MASTER OF HEALTH CARE DELIVERY SCIENCE

DARTMOUTH GRADUATE STUDIES

THAYER SCHOOL OF ENGINEERING AT DARTMOUTH

THE DARTMOUTH INSTITUTE FOR HEALTH POLICY & CLINICAL PRACTICE

DARTMOUTH GRADUATE APPLICATION

First Name (Given Name)	Middle Name	Last Name (Family Name)	Preferred First	Suffix
Kittipat		**Apicharttrisorn**		

Gender: **Male**	Full-time or Part-time: **Full-time**	Applying for Financial Aid: **Y**

Date of Birth: **SEP 2 1982** Birthplace: **Thailand** Country of Citizenship: **Thailand** Current Visa Type (non U.S. Citizens): **Non-Citizen**	Ethnicity (optional): Are you Hispanic or Latino: **N** ___ American Indian or Alaskan Native ✔ Asian (including Indian Subcontinent and Philippines) ___ Black or African American (including Africa and Caribbean) ___ Native Hawaiian or Other Pacific Islander (Original Peoples) ___ White (including Middle Eastern) If Native American, Tribal Affiliation: **Thailand**
Current Address [Expiration Date:__________] **7/639 Vibhavadee-Rangsit Rd. Chatuchak** **Bangkok 10900** **Thailand** Current Telephone Number: Home: **66025370097** Office: **66022859177** Fax:	Permanent Address: **7/639 Vibhavadee-Rangsit Rd. Chatuchak** **Bangkok 10900** **Thailand** Permanent Telephone Number: Home: Office: Fax:
Email Address: **kittipat.api@gmail.com**	(optional) Web page:

Graduate Program: **Computer Science** Intended Area of Specialization: **Systems**	Graduate Degree: **Doctor of Philosophy**
How did you hear about our program?	

If required and you have not taken the GRE or MCAT, when will you take the exam?

GRE Test Date 1	GRE Verbal	GRE Verbal %	GRE Quant	GRE Quant%	TOT Verbal & Quant	GRE Anal	GRE Anal%	GRE Subject Date 1	GRE Subject	GRE Subject%
10/2013	**153**	**53**	**164**	**89**		**4.0**	**54**			
GRE Test Date 2	GRE Verbal	GRE Verbal %	GRE Quant	GRE Quant%	TOT Verbal & Quant	GRE Anal	GRE Anal%	GRE Subject Date2	GRE Subject	GRE Subject%

TOEFL Test Date	Listening Comprehension	Reading Comprehension	Structure Writing	Speaking	Total Score
11/2013	**29**	**26**	**30**	**22**	**107**
IELTS Test Date	Listening	Reading	Writing	Speaking	Overall Band Score

MCAT Test Date	Verbal	Verbal %	Phy Sci	Phy Sci %	Essay	Essay %	Bio Sci	Bio Sci %	Total Score	Total %

GMAT Test Date	Verbal	Verbal %	Quantitative	Quantitative %	Total Score	Total %

GRADUATE DEGREE INSTITUTIONS:

Prior School 1: Chulalongkorn U. Attendance Dates: 05/2007 to 11/2010				Prior School 1 Location: Bangkok		
Major	Degree	Degree Date	GPA: 3.75 Major GPA:	Class Rank		Date Transcript Received
Computer Science	Master of Science	11/2010				

Prior School 2: Kasetsart University 06/2000 to 10/2004				Prior School 2 Location: Bangkok		
Major	Degree	Degree Date	GPA: 2.49 Major GPA:	Class Rank		Date Transcript Received
Electrical Engg.	Bachelor of Engg.	10/2004				

UNDERGRADUATE DEGREE INSTITUTIONS:

Prior School 3: Attendance Dates:				Prior School 3 Location:		
Major	Degree	Degree Date	GPA: Major GPA:	Class Rank		Date Transcript Received

Prior School 4: Attendance Dates:				Prior School 4 Location:		
Major	Degree	Degree Date	GPA: Major GPA:	Class Rank		Date Transcript Received

Prior School 5: Attendance Dates:				Prior School 5 Location:		
Major	Degree	Degree Date	GPA: Major GPA:	Class Rank		Date Transcript Received

Prior School 6: Attendance Dates:				Prior School 6 Location:		
Major	Degree	Degree Date	GPA: Major GPA:	Class Rank		Date Transcript Received

OTHER PROGRAM:

Prior School 7: Attendance Dates:				Prior School 7 Location:		
Major	Degree	Degree Date	GPA: Major GPA:	Class Rank		Date Transcript Received

Publications:

"A Moving Object Tracking Algorithm Using Support Vector Machines in Binary Sensor Networks"
"Desynchronization with an artificial force field for wireless networks"
"Energy-Efficient Gradient Time Synchronization for Wireless Sensor Networks"

Honors,Prizes, and Academic Distinctions:

Extra Curricular Activities such as internships and student teaching or participation in any activities that you consider of academic significance:

VOLUNTEER SERVICES
• Event Name: CANSO* Global ATM Summit and 15th Annual General Meeting (AGM)
Period 11 June 2011 - 14 June 2011
Contributions : Help organize meeting rooms

Recommended by: Name: **Chalermek Intanagonwiwat**	Title: **Senior Software Engineer**	Institution: **Cisco Systems, Inc.**	Date Letter Received:
Recommended by: Name: **Teerasit Kasetkasem**	Title: **Assistant Professor**	Institution: **Kasetsart University**	Date Letter Received:
Recommended by: Name: **Yunyong Teng-amnuay**	Title: **Head of Information Systems Engineering Lab**	Institution: **Chulalongkorn University**	Date Letter Received:

Are you the first in your immediate family to have enrolled in a bachelor's degree program? N

Would you be the first in your immediate family to attend post graduate education? N

Please rate the following factors for their importance in your choice of graduate school:
1 Not Applicable 2 Not Important 3 Somewhat Important 4 Moderately Important 5 Very Important

5 Very Important Institution Reputation
4 Moderately Impo Department Reputation
4 Moderately Impo Special Training Programs
5 Very Important Faculty Research Interests
3 Somewhat Impor Type of Courses Offered
3 Somewhat Impor Smaller Sized Graduate School
5 Very Important Guaranteed Funding
4 Moderately Impo Social Atmosphere
3 Somewhat Impor Geographic Location
3 Somewhat Impor Career Potential/Placement
2 Not Important Diverse Student Body
4 Moderately Impo Alumni Network
5 Very Important Career Goal: Academic Professor

4 Moderately Impo Career Goal: Work in Industry
3 Somewhat Impor Career Goal: Teaching
3 Somewhat Impor Gaining Breadth of Experiences
4 Moderately Impo Gaining a Depth of Knowledge
5 Very Important Faculty Mentor Identified
4 Moderately Impo Professional Development Training
4 Moderately Impo Training/Experience in Grant Writing
3 Somewhat Impor Teaching Experience
4 Moderately Impo Business Management Training
3 Somewhat Impor Ethical Integrity Emphasis
4 Moderately Impo Interdisciplinary Research Opportunities/Training
3 Somewhat Impor International Opportunities

How did you hear about our program initially? (Please check all that apply)

- ☐ Undergraduate/Masters Advisor
- ☐ Faculty at Undergraduate/Masters Institution
- ☐ Students at Undergraduate/Masters Institution
- ☐ Graduate School Fair
  Where?______________________
- ☐ Professional Society Meeting
- ☐ Internet advertisement
- ☐ Email Invitation
- ☐ Mail Invitation
- ☐ Dartmouth Alumni Recommendation
- ☐ Petersons.com
- ☐ Grad Schools.com
- ☐ Other Web Resource____
- ☐ Other (please specify)____

How did you get most of your information about our graduate program? (Please check all that apply):

- ☑ Departmental or Program Website
- ☐ Campus Interview
- ☐ Requested Brochures from the Department/Program
- ☐ Undergraduate/Masters Advisor
- ☐ Faculty at Undergraduate/Masters Institution
- ☐ Other (please specify)_______________________

Statement of Purpose

This statement of purpose is intended for use with my application to the Doctor of Philosophy graduate program at the Department of Computer Science, Dartmouth College. This document starts by portraying my educational background, of both the Bachelor and Master's degrees. Then, it briefly explains my research experience during the Master's degree study and states my professional experience during the employment as a systems engineer. Then my interest in Dartmouth's research is elaborated and finally my future plans after Ph.D. graduation are described. After finishing reading this statement of purpose, the committee will learn why I am qualified to be an excellent student of the program, what motivates me to pursue the doctoral degree at Dartmouth and why it is so important for my future profession that I earn this degree.

