3D Relative Localization of Mobile Systems using Distance-only Measurements via Semidefinite Optimization

Bomin Jiang, Brian D.O. Anderson and Hatem Hman

Abstract—In a network of cooperating unmanned aerial vehicles (UAVs), individual UAVs usually need to localize themselves in a shared and generally global frame. This paper studies the localization problem for a group of UAVs navigating in 3D space with limited shared information, viz., noisy distance measurements are the only type of inter-agent sensing that is available, and only one UAV knows its global coordinates, the others being GPS-denied. Initially for a two-agent problem, but easily generalized to some multiagent problems, the paper first establishes constraints on the minimum number of distance measurements required to achieve the localization. The paper then proposes a composite algorithm based on semidefinite programming (SDP) in a first step, followed by maximum likelihood estimation using gradient descent on a manifold initialized by the SDP calculation. The efficacy of the algorithm is verified with experimental noisy flight data.

I. INTRODUCTION

Research on multiagent systems has been attracting increasing interest in recent years. This is due to the potential applications of this research in a variety of fields, including sensor networks [1], distributed power grids [2], distributed computation and so on. For certain multiagent systems, such as a cooperating group of unmanned aerial vehicles (UAVs), there are some tasks, e.g. formation shape control [3], state consensus [4], or self-localization [1], which often need to be performed in a distributed way and using limited sensing.

A. The broad problem statement with rationale

A common task for UAV formations is that of localizing a target, such as a radio-frequency emitter. A prerequisite for this is for all agents to know their positions in a common, possibly global coordinate basis, since otherwise vehicles cannot sensibly process and fuse targeting information obtained by other vehicles. However, not all vehicles may have access to GPS, because of, for example, obstacles obstructing the line-of-sight to satellites (e.g. receivers in a dense urban canyon) or GPS signal interference (jamming). Nevertheless, if one vehicle does have such access, then it can assist the other vehicles in the same formation to geolocate themselves, making possible collective localization of an emitter. Those other vehicles, being GPS-denied or not GPS-equipped, are assumed to be equipped with inertial navigation sensing, which means that each views its position and motion in a local coordinate frame whose orientation and position relative to the global frame are unknown. The purpose of this work is to present an approach that expresses the local UAV position data in a global reference system, determined by the single UAV, which has access to GPS signals. Such problems are not completely novel. The book chapter [5] provides a high-level introduction to localization in a 3D environment, including limited consideration of mobility. While localization can be easily achieved in principle if vehicles have sensors delivering relative positions (i.e. both range and bearing) of their neighbors, and an ability to communicate with their neighbors, distance-only sensing is preferred in many scenarios [3], [6], [7] due to its reliability, low cost, low power and lightweight sensors.

We note however that bearing-only measurements are often preferred when passive sensing is required. Early analysis of bearing-only localization can be found in [8], and also [9] discusses the mobile localization problem with bearing-only measurements. It is evident though that a method that can localize non-GPS-equipped UAVs using a set of distance-only measurements between mobile vehicles together with INS data from non-GPS-equipped vehicles is of great appeal in many applications. Providing such a method is the motivation for this paper.

Additionally, we seek a method which degrades gracefully, rather than collapses, in the presence of increasing noise. Quite often in nonlinear estimation problems, estimators can perform well in small noise, but when the noise reaches a certain threshold, collapse in performance occurs with the error of estimates sharply increasing. With the performance metric we introduce in Section IV-A, we find no such collapse occurs with our method.

B. Relation to other relevant work

In considering other relevant work, it is useful to distinguish between the application context and the methodologies. One paper with overlapping authorship, viz. [10], reports work on a rather similar problem, conducted in parallel with the work of this paper, and differing from it through the assumption of direction rather than distance measurement availability. By way of more general comment, in the context of our application, understanding the tradeoffs between numbers of measurements and SNR is critical, as is the identification of the minimum measurement requirements, since
for reasons of maintaining covention and time window of likelihood optimization using gradient descent on the manifold availability for measurements, often there will be a desire, even requirement, to minimize the number of measurements. Our approach These matters are little addressed in other work. Other unique aspects of our application context in comparison with other work focused on similar applications include the treatment of time synchronization offsets, consideration of adverse geometries and the establishment of a framework for treating localization of a single point (equivalently a stationary agent).

For the sake of completeness, we remark that there are several methods have been proposed to solve nonconvex optimization problems, frequently involving fixed time synchronization offsets, consideration of adverse geometries and the establishment of a framework for treating localization of a single point (equivalently a stationary agent).

For the sake of completeness, we remark that there are several methods have been proposed to solve nonconvex optimization problems, frequently involving fixed time synchronization offsets, consideration of adverse geometries and the establishment of a framework for treating localization of a single point (equivalently a stationary agent).

We now move on to consider aspects of our composite algorithm and its relation to the literature. Many authors have recognized that localization problems can often be formulated as nonconvex optimization problems, frequently involving a quadratic performance index and quadratic constraints, GPS-equipped agent. It also discusses the minimal number of measurements required, including insights from the theory of programming (SDP) relaxation followed by a maximum like several major steps. These steps include SDP followed by a
Consider two agents in 3D space, as shown in Figure 1. Agent $x$ is the reference agent, whose position in the global coordinate system is denoted by $p_x = [x_1, x_2, x_3]^T$. Agent $y$ is the agent to be localized, whose position in its own local coordinate system is denoted by $p_y = [y_1, y_2, y_3]^T$. Suppose further that a coordinate system transformation, including a rotation matrix $R$ and a $3 \times 1$ translation vector $T$, can transfer agent $y$’s local coordinates into global coordinates, so that the position of agent $y$ in the global coordinate system is $Rp_y + T$. The matrix $R$ and vector $T$ are assumed to be constant and unknown through the interval over which measurements are taken, i.e., the cumulative drift in the INS system over the time interval of measurements is assumed negligible. Equivalently, the time interval is sufficiently short.

