SELEST: Secure Elevation Estimation of Drones using MPC

Marinos Vomvas
Northeastern University

Erik-Oliver Blass
Airbus

Guevara Noubir
Northeastern University

ABSTRACT

Drones are increasingly associated with incidents disturbing air traffic at airports, invading privacy, and even terrorism. Wireless Direction of Arrival (DoA) techniques, such as the MUSIC algorithm, can localize drones, but deploying a system that systematically localizes RF emissions can lead to intentional or unintentional (e.g., if compromised) abuse. Multi-Party Computation (MPC) provides a solution for controlled computation of the elevation of RF emissions, only revealing estimates when some conditions are met, such as when the elevation exceeds a specified threshold. However, we show that a straightforward implementation of MUSIC, which relies on costly computation of complex matrix operations such as eigendecomposition, in state of the art MPC frameworks is extremely inefficient requiring over 20 seconds to achieve the weakest security guarantees. In this work, we develop a set of MPC optimizations and extensions of MUSIC. We extensively evaluate our techniques in several MPC protocols achieving a speedup of 300-500 times depending on the security model and specific technique used. For instance a Malicious Shamir execution providing security against malicious adversaries enables 536 DoA estimations per second, making it practical for use in real-world setups.

CCS CONCEPTS

• **Security and privacy → Domain-specific security and privacy architectures**; *Wireless security*; Privacy protections.

KEYWORDS

private drone localization, DoA estimation, multiparty computation

ACM Reference Format:
Marinos Vomvas, Erik-Oliver Blass, and Guevara Noubir. 2021. *SELEST*: Secure Elevation Estimation of Drones using MPC. In *Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '21), June 28–July 2, 2021, Abu Dhabi, United Arab Emirates.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3448300.3468228

1 INTRODUCTION

Localization of RF emissions is an increasingly useful primitive with many applications. Current applications include pinpointing the location of drone intrusions or locating and tracking the source of malicious emissions (e.g, a jammer). Especially drone intrusions have been at the center of a multitude of security incidents in recent years, with a dramatic increase in airport incidents [42], and the potential of terrorist attacks [1, 50]. With low cost commercial drones causing billion dollar damages [51] and being used to smuggle illegal substances and equipment [29], authorities are trying to find ways to monitor, control and even take down these targets [30].

As a first step to control drone intrusions, the US DHS/FAA has introduced new regulations, e.g., requiring permission authorizations using mobile apps such as B4UFLY [20]. As such requests are increasing in frequency, in some areas such authorizations are automatically processed and approvals are conditioned on respecting altitude limits. The incorporation of tracking technology into drones is also being considered [52], and the European Union Aviation Safety Agency (EASA) passed similar regulations for drones [3]. However, locating intrusion violations from non-compliant drones remains a challenge. While agencies such as DHS, EASA, FAA, FCC have installed regulations, there exist no mechanisms to enforce them. In practice, localization and tracking might need to be triggered in real time or a-posteriori based on proximity to a sensitive location (airports) or observations of other events such as interference or suspicious activity from a Wi-Fi MAC address.

Various techniques have been developed over the last few decades to automatically locate RF emissions, from active radars [55], to algorithms for multi-antenna systems such as MUSIC [48], ESPRIT [43], and Matrix Pencil [28]. Fundamentally, these techniques estimate the Direction of Arrival (DoA) of RF emissions by analyzing and correlating signals received by multiple antennas with a computer. Today, techniques are fairly accurate and reasonably efficient for a small number of emissions [2, 14]. Recent years have also seen the emergence of several commercial systems to locate drones such as Fortem TrueView Radar [23] or DJI Aeroscope [18].

However, the ability of a single system to permanently eavesdrop and analyze the wireless spectrum and track all RF emissions raises concerns and violates fundamental privacy laws. A compromise or intentional misuse of such systems could result in an indiscriminate tracking of users. Consequently, there is a need for systems which can collaboratively and in a privacy-preserving way track RF emissions, but only when they are deemed of interest. Such emissions could belong to a jammer, or a drone flying in a restricted zone or altitude without authorization. Locating and tracking RF emissions should only be possible when specific rules are violated and should not allow indiscriminately tracking of users.

A conceptually straightforward way to achieve privacy preserving localization consists of evaluating a localization algorithm using secure Multi-party Computation (MPC). Essentially, antennas send secret shared signals to a group of parties such that no single party has access to plain signals at any time. As a group, these parties are trusted to not collaborate, intentionally or unintentionally (e.g., when a subset is compromised) on trying to compute anything but the MPC-specified functionality. Jointly, the parties then analyze the signals working only on the secret shared data and output a

WiSec '21, June 28–July 2, 2021, Abu Dhabi, United Arab Emirates

ACM ISBN 978-1-4503-8349-3/21/06…$15.00
https://doi.org/10.1145/3448300.3468228

Table 1: Total time (s) for MPC elevation angle estimation of a single emission (3 parties) over 10ms latency network

Protocol	Malic. Advers.	Hon. Maj.	Standard MUSIC	This paper Opt-MUSIC	This paper ***SELEST***	**Speed up** ×
MASCOT	✓	N	**24**	2.52	**0.08**	**300**
Lowgear	✓	N	**24**	2.52	**0.08**	**300**
Cowgear	✓	N	**24**	2.52	**0.08**	**300**
Semi	-	N	**23.9**	2.47	**0.05**	**478**
Hemi	-	N	**23.9**	2.47	**0.05**	**478**
Mal-Shamir	✓	Y	**25.5**	2.63	**0.05**	**510**
Sy-Shamir	✓	Y	**43.6**	4.5	**0.16**	**272**
Ps-Rep	✓	Y	**25.6**	2.67	**0.08**	**320**
Shamir	-	Y	**25.4**	2.62	**0.05**	**508**
Rep3	-	Y	**25.3**	2.61	**0.05**	**506**

function of the emission source location (e.g., elevation estimate). However, as we show in Table 1, the implementation of an existing localization algorithm (such as MUSIC) in MPC frameworks results in exorbitant computation and communication costs.

In this work, we develop a DoA technique amenable to efficient optimization within several MPC frameworks, and in particular secure against malicious parties performing the analysis. We show that spatio-temporal cropping of RF samples results in a covariance matrix (of antenna samples) that has a single non-negligible eigenvalue. We also show that computing the pseudo-spectrum by MUSIC becomes equivalent to projecting steering vectors on a random combination of the covariance matrix. This technique is significantly more efficient than the use of a *QR* algorithm for eigenvalue and eigenvector computation as in MUSIC. We implement our technique within the MP-SPDZ framework [31], further optimizing to reduce complexity by, adjusting the norm, exploiting parallelism, avoiding divisions, square roots, comparisons. We evaluate our resulting algorithm (along our optimizations of MUSIC) in all common security models with arithmetic MPC protocols including MASCOT [32] and Malicious Shamir [13]. We perform extensive benchmarks in terms of computation and communication cost for both online and offline operations. We discuss the various trade-offs such as communication cost vs. computation, offline vs. online computation, and trust assumptions. In particular, we show that it is possible to process up to 775 emissions per second in the passive adversary setting, and up to 586 emissions in the active adversary setting with sufficient offline-computed Beaver triples.

We apply for ACM WiSec's replicability label. Our source code can be found here [54]. The technical highlights of this paper are:

(1) *SELEST*: A practical system for secure, privacy-preserving DoA estimation on top of MPC.
(2) An implementation and thorough performance evaluation of *SELEST* using arithmetic and binary circuits in two prominent MPC frameworks (MP-SPDZ [31] and EMP-Toolkit [56]), capable of elevation angle estimation with high rates and satisfying real-time requirements.
(3) An efficient implementation of arithmetic for complex numbers in MP-SPDZ and EMP-Toolkit.
(4) MP-SPDZ libraries for the QR Algorithm over complex numbers, together with array and matrix multiplications.

2 BACKGROUND

2.1 Basic Notation

Throughout the paper, we refer to matrices using uppercase bold letters (**A**) and column vectors using lowercase bold letters (**v**). The expectation of a random variable X is denoted as $\mathbb{E}[X]$, $\bar{z}$ denotes the complex conjugate of a complex number z, $\mathbf{E}^*$ denotes the conjugate transpose of matrix $\mathbf{E}$, and the nullspace of a matrix M is denoted by $\ker(M)$. Furthermore, with $[x]$ we denote the complete representation of a secret shared value x in an MPC circuit, and with $[x]_i$ we denote the share of x held by the party i.

2.2 Direction of Arrival Estimation

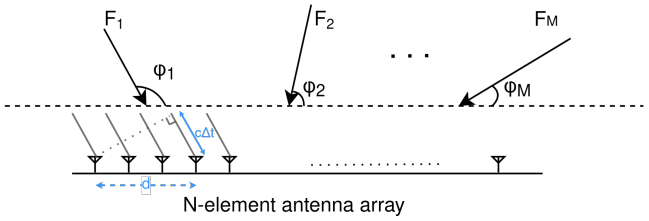


Figure 1: The DoA estimation problem.

Assume an antenna array consisting of N elements (Figure 1). We refer to incident signals or waveforms as the set of electromagnetic fields F_i created by distinct emissions that arrive on the antenna at an *angle* ϕ_i. In this setup, the measured signal (I/Q sample, referred to as complex sample) on an antenna element is called the antenna response and the vector of N antenna responses caused by an incident signal from angle ϕ is called *steering vector*, denoted as $\boldsymbol{\alpha}(\phi)$. Intuitively, there is a phase difference between the observed signals at two antennas due to the distance between them. The steering vector describes the phase differences within the array depending on its topology, and can be used to estimate the direction of arrival: the azimuth and/or elevation angles with respect to the antenna array orientation. Estimating the DoA has been a problem of interest in wireless communications research for many years and can be very challenging in practical applications [15, 21, 60, 61, 64].

