Indoor localization methods and services

by

Gergely Zachár

Supervisor: Gyula Simon, PhD

DOI:10.18136/PE.2019.702

A thesis submitted to the University of Pannonia in fulfillment
of the requirements for the degree of
Doctor of Philosophy

University of Pannonia
2018

Indoor localization methods and services

Thesis for obtaining a PhD degree in the Doctoral School of Information Science of the University of Pannonia

in the branch of Information Sciences

Written by Gergely Zachár

Supervisor: Gyula Simon

propose acceptance (yes / no) ……………………….
(supervisor)

The PhD-candidate has achieved .... % in the comprehensive exam,
Veszprém, ……………………….
(Chairman of the Examination Committee)

As reviewer, I propose acceptance of the thesis:

Name of Reviewer: …...................... …................. yes / no
……………………….
(reviewer)

Name of Reviewer: …...................... …................. yes / no
……………………….
(reviewer)

The PhD-candidate has achieved …..........% at the public discussion.

Veszprém, ……………………….
(Chairman of the Committee)

The grade of the PhD Diploma …..................................... (…….. %)
Veszprém,

……………………….
(Chairman of UDHC)

Abstract

Object localization and tracking is a crucial element of various autonomous systems. While in outdoor localization mainly GPS-based services are used, indoor localization systems use other competitive solutions, based on e.g. ultrasound, Wi-Fi signals, graphic markers, RFID tags, and visible light. The goal of this thesis is to provide contributions to radio-interferometric and image-processing-based indoor localization systems.

In this thesis two novel radio-interferometric tracking algorithms are presented. In radio-interferometric localization systems infrastructure transceivers generate interferometric signals, the phase of which are measured by receiver nodes deployed in the infrastructure and on the followed object. The position or the track of the object is calculated from the phase measurements and the known positions of the infrastructure transceivers. The first proposed solution is a geometric based approach, utilizing robust phase unwrapping, with which a moving target can be tracked from a known initial position. The second solution is based on a novel confidence map, generated from the measurements and the theoretical phase distribution, and it can either track an object from a known initial position or determine the full track and the initial position of the object retroactively. The proposed methods provide centimeter-range accuracy when the speed of the object is low (<1 m/s in the current implementation). The theoretical results are analyzed and supported by simulations and real measurements.

To aid the measurement process in radio interferometric applications, a novel distributed and efficient method was proposed to measure the phase differences. The new solution provides more accurate phase difference measurements than the current solutions, even in case of various waveforms, amplitudes, and noisy conditions. The proposed solution is distributed such that both the computational demand on the low-end measurement devices and the communication bandwidth required to transmit measurement results are low.

In a different application domain (image-processing-based indoor localization systems), a novel method was proposed to create long-distance beaconing, using Visible Light Communication (VLC). Beacons transmit their own IDs using VLC with high enough frequency (>100 Hz) to avoid flickering effect. The novel solution uses undersampling, which allows the utilization of simple sensors (e.g. inexpensive digital cameras with frame rates as low as 30 fps). Since the decoding requires only a few pixels on the image, the beacons can be reliably detected even from long distances (up to 50 m).

Kivonat

Objektumok lokalizációja és nyomon követése számos autonóm rendszer kulcsfontosságú eleme. Amíg a kültéri lokalizációban elsősorban a GPS-alapú szolgáltatásokat használják, a beltéri lokalizációs rendszerek számos egyéb megoldást alkalmaznak, pl. ultrahang, Wi-Fi jelek, grafikus markerek, RFID címkék és látható fény alapú rendszerek. Az értekezés célja, hogy új megoldásokat adjon rádió-interferometrikus és képfeldolgozásra épülő beltéri lokalizációs rendszerek megvalósításához.

A dolgozat két új rádió-interferometrikus nyomkövetési algoritmust mutat be. A rádió-interferometrikus lokalizációs rendszereknél az infrastruktúra adók interferometrikus jeleket generálnak, amelyek fázisát az infrastruktúrában elhelyezett és a követett objektumra telepített vevőegységekkel mérjük. Az objektum pozícióját vagy útvonalát a fázismérésekből és az infrastruktúra-eszközök ismert pozícióiból számítjuk. Az első javasolt megoldás egy geometriai alapú megközelítés, amely egy új robusztus fáziskorrekciós eljárást alkalmaz, amellyel egy mozgó célpont egy ismert kezdeti pozícióból követhető. A második megoldás egy új konfidencia térképen alapul, amelyet a fázismérésekből és az elméleti fáziseloszlásból származtatok, és amelynek segítségével egy objektumot nyomon követhető egy ismert kezdeti pozícióból, vagy visszamenőlegesen meghatározható annak teljes útvonala és kiindulási pozíciója. A javasolt módszerek centiméteres pontosságot nyújtanak alacsony objektum-sebesség esetén (<1 m/s). Az elméleti eredményeket szimulációkkal és valós mérésekkel támasztottam alá.

A rádió-interferometrikus mérési eljárás támogatásához új, elosztott és hatékony módszert javasoltam fáziskülönbségek mérésére. Az új megoldás nagyobb pontosságot biztosít a jelenlegi megoldásoknál különféle előforduló hullámformák, amplitúdók és zajos körülmények esetén is. A javasolt megoldás elosztott számításokat alkalmaz, így az kis számítási teljesítményű mérőeszközökkel is megvalósítható, ráadásul igen alacsony kommunikációs sávszélességet igényel.

Egy másik alkalmazási területen (képfeldolgozás alapú beltéri lokalizációs rendszerek) új módszert javasolok nagy hatótávolságú jeladók megvalósítására. A jeladók látható fény alapú kommunikáció (VLC) segítségével magas villogási frekvenciával (>100Hz) továbbítják saját azonosítóikat, így elkerülve az emberi szem számára zavaró villódzást. Az új megoldás alulmintavételezést alkalmaz, amely lehetővé teszi költséghatékony (akár 30 képkocka / másodperc sebességű) kamerák használatát is. Mivel a dekódolás csak néhány képpontot igényel, a jeladók nagy távolságról is megbízhatóan észlelhetők (akár 50 m-ig).

Abstrakt

Objekt lokalisering og sporing er et viktig element i ulike autonome systemer. Mens i utendørs lokalisering hovedsakelig brukes GPS-baserte tjenester, innendørs lokaliseringssystemer fortrekker andre konkurransedyktige løsninger, basert på f.eks. ultralyd, Wi-Fi-signaler, grafiske markører, RFID-koder og synlig lys. Målet med denne avhandlingen er å levere bidrag til radio-interferometriske og bildebehandlingsbaserte innendørs lokaliseringssystemer.

I denne avhandlingen presenteres det to nye radio-interferometriske sporingsalgoritmer. I radio-interferometriske lokaliseringssystemer genererer infrastruktur-transceivere interferometriske signaler, fasen måles av mottakernoder i infrastrukturen og på det etterfølgende objektet. Objektets posisjon eller spor beregnes ut fra fasemålingene og de kjente posisjonene til infrastruktur enheter Den første foreslåtte løsningen er tilnærming basert på geometri, ved bruk av robust faseutpakking hvor et bevegelig mål kan spores fra en kjent startposisjon. Den andre løsningen er basert på et nytt selvtillitskart, generert fra målingene og den teoretiske fasedistribusjonen, og den kan enten spore et objekt fra en kjent startposisjon eller bestemme hele sporet og den opprinnelige posisjonen til objektet med tilbakevirkende kraft. De foreslåtte metodene gir nøyaktighet i centimeterområdet når objektets hastighet er lav (<1 m / s i dagens implementering). De teoretiske resultatene blir analysert og støttet av simuleringer og virkelige målinger.

For å støtte måleprosessen i radiointerferometriske applikasjoner, ble det foreslått en ny distribuert og effektiv metode for å måle faseforskjellene. Den nye løsningen gir mer nøyaktige faseforskjellmålinger enn de nåværende løsningene, selv om det er forskjellige bølgeformer, amplituder og støyende forhold. Den foreslåtte løsningen distribueres slik at både beregningsbehovet på måleinstrumentene og kommunikasjonsbåndbredden som kreves for å overføre målesultater, er lave.

I et annet applikasjonsdomene (bildebehandlingsbaserte innendørs lokaliseringssystemer) ble det foreslått en ny metode for å lage langdistanse beacons ved bruk av Visible Light Communication (VLC). Beacons sender sine egne ID-er ved hjelp av VLC med høy nok frekvens (> 100 Hz) for å unngå den flimrende effekten. Den nye løsningen ansetter underprøvetaking, som tillater bruk av enkle sensorer (for eks. billige digitale kameraer med opptaksfrekvens så lave som 30 fps). Siden dekodningen krever bare noen få piksler på bildet, kan beacons påvises pålitelig selv fra lange avstander (opptil 50 m).

INDEX

1. Introduction

Object localization and tracking is a crucial element of autonomous systems. While in outdoor localization mainly GPS-based services are used, indoor localization systems use other competitive solutions, based on e.g. ultrasound, Wi-Fi signals, graphic markers, RFID tags, and visible light. Localization systems contain several building blocks; at the lowest level physical measurement devices are present, producing physical parameters (e.g. signal strength, phase, time-of-flight, image frames, etc.). The next processing level provides higher-level and more abstract properties (e.g. distance, angle, relative position, etc.). At the highest level of the processing chain fusion and localization algorithms accumulates and fuse the data, and as a result, present the location of the tracked object.

This thesis provides contributions at all levels of the signal processing chain: at the lower levels (1) a new robust, accurate, and low complexity method will be presented to measure the phase-difference of the radio-interferometric signals, and (2) a new beaconing method will be proposed for optical-based localization systems. At the fusion level two new algorithms will be introduced to provide accurate and robust tracking utilizing radio-interferometric measurement architectures.

In Section 2 related work and current solutions are overviewed.

In Section 3 a novel radio-interferometric tracking application is proposed. Fixed radio transceiver nodes are used to generate interferometric signals, allowing the tracking of a moving object, equipped with a radio receiver. Low-frequency phase measurements of the interferometric signals are processed by the proposed Simultaneous Phase Unwrapping algorithm, and then the real-time position estimates are calculated. Simultaneous Phase Unwrapping is proposed as a solution of a newly identified phase unwrapping problem set; where the separate wrapped 1D phase signals are unwrapped utilizing a higher level relationship between them. The actual object location is calculated with a self-correcting algorithm based on an error surface, which provides a fault-tolerant and robust tracking method. The performance of the tracking system is analyzed by simulations and real measurements, indicating that the proposed method tolerates well substantial measurement phase noise and even sporadic large measurement errors.

In Section 4 another novel robust radio-interferometric object tracking method is proposed. The system utilizes the same radio-interferometric phase measurement process as the tracking solution presented in Section 3. However, from the measured phase values now a confidence map is computed, which is used to generate the track of the moving receivers. The proposed method can be operated in two modes: (1) the object path is determined based on the initial position and the successive measurements, (2) the object path is determined retroactively based on the progression of the confidence map. The proposed method also enhances the track estimation by an adaptive evaluation method of the confidence map, and provides more robust estimation by allowing temporary bad or missing measurements, which are tolerated by a prediction mechanism, extracted from the evolution of the confidence map in time. The performance of the proposed system is illustrated by simulations and real measurements

Phase difference measurement of interfering radio signals is an essential part of radio-interferometric positioning applications. In Section 5 a novel phase measurement system is proposed, which is able to accurately measure the phase difference of the received radio signals. The proposed solution contains a novel algorithm, which can measure the phase of the signals even when the waveforms are distorted, the amplitudes vary or the measurement noise is significant. The signal processing and the calculations of the phase difference are distributed; preprocessing of the signal is performed on the measurement devices with integer arithmetic and only a small amount of data is sent to the base station. The final steps of the calculation take place on the base station (usually a PC) utilizing floating point arithmetic. Thus the suggested measurement system can be operated on inexpensive devices with very limited computing capabilities (e.g. on 8-bit microcontrollers) and the precision of the phase estimation still can be high. The performance of the proposed system is illustrated with real measurements results.

In Section 6 the design of an LED-based beaconing infrastructure is introduced, which can be utilized in indoor localization systems. The beacons can be identified with regular cameras as detectors. The proposed solution detects and extracts the IDs of the beacons from a series of frames. The detected IDs and their positions can be used for localization purposes. The LED beacons are blinking with high frequency, invisible for human eyes, thus they can be part of the existing lighting infrastructure. The proposed coding system utilizes the undersampling of the signal: the sampling frequency of the cameras is much lower than the blinking frequency, thus the proposed system can be operated using inexpensive devices. The performance of the proposed system is analyzed and a possible application is introduced.

Section 7 summarizes the new results and provides an outlook for possible further research and applications.

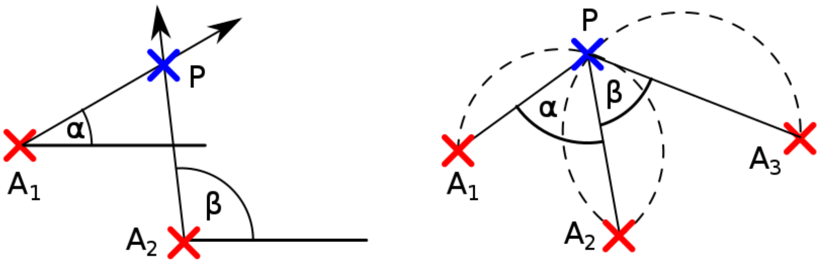
2. Background and previous work

This section provides a brief overview of the current localization and tracking solutions and also introduces the field of phase unwrapping, phase measurement, the fundamental radio-interferometric principles and measurement processes, which are necessary foundations of the proposed tracking algorithms and applications.

2.1. Object positioning and tracking principles

Positioning and tracking applications are utilizing several basic principles. These are mostly based on geometry, where the measured properties are angles and distances. Other solutions compare measured quintities to *a priori* information, mainly maps from previously collected measurements, or measure the relative movements of the object. In this chapter these principles will be briefly summarized. Note that these methods are presented in two dimensions, but their extension to 3D is trivial.

In several applications measured angles (Angle of Arrival, AoA) are available for localization (see Figure 1). The object position P can be estimated based on the measured angles (which are measured to a global reference direction) and on the known positions of the anchor points, as can be seen in Figure 1(a). The required calculations are trivial; the location estimate is at the intersection of the measured direction vectors. Object positioning is also possible if only the differences of the angles are known (see Figure 1(b)). In this scenario the difference between the measured angles are inscribed angles in circles. With multiple circles the position can be estimated as the intersection of the circles.

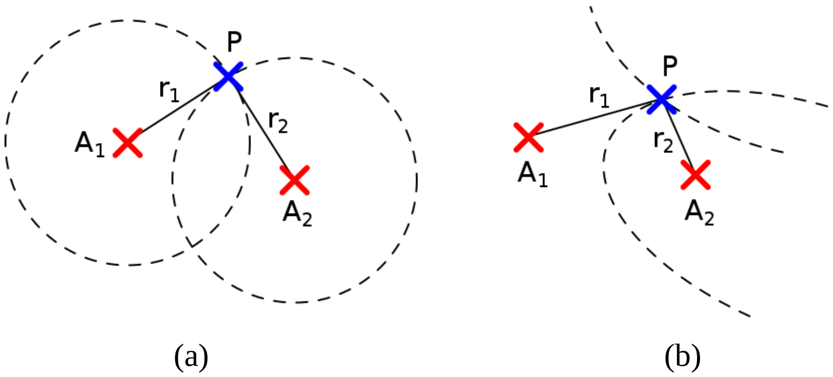


(a) (b)

Figure 1. Location estimation utilizing the measured angle (Angle of Arrival, AoA): (a) object position P (blue mark) is estimated based on the known positions of the anchor points A_1, A_2 (red marks) and the measured angles α, β, (b) object position is estimated based on the known positions of the anchor points and the measured angle differences α, β as the inscribed angles of circles C_1, C_2.

In several localization applications the range between the anchor points and the tracked object can be used. The absolute distance can be measured with a Time of Flight (ToF, or

Time of Arrival, ToA) approach, which utilizes the fact that the distance and the required time for a signal to travel this distance thought a medium are proportional. If the distances between the object and the anchor points with known locations are known then the localization is trivial, as can be seen in Figure 2. The object position can be estimated as the intersection of the circles with radii equal to the measured distances. If only the Time Difference of Arrival (TDoA) can be measured, the position of the object will be on a hyperbola, as can be seen in Figure 2(b). With multiple measurements the exact location can be calculated as the intersection of the hyperbolas.



(a) (b)

Figure 2. Location estimation utilizing the measured range: (a) Time of Flight (ToF) location estimation of an object P (blue mark) based on the measured absolute ranges and the known positions of anchor points A_1, A_2 (red marks), (b) Time Difference of Arrival object location estimation based on the measured range differences and the known positions of the anchor points.

The fingerprinting localization method, which is presented in Figure 3, requires *a priori* information for the object location estimation. Before the positioning, a reference map is created with several measurements in known positions. When the object is localized, the measurement results are compared to the reference map; the object position is estimated as the location where the correlation between the measurements and the reference map are the highest. Note that this localization solution can be used with several different types of measured parameters, e.g. magnetic field strength, received radio signal strength, etc.

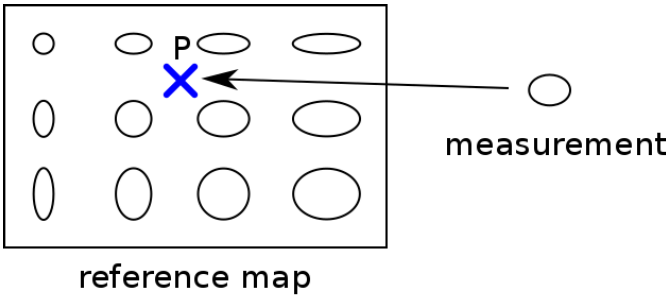


Figure 3. Location estimation utilizing a reference map (fingerprinting): object position P is estimated based on the measured values and the reference map. The estimated position is the point where the correlation between the map and the measurements are the highest.

Localization can be done utilizing only the measured movement parameters of the object (see Figure 4). In this scenario the parameters of consecutive movements are measured (e.g. using rotary encoders) or are estimated (e.g. using accelerometers, gyroscopes). For an exact location the origin of the object path should be known. Note that the localization error based on each movement is accumulating, thus for accurate positioning the exact object location should periodically be corrected with other methods.

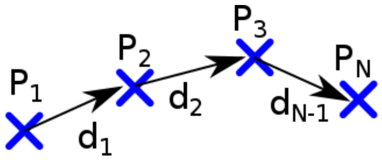


Figure 4. Location estimation utilizing the measured consecutive displacements: for the exact location of the object the origin of the path should be known

2.2. Object positioning applications

While in outdoor positioning the most dominant method is GPS, for indoor positioning several alternative methods have been proposed. In this section the most prevalent methods will be reviewed, through a representative application.

Acoustic (most commonly ultrasound) solutions calculate the pairwise distances based on the measured time-difference-of-flight of the emitted pulses, and the known position of the beacons. The system proposed in [1] utilizes several devices on known positions with microphones to locate the object equipped with a loudspeaker. The proposed solution records the time instances on each device when the signal is arrived, which will be proportional to the distances. Since the clocks of the receivers are synchronized, the position of the source can be calculated based on the known positions of the receivers using the TDoA method.

Radio waves are commonly used for indoor localization. The most common methods are using a signal-strength based approach, and can be categorized into two classes. The methods corresponding to the first class utilize the mathematical model of the propagation of the radio waves. The sources of the signal are at known positions and the receiver measure the received strength (in some cases also the phase) of the signal. Based on the model the range between the source and the receiver can be approximated. The localization principle is mathematically equivalent to the ToF principle. Note that the drawback of this approach is that the environment has heavy influence on the signal strength, resulting in limited ranging accuracy, and thus modest localization accuracy. [2]

The methods of the second class of the signal strength based solutions utilize a priori information. Before the actual localization a statistical reference map is created with several measurements in known locations. The localization itself is based on the fingerprinting principle. The drawback of this approach is that any environmental change (e.g. moving the furniture) may corrupt the reference map. [3]

Great advantage of radio RSSI methods is that existing infrastructure can be used as beacons e.g. WiFi routers [4].

Localization is also possible with measuring the time-of-flight of the radio-waves. Solutions presented in [5], [6] measure the required time to travel a signal between two radios. The accuracy can be increase with the combined measurements of the time and phase. The main drawback of the radio-wave ToF method was the special, expensive, high speed measurement hardware. In the past few years, however, inexpensive commercial measurement devices have become available [7].

Economic beacon-free solutions can be provided e.g. for smart phones using image processing techniques: room-level localization was proposed using simple graphical markers [8]. The environment itself was used as a generalized marker in [9]: a picture taken by the camera was compared to pictures stored in a data-base, containing a large set of reference images, to detect the current camera location.

Localization can be performed with measuring or calculating the direction and length of the movements. The system presented in [10] is a pedestrian dead reckoning solution, where inertial sensors utilized for short range positioning. Note that, due to the accumulation of the localization error, these systems utilize other localization methods to periodically correct these deviations. In [10] a combined approach was used, where WiFi signal strength measurements were utilized to decrease the accumulating error of the dead reckoning.

Light based approaches utilize LEDs as anchors, which can be installed as a part of the lighting infrastructure. The ranging is performed either by light intensity measurements [11] or using image processing techniques [12] to estimate the position of a camera. In these systems visible light communication (VLC) is utilized to transmit the identifier of beacons.

A completely new approach utilizes radio interferometry, which provides measurement results that can be considered as generalized distances [13], [14]. Although the radio interferometric approach provides potentially high accuracy localization, due to its novelty its properties and potential application fields have not yet been investigated thoroughly. The main part of this dissertation contributes to this research area.

2.3. Interferometry in measurement systems

Measurement systems using optical interferometry are utilized in a wide range of applications, from precision manufacturing to medical applications, to measure small displacements with high accuracy. Such measurement systems produce interferometric signals, where the phase change corresponds to the displacement to be measured. Recently in [15] a homodyne measurement system to characterize piezoelectric actuators has been proposed, while in [16] a heterodyne interferometric measurement system has been introduced for semiconductor manufacturing. For surface contouring laser diode speckle interferometer was proposed in [17]. In medicine, chest wall displacement was measured using laser interferometry [18]. For speed measurements from larger distances (up to 10m) a self-mixing interferometer with frequency domain signal processing was proposed in [19].

Interferometry is widely used in radio astronomy, where radio emissions from distant celestial objects are observed. The angular resolution of conventional radio antennas is greatly increased by synthesis telescopes, where signals from arrays of telescopes are combined [20]. Radio-interferometry is also utilized in low-distance telemetry: high precision distance measurements up to 50m using frequency-scanning interferometry was proposed in [21], while dispersive interferometry was successfully used for measurements of sub-millimeter distances [22].

In [23] a measurement method was proposed, which can determine the speed profile of a radio transmitter, moving along a known linear trajectory. The measurement system utilizes one moving (tracked) transmitter, emitting a carrier wave, and two fixed receivers, which produce the interference signal used to determine the actual speed of the transmitter. More general architectures were proposed for radio-interferometric positioning and tracking, where multiple transmitters and receivers were utilized to determine the location of an object (a receiver, in these cases) in two or three dimensions. The foundation of these methods was laid in [13], which will be detailed in Sections 2.3 and 2.4.

2.4. Radio-interferometric localization

Instead of using high frequency signal processing, Radio-Interferometric Positioning [13] processes low frequency interference signals, generated by radio-interferometry. In the measurement setup, shown in Figure 5, four nodes, referred to as *quad*, are used. Two nodes

(A and B) are used as transmitters to create an interference, while two receivers (C and D) measure the interference signal, as shown in Figure 5. The transmitter frequencies are approximately the same $(f_A \approx f_B \approx f)$ thus the interference signal has a low frequency envelope, with beat frequency $\Delta f = |f_A - f_B|$.

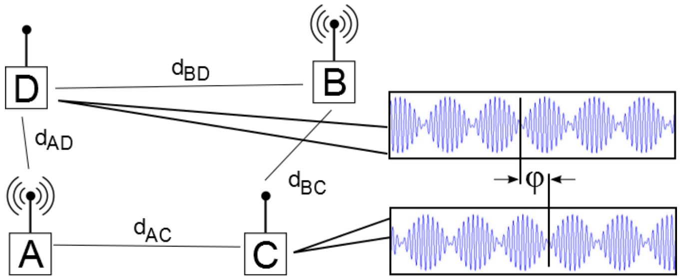


Figure 5. Radio-interferometric phase measurement

The envelope of the interference signal is sensed as the RSSI at the receivers C and D, which measure the phase difference of the detected RSSI signals, which depends on the relative locations (called quad-range) d_{ABCD} of the quad elements, as follows [13]:

$$\varphi(f) = 2\pi \frac{d_{ABCD}}{\lambda} \ (mod\ 2\pi), \tag{1}$$

where $\lambda = c/f$ is the wavelength of the signal, c is the speed of light, and quad-range d_{ABCD} is defined as follows:

$$d_{ABCD} = d_{AD} - d_{BD} + d_{BC} - d_{AC}, \tag{2}$$

where d_{XY} is the Euclidean distance between nodes X and Y, as shown in Figure 5.

The modulo operation in (1) is the main challenge in radio-interferometric localization: the exact quad-ranges cannot be computed from one measurement $\varphi(f)$. In [13] the ambiguity problem was solved by using multiple carrier frequencies f and thus multiple $\varphi(f)$ values, but this solution requires large amount of computations and is sensitive to phase errors.

In the radio-interferometric measurement process there are several error sources which can affect the accuracy of the localization. These sources are identified and discussed in details in [13]. The error sources are the following:

Carrier frequency inaccuracy: the two transmitters emit slightly different (Δf) carrier frequencies, which affect the phase estimation and the d_{ABCD} distance calculation. However this Δf frequency deviation is several order smaller than the carrier frequency, thus represents a negligible error source in real application scenarios.

Carrier frequency drift and phase noise: the interferometric measurement process relies on the fact that the carrier frequencies, and thus the interference, is stable. This stability can be achieved with short measurement times and by minimizing the chance of abrupt thermal changes and mechanical shocks.

Multipath effects: these effects can cause disruption in the interference signal, which results in erroneous or missing phase measurements. Such measurement errors result in significant localization error. In the literature so far there has been no solution proposed to mitigate this effect. The contributions presented in Sections 3 and 4 provide novel robust and fault tolerant estimation methods.

Antenna orientation: theoretically orientation and antenna shape can be a source of a negligible error, but empirically this was not verified. However the antenna radiation pattern and the orientation can drastically modify the interference signal amplitude, thus lowering the signal-to-noise ratio.

RSSI measurement delay jitter: this jitter introduces a relative phase offset error, however in real measurements the effect is not noticeable.

RSSI signal-to-noise Ratio: the signal-to-noise ratio of the recorded RSSI values depend on the distances of the devices and also on other physical properties, e.g. antenna orientation, receiver hardware.

Signal processing error: this type of error highly depends on the utilized phase estimation algorithm. In Section 5 a distributed phase estimation method is proposed, which can achieve good accuracy, despite the low computation cost and low resource requirements.

Time synchronization error: time synchronization is a crucial point for the phase difference measurement. It can be achieved with appropriate time stamping of the control messages sent to the measurement devices. A potentially more accurate method is presented in Section 5.2.3 which utilizes the interferometric signal itself for synchronization.

In Sections 3 and 4, novel radio-interferometric tracking solutions will be proposed. The new methods provide *tracking,* instead of solving the localization problem, with increased speed and robustness.

2.5. Phase difference measurement

Phase difference measurements are essential in radio-interferometric localization and tracking. In such applications two radio transmitters generate the interference signal (see

Figure 5). The interference signal has a low-frequency envelope (the frequency of the beat equals to the frequency difference of the two transmitters). The signal envelope is in fact the RSSI signal, which can be measured in most radios. Radio-interferometric localization algorithms measure the RSSI phase difference between pairs of receivers, and from these measurements the location of a tracked receiver can be computed. The fundamental measurement is illustrated in Figure 5, where two receivers A and B simultaneously measure the interference signal and from the time delay between the signal envelopes the phase difference is computed. Notice that in this application not the phase of the high frequency carrier wave is measured, but rather the phase of the low frequency envelopes, which allows the utilization of low-cost devices.

