

Real-Time Gunshot Localization Using Acoustics

by

Aniruddha Sinha and Hemendra Arya

Associate Professors

Department of Aerospace Engineering

Indian Institute of Technology Bombay

under the aegis of

National Center of Excellence in Technology for Internal Security (NCETIS)



May 2021

Executive Summary

Real-time localization of shooters can be achieved by acquiring and processing the acoustics generated from a gunshot event, which is helpful for security forces encountering hostile gunfire. The acoustic wave generated from the expanding gases at the muzzle of a gun is captured using one or more sensor clusters, each consisting of multiple microphones, that are deployed for the purpose. The range and bearing of the shooter may be estimated by various localization algorithms that take as input the differential-time-of-arrival of the blast wave's acoustic signal at all microphone pairs in a cluster.

This report describes the development of the hardware and software for realizing such a localization system. It is intended for use in securing a compound that may be under enemy gunfire. At present, we use a single cluster of four microphones, having a total footprint of 1 meter square. The system is able to estimate a shooter's location within 0.1 s of the gunshot event. Our implementation is evaluated in several experiments performed using firecrackers (that generate blast waves similar to a gun) and firearms. In general, the bearing angle of the shooter, which is the main parameter of interest, is predicted very well, with maximum error of 3° only. The range of the shooter is inherently more difficult to predict since one has to rely on the very slight curvature of the spherical wavefront of the blast at large distances from the muzzle. Indeed, significant range error is found in our tests, although some mitigation is observed on increasing the sampling rate and deploying redundant sensors in a cluster.

The shock wave trailed by a supersonic bullet in a gunfire event is an additional acoustic signature that is very useful for localization. This is especially true for a very distant sniper whose gun muzzle blast may fade before reaching the sensors. Although we have not implemented any shock wave-based localization method in hardware, we have evaluated a simple algorithm in extensive computer simulations. This method does not attempt to estimate the range of the sniper, instead focusing on determining the more important bearing angle only. Our simulations consider the problem of securing a 50 meter square compound using four sensor clusters at its four corners, each cluster consisting of four virtual microphones. The computer tests indicate that this approach is very promising, with maximum bearing angle error of 2° for snipers that are 1 km away from the sensors. If the shooter is nearby, say 100 m away, then the results are less accurate, with maximum error being 15°. However, the muzzle blast becomes usable in these cases, and may be used to refine the location estimate.

Contents

1	Introduction	1
2	Background	3
2.1	Acoustic Signature of Gunshots	3
2.1.1	Muzzle blast from propellant at the gun location	3
2.1.2	Shock waves generated by supersonically-moving projectiles	6
2.1.3	Effect of reflection	7
2.1.4	Simulating gun muzzle blast with firecracker	8
2.2	Gunshot Localization Preliminaries	8
2.2.1	Detection	9
2.2.2	Reflection artifact minimization	9
2.2.2.1	Cross-correlation	10
2.2.2.2	Using auto-correlation to minimize reflection artifact	10
2.2.3	Gunshot event categorization	11
2.2.4	DTOA calculation using cross-correlation	13
2.2.4.1	Parabolic interpolation of cross-correlation peak	14
2.3	Algorithms for localizing gun-shooter from muzzle blast	15
2.3.1	Wavefront curvature method	16
2.3.2	Nonlinear least-squares method	18
2.3.3	Bancroft's method	20
2.3.4	Localization using multiple direction-of-arrival vectors	20
2.4	Algorithms for localizing gun-shooter using bullet shock wave	22
2.4.1	Geometric attributes of ballistic shock wave	22
2.4.2	Estimation of bullet trajectory direction	23
3	Hardware and Software for Acoustic Gunshot Localization	26
3.1	Sensor basics and arrangement	26
3.2	Data acquisition and processing	28
3.2.1	MCC USB-1608FS DAQ board for data acquisition	28
3.2.2	Tiva board for data acquistion	28
3.2.3	RPi board for gunshot localization	31
4	Results and Discussions	33
4.1	Validation of offline localization based on blast wave using firecrackers and the tetrahedral microphone array	33
4.2	Validation of online localization based on blast wave using firecrackers and the planar square microphone array	35

4.3	Offline localization of gunfire	38
4.4	Bullet trajectory direction estimation in simulations	38
4.4.1	Simulation of ballistic shock wave recorded at a virtual microphone	39
4.4.2	Simulation of an installation security system that estimates the bearing of a far shooter from ballistic shock	40
4.4.3	Effect of signal noise on performance of shock wave-based localization	43
5	Summary and Conclusions	45

Chapter 1

Introduction

Security forces often face difficulties in ascertaining the origin of gunfire from their sound. This is especially true of obstacle-dense and reverberative environments like forests, valleys and cities. Moreover, security personnel guarding installations in hostile environments may not be able to keep up a round-the-clock vigil. Yet, they will desire instant knowledge of the location of the enemy when they do shoot, so as to undertake prompt and effective counter-measures. Hence, there is a felt need for machine-assisted real-time gunshot localization.

The following is a list of signatures left by a gunfire event: (1) Sound of charge explosion in gun muzzle, (2) Light of the charge explosion, (3) Sound of the bullet travelling, which is only perceptible if the bullet's speed is supersonic, (4) 'Sight' of the bullet travelling, (5) Sound of the bullet hitting the target, and (6) Smell of the gunfire. Evidently, only the first and third signatures are useful for reliable localization of the shooter. It will be noticed that both are acoustic in nature.

The objective of the project was thus to develop a system that can robustly detect and localize gunshots based on their acoustic signature in scenarios of relevance to internal security. This consists of a minimal number of low-cost microphones and associated electronic circuitry distributed over the zone of interest. They feed into an efficient algorithm running on a micro-processor to deliver actionable information in real time.

Acoustic targeting devices may be classified based on their mobility. At one end of the spectrum is the static device that is appropriate for guarding installations, like sensitive areas in a city, or forward camps in forests. Next in terms of mobility are vehicle-mounted devices that are useful for protecting convoys moving in hostile territory. The most portable solutions are helmet-mounted or shoulder-worn, that assist soldiers in on-foot operations. As will become evident in the following, these classes differ in the sophistication of their hardware and software capabilities. The current project is a preliminary attempt at developing such devices indigenously; hence, the focus is on the static variant. Figure 1.1 exemplifies a typical setup for implementing and validating a static acoustic targeting system; this is representative of our work too.

This report describes the scientific background of the acoustic targeting system, as well as its detailed development and validation. The basic attributes of the noise signature of gunshots is first discussed in Chapter 2. The same chapter then goes on to delineate the successive steps involved in acoustic targeting, and the various hardware capabilities and software algorithms needed to accomplish them. The specific details of our implementation are discussed in Chapter 3. In particular, we first describe the choice of microphones as well as data acquisition and microprocessor hardware. Then we go on to explain the software implemented in the microprocessor. We have only implemented the muzzle blast-based localization approach in hardware; the one based on the bullet-trailed wave is only implemented in computer simulations for possible translation to hardware



Figure 1.1: Setup for a shooter localization method using four sensor nodes placed in a field (Sallai et al., 2013).

in the future. The gunshot localization system developed has been validated in several situations, and the setup and outcome of these tests are described in Chapter 4. Chapter 5 ends the report with a summary of the work done, and directions in which it may be extended.

Chapter 2

Background

2.1 Acoustic Signature of Gunshots

2.1.1 Muzzle blast from propellant at the gun location

A conventional firearm uses an explosive propellant in its muzzle to discharge the bullet. The local density fluctuations created by this blast are captured clearly in the shadowgram of a Beretta gun in action (Settles et al., 2005), presented in fig. 2.1. The sound from this explosion travels in all directions at sonic speed.

Although the speed of propagation of the blast wave is uniform in all directions, the amplitude is loudest in the direction of firing, as represented in the simplified directivity plot of fig. 2.2a. Figure 2.2b also shows that the typical acoustic signature varies across observer locations. The microphone signal recorded in the front sector of the gun displays a sharp rise (within 50 to 200 μ sec) followed by a relatively slower decay. The signature recorded in the rear sector shows several oscillations. The blast wave suffers attenuation and broadening due to atmospheric absorption, wind, ground effects, and hindrance from solid obstacles – this renders them less useful for sensors at large distances from the shooter. Since the muzzle blast emanates from the gun (and not the bullet), it can be used to directly localize the shooter (instead of resorting to ballistics modelling, as discussed below).

Figure 2.3 demonstrates the change in amplitude and shape of the muzzle blast wave with distance from the gun. The attenuation of the wave is primarily due to spherical propagation; it is well explained by the inverse-distance decay law. Atmospheric absorption is responsible for the distortion of the waveform, especially the preferential reduction of sharpness of the initial

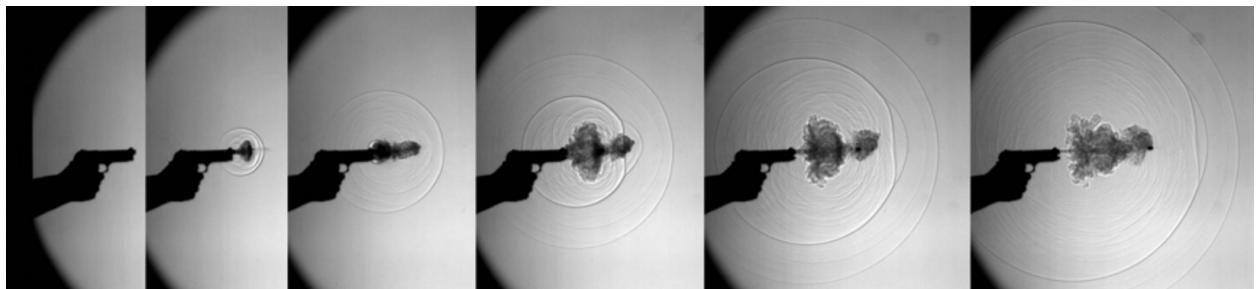


Figure 2.1: Consecutive shadowgrams of the firing of a Pennsylvania State Police Beretta Model 96D (Settles et al., 2005).

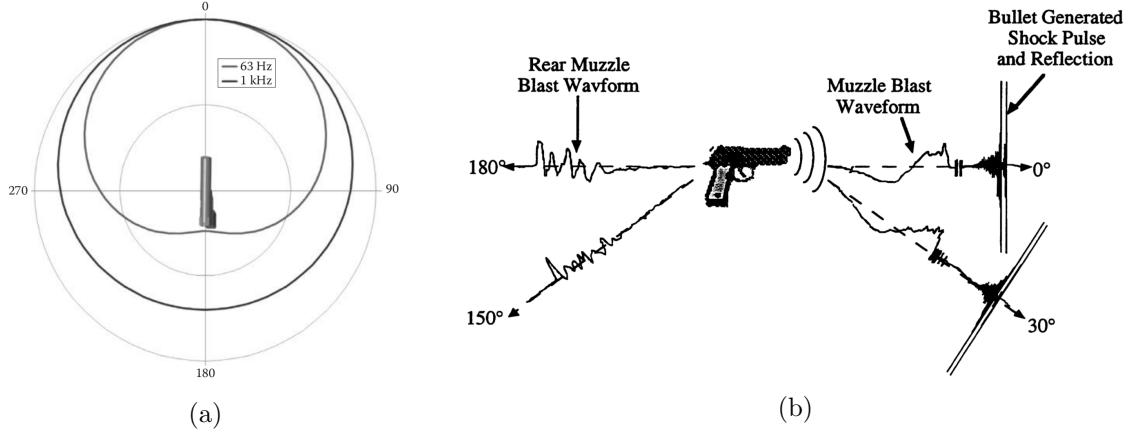


Figure 2.2: (a) Simplification of muzzle blast directivity (Aguilar, 2015). (b) Variation of muzzle blast's acoustic signature around the gun (Page and Sharkey, 1995).

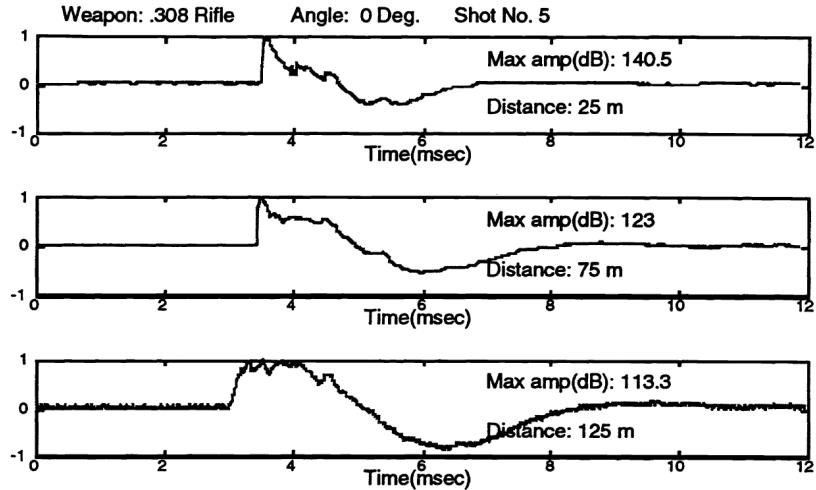


Figure 2.3: Variation of muzzle blast waveform with distance from the gun (a .308 rifle) (Page and Sharkey, 1995).

over-pressure.

Every firearm has a distinct muzzle blast signature, as evidenced by fig. 2.4. This is due to the specific design of the gun, one of the main parameters being the length of the muzzle itself. However, it will be observed that they share some common attributes: the initial sharp rise, followed by gentler decay, a negative excursion, and final settling. In our work, these features are all that are used for identification and localization, and not the specific details of the waveform. Hence, the implementation works for all firearms.

It is evident by now that the muzzle blast is an impulsive event having a sharp and short-lived acoustic signature. There are other acoustic events which may sound similarly impulsive to the ear. Figure 2.5 presents the acoustic signatures of several such occurrences. It is evident that their actual waveform are quite different from that of a muzzle blast. Thus, algorithms may be developed to identify gunshots, and disregard all other potentially similar events, prior to further processing and localization.

The muzzle blast may be suppressed if a silencer is used, in which case it cannot be used for

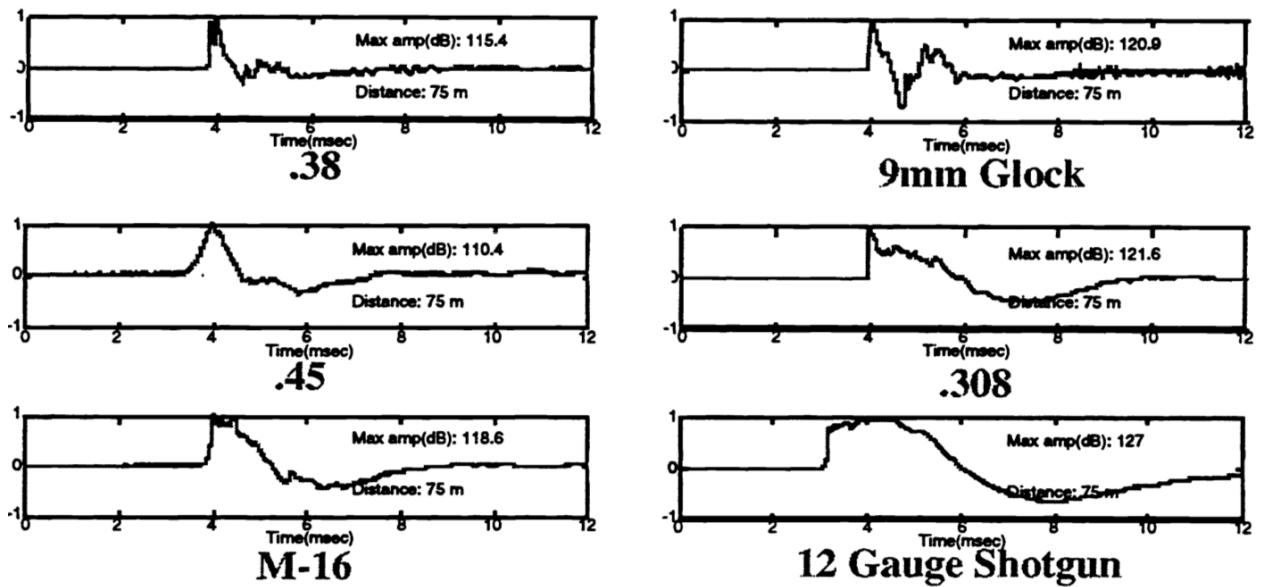


Figure 2.4: Forward direction muzzle blast waveforms for a variety of weapons (adapted from Page and Sharkey (1995)).

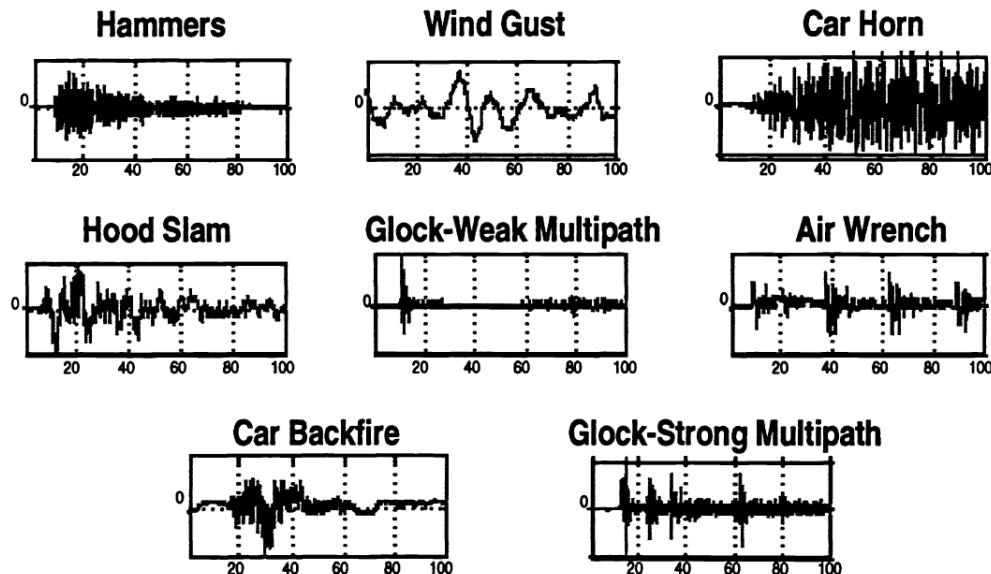


Figure 2.5: Sample noise from acoustic events that are also impulsive like gun muzzle blasts (Page and Sharkey, 1995).

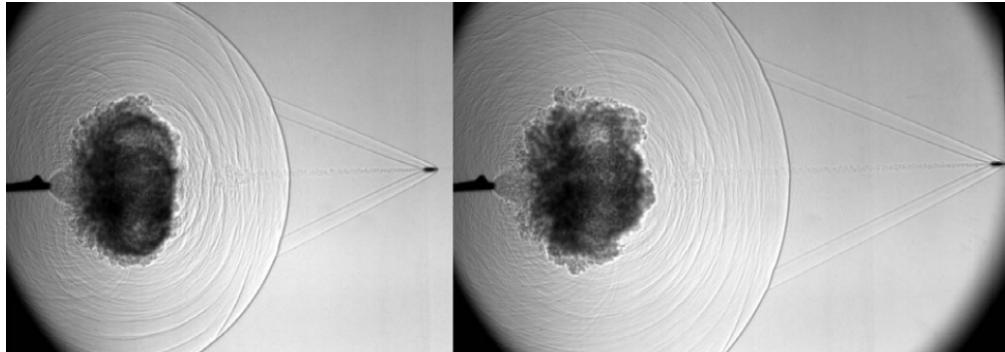


Figure 2.6: Consecutive shadowgrams of the firing of a Remington .30-06 deer rifle (Settles et al., 2005).

