

UNIVERSIDADE FEDERAL FLUMINENSE

HEDER DORNELES SOARES

**SINCRONIZAÇÃO EM REDES SENSORES SEM
FIO**

NITERÓI

2016

UNIVERSIDADE FEDERAL FLUMINENSE

HEDER DORNELES SOARES

SINCRONIZAÇÃO EM REDES SENSORES SEM FIO

Dissertação de Mestrado apresentada ao
Programa de Pós-Graduação em Compu-
tação da Universidade Federal Fluminense
como requisito parcial para a obtenção
do Grau de Mestre em Computação.
Área de concentração:
Redes e Sistemas Distribuídos e Paralelos

Orientador:

CÉLIO VINICIUS NEVES DE ALBUQUERQUE

Co-orientador:

RAPHAEL PEREIRA DE OLIVEIRA GUERRA

NITERÓI

2016

HEDER DORNELES SOARES

SINCRONIZAÇÃO EM REDES SENSORES SEM FIO

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Redes e Sistemas Distribuídos e Paralelos.

Aprovada em Janeiro de 2016.

BANCA EXAMINADORA

Prof. Célio Vinicius Neves de Albuquerque - Orientador, UFF

Prof. Raphael Pereira de Oliveira Guerra, Coorientador, UFF

Prof. <NOME DO AVALIADOR>, <INSTITUIÇÃO>

Prof. <NOME DO AVALIADOR>, <INSTITUIÇÃO>

Prof. <NOME DO AVALIADOR>, <INSTITUIÇÃO>

Niterói

2016

Dedicatória(s): Elemento opcional onde o autor presta homenagem ou dedica seu trabalho (ABNT, 2005).

Agradecimentos

Elemento opcional, colocado após a dedicatória (ABNT, 2005).

Resumo

Elemento obrigatório, constituído de uma sequência de frases concisas e objetivas e não de uma simples enumeração de tópicos, não ultrapassando 500 palavras (ABNT, 2005).

Palavras-chave: Palavras representativas do conteúdo do trabalho, isto é, palavras-chave e/ou descritores, conforme a ABNT NBR 6028 (ABNT, 2005).

Abstract

Wireless sensors networks are distributed system composed of battery-powered nodes with low computational resources. Like in many distributed systems, some applications of WSN require time synchronization among their nodes. In the particular case of WSN, synchronization algorithms must respect the nodes computational constraints. The well known FTSP protocol is famous for achieving nanosecond precision with low overhead. However, it relies on MAC timestamp, a feature not available in all hardware. In this work, we propose MAC timestamp independent version in order to extend and adapt FTSP to work on hardware that do not have MAC timestamp while keeping the low overhead and high synchronization precision our results indicate an average synchronization error of $3\mu s$ per hop, while adding a correction message every three seconds.

Keywords: WSN, Time Synchronization, FTSP, MAC Timestamp.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

RSSF	:	Rede de Sensores Sem Fio;
FTSP	:	Flooding Time Synchronization Protocol;
WSN	:	Wireless Sensor Network;
TI	:	Tecnologia da Informação;
MAC	:	Media access control;
NTP	:	Network Time Protocol;
RBS	:	Reference Broadcast Synchronization;
TDMA	:	Time Division Multiple Access;
TPSN	:	Timing-sync Protocol for Sensor Networks;

Sumário

1	Introdução	1
1.1	Sincronização em Redes Sensores Sem Fio	2
1.1.1	Conceitos Básicos	3
1.1.2	Mensagens de Sincronização	4
1.1.3	Fontes de Imprecisão em Comunicação	4
1.2	Definição do Problema	5
1.3	Motivação do Trabalho	5
1.4	Contribuição da Dissertação	5
1.5	Organização do Documento	5
2	Trabalhos Relacionados	7
2.1	RBS	7
2.2	LTS	7
2.3	TPSN	7
2.4	PulseSync	8
2.5	FTSP	8
3	Revisão detalhada do FTSP	9
3.1	<i>Timestamp</i>	9
3.2	Escorregamento do Relógio	9
3.3	Sincronização Multi-saltos	10
4	FTSP+	11

4.1	Técnica	11
4.2	Modificação	11
5	Implementação	13
5.1	Sistema Operacional	13
5.2	Diagramas e Componentes	13
6	Experimentos	14
6.1	Experimento 1	14
6.2	Experimento 2	14
6.3	Experimento 3	15
7	Conclusão e Trabalhos Futuros	16
	Referências	17
	Apêndice A - <TÍTULO DO APÊNDICE>	19

