Mike Hazas
John Krumm
Thomas Strang (Eds.)

LNCS 3987

Location- and Context-Awareness

Second International Workshop, LoCA 2006
Dublin, Ireland, May 2006
Proceedings

Springer

Mike Hazas John Krumm
Thomas Strang (Eds.)

Location- and Context-Awareness

Second International Workshop, LoCA 2006
Dublin, Ireland, May 10-11, 2006
Proceedings

Springer

Volume Editors

Mike Hazas
Lancaster University
Computing Department, Infolab
South Drive, Lancaster, LA1 4WA, UK
E-mail: hazas@comp.lancs.ac.uk

John Krumm
Microsoft Corporation
One Microsoft Way, Redmond, WA 98052, USA
E-mail: jckrumm@microsoft.com

Thomas Strang
Deutsches Zentrum für Luft- und Raumfahrt
P.O. Box 1116, 82234 Wessling/Oberpfaffenhofen, Germany
E-mail: Thomas.Strang@dlr.de

Library of Congress Control Number: 2006924878

CR Subject Classification (1998): H.3, H.4, C.2, H.5, K.8

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-34150-1 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-34150-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11752967 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3987

Lecture Notes in Computer Science

For information about Vols. 1–3858

please contact your bookseller or Springer

Vol. 3987: M. Hazas, J. Krumm, T. Strang (Eds.), Location- and Context-Awareness. X, 289 pages. 2006.

Vol. 3968: K.P. Fishkin, B. Schiele, P. Nixon, A. Quigley (Eds.), Pervasive Computing. XV, 402 pages. 2006.

Vol. 3958: M. Yung, Y. Dodis, A. Kiayias, T. Malkin (Eds.), Public Key Cryptography - PKC 2006. XIV, 543 pages. 2006.

Vol. 3956: G. Barthe, B. Gregoire, M. Huisman, J.-L. Lanet (Eds.), Construction and Analysis of Safe, Secure, and Interoperable Smart Devices. IX, 175 pages. 2006.

Vol. 3955: G. Antoniou, G. Potamias, C. Spyropoulos, D. Plexousakis (Eds.), Advances in Artificial Intelligence. XVII, 611 pages. 2006. (Sublibrary LNAI).

Vol. 3954: A. Leonardis, H. Bischof, A. Pinz (Eds.), Computer Vision – ECCV 2006, Part IV. XVII, 613 pages. 2006.

Vol. 3953: A. Leonardis, H. Bischof, A. Pinz (Eds.), Computer Vision – ECCV 2006, Part III. XVII, 649 pages. 2006.

Vol. 3952: A. Leonardis, H. Bischof, A. Pinz (Eds.), Computer Vision – ECCV 2006, Part II. XVII, 661 pages. 2006.

Vol. 3951: A. Leonardis, H. Bischof, A. Pinz (Eds.), Computer Vision – ECCV 2006, Part I. XXXV, 639 pages. 2006.

Vol. 3947: Y.-C. Chung, J.E. Moreira (Eds.), Advances in Grid and Pervasive Computing. XXI, 667 pages. 2006.

Vol. 3946: T.R. Roth-Berghofer, S. Schulz, D.B. Leake (Eds.), Modeling and Retrieval of Context. XI, 149 pages. 2006. (Sublibrary LNAI).

Vol. 3945: M. Hagiya, P. Wadler (Eds.), Functional and Logic Programming. X, 295 pages. 2006.

Vol. 3944: J. Quiñonero-Candela, I. Dagan, B. Magnini, F. d'Alché-Buc (Eds.), Machine Learning Challenges. XIII, 462 pages. 2006. (Sublibrary LNAI).

Vol. 3942: Z. Pan, R. Aylett, H. Diener, X. Jin, S. Göbel, L. Li (Eds.), Technologies for E-Learning and Digital Entertainment. XXV, 1396 pages. 2006.

Vol. 3939: C. Priami, L. Cardelli, S. Emmott (Eds.), Transactions on Computational Systems Biology IV. VII, 141 pages. 2006. (Sublibrary LNBI).

Vol. 3936: M. Lalmas, A. MacFarlane, S. Rüger, A. Tombros, T. Tsikrika, A. Yavlinsky (Eds.), Advances in Information Retrieval. XIX, 584 pages. 2006.

Vol. 3935: D. Won, S. Kim (Eds.), Information Security and Cryptology - ICISC 2005. XIV, 458 pages. 2006.

Vol. 3934: J.A. Clark, R.F. Paige, F.A. C. Polack, P.J. Brooke (Eds.), Security in Pervasive Computing. X, 243 pages. 2006.