The challenges of phase measurement in the target application field are the following:

a) the measured signal is highly variable (e.g. due to fading, or movements around the devices),
b) measurement devices are wireless, with no wired link anywhere in the system,
c) the measurement devices must be small and inexpensive, only one central node has high computing capability,
d) the measurement must be as fast as possible, to allow frequent measurements and thus good quality tracking,
e) the transmitted measurements must be of small size, due to the rather limited communication bandwidth,
f) the measurements must be continuously repeated as fast as possible with different pairs of receivers.

Due to the above limitations raw measurements cannot be transmitted, thus the signal processing must be performed in place, on the measurement devices. A low complexity processing algorithm is required, which can efficiently be executed on the low-end devices. Since in real applications the measurements involve several quads, the measurements must be scheduled in time, and the schedule must be as tight as possible, in order to provide high responsiveness of the system.

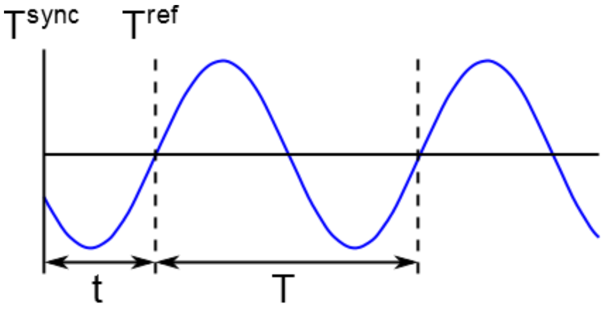


Figure 6. Definition of phase and a possible measurement method of the phase of a signal.

Frequency and phase measurement of periodic signals is a widely studied field. The classical definition of phase can be seen in Figure 6, which also illustrates a possible measurement method. If t and T are measured, where T is the period length and t is the time between a time synchronization point T^{sync} and a signal reference point T^{ref} (in the example of Figure 6 the signal reference point is defined as the time instant when the rising edge of the signal crosses zero), then the phase is defined as

$$\varphi = 2\pi\frac{t}{T} \tag{3}$$

Such simple measurements are often used when the signal-to-noise ratio is high and the accuracy requirements are modest, e.g. in [13]. For higher accuracy, other time domain methods also exist in large number. The method proposed in [24] uses digital sampling of high resolution voltmeters to provide accurate harmonic and phase estimation of two low-noise, low-frequency signals, with primary application of power line measurement. In [25] least-square fitting is analyzed for the amplitude and phase estimation of periodic components of a signal. In [26] digital filters are used to suppress higher harmonics to provide accuracy below 1mrad.

A large range of phase measurement techniques use frequency domain. For spectral estimation the Fast Fourier Transform (FFT) is widely used (e.g. [27]). Several enhancements were proposed to decrease the effect of leakage and thus increase accuracy, e.g. windowing [27], synchronous or quasi-synchronous sampling [28], iterative methods [29], just to name a few. An alternative method, which reduces phase error, and the computational complexity is similar to that of the FFT, was proposed in [30]. Resonator-based digital filter banks [31] and PLL-like adaptive filter banks [32] were also proposed for real-time and high accuracy measurements of periodic signals.

The above mentioned solutions focus on the accuracy of the phase and frequency measurement in contrast to the computation cost on the device. The target application requires low complexity algorithms and fast measurement on the devices, meanwhile enables post-processing on the host computer (distributed computation). Phase measurement methods in the frequency domain require transformation from the time domain, which is a computationally intensive task, while filter banks require several periods to settle to the correct phase and frequency, causing longer signal sampling periods. Time domain, statistical solutions (e.g. correlation, least-square fitting) also require excessive computation. Note, that most of the simple solutions (e.g. the zero-crossing detection demonstrated on Figure 6) are also not capable of producing appropriate phase estimation, which can be used for phase difference measurement: on different devices the measured signals can be distorted in different ways, which causes different, non-comparable reference points (see e.g. Figure 58 for a real measurement).

In Section 5 a novel measurement setup will be proposed, which can accurately estimate the phase difference between signal envelops, measured on two devices, even in the presence of noise. The proposed distributed computation method can be executed on simple devices with low computational complexity and without floating point operations, and requires very low communication bandwidth. The system also includes support for fast measurement scheduling.

2.6. Phase unwrapping techniques

Typical application area of one dimensional phase unwrapping is when the full phase of a signal spectrum is required. Such requirement arises in a wide range of signal processing problems, from speech processing and synthesis to EEG processing.

Let $X(x): \mathbb{R} \to \mathbb{C}$ be a continuously real-differentiable complex valued function, the phase of which is denoted by $\phi(X): \mathbb{C} \to \mathbb{R}$. The phase can be calculated as

$$\phi(X) = atan2\big(Im(X), Re(X)\big) + 2k\pi, \tag{4}$$

where $atan2$ is the extended arctan function, k is an arbitrary integer, and $Re(X)$ and $Im(X)$ are the real and imaginary parts of X, respectively. Due to the phase ambiguity in (4), the phase value is often represented by its *wrapped* or *principal value* φ, which is restricted to a predefined 2π-long range, e.g. $(-\pi, \pi]$, thus $\varphi(X): \mathbb{C} \to (-\pi, \pi]$, or $\varphi(x): \mathbb{R} \to (-\pi, \pi]$. The calculation of the true phase value ϕ from the principal value φ is called unwrapping, defined by the unwrapping integral. For this purpose let us define unwrappable intervals. The interval $\Gamma_i = [x_i, x_{i+1})$ is called unwrappable if $\forall x \in \Gamma_i, \varphi(X(x)) \neq \pi$. Then the unwrapping integral is defined as

$$\phi(x_k) = \sum_{i=0}^{k-1} \int_{\Gamma_i} \nabla\varphi(X(x))dx, \tag{5}$$

where $\nabla\varphi$ is the gradient of φ and $\phi(x_0) = 0$ by definition. Notice that the unwrapping process is performed piecewise through the unwrappable intervals Γ_i to produce the full integration interval $\Gamma = \sum_{i=0}^{k-1} \Gamma_i$.

In practical cases $\varphi(x)$ is known at discrete grid $x(k), k = 1..N$; and the true phase values are searched for in the same grid. Let us denote the discrete phase samples by $\varphi_k = \varphi\big(x(k)\big)$ and $\phi_k = \phi\big(x(k)\big)$. Assuming that Shannon's sampling theorem is fulfilled for the sampling grid, i.e. $\forall k = 1..N-1\colon |\phi_{k+1} - \phi_k| < \pi$, discontinuities in the principal phase record mean transitions between unwrappable intervals. Thus a straightforward unwrapping method can be performed by Detecting and Correcting Discontinuities (DCD) in the principal phase record, as follows [33]:

First the gradient of φ_k is calculated:

$$\begin{aligned} &\nabla\varphi_1 = 0, \\ &\nabla\varphi_k = \varphi_k - \varphi_{k-1}, \quad \text{for } k = 2..N, \end{aligned} \tag{6}$$

then the correction terms Λ_k are determined:

$$\begin{array}{ll} \Lambda_k = 0, & \text{if } \nabla\varphi_k \in (-\pi,\pi] \\ \Lambda_k = i2\pi, \text{s.t.} \nabla\varphi_k + \Lambda_k \in (-\pi,\pi] & \text{otherwise,} \end{array} \tag{7}$$

where i is an integer value. The unwrapped phase is then calculated for $k = 1..N$, as follows:

$$\phi_k = \varphi_k + \sum_{i=1}^{k} \Lambda_i. \tag{8}$$

The applicability of DCD method is universal where sampled principal phase values are available. However, the DCD method assumes that the samples are "dense" enough, .i.e. the sampling theorem is fulfilled. This is not always the case, which is addressed by the Adaptive Numerical Integration (ANI) method [34].

The ANI method calculates the unwrapped phase values similarly to DCD, but ensures that the principal value samples are dense enough, using a discrete approximation (trapezoidal integration formula) of the unwrapping integral of (5) for checking. If the calculated integral is inconsistent with the measured phase values (i.e. the phase gradient changes too much between two points of the grid), the adaptive mechanism doubles the density of the grid, and thus halves the integration step-size. The adaptation mechanism of ANI increases the sampling frequency until the principal phase values are provided with sufficient density. In the typical application (cepstrum calculation) of ANI the change of sampling frequency is possible, since the data points are generated by FFT, the width of which can be adaptively changed. However, in several applications (including radio-interferometric tracking) the principal phase values are generated by a measurement process with predefined sampling frequency, where there is no way to generate more measurement points on demand.

Other approaches were proposed in both the time and frequency domain. For applications where the data can be efficiently modelled as an output of an autoregressive moving average (ARMA) process, the Polynomial Factoring method can be applied. This method first identifies the process, i.e. calculates the roots of the model polynomials. Once the model is known, the exact spectrum (including the phase) can be computed on the unit circle [35]. The correcting factor Λ can directly be determined from the time domain data as well. Using Sturm's Theorem, the integer value i in (7) can be calculated. The method, proposed in [36], uses Sturm polynomials to calculate Λ at any frequency.

Imaging applications, such as Synthetic Aperture Radar [20], Magnetic Resonance Imaging [37], or profilometry [38], utilize two or three-dimensional (from now on n-D) phase maps, where the unwrapping problem is analogous to the 1-dimensional case: the input phase map contains the principal phase values on an n-D grid and the goal is to provide the unwrapped phase values. The function $X(x)$ can now be defined as $X(x): \mathbb{R}^n \to \mathbb{C}$. Instead of unwrappable intervals, Γ_i now represent unwrappable *paths* in the domain of x. Formally, the unwrapping process is performed again by the unwrapping integral (5): from a reference point x_0, using (5), the full phase of any other point x_1 can be calculated. A major difference between the 1-D and n-D case is that now there are *multiple choices* to reach x_1 from x_0 (since several possible paths Γ exist between points x_0 and x_1). If there are no phase singularities [37] present (caused either by noise or natural discontinuities in the data, e.g. an outline of a hill in front of the background in an SAR image), the evaluation of (5) along any of the possible paths Γ provides the same result for $\phi(x_1)$. However, in the presence of phase singularities, different paths may provide different results. The core problem of n-D phase unwrapping is the correct choice of the integration path so that the result be meaningful.

In n-D domain several approaches are used to identify and avoid phase singularities during phase integration, including least-squares methods [39], brunch-cut algorithms [40], pattern analysis [41], or optimization methods based on flow cost minimization [42].

In radio-interferometric tracking, phase measurements are given above a two-dimensional plane, but these measurement values are known only along the object trajectory (as opposed to full 2-D representations, as in SAR), thus the problem is inherently one dimensional. Since multiple measurements are performed at the same time (each measurement corresponding to a quad), 1-D unwrapping must be performed for each measurement sequence. Notice, that phase measurements of the different quads are not independent, since they belong to the same object trajectory, thus this piece of information may be utilized to increase the accuracy. The simultaneous phase unwrapping problem (which will be addressed in Section 3.2) is a new class of the unwrapping problems, which, to the best of my knowledge, has not been discussed in the literature yet.

3. Radio-interferometric tracking based on phase-unwraping

General localization techniques are able to provide position information of the localized object in every time instant. When *a priori* information is available on earlier positions of the object, tracking solutions can be used. This chapter will discuss solutions related to tracking, e.g. the initial position of the object is assumed to be known and the movement of the object is tracked in time by successive measurements.

The system uses fixed reference nodes (transmitters and receivers) and arbitrary number of tracked receiver nodes. For sake of simplicity I will discuss only the case of one tracked node, but the proposed solution can trivially be extended to any number of tracked devices.

3.1. Basic hyperboloid-based tracking

In this section I will discuss the Basic Hyperboloid-based Tracking (BHT) algorithm in one dimension, then I will extend the method to higher dimensions to provide a location estimation, and finally I will introduce a location estimator using redundant measurements, which provides better accuracy and robustness against phase measurement errors and distortions.

3.1.1. Tracking in one dimension

For tracking in one dimension, two fixed transmitter nodes (A and B) and one fixed receiver node C is used, while the tracked receiver is denoted by D (see Figure 5). The transmitters utilize a single carrier frequency (as opposed to multiple frequencies in [13]) and the receivers continuously monitor the phase difference ϑ. The quad-range d_{ABCD} is calculated. According to (2), the solutions for the position of moving node D are located on a set of hyperboloids (hyperbolas in 2D, as illustrated in Figure 7). When a ϑ value is measured, the associated quad-range d_{ABCD} determines the possible positions of D. In Figure 7(a) for illustration purposes hyperbolas corresponding to $2k\pi$ phase values are shown. Note that from one measurement there is no way to determine on which of the possible hyperboloids D is located (i.e. k is unknown), and even if the particular hyperboloid was knew, it is impossible to tell where on the hyperboloid D is exactly located.

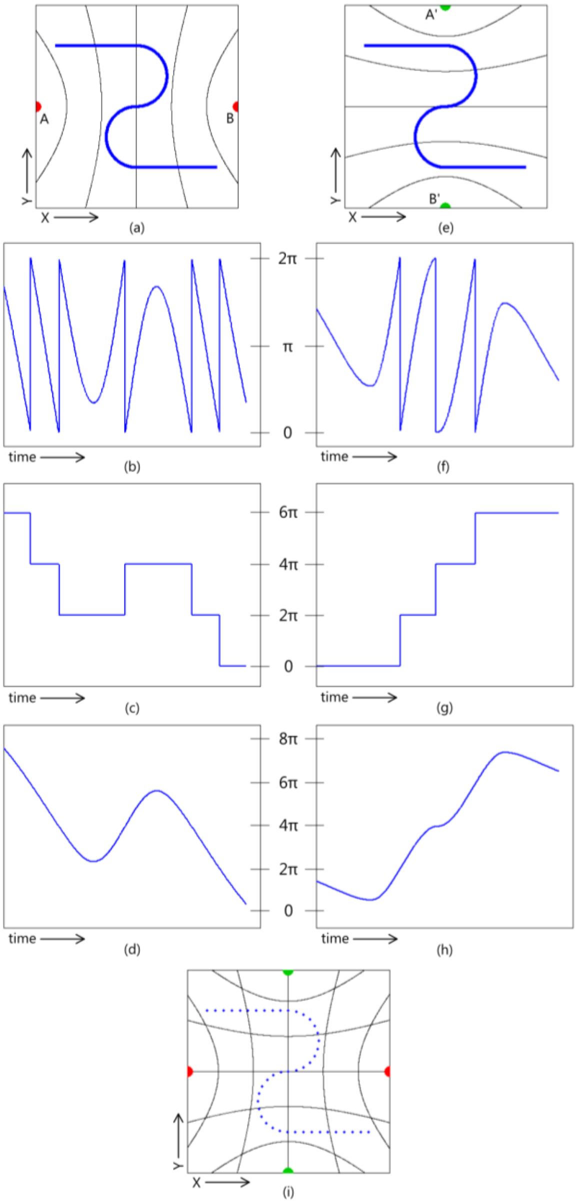


Figure 7. Basic radio-interferometric tracking with fixed transmitters A, B, A', B', fixed receiver C=C', positioned at the center (not shown for clarity), and moving node D. (a) The generated phase pattern by transmitters A and B: hiperbolas corresponding to $\vartheta = 2k\pi$ phase values are shown. The trajectory of the moving object is shown by thick line, the origin is assumed to be known, (b) the measured phase difference values between nodes C and D, (c) phase correction values, (d) unwrapped phase values, (e)-(h) independent measurement with transmitters A' and B', (i) tracking using the phase esimates of the two measurements. The trajectory estimate is shown by the dotted line.

Assuming that the original location of D is known and the motion of the object is relatively slow, i.e. the successive phase measurements have a difference smaller than π then the actual hyperboloid can be selected, based on the *expected continuity* assumption, using the estimated hyperboloid at the previous time instant, as follows. If the known phase in time instant k is φ_k and the measurement in time instant $k+1$ results the ambiguous phase values $\varphi_{k+1}(i) = \vartheta_{k+1} + 2\pi i$, then the closest possible value to φ_k is selected:

$$i = argmin\,|\vartheta_{k+1} + 2\pi i - \varphi_k| \tag{9}$$

Note that the underlying problem here is in fact the well-known phase-unwrapping problem (see Section 2.6). In Figure 7(b) the phase measurements are shown in the range of $(0 \dots 2\pi)$, while Figure 7(c) shows the phase correction values, according to (9), resulting the unwrapped phase measurement in Figure 7(d).

Knowing the exact phase value, the actual hyperboloid surface can be identified, using (1) and (2). Thus with a series of measurements the hyperboloid surfaces, on which the moving node D is located, can be tracked. However, the exact location of D on the hyperboloid is not known yet: for this purpose multiple independent quad-ranges will be used.

3.1.2. Tracking in higher dimensions

If in addition to node triplet A,B,C another node triplet A’,B’,C’ is used then two quad-ranges $d_{ABCD}$and $d_{A'B'C'D}$ can be measured, as shown in Figure 7(a) and Figure 7(e). Note that node triples A, B, C and A’, B’, C’ may be composed of physically different nodes or alternatively may contain the same nodes but in different roles: e.g. a possible solution is A’=A, B’=C, C’=B.

Using two quad-ranges, two independent one-dimensional tracking can be simultaneously performed, as described in the previous section. In each time instant each tracking identifies a hyperboloid surface on which node D is located. In Figure 7 the left and right columns represent the two independent trackings. Naturally node D is located in the intersection of the two hyperboloids, thus if the node is moving on a plane (in 2D), the node’s location can be calculated as the intersection of the two hyperbola, as illustrated in Figure 7(i).

For 3D localization three independent quad-ranges are necessary, for which at least four fixed devices must be used, out of which three independent triplets can be chosen.

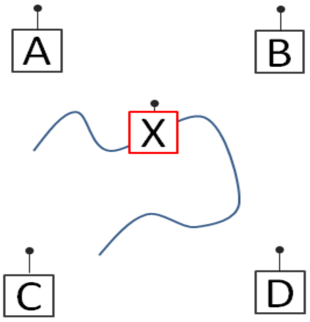
The tracking infrastructure required for the tracking will be discussed in Section 3.1.3, while the utilized phase difference measurement solution will be discussed in details in Section 5.

3.1.3. Tracking infrastructure

For measuring several independent quad-ranges an appropriate tracking infrastructure is required. The infrastructure contains transceivers at known positions, which can either play the role of transmitters to generate interference signals at the receivers, or receivers to allow phase difference measurements, as described in Section 2.4. The tracked node is always a receiver. Infrastructure nodes alter their roles, thus different interference signals can be generated. The measurements are scheduled using TDMA: different time slices are allocated for each quad-range.

A simple measurement uses three infrastructure nodes (two transmitters and one receiver) and the tracked receiver node, in a measurement configuration. The four nodes in the *configuration* can measure a phase difference value ϑ, which depends on the positions of both the infrastructure nodes and the tracked node. Such simple measurements are carried out with different configurations, to provide a measurement round, containing C simple measurements. The measurement results of a complete round will be used as inputs in each step of the tracking algorithm.

Figure 8 illustrates a scenario with four fixed nodes A, B, C, D, and one tracked node X. In this case $C_{max} = 12$ possible configurations exist, as shown in the table of Figure 8.



Config	A	B	C	D	X
C1	T	T	R		R
C2	T	R	T		R
C3	T	R		T	R
C4	R	T	T		R
C5	R	T		T	R
C6	R		T	T	R
C7	T	T		R	R
…					
C12		R	T	T	R

Figure 8. An example tracking infrastructure with for fixed nodes (A, B, C, D) and one tracked node (X). The possible configurations are listed in the table.

3.1.4. Utilizing redundant phase measurements

The object tracking method, introduced in Section 3.1.2, calculates the object location as the intersection of two hyperbolas. This solution, however, is very sensitive to phase unwrapping errors, as illustrated in Figure 9. The robustness of the method can be increased by utilizing redundancy, namely with more quad-range measurements: the over-determined system is more resistant to bad or missing measurement values.

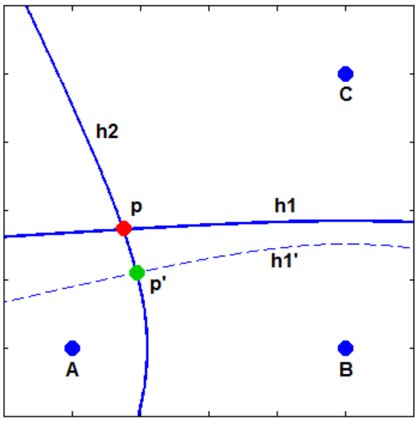


Figure 9. Estimating an object position (*p*) with two hyperbolas *h1* and *h2*, generated by the transmitter pairs of *B*, *C* and *A*, *B*, respectively. Hyperbola *h1'* is the result of an incorrect phase estimation resulting incorrect position estimate *p'*.

In my proposed solution multiple hyperbolas are used. For the position estimation with multiple hyperbolas the following error function ε is specified for a given point (x, y):

$$\varepsilon(x, y) = \sum_{i=1}^{n} d_i^2(x, y) \tag{10}$$

where $d_i(x, y)$ is the distance between point (x, y) and hyperbola i.

The distance $d_i(x, y)$ is the minimum Euclidean distance between the given point (x, y) and the hyperbola h_i, formally defined as $d_i(x, y) = min_{(x_{h_i}, y_{h_i})}\left(\sqrt{(x - x_{h_i})^2 + \left(y - y_{h_i}\right)^2}\right)$, where x_{h_i}, y_{h_i} are the points of the corresponding hyperbola.

The optimal estimated position $(\hat{x}, \hat{y})$ is located where the error function ε has its minimum (Figure 10):

$$(\hat{x}, \hat{y}) = \operatorname{argmin}_{x,y} \varepsilon(x, y) \tag{11}$$

Note that in the proposed BHT method neither the redundant phase measurements nor the utilized error function can detect or correct the $2k\pi$ unwraping phase errors introduced by erroneous or missing measurements. Thus this solution may produce large bias in the presence of bad quality measurements. In the next section a novel phase unwrapping technique will be proposed, which is robust against unwrapping errors.

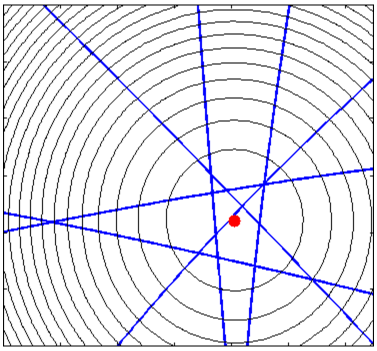


Figure 10. Using multiple hyperbolas the localization is performed by a minimum search on an error surface.

3.2. Simultaneous Phase Unwrapping

The goal of the proposed Simultaneous Phase Unwrapping (SPU) is to enhance the robustness and accuracy of the unwrapping process, applied to the measured (wrapped) phase signals. Other methods, which have extra information on the phase signal, successfully apply signal models and thus make the estimator more robust, e.g. in [35], [43]. In our application such knowledge on the individual phase measurements is not available, but another type of constraints can be applied here as well: the phase signals measured in parallel reflect the same (but unknown) location of the tracked object.

The Simultaneous Phase Unwrapping problem in 1-D can be formalized as follows. Discrete time series of wrapped phase values are determined from M separate measurement channels, where $\varphi_{m,k}$ denotes the principal phase measurement from measurement channel $m, m = 1, 2, \dots, M$, at time instant k. The set of past measurements $\varphi_{m,1}, \varphi_{m,2}, \dots, \varphi_{m,k}$, for $m = 1, 2, \dots, M$, are given, and a cost function $F(\phi_{1,k}, \phi_{2,k}, \dots, \phi_{M,k})$ is applied, which measures the consistency of the full phase values. The full phase values $\phi_{m,k}$ are determined by minimizing $F(.)$, such that $\phi_{m,k} = \varphi_{m,k} + \Lambda_{\mathrm{m},k}$, where correction parameters $\Lambda_{\mathrm{m},k}$ are integer multiples of 2π. The cost function $F(.)$ is application specific and may also be influenced by the utilized optimization method. For radio-interferometric tracking a special cost function will be defined in Section 3.3.

The block diagram of the system is shown in Figure 11. The individual one-dimensional wrapped phase sequences $\varphi_{m,k}$ are fed to DCD unwrapping blocks, where the unwrapping procedure in each channel is performed according to (6)-(8). Note that the value of $\phi_{m,k}$ depends on measurements $\varphi_{m,1}, \varphi_{m,2}, \dots, \varphi_{m,k}$ and state variables $\mathit{\Lambda}_{m,1}, \mathit{\Lambda}_{m,2}, \dots, \mathit{\Lambda}_{m,k}$. After the initial unwrapping by the DCD blocks, at each time instant k, based on the actual cost function, the optimizer block readjusts the last correction parameters as $\mathit{\Lambda}_{m,k} = i_m 2\pi$, where

i_m is an integer. The optimizer tunes parameters $\Lambda_{m,k}$, $m = 1, 2, \dots, M$, until it finds the smallest cost function:

$$(\Lambda_{1,k}, \Lambda_{2,k}, \dots, \Lambda_{M,k}) = \operatorname{argmin}_{\Lambda_{m,k}, \forall m} F\left(\phi_{1,k}(\Lambda_{1,k}), \phi_{2,k}(\Lambda_{2,k}), \dots, \phi_{m,k}(\Lambda_{M,k})\right). \quad (12)$$

For radio-interferometric tracking a fast and approximate solution of (12) will be proposed in the next section.

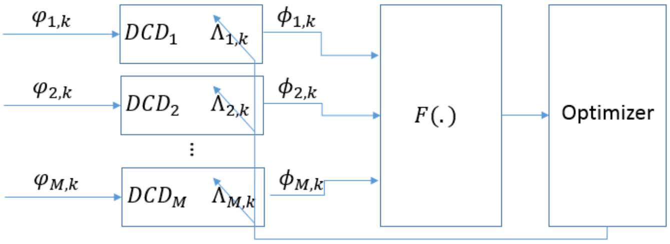


Figure 11. Block diagram of the phase unwrapping system

3.3. Fault-tolerant hyperboloid-based tracking

The BHT algorithm in Section 3.1 utilizes full phase values, thus the quality of the position estimator greatly affected by potential unwrapping errors. According to the sampling theorem, if the phase difference between consecutive phase samples $\varphi_{m,k-1}$ and $\varphi_{m,k}$ is smaller than π then the unwrapping can be performed unambiguously (see Section 3.1.1). However, in practical applications several error sources may be present, which can cause the violation of the sampling theorem:

Phase measurement errors: The distribution of the measured phase difference error was measured, with normal human movement in an indoor measurement area (for detailed description of the measurement see Section 5.3.2). The distribution of the phase error can be approximated with a zero mean Gaussian distribution with standard deviation of $\sigma \sim 0.1\pi$. Since this value represent nominal circumstances only, it is not very unlikely that additional disturbances may cause larger errors and thus violate the sampling theorem. Interference resulting from devices emitting radio waves around the measurement frequency f may also cause phase measurement errors.

Fast object movement: Moving targets cause phase changes during their movements. If the speed of the tracked object is v_{obj} and the phase sampling frequency is f_s then the object covers distance v_{obj}/f_s between two samples, which cannot be more than $\lambda/4$, corresponding to phase shift of π in worst case. This implies the constraint on the speed of the object:

$$v_{obj} < \frac{\lambda f_s}{4} \quad (13)$$

If constraint (13) is violated, the sampling theorem may also be violated. Notice that this source of error affects all of the measurement channels and thus cannot be eliminated by the proposed method.

Missing samples: The communication mechanism of the proposed measurement system allows packet losses. If measurement results are missing, the sampling theorem may be violated on some channels. In practical situations this is the main error source.

The above effects, or their combination, can cause phase unwrapping errors, which may result in permanent error of $2i\pi$ (i is integer) in the corresponding full phase value, causing a permanent localization bias. To handle the occasional phase unwrapping errors, a novel Fault-tolerant Hyperboloid-based tracking algorithm (FHT) is proposed, utilizing SPU. The proposed algorithm detects and corrects phase unwrapping errors, assuming that the number of error free phase measurements is significantly larger than the ones containing unwrapping errors.

Notice that when an erroneous phase estimate $\phi_{m,k}$ is present then the d_m distance of the corresponding hyperbola h_m from the estimated location becomes large. Analogously, large $d_m(x_e, y_e)$ value indicates that the corresponding $\phi_{m,k}$ has potentially large (possibly unwrapping) error. The proposed algorithm uses this idea to detect and correct phase unwrapping errors.

