Secure Fast Covariance Intersection Using Partially Homomorphic and Order Revealing Encryption Schemes

Marko Ristic, Benjamin Noack, and Uwe D. Hanebeck

***Abstract*— Fast covariance intersection is a widespread technique for state estimate fusion in sensor networks when cross-correlations are not known and fast computations are desired. The common requirement of sending estimates from one party to another during fusion means they do not remain private to their producing party. Current secure fusion algorithms have a reliance on encryption schemes that do not provide sufficient flexibility and as a result require, often undesired, excess communication between estimate producers. We propose a novel method of homomorphically computing the fast covariance intersection algorithm on estimates encrypted with a combination of encryption schemes. Using order revealing encryption we show how the approximate solutions to the fast covariance intersection coefficients can be computed and combined with partially homomorphic encryptions of estimates, to compute an encryption of the fused result. The described approach allows the secure fusion of any number of private estimates, making third-party cloud processing a viable option when working with sensitive state estimates, or when performing estimation over insecure networks.**

I. Introduction

Sensor data processing and state estimation have been increasingly prevalent in networked systems [1], [2]. Bayesian state estimation has become a particularly common application since the beginning of Kalman estimation theory [3] and has led to a large interest in the field of state estimation fusion [4]–[8]. Challenges of estimation fusion are closely tied to the handling and merging of estimation error statistics [9]. Cross-correlation between estimation errors characterise dependencies between local estimates and must be considered when performing consistent or optimal fusion [10], [11]. Methods that keep track of the cross-correlation of errors omit the need for repeated reconstruction [12] however typically add local computational complexity and limit usability. An alternative strategy sees the approximation of estimate error cross-correlation based on conservative suboptimal strategies, and has been implemented in a variety of methods [13]–[18]. Covariance Intersection (CI) [14] provides one such popular conservative strategy, from which a less computationally expensive method, the Fast Covariance Intersection (FCI) [17] has been derived. CI is particularly well paired with the information form of the Kalman filter [19]. This algebraically equivalent form of the standard Kalman filter requires the persistent storing of the information matrix and vector instead of the usual state estimate and covariance and reduces fusion operations to simple summations. It has been used to subtract common information between estimates when cross-correlations are known [8] and within fully distributed filter implementations [20].

A key step in distributed sensor fusion, and our topic of interest in this paper, is the requirement of transmitting sensor state estimate information between network nodes for the computation of a final fused result. Network eavesdroppers or curious fusion nodes are not prevented from learning possibly sensitive local state estimates and errors. Encryption has until recently been primarily used to secure information transfer between communicating parties, with common symmetric-key encryption schemes such as AES [21] being used to encrypt sent information to its destination, and public-key encryption schemes such as RSA [22] to distribute symmetric keys. However, recent developments in public-key homomorphic encryption (HE) schemes [23]–[25], which allow algebraic operations to be performed on encryptions, are leading to more secure cloud or network applications for signal processing [26]–[28]. Although implementations of Fully Homomorphic Encryption (FHE) schemes exist [29], and provide all algebraic operations on encryptions, current implementations are still computationally infeasible for large-scale signal processing [30], [31]. Instead, Partially Homomorphic Encryption (PHE) schemes [24], [25], providing typically only one algebraic operation, have been a focus for such processing tasks [27], [28]. However, due to the limited operations provided by PHE (most commonly addition provided by the Paillier encryption scheme due to its speed and simplicity, [25]), securely computable processing algorithms have thus far been relatively restricted in complexity and application. The recent development of new encryption schemes, such as Order Revealing Encryption (ORE) [32]–[34], have provided new light on the possible complexity of signal processing algorithms that can be computed securely. Thus far, ORE has found little application in the context of signal processing algorithms or in combination with HE schemes. In this paper, we make use of a combination of ORE and PHE schemes to develop a Secure FCI (SecFCI) fusion approximation method that enables us to protect sensor estimates from both eavesdroppers and other algorithm participants.

In section II we will introduce CI and FCI methods relevant to our proposed fusion algorithm, and in section III, the relevant encryption schemes. Sections IV and V will introduce the secure FCI algorithm for the 2 sensor and multi-sensor cases respectively, and VI discusses simulation results and comparisons to the ordinary FCI fusion algorithm.

Marko Ristic, Benjamin Noack, and Uwe D. Hanebeck are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT), Germany.
{marko.ristic,noack,uwe.hanebeck}@kit.edu

We conclude our findings and plans for future work in VII.

A. Notation

Throughout this paper we will use the following notation. Lowercase characters represent scalars, lowercase underlined characters, $\underline{x}$, represent vectors. Uppercase bold characters, $\mathbf{M}$, are reserved for matrices, where $\mathbf{M}^\top$ denotes the matrix transpose, $\mathbf{M}^{-1}$ the matrix inverse, and $\mathrm{tr}(\cdot)$ the trace function. Covariance matrices will be represented by the matrix $\mathbf{P}$. $\mathcal{E}_{pk}(a)$ and $\mathcal{E}_{ORE,k}(a)$ denote the public-key pk and ORE key k encryptions of a, and similarly with the decryption functions $\mathcal{D}_{pk}(\cdot)$ and $\mathcal{D}_{ORE,k}(\cdot)$, where any required real-number encodings of the number a are assumed to be performed. $\mathcal{E}(a)$ and $\mathcal{E}_{ORE}(a)$ may be used for brevity when the encryption keys can be inferred from context. Encryption of vectors and matrices are defined element-wise, with elements given by $\mathcal{E}(\mathbf{P}_{i,j}) = \mathcal{E}(\mathbf{P})_{i,j}$.

