Object Location In A Multiply Reflective Environment

Brendan Walker, Hermann Chong, Iqbal Binderwala,
Prof. Tony Woo

University of Washington, Dept. of Industrial Engineering

Overview

- Motivation for object location in a static system
- Related problems
- Methods for signal triangulation
- Reflection and synchronization issues
- Algorithm for solving these problems
- Theoretical and experimental implementations
- Conclusion

Motivation for Object Location

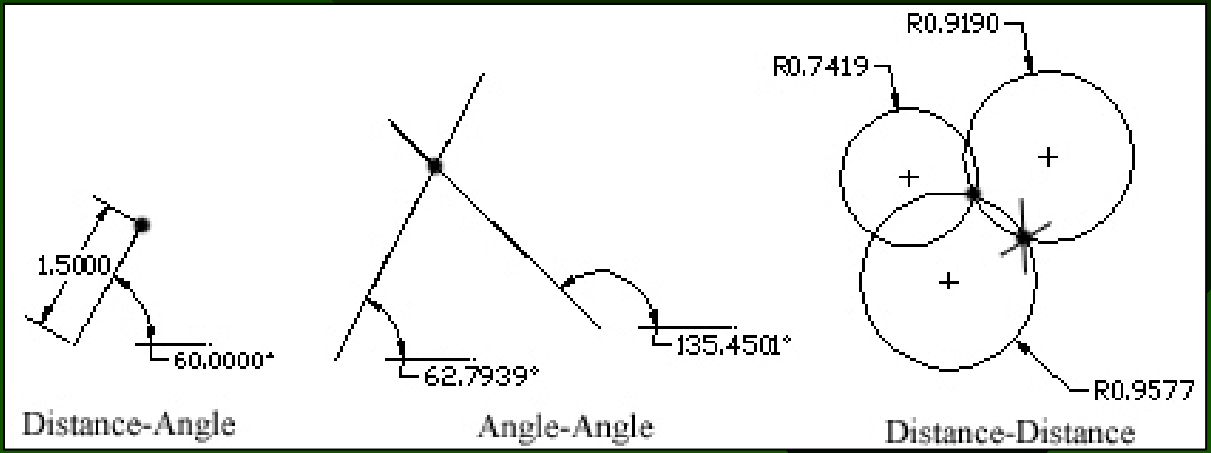
- Want to find an object in a system
  - Inventoried item in a warehouse
  - Someone who called 911 on their cell phone
- Work within the existing constraints of the environment

Related Technologies

- Beaconing bar code
  - Radio sensitive tag placed on inventoried item
  - A radio “ping” is sent out to the tag
  - Tag responds back with reply
- Augmented GPS
  - GPS receiver stations set up at known coordinates
  - End user’s receives SAT GPS corrections
  - Possible technology for use in E911 systems

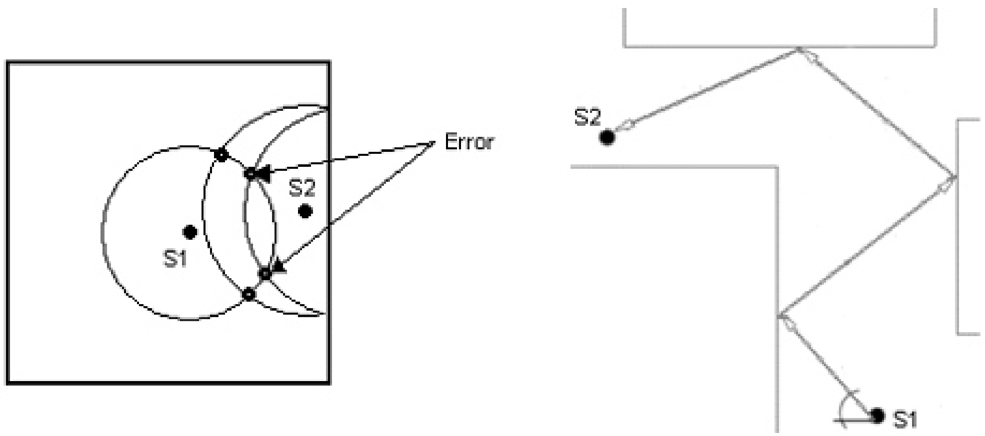
Triangulation Methods

- Application determines receiver type
- Generally, angle distance receivers are more complicated