MUSIC Algorithm. Multiple Signal Classification (MUSIC) [48] and its derivations is a set of popular high-resolution techniques for DoA estimation that uses *subspace* separation. In a nutshell, MUSIC analyzes the covariance within the received complex samples of all antennas and maps highly correlated values to signals, and low correlated values to noise. Due to the orthogonality between the noise and signal subspaces, the DoA is computed as the angle that minimizes the projection of the steering vector (triggered by the signals) on the noise subspace. The *pseudospectrum* output is the magnitude of this projection with respect to the angle θ. We provide a more detailed description and discussion of MUSIC in Section 4.

2.3 Secure Multiparty Computation

Secure multiparty computation (MPC) [40] allows a set of n parties $(P_1, \ldots, P_n)$ to jointly compute output $Y = F(x_1, \ldots, x_n)$ of any function on their respective inputs $(x_1, \ldots, x_n)$ without revealing more than what can be inferred from the output Y. Essentially, MPC emulates an ideal world where parties would send their inputs to a trusted third party which computes the desired functionality and sends the final output back to the parties.

As described in Sections 3 and 4, there exist several different techniques for different settings of secure multiparty computation, and in the following we informally summarize the intuition behind one idea. Function F is converted into a circuit representation, comprising of (Boolean or arithmetic) gates. Instead of evaluating the circuit on their real inputs x_i, parties distribute secret shares ($[x_i]$) to other parties and evaluate the circuit's gates on the shares $[x_i]$, thus hiding parties' inputs. The circuit is evaluated one gate at a time until output $[Y]$ is computed and revealed to all parties. Input privacy follows from the fact that no single party P_i is able to learn anything about the inputs of other parties.

Security Models. MPC allows participants to jointly evaluate a function even in the presence of *corrupted* parties. Corrupted parties may be assumed under the control of an adversary trying to extract information, affect the output or completely disrupt the computation. Various formal security definitions [12] have been proposed, informal definitions to some of the most important properties follow. Specific MPC protocols and techniques provide security guarantees for combinations of these properties.

Privacy: No party must learn anything from the computation other than what can be inferred from the output.

Correctness: Each party receives a correct output.

Guaranteed output delivery: The adversary may not obstruct the honest parties from learning the output of the computation.

Fairness: Either all, or no parties, learn the output.

Abort: The adversary is allowed to learn the output and abort the execution, before making it known to the honest parties.

In terms of adversarial behavior, a *semi-honest, passive* adversary (honest-but-curious) follows the protocol, but may use any observed messages from other parties to recover sensitive information. On the other hand, a *malicious* (active) adversary may arbitrarily deviate from the allowed protocol execution in order to affect security or correctness. A *covert* adversary may potentially act like a malicious adversary with a high probability of being caught, and penalized.

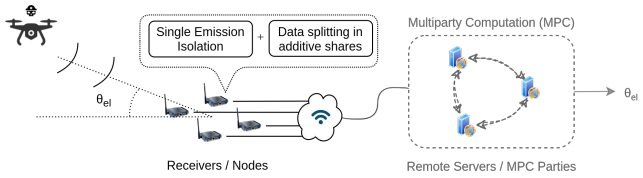


Figure 2: Private drone detection workflow

Arithmetic Operations in MPC. Since its introduction [10, 26, 62] many different MPC protocols have been proposed. The focus of this work is the private DoA estimation evaluation using arithmetic circuits, although we briefly discuss the evaluation in Boolean circuits as well in Section 5. The reasoning behind this is that arithmetic circuits are significantly more efficient in evaluating arithmetic operations such as addition and multiplication than Boolean circuits. The computation domain of arithmetic circuits is usually computation modulo a large number: either prime (denoted $\mathbb{Z}_p$), or power of two (denoted $\mathbb{Z}_{2^k}$).

A *secret sharing* scheme allows a party to split a value x into *shares* $[x]_i$, where $[x]_i$ is distributed to party P_i. Without loss of generality, we briefly illustrate an additive secret sharing mechanism between two parties. There, x will be shared such that $[x] = [x]_1 + [x]_2$. To actually share its input x, P_1 randomly samples r, sets their share to $[x]_1 = x - r$, sets $[x]_2 = r$, and sends $[x]_2$ to P_2. Informally, value x is hidden by r, and by adding their shares the parties can *reconstruct* x. We denote as $[x]$ the complete set of shares that define the value x, in this case: $[x] = \{[x]_1, [x]_2\}$.

We stretch notation and write $[x] + [y]$ to describe the addition (resp. other operations) of values x and y based on their shares. More specifically, by adding their shares $[x]_i + [y]_i$, P_i obtains a share $[x + y]_i$ of the sum $x + y$. Note that the same does not hold for local multiplication of shares. Instead, parties need to interact and exchange further information for each multiplication (see below) which introduces a significant communication overhead in the evaluation of an MPC circuit. Interactivity, the number of *communication rounds* relates to the *multiplicative depth* of the circuit. The *communication complexity* denotes the *amount of data* to be exchanged between parties during the computation. Finally, a sharing scheme defines a reconstruction mechanism that allows the parties to combine the shares of a value they are holding in order to learn the output. As local computations by each party are highly efficient, the total runtime of securely evaluating a circuit with MPC is dominated by network latency and throughput.

MPC using preprocessed data. Many MPC protocols take advantage of an input independent *offline* phase to generate correlated randomness to speed up computation in the subsequent *online* phase. A typical example of such randomness are Beaver triples [9] for multiplication. The idea is that a trusted third party prepares random products $[z] = [x][y]$ during the offline phase, which are *'corrected'* into the desired product $[c] = [a][b]$ during the online phase by revealing $d = [x] - [a]$ and $e = [y] - [b]$. Parties are now able to compute the product using only local operations (additions, multiplications by constant): $[c] = [a][b] = [z] - e \cdot [a] - d \cdot [b] - d \cdot e$. Other examples of preprocessed data include *random input gadgets* [17], and *random bits* [16, 19, 41].

3 PROBLEM STATEMENT

3.1 Problem description

In a traditional DoA estimation system consisting of one computer processing the data collected by an N-antenna array, this computer alone can track all emissions unconditionally. Instead, we propose a system of N separate *receivers*, each equipped with a receiving antenna, that forward received data to a set of *remote servers*, see Figure 2. The servers then privately estimate the DoA using MPC and reveal the output under a certain condition. In this setup, a *receiver* describes a deployed receiver in the area of interest with the ability to record data and forward them to a remote server. We assume that receivers are properly time and phase synchronized using standard techniques [6, 7, 36]. By *server*, *MPC party* or simply *party* we refer to a remote server participating in the MPC evaluation of the detection. The setting is described in Outline 1.

We note two major advantages of this approach. First, the computationally intensive MPC evaluation is performed on remote servers, which allows for the affordable deployment of large numbers of receivers. The computation can be run on the cloud, on demand, or in real time. Second, the data received from all antennas are never owned by a single entity in the clear, therefore protecting the location of regular users from systematic tracking.

3.2 Technical Challenges

In the following, we identify several technical challenges arising from both MUSIC and MPC.

MUSIC *in Practice.* Received data rates in practical wireless systems easily reach the order of millions of samples per second (e.g., monitoring 1MHz of spectrum results in 1 million complex samples per second), and estimating the received data covariance matrix and performing eigendecomposition is computationally highly demanding, even without MPC. Furthermore, there is the need for an extensive angle search to find the angle that maximizes the DoA estimation function. Moreover, to identify the noise subspace, MUSIC requires that the number of antennas is larger than the number of incident signals in the air, therefore the number of incident signals must be known or precisely estimated, which on its own is a difficult problem [22, 44, 58].

Secure computation overhead. A recent flurry of research on making MPC practical has resulted in multiple, ready-to-use open source frameworks to securely implement functionalities [11, 32, 34, 46, 47, 56, 63]. Today, MPC is increasingly used in a wide range of applications, including statistics [38], data analysis or end-to-end encryption [49], but available frameworks are still limited in terms of general usability. For example, there is no implementation that supports efficient complex arithmetic, a vital requirement in wireless system analysis (and crucial for DoA). Moreover, apart from communication overhead depending on specific MPC operations, mathematical operations such as comparisons, trigonometric functions or computing roots are converted to large, complex circuits which compute approximate results. Internally, these operations require fixed point arithmetic which is significantly more expensive and numerically sensitive than integer arithmetic. For reference, a single 32-bit fixed-point inverse utilizes 329 multiplications to approximate the result. MUSIC includes a wide range of such operations: fixed-point divisions, computing norms and square roots, comparisons, as well as trigonometric and logarithmic functions.

Outline 1 Secure DoA

Inputs: Received samples $\mathbf{x_j} = \{x_{i,1}, \ldots, x_{i,K}\}$ for K snapshots
Output: $f(\theta)$
Functionality:
1: Receiver j splits $\mathbf{x_j}$ into additive shares and forwards $\left[\mathbf{x_j}\right]_s$ to remote server (party) s.
2: Servers perform DoA estimation in MPC.
3: **return** $f(\theta)$, e.g.: output $f(\theta) = \theta$ if $\theta > \theta_{thr}$, otherwise $\perp$.

MPC protocols leverage multiple computationally heavy tools and cryptographic primitives, such as commitment schemes, zero knowledge proofs, large field operations, oblivious transfer, and more. Table 1 shows complete online execution times of MUSIC with 4 input samples using various state-of-the-art MPC protocols.