II. Covariance Intersection and Approximations

Covariance Intersection (CI) introduced in [14] provides a consistent state estimation fusion algorithm when model cross-correlations are not known. The resulting fused estimate $\hat{\underline{x}}$ and estimate covariance $\mathbf{P}$ can be easily derived from (1) and (2). Note that (1) and (2) compute the fusion of the information matrix and vectors as defined in [17] and reduce the fusion to a simple weighted sum.

$$\mathbf{P}^{-1} = \sum_{i=0}^{n} \omega_i \mathbf{P}_i^{-1} \tag{1}$$

$$\mathbf{P}^{-1}\hat{\underline{x}} = \sum_{i=0}^{n} \omega_i \mathbf{P}_i^{-1} \hat{\underline{x}}_i \tag{2}$$

Values for ω_i must satisfy (3) and (4), and guarantee consistency of the fused estimates. They are chosen in a way to speed up convergence, by minimising a property of the resulting fused estimate covariance.

$$\omega_0 + \omega_1 + \cdots + \omega_n = 1 \tag{3}$$

$$0 \leq \omega_i \leq 1 \tag{4}$$

One such property which may be minimised to guarantee faster convergence is the fused estimate covariance trace, requiring the solution to (5).

$$\underset{\omega_0,\ldots,\omega_n}{\arg\min}\{\mathrm{tr}(\mathbf{P})\} = \underset{\omega_0,\ldots,\omega_n}{\arg\min}\{\mathrm{tr}((\sum_{i=0}^{n} \omega_i \mathbf{P}_i)^{-1})\} \tag{5}$$

However, minimising this non-linear cost function can be very costly computationally and has led to the development of the non-iterative approximation technique in [17].

A. Fast Covariance intersection

The Fast Covariance Intersection (FCI) algorithm from [17] is a common method used for approximating the solution to (5) without the loss of guaranteed consistency. It is computed by defining a new constraint (6) on ω_i and solving the resulting equations instead.

$$\omega_i \,\mathrm{tr}(\mathbf{P}_i) - \omega_j \,\mathrm{tr}(\mathbf{P}_j) = 0, \;\; (i,j = 1, 2, \ldots, n) \tag{6}$$

In the two sensor case, this results in the solving of (7) and (8).

$$\omega_0 + \omega_1 = 1 \tag{7}$$

$$\omega_0 \,\mathrm{tr}(\mathbf{P}_0) - \omega_1 \,\mathrm{tr}(\mathbf{P}_1) = 0 \tag{8}$$

With analytical solutions given by (9).

$$\omega_0 = \frac{\mathrm{tr}(\mathbf{P}_1)}{\mathrm{tr}(\mathbf{P}_0) + \mathrm{tr}(\mathbf{P}_1)}, \; \omega_1 = \frac{\mathrm{tr}(\mathbf{P}_0)}{\mathrm{tr}(\mathbf{P}_0) + \mathrm{tr}(\mathbf{P}_1)} \tag{9}$$

When computed for the n sensor case, the highly redundant (6) can have it largest linearly independent subset represented by (10), and requires the solution to the linear problem (11), where we let $\mathcal{P}_i = \mathrm{tr}(\mathbf{P}_i)$.

$$\omega_i \,\mathrm{tr}(\mathbf{P}_i) - \omega_{i+1} \,\mathrm{tr}(\mathbf{P}_{i+1}) = 0, \;\; (i = 1, 2, \ldots, n) \tag{10}$$

$$\begin{bmatrix} \mathcal{P}_0 & -\mathcal{P}_1 & 0 & \cdots & 0 \\ 0 & \mathcal{P}_1 & -\mathcal{P}_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \mathcal{P}_{n-1} & -\mathcal{P}_n \\ 1 & \cdots & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \omega_0 \\ \omega_0 \\ \vdots \\ \omega_{n-1} \\ \omega_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{11}$$

Our proposed filter aims to solve FCI fusion, namly (1), (2) and (11), homomorhically, such that using only encryptions from each sensor i we are able to produce valid encryptions of fused estimates without the need for decryption.