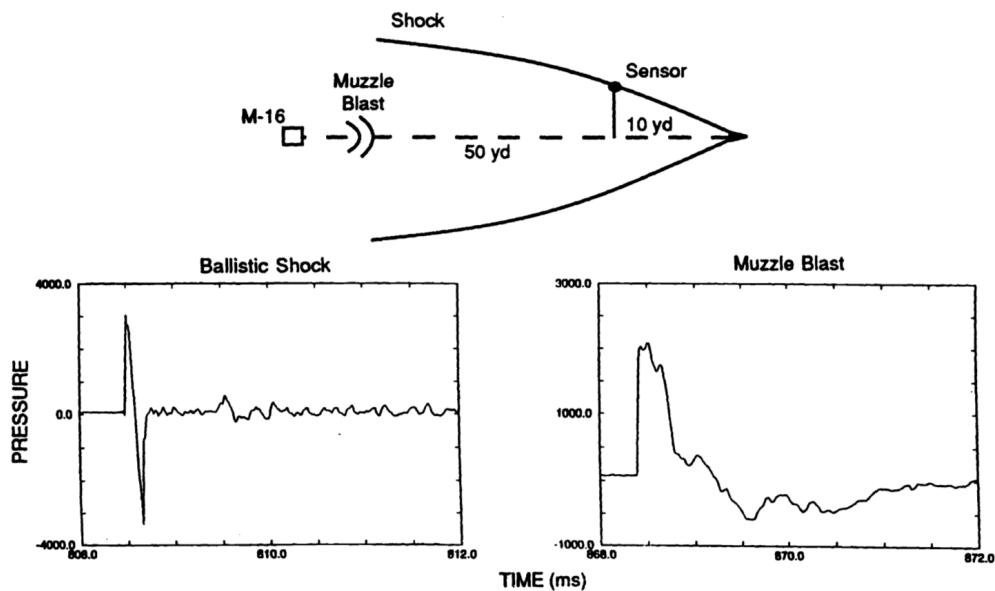


Figure 2.7: Example pressure time series of ballistic shock wave and gun blast wave recorded from an M-16 rifle. Pressure units here are A/D counts (Stoughton, 1997).

acoustic detection. However, as discussed below, this is the only acoustic signature available for the purpose of detection and localization in the case of a subsonically-moving bullet.

2.1.2 Shock waves generated by supersonically-moving projectiles

If the bullet travels faster than the speed of sound then it trails a shock wave in the form of a cone behind it; a shadowgram of this case is shown in fig. 2.6. The leading edge of the bullet suddenly compresses the air in front of it and its trailing edge creates a corresponding expansion. This gives rise to the characteristic ‘N’ wave of a shock shown in fig. 2.7. The duration of this pulse is related to the calibre of the bullet. The amplitude is primarily a function of the ‘miss distance’ – how far away from the microphone the bullet passes. In particular, a microphone placed behind the gun will never detect the shock wave from the bullet (see also fig. 2.2b).

It is clear from fig. 2.6 and 2.7 that the shock wave from a supersonically-travelling bullet reaches the microphone before the muzzle blast. Also, the frequency and amplitude characteristics of the

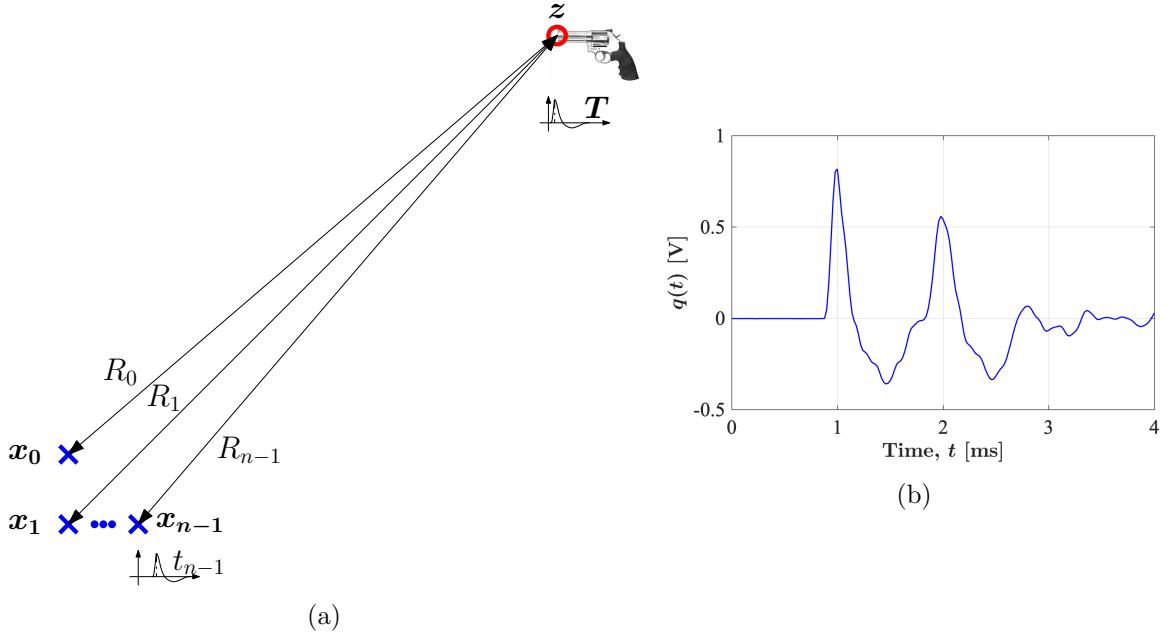


Figure 2.8: (a) Schematic of setup to record gunshot acoustics on a microphone, and the ground reflected muzzle blast arriving at it (adapted from Maher (2006)). (b) The audio recording of a Glock pistol’s muzzle blast, with ground reflection.

two waves are quite distinct. Algorithms may be devised to automatically classify such signatures. The shock-wave data from multiple sensors may yield information on the trajectory of the bullet. Additional modelling of ballistics, like accounting for atmospheric deceleration of bullets, drooping of trajectory due to gravity, and precise categorization of the bullet, may allow the approximate determination of the shooter’s location. However, all this is of course contingent on the bullet being supersonic.

2.1.3 Effect of reflection

The major reason for human confusion regarding the origin of an acoustic event (like gunfire) is echo from the ground and other reflectors. (Another important reason is of course the pre-occupation of the mind, for instance with a difficult combat situation.) Although the echo of an event always arrives at the ear *after* the sound of the actual event, the distinction may not be perceived clearly by the brain, especially in a state of emergency. This may cause a reflection point to be mistaken for the origin of the actual event. The situation is further aggravated in reverberative environments due to multiple echoes.

Figure 2.8 exemplifies the issue with an audio recording of the firing event of a pistol. Due to reflection from the ground, a delayed and attenuated replica of the muzzle blast is recorded at the microphone. Similar reflection issues also arise in the case of the bullet shock wave. Subsequently, we are going to use this understanding to explain an algorithm for mitigating the effect of such reflections on our localization efforts.

Subtle differences in time-of-arrival are of course easily resolved by electronic means (as in the system developed here), allowing unambiguous localization of the original event.

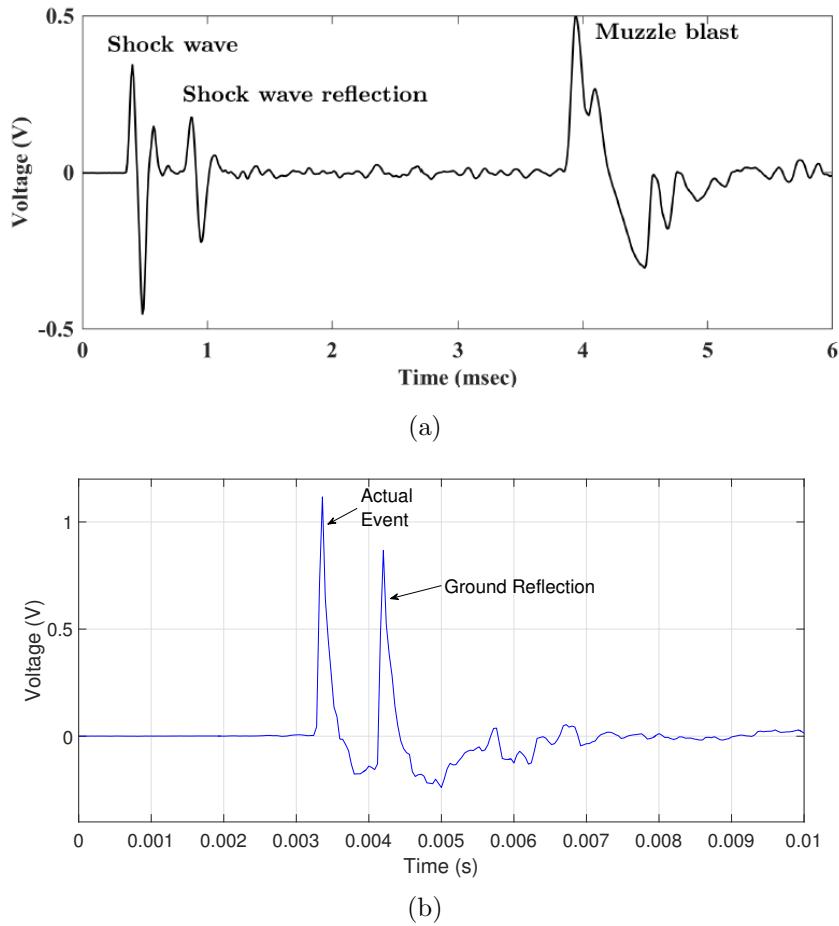


Figure 2.9: Acoustic signature of (a) an AK-47 rifle, and (b) a firecracker.

2.1.4 Simulating gun muzzle blast with firecracker

It is not always easy to get access to live gunfire for testing and validation of our targeting device. Hence, we explored the possibility of using some other sound source to simulate gunfire. Figure 2.9 demonstrates that firecrackers have an acoustic signature that is not too dissimilar from the muzzle blast of a gun – an AK-47 rifle in particular. Both waveforms display the sharp rise, followed by a gradual fall, a negative excursion, and then eventual dissipation. Even the time duration of the pulses are of the same order – about 1 to 2 ms. Thus, we made extensive use of firecrackers in the evaluation of our device. Of course, there is no such simple substitute for the ballistic shock wave.

2.2 Gunshot Localization Preliminaries

Figure 2.10 presents the basic software architecture for gunshot localization. One has to begin by detecting a gunshot event at one or more microphones in real time. This must be followed by a pre-processing step to remove reflection artifacts from the signal. Subsequently, one must classify the acoustic event as that of a gun muzzle blast, a bullet-trailed shock wave, or neither. The identification of the first two classes of events then trigger further steps of cross-correlation across microphone channels (more about this later) and application of the actual localization algorithm specialized for the respective event. If both the muzzle blast and the ballistic shock are received

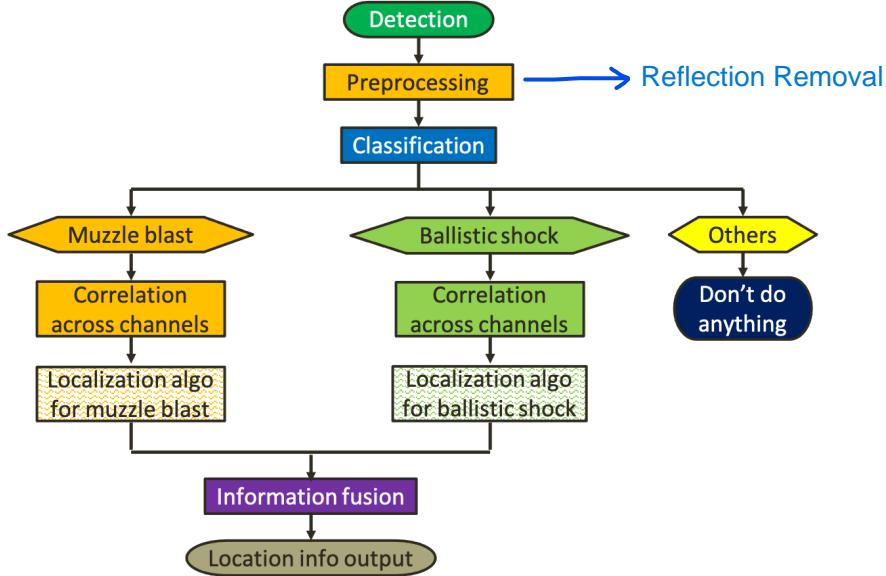


Figure 2.10: Components of gunshot localization software.

unambiguously by the device, then their processed location results may be merged judiciously to obtain a more robust overall answer.

2.2.1 Detection

Detection of an acoustic event of interest is usually done by setting a threshold for the amplitude of the microphone signal. An acoustic event is deemed to have arrived at a sensor if the absolute value of the voltage crosses this threshold. The actual threshold value to use has to be decided by trial and error. If it is too high, then we will miss bona fide events; if it is too low, then we will wrongly identify background noise as gunshots.

Once a sample of any microphone signal crosses the threshold, we start ‘recording’ data. That is, we store a subsequent segment of signal on all microphones in the array. This segment is designed to be long enough to capture the entire duration of the acoustic event at all microphones. On the other hand, if it is too long, then we will increase our computational overhead in subsequent steps, and/or capture more than one acoustic event.

2.2.2 Reflection artifact minimization

A perusal of the reflection-corrupted gunshot signals in fig. 2.8b and 2.9 reveals that the reflection appears as a delayed and diminished version of the original event signature. Thus, if one were to know the differential time of arrival (DTOA) between the main event and its reflection as well as the attenuation factor, then one may be able to minimize the reflection artifact significantly (Braasch, 2013).

Apparently, the simplest way of estimating the DTOA and attenuation is to focus on the sharp positive peaks of the main event and its reflection. However, this approach may be erroneous as it completely ignores the rest of the ‘body’ of the signatures. Moreover, in a sampled signal, one can easily miss sampling the actual peak, as it can occur in between two samples. Thus, a much more robust approach is to use auto-correlation. For later reference, we first introduce the concept of cross-correlation, which is a generalization of auto-correlation.

2.2.2.1 Cross-correlation

Let us consider two signals $q(t)$ and $r(t)$, both sampled with periodicity Δt . For simplicity, we will assume that both signals have N samples available; this simplification holds true for our implementation. Then, we denote the two sampled sequences as $\{q[i]\}_{i=0}^{N-1}$ and $\{r[i]\}_{i=0}^{N-1}$, where $q[i] := q(i\Delta t)$, and so on. In our notation, square brackets for the argument indicates the sampled sequence counterpart of the continuous signal that has parentheses for the corresponding argument.

The sequences of interest here are square-summable – i.e., they vanish towards both ends. Basically, this means that acoustic events have a finite duration, outside of which the sequences relax to zero. The cross-correlation between two such sampled sequences q and r at a shift of k samples is given by

$$R_{qr}[k] := \sum_{i=\max(0,-k)}^{N-1-\min(0,-k)} q[i+k]r[i]. \quad (2.1)$$

That is, the cross-correlation of two sequences q and r at a shift of k samples is the sum of the sample-wise product of the two sequences with the first one shifted ahead by k samples. The summation limits account for the finiteness of the sequences. The auto-correlation is obtained when both input sequences to the cross-correlation are the same.

To make better sense of the cross-correlation, it is common to report the cross-correlation coefficient instead, which is given as

$$\rho_{qr}[k] := \frac{R_{qr}[k]}{\sqrt{R_{qq}[0]R_{rr}[0]}}. \quad (2.2)$$

It will be noted that $\sqrt{R_{qq}[0]}$ and $\sqrt{R_{rr}[0]}$ are simply the root mean square (RMS) of the q and r sequences, respectively. The cross-correlation coefficient has the advantage of being bounded within $+1$ and -1 . Two sequences that are identical for a particular time shift will have $\rho = +1$ for that shift k . On the contrary, if the two sequences are equal and opposite of each other for a particular shift, then $\rho = -1$. Thus, the cross-correlation coefficient gives the degree of similarity between two sequences, modulo all possible time shifts. In particular, the auto-correlation coefficient of all sequences are unity at zero lead/lag.

2.2.2.2 Using auto-correlation to minimize reflection artifact

As an example, consider the raw voltage signal (output of microphone) of fig. 2.11a recorded for a Glock pistol firing at 50 kHz sampling rate. It evidently has a strong reflection artifact. The attenuation ratio α of the reflection amplitude to the main event amplitude is 0.66, and the two are separated by $k^* = 49$ samples (corresponding to 0.98 ms). The reflection artifact appears as the second peak in the auto-correlation of the sequence shown in fig. 2.11b. The ratio γ of the second peak of R_{qq} to the peak at zero shift, was shown by Braasch (2013) to be related to the attenuation ratio α as

$$\alpha = 0.5(1 - \sqrt{1 - 4\gamma^2})/\gamma. \quad (2.3)$$

We measure $\gamma = 0.4512$, which gives $\alpha = 0.63$. Also, the second peak occurs at $k^* = 49$. Both are sufficiently close to the corresponding quantities estimated directly from the sequence.

Next, we delay the original sequence $q[i]$ by k^* and scale it down by the factor α , as shown in fig. 2.11c. Evidently, the main event now matches quite well with the reflection artifact. Subtraction of the shifted and scaled sequence $\alpha q[i-k^*]$ from the original sequence yields a result (see fig. 2.11c) that has significantly reduced reflection artifact. However, the original reflection artifact now

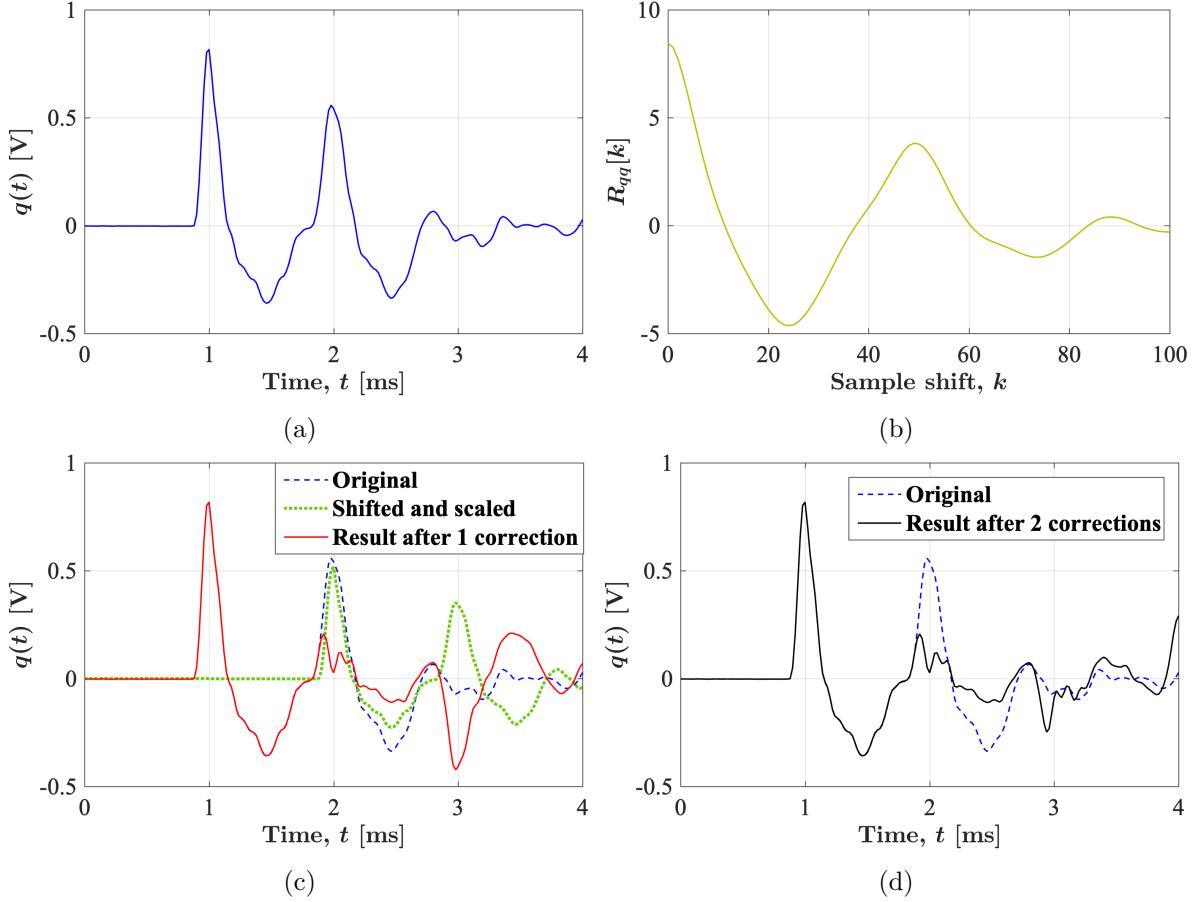


Figure 2.11: (a) Muzzle blast signature of a Glock pistol with reflection artifact. (b) The auto-correlation of the sequence. (c) Result of reflection artifact minimization effort after subtraction of delayed and scaled version of the original from itself (first correction). (d) Result after two such corrections, demonstrating further minimization of the reflection artifact.

appears as a further delayed, scaled and *inverted* effect (at around 3 ms). If required, this too can be minimized by *adding* the further shifted and scaled sequence $\alpha^2 q[i - 2k^*]$ to the first result, as shown in fig. 2.11d. In fact, the general formula for applying M corrections is

$$\hat{q}[i] = \sum_{j=0}^M (-\alpha)^j q[i - jk^*]. \quad (2.4)$$

Usually, $M = 2$ or 3 corrections are enough for the purposes of cleaning up the gunshot acoustic event sequence sufficiently for subsequent processing.