An additional challenge of computing over encrypted data is efficiently achieving the required fractional precision. Currently, many MPC frameworks omit a floating point implementation completely and rely on fixed-point arithmetic. Even though MP-SPDZ [31] provides both options, our own MP-SPDZ benchmarks have found fixed-point arithmetic both more efficient and better supported than floating-point arithmetic. This, in turn, introduces limits in the arithmetic range, resolution, and accuracy. Complex samples from received emissions aggravate these limitations: their scalars can vary a lot in magnitude based on the received signal strength and phase, both within, and across executions. Furthermore, the chosen bit-precision is an important factor in the performance of the MPC evaluation, as achieving higher range and resolution requires more computation and potentially larger field size. On the other hand, operations like norms, squaring and dividing by very small numbers are very susceptible to causing overflows.

Working on Isolated Emissions. The computational complexity of DoA algorithms, amplified by the computation and communication cost of MPC, make it clear that maintaining a practical, secure localization system in a modern congested RF environment is impossible without further optimization. However, processing one emission at a time lowers the complexity and run time of the algorithm as the number of antennas required is minimal, and fewer samples are sufficient to estimate the covariance matrix [45]. The problem arising in this scenario is the precise isolation of a few emission samples from a stream of wideband collected data.

With the rapid growth of Machine Learning and Artificial Intelligence in wireless communication systems, multiple tools have emerged that allow for wireless emission classification. [8] propose a tool for incremental learning of the surrounding RF environment by classifying known emissions and detecting newly encountered types of signals. Their evaluation shows real-time, high accuracy signal classification in the 2.4 GHz and 5 GHz bands. Using such tools, we can obtain samples that belong to exactly one signal and our working assumption onward is an isolated emission, i.e., one incident signal on the antenna array. Yet, Table 1 shows that even with the single emission assumption the performance improvement is not sufficient for a real time private detection system.

4 TECHNIQUES

Overview of our contributions. We propose a set of techniques for secure DoA estimation using MPC. For this, we introduce new techniques for MPC arithmetic over complex numbers in MP-SPDZ, an MPC-optimized version of the original MUSIC algorithm, and our novel extension of MUSIC, dubbed *SELEST*. This extension features optimized angle search, outperforming MUSIC in an MPC environment at the cost of acceptable loss of precision. All techniques have been both implemented and evaluated in relevant arithmetic-circuit MPC protocols of the MP-SPDZ framework [31] as well as in Yao's GC [62]. For comparison, we have also implemented *SELEST* in EMP-toolkit [56]. We first describe the basic techniques and optimizations (parallelism and relevant arithmetic optimizations) and their application to MUSIC. We report that while these optimizations result in over an order of magnitude improvement of performance relative to the standard MUSIC (Table 1), they remain impractical for real-time DoA computation. We then present our extension of MUSIC along with a proof of correctness.

Parallel communication. A major bottleneck in the evaluation of a circuit its multiplicative depth. In case of multiple independent multiplications, parties' interactions can be *grouped*, so the data for all multiplications can essentially be exchanged in one round of communication. Our complex arithmetic implementation supports and extensively uses this optimization offered by the MP-SPDZ compiler [32]. From now on, we will refer to it as *parallel communication*, noting the difference from *parallel executions*.

Secure complex arithmetic for MPC. A *fixed-point* representation is essentially an extension of the *integer* representation, consisting of an integer value and a scaling factor. Fixed-point arithmetic is usually preferred in MPC, because of its efficiency compared to *floating-point* arithmetic [31]. In most cases the range of the values supported is configured by the application such that fixed-point representation is sufficient for computation.

Operating on *complex* numbers (magnitude, phase, divisions) introduces fractional values and requires fixed point scalars for the *real* and *imaginary* parts. In our design, we define a complex number $z = x + jy$ as a tuple of two fixed-point scalars (x, y), and reduce complex operations to operations on the scalars. For complex multiplications, we use Knuth's standard technique which is considered numerically stable for practical use [27, 35]: $z = (a+jb)(c+id) = ac-bd+j[(a+b)(c+d)-ac-bd]$. Note that each complex multiplication requires only three scalar multiplications instead of four, grouped in a single communication round.

During our development process, we contributed to the MP-SPDZ framework by identifying a significant number of bugs, raising performance issues, and providing minor additional features. An overview of our contributions to MP-SPDZ can be found here [53].

4.1 Arithmetic optimizations

Besides optimization specific to MUSIC (below), we also perform the following general operations to speed up total runtime. All complex operations are *vectorized* [32] whenever possible, meaning that the same instruction is executed for consecutive memory registers to boost both compilation time and runtime (SIMD). Given our specific application requirements, we have also extended structures like *Arrays* and *Matrices* in MP-SPDZ to store and operate on data. These structures support parallelized scalar multiplication, matrix-vector multiplication, matrix multiplication, dot product, vector covariance, and norm ($L1, L2$) computation.

Pseudocode 2 GS orthogonalization(**A**)

```
B, R ← 0
for i = 1, ..., n_cols(A) do
    v ← A^T[i]
    B[i] ← v − proj_B(v)        // comm. parallelism in projection
for i = 1, ..., n_rows(R) do    // comm. parall. in R computation
    for j = i, ..., n_cols(R) do
        R[i][j] = ⟨B[i] · A^T[j]⟩
return B^T, R
```

QR Algorithm. MUSIC relies on the QR algorithm [24, 37] for the eigendecomposition of covariance matrices (see pseudocode 2). We implement the complex QR Algorithm using Gram-Schmidt orthogonalization. While it converges quickly for the small size of our input matrix, our implementation supports arbitrary sizes of input matrices and is extensible to different methods. We leverage parallel communication wherever possible, and we minimize the number of expensive operations such as norms, divisions and comparisons. Typically, the QR Algorithm iterates until the input matrix becomes almost diagonal, which in our experiments was in 1 to 3 iterations. Ours and any MPC evaluation requires the number of iterations to be fixed to not leak any information.

Input profiling. To balance the trade-off between the desired fractional resolution and arithmetic efficiency while avoiding overflows we perform input profiling and track the numerical progression during test MPC executions. As a first step, we scale the received samples to a range of $[0.1 - 1]$ at each receiver to ensure that the empirical covariance will not be arbitrarily large or small. This does not affect the outcome as it is simply a per-receiver gain adjustment, and our approaches exploit the phase correlation of the received signals. At every step of the execution, we mark the required arithmetic range and maintain an accuracy of at least 3 decimals, to identify the minimum fixed-point precision. This leads to a 32-bit signed fixed-point number with 24 bits for the fractional part for the Optimized MUSIC implementation and to a 13-bit signed fixed-point number with 9 bits for the fractional part for *SELEST*.

4.2 Data split

Samples received by antennas are forwarded to potentially untrusted servers and subsequently used as inputs for MPC evaluation. This raises security concerns. First, a semi-honest adversary in control of a subset of servers is able to process the correlation of the data their servers hold. Additionally, a malicious adversary can alter the data before engaging in the MPC. We stress that this is different than a standard input substitution attack [25], as in this case the input comes from another, honest entity and must not be substituted. Informally, the server is playing the role of a middle-man.

To address these concerns, the received data sample $\mathbf{x}$ is simply split at the receivers' level into additive shares, such that $\mathbf{x} = \mathbf{x}_1 + \ldots + \mathbf{x}_M$, and $\mathbf{x}_s$ is forwarded to remote server s. Observe that this data split is extremely cheap and can be performed with

Pseudocode 3 Empirical covariance matrix calculation

```
for i = 1, ..., K do                    // Parallelism across snapshots
    S ← S + xx*                         // Parallelism in covariance
S ← 1/(K−1) S                           // aggregate instead of averaged
```

Pseudocode 4 Pseudospectrum search

```
for i = 1, ..., 180 do                  // Parallelism across angles
    P_MUSIC(φ) ← 1/||a(φ)E_n^*||^2      // avoid inversion
```

high rates even by resource constrained receivers. After the input sharing phase of the MPC protocol, server s obtains share $[\mathbf{x}]_s$ by adding $[\mathbf{x}_l]_s$, for $l = 1, \ldots, M$.

4.3 Optimized MUSIC

We apply our MPC optimization techniques to the MUSIC algorithm and provide a secure, efficient implementation for arithmetic MPC circuits. Given the scenario discussed in Section 2.2, the array of received samples can be written as $\mathbf{x} = \mathbf{Af} + \mathbf{w}$, where $\mathbf{A}$ is a $N \times M$ matrix of steering vectors, $\mathbf{f}$ is a complex vector of the incident signals on the antenna, and $\mathbf{w}$ is the (environmental and instrumental) noise vector. MUSIC takes advantage of the *signal* covariance matrix $\mathbf{S_s} = \mathbf{Aff}^*\mathbf{A}^*$ that contains the collective information of all antenna responses stimulated by the incident signals $\mathbf{f}$, without the noise. Assuming that the signals and noise are uncorrelated, any vector $\mathbf{q_m}$ uncorrelated to the signals must belong to $\ker(\mathbf{S_s})$ and, by definition, be orthogonal to all steering vectors in $\mathbf{A}$. Thus, expression $\|\mathbf{A}^*\mathbf{q_m}\|^2$ is equal to zero. As the signal correlation $\mathbf{S_s}$ cannot be obtained in practice, $\ker(\mathbf{S_s})$ is estimated by the covariance matrix of the received data over K snapshots in time:

$$\mathbf{S} = \mathbb{E}\left[\mathbf{xx}^*\right] = \frac{1}{K-1}\sum_{i=1}^{K}\mathbf{x_i}\mathbf{x_i}^* = \mathbf{S_s} + \mathbb{E}\left[\mathbf{ww}^*\right] = \mathbf{S_s} + \sigma^2\mathbf{I}, \quad (1)$$

where σ^2 is the noise variance. In fact, the eigenvectors of $\mathbf{S_s}$ corresponding to the zero eigenvalue are the exact eigenvectors of $\mathbf{S}$ that correspond to the σ^2 eigenvalue. Then, if $\mathbf{E_n}$ denotes the matrix of these eigenvectors, MUSIC plots the *pseudospectrum*, i.e., levels corresponding to the magnitude of the above projection for different possible angles of incident signals:

$$P_{MUSIC}(\phi) = \frac{1}{\boldsymbol{\alpha}^*(\phi)\mathbf{E_n}\mathbf{E_n}^*\boldsymbol{\alpha}(\phi)} = \frac{1}{\|\mathbf{E_n}^*\boldsymbol{\alpha}(\phi)\|^2} \quad (2)$$

The points of the peak values of eq. (2) correspond to the estimated DoA, and the magnitude of the peaks directly relate to the strength of each received signal.