During my undergraduate study, in addition to a number of Electrical Engineering subjects, I studied a wide range of mathematical subjects including four Calculus courses, a course on Probability, and another on Linear Algebra and Complex Numbers, all of which are basic principles of today's computer science. Moreover, I passed two courses on computer programming, data structures and algorithms, which are a prerequisite knowledge for a successful computer scientist. For the Master's degree study, I passed eight credited computer science graduate courses. I studied two theoretical courses, namely Theory of Computation and Computer Algorithms, five systems courses namely Information Systems Architecture, Distributed Systems, Advanced Topics on Computer Networks (Multimedia, Wireless and Adhoc Networks), Embedded Systems, and Database Management Systems, and one Artificial Intelligence course. Moreover, I passed two non-credit courses - namely Computer Security and Special Topics on Distributed Systems (Service Computing). In sum, I earn a solid foundation in computer science as a result of my undergraduate and graduate study.

In addition, I gain valuable research experience during my Master's degree study and I would like to explain three principal research skills in this letter. First, I learn the critical reading skills. As an important part of research methodology in computer science, literature reviewing is an everyday activity of graduate students. Researchers study research papers not only to understand the overall concepts but also to critique them, find weak points and discover hidden assumptions. With this critical reading, I can find a research opportunity hidden in a research paper and can think of "what to do next" instead of just "this work is interesting". Second, I learn how to give an intelligible academic presentation. At the UbiNet lab under the supervision of Assistant Professor Dr. Chalermek Intanagonwiwat, each lab member took turns giving one progress presentation reporting the progress toward the thesis work and one paper presentation illustrating the ideas and results of a research paper of interest. Through this regular lab activity, I learned to select an interesting paper published in a well-known conference or journal publisher, to extract outstanding points in the paper and to present them in a way that made it easier for the audience to understand. Third, I learn scientific publication skills. After completing a certain amount of literature review and implementation work, I need to publish a paper in order to organize my ideas into a standard format, to distribute my work for other researchers to study and to welcome feedbacks and comments from reviewers, all of which help further strengthen my work. According to my advisor, a high-quality paper in computer science should not only allow the readers to understand the overall picture of the work, but also enable them to implement it into the code themselves. Therefore, I learn to explain the data structures, algorithms, and communication packets so clearly that one could use all this information for further experimentation. In sum, I earn the research experience and skills not through lectures or workshops but by application and repetition throughout the years of the Master's degree study.

Up to now, I have published three academic publications, two of which are in international conferences' proceedings and the other is in an ACM journal. First, "Energy-Efficient Gradient Time Synchronization for Wireless Sensor Networks" was published in the proceedings of the Second International Conference on Computational Intelligence, Communication Systems and Networks or CICSyN. In the paper, I designed an extended version of gradient time synchronization protocols that was more time-accurate and energy-efficient, while maintaining a "gradient" property. With the gradient property, geographically adjacent nodes are able to maintain minimal synchronization errors. Second, "Desynchronization with an artificial force field for wireless networks" was published in ACM SIGCOMM's *Computer Communication Review*. The desynchronization problem was analogous to a resource allocation problem in which nodes cooperated to take turns accessing to the same resource. In this paper, we provided a prove of convexity of this problem. Additionally, we designed a desynchronization protocol, inspired by electromagnetic force field, that performed in a distributed manner, better scaled with network sizes and densities and produced less desynchronization errors. The first two papers were my work under the supervision of Assistant Professor Dr. Chalermek Intanagonwiwat. Third, in 2013, I had a change to work on a research project with Associate Professor Dr. Teerasit Kasetkasem of Kasetsart University. In this project, we used a signal processing technique to track a moving object in a field given binary sensor observations. In this paper, I was fully responsible for the manuscript preparation and partly for experimental simulation. Finally, the paper titled "A Moving Object Tracking Algorithm Using Support Vector Machines in Binary Sensor Networks", was finally published in the proceedings of The 13th International Symposium on Communications and Information Technologies.

I also have seven-year professional experience working at Aeronautical Radio of Thailand or Aerothai, a state enterprise under the Ministry of Transport, Thailand. One of Aerothai's principal missions is to provide safe and efficient air navigation services or air traffic control within Thailand's airspace. Specifically, the department of air traffic data systems engineering is responsible for the provision and administration of data systems that support air traffic controllers' operations. At the department, my colleagues and I design, configure, and implement those systems by taking advantage of enterprise-graded computing system products, mostly of the USA, such as HP and Dell servers, Oracle and Microsoft databases, Cisco network equipment, and VMWare's virtualization technology, etc. One of the interesting aeronautical applications that runs on these infrastructures is the flight scheduling service, named Bay of Bengal Cooperative Air Traffic Flow Management System or BOBCAT. BOBCAT manages the air traffic over the Bay of Bengal, which has the security constraints. Approximately 60 flights per day request to fly through this narrow airspace; therefore, International Civil Aviation Organization or ICAO demands that the airspace be managed by Aerothai, after the systems competition with other organizations. Nowadays, BOBCAT smoothly serves tens of airline customers requesting air space slots over the area every day thanks to Aerothai's effective software systems and responsive operational procedures. Therefore, I have witnessed how these innovative products help enhance reliability and efficiency of air traffic data systems. This hand-on experience has provided me with practical aspects of enterprise information systems with the safety-critical applications, and motivates me to study more deeply and broadly in computer science, a core foundation of computer-related products and services.

I determine to advance my study to a PhD level in the US because of the following three main reasons. First and most importantly, I want to be a professional researcher in computer science in the future, either in an academic institution or in a research laboratory and a doctoral degree is an important precursor to the research profession. Second, I agree with Matt Welsh, previously a professor of Computer Science at Harvard University, about a PhD study. He suggests that "You get an intense exposure to every subfield of Computer Science, and have to become the leading world's expert in the area of your dissertation work." For example, during my PhD study, I will have an opportunity to get exposed to a variety of academic subjects and research projects in computer science, such as Artificial Intelligence, Computer Graphics, Robotics, Databases, Systems, Software Engineering, and Computational Science, etc., all of which will considerably expand my intellectual horizons in computer science. Moreover, the PhD study will train me to be an expert in the field of my dissertation through the educational systems and processes, and through my assiduous and persevering efforts. Third, I am conscious that studying at a PhD level requires an academically vibrant environment which includes surroundings with brilliant students and inspirational faculty members, as well as accessible academic conferences and seminars. In my opinion, all of these are prevalent in the US educational systems and universities.

I aspire to become a PhD student at the Department of Computer Science, Dartmouth College, a prestigious university in the US, because I am particularly interested in its research. The following are Dartmouth's faculty members whose research projects interest and excite me. First, Professor David Kotz's mHealth project takes advantage of ubiquitous mobile devices and pervasive cellular networks in order to promote public health and wellness of people around the world. For example, a doctor can remotely monitor her client which has a smart phone connecting medical sensors to the Internet. As another example, a patient's mobile phone can help him collect his personal health data such as pictures of his diet or social network traces during the last week or month so that his doctor can use them for more appropriate care and better analysis. Although these data are beneficial for patients themselves and valuable for hospitals and insurance companies, security and privacy of such data are of particular concerns. Without a level of security and privacy guaranteed, no clients or patients are willing to share such data and no hospitals or insurance companies are likely to use them. For example, a patient is reluctant to share the data that cost him a higher rate of health insurance plans. However, such data must be reliable without manipulation or alteration; or else, a hospital or health insurance company may refuse to use them. As a result, security and privacy of mHealth data are among key factors of successfully deployed mHealth systems in the future. Professor David Kotz's publications "Provenance framework for mHealth" and "Privacy in Mobile Technology for Personal Healthcare" propose to address mHealth security and privacy issues respectively. However, these frameworks should be further investigated by implementing them in an experimental setting so that researchers could see a clearer picture of how the systems would work in a real world situation.

Second, Professor Andrew T. Campbell's CarSafe App project interestingly uses a smart phone device to imitate the features of high-end car safety devices that aim to alert drowsy or distracted drivers. Drowsiness and distraction are among the main factors of car accidents not only in the US but also in Thailand. I hear every year about car accidents in Thailand that cause public bus passengers to die because of the driver's drowsiness. This mobile app installed on smart phones is a potential and affordable solution to this problem. It manages not only to alert the bus or car driver, but also, possibly via the cellular network, to alarm the bus service company which then may trigger an appropriate measure to its driver's drowsiness problem. In sum, this mobile application can help reduce the number and severity of car accidents in Thailand that cause thousands of Thai people to die every year by using the CarSafe App on a smart phone in any car model.