Capítulo 1

Introdução

- | |
|--|
| <ul style="list-style-type: none">• WSN são redes de sensores com recursos . . . |
|--|

As Redes de Sensores Sem Fio (RSSF) são compostas de pequenos dispositivos equipados com uma antena de comunicação sem fio, um ou mais tipos de sensores, e uma CPU de baixa capacidade de processamento de dados [2]. Estes dispositivos geralmente são chamados de *motes*. Devido ao alcance do sinal de rádio limitado e da restrição energética uma RSSF tem limitações e características únicas que a difere de redes de computadores e sistemas distribuídos tradicionais [1].

As RSSF tem várias aplicações em diversos campos. A implantação de sensores em aplicações militares foi sempre muito difundido, de modo que a introdução de *motes* era uma incorporação natural para o avanço dos sistemas já utilizados. Aplicações que foram aprimoradas com o uso de RSSF incluem o rastreamento de inimigos e alvos [21], monitoramento de veículos [16], sistema contra atirador [15] e sistemas de vigilância [8]. Monitoramento ambiental também provê oportunidades para aplicação de redes sensores sem fio. Nosso meio ambiente tem uma gama muito grande de informações que desempenham um papel importante em nossa qualidade de vida, como a qualidade do ar, água, som e radiação solar que estamos expostos todos os dias e que diretamente afetam nossa saúde [13, 3]. Com o interesse cada vez maior em computação verde nos leva as preocupações com o consumo de energia elétrica em instalações de TI [4], e as RSSF têm um papel estratégico no monitoramento e controle destes ambientes [22].

Algumas dessas aplicações requerem mecanismos de sincronização de tempo com boa

precisão e escalabilidade, tudo isso em conformidade com os seus baixos recursos computacionais e disponibilidade energética [22, 15, 21]. Métodos de sincronização tradicionais, como NTP [11], amplamente utilizado em servidores e clientes de rede não se aplicam em RSSF devido a fatores não deterministas relacionados com o acesso ao meio na transmissão sem fio dos sensores. O Flooding Time Synchronization Protocol (FTSP) é o algoritmo de sincronização de relógio mais popular para RSSF [10], ele é tolerante a falhas, consegue alta precisão ($\sim 1,5\mu s$ por salto) utilizando *timestamps* em camadas baixas da pilha de rádio, usa a técnica de regressão linear para compensar o escorregamento do relógio usando poucas mensagens pela rede. Contudo, *timestamps* na camada MAC não é um recurso padronizado, e portanto, não interoperáveis entre diferentes hardware e protocolos da camada física. Há um esforço do Google para padronizar *timestamping* na camada MAC em RSSF [20], mas até agora não há muita conformidade.

Muitos protocolos de sincronização usam MAC timestamp: alguns têm menos precisão do que o FTSP, eles se concentram em outros problemas e fazem suposições mais restritivas [7, 19]; outros podem conseguir uma melhor precisão entre os nós distantes [12, 9, 17]. Elson et al. propôs o Reference Broadcast Synchronization [6] (RBS) para eliminar a incerteza do remetente sem MAC timestamp removendo o *sender* do caminho crítico. A idéia é que um terceiro irá transmitir um *beacon* para todos os receptores. O *beacon* não contém qualquer informação de tempo; em vez disso os receptores irão comparar seus relógios um ao outro para calcular o escorregamento de seus relógios. Tem $\sim 30\mu s$ erro por salto e é independente da MAC timestamp. Ranganathan e Nygard oferecem uma boa visão geral desses protocolos [14].

1.1 Sincronização em Redes Sensores Sem Fio

Razões e desafios. NTP e GPS, Necessidade de algoritmos de sincronização específicos para RSSF.

- Intro sobre redes sensores sem fio.
- Necessidade da Sincronização em sistemas distribuídos.
- Problemas inerentes a sincronização, fontes de atraso e imprecisão em RSSF

Em sistemas distribuídos como as redes de sensores sem fio, cada nó tem seu próprio relógio e sua própria percepção de tempo. No entanto, uma escala de tempo comum entre nós sensores é importante para identificar relações entre eventos que estejam sendo

monitorados, para apoiar a eliminação de dados redundantes de sensores e para facilitar a operação da rede. Uma vez que cada nó em uma rede de sensores opera de forma independente e conta com o seu próprio relógio, as leituras do relógio de diferentes nós de sensores também será diferente. Além destas diferenças aleatórias, a diferença entre os relógios de sensores diferentes vai aumentar ainda mais devido às taxas de escorregamento de diferentes osciladores. Portanto, o relógio ou tempo de sincronização é necessário para assegurar que os tempos de detecção podem ser comparados de uma forma significativa.