2.2.3 Gunshot event categorization

The muzzle blast and the bullet shock wave have fundamentally different propagative properties, and hence must be processed differently for shooter localization. Thus, event categorization is essential to distinguish between these two signatures of a gunshot, as well as discriminate against all other similar impulsive sounds.

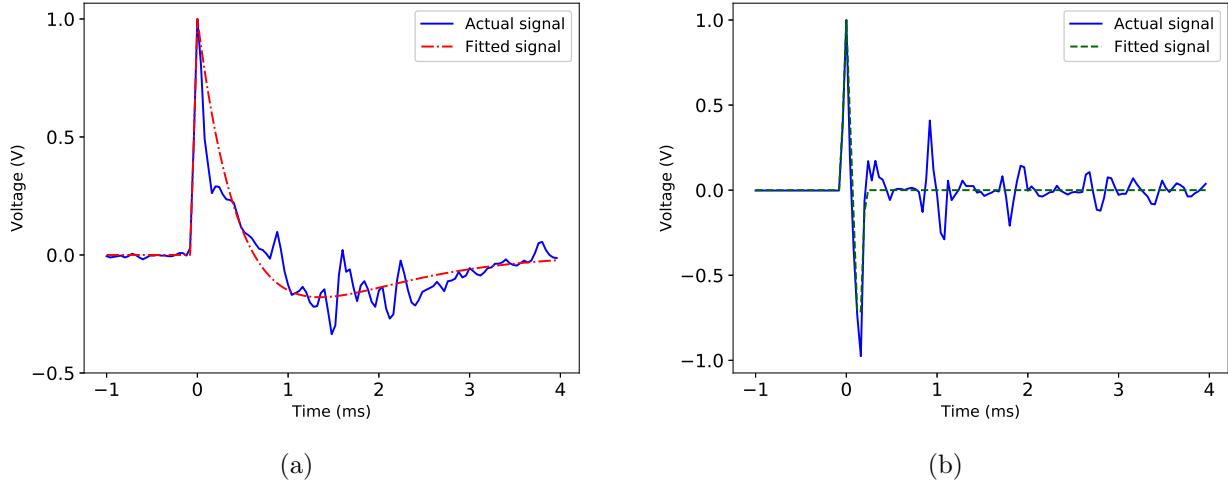


Figure 2.12: Distinctive signatures of a light machine gunfire fitted with corresponding waveforms. (a) Muzzle blast fitted with the Friedlander wavef. (b) Bullet shock wave matched to an ‘N’ wave.

Looking at the representative gunshot signal in fig. 2.9a, one notices clear distinction between the muzzle blast and ballistic shock wave in terms of duration and shape that may be exploited for categorization. We have found it more convenient to use the wave shape for this purpose.

Figure 2.12a demonstrates that the following modified ‘Friedlander wave’ is a very good fit for muzzle blast signatures:

$$f_{\text{Friedlander}}(t) = \begin{cases} 0, & t \leq t_0, \\ A(t - t_0)/t_r, & t_0 < t \leq t_0 + t_r, \\ A[1 - (t - t_0 - t_r)/t_d] e^{-(t-t_0-t_r)/t_d}, & t > t_0 + t_r. \end{cases} \quad (2.5)$$

Here, A is peak amplitude of the event signature, t_0 is its time of arrival, t_r is its initial rise time, and t_d characterizes its subsequent decay time. The function are only four parameters to be determined by a **nonlinear least squares fitting exercise**. A common simplification is to **neglect the rise time altogether** (Beck et al., 2011), which reduces the number of free parameters to three.

On the other hand, fig. 2.12b substantiates that the following ‘N’ wave (Maher, 2006) is an excellent representation of typical ballistic shocks:

$$f_{\text{N-wave}}(t) = \begin{cases} 0, & t \leq T_0, \\ B(t - T_0)/T_r, & T_0 < t \leq T_0 + T_r, \\ B[1 - 2(t - T_0 - T_r)/T_d], & T_0 + T_r < t < T_0 + T_r + T_d, \\ B[(t - T_0 - T_r - T_d)/T_r - 1], & T_0 + T_r + T_d < t < T_0 + 2T_r + T_d, \\ 0, & t > T_0 + 2T_r + T_d. \end{cases} \quad (2.6)$$

Here, B is the amplitude of the peak (both positive and negative), T_0 is its time of arrival, T_r is the common rise times of the initial and final segments, and T_d is the signal ‘duration’ from the positive to the negative peak. Thus, there are again only four parameters to be determined by the nonlinear least squares fitting procedure. In fact, one can even neglect the two rising sections owing to their extreme rapidity, thereby bringing down the parameter count to three.

The overall algorithm for event categorization goes as follows. We try to fit the event signature with both the Friedlander and the ‘N’ wave shapes, and determine their respective goodness-of-fit (say, the chi-square measure). The better fit determines the category of the event – muzzle blast

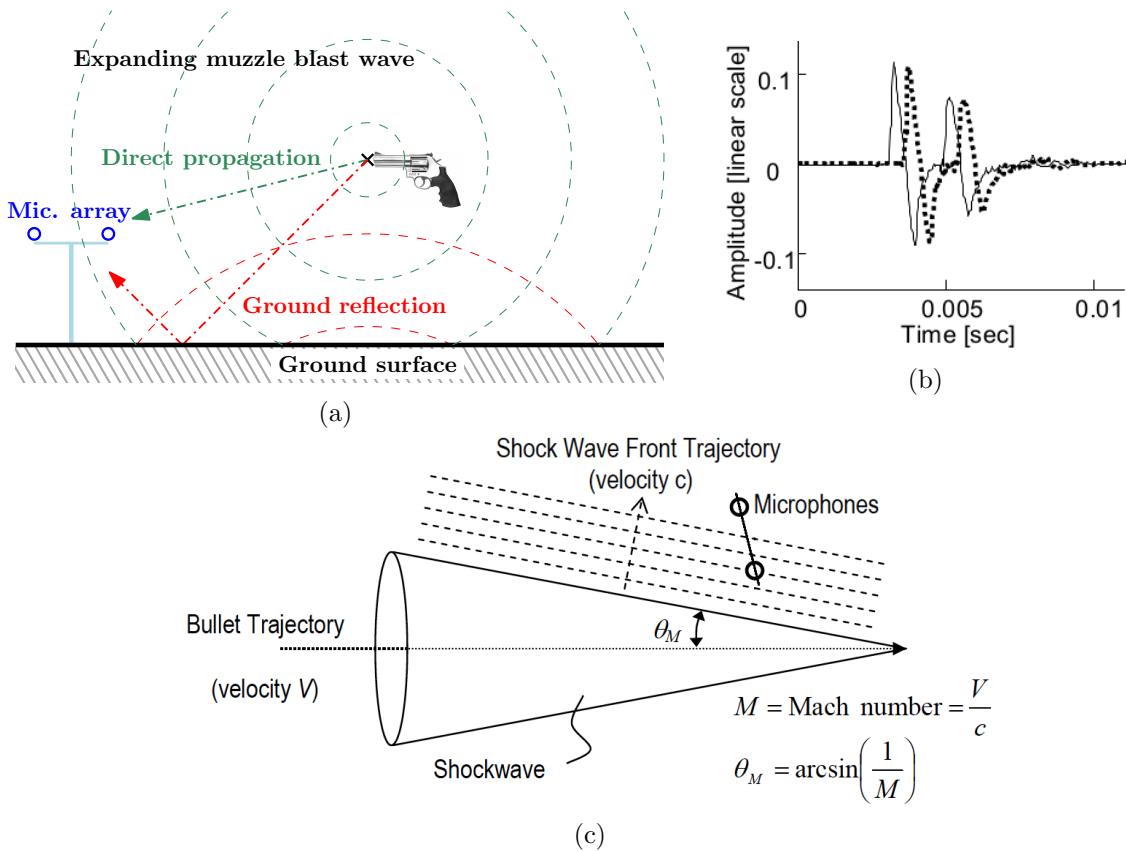


Figure 2.13: (a) Schematic of propagation of muzzle blast and its ground reflection near a two-microphone array. (b) The resulting 2-channel audio recording of an HK USP compact handgun's muzzle blast (solid line is for the nearer microphone; dotted line is for the other one). (c) Schematic of bullet shock wave propagation over a two-microphone array. Adapted from Maher (2006).

or ballistic shock wave. If neither fit is deemed good enough, then the event is not considered to be from a gunshot.

The present algorithm can properly categorize the event only if the sequence has either a muzzle blast or a ballistic shock signature. If both the acoustic signatures are present simultaneously in the abstracted segment of the signal, then the classification cannot be trusted.

2.2.4 DTOA calculation using cross-correlation

Per the overall flowchart presented in fig. 2.10, the next step is the calculation of differential time of arrival (DTOA) of an acoustic event at two sensors. The schematic of the propagation of gunshot acoustic waves (muzzle blast and ballistic shock wave) in the vicinity of two microphones in array shown in fig. 2.13a and 2.13c clarifies the centrality of DTOA in the process. Owing to the spatial separation of the microphones in an array, the acoustic wave from the gunshot (be it the muzzle blast or the ballistic shock) arrives with a slight delay at the farther microphone compared to the nearer one. This is displayed by the two-channel recording of a handgun's muzzle blast in fig. 2.13b. We will be relying entirely on the DTOA for each pair of microphones in an array to localize the origin of the gunfire. Here we discuss the method for estimating the DTOA.

A study of the representative two-channel recording of fig. 2.13b indicates that the signature

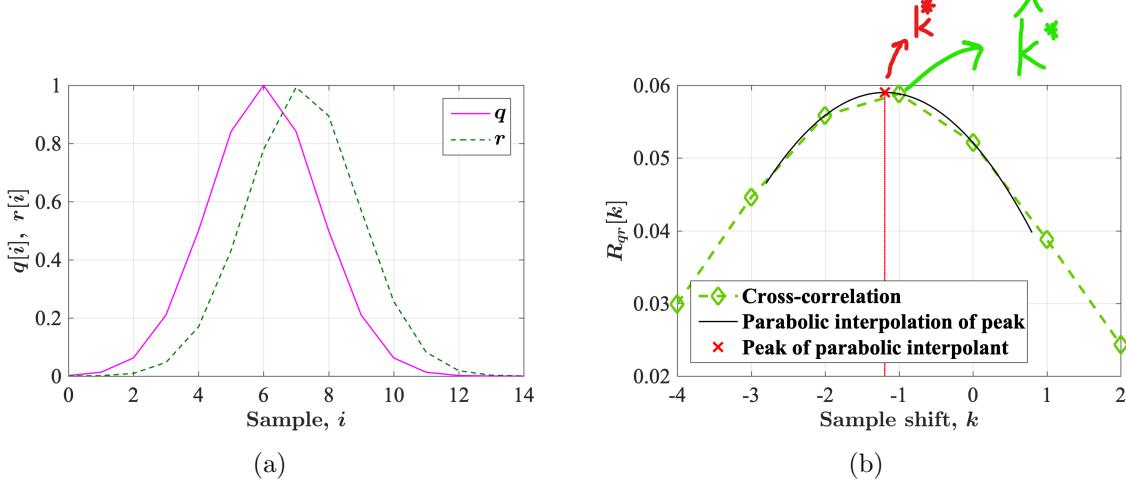


Figure 2.14: (a) Two discretely-sampled Gaussian signals q and r , both having half-width of two samples, with r delayed by 1.2 samples relative to q . (b) Parabolic interpolation of the peak of the cross-correlation of these two signals to refine the estimate of the (fractional) sample delay.

in the two channels are essentially time-shifted and amplitude-scaled versions of each other. This is of course similar to the reflection artifact reduction problem discussed in Section 2.2.2, although there the two versions of the signature (the original and the reflected) were recorded on the same channel. In that context, we have discussed the reasons for not relying on the shift and scaling of the positive peaks of the two versions for estimating the overall shift and scaling of the signatures. Instead, we showed that the auto-correlation of the channel signal delivers a robust estimate of these necessary parameters. Extending the same idea to the present scenario, the DTOA can be best estimated as the sample count k^* corresponding to the peak of the cross-correlation of the two channels' signals. This k^* is of course the number of samples by which one signal has to be shifted (advanced or delayed) to match best with the other signal.

2.2.4.1 Parabolic interpolation of cross-correlation peak

The cross-correlation of two discretely sampled signals is also available at integer sample shifts. Thus, it will peak at an integer sample shift (\hat{k}^* , say), whereas the true DTOA is fractional in general. At worse, such discretization error in the DTOA estimate can be $\Delta t/2$. Since the accuracy of the DTOA estimate is of utmost importance for the success of the gunshot localization, we use the following parabolic interpolation technique to refine the estimate.

To exemplify the procedure, consider two sampled Gaussians signals shown in fig. 2.14a, both having half-width of 2 samples, but one being delayed from the other by 1.2 samples. The cross-correlation of these two signals is seen in fig. 2.14b to peak at $\hat{k}^* = -1$, which will be taken as the first approximation of the DTOA. However, we can improve upon this by fitting a parabola through the peak point of the cross-correlation sequence $(\hat{k}^*, R[\hat{k}^*])$ and its two side points on either side $(\hat{k}^* - 1, R[\hat{k}^* - 1])$, and $(\hat{k}^* + 1, R[\hat{k}^* + 1])$, as shown in fig. 2.14b. Then, the location k^* of the peak of this fitted parabola serves as a more refined estimate of the DTOA.

The general expression for the fitted parabola is of course $\mathcal{R}(k) = ak^2 + bk + c$, where k is now allowed to be a real number. The three parameters a , b and c can be ascertained by requiring that the above equation be exactly satisfied at the three points near the peak of the cross-correlation $R[k]$, as mentioned above. Some elementary manipulation then yields the k^* corresponding to the

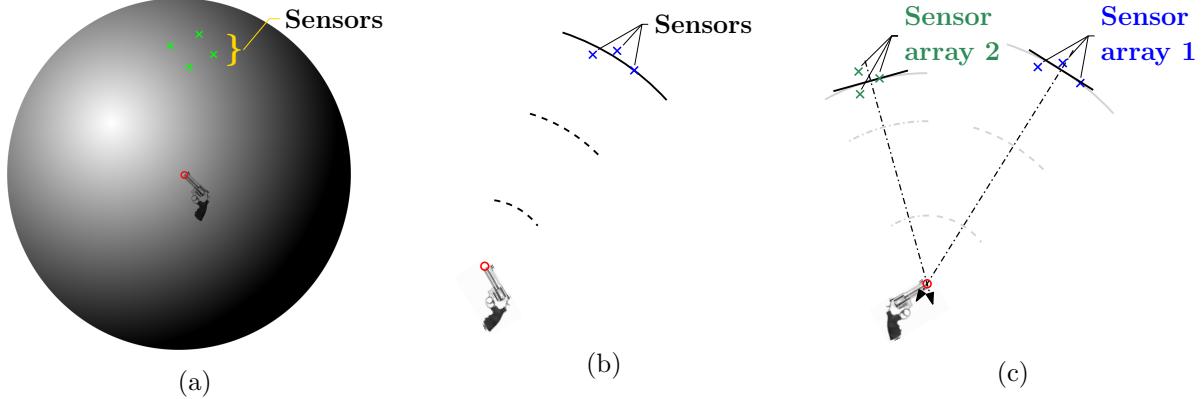


Figure 2.15: (a) Unlikely scenario of four sensors just happening to be on a spherical muzzle blast wavefront centered on shooter. (b) A realistic situation in 2-D where sensors experience differences of time-of-arrival of the circular blast wave. (c) With multiple sensor arrays, the wavefront’s local curvature can be neglected in favour of triangulation using two directions-of-arrival for localization.

peak of this parabola as

$$k^* = -\frac{b}{a} = \hat{k}^* + \frac{1}{2} \frac{R[\hat{k}^* - 1] - R[\hat{k}^* + 1]}{R[\hat{k}^* - 1] - 2R[\hat{k}^*] + R[\hat{k}^* + 1]}. \quad (2.7)$$

For our example, $R[\hat{k}^* - 1]$, $R[\hat{k}^*]$ and $R[\hat{k}^* + 1]$ are respectively 0.0559, 0.0588 and 0.0521. From these, we calculate $k^* = -1.198$ samples, which is much closer to the true value of -1.2 samples. As a final note, if one were to use the two signals’ peak indices for DTOA estimation alone, then the best outcome for the current example would have been -1 sample.

2.3 Algorithms for localizing gun-shooter from muzzle blast

As described in Section 2.1.1, the muzzle blast propagates **approximately spherically** in all directions from the gun muzzle. Thus, if one were able to identify any spherical wavefront of the blast, then the shooter may be directly located at its center. To define a spherical surface in **three dimensions**, one needs to know the coordinates of **at least four points** on it (see fig. 2.15a). However, it will indeed be unlikely that four arbitrarily arrayed microphones just happen to be on the spherical muzzle blast wavefront of a particular gunshot. The much more likely scenario is that the same muzzle blast is recorded at four or more microphones, but with some relative time delays (i.e., the DTOAs that were discussed earlier). Even in this case, we will show below that one can localize the shooter if one knows the ambient speed of sound.

A significantly simpler scenario, which still happens to be almost equally relevant, is the corresponding two-dimensional problem. Here, we will have an array of at least three microphones in a plane. We will assume that the shooter is also in the same plane, so that the muzzle blast wavefront can be taken to be circular (see fig. 2.15b). This situation is of interest since the difference in altitude between the microphones and the shooter is expected to be insignificant compared to the 3-D separation distance.

The typical situation of interest is when the dimension of the microphone array is much smaller compared to its distance from the possible gun-shooter. A large array is of much less practical utility than a compact, possibly-portable one. Also, precise relative location of the sensors in an

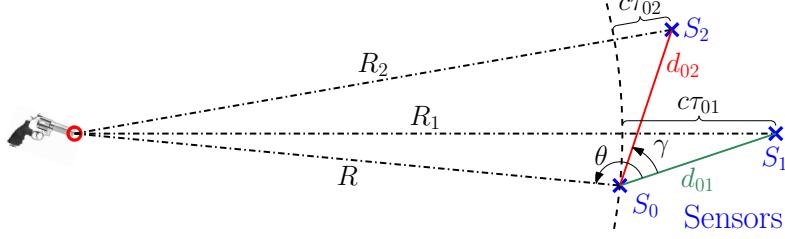


Figure 2.16: Schematic of the wavefront curvature method for 2-D localization of a gun-shooter using the circular muzzle blast wave's acoustic signature.

array is crucial for accuracy of the localization, but this precision becomes difficult to achieve on a large scale. However, for the same reason, the **spherical or circular muzzle blast wavefront** also **becomes very flat** by the time it arrives at the sensor array, so that the **calculation of its center from the DTOA measurements is very problematic**. In other words, we expect significant inaccuracy in the range estimate of a shooter if we use a small array of sensors. It turns out, though, that the **bearing angle** of the shooter relative to the array (also the direction of arrival – DOA – of the muzzle blast wavefront) **can be estimated with much greater precision**. This is because there is a greater degree of redundancy in having three sensors in a 2-D array for this purpose. The above observation suggests that, if we could ‘fuse’ the information regarding DOA from more than one spatially-separated sensor arrays (each possibly small), then we can triangulate the location of the shooter more precisely using the multiple DOAs as shown in fig. 2.15c.

Now we describe the the algorithms proposed in the literature for implementing the various localization philosophies outlined above.

2.3.1 Wavefront curvature method

The wavefront curvature method was proposed by Carter (1981) for locating a gunshot source using its muzzle blast wave. This is a **planar localization technique**, and thus assumes **circular expansion** of the acoustic wavefront. Three microphones placed in a planar arrangement to capture the acoustic wavefront is shown in fig. 2.16. It may be appreciated that three is the minimum number of sensors needed for this purpose. Thus, the wavefront curvature method seeks an *exact* solution to the problem, instead of a *robust* one from redundant measurements. In particular, it cannot be used directly with more than three microphones. We will later discuss ways of extending this approach to make it more robust.