III. Homomorphic and Order Revealing Encryption

To achieve a secure solution to the FCI fusion problem, we have made use of two types of function-providing encryption schemes. Public-key additive Partially Homomorphic Encryption schemes [25], [35] provide a single homomorphic addition operation on cyphertexts such that (12) holds.

$$\mathcal{E}(a) \oplus \mathcal{E}(b) = \mathcal{E}(a + b) \tag{12}$$

While symettric-key Order Revealing Encryption schemes [32], [33] provide a secure comparison function, allowing the comparison of encrypted values via (13).

$$f(\mathcal{E}_{ORE}(a), \mathcal{E}_{ORE}(b)) = cmp(a, b) \tag{13}$$

The formal security of encryption schemes consists of a security goal and a formal threat model [36]. Indistinguishability of ciphertexts under the adaptive chosen ciphertext attack model (IND-CCA2) is the commonly considered strongest security guarantee [37], however, cannot be satisfied by any homomorphic encryption scheme due to their apparent ability to create valid cyphertexts via homomorphic operations. Instead, PHE schemes aim to protect against the weaker assumption of the chosen plaintext attack model (IND-CPA) [38]. Similarly, ORE schemes aim to protect against the ordered chosen-plaintext attack (IND-OCPA) or weaker simulation-based security defined in [32].

A. Additive Partially Homomorphic Encryption

The additive PHE scheme we have used is the Paillier encryption scheme [25] due to its implementation simplicity, and computational speed. The Paillier scheme provides two homomorphic operations on encrypted data, (14) and (15), where the modulus N is computed as the product of 2 large primes chosen randomly during key generation. The public and secret keys are shown as pk and sk respectively, and plaintext messages $a, b \in \mathbb{Z}_N$.

$$\mathcal{E}_{pk}(a)\mathcal{E}_{pk}(b) \pmod{N^2} = \mathcal{E}_{pk}(a+b \pmod{N}) \quad (14)$$

$$\mathcal{E}_{pk}(a)^c \pmod{N^2} = \mathcal{E}_{pk}(ca \pmod{N}),\ c \in \mathbb{Z}_N^* \quad (15)$$

The Paillier encryption scheme successfully protects against the IND-CPA security goal and attacker model.

B. Left-Right Order Revealing Encryption

The ORE scheme we have used is Lewi's symmetric-key Left-Right encryption scheme [33] which has the added property of only allowing certain comparisons between cyphertexts. This property can be used to decide which values may not be compared as will be shown in section IV, and is described as follows. Two encryption functions allow integers to be encrypted as either a "Left" (L) or "Right" (R) encryption (16), and comparisons are only possible between an L and an R encryption (17).

$$\begin{aligned} enc^L_{ORE}(k,x) &= \mathcal{E}^L_{ORE,k}(x) \\ enc^R_{ORE}(k,y) &= \mathcal{E}^R_{ORE,k}(y) \end{aligned} \quad (16)$$

$$cmp_{ORE}(\mathcal{E}^L_{ORE}(x),\ \mathcal{E}^R_{ORE}(y)) = cmp(x,y) \quad (17)$$

Note that no decryption function is provided, as encryptions are only required to provide a means of secure comparison. The Lewi ORE encryption scheme does not satisfy IND-OCPA security due to its apparent difficulty, and instead satisfies the simulation-based notion of ORE security [32], [33].

C. Real Number Encoding for Homomorphic Encryption

Both encryption schemes in sections III-A and III-B are defined only over positive integers, while the Paillier scheme further gives an upper bound N to the size of an encryptable integer. Due to the prevalence of real numbers in estimation theory, typically stored as floating-point numbers in modern-day hardware, an integer encoding of real number values is required for their encryption. This requires the handling of both fractional, and negative numbers.

Negative numbers can be handled using the common two's complement method of representing negative integers [39]. This is done by splitting the total range of allowable integers $[0, N)$ in half, and letting the upper half $[\frac{N}{2}, N)$ represent negative integers. From this, we can see that the value of the largest encryptable integer is now given by $N/2-1$ and that the addition of two's complement numbers is automatically preserved due to modulo arithmetic.

The handling of fractional numbers proves to be a more complicated matter, due to the homomorphic multiplication property of the Paillier encryption scheme (15). Fractional numbers are represented as integers using the quantisational Q number format [40]. The encoding of a real number a, with maximum integer bits i and fractional bits f is represented by an $i+f$ bit long integer e, such that the maximum encoding is given by $2^{(i+f)}-1$. Encoding is performed by (18).

$$e = \lfloor 2^f a \rceil \quad (18)$$

While encoded Q numbers are consistent under addition, multiplication requires a factor of $1/2^f$ to be removed. As homomorphic division is not supported by the Paillier encryption scheme, the number of multiplications performed on encrypted values must be bounded and handled when decoding. As will be shown in section IV, our fusion method will always require that a single multiplication factor be removed and leads to the decoding of an integer e to a real number a being given by (19).

$$a = \frac{e}{2^{2f}} \quad (19)$$

While the largest encryptable integer is given by $N/2-1$, the largest encodable real number must account for the additional multiplication factor and must have i and f chosen such that (20) holds.

$$(2^{(i+f)}-1)^2 \le \frac{N}{2} - 1 \quad (20)$$

IV. Secure Fast Covariance Intersection with 2 Sensors

- First we will consider the 2 sensor case of secure FCI (SecFCI), before extending it to any number of sensors in section V. We consider sensor 0 which produces the estimate $\underline{\hat{x}}_0$ and covariance $\mathbf{P}_0$, and similarly sensor 1 producing $\underline{\hat{x}}_1$ and $\mathbf{P}_1$.
- From (1) and (2) we can see that CI is particularly suited to PHE schemes. Sensor encrypted estimates and covariances can be combined additively using (14) and (15). The equations using the Paillier encrtyption systems to compute CI are given in (21) and (22).