Vol. 3933: F. Bonchi, J.-F. Boulicaut (Eds.), Knowledge Discovery in Inductive Databases. VIII, 251 pages. 2006.

Vol. 3931: B. Apolloni, M. Marinaro, G. Nicosia, R. Tagliaferri (Eds.), Neural Nets. XIII, 370 pages. 2006.

Vol. 3930: D.S. Yeung, Z.-Q. Liu, X.-Z. Wang, H. Yan (Eds.), Advances in Machine Learning and Cybernetics. XXI, 1110 pages. 2006. (Sublibrary LNAI).

Vol. 3929: W. MacCaull, M. Winter, I. Düntsch (Eds.), Relational Methods in Computer Science. VIII, 263 pages. 2006.

Vol. 3928: J. Domingo-Ferrer, J. Posegga, D. Schreckling (Eds.), Smart Card Research and Advanced Applications. XI, 359 pages. 2006.

Vol. 3927: J. Hespanha, A. Tiwari (Eds.), Hybrid Systems: Computation and Control. XII, 584 pages. 2006.

Vol. 3925: A. Valmari (Ed.), Model Checking Software. X, 307 pages. 2006.

Vol. 3924: P. Sestoft (Ed.), Programming Languages and Systems. XII, 343 pages. 2006.

Vol. 3923: A. Mycroft, A. Zeller (Eds.), Compiler Construction. XIII, 277 pages. 2006.

Vol. 3922: L. Baresi, R. Heckel (Eds.), Fundamental Approaches to Software Engineering. XIII, 427 pages. 2006.

Vol. 3921: L. Aceto, A. Ingólfsdóttir (Eds.), Foundations of Software Science and Computation Structures. XV, 447 pages. 2006.

Vol. 3920: H. Hermanns, J. Palsberg (Eds.), Tools and Algorithms for the Construction and Analysis of Systems. XIV, 506 pages. 2006.

Vol. 3918: W.K. Ng, M. Kitsuregawa, J. Li, K. Chang (Eds.), Advances in Knowledge Discovery and Data Mining. XXIV, 879 pages. 2006. (Sublibrary LNAI).

Vol. 3917: H. Chen, F.Y. Wang, C.C. Yang, D. Zeng, M. Chau, K. Chang (Eds.), Intelligence and Security Informatics. XII, 186 pages. 2006.

Vol. 3916: J. Li, Q. Yang, A.-H. Tan (Eds.), Data Mining for Biomedical Applications. VIII, 155 pages. 2006. (Sublibrary LNBI).

Vol. 3915: R. Nayak, M.J. Zaki (Eds.), Knowledge Discovery from XML Documents. VIII, 105 pages. 2006.

Vol. 3914: A. Garcia, R. Choren, C. Lucena, P. Giorgini, T. Holvoet, A. Romanovsky (Eds.), Software Engineering for Multi-Agent Systems IV. XIV, 255 pages. 2006.

Vol. 3910: S.A. Brueckner, G.D.M. Serugendo, D. Hales, F. Zambonelli (Eds.), Engineering Self-Organising Systems. XII, 245 pages. 2006. (Sublibrary LNAI).

Vol. 3909: A. Apostolico, C. Guerra, S. Istrail, P. Pevzner, M. Waterman (Eds.), Research in Computational Molecular Biology. XVII, 612 pages. 2006. (Sublibrary LNBI).

Vol. 3908: A. Bui, M. Bui, T. Böhme, H. Unger (Eds.), Innovative Internet Community Systems. VIII, 207 pages. 2006.

Vol. 3907: F. Rothlauf, J. Branke, S. Cagnoni, E. Costa, C. Cotta, R. Drechsler, E. Lutton, P. Machado, J.H. Moore, J. Romero, G.D. Smith, G. Squillero, H. Takagi (Eds.), Applications of Evolutionary Computing. XXIV, 813 pages. 2006.

Vol. 3906: J. Gottlieb, G.R. Raidl (Eds.), Evolutionary Computation in Combinatorial Optimization. XI, 293 pages. 2006.

Vol. 3905: P. Collet, M. Tomassini, M. Ebner, S. Gustafson, A. Ekárt (Eds.), Genetic Programming. XI, 361 pages. 2006.

Vol. 3904: M. Baldoni, U. Endriss, A. Omicini, P. Torroni (Eds.), Declarative Agent Languages and Technologies III. XII, 245 pages. 2006. (Sublibrary LNAI).

Vol. 3903: K. Chen, R. Deng, X. Lai, J. Zhou (Eds.), Information Security Practice and Experience. XIV, 392 pages. 2006.

