IEEE Transactions on Robotics and Automation February 1995, Vol. 11, No. 1, pp 132-138 .

Error Eliminating Rapid Ultrasonic Firing for Mobile Robot Obstacle Avoidance

by
Johann Borenstein and Yoram Koren
The University of Michigan
Department of Mechanical Engineering And Applied Mechanics (MEAM)
Ann Arbor, MI 48109-2110
Ph.: (313) 763-1560, FAX: (313) 763-1260

ABSTRACT

This paper introduces *error eliminating rapid ultrasonic firing* (EERUF), a new method for firing multiple ultrasonic sensors in mobile robot applications. EERUF allows ultrasonic sensors to fire at rates that are five to ten times faster than those customary in conventional applications. This is possible because EERUF reduces the number of erroneous readings due to ultrasonic noise by one to two orders of magnitude.

While faster firing rates improve the reliability and robustness of mobile robot obstacle avoidance and are necessary for safe travel at higher speed (e.g., V > 0.3 m/sec), they introduce more ultrasonic noise and increase the occurrence rate of crosstalk. However, EERUF almost eliminates crosstalk, making fast firing feasible. Furthermore, ERRUF's unique noise rejection capability allows multiple mobile robots to collaborate in the same environment, even if their ultrasonic sensors operate at the same frequencies.

We have implemented and tested the EERUF method on a mobile robot and we present experimental results. With EERUF, a mobile robot was able to traverse an obstacle course of densely spaced, pencil-thin (8 mm-diameter) poles at up to 1 m/sec.

Keywords: Ultrasonic sensors, Mobile robots, Obstacle avoidance, Crosstalk, Noise rejection

---

1) This work was sponsored in part by the Department of Energy Grant DE-FG02-86NE37969 and in part by the Veterans Administration Grant 028085.

2) Parts of the material in this paper were presented at the *IEEE Conference on Robotics and Automation*, Nice, France, May 10-15,1992

NOMENCLATURE

a,b Index to alternating delays $T_{wait,a}$ or $T_{wait,b}$.

n,m Index to the *nth* (*mth*) occurrence of an event.

t_0 Start of timing diagrams.

t_{ct} The (absolute) time at which crosstalk occurred.

x Sensor causing crosstalk, located at the beginning of a critical path.

y Sensor affected by crosstalk, located at the end of a critical path.

L Length of a critical path.

T_δ Maximum allowable time difference between any two consecutive readings.

T_{echo} Time from firing to receiving an echo.

T_{err} Erroneous reading, caused by crosstalk.

T_{fire} Amount of time from the beginning of a period to the actual firing of a sensor.

T_{idle} Amount of time from receiving an echo to the beginning of the next period.

$T_{i,nom}$ Nominal time interval between scheduled firings of the sensors in a group.

$T_{i,min}$ Shortest time interval between actual firings of the sensors in a group.

T_{lag} Amount of time a sensor is scheduled for firing after the beginning of a period.

T_p Time period. The amount of time in which each sensor fires once.

T_{wait} The amount of time EERUF waits before firing a sensor, after the sensor was already scheduled for firing.

T_{wind} Time window) the amount of time a sensor is "open" to await an echo.

$T_{L,n}$ Amount of time soundwaves spend on traveling through the critical path.

<u>Abbreviations</u>

EERUF Error Eliminating Rapid Ultrasonic Firing

URS Ultrasonic Range Sensor

1. INTRODUCTION AND BACKGROUND

This paper introduces *error eliminating rapid ultrasonic firing* (EERUF), a new method for noise rejection with ultrasonic range sensors (URSs). The EERUF method is designed to work with the widely used URSs manufactured by POLAROID [19]. A comprehensive discussion of the characteristics and limitations of these sensors can be found in the literature and is omitted here (see [1; 5; 10; 12; 15; 16]).

In order to guarantee complete coverage of the area around a mobile robot in all directions, many mobile robots have URSs installed on their periphery at 15° intervals[1]. For omnidirectional robots of circular shape, this design requires 24 (=360°/15°) URSs mounted on a ring around the robot. Similar designs using 24 URSs in 15° intervals are described in the literature [17; 7; 2; 18; 9; 11; 6] and were used in the previously commercially available robot manufactured by DENNING.

While using multiple URSs reduces the risk of collision, it increases the amount of ultrasonic noise in the environment in two ways:

<u>a) Environmental Noise From Other Ultrasonic Sensors</u>

This type of noise is typically a discrete disturbance. It is very likely to occur when more than one vehicle with ultrasonic sensors operate in the same environment. In this case, interferences may occur over distances of up to 20 m.

<u>b. Internal Noise From Onboard Ultrasonic Sensors) *Crosstalk*</u>

Crosstalk (also called *multipath*) is an undesirable phenomenon in which one sensor receives the ultrasound waves emitted by another sensor. Figure 1 shows a mobile robot equipped with multiple URSs in two typical indoor environments; both environments differ substantially in the way they promote crosstalk. For the following discussion, we define the term "*critical path*" as any path of ultrasound waves that are transmitted by one sensor and are received by one or more others, thus creating crosstalk. The sensor that transmitted the

---

[1] Kuc [14] shows that theoretically it would be necessary to use even denser spacing (e.g., 5°) to cover all possible obstacles. However, we found that in practice 15°-spacing reliably detects obstacles as small as 8 mm diameter vertical poles.

ultrasound waves is labeled **x**, and each of the receiving sensors is labeled **y**.

Figure 1a shows a direct critical path, where the robot is near a single wall. Because of the symmetry in Fig. 1a, two sensors are labeled '**y**,' since they are both on a critical path with sensor **x**. If any sensor **y** fired shortly after sensor **x**, this sensor **y** would be awaiting the echo to its own signal by the time the echo from sensor **x** reaches it. Thus, the reading from sensor **y** would result in some arbitrary error, depending on the time difference, T_{lag}, between firing sensors **x** and **y**.