$$\mathcal{E}(\mathbf{P}) = \mathcal{E}(\mathbf{P}_0)^{\omega_0}\mathcal{E}(\mathbf{P}_1)^{(1-\omega_0)} \quad (21)$$

$$\mathcal{E}(\mathbf{P}\underline{\hat{x}}) = \mathcal{E}(\mathbf{P}_0\underline{\hat{x}}_0)^{\omega_0}\mathcal{E}(\mathbf{P}_1\underline{\hat{x}}_1)^{(1-\omega_0)} \quad (22)$$

$$(23)$$

- What remains is the computation of the parameter ω. The FCI solutions for the 2 sensor case given by (9), cannot be computed with PHE encryptions due to the required division.
- Instead we discretise ω to steps of some size $s < 1$ such that s divides 1, and use the ORE scheme to compute (8). Each sensor discretises ω and multiplies each discretisation with $\mathrm{tr}(\mathbf{P}^{-1})$ of its current coveriance $\mathbf{P}$. Each result is then encrypted with the Left-Right ORE scheme.
- Sensor 0 encrypts using the Left scheme as shown in (24) and similarly for sensor 1 using Right encryption in (25).

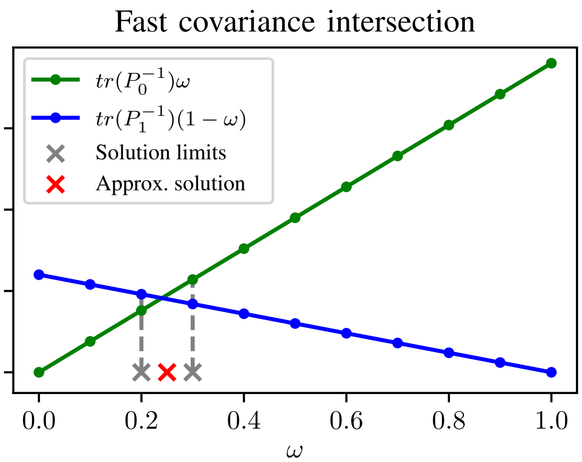

Fig. 1. Approximation of ω_0 with discretisation step size $s = 0.1$. Only comparisons of the ordered values sent from either estimator are used.

$$[\mathcal{E}^L_{ORE}(\omega\,\mathrm{tr}(\mathbf{P}_0^{-1})),\ \omega \in [0, 0+s, \ldots, 1-s, 1]] \tag{24}$$
$$[\mathcal{E}^R_{ORE}(\omega\,\mathrm{tr}(\mathbf{P}_1^{-1})),\ \omega \in [0, 0+s, \ldots, 1-s, 1]] \tag{25}$$

- The two ordered encrypted lists (24) and (25) are recieved at the fusion center, and used to estimate ω.
- To compute the FCI value for ω we want the intersection between the two lines described by $\mathrm{tr}(\mathbf{P}_0)\omega$ and $\mathrm{tr}(\mathbf{P}_1)(1-\omega)$. Note that in the 2 sensor case, the discretised list of $\mathrm{tr}(\mathbf{P}_i)(1-\omega)$ can be obtained by simply reversing the list for $\mathrm{tr}(\mathbf{P}_i)\omega$.
- Fig. 1 shows the list from sensor 0 and the reversed list of sensor 1 plotted over ω with a step size $s = 0.1$. Since values from one list can be compared with those from the other using the ORE operation (17), the exact intersection can be approximated in $O(log(\frac{1}{s}))$ steps by performing a binary search.

- Once the two consecutive differing comparisons from vertically aligned points in 1 are found, the FCI ω can be approximated by taking the middle value between the two bounds. This is computed simply with (26).
- In the case where the comparison function from the ORE scheme returns exact equality, the exact value of ω is known and can be taken as the approximation.

$$\omega_0' = \frac{1}{2}(a+b),\ \omega_1' = (1-\omega_0) \tag{26}$$

V. Multi-sensor Secure Fast Covariance Intersection

- In the multi-sensor case, the same method of using PHE encryptions of each sensor's estimate and covariance is used to compute (1) and (2). Again we are left with the task of computing the weights ω_i.
- For the ease of diagrams, we will demonstrate how this can be done in the three sensor case, and provide equations for the n sensor case.