Reflection and Multi-path

- Bounce path longer than true distance
- Signal direction is changed



Distance-Distance triangulation with reflection

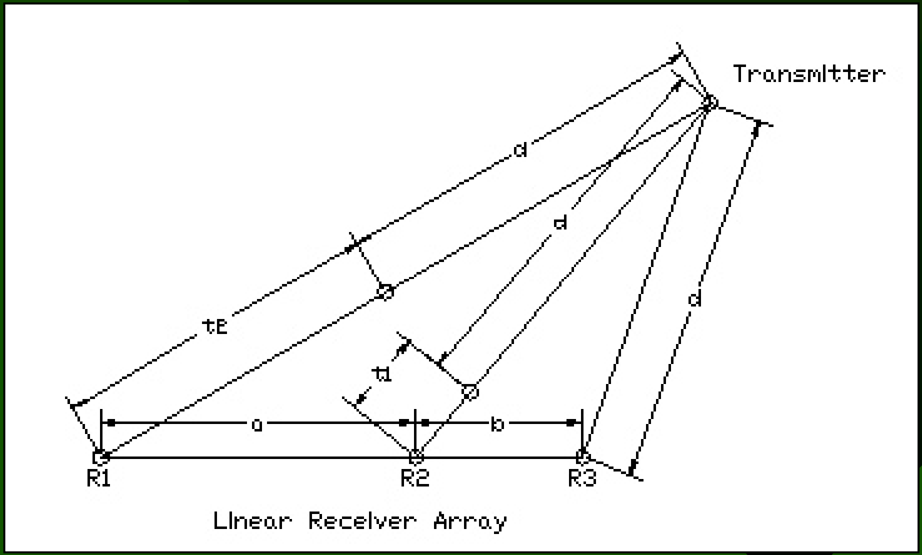
Distance-Angle triangulation with reflection

Implicit Problem of Synchronization

- Many signal based distance methods use time of flight (TOF)
- Like BBC, assumes we receiver can respond back
- Sometimes synchronization is not possible
- Removing need to respond makes it easier to track objects

Asynchronous Triangulation

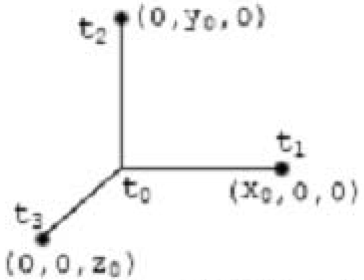
- Putting receivers in a line allows for asynchronous distance calculation
- Consider free space (no obstacles)



Multi-path Triangulation

- Possible if we know the following
  - Geometry of what is reflection
  - Position of receiver
  - Incoming signal direction
  - Distance to origin of signal
- Can then ray trace back to signal's origin

Calculating Signal Direction



$$x^2 + y^2 + z^2 = (r + t_0)^2$$

$$(x - x_0)^2 + y^2 + z^2 = (r + t_1)^2$$

$$x^2 + (y - y_0)^2 + z^2 = (r + t_2)^2$$

$$x^2 + y^2 + (z - z_0)^2 = (r + t_3)^2$$

where t_0 or t_1 or t_2 or $t_3 = 0$

$$x = \left((t_0 - t_1)(2r + t_0 + t_1) + x_0^2\right) / 2 \qquad x(r) = r(t_0 - t_1)/x_0 + \left(t_0^2 - t_1^2 + x_0^2\right)/2x_0$$

$$y = \left((t_0 - t_2)(2r + t_0 + t_2) + y_0^2\right) / 2y_0 \longrightarrow y(r) = r(t_0 - t_2)/y_0 + \left(t_0^2 - t_2^2 + y_0^2\right)/2y_0$$

$$z = \left((t_0 - t_3)(2r + t_0 + t_3) + z_0^2\right) / 2z_0 \qquad z(r) = r(t_0 - t_3)/z_0 + \left(t_0^2 - t_3^2 + z_0^2\right)/2z_0$$