Vol. 3901: P.M. Hill (Ed.), Logic Based Program Synthesis and Transformation. X, 179 pages. 2006.

Vol. 3900: F. Toni, P. Torroni (Eds.), Computational Logic in Multi-Agent Systems. XVII, 427 pages. 2006. (Sublibrary LNAI).

Vol. 3899: S. Frintrop, VOCUS: A Visual Attention System for Object Detection and Goal-Directed Search. XIV, 216 pages. 2006. (Sublibrary LNAI).

Vol. 3898: K. Tuyls, P.J. 't Hoen, K. Verbeeck, S. Sen (Eds.), Learning and Adaption in Multi-Agent Systems. X, 217 pages. 2006. (Sublibrary LNAI).

Vol. 3897: B. Preneel, S. Tavares (Eds.), Selected Areas in Cryptography. XI, 371 pages. 2006.

Vol. 3896: Y. Ioannidis, M.H. Scholl, J.W. Schmidt, F. Matthes, M. Hatzopoulos, K. Boehm, A. Kemper, T. Grust, C. Boehm (Eds.), Advances in Database Technology - EDBT 2006. XIV, 1208 pages. 2006.

Vol. 3895: O. Goldreich, A.L. Rosenberg, A.L. Selman (Eds.), Theoretical Computer Science. XII, 399 pages. 2006.

Vol. 3894: W. Grass, B. Sick, K. Waldschmidt (Eds.), Architecture of Computing Systems - ARCS 2006. XII, 496 pages. 2006.

Vol. 3893: L. Atzori, D.D. Giusto, R. Leonardi, F. Pereira (Eds.), Visual Content Processing and Representation. IX, 224 pages. 2006.

Vol. 3891: J.S. Sichman, L. Antunes (Eds.), Multi-Agent-Based Simulation VI. X, 191 pages. 2006. (Sublibrary LNAI).

Vol. 3890: S.G. Thompson, R. Ghanea-Hercock (Eds.), Defence Applications of Multi-Agent Systems. XII, 141 pages. 2006. (Sublibrary LNAI).

Vol. 3889: J. Rosca, D. Erdogmus, J.C. Príncipe, S. Haykin (Eds.), Independent Component Analysis and Blind Signal Separation. XXI, 980 pages. 2006.

Vol. 3888: D. Draheim, G. Weber (Eds.), Trends in Enterprise Application Architecture. IX, 145 pages. 2006.

Vol. 3887: J.R. Correa, A. Hevia, M. Kiwi (Eds.), LATIN 2006: Theoretical Informatics. XVI, 814 pages. 2006.

Vol. 3886: E.G. Bremer, J. Hakenberg, E.-H.(S.) Han, D. Berrar, W. Dubitzky (Eds.), Knowledge Discovery in Life Science Literature. XIV, 147 pages. 2006. (Sublibrary LNBI).

Vol. 3885: V. Torra, Y. Narukawa, A. Valls, J. Domingo-Ferrer (Eds.), Modeling Decisions for Artificial Intelligence. XII, 374 pages. 2006. (Sublibrary LNAI).

Vol. 3884: B. Durand, W. Thomas (Eds.), STACS 2006. XIV, 714 pages. 2006.

Vol. 3882: M.L. Lee, K.-L. Tan, V. Wuwongse (Eds.), Database Systems for Advanced Applications. XIX, 923 pages. 2006.

Vol. 3881: S. Gibet, N. Courty, J.-F. Kamp (Eds.), Gesture in Human-Computer Interaction and Simulation. XIII, 344 pages. 2006. (Sublibrary LNAI).

Vol. 3880: A. Rashid, M. Aksit (Eds.), Transactions on Aspect-Oriented Software Development I. IX, 335 pages. 2006.

Vol. 3879: T. Erlebach, G. Persinao (Eds.), Approximation and Online Algorithms. X, 349 pages. 2006.

Vol. 3878: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing. XVII, 589 pages. 2006.

Vol. 3877: M. Detyniecki, J.M. Jose, A. Nürnberger, C. J. '. van Rijsbergen (Eds.), Adaptive Multimedia Retrieval: User, Context, and Feedback. XI, 279 pages. 2006.

Vol. 3876: S. Halevi, T. Rabin (Eds.), Theory of Cryptography. XI, 617 pages. 2006.

Vol. 3875: S. Ur, E. Bin, Y. Wolfsthal (Eds.), Hardware and Software, Verification and Testing. X, 265 pages. 2006.