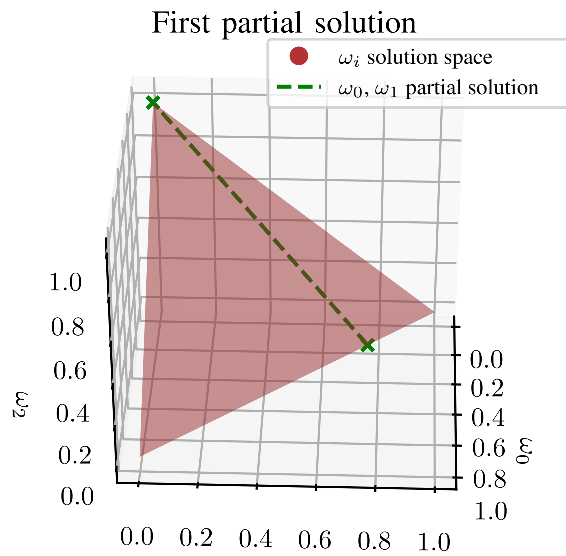

Fig. 2. Partial solution from equation (27) plotted on the plane of all possible values of ω_0, ω_1, and ω_2.

- Computing the ω_i values for FCI in the three sensor case requires the solving of (27) and (28).

$$\omega_0\,\mathrm{tr}(\mathbf{P}_0) - \omega_1\,\mathrm{tr}(\mathbf{P}_1) = 0 \tag{27}$$
$$\omega_1\,\mathrm{tr}(\mathbf{P}_1) - \omega_2\,\mathrm{tr}(\mathbf{P}_2) = 0 \tag{28}$$

- Our method solves the multiple equations by approximating the partial solutions of each, similarly to the two sensor case, and solving the newly obtained, unencrypted, multiple equations.
- Partial solutions are treated as hyperplanes on the possible solutions space, and their intersection gives the solution to the multiple equations. We consider each ω_i a dimension, and defined hyperplane points as $(\omega_0, \omega_1, \ldots, \omega_n)$ accordingly.
- Solving (27) as was done in the two sensor case with (24) and (25), gives the partial solution of (27) when $\omega_2 = 0$ in the three sensor case. The other defining points of the partial solution of ω_0 and ω_1 are obtained when each remaining $\omega_i = 1$. In the three sensor case, this gives one aditional point; (0,0,1). In Fig. 2 the partial solution of (27) has been plotted over the solution space defined by (3) and (4) when $n = 2$.

- Next we compute the partial solution for ω_1, and ω_2 in the same way. The additional known point defining this partial solution space is $(1, 0, 0)$. Fig 3 shows both partial solutions plotted over the solution space. Note that the intersection of the partial solutions gives the approximate solution for the FCI ω values.

- The partial solutions provide $n-1$ hyperplanes, each of dimension $n-2$. When computing the partial solutions' intersection, we define hyperplanes of dimension $n-1$ for each partial solution, by adding a dimension

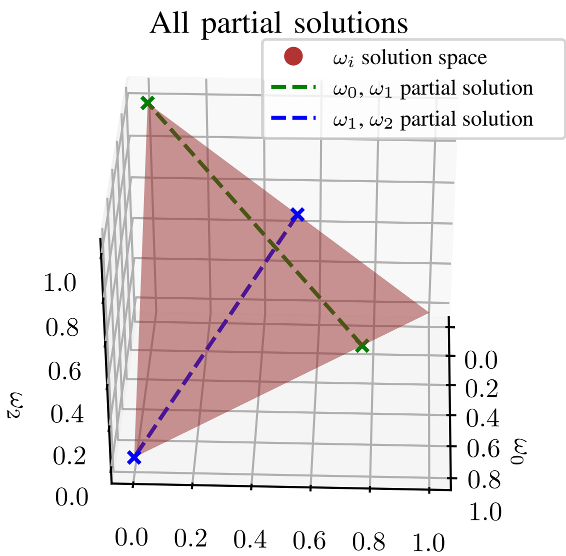

Fig. 3. Partial solutions from equations (27) and (28) plotted on the plane of all possible values of ω_0, ω_1, and ω_2.

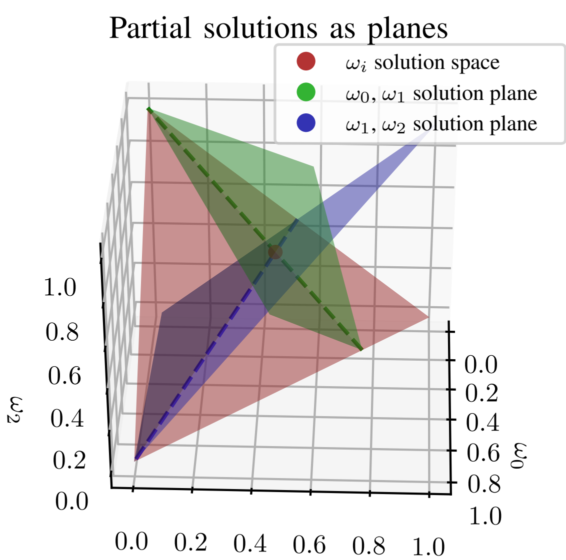

Fig. 4. Partial solutions from fig. 3 plotted as planes perpendicular to the plane of possible solutions. Intersection point gives solution values of ω_i for Fast Covariance Intersection.

perpendicular to the solution space. This can be seen in Fig. 4.

