Interval-based Clock Synchronization for Ad-Hoc Sensor Networks

Lennart Meier*

Computer Engineering and Networks Laboratory
Swiss Federal Institute of Technology (ETH) Zurich
CH-8092 Zurich, Switzerland

Abstract. Clock synchronization is a crucial basic service in typical sensor networks, since the observations of distributed sensors more often than not need to be ordered ("*a* happened before *b*") or otherwise related ("*a* and *b* happened within a time window of size *x*") in time. Ad-hoc sensor networks exhibit characteristics which make the use of traditional clock-synchronization algorithms infeasible. More appropriate algorithms have been presented recently.
In this paper, we argue that interval-based synchronization is particularly suitable for ad-hoc sensor networks. Interval-based algorithms maintain lower and upper bounds on time instead of time estimates. We discuss the benefits of the interval-based approach for ad-hoc sensor networks and present recent results.

1 Introduction

Clock synchronization is an important service in typical ad-hoc sensor networks. For example, the correct evaluation of distributed sensor data may require knowledge about the chronology of the sensor observations [13]. In addition, energy consumption can be reduced by synchronous power-on and shutdown of the wireless-communication circuits of a sender–receiver pair [3, 4].

There has been extensive work on clock synchronization in infrastructure-based networks [10, 16, 11]. Ad-hoc (and thus wireless) sensor networks pose some substantially different challenges. *Robustness:* There is no stable connectivity between nodes. *Energy efficiency:* Synchronization can only be achieved and maintained by communication. Communication is expensive in terms of energy, which typically is a scarce resource for sensor nodes. *Ad-hoc deployment:* The clock-synchronization service must not rely on any a-priori configuration or on infrastructure.

The conclusion is that algorithms for infrastructure-based networks cannot be directly applied in ad-hoc sensor networks. Algorithms particularly suitable for such networks have been proposed recently [5, 6, 13, 15, 17, 18]. Most of the proposed algorithms let nodes maintain single-value time estimates. Interval-based synchronization uses guaranteed bounds on time. It was first proposed in [8] and was further studied in [14]. In [1, 9], it was proposed as particularly suited for ad-hoc sensor networks.

* The author was supported by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322.

1.1 Overview

In Sect. 2, we define the system model and state the problem we want to solve. In Sect. 3, we discuss the particular suitability of interval-based synchronization for ad-hoc sensor networks. We present recent results and ongoing work in this area in Sect. 4.

2 System Model and Problem Statement

In this section, we give a model that captures those aspects of an ad-hoc sensor network that are essential to our analysis. We then state the problem of internal and external clock synchronization as we understand it.

2.1 Clock model

Each node of the network is equipped with a local clock h; the reading of node N_i's clock h_i at real time t is denoted as $h_i(t)$. The drift of a clock h_i at time t is defined as the deviation of its speed from the ideal speed 1 and is thus given by

$$\rho_i(t) = \frac{\mathrm{d}h_i(t)}{\mathrm{d}t} - 1 \ . \tag{1}$$

The quality of synchronization that can be achieved depends on the assumptions about the nodes' clocks. We suppose that $\rho_i(t)$ is limited by a constant $\rho^{\max}$ according to

$$-\rho^{\max} \leq \rho_i(t) \leq \rho^{\max} \quad \forall t \ . \tag{2}$$

We require $\rho_i(t) > -1$ for all times t. This means that a clock can never stop ($\rho_i(t) = -1$) or run backward ($\rho_i(t) < -1$). Thus, for two events a, b with $t_a < t_b$ occuring at node N_i (whose clock's drift ρ_i is bounded according to (2)), node N_i can compute bounds $\Delta_i^l[a,b]$, $\Delta_i^u[a,b]$ on the real-time difference $\Delta[a,b] := t_b - t_a$ as

$$\Delta_i^l[a,b] := \frac{h_i(t_b) - h_i(t_a)}{1 + \rho^{\max}} \qquad \Delta_i^u[a,b] := \frac{h_i(t_b) - h_i(t_a)}{1 - \rho^{\max}} \ .$$

2.2 Problem statement

We define the problem of internal synchronization as follows: Given an event occurring at node N_j at local time $h_j(t)$, we are interested in tight bounds $H_i^l(t)$, $H_i^u(t)$ on the local time $h_i(t)$ of another node N_i at time t, such that $H_i^l(t) \leq h_i(t) \leq H_i^u(t)$. In the context of a sensor network, this situation arises when the observations of multiple sensor nodes have to be combined.