Vol. 3874: R. Missaoui, J. Schmidt (Eds.), Formal Concept Analysis. X, 309 pages. 2006. (Sublibrary LNAI).

Vol. 3873: L. Maicher, J. Park (Eds.), Charting the Topic Maps Research and Applications Landscape. VIII, 281 pages. 2006. (Sublibrary LNAI).

Vol. 3872: H. Bunke, A. L. Spitz (Eds.), Document Analysis Systems VII. XIII, 630 pages. 2006.

Vol. 3871: E.-G. Talbi, P. Liardet, P. Collet, E. Lutton, M. Schoenauer (Eds.), Artificial Evolution. XI, 310 pages. 2006.

Vol. 3870: S. Spaccapietra, P. Atzeni, W.W. Chu, T. Catarci, K.P. Sycara (Eds.), Journal on Data Semantics V. XIII, 237 pages. 2006.

Vol. 3869: S. Renals, S. Bengio (Eds.), Machine Learning for Multimodal Interaction. XIII, 490 pages. 2006.

Vol. 3868: K. Römer, H. Karl, F. Mattern (Eds.), Wireless Sensor Networks. XI, 342 pages. 2006.

Vol. 3866: T. Dimitrakos, F. Martinelli, P.Y.A. Ryan, S. Schneider (Eds.), Formal Aspects in Security and Trust. X, 259 pages. 2006.

Vol. 3865: W. Shen, K.-M. Chao, Z. Lin, J.-P.A. Barthès, A. James (Eds.), Computer Supported Cooperative Work in Design II. XII, 659 pages. 2006.

Vol. 3863: M. Kohlhase (Ed.), Mathematical Knowledge Management. XI, 405 pages. 2006. (Sublibrary LNAI).

Vol. 3862: R.H. Bordini, M. Dastani, J. Dix, A.E.F. Seghrouchni (Eds.), Programming Multi-Agent Systems. XIV, 267 pages. 2006. (Sublibrary LNAI).

Vol. 3861: J. Dix, S.J. Hegner (Eds.), Foundations of Information and Knowledge Systems. X, 331 pages. 2006.

Vol. 3860: D. Pointcheval (Ed.), Topics in Cryptology – CT-RSA 2006. XI, 365 pages. 2006.

¥385.00元

Preface

These proceedings contain the papers presented at the 2nd International Workshop on Location- and Context-Awareness in May of 2006. As computing moves increasingly into the everyday world, the importance of location and context knowledge grows. The range of contexts encountered while sitting at a desk working on a computer is very limited compared to the large variety of situations experienced away from the desktop. For computing to be relevant and useful in these situations, the computers must have knowledge of the user's activity, resources, state of mind, and goals, i.e., the user's context, of which location is an important indicator. This workshop was intended to present research aimed at sensing, inferring, and using location and context data in ways that help the user.

Our call for papers resulted in 74 submissions, each of which was assigned to members of our Program Committee. After reviews and email discussion, we selected 18 papers for publication in these proceedings. Most of the accepted papers underwent a shepherding process by a reviewer or a member of the Program Committee to ensure that the reviewers' comments were accounted for in the published version. We feel our selective review process and shepherding phase have resulted in a high-quality set of published papers.

We extend a sincere "thank you" to all the authors who submitted papers, to our hard-working Program Committee, our thoughtful reviewers, and our conscientious shepherds.

May 2006

Mike Hazas and John Krumm, Program Co-chairs
Thomas Strang, Workshop Chair

Organization

Program Committee

Gaetano Borriello	University of Washington and Intel Research Seattle
Anind Dey	Carnegie Mellon University
William Griswold	University of California, San Diego
Robert Harle	University of Cambridge
Jeffrey Hightower	Intel Research Seattle
Minkyong Kim	Dartmouth College
Gabriele Kotsis	Johannes Kepler University of Linz
Marc Langheinrich	ETH Zurich
Claudia Linnhoff-Popien	Ludwig Maximilian University Munich
Henk Muller	University of Bristol
Chandrasekhar Narayanaswami	IBM T.J. Watson Research Center
Harlan Onsrud	University of Maine
Donald Patterson	University of California, Irvine
Thorsten Prante	Fraunhofer IPSI
Aaron Quigley	University College Dublin
Bernt Schiele	Darmstadt University of Technology
Chris Schmandt	MIT Media Lab
Flavia Sparacino	Sensing Places and MIT
Thomas Strang	German Aerospace Center and University of Innsbruck
Yasuyuki Sumi	Kyoto University
Hiroyuki Tarumi	Kagawa University
Daniel Wilson	Author