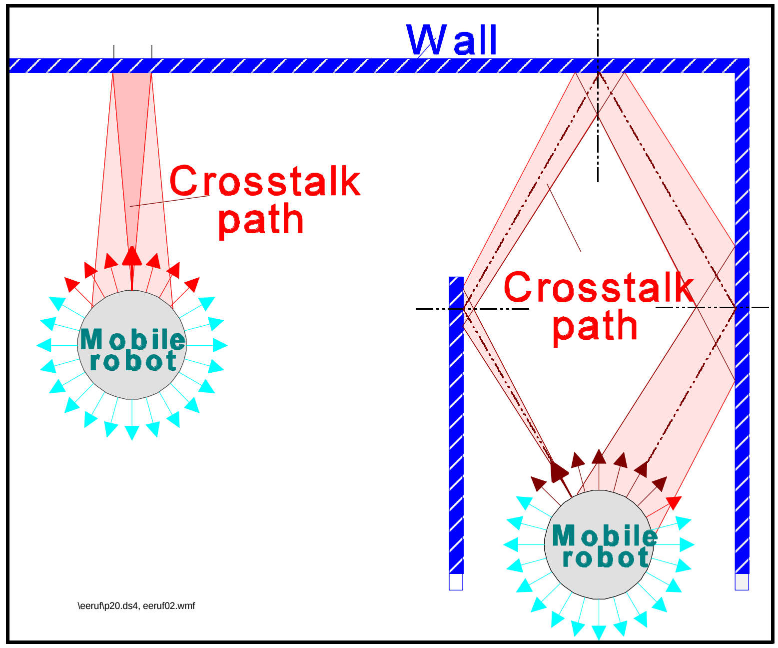

Figure 1: How *crosstalk* from onboard sensors is generated: a. *Direct* critical path.; b. *Indirect* critical path.

The situation is more complex for the indirect critical path in Fig. 1b. Here, at an instance t_0, sensor **x** fires and its ultrasound waves are reflected off three walls. Assuming the walls are fairly smooth, the reflected wavefront will reach sensor **y** after traveling through the distance $L=l1+l2+l3+l4$. If, at this time, sensor **y** is awaiting an echo of its own, then it will receive the signal from sensor **x** and interpret it as its own echo.

As is evident from Fig. 1b, crosstalk is not a phenomenon that occurs only under very extreme conditions. Furthermore, once a critical path exists, crosstalk is a particularly damaging condition because it will continuously cause false readings in sensor **y**, until the robot moves out of the critical path situation.

2. REJECTING NOISE AND CROSSTALK WITH THE EERUF METHOD

In this Section we introduce two methods for noise rejection. The first method, *comparison of consecutive readings*, is straight-forward. However, as we will show, this method can only reject non-systematic external noise. In order to also reject the systematic error caused by crosstalk, we introduce a modification called *comparison with alternating delays*.

2.1 Comparison of Consecutive Readings

One simple approach to eliminating occasional random noise is to compare two consecutive readings from the same sensor. The difference between any two consecutive readings, T_δ, is small if the readings result from "good" measurements (i.e., not caused by noise). One cannot assume $T_\delta = 0$ because of the discrete resolution of the sensor system. In

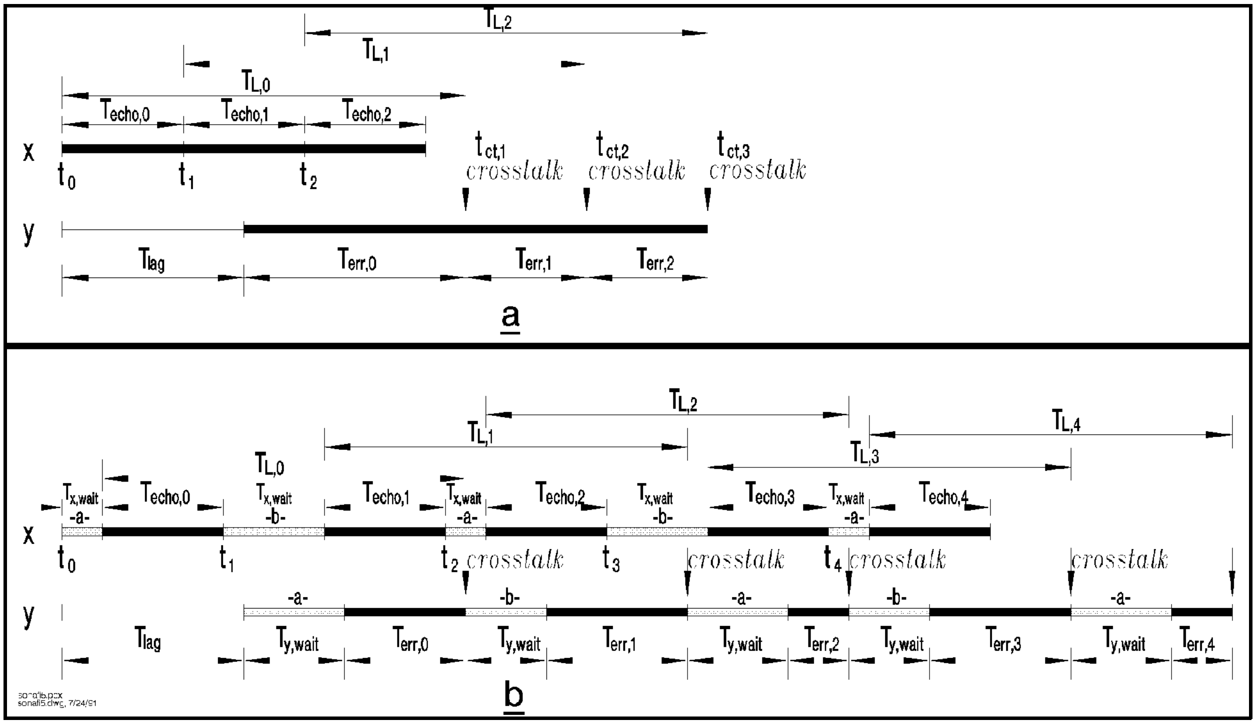

Figure 2: Timing diagram for *immediate refiring*. a. No wait-times; b. Alternating wait-times