Third, Assistant Professor Xia Zhou's publication "Practical Conflict Graphs for Dynamic Spectrum Distribution" is of my particular research interest because it applies graph theory to resource allocation problems. One of my research projects at Chulalongkorn University was "Desynchronization as Distributed Resource Allocation" in which we model a framework for desynchronization and develop desynchronization algorithms to solve TDMA problems in wireless networks. Therefore, these two projects seem to solve almost the same problems from different perspectives and with different targeted applications (Dynamic Spectrum Distribution and TDMA). Moreover, in the forth year of my undergrad education, I studied Wireless Communications which covered topics of cellular networks and wireless signal models which were the fundamentals of the signal interference modeling. In sum, my research experience and educational background could be a useful resource for this project of Assistant Professor Xia Zhou.

My plan after graduation with a doctoral degree is that I will look for a research or post-doc position that is related to the field of my dissertation in order to continue to accumulate research knowledge and experience. Therefore, within five years after graduation, I will become a real expert in the field and plan to lead my own research laboratory. Research experience gained during the PhD study and accumulated after graduation will play an important role in attracting funds and students into my lab.

I would like to express my appreciation to the graduate admission committee of Dartmouth College for taking my statement of purpose along with other application materials into consideration for admissions. I hope that the committee will be convinced that my educational background, academic and professional experience, and research interest and motivation are the sufficient evidences to suggest that I will be an excellent student of the PhD program and a competent researcher in computer science.

KITTIPAT APICHARTTRISORN

Office Address
Air Traffic Data Systems Engineering Department
Aeronautical Radio of Thailand
Sathon, Bangkok, Thailand 10120
(+66) 2285-9177

Permanent Address
7/639 Vibhavadee-Rangsit Rd.
Chatuchak, Bangkok 10900
(+66) 2537-0097

OBJECTIVE

A Ph.D. student position in computer science with research interest in computer networks, distributed resource allocation, sensor networks, software-defined networking, and internet of things.

EDUCATION

Master of Science, Computer Science
Chulalongkorn University, Bangkok, Thailand GPA 3.75 / 4.00 May 2007 - November 2010
THESIS - Distributed Time Synchronization in Wireless Sensor Networks
ADVISOR - Asst. Prof. Dr. Chalermek Intanagonwiwat

Bachelor of Engineering, Electrical Engineering
Kasetsart University, Bangkok, Thailand GPA 2.49 / 4.00 June 2000 - October 2004
SENIOR PROJECT: Adaptive Multi-Rate - Wideband (AMR-WB) speech codec Testing
SENIOR PROJECT SUPERVISOR: Assoc. Prof. Dr. Mongkol Raksapatcharawong

EMPLOYMENT

Senior Systems Engineer January 2007 - Present
Air Traffic Data Systems Engineering Department
Aeronautical Radio of Thailand, Bangkok, Thailand

- Administer, monitor, and maintain aeronautical data systems for which the Air Traffic Data Systems Engineering Department take responsibility so that the systems operate to support availability, safety and continuity of air navigation services
- Perform preventive maintenance, corrective maintenance, software and hardware installation, and deployment of monitoring systems (e.g. ICMP, SNMP)
- Inspect and troubleshoot problems, coordinate and consult with related internal and external aeronautical units to troubleshoot problems and investigate causes of interruption or outage of data systems services
- Gather information from users and report usage and service problems to managers, programmers and the director, to improve systems' reliability, availability and serviceability

Network Engineer March 2005 - September 2006
1tonet Co., Ltd., Bangkok, Thailand

- Design and implement voice over IP subsystems
- Integrate IP telephony with customers' existing public exchange systems

PUBLICATIONS

- **"A Moving Object Tracking Algorithm Using Support Vector Machines in Binary Sensor Networks"**

  Authors Dusadee Apicharttrisorn, Kittipat Apicharttrisorn and Teerasit Kasetkasem
  Publication Name The 13th International Symposium on Communications and Information Technologies
  Publication Date September 2013
  Abstract *Wireless sensor technologies have enabled us to deploy such small sensors to monitor an area of interest. Object tracking is one of the most attractive applications to be implemented with wireless sensor networks (WSNs). However, many solutions are struggled with energy-draining global positioning system (GPS), poorly-performed trilateration for indoor usage, and impractical, complex algorithms to be implemented in sensor nodes. This paper proposes a moving object tracking algorithm using support vector machines (MOT-SVM). The MOT-SVM takes advantage of light-weighted directional binary sensor networks, and state-of-the-art signal processing algorithms, namely the support vector machines and particle filters. We compare our proposed algorithm with the Aslam's work through the simulation. We examine our algorithms for various movement scenarios such as the linear, random and the 8-model trajectories, and the scenarios in which observing sensors make observation errors.*

- **"Desynchronization with an artificial force field for wireless networks"**

  Authors Supasate Choochaisri, Kittipat Apicharttrisorn, Kittiporn Korprasertthaworn, Pongpakdi Taechalertpaisarn and Chalermek Intanagonwiwat
  Publication Name SIGCOMM Computer Communication Review
  Publication Date March 2012
  Abstract *Desynchronization is useful for scheduling nodes to perform tasks at different time. This property is desirable for resource sharing, TDMA scheduling, and collision avoiding. Inspired by robotic circular formation, we propose DWARF (Desynchronization With an ARtificial Force field), a novel technique for desynchronization in wireless networks. Each neighboring node has artificial forces to repel other nodes to perform tasks at different time phases. Nodes with closer time phases have stronger forces to repel each other in the time domain. Each node adjusts its time phase proportionally to its received forces. Once the received forces are balanced, nodes are desynchronized. We evaluate our implementation of DWARF on TOSSIM, a simulator for wireless sensor networks. The simulation results indicate that DWARF incurs significantly lower desynchronization error and scales much better than existing approaches.*

- **"Energy-Efficient Gradient Time Synchronization for Wireless Sensor Networks"**

  Authors Kittipat Apicharttrisorn, Supasate Choochaisri and Chalermek Intanagonwiwat
  Publication Name 2010 Second International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)
  Publication Date July 2010
  Abstract *Wireless sensor network (WSN) applications usually demand a time-synchronization protocol for node coordination and data interpretation. In this paper, we propose an Energy-Efficient Gradient Time Synchronization Protocol (EGTSP) for Wireless Sensor Networks. In contrast to FTSP, a state-of-the-art synchronization protocol for WSNs, EGTSP is a completely localized algorithm that achieves a global time consensus and gradient time property using effective drift compensation and incremental averaging estimation. In contrast with GTSP, a gradient-based fixed-rated time synchronization protocol, our protocol provides adaptive beaconing for applications to optimize energy savings by selecting appropriate message-broadcast periods. The protocol is implemented and evaluated on multi-hop networks that consist of Telosb motes running TinyOS. The experimental results indicate that our protocol achieves a network-wide global notion of time, attains small synchronization errors, and utilizes energy efficiently.*

ACADEMIC PROJECTS

- Project Name: Time Synchronization for Wireless Sensor Networks

  Objective MS Thesis's Research Project
  Description Time synchronization is a challenging but important task for wireless sensor networks (WSNs) because of the resource-constrained characteristics. This project aims to explore a distributed protocol and algorithm of time synchronization that is time-accurate and energy-efficient while maintaining a gradient time property.
  Period January 2008 - October 2010
  Roles and Responsibility Main investigator who reviews literature, designs, analyzes, and implements algorithms, finally produces a publication
  Tools and Environments TinyOS, Ubuntu, Gnuplot, TelosB* motes

- Project Name: Desynchronization as Distributed Resource Allocations and TDMA

  Objective Research Project
  Description Desynchronization is an abstraction that arranges nodes declaring to access a shared resource in a round-robin schedule. It can be applied to solve resource allocation problems especially in distributed systems. This research project aims to explore a novel distributed desynchronization algorithm.
  Period March 2010 - Present
  Roles and Responsibility Literature review, experiments, and publications
  Tools and Environments TinyOS, TOSSIM, Ubuntu, Gnuplot

- Project Name: Moving Object Tracking in Binary Sensor Networks

  Objective Research Project
  Description Moving object tracking is a potential application of wireless sensor networks. Binary sensor networks require nodes only to send one-bit information to the central processing node which is responsible for signal processing tasks to track a moving object. This research project aims to explore a signal processing algorithm that tracks the object more accurately with tolerance to signal errors.
  Period March 2013 - Present
  Roles and Responsibility Literature review, experiments, and publications
  Tools and Environments Matlab