Embora as técnicas de sincronização de tempo para redes com fios receberam uma quantidade significativa de atenção, esses métodos não são apropriados para uso em RSSF sem modificação, devido aos desafios colocados pelos ambientes sensores sem fio. Estes desafios incluem o tamanho das redes de sensores, a necessidade de auto-configuração e robustez, a mobilidade dos sensores além da necessidade primordial de conservação de energia [5].

Nas RSSF a eficiência energética é uma necessidade básica para seu funcionamento, em uma rede de larga escala não é possível fornecer uma fonte de energia para toda a rede, esses sensores geralmente são acoplados a uma bateria. Devido ao tamanho dos dispositivos a quantidade de energia que eles podem produzir ou armazenar é muito limitada [18].

Sincronização de tempo é um serviço necessário para muitas aplicações e serviços em sistemas distribuídos em geral. Numerosos protocolos de sincronização de tempo têm sido propostos para ambos os sistemas com e sem fio, por exemplo, o Network Time Protocol (NTP) [11] é uma abordagem de sincronização escalável, robusto e auto-configurável amplamente utilizado. Especialmente em combinação com o Sistema de Posicionamento Global (GPS), tem sido utilizado para alcançar a precisão da ordem de alguns microssegundos. No entanto, abordagens como NTP não são adequados para RSSFs devido a ser ineficiente neste contexto pois necessita de maior quantidade de memória. Já o uso de GPS pode elevar o custo de implantação de um sistema de monitoramento, além de requerer vários minutos para sintonizar, GPS também necessita de um uso maior de energia.

1.1.1 Conceitos Básicos

Efeitos do Ambiente. Restrição energética. Mobilidade e acesso ao meio sem fio. Capacidade dos dispositivos.
--

Os relógios dos *motes* nas RSSF são baseados em osciladores de quartzo, o tempo é mensurado nas oscilações do cristal em conjunto a um contador, este contador tem seu valor decrementado até atingir zero, então ele reinicia novamente para seu valor padrão e o processo continua indeterminadamente, toda vez que o contador chega a zero é gerado uma interrupção e essa interrupção é chamada de *tick* ou *clock tick*, ela incrementa outro contador o do relógio a nível de *software* que é de onde vem tempo que utilizamos no sistema operacional. Com isso, esse relógio de *software* fornece para o *mote* um valor de tempo local (*local time*).

Supondo a existência de dois nós, cada um com o seu próprio relógio, onde $R(t)$ simboliza o valor do relógio no tempo t , sendo t o tempo de referência. Se verificarmos $R(t)$ de ambos os nós podemos encontrar uma diferença entre os tempos, essa diferença é chamada de *clock offset*. Para corrigir esta diferença é necessário realizar uma sincronização para que os tempos sejam iguais ou que a diferença seja tão pequena quanto possível.

1.1.2 Mensagens de Sincronização

Pairwise synchronization, One-Way Message Exchange, Two-Way Message Exchange, Receiver-Receiver Synchronization

1.1.3 Fontes de Imprecisão em Comunicação

Send delay: Access delay: Propagation delay: Receive delay:

1.2 Definição do Problema

Dependência de hardware, padronização do recurso, esforço para que exista futuramente um padrão que esteja presente em todos os tipos de *motes* e possibilite assim a interoperação de protocolos de MAC *timestamp* em sistemas heterogêneos.

1.3 Motivação do Trabalho

- Devido a diversidade de dispositivos e o bom desempenho do FTSP, torna-lo um protocolo independente de recurso de hardware possibilitando o seu funcionamento em *motes* sem o recurso de MAC timestamp.

1.4 Contribuição da Dissertação

- Fazer o FTSP funcionar com timestamp a nível de aplicação.

1.5 Organização do Documento

Esta dissertação é composta por seis capítulos. Descrever no final o que cada capítulo apresenta.

- **Capítulo 2:** Descrever conteúdo aqui.
- **Capítulo 3:** Descrever conteúdo aqui.
- **Capítulo 4:** Descrever conteúdo aqui.
- **Capítulo 5:** Descrever conteúdo aqui.
- **Capítulo 6:** Descrever conteúdo aqui.