Reviewers

Ian Anderson	University of Bristol
Michael Beigl	University of Karlsruhe
Alastair Beresford	University of Cambridge
David Cottingham	University of Cambridge
Florian Fuchs	Siemens and Ludwig Maximilian University Munich
Caroline Funk	Ludwig Maximilian University Munich
Thomas Grill	Johannes Kepler University of Linz
Tom Gross	Bauhaus University Weimar
Sinem Guven	Columbia University
Ismail Ibrahim	Johannes Kepler University of Linz
Axel Küpper	Ludwig Maximilian University Munich
David Molyneaux	Lancaster University
Mandayam Raghunath	IBM T.J. Watson Research Center

Anand Ranganathan	IBM T.J. Watson Research Center
Wieland Schwinger	Johannes Kepler University of Linz
Peter Tandler	Fraunhofer IPSI
Georg Treu	Ludwig Maximilian University Munich
Diana Weiss	Ludwig Maximilian University Munich

Shepherds

Alastair Beresford	University of Cambridge
Gaetano Borriello	University of Washington and Intel Research Seattle
Sinem Guven	Columbia University
Robert Harle	University of Cambridge
Mike Hazas	Lancaster University
Jeffrey Hightower	Intel Research Seattle
Minkyong Kim	Dartmouth College
Marc Langheinrich	ETH Zurich
Henk Muller	University of Bristol
Aaron Quigley	University College Dublin
Flavia Sparacino	Sensing Places and MIT
Daniel Wilson	Author

Table of Contents

Location Sensing

Mapping

Privacy and Access

Context Sensing

Social Context

Representation and Programming

Particle Filters for Position Sensing with Asynchronous Ultrasonic Beacons*

Henk L. Muller, Michael McCarthy, and Cliff Randell

Department of Computer Science, University of Bristol, UK
http://www.cs.bris.ac.uk/

Abstract. In this paper we present a user-centric position sensing system that is based on asynchronous, independent ultrasonic beacons. These stationary transmitter units are small, cheap to manufacture, and have power requirements low enough to run each from a small solar cell and a nearby light source. Each beacon is programmed to emit a short, 40 kHz ultrasonic signal with a unique transmission period. The mobile receiver unit first associates a received signal with a beacon based on the observed periodicity, then measures the Doppler shift in the periodicity that results from movements of the receiver. Using Doppler shifts from a number of different beacons, the receiver is able to estimate both its position and velocity by employing a particle filter. In this paper, we describe our positioning algorithm, the hardware, and proof-of-concept results.

1 Introduction

There are many design decisions involved in building a positioning system. Traditionally, accuracy and system costs are the two most important factors. Other factors include the weight and power consumption of the mobile unit, the refresh rate, whether the system is user-based or infrastructure-based, the positioning coverage, the ease of setup and maintenance, and aesthetic impact.

We present a system which has been designed to optimise aesthetics and component costs. The system is based around small, low power beacons that are fixed within a room. The beacons are also wireless and independent, making it possible to minimise their visual impact. We do not use RF or any other synchronising technology, which helps to minimise power consumption and bring component numbers down. The power consumption of the beacons is sufficiently low to enable them to be powered by a small solar cell, illuminated by a nearby domestic or office light source.

The beacons produce only minimal information: they transmit ultrasonic signals (or 'chirps') with a regular periodicity, usually around 500 ms. The receiver unit consists of an ultrasonic transducer and a medium-power processing device. It approximates its position with an accuracy of, at present, 25 cm by measuring the Doppler shift in the transmission periods. This accuracy may seem low, but we expect it to be adequate for use in coarse-grained location-based applications such as automated museum guides or indoor navigation aids. Also, we believe that the accuracy can be improved by fine tuning the system.

* Funding for this work is received from the U.K. Engineering and Physical Sciences Research Council as part of the Equator IRC, GR-N-15986.

M. Hazas, J. Krumm, and T. Strang (Eds.): LoCA 2006, LNCS 3987, pp. 1–13, 2006.

In the design space, we have made a trade-off between costs and aesthetics on the one hand, and accuracy on the other hand. Our accuracy is better than that obtained with some RF based systems such as those using WiFi [1] but not as accurate as other ultrasound-based positioning systems [2, 3, 4]. Our system is similar to the Cricket [5] in that wireless beacons are used to infer position. However, we believe that our design provides a new contribution by minimising the size and power consumption of the devices in the infrastructure. By setting this as our main design constraint we have been forced to come up with novel methods for inferring position.

In the rest of this paper, we first introduce the problem domain in Section 2, then present our algorithms in Sections 3, 4 and 5. Section 6 details the hardware that we used for the initial results, which are shown in Section 7. We discuss improvements that we believe can be made to our algorithms in Section 8.