- Each partial solution in the three sensor case is therefore now defined as a plane of the form (29), where x, y, z represent the $\omega_0, \omega_1, \omega_2$ axes.

$$a_0 x + a_1 y + a_2 z + d = 0 \tag{29}$$

- We now have n equations, one of which is the solution space equation, and n values of $'omega_i$ to solve for. In the three sensor case, the solution to the linear problem (30) provides the approximate solution to each ω_i value in FCI.

$$\begin{bmatrix} a_0^0 & a_1^0 & a_2^0 \\ a_0^1 & a_1^1 & a_2^1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \omega_0 \\ \omega_1 \\ \omega_2 \end{bmatrix} = \begin{bmatrix} d^0 \\ d^1 \\ 1 \end{bmatrix} \tag{30}$$

- In the case of n sensors, hyperplanes are defined similarly and produce the linear equation (31).

$$\begin{bmatrix} a_0^0 & a_1^0 & \cdots & a_n^0 \\ a_0^1 & a_1^1 & \cdots & a_n^1 \\ \vdots & \vdots & \ddots & \vdots \\ a_0^n & a_1^n & \cdots & a_n^{n-1} \\ 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \omega_0 \\ \omega_1 \\ \vdots \\ \omega_{n-1} \\ \omega_n \end{bmatrix} = \begin{bmatrix} d^0 \\ d^1 \\ \vdots \\ d^{n-1} \\ 1 \end{bmatrix} \tag{31}$$

- Each of the partial solutions is approximated using consecutive sensor lists encrypted with ORE. This allows "Left-Right" ORE to still be used, by alternating which sensor uses which encryption.
- The required ordered lists sent from each sensor i are described by (32).

$$\begin{aligned} &[\mathcal{E}_{ORE}^{L}(\omega \operatorname{tr}(\mathbf{P}_i^{-1})),\ \omega \in [0, 0+s, \ldots, 1-s, 1]],\ i \text{ is even} \\ &[\mathcal{E}_{ORE}^{R}(\omega \operatorname{tr}(\mathbf{P}_i^{-1})),\ \omega \in [0, 0+s, \ldots, 1-s, 1]],\ i \text{ is odd} \end{aligned} \tag{32}$$

VI. Simulation Results

- A simulation was implemented to demonstrate the accuracy of SecFCI fusion when compared to traditional FCI fusion.
- A constant-speed linear process model was used, with two independent cartesian sensors making white Gaussian noisy measurements of the ground truth. Both sensors ran linear Kalman Filters [] on their measurements producing a local estimate and estimate covariance.
- Sensors provided their estimates to a fusion center both in unencrypted form, and encrypted form. Unencrypted estimates consisted of only the estimates $\hat{\underline{x}}_i$ and $\mathbf{P}_i$, while encrypted estimates were composed of PHE encryption of the estimate in the information filter form, $\mathcal{E}(\mathbf{P}_i^{-1})$ and $\mathcal{E}(\mathbf{P}_i^{-1}\hat{\underline{x}}_i)$, and the ordered ORE encryptions defined in (32).
- The fusion center then performs FCI fusion on the unencrypted estimates and our SecFCI fusion on the encrypted information. A portion of the trajectory and fused estimates are shown in Fig. 5.
- Fig. 6 plots the resulting fused estimate covariances' traces over the course of the simulation. Approximation of exact FCI fusion results in slightly differing covariance traces as expected. In our simulation, the resulting covariance trace of the SecFCI fusion algorithm is consistently higher than that of FCI fusion. However, this is not always the case, due to FCI itself computing an approximation to the trace minimising choice of ω.
- In Fig. 7 FCI and SecFCI values for each ω_i are plotted throughout a portion of the trajectory. SecFCI values for ω_i stay constant over time as can be expected due to the discretisation of partial solution estimates.

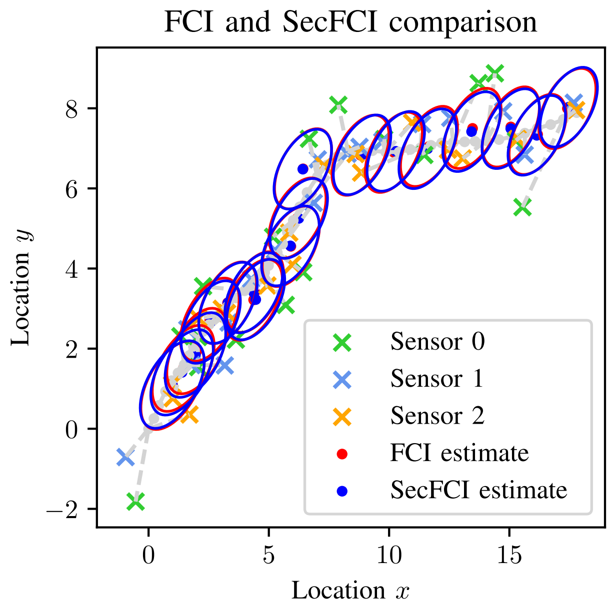

Fig. 5. Tracking simulation comparing Fast Covariance Intersection and our Secure Fast Covariance Intersection fusion methods.