The SPU scheme, proposed in Section 3.2, is utilized in the improved radio-interferometric tracking (FHT). The application-specific cost function $F(.)$, which quantifies the relationships between different channels of phase measurements, in the radio-interferometric tracking application is defined as

$$F = min_{(\mathrm{x,y})}\, \eta(x, y) = \eta(\hat{x}, \hat{y}), \tag{14}$$

where the error map η is the following:

$$\eta(x, y) = \sum_{m=1}^{M} L_\delta(d_m(x, y)), \tag{15}$$

where L_δ is the Huber loss function with parameter δ as follows:

$$\begin{array}{ll} L_\delta(q) = 0.5q^2, & \text{if } |q| \leq \delta \\ L_\delta(q) = \delta|q| - 0.5\delta^2 & \text{otherwise} \end{array} \tag{16}$$

Notice that in η the Huber function is used, which is less sensitive to outliers than the square error (which is used e.g. used in (10)), due to its linear nature for higher input values. Parameter δ should be chosen to reflect the expected range of d_m for correct measurements.

In the proposed system $\delta = 0.15m$ was used. The flowchart of the tracking algorithm can be seen in Figure 12. At time instant k, using M quads, a set of new phase difference measurements $\varphi_{m,k}$ $(m = 1, 2, \dots, M)$ are collected in the range of $[0, 2\pi]$. Then the phase values are unwrapped, using DCD equations (6)-(8), resulting full phase values $\phi_{m,k}$ in the range of $[-\infty, \infty]$. In the next step the error surface is generated using (15), and the initial position estimate $(\hat{x}, \hat{y})$ is calculated using (14). If there are potentially incorrectly unwrapped phase values then in the following steps the algorithm tries to detect the misplaced hyperbolas and correct the corresponding phase values. At first (Step *Select Worst Hyperbola*) the m^{th} hyperbola h_m is selected, such that

$$m = \operatorname{argmax}_i d_i(\hat{x}, \hat{y}) \tag{17}$$

where d_i is the distance between hyperbola h_i and the estimated position $(\hat{x}, \hat{y})$. Then in Step *Correction* two new trial correction terms $\Lambda_{m,k} + 2\pi$ and $\Lambda_{m,k} - 2\pi$ are generated. With these correction terms the full phase values and the associated error surfaces are recalculated using (15). The next step (*Select Best and Update*) evaluates the initial and the corrected error surfaces and selects the one with the lowest minimum value. The correction term $\Lambda_{m,k}$ and the corresponding unwrapped phase value $\phi_{m,k}$ is permanently updated with the one producing the best result. The position estimate $(\hat{x}, \hat{y})$ is calculated using (15) and (14). The correction phase is repeated M_{MAX} times, i.e. at most M_{MAX} erroneous phase measurements are corrected at a time instant. In rare cases, when the number of simultaneous erroneous phase values is higher than M_{MAX}, the corrections are made in subsequent time instances.

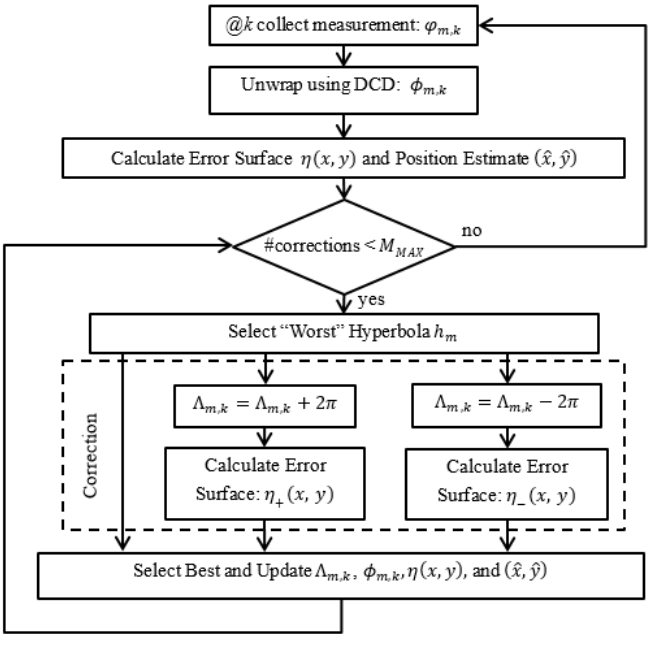


Figure 12. Flowchart of the Fault-tolerant Hyperboloid-based Tracking (FHT) algorithm

The pseudo code representation of the tracking algorithm can be seen in Figure 13 for a single time instant k. The unwrapped phase values are denoted by *p1..pm*. The *calculate_error* function implements the application specific cost function defined in (14) and returns the minimum cost value and position. The *calculate_distances* function computes the d_m distance between hyperbolas h_m and the estimated position (*x*, *y*).

```
P0 = {p1 .. pm}  // set of $\phi_{m,k}$
Hchecked = {}    // set of already checked hyperbolas

[cost, x, y] = calculate_error(P0) // see (14) and (15)
[d1..dm] = calculate_distances(P0, x, y) // see (17)
WHILE |Hchecked| < $M_{MAX}$ DO

  j = argmaxi( d1..dm ), where j ∉ Hchecked // Select Worst
  P+ = (P0 \ pj)  ∪ pj + 2π
  P- = (P0 \ pj)  ∪ pj - 2π

  CHOSE Pnew ∈ {P0, P+, P-} FOR WICH cost IS MINIMAL, WHERE
    [ncost, nx, ny] = calculate_error(Pnew) // see (14) and (15)
    [nd1..ndm] = calculate_distances(Pnew, x, y) // see (17)

    cost = ncost
    x = nx
    y = ny
    d1 = nd1, d2 = nd2, … , dm = ndm
    P0 = Pnew
    Hchecked = Hchecked ∪ j
  END
END
```

Figure 13. Pseudo code of the Fault-tolerant Hyperboloid-based Tracking (FHT) algorithm

3.4. Evaluation

In this section the proposed concept of the basic and fault-tolerant hyperboloid-based tracking (BHT and FHT) will be evaluated using simulations and real measurements. First a theoretical error analysis method will be shown then the simulation results will be introduced and finally real measurements will be presented. The effects of the various node placement will be also examined.

3.4.1. Error analysis

The location estimate is given as the intersection of two hyperbolas. When measurement noise and inaccuracies are present, the hyperbolas become inaccurate (i.e. the hyperbolas are moved from the ideal position, see e.g. Figure 9). The intersection of two inaccurate hyperbolas results an inaccurate location estimate. In order to characterize the localization error, let us use the following linearized model (see Figure 14):

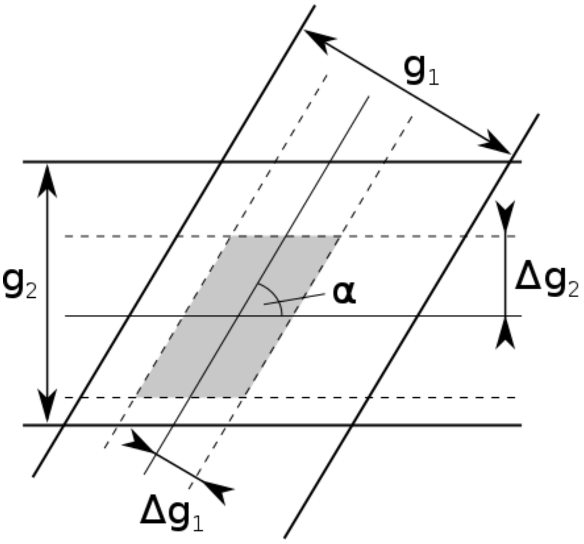


Figure 14. Localization error with linearized hyperbolas

Near the intersection of the hyperbolas, let us use a linear approximation of the hyperbolas as shown in Figure 14. The ideal hyperbolas are represented by thin solid lines, the measured inaccurate hyperbolas are shown by dashed lines. In the model we assume the ideal and measured hyperbolas are parallel. Let us denote the angle between the two linearized ideal hyperbolas by α, where $\alpha \leq \pi/2$.

Let us approximate the maximal distance between the ideal and the measured hyperbolas by Δg_1 and Δg_2, as shown in Figure 14. Δg_i, $i = 1, 2$ depends on the measurement phase noise $\Delta\vartheta$, and the distance g_i between the fringe patterns (hyperbolas corresponding to $\vartheta = 2k\pi$ phase values) near the intersection point. The fringe pattern is shown by thick lines in Figure 14. Δg_i can be approximated as $\Delta g_i = g_i \cdot \Delta\vartheta/2\pi$.

If the maximal distance Δg_i between the ideal and the measured hyperbolas is known, then the location estimate can be anywhere inside the shaded area of Figure 14. Thus the maximum location error equals to half of the longer diagonal d of the shaded parallelogram.

Using elementary geometric calculations the maximum location error d can be estimated as

$$d = \sqrt{y_1^2 + y_2^2 + 2y_1y_2cos(\alpha)}, \qquad (18)$$

where $y_i = \Delta g_i/sin(\alpha)$.

Examining Figure 14 and (18), it is clear that the localization error is small if Δg_1 and Δg_2 are small and α is close to $\pi/2$. The localization error increases if Δg_1 or Δg_2 increase, or α decreases to 0.

The quality of the localization greatly depends on the transmitter nodes deployment. For a simple example see Figure 15, where two pairs of transmitters, shown by red and blue, are deployed. As Figure 15 shows, the fringe patterns are closest to each other near the axis between transmitter pairs (the axes are shown by thick lines in Figure 15). As the distance from the axes increases the distance between the fringe lines also increases, causing increase in variables g_i and Δg_i. Thus, according to (18), the most accurate results are expected between the transmitter nodes, and as the distance from the axis increases, the potential accuracy decreases.

In the example setup of Figure 15 fringe lines are close to orthogonal in the area encircled by the transmitters, here higher accuracy is expected, according to (18). Outside this region the angle between red and blue fringe lines decreases, at certain points red and blue lines become almost parallel, here low localization accuracy can be expected.

Since the relationship between accuracy and node placement is fairly complex in general, the design of the node placement can be supported by numerical simulation, based on the principles discussed above, see Section 3.4.3.

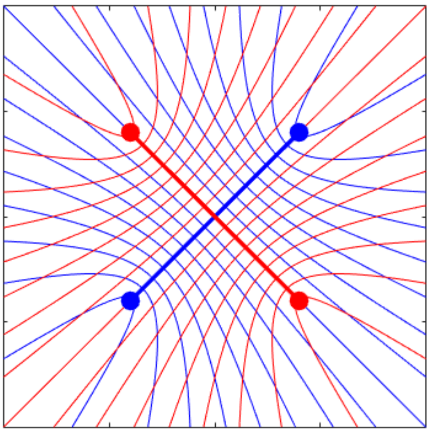
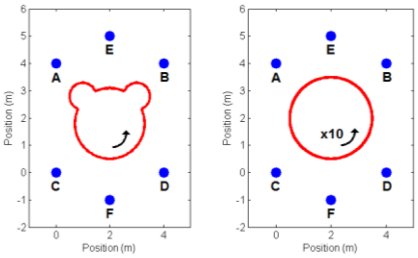


Figure 15. Fringe pattern of a simple measurement setup with two pairs of transmitters. Thick lines represent the axes between the transmitter node pairs. Thin lines represent the fringe pattern, i.e. hyperbolas corresponding to $\vartheta = 2k\pi$ phase values.

3.4.2. Simulation results

The performance of the FHT algorithm was tested using simulations. In the simulations, corresponding to the real measurements, $\lambda = 0.35$m was used, and the sampling frequency of phase measurements was $f_s = 16.7Hz$. The speed of the simulated object in both tests was approximately 3km/h. The test setups are shown in Figure 16(a) and Figure 16(b), where positions of nodes A, B, C, D, E, and F are illustrated. The bear-head and circle trajectories, used in the tests, are also shown.



(a) (b)

Figure 16. Placement of the nodes and object trajectories in the simulations

In the simulations five configurations were used to produce virtual phase measurements with number of quads $M = 2$, 4, 6, 10, and 15. Figure 17 shows the configurations: nodes of the graphs represent the sensor nodes A-F, and edges represent pairs of nodes which served as fixed transmitters in a quad. The fixed receiver was a node which had the largest distance from both transmitters (not shown in the figure), and the fourth node in each quad was the tracked receiver itself (also not shown in the figure).

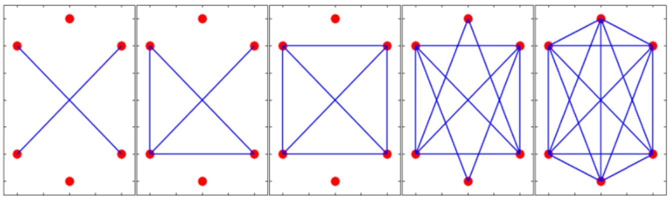


Figure 17. Node configurations used to select 2, 4, 6, 10, and 15 quads: connected nodes represent the transmitter pairs.

In the first test the noise sensitivity of the proposed algorithm was tested, with object trajectory shown in Figure 16(a). In each point of the trajectory the ideal phase measurements were calculated using (1) and (2), then random noise was added to the ideal values. In the tests Gaussian noise was used, with standard deviation of $\sigma = 0.1\pi$, 0.2π, and 0.3π, representing nominal, high, and very high noise levels, respectively (see e.g. Section 5.3.2 for real noise measurement data). The object location was estimated from the noisy phase values, using the proposed FHT method with $M_{MAX} = 2$. The results in Figure 18 show that for nominal amount of noise the algorithm performs well, even for small number of quads. As the noise level is increased, for low M there is significant position error, due to phase unwrapping errors. As the redundancy (i.e. the number M of quads) is increased, the quality of the estimator gets better, as the numerical results (mean and standard deviation of the position error) in Table 1 also show. The highlighted cells in the table show those configurations where there was no permanent unwrapping error.

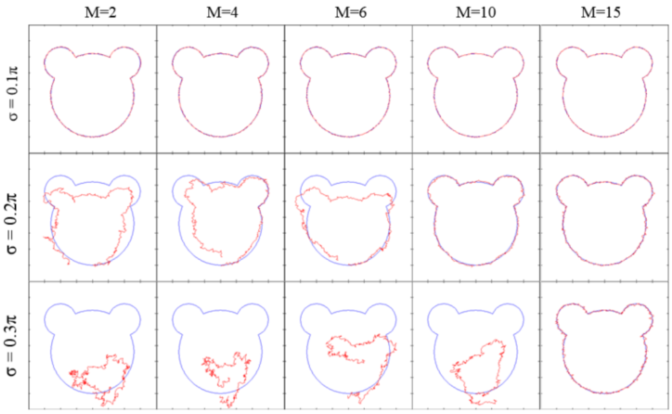


Figure 18. Simulation results for various phase noise levels σ and various numbers of quads M

Table 1. The mean and the standard deviation of tracking errors with the simulated phase measurements. The number of quads and the variance of the simulated phase noise were M and σ, respectively.

	mean/std (mm)				
M σ	2	4	6	10	15
0.1π	**13/14**	**11/13**	**10/11**	**8/9**	**9/10**
0.2π	357/390	238/300	293/337	**36/40**	**23/27**
0.3π	1262/1381	1165/1261	866/925	1058/1167	**31/36**

The second simulation test compares the performances of the proposed BHT and the self-correcting FHT algorithms. The moving object made 10 rounds along a circular path shown in Figure 16(b). The standard deviation of the additive phase noise was set to 0.1π, corresponding to nominal phase noise conditions. In addition to the Gaussian measurement noise, large measurement errors were also injected at random positions of the trajectory: in each round four phase measurements were replaced by a random number in the range of $[0,2\pi]$ with uniform distribution, simulating faulty measurements. The results of the tracking are shown in Figure 19. The large measurement errors caused permanent phase unwrapping error in the BHT algorithm, independently of the number of utilized quads. The self-correcting FHT method (with parameter $M_{MAX} = 1$) was able to tolerate the errors as the number of quads increased: for $M \leq 4$ there is large position error, due to the permanent unwrapping error, but for $M = 6$ only a small section can be observed where the estimated trajectory deviates from the true trajectory, otherwise the algorithm corrected the false measurement. For $M = 10$ and $M = 15$ there was no observable permanent unwrapping error. The numerical results can be seen in Table 2, the highlighted cells containing cases with no unwrapping error.

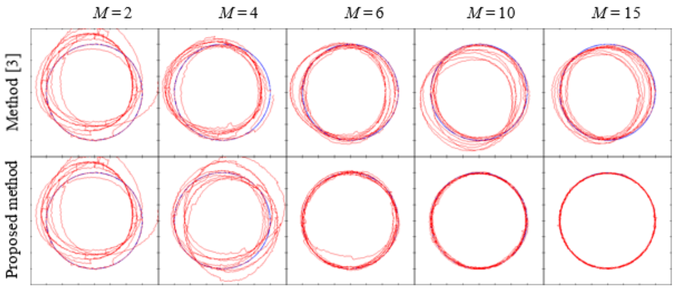
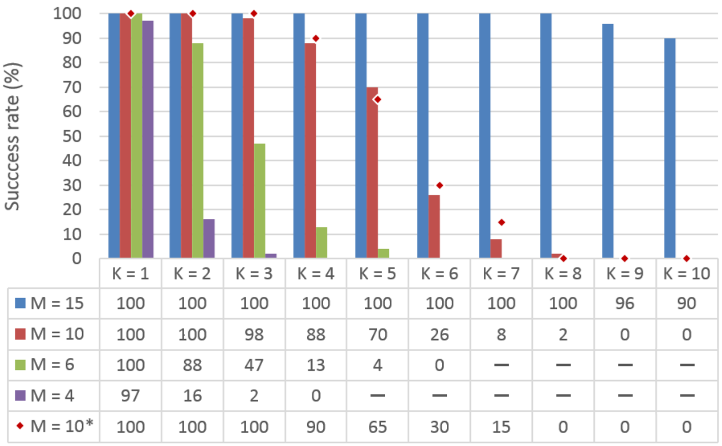


Figure 19. Simulation results of the BHT algorithm and the proposed self-correcting FHT method, with various numbers of quads M, in the presence of both phase noise and sporadic large measurements errors.

Table 2. The mean and the standard deviation of tracking errors with the simulated phase measurements. The number of quads used in the measurement was M. The applied phase unwrapping mechanism was the standard DCD in BHT, while the FHT utilized the proposed SPU method.

	mean/std (mm)				
M	2	4	6	10	15
BHT	330/373	248/265	193/223	230/277	233/268
FHT	NA	249/296	60/81	**38/42**	**19/21**

Another test was performed to check the self-correction mechanism of the proposed FHT algorithm. In the test multiple simultaneous large errors were injected in the input phase series. The input trajectory was one quarter of a circle from the previous test (i.e. the movement was simulated from the 6 o'clock position to the 3 o'clock position), the measurements were generated using the noisy ideal values, using $\sigma = 0.15\pi$. At the half of the trajectory, K out of M unwrapped phase values $\phi_{m,k}$ were altered by $\pm 2\pi$, modelling unwrapping errors. For each (K, M) pair 100 independent inputs with different noise realizations were generated and the proposed FHT algorithm was run on each of them. To evaluate the efficiency of the self-correcting mechanism, each of the M channels was individually monitored and the unwrapped phase values were compared to the known true values at the end of the trajectory. A run was successful where *all* of the channels were correctly restored. In the tests the value of M was 4, 6, 10, and 15 (see Figure 17 for the associated configurations). The number of simultaneous injected errors K was changed between 1 and $\min(M, 10)$. The parameter M_{MAX} was changed between 1 and M. The bars in Figure 20 show only the percentage of successful runs for $M_{MAX} = 1$. Other cases will be discussed later.



	K = 1	K = 2	K = 3	K = 4	K = 5	K = 6	K = 7	K = 8	K = 9	K = 10
■ M = 15	100	100	100	100	100	100	100	100	96	90
■ M = 10	100	100	98	88	70	26	8	2	0	0
■ M = 6	100	88	47	13	4	0	—	—	—	—
■ M = 4	97	16	2	0	—	—	—	—	—	—
◆ M = 10*	100	100	100	90	65	30	15	0	0	0

Figure 20. Percentage of successful error corrections for K simultaneous injected errors and M measurement channels. Bars show cases for $M_{MAX} = 1$ (first four lines in the table), while diamonds illustrate a case for $M_{MAX} = 4$ (last line in table).

As Figure 20 shows, the robustness of the algorithm greatly increases as the number of utilized quads increases. For $M = 4$, 97% of the single-error cases were successfully handled, but for higher number of errors the success rate is low. For $M = 6$ all the single-error cases were corrected and a significant portion (88%) of the double errors were also corrected. When $M = 10$, two simultaneous errors can always be compensated for and even three and four simultaneous errors can be handled relatively well. The case $M = 15$ was very robust: it was able to handle as much as 8 simultaneous errors.

In the experiments the effect of M_{MAX} was also examined by changing M_{MAX} between 1 and M (the bars in Figure 20 show the results for $M_{MAX} = 1$). The behavior of the algorithm was very similar for $M_{MAX} > 1$, thus the results are not presented here in detail, only one case is illustrated in Figure 20: the data denoted by $M = 10^*$ was generated with $M_{MAX} = 4$. The performance of the algorithm is very similar to the case of $M_{MAX} = 1$ (see data denoted by $M = 10$). This similarity was observed in all experiments, the algorithm is not sensitive to the value of M_{MAX}: if simultaneous errors can be corrected in one time instant (performing $M_{MAX} > 1$ corrections in one time instant) then the algorithm is also able to correct the same errors in multiple subsequent time instants, with one correction only at each time instant ($M_{MAX} = 1$), providing almost the same performance. The result is intuitive for cases where the input data is changing slowly (low object speed), and with sparse errors where the correction phases do not overlap. Thus to decrease the computational complexity, $M_{MAX} = 1$ is a reasonable choice, according to the experiments.

3.4.3. Node placement analysis

In this section an important design problem, the placement of the infrastructure nodes in radio-interferometric localization systems, is discussed. For analysis purposes a metric called maximum localization error $E(A, \Delta\varphi_{max}, C)$ will be used. The localization is performed in the *observed area* A, thus the error is calculated here and the error value is irrelevant outside of this area. The measurement uncertainties are modelled with maximum phase measurement error $\Delta\varphi_{max}$. Node configuration C contains the placement of the infrastructure nodes and also the utilized quad configurations.

The mean localization error E_{mean} and maximum localization error E_{max} are computed along a grid (x, y), where $(x, y) \in A$, as follows:

$$\begin{gathered} E_{mean}(A, \Delta\varphi_{max}, C) = \mathrm{mean}_{(x,y)\in A}\big(\Delta p_{max}(x, y, \Delta\varphi_{max}, C)\big) \\ E_{max}(A, \Delta\varphi_{max}, C) = \mathrm{max}_{(x,y)\in A}(\Delta p_{max}(x, y, \Delta\varphi_{max}, C)) \end{gathered} \tag{19}$$

where $\Delta p_{max}(x, y, .) = \mathrm{max}_{\forall\bar{\boldsymbol{\varphi}}}(\|\hat{x}(\boldsymbol{\varphi}_{id} + \bar{\boldsymbol{\varphi}}) - x, \hat{y}(\boldsymbol{\varphi}_{id} + \bar{\boldsymbol{\varphi}}) - y\|_2)$ is the maximum localization error at location (x, y), $\boldsymbol{\varphi}_{id}$ is the ideal phase measurement vector with M elements for configuration C, calculated at (x, y) using (1) and (2), and $\bar{\boldsymbol{\varphi}}$ is the injected phase error vector, where for each element $\bar{\varphi}_i$, $i = 1, 2, \dots, M$ of $\bar{\boldsymbol{\varphi}}$ it is true that $\Delta\varphi_{max} \leq \bar{\varphi}_i \leq \Delta\varphi_{max}$.

An approximate fast calculation of Δp_{max} is illustrated in Figure 21(a). For each grid point (x, y) and for each channel i the ideal solution (a hyperbola) is calculated, along with two other solutions, where distorted phase measurement $\varphi_{id,i} \pm \Delta\varphi_{max}$ was used in (1) and (2); in Figure 21 two channels are shown. For each channel the distorted solutions provide two hyperbolas, with some distance from the ideal one: any possible solution, with maximum phase error $\Delta\varphi_{max}$ lies in the band between the two hyperbolas. The intersection of the bands (also calculated on a finite grid) provide the set of possible solutions (the uncertainty region) for max phase measurement error $\Delta\varphi_{max}$, shown by striped region in Figure 21 . The largest distance in this set from point (x, y) provides the maximum localization error $\Delta p_{max}(x, y, .)$. Using this method, for any configuration C, observation area A, and phase error $\Delta\varphi_{max}$, the error map Δp_{max} can be computed (examples are shown in Figure 23), and the maximum of the error map provides the maximum localization error. Notice that Δp_{max} is an alternative way to quantify the geometric dilution of precision (GDOP) [44]. As Figure 21(a) illustrates, Δp_{max} is primarily affected by (1) the width of the individual error bands, wider bands causing larger error, and (2) the angle of the hyperbolas, the best case being perpendicular hyperbolas. The error sources and their effect will be illustrated by numerical analysis examples.

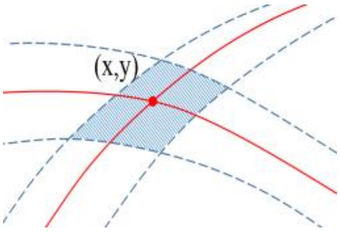


Figure 21. Localization error map Δp_{max}. (a) calculation of Δp_{max}. (x,y): true position, solid lines: hyperbolas corresponding to the true position, dashed lines: hyperbolas corresponding to noisy measurement with phase error $\Delta\varphi_{max}$, striped area: uncertainty region.

The numerical analysis was performed using the configurations shown in Figure 22. The nodes were placed in the corners of a the *node area*, which is a rectangle of size L_1 x L_2, as shown in Figure 22(a). The observed area, where the localization is to be performed, was a square of size DxD, with $D = 5$m. The observed area was always placed in the center of the node area, as shown in Figure 22(a). In the analysis four typical configurations were used, as shown in Figure 22(b), where four infrastructure nodes are denoted by red dots and the transmitter pairs of the utilized quads are denoted by edges between corresponding nodes. In configurations cL and cX, shown in Figure 22(b), two quads were used ($M = 2$). In configuration cO, four quads were used ($M = 4$) in a circular pattern, while configuration cF (full) used all the possible quads ($M = 6$).

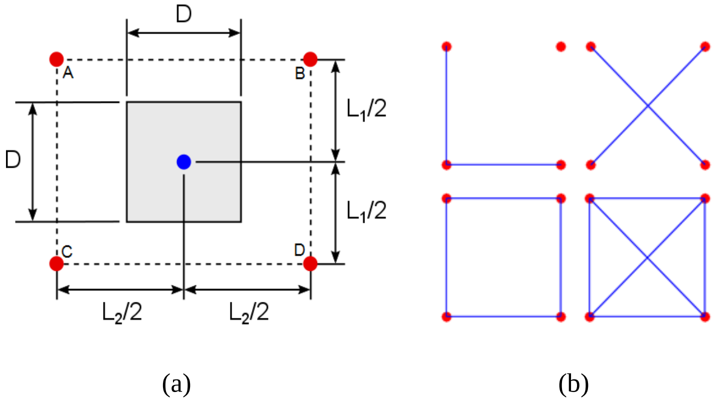
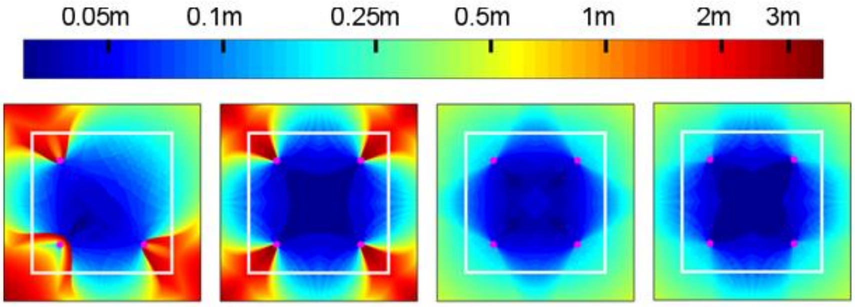


Figure 22. Configurations used for infrastructure node placement analysis. (a) distance notation, (b) configurations cL, cX, cO, and cF (full). Nodes are denoted by red vertices, transmitter pairs used in the configurations are denoted by edges.

During the analysis first the error map Δp_{max} was calculated. Notice that for illustration purposes, Δp_{max} was calculated for a somewhat larger area than the observed area. In Figure 23 representative examples can be seen for all test configurations: the error map is color coded, transmitter node positions are shown by light red dots, and the observed areas are shown by white rectangles.