the following discussion we will call consecutive readings that differ by less than a small amount T_δ "*near-identical readings*." We will also use the term "*external erronous readings*" for erroneous readings caused by external sources. External noise sources are typically not synchronized with the robot's internal firing intervals and it is therefore highly unlikely that one external erronous readings is near-identical to the previous reading, whether the previous reading was "good" or caused by noise, too. Thus, comparison of consecutive readings can identify external erroneous readings and subsequently reject such readings.

While comparison of consecutive readings is an efficient way for rejecting external erroneous readings, it is unsuitable for reducing crosstalk. This is so because crosstalk does not occur at random, as can be seen in Fig. 1. If, for example, sensor **y** always fired a fixed period of time (T_{lag}) after sensor **x**, then sensor **y** would repeatedly produce near-identical erroneous readings T_{err}. This process would repeat until the spatial conditions in the environment have changed so much (due to the robot's motion) that sensor **y** does not receive the signal from sensor **x** (i.e., the critical path is interrupted).

To support this claim, we introduce the timing diagram in Fig. 2a. The first firing of sensor **x** after a critical path is established is labeled t_0. Sensor **y** fires some arbitrary time T_{lag} later. For the configuration given in Fig. 1b, the ultrasound waves travel through the critical path L in the time $T_{L,n}$ (for the nth firing, with n = 0,1,..), resulting in an erroneous reading $T_{err,n}$ in sensor **y**. Sensor **x** receives its "legitimate" echo after a period $T_{echo,n}$. We will assume that $T_{echo,n}$ and $T_{L,n}$ change only insignificantly between subsequent readings. To simplify the mathematical treatment, we will thus assume that $T_{L,n} \cong T_{L,n-1}$ and $T_{echo,n} \cong T_{echo,n-1}$ for all n. This assumption is justifiable since we are only interested in comparing consecutive readings that differ at most by a small amount T_δ.

If each sensor re-fires immediately after it receives its first echo, we can express the times at which crosstalk occurs (t_{ct}) as follows:
For sensor **x**:

$$t_{ct}(n) = (n\text{-}1)T_{echo} + T_{L,n} \qquad \text{(for } n = 1, 2, \ldots) \tag{1}$$

For sensor **y**:

$$t_{ct}(n) = T_{lag} + \sum_{m=0}^{n-1} T_{err}(m) \qquad \text{(for } n = 1, 2\ldots \text{ and } m = 0, 1, \ldots) \tag{2}$$

The first erroneous reading due to crosstalk is denoted as $T_{err,0}$ and is given by (see Fig. 2a):

$$T_{err,0} = T_{L,0} - T_{lag} \tag{3}$$

rewriting Eq. (2) as

$$t_{ct}(n) = T_{lag} + T_{err,0} + \sum_{m=1}^{n-1} T_{err}(m) \tag{4a}$$

we can substitute (3) into (4a)

$$t_{ct}(n) = T_{L,0} + \sum_{m=1}^{n-1} T_{err}(m) \tag{4b}$$

equating (4b) to (1) yields

$$(n\text{-}1)T_{echo} = T_{L,0} - T_{L,n} + \sum_{m=1}^{n-1} T_{err}(m) \tag{5}$$

assuming that $T_{L,n} \cong T_{L,n-1} \pm T_\delta$, the term $T_{L,0} - T_{L,n}$ is bounded by $\pm(n\text{-}1)T_\delta$. Thus

$$(n\text{-}1)T_{echo} = \pm(n\text{-}1)T_\delta + \sum_{m=1}^{n-1} T_{err}(m) \tag{6}$$

or

$$(n\text{-}1)\,(T_{echo} \pm T_\delta) = \sum_{m=1}^{n-1} T_{err}(m) \tag{7}$$

Equation (7) holds true for all n only if

$$T_{err} = T_{echo} \pm T_\delta \tag{8}$$

Therefore, if a critical path exists between sensor **x** and sensor **y**, then sensor **y** will first produce one erroneous reading $T_{err,0}$, and will then continuously produce near-identical erroneous readings. Since the method of comparison of consecutive readings considers near-identical readings as valid, it **cannot** identify and reject this crosstalk error.

2.2 Comparison With Alternating Delays

To overcome this problem, we introduce an additional measure: an **alternating** delay (T_{wait}), before each sensor fires (Fig. 2b). For each sensor i, T_{wait} alternates between two values, a_i and b_i. a_i and b_i can be very small, on the order of a few milliseconds. Now, any sensor i can re-fire immediately after receiving an echo **and** after waiting for the short delay, $T_{i,wait}$. After each firing, $T_{i,wait}$ is toggled between $T_{i,wait,a}$ and $T_{i,wait,b}$.