2 The Problem

Our positioning system comprises a number of beacons scattered around a room (mostly along the ceiling and walls) and one or more mobile receivers that can position themselves using the signals received.

The beacon units (described in detail in Section 6) have been designed to be simple, low power, and require no wiring. In order to reduce complexity, they only transmit ultrasound. The mobile receivers are untethered, and comprise an ultrasonic microphone, amplifier, and a processing unit to estimate position and velocity.

The principle mode of operation of our system is to estimate the position by measuring the Doppler shift in the periodicity of each of the beacons. We assume that the receiver knows that there are N beacons in a room (around eight), and that the receiver knows the position $\boldsymbol{T}_i$ of each beacon i ($0 \leq i < N$). Furthermore, the receiver has knowledge of the transmission periods P_i, which are around 500 ms.

The receiving unit will ideally receive a pulse train as shown in Figure 1. The horizontal axis is time in seconds, the vertical bars denote the arrival of a chirp from a beacon. Because of the different periodicities of, for example, Beacons 6 and 0, their reception times shift relative to each other. A real chirp train that has been recorded using our hardware is shown in Figure 2. Some signals have gone missing altogether (such as the signal from Beacon 6), some signals collide (chirps from Beacons 2 and 3 arrive simultaneously at time 1006), and there are reflections and noise (signified by shorter bars) which muddle the picture.

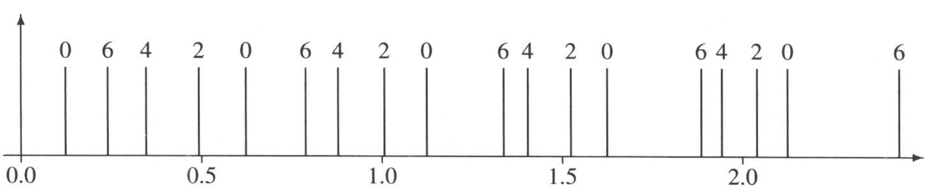

Fig. 1. Ideal chirp trace received from four beacons

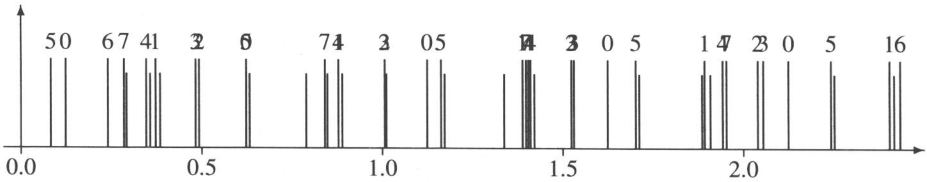

Fig. 2. Actual chirp trace received from eight beacons. Note that the labelling is added for clarification only, no identification is received.

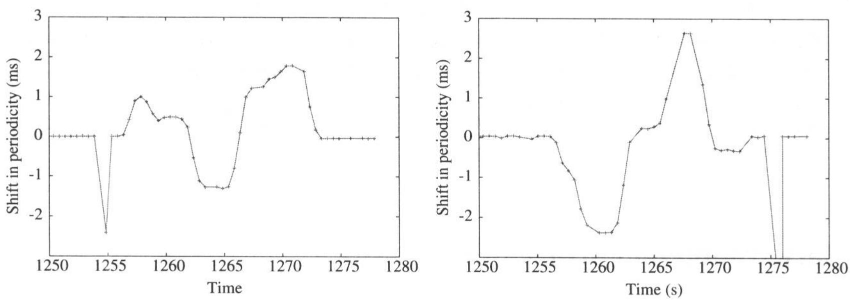

Fig. 3. Deviations from mean periods for two receivers over a period of time. The time (X-axis) is measured in seconds, the deviation (Y-axis) is measured in milliseconds.

When the receiver is static, the reception time of the pulses from each beacon are spaced by their periods P_i. This is a method for determining the correspondence between pulses and beacons; a series of chirps separated by P_i is likely to come from Beacon i. Measurements from a static receiver also enable us to measure transmission periods P_i to a high precision. This may be necessary as clock crystals used in the beacons and the receiver will have slightly different frequencies (up to 500 ppm). This difference may cause drifting of the position when not corrected for.

When the receiver is moving, the transmission periods will appear to vary. As it moves through the tracking area, the receiver's distance to each beacon will change. This will affect the time that it takes for the signals to travel, causing the separation between chirps to vary. The changes in separation observed by the receiver is, essentially, a Doppler shift in the periodicity of each beacon. If the receiver moves towards a beacon, the pulses will be brought closer together; if the receiver moves away, the pulses will be pushed further apart. To illustrate, some measured Doppler shifts are provided in Figure 3. Note that the outlying downward spikes are caused by missing chirps, all other deviations are caused by the receiver's movement.