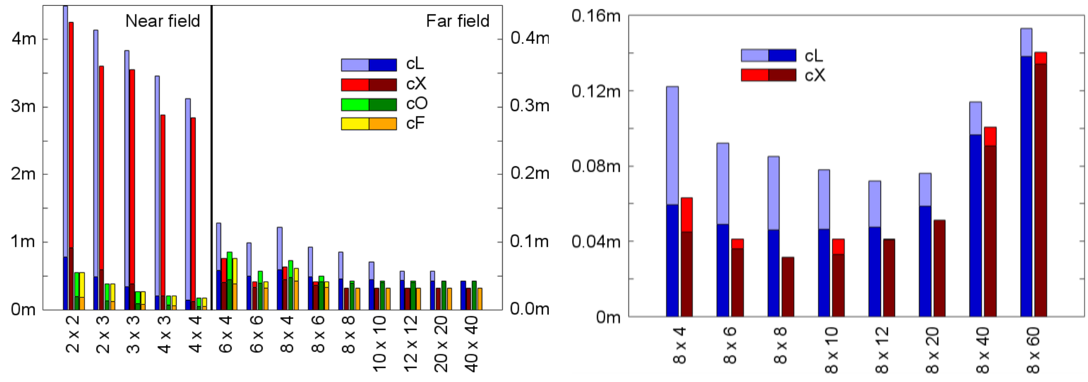


(a) (b) (c) (d)

Figure 23. Localization error maps (Δp_{max}) of configurations L, X, O, and F, respectively. Nodes placed in the corners of a 3m x 3m rectangle.

The analysis was carried out in two main setups, for distance notation refer to Figure 22(a). In Setup 1 ("square") $L_1 \approx L_2$, with 14 test cases from 1m x 1m to 20m x 20m. In Setup 2 ("stripe") $L_1 = 8\text{m}$ and L_2 changed from 4m to 60m, producing 8 test cases. The maximum phase error in the test was set to $\Delta\varphi_{max} = 0.1\pi$. Those cases where the infrastructure nodes are inside and outside of the observed area will be called *near-field* and *far-field* cases, respectively. In Figure 23 near-field examples are shown for each configurations. In cL and cX the localization error is large near the transmitter positions, while in cO and cF this effect is much lower. In all cases the center of the observed area has good coverage and small error.

The calculated E_{mean} and E_{max} localization error values are shown in Figure 24 for each test case, with cL, cO, cX, and cF corresponding to color codes blue, red, green, and yellow, respectively, and darker and lighter shades showing E_{mean} and E_{max}, respectively. In Figure 24 results for Setup 1 can be seen, for all four test configurations. In near-field test cases the error is significantly larger, especially for cL and cX (see the error maps for case 3m x 3m in Figure 23) In far field cases the maximum localization error is in the range of 31-128 mm. Notice that after reaching a "safe" far-field distance, further increase of the sensor distance has no significant effect on the error. In the simulation this is partly due to the discrete evaluation of the error map (the resolution of the grid was 10mm), but in practice this error is already in the range of the position accuracy of the beacons.



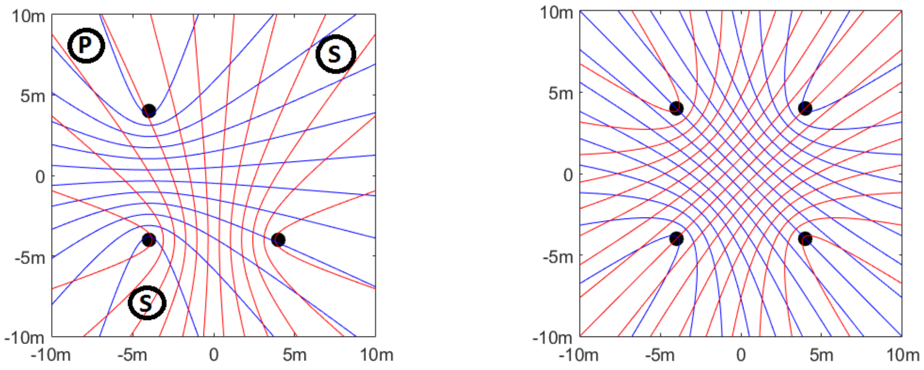
(a) (b)

Figure 24. Mean and maximum localization error. (a) Setup 1. Notice the different vertical scale for near-field and far-field cases (b) Setup 2.

According to Figure 24(a), the error is systematically the largest in cL, in both near and far-field cases. The sources of error are explained using Figure 25, which shows the fringe patterns of cL and cX, for the case of 8m x 8m. For sake of clarity, only every fourth contour line is plotted, corresponding to unwrapped phase values of $8k\pi$, where k is integer. Notice that in reality the fringe pattern is four times as dense. Blue and red contours correspond to transmitter pairs: (A,C) and (C,D) in case of cL, see Fig. 13(a), and (A, D) and (B, C) in case of cX, see Figure 25(b). Clearly, contour lines are the densest between the transmitter pairs (it is easy to show that along the transmitter axis the distance between hyperbolas of Figure 25 is $4 \cdot \lambda/2 \cong 0.7\text{m}$), and the fringe density decreases as the distance from the transmitter axis grows. The distance between contour lines is proportional with the width of the uncertainty bands, shown in Figure 21(a). In case of cL in Figure 25(a), the distance between contour lines significantly increases towards the upper right corner (for both blue and red contours), indicated by S. In this area the error is higher (according to (18)), as can be seen in the corresponding green region of Figure 21(b). Similar, but less obvious effect can be seen along the transmitter axis, outside of the sensors: here the phase changes very slowly in a large area, indicated by the lack of contour lines; one of these areas for the blue contours are also marked by S in Figure 25(a). In these areas the position error is the largest, shown by red in Figure 21.

Another possible reason of increased error is when the angle of intersecting hyperbolas is small ($\ll \pi/2$), according to (18): in this case the GDOP is large. This effect can be seen in the corners, outside of the sensor positions; one such area is marked by P in Figure 25(a), and the high error is indicated by the corresponding red regions of Figure 21(b). The latter effect can also be seen for cX in Figure 25(b) and the corresponding red regions of Figure 21(c). In the center (far-field) region of cX, however, hyperbolas are nearly orthogonal, as shown in

Figure 25(b), thus cX performs better in far-field than configuration cL. Configurations cO and cF have redundancy, e.g. cO can be considered as the union of two cL's, while cF can be constructed from cX and two cL's. In the critical regions, where cL and cX perform poorly, the redundant configurations cO and cF have other hyperbolas to help to reduce the maximum error, providing better performance in both far and near fields.



(a) (b)

Figure 25. Fringe patterns for test case 8m x 8m, node positions are shown by dots, (a) configuration cL, (b) configuration cX. S and P indicate selected areas where the position error is potentially high, due to sparse or parallel contour lines, respectively.

To further illustrate the effect of the possible error sources, the results of Setup 2 are shown in Figure 24(b), for cL and cX. Since $L_1 > D$, all the cases can be considered far-field. As the figure clearly shows, the localization error decreases as the distance L_2 increases until $L_2 = 6\text{m}$, then the error increases with the increase of L_2. The explanation is shown in Figure 26(a) and (b), showing fringe patterns of configurations cL and cX, respectively, for $L_2 = 60\text{m}$. Notice that only the central area, around the observed region is shown. In Figure 26(a) the blue lines correspond to parallel and dense hyperbolas generated by transmitter pair (A,C). These lines are dense since the distance of the observation area from the axis A-C is small. Red hyperbolas, generated by transmitter pair (C,D), diverge at the center of the sensor area (being far from axis B-C), producing larger localization error in the observed region. The larger L_2 the larger the error, as was measured in Figure 24(b). In case of configuration X, shown in Figure 25(b), both red and blue hyperbolas are dense in the observation area, but their angle is far from 90 degrees: as L_2 increases the red and blue hyperbolas become more parallel, thus small perturbations in the hyperbolas' positions cause large GDOP.

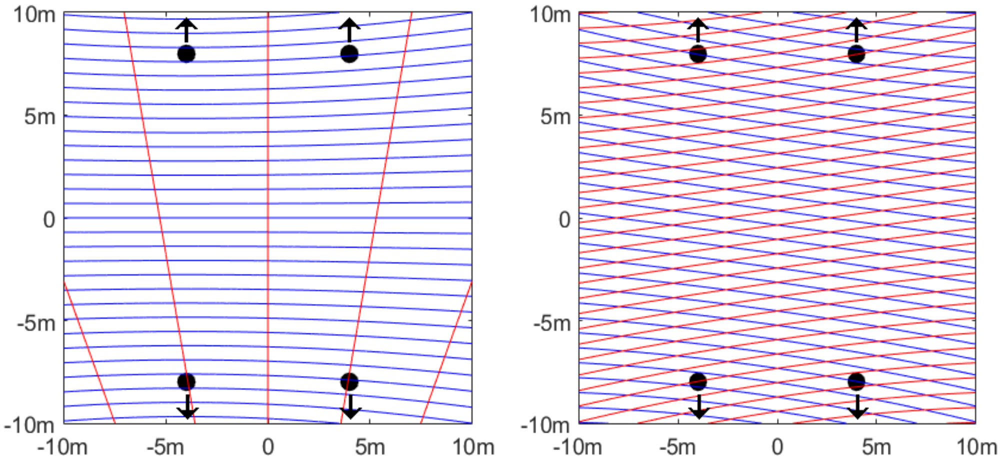


Figure 26. Fringe patterns for test case 8m x 60m, showing the central 20m x 20m area. (a) configuration cL, (b) configuration cX.

The reported case studies do not intend to provide full and general analysis, rather they illustrate the operation of the numerical analysis method, and also offer important general design strategies: the nodes should be outside of the observed area, but the additional increase of the distance has little effect; the axis of the transmitter pairs should be close to the observed area; transmitter pairs with orthogonal axes are the most valuable; redundant configurations have potentially smaller error. The tests also indicate that with a reasonable selection of node positions good results can be achieved, since the method is not sensitive to moderate changes of the node positions. In practice these rules of thumb can be used to design rough deployment setups of the infrastructure nodes, taking into consideration the environmental constraints along with the effect of possible node failures, and then the best candidate can be chosen performing the error analysis using (19).

3.4.4. Measurements

First a test was carried out to illustrate the accuracy of the phase offset model (1) and (2) with measurements. Figure 27 shows the test setup with two fixed transmitters A and B, a fixed receiver C, and a moving node D, which moved along trajectories T1, T2, and T3. The tests were performed outdoors in an orchard where the nodes were placed 1.2m above ground level on a wooden table. Phase offset values were measured along the trajectories in 0.02m increments, collecting 100 measurements in each position. During the measurements the effect of external disturbances was minimized (e.g. there were no movements or operating electronic devices nearby).

The results for trajectories T1, T2, and T3 are shown in Figure 28. The figures show both the wrapped and unwrapped phase offset values. Blue lines denote the computed ideal phase offset, while red dots show the actual measured values. Note the significant variance in the phase offset measurements, despite of the relatively disturbance-free environment. Also note that the variance is safely below the limit of π, thus phase unwrapping is indeed possible using the expected continuity assumption in a disturbance free or low noise environment.

The frequency difference Δ*f* varied between 892Hz and 1420Hz during measurement T1, between 3472Hz and 3906Hz during T2, and between 416Hz and 1136Hz during T3. The variance of the phase measurement was the largest in T2, due to the large Δ*f*. Note that the transmission frequency is sensitive to temperature, thus Δ*f* changed quite rapidly during the measurements.

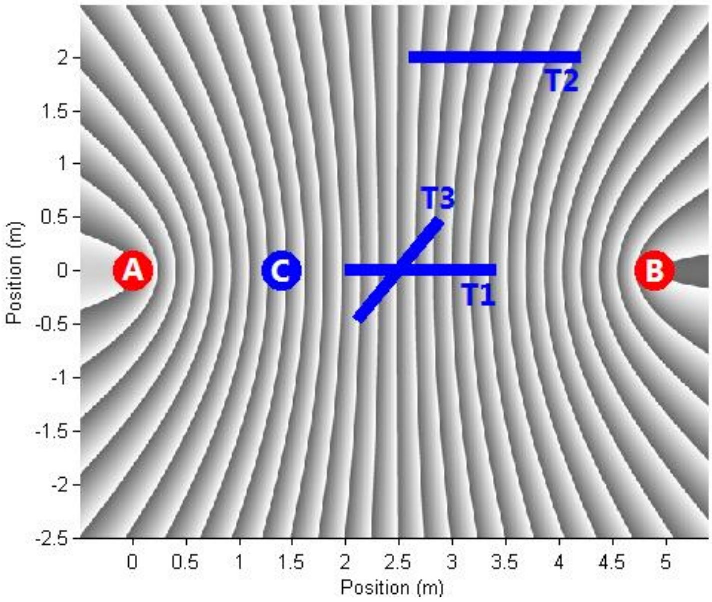
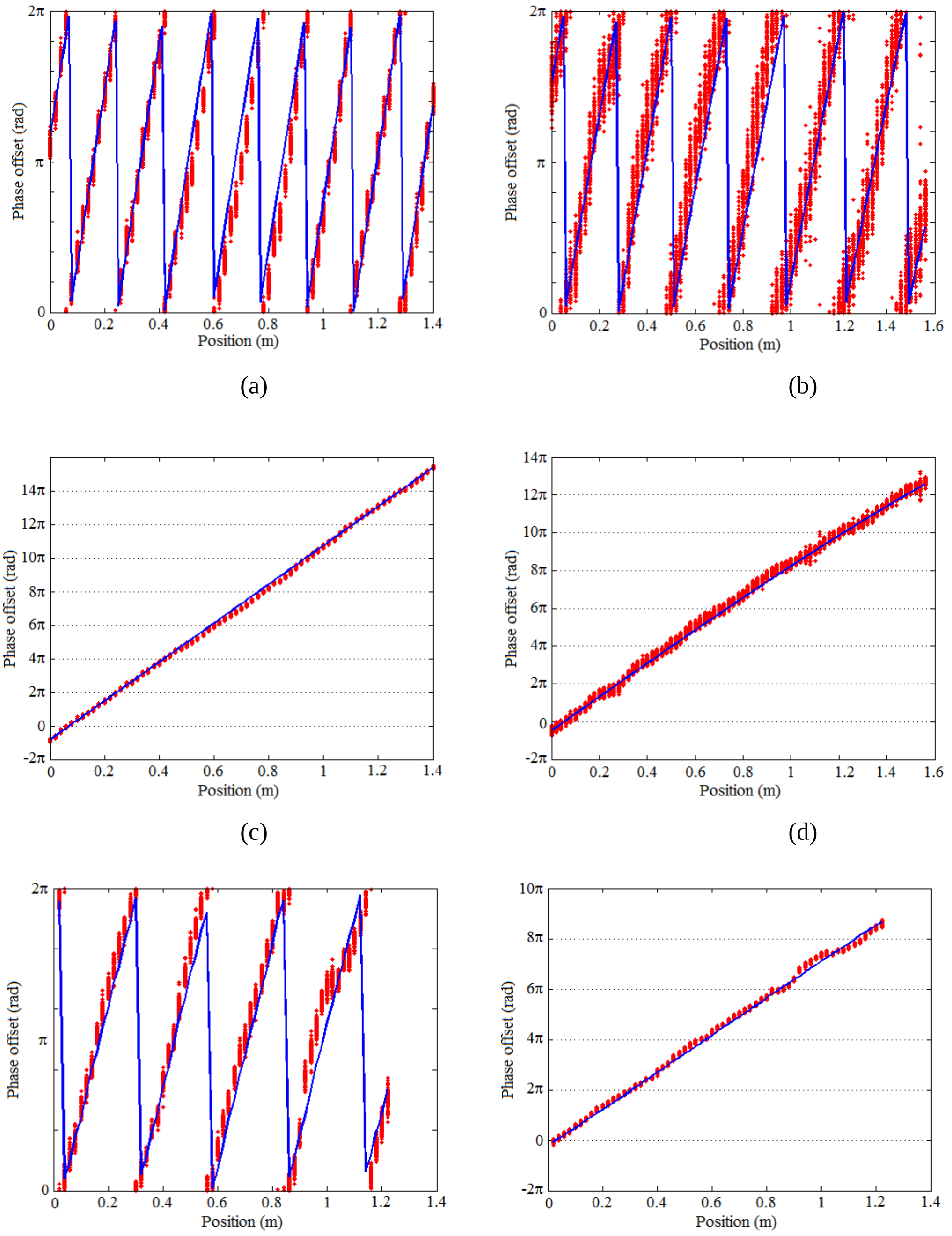


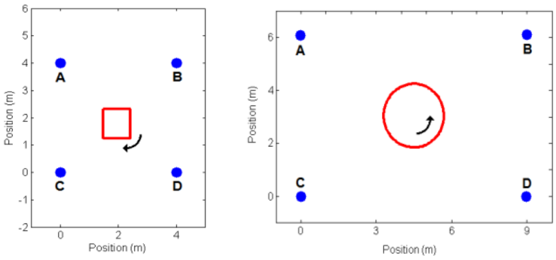
Figure 27. Setup for phase-offset test measurements using fixed transmitters A,B, fixed receiver C, and moving receiver D. Blue line segments show trajectories T1, T2, and T3 of the moving object D. The calculated ideal phase-offset pattern is also shown.



(a) (b) (c) (d) (e) (f)

Figure 28. Phase offset measurements for trajectory T1 (a)-(b), T2 (c)-(d) and T3 (e)-(f). Blue lines show the ideal phase offset values, scattered plot with red dots show the actual measurements. (a),(c),(e) wrapped and (b),(d),(f) unwrapped phase offset values.

To test the proposed BHT and FHT algorithms, real measurements were also performed, utilizing the phase-difference measurement method introduced in Section 5. The 4 fixed infrastructure devices generated the interferometric signals in the 868MHz ISM frequency band, by tuning the frequencies of the transmitters to provide beat signal with Δf in the range of a few kHz. One moving receiver was utilized to measure the interferometric signals. The measurement process utilized 6 quads, by selecting all the possible pairs of the 4 fixed nodes for transmitters. The measurement setups and the paths of the tracked object are illustrated in Figure 29. The phase unwrapping and location estimation was performed, using parameter $M_{MAX} = 1$.



(a) (b)

Figure 29. Placement of the nodes and object trajectories in the real measurements

In the first tracking experiment the fixed nodes were placed on tripods in an empty parking lot, to provide disturbance-free environment, and the tracked node was moved on a 2D plotter-like table, as shown in Figure 30. The table was made of wood and plastic to eliminate interference with radio signals, only the driving motors and the electronics contained some metal. The measurement setup and the path of the tracked object are shown in Figure 29(a). The signal wavelength and sampling frequency of the phase measurement were $\lambda = 0.35m$ and $f_s = 16.7Hz$, respectively. In the test the object was moved with constant speed, except for the corners. Measurements were conducted with 8 different object speed values between $v_{obj} = 1.1km/h$ and $v_{obj} = 7.0km/h$, and at each speed 10 experiments were made.

Figure 30. Radio-interferometric measurement devices (on tripods) and the plotter-like table used to move the tracked device.

The results for one particular experiment with speed v_{obj}=3.0 km/h are shown in Figure 31. The true trajectory is a rectangle, shown in blue in Figure 31(a), which was tracked by the proposed algorithm, shown in red. The phase unwrapping was free of permanent error, thus the tracking has no visible bias. The absolute tracking error is shown in Figure 31 (b), with mean error of 0.015m and maximum error of 0.056m.

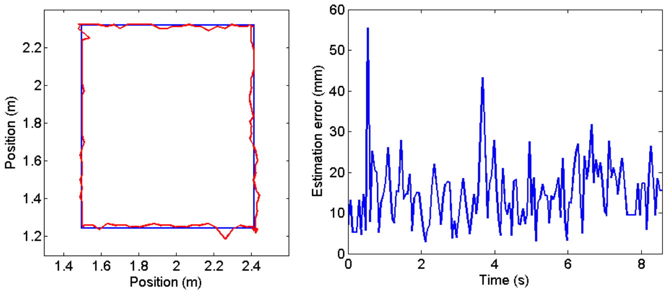


Figure 31. (a) Tracking results of the plotter-table experiment for v_{obj}=3.0 km/h. Blue: true trajectory, red: estimated trajectory. (b) Absolute position error as a function of time

For higher speed, naturally, the performance of the algorithm degrades. The results for different speed values are shown in Figure 32, where the percentage of the measurements with no remaining tracking error are plotted. At low speed 100% of the trackings was correct, while at higher speed the ratio of the correct trackings decreases. According to (13), in worst *noise-free* case the maximum speed not violating the sampling theorem is $v_{max} = 5.2\ km/h$. In the presence of measurement noise permanent tracking errors occur somewhat earlier, at $v_{obj} = 3.5\ km/h$, corresponding well with the theoretical expectations.

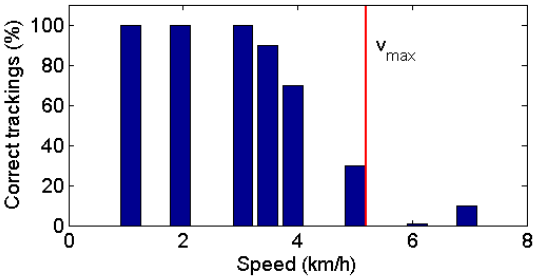


Figure 32. Percentage of correct trackings vs. object speed. v_{max} is the theoretical worst case limit in noise-free case

The detailed results are shown in Table 4, along with the mean and maximum deviation of the error. For low speeds the system provided low mean error around 15mm, independently of the speed. At higher speed values the object speed and measurement errors caused unrepairable phase errors, resulting permanent error in the localization as well.

The results clearly indicate that with the current hardware solution only limited speed (max. 3km/h, equivalent with slow walking) can be handled. The main bottleneck is the sampling frequency, the minimum of which is directly proportional with the maximum object speed, according to (13). Future work includes the speed-up of the measurement process, allowing the tracking of objects with higher speed.

Table 4. The mean and the maximum deviation (in mm) of positioning errors, and the percentage of correct trackings for different object speed v_{obj} (in km/h).Highlighted cells show cases with no permanent error.

v_{obj}	1.1	2.0	3.0	3.5	3.9	5.0	6.0	7.0
Mean	**15**	**15**	**15**	25	31	71	144	126
Max.	**77**	**77**	**68**	252	283	355	457	357
R %	**100**	**100**	**100**	90	70	30	0	10

The second tracking measurement was conducted indoors, with human presence in the experimental area. The moving object was carried by a person along a circular path, making five rounds, with short pauses after each circle. The speed of the movement was approx. 1km/h. In order to provide accurate positioning, the carried object was attached to the end of a rod, the other end of which was joined to a fixed axle. The measurement setup and the path of the tracked object are illustrated in Figure 29(b). The tracking results are shown in Figure 33(a), the true and estimated trajectory plotted in blue and red, respectively. The calculated absolute error is shown in Figure 33(b). In the experiment the mean absolute error was 0.02m with max. error of 0.13m. The error in this experiment was somewhat larger than in the disturbance-free outdoor experiment, probably due to indoor multipath effects and also the human presence. Notice also the systematic offset along certain parts of the trajectory; this is probably due to the distortion of the phase map resulting from multipath effects.

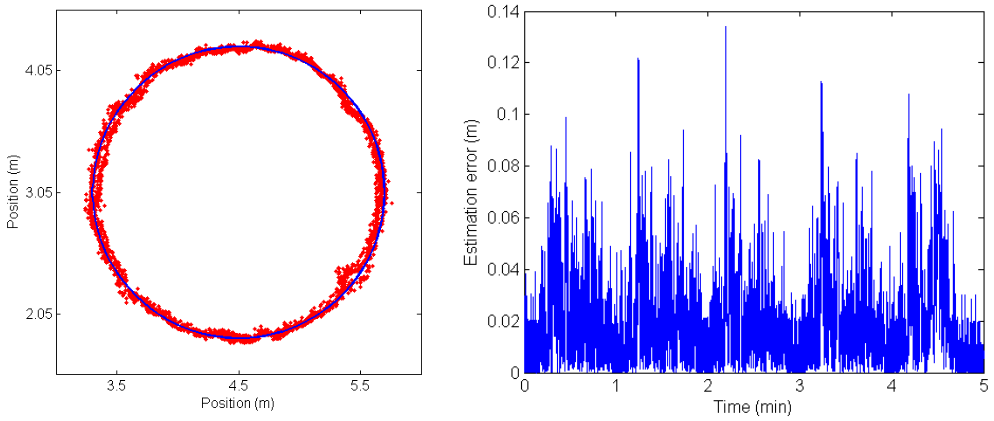


Figure 33. (a) Tracking results of the walking experiment. Blue: true trajectory, red: estimated positions along 5 cycles. (b) Absolute position error as a function of time

The performance of the BHT algorithm was also tested with the measurement data. Without the proposed phase correction, the mean and max. absolute position errors were 0.1m and 0.42m, respectively. During the experiment, phase unwrapping errors occurred three times (at 20 sec, 51 sec, and 3 min 47 sec), increasing the error of the BHT algorithm, while these sources of error were successfully handled by the proposed FHT algorithm.

3.5. Conclusion

A new class of phase unwrapping problems was identified, where simultaneous channels of one-dimensional wrapped phase measurements are used to calculate the full phase values at each channel. A novel Simultaneous Phase-Unwrapping algorithm (SPU) was proposed, which utilizes a high-level score function and an optimization mechanism to identify channels with faulty unwrapped values and correct them.

Novel radio-interferometric tracking methods were proposed, which utilize fixed infrastructure nodes to create interferometric signals, to track a moving radio receiver. The tracking methods utilized phase difference measurements from quads, consisting of two fixed transmitters, one fixed receiver, and the moving (tracked) receiver. In the Basic Hyperboloid-based Tracking (BHT) algorithm, multiple individually unwrapped phase values are used to generate the estimated location of the tracked object. In the Fault-tolerant Hyperboloid-based Tracking (FHT) algorithm multiple quad measurements are used, which are simultaneously unwrapped using SPU to provide a fault-tolerant method. The performances of the proposed algorithms were evaluated and compared using simulations and real tracking experiments. The simulation tests confirmed that redundant measurements along with SPU increased the accuracy of the tracking and also provided fault tolerance: the proposed FHT method is operational in noisy environments even in the presence of sporadic large measurement errors.

The proposed methods were implemented in a real-time measurement setup, where 4 fixed infrastructure nodes were utilized to provide 6 simultaneous phase measurement channels. The tests showed that the system with the current phase measurement solution is able to track objects with max. speed of 3 km/h with accuracy of a few centimeters.

4. Radio-interferometric tracking based on confidence maps

In this chapter a radio-interferometric tracking is proposed, which is essentially another solution for the same problem (presented in Section 2.4), but with a different perspective. In contrast to the hyperboloid-based tracking algorithms, where the problem is mathematically analyzed and an appropriate model were created, the proposed confidence map based methods are utilizing the empirical and inherently statistical properties of the radio-interferometric field. Despite the different point of views the underlying problem is the same, thus the proposed solutions have similar properties and are closely related.

The proposed confidence map based solution offers two operation modes:

- *Mode 1*: on-line tracking of objects with known initial position. In this mode the movement of the object is tracked in real-time from the known initial position.
- *Mode 2*: off-line tracking of objects with unknown initial position. In this mode a sufficiently long data must be recorded, while the object moves; after sufficient amount of data is collected the full object track is determined (retroactively) and the tracking can be continued as in on-line Mode 1.

Since Mode 1 is a subcase of Mode 2, we will discuss only the operation of Mode 2 in detail. The proposed solution has some realistic assumptions and requirements, as follows:

- R1: The exact positions of the infrastructure nodes are known.
- R2: The movement of the object is slow, compared to the sampling frequency. (According to experiments, the object should not move more than a few tens of millimeters between two consecutive phase measurement rounds. Tracking with maximum object velocity ~0.3 m/s was possible at a sampling frequency of 6 rounds/s. This is similar to and corresponds well with the speed constraint presented in (13) in Section 3.3)
- R3: The object trajectory is long enough to allow the resolution of the ambiguity problems (Mode 2 only). There is no explicit known formula yet on how long the trajectory should be; according to our experiments the more complex the movement (containing multiple directions) the shorter trajectory is enough. See the simulations and the measurement result is Section 4.7.
- R4: The initial object position must be known (Mode 1 only). In Mode 2 the initial position is unknown and is determined by the algorithm.

4.1. Position confidence map

The proposed tracking algorithm requires multiple independent phase measurements, which can be provided by utilizing the tracking infrastructure proposed in Section 3.1.3.