- Project Name: Distributed Online Ticket Reservation with Display on Google Maps

  Objective Term Project (Graduate Course: Distributed Systems)
  Description This project aims to provide an opportunity for students to design and implement a distributed system which reserves online tickets and displays the status through Google Maps.
  Period June 2008 - October 2008
  Roles and Responsibility Design overall systems and demonstration
  Tools and Environments Microsoft .NET and Google Map APIs

- Project Name: Thailand's Undergrad Admission Systems: Information Systems Architecture

  Objective Term Project (Graduate Course: Information Systems Architecture)
  Description This project aims to provide an opportunity for students to design Thailand's Undergrad Admission Systems. During this term project, we combine each other's experience and viewpoints of information systems and brainstorm the viable solutions for the systems. The final document consists of the design of network, database, hardware, middleware, and software. The designed architecture is supposed to support thousands of concurrent users who use the system from registrations to final admission reports.
  Period June 2007 - October 2007
  Roles and Responsibility Part of group discussion and brainstroming sessions
  Tools and Environments MS Words, MS Visio

- Project Name: Adaptive Multi-Rate - Wideband (AMR-WB) speech codec Testing

  Objective Undergraduate Senior Project (Electrical Engineering Project)
  Description Adaptive Multi-Rate Wideband (AMR-WB) is a patented wideband speech coding standard developed based on Adaptive Multi-Rate encoding, using similar methodology as Algebraic Code Excited Linear Prediction (ACELP). AMR-WB provides improved speech quality due to a wider speech bandwidth of 50 - 7000 Hz compared to narrowband speech coders which in general are optimized for POTS wireline quality of 300 - 3400 Hz. This project aims to document the study of AMR-WB in both theoretical and practical aspects.
  Period June 2003 - Mar 2004
  Roles and Responsibility Design and conduct experiments, and document a project report
  Tools and Environments MS Visual C

* TelosB is a WSN platform that is widely used by research laboratories worldwide.

PROFESSIONAL PROJECTS

- Project Name: Aeronautical Message Switching Systems (AMSS)

  Description AMSS is a core aeronautical data system that switches, stores and manipulates aeronautical messages interexchanged between aeronautical units worldwide so that flights are operated and managed properly and continuously.
  Roles and Responsibilities Administer, monitor, and maintain the system, inspect and troubleshoot problems
  Tools and Environments Redhat Enterprise, Windows Servers, Oracle Database 10g, Cisco switches and routers

- Project Name: Aeronautical Message Handling Systems (AMHS) and X.400

  Description According to ICAO*, Aeronautical Message Handling System is a new standard for aeronautical ground-ground communications (e.g. for the transmission of NOTAM**, Flight Plans or Meteorological Data) based on X.400 profiles. Aeronautical Radio of Thailand progresses to establish AMHS connectivity with several countries such as India, Singapore, Hong Kong, Italy, Laos, Vietnam, and Cambodia.
  Roles and Responsibilities Test and record system connectivity and functionality
  Tools and Environments Redhat Enterprise, Oracle Database 10g, ATN Routers

- Project Name: Flight Data Management Center

  Description Flight Data Management Center was established to unify clearance of national flight plans and their modifications to a single center in order to streamline air navigation operations. Computer-based systems are used to provide the functionality of FDMC.
  Roles and Responsibilities Administer, monitor, and maintain the system, inspect and troubleshoot problems
  Tools and Environments Java, Redhat Enterprise, MS Windows Servers, Oracle Database, Cisco switches and routers

- Project Name: Operational Aeronautical Meteorological Data (OPMET) and Regional OPMET Bulletins Exchange (ROBEX) Systems

  Description Aeronautical Radio of Thailand was designated to provide a regional OPMET data bank of the Asia/Pacific region. Its core function is to accumulate and store aeronautical meteorological data that can be retrieved remotely and automatically by queries from relevant aeronautical organizations. ROBEX processes such data in the form of bulletins, a periodic conclusive report, and periodically send them to related aeronautical units.
  Roles and Responsibilities Administer, monitor, and maintain the systems, inspect and troubleshoot problems

Tools and Environments Java, Redhat Enterprise, MS Windows Servers, Oracle Database, Cisco switches and routers

* ICAO (International Civil Aviation Organization) is a specialized agency of the United Nations which codifies the principles and techniques of international air navigation and fosters the planning and development of international air transport to ensure safe and orderly growth. Its headquarters are located in the Quartier International of Montreal, Quebec, Canada.

** NOTAM (Notice to Airmen) is a notice filed with an aviation authority to alert aircraft pilots of potential hazards along a flight route or at a location that could affect the safety of the flight. Aeronautical Radio of Thailand is authorized to provide a NOTAM data bank that stores and retrieves NOTAM messages which are distributed by AMSS and AMHS.

GRANTS

- Grant Name: International Conference Attendance Support Grants for Graduate Students

  Period July 2010
  Purpose This grant provides partial financial support for graduate students whose academic papers are accepted to be presented at an international conference.
  Amount Approximately 900 USD
  Granted by Graduate School, Chulalongkorn University Bangkok, Thailand

- Grant Name: AINTEC* 2010 Conference Attendance Grants

  Period November 2010
  Purpose This grant provides full financial support for graduate students who are interested in Internet research so that they can attend and participate in this academic conference.
  Amount Attendance Fee (Unknown)
  Granted by Thailand Research Education Network Association (ThaiREN), Bangkok, Thailand

- Grant Name: AINTEC* 2008 Conference Attendance Grants

  Period November 2008
  Purpose This grant provides full financial support for graduate students who are interested in Internet research so that they can attend and participate in this academic conference.
  Amount Attendance Fee (Unknown)
  Granted by Thailand Research Education Network Association (ThaiREN), Bangkok, Thailand

* AINTEC (Asian INTernet Engineering Conference) is an international conference held in Thailand and hosted by Internet Education and Research Laboratory, Asian Institute of Technology, Thailand `http://www.interlab.ait.ac.th/`. This single-tracked conference attracts high-quality papers from global Internet research communities.

ACADEMIC ACTIVITIES

- **Event** IEEE International Conference on Computer Communications (INFOCOM 2012)
  Activity Review papers delegated by Asst. Prof. Dr. Chalermek Intanagonwiwat
- **Event** IEEE International Conference on Computer Communications (INFOCOM 2011)
  Activity Review papers delegated by Asst. Prof. Dr. Chalermek Intanagonwiwat

CERTIFICATES

- Certificate Name: *"Embedded Software Engineering"*

  Content Embedded Hardware Architecture, Operating Systems for Embedded Systems, Programming Embedded Systems, Embedded Systems I/O, Embedded Software Engineering

  Certified by Computer Engineering Department, Chulalongkorn University and Software Industry Promotion Agency (SIPA)

  Duration 22 - 27 October 2007

- Certificate Name: *"Certified Thaicom Users"*

  Content General functionality of THAICOM satellites, Basic VSAT setup, Signal optimization and interference

  Certified by THAICOM Public Company Limited

  Duration 3 April 2007

- Certificate Name: *"Network Design and Implementation I"*

  Content Design, analysis, implementation and troubleshooting of computer networks and hands-on workshops with CISCO routers and switches

  Certified by Continuing Education Center, Chulalongkorn University

  Duration 29 January 2005 - 23 April 2005

SKILLS

Programming Languages
- C, C++, NesC, TinyOS, Matlab, Java, Python, SQL

Computer Software
- Ubuntu, UNIX, Gnuplot, Latex.