Writing individual timing equations in a similar manner as above but for each n separately, we derive the following expressions for error readings n.

$$T_{err,0} = T_{x,wait,a} + T_{L,0} - T_{y,wait,a} - T_{lag} \quad \text{(for } n = 0\text{)} \tag{9}$$
$$T_{err,n} = T_{x,wait,b} - T_{y,wait,b} + T_{echo,n} \quad \text{(for } n = 1, 3, 5, \ldots\text{)} \tag{10a}$$
$$T_{err,n+1} = T_{x,wait,a} \;)\; T_{y,wait,a} + T_{echo,n+1} \quad \text{(for } n = 2, 4, 6, \ldots\text{)} \tag{10b}$$

The first crosstalk reading produces an arbitrary erroneous reading $T_{err,0}$. However, subsequent errors alternate between Eqs. (10a) and (10b). We recall that we wish to identify and discard crosstalk readings based on an artificially introduced difference between consecutive readings. We achieve protection from crosstalk if we reject every reading that differs by more than T_δ from the preceding one, that is

$$|T_{err,n+1} - T_{err,n}| > T_\delta \tag{11}$$

We substitute Eqs. (10) into (11), noting that all elements of Eqs. (10) are positive

$$\big|\,|T_{x,wait,a} - T_{y,wait,a}| - |T_{x,wait,b} - T_{y,wait,b}|\,\big| - |T_{echo,n+1} - T_{echo,n}| > T_\delta \tag{12}$$

Since consecutive legitimate echoes differ by no more than T_δ (i.e., $|T_{echo,n+1} - T_{echo,n}| < T_\delta$), we can rewrite Eq. (12) as

$$\big|\,|T_{x,wait,a} - T_{y,wait,a}| - |T_{x,wait,b} - T_{y,wait,b}|\,\big| > 2T_\delta \tag{13a}$$

Finally, it should be noted that the timing diagram in Fig. 2b and the timing equations (9) and (10) were written for the case where both sensors **x** and **y** started with T_{wait} in status a. Just as well, either one may have started in status **b**. To account for this (equal) possibility, we also have to make sure that

$$\big|\,|T_{x,wait,a} - T_{y,wait,b}| - |T_{x,wait,b} - T_{y,wait,a}|\,\big| > 2T_\delta \tag{13b}$$

We should recall and emphasize that indices x and y don't pertain to any particular sensor; rather, any sensor can be indexed x (i.e., originating the critical path) or y (i.e., receiving an "illegitimate" echo). Thus, if a mobile robot uses k sensors, we must find a set of $2k$ values for **all** $T_{i,wait,a}$ and $T_{i,wait,b}$ (with $i = 1,2...k$) that meets conditions (13) **in all possible combinations**. In other words, conditions (13) must be met if we substitute any one of the k wait-times $T_{i,wait,a}$ for either $T_{x,wait,a}$ or $T_{y,wait,a}$ and any one of the k wait-times $T_{i,wait,b}$ for either $T_{x,wait,b}$ or $T_{y,wait,b}$. Section 3 gives examples of such values and explains how they are found.

One should note that "good" readings are unaffected by this scheme, since echo-readings from actual obstacles are independent of the alternating delays and will be near-identical, differing at most by T_δ.

3. IMPLEMENTATION

The actual implementation of our method for rapid firing with alternating delays can be further enhanced by a slight modification to the firing scheme shown in Fig. 2b. The implementation described here combines the theoretical consideration of Section 2 with experimental observations and engineering considerations, resulting in a robust system for real-world applications.

The basic set-up for our implementation comprises *k* URSs spaced at 15° intervals and labeled 1, 2, ... *k*. We have experimentally determined that for a near-by wall (worst case), direct path crosstalk can affect three neighboring sensors (for example, when sensor #1 fires, sensors #2, #3, and #4 can receive the direct path echo). In order to reduce crosstalk in the first place) rather than having to reject an erroneous echo) sensors in a group of four neighboring sensors fire at scheduled intervals (instead of being re-fired immediately, as shown in Fig. 2). Intervals should be large enough to allow the echo of, say, sensor #1 to return from a near-by wall before any other of the 4 sensors in the group fires. Experimentally we found that intervals should be at least 15 ms, corresponding to a distance of 2.5 m between the wall and the sensors. Thus, firing sensors #1 to #4 at scheduled times T_{lag} = 0, 15, 30, and 45 ms (respectively) avoids most direct path crosstalk resulting from objects up to 2.5 m away.

We now combine the scheduled firing scheme with the method of comparison of consecutive readings and the method of alternating delays as follows (see Fig. 3):

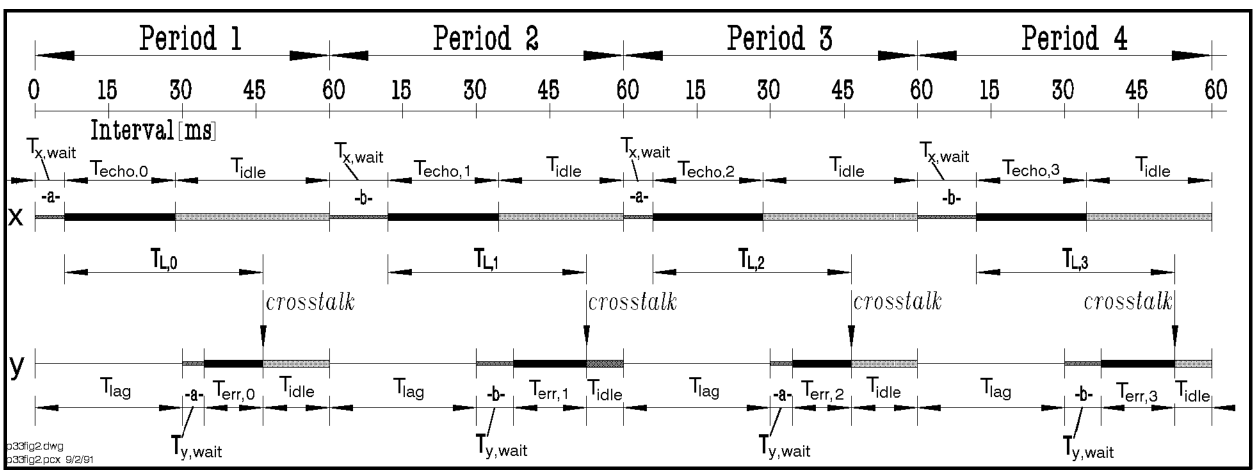

Figure 3: Timing diagram with *scheduled firing* and *alternating delays*.

