High Resolution Distance Sensing for Mini-Robots using Time Difference of Arrival

George Sineriz*, Michael J. Kuhlman* and Pamela A. Abshire
Department of Electrical and Computer Engineering, University of Maryland, College Park
Email: gsine@umd.edu, mkuhlman@umd.edu, pabshire@umd.edu

***Abstract*—This paper presents an efficient, compact, and robust distance-only sensor for networked small robotic platforms with wireless communication and signal processing capabilities. The sensor determines inter-robot distances by measuring the Time Difference of Arrival (TDOA) between wireless radio frequency packets and audio pulses. Computational overhead has been reduced by an order of magnitude while the sensor resolution has been improved to 0.27 cm over a range of 75 cm, compared to a previous resolution of 1.1 cm. Error analysis identified timing jitter as the dominant contribution to measurement error and a significant factor in heading estimation error using distance-only measurements. The improvement in distance resolution is shown to improve system performance by reducing the motion planning decision error rate due to measurement uncertainty.**

I. Introduction

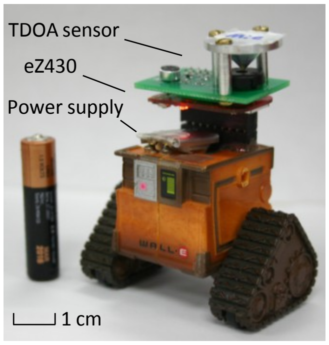

Fig. 1. Miniature robot platform used in the experiments. AAA battery shown for reference.

Distance-only sensors offer miniature robots such as the one shown in Fig. 1 a means to determine the location of other robots in a swarm. The TDOA sensor is realized using a TI eZ430-RF2500 wireless development board, which is also used for communications and motor control, with auxillary audio components as shown in Fig. 2. The eZ430 includes an MSP430 microcontroller and a CC2500 wireless radio chip. An auxiliary TDOA board consists of an omnidirectional microphone (CMC-2742PBJ-A, CUI Inc.), a piezo buzzer (PS1240P02CT3, TDK), bias capacitors and resistors, and headers. It is connected to two internal cascaded non-inverting op-amps of the MSP430 with a total gain $A_V = 130\ \frac{\mathrm{V}}{\mathrm{V}}$. Each board has both a microphone and piezo buzzer for bi-directional distance sensing.

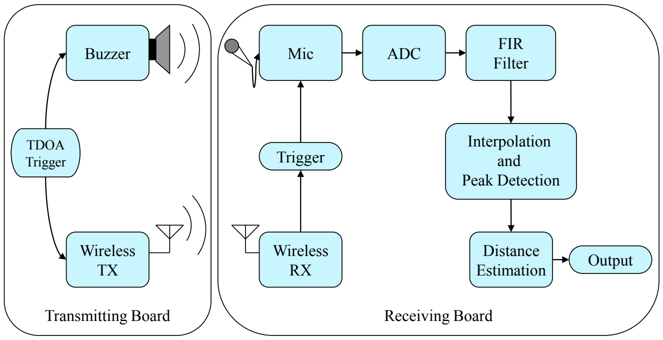

Fig. 2. TDOA system flow chart.

The transmitting board (left, Fig. 1) produces an audio signal through pulse width modulation by utilizing an onboard timer. The 12 kHz audio pulse is emitted by the buzzer. The receiving board (right, Fig. 2) first receives the wireless packet which triggers the microphone to start recording. This analog signal is stored digitally and filtered. Peak detection techniques extract the relevant features of the signal to measure the relative arrival times of the RF packet and audio pulse.

II. Signal Processing

When the wireless signal is detected by the receiving board, the audio pulse is amplified and then sampled by the 10-bit Analog-to-Digital converter (ADC) internal to the MSP430 at a sampling frequency of F_s = 91 kHz. 255 samples are acquired to provide a $\sim$3 millisecond window for capturing the sound impulse within a $\sim$1 m distance. Larger sampling windows are not needed due to significant attenuation of the audio pulse beyond $\sim$0.8 m.

The complete discrete-time sound waveform is filtered using a 32nd order discrete-time FIR filter implemented using Horner's method [1] to remove environmental noise. A bandpass filter with f_{center} = 12 kHz and 3-dB passband edge frequencies of f_p = 12 ± 1.3 kHz was designed to pass the 12 kHz sound pulse emitted by the transmitting board, encompassing the range of frequencies observed during experimentation. This variance is caused by timing jitter in the system clock. In Fig. 3 a sampled signal, along with its Fast Fourier Transform, is displayed before and after filtering.