Language Proficiency
- English: TOEFL 107 iBT (Test Date: 24 November 2013)
  Reading: 26 / 30, Listening: 29 / 30, Speaking: 22 / 30, Writing: 30 / 30
- Thai: Native

Graduate Record Examination
- Test Date: 21 October 2013
- Verbal Reasoning Score: 152 / 170 (53^{rd} *Percentile Rank*)
- Quantitative Reasoning Score: 164 / 170 (89^{th} *Percentile Rank*)
- Analytical Writing Score: 4.0 / 6.0 (54^{th} *Percentile Rank*)

VOLUNTEER SERVICES

- Event Name: CANSO* Global ATM Summit and 15th Annual General Meeting (AGM)

  Period 11 June 2011 - 14 June 2011
  Description : As Air Chief Marshal Somchai Thean-anant, a former President of Aeronautical Radio of Thailand delivered a policy to recruit the company's employees to volunteer to help organize these eminent events that welcomed hundreds of worldwide dignitaries and executives from all segments of the aviation industry. I applied for a volunteer position and was then selected, under the supervision of Ms. Tipaporn Nippakakorn, Vice President (Human Resource), to help organize the conference and seminar rooms at the Renaissance Hotel, Bangkok.
  Contributions : Help organize meeting rooms
  Benefits : Overall, the company succeeded in organizing these meetings which brought about and strengthen collaboration and understanding between global aeronautical organizations. My personal benefits included friendship with other employees from various departments of the company and awareness of aviation industry's next generation gathered during the conference and seminar attendance. Most importantly, I learn to volunteer myself to contributing back to my organization and aviation society without any pay.

  *CANSO the Civil Air Navigation Services Organization is the global voice of air navigation service providers (ANSPs) worldwide. CANSO's members support over 85 percent of world air traffic and share information and develop new policies, with the ultimate aim of improving air navigation services (ANS) on the ground and in the air.

INTERESTS AND HOBBIES

Jazz and blues guitar, photography, cooking, swimming

REFERENCES

- Ms. Tipaporn Nippakakorn

  Position Vice President (Human Resource)
  Address Aeronautical Radio of Thailand, Bangkok, Thailand, 10120
  Email tipaporn.ni@aerothai.co.th
  Tel. (+66) 2285-9179

- Dr. Chalermek Intanagonwiwat

  Position Senior Software Engineer
  Address Cisco Systems, Inc., California, USA
  Email cintanag@cisco.com
  Tel. (+1) 408 525 3795

- Mr. Pongnarin Anantasirijinda

  Position Director of Air Traffic Data Systems Engineering Department
  Address Aeronautical Radio of Thailand, Bangkok, Thailand, 10120
  Email add@aerothai.co.th
  Tel. (+66) 2285-9101

- Asst. Prof. Dr. Teerasit Kasetkasem

  Position Assistant Professor of Electrical Engineering
  Address Electrical Engineering Department, Kasetsart University, Bangkok, Thailand, 10900
  Email fengtsk@ku.ac.th
  Tel. (+66) 2942-8555 ext 1536

Sample Code

The following are pieces of code written in nesC, an extension of C that is designed to be used with embedded systems especially TinyOS, an operating system for wireless sensor networks. They are core code of my time synchronization that resulted in the publication "Energy-Efficient Gradient Time Synchronization for Wireless Sensor Networks" and my Master's degree thesis.

```
#include "TimeSyncMsg.h"

configuration TimeSyncC
{
  uses interface Boot;
  provides interface Init;
  provides interface StdControl;
  provides interface GlobalTime<TMilli>;
  provides interface TimeSyncInfo;
}

implementation
{
  components new TimeSyncP(TMilli);

  GlobalTime      =   TimeSyncP;
  StdControl      =   TimeSyncP;
  Init            =   TimeSyncP;
  Boot            =   TimeSyncP;
  TimeSyncInfo    =   TimeSyncP;

#ifdef TIMESYNC_TOSSIM
  components ActiveMessageC;
  TimeSyncP.RadioControl   ->  ActiveMessageC;
  TimeSyncP.AMSend         ->  ActiveMessageC.AMSend[AM_TIMESYNCMSG];
  TimeSyncP.Receive        ->  ActiveMessageC.Receive[AM_TIMESYNCMSG];
#else
  components TimeSyncMessageC as ActiveMessageC;
  TimeSyncP.RadioControl      ->  ActiveMessageC;
  TimeSyncP.AMSend            ->  ActiveMessageC.TimeSyncAMSendMilli[AM_TIMESYNCMSG];
  TimeSyncP.Receive           ->  ActiveMessageC.Receive[AM_TIMESYNCMSG];
  TimeSyncP.TimeSyncPacket    ->  ActiveMessageC;
#endif
  components HilTimerMilliC;
  TimeSyncP.LocalTime         ->  HilTimerMilliC;

  components new TimerMilliC() as TimerC;
  TimeSyncP.SendTimer ->  TimerC.Timer;

  components new TimerMilliC() as DriftCaptureDelay;
  TimeSyncP.DriftCaptureDelay ->  DriftCaptureDelay;

  components new TimerMilliC() as DriftCapturePeriod;
  TimeSyncP.DriftCapturePeriod ->  DriftCapturePeriod;

  components new TimerMilliC() as CompensateUnitDriftPos;
  TimeSyncP.CompensateUnitDriftPos ->  CompensateUnitDriftPos;

  components new TimerMilliC() as CompensateUnitDriftNeg;
  TimeSyncP.CompensateUnitDriftNeg ->  CompensateUnitDriftNeg;

  components LedsC;
  TimeSyncP.Leds  ->  LedsC;
}
```

The above piece of code shows the wiring and modular style of nesC. If you want to use a code module, you need to wire it to your code. My code mainly wires Active Message interfaces that make it easy to implement message transmission and reception over a wireless network, and timer interfaces that supports multiple fine-grained milli-second clocks.

```
event message_t* Receive.receive(message_t* msg, void* payload, uint8_t len)
{
  #ifdef TIMESYNC_DEBUG   // this code can be used to simulate multiple hopsf
          uint8_t incomingID = (uint8_t)((TimeSyncMsg*)payload)->nodeID;
          int8_t diff = (incomingID & 0x0F) - (TOS_NODE_ID & 0x0F);
          if( diff < -1 || diff > 1 )
              return msg;
          diff = (incomingID & 0xF0) - (TOS_NODE_ID & 0xF0);
          if( diff < -16 || diff > 16 )
              return msg;
  #endif

  //not currently processing and TimeSyncPacket is valid
  if  ( (state & STATE_PROCESSING) == 0
    #ifndef TIMESYNC_TOSSIM
          && call TimeSyncPacket.isValid(msg)
    #endif
    )
  {
      uint32_t localTime, globalTime;
            message_t* old = processedMsg;
              // old <-- processedMsg <-- msg
            processedMsg = msg;

  #ifdef TIMESYNC_TOSSIM
      globalTime = localTime = call GlobalTime.getLocalTime();
  #else
      globalTime = localTime = call TimeSyncPacket.eventTime(msg);   // MAC timestamping the packet and convert the event time to local of receiver
  #endif

    call GlobalTime.local2Global(&globalTime);
  ((TimeSyncMsg*)(payload))->receiveEventLocalTime = localTime;
    ((TimeSyncMsg*)(payload))->receiveEventGlobalTime = globalTime;
  printf("TimeSync: \t\nnodeID = %u \t\nglobalTime = %lu\n", TOS_NODE_ID, globalTime);

     state |= STATE_PROCESSING;
          post processMsg();
     return old;
     }
     return msg;
}
```


The above piece of code, processing upon reception of a new time sync packet, shows how I design the code implementation so that it can be used with different scenarios and settings. For example, if the makefile has been marked TIMESYNC DEBUG, this code will simulate multi hop wireless networks using node identification numbers as a controller. If the makefile has been marked TIMESYNC TOSSIM, a clock module for a TinyOS simulation called TOSSIM, which differs from that of sensor nodes or motes, will be used. Moreover, *post processMsg();* shows my implementation of nesC tasks. Tasks in TinyOS are a form of deferred procedure call (DPC), which enable a program to defer a computation or operation until a later time. TinyOS tasks run to completion and do not pre-empt one another. These two constraints mean that code called from tasks runs synchonously with respect to other tasks.