As shown in fig. 2.16, the sound waves propagating circularly from the source reaches the sensors S_0 , S_1 and S_2 through wavefronts of radius R , R_1 and R_2 respectively. Taking S_0 as the origin of the sensor triplet, and the $S_0 - S_1$ arm of the array as the axis reference, the bearing angle θ and the range R of the source can be calculated using the inter-sensor distances d_{01} and d_{02} , the included angle γ , the ambient speed of sound c , and the measured DTOAs τ_{01} and τ_{02} (see fig. 2.16). Here, $\tau_{ij} := t_j - t_i$, with t_k being the time of arrival of the wavefront at the sensor S_k . Note that $\tau_{ij} = -\tau_{ji}$. From the geometry of the problem shown in fig. 2.16, one can observe that

$$R_1 = R + c\tau_{01}. \quad (2.8)$$

Note that τ_{01} will be negative if S_0 is farther from the source than S_1 , but the above relation will still hold. Now, from the cosine law for the triangle formed by S_0 , S_1 and the source, with the

included angle being the bearing angle θ , we get

$$\cos \theta = \frac{R^2 + d_{01}^2 - R_1^2}{2Rd_{01}} \implies R_1 = \sqrt{R^2 + d_{01}^2 - 2Rd_{01} \cos \theta}.$$

Eliminating R_1 from the above two equations, we get (after some algebraic manipulation)

$$R = \frac{d_{01}}{2} \frac{1 - \eta_{01}^2}{\cos \theta + \eta_{01}}, \quad (2.9)$$

where, we have defined the normalized DTOA between the sensor pair $S_i - S_j$ as

$$\eta_{ij} := c\tau_{ij}/d_{ij}. \quad (2.10)$$

Note that $-1 \leq \eta_{ij} \leq +1$ for all possible i and j .

A very similar derivation for the triangle formed by S_0 , S_2 and the source (with the included angle being $\theta - \gamma$) gives

$$R = \frac{d_{02}}{2} \frac{1 - \eta_{02}^2}{\cos(\theta - \gamma) + \eta_{02}}. \quad (2.11)$$

After eliminating the range R from the above two equations, and applying some trigonometric identities, the expression for the bearing angle θ is obtained as

$$\theta = \tan^{-1} \left(\frac{\sin \gamma}{\cos \gamma - \delta} \right) \pm \cos^{-1} \left(\frac{\delta \eta_{01} - \eta_{02}}{\sqrt{1 + \delta^2 - 2\delta \cos \gamma}} \right), \quad \delta := \frac{d_{02}(1 - \eta_{02}^2)}{d_{01}(1 - \eta_{01}^2)}. \quad (2.12)$$

Finally, the range R is calculated by substituting the above bearing angle solution in eqn. (2.9).

It is evident from the bearing angle estimation formula of eqn. (2.12) that there are two mathematically-possible solutions delivered by the method. An in-depth study of the solution space was conducted by Kothiwala (2017), who concluded that one solution invariably corresponds to a very nearby source location, whereas the other one is much farther away from the sensor array. Since most practical cases will be of the latter type, one can safely ignore the nearby solution, and say that the farther one is *the* solution of the wavefront curvature method.

For later reference, one can obtain a measure of the accuracy of the source location estimate as

$$\epsilon_{wc} = |R_2 - R_1 - c\tau_{12}|. \quad (2.13)$$

Although, mathematically this error should be zero, in reality it is never so due to the inevitable finiteness of the sampling rate that limits the accuracy of DTOA measurements.

Note that we obtain a closed-form solution for the location (bearing angle and range) of the gunshot source in the plane of the sensor arrangement for an exactly-determined system. That is, the minimal count of three sensors are needed, and no more sensor can be used directly to refine the location estimate. Let us say that we have four sensors in a planar array instead of three. Then, one way to continue using the wavefront curvature method is to create sets of three sensors, estimate the shooter location from each such triplet, and finally take some kind of average of all the results. However, there are other methods that more directly exploit such sensor redundancy to deliver more reliable estimate of the source location, as described next.

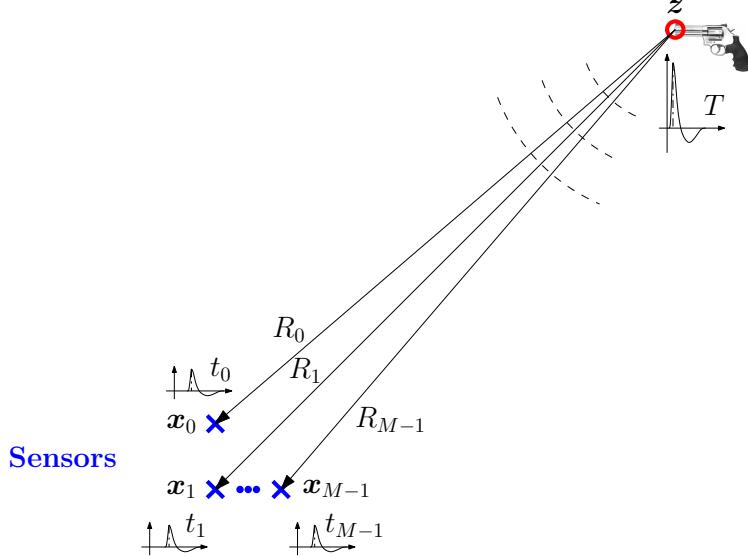


Figure 2.17: Schematic of 2-D shooter localization using the gun’s muzzle blast acoustic signature captured on $M \geq 3$ sensors.

2.3.2 Nonlinear least-squares method

The wavefront curvature method is a geometric approach to solving the localization problem for an exactly-determined system. In particular, we can only work with three sensors’ data (at a time) in two-dimensional cases. Apart from the difficulty in extending it to three-dimensional localization problems, it is also not directly applicable to the more robust case where we happen to have more sensors in an array than the strict minimum. In these general scenarios, the problem of localization may be formulated as an algebraic nonlinear least-squares problem, and solved using the classical **Gauss-Newton method** Sirola (2010).

Referring to fig. 2.17, the straightforward muzzle blast propagation equations can be written down for M microphones as

$$ct_i = \mathcal{D}(\mathbf{x}_i, \mathbf{z}) + cT, \quad i \in \{0, 1, \dots, M-1\}.$$

Here, T is the instant of gunfire at the position \mathbf{z} (which is a column vector), t_i is the time-of-arrival (TOA) of the muzzle blast at the i th microphone located at \mathbf{x}_i , c is the ambient speed of sound, and $\mathcal{D}(\cdot, \cdot)$ is the distance function that takes as argument two position vectors and gives as output the distance between them. Referring to our setup in fig. 2.17, $\mathcal{D}(\mathbf{x}_i, \mathbf{z})$ is exactly equivalent to the ‘range’ R_i ; the notation simply renders the solution variable \mathbf{z} explicit.

For an N -dimensional problem (where $N = 2$ or 3), all position vectors are N -dimensional column vectors. For later use, we introduce the notation for the j th coordinate of a general position vector \mathbf{y} as y_j . In particular, the j th coordinate of the source position vector is z_j , and that of the i th sensor’s position vector is x_{ij} .

The similarity of this setup with the GPS problem may be noted. There, \mathbf{x}_i represents the known instantaneous location of the i th satellite that emitted a signal at time t_i which happens to reach the GPS device located at the (unknown) \mathbf{z} at the instant T of interest.

Now, in our gunshot localization problem, we do not know and neither do we need to know the instant of gunfire T . To avoid its occurrence in our equations, we can subtract the equation for a reference sensor, say sensor $i = 0$, from the equations for all other sensors. Then, we obtain

$M - 1$ equations in terms of the differential times of arrival DTOAs (i.e., $\tau_{0i} = t_i - t_0$) instead of the TOAs:

$$c\tau_{0i} = \mathcal{D}(\mathbf{x}_i, \mathbf{z}) - \mathcal{D}(\mathbf{x}_0, \mathbf{z}), \quad i \in \{1, \dots, M - 1\}. \quad (2.14)$$

Here, the known quantities are the M sensors' locations $\{\mathbf{x}_i\}_{i=0}^{M-1}$, the ambient speed of sound c , and the $(M - 1)$ DTOAs $\{\tau_{0i}\}_{i=1}^{M-1}$. The unknowns are the two or three coordinates of the source position vector \mathbf{z} , in 2-D and 3-D respectively. Evidently, we need at least two (resp. three) such equations to solve for \mathbf{z} in two (resp. three) dimensions. Thus, the number of sensors required is $M \geq 3$ (resp. 4) in two (resp. three) dimensions, as concluded earlier from purely geometric arguments.

In over-determined systems (i.e., when there are more than the bare minimum number of sensors), we cannot expect an exact solution. Due to various sources of inaccuracy (measurement, estimation, etc.), the ‘true’ source location solution may not exactly satisfy any of the equations. In this situation, the best that one can hope for is to *minimize* the error in satisfying the set of equations. Thus, we define the errors or ‘residuals’ of the $M - 1$ nonlinear equations as

$$r_i(\mathbf{z}) := c\tau_{0i} - \{\mathcal{D}(\mathbf{x}_i, \mathbf{z}) - \mathcal{D}(\mathbf{x}_0, \mathbf{z})\}, \quad i \in \{1, \dots, M - 1\}. \quad (2.15)$$

In the above, we have made it explicit that the sole parameter of r_i is the source position vector \mathbf{z} , all other quantities being given. For ease of notation, we introduce the $(M - 1)$ -dimensional vector of residuals $\mathbf{r} := [r_1, r_2, \dots, r_{M-1}]^T$, where $(\cdot)^T$ denotes the matrix transpose. The Gauss-Newton algorithm iteratively finds the value of \mathbf{z} that minimizes the sum of the squares of the residuals:

$$S(\mathbf{z}) := \sum_{i=1}^{M-1} |r_i(\mathbf{z})|^2 = \mathbf{r}(\mathbf{z})^T \mathbf{r}(\mathbf{z}).$$

The algorithm requires an initial guess for the solution, say $\mathbf{z}^{(0)}$. This can come from, say, an application of the wavefront curvature method with a set of minimal sensors. Then, the $(k + 1)$ th iteration of the solution, $\mathbf{z}^{(k+1)}$, is given by

$$\mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} - \left(J(\mathbf{z}^{(k)})^T J(\mathbf{z}^{(k)}) \right)^{-1} J(\mathbf{z}^{(k)})^T \mathbf{r}(\mathbf{z}^{(k)}). \quad (2.16)$$

Here, $J(\mathbf{z}^{(k)})$ is the Jacobian matrix of the residual vector function $\mathbf{r}(\mathbf{z})$ evaluated at the k th iterate of the solution $\mathbf{z} = \mathbf{z}^{(k)}$. In particular, the entry in the i th row and j th column of the J matrix, evaluated at a trial source position vector \mathbf{z} , is

$$J_{ij}(\mathbf{z}) := \frac{\partial r_i}{\partial z_j} = \frac{x_{ij} - z_j}{\mathcal{D}(\mathbf{x}_i, \mathbf{z})} - \frac{x_{0j} - z_j}{\mathcal{D}(\mathbf{x}_0, \mathbf{z})}.$$

The Jacobian matrix evidently has $M - 1$ rows and N columns. If it were square (i.e., if the system of equations was exactly determined), then its inverse would have appeared in eqn. (2.16) as $\mathbf{z}^{(k+1)} = (\mathbf{z} - J(\mathbf{z})^{-1} \mathbf{r}(\mathbf{z}))|_{\mathbf{z}=\mathbf{z}^{(k)}}$. In the general over-determined system, it is the Moore-Penrose pseudo-inverse of J , given by $J^\dagger := (J^T J)^{-1} J^T$ that appears in eqn. (2.16).

The iterations can be limited by one or more of the following stopping criteria: (a) if the residual is below some preset tolerance ϵ_r (i.e., if $\sqrt{S(\mathbf{z}^{(k)})} \leq \epsilon_r$), (b) if the change in the solution between two iterations is below some tolerance ϵ_z (i.e., $\|\mathbf{z}^{(k)} - \mathbf{z}^{(k-1)}\| \leq \epsilon_z$), and/or (c) if a preset maximum number of iterations is exceeded. If the initial guess is appropriate, the Gauss Newton algorithm provides a very robust approach for source localization.

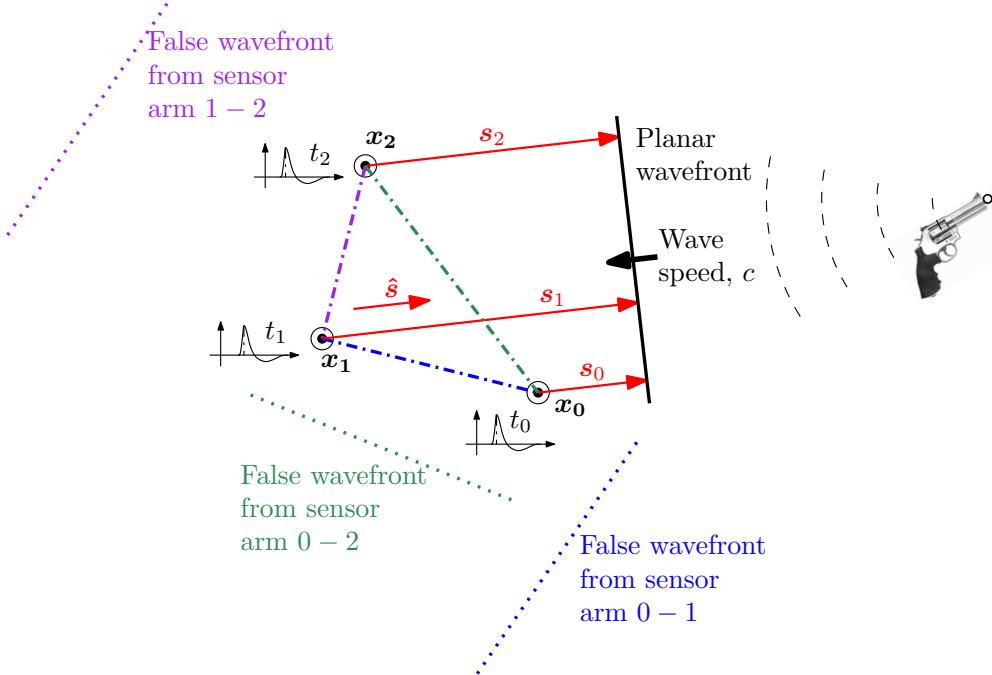


Figure 2.18: Schematic of estimation of the direction-of-arrival (DOA) of a planar wavefront in two-dimensional approach using three sensors in an array.

2.3.3 Bancroft's method

Bancroft (1985) proposed a closed-form solution method for a possibly over-determined localization problem (as considered in Section 2.3.2). The original development was for solving the GPS equations, but the method readily extends to our problem of acoustic localization of gunshots too. The actual algorithm for solving this system of possibly over-determined system is not repeated here; it may be found in the original work of Bancroft (1985).

Similar to the wavefront curvature method, Bancroft's algorithm also involves a quadratic equation leading to two solutions. In order to disambiguate this, we calculate the error in the solution or the residual (using the notation introduced in Section 2.3.2) as

$$\text{Residual} = \sum_{i=1}^{M-1} |\mathcal{D}(\mathbf{x}_i, \mathbf{z}) - \mathcal{D}(\mathbf{x}_0, \mathbf{z}) - c\tau_{0i}|^2.$$

The solution with the smaller residual is chosen as the final solution, as discussed by Sirola (2010).

The method was found to produce poor results for overdetermined systems. For exactly-determined two-dimensional systems, the results are akin to the wavefront curvature method. It also provides a solution for the exactly-determined three-dimensional system, for which the 3-D analogue of the wavefront curvature approach becomes analytically convoluted. We have found the Bancroft method to be less robust than the nonlinear least-squares approach, as observed by Sirola (2010) earlier.

2.3.4 Localization using multiple direction-of-arrival vectors

It was discussed in the context of fig. 2.15c that the muzzle blast signature may be used for more precise localization of the gun shooter if more than one array of sensors is available. It is assumed

that the size of each sensor array is small compared to the range of the shooter, such that the spherical (or circular) muzzle blast wavefront may be considered to be planar (or linear). Then, each sensor array yields a direction of arrival (DOA), and all such DOAs are combined to triangulate upon the shooter location (McNelis, 1996). Note that the determination of the DOA by a sensor array is the basic processing step in localization using the ballistic shock wave too, as discussed next in Section 2.4.

Figure 2.18 presents a detailed schematic of this scenario in the context of two-dimensional localization using a planar array of three sensors S_0, S_1 and S_2 located at $\mathbf{x}_0, \mathbf{x}_1$ and \mathbf{x}_2 , respectively. Let the linear wavefront of the muzzle blast (or the bullet shock wave) be at the location shown at the current time instant t , such that the normal vector from sensor i on to the wavefront is denoted \mathbf{s}_i . The length of each of these vectors is the distance that the wavefront must travel to reach the respective sensors. These vectors are of course all parallel to each other. Let the unit vector along the direction of arrival at the sensor array be $\hat{\mathbf{s}}$; this is called the *unit sighting vector* of the sensor array for the acoustic event under consideration. Note that it is opposite to the direction of propagation of the wave. Also, if the wavefront is yet to reach the sensors, then \mathbf{s}_i are aligned with $\hat{\mathbf{s}}$ (as shown); on the other hand, if the wavefront has already crossed the sensors then they are all opposite to the latter.

Let the time of arrival (TOA) of the wavefront at the i th sensor be t_i . Then, given the ambient sound speed c , we have

$$ct_i = ct + \mathbf{s}_i \cdot \hat{\mathbf{s}}, \quad i \in \{0, 1, 2\}.$$

As before, we convert these TOA-based equations to those involving differential time of arrival (DTOA), by subtracting the equation for the reference sensor (say S_0) from those of the other sensors. It will be recalled that the DTOA between sensors S_i and S_j is denoted by $\tau_{ij} = t_j - t_i$. We thus have

$$c\tau_{0i} = ct_i - ct_0 = \mathbf{s}_i \cdot \hat{\mathbf{s}} - \mathbf{s}_0 \cdot \hat{\mathbf{s}} = (\mathbf{x}_0 - \mathbf{x}_i) \cdot \hat{\mathbf{s}}, \quad i \in \{1, 2\}. \quad (2.17)$$

The last step follows from the geometry depicted in fig. 2.18. The above system of equations can be written in matrix form as

$$\underbrace{\begin{bmatrix} (\mathbf{x}_0 - \mathbf{x}_1)^T \\ (\mathbf{x}_0 - \mathbf{x}_2)^T \\ \vdots \\ (\mathbf{x}_0 - \mathbf{x}_{M-1})^T \end{bmatrix}}_{=: \mathbb{A}} \hat{\mathbf{s}} = \underbrace{\begin{bmatrix} c\tau_{01} \\ c\tau_{02} \\ \vdots \\ c\tau_{0(M-1)} \end{bmatrix}}_{=: \mathbb{D}}. \quad (2.18)$$

For this linear exactly-determined system, the solution is simply obtained as

$$\hat{\mathbf{s}} = \mathbb{A}^{-1} \mathbb{D}. \quad (2.19)$$

Note that the formulation readily extends to 3-D localization problems, where a minimum of four non-coplanar sensors are required in the array.

If in case the number of sensors (say, M) is more than the necessary minimum (three in 2-D localization or four in 3-D problem), then eqn. (2.18) can be extended as

$$\underbrace{\begin{bmatrix} (\mathbf{x}_0 - \mathbf{x}_1)^T \\ (\mathbf{x}_0 - \mathbf{x}_2)^T \\ \vdots \\ (\mathbf{x}_0 - \mathbf{x}_{M-1})^T \end{bmatrix}}_{=: \tilde{\mathbb{A}}} \hat{\mathbf{s}} = \underbrace{\begin{bmatrix} c\tau_{01} \\ c\tau_{02} \\ \vdots \\ c\tau_{0(M-1)} \end{bmatrix}}_{=: \tilde{\mathbb{D}}}. \quad (2.20)$$

Evidently, $\tilde{\mathbb{A}}$ is non-square (it is tall and thin). In this case, one can readily look for a linear least-squares solution using the pseudo-inverse of $\tilde{\mathbb{A}}$ as

$$\hat{\mathbf{s}} = (\tilde{\mathbb{A}}^T \tilde{\mathbb{A}})^{-1} \tilde{\mathbb{A}}^T \tilde{\mathbb{D}}. \quad (2.21)$$

In an actual experiment, there is no guarantee that the length of the unit sighting vector \hat{s} obtained as above will indeed be unity. This may be due to inaccuracies in measurement of the DTOA, the sensor spacing and/or the speed of sound. Another possible source of error (the one considered by McNelis (1996)) is that the sighting vector may be slightly inclined to the sensor plane (due to the shooter being out-of-plane with the sensor array), even though the problem is presumed to be two-dimensional. One workaround is to pose the problem instead as a linear least-squares one (as above for the over-determined system), but with the nonlinear constraint of $\|\hat{s}\| = 1$. This was not pursued in this work.