*These authors contributed equally to this manuscript.

978-1-4673-0219-7/12/$31.00 ©2012 IEEE

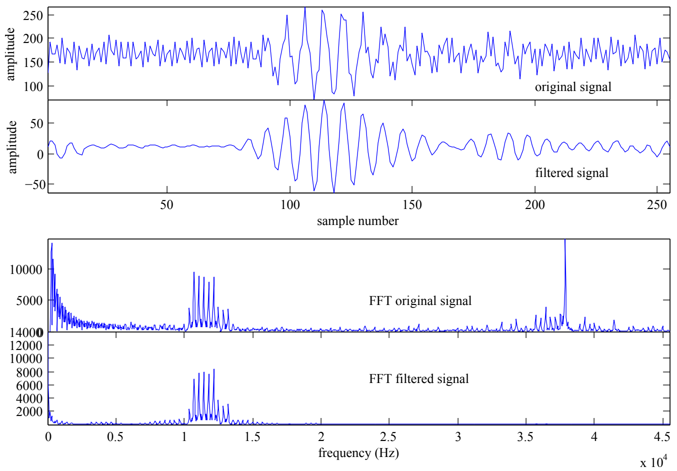

Fig. 3. Results of MATLAB signal processing: Original Signal, FFT of Original Signal, Filtered Signal, FFT of Filtered Signal

III. Distance Estimation

In order to measure distance a unique signal feature must be extracted. The detected audio pulse has a carrier frequency of 12 kHz, and the unique feature is taken to be the peak of the signal. While many sonar systems use the leading edge of the audio pulse as a feature [2], this work uses the signal peak because of its robustness to noise and simplicity of detection. One can either locate the peak of the signal or the peak of the signal envelope. Using the signal peak the resolution would be limited to approximately the wavelength of the carrier signal. For the speed of sound in air at sea level (343 m/s) this resolution is 2.9 cm. However, the signal envelope cannot be directly measured and must be approximated.

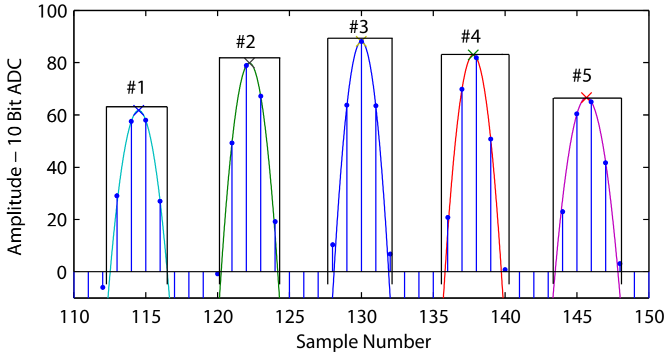

Fig. 4. Enlarged view of audio pulse displaying data regions used for different methods of interpolation, with the maximum amplitude in the center.

To provide subsample resolution, we used Lagrange polynomial interpolation [3] to fit parabolic functions to the signal peaks, approximating the signal envelope between the crests of the carrier signal. Each interpolation uses three (t, x) pairs, but there is ambiguity in selecting the input parameters from the audio signal. Given the known pattern of the signal, we have developed two techniques to address ambiguity and to better represent the signal envelope:

Iterative Interpolation: For a local peak, interpolate the signal locally to better approximate the local peak. The interpolation algorithm is then executed iteratively to find the signal envelope peak using interpolated local peaks. Type A methods do not use this technique, calling the interpolation algorithm once, while Type B methods use this technique, calling the interpolation algorithm four times.

Data Region Selection: Select different local signal peaks relative to the peak of peaks to pass to the interpolation algorithm. We have numbered the relevant data regions containing local peaks accordingly. Each region contains a local peak and its neighbors, with region 3 defined to contain the peak of peaks as in Fig. 4.