2010 Second International Conference on Computational Intelligence, Communication Systems and Networks

Energy-Efficient Gradient Time Synchronization for Wireless Sensor Networks

Kittipat Apicharttrisorn, Supasate Choochaisri, Chalermek Intanagonwiwat
Department of Computer Engineering, Faculty of Engineering
Chulalongkorn University
Bangkok, Thailand
kittipat.ap@student.chula.ac.th, supasate.c@student.chula.ac.th, chalermek.i@chula.ac.th

***Abstract*—Wireless sensor network (WSN) applications usually demand a time-synchronization protocol for node coordination and data interpretation. In this paper, we propose an Energy-Efficient Gradient Time Synchronization Protocol (EGTSP) for Wireless Sensor Networks. In contrast to FTSP, a state-of-the-art synchronization protocol for WSNs, EGTSP is a completely localized algorithm that achieves a global time consensus and gradient time property using effective drift compensation and incremental averaging estimation. In contrast with GTSP, a gradient-based fixed-rated time synchronization protocol, our protocol provides adaptive beaconing for applications to optimize energy savings by selecting appropriate message-broadcast periods. The protocol is implemented and evaluated on multi-hop networks that consist of Telosb motes running TinyOS. The experimental results indicate that our protocol achieves a network-wide global notion of time, attains small synchronization errors, and utilizes energy efficiently.**

***Keywords*-time synchronization; wireless sensor networks; energy efficiency**

I. Introduction

Wireless sensor networks (WSNs) have been extensively explored by research communities worldwide for a decade. Small-sized, resource-constraint, and battery-powered sensor nodes are wirelessly connected to each other to disseminate sensed data to sinks. Given that energy is one of the most precious resources in WSNs, it is mandatory that protocols in WSNs are energy-efficient. WSNs are unprecedentedly capable of monitoring environments which are sensitive to human access, infrastructure which is hard to maintain, and patients who are annoyed by electrical wires. Sensor network applications usually require that time and space information of an event be embedded into sensed data so that the data are more meaningful and usable. For all sensor nodes in the network to have a common notion of time, time synchronization is needed.

However, time synchronization is classified into three models proposed by Ganeriwal et al. [1]: event ordering, relative clock and global clock (or always-on). An event-ordering model is the simplest model whereby nodes in the system maintain only a right order of events. An example of this model is a protocol by Romer[2]. In a relative clock model, a node in sensor networks only maintains its clock relations between itself and neighbors. Example protocols of this model are RBS[3], and Tiny-Sync[4]. A global clock model (the most complex but most useful one) requires that all nodes maintain a common notion of time (the global time). With synchronized global time , nodes can easily have a right order of events, and a right relation of time with other nodes. Additionally, nodes can perform TDMA or schedule their duty cycle with minimal wake-up guard time.

Global time synchronization is a classical problem in distributed systems [5]. Network Time Protocol (NTP) [6] on Internet has provided global time synchronization for decades but, because of its high complexity and low accuracy in the millisecond scale, NTP is not well suited for use in WSNs. A state-of-the-art global time-synchronization protocol for WSNs is Flooding Time Synchronization Protocol (FTSP) [7].

However, FTSP lacks a gradient time property or locality. Most WSN applications require that nearer nodes have tighter time synchronization than farther nodes do. For example, most object tracking applications require that nodes adjacent to each other have tighter time synchronization to correctly detect the direction and speed of an object. Conversely, nodes with multi-hop away can tolerate larger synchronization error.

Unlike FTSP, Gradient Time Synchronization Protocol (GTSP) [8] is a distributed time synchronization protocol with the gradient time property. In GTSP, each node estimates its perceived global (logical) time with the average perceived global time of all neighbors. Every node periodically broadcasts a reference-point message containing its global time. The received reference points from all neighbors are used to calculate an average time offset and relative global rate. According to this computed time information, the global clock of the computing node is adjusted.

In addition, the global clock of each node is adjusted once every broadcasting period. Consequently, the amount of adjustment can be large. Therefore, the global time will not be smoothed (continuous). This may cause an error in data interpretation of an application. Furthermore, in GTSP, the broadcasting period is fixed. To achieve small synchronization error, this fixed broadcasting period of GTSP must also be small. This will linearly and significantly drain sensor networks' energy.

In this paper, we propose an Energy-Efficient Gradient Time Synchronization Protocol (EGTSP) for wireless sensor

978-0-7695-4158-7/10 $26.00 © 2010 IEEE
DOI 10.1109/CICSyN.2010.14

networks. Similar to GTSP, EGTSP is a localized algorithm to achieve the global time consensus and the gradient time property. However, unlike GTSP, EGTSP achieves those properties using effective drift compensation and incremental averaging estimation without the mentioned drawbacks of GTSP.

To improve the continuity of the global time, each EGTSP node computes the incremental average of time immediately after receiving a broadcast message from a neighbor (instead of computing the average after broadcast messages from all neighbors are received). Given that this incremental adjustment is smaller, the continuity of the global time is improved.

Additionally, after the drift estimation, our algorithm can adaptively adjust the broadcasting period to save energy without degrading the global time accuracy. Applications are also allowed to select an extensive broadcasting period by trading off metrics such as synchronization error, system responsiveness, and energy consumption.

We have also implemented and evaluated our protocol on a testbed consisting of Telosb motes, each of which runs TinyOS. EGTSP is compared with GTSP to verify the viability of our approach. The experimental results indicate that EGTSP consumes 82% less energy as well as incurs slightly smaller synchronization error over GTSP does.

II. Related Work

Time synchronization techniques can be simply divided into two categories: receiver-to-receiver and sender-to-receiver. In a receiver-to-receiver protocol, one node broadcasts a beacon message to let all nodes in the broadcast domain timestamp upon reception. The receivers of the broadcast later exchange timestamp information to each other and calculate time offset relative to each other. Using this technique, the effect of delay uncertainties at sender side is minimized; thus, the accuracy of time synchronization is improved. An Example of receiver-to-receiver protocols includes Reference Broadcast Synchronization (RBS) [3].

In a sender-to-receiver protocol, a sender stamps its sending time into its broadcasting messages. A receiver of the message also records its receiving time and then calculates the time offset relative to the sender. The receiver can use this time offset to adjust its clock accordingly. In Timing-sync Protocol for Sensor Networks (TPSN) [1], Ganeriwal et al have mathematically proved that, by time-stamping messages in the MAC layer, a sender-to-receiver protocol achieves better synchronization accuracy than a receiver-to-receiver protocol does. However, TPSN depends time synchronization on a tree structure in which a child synchronizes to only its parent node. Consequently, TPSN is vulnerable to network dynamics, which potentially occur in WSNs. Our proposed protocol (EGTSP) also uses the sender-to-receiver synchronization but does not construct any tree of time relationships. EGTSP relies on time information from all neighbors rather than one. As a result, our distributed and localized protocol is robust to network dynamics.

To solve the mentioned problem of TPSN, FTSP [7] floods time-synchronization messages from an elected root to provide redundant messages from several routes. However, FTSP's performance depends heavily on the root. Any change on the root (e.g., root failure, root position in the network, root re-election) will significantly degrade the performance. Furthermore, flooding techniques are notorious in draining network energy. Extending flooding periods will consequentially reduce synchronization accuracy. Furthermore, FTSP lacks the gradient time property that is desirable to most WSN applications.

Conversely, our proposed protocol does not depend on any root node. Every node equally participates in synchronization processes. Therefore, EGTSP is robust to a single point of failure. Additionally, with adaptive beaconing, EGTSP is also energy-efficient. At the beginning, nodes broadcast messages at a fixed period to compensate for time offsets and concurrently evaluate clock drifts. After the evaluation completes, nodes can extend a synchronization period, saving energy in broadcasting. In addition, given that EGTSP nodes compute their global time from all neighbors' time, EGTSP does not lack the gradient property.

Of a particular interest is Elapsed Time on Arrival (ETA) from Kusy et al. [9]. ETA provides the source-to-destination traveling time (elapsed time) of a message. Based on this elapsed time, the destination node can compute the relative time between the source and the destination. Understandably, this relative time is a basis to build a time synchronization protocol. However, ETA alone does not provide the global notion of time whereas our protocol does. Similarly, Tiny-sync [4] has been proposed to synchronize time between a pair of nodes. However, unlike our approach, Tiny-sync does not provide a mechanism to efficiently synchronize the time for the whole network.

In contrast to Tiny-sync, Time-diffusion [10] provides a converged global time of a sensor network. Nevertheless, each node must be a member of several trees as well as synchronize time with an elected diffused leader, causing high complexity in terms of overhead and memory. Conversely, the overhead and memory usage of our work are quite small.

To minimize overhead, Asynchronous Diffusion [11] requires only that each node participates in synchronization processes with a non-zero probability. It is proved that the time of an entire network can eventually converge to the global time. Each operation of Asynchronous Diffusion requires three-way message handshakes whereas each operation of our algorithm requires only one message broadcast. Hence, Asynchronous Diffusion potentially incurs more overhead than our work does when the participation probability is higher than 0.33. With a lower probability, the overhead should be reduced but the converging time may be adversely affected.

