.RBS协议采用最小二乘法估计时钟飘移.RBS协议的单跳同步误差在Mica2 平台上达到29.3 μs ,同步精度有限并且同步能耗较大.

与RBS协议一样,FTSP协议也是典型的单向同步.FTSP协议采用洪泛的方法来保证强健性.它通过在一个报文的的不同位置处嵌入多个低层时间戳,以及静态的编解码时间排除方法来提高同步精度,在Mica2 平台上达到了1 μs 的单跳精度.由于采用洪泛方式,FTSP是一个高能耗的协议.与FTSP协议相比,虽然TPLSN协议单跳同步精度逊于FTSP协议,但这种差距完全是应为时间戳的方法不同造成的,FTSP的多时间戳方法也可用于TPLSN进一步提高精度;TPLSN协议的同步能耗比FTSP协议要低得多,并且TPLSN协议的同步精度对平台的依赖性弱也决定了它比FTSP协议有更广泛的应用范围.类似地,GCS协议^[15]也采用了散布时间基准节点的时间戳的方法.

TPSN 协议采用了经典的双向同步机制.其创新之处在于证明了发送者-接收者同步方法并不逊色于接收者-接收者同步方法.但其多跳扩展方法较为粗糙.对于 n 跳的线性网络,要达到和 TPLSN 类似的同步精度性能,其同步报文数为至少为 $3n$ 个,并且其同步精度劣于 TPLSN 协议.

表 6 从同步精度、同步能耗、所采用的同步机制和同步精度对 MAC 时间戳的依赖性 4 个角度总结了 TPLSN,TPSN,RBS 和 FTSP 协议之间的区别和特点.

对于时钟飘移估计技术来说,线性时间模型^[16]是最常用的模型.最小二乘法是最常用的方法,并得到广泛应用.Tiny-sync和Mini-sync协议^[16]估计时钟飘移的方法的基本思想为:在双向同步的 4 个时间戳基础上可以得到两个关于时钟飘移的约束集,根据这两个约束集得到了时钟飘移的上下限.时钟飘移的估计值取为上下限的均值.文献[17]提出了一个加权最小二乘时钟飘移估计器,以反映时钟飘移的实时动态性.文献[18]是一个基于数字锁相环的时间同步算法,它并不对时钟飘移进行估计,而是直接跟踪时间偏移的变化来保证同步.

Table 6 Comparison among TPLSN, TPSN, RBS and FTSP**表6** TPLSN,TPSN,RBS和FTSP之间的比较

Protocols performance metrics	TPLSN	TPSN	RBS	FTSP
Synchronization accuracy	Very high	High	Low	Very high
Synchronization energy consumption	Very low	Low	High	High
Synchronization scheme adopted	Enhanced two-way packets exchange	Classic two-way packets exchange	Reference broadcast synchronization	One-Way packet synchronization
Relationship with MAC timestamp	Weak	Weak	Strong	Strong

TPLSN也可被扩展到非线性无线网络中.当节点A需要和节点B同步时,它需要首先建立一条到达节点B的路径,然后在这条路径上运行TPLSN协议即可.因此一个合适的路由协议,例如Ad-hoc On-demand Distance Vector Routing(AODV)协议^[19]是必须的.事实上,由于对经典的双向同步机制进行了改进,TPLSN协议对路由延迟并不敏感.此外,考虑到同步误差随跳数累积的现象,因此在进行路由协议的选择时,首先应考虑的因素是路由跳段数最少,而路由延迟无须过多考虑.而对于由基站节点发起的全网同步的场合,扩展方法可以与TPSN完全相同,只是任何节点可以和其邻居节点以改进的双向报文交换同步方法进行同步.

7 结 论

基于线性时间模型,我们提出了一个最小方差的无偏时钟漂移估计器.该估计器以相邻两个报文的时间戳差异为数据源,因此对传输延迟及传播延迟不敏感,从而获得了更准确的估计值.

在时钟漂移估计技术的基础上,我们对传统双向同步机制进行了两处改进:(1) 对非同步期内的由时钟漂移导致的误差进行补偿;(2) 对非同步期内的节点时间突变进行补偿.从而我们提出的 TPLSN 协议把双向同步几乎无损地扩展到多跳无线网络中.在 GAINS 节点上的实验表明:单跳同步精度可达 $10\mu\text{s}$ 左右,同步误差随跳距的增长率约为 $1\mu\text{s}/\text{跳}$;对于 n 跳同步路径,同步报文个数仅为 $2n$,这也是全程采用双向同步机制最少需要的同步报文数.与 RBS,TPSN 和 FTSP 协议相比,TPLSN 协议同时获得了对同步精度和同步能耗性能的大幅改进.