We have tested four methods using different selection options guided by the two techniques:

- Method 1: Peak Type A using data regions 2,3,4
- Method 2: Peak Type A using data regions 1,3,5
- Method 3: Peak Type B using data regions 2,3,4
- Method 4: Peak Type B using data regions 1,3,5

The peak of the audio pulse is thus approximated by the peak of the interpolant. A linear (i.e. affine) relationship between the fractional sample number of the signal envelope peak n and the estimated inter-robot distance exists δ with parameters (M_s, S_0) determined empirically from calibration data (least squares regression).

$$\delta(n) = M_s n - S_0 \tag{1}$$

M_s is a slope parameter mapping sample numbers to audio pulse distance travelled. The delay shift S_0 is a consequence of hardware latency (from both the transmitting and receiving boards) and the constant delay imposed by the FIR filter. For calibration, a set of 20 distance readings (in terms of sample number) per distance were obtained at 10 cm increments from 10 to 70 cm (140 measurements total). The calibrated parameters were $M_s = 0.379 \frac{\text{cm}}{\text{sample}}$ and $S_0 = 19.8$ cm, which corresponds to a measured speed of sound in air $v_s = F_s M_s = 345 \frac{\text{m}}{\text{s}}$.

A. Experimental Results

The distance resolution σ_i was taken to be the standard deviation of the measurements δ_i given true distance D_i. Using leave one out cross validation techniques to calculate (M_s, S_0) for a given data subset, errors on all of the measurements in the dataset were calculated, thus computing σ_i for each distance D_i. The mean resolution $\bar{\sigma}$ is the mean resolution over all test distances. For each of the methods studied, Table I shows $\bar{\sigma}$ along with the number of clock cycles required for the FIR filter, Goertzel Algorithm and interpolation to execute. The new FIR-based methods use fixed-point computations, whereas the original algorithm used floating point computation. The new design requires 10% of the execution time with improved accuracy due to interpolation.

Method	FIR-1	FIR-2	**FIR-3**	FIR-4	old [4]
$\bar{\sigma}$ [cm]	0.56	0.36	**0.27**	0.31	1.1
filtering [cycles]	281k	281k	**281k**	281k	3,170k
interpolation [cycles]	7.65k	7.65k	**30.2k**	30.2k	0
total [cycles]	289k	289k	**311k**	311k	3,170k

TABLE I
INTERPOLATION METHOD ACCURACY VS. COMPUTATIONAL EFFICIENCY IN THOUSAND CLOCK CYCLES.

IV. Noise Error Propagation Model

Error analysis was performed to determine the dominant sources of noise in measurement error $\bar{\sigma}$. Error models were developed to map the sensitivity of the interpolated signal envelope peak to variations in timing Δt_i or amplitude Δx_i. One technique to approximate $t_{peak} = f(t, x)$ is by using a linear approximation of the nonlinear function f as in the Extended Kalman filter, where $\Delta t^2_{peak} \approx \mathbf{J}\Sigma_X\mathbf{J}^{'}$ where Σ_X is the covariance matrix. We assume that each of the random variables are independent. Δt_{peak} thus linearly maps to a distance measurement error, $\bar{\sigma}_{noise}$.

A. Flicker Noise Model

$$t_{peak} = f(t, X) \tag{2}$$

Observation of the FFTs in Fig 3 suggests that the dominant feature is flicker noise. The spectral noise density as in Fig 3 was fit using linear regression, then integrated over frequencies of interest to determine the RMS contribution due to flicker noise. We further assume that all x_i are independent and normally distributed, i.e. $\perp X_i$, $X_i \sim \mathcal{N}(x_i, (\Delta x_i)^2)$, and propagate Δx_i through the interpolation algorithm (2) using $\mathbf{J_i} = \frac{\partial f}{\partial X_i}$ to estimate Δt_{peak}.

B. Timing Jitter Model

$$\begin{aligned} t_{peak} &= g(t_1, T_1, T_2, x) \\ &= f(t_1, t_1 + T_1, t_1 + T_1 + T_2, x) \end{aligned} \tag{3}$$

The nondeterministic timing jitter σ_{jitter} of the ADC's built-in oscillator was measured and is 4.7% of the clock period. To ensure independence of random variables, one most reformulate the interpolation equations in terms of the time intervals between peaks instead of the peak times themselves. This assumes that t_1 and the amplitudes of the three points are held fixed but the time intervals T_i between the measurements are independent and normally distributed, i.e. $T_1 \perp T_2$, $T_i \sim \mathcal{N}(t_{i+1}, (\sigma_{jitter})^2)$. The timing jitter σ_{jitter} propagates through (3) using $\mathbf{J_i} = \frac{\partial g}{\partial T_i}$ to estimate Δt_{peak}.