III. Energy-Efficient Gradient Time Synchronization Protocol

A wireless sensor network can be simply modeled as a connected graph, $C = (V, E)$, whereby V is a set of nodes and E is a set of communication links between nodes. Each sensor node is equipped with a local clock, L_i , that is initialized to Θ_i. The local clock progresses in accordance with its hardware clock tick h_i. Even if all nodes start with the same clock value, the local clocks of the nodes will drift apart from each other as time passes, called clock drifts. Therefore, nodes need to periodically exchange time information in order to maintain an acceptable level of synchronization error. The drift rate of each sensor node depends on temperature, battery power, and oscillator age. The clock model is defined as

$$L_i(t) = \int_{t_0}^{t} h_i(\tau)\, d\tau + \Theta_i(t_0) \, . \tag{1}$$

EGTSP estimates the global time and the average neighbor clock drift. The global time is the local time compensated by the average time offset (relative to the neighbors' perceived global time) and the average drift. If a node has the perceived global time higher than the average of the neighbors' global time, it will decrease its global time. Conversely, if a node has the global time lower than the average of the neighbors' global time, it will increase its global time accordingly. By repeating this process, the clocks of the sensor nodes in a network will converge to a common value, which is the global clock. Subsequently, the average neighbor clock drift is estimated using local clocks of the neighbors (relative to the computing node's local clock). If the average drift is more than 1 (the below line in Fig. 1), the clock of the computing node is running more slowly than that of its neighbors. Therefore, the node has to increase its global clock. If the average drift is less than 1 (the above line in Fig. 1), the clock of a node is running faster than that of its neighbors. Thus, the node has to decrease its global clock.

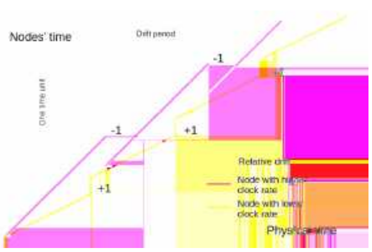

Figure 1. Neighborhood drift compensation

The inverse of an average drift indicates the amount of time for the node's perceived global clock to drift one time unit apart from the neighbors' global clock. This time amount is called a *drift period*. At the end of every drift period, nodes increase or decrease their global clocks one time unit. The benefits of the average-drift estimation are twofold.

1) Once the global clock is converged, the correctly estimated drift alone is sufficient to maintain the global time. Therefore, without network dynamics, nodes do not have to broadcast synchronization messages at all. Even if they do, the broadcast period can be extended. This will save energy in overhead significantly.
2) Sensor nodes can sleep and turn off the radio for a longer period to save even more energy, because they no longer have to frequently exchange the synchronization messages with each other. To maintain the global time after waking up, the node simply calculates how long it has slept and estimates how many time units it has to compensate for the drift.

In this paper, the global clock is modeled as

$$G_i(t) = L_i(t) + \theta_i(t) + \omega_i(t) \tag{2}$$

where $G_i(t)$is the global time of node i at time t, $L_i(t)$ is the local time and $\theta_i(t)$ is an offset compensation value

$$\theta_i(t) = \theta_i(t-1) + \frac{G_j(t-1) - G_i(t-1) + \gamma_i(t-1)}{|N_i| + 1} \tag{3}$$

where N_i is a neighbor set of node i and node $j \in N_i$, and $\gamma_i(t-1)$ is the remainder of the previous integer division. The remainder of an iteration is calculated by modular operation. $\omega_i(t)$ is a drift compensation value, defined as

$$\omega_i(t) = \begin{cases} \omega_i(t-1) + 1 & \text{every positive drift period} \\ \omega_i(t-1) - 1 & \text{every negative drift period} \\ \omega_i(t-1) & \text{else} \end{cases} \tag{4}$$

EGTSP's table structure, message structure, and algorithms can be described in Table I, Fig. 2 and, Alg. 1 and 2, respectively.

Table Field	Description
NodeID	A neighbor's node ID
SeqNum	A sequence number of a broadcast from a neighbor
SendLocalTime	Local sending time of the sender
RcvLocalTime	Local receiving time of the receiver

Table I
Table structure

Alg 1. starts by statically declaring the maximum number of entries of the table. This number should not be smaller than the maximum degree of nodes in the network. Upon

NodeID	SeqNum	SendLocalTime	SendGlobalTime

Figure 2. Message structure

Algorithm 1 Offset compensation

TableItem table[MAX_ENTRIES]
Node i receives a time sync msg from node j
Timestamp the local receiving time, L_i and also the global receiving time G_i using (2)
Sending node's time, $L_j = msg- > SendLocalTime$, $G_j = msg- > SendGlobalTime$
Update table, table.NodeID = j , $table.SeqNum = msg- > SeqNum$, table.SendLocalTime = L_j , table.RcvLocalTime = L_i
Calculate time offset between node i and j, $\theta_{i,j}$ = $G_j - G_i$
if $\theta_{i,j} > JUMP\_THRESHOLD$ **then**
    $G_i = G_i + \theta_{i,j}$
else
    $G_i = G_i + \theta_{i,j}/(N_i + 1)$ *// incremental averaging, see (3) for details*
end if
When the sync timer expires, node i broadcasts a time sync msg and resets the sync timer

receiving a message, node i timestamps both the global clock and the local clock. The global reception time is used to calculate the relative offset while the local reception time is used to further calculate the drift. If the relative offset is positively over a threshold, the receiver's global time will be adjusted to the sender's global time. Otherwise, the receiver's global time is added with a fraction of the relative offset. In a sense, the global time is incrementally averaged.

Alg 2. starts by declaring two more tables of the table structure in Table 1. The two tables are used to capture the relative neighbor drift for an extended period (CAPTURE_PERIOD). The longer period to estimate drifts, the less probability of estimation error [12]. In contrast to GTSP whereby a fixed period is used for both offset estimation and drift estimation, we argue that the periods for both estimations should be treated differently. A period for effective offset compensation should be shorter than the time-sync period with incremental averaging for smoother global time. Conversely, the period for effective drift compensation must be long enough to avoid estimation error but be short enough for the system to quickly start taking advantage of the drift estimation. For example, D_i of 1.000001 indicates that node i's clock drifts below the neighbor drift. Therefore, DP_i = 1/(1.000001 - 1) = 10^6 . If the precision of the hardware clock is millisecond, the node will increase the global time one millisecond every 16.67 minutes. As a result, WSN applications on this node can extend the time-sync period and the sleep-wake-up period as long as the system responsiveness to network dynamics and other metrics are still acceptable. This allows the WSN applications to optimize energy management, more flexibly and effectively than using a fixed beacon rate as proposed by other work.

Algorithm 2 Drift Compensation

TableItem startCapture[MAX_ENTRIES], stopCapture[MAX_ENTRIES];
Allow a period for nodes to collect initial reference points
Start capturing the drift by duplicating *table* to *startCapture*
For the CAPTURE_PERIOD, allowing newcoming reference points to update *table*
Stop capturing the drift by duplicating an updated *table* to *stopCapture*
Calculate relative drift by $D_i = \frac{\left(\sum_{j \in N_i} \frac{L_j^{(stop)} - L_j^{(start)}}{L_i^{(stop)} - L_i^{(start)}} + 1\right)}{N_i + 1}$
Calculate drift period, $DP_i = \frac{1}{(D_i - 1)}$
if $DP_i > 0$ **then**
    for every $|DP_i|$ period, $\omega_i = \omega_i + 1$
else if $DP_i < 0$ **then**
    for every $|DP_i|$ period, $\omega_i = \omega_i - 1$
end if
If the drift is converged, inform the application to choose a longer time-sync period.

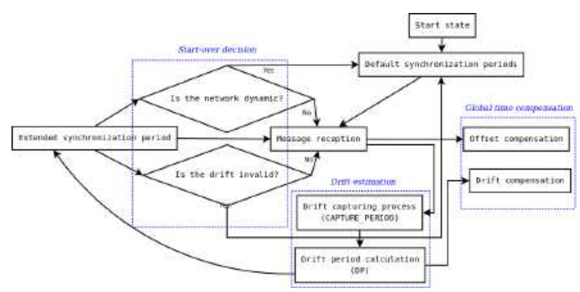

Figure 3. Algorithm Diagram

From Fig. 3, the algorithms will start over if the network dynamic is detected or the drift estimation is no longer accurate. Network dynamics include newly joining nodes, leaving nodes, or moving nodes. Furthermore, we need to detect if the neighbor drift estimation has become invalid. From Fig 1, one may observe that if sensor nodes' clocks are still compensating for a valid relative drift, the offset between any two nodes will not exceed two time unit. To enhance the confidence of the observation, nodes may count the consecutive times when this situation occurs.