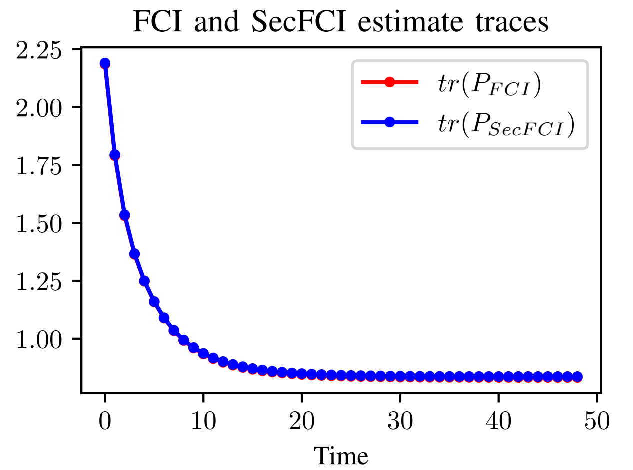

Fig. 6. Plot showing the fused estimate covariance trace throughout a tracking simulation, for both Fast Covariance Intersection and our Secure Fast Covariance Intersection

VII. Conclusion

- Fast Covariance Intersection commonly used method for the approximation of the non-linear optimisation problem of Covariance Intersection fusion but requires the sharing of local sensor estimates with each other or a centralised fusion server. We have proposed an approximation to the FCI which can be computed given only encrypted estimate and estimate covariance information from each sensor.
- Uses for a secure fusion algorithm can be found in various security-critical applications, or with untrusted networks and fusion centers.
- In future work, we would like to assess the estimate data leakage implications of ORE encrypted covariance information and produce security assumptions and proofs for the signal processing technique.
- We are also interested in how the computational performance of SecFCi may compare with alternative FHE and real number encoding techniques.

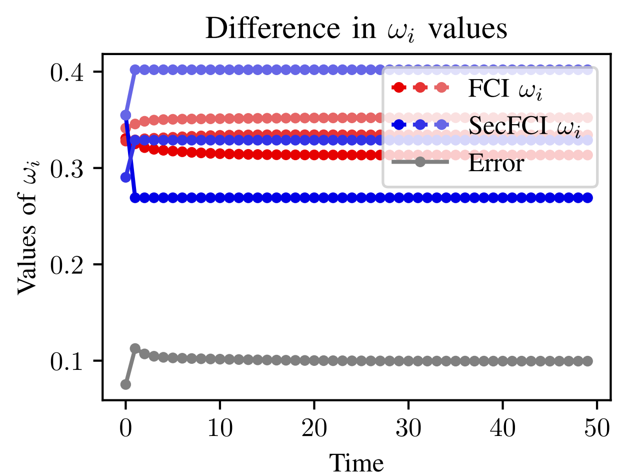

Fig. 7. Plot showing the difference in ω_i values between Fast Covariance Intersection and our Secure Fast Covariance Intersection, throughout a tracking simulation.

References

[1] D. Hall, C.-Y. Chong, J. Llinas, and M. Liggins, *Distributed Data Fusion for Network-Centric Operations*, 1st ed. USA: CRC Press, Inc., 2012.
[2] C.-Y. Chong, "Forty years of distributed estimation: A review of noteworthy developments," in *2017 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, Oct. 2017, pp. 1–10.
[3] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, Mar. 1960.
[4] D. Willner, C.-B. Chang, and K.-P. Dunn, "Kalman filter algorithms for a multi-sensor system," Jan. 1977, pp. 570–574.
[5] C. Y. Chong, "HIERARCHICAL ESTIMATION," p. 17.
[6] C.-Y. Chong, K.-C. Chang, and S. Mori, "Distributed Tracking in Distributed Sensor Networks," in *1986 American Control Conference*, Jun. 1986, pp. 1863–1868.
[7] H. Hashemipour, S. Roy, and A. Laub, "Decentralized structures for parallel Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 33, no. 1, pp. 88–94, Jan. 1988, conference Name: IEEE Transactions on Automatic Control.
[8] S. Grime and H. F. Durrant-Whyte, "Data fusion in decentralized sensor networks," *Control Engineering Practice*, vol. 2, no. 5, pp. 849–863, Oct. 1994.
[9] H. Fourati, *Multisensor Data Fusion : From Algorithms and Architectural Design to Applications*. CRC Press, Dec. 2017.
[10] Y. Bar-Shalom, "On the track-to-track correlation problem," *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 571–572, Apr. 1981, conference Name: IEEE Transactions on Automatic Control.
[11] S.-L. Sun and Z.-L. Deng, "Multi-sensor optimal information fusion Kalman filter," *Automatica*, vol. 40, no. 6, pp. 1017–1023, Jun. 2004.
[12] J. Steinbring, B. Noack, M. Reinhardt, and U. D. Hanebeck, "Optimal sample-based fusion for distributed state estimation," in *2016 19th*

International Conference on Information Fusion (FUSION), Jul. 2016, pp. 1600–1607.

[13] N. Carlson, "Federated filter for fault-tolerant integrated navigation systems," in *IEEE PLANS '88.,Position Location and Navigation Symposium, Record. 'Navigation into the 21st Century'.*, Nov. 1988, pp. 110–119.