We have implemented MUSIC for MPC in MP-SPDZ, but institute the following changes resulting in major performance improvements. In estimating the covariance matrix for every snapshot (pseudocode 3), communication rounds are kept to a minimum since the parties only need to interact in order to compute half the matrix (plus the diagonal). The rest of the matrix is computed locally by Hermitian symmetry. Additionally, parallel communication is exploited both in computing the covariance during every snapshot and across all K snapshots. Finally, we use the aggregate covariance matrix instead of the average, skipping the last step to avoid the unnecessary overhead of fixed-point division. This results to an estimate scaled by some factor, but we stress that this factor has no impact on the matrix eigenstructure.

Another computationally heavy part of MUSIC is the pseudospectrum search (pseudocode 4), which evaluates eq. (2) for every angle of the plane. Even though this operation is also parallelized, for every angle a significant amount of multiplications and one division are evaluated. We implement the square norm with parallel communication and omit inversion to minimize the communication rounds. After the calculation of the pseudo-spectrum, a function of the elevation angle is revealed according to the application requirements. For instance, the elevation angle is revealed if it is over a certain threshold. This requires $n-1$ comparisons, where n is the size of the calculated pseudospectrum, or, comparing the elements in pairs results in $\lceil \log n \rceil$ comparisons.

4.4 *SELEST*

We show that for the single emission case we can avoid a lot of the computation of standard MUSIC and still obtain accurate results. Instead of using the noise subspace, we revert to the signal subspace and use the covariance matrix of the received signals to estimate the DoA to avoid the overhead of the complex QR Algorithm. At the same time, we optimize the operations for the multi-party computation case.

We prove correctness of our single emission detection, starting from the MUSIC algorithm. Let $\mathbf{S}$ be the covariance matrix of eq. (1): $\mathbf{S} = \mathbb{E}\left[\mathbf{xx}^*\right] = \mathbf{APA}^* + \sigma^2\mathbf{I}$, with $\mathbf{P} = \mathbf{ff}^*$.

MUSIC utilizes the fact that the eigenvectors of $\mathbf{S_s}$ which correspond to the zero eigenvalue are orthogonal to all M signal steering vectors, in order to maximize (2). Let $\mathbf{E_n}$ be the matrix of these eigenvectors as columns, and $\mathbf{E_s}$ be the matrix of the eigenvectors that correspond to non-zero eigenvalues as columns.

Lemma 4.1. *The values that minimize the denominator of (2), are the same values that maximize the expression* $P_s = \boldsymbol{\alpha}^*(\phi)\mathbf{E_s}\mathbf{E_s}^*\boldsymbol{\alpha}(\phi)$

Proof. By the Hermitian property of $\mathbf{S_s}$, eigenvectors corresponding to distinct eigenvalues are orthogonal, therefore $\mathbf{E_n} \perp \mathbf{E_s}$. Because $\mathbf{E_s}$, $\mathbf{E_n}$ completely define the eigenvectors of $\mathbf{S_s}$, the signal steering vectors $\boldsymbol{\alpha}(\phi_s)$ that are orthogonal to $\mathbf{E_n}$ must be parallel to a corresponding signal eigenvector $\mathbf{q_s}$ in $\mathbf{E_s}$. This naturally creates a local maximum of $P_s = \|\mathbf{E_s}^*\boldsymbol{\alpha}(\phi)\|^2$ at ϕ_s. □

Lemma 4.2. *In the case of a single incident waveform, projecting on any column vector of the signal covariance matrix* $\mathbf{S_s}$ *is equivalent to projecting on the eigenvector of* $\mathbf{S_s}$ *that corresponds to the single non-zero eigenvalue.*

Proof. In the case of a single emission, $\mathbf{P} = \mathbf{ff}^* = \mathbb{E}\left[|f_1|^2\right]$, $\mathbf{S_s}$ is clearly a $N \times N$ matrix with rank 1 and has $(N-1)$ zero eigenvalues. $\mathbf{S_s}$ can be decomposed into matrices $\mathbf{E_s}\boldsymbol{\Lambda}\mathbf{E_s}^*$ with $\mathbf{E_s} = [\mathbf{q_s}, \mathbf{n_1}, \mathbf{n_2}, \mathbf{n_3}]$, $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_s, 0, 0, 0)$; $(\lambda_s, \mathbf{q})$ being the single signal eigenvalue-eigenvector pair. Therefore $\mathbf{S_s} = \mathbf{E_s}\boldsymbol{\Lambda}\mathbf{E_s}^* = \lambda_s\mathbf{qq}^*$ and by the property of the outer product of a vector with itself, $\mathbf{S_s}$ has rank 1 and all its columns are linearly dependent on $\mathbf{q}$. □

It is crucial to note that the important information is contained in the phase of the samples and the norm of a vector is rather irrelevant. Also, in practice it is impossible to obtain a column vector of $\mathbf{S_s}$ but we can closely estimate it from the received covariance matrix $\mathbf{S}$.

Theorem 4.3. *In the case of a single incident waveform on an antenna array, DoA estimation using the covariance matrix is equivalent to DoA estimation using the noise subspace (MUSIC).*

$$P_s = \boldsymbol{\alpha}^*(\phi)\mathbf{E_s}\mathbf{E_s}^*\boldsymbol{\alpha}(\phi) \approx \boldsymbol{\alpha}^*(\phi)\mathbf{S_i}\mathbf{S_i}^*\boldsymbol{\alpha}(\phi) = P_{SELEST}$$

Please refer to the Appendix for the proof of the theorem.

4.4.1 Localization Output.

Coarse pseudospectrum output. DoA estimation algorithms such as MUSIC output a complete view of the pseudospectrum, with the peaks denoting the magnitude and angle of arrival of the incident signals. Our goal is to provide a flatter, *coarser* representation of the pseudospectrum by averaging the output power over a range of angles. This way, we create a rough picture of the elevation plane without explicitly revealing the elevation of the targeted emissions. We achieve this by using a modified steering vector $\hat{\boldsymbol{\alpha}}(\phi)$ which includes the average information of the array's original steering vector over a certain range of angles f. In addition to the coarse pseudospectrum view, this provides significant boost in the performance of the MPC circuit because of the reduced number of operations. Figure 3 demonstrates the accuracy as well as the output coarseness of this method, for a factor $f = 18$.

If k is the degree resolution of the original steering vector of eq. (4), we obtain s points for our modified steering vector with $b := \lceil \frac{k}{f} \rceil$ and we compute $\hat{\boldsymbol{\alpha}}(\phi) = [\hat{\boldsymbol{\alpha}}(\phi)_1, \cdots, \hat{\boldsymbol{\alpha}}(\phi)_b]$, $\hat{\boldsymbol{\alpha}}(\phi)_i = \frac{1}{f}\sum_{j=i\cdot f}^{i\cdot f+f-1} \boldsymbol{\alpha}(\phi)_j$ and from eq. (2):

$$P_{SELEST}(\phi) = \hat{\boldsymbol{\alpha}}^*(\phi)\mathbf{v_s}\mathbf{v_s}^*\hat{\boldsymbol{\alpha}}(\phi) = \|\mathbf{v_s}^* \cdot \hat{\boldsymbol{\alpha}}(\phi)\|^2 \tag{3}$$

In the above, $\mathbf{v_s}^*$ denotes the chosen signal vector from the covariance matrix according to Theorem 4.3. The effect of inversing expression 2 is reflected in Figure 3b by a wider, flatter peak, compared to Figure 3a. This coarse approximation achieves high performance without leaking sensitive information, but still reveals more than nothing about the location of the target emission.

Conditioned output. In addition to the above output method, we consider the case where nothing is revealed besides whether a signal matches a certain condition in the monitored area. For example, we consider areas where flying a drone is permitted under regulations such as a maximum allowed flying altitude. Thus, we modify the output of our algorithm to compare the calculated pseudospectrum values and output only *Detection* if the estimated elevation angle exceeds some threshold. In general, the output as a function of the elevation angle $Y = f(\theta)$ incurs the extra cost of evaluating the output condition in MPC given the computed pseudospectrum from eq. (3). Both of these techniques are used in both Optimized MUSIC and *SELEST*.

We present *SELEST* in Algorithm 5. *SELEST* supports the general case of M servers processing K snapshots collected by N antennas for two output scenarios: Coarse and Conditioned. Protocol P can be any arbitrary M-party MPC protocol. The algorithm for Optimized MUSIC replaces $[\mathbf{v_S}]$ in line 7 of Algorithm 5 by $[\mathbf{E}_N] = \mathcal{F}_{\mathcal{QR}}([S])$ using our optimized QR Algorithm $\mathcal{F}_{\mathcal{QR}}$ (Algorithm 7) and continues accordingly.

Algorithm 5 *SELEST*

Inputs: Received samples $\mathbf{x_j} = \{x_{j,1}, \ldots, x_{j,K}\}$ for snapshots $i \in \{1, \ldots, K\}$, for antennas $j \in \{1, \ldots, N\}$. Angle θ_{thr}. Algorithm $A \in \{\text{Coarse, Conditioned}\}$. Protocol P.