In each configuration c $(c = 1, 2, \dots, C)$ (see Section 3.1.3 for details) the ideal phase values $\vartheta_c^{(id)}(p)$ can be calculated for every possible object position p, using (1) and (2). For two dimensions, this gives a 2D phase map as can be seen on Figure 34. Note that phase maps can be pre-computed and stored, to increase the speed of the algorithm.

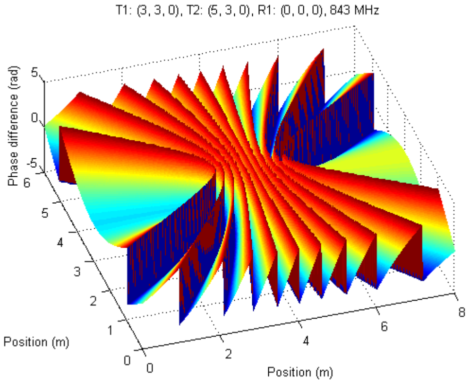


Figure 34. Phase map for a given configuration. The height of the map represents the ideal phase difference value between the fixed and the moving receiver, which operates on a 2D plane.

Measurement round k produces C phase measurements, each measurement corresponding to one measurement configuration, as follows:

$$\vartheta_{meas}(k) = [\vartheta_1(k), \vartheta_2(k), \dots, \vartheta_C(k)]. \tag{18}$$

Using the ideal phase maps and the measurements, a phase offset is calculated for each scenario c, as follows:

$$\Delta\vartheta_c(p,k) = \min_{i=-1,0,1}\left|\vartheta_c^{(id)}(p) + i2\pi - \vartheta_c(k)\right|. \tag{19}$$

Note that the phase offset values $\Delta\vartheta_c$ are between 0 and π. From the phase offsets an error map is calculated, as follows:

$$\varepsilon(p,k) = \frac{1}{C\pi^2}\sum_{c=1}^{C}\left(\Delta\vartheta_c(p,k)\right)^2 \tag{20}$$

The error ε is zero if the measurements exactly correspond to the ideal values. The maximum error is 1, indicating large difference between the ideal and measured phase values. Thus from the error map a confidence map can be defined, as follows:

$$\beta(p,k) = 1 - \varepsilon(p,k) \tag{21}$$

Confidence value $\beta(p,k)$ close to one indicates that position p can indeed be the real object position in time instant k, while low confidence values show that it is unlikely that the object is in position p in time instant k.

Figure 35 shows a confidence map computed for a scenario similar to the one shown in Figure 8. The true object position is at the center of the figure. Figure 35(a) shows the case when there is no measurement noise; in this case there are significant and sharp peaks in the confidence map. Note that in this case the confidence value is exactly 1 at the true object position, but there are several other significant peaks at phantom positions. This phenomenon is due to the phase wrapping in (1), and thus the true object position cannot be determined from one measurement. The noisy case is shown in Figure 35(b), where the peaks are less high and somewhat blurred. The phantom positions are clearly observable here as well.

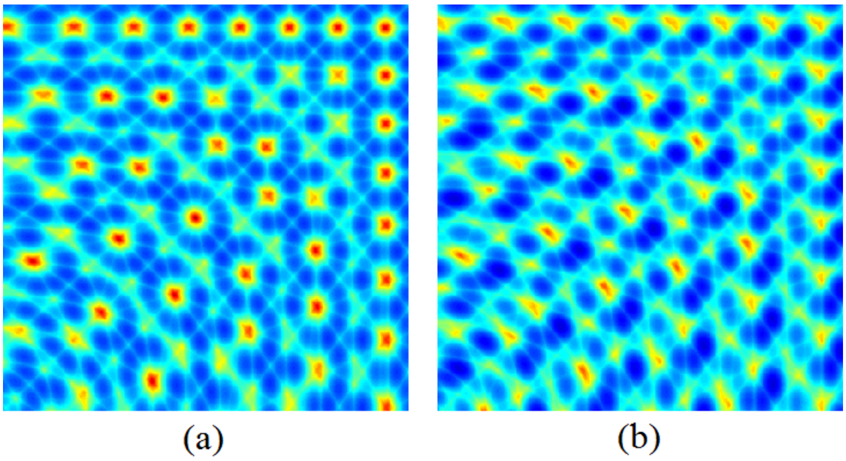


Figure 35. Calculated confidence map for (a) noise free and (b) noisy measurements. Red colors show high confidence values, dark blue denotes low confidence values. The true position is at the center, the other peaks represent phantom positions.

An important observation, which is the basis of the proposed algorithm, is the following: the true object location has high confidence value all along the track of the object (exactly one in noiseless case, and close to one in noisy case). The phantom positions, however, change their confidence values, as the object moves along its trajectory. This phenomenon is quite salient when a series of confidence maps is observed: the peak, corresponding to the true object position, is moving along the object trajectory; at the same time the phantom peaks fade and new phantoms appear, only few phantoms living longer than a few meters.

The phenomenon is illustrated in Figure 36, where the confidence values, corresponding to the true and a phantom position, are denoted by red and blue lines, respectively. Note that in noisy case the confidence value of a phantom position may be higher than that of the true position. The phantom's confidence, however, will eventually decrease. Thus the tracking algorithm, described in the next section, monitors the (true or phantom) trajectories, and keeps only those, which have steadily high confidence values.

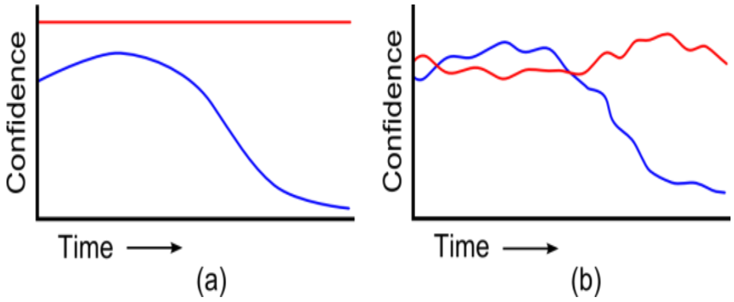


Figure 36. Illustration of phase confidence values at the real (red line) and a phantom (blue line) position of a moving object, as a function of time. (a) ideal, noise free case, (b) noisy case.

4.2. Basic confidence-map based tracking

The input of the Basic Confidence-map based Tracking (BCT) algorithm in each time instant k ($k = 1, 2, \dots, N$) is the measured phase vector $\vartheta_{meas}(k)$, where the vector contains C phase measurements, corresponding to the utilized configurations, according to (21). The output of the algorithm is the actual track list (*atrack*), which ideally contains one and only one track. At the beginning of the algorithm, however, several possible starting points are identified: the true one and many phantoms. As the object moves and new measurements are available, the algorithm checks whether the current tracks can be continued, according to the new measurements, or not. A track can be continued if a possible location (true or phantom) is close enough to the end of the track. The required maximum distance is defined in variable *limit*. Tracks which cannot be continued (thus proved to be phantom tracks) are removed from the actual track list and are stored in list *phtrack*. The list of the actual tracks is thus shrinking, as the moving object provides more and more information to resolve ambiguities, and finally contains only the true track alone. The pseudo-code of the algorithm can be seen in Figure 37.

The helper functions in the algorithm are the following:

`confidence_map(phase_values)`: calculates the confidence map for a given phase measurement set, corresponding to one time instant. See Figure 35 for illustration of a confidence map.

`possible_positions(map)`: analyses the confidence map and determines possible positions. In the current implementation a hard threshold *confmin* is used to select the high peaks in the map, then a blob analysis is run to determine the connected areas, finally the center of each area is selected as the possible position.

`min_distance(p, pv)`: from a vector of points *pv* selects the closest point to a point *p*. Returns both the closest point and the distance.

```
input: ϑ_meas(k), k=1..N
output: atrack, phtrack
Initialization:
atrack = {}
phtrack = {}
map = confidence_map(ϑ_meas(1))
points = possible_positions(map)
for each p ∈ points
   t= new Track
   t.add(p)
   atrack = atrack ∪ t

Tracking:
for each ph ∈ ϑ_meas(2..n)
   map = confidence_map(ph)
   points = possible_positions(map)
   for each t ∈ atrack
      [d, p] = min_distance(t.last_point, points)
      if d < limit
         t.add(p)
      else
         atrack = atrack \ t
         phtrack = phtrack ∪ t
```

Figure 37. Pseudo-code of the Basic Confidence-map based Tracking (BCT) algorithm

4.3. Peak detection anomalies

The proposed BCT algorithm in Section 4.2 tracks all of the possible positions as long as the corresponding peaks in the confidence map are high, and disposes those phantom tracks, where the peak value decreases below a limit, thus after a sufficiently long track only the true trajectory remains.

Unfortunately the BCT algorithm lacks the possibility to continue a track if a peak is disappearing for a few measurement rounds due to e.g. erroneous or missing phase measurements or faulty peak detection on the confidence map. In practical situations bad measurements are present, measurement results may be lost due to communication problems, and in such cases the track may be lost. In the next sections two means are proposed to avoid such problems, thus to provide more robust trajectory estimates.

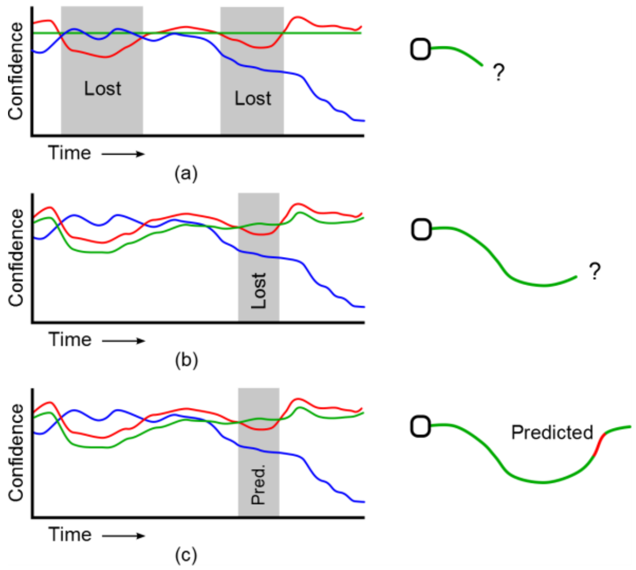


Figure 38. Left column: confidence values of real objects (red), phantom objects (blue) and the thresholds (green); Right column: estimated object trajectories. (a) constant threshold, with two peak-losses, resulting in a track loss, (b) adaptive threshold, still peak loss possible, causing track loss, (c) adaptive threshold with prediction, which tolerates temporary peak losses.

The proposed solution provides an enhanced and more robust tracking method based on a confidence map. The proposed techniques are illustrated in Figure 38. The original method of the BCT uses a fixed threshold. As opposed to the ideal case, illustrated in Figure 36, real measurements often results in true peaks with smaller amplitude, thus the original method may lose tracks, as illustrated in Figure 38(a). The proposed adaptive peak search method, illustrated in Figure 38(b) and discussed in Section 4.4, uses an adaptive threshold, thus the number of peak losses can be decreased (but not necessarily fully prevented). The peaks, which are lost for a short time, would result in complete track loss, which is prevented by the proposed prediction method, described in Section 4.5: if a track seems to end, due to the lost peak, the track it is maintained for a while, using predicted peaks from neighborhood information. Thus temporary peak losses can be tolerated, as illustrated in Figure 38(c).

The new tracking algorithm, using adaptive threshold and predictive track enhancement will be formalized in Section 4.6.

4.4. Adaptive peak extraction

The proposed object tracking algorithm operates on the possible positions, extracted from the peaks of the confidence map in each measurement round, thus the quality of the peak extraction heavily influences the accuracy and robustness of the tracking. In the case of noisy measurements the confidence map can be blurred and may contain multiple smaller peaks rather than one, sharp peak in a possible position. The proposed solution uses binarization with an adaptive threshold.

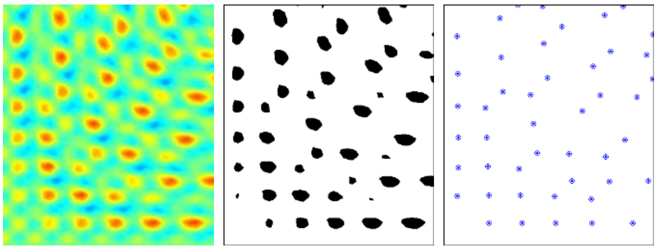


Figure 39. Peak extraction. (a) Confidence map, (b) binary map, (c) peak position estimation.

The proposed peak extraction is illustrated in Figure 39. The algorithm uses the confidence map, as input (see Figure 39(a)). In the first step high peaks are selected using the threshold, resulting a new binary map where peaks higher than the threshold are represented as blobs, as shown in Figure 39(b). In the second step, shown in Figure 39(c), possible positions are determined as the geometric centers of each region, thus eliminating the issues with blurred or multiple peaks.

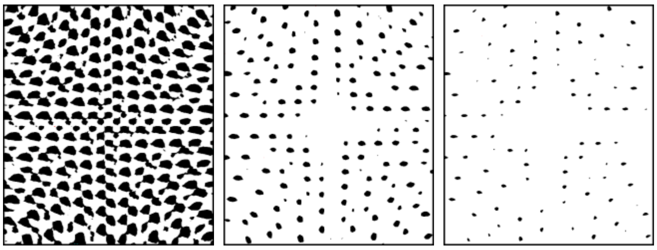


Figure 40. Peak selection with too low (a), correct (b), and too high (c) threshold.

The key part of the extraction method is the threshold level selection. An illustrative example is shown in Figure 40, with peaks selected with too low, correct, and too high thresholds. In the case of low threshold values of separate regions may merge (see Figure 40(a))), while too high threshold values may miss peaks (see Figure 40(c)). In real cases the confidence values highly depend on the measurement noise and vary in each measurement round. Thus a constant threshold values, as in Figure 38(a), should be avoided.

The proposed dynamic threshold value selection (shown on Figure 38(b)) is based on the histogram of the confidence map. A typical histogram and cumulative histogram can be seen in Figure 41(a) and Figure 41(b), respectively. The dynamic threshold value is determined as the value where 90% of the confidence map values are lower than the selected value, as shown in Figure 41.

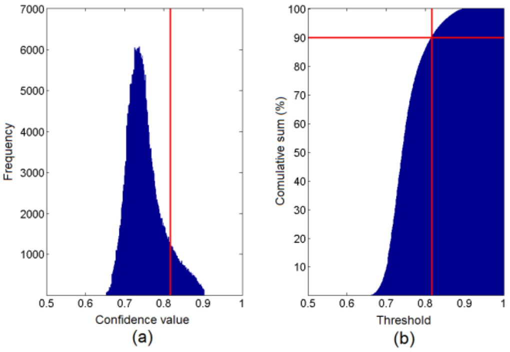


Figure 41. Typical (a) histogram and (b) cumulative histogram of a confidence map. Red lines show the selected threshold value.

4.5. Confidence map-based peak prediction

The proposed peak prediction method enhances the tracking robustness by predicting the peaks' movements based on the confidence map itself. There are time instances where the confidence value of a possible position can go below the threshold value as can be seen on Figure 38(b), which makes the associated track impossible to follow. The BCT algorithm presented in Section 4.2 marks the track as dead and stops tracking the real or phantom object positions in these cases.

In the enhanced solution three track attributes are used: alive, alive but invalid, and dead. Alive tracks become invalid, when the peak is lost, but kept alive for a limited number of measurement rounds. For invalid tracks the position is estimated from the last known or estimated position (i.e. peak location).

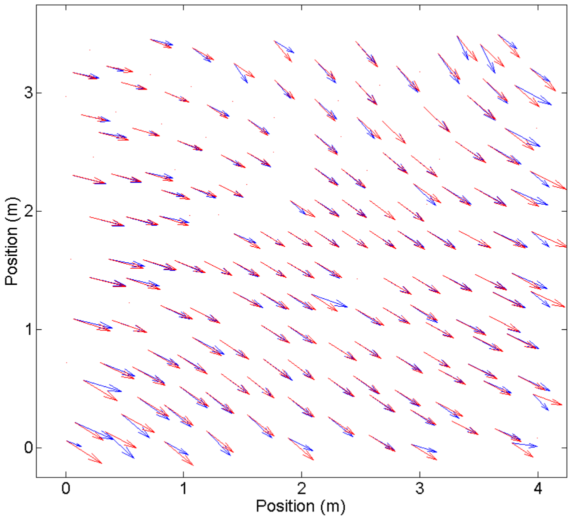


Figure 42. Illustration of the predictive peak tracking. Blue arrows show the calculated (true) movement vectors of the peaks in one iteration, red arrows show the predictions for the same vectors. For better visibility the vectors are enlarged.

The proposed solution focuses on solving the recovery problem; making the tracks alive for a longer period and assign them to the appropriate possible position. The key idea can be seen in Figure 42; possible positions of a peak can be estimated in consecutive measurement rounds using the confidence map. The displacement of a lost peak can estimated from the displacement of other peaks in the close vicinity. This idea is illustrated and also justified the by Figure 42, where the movement vectors, calculated from a real measurement, are approximately the same in a small neighborhood. Thus the proposed solution predicts the displacement of a given point P as follows:

$$\overrightarrow{P_{pred}} = \frac{\sum_{k=1}^{n} \overrightarrow{A_k B_k}}{n}, \left\| \overrightarrow{PA_k} \right\| < L \tag{22}$$

where A_k and B_k are the initial and the end points of a nearby movement vector, calculated from two consecutive confidence maps, and n is the number of vectors within a specified distance L.

4.6. Prediction-aided, confidence-map based tracking algorithm

The pseudo-code of the prediction-aided, confidence-map based tracking algorithm (PCT) is shown in Figure 43. The input of the proposed tracking algorithm in each time instant k $(k = 1, 2, \ldots, N)$ is the measured phase vector $\vartheta_{meas}(k)$. The vector length varies depending on the on the number of the utilized configurations (see Figure 8). The outputs of the algorithm are the lists of active, inactive and dead tracks; *aTrack*, *iTrack*, *dTrack* respectively (see lines 1-2). Ideally the active tracks list contains only one element which belongs to the real object movement. At the beginning of the algorithm new tracks are created for each of the possible positions, containing the real object position and several phantom positions (lines 4-13). In each of the following measurements rounds the new possible positions are calculated and the movements of the tracks are computed (lines 14-54). First the new confidence map is calculated and the peak positions are identified (lines 15-18). Then each active track is checked whether a detected peak can be regarded as the follow-up of the track (line 23). If a track can be followed then it will be marked as active and the measured new position is used as the last position of the track (lines 24-28). If a track cannot be followed (due to a lost peak) then the new position will be predicted and the track will be marked as inactive (lines 30-33). Inactive tracks are similarly handled in lines 37-50. At the end of the iteration the dead tracks are separated from the inactive tracks (lines 51-53). The current solution marks a track dead if it is inactive for N_{dead} consecutive iterations.

```
input: phases(k), k=1..n
output: aTrack, iTrack, dTrack

function TrackPositions

aTrack = {}
fTrack = {}
dTrack = {}

map = genConfidenceMap(phases(1))
points = possiblePositions(map)
for each p ∊ points
  t= new Track
  t.add(p)
  aTrack = aTrack ∪ t
endfor

for each ph ∊ phases(2..n)
  prevPoints = points
  map = genConfidenceMap(ph)
  points = possiblePositions(map)
      newAtrack = newAtrack ∪ t
    else
      p = predict(ap1, t.last)
      t.add(p)
      t.invalid = 1
      newITrack = newITrack ∪ t
    endif
  endfor

  iTPos = collectPoints(iTrack)

  ap2 = assignPoints(iTPos,
     points\usedPoints, PREDICT_LEVELS)

  for each t ∊ iTrack
    if isAssigned(ap2, t.last)
      p = getNewPoint(ap2, t.last)
      t.add(p)
      t.invalid = 0
      newATrack = newATrack ∪ t
    else
      p = predict(ap1, t.last)
```

Figure 43. Pseudo code of the prediction-aided tracking algorithm

4.7. Evaluation

In this section the proposed BCT and PCT methods will be evaluated using simulations and real measurements. In the simulations and in the first real measurement a 4-by-4 meter area was used where the four infrastructure nodes were placed into the corners, i.e. the fixed nodes were placed at positions (0, 0), (0, 4), (4, 0), and (4, 4), respectively. In these experiments six configurations were used, corresponding to configurations C1…C6 in Figure 8.

4.7.1. Simulations

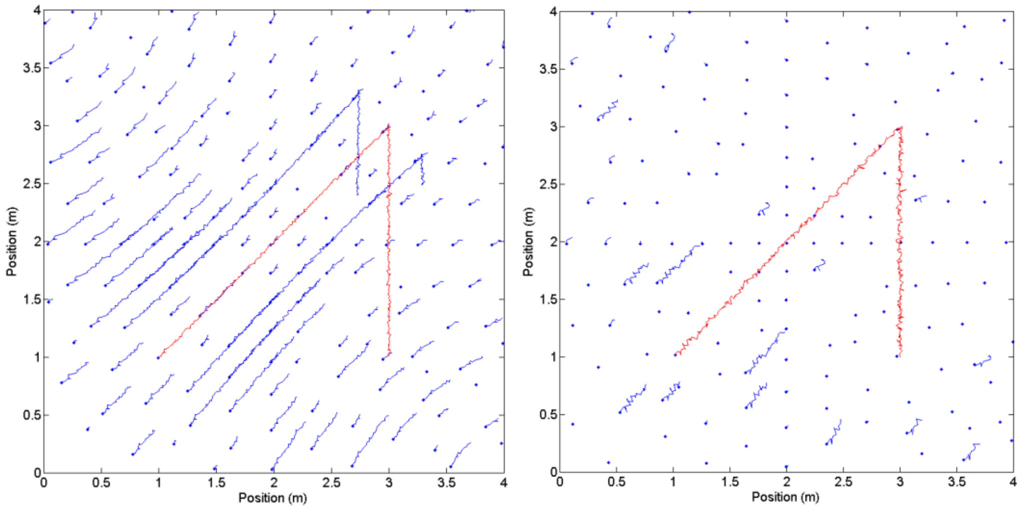
The proposed BCT algorithm was tested with a simulated object trajectory, which started from position (1, 1), moved to (3, 3) and then moved to (3, 1). To the ideal phase values various amount of additive phase noise was added to simulate noisy measurements. In the two experiments zero-mean normal distribution noise was used with $\sigma = 0.1\pi$ and $\sigma = 0.2\pi$, respectively. In both simulations parameters *confmin* and *limit* were set to 0.8 and 0.1m, respectively.

The results of the tests can be seen in Figure 44, where red lines represent the identified true object track, while blue lines show the phantom trajectories. Blue dots show the starting track positions.

As can be seen in Figure 44, from the initial tracking positions phantom tracks of various lengths were detected. Note that the directions of the real and phantom tracks were approximately the same. Also note that in the noisier simulation the phantom tracks are much shorter, because the same *confmin* threshold for lower confidence values (see Section 4.4) results an earlier abortion of phantom tracks.

The length of the true trajectory is $N = 400$. With $\sigma = 0.1\pi$ the five longest phantom tracks have 285, 228, 116, 108, and 98 points, while with $\sigma = 0.2\pi$ their corresponding lengths are 35, 35, 25, 19, and 19 points. The number of starting points somewhat decreased in the noisier experiment from 164 to 143. This is again due to the fact that fewer initial points exceeded the same confidence limit.

The accuracy of the position estimation was also evaluated in the simulations. In the $\sigma = 0.1\pi$ case the maximum, average and the standard deviation of the estimation error are 25.2mm, 8.3mm, and 18.6mm, respectively. In the $\sigma = 0.2\pi$ case the maximum, average and the standard deviation of the estimation error increased to 59.9mm, 17.0mm, and 19.1mm, respectively.



(a)

(b)

Figure 44. Simulation result of the BCT algorithm with various measurement phase noise and constant threshold. The standard deviation of the additive noise was (a) $\sigma = 0.1\pi$ and (b) $\sigma = 0.2\pi$. Active and phantom object trajectories are shown with red and blue lines, respectively. Blue dots present the initial starting positions of the tracks.

4.7.2. Measurements

To test the proposed BCT method a real measurement was also performed, utilizing the measurement system proposed in Section 5. During the measurements the 868 MHz ISM frequency band was used and the radios were fine-tuned with the available 312.5 Hz accuracy. The synchronized receivers performed phase difference measurement on the RSSI data, sampled with frequency of 62.5 kHz.

In the test, shown in Figure 45, four nodes and six configurations were used, as was described at the beginning of Section 4.7. Each of the devices were placed 1.25m above ground level and the tracked node was carried in hand by a person. The test took 50 seconds while 271 rounds were measured (altogether 1626 phase measurements were performed).

The BCT algorithm was executed with parameter values of *limit* = 0.1m, and *confmin* = 0.6. Initially 169 possible locations were found.

The length of the actual track is 271, as shown in Figure 45, with red line. The longest five phantom tracks have lengths of 86, 71, 69, 69, and 69 steps.

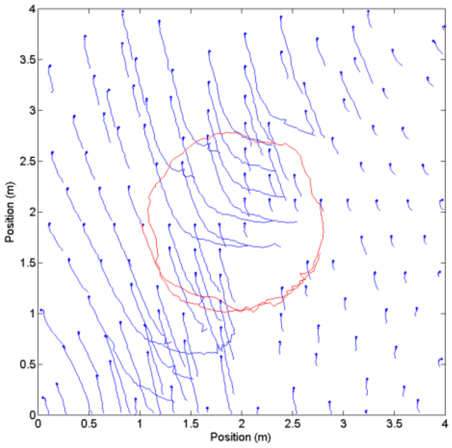


Figure 45. Output of the tracking algorithm based on a real measurement. The computed real object track is shown by red line, while the phantom tracks are shorter blue lines. The initial track positions are denoted by blue dots.

The performance of the adaptive peak extraction and the peak movement prediction utilized in the proposed PCT algorithm was also tested with a real measurement. In the test four infrastructure nodes and one tracked node were used. The algorithm utilized all the twelve possible configurations *C1-12* (see Figure 8). The test was performed in a 4.5 m by 4.5 m room where the infrastructure nodes were placed in the corners of a 3 m by 3.5 m rectangle, 2 m above the ground level (at coordinates (0, 0, 2), (0, 3, 2), (3.5, 0, 2), and (3.5, 3, 2) in Figure 46). The moving node was carried in hand by a person (probably causing small deviations in the range of a few dozens of millimeters). The tracking results of the proposed algorithm can be seen in Figure 46, where phantom and real tracks are shown in blue and red, respectively. With no prediction (Figure 46(a)) the tracking failed after a few seconds because of noisy measurements. With the same measurement data the proposed algorithm performed well and was able to follow the movement, since the missing estimates were successfully replaced by predictions (see Figure 46(b)). In the test the parameter N_{dead} was set to five.

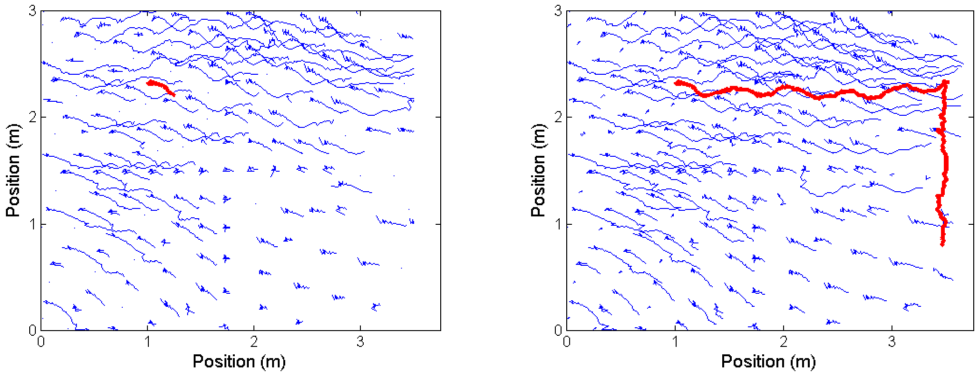


Figure 46. Tracking results of the measurement (a) without and (b) with peak prediction.

Figure 47 shows the reference trajectory in magenta and the estimated trajectory by dots. The maximum deviation from the reference track is 91 mm, part of which is probably caused by the carrying person.