1. Sensors #1 - #4 are scheduled for firing at intervals T_{lag} = 0, 15, 30, and 45 ms.
2. Subsequent groups of four sensors (e.g., #5 - #8) use the same intervals (0, 15, 30, and 45 ms).
3. Sensors don't actually fire at their scheduled times, but rather delay firing by T_{wait}.

4. Delays T_{wait} alternate between two different values, a and b. Each sensor i has its own distinct set of $T_{wait,a}$ and $T_{wait,b}$.
5. Thus, a sensor actually fires at time $T_{lag} + T_{wait}$ (relative to the beginning of each period).
6. Every sensor fires exactly once within each period of 4×15 = 60 ms.

The top row in Fig. 3 shows several periods, each divided into four intervals of 15 ms. The middle row shows the timing of a given sensor **x**, which, in this example here, happens to be scheduled at $T_{lag} = 0$. At $t = T_{lag}$, sensor **x** delays firing by $T_{x,wait} = \boldsymbol{a}$. Then, sensor **x** fires and awaits its echo. After the first echo is register ($T_{echo,0}$) sensor **x** does nothing (T_{idle}) until the end of the first period. This sequence repeats itself during the second period, with the exception that now sensor **x** delays firing by $T_{x,wait} = \boldsymbol{b}$.

The bottom row shows the events for sensor **y** (the sensor affected by crosstalk from sensor **x**). In the example here, sensor **y** is scheduled for firing at T_{lag} = 30 ms. After a delay of $T_{y,wait} = \boldsymbol{a}$ (recall that each sensor has its own individual pair of values ***a*** and ***b***), **y** fires and awaits its echo. However, assuming a crosstalk path of length L exists between sensors **x** and **y**, a crosstalk echo is received by sensor **y** $T_{L,0}$ ms after sensor **x** fires, causing an erroneous reading of $T_{err,0}$ in sensor **y**. After receiving the erroneous echo, sensor **y** idles until the end of the period (T_{idle}). This sequence repeats itself during the second period, with the exception that now sensor **y** delays firing by $T_{y,wait} = \boldsymbol{b}$. As can be readily seen from this timing diagram, the erroneous readings $T_{err,n}$ differ from $T_{err,n\text{-}1}$ and can thus be identified and rejected.

3.1 Choosing Timing Parameters for EERUF

To implement scheduled firing with alternating delays the following constraints must be met:

a. <u>The Alternating Delays Constraint</u>
   This constraint is the set of conditions established in Equations (13a) and (13b). These equations limit the choice of distinct pairs of wait-times $T_{x,wait,a}$ and $\mathrm{T}_{y,wait,b}$

b. <u>The Minimum Resolution Constraint</u>
   For real-life applications it is necessary to consider processing speed, sensor repeatability, and other engineering constraints that influence the maximum difference between two consecutive readings. The larger the potential difference between two consecutive "good readings," T_δ, the more difficult it is to meet Equations (13a) and (13b).

c. <u>The Minimum Interval Constraint</u>
   As we noted in the beginning of Section 3, it is desirable (but not crucial) to fire any 4 neighboring sensors at intervals of at least 15 ms, to minimize noise saturation due to direct path crosstalk.

It is not quite trivial to find timing parameters that meet all three constraints listed above. In the following discussion we explain the reasoning behind our choice of timing parameters. We will consider only 12 sensors, because in our environment this number is sufficient to cover a semi-circular area around the front half of the vehicle and completely protect it from collisions.

Parameter selection begins with determining a period, T_p. In the following example we start out with T_p = 100 ms. This period is divided into 4 equal nominal[2] intervals T_{nom} = 25 ms, as shown in Table I-a.

Table I-a: Scheduled firing times for 12 sensors

Sensor	1	2	3	4	5	6	7	8	9	10	11	12
T_{lag} [ms]	0	25	50	75	0	25	50	75	0	25	50	75

Next, for every sensor we choose pairs of $T_{wait,a}$ and $T_{wait,b}$ that meet constraint 3.1a. One such set is shown in Table I-b. These values were found by trial-and-error, using a computer program to test any set of suggested values for compliance with Eqs. (13a) and (13b). For the values in Table I-b, Eqs. (13a) and (13b) were met for T_δ < 1 ms. Note that with the method of scheduled firing delays (*a* or *b*) can alternate at the same time for all sensors. Thus, during the first period $T_{p,a}$ (e.g., $0 < t \leq 100$ ms) $T_{wait,a}$ is in effect for all sensors, and during the following period $T_{p,b}$ (e.g., $100 < t \leq 200$ ms) $T_{wait,b}$ is in effect for all sensors.