Returning to eqn. (2.17) for the 2-D problem, note that we arrive at only two possibilities for the DOA just from a single pair of sensors (say S_0 and S_1). One of them is the correct solution, and the other is its mirror image across the line joining the two sensors (see fig. 2.18). Thus, the result from the second pair of sensors can be used to simply confirm one of these, and reject the false one. This demonstrates the greater degree of redundancy (and hence robustness) built into this approach for bearing angle estimation. Of course, the downside is that by neglecting the curvature of the wavefront, we cannot predict the range of the shooter.

2.4 Algorithms for localizing gun-shooter using bullet shock wave

The shock wave trailed by a supersonically moving bullet is the other acoustic signature of a firearm that is useful in localizing the shooter. Before we describe the actual algorithms, we first establish the pertinent geometric features of the bullet shock wave.

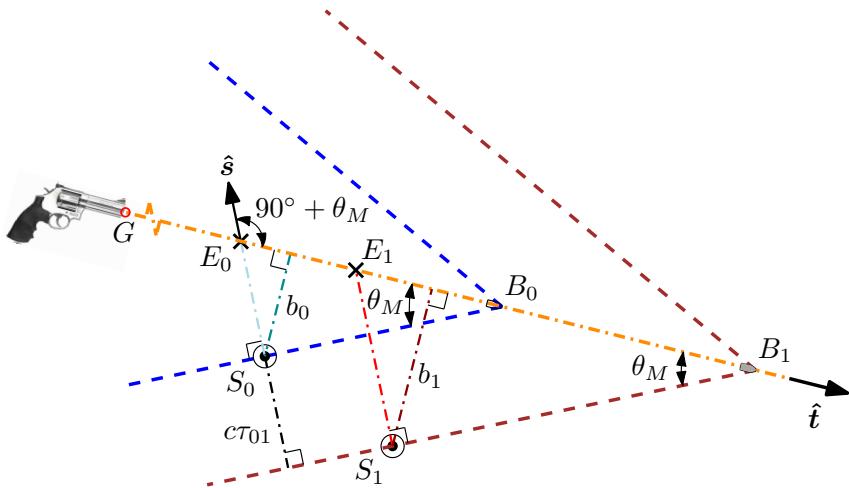
2.4.1 Geometric attributes of ballistic shock wave

A schematic of the bullet shock wave was depicted in fig. 2.13c; this is now presented in greater detail in fig. 2.19. Although the shock wave is a cone in 3-D, it will be a pair of plane waves in 2-D, as shown in the figure. Indeed, we will assume that the bullet trajectory lies in the plane of the sensor array. A similar assumption was justified on grounds of practicality in the context of localization using the muzzle blast signature in Section 2.3.

Let the bullet be moving at speed V_b that is greater than the ambient sound speed c . The bullet actually decelerates due to aerodynamic drag, so that its speed does not remain constant over its trajectory. However, in this work, we will only be interested in the local speed of the bullet over a small segment of its trajectory, which can be assumed to be constant. Its Mach number is $M := V_b/c > 1$. The shock wave trailed by the bullet propagates normal to itself at speed c . Then, the geometry depicted in fig. 2.19 implies that the shock wave makes an angle to the bullet trajectory given by the Mach angle $\theta_M = \sin^{-1}(1/M)$. This means that a slower bullet will have a broader Mach cone compared to a faster one.

The ballistic shock wave that reaches a sensor, say sensor index i , was emitted by the bullet when it was at the corresponding emission point E_i (see fig. 2.19). By the time the shock wave actually reaches S_i , the bullet has already moved on to the point B_i . The nearest that the bullet comes to a sensor is called its ‘miss distance’; the miss distance for the i th sensor is denoted b_i .

In the context of gunshot acoustic event categorization in Section 2.2.3, it was shown that the ballistic shock wave appears as an ‘N’ wave in a microphone signal. The functional form of the ‘N’ wave given in eqn. (2.6), consists of a rise time T_r to a positive peak, followed by a decay time T_d to an equal negative peak, and terminated by another rise time of the same T_r back to neutral. If one ignores the almost instantaneous rising portions of the wave, then an even simpler approximate function for the over-pressure signal of a shock wave w.r.t. the atmospheric pressure p_0 is (Sadler



Symbol	Description
\hat{t}	Bullet trajectory direction (unit vector)
θ_M	Mach angle corresponding to bullet Mach number M (> 1)
\hat{s}	Shock wave DOA for sensor array (unit vector)
G	Gun location
E_i	Emission point of shock wave that reaches i th sensor
B_i	Bullet position when shock wave reaches i th sensor
b_i	Miss distance for i th sensor (its distance from trajectory)

Figure 2.19: Supersonically moving bullet emanating shock wave recorded by an array of sensors located on one side of the trajectory. Symbols newly introduced here are described in the table.

et al., 1998)

$$p_{sw}(t) - p_0 = \begin{cases} 0, & t \leq T_0, \\ \Delta p_{sw} [1 - 2(t - T_0)/T_d], & T_0 < t < T_0 + T_d, \\ 0, & t > T_0 + T_d. \end{cases} \quad (2.22)$$

Here, T_0 is the time of arrival of the shock wave at a particular microphone, T_d is the duration of the shock wave (from positive to negative peak), and Δp_{sw} is the positive (or negative) over-pressure.

For sensors placed far enough away from slender projectiles moving with supersonic speed in air, Whitham (1952) derived the following simple relations between the geometry of the bullet and the parameters of the ‘N’ wave:

$$\Delta p_{sw} = \frac{0.53p_0(M^2 - 1)^{1/8}d}{b^{3/4}\ell^{1/4}}, \quad \frac{dp_{sw}}{dt} = -\frac{0.58p_0c\sqrt{M^2 - 1}}{Mb} \implies T_d = \frac{1.82Mb^{1/4}d}{c(M^2 - 1)^{3/8}\ell^{1/4}}. \quad (2.23)$$

Here, d and ℓ are respectively the caliber (diameter) and length of the bullet, and b is the miss distance of the sensor from the trajectory.

2.4.2 Estimation of bullet trajectory direction

In this project, we take up the simplest localization problem involving ballistic shock waves, namely estimating only the **direction of the bullet trajectory**. The next problem in terms of complexity would have been locating the trajectory itself (i.e., finding a point on the trajectory, apart from its direction already found above). The final challenge would of course have been locating the shooter

(or the origin of the trajectory). Note that if the interest is in the approximate bearing angle of a distant shooter, then only the first problem needs to be solved. The solution of the second problem refines the estimate of the bearing angle slightly. However, the last problem has to be solved to obtain the exact bearing angle as well as the range of the shooter.

It will be evident from the following discussion that we need at least two spatially-separated clusters of sensors for solving any and all of the above localization problems involving the ballistic shock wave. A cluster is defined herein as a collection of relatively closely-spaced sensors (microphones, say). For 2-D localization, we need at least three sensors in a cluster; having more (coplanar) sensors in a cluster of course improves the robustness of the estimation. Analogously, for 3-D localization (not pursued here), we need at least four non-coplanar sensors in a cluster. A single cluster can be used independently to record and process sensor data. So, in a decentralized real-time system, all clusters are replicas of each other and, thus, can work in parallel to process the acoustic information. The two (or more) sensor clusters deployed for localization have to be far enough apart to yield distinct information; this will be clearer from the subsequent discussion.

If all sensors in a cluster are on the same side of the bullet trajectory (as shown in fig. 2.19), then the shock wave will appear to arrive from a common direction with the ambient sonic speed. This direction of arrival (DOA) is indicated by the unit sighting vector \hat{s} in the figure (see also Section 2.3.4). Note that, unlike the DOA in the muzzle blast described in Section 2.3.4, the linear wavefront in the definition of the DOA here does not require the sensor array to be far from the bullet trajectory. This is because the linearity of the wavefront is a fact instead of an assumption for the ballistic shock. The algorithm for determining the DOA for a cluster has been described in Section 2.3.4.

If, on the other hand, the bullet trajectory passes through a sensor cluster, then it is meaningless to look for a unique DOA (even if the sensors themselves remain unharmed). A giveaway of this condition is the large deviation of the length of the sighting vector \hat{s} from unity. If there are more than the minimal number of sensors in the cluster (e.g., four in a 2-D problem), then an even better way of identifying this degenerate scenario is the largeness of the ‘residual’ $\|\tilde{A}\hat{s} - \tilde{D}\|$ (see eqns. (2.20) and (2.21)). Thus, if the above residual is larger than a preset threshold for a particular cluster, then that cluster should be disregarded in the application of this algorithm.

There are two scenarios in the problem of determining the direction of the bullet trajectory (i.e., the unit vector \hat{t}) concerning the arrangement of the clusters relative to the bullet trajectory. In the first case, there are sensor clusters on either side of the trajectory (see fig. 2.20 next). The alternative scenario is if all clusters are on the same side of the trajectory. In the latter case, some *a priori* knowledge about the bullet parameters (viz. calibre, length, muzzle speed) is required to estimate the trajectory direction. Here, the focus is only on the former scenario as this problem is of greater relevance (see Section 4.4.1) apart from being easier to solve.

If two clusters happen to be on opposite sides of the bullet trajectory, the unit sighting vector (or DOA) of the shock wave will be different for them. A schematic of this particular case is displayed in fig. 2.20. We might have multiple clusters deployed in a particular setup. From these, we will identify all pairs of clusters (for a particular firing event) whose DOAs are significantly different. Each such pair will yield one estimate of \hat{t} , and we will average the result over all pairs of straddling clusters.

Let us denote the pair of clusters chosen for a particular estimate as C_1 and C_2 . Let their respective unit sighting vectors be \hat{s}_1 and \hat{s}_2 . From the geometry of fig. 2.20, it is clear that each sighting vector makes an angle of $(90^\circ + \theta_M)$ with the (unit) trajectory vector \hat{t} . Applying trigonometric identities, this can be expressed as

$$\hat{t} \cdot \hat{s}_i = \cos(90^\circ + \theta_M) = -\sin \theta_M, \quad \forall i \in \{1, 2\}.$$

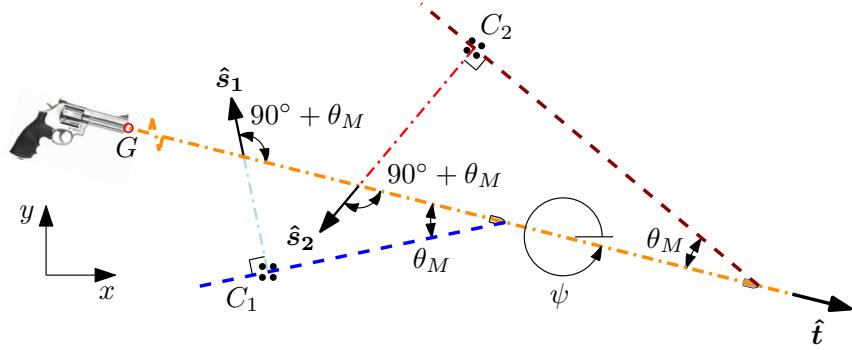


Figure 2.20: Schematic of the case where a bullet trajectory passes in between two clusters of sensors C_1 and C_2 , reporting respective shock wave unit sighting vectors (or DOAs) \hat{s}_1 and \hat{s}_2 .

Subtracting the second equation from the first, and introducing the Cartesian coordinates of the vectors involved as $\hat{\mathbf{t}} = [\hat{t}_x, \hat{t}_y]^T$ and $\hat{\mathbf{s}}_i = [\hat{s}_{i,x}, \hat{s}_{i,y}]^T$, we obtain one equation for the components of $\hat{\mathbf{t}}$. The other equation that has to be satisfied is that $\hat{\mathbf{t}}$ has unit length. That is, \hat{t}_x and \hat{t}_y have to simultaneously satisfy the two equations

$$\begin{aligned} \hat{t}_x(\hat{s}_{1,x} - \hat{s}_{2,x}) + \hat{t}_y(\hat{s}_{1,y} - \hat{s}_{2,y}) &= 0, \\ \hat{t}_x^2 + \hat{t}_y^2 &= 1. \end{aligned}$$

By eliminating one component, say \hat{t}_y , between the two equations, a quadratic equation is obtained for the other component, \hat{t}_x , that will yield two solutions. Each in turn will be associated with its respective solution for \hat{t}_y . Out of these two, the correct solution for the unit trajectory vector is obtained from the conditions

$$\hat{\mathbf{t}} \cdot \hat{\mathbf{s}}_i = -\sin \theta_M < 0, \quad \forall i \in \{1, 2\},$$

since the Mach angle θ_M is acute for a supersonic bullet.

The angle that the trajectory makes with a chosen reference direction (the x -axis shown in fig. 2.20) is of course given by $\psi = \tan^{-1}(\hat{t}_y/\hat{t}_x)$. To avoid ambiguity in the `atan` function, one should always use the standard `atan2` function of the programming language (or encode it in hardware) to calculate

$$\psi = \text{atan2}(\hat{t}_y, \hat{t}_x). \quad (2.24)$$

The trajectory direction is calculated similarly for all the chosen cluster pairs and the average value is computed to get the unit trajectory vector for the event. The Mach angle and the bullet speed is computed from the trajectory direction (and the unit sighting vector for the first cluster, say) using

$$\theta_M = \sin^{-1}(-\hat{\mathbf{t}} \cdot \hat{\mathbf{s}}_1), \quad V_b = c / \sin \theta_M.$$

In more sophisticated localization algorithms (Duckworth et al., 1997), the bullet speed estimated above can be used along with its assumed deceleration rate and muzzle exit velocity to get the actual location of the gun, but this has not been pursued in this work.

Chapter 3

Hardware and Software for Acoustic Gunshot Localization

The development of the hardware and software for the present project has been reported earlier by Kothiwala (2017) and Bhattacharya (2020) as part of their Master's dissertations.

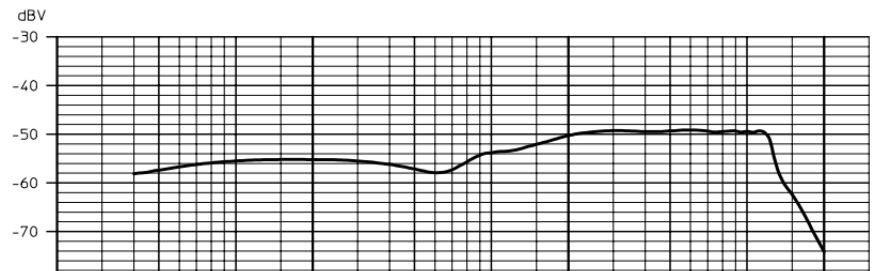
3.1 Sensor basics and arrangement

Sennheiser MD-42 microphones are used for capturing the acoustic signals of gunfires and firecracker bursts. These are omni-directional dynamic microphones with sensitivity of 2 mV/Pa. The microphone and its frequency response are shown in fig. 3.1.

All the localization algorithms discussed in the previous chapter are based on calculating the DTOA of an event signal at known sensor locations. Prior to settling on the final two-dimensional (2-D) localization approach, we evaluated a three-dimensional (3-D) approach. For locating a source in 3-D using DTOA measurement, at least four non-planar microphones are required to



(a)



(b)

Figure 3.1: (a) The Sennheiser MD-43 omni-directional microphone used in the device, and (b) its frequency response.

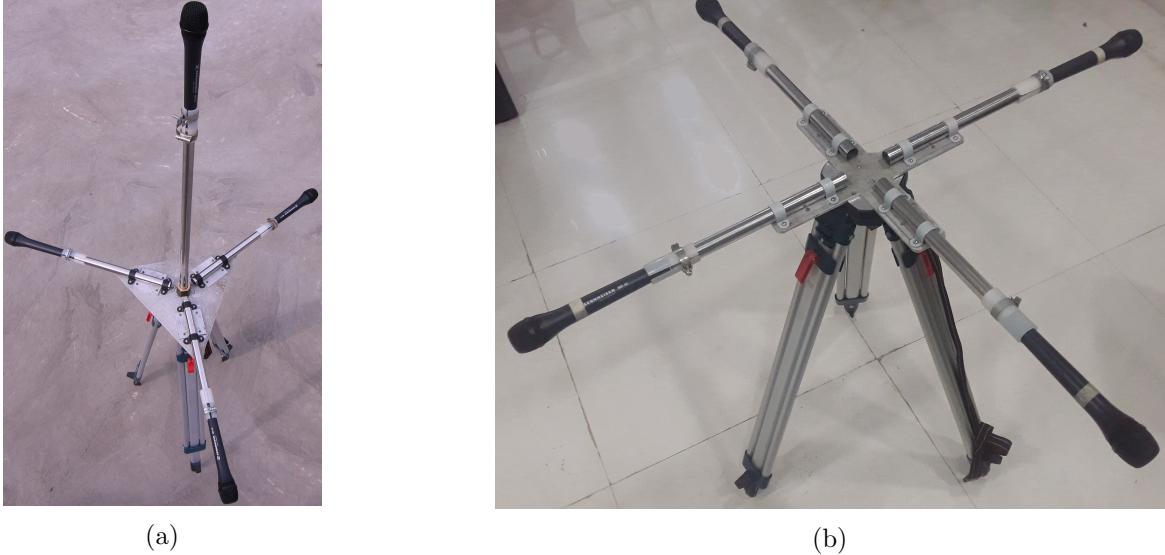


Figure 3.2: (a) Tetrahedral microphone array on a tripod for gunshot localization in 3-D. (b) Planar square microphone array for 2-D approach.

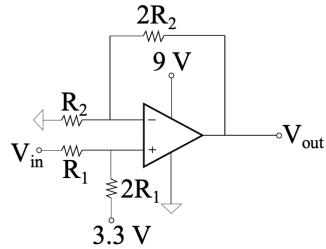


Figure 3.3: Schematic of the biasing circuit connected between the microphone output and the Tiva DAQ board's input to allow the capture of negative signal values.

capture the signals. The inter-sensor distance is maximized by arranging the microphones in a tetrahedron structure. Moreover, placing three of them in the horizontal plane improves the in-plane bearing estimate, which is the most crucial piece of information desired. The microphone array designed per these principles is shown in fig. 3.2a. Every pair of sensor has 1 meter separation. The microphones are connected using XLR connector through microphone cables to stereo jacks attached to the pins of data acquisition device. Using the tetrahedron setup, we can compute three unknown parameters, of which the most important ones are the range and the bearing angle of the source. The third parameter is the zenith angle or altitude, which is not very important in the present scenario. To get a more robust estimate of the range and bearing angle of the source, a new rig was developed with the same four microphones arranged in a planar square array (see fig. 3.2b). The separation between any pair of microphones is 1 meter again.

Both the muzzle blast and shockwave signals consist of both positive and negative voltage values, as shown in fig. 2.9a. However, the Tiva data acquisition (DAQ) board (see Section 3.2.2) used to record the microphone signals limits the input voltage to the range of 0 to 3.3 V. Thus, an additional piece of analog circuitry is connected between each microphone's output and the corresponding input channel of the Tiva DAQ to allow recording of the negative parts of the microphone signals. The circuit shown in fig. 3.3 shifts the range of the input signal that can be acquired by the Tiva DAQ board from 0 to 3.3 V to the range of -1.65 to $+1.65$ V for a single



Figure 3.4: MCC USB-1608FS DAQ board powered from a laptop.

channel. The actual circuit is made on a general purpose PCB which has a quad LM324 op-amp and the required resistors. The supply voltage for the op-amp is given from a 9 V battery. The input 3.3 V power shown in the figure is supplied by the Tiva board's output pin.

3.2 Data acquisition and processing

3.2.1 MCC USB-1608FS DAQ board for data acquisition

In the initial stages of the development, the four microphone signals were acquired for offline processing on a 16-bit MCC (Measurement Computing Company) USB-1608FS DAQ board with a maximum simultaneous sampling frequency of **25 kHz**. This board (shown in fig. 3.4) had to be connected to a computer using a USB cable both for power supply and for sending data. An off-the-shelf software – MCC's DAQami – provided a GUI for the acquired signals to be monitored. The signal from all the channels could be analyzed and the relevant event data could be saved as files on the computer. The data thus logged in files was post processed in a personal computer to localize the origin of a gunshot-like event – this is referred to as the '**offline**' implementation.