The estimated and predicted values in Figure 47 are shown in blue and red dots, respectively. It is interesting to note that at the magnified part of the track the predictions follow very well the true estimated trajectory, despite the fact that the trajectory changed its direction while only the predicted values were available. Such prediction, based on models on the movement itself, would be very troublesome, but the proposed prediction scheme using the peaks in the neighborhood handles the situation properly.

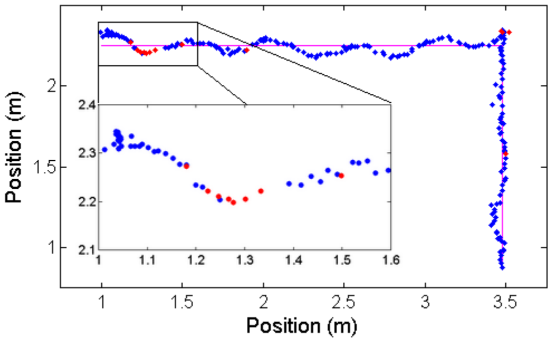


Figure 47. The longest track of the measurement. The estimated and predicted positions are shown with blue and red dots, respectively. The true track is shown in magenta.

In order to evaluate the performance of the prediction subsystem, another experiment was carried out. In each time-instant k of the previous experiment, a one-step prediction was made from the current estimated position $p(k)$, according to (25). The predicted position is $\hat{p}(k+1) = p(k) + \overrightarrow{P_{pred}}$. The difference between the predicted and measured positions in time instant $k+1$ is $\Delta(k+1) = \hat{p}(k+1) - p(k+1)$. Figure 48 shows values of $\Delta(k)$ as function of time, in red color. For easier reference, Figure 48 also contains values of the real displacement $p(k+1) - p(k)$, show in blue. The average displacement (average of the blue curve) is 0.023 m, corresponding to an object speed of approximately 0.25 m/s. The average prediction error is 5 mm, while the maximum error was 21 mm. Clearly the proposed prediction scheme is able to accurately replace the estimates when they are not available.

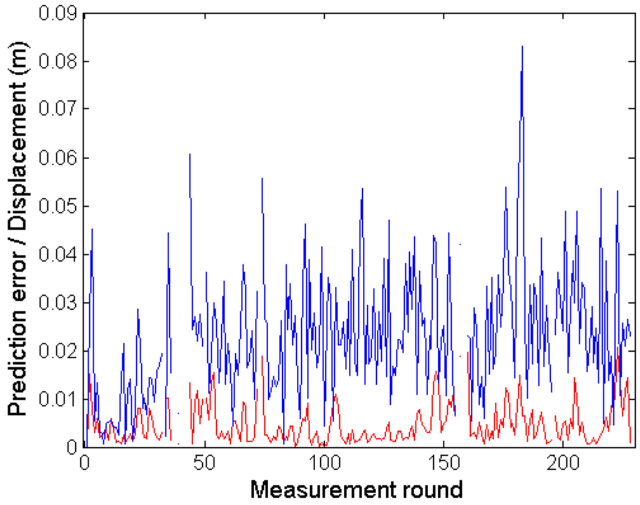


Figure 48. Prediction error (red values) and the lengths of displacement vectors (blue values) of the real object track

4.8. Conclusion

The experimental results show that the proposed PCT method is capable of tracking a moving object equipped with a sensor node device. In contrast to RIPS [13] or SRIPS [14], the proposed tracking algorithm utilizes only one measurement frequency, resulting shorter measurement times for each measurement round. Thus the phase difference measurement errors, caused by the object movement, can be minimized and the proposed method is able to provide a robust tracking, while preserving the few centimeters of accuracy of the interferometric-based localization techniques.

5. Distributed Phase Difference Measurement

Phase difference measurement of interfering radio signals is an essential part of radio-interferometric positioning applications. A measurement method was proposed for radio-interferometric localization in [13], which can perform phase and frequency estimation on the recorded RSSI values. The algorithm is divided into online and post processing parts. The online part is executed on each of the 256 samples. The raw signal is filtered with a moving average. First the minimum and maximum values are determined from the leading 24 samples, which is long enough to contain at least one cycle. Then an upper and lower threshold is set at the 20% and 80% of the calculated range. Peaks are detected as the center points between two consecutive high level crossings. The post processing phase works only on the online detected and filtered peak indices. The phase of the RSSI signal is estimated as the average phases of the peaks.

The application field has numerous challenges, which are discussed in detail in Section 2.5. The proposed solution aims to provide several improvements over the current solution [13]. The targeted improvements and requirements are the following:

a) the measured signal is highly variable: improved adaptive signal processing is required to handle the various potential signal shapes (e.g. logarithmic sine wave)
b) measurement devices are wireless: improved economic data communication is required to lower the communication demand and possibly improve the synchronization
c) measurement devices have limited capabilities: novel distributed signal processing algorithm is required, which can run on low-end devices
d) fast measurements are necessary to allow tracking of moving objects: improved signal processing is required to provide more frequent measurements
e) the communication bandwidth is limited: transmitted data packets must be small, the distributed processing must be tailored to meet this demand
f) repeated measurements are necessary for tracking: the overhead for measurement control must be low

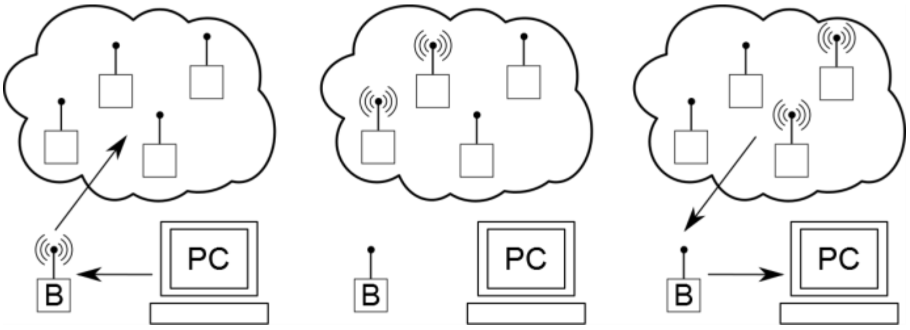
Following the above requirements, a novel phase measurement system is proposed, which is able to accurately measure the phase difference of the received radio signals, even in the presence of significant measurement noise. Due to the distributed nature of the suggested measurement system, it can be operated on inexpensive elements with very limited computing capabilities (e.g. on 8-bit microcontrollers). The performance of the proposed system is illustrated with real measurements results.

5.1. System overview

In this section the measurement system is presented, including the measurement setup and the hardware and software architectures.

5.1.1. Measurement setup

The localization application requires phase difference measurements between predefined pairs of nodes (see Section 3.1.3). Thus the measurement application must schedule and perform phase difference measurements, and finally the results must be forwarded to the localization algorithm. The measurement setup is shown in Figure 49.



(a) (b) (c)

Figure 49. Measurement setup for measuring phase differences: (a) broadcast of measurement and synchronization command, (b) data collection and phase estimation, (c) transmission of measurement results

The measurement network contains several measurement nodes, which can perform phase estimation, and a central base station, which is connected to a PC. The base station can transmit commands to the measurement network and can receive results from the network. In the first step (see Figure 49(a)) the base station broadcasts a measurement command, which has dual role: it contains scheduling information (the IDs of nodes, which have to perform the measurement; the starting time of measurement; and the schedule for transmission of the results), and also supports the synchronization of the selected nodes. All measurement nodes receive the measurement command, and the selected nodes perform the measurement, according to the received schedule (see Figure 49(b)). Finally the measurement results are transmitted back to the base station, where the post-processing is performed, as shown in Figure 49(c).

5.1.2. Hardware

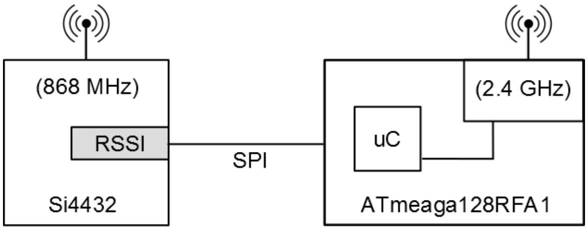


Figure 50. Block diagram of the measurement device

The block diagram of the utilized hardware is shown in Figure 50. The device contains an Si4432 radio chip, which is used to receive the interference signal. The device also features an integrated ATmega128RFA1 microcontroller-radio unit, which is used to perform computations and communicate with the base station.

Figure 51. Dual-radio interferometric phase measurement device

The photo of the measurement node, including a metal housing, two antennas, and the UCMote dual-radio mote [45], is shown in Figure 51.

5.1.3. Measurement software

The block diagram of the software, running on the measurement device is shown in Figure 52. The software is built of modules, using the TinyOS operating system [46]. The blocks shown in grey are part of the operating system, while the white blocks were developed to support the measurement device. The 2.4GHz radio is connected to the communication module, which receives commands and transmits the results. Based on the received commands the measurement is configured, including measurement scheduling and reply scheduling. During the measurement, first a sample buffer is filled with RSSI measurements, then the phase estimation algorithm is executed. The results are transmitted according to the reply schedule.

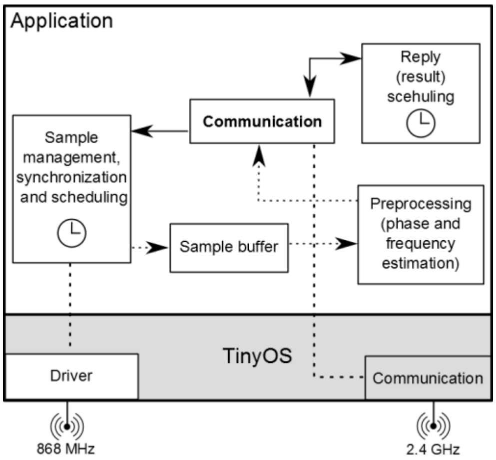


Figure 52. Software block diagram of the measurement device

5.2. Phase difference estimation

The proposed phase difference estimation provides solutions for the various challenges presented in Section 2.5: it offers fast computation in a distributed manner and the issues of the phase difference calculation are addressed by accurate clock synchronization.

5.2.1. Phase estimation algorithm

The fundamental operation of the phase estimation algorithm can be seen in Figure 53. First RSSI samples are collected and stored in the sample buffer with a fixed sampling frequency. The phase estimation algorithm works in two consecutive loops on the buffered samples. The first loop calculates the average of the signal (magenta line on Figure 53) and the lower and upper hysteresis limits (red lines on Figure 53) are determined as ±*Hist* from the average, where *Hist* is a predefined value. This way the signal is divided to four zones with the two hysteresis limits and the average, as shown in Figure 53. The second loop utilizes a state machine, which changes state only when zone transitions occur with a defined sequence. Thus the state machine follows the signal through zones 1-2-3-4-3-2-1-… The reference points of the signal Ref_i (marked with vertical black lines on Figure 53) are determined using the times of transitions between zones 2 and 3, as follows:

$$Ref_i = \frac{t_{32}(i)+t_{23}(i)}{2}, \tag{23}$$

where t_{23} and t_{32} are the time instants of transitions 2-3 and 3-2, respectively, measured from the start of the signal. Note that t_{23} and t_{32} are not suitable reference points, because their time depends strongly on the actual signal shape (see Figure 58 for real signals illustrating this phenomenon).

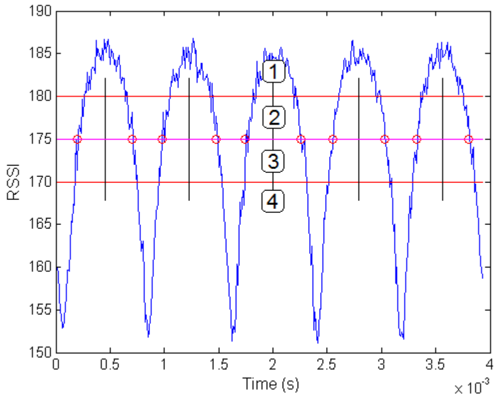


Figure 53. Illustration of the phase measurement: magenta line: mean value, red lines: hysteresis limits, 1-4: amplitude ranges, red circles: 2-3 and 3-2 transitions, and black vertical lines: estimated times of the reference points

The phase of a signal can be defined as follows:

$$\varphi = 2\pi \frac{t}{T} \tag{24}$$

where T is the period length and parameter t is the is the time between a time synchronization point and a signal reference point (see Figure 6).

The proposed phase calculation is performed to estimate the period length T and parameter t in (27). Period length is determined as

$$T = \frac{\Delta}{N-1} \tag{25}$$

where N is the number of reference points (thus the number of full periods is $N-1$) and Δ is calculated as

$$\Delta = \sum_{i=2}^{N} Ref_i - Ref_{i-1} = Ref_n - Ref_1 \tag{26}$$

Notice that the following equation holds for each Ref_i:

$$Ref_i = Ref_1 + T(i-1) \tag{27}$$

Let us define a parameter S as the sum of the distances between each Ref_i and the synchronization point (i.e. that is the beginning of the recorded signal):

$$S = \sum_{i=1}^{N} Ref_i \tag{28}$$

Using (30) and (31) the estimate of parameter t is calculated as

$$t = \frac{\left(S - \frac{(N-1)*N}{2} * T\right)}{N} \tag{29}$$

which can be simplified to

$$t = \frac{S}{N} - \frac{\Delta}{2}. \tag{30}$$

The phase estimate can be calculated as follows, using (27), (28), and (33):

$$\varphi = 2\pi \frac{S/N - \ \Delta/2}{\Delta/(N-1)} \tag{31}$$

Then the phase difference $\Delta\varphi_{i,j}$ between nodes i and j, is calculated from the phase estimates φ_i and φ_j:

$$\Delta\varphi_{i,j} = \left(\varphi_i - \varphi_j\right) mod\ 2\pi\ . \tag{32}$$

Phase estimator (34) contains a division, which is either imprecise or slow operation on low-end devices. To overcome this problem a distributed solution is proposed.

5.2.2. Distributed calculation

The distributed phase estimator is illustrated in Figure 54. The low-end measurement devices compute only integer variables N, S, and Δ, using fast integer arithmetic, according to (29) and (31). The variables are then sent to the base station in the result message. The calculation of the phase difference estimate is performed on the PC, according to (34) and (35).

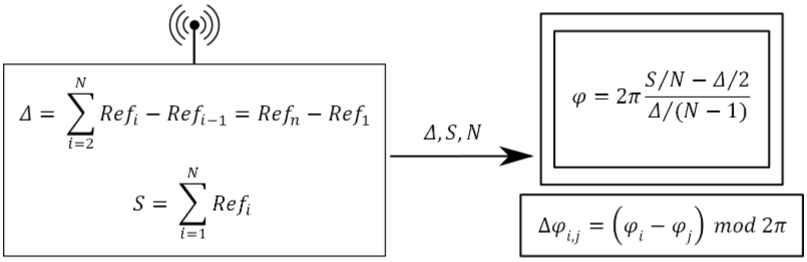


Figure 54. Operation of the distributed phase estimator

Sending intermediate results also ensures lower bandwidth utilization. Sending the phase and period time values as floating point numbers would require 8 (2x4) bytes, while only 5 bytes are necessary for sending $\Delta,\ S,\ N$ as 16, 16 and 8 bit length values, respectively.

5.2.3. Measurement synchronization

The synchronization between the measurement devices and the scheduling of a single measurement is illustrated in Figure 55. The command message, sent by the base station, contains the node IDs of the devices which should perform the measurement (in the example i and j) and the result transmission delays (T_i^{trans} and T_j^{trans} in Figure 55). When the message is received, the nodes create timestamps $T_i^{timestamp}$ and $T_j^{timestamp}$, according to their local clock. The low-layer time-stamping guarantees, that the time stamp is accurate, independently on the various software induced delays [47]. Each node starts two timers, which provide delays, starting from $T^{timestamp}$. The first timer, with delay t^{start}, starts the sampling, followed by the signal processing (*Proc*. in Figure 55) while the second timer initiates the transmission of the results, with delay t^{trans} (*Send result* in Figure 55).

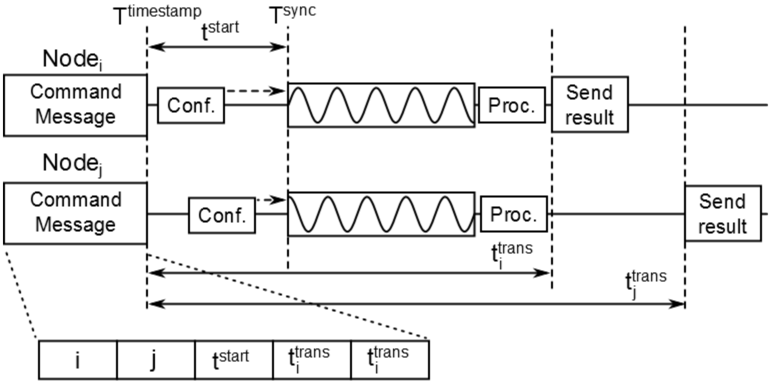


Figure 55. Measurement synchronization and scheduling

The synchronization mechanism guarantees, that the sampling is started on both devices at the same time (when the local clocks show $T^{sync} = T^{timestamp} + t^{start}$). The different result transmission time allow collision-free communication between the measurement devices and the base station.

An additional and more precise synchronization can be achieved by utilizing the interferometric signal itself. The basic concept can be seen in Figure 56, which utilizes a special, scheduled on-off switching sequence of the transmitters (“TX1” and “TX2”). First, the receivers and the transmitters are “roughly” synchronized with the time stamping method described above. In the designated time the receivers start to record the RSSI values, meanwhile the transmitters are still switched off. In this initial state the buffer contains values reflecting the background radio noise. In the next step the first transmitter is switched on (see Figure 56: “TX1: On”), which causes a sudden and significant increase in the RSSI values. The rising edge is sharp enough (in our hardware equipment) to be utilized as a synchronization point. The synchronization point T_{sync} is defined as the first sample which exceeds a predefined limit, shown in red in Figure 56. (In the current implementation the limit is a simple constant threshold, set experimentally.) Later the second transmitter is switched on, causing the generation of the interference signal. At the end of the measurement sequence both transmitters are switched off.

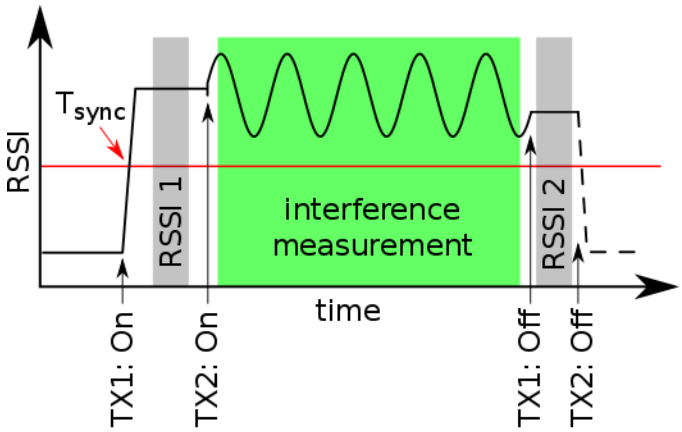


Figure 56. Received signal based time synchronization, using the on-off switching sequence of the transmitters: the recorded RSSI values in the buffer are shown as black line, the threshold used to detect the synchronization point shown as red line, time intervals used to calculate the average RSSI of the individual transmitters are shown in gray, time interval used to perform phase and frequency estimation is shown in green.

Note that the presented switching sequence not only allows to define a synchronization point T_{sync}, but also enables to collect and average individual RSSI values from both of the transmitters (see grey intervals in Figure 56). This information can be utilized as additional quality indicator of the interference measurement and also can be used to identify problems: large difference between RSSI1 and RSSI2 indicates poor measurement results, low RSSI measurements may indicate transmitter problems (e.g. low battery).

In our equipment the achievable time synchronization accuracy with time stamping is ±1 tick with 32.5 kHz clock source, i.e. ±30 microseconds, while the accuracy of the presented method is ±2 samples with a sampling speed of around 110 kHz, i.e. ±18 microseconds. Note that in a measurement round, where consecutive interferometric measurements are made with different configurations (see Section 3.1.3 and Figure 8) only one time stamped message is enough for “rough” time synchronization, because the presented method, based on the recorded values, can compensate for the clock drift between the measurements. This is advantageous, because it requires fewer messages and further shortens the time between consecutive phase measurements in one round.

5.3. Evaluation

In this section the proposed phase difference estimation will be evaluated using simulations and real measurements.

5.3.1. Simulation

The performance of the phase estimation algorithm was first tested by simulations. In the experiment an input sine-wave was generated and Gaussian noise was added to provide signal to noise ratio of 20dB. The proposed algorithm and the algorithm presented in [13] were compared.

For the simulation 160 phase values were generated between 0 and 2π and for each value 100 independent noisy input signals were created. The phase error values (the differences between the ideal and the estimated phase) were recorded and plotted in Figure 57, for the proposed and the reference [13] algorithms, respectively. The variance of the proposed and the reference algorithms were 0.0228π and 0.0343π, respectively. The superiority of the proposed algorithm is clearly visible.

The reason of the better performance of the proposed method lies in the indirect maximum search. The reference algorithm [13] tries to find the maximum positions of the RSSI signal by looking for the maximum values of the signal where the slope is low, thus producing significant errors due to noise. In the proposed solution the indirect maximum position estimation clearly provides more accurate results, since the signal processing was performed where the signal slope is high.

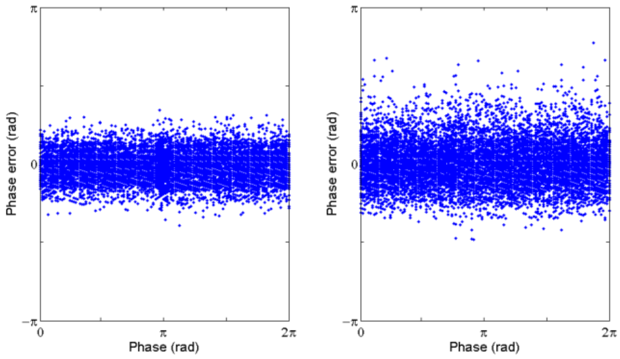


Figure 57. Scattered plot of phase estimation error of (a) proposed algorithm and (b) reference algorithm [13].

Another experiment was performed using logarithmic sine-waves as input signals (see Figure 53 and Figure 58). In this case the performance of the proposed algorithm was similar to the previous experiment, while the reference algorithm failed to provide usable measurements in most cases. The cause of the failures can be explained by the occasional extreme minimum values; the reference method computes the hysteresis limits (20% and 80%) based on the range of minimum and maximum values of the leading samples, thus outlier values can result in oversized hysteresis gap (see Figure 58). In contrast, the proposed solution utilizes the average of the signal, which is more robust against extreme values.

5.3.2. Measurement results

A real measurement setup was built, according to Section 5.1, using two measurement nodes. A typical measured RSSI signals are shown in Figure 58.

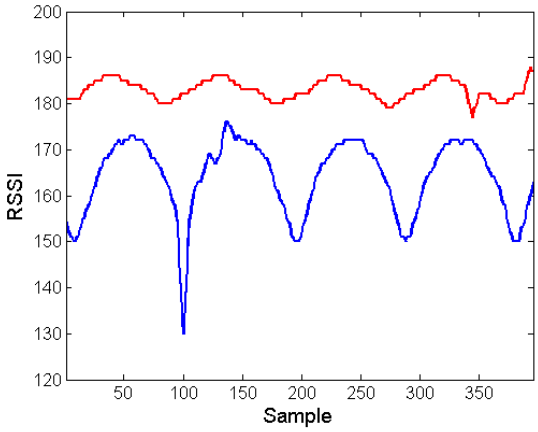


Figure 58. Measured RSSI values by a pair of measurement devices

An experiment was conducted to check the accuracy of the proposed phase difference measurement method. In a static (disturbance-free and outdoor) setup 5000 consecutive measurements were conducted, the histogram of which is shown in Figure 59. The variance of the phase measurements was 0.024π. Note that part of the variance was probably due to environmental effects and noise.

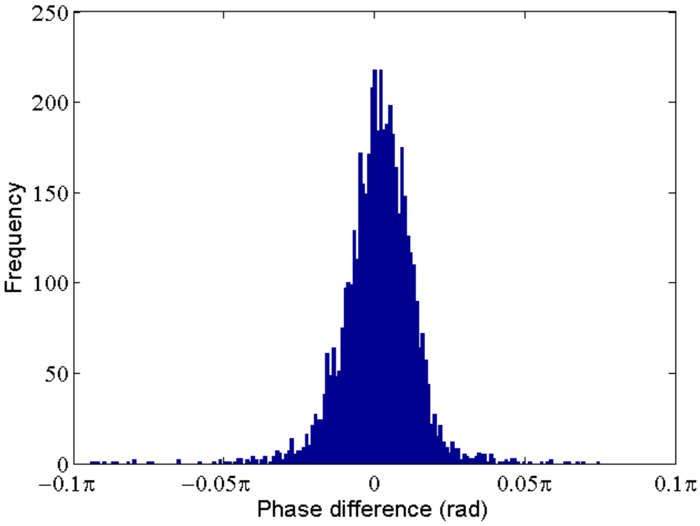


Figure 59. Measured phase error distribution in a disturbance-free, outdoor setup

For radio-interferometric localization and tracking algorithms the characteristics and the variance of additive phase noise is a crucial information, thus the distribution of the phase offset was tested in various environments. The tests utilized static nodes, and had a frequency difference $\Delta f = 595$Hz-744Hz during the measurements. Figure 60(a) shows the disturbance-free, indoor case, while Figure 60(b) illustrates the situation when people were moving around the sensors, using 1500 measurements in each case. As the measurement results clearly indicates the distribution of the phase error can be approximated with a zero mean Gaussian distribution. The standard deviation in the disturbance-free case was 0.03π, which increased to 0.09π when people were present in the measurement space. Figure 60 shows that in both cases the noise level is small enough to perform phase unwrapping, but when disturbance is present, the phase error between two consecutive measurements may be close to π, thus in this case more sophisticated unwrapping techniques should be used, like the one presented in Section 3.2

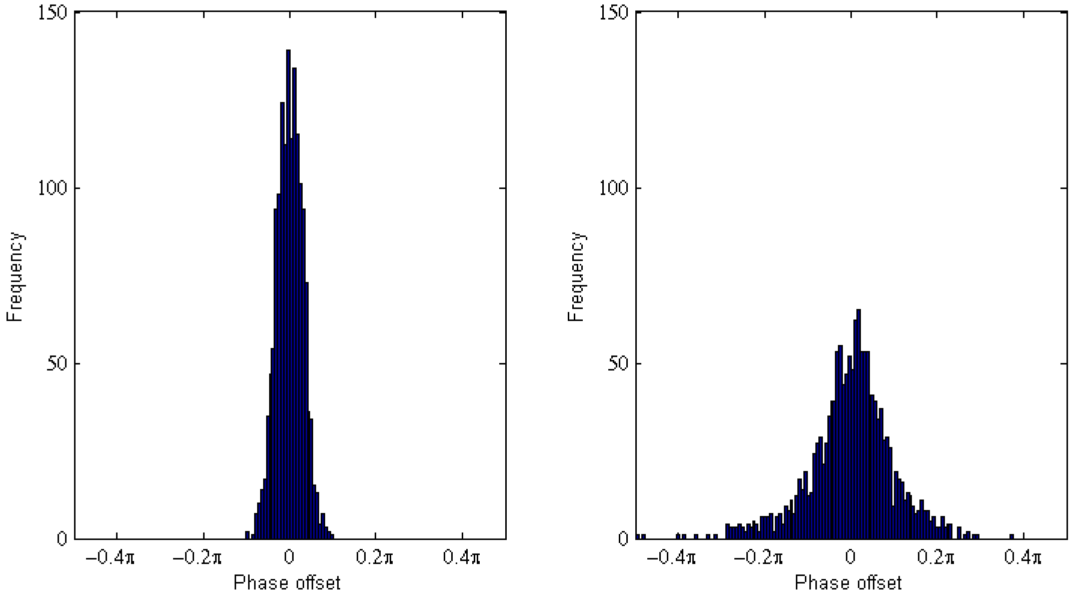


Figure 60. Distribution of 1500 phase-difference indoor measurement: (a) disturbance-free case, (b) human presence and movement

5.4. Conclusion

A measurement method and the corresponding hardware and software infrastructure were proposed, which is able to estimate phase differences between different points of a radio-interferometric space. During a measurement round a pair of synchronized receiver nodes are used, each of which individually records RSSI values and performs pre-processing. The pre-processing algorithm can be executed on low-end measurement devices (e.g. 8 bit microcontrollers). The intermediate results are transmitted through a low-bandwidth channel and the phase difference estimates are calculated at the host computer. Due to the distributed calculation method and the utilized synchronization the proposed measurement setup is able to perform fast and accurate phase difference measurements to support modest real-time requirements of the targeted radio-interferometric tracking applications. The performance of the proposed measurement method was illustrated by simulations and real measurement examples.