The problem of external synchronization consists in providing all nodes in the network with a common system time that is provided from outside the system, e.g. via a GPS receiver.

3 Guaranteed Bounds in Ad-Hoc Sensor Networks

Clock-synchronization algorithms face two problems: The information a node has about the local time of another node degrades over time due to clock drift (see Fig. 1), and its improvement through communication is hindered by message-delay uncertainty. The influence of drift and delay uncertainty can to a large extent be studied separately. A third problem arises from the fact that it is not always clear which node's time shall be used as the reference.

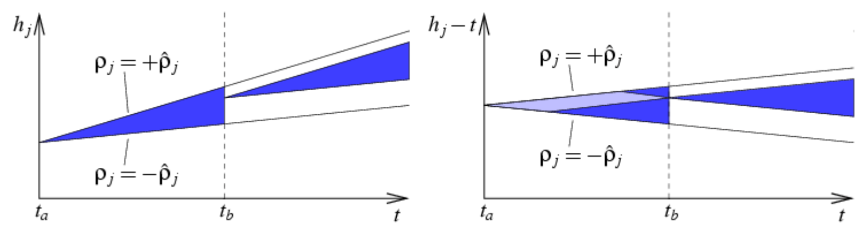

Fig. 1. Graphical representation of the knowledge of node N_i about the local time h_j of node N_j as a function of real time t. The shaded area is the region in which h_j can lie. The two nodes communicate at events a and b. On the right, $h_j - t$ is plotted against t; additionally, the information about past values of h_j that N_i gathers at event b is shown as a lighter-shaded area.

Most of the work on synchronization in ad-hoc networks has concentrated on delay uncertainty; recent algorithms reduce it to a few microseconds [2, 5, 7, 12]. They then achieve good synchronization by continued and frequent communication, which keeps the impact of clock drift negligible.

In settings with *sporadic* communication, the impact of clock drift has to be taken into account explicitly. Sporadic communication is typical for ad-hoc sensor networks where nodes may fail, be temporarily out of reach or abstain from communication to save energy. Our interval-based approach is particularly suited for such settings. Note that we do not neglect delay uncertainty: Methods as in [2, 5, 7, 12] can be used to exchange time bounds with negligible delay uncertainty. Delay and drift uncertainty are orthogonal issues.

Interestingly, the use of guaranteed bounds has not received much attention, although it has a number of advantages over using time estimates: (a) Guaranteed bounds on the local times at which sensor events occurred allow to obtain guaranteed bounds from sensor-data fusion. (b) The concerted action (sensing, actuating, communicating) of several nodes at a predetermined local time of a specific clock always succeeds: each node can minimize its uptime while guaranteeing its activity at the predetermined time. (c) The combination of several bounds for a single local time is unambiguous and optimal, while the reasonable combination of time estimates requires additional information about the quality of the estimates. Note that interval-based synchronization naturally solves the problem of choosing reference nodes in large networks.

The influence of the clock drift on the quality of synchronization may dominate over the influence of the message delays. This is the case in those ad-hoc sensor networks where communication is sporadic not only in the sense of unpredictable, but also in the sense of *infrequent*. With decreasing frequency of communication, the uncertainty due to clock drift increases, while the uncertainty due to message delays remains constant. *A numeric example:* Suppose the message delay contributes 1 millisecond to a node's uncertainty, and the clock drift is bounded by $\rho^{max} = 100$ppm. After 5 seconds, the drift's contribution to the uncertainty equals that of the delay. After one hour, it is 720 times larger. In such settings, neglecting the delay uncertainty is acceptable.

4 Recent Results

In [8], a simple algorithm for interval-based external synchronization was proposed: algorithm IM lets two communicating nodes intersect their time intervals. In [1], a model for the path-based analysis of interval-based clock synchronization was used to show that algorithm IM is worst-case optimal. Furthermore, a new algorithm was presented, the Back-Path Interval-Synchronization Algorithm (BP-ISA). The BP-ISA extends algorithm IM by letting each node keep a history of the intervals from the last communication with every other node. The BP-ISA was shown to be worst-case optimal and to perform significantly better than algorithm IM in the average case. This is due to the typical drift diversity of the nodes' clocks.