Table I-b: Alternating Wait-Times

Sensor	1	2	3	4	5	6	7	8	9	10	11	12
$T_{wait,a}$ [ms]	24	24	24	24	24	24	24	24	24	24	24	24
$T_{wait,b}$ [ms]	0	2	4	6	8	10	12	14	16	18	20	22

We can now combine scheduled times T_{lag} with alternating delays T_{wait}, to obtain the actual firing schedule listed in Table I-c:

Table I-c: Actual Firing Schedule

Sensor	1	2	3	4	5	6	7	8	9	10	11	12
$T_{fire,a}$ [ms]	24	49	74	99	24	49	74	99	24	49	74	99
$T_{fire,b}$ [ms]	0	27	54	81	8	35	62	89	16	43	70	97

One problem with the actual firing schedule in Table I-c is that the recommended 15 ms intervals between firings of any 4 neighboring sensors is not maintained. For example, during the first period sensor #4 fires at t = 99 ms. During the following period (which starts at t = 100 ms) sensor #1 fires at $T_{fire,b}$ = 0 ms (i.e., at t = 100 ms)) only 1 ms after sensor #4. This problem can be overcome by rearranging delay pairs T_{wait} as shown in Table I-d. The resulting actual firing schedule is also shown in Table I-d. Note that the shortest time difference between firings of any 4 neighboring sensors is now 19.0 ms (e.g., between sensor #4 and sensor #1) and constraint 3.1c is met.

[2] We will see later that the *actual* firing times differ from the *nominal* ones.

Table I-d: Modified Firing Schedule

Sensor	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
T_{lag} [ms]	0	25	50	75	0	25	50	75	0	25	50	75
$T_{wait,a}$ [ms]	24	24	24	24	24	24	24	24	24	24	24	24
$T_{wait,b}$ [ms]	18	12	6	0	20	14	8	2	22	16	10	4
$T_{fire,a}$ [ms]	24	49	74	99	24	49	74	99	24	49	74	99
$T_{fire,b}$ [ms]	18	37	56	75	20	39	58	77	22	41	60	79

It is straightforward to derive other timing schedules by proportionally scaling the firing times $T_{fire,a}$ and $T_{fire,b}$, allowing the design of URS systems with different characteristics. For example, faster firing can be achieved by using a firing schedule based on, say, T_p = 60 ms, although this measure may increase the number of readings that must be rejected due to crosstalk. Conversely, a slower schedule reduces the rejection rate but provides fewer readings. Combining these considerations, we have successfully tested an approach called "adaptive scheduling." In this approach the algorithm monitors its rejection rate and adaptively changes its firing characteristics accordingly. The benefit of adaptive scheduling is that the robot can fire very rapidly (e.g., T_p < 60 ms) in environments with little crosstalk, while automatically reducing its firing rate (and speed of travel) in crosstalk-promoting environments or in the presence of other mobile robots.

4. EXPERIMENTAL RESULTS

In this Section we report on two different sets of experiments. The first set concerns the error rejection performance of the EERUF method, while the second group tests the impact of EERUF's fast firing rates on the performance of mobile robot obstacle avoidance.

Set I: Error Rejection

In this set of experiments we tested EERUF in a reproducible, stationary test-environment like the one shown in Fig. 4. This set-up comprised 8 sensors, spaced at 15° intervals. The sensors faced three perpendicular, highly reflective walls) a configuration that strongly promotes crosstalk. To identify errors during the experiments, we took initial range measurements in the beginning of each experiment for reference. Reference measurements were taken by firing each sensor individually and waiting for 200 ms before firing the next sensor, to make sure there was no crosstalk. We compared the performance of the EERUF algorithm with a generic "conventional" algorithm. The

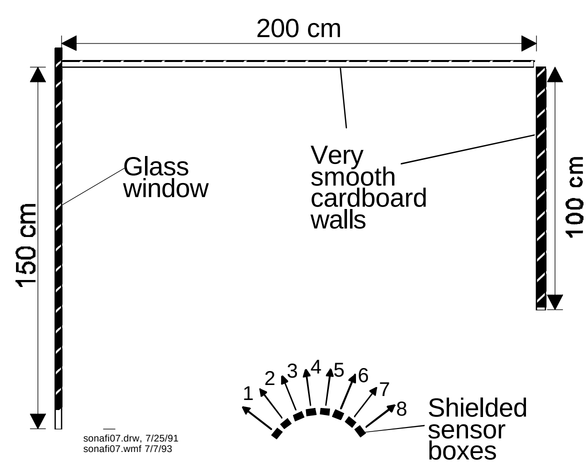

Figure 4: Experimental set-up in crosstalk-promoting environment.

conventional algorithm fired the sensors at the same rate, but without noise rejection. With both EERUF and the conventional firing method, approximately 8,000 readings were taken (at a rate of T_p = 60 ms) and compared to the initial reference measurements. EERUF consistently produced fewer than 3% errors (typically 1-2%), while the conventional firing scheme consistently produced more than 30% errors (typically 40-80%). It is difficult to quantify these results more accurately, because small changes in the geometry of the experimental setup would change the sensors affected by crosstalk and thus the percentual results. One particular limitation of this experiment is that it was not able to distinguish between errors due to crosstalk or other factors (e.g., marginal readings, because of specular reflections). Nonetheless, the qualitative results reported here clearly show the significance of the improvements obtained with EERUF in our environment.

Set II: Mobile Robot Obstacle Avoidance

Error rejection with the EERUF method allows fast firing of the URSs. We tested the impact of fast firing on the obstacle avoidance performance of a mobile robot in a densely cluttered environment with difficult-to-detect obstacles. These experiments were performed on the commercially available *LabMate* platform [20]. The LabMate is 75 cm long, 75 cm wide, and has a maximum speed of 1 m/sec. In our experimental system we used eight POLAROID sensors that were symmetrically spaced at 15° intervals (see Fig. 5) and fired at a fixed rate of T_p = 60 ms).