- **Capítulo 7:** Descrever conteúdo aqui.

Capítulo 2

Trabalhos Relacionados

Falar sobre os protocolos abaixo:

2.1 RBS

É um método no qual o receptor usa as transmissões da camada física para comparar os relógios. Usa o tempo de chegada do pacote como tempo de referência de sincronização.

2.2 LTS

Lightweight Tree-Based Synchronization.

2.3 TPSN

TPSN não elimina incerteza no sender, apenas minimiza o erro. Ele tenta reduzir este não determinismo utilizando timestamping da mensagem na camada MAC.

2.4 PulseSync

No trabalho [9] é realizado um estudo sobre o FTSP e é observado que o protocolo apresenta um crescimento do erro de sincronização baseado no diâmetro da rede. Então propõe o protocolo PulseSync para melhorar esse cenário. Também é dependente de MAC timestamping.

2.5 FTSP

Descrição de alto nível sobre o FTSP.

Capítulo 3

Revisão detalhada do FTSP

- Explicar a técnica.
- Inserir ilustração do processo de sincronização com o relógio externo.
- Pseudo código
- FTSP+ e subsection

3.1 *Timestamp*

Explicar como funciona a marcação de tempo no FTSP

3.2 Escorregamento do Relógio

Descrever os problemas relacionados ao clock drift. Como o FTSP corrige essa fonte de imprecisão.

3.3 Sincronização Multi-saltos

- Formato da mensagem de sincronização.
- Gerenciamento de informação redundante.
- Eleição do nó raiz.
- Evento: enviar/receber mensagem de sincronia.

Capítulo 4

FTSP+

Intro

- Explicar a técnica.
- Inserir ilustração do processo de sincronização com o relógio externo.
- Pseudo código
- FTSP+ e subsection

4.1 Técnica

Descrever o processo de cálculo do tempo de acesso ao meio, e onde foi baseado.

4.2 Modificação

Pseudo código com explicação do passo-a-passo do algoritmo (no sender e no receiver).

Capítulo 5

Implementação

Descrever o processo de Implementação no tinyos (libs, organização do código e TEPs).

5.1 Sistema Operacional

- Gerencia de memória.
- Gerenciamento energético.
- Redes.
- Linguagem.
- Manipulação de interrupções.
- Programação baseada em eventos.

Tinyos

5.2 Diagramas e Componentes

O Tinyos tem o recurso de componentes para disponibilizar seus mais diversos recursos, descrever os componentes relevantes para o trabalho e os componentes resultantes das alterações.

Capítulo 6

Experimentos

Listar o experimentos de maneira numérica 1,2 e 3. Em cada item explicar os exp em si.

- O que eu quero responder com os experimentos.
- Quais métricas utilizar.
- Definir ambiente.
- Colocar os gráficos e tabelas gerados pelos experimentos.
- Comentar e analisar os resultados.

6.1 Experimento 1

Descrever o ambiente, tipo de sensor, tempo de duração e demais características do cenário testado.

6.2 Experimento 2

Descrever o ambiente, tipo de sensor, tempo de duração e demais características do cenário testado.

6.3 Experimento 3

Descrever o ambiente, tipo de sensor, tempo de duração e demais características do cenário testado.

Capítulo 7

Conclusão e Trabalhos Futuros

...