Time synchronization messages are not only used to attain offset compensation, but also used to detect network dynamics and invalid relative drift. WSN applications will have to trade off system responsiveness (to network dynamics) with energy consumption. However, we argue that our time synchronization protocol consumes significantly less energy, poses considerably less overhead, and maintains acceptably small synchronization errors.

IV. Experimental Results

Our experiments are conduct on Telosb sensor nodes [13] running TinyOS 2.1. Our experiments consist of two scenarios: one-hop and multi-hop. Our multi-hop networking is software controlled. Specifically, the received packets will be dropped if the packets are sent from nodes that are not neighbors in the logical topology (even though they may be physically neighbors).

Variable	Value
Number of nodes	7
Default sync period	30 s
Global time sampling period	30 s
Experiment period (approx)	~10000 s
Drift capture period (of EGTSP)	300 s

Table II
EXPERIMENTAL SETTINGS

Algorithms	Total messages (bytes)
GTSP	54432
EGTSP	9856

Table III
MESSAGE OVERHEAD AND ENERGY CONSUMPTION : TOTAL TIME SYNCHRONIZATION MESSAGES (BYTES)

Algorithms	Single-hop	Multi-hop
GTSP	0.52	1.52
EGTSP	0.52	1.45

Table IV
ACCURACY COMPARISON: AVERAGE SYNCHRONIZATION ERRORS (MILLISECONDS)

EGTSP poses significantly less synchronization-message overhead than GTSP does in our experiment (see Table III). The main reason is that EGTSP can extend the synchronization period once the neighborhood drift is successfully estimated without degrading the synchronization error but GTSP cannot. If GTSP extends its synchronization period, the synchronization error of GTSP will increase (see below). Specifically, EGTSP can achieve 82% energy savings under investigated scenarios given that GTSP sends 5.5 times of the total synchronization messages sent by EGTSP.

In Fig. 4 and 5, our experimental results indicate that network time synchronization errors of EGTSP are comparable to those of GTSP in a single-hop network. Network synchronization errors are computed by averaging differences of perceived global time from all nodes in the network

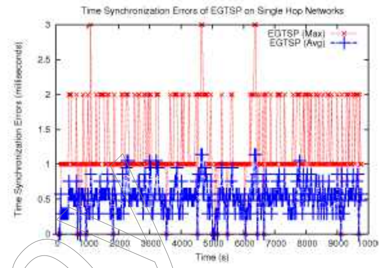

Figure 4. EGTSP's time synchronization errors on single-hop networks

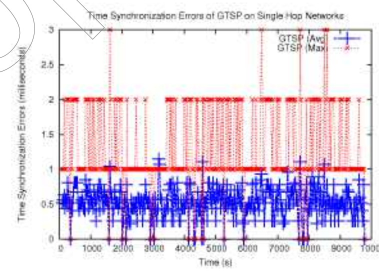

Figure 5. GTSP's time synchronization errors on single-hop networks

whereas neighbor synchronization errors are computed using only global time differences between neighbors. In a single-hop network, neighbor synchronization errors and network synchronization errors are equal.

In Fig. 6, EGTSP's neighbor synchronization errors (below) are smaller than network synchronization errors (above) in all samplings of the experiment. This result indicates that EGTSP also holds the gradient property of GTSP. In Fig. 7, ineffective drift estimation of GTSP causes synchronization errors to accumulate in the middle of the experiment. Because of its frequent synchronization messages, GTSP's synchronization errors are pulled back to a lower level again. Therefore, if the synchronization period of GTSP is extended, the duration of the higher-error level will be as well extended. Conversely, in Fig. 6, EGTSP is more stable than GTSP due to its incremental averaging (which adjusts time offset upon reception of a message) and its effective drift estimation (which is carefully used for drift compensation of the global time). As a result, our protocol is slightly more accurate but significantly more energy-efficient than GTSP (Table III and IV).

V. Conclusions

In this paper, we propose EGTSP, an energy-efficient gradient time synchronization protocol for providing a global notion of time in wireless sensor networks. EGTSP is distributed, gradient-based, and energy-efficiency-oriented. Therefore, we introduce a concept of neighborhood drifts for effective drift compensation and adaptive adjustment of broadcasting periods to save energy. EGTSP also includes frequent offset compensation with incremental averaging to

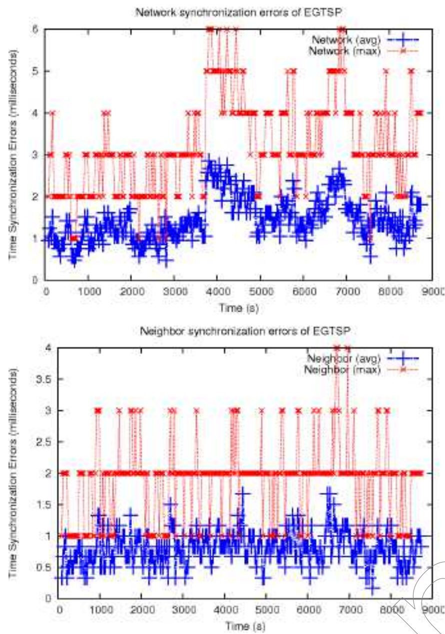

Figure 6. EGTSP's synchronization errors on multi-hop networks between all nodes (above) and between neighbors (below)

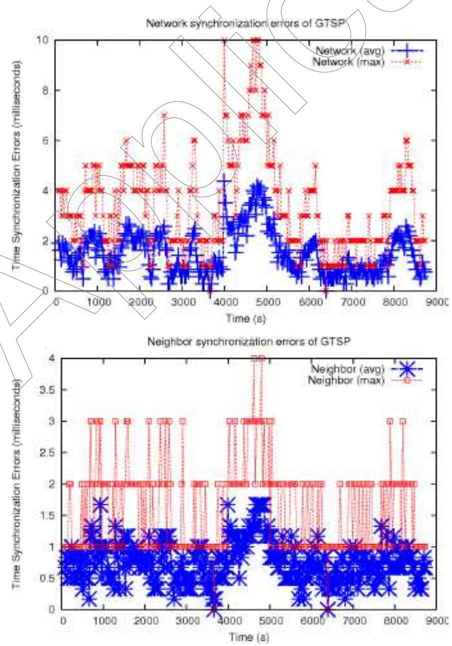

Figure 7. GTSP's synchronization errors on multihop networks between all nodes (above) and between neighbors (below)

smooth the global time estimation. We evaluate our protocol on Telosb sensor motes and compare the performance of EGTSP with that of GTSP. The experimental results indicate that, under investigated scenarios, our synchronization protocol is much more energy-efficient than GTSP is, without degrading the time synchronization accuracy. Future work includes improving an accuracy of clock drift evaluation and simulating EGTSP on a more scalable topology.

References

[1] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*. ACM, 2003.

[2] K. Römer, "Time synchronization in ad hoc networks," in *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*. ACM, 2001.

[3] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *OSDI '02: Proceedings of the 5th symposium on Operating systems design and implementation*. ACM, 2002.

[4] S. Yoon, C. Veerarittiphan, and M. L. Sichitiu, "Tiny-sync: Tight time synchronization for wireless sensor networks," *ACM Trans. Sen. Netw.*, vol. 3, no. 2, p. 8, 2007.

[5] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Commun. ACM*, vol. 21, no. 7, pp. 558–565, 1978.

[6] D. L. Mills, "Internet time synchronization: The network time protocol," 1989.

[7] M. Maróti, B. Kusy, G. Simon, and A. Lédeczi, "The flooding time synchronization protocol," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM, 2004.

[8] P. Sommer and R. Wattenhofer, "Gradient clock synchronization in wireless sensor networks," in *IPSN '09: Proceedings of the 2009 International Conference on Information Processing in Sensor Networks*. IEEE Computer Society, 2009.

[9] B. Kusy, P. Dutta, P. Levis, M. Maroti, A. Ledeczi, and D. Culler, "Elapsed time on arrival: a simple and versatile primitive for canonical time synchronisation services," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 1, no. 4, pp. 239–251, 2006.

[10] W. Su and I. F. Akyildiz, "Time-diffusion synchronization protocol for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 384–397, 2005.

[11] Q. Li and D. Rus, "Global clock synchronization in sensor networks," *IEEE Trans. Comput.*, vol. 55, no. 2, pp. 214–226, 2006.

[12] C. Lenzen, P. Sommer, and R. Wattenhofer, "Optimal clock synchronization in networks," in *SenSys '09: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2009.

[13] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*. IEEE Press, 2005.