Solving for Radius in 2 D :

$$Ar^2 + Br + C = 0$$

$$A = 4\left[y_0^2 (t_0 - t_1)^2 + x_0^2 (t_0 - t_2)^2 - x_0^2 y_0^2\right]$$

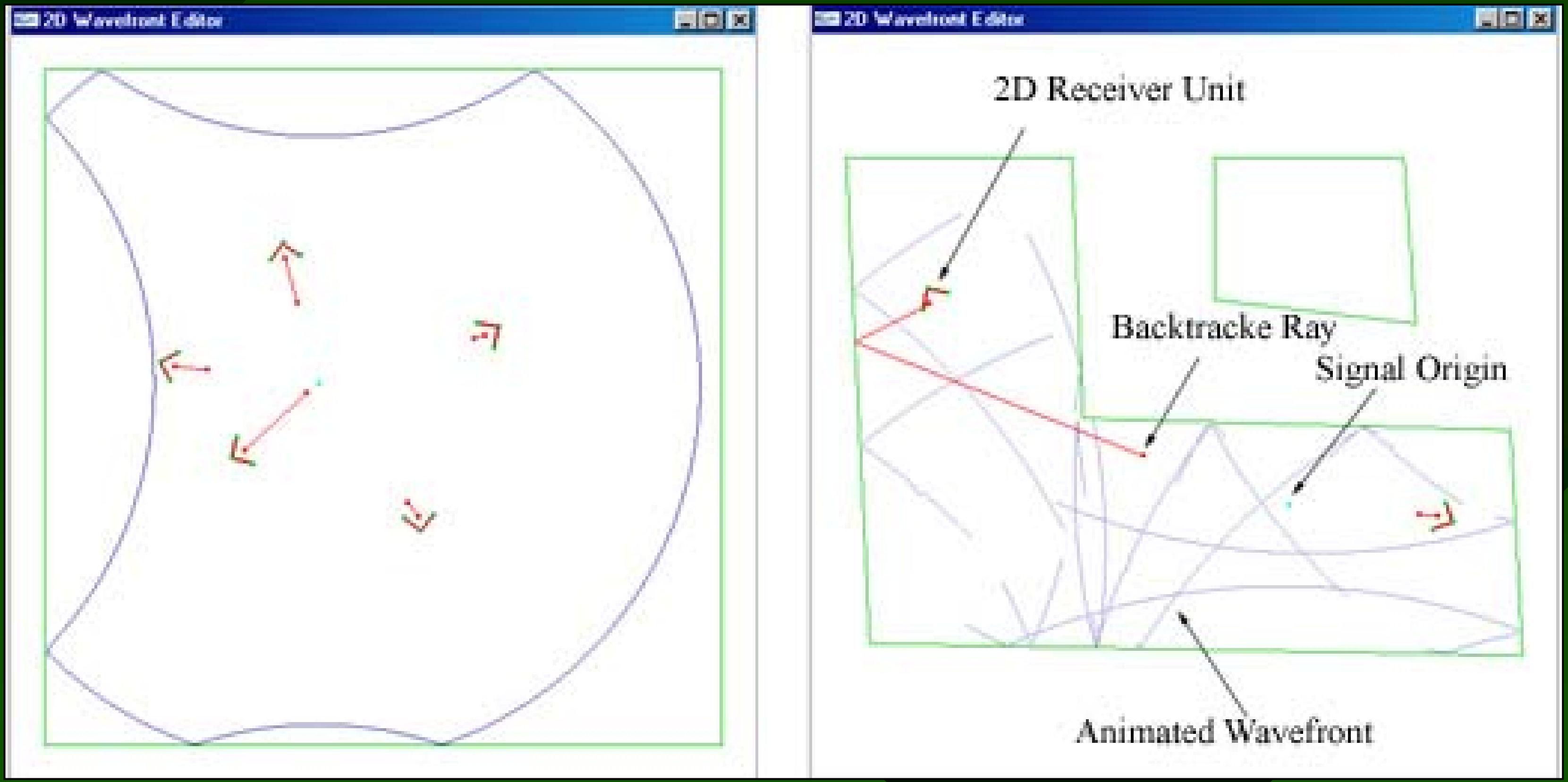
$$B = 4\left[y_0^2 (t_0 - t_1)^2 \left[t_0^2 - t_1^2 + x_0^2\right] + x_0^2 (t_0 - t_2)^2 \left[t_0^2 - t_2^2 + y_0^2\right] - 2 t_0 x_0^2 y_0^2\right]$$

$$C = y_0^2 \left(t_0^2 - t_1^2\right)\left[t_0^2 - t_1^2 + 2x_0^2\right] + x_0^2 (t_0 - t_2)^2 \left[t_0^2 - t_2^2 + 2y_0^2\right] + x_0^2 y_0^2 \left[x_0^2 + y_0^2 - 4t_0^2\right]$$