Stage 1: Receiver Setup

1: Receiver j splits samples $\mathbf{x_j}$ into additive shares $x_{i,j}^{(1)}, \ldots, x_{i,j}^{(M)}$ for all $i \in \{1, \ldots, K\}$. and forwards shares $x_{i,j}^{(s)}$ to server s for all $s \in \{1, \ldots, M\}$.

Stage 2: MPC

2: Server s secret shares $x_{i,j}^{(s)}$ according to protocol P for all $i \in \{1, \ldots, K\}$ and $j \in \{1, \ldots, N\}$.

3: Server s computes $\left[x_{i,j}\right]_s = \sum_{\ell=1}^{M} \left[x_{i,j}^{(\ell)}\right]_s$ for all $i \in \{1, \ldots, K\}$, $j \in \{1, \ldots, N\}$.

4: **for** $i = 1, \ldots, K$ **do**

5: $[\mathbf{S_i}] = [\mathbf{x_i}][\mathbf{x_i}^*]$, where $[\mathbf{x_i}] = \left[\left[x_{i,1}\right], \ldots, \left[x_{i,N}\right]\right]$

6: $[\mathbf{S}] = \sum_{i=1}^{K} [\mathbf{S_i}]$

7: Choose a column of $[\mathbf{S}]$, $[\mathbf{v_S}] := [\mathbf{S}]^T(0)$.

8: **for** $\psi = 1, \ldots, b$ **do**

9: $[P_{SELEST}(\psi)] = [\mathbf{v_S}] \cdot \hat{\boldsymbol{\alpha}}(\theta)^T[\psi]$

10: **if** A = Coarse **then**

11: **return** P_{SELEST}

12: **else**

13: **if** A = Conditioned **then**

14: $[\theta_{el}] := \arg\max_{\psi \in \{1,\ldots,b\}} \{[P_{SELEST}(\psi)]\}$

15: **if** $[\theta_{el} > \theta_{thr}]$ **then**

16: **return** θ_{el}

17: **else**

18: **return** $\perp$

Algorithm 6 Optimized Gram Schmidt $\mathcal{F}_{\mathcal{GS}}$

Inputs: Secret shares of input matrix $[\mathbf{A}]$

Output: Secret shares of orthonormal basis $[\mathbf{Q}]$, triangular $[\mathbf{R}]$

1: $[\mathbf{B}] = \mathbf{0}$

Stage 1: Form $[\mathbf{Q}]$

2: **for** $i = 1, \ldots, n_{cols}([\mathbf{A}])$ **do**

3: **for** $j = 1, \ldots, n_{rows}([\mathbf{B}])$ **do**

4: $[\mathbf{A}[i]] = [\mathbf{A}[i]] - \langle [\mathbf{A}[i]], [\mathbf{B}[j]] \rangle \cdot [\mathbf{B}[j]]$

5: $[\mathbf{B}[i]] = \frac{[\mathbf{A}[i]]}{[\|\mathbf{A}[i]\|]}$

Stage 2: Form $[\mathbf{R}]$

6: $[\mathbf{R}] = \mathbf{0}$

7: **for** $i = 1, \ldots, n_{rows}([\mathbf{R}])$ **do**

8: **for** $j = i, \ldots, n_{cols}([\mathbf{R}])$ **do**

9: $[\mathbf{B}[i][j]] = \langle [\mathbf{A}[j]], [\mathbf{B}[i]] \rangle$

10: **return** $[\mathbf{Q}]$, $[\mathbf{R}]$

Algorithm 7 Optimized QR Algorithm $\mathcal{F}_{\mathcal{QR}}$

Inputs: Secret shares of input matrix $[\mathbf{A}]$, iterations $iter$

Output: Secret shares of triangular $[\mathbf{A}]$, eigenvectors $[\mathbf{Q}_c]$

1: $[\mathbf{Q}_c] = \mathbf{I}$, $[\mathbf{A}_1] = [\mathbf{A}]$

2: **for** $i = 1, \ldots, iter$ **do**

3: $[\mathbf{Q}], [\mathbf{R}] = \mathcal{F}_{\mathcal{GS}}([\mathbf{A}_i])$

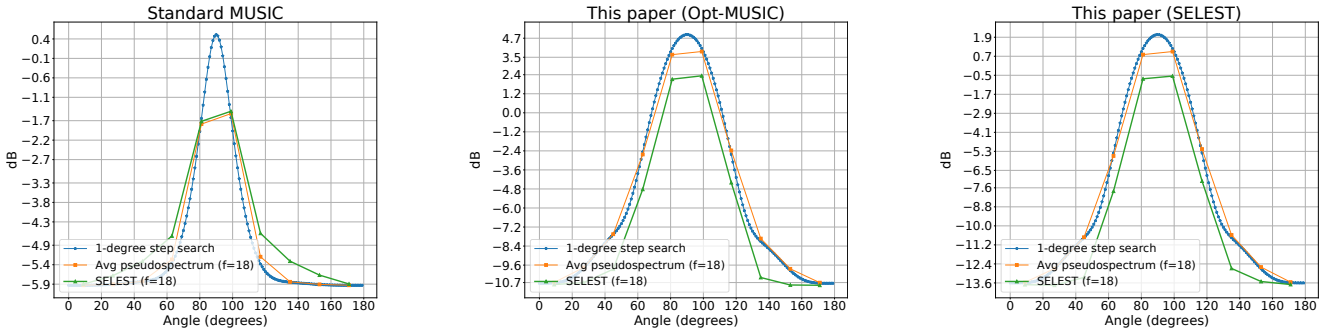
4: $[\mathbf{A}_{i+1}] = [\mathbf{R}] \cdot [\mathbf{Q}]$

5: $[\mathbf{Q}_c] = [\mathbf{Q}_c] \cdot [\mathbf{Q}]$

6: **return** $[\mathbf{A}_{iter+1}]$, $[\mathbf{Q}_c]$

5 IMPLEMENTATION AND EVALUATION

We have implemented and evaluated MUSIC, Opt-Music, and *SELEST* in MP-SPDZ to show efficacy of our optimizations and, specifically, the practicality of *SELEST* as a privacy-preserving drone localization system. The source code is available for download [54].



(a) MUSIC **coarse pseudoscpectrum** **(b)** ***Opt-MUSIC*** **coarse pseudoscpectrum** **(c)** ***SELEST*** **coarse pseudoscpectrum**

Figure 3: Coarse DoA Estimation of a single emission at 90° angle with averaging factor 18

In our implementation, we choose signed fixed-point arithmetic as required by each of our algorithms according to Section 4.1. For computations on fixed-point numbers, we use MP-SPDZ standard 128 bit field size to maintain security for operations on numbers of length 32 bit. All evaluations were performed on a single server with a 16-Core Intel Xeon E5-2660@2.20GHz processor and 128 GByte of RAM. As latency dominates total runtime, communication between parties takes place over the loopback interface, and all parties run on the same machine. This allows us to precisely control latency via the Linux kernel Traffic Control (tc/netem) interface.

All techniques have been tested on both synthetic and real data. Synthetic data was created in Matlab from random data streams and additive white Gaussian noise. Real data was captured from WiFi dongles and a DJI Phantom 4 commercial drone in outdoor spaces to reflect realistic environments. The emissions were recorded using a Uniform Linear Array (ULA) consisting of 4 antennas, a single Ettus USRP X310 synchronized using an Ettus OctoClock CDA-2990, and 4 snapshots samples were processed per antenna.

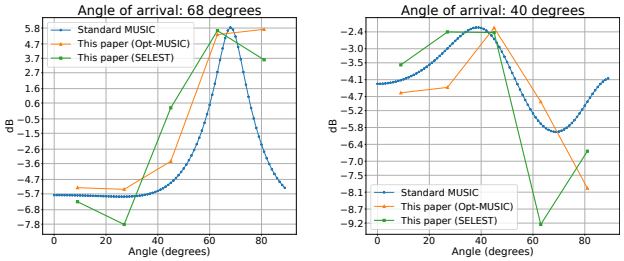
In the following, we analyze accuracy, security, performance, and costs of our methods with respect to latency and protocol used. We focus on performance during protocols' online phases and discuss the preprocessing separately. We measure throughput in terms of DoA estimations per second to reflect requirements of a real-world localization system, and present indicative monetary costs for the deployment of a system based on our techniques, taking computational and network traffic costs into account. To compare performance for different levels of security, we evaluated our techniques using various state of the art MPC protocols, covering a wide range of adversary models. Table 3 provides a comprehensive summary of our findings.

5.1 *SELEST* evaluation

Detection accuracy. Figure 4 compares the accuracy of our techniques against the results of the standard MUSIC algorithm on the same input data. The results of standard MUSIC were produced by our Python implementation, using a discretized angle search of 1 deg. The results of our algorithms were obtained by the output of the MPC Coarse pseudospectrum evaluation using a modified steering vector with $f = 18$. All results were then shifted into the same logarithmic scale for better comparability.

Table 2: *SELEST* offline cost for maximum throughput case in consumed triples per hour and triple generation cost in $/h

Protocol	Usage Rate (10^6 triples/h) Coarse	Conditioned	Cost ($/h) Coarse	Conditioned
MASCOT	43.87	48.76	0.693	0.77
Lowgear	43.99	69.68	0.695	1.101
Cowgear	43.26	66.15	0.683	1.044
Semi	65.87	79.25	1.04	1.251
Hemi	63.45	84.67	1.002	1.337
Mal-Shamir	220	290.7	3.474	4.59



(a) DoA = 68 **(b) DoA = 43**

Figure 4: Coarse pseudospectrum identifying the DoA of transmitted signals from various angles

The effect of the modified steering vector is a pseudospectrum view with the same trend as MUSIC's, simultaneously hiding any particular position of the peak up to an extent depending on the chosen f. Higher values for f result in an even coarser view of the pseudospectrum with slightly improved performance, because of the decreased amount of computation. Lower values of f provide a more refined view. Figure 4 shows the expected coarse approximation of up to $f/2$ off the actual angle for over-the-air transmitted sine waves (Fig. 4a) and over-the-air transmitted drone emission (Fig. 4b) Our coarse pseudospectrum output accurately reflects the actual angle of arrival of the target emission in order to infer whether a device violates a regulation such as altitude restriction.