我们研究了 TPLSN 协议同步误差随跳数累积的现象.首先发现了影响相邻节点间时间偏移的 3 个因素:近似精度、双向报文交换的不对称性和相对时钟漂移,并指出了它们对时间偏移的影响.进而,基于该结论,指出线性网络的时钟序对同步误差的累积至关重要,并对该结论作了实际的实验验证.

References:

- [1] Elson J, Römer K. Wireless sensor networks: A new regime for time synchronization. ACM SIGCOMM Computer Communications Review, 2003,33(1):149–154.
- [2] Sivrikaya F, Yener B. Time synchronization in sensor networks: A survey. IEEE Network, 2004,18(4):45–50.
- [3] Mills DL. Internet time synchronization: The network time protocol. IEEE Trans. on Communications, 1991,39(10):1482–1493.
- [4] Hill JL, Culler DE. Mica: A wireless platform for deeply embedded networks. IEEE Micro, 2002,22(6):12–24.
- [5] Elson J, Girod L, Estrin D. Fine-Grained time synchronization using reference broadcasts. In: Proc. of the 5th Symp. on Operating Systems Design and Implementation. Boston: USENIX, 2002. 147–163.
- [6] Ganeriwal S, Kumar R, Srivastava MB. Timing-Sync protocol for sensor networks. In: Proc. of the 1st ACM Conf. on Embedded Networked Sensor Systems. LA: ACM, 2003. 138–149.
- [7] Yoon S, Veerarittiphan C, Sichert ML. Tiny-Sync: Tight time synchronization for wireless sensor networks. ACM Trans. on Sensor Networks, 2007,3(2):8–es.
- [8] Maróti M, Kusy B, Simon G, Lédeczi A. The flooding time synchronization protocol. In: Proc. of the 2nd ACM Conf. on Embedded Networked Sensor Systems. Baltimore: ACM, 2004. 39–49.
- [9] Ganeriwal S, Tsigkogiannis I, Shim H, Tsiatsis V, Srivastava MB. Estimating clock uncertainty for efficient duty-cycling in sensor networks. IEEE/ACM Trans. on Networking, 2009,17(3):843–856.

- [10] Hill J, Culler D. A wireless embedded sensor architecture for system-level optimization. Intel Research Report, IRB-TR-02-00N, USA: UC Berkeley, 2002.
- [11] Kay SM. Fundamentals of Statistical Signal Processing: Estimation Theory. Prentice-Hall Inc., 1993. 70–105.
- [12] Syed AA, Heidemann J. Time synchronization for high latency acoustic networks. In: Proc. of the 25th IEEE Conf. on Computer Communications. Barcelona: IEEE, 2006. 1–12.
- [13] So HW, Nguyen G, Walrand J. Practical synchronization techniques for multi-channel MAC. In: Proc. of the 12th Int'l Conf. on Mobile Computing and Networking. LA: ACM, 2006. 134–145.
- [14] Elson J, Estrin D. Time synchronization for wireless sensor networks. In: Proc. of the 15th Int'l Parallel and Distributed Processing Symp. San Francisco: IEEE, 2001. 1965–1970.
- [15] Li Q, Rus D. Global clock synchronization in sensor networks. IEEE Trans. on Computers, 2006,55(2):214–226.
- [16] Sichert ML, Veerarittiphan C. Simple, accurate time synchronization for wireless sensor networks. In: Proc. of the IEEE Wireless Communications and Networking Conf. New Orleans: IEEE, 2003. 1266–1273.
- [17] Svein J. Time synchronization in a local area network. IEEE Control Systems Magazine, 2004,24(2):61–69.
- [18] Ren FY, Dong SY, He T, Lin C. A time synchronization mechanism and algorithm based on phase lock loop. Journal of Software, 2007,18(2):372–380. <http://www.jos.org.cn/1000-9825/18/372.htm>
- [19] Perkins CE, Royer EM. Ad hoc on-demand distance vector routing. In: Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications. New Orleans: IEEE, 1999. 90–100.

附中文参考文献:

- [18] 任丰原,董思颖,何滔,林闯.基于锁相环的时间同步机制与算法.软件学报,2007,18(2):372–380 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/18/372.htm>



徐朝农(1975—),男,安徽巢湖人,博士,讲师,主要研究领域为无线传感器网络,嵌入式系统.



邓志东(1966—),男,博士,教授,博士生导师,主要研究领域为计算智能,计算生物学,复杂网络理论,虚拟现实,无线传感器网络与机器人.



徐勇军(1979—),男,博士,副研究员,主要研究领域为无线传感器网络,低功耗系统.