The amount the pulses shift is proportional to the distance moved over the period:

$$\Delta d = v_s \Delta P_i$$

Here, Δd is the movement of the receiver relative to the beacon and v_s is the speed of sound, 343 ms^{-1}. This distance is equal to the distance travelled in the direction of the beacon over the time period $P_i + \Delta P_i$ [6]:

$$\Delta d = \frac{(\boldsymbol{X_0} - \boldsymbol{T_i})}{|\boldsymbol{X_0} - \boldsymbol{T_i}|} \cdot (\boldsymbol{X_0} - \boldsymbol{X_{P_i+\Delta P_i}})$$

Hence,

$$v_s \Delta P_i = \frac{(\boldsymbol{X_0} - \boldsymbol{T_i})}{|\boldsymbol{X_0} - \boldsymbol{T_i}|} \cdot (\boldsymbol{X_0} - \boldsymbol{X_{P_i+\Delta P_i}}) \tag{1}$$

Equation 1 formulates a relationship between two expressions: one containing the speed of sound v_s and the shift in periodicity ΔP_i on the left-hand side; and the other containing the previous position of the mobile receiver $\boldsymbol{X_0}$, the location $\boldsymbol{T_i}$ of transmitter i and the current position $\boldsymbol{X}_{P_i+\Delta P_i}$ (the current time is $P_i + \Delta P_i$, where P_i is the periodicity of transmitter i) on the right-hand side. With a sufficient number of readings, it is possible to iteratively estimate the receiver position using this equation. We have found that a particle filter is capable of performing this estimation, even in the presence of noise.

3 Background on Particle Filters

In order to estimate position, we use *particle filters* in two different parts of our algorithm. In this section, we provide a brief background on this type of estimator.

Particle filters [7] are a method to approximate random variables. This is similar to a Kalman filter [8, 9], except that a particle filter can cope with arbitrary error distributions and, in particular, with multi-modal distributions.

A particle filter requires two sets of equations to operate. One set of equations models state progression. This is similar to the state transition of a Kalman filter, except that random noise is added to the state on every iteration. This noise should be distributed according to the variation that is expected in the system over that period of time.

The second set of equations should compute the likelihood of a measured observation, given a particular state of the system. This likelihood function models the error distribution of the measurements.

A particle filter operates by maintaining a number of states in parallel, called *particles*. Each particle is progressed using the state equations, after which the likelihood of each particle is computed. The probability distribution function (PDF) of the state space is modelled by the particles, and can have any shape, as long as there are enough particles to sample this shape.

In order to prevent the filter from deterioating, the filter can be periodically *resampled*. Resampling is performed by first translating the set of likelihoods into a cumulative distribution function (CDF). The CDF can be sampled using a uniform random number, duplicating particles in areas of the state space that have a high probability, and removing particles in the areas of the state space that have a low probability.

The final stage of a particle filter is to distill a single state from the particles. In many cases the *mean* state of all particles is useful, especially on single-modal distributions. In multi-modal state spaces one can compute the *mode* of the distribution, which can be problematic as the mode of the state can change dramatically from one iteration to the next.

In comparison with a Kalman filter, a particle filter is advantageous if it is expensive to compute the Jacobian of the measurement equation, if the state space errors are multi-modal, or if one intends to use low precision arithmetic.

Particle filters have been used in the estimation of location successfully before. They have been used both for low level filtering of data, and in order to fuse sensor information at a high level (see for example [10, 11]).

4 Associating Chirps with Beacons

The first problem is that of associating each chirp in Figure 2 with a beacon, or classifying it as a reflection or noise. The only information that is available to the receiver is an estimate of the periodicity of each beacon. It is an estimate in that the clock crystals on each beacon are only cut with a limited precision, and their frequency varies with temperature. The typical variation is in the order of 50 ppm, or 25 μs in a 500 ms period. In addition, movement of the receiver will cause further variation. Assuming a maximum movement of the receiver of 2 ms^{-1}, we expect this to be on the order of $2/v_s = 2/343 = 0.58\%$, or 2.9 ms in a 500 ms period.

Our strategy to solve the measurement association problem is to use a particle filter to model each beacon. The particle filter modelling beacon i estimates the time that this beacon produced its first chirp, a value that we call the *Epoch* and represent using the variable E (a number between 0 and P_i). Hence, the state of the filter is just a one-dimensional variable that tracks E. The state progression function adds noise to the state with a standard deviation of 50 μs.