Security & Privacy. To evaluate different performance vs. security trade-offs, we evaluate *SELEST* for different adversary models

as shown in Table 3. MASCOT and Lowgear provide the strongest security guarantees out of the evaluated protocols, being secure in the presence of a dishonest majority of malicious adversaries, while Cowgear is secure against covert adversaries. Semi and Hemi are semi-honest secure variations of the above protocols, but maintain security against a dishonest majority. These protocols provide the flexibility of arbitrary number of honest and corrupt parties at the expense of increased computation.

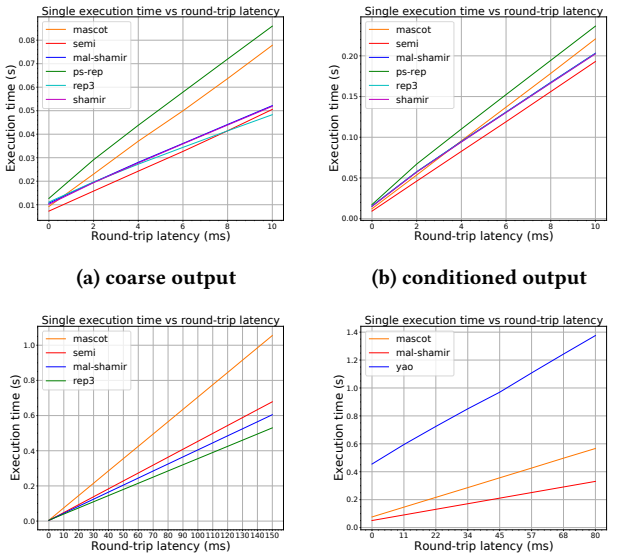
MPC based on Shamir's secret sharing protocols achieve security against an honest majority of malicious adversaries with a noticeable improvement in performance and cost. Similarly, the replicated secret sharing, 3-party computation protocols Ps-Rep and Rep3 achieve malicious and semi-honest security respectively for a honest majority of parties, by having two parties holding a share of a value unknown to the third. This way, two out of three parties are able to reconstruct a value. The above compromises are reflected in the performance evaluation in Table 3.

Our system inherently requires trust in the input shares sent by the receivers to the servers. In any system, an untrusted receiver could provide incorrect inputs to affect the correctness of the result, and application specific techniques (e.g. redundancy based) are required to cope with this type of attack. Additionally, a malicious server could modify a receiver's shares. There are standard techniques to mitigate this: an antenna could send cryptographic hashes of every sample to all parties, which can be verified in the MPC circuit after the input reconstruction. During the evaluation of the MPC circuit, security is obtained by the chosen protocol, and no information is leaked about the recorded samples. In the case of the coarse pseudospectrum output, the elevation peaks cannot be traced back to a specific set of input samples and the exact angle of arrival remains private.

Network impact. We evaluate two crucial factors in the performance of the system, communication rounds and communication complexity. For the former, we show the effect of latency in various scenarios and protocols. For the latter, we show network traffic costs based on the total data exchanged for every second of the evaluation, and the current $0.01 cents/GByte traffic cost between certain AWS instances[4]. Running the same experiments on a similar setup using EC2 AWS instances, a reserved m4.4xlarge instance would cost $0.496 effective hourly. Furthermore, the data transferred between the parties during a single online execution of *SELEST* are in the worst case ~0.33 MByte one-way for protocol Sy-Shamir for conditioned output. Even at very high throughput, this requires speeds of ~50 MBytes/sec, far within the range of the 25 GBit/sec connections between AWS instances [5].

Figure 5 presents the effect of communication rounds for various protocols in a single MPC evaluation of *SELEST*, for round trip time (RTT) latency up to 150 ms which reflects high quality intra-continental WAN connections. For consistency, we conduct the rest of the experiments in a 10 ms RTT setting, at the time of writing the worst case for cloud server instances like AWS EC2 within a local zone [59]. The amount of exchanged data and therefore the bandwidth is a negligible factor in the performance compared to the communication rounds and implied network latency.

Performance. With multiplications requiring interaction, in a single execution of *SELEST* communication between parties is invoked



(a) coarse output **(b) conditioned output**

(c) coarse output, high latency **(d) GC vs. arithmetic protocols**

Figure 5: *SELEST* execution times varying network latency

10 to 28 times depending on the underlying protocol. This results to idle computational resources, waiting for the data exchange, especially as the network latency increases. We have heuristically estimated maximum parallelism for each protocol by measuring the performance over increased executions for various latency values. For low latency, we find that most protocols peak in performance at 1 to 10 parallel executions. For a 10 ms RTT the maliciously-secure protocols' performance peaked at 60 parallel executions, and the semi-honest protocols' performance peaked at $120 - 180$ parallel executions. This verifies our assumption that executions can be highly parallelized to increase the system throughput, in terms of secure DoA estimations per second. We attribute this difference in the additional computation required by the malicious protocols for operating on the MACs of the shares to detect inconsistencies in the parties' shares and computations. We note that for every execution, N party instances are invoked (in our case $N = 3$), and, being executed on the same server, the player instances share a common pool of resources, thereby restricting the potential parallelism by a factor of 3.

We demonstrate the online performance of *SELEST*, evaluated as a throughput of achieved DoA estimations per second in two cases, *coarse pseudospectrum* and *conditioned*. For reference, we also evaluated the performance of MUSIC in Python (using Hermitian-optimized eigendecomposition by NumPy) on the same inputs, and hardware/DoA configuration, and measured 7.44ms per execution, as opposed to 24s (Table 1). In order to account for MUSIC's ability to process multiple emissions, we scale up its performance by a factor of three (there are four antennas in our system), and compute a maximum throughput of 403.2 DoA/s, almost 4 orders of magnitude faster than the Standard MUSIC implementation in MPC (Table 1).

Preprocessing costs. Several protocols in Table 3 achieve a faster online phase when the required randomness is computed during an offline phase. While all of the arithmetic protocols benefit from

Table 3: *SELEST* online performance (secure DoA estimations/second), 3 parties, 10ms RTT, traffic cost in cents/hour

	Model		Coarse				Conditioned			
	Malic.	Hon.	Single		Parallel		Single		Parallel	
Protocol	Advers.	Maj.	DoA/s	Cost (c/h)	DoA/s	Cost (c/h)	DoA/s	Cost (c/h)	DoA/s	Cost (c/h)
MASCOT	✓	N	13.1	0.001	106.9	0.09	5	0.002	38.7	0.126
Lowgear	✓	N	13.1	0.001	107.2	0.119	4.9	0.002	55.3	0.377
Cowgear	✓	N	13.1	0.001	105.4	0.119	5.0	0.002	52.5	0.377
Semi	-	N	20.2	0.001	160.5	0.117	5.8	0.002	62.9	0.373
Hemi	-	N	20.3	0.001	154.6	0.263	5.8	0.002	67.2	0.373
Mal-Shamir	✓	Y	19.8	0.001	536.1	0.09	5.6	0.002	230.7	0.124
Sy-Shamir	✓	Y	6.3	0.036	227.1	2.893	2.5	0.037	151.8	4.413
Ps-Rep	✓	Y	12.0	0.001	601	0.117	4.7	0.001	296.8	0.171
Shamir	-	Y	20.0	0.001	586.6	0.005	5.6	0.001	293.2	0.149
Rep3	-	Y	21.8	0.001	775.9	0.055	5.6	0.001	333.0	0.084
Yao (2PC)	-	n/a	2.1	0.99	20.3	59.54	2.2	0.99	19.5	59.5
EMP (2PC)	-	n/a	-	-	-	-	0.8	0.002	-	-

randomly generated bits for bit-wise operations, the greatest impact is caused by the preprocessing of multiplication (Beaver [9]) triples.

The SPDZ family of protocols, i.e., MASCOT, Semi [32], Lowgear, Cowgear and Hemi [33], follow the online phase of the SPDZ protocol [17] which heavily relies on Beaver triples for performing multiplications, but offer improved offline phase performance. We focus on the Lowgear and Cowgear protocols because they perform better than MASCOT for large number of triples, yet we note their higher computation requirement. On the other hand, MASCOT relies more on communication and performs better in very low latency scenarios [33]. Keller et al. [33] measure over 100.000 triples/sec throughput for 3 parties and the same computation setting as in our experiments. Moreover, they estimate near 190 million triples per dollar and per party given the hourly cost of one hour AWS r4.16xlarge instance in Amazon's US East data center. *SELEST* requires a varying number of triples depending on the size of the input and the type of output. For reference, we examine the cost of the offline phase based on 4 samples of input per antenna (total of 32 samples) for both coarse and conditioned outputs (Table 2).

Summary of results. Our results show that we are able to achieve high performance DoA estimation with high accuracy in an MPC circuit, thus maintaining data privacy and security in the presence of different types of adversaries. For a coarse spectrum estimation, we achieve more than 100 DoA/s in the presence of a majority of compromised servers in the malicious model and more than 160 DoA/s in the semi-honest model. In the case of an honest majority of servers, our results boast more than 536 DoA/s in the malicious model and more than 775 DoA/s in the semi-honest model. The more demanding circuit producing conditional outputs also achieves more than 230 DoA/s in the malicious model and more than 330 DoA/s in the semi-honest model. We find Ps-Rep protocol balanced between high performance and security against one malicious adversary out of three parties, without the requirement of preprocessed multiplication triples. We also observe a speedup compared to non-private MUSIC (403.2 DoA/s), which reflects the lower computation of *SELEST* and the potential of parallel executions due to party interactions. However, *SELEST* processes a single emission and its performance is affected by network latency.