6. Long distance VLC-based beaconing for indoor localization

In this chapter the design of an LED-based beaconing infrastructure is introduced, which can be utilized in indoor localization systems. The LED beacons, which may be part of the existing lighting infrastructure, are blinking with high frequency invisible for human eyes, while the detectors are regular cameras. Since the sampling frequency of the cameras is much lower than the blinking frequency, the detection is based on a coding system, which tolerates the undersampling of the signal. The performance of the proposed system is analyzed and a possible application is introduced.

Most of the camera-based decoder systems utilize the rolling shutter effect: since the image sensor samples different rows of pixels at different time instants, high frequency blinking appears in the form of alternating dark and light bands in the picture, which is utilized to decode the transmitted code [12]. The main advantage of these systems is that a single image frame is enough for decoding, but they require large image size for the correct operation, thus the beacon must be close to the camera. Contrary to these solutions, in the proposed solution the detection is performed by utilizing a series of pictures, allowing very small image sizes, and thus distant beacons can also be reliably detected.

The block diagram of the proposed system is shown in Figure 61. The emitted light of each LED is modulated by a LED driver and a unique beacon ID is transmitted by each LED. The modulation utilizes high frequency, providing flicker-free operation. In the detector a static camera is utilized, which takes pictures with approximately 30 fps. The series of images are used to detect those pixels, which correspond to any of the transmitted beacon IDs. The detected pixels are then used to find blobs on the image, each blob corresponding to one beacon. The localization system then estimates the unknown location from the detection image and the a priori information on the beacon positions. Note that the last step (localization) is not in the scope of this thesis, but a case study application will be presented to illustrate the achievable accuracy.

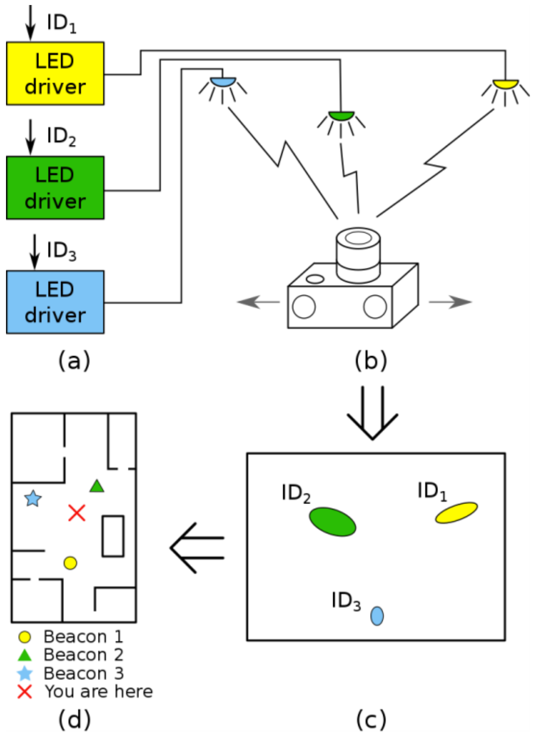


Figure 61. The proposed system architecture: (a) Modulated LED beacons. (b) Wide angle camera. (c) Beacon detection (d) Position estimation.

6.1. System overview

The architecture layers of the proposed long distance VLC beaconing can be seen on Figure 62. The beaconing part physically includes the LEDs and the associated driver circuits. In terms of logical layers, each beacon has a unique ID, which should be encoded to a bit stream, and the channel encoding is an on-off keying with an appropriate carrier frequency and duration. The beacon detection is performed with a camera, where the fundamental operation (sampling) properties are highly utilized in the proposed system. The bit and code detection are performed pixel-wise on consecutive frames.

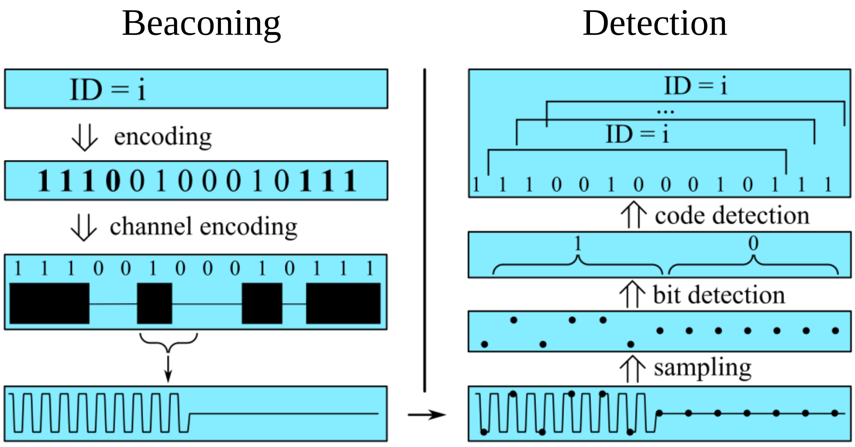


Figure 62. Architecture layers of the proposed long distance VLC beaconing

6.1.1. Beaconing

The beacon ID is encoded as a continuously repeated bit pattern. The codes begin with a header (11100), which is not allowed in any other position of the ID's bit pattern. In the tested system the length of the code (including header) was $N_c = 13$ bits.

The channel encoding is illustrated in Figure 63(a). The encoder generates the on/off sequence and power level settings for the LEDs as follows: during bits *one* the LED is driven at full power and blinked with 50% duty cycle, while bits *zero* are encoded as constant (not blinking) light with half brightness. The applied encoding scheme has the advantage that human eyes cannot see the modulation and flicker-free operation can be provided, if the modulation frequency is sufficiently high. The finite sampling (shutter) time of the camera somewhat limits the maximum usable frequency, but most cameras allow modulation frequencies much higher than 100 Hz. In the proposed system $f_b = 165$ Hz was utilized. The design considerations related to the modulation frequency f_b and the lengths of the pulses, coding the bits, will be discussed in Section 6.2.

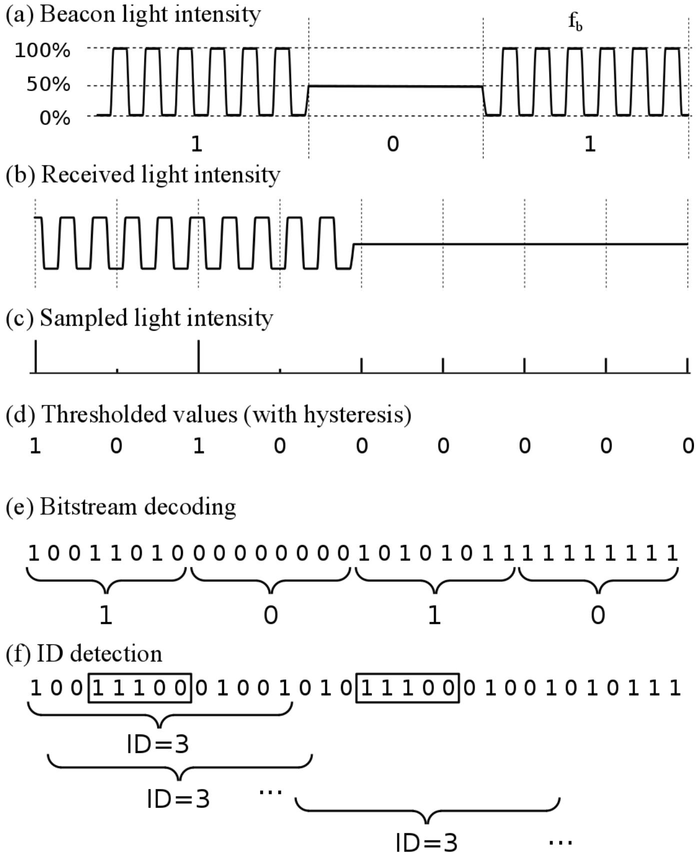


Figure 63. Coding and decoding of the beacon IDs.

6.1.2. Bit detection

The LED beacons are observed with a camera, as shown in Figure 61. Although regular cameras cannot follow high speed light intensity changes, with proper settings suitable undersampling can be provided. With high speed shutter mode (where the shutter time $T_s \ll 1/f_b$) the camera takes short samples from each frame, as shown in Figure 63(b) and (c), thus blinking light provides frame sequences where the corresponding pixels show changes in intensity values. E.g. with $f_b = 165$ Hz a camera with 30 FPS sampling frequency provides 15 Hz alias frequency.

The detector utilizes a comparator with hysteresis to convert each pixel of the detected image frame to a binary value, as shown in Figure 63(d). A state machine is operated for each pixel: it examines whether the last K value of the pixel contains changes or not, and provides the length of the last observed blinking segment T^{D1} (which is the length of a bit 1 or a series of consecutive bits 1), or the length of the last observed static segment T^{D0} (which is the length of a bit 0 or a series of consecutive bits 0). From the output of the state machine the bit values are identified and thus the bit stream is decoded, as shown in Figure 63(e). For the detection the designed symbol length values are used, see later in Table 5. To determine the number of consecutive bits, the designed signal length values (see in Table 5) and the corresponding T^{D1} and T^{D0} values are used (see Figure 65)

6.1.3. ID detection

The code detection algorithm continuously monitors whether the last N_c detected bits correspond to any of the beacon IDs, where N_c is the length of the transmitted code (including the header), as shown in Figure 63(f). The search is performed in parallel, using a code table T_{CODE} with T^{N_c} entries.

The index of a code table entry corresponds to a possible detected bit sequence, while the table contains the corresponding ID (or an invalid symbol). Thus any possible rotation of a code can be detected after the reception of N_c bits, there is no need to wait for the header before decoding.

The detection is performed as follows. For each valid beacon ID a detection image DIM_{id} is stored, initially each pixels being unmarked. $N_{frame} > N_c$ consecutive frames are processed, and in each frame for every pixel, using the last N_c detections Ω_{N_c}, $id =$

$T_{CODE}(\Omega_{N_c})$ is determined: if id is a valid ID, then the corresponding pixel is marked in detection image DIM_{id}.

After N_{frame} samples the detection images are further processed to calculate the beacon positions. In the prototype system N_{frame} was set to 250 frames.

6.1.4. Beacon detection

In the detection image DIM_{id} those pixels are marked, where beacon id was detected. First connected areas (blobs) are detected, then the centroid of each blob is calculated, along with the size of the blob. Blobs with size smaller than B_{min} are ignored, the others are used in the localization algorithm. The minimal blob size in the proposed system were set to $B_{min} = 7$, based on the analysis presented in Section 6.4.

6.2. Parameter design

In this section methods will be presented to design the two key parameters for the system: the symbol length and the blinking frequency.

6.2.1. Code parameter design

Symbol length is one of the key parameters of a data transmission system. To examine the effect of different symbol sizes, a simulator was used. The simulator can virtually emit and observe an arbitrary lighting pattern, with tunable parameters (blinking frequency and camera FPS).

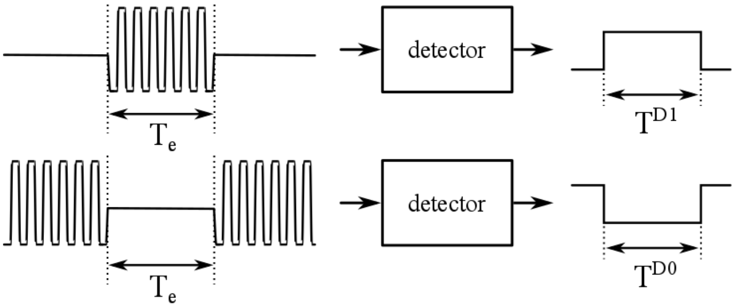
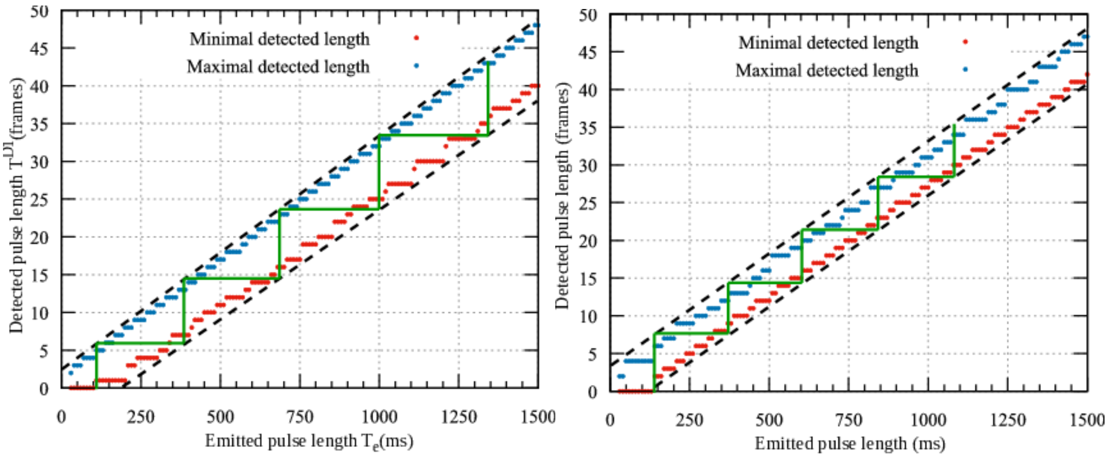


Figure 64. Illustration of the test setup and signals: (left) emitted signals with symbol length T_e and (right) detected 1/0 symbols with lengths T^{D1} and T^{D0}.

The test setup and the test signals are shown in Figure 64. Test signal #1 contains a blinking pattern with length T_e, preceded and followed by 500 ms constant signal segments. Similarly, test signal #2 contains a constant signal segment with length T_e, preceeded and followed by 500 ms blinking signal segments.

In the tests T_e was varied between 30 ms and 1500 ms, and for each length, 1000 different beacon-camera time offsets and 15 different camera FPS setups (between 29 Hz and 31 Hz) were used.

Figure 65 shows the minimal (red dots) and the maximal (blue dots) observed pulse length values for test signal #1 (T^{D1}, Figure 65(a)) and for test signal #2 (T^{D0}, Figure 65(b)).



(a) (b)

Figure 65. Detected length of (a) emitted blinking signals and (b) constant intensity signals. Red dots: minimal detected length, blue dots: maximal detected length.

The minimal usable symbol length T_1^{Dx} is the shortest pulse, which is always detected, i.e.:

$$0 < min(T^{D1}(T_1^1)) \tag{33}$$

$$0 < min(T^{D0}(T_1^0)) \tag{34}$$

Two consecutive symbols must always be different from one symbol, thus the shortest usable length T_2^{Dx} for two consecutive symbols must satisfy the following constraints:

$$max(T^{D1}(T_1^1)) < min(T^{D1}(T_2^1)) \tag{35}$$

$$max(T^{D0}(T_1^0)) < min(T^{D0}(T_2^0)) \tag{36}$$

Similarly, for any consecutive k symbols the following unequality must be satisfied:

$$max(T^{D1}) < min(T^{D1}(T_{\mathrm{k}+1}^1)) \tag{37}$$

$$max(T^{D0}) < min(T^{D0}(T_{\mathrm{k}+1}^0)) \tag{38}$$

To simplify the design and to provide a safety margin, calculations were made using the linearized upper and lower bounds for T^{D1} and T^{D0} (Figure 65 (a) and (b), dashed lines). The design procedure is symbolized with green line segments (Figure 65(a) and (b)), implementing equations (36) -- (41). The design results for symbol lengths of 1 … 5 consecutive 0 and 1 bits are summarized in Table 5.

Table 5. Designed minimal impulse length values for 1 … 5 consecutive 0 or 1 bits (in milliseconds).

k	1	2	3	4	5
T_k^0	140	370	600	845	1080
T_k^1	110	380	690	1000	1340

6.2.2. Error rate vs. blinking frequency

The effect of using different blinking frequencies were demonstrated using the simulator described in Section 6.2.1. Test signal #1 was used, where the blinking frequency was varied between 150 Hz and 180 Hz and three different T_e values (200 ms, 400 ms and 600 ms) were used. The camera FPS was set to 30 Hz. In each test run 1000 different beacon-camera time offsets were simulated. A test run was considered to be successful if exactly one continuous blinking signal (i.e. a series of 1 bits) was observed, other cases (e.g. no detection or incorrect detections of 0 bits in between) were considered erroneous. For each blinking frequency the number of successful and erroneous detections were recorded. Figure 66 shows the error rate vs. the blinking frequency for different pulse lengths (200 ms, 400 ms and 600 ms).

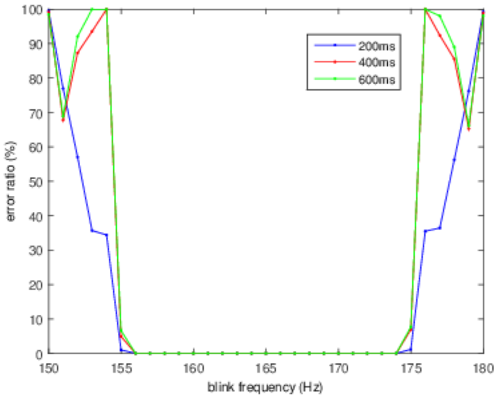


Figure 66. Error rate of detection with 30 FPS camera vs. blinking frequency f_b, for impulse lengths of 200 ms (blue), 400 ms (red) and 600 ms (green).

As Figure 66 clearly shows, blinking frequencies between 156 Hz and 174 Hz (generating a 6-15 Hz alias frequency with the 30 FPS camera), give perfect results. The error rate dramatically increases below 155 Hz and above 175 Hz, where the alias frequency gets closer to the 0 Hz beat.

6.3. Implementation

The proposed design was validated by a proof of concept system, shown in Figure 67. Each beacon contained a 10 Watt power LED, an LM3414 driver, providing constant current with good efficiency, and a pattern generator using an Atmega128RFA1-based sensor network node running TinyOS. The communication capability of the driver was used to implement remote control on the devices, but was not used in the beaconing process itself.

Figure 67. The photo of the pattern generator, LED drivers (in plastic enclosure), and the power LED.

The detection algorithm was implemented as a multithreaded C program. The images were split into slices, each processed by a separate thread. The image loading and processing subtasks were synchronized with a mutex-based barrier method.

In the tests an Intel i5-3210M CPU was used with 4 cores, where 3 cores were reserved for image processing tasks. The implemented detector was able to process the 30 FPS 1080p video stream in real time.

6.4. Evaluation

In this section the performance of the long distance beaconing will be evaluated using real measurements, and an application example will be presented, which utilizes the proposed method.

6.4.1. Detection accuracy

The detection process may result in two types of error: a visible beacon may not be detected (false negative), or a beacon not present may be detected (false positive). To provide immunity against small-sized false positives, a minimal blob size B_{min} is defined: detections with size smaller than B_{min} are ignored.

To determine the optimal value of B_{min}, a test was performed, where 500 measurements were recorded using 4 beacons, placed at various positions, and 16 IDs (including the 4 utilized ones) were used in the detector. Figure 68 shows the frequency of false positives and false negatives (straight red line and dotted blue line, respectively) vs. blob size threshold B_{min}. According to the measurements the optimal value of B_{min} is around 7, where the rates of both the false positive and false negative detections are around 5%.

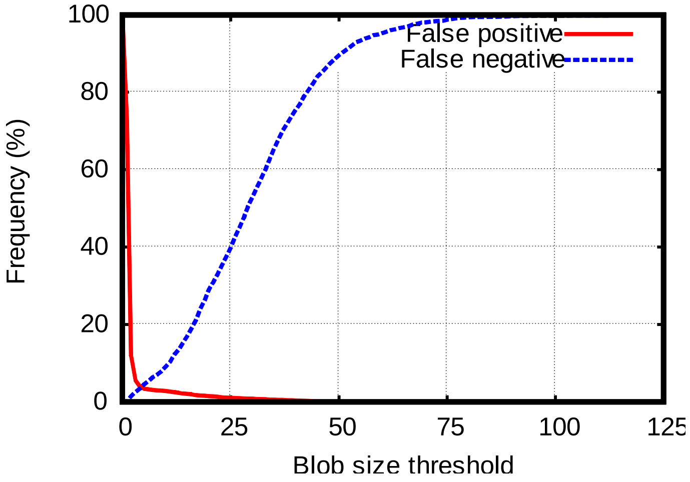


Figure 68. Probability of false positives and false negatives.

6.4.2. Cluster size vs. distance

As results of Section 6.4.1 show, the quality of the detection highly depends on the blob size. To characterize the relationship between the detected blob size and the beacon-camera distance, a measurement was performed. In the experiment the beacon-camera distance was varied between 5 and 50 meters and in each position 50 measurements were made. To model realistic scenarios, the camera and the LED was not facing each other, but rather each device pointed 45 degrees into the camera--beacon axis. In the tests 3 configurations were tested, with different average LED power values (5W, 2.5W, and 1W).

For each measured distance the average, minimum, and maximum detected blob sizes are shown in Figure 69. The detected blob sizes increase as the power increases. Using $B_{min} = 7$, the maximum distance for each power value can be determined from Figure 69: for 1W and 2.5W the maximum distance was around 25m, while for 5W it was more than 30m.

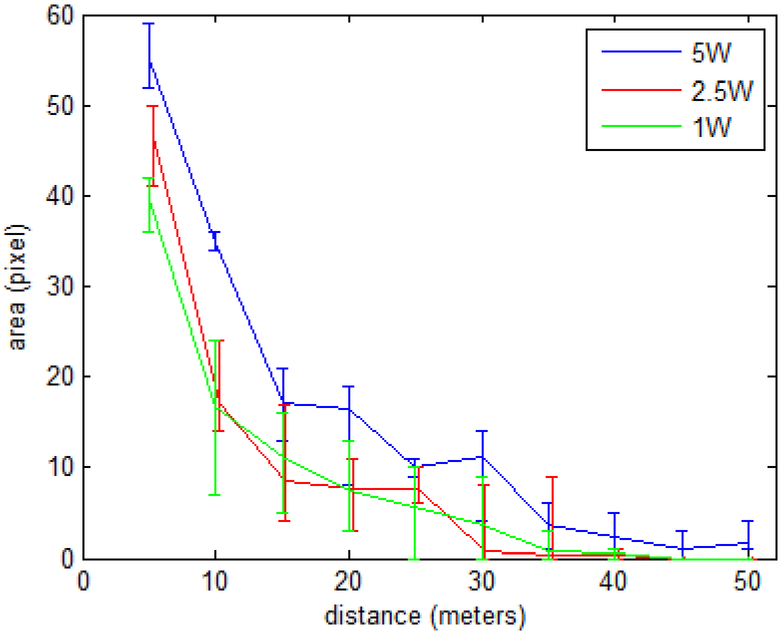


Figure 69. Detected blob size vs. the distance between the beacon and the camera. Blue, red and green curves are recorded with average beacon powers of 5 W, 2.5 W and 1 W, respectively.

6.4.3. Application

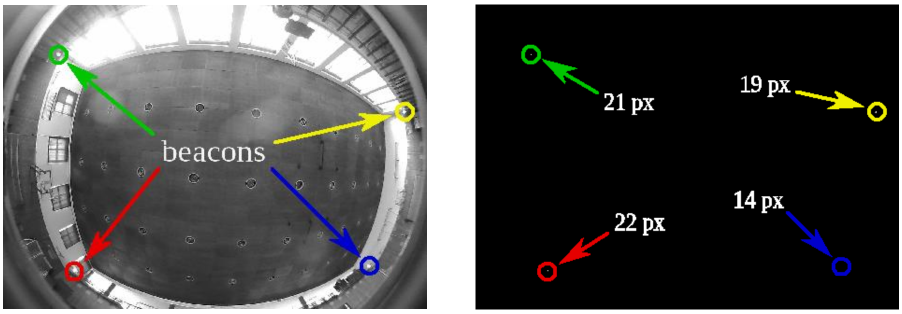
The proposed beaconing method was applied in an indoor localization system [48]. The system utilizes beacons deployed at fixed and known positions, while the camera is to be localized.

The beacons can be part of the existing lighting infrastructure, which is convenient and unnoticeable for the users. Since the detection does not require special cameras, the system can be used with smartphones as well, to provide accurate indoor positioning. In the tests inexpensive wide angle (fisheye) web-cameras were utilized.

The system can utilize arbitrary number of beacons, which are detected using the proposed solution. From the detections and the known beacon positions the actual camera position can be calculated using trilateration or other fusion techniques [48].

A test scenario is shown in Figure 70(a). In this measurement 4 beacons were placed at the corners of a gymnasium of size 17 m x 30 m. The altitude of the sensors varied between 3 and 4 meters.

The image taken by the fisheye camera is shown in Figure 70(a). The beacon positions are marked and the beacons are color-coded for better visibility.



(a) (b)

Figure 70. Localization of the camera in a gymnasium.
(a) Camera image. (b) Detected beacons (ID is color coded) with blob sizes.

Figure 70(b) shows the detection images $DIM_1 \ldots DIM_4$, plotted in one figure, using the same color coding to identify beacons. The sizes of the blobs, corresponding to the detected beacons, are also shown.

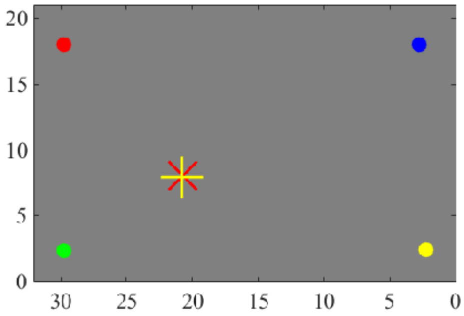


Figure 71. Localization of the camera in a gymnasium. Dots: fixed beacons, yellow cross: estimated camera position, red cross: true position

In Figure 71 the localization result is shown. The figure shows a map, where the known beacon positions are shown with colored dots. The true and estimated positions are denoted by red and yellow crosses, respectively. In the experiment the true camera position was (20.8 m, 7.91 m), while the detected position was (20.73 m, 8.03 m), the error being 0.14 m.

Multiple tests were performed to characterize the accuracy of the localization system. Using 4 beacons and 49 test positions, scattered in the test area shown in Figure 70, the mean absolute error was 0.17 m.

6.5. Conclusion

A beaconing and a corresponding beacon detection scheme was proposed using LED lights and inexpensive cameras. The beacons use VLC to transmit their IDs, using a coding scheme enabling flicker-free user experience. The high frequency blinking of the beacons is detected by ordinary cameras, utilizing proper undersampling of the signal. A method to design the system parameters was presented.

According to measurement results, the proposed detector requires only a small number of image pixels (7 pixels) to provide robust beacon detection and identification. In the presented proof of concept system, using average power of 5 W for the LEDs, the beacons were reliably detected form distances as large as 30 m. The proposed beaconing scheme was illustrated in a localization case study, where the unknown camera location was determined using 4 fixed beacons, with accuracy in the decimeter range.

7. Summary

In this thesis novel methods and algorithms for object localization and tracking services were presented. The result covered algorithms for radio-interferometric tracking, radio-interferometric phase-difference measurements, and a beaconing solution for image-based localization services. In the following chapters the presented solutions will be summarized, compared to applications in the literature and possible application fields will be presented.

7.1. New results

In this thesis novel methods and algorithms for object localization and tracking services were presented. Two radio-interferometric tracking algorithms were presented, the first solution (Section 3) is a geometric based approach, with which a moving target can be tracked from a known initial position. The second solution (Section 4) is based on a novel error function (confidence map) and it can both (1) track an object from a known initial position and (2) determine the full track and the initial position of the object retroactively, in case the method converges to a single solution. The proposed methods provide centimeter accuracy when the speed of the object is low (<1 m/s in the current implementation).

To aid the measurement process in radio interferometric applications, a novel distributed and efficient method was proposed to measure the phase differences. The new solution provides more accurate phase difference measurements than the current solutions, even in case of various waveforms, amplitudes, and noisy conditions. The proposed solution is distributed such that both the computational demand on the low-end measurement devices and the communication bandwidth required to transmit measurement results are low.

In a different application domain (image-processing-based indoor localization systems), a novel method was proposed to create long-distance beaconing, using Visible Light Communication (VLC). Beacons transmit their own IDs using VLC with high enough frequency (>100 Hz) to avoid flickering effect. The novel solution uses undersampling, which allows the utilization of simple sensors (e.g. inexpensive digital cameras with frame rates as low as 30 fps). Since the decoding requires only a few pixels on the image, the beacons can be reliably detected even from long distances.