C. Results

120 audio recordings filtered using the FIR filter were analyzed to characterize flicker noise and estimate measurement error. Results are shown in Table II which suggests that timing jitter is the dominant contribution to variations in t_{peak} and $\bar{\sigma}_{noise}$.

4.7% timing jitter	Flicker noise (after filtering)
0.10 cm	0.011 cm

TABLE II
RMS EXPERIMENTAL AVERAGES FOR STANDARD DEVIATIONS.

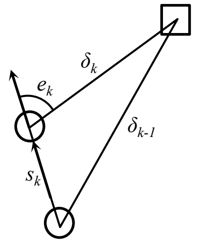

Fig. 5. Law of cosines triangle for heading estimation

V. Application: Heading Estimation on a Miniature Robot

The improved distance sensor results in improved system-level performance when estimating heading between two robots in a swarm using distance-only information. The heading angle (4) between a leader robot and a follower robot can be calculated using the law of cosines in Fig. 5. δ_k is measured using a distance-only sensor and s_k is estimated using odometry/calibration information. Due to the symmetry of the cosine function, there is no unique solution for the heading angle. This is resolved by tracking multiple heading estimates and tracking the turning directions of the robot, using the conditional in (5) to drive the heading angle to zero. Fig. 6 shows an example of a robot trajectory using this method to attempt to rendezvous with its leader as captured by an overhead vision system. The ideal heading estimation triangles have been superimposed onto the trajectory, representing the information available to the robot at a given time.

$$q_k = 1 - \cos(e_k) = 1 + \frac{(\delta_k^2 - \delta_{k-1}^2 + s_k^2)}{2\delta_k s_k} \tag{4}$$

$$\text{If } q_k - q_{k-1} < 0 \Rightarrow \text{Change turning direction} \tag{5}$$

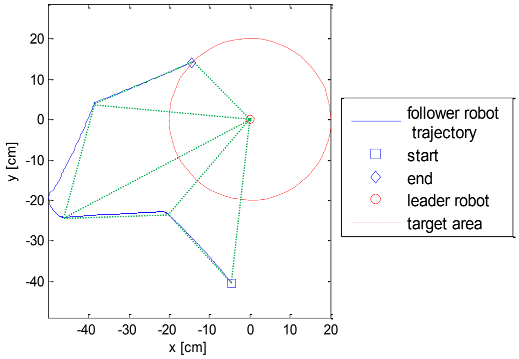

Fig. 6. Sample robot trajectory with ideal heading triangles (dotted green) overlaid

A. Linearized Error Propagation Model

Each of the sides of the triangle in Fig. 5, $(\delta_k, \delta_{k-1}, s_k)$, is a random variable with measurement uncertainty. We apply error propagation analysis as in Section IV, assuming each measurement. $\sigma^2_{s_k}$ was calculated by using a vision system observing the normed displacement of the robot between distance measurements. The robots were programmed to turn for a time interval $\propto q_k$, then move forward 10 centimeters. $\sigma^2_{\delta_k} = \sigma^2_{\delta_{k-1}} = \bar{\sigma}^2$ is the distance sensor resolution calculated by calibration experiments. The new method reduces both the RMS and maximum standard deviations by 27%, as seen in Table III.

sensor	σ_{δ_k} (mm)	σ_{s_k} (mm)	σ_{q_k} (RMS)	σ_{q_k} (Max)
Old TDOA	11	16	0.21	0.78
New TDOA	3	16	0.15	0.57

TABLE III
HEADING ESTIMATION STANDARD DEVIATION σ_{q_k}

One critical area of system performance is the sensitivity of choosing the turning direction using (5). Decision errors fall into two distinct classes when using uncertain measurements, $\hat{q}_k$, instead of the true values q_k to calculate (5). For notational convenience, define $z = q_k - q_{k-1}$ and $\hat{z} = \hat{q}_k - \hat{q}_{k-1}$.

False Positive (FP) Error: The robot changed turning direction when it was not supposed to, i.e. $\hat{z} < 0$ when $z > 0$.

False Negative (FN) Error: The robot did not change turning direction when it was supposed to, i.e. $\hat{z} > 0$ when $z < 0$.