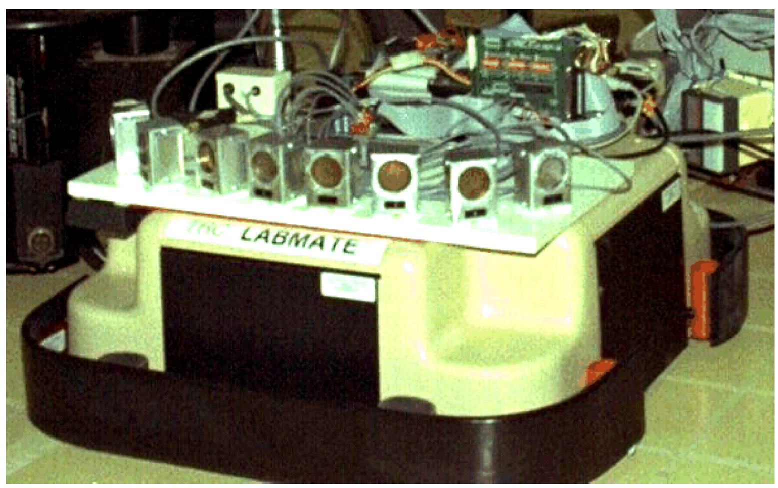

Figure 5: One of the University of Michigan's four *Labmate* robots, equipped with an array of eight URSs (spaced at 15 ° intervals).

A '386-20 Mhz computer ran the EERUF algorithm as an interrupt-driven background task. The main task was the *vector field histogram* (VFH) obstacle avoidance method [3] combined with the histogrammic in motion mapping (HIMM) method [4].

We set up an obstacle course comprising of pencil-thin (8 mm diameter) vertical poles spaced approximately 1.6 m from each other (see Fig. 6). With the EERUF method, the robot was able to traverse this course at its maximum speed of 1 m/sec and an average speed of 0.8 m/sec (the maximum speed was reduced before and during tight turns, for dynamic reasons).

In another experiment we found that EERUF allowed equally fast obstacle avoidance even in the presence of intense ultrasonic noise from a second mobile robot with 24 URSs. Also, as can be seen in Fig. 6, the far corner of the lab has highly reflective smooth walls, which strongly promote crosstalk; so do the reflective poster boards (with surface-smoothness similar to that of plexiglass) shown in Fig. 6. With EERUF, the robot was able to avoid all obstacles while traveling at a speed of up to 1 m/sec.

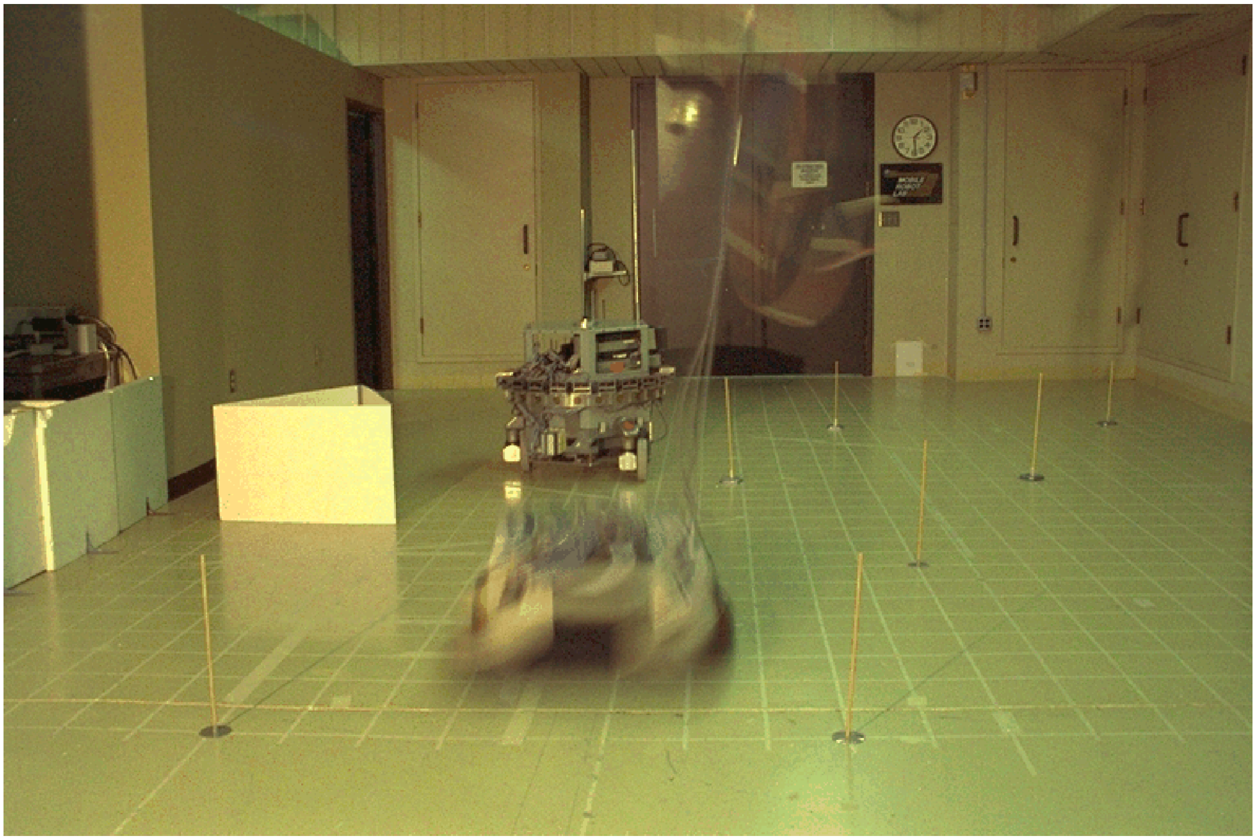

Figure 6: With EERUF, the *Labmate* zaps through an obstacle course at 1 m/sec. Another mobile robot nearby generates ultrasonic noise at a rate of 24 firings per 80 ms.