In [9], the algorithm for internal synchronization from [13] was improved. Furthermore, it was shown that for optimal synchronization, nodes need to store, communicate and use their entire histories. Clearly, this is infeasible in a realistic scenario. But typically, time information degrades quickly and can thus be discarded without a loss in synchronization quality. Current work is focusing on quantifying the trade-off of synchronization overhead and quality.

Another issue that has not been addressed so far is how node mobility affects interval-based synchronization. Simulation results suggest that mobility has only positive effects. This is due to the special nature of interval-based time information, which makes the combination of two nodes' time information unambiguous and optimal.

5 Conclusion

In this paper, we argued that interval-based synchronization is particularly suited for ad-hoc sensor networks. Most importantly, it naturally solves the problem of choosing reference nodes in large networks. We presented recent advances in interval-based synchronization for ad-hoc sensor networks. Open questions about the overhead–quality trade-off and about the effects of node mobility remain to be investigated.

References

1. Philipp Blum, Lennart Meier, and Lothar Thiele, *Improved interval-based clock synchronization in sensor networks*, Proceedings of the Third International Symposium on Information Proces sing in Sensor Networks (IPSN '04), 2004, pp. 349–358.
2. Philipp Blum and Lothar Thiele, *Clock synchronization using packet streams*, DISC 2002, Brief Announcements (Dahlia Malkhi, ed.), 2002, pp. 1–8.
3. Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris, *Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks*, Mobile Computing and Networking, 2001, pp. 85–96.
4. IEEE Computer Society LAN MAN Standards Committee, *IEEE 802.11: Wireless LAN medium access control and physical layer specifications*, August 1999.
5. Jeremy Elson, Lewis Girod, and Deborah Estrin, *Fine-grained network time synchronization using reference broadcasts*, SIGOPS Oper. Syst. Rev. **36** (2002), no. SI, 147–163.
6. Saurabh Ganeriwal, Ram Kumar, and Mani B. Srivastava, *Timing-sync protocol for sensor networks*, Proceedings of the 1st international conference on Embedded networked sensor systems (SenSys '03), ACM Press, 2003, pp. 138–149.
7. Jason Hill and David Culler, *MICA: A wireless platform for deeply embedded networks*, IEEE Micro **22** (2002), no. 6, 12–24.
8. Keith Marzullo and Susan Owicki, *Maintaining the time in a distributed system*, Proceedings of the second annual ACM symposium on Principles of distributed computing, ACM Press, 1983, pp. 295–305.
9. Lennart Meier, Philipp Blum, and Lothar Thiele, *Internal synchronization of drift-constraint clocks in ad-hoc sensor n etworks*, Proceedings of the Fifth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2004), 2004.
10. David L. Mills, *Bibliography on computer network time synchronization*, available on-line at http://www.eecis.udel.edu/˜mills/bib.html.
11. ———, *Internet time synchronization: The network time protocol*, IEEE Transactions on Communications **39** (1991), no. 10, 1482–1493.
12. Michael Mock, Reiner Frings, Edgar Nett, and Spiro Trikaliotis, *Clock synchronization in wireless local area networks*, Proceedings of the 12th Euromicro Conference on Real Time Systems, June 2000, pp. 183–189.
13. Kay Römer, *Time synchronization in ad hoc networks*, Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing, ACM Press, 2001, pp. 173–182.
14. Ulrich Schmid and Klaus Schossmaier, *Interval-based clock synchronization*, Real-Time Systems **12** (1997), no. 2, 173–228.
15. Mihail L. Sichitiu and Chanchai Veerarittiphan, *Simple, accurate time synchronization for wireless sensor networks*, IEEE Wireless Communications and Networking (WCNC 2003), vol. 2, IEEE, 2003, pp. 1266–1273.
16. Barbara Simons, Jennifer L. Welch, and Nancy A. Lynch, *An overview of clock synchronization*, Fault-Tolerant Distributed Computing (Barbara B. Simons and Alfred Z. Spector, eds.), Lecture Notes in Computer Science, vol. 448, Springer, 1990, pp. 84–96.
17. Fikret Sivrikaya and Bülent Yener, *Time synchronization in sensor networks: A survey*, IEEE Network **18** (2004), no. 4, 45–50.
18. Jana van Greunen and Jan Rabaey, *Lightweight time synchronization for sensor networks*, Proceedings of the 2nd ACM international conference on Wireless sensor networks and ap plications (WSNA '03), ACM Press, 2003, pp. 11–19.