The main purpose of the project is to make the localization process real-time, something that cannot be achieved using the MCC DAQ board. Also, wired connections to personal computers are not feasible for deployment in the field. Another drawback of using this board is that the maximum sampling frequency that can be achieved for four channels is 25 kHz. The precision of the DTOA calculation (the sole input to the localization algorithm) is improved by increasing the sampling frequency of the data acquisition.

3.2.2 Tiva board for data acquisition

To implement the real-time localization algorithm in a standalone ('online') manner, a Texas Instruments **Tiva C Series TM4C1294** Connected LaunchPad Evaluation Board (Tiv) is used to acquire the microphone signals. It has an ARM-based microcontroller with two 12-bit ADC modules and Ethernet connectivity. The board has an in-circuit debug interface (ICDI) with USB connection. The debug USB is connected to a computer to develop the software which is to be run on the chip. Power is also provided to the board from the computer through this USB. Once the binary code is flashed on the chip, power can be provided from any external 5 V DC source through VCC pins on the board making the device standalone.

The software is developed for the board on an environment called Code Composer Studio (CCS).

The program is written in Embedded C language with the help of Tivaware software tool. The software implemented on the Tiva board has the following aims:

- Continuously ‘listen’ on all microphone channels (currently four in number),
- Detect the acoustic event on any microphone channel,
- Acquire data in a brief time window covering the acoustic event on all four microphones, once an ‘interesting’ event is detected, and
- Pass on the collected data over Ethernet to a downstream microprocessor (the RPi board described in Section 3.2.3).

The on-board ADC is triggered by a hardware timer. The timer is set with a period to interrupt the processor and trigger the ADC. This period then becomes the sampling period for acquiring analog data from the microphone. The timer is made to run periodically as it reloads each time after going to zero.

In our first implementation, an interrupt service routine (ISR) was written to handle the interrupts from the timer. As the ADC was triggered at the end of each timer cycle, a data sample from a microphone was written into the ADC register. This was done for all four channels connected to their corresponding microphones. The ISR wrote the sampled data from the ADC registers into 2-dimensional primary circular arrays for all four channels for further processing, and sent it out on the Ethernet socket. This implementation suffered from some glitches at high sampling rate as it was not very efficient. This was due to the fact that the timer was responsible for both the triggering of the ADC and the interrupting of the processor; this resulted in the processor remaining idle for some time. Indeed, the highest sampling rate achieved was about 40 kHz.

Our subsequent implementation has improved upon this; so that we can sample at 80 kHz at present. In this version of the software, the timer triggers the ADC, and the ADC interrupts the processor once it has acquired the data. The ISR responds to the ADC interrupt and copies the data from the ADC registers to their respective primary buffers, as discussed below. The schematic of the present algorithm running on the Tiva board is shown in fig. 3.5. As seen from the schematic, the ISR (handler) is an independent process and runs when the ADC raises the interrupt flag. It has the highest priority amongst all the processes running on the board.

Along with the sampling of the microphone signal at periodic intervals, a main program runs in a loop which checks for the event in the primary buffers of all channels. An event is deemed to be detected if the signal value crosses a preset threshold. The threshold has to be chosen carefully such that no events are missed, and also that there are no false positives. Some initial trials were done for both firecrackers and gunshots to set the threshold used in further testing. Furthermore, microphone signals can drift over time. Thus, one first needs to determine a baseline (a DC value, so to say) from which the threshold value should be measured. For this purpose, the event checking algorithm starts only after 6.25 ms has elapsed since the booting of the board (which corresponds to 500 samples for 80 kHz sampling rate). The average of the acquired voltage values for this initial 6.25 ms is calculated for each channel to get their respective DC voltages. The absolute-value threshold is set around this DC voltage to detect the event.

The typical muzzle blast and shock wave components of a gunshot’s acoustic signature (as depicted in fig. 2.9a) have very short rise times, and much longer decay times. However, if we start recording the microphone signals only after they cross the set threshold, then we stand to lose the initial rising portion. Instead, we would like to record a small segment of data *before* the event sample (the sample at which the ADC value first crosses the threshold) and a larger segment of data after the event sample. This is facilitated in our implementation by storing the

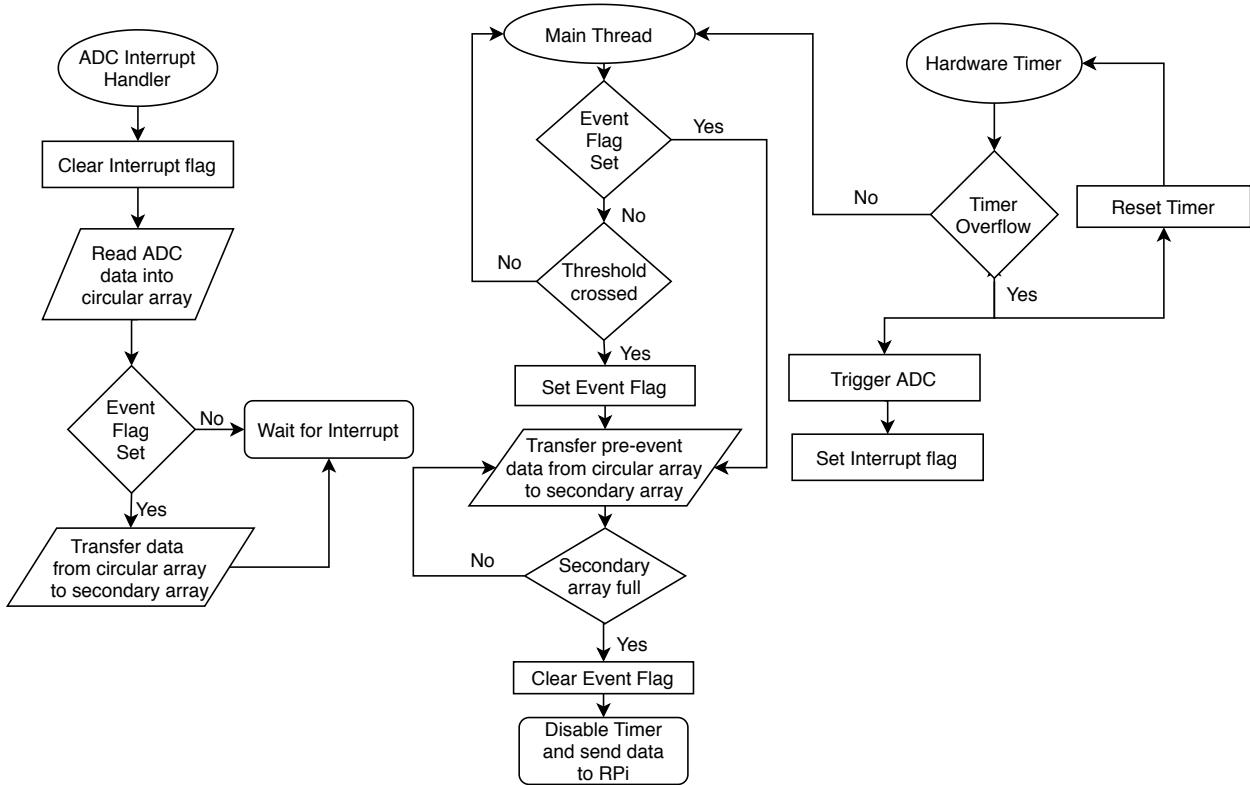


Figure 3.5: Flowchart of the algorithm running on the Tiva board.

'heard' signal on each channel in a *circular* primary buffer. Looking at the gunshot signatures, we conservatively estimated the length of the **pre-event segment** to be **0.3 ms**. With sampling at 80 kHz, this corresponds to 24 samples of each of the four microphone signals that we need to retain in memory at all times. The actual circular buffer is oversized by a factor of four to contain 100 samples; this allows for future improvements in sampling rate.

If and when an event is detected on any microphone channel, we pause checking for events on all channels, and instead transfer the sampled data directly into a secondary (linear) buffer, starting from the 25th sample. Simultaneously, we also populate the first 24 entries of the secondary buffer with as many of the recentmost entries of the circular primary buffer. Note that this is done for all four channels, even though the event is detected on only one channel.

The secondary buffer is sized so as to contain the maximum duration of gunshot acoustic events. A study of many such event signatures yielded a conservative estimate of **4 ms** for this duration. Additionally, we also have to allow for propagative time delay of the acoustic wave from the first microphone it encounters (which triggers the detector, and subsequent filling of the secondary buffer) to the last one. For our meter-square microphone array, this delay is at most about $(\sqrt{2}/c) \approx$ **4.2 ms**. Here, $\sqrt{2}$ m is the length of the diagonal of the planar array, and $c \approx 340$ m/s is the typical speed of sound. Adding these two time durations with the **0.3 ms pre-event** segment yields a total of around 8.5 ms. Rounding this up to 8.75 ms, we arrive at the secondary array length of 700. The secondary buffer is thus a 2-D array with **700 columns and 4 rows** (one for each channel).

The two-dimensional secondary array containing the event data of all four channels is sent to the Raspberry Pi (RPi) microprocessor (to be discussed next) through Ethernet for further processing towards the final source localization. The timer interrupt is disabled during the data exchange



Figure 3.6: The Tiva board (red) and RPi card (green) connected by an Ethernet cable.

through Ethernet. All these processes running simultaneously limits the maximum sampling rate that can be achieved for the acquisition to the present value of 80 kHz.

The Ethernet data transfer is done using an open source lightweight TCP/IP stack known as **lwIP on the Tiva board side**. On the **RPi side**, the data transfer is done using **Unix socket**. The connection between the boards is achieved through **IPv4** internet layer protocol (IPv). The transport protocol used for the exchange is Transmission Control Protocol (TCP). Raw lwIP APIs are used for sending the data from the Tiva board, and Berkeley socket APIs are used to receive the data on the RPi. After the secondary array is filled with 8.75 ms of an event's data, two data packets containing the channel number and the channel data respectively is sent consecutively to the RPi for the first channel. The same sequence is repeated for the next three channels. An image of the two boards connected by an Ethernet cable is shown in fig. 3.6.

3.2.3 RPi board for gunshot localization

The actual acoustic localization algorithm requires computational power that the Tiva board cannot deliver. Thus, we implement this on a Raspberry Pi Compute Module 3+ (RPi) (RPi), which has a better processing capability. The Tiva board provides a good interface between the analog and the digital domain, whereas RPi does the digital processing of the acoustic signals. The RPi is currently powered by using a micro USB port from a personal computer, but it can also be powered using GPIO pins from a battery to render it standalone. To transfer the program to the RPi, we interface it with a laptop using Secure Shell (SSH) networking over Ethernet or WiFi module. RPi does not have ROM or flash storage on the board, but provides a microSD card slot. The SD storage is loaded with a Linux based operating system called Raspbian and the programming is done in Python language.

The event signal from the four channels of the Tiva board received by the RPi is provided to the cross-correlation function for computing the DTOA of the acoustic signal between all microphone pairs. The data for all channels are for the same time window. So, the cross-correlation operation discussed in Section 2.2.4 will directly yield the time delay between the arrival of the acoustic event at the two microphones. Before the calculation of the DTOA, the ground reflection removal algorithm as discussed by Braasch (2016) is applied to the signals for each channel. The event categorization algorithm as discussed in Section 2.2.3 is applied to the event data for each channel. If the event is categorized as a muzzle blast, the DTOA is calculated for the data windows. The

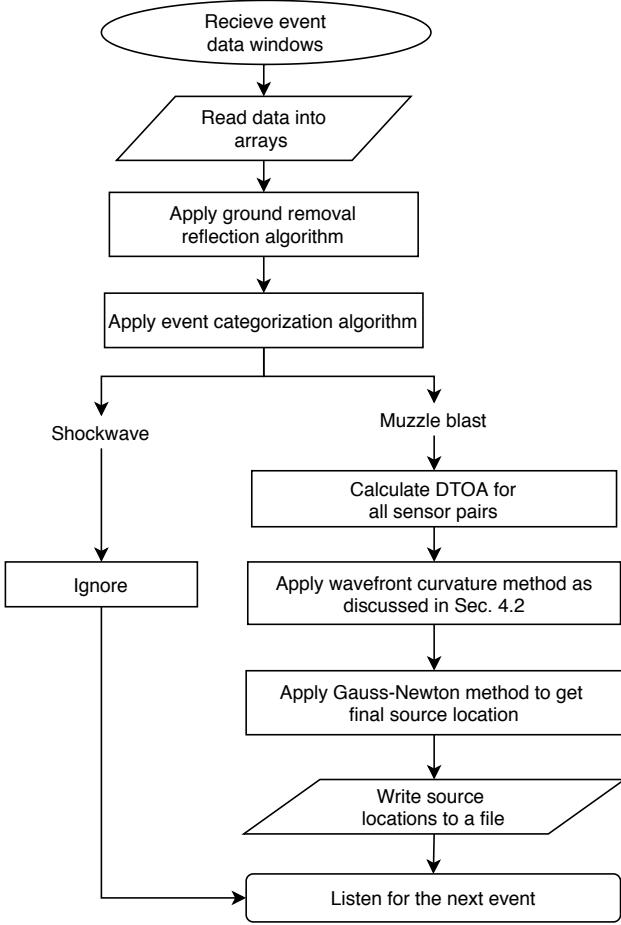


Figure 3.7: Flowchart of the algorithm running on the RPi board.

DTOA values are then fed as input along with the sensor locations to the localization algorithm.

The localization methods give the source locations as output from which the bearing and the range of the source is estimated. The criteria to pick one solution between multiple solutions for each algorithm and the choice for the algorithm is explained in Chapter 2. The present implementation of the localization method involves getting the 2-D coordinates of the source from wavefront curvature method and providing the solution as initial guess for the classical non linear least square estimation method to get the final solution. Once the solution for a particular event is estimated, it is written to a file and the software goes back to listen for further events from the Tiva board. A flowchart of the algorithm running on the RPi board is shown in fig. 3.7. At present, the source location is read off the screen of a laptop connected to the RPi board, but an LCD screen may be connected to the RPi board in the future.

The RPi board requires some time to run the localization algorithm. We avoid sending fresh acoustic event data from the Tiva board during this period, so as not to overwhelm the RPi program. Thus, we pause the ‘listening’ for acoustic events (i.e., the detection and all downstream activities) on the Tiva board for a total of 100 ms (0.1 s) after an event is detected. Note that the first 8.55 ms of this time window goes in recording the corresponding duration of acoustic signal. This sets the current minimum time between two gunshot events that can be unambiguously localized by our device to 0.1 s. Also, our implementation requires this 0.1 s to report the location estimate from the time of gunfire (actually from the time its sound reaches the sensor array).

Chapter 4

Results and Discussions

The gunshot localization technology based on the muzzle blast that has been developed as part of this project has been tested under various circumstances; here we describe these results. As mentioned earlier, firecrackers are a useful stand-in for gun muzzle blast, and thus they have been employed extensively in our validation assessments. In fact, Sections 4.1 and 4.2 describe results obtained with the earlier offline and the later online implementations of the technology, respectively, applied to localizing firecrackers when they burst. Subsequently, Section 4.3 presents validation results of the offline implementation for gunshot localization in a firing range. Finally, Section 4.4 describes the validation of the bullet trajectory estimation algorithm based on the ballistic shock wave, in computer simulations. The work presented here has been reported by Bhattacharya (2020) as part of his Master's dissertation.

4.1 Validation of offline localization based on blast wave using firecrackers and the tetrahedral microphone array

This experiment involved localization of an acoustic source produced by bursting a firecracker. The tetrahedral microphone array shown in fig. 3.2a was used, and the MCC DAQ board discussed in Section 3.2.1 was used for data acquisition. The Bancroft method for blast wave-based source localization discussed in Section 2.3.3 was applied in offline mode on a computer to get the firecracker burst location. The reflection removal algorithm was employed as described in Section 2.2.2, before calculating the DTOA using the cross-correlation technique as explained in Section 2.2.4. The sampling rate for all the channels was taken to be **25 kHz** (the maximum possible on the MCC DAQ when acquiring data on four channels). The microphone array was kept fixed at a known location and firecrackers were burst at different locations. A schematic showing the position of the crackers and the microphone location from the top is presented in fig. 4.1. The centroid of the base triangle of the tetrahedral array forms the origin of the coordinate axes, and the microphone M_1 lies on the x -axis w.r.t. which the bearing angle is measured.

Although the microphone coordinates are known and are provided to the localization methods as input, but the exact location of the diaphragm which captures the signal is ambiguous. Also, the microphone array, although built to be symmetrical will not be exactly so due to inevitable variations in fabrication. From an analysis of the localization results it is found that a small change in inter-microphone distance could result in high errors in the source location. To overcome this issue, a calibration of the microphone positions is done by implementing the nonlinear least-squares method as discussed in Section 2.3.2, but in reverse. This gives the microphone locations from an overdetermined set of known source locations. A comparison of the range and bearing errors for

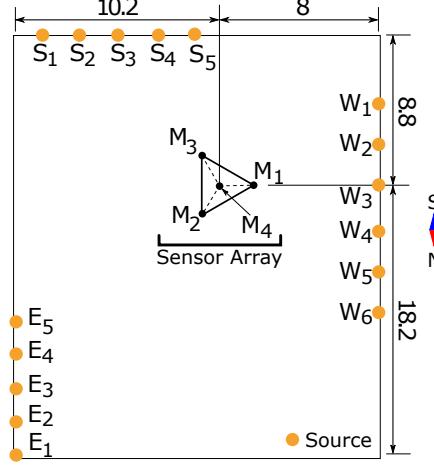


Figure 4.1: Top view schematic of the firecracker experiment set up to validate the offline localization method using blast wave. Locations of the four microphones and cracker bursts are shown (all dimensions are in meters). The tetrahedral sensor array is not shown to scale; each of its side is actually 1 m in length.

Sl. no.	Bearing error (deg)		Normalized range error	
	Without calibration	With calibration	Without calibration	With calibration
1	1.7	0.9	0.7	0.1
2	1.9	0.9	0.8	0.3
3	178.7	0.6	0.1	0.3

Table 4.1: Range and bearing errors of firecracker burst localization in offline mode, without and with calibration of the tetrahedral microphone array.

three source locations before and after calibration is shown in table 4.1. Here, and later, the normalized range error refers to the error in range estimation as a ratio of the actual range.

For calculating the DTOA of the event at a microphone, a reference microphone is selected relative to which the time delays are computed. Kothiwala (2017) carried out extensive computer simulations to understand the performance of localization algorithms using a planar equilateral triangular array of sensors. He found that the estimated source location changes with the choice of the reference microphone, and proposed the following algorithm for choosing the same. For any reference sensor, the remaining sensors yield corresponding normalized DTOA values (η 's), as defined in eqn. (2.10), from which we identify the maximum value, say η_{\max} . Similarly, we calculate the η_{\max} values for all the other choices of the reference microphone. Kothiwala (2017) showed that the best choice of the reference for accuracy of the location estimate is the one for which η_{\max} is minimum. This was also found to be true in the experiments reported here, and has been applied throughout.

The Bancroft method applied here gives two solutions for the source location and the time for the signal to reach the reference microphone. The solution having the least residual is chosen as the valid solution as discussed in Section 2.3.3. This algorithm provides a 3-D acoustic localization of the source but the zenith angle estimate is ignored as it is of little relevance for the purpose of this project, as argued in Chapter 2.

Table 4.2 shows the actual bearing and range of all the sources, their offline estimates, and the

Source ID	Bearing angle (deg)			Range			
	Actual	Estimated	Error	Actual (m)	Estimated (m)	Normalized	Error
W_1	35.7	34.7	1.0	10.54	9.42	0.11	
W_2	19.6	19.4	0.2	9.09	11.57	0.27	
W_3	0.0	-0.2	0.2	8.56	5.19	0.39	
W_4	-19.6	-19.8	0.2	9.09	14.06	0.55	
W_5	-35.5	-33.6	1.9	10.51	4.78	0.55	
W_6	-46.7	-46.5	0.2	12.49	13.67	0.09	
S_2	124.4	125.4	1.0	10.77	13.69	0.27	
S_3	119.1	121.0	1.9	10.17	32.78	2.22	
S_4	105.4	107.1	1.7	9.21	15.66	0.70	
S_5	97.9	100.7	2.8	8.96	28.42	2.17	
E_1	-118.2	-118.8	0.6	20.64	27.62	0.34	
E_2	-122.4	-122.0	0.4	18.22	13.34	0.27	
E_3	-128.5	-128.6	0.1	15.69	13.52	0.14	
E_4	-136.6	-136.8	0.2	13.43	8.07	0.40	

Table 4.2: Performance of offline localization in the firecracker experiment.

bearing and range errors. The error in the bearing angle is fairly low, with the maximum error being less than 3°. The normalized range error is largely within 50%, but in some of the cases it reaches up to 200%. Figure 4.2 provides a visualization of the tabulated data. It reinforces the conclusion that the localization performance is excellent vis-à-vis the direction, but is much poorer in terms of the distance.