[14] S. J. Julier, "A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations," p. 5.

[15] J. Sijs, M. Lazar, and P. Bosch, "State fusion with unknown correlation: Ellipsoidal intersection," in *Proceedings of the 2010 American Control Conference*, Jun. 2010, pp. 3992–3997.

[16] B. Noack, J. Sijs, and U. D. Hanebeck, "Inverse covariance intersection: New insights and properties," in *2017 20th International Conference on Information Fusion (Fusion)*. Xi'an, China: IEEE, Jul. 2017, pp. 1–8.

[17] W. Niehsen, "Information fusion based on fast covariance intersection filtering," in *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997)*, vol. 2, Jul. 2002, pp. 901–904 vol.2.

[18] D. Franken and A. Hupper, "Improved fast covariance intersection for distributed data fusion," in *2005 7th International Conference on Information Fusion*, vol. 1, Jul. 2005, pp. 7 pp.–.

[19] A. G. O. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. Routledge, May 2019.

[20] F. Pfaff, B. Noack, U. D. Hanebeck, F. Govaers, and W. Koch, "Information form distributed Kalman filtering (IDKF) with explicit inputs," in *2017 20th International Conference on Information Fusion (Fusion)*. Xi'an, China: IEEE, Jul. 2017, pp. 1–8.

[21] "Announcing the Advanced Encryption Standard (AES)," NATIONAL INST OF STANDARDS AND TECHNOLOGY GAITHERSBURG MD, Tech. Rep., Nov. 2001.

[22] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[23] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 41st Annual ACM Symposium on Symposium on Theory of Computing - STOC '09*. Bethesda, MD, USA: ACM Press, 2009, p. 169.

[24] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," in *Advances in Cryptology*, ser. Lecture Notes in Computer Science, G. R. Blakley and D. Chaum, Eds. Berlin, Heidelberg: Springer, 1985, pp. 10–18.

[25] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," in *Advances in Cryptology — EUROCRYPT '99*, ser. Lecture Notes in Computer Science, J. Stern, Ed. Springer Berlin Heidelberg, 1999, pp. 223–238.

[26] R. L. Lagendijk and M. Barni, "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 82–105, Jan. 2013.

[27] A. B. Alexandru, M. S. Darup, and G. J. Pappas, "Encrypted Cooperative Control Revisited," p. 7.

[28] M. Aristov, B. Noack, U. D. Hanebeck, and J. Muller-Quade, "Encrypted Multisensor Information Filtering," in *2018 21st International Conference on Information Fusion (FUSION)*. Cambridge, United Kingdom: IEEE, Jul. 2018, pp. 1631–1637.

[29] C. Gentry and S. Halevi, "Implementing Gentry's Fully-Homomorphic Encryption Scheme," in *Advances in Cryptology – EUROCRYPT 2011*, ser. Lecture Notes in Computer Science, K. G. Paterson, Ed. Springer Berlin Heidelberg, 2011, pp. 129–148.

[30] Y. Du, L. Gustafson, D. Huang, and K. Peterson, "Implementing ML Algorithms with HE," p. 14.

[31] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A Survey on Homomorphic Encryption Schemes: Theory and Implementation," *ACM Computing Surveys*, vol. 51, no. 4, pp. 79:1–79:35, Jul. 2018.

[32] N. Chenette, K. Lewi, S. A. Weis, and D. J. Wu, "Practical Order-Revealing Encryption with Limited Leakage," in *Fast Software Encryption*, T. Peyrin, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, vol. 9783, pp. 474–493.

[33] K. Lewi and D. J. Wu, "Order-Revealing Encryption: New Constructions, Applications, and Lower Bounds," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16*. Vienna, Austria: ACM Press, 2016, pp. 1167–1178.

[34] D. Bogatov, G. Kollios, and L. Reyzin, "A Comparative Evaluation of Order-Preserving and Order-Revealing Schemes and Protocols," p. 19.

[35] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Sciences*, vol. 28, no. 2, pp. 270–299, Apr. 1984.

[36] J. Katz, *Introduction to Modern Cryptography: Principles and Protocols*, 1st ed. Chapman and Hall/CRC, Aug. 2007.

[37] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, "Relations among notions of security for public-key encryption schemes," in *Advances in Cryptology — CRYPTO '98*, G. Goos, J. Hartmanis, J. van Leeuwen, and H. Krawczyk, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, vol. 1462, pp. 26–45, series Title: Lecture Notes in Computer Science.

[38] M. Chase, H. Chen, J. Ding, S. Goldwasser, S. Gorbunov, J. Hoffstein, K. Lauter, S. Lokam, D. Moody, T. Morrison, A. Sahai, and V. Vaikuntanathan, "SECURITY OF HOMOMORPHIC ENCRYPTION," p. 27.

[39] D. J. Lilja and S. S. Sapatnekar, "Designing Digital Computer Systems with Verilog," p. 176.

[40] E. L. Oberstar and O. Consulting, "Fixed-Point Representation & Fractional Math," p. 19.