Theoretical Implementation

- Assume a model in which each surface is perfectly reflective
- Signal is an expanding circular (2D) or spherical (3D) wave front
- Once signal crosses an “Orthant” receiver, back trace to origin

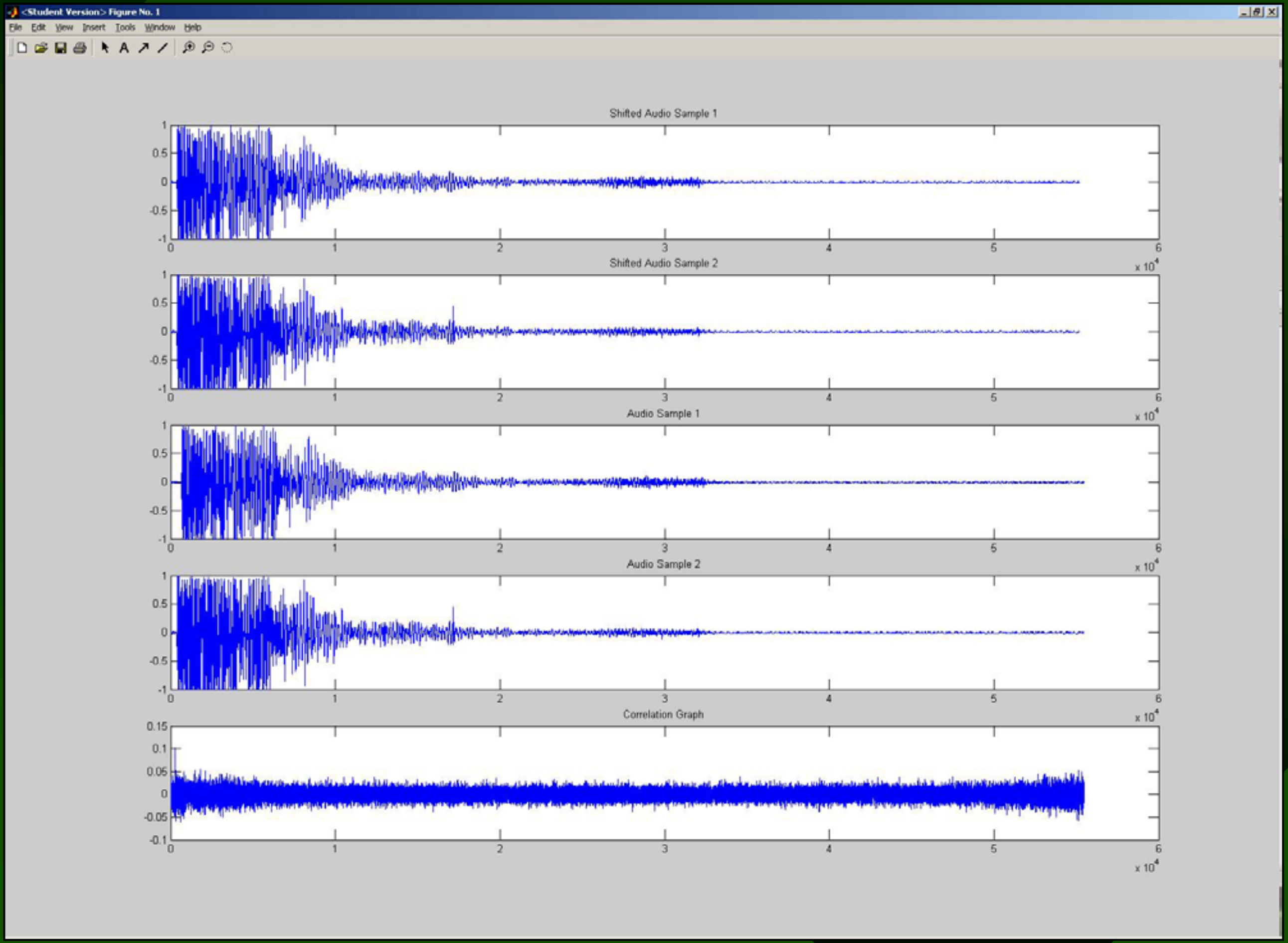
Theoretical Implementation



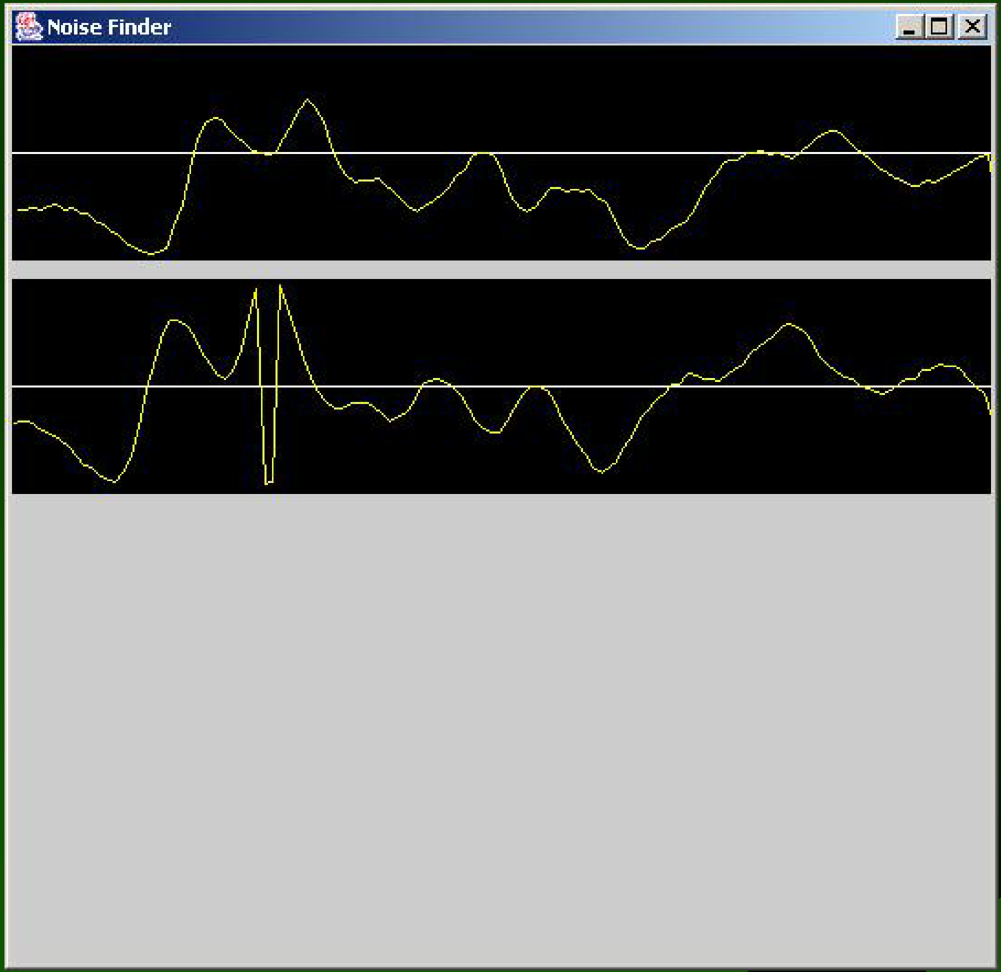
Experimental Implementation

- Working in the audio domain
- Non trivial to determine when a signal started crossing a microphone
- Incoming data has to be continually processed from a buffer
- FFT and autocorrelation used to find signal thumbprint

Auto Correlation Experiment



Java Implementation



Future Work

- Run some experiments
- Implement a more rigorous simulation
- Implement the autocorrelation in hardware
- Considerworking in the RF domain

Conclusion

- Wide range of applications for this technology
- Asynchronous transmission means little of no hardware on the transmitters end
- Our method of multi-path triangulation can be implemented very cheaply for audio applications

Questions?