5. CONCLUSIONS

We have introduced *error eliminating rapid ultrasonic firing* (EERUF), a new method that allows fast firing of multiple URSs. EERUF is able to identify and reject erroneous readings due to crosstalk and discrete external noise.

EERUF is based on the principle of comparison of consecutive readings, but, in addition, employs alternating delays before firing each sensor. The latter measure artificially creates differences between consecutive crosstalk readings, while leaving "good" readings unaltered.

Experimental results show successful rejection of both direct and indirect path crosstalk (from onboard sensors), and errors caused by external sources. In error prone test-environments, the EERUF method consistently produced one to two orders of magnitude fewer errors than a convebtional firing scheme (i.e., one without error rejection) firing at the same rate. In summary, these are the advantage of EERUF over conventional firing methods:

1. With EERUF, mobile robots are able to traverse obstacle-cluttered environments safely and much faster than with conventional methods. We have successfully demonstrated obstacle avoidance at 1 m/sec, which was limited only by the physical capability of the mobile platform. We expect that with some optimization EERUF will allow safe obstacle avoidance at speeds of up to 2 m/sec.

2. Multiple mobile robots can operate in the same environment, without interference among their URSs.

Acknowledgements

This research was motivated by the needs of the *Robotics for Advanced Reactors* program sponsored by the Department of Energy Office of Nuclear Energy. This program combines in a unique way the team-efforts of the Universities of Michigan, Florida, Tennessee, and Texas, as well as of the Oak Ridge National Lab (ORNL). Special thanks also to Mr. Lincoln Jaros from the Rehabilitation Engineering Center at the University of Michigan Hospital for developing ingenious interrupt drivers, which allow micro-second timing without additional hardware.

---

6. REFERENCES

[1] Borenstein, J. and Koren, Y., 1988, "Obstacle Avoidance With Ultrasonic Sensors." *IEEE Journal of Robotics and Automation*, Vol. RA-4, No. 2, pp. 213-218.

[2] Borenstein, J. and Koren, Y., 1989, "Real-time Obstacle Avoidance for Fast Mobile Robots." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 5, Sept/Oct, pp. 1179-1187.

[3] Borenstein, J. and Koren, Y., 1991, "The Vector Field Histogram) Fast Obstacle-Avoidance for Mobile Robots." *IEEE Journal of Robotics and Automation*, Vol. 7, No. 3., June, pp. 278-288.

[4] Borenstein, J, and Koren, Y., 1991, "Histogramic In-motion Mapping for Mobile Robot Obstacle Avoidance." *IEEE Journal of Robotics and Automation*, Vol. 7, No. 4, pp. 535-539.

[5] Bozma, Ö. and Kuc, R., 1991, "Characterizing Pulses Reflected From Rough Surfaces Using Ultrasound." *Journal of the Acoustical Society of America*, vol. 89, no. 6, pp. 2519 -2531.

[6] Buchenberger, M., Jörg, W., and Von Puttkamer, E., 1993, Laserradar and Sonar-based World Modelling .." *1993 IEEE Conference on Robotics and Automation*, Atlanta Georgia, May 10-15 pp. 534-541.

[7] Crowley, J. L., 1989, "World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging." *Proceedings of the 1989 IEEE International Conference on Robotics and Automation*, Scottsdale, Arizona, May 14-19, pp. 674-680.

[8] Denning Mobile Robotics, Inc., 21 Cummings Park, Woburn, MA 01801.

[9] Everett, H. R., Gilbreath, G. A., Tran, T. and Nieusma, J. M., 1990, "Modeling the Environment of a Mobile Security Robot." *Technical Document 1835, Advanced Systems Division, Naval Ocean Systems Center, San Diego, California 92152-5000*, June.

[10] Flynn, A. M., 1988, "Combining Sonar and Infrared Sensors for Mobile Robot Navigation." *The International Journal of Robotics Research*, Vol. 7, No. 6, December pp. 5-14.

[11] Holenstein, A. A. and Badreddin, E., 1991, "Collision Avoidance in a Behavior-based Mobile Robot Design." *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, Sacramento, California, April, pp. 898-903.

[12] Kuc, R. and Barshan, B., 1989, "Navigating Vehicles Through an Unstructured Environment With Sonar." *Proceedings of the 1989 IEEE International Conference on Robotics and Automation* Scottsdale, Arizona, May 14-19, pp. 1422-1426.

[13] Kuc, R. and Viard, V. B., 1991, "A Physically-Based Navigation Strategy for Sonar Guided Vehicles. *International Journal of Robotics Research*, vol. 10, no. 2, pp. 75-87.

[14] Kuc, R., 1991, "Why Sonar Works (when it does)." Submitted to the *Journal of IEEE Trans. Robotics and Automation*.

[15] Leonard, J.J. and Durrant-Whyte, H.F., 1992, *Directed Sonar Sensing For Mobile Robot Navigation*. Kluwer Academic Publishers, Hingham, MA. ISBN: 0-7923-9242-6.

[16] Manz, A., Liscano, R., and Green, D., 1991, "A Comparison of Realtime Obstacle Avoidance Methods for Mobile Robots." *Experimental Robotics*, Toulouse, France, June.

[17] Moravec, H. P., 1988, "Sensor Fusion in Certainty Grids for Mobile Robots." *AI Magazine*, Summer, pp. 61-74.

[18] Pin, F. G. et al., 1989, "Autonomous Mobile Robot Research Using the Hermies-III Robot." *IROS International Conference on Intelligent Robot and Systems*, Tsukuba, Japan, Sept.

[19] POLAROID Corp, Ultrasonic Components Group, 119 Windsor Street, Cambridge, MA.

[20] TRC (Transition Research Corporation), Shelter Rock Lane, Danbury, CT.