When a chirp is received at time t, the likelihood function is applied to the particle filter. For a particle j with value E_j, the likelihood of $t - E_j$ is computed using the sawtooth curve in Figure 4. If the chirp corroborates the epoch time of the particle, then the probability will fall on one of the peaks of the sawtooth. If it is off by, say, half a period, then the particle will be unsupported. If the signal is slightly late, then the probably function shows intermediate support (the diagonal of the sawtooth) as the chirp may be a reflection of the original signal. The closer to the peak a measurement is, the more likely that it is a reflection. Particles will converge first on all sequences of signals that are P apart, including systematic reflections. The groups of particles

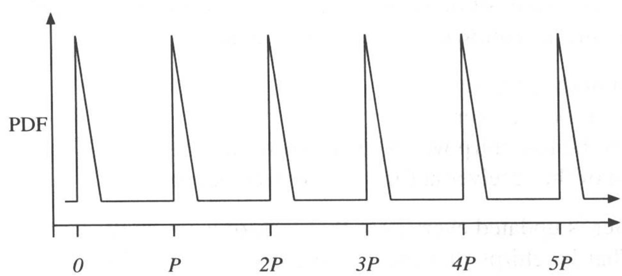

Fig. 4. Likelihood for a particle to associate with a beacon

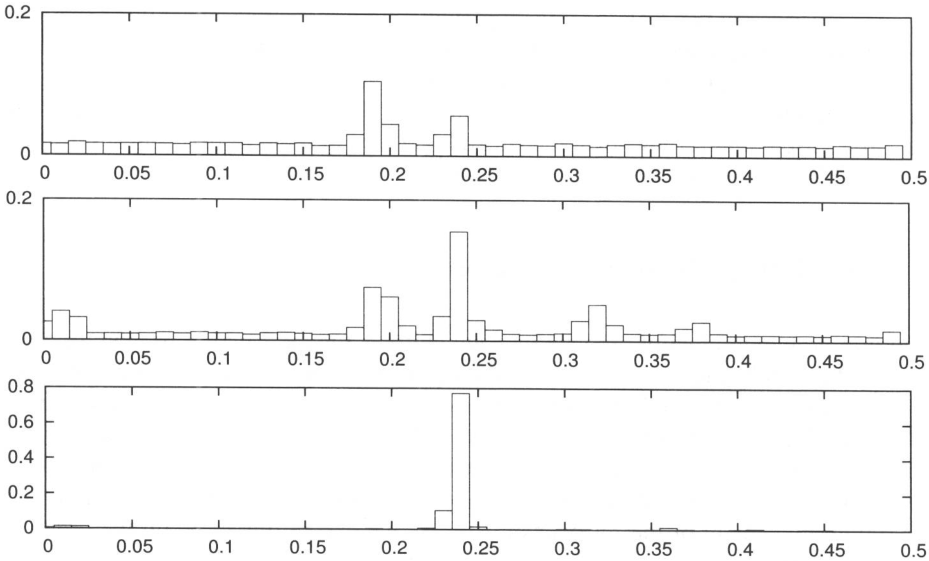

Fig. 5. Distribution of the particle filter after 0.5, 1 and 2 seconds

tracking reflections will climb up the slope and merge with the main group, latching on to the actual chirps from the beacon.

The evolution of the distribution of particles is visualised in Figure 5, which shows the probability distribution functions of the particles for Beacon 0. The graphs are 0.5, 1, and 2 seconds into the run, showing that the particles latch onto Beacon 0 after 2 seconds (equivalent to four chirps).

It is possible for the particle filter to latch onto noise that is roughly separated by P_i. In order to avoid this, we keep refreshing a fraction of the particles if we have not yet observed a consistent signal. Using this method, we usually latch onto all beacons within a 10 second period.

5 Estimating the Position of the Receiver

In order to estimate the position of the receiver we use another particle filter. This position filter maintains the following state for each particle:

$\boldsymbol{X}$ the position of the receiver
$\boldsymbol{V}$ the velocity of the receiver
R_i the time that the last chirp was received from each beacon i ($0 \leq i < N$)
$\boldsymbol{O}_i$ the position of the receiver at time R_i for each beacon i

The particle filter is updated every time that a chirp is uniquely associated with one of the beacons. That is, chirps that may have come from multiple beacons are ignored, as are chirps that cannot be associated with any beacons. A chirp is associated with a beacon if it falls within a 4 ms window around the expected arrival time of that beacon.