We note that ML-based detection techniques (Baset et al. [8]) are increasingly practical in isolating RF-emissions from millions of received samples, making our assumption of single emission reasonable. Furthermore, in a reasonably congested residential setting with hundreds of wireless devices, our results show the feasibility of a real time, secure, drone localization system.

5.2 Discussion

Constant round protocols. Being the first complex number implementation for a wireless application, it is natural that our implementation and evaluation focuses on arithmetic circuits. It is worth noting another growing field of MPC, *Constant Round* protocols which are based on Yao's GC protocol [62]. In a nutshell, such protocols usually operate in the binary domain and achieve constant rounds of communication by having one party (the garbler) create the circuit, send it to the other party (the evaluator) which evaluates it. This comes at the cost of a very large circuit that has to be created, transmitted and evaluated efficiently. We implemented and tested our algorithm in Yao's GC in MP-SPDZ [31] and EMP-Toolkit, both 2-party protocols for semi-honest adversaries with the EMP-Toolkit implementation easily extensible for malicious adversaries using Authenticated Garbling [57]. Figure 5d shows that 3 party arithmetic protocols outperformed constant round protocols for RTT up to 80ms. We observed that constant round protocols only take the lead for larger number of parties and higher RTT. We attribute this to the network volume caused by the garbled circuit which overshadows the few, low-latency communication rounds between a small number of parties in arithmetic evaluation: we observed 655 times more communication in Yao/EMP-Toolkit compared an equivalent MASCOT online phase. Moreover, a faster CPU would favor the garbled circuit approach, since it relies in computation for garbling and evaluating the circuit more than arithmetic protocols do.

Other applications. Our results are also promising for use in extended DoA estimation, such as higher dimension search and similar applications. Some applications aim for Acoustic Source Localization (ASL) by using microphone arrays to collect samples and pose similar security concerns [39].

ACKNOWLEDGMENTS

This work was partially supported by grants NAVY/N00014-20-1-2124, NCAE-Cyber Research Program, and NSF/DGE-1661532. The authors would like to thank the anonymous reviewers for their valuable feedback. Many thanks to Hai N. Nguyen for the provided raw data and helpful comments.

REFERENCES

[1] Chris Abbott, Matthew Clarke, Steve Hathorn, and Scott Hickie. 2016. Hostile Drones: The Hostile Use of Drones by Non-State Actors against British Targets. https://css.ethz.ch/en/services/digital-library/publications/publication.html/195685

[2] Elias Aboutanios, Aboulnasr Hassanien, Amr El-Keyi, Youssef Nasser, and Sergiy A. Vorobyov. 2017. Advances in DOA Estimation and Source Localization. *International Journal of Antennas and Propagation* (2017).

[3] EU Aviation Safety Agency. 2021. Civil drones regulations. https://www.easa.europa.eu/domains/civil-drones-rpas Retrieved 3-20-2021.

[4] aws.amazon.com. 2021. Amazon EC2 On-Demand Pricing. https://aws.amazon.com/ec2/pricing/on-demand/ Retrieved 3-20-2021.

[5] aws.amazon.com. 2021. Announcing improved networking performance for Amazon EC2 instances. https://aws.amazon.com/about-aws/whats-new/2017/09/announcing-improved-networking-performance-for-amazon-ec2-instances/ Retrieved 3-20-2021.

[6] Ettus Knowledge Base. 2016. GPSDO Selection Guide — Ettus Knowledge Base,. http://kb.ettus.com/GPSDO_Selection_Guide

[7] Ettus Knowledge Base. 2020. OctoClock CDA-2990 — Ettus Knowledge Base,. https://kb.ettus.com/OctoClock_CDA-2990 Retrieved 3-20-2021.

[8] Aniqua Baset, Christopher Becker, Kurt Derr, Samuel Ramirez, Sneha Kumar Kasera, and Aditya Bhaskara. 2019. Towards Wireless Environment Cognizance Through Incremental Learning. In *2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*.

[9] Donald Beaver. 1991. Efficient Multiparty Protocols Using Circuit Randomization. In *Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO '91)*.

[10] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. 1988. Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing (STOC '88)*.

[11] Lennart Braun, Daniel Demmler, Thomas Schneider, and Oleksandr Tkachenko. 2020. MOTION - A Framework for Mixed-Protocol Multi-Party Computation. Cryptology ePrint Archive, Report 2020/1137. https://eprint.iacr.org/2020/1137.

[12] Ran Canetti. 2000. Security and Composition of Multiparty Cryptographic Protocols. *J. Cryptol.* (2000).

[13] Koji Chida, Daniel Genkin, Koki Hamada, Dai Ikarashi, Ryo Kikuchi, Yehuda Lindell, and Ariel Nof. 2018. Fast Large-Scale Honest-Majority MPC for Malicious Adversaries. In *Advances in Cryptology – CRYPTO 2018*.

[14] Pei-Jung Chung, Mats Viberg, and Jia Yu. 2014. Chapter 14 - DOA Estimation Methods and Algorithms. In *Academic Press Library in Signal Processing: Volume 3*.

[15] M. P. Clark and L. L. Scharf. 1994. Two-dimensional modal analysis based on maximum likelihood. *IEEE Transactions on Signal Processing* (1994).

[16] Ivan Damgard, Marcel Keller, Enrique Larraia, Valerio Pastro, Peter Scholl, and Nigel P. Smart. 2012. Practical Covertly Secure MPC for Dishonest Majority – or: Breaking the SPDZ Limits. Cryptology ePrint Archive, Report 2012/642. https://eprint.iacr.org/2012/642.

[17] I. Damgard, V. Pastro, N.P. Smart, and S. Zakarias. 2011. Multiparty Computation from Somewhat Homomorphic Encryption. Cryptology ePrint Archive, Report 2011/535. https://eprint.iacr.org/2011/535.

[18] dji.com. 2021. DJI AEROSCOPE. https://www.dji.com/aeroscope Retrieved 3-20-2021.

[19] Daniel Escudero, Satrajit Ghosh, Marcel Keller, Rahul Rachuri, and Peter Scholl. 2020. Improved Primitives for MPC over Mixed Arithmetic-Binary Circuits. Cryptology ePrint Archive, Report 2020/338. https://eprint.iacr.org/2020/338.

[20] faa.gov. 2021. B4UFLY Mobile App. https://www.faa.gov/uas/recreational_fliers/where_can_i_fly/b4ufly/ Retrieved 3-20-2021.

[21] J. E. Fernandez del Rio and M. F. Catedra-Perez. 1997. The matrix pencil method for two-dimensional direction of arrival estimation employing an L-shaped array. *IEEE Transactions on Antennas and Propagation* (1997).

[22] Eran Fishler and H. Vincent Poor. 2005. Estimation of the number of sources in unbalanced arrays via information theoretic criteria. *IEEE Transactions on Signal Processing* (2005).

[23] foremtech.com. 2021. TrueView Radar. https://fortemtech.com/products/trueview-radar/ Retrieved 3-20-2021.

[24] J. G. F. Francis. 1961. The QR Transformation A Unitary Analogue to the LR Transformation—Part 1. *Comput. J.* (1961).

[25] Oded Goldreich. 2004. *Foundations of Cryptography*.

[26] Oded Goldreich, Silvio Micali, and Avi Wigderson. [n.d.]. A high-level approach to computer document formatting, In How to Play any Mental Game or a Completeness Theorem for Protocols with Honest Majority. *In Proceedings of the 19th Annual ACM Symposium on Theory of Computing*.

[27] Nicholas J. Higham. 1992. Stability of a Method for Multiplying Complex Matrices with Three Real Matrix Multiplications. *SIAM J. Matrix Anal. Appl.* (1992).

[28] Yingbo Hua and Tapan K.Sarka. [n.d.]. ([n. d.]).

[29] wsj.com Jack Nicas. 2019. Criminals, Terrorists Find Uses for Drones, Raising Concerns. https://www.wsj.com/articles/criminals-terrorists-find-uses-for-drones-raising-concerns-1422494268

[30] wsj.com Joseph De Avila. 2019. New York Police Seek Authority to Take Down Drones. https://www.wsj.com/articles/new-york-police-seek-authority-to-take-down-drones-11550419320

[31] Marcel Keller. 2020. MP-SPDZ: A Versatile Framework for Multi-Party Computation. Cryptology ePrint Archive, Report 2020/521. https://eprint.iacr.org/2020/521.

[32] Marcel Keller, Emmanuela Orsini, and Peter Scholl. 2016. MASCOT: Faster Malicious Arithmetic Secure Computation with Oblivious Transfer. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*. Association for Computing Machinery.

[33] Marcel Keller, Valerio Pastro, and Dragos Rotaru. 2017. Overdrive: Making SPDZ Great Again. Cryptology ePrint Archive, Report 2017/1230. https://eprint.iacr.org/2017/1230.

[34] B. Knott, S. Venkataraman, A.Y. Hannun, S. Sengupta, M. Ibrahim, and L.J.P. van der Maaten. 2020. CrypTen: Secure Multi-Party Computation Meets Machine Learning. In *Proceedings of the NeurIPS Workshop on Privacy-Preserving Machine Learning*.

[35] Donald E. Knuth. 2014. *Art of Computer Programming, Volume 2: Seminumerical Algorithms (3rd ed.)*.

[36] Markus Krueckemeier, Fabian Schwartau, Carsten Monka-Ewe, and Joerg Schoebel Technische. 2019. Synchronization of Multiple USRP SDRs for Coherent Receiver Applications. In *2019 Sixth International Conference on Software Defined Systems*.