These FP/FN errors are analogous to Type I/II errors in statistical hypothesis testing. Errors of this nature drastically harm the convergence rate of the robot to its desired location since they cause the robot to turn in the wrong direction. Using the variances calculated using the error propagation model, we assume that $\hat{q}_k$ is normally distributed with variance $\sigma^2_{q_k}$, unbiased mean centered around the true value q_k (6). We use the linear error propagation model to estimate the probabilities of these errors occurring given sensor uncertainty.

$$\hat{q}_k - \hat{q}_{k-1} \sim \mathcal{N}(q_k - q_{k-1}, \sigma^2_{q_k} + \sigma^2_{q_{k-1}}) \tag{6}$$

(6) assumes that q_i's are normally distributed and that $\hat{q}_k \perp \hat{q}_{k-1}$. This is clearly not the case since each computation of q_k is dependent on $(\delta_k, \delta_{k-1}, \delta_{k-2})$ and q_{k-1} is dependent on $(\delta_{k-1}, \delta_{k-2}, \delta_{k-3})$. The assumption simplifies the addition of two normal random variables.

B. Experiment Results

The experimental dataset consists of 14 trajectories of a traveling robot using the old TDOA sensor in [4] with the heading estimator. Each trajectory comprises a sequence of true heading error q_i as captured by an overhead vision system, and a sequence of estimated angle error $\hat{q}_i$ calculated on the embedded system using distance-only measurements. The relationships between z and $\hat{z}$ as observed over 14 different robot trajectories are shown in Fig. 7. Each quadrant of this graph corresponds to either a correct decision or FP/FN error.

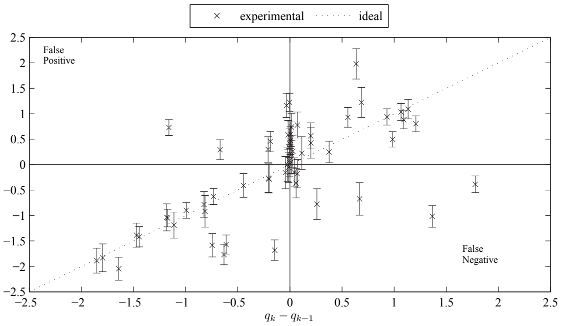

Fig. 7. Comparison of z to $\hat{z}$ highlighting turning decision errors from observed robot trajectories. The system used the old TDOA distance sensor when acquiring this data.

In Fig. 7, the error bars shown for each data point are the estimated measurement standard deviations. The residuals of the dataset were distributed with sample mean 0.007 $\frac{\text{cm}^2}{\text{cm}^2}$ and standard deviation 0.75 $\frac{\text{cm}^2}{\text{cm}^2}$. Empirical probabilities yielding the percentage of data points in each quadrant of Fig. 7 are summarized in Table IV.

Region	Correct	False Positive	False Negative
Fraction in region	0.71	0.16	0.12

TABLE IV
EMPIRICAL DECISION PROBABILITIES FOR THE OLD TDOA SENSOR.

VI. CONCLUSION

The TDOA distance sensor can be used in any system that has wireless communication and signal processing capabilities. Parabolic interpolation of the signal envelope has proven to be an efficient means to reduce distance measurement uncertainty. The effects of distance resolution were analyzed in the system context of heading estimation on a miniature robot. The new TDOA sensor reduces distance uncertainty so that odometry uncertainty is now the dominant factor in heading estimation, whereas both were important before. We are currently working to reduce odometry uncertainty through improvements in calibration and hardware.

ACKNOWLEDGMENT

The authors thank Susana Galicia, the MERIT BIEN program, and the rest of the ANTBOT team. This material is based upon work supported by the National Science Foundation under Award Nos. 0647321, 0755224, and 0931878.

REFERENCES

[1] K. Venkat, *Efficient MSP430 Code Synthesis for an FIR Filter*. Texas Instruments, 2007.

[2] B. Barshan and B. Ayrulu, "Performance comparison of four time-of-flight estimation methods for sonar signals," *Electronics Letters*, vol. 34, no. 16, pp. 1616–1617, 1998.

[3] M. Heath, *Scientific Computing*. McGraw-Hill, 1997.

[4] C. Perkins, L. Lei, M. Kuhlman, T. Lee, G. Gateau, S. Bergbreiter, and P. Abshire, "Distance sensing for mini-robots: RSSI vs. TDOA," in *2011 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2011, pp. 1984–1987.