Referências

- [1] ARAMPATZIS, T.; LYGEROS, J.; MANESIS, S. A survey of applications of wireless sensors and wireless sensor networks. In *IEEE International Symposium on Intelligent Control, Mediterrean Conference on Control and Automation* (2005), IEEE, pp. 719–724.
- [2] BARONTI, P.; PILLAI, P.; CHOOK, V. W. C.; CHESSA, S.; GOTTA, A.; HU, Y. F. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and zigbee standards. *Comput. Commun.* (2007), 1655–1695.
- [3] CARDELL-OLIVER, R.; SMETTEM, K.; KRANZ, M.; MAYER, K. Field testing a wireless sensor network for reactive environmental monitoring [soil moisture measurement]. In *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004. Proceedings of the 2004* (Dec 2004), pp. 7–12.
- [4] CHONG, F.; HECK, M.; RANGANATHAN, P.; SALEH, A.; WASSEL, H. Data center energy efficiency:improving energy efficiency in data centers beyond technology scaling. *Design Test, IEEE* 31, 1 (Feb 2014), 93–104.
- [5] ELSON, J.; ESTRIN, D. Time synchronization for wireless sensor networks. In *Parallel and Distributed Processing Symposium., Proceedings 15th International* (April 2001), pp. 1965–1970.
- [6] ELSON, J.; GIROD, L.; ESTRIN, D. Fine-grained network time synchronization using reference broadcasts. *SIGOPS Oper. Syst. Rev.* 36, SI (Dec. 2002), 147–163.
- [7] GANERIWAL, S.; KUMAR, R.; SRIVASTAVA, M. B. Timing-sync protocol for sensor networks. In *1st International Conference on Embedded Networked Sensor Systems* (2003), ACM, pp. 138–149.
- [8] GUI, C.; MOHAPATRA, P. Power conservation and quality of surveillance in target tracking sensor networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, MobiCom '04*, ACM, pp. 129–143.
- [9] LENZEN, C.; SOMMER, P.; WATTENHOFER, R. Optimal clock synchronization in networks. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems* (2009), ACM, pp. 225–238.
- [10] MARÓTI, M.; KUSY, B.; SIMON, G.; LÉDECZI, Á. The flooding time synchronization protocol. In *Proceedings of the 2nd international conference on Embedded networked sensor systems* (2004), ACM, pp. 39–49.
- [11] MILLS, D. Internet time synchronization: the network time protocol. *Communications, IEEE Transactions on* 39, 10 (Oct 1991), 1482–1493.

- [12] NAZEMI GELIAN, S.; EGHBALI, A.; ROUSTAPOOR, L.; YAHYAVI FIROUZ ABADI, S.; DEHGHAN, M. Sltp: Scalable lightweight time synchronization protocol for wireless sensor network. In *Mobile Ad-Hoc and Sensor Networks*, vol. 4864. Springer Berlin, 2007, pp. 536–547.
- [13] OLIVEIRA, L. M.; RODRIGUES, J. J. Wireless sensor networks: a survey on environmental monitoring. *Journal of communications* 6, 2 (2011), 143–151.
- [14] RANGANATHAN, P.; NYGARD, K. Time synchronization in wireless sensor networks: a survey. *International journal of UbiComp (IJU)* 1, 2 (2010), 92–102.
- [15] SIMON, G.; MARÓTI, M.; LÉDECZI, Á.; BALOGH, G.; KUSY, B.; NÁDAS, A.; PAP, G.; SALLAI, J.; FRAMPTON, K. Sensor network-based countersniper system. In *Proceedings of the 2nd international conference on Embedded networked sensor systems* (2004), ACM, pp. 1–12.
- [16] SINOPOLI, B.; SHARP, C.; SCHENATO, L.; SCHAFFERT, S.; SASTRY, S. S. Distributed control applications within sensor networks. *Proceedings of the IEEE* 91, 8 (2003), 1235–1246.
- [17] SOMMER, P.; WATTENHOFER, R. Gradient clock synchronization in wireless sensor networks. In *International Conference on Information Processing in Sensor Networks* (April 2009), pp. 37–48.
- [18] SUNDARARAMAN, B.; BUY, U.; KSHEMKALYANI, A. D. Clock synchronization for wireless sensor networks: a survey. *Ad Hoc Networks* 3, 3 (2005), 281–323.
- [19] VAN GREUNEN, J.; RABAEY, J. Lightweight time synchronization for sensor networks. In *Proceedings of the 2Nd ACM International Conference on Wireless Sensor Networks and Applications* (2003), ACM, pp. 11–19.
- [20] WANG, S.; AHN, T. Mac layer timestamping approach for emerging wireless sensor platform and communication architecture, Dec. 17 2009. US Patent App. 12/213,286.
- [21] YANG, H.; SIKDAR, B. A protocol for tracking mobile targets using sensor networks. In *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on* (2003), IEEE, pp. 71–81.
- [22] ZANATTA, G.; BOTTARI, G. D.; GUERRA, R.; LEITE, J. C. B. Building a WSN infrastructure with COTS components for the thermal monitoring of datacenters. In *Symposium on Applied Computing, SAC 2014, Gyeongju, Republic of Korea - March 24 - 28, 2014* (2014), pp. 1443–1448.

APÊNDICE A - <TÍTULO DO APÊNDICE>

Elemento opcional. O(s) apêndice(s) são identificados por letras maiúsculas consecutivas, travessão e pelos respectivos títulos. Excepcionalmente utilizam-se letras maiúsculas dobradas, na identificação, quando esgotadas as 23 letras do alfabeto (ABNT, 2005).