[37] V.N. Kublanovskaya. 1962. On some algorithms for the solution of the complete eigenvalue problem. *U. S. S. R. Comput. Math. and Math. Phys.* (1962).

[38] Andrei Lapets, Frederick Jansen, Kinan Dak Albab, Rawane Issa, Lucy Qin, Mayank Varia, and Azer Bestavros. 2018. Accessible Privacy-Preserving Web-Based Data Analysis for Assessing and Addressing Economic Inequalities. In *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies (COMPASS '18)*.

[39] R. Levorato and E. Pagello. 2015. DOA Acoustic Source Localization in Mobile Robot Sensor Networks. In *2015 IEEE International Conference on Autonomous Robot Systems and Competitions*.

[40] Yehuda Lindell. 2020. Secure Multiparty Computation. *Commun. ACM* (2020).

[41] Eleftheria Makri, Dragos Rotaru, Frederik Vercauteren, and Sameer Wagh. 2021. Rabbit: Efficient Comparison for Secure Multi-Party Computation. Cryptology ePrint Archive, Report 2021/119. https://eprint.iacr.org/2021/119.

[42] BBC News. 2018. Gatwick Airport: Drones ground flights. https://www.bbc.com/news/uk-england-sussex-46623754

[43] Arogyaswami Paulraj, R. Roy, and Thomas Kailath. 1985. Estimation Of Signal Parameters Via Rotational Invariance Techniques- Esprit. In *Nineteeth Asilomar Conference on Circuits, Systems and Computers*.

[44] Raviraj S. Adve Pinyuen Chen, Michael C. Wicks. 2001. Development of a statistical procedure for detecting the number of signals in a radar measurement. *IEEE Proceedings - Radar, Sonar and Navigation* (2001).

[45] Irving S. Reed, J. D. Mallett, and Lawrence E. Brennan. 1974. Rapid convergence rate in adaptive arrays. *IEEETransAeroElec* AES-10 (1974).

[46] Peter Rindal. [n.d.]. The ABY3 Framework for Machine Learning and Database Operations. https://github.com/ladnir/aby3.

[47] Peter Rindal and Mike Rosulek. 2016. Faster Malicious 2-party Secure Computation with Online/Offline Dual Execution. Cryptology ePrint Archive, Report 2016/632. https://eprint.iacr.org/2016/632.

[48] Ralph Schmidt. 1986. Multiple emitter location and signal parameter estimation. (1986). https://ieeexplore.ieee.org/abstract/document/1143830

[49] sharemind.cyber.ee. 2021. Sharemind. https://sharemind.cyber.ee/sharemind-mpc/ Retrieved 3-20-2021.

[50] wsj.com Thomas Braun. 2020. Miniature Menace: The Threat of Weaponized Drone Use by Violent Non-state Actors. https://www.airuniversity.af.edu/Wild-Blue-Yonder/Article-Display/Article/2344151/miniature-menace-the-threat-of-weaponized-drone-use-by-violent-non-state-actors/

[51] rferl.org Tony Wesolowsky. 2017. Ukraine's Exploding Munition Depots Give Ammunition To Security Concerns. https://www.rferl.org/a/ukraine-exploding-munitions-security-concerns-russia/28777991.html

[52] US Federal Aviation Administration (FAA). 2021. UAS Remote Identification Overview. https://www.faa.gov/uas/getting_started/remote_id/ Retrieved 3-20-2021.

[53] Marinos Vomvas. 2021. Issue No.: 181, 177, 124, 126, 120, 81, 85, 86, 87. https://github.com/data61/MP-SPDZ/issues/.

[54] Marinos Vomvas. 2021. Source Code. https://github.com/Vomvas/Selest.git.

[55] Jun Wang, Feng gang Yan, Shuai Liu, and Xiao lin Qiao. 2015. Compressive sensing DOA estimation for active radar in forward-looking direction. *IET Conference Proceedings* (2015).

[56] Xiao Wang, Alex J. Malozemoff, and Jonathan Katz. 2016. EMP-toolkit: Efficient MultiParty computation toolkit. https://github.com/emp-toolkit.

[57] Xiao Wang, Samuel Ranellucci, and Jonathan Katz. 2017. Authenticated Garbling and Efficient Maliciously Secure Two-Party Computation. Cryptology ePrint Archive, Report 2017/030. https://eprint.iacr.org/2017/030.

[58] Mati Wax and Thomas Kailath. 1985. Detection of signals by information theoretic criteria. *IEEE Transactions on Acoustics, Speech, and Signal Processing* (1985).

[59] Emma White. 2021. Low-latency computing with AWS Local Zones. https://aws.amazon.com/blogs/compute/low-latency-computing-with-aws-local-zones-part-1/ Retrieved 3-20-2021.

[60] Lingyun X. Xiaofei Z., Jianfeng L. 2011. Novel two-dimensional DOA estimation with L-shaped array. *EURASIP J. Adv. Signal Process.* (2011).

[61] X.Zhang, X.Gao, and W.Chen. 2009. Improved Blind 2 D - Direction of Arrival Estimation with L - Shaped Array Using Shift Invariance Property. *Journal of Electromagnetic Waves and Applications* (2009).

[62] Andrew Chi-Chih Yao. 1986. How to Generate and Exchange Secrets. In *Proceedings of the 27th Annual Symposium on Foundations of Computer Science (SFCS '86)*.
[63] Samee Zahur and David Evans. 2015. Obliv-C: A Language for Extensible Data-Oblivious Computation. Cryptology ePrint Archive, Report 2015/1153. https://eprint.iacr.org/2015/1153.
[64] Michael D. Zoltowski, Martin Haardt, and Cherian P. Mathews. 1996. Closed-form 2-D angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT. *IEEE Transactions on Signal Processing* (1996).

A APPENDIX

THEOREM A.1. *In the case of a single incident waveform on an antenna array, DoA estimation using the covariance matrix is equivalent to DoA estimation using the noise subspace (MUSIC).*

$$P_s = \boldsymbol{\alpha}^*(\phi)\mathbf{E_s}\mathbf{E_s}^*\boldsymbol{\alpha}(\phi) \approx \boldsymbol{\alpha}^*(\phi)\mathbf{S_i}\mathbf{S_i}^*\boldsymbol{\alpha}(\phi) = P_{SELEST}$$

PROOF. When $\mathbf{P} = \mathbf{f}\mathbf{f}^* = \mathbb{E}\left[|f_1|^2\right]$ the signal covariance matrix can be written $\mathbf{S_s} = \mathbf{APA}^* = \mathbf{PAA}^*$. For simplicity we consider a *Uniform Linear Array* with the following steering vector, but the proof holds for arbitrary antenna arrays:

$$\begin{aligned}\boldsymbol{\alpha}(\phi) &= [1, e^{jkdcos\phi}, e^{jk2dcos\phi}, \cdots, e^{jk(N-1)dcos\phi}] \\ &= [1, z, z^2, \cdots, z^{N-1}]\end{aligned} \tag{4}$$

with $z = e^{jkdcos\phi}$ and $|z| = 1$. Then:

$$\mathbf{S} = \mathbf{S_s} + \sigma^2\mathbf{I} = \mathbf{PAA}^* + \sigma^2\mathbf{I}$$

$$= \mathbb{E}\left[|f_1|^2\right]\begin{bmatrix} 1+\tilde{\sigma} & \bar{z} & \bar{z}^2 & \cdots & \bar{z}^{N-1} \\ z & z\bar{z}+\tilde{\sigma} & z\bar{z}^2 & \cdots & z\bar{z}^{N-1} \\ z^2 & z^2\bar{z} & z^2\bar{z}^2+\tilde{\sigma} & \cdots & z^2\bar{z}^{N-1} \\ \vdots & \cdots & \cdots & \ddots & \vdots \\ z^{N-1} & z^{N-1}\bar{z} & z^{N-1}\bar{z}^2 & \cdots & z^{N-1}\bar{z}^{N-1}+\tilde{\sigma} \end{bmatrix}$$

where $\tilde{\sigma} = \frac{\sigma^2}{\mathbb{E}[|f_1|^2]}$.

We can now express a column vector of S as

$$\mathbf{s}_i = [\bar{z}^i, z\bar{z}^i, \cdots, z^i\bar{z}^i + \tilde{\sigma}, \cdots, z^{N-1}\bar{z}^i]^T$$

and have the dot product of two arbitrary column vectors of S be

$$\langle \mathbf{s}_i \cdot \mathbf{s}_j \rangle = \sum_{k=0}^{N-1} \bar{\mathbf{s}}_{ik}\mathbf{s}_{jk} = z^i\bar{z}^j(N + 2\tilde{\sigma}) \tag{5}$$

For the norm of a column vector of S, we have $\|\mathbf{s}_i\| = \sqrt{\sum_{k=0}^{N-1}|s_{ik}|^2} = \sqrt{(N+\tilde{\sigma})}$, and therefore $\|\mathbf{s}_i\| \cdot \|\mathbf{s}_j\| = (N+\tilde{\sigma})$. Then by the vector dot product:

$$\cos\theta = \frac{\mathrm{Re}(\langle \mathbf{s}_i \cdot \mathbf{s}_j \rangle)}{\|\mathbf{s}_i\| \cdot \|\mathbf{s}_j\|} = \mathrm{Re}(z^{|i-j|})(1 + \frac{\tilde{\sigma}}{N+\tilde{\sigma}}) \leq \mathrm{Re}(z^{|i-j|})(1+\tilde{\sigma})$$

However, $\tilde{\sigma} = \frac{\sigma^2}{\mathbb{E}[|f_1|^2]}$ and the noise variance can be safely assumed orders of magnitude less than the expected norm of the incident signal. This shows that choosing two arbitrary vectors from S is equivalent to choosing two arbitrary vectors from $\mathbf{S_s}$. □