4.2 Validation of online localization based on blast wave using firecrackers and the planar square microphone array

We now report on the validation of the improved and online implementation of localization technology, again using firecrackers. The changes have been already described in Chapter 3; these are now discussed in the context of the shortcomings exposed by the experiments reported in Section 4.1.

From extensive analysis of the performance of localization algorithm in simulations, Kothiwala (2017) reported that the error in range estimation increases monotonically with the error in measuring the DTOA. In fact, the maximum error in the DTOA calculation will be equal to half of the sampling period of the ADC (Analog to Digital Converter) used. Also, it has been found from the experiments that a slight error in the DTOA leads to a large error in the range estimation. So, we concluded from the preliminary assay reported in Section 4.1 that we needed to increase the sampling rate of the data acquisition system significantly in order to reduce the range error.

The localization accuracy may also be expected to improve if we have more sensors in a cluster than the strict minimum. Given that the zenith angle (or relative altitude) of the shooter is of minimal interest, one may pursue the gunshot localization in 2-D instead of 3-D. This reduces the minimum number of required sensors from four to three. Then, robustness of the location estimate may be improved if the microphone array consists of more than three sensors in a horizontal plane.

Motivated by the above considerations, we have implemented the following improvements in our blast wave-based localization technology. The square planar array shown in fig. 3.2b is now used for capturing the acoustic signals; it has four co-planar microphones, which is more than the minimum

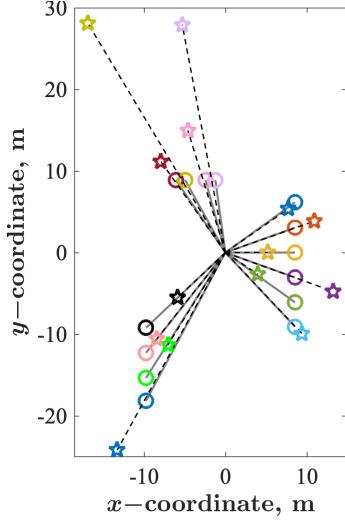


Figure 4.2: Comparison between the actual (circle marker) and offline-estimated (pentagram marker) source locations in the firecracker experiment. Pairs of actual and estimated locations are depicted with the same color of marker.

required for 2-D localization. For the data acquisition and preliminary analysis, the Tiva board discussed in Section 3.2.2 is used with a sampling rate of 80 kHz per channel (compare this with the 25 kHz that could be achieved with MCC DAQ earlier). Finally, the remaining pre-processing as well as the actual localization algorithm are now implemented on the RPi board introduced in Section 3.2.3, which affords real-time online estimation of the source location.

As discussed in Section 2.3.3, Bancroft’s method performs poorly for over-determined systems (i.e., when there are more sensors than the minimum required). So, a combination of the wavefront curvature method (see Section 2.3.1) and the nonlinear least squares approach (see Section 2.3.2) is implemented in the online localization system. In particular, we use the former algorithm to obtain an initial guess of the source location for use in the latter method.

The wavefront curvature method involves exactly three microphones to get the source location in a plane. This introduces the task of first choosing the ‘best’ triplet from the four possible triplets in the four-microphone square array. The steps involved in estimating the source location is listed below; it is implemented on the RPi board (see Section 3.2.3) for online calculation.

1. For each available microphone triplet, identify the reference using the procedure described in Section 4.1, before applying the wavefront curvature method to get a preliminary estimate of the source location.
2. Amongst the four solutions from the four triplets of sensors, choose the source location estimate from the triplet that yields the minimum error ϵ_{wc} as defined in eqn. (2.13).
3. Use this source location estimate as the initial guess in the nonlinear least-squares algorithm applied to the information available from all sensors together. For this, the reference microphone is identified again as done for the wavefront curvature method, but amongst all available microphones. The final source location is obtained from the nonlinear least squares estimation terminating for a given error tolerance.

The validation experiment was performed by bursting firecrackers, similar to the previous experiment described in Section 4.1. The planar microphone rig was kept fixed at a known location,

Source ID	Bearing angle (deg)			Range		
	Actual	Estimated	Error	Actual (m)	Estimated (m)	Normalized Error
P_1		-66.2	0.6		6.89	0.19
	-65.6	-65.8	0.2	8.46	6.51	0.23
	-65.8	0.2			7.63	0.10
P_2		-34.4	1.7		8.55	0.40
	-32.7	-34.2	1.5	14.26	10.63	0.25
	-34.7	2.0			11.56	0.19
P_3	9.3	8.0	1.3	9.32	9.03	0.03

Table 4.3: Performance of online estimation of source locations in the firecracker experiment. N.B.: three crackers were burst separately at both P_1 and P_2 .

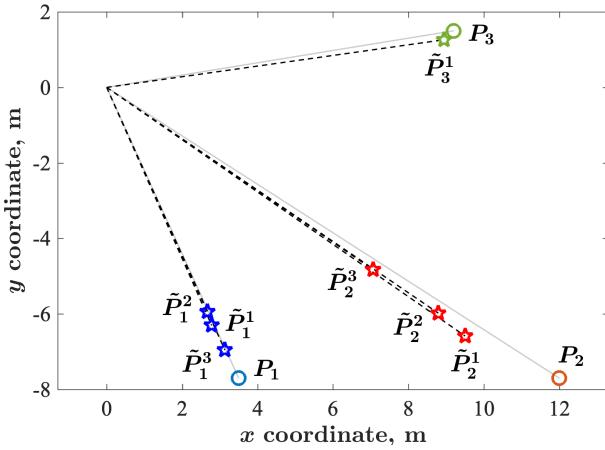


Figure 4.3: Comparison between the actual (circle marker) and online-estimated (pentagram marker) source locations in the firecracker experiment. Pairs of actual and estimated locations are depicted with the same color of marker. N.B.: three independent tests were conducted for crackers located at P_1 and P_2 ; their respective estimates are denoted by $\{\tilde{P}_1^k\}_{k=1}^3$ and $\{\tilde{P}_2^k\}_{k=1}^3$. Only one test could be conducted for the location P_3 .

while the crackers were burst at different locations. The performance of the online implementation of the gunshot localization technology is presented in table 4.3. Three different locations are chosen for bursting the firecrackers, and multiple iterations are conducted for some of these locations. The maximum error in the bearing angle is only 2° . The maximum normalized range error is 40%, which is an improvement over the results reported in Section 4.1. A visualization of the performance is also presented in fig. 4.3. It reinforces the drastic improvement achieved in comparison with the earlier offline implementation, shown in fig. 4.2

To showcase the improvement in the range estimation, table 4.4 compares the mean, median and maximum of the normalized range errors for the current experiment to that obtained earlier in Section 4.1. Increasing the sampling rate for acquisition from 25 kHz to 80 kHz and making the system over-determined are the principal reasons for the improvement in the estimated source range.

Parameter	Normalized Range Error	
	Improved Online Implementation	First Offline Implementation
Maximum	0.40	2.22
Median	0.21	0.36
Mean	0.21	0.60

Table 4.4: Reduction in range estimation error achieved by the improved online implementation reported here, vis-à-vis the first offline version discussed in Section 4.1.

Source ID	Bearing angle (deg)		Range	
	Actual	Error	Actual (m)	Normalized Error
Q_1	97.4	1.5	5.53	0.28
Q_2	96.1	0.9	9.80	0.07
Q_3	96.8	0.4	14.15	0.30

Table 4.5: Performance of offline estimation of source locations in the gunfire experiment.

4.3 Offline localization of gunfire

A validation experiment was conducted in a firing range with a Glock pistol being fired from three different locations. The earlier offline version of the source localization technology described in Section 4.1 was deployed. In particular, the microphone signals were sampled at 25 kHz on the MCC DAQ board, the tetrahedral microphone array was employed, and Bancroft’s algorithm was used for the localization using the muzzle blast wave.

The performance is presented in table 4.5. In particular, we note that the bearing angle error is limited to 1.5° only, whereas the normalized range error is at most 0.28. It must be admitted that the actual range of the gunfire is quite small in these preliminary experiments. More issues may be anticipated in cases where the gun is fired farther from the microphone array.

4.4 Bullet trajectory direction estimation in simulations

Till now, this chapter has described the performance of the blast wave-based source localization technology implemented in hardware – either in offline or online mode. If the bullet travels supersonically, then it trails a shock wave that can be used for additional refinement of the estimate of the shooter’s position. In fact, if the gun is very far away from the microphones, then the muzzle blast wave may be too attenuated to be reliable, and the ballistic shock may be the only acoustic signature available for localization. Thus, gunshot localization technology must also utilize the bullet shock wave, if any.

We have described a method for estimating the bullet trajectory direction in Section 2.4.2. As mentioned in Chapter 1, we have not implemented this in the hardware as yet. Instead, it has been implemented in software and validated in computer simulations. For this, we first need to simulate a bullet shock wave’s signature at a virtual microphone, which is described in Section 4.4.1. Next, Section 4.4.2 presents an implementation (in simulation) of the bullet trajectory direction estimation algorithm exemplified by a gunshot localization system deployed to secure a physical area of interest (say, a compound). Finally, Section 4.4.3 reports on an assessment of the robustness of the above algorithm in the presence of simulated noise in the microphone signals.

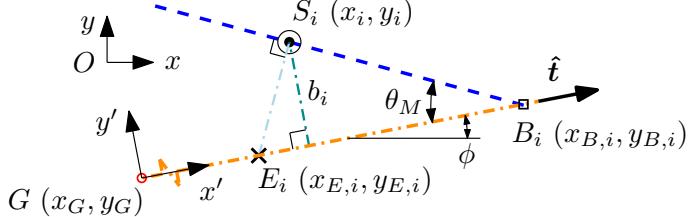


Figure 4.4: Calculations needed to simulate arrival time of shock wave trailed by supersonically moving bullet at a sensor.

It will be recalled that the algorithm discussed in Section 2.4.2 and evaluated here is the simplest one possible that uses the ballistic shock signature. It is appropriate if the shooter is located far away from the sensors (i.e., from the compound in need of security), and if the bullet trajectory passes in between one or more pairs of the sensor clusters deployed. Moreover, we assume that the bullet maintains a constant speed during its passage in the vicinity of the sensors. Under these conditions, the algorithm estimates the unit vector along the bullet trajectory, which gives the approximate bearing angle of the distant shooter.

4.4.1 Simulation of ballistic shock wave recorded at a virtual microphone

Here, we describe how the ballistic shock wave arriving at a particular (virtual) microphone is simulated for the purpose of validation of the ballistic trajectory direction estimation algorithm.

The geometry of the problem is redrawn from fig. 2.19 in fig. 4.4, focusing on some relevant details. In particular, let the origin of the $x - y$ coordinate system be at O . Let the gun be fired at the point G with coordinates (x_G, y_G) , and let the bullet trajectory make an angle ϕ with the x -axis. The bullet Mach number is M . We are interested in simulating the shock wave recorded at the i th sensor (i.e., at S_i) with coordinates (x_i, y_i) . This requires knowledge of two things – the time of arrival (TOA) of the shock wave (for use in eqn. (2.22)) and the miss distance (for determining the ‘N’ wave from eqn. (2.23)). In turn, for determining the TOA, we need to locate the emission point E_i with coordinates $(x_{E,i}, y_{E,i})$.

The problem is conveniently solved by transforming to the $x' - y'$ coordinate system, whose origin is at the gun location and whose x' -axis is along the bullet trajectory vector \hat{t} . In this new coordinate system, the coordinates of the i th sensor are

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = R_\phi \left(\begin{bmatrix} x_i \\ y_i \end{bmatrix} - \begin{bmatrix} x_G \\ y_G \end{bmatrix} \right), \quad R_\phi = \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix},$$

where R_ϕ is the rotation matrix for this transformation. The miss distance is simply $b_i = |y'_i|$. The coordinates of the emission point for the i th sensor E_i in the new system are $(x'_{E,i}, y'_{E,i}) = (x'_i - b_i \tan \theta_M, 0)$. We note that for E_i to be a bona fide emission point, $x'_{E,i}$ must be positive; if it comes out as negative (i.e., behind the gun) then that means that the ballistic shock wave never reaches the particular sensor. Finally, the emission point coordinates in the original system are obtained by the inverse transformation

$$\begin{bmatrix} x_{E,i} \\ y_{E,i} \end{bmatrix} = R_\phi^T \begin{bmatrix} x'_{E,i} \\ y'_{E,i} \end{bmatrix} + \begin{bmatrix} x_G \\ y_G \end{bmatrix},$$

where R_ϕ^T denotes the transpose of R_ϕ , which is also its inverse (since it is unitary).

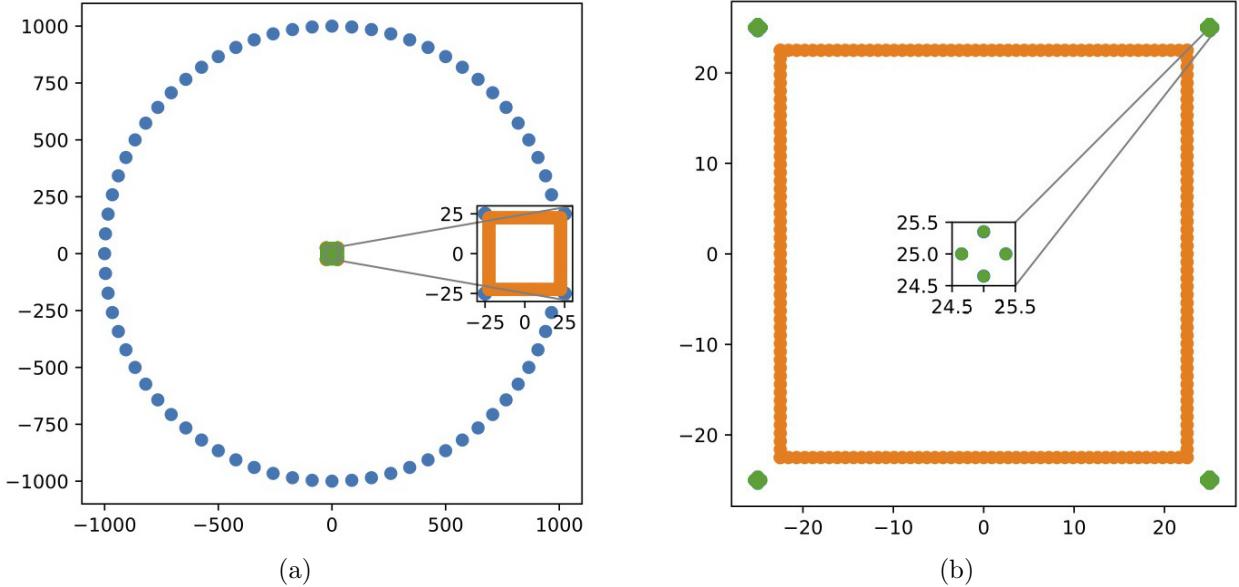


Figure 4.5: (a) Simulation arrangement for sniper firing on a square compound, showing the sources and sensor cluster locations – dimensions are in meter. Blue dots represent choices of the sniper positions all around the compound; green dots denote the sensors deployed. (b) Zoom in view of the target compound with clusters of four sensors each located at the four corners. The orange square demarcates an inner area within which the sniper’s target is located.

For completeness, we can calculate the bullet position B_i when the shock wave reaches the i th sensor. Its coordinates in the new system are $(x'_{B,i}, y'_{B,i}) = (x'_i + b_i/\tan\theta_M, 0)$. Transformation back to the original coordinate system follows the recipe for the emission point described above.

The bullet travels with speed $V_b = Mc$ from the gun at G to the emission point E_i , i.e., the over the distance $x'_{E,i}$ found above. Thereafter, the shock wave moves at sonic speed to the sensor S_i , i.e., over the distance $b_i/\cos\theta_M$. Thus, given the time of gun fire, t^G , say, we can find the time of arrival of the ballistic shock wave at the sensor S_i as

$$T_{0,i} = t^G + c^{-1}(x'_{E,i}/M + b_i/\cos\theta_M). \quad (4.1)$$

Finally, the shape of the ‘N’ wave recorded at the sensor S_i is determined by the over-pressure $\Delta p_{sw,i}$ and the duration $T_{d,i}$, both of which are related to the miss distance (and the bullet caliber and length) through eqn. (2.23).

Of course, the shock wave signals recorded in gunfire experiments shown in fig. 2.9a and 2.12b are not clean; they may be said to have noise superimposed on the basic ‘N’ shape. Moreover, the finiteness of the sampling rate of the data acquisition is also important in determining the actual shape of the signal. These are accounted for in the current simulations.

4.4.2 Simulation of an installation security system that estimates the bearing of a far shooter from ballistic shock

Imagine a 50 meter square area, perhaps a compound, to be secured using the gunshot localization system, as shown fig. 4.5a. Four sensor clusters are deployed, one at each of the four corners of the square area, as shown in the zoomed view in fig. 4.5b. Each cluster consists of four microphones that form a co-planar square of sides 70 cm. (Note that this is a slight reduction in footprint from

Name	Muzzle speed (m/s)	Calibre (mm)	Length (mm)	Weight (g)
5.6×50 Magnum	1050	5.56	50.0	3.24
.30-06 Springfield	750	7.62	63.0	14.3

Table 4.6: Bullet details used in the simulations of shock wave-based localization.

our current implementation, where sensors are separated by 1 m.) Now consider a hostile sniper located anywhere on a circle of radius 1 km centered on the compound. The target of the sniper is some point inside the inner orange square within the compound. This inner target zone ensures that the bullet trajectory passes between at least a pair of clusters, as required by the localization algorithm described in Section 2.4.2. Bullet trajectories crossing overhead of a compound to be secured are of much greater concern than those that pass outside of it, making the above-described problem worthy of investigation.

The simulation of a gunfire event starts by selecting the sniper location and the target at random from the 1 km-radius circle and the compound’s internal (orange) square, respectively. This defines the (random) bullet trajectory direction \hat{t} , and also anchors it in the (horizontal) plane. Two bullet types are used for the simulations, with details provided in table 4.6; we choose one for the simulation. The ballistic shock arrival time (assuming that the gunfire was at time $t = 0$) is then calculated for each of the 16 sensors, as described in Section 4.4.1; for this we assume that there is no deceleration of the bullet. The bullet miss distance and its specifications are then used to determine the over-pressure and duration of the ‘N’-shaped shock wave at each sensor, per Whitham’s relations of eqn. (2.23). These ‘N’-waves are sampled at 200 kHz (which is 2.5 times that of the current implementation), and only these sampled data are supplied to the localization algorithm presented in Section 2.4.2. No additional noise is superimposed on the virtual microphone signals in the numerical experiments described here; the effect of noise is investigated subsequently in Section 4.4.3.

The algorithms for event detection, signal segment abstraction and DTOA calculation described in Chapters 2 and 3 are applied to the sampled signals simulated for each of the four sensors in any cluster. Note that any sensor can be chosen as the reference for the DTOA since we are dealing with a plane wave here. This is unlike the case of DTOA calculation for a blast wave, for which additional care had to be exercised in choosing the reference sensor as described in Section 4.1. The DTOAs, along with the known coordinates of the sensors, are then used to estimate the unit sighting vector of the shock wave for the cluster, as detailed in Section 2.3.4. The four sensors present in the planar cluster makes for an overdetermined system of equations governing the unit sighting vector; this redundancy is expected to improve the accuracy of the estimation.

The unit sighting vectors estimated above for the four clusters form the sole input to the bullet trajectory direction estimation algorithm described in Section 2.4.2. We first need to identify a pair of clusters that are on either side of the trajectory. This is deemed to be the case if the dot product of the unit sighting vectors of the pair is less than 0.866; physically, this stipulates that the shock wave arrival angle at the two clusters must differ by more than 30° . This criterion will fail only if the bullet Mach number falls below $1.035 (= 1 / \sin(90^\circ - 30^\circ / 2))$, which is barely supersonic anyway. From the design of the simulation experiment, one can conclude that there can be a minimum of 3 such pairs of straddling clusters and a maximum of 4 for any bullet trajectory. The trajectory directions estimated from all these cluster pairs are averaged to arrive at the final estimate of the same. Finally, the trajectory angle ψ can be determined (with respect to one side of the square compound identified as the x -axis) from the above-estimated unit trajectory vector \hat{t} using eqn. (2.24).