7.2. Future research

The main bottleneck of current radio-interferometric solutions is the rather limited allowed object speed. The upper limit for the object speed was derived in (13), as a function of the applied wavelength and sampling frequency. According to (13), by increasing the sampling frequency the achievable object speed could be increased. For this, however, new and faster measurement process must be developed, and also the scheduling mechanism of the application must be enhanced.

The confidence-map based tracking solution utilizes ideal phase maps. Real phase maps, however, may significantly be distorted, e.g. around large metallic objects, see e.g. [S11]. This phenomenon may lead to significant bias of the position estimate in certain space regions. Improved solutions may use more precise ideal phase maps (e.g. derived from more accurate models or physical measurements).

The proposed VLC beaconing method is suitable for localization of static objects. A possible improvement of the beaconing scheme is to make detection possible when the tracked object (i.e. the camera) is moving. Also the speed of detection could be improved to provide more responsive systems.

7.3. Application and comparative evaluation

Radio-interferometric tracking does not require the creation of reference maps, as several other radio-signal based methods do, thus the deployment and maintenance of the proposed solutions are easier and more cost efficient. The accuracy of the proposed methods would allow its application in asset tracking or personal tracking applications. The allowed object speed of the current solution, however, is a bottleneck yet.

The radio-interferometric tracking method, described in this thesis, was patented in [S1]. At the time of its invention, the proposed solution was unique in the sense that it could be implemented in low-end, inexpensive devices and its accuracy was still among the bests. In the last few years, however, radio time-of-flight based solutions matured and nowadays they provide similar accuracy with mass-produced inexpensive devices.

VLC-based localization is able to provide viable alternatives to today’s most accurate but still expensive solutions. For applications with high accuracy demand the most accurate solutions utilized 2D or 3D laser scanners, providing accuracy in the mm range. For the cm-range domain the proposed VLC-based method can provide much more economic solutions. The beaconing method, described in the thesis, was utilized in an indoor localization system, which participated in the Microsoft Indoor Localization Competition 2015, where our system proved to be the 6th most accurate system.

In the following paragraphs the proposed radio-interferometric tracking methods and an optical localization solution, using the proposed beaconing method, will be compared to other typical methods, presented in the literature. The selected representative methods include acoustic TDoF, WiFi signal-strength (with and without a priory measurements), dead reckoning, LIDAR, and radio ToF based solutions. The detailed descriptions of the methods can be seen in Section 2.2. The comparison of the methods is made using the following system properties: relationship of localization and tracking, achievable accuracy, hardware requirements and cost, as shown in Table 6. The comparison in Table 6 also shows the utilized basic geometric model, and the need of *a priory* measurement.

Relationship of localization and tracking: Most of the applications presented in the literature perform object tracking by periodical object localization. Thus the exact position is determined in every time instant and the path of the object is defined as the series of the consecutively localized object positions. In contrast, the proposed radio-interferometric methods measure the movement of the object, thus for the exact object location the origin of the path should be known. In that sense the radio-interferometric tracking is similar to odometry and dead reckoning based methods, but with the robust algorithms, presented in Section 3.2, the accumulation of errors can be avoided.

Accuracy: With proper installation (i.e. well established infrastructure) acoustic localization and radio-interferometric tracking can achieve a positioning error in the range of a few centimeters. In [1] the error of the localization is on the same magnitude as the deployment error of the infrastructure devices. Lower accuracy (in the range of 1.5 m – 3 m) can be achieved with signal strength based methods, utilizing a reference map (fingerprinting methods). In [3] the average error was 2.1 m, and 1.85 m with 90% probability. The localization error of the signal strength based methods, utilizing only a mathematical model (i.e. without reference map), is heavily influenced by the environment and the measurement method. The positioning accuracy can be 2-5 m, or in worst cases it is only suitable for rough zone identification. In [2] the reported maximum accuracy was 3 m. LIDAR (laser scanner) based solutions utilize the point cloud derived from the device's precise angular and range measurements. With structural recognition and matching between the point cloud and the known model of the environment the achievable accuracy is in the range of a couple of centimeters [49]. Imaging based positioning methods have an accuracy range from sub-millimeter (e.g. part inspection) to a few meters (e.g. zone identification). The localization error heavily depends on the distance between the localized object and the camera, and also on the utilized image resolution. In [S14] an indoor localization system, utilizing the proposed beaconing method was introduced, which has accuracy in the low decimeter range in large indoor setups.

At the inception of the proposed radio interferometric tracking methods the localization error of time-of-flight based solutions, utilizing radio signals, were 1-2 m or >10 m in cases where the environment heavily scattered the signal. In [5] the reported error was 3.4 m in a room, 5.5 m in a corridor, and 18 m in a coal mine. In the last few years not only the hardware cost has been reduced (due to the mass-production, see [7]), but also the accuracy has highly been increased. In the current methods the achievable accuracy is in the range of a few centimeters [6].

Hardware requirements, hardware cost: Acoustic (mainly ultrasonic) localization methods require special hardware elements; loudspeakers and microphones. The integration of the loudspeakers to small form factor or waterproof devices is challenging and the required power for the driving is also high. The processing of the recorded signal of the microphone also requires specialized hardware or high computation power, thus preventing the application in low-cost devices. Signal strength based and radio interferometric methods require the readily available radio-module of the devices, thus eliminating the additional cost for a specialized hardware component. At the inception of the radio-interferometric tracking, the time-of-flight based radio chips were specialized and expensive, but in the last few years their cost have heavily dropped due to the mass-production, thus radio-modules with integrated time-of-flight measurement hardware are widely available [7]. Imaging applications require cameras and high computational power, thus costly hardware. Note that in several applications smart phones with an integrated camera are used for self-localization. LIDAR based applications require very expensive laser scanners [49].

Table 6. Comparison of object localization and tracking solutions

	Radio-inter-ferometric tracking	**Radio-inter-ferometric localization [13]**	**VLC based optical localization [S14]**	**Laser scanner [49]**	**Acoustic localization [1]**	**Signal-strength based localization [2]**	**Signal strength and reference map based localization [3]**	**Radio ToF localization in 2006 [5]**	**Radio ToF localization in 2016 [6]**	**Pedestrian dead reckoning [10]**
Relationship of localization and tracking	localization ↑ tracking	(for stationary objects only)	localization ↓ tracking	localization ↓ tracking	localization ↓ tracking	localization ↓ tracking	localization ↓ tracking	localization ↓ tracking	localization ↓ tracking	localization ↑ tracking
Basic geometric model	set of hyperboloids	set of hyperboloids	sphere	--	hyperboloids	sphere	sphere	sphere	sphere	--
Special hardware requirement	none	none	yes (low cost)	yes (very expensive)	yes (high power)	none	none	yes (expensive)	yes (low cost)	yes (low cost)
***a priori* measurements (reference map)**	none	none	none	reference model	none	none	reference map	none	none	none
Accuracy	few centimeters	few centimeters (noise free environment)	few centimeters	sub-centimeter	few centimeters (noise free environment)	max: 3 m	avg. err: 2,1 m 90% probability: 1,85 m	indoor: max. 3,4 m corridor: max.: 5,5 m coal mine max.: 18 m	few centimeters	few centimeters (short range)

The methods and solutions of Table 6 can serve for different application fields. Clearly the applicability of a method is mainly determined by the cost and the accuracy of the solution.

A typical application area is person tracking where the accuracy demand is relatively low (few meters or room-level) and at the same time it is very cost sensitive. Thus a suitable solution would use either existing infrastructure or none at all. WiFi based solutions are popular along with dead reckoning methods: these operate on existing hardware with special software. For higher accuracy demands the proposed solutions may be viable alternatives, since the cost of the infrastructure (a few radio-transceivers or LED beacons) is not high and the cost per tracked object (one radio-transceiver or one inexpensive camera) is low, and at the same time the provided accuracy fulfills the requirements of the application domain. Typical applications include tracking of expensive assets in industrial environments, tracking of mobil devices in hospitals, and tracking of persons in either homecare or in hospitals.

Higher accuracy demands arise from autonomous vehicles, where low centimeter-range accuracy is necessary for reliable operation. Current solutions apply expensive LIDAR (laser scanner) devices, where the cost per tracked object can exceed $10.000. In such application areas the proposed solutions (both the radio-interferometric and the optical) can provide the required accuracy in a much more cost efficient way.

8. Summary of Theses

Thesis 1

I proposed a novel radio-interferometric object tracking solution. The proposed system is able to track a moving radio transceiver using infrastructure transceiver nodes deployed in known and fixed positions. The proposed solution is based on the radio-interferometric measurement presented earlier in the literature. The proposed solution is able to track an object from a known initial position, using a series of phase measurements. The method provides centimeter accuracy when the speed of the object is low (<1 m/s in the current implementation).

1.1 I proposed a new object tracking method using a series of radio-interferometric phase measurements. I provided design criteria on the necessary sampling frequency of the phase measurements as a function of the tracked object's speed. The proposed solution has geometric interpretation known in the literature: in 2D, one wrapped phase difference measurement (due to its 2π periodicity) provides a set of hyperbolas, as possible location of the object. From two independent measurements the intersections of the two sets of hyperbolas provide a set of points, as possible position estimates. In the proposed method, if the object position is known in the previous time instant then the new position estimate will be one of the intersections, closest to the previous object position. The proposed solution can be generalized to 3D, using three independent measurements and hyperboloid surfaces.

1.2 To increase the accuracy of the radio-interferometric tracking I proposed an enhanced solution using arbitrary number of independent phase difference measurements. In the proposed solution an error surface was applied, the minimum of which defines the position estimate. The performance of the proposed redundant tracking method was verified using simulations and real measurements.

1.3 I formalized a new Simultaneous Phase Unwrapping problem as follows. Discrete time series of wrapped phase values are determined from M separate measurement channels, where $\varphi_{m,k}$ denotes the principal phase measurement from measurement channel $m, m = 1, 2, \dots, M$, at time instant k. The unwrapped phase values $\phi_{m,k}$ are determined by minimizing $F(\phi_{1,k}, \phi_{2,k}, \dots, \phi_{M,k})$, such that $\phi_{m,k} = \varphi_{m,k} + \Lambda_{m,k}$, where correction parameters $\Lambda_{m,k}$ are integer multiples of 2π. The cost function $F(.)$ describes the high-level (application specific) relationship between the channels.

1.4 I proposed a robust method to compensate for faulty or missing measurements, using Simultaneous Phase Unwrapping. I defined an appropriate error function $F(.)$ for the radio-interferometric application domain. The proposed method is able to detect and automatically compensate for faulty measurements. The performance of the proposed fault-tolerant tracking method was verified using simulations and real measurements.

Related publications: [S1], [S4], [S5], [S6], [S10], [S12]

Thesis 2

I proposed a novel radio-interferometric object tracking solution in 2D. The proposed solution uses multiple series of measurements to (1) track an object from a known initial position or (2) determine the full track and the initial position of the object retroactively, in case the method converges to a single solution. In case (2) the exact conditions of the convergence are not known. The proposed solution uses the phase measurement principle applied in Thesis 1, but utilizes a novel error function, derived from the differences of the measured phase values and the ideal phase map, calculated from the positions of the reference devices.

2.1 I defined a new error function to support radio-interferometric tracking. For *M* independent measurement channels, *M* error maps are calculated as the difference of the ideal phase maps and the actual measurements. The composite error function over the full search space is constructed as a squared sum of the error maps. From the composite error function a confidence map was created where high values show high likelihood of the estimated position.

2.2 Using a confidence map I defined a method to track an object from a known initial position. If the object position is known in the previous time instant and the measurements are performed with a sampling frequency allowing unambiguous phase determination then the new position estimate will be the peak of the confidence map, which is closest to the previous object position. The performance of the proposed tracking method was verified using simulations and real measurements.

2.3 Using the confidence map I proposed a novel solution to calculate the full track of the object from an unknown initial position. The proposed method is based on the phenomenon that along the real track the confidence values are always high. The novel method follows every possible track until its confidence value decreases below a threshold. When the method converges to a single solution, the last surviving track contains the full path of the object, including its initial position. Although the exact conditions of the convergence are not known, due to the complexity of the problem, the performance of the proposed tracking method was verified using simulations and real measurements.

2.4 I defined an adaptive solution to calculate the peak positions of the confidence map. The peaks are located using a threshold, derived empirically from the actual statistical properties of the confidence map. The advantage of the proposed solution is that it provides more robust tracking than the method using constant threshold, when non-ideal measurement conditions are present. The performance of the proposed adaptive peak extraction method was verified using real measurements

2.5 I proposed a new predictive method to compensate for path losses, caused by faulty measurements. The proposed solution does not delete a track when its confidence value decreases below the threshold, but rather the track is suspended. In suspended state the position is predicted using the detected movements of nearby peaks of the confidence map. When the confidence level of a suspended track exceeds the threshold again, it becomes active. A track is deleted if it stays in suspended state for a time longer than an empirical time limit. The robustness of the proposed predictive method was verified using real measurements

Related publications: [S2], [S4], [S6], [S7], [S8], [S11]

Thesis 3

I proposed a novel distributed and efficient method to measure the phase difference in radio-interferometric tracking applications. The proposed solution allows accurate phase difference measurement even in case of various waveforms, amplitudes, and noisy conditions, occurring in real situations. An additional advantage of the proposed method is that the calculations required for phase estimation are distributed such that both the computational demand on the low-end measurement devices and the communication bandwidth required to transmit measurement results are low.

3.1 A proposed a novel method to estimate the phase of periodic signals with respect to a reference. Since in this application domain the shape and amplitude of the signal, measured in different places of the space by different devices, may differ significantly, the peak positions of the signal were selected as invariant points for phase measurement. Unfortunately, when the SNR is low, peaks are hard to detect accurately, thus in the proposed solution the peak positions are estimated using the steep rising and falling edges of the signal, exploiting the symmetry of the signal. The phase estimation is performed by using multiple detected peak positions.

3.2 I proposed a distributed calculation method to support phase and phase difference measurements in low complexity measurement devices. On the low-end devices only the most necessary operations are performed using integer arithmetic. The partial results are transmitted using wireless communication and are further processed in the higher complexity central computer using floating point arithmetic. The size of the transmitted partial results is even smaller than the size of the final result. Thus the proposed solution has low computational demand on the measurement devices and at the same time requires low communication bandwidth.

3.3 I proposed a distributed measurement system to support synchronized radio-interferometric measurements. The devices of the proposed system utilize the measurement method described in 3.1, the phase difference estimation process utilizes the distributed calculation method described in 3.2, and the whole measurement process is controlled and synchronized by a central controller unit, using time division multiple access scheduling. The performance of the proposed system was verified using simulations and real measurements.

Related publications: [S3], [S7], [S9], [S12]

9. References

[S1] S. Dóra, M. Maróti, G. Simon, G. Vakulya, G. Zachár: "***Eljárás és rendszer mozgó objektum pozíciójának rádióinterferencia-alapú meghatározására (Method and system for determining the position of a moving object based on radio-interference)***," P1300645, Nov. 8, 2013.

[S2] G. Zachár, and G. Simon, "***Radio-interferometric Object Trajectory Estimation***," In Proc. of the 3rd International Conference on Sensor Networks, SENSORNETS '14, Lisbon, 2014, pp. 268-273

[S3] G. Zachár, and G. Simon, "***Robust Phase Estimation for RSSI signals***," 11th European Conference on Wireless Sensor Networks, EWSN '14, Oxford, 2014 (poster)

[S4] G. Zachár, and G. Simon, "***Robust Sensor Fusion for Radio Interferometric Tracking***," 11th European Conference on Wireless Sensor Networks, EWSN '14, Oxford, 2014 (poster)

[S5] G. Zachár, G. Simon, M. Maróti, "***Radio Interferometric Object Tracking***", in Proc. 8th International Conference on Sensing Technology (ICST 2014), Liverpool, 2-4 September 2014, pp. 453-458 "***Best Student Presentation, Runner-Up Prize***"

[S6] G. Zachár, G. Simon, "***Novel Approaches for Object Position Estimation in Wireless Sensor Networks***", in Proc. ASCONIKK 2014: Extended Abstracts III, (ASCONIKK 2014), Veszprém, 14-17 December 2014, pp. 66-72

[S7] G. Zachár, G. Simon, "***Demo: Radio Interferometric-Based Indoor Tracking***", 12th European Conference on Wireless Sensor Networks (EWSN 2015), Porto, 9-11 February 2015 (demo) "***Best Demo Award***"

[S8] G. Zachár, G. Simon, "***Prediction-Aided Radio-Interferometric Object Tracking***", 4th International Conference on Sensor Networks (SENSORNETS 2015), Angers, 11-13 February, 2015

[S9] G. Zachár, G. Simon, "***Phase Difference Measurement of Radio-Interference Signals with Low-Cost Devices***", 2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC 2015), Pisa, 11-14 May, 2015

[S10] G. Zachár, G. Simon, "***Radio Interferometric Tracking Using Redundant Phase Measurements***", 2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC 2015), Pisa, 11-14 May, 2015

[S11] Gergely Zachár, Gyula Simon, "***Towards Distortion-tolerant Radio-interferometric Object Tracking***" In Proc. of the 5th International Confererence on Sensor Networks: (SENSORNETS 2016), Rome, Italy, 19-21 February 2016, Portugal: SciTePress, 2016. pp. 207-213.

[S12] G Zachár, G Simon, "***Radio-interferometric tracking using fault-tolerant phase unwrapping***" MEASUREMENT SCIENCE & TECHNOLOGY 27:(4) Paper 045106. 15 p. (2016)

[S13] G. Zachár, G. Vakulya, G. Simon, "***Design of a VLC-based Beaconing Infrastructure for Indoor Localization Applications***" In Proc. of the 2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Torino, Italy, 22-25 May 2017, pp. 1005-1010. 2017

[S14] G. Simon, G. Zachár, G. Vakulya, "***Lookup: Robust and Accurate Indoor Localization Using Visible Light Communication***" IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT 66:(9) pp. 2337-2348. (2017)

[1] T Ajdler, I. Kozintsev, R. Lienhart, M. Vetterli, “Acoustic source localization in distributed sensor networks,” Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004, vol.2, pp.1328-1332, Nov. 2004

[2] N. K. Sharma, "Weighted Center of Mass based Trilateration Approach for Locating Wireless Devices in Indoor Environment," Proceedings of the Forth ACM International Workshop on Mobility Management & Wireless Access (MOBIWAC 2006), Terromolinos, Spain, pp. 112-115, 2006

[3] R. Singh, L. Macchi, C. S. Regazzoni, K. Plataniotis "A statistical modelling based location determination method using fusion technique," Int’l Workshop Wireless Ad-Hoc Networks (In WLAN), 2005

[4] A.W.S. Au et al, “Indoor Tracking and Navigation Using Received Signal Strength and Compressive Sensing on a Mobile Device,” IEEE Transactions on Mobile Computing, 06 Aug. 2012.

[5] S. Lanzisera, D.T. Lin, K.S.J. Pister, “RF Time of Flight Ranging for Wireless Sensor Network Localization,” Proc. 2006 International Workshop on Intelligent Solutions in Embedded Systems, pp.1-12, June 2006

[6] A. Alarifi, et al., "Ultra Wideband Indoor Positioning Technologies: Analysis and Recent Advances, " Sensors, Vol. 16., no. 5, p. 707, 2016.

[7] DecaWave DW1000 homepage. Online: https://www.decawave.com/products/dw1000

[8] A. Mulloni, D. Wagner, I. Barakonyi, D. Schmalstieg, “Indoor Positioning and Navigation with Camera Phones,” IEEE Pervasive Computing, vol. 8, no. 2, pp. 22-31, April-June 2009.

[9] Kai Ni; A. Kannan, A. Criminisi, J. Winn, “Epitomic location recognition,” IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008., pp.1-8, June 2008

[10] F. Evennou and F. Marx, "Advanced integration of WIFI and inertial navigation systems for indoor mobile positioning, " EURASIP J. Appl. Signal Process., Vol. 2006, pp. 164-164, 2006

[11] Li, Liqun and Hu, Pan and Peng, Chunyi and Shen, Guobin and Zhao, Feng, "Epsilon: A Visible Light Based Positioning System," Proceedings of the 11th USENIX Conference on Networked Systems Design and Implementation (NSDI’14), Berkeley, CA, USA: USENIX Association, pp. 331-343, 2014

[12] Y.S. Kuo, P. Pannuto, K.J. Hsiao,P. Dutta, “Luxapose: indoor positioning with mobile phones and visible light, ” in Proc. 20th Annual International Conference on Mobile computing and networking (MobiCom '14), Maui, Hawaii, USA, Sept. 7-11, 2014, pp. 447-458.

[13] M. Maroti et al, “Radio Interferometric Geolocation”, ACM Third International Conference on Embedded Networked Sensor Systems (SenSys 05), San Diego, CA, pp. 1-12, November, 2005.

[14] B.J. Dil, P.J.M. “Havinga, Stochastic Radio Interferometric Positioning in the 2.4 GHz Range,” Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems (SenSys 11), Seattle, WA, pp. 108-120, 2011

[15] J.H. Galeti, R.T. Higuti, E.C.N. Silva, C. Kitano, “Nanodisplacement Measurements of Piezoelectric Flextensional Actuators Using a New Interferometry Homodyne Method,” IEEE Transactions on Instrumentation and Measurement, vol.64, no.5, pp.1256-1265, May 2015

[16] C. Wang, J.D. Ellis, “Dynamic Doppler Frequency Shift Errors: Measurement, Characterization, and Compensation,” IEEE Transactions on Instrumentation and Measurement, vol.64, no.7, pp.1994-2004, July 2015

[17] D. Francis, D. Masiyano, J. Hodgkinson, R. P. Tatam, “A mechanically stable laser diode speckle interferometer for surface contouring and displacement measurement,” Measurement Science and Technology, vol. 26, no. 5., 055402 (7pp), Apr. 2015

[18] I. Milesi, M. Norgia, P.P. Pompilio, C. Svelto, R.L. Dellaca, “Measurement of Local Chest Wall Displacement by a Custom Self-Mixing Laser Interferometer,” IEEE Transactions on Instrumentation and Measurement, vol.60, no.8, pp.2894-2901, Aug. 2011

[19] A. Magnani, A. Pesatori, M. Norgia, “Real-Time Self-Mixing Interferometer for Long Distances,” IEEE Transactions on Instrumentation and Measurement, vol.63, no.7, pp.1804-1809, July 2014

[20] K. Ouchi, “Recent Trend and Advance of Synthetic Aperture Radar with Selected Topics”, Remote Sensing 2013, Vol.5, No.2, pp.716-807

[21] G. Prellinger, K. Meiners-Hagen, F.Pollinger “Spectroscopically in situ traceable heterodyne frequency-scanning interferometry for distances up to 50 m,” Measurement Science and Technology, vol. 26, no. 8., 084003 (9pp), July 2015

[22] K-N. Joo, “Dichroic spectrally-resolved interferometry to overcome the measuring range limit,” Measurement Science and Technology, vol. 26, no. 9., 095204 (7pp), July 2015

[23] A. Benlarbi, J.C. van de Velde, D. Matton, Y. Leroy, “Position, velocity profile measurement of a moving body by microwave interferometry,” IEEE Transactions on Instrumentation and Measurement, vol.39, no.4, pp.632636, Aug 1990

[24] G.A. Kyriazis, M.L.R. De Campos, "An algorithm for accurately estimating the harmonic magnitudes and phase shifts of periodic signals with asynchronous sampling," IEEE Transactions on Instrumentation and Measurement, vol.54, no.2, pp.496-499, April 2005.

[25] P.M. Ramos, M. Fonseca da Silva, R.C. Martins, A.M.C. Serra, "Simulation and experimental results of multiharmonic least-squares fitting algorithms applied to periodic signals," IEEE Transactions on Instrumentation and Measurement, vol.55, no.2, pp.646-651, April 2006.

[26] B. Djokic, E. So, "Phase measurement of distorted periodic signals based on nonsynchronous digital filtering," IEEE Transactions on Instrumentation and Measurement, vol.50, no.4, pp.864-867, Aug 2001.

[27] P. Stoica, R.L. Moses, Spectral Analysis of Signals. Upper Saddle River, NJ, USA: Prentice Hall, 2005.

[28] X. Dai, R. Gretsch, "Quasi-synchronous sampling algorithm and its applications," IEEE Transactions on Instrumentation and Measurement, vol.43, no.2, pp.204-209, Apr 1994.

[29] P.T. Gough, "A fast spectral estimation algorithm based on the FFT," IEEE Transactions on Signal Processing, vol.42, no.6, pp.1317-1322, Jun 1994.

[30] A. Ferrero, R. Ottoboni, "A new approach to the Fourier analysis of periodic signals for the minimization of the phase errors," IEEE Transactions on Instrumentation and Measurement, vol.40, no.4, pp.694-698, Aug 1991.

[31] G. Peceli, "Resonator-based digital filters, "Circuits and Systems, IEEE Transactions on , vol.36, no.1, pp.156,159, Jan 1989.

[32] F. Nagy, "Measurement of signal parameters using nonlinear observers," IEEE Transactions on Instrumentation and Measurement, vol.41, no.1, pp.152-155, Feb 1992.

[33] A. V. Oppenheim and R.W. Schafer. Discrete-Time Signal Processing, Chapter 13. Prentice Hall, Englewood Cliffs, NJ, 2009.

[34] J.M. Tribolet, “A new phase unwrapping algorithm”, IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP, Vol. 25, No.2, pp. 170-177, Apr. 1977.

[35] K. Steiglitz and B. Dickinson, “Phase unwrapping by factorization”. IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol.30, No.6, pp.984-991, Dec. 1982.

[36] R. McGowan and R. Kuc, “A direct relation between a signal time series and its unwrapped phase,” IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol.30, No.5, pp.719-726, Oct 1982.

[37] Q. Fang, P.M. Meaney, K.D. Paulsen, “The Multidimensional Phase Unwrapping Integral and Applications to Microwave Tomographical Image Reconstruction,” IEEE Transactions on Image Processing, vol.15, no.11, pp.3311-3324, Nov. 2006

[38] X. Zhao, J. Liu, H. Zhang, Y. Wu , “Measuring the 3D shape of high temperature objects using blue sinusoidal structured light,” Measurement Science and Technology, vol. 26, no. 12., 125205 (10pp), Nov. 2015

[39] D.L. Fried , “Least-square fitting a wave-front distortion estimate to an array of phase-difference measurements,” J. Opt. Soc. Am. Vol. 67, No.3, pp. 370-375, 1977

[40] Dongliang Zheng, Feipeng Da, “A novel algorithm for branch cut phase unwrapping,” Optics and Lasers in Engineering, Vol. 49, No.5, pp. 609-617, May 2011.

[41] C. Zhang, H. Zhao, F. Gu, Y. Ma, “Phase unwrapping algorithm based on multi-frequency fringe projection and fringe background for fringe projection profilometry,” Measurement Science and Technology, vol. 26, no. 4., 045203 (10pp), March 2015.

[42] G.F. Carballo, P.W. Fieguth, “Probabilistic cost functions for network flow phase unwrapping,” IEEE Transactions on Geoscience and Remote Sensing, vol.38, no.5, pp.2192-2201, Sep 2000.

[43] A. Vallan, F. Molinari, “A Vision-Based Technique for Lay Length Measurement of Metallic Wire Ropes,” IEEE Transactions on Instrumentation and Measurement, vol.58, no.5, pp.1756-1762, May 2009

[44] R. B. Langley, “Dilution of precision,” GPS World, vol. 10, no. 5, pp. 52–59, May 1999.

[45] UCMote homepage. Online: http://www.ucmote.com

[46] P. Levis and D. Gay, TinyOS Programming, 1st ed. New York, NY, USA: Cambridge University Press, 2009.

[47] M. Maróti, B. Kusy, G. Simon, A. Lédeczi, "The flooding time synchronization protocol," SenSys ’04, pp. 39–49, 200

[48] "Microsoft Indoor Localization Competition – IPSN 2015" [Online] https://www.microsoft.com/en-us/research/event/microsoft-indoorlocalization-competition-ipsn-2015

[49] Y. Chen, et al., "Knowledge-based indoor positioning based on LiDAR aided multiple sensors system for UGVs," Proceedings of the 2014 IEEE/ION Position, Location and Navigation Symposium, pp. 109-114, 2014