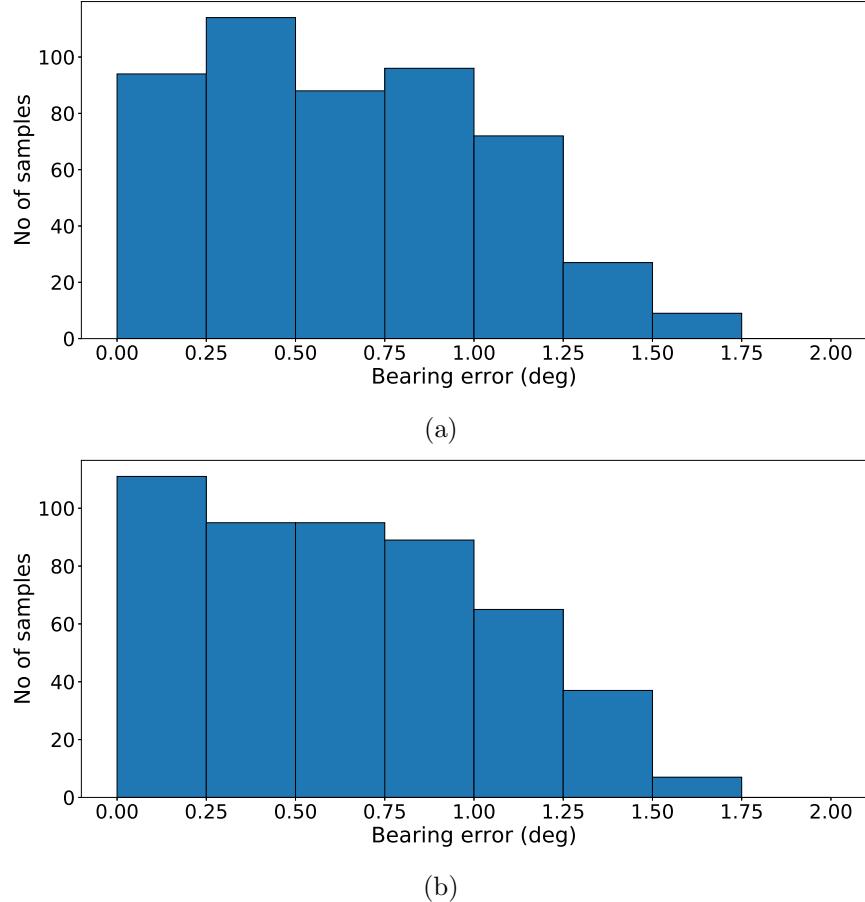


Figure 4.6: Histograms of bearing angle estimation errors for (a) 5.6×60 Magnum, and (b) .30-06 Springfield bullets.

The bearing angle θ of the shooter is defined with respect to a point (origin) and a direction (axis) of a coordinate system. The centroid of the square compound and its above-mentioned side naturally serve these respective purposes. The trajectory angle ψ found above is not exactly identical to this bearing angle, since, in general, the bullet trajectory will not pass through the centroid. However, since the sniper is very far from the compound (see fig. 4.5) and since the trajectory passes through a point inside the compound, it is assumed here that $\theta \approx \psi$.

A total of 500 gunshot events were simulated for each of the two bullet types, with the origin and target both randomized as described above. Figure 4.6 presents histograms of the bearing angle estimation errors for the two bullet types. The maximum bearing error observed across the 1000 cases is 1.75° only. The average error is 0.64° for the first bullet and 0.65° for the second one. Although two bullet types are used, the performance of the algorithm is similar for both of them. This is not surprising since the bullet-specific parameters are needed only in the simulation and not in the localization for this particular algorithm. Figure 4.7 provides a visualization of the performance for 10 representative shots of each of the two bullets investigated. Evidently, the algorithm is able to precisely estimate the shooter's bearing angle, which is the most important information sought in gunshot localization in any case.

Although all the preceding results are for a shooter range of 1 km, the localization algorithm should work (within limits) for other ranges too. To test this hypothesis, we investigated shooter

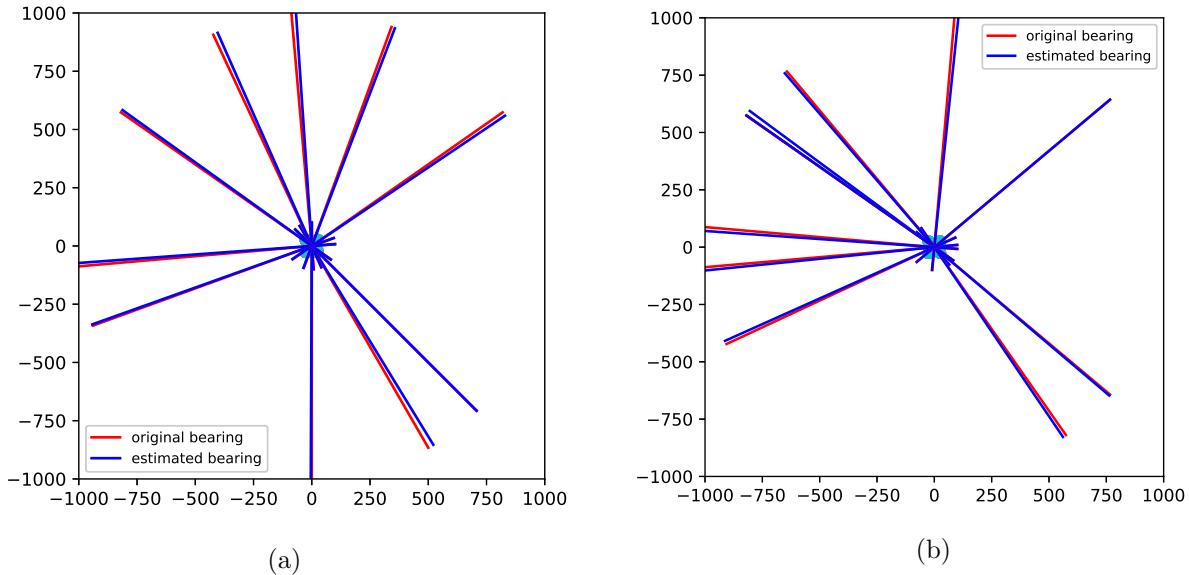


Figure 4.7: Actual and estimated bearing directions of the shooter for 10 representative gunshot events each, using (a) the 5.6×60 Magnum bullet, and (b) the $.30\text{-}06$ Springfield bullet. All dimensions are in meters.

ranges starting from 100 m up to 800 m in steps of 100 m (recall that the compound itself is 50 meter square). For each range, 100 random trajectories were simulated based on the procedure delineated above (for 1 km range), and the localization method was applied. The results are presented in fig. 4.8. In general, the performance continues to be quite encouraging for all ranges, down to 100 m. As expected, the error increases as the shooter nears the compound because of the greater inaccuracy in the approximation of the bearing angle by the calculated bullet trajectory direction (i.e., $\theta \approx \psi$). However, the maximum bearing angle estimation error (found when the shooter is nearest at 100 m) is 14.7° , which is still quite acceptable for our purpose. Of course, the performance will degrade further for ranges below 100 m, but such nearby gunshots can be located using the muzzle blast signature also (if not visually).

4.4.3 Effect of signal noise on performance of shock wave-based localization

The validation tests discussed in Section 4.4.2 did not incorporate any noise in the ballistic shock-wave signals simulated at the virtual microphones for a gunfire event. However, as alluded to earlier, no real signal is perfectly clean, and it is worthwhile to investigate the robustness of the localization algorithm in the presence of signal noise. In this section, we add a zero-mean Gaussian noise to the ideal ‘N’-wave signal.

For the constant-velocity bullet motion under consideration, the over-pressure amplitude of the shock wave signal captured by a sensor depends only on the corresponding miss distance of the bullet. Apart from moving forward along its trajectory, a bullet will also have perturbations in its other degrees of freedom. These disturbances may be received by a sensor along with the ideal signal. As the source of these disturbances is the same as the actual shock wave itself, one can expect the amplitude of the signal noise to scale with the miss distance, and hence with the over-pressure amplitude of the primary shock wave signal reaching the sensor. Thus, we will specify the noise level in the current simulations as a fraction – the noise-to-signal ratio. This will signify

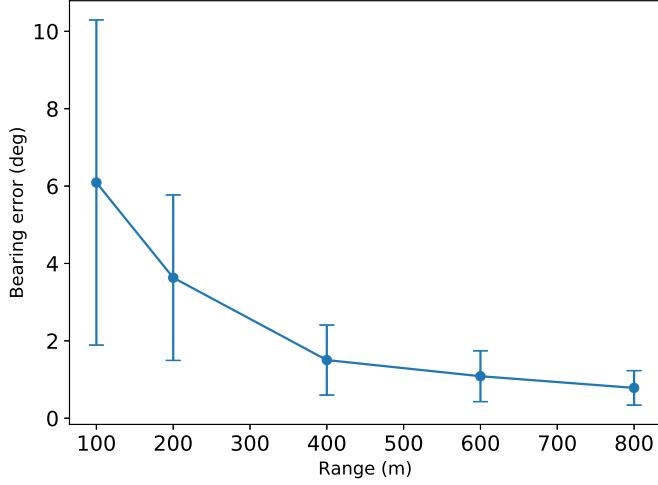


Figure 4.8: Performance of the shock wave-based localization algorithm in terms of the average error in estimated bearing angle for different values of the shooter range. The error bars signify the $\pm\sigma$ limits, where σ is the standard deviation of the bearing angle errors.

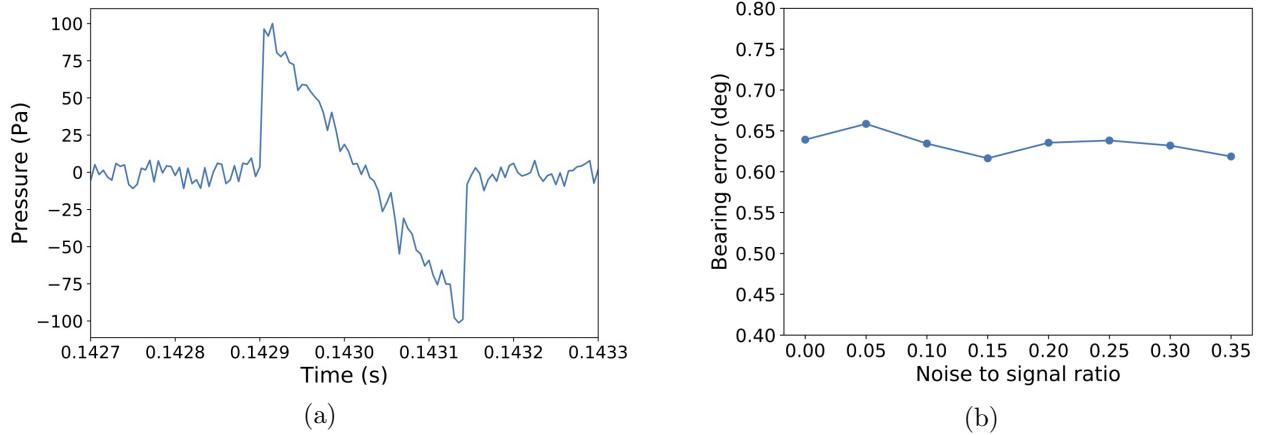


Figure 4.9: (a) Simulated shock wave signal at a sensor with the standard deviation of the additive Gaussian noise being 0.066 times the over-pressure amplitude of the ideal shock wave signal. (b) Mean error in bearing angle estimation as a function of the noise to signal ratio.

the standard deviation of the Gaussian noise corrupting the signal simulated at each sensor as a fraction of the over-pressure amplitude recorded at the same sensor. As an example, fig. 4.9a shows a simulated ‘N’-wave signal with 6.67% noise superimposed.

The bearing angle estimation performance was investigated by simulating cases with increasing noise percentage. In particular, 50 random bullet trajectories were simulated in the manner described in Section 4.4.2, with the sniper located 1 km away from the compound. Figure 4.9b demonstrates that the performance of the localization algorithm is essentially independent of the noise to signal ratios investigated, within the indicated range. This behaviour can be understood by recalling the fact that cross-correlation is used to identify the DTOA for each pair of sensors in a cluster, and this is inherently robust to noise.

Chapter 5

Summary and Conclusions

This project set out to develop and implement a real-time device for localizing gunshots based on their acoustic signature. Such a technology can have several use cases. Static devices can be deployed for surveillance of a compound that may encounter hostile gunfire. Portable ones may be fixed to vehicles for the same purpose. Finally, miniature localization devices can be mounted on the armour or even the person of security personnel to help them orient themselves in ambush situations.

There are two distinct acoustic signatures of a gunfire event – the spherically-expanding blast wave of the muzzle explosion, and the conical shock wave trailed by the bullet if it travels supersonically. Localization based on the muzzle blast is simpler as the center of the spherical acoustic wave directly reveals the shooter position, and it requires fewer sensors to do so. However, for very distant (but relevant) gunshot events like sniper fire, the muzzle blast may be too attenuated to be useful, leaving only the ballistic shock for use in localization. In the work reported here, the main focus has been on the hardware implementation of blast wave-based localization technology; the shock wave-based approach has been explored to a very limited degree.

Several muzzle blast localization algorithms have been explored in the project. The input for all of them is the differential time of arrival (DTOA) of the acoustic signature at the various microphone pairs in a cluster of closely-spaced sensors, along with their relative positions. The blast-wave algorithms attempt to estimate the bearing and range of the shooter by relating the DTOAs to the curvature of the wavefront. Identifying the center of the wavefront is inherently difficult if the shooter is far from the sensor cluster because the curvature becomes very small. Although this impedes the accurate estimation of the range of the shooter, the bearing angle can still be predicted easily as it is given by the local propagation direction of the (almost planar) wavefront. It is fortunate that bearing angle is the most critical information desired from a gunshot localization device; range, while desirable, is of secondary importance.

The localization technology implemented in this project (after several iterations) consists of the following physical components. Four microphones form a horizontal 1 meter square array (a cluster) mounted on a tripod. The microphones' signals are acquired by the high-rate analog-to-digital converter (ADC) of a Texas Instruments Tiva microprocessor; it performs some preliminary processing to abstract and send a short segment of the acquired data over an Ethernet cable to the more versatile Raspberry Pi (RPi) microprocessor that implements the actual localization algorithms. The result is currently written to a file on the RPi board for later access from a connected laptop, but may easily be displayed immediately on an on-board LCD in the future. The entire processing from arrival of the acoustic event at the microphones to the writing of the file (or future on-screen display) is completed within 0.1 second.

The device has been validated extensively by bursting firecrackers from known locations, since they produce a blast wave similar to firearms. The results were very encouraging, with the maximum error in bearing angle being 3° . The range prediction was more erroneous, as anticipated from the discussion above. The limited validation exercise pursued with firearms at a firing range were similarly successful.

The above-reported real-time localization assays were based on the muzzle blast signature of gunfire. A simple algorithm was also found to be effective, albeit in computer simulations, for determining the bearing angle of a distant shooter (a sniper) based on the ballistic shock wave.

The real-time gunshot localization device developed here needs to be augmented and improved in several aspects before it can be deployed. The program running on the Tiva microprocessor should be made more efficient to allow further increase of the sampling rate of the ADC from its current 80 kHz rate. This can either enhance the range estimation accuracy with the current microphone setup, or allow for the microphones to be brought closer to each other. Such miniaturization will also necessitate use of smaller microphones.

The shock-wave based localization algorithm needs to be implemented on the RPi board, and in a manner that meshes it properly with the existing blast-wave based program. The former will require data from two or more clusters of microphones, which will also mandate a corresponding enhancement of processing capacity of the microprocessors.

The 0.1 second latency in the current implementation from arrival of a gunfire's acoustic signature to production of location estimate is quite sufficient for real-time human use. However, this latency also means that the device pauses 'listening' for new gunshot events for the same duration after one event is captured by it. This limitation has to be mitigated to allow the localization of (almost) simultaneous gunfire coming from two different sources.

More and varied on-field experiments are needed to identify issues in the implementation, as well as to validate its performance more stringently. In particular, the tests done till now involved nearby gunfire (or firecracker bursts), with the maximum range tested being about 20 m. This constraint was imposed by operational considerations, but such small ranges are not very relevant in applications. Experiments need to be conducted with greater range of gunshots to determine the limitations of the implemented solution.

All in all, while a viable preliminary implementation of gunshot localization technology has been achieved, more remains to be done to mature it.

Bibliography

- IPv4 reference.* URL <https://tools.ietf.org/html/rfc791>.
- Raspberry Pi datasheet.* URL https://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM3plus_1p0.pdf.
- Tiva C Series TM4C1294 Connected LaunchPad Evaluation Kit – User’s Guide.* URL <http://www.ti.com/lit/ug/spmu365c/spmu365c.pdf>.
- J. R. Aguilar. Acoustic sensors and algorithms for urban security. In J. R. Vacca, editor, *Handbook of Sensor Networking: Advanced Technologies and Applications*, pages 235–248. Chapman and Hall/CRC, 2015.
- S. Bancroft. An algebraic solution of the GPS equations. *IEEE Transactions on Aerospace and Electronic Systems*, AES-21(7):56–59, 1985.
- S. D. Beck, H. Nakasone, and K. W. Marr. Variations in recorded acoustic gunshot waveforms generated by small firearms. *The Journal of the Acoustical Society of America*, 129(4):1748–1759, 2011.
- S. Bhattacharya. Real-time gunshot localization using acoustics. Master’s thesis, Indian Institute of Technology Bombay, 2020.
- J. Braasch. A precedence effect model to simulate localization dominance using an adaptive, stimulus parameter-based inhibition process. *The Journal of the Acoustical Society of America*, 134(1):420–435, 2013.
- J. Braasch. Sound localization in the presence of multiple reflections using a binaurally integrated cross-correlation/auto-correlation mechanism. *The Journal of the Acoustical Society of America*, 140(1):EL143–EL148, 2016.
- G. C. Carter. Time delay estimation for passive sonar signal processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(3):463–470, 1981.
- G. L. Duckworth, D. C. Gilbert, and J. E. Barger. Acoustic counter-sniper system. *Proc. SPIE*, 2938:262–275, 1997.
- A. Kothiwala. Development of an algorithm for gun-shot localization. Master’s thesis, Indian Institute of Technology Bombay, 2017.
- R. Maher. Modeling and signal processing of acoustic gunshot recordings. In *12th IEEE DSP Workshop & 4th IEEE Signal Processing Education Workshop*, pages 257–261, 2006.
- N. B. McNelis. Method and apparatus for determining the general direction of the origin of a projectile, US Patent 5,544,129, 1996.
- E. A. Page and B. Sharkey. SECURES: system for reporting gunshots in urban environments. *Proc. SPIE*, 2497:160–172, 1995.
- B. M. Sadler, T. Pham, and L. C. Sadler. Optimal and wavelet-based shock wave detection and estimation. *The Journal of the Acoustical Society of America*, 104(2):955–963, 1998.
- J. Sallai, P. Völgyesi, Á. Lédeczi, K. Pence, T. Bapty, S. Neema, and J. R. Davis. Acoustic shockwave-based bearing estimation. In *Proceedings of the 12th international conference on Information processing in sensor networks*, pages 96–107, 2013.
- G. S. Settles, T. P. Grumstrup, J. D. Miller, M. J. Hargather, L. J. Dodson, and J. A. Gatto. Full-scale high-speed “Edgerton” retroreflective shadowgraphy of explosions and gunshots. In *5th Pacific Symposium on Flow Visualisation and Image Processing*, 2005.
- N. Sirola. Closed-form algorithms in mobile positioning: Myths and misconceptions. In *7th Workshop on Positioning navigation and communication (WPNC)*, pages 38–44, 2010.
- R. B. Stoughton. SAIC SENTINEL acoustic counter-sniper system. *Proc. SPIE*, 2938:276–284, 1997.
- G. B. Whitham. The flow pattern of a supersonic projectile. *Communications on Pure and Applied Mathematics*, 5(3):301–348, 1952.