In addition, suppose $d$ is the distance between the two agents and define

$$
\bar{p} = Rp_y + T - p_x
$$

Thus $\bar{p}$ is the relative position vector of agent $y$ as seen from agent $x$, expressed in global coordinates. Immediately we have

$$
d = ||\bar{p}|| = ||Rp_y + T - p_x||
$$

Suppose at time instants $\tau = \tau_k$, $k = 1, 2, \ldots, N$, each agent acquires the distance measurement $d[k]$ and each knows the coordinates $p_x[k]$ and $p_y[k]$ at these time instants through message interchange. The objective is to infer $R$ and $T$ after a finite number of measurements. Knowledge of $R$ and $T$ enables one to estimate the positions of agent $y$ in global framework is again globally rigid. This means that if one of the frameworks before merging is localized, the whole framework is rigid but is in fact uniquely defined up to a congruence. Now the coordinate-assignment problem of interest to us can generally be reformulated as one in which (a) associates a 3D (globally rigid) body with each agent (the body being a polyhedron with vertices defined by the agent positions at each time an inter-agent distance is measured), (b) associates a bar (or link between two bodies) with each inter-agent distance measurement, and (c) seeks conditions on the overall number of links and rules concerning the points on the bodies to which they can be incident to ensure that the entire framework is globally rigid, i.e. every joint or bar end in the framework has a computable location in a coordinate frame fixed to the framework, which cannot flex. The result states that if two globally rigid frameworks in an ambient three-dimensional space are joined by 7 or more bars of fixed lengths (and the endpoints of the bars can only coincide to a limited extent, or are distinct) then the merged framework is again globally rigid. This means that if one of the frameworks before merging is localized, the whole framework is globally rigid.

Note that there are 6 degrees of freedom to pin down, three each associated with $R$ and $T$. Therefore, one must expect that at least 6 measurements would be needed to solve the relevant equation set. With precisely 6 measurements, as

$$
1.4 \text{ The equation set is actually a set of simultaneous polynomial equations. Unless all the equations are linear, as a matter of algebra } n \text{ simultaneous polynomial equations in } n \text{ unknowns always have multiple solutions, provided they are solvable in the first place.}
$$
framework after merging (and thus the second framework before merging) is localized.

The lower bound, 7, carries over to the localization problem. Further, if all measurements were noiseless, there would be no reason to use more than 7 measurements. Noise in practice is very likely, and so having more than 7 rigid transformation, \( \hat{p}_g = R p_g + T \), where \( R \) is a \( 3 \times 3 \) rotation matrix and \( T \) is a 3-vector. As before, \( d \) denotes localization error and, less obviously, to mitigate errors in an inter-agent distance.

This is discussed much later, as also is the existence of exceptional trajectories (which can always be avoided) for which no solution can be found, irrespective of the number of measurements and solution methods.

III. SOLUTION TO THE TWO-AGENT PROBLEM

In this section, we focus on the two-agent case, and later note in a Remark, how the ideas can be applied to certain multiagent arrangements. We first present, in subsection A, an SDP-based approach for solving a constrained least squares problem where no less than 7 measurements are assumed available. This solution actually is a relaxed solution (in the sense of optimization theory) of the problem of interest, and so a further step may have to be undertaken to deal with possible consequences of the relaxation. This further step may induce failure of the constraint that the matrix \( R^2 \) is orthogonal with determinant 1, and so we show how to modify the outcome of the calculations of Subsection III-A to achieve an orthogonal matrix in Subsection III-B.

As we shall see later in this section, the SDP relaxation solves a least squares error problem subject to a number of quadratic equality constraints, and therefore any obtained optimal solution may not be optimal in a maximum likelihood sense. The reason is that the distance measurement errors are mixed and transformed nonlinearly during the process of deriving the least squares error objective function. This holds true in the case where the measurement errors are zero-mean Gaussian. In the special case where the measurements are error-free, both optimization problems are error-free, and so the SDP relaxation solves a least squares error problem subject to constraints.

A. A semidefinite programming relaxation

Suppose \( R = \{r_{ij}\}, \ T = [t_1, t_2, t_3]^T \). There holds for measurements at time \( \tau_1, \tau_2, \ldots, \tau_N \) (with the time index suppressed in the following equations)

\[
d^2 = \| \hat{p}_g - p_x \|^2
\]

The equation for \( d^2 \) can be written as

\[
d^2 = \| \hat{p}_g - p_x \|^2
= \| R p_g + T - p_x \|^2
= \| p_g \|^2 + \| T \|^2 + \| p_x \|^2 + 2T^T R p_g - 2p_g^T R p_g - 2p_g^T T
\]

The fourth term involves quadratic expressions in the unknowns, i.e. the entries of \( R \) and \( T \). The fifth and sixth terms involve linear terms only. The end result is the following quadratic constraint on the entries of \( R \) and \( T \), presented to place all the unknowns on the right side of the equation:

\[
d^2 - \| p_x \|^2 - \| p_g \|^2
= -2x_1 y_1 r_{11} - 2x_1 y_2 r_{12} - 2x_1 y_3 r_{13}
-2x_2 y_1 r_{21} - 2x_2 y_2 r_{22} - 2x_2 y_3 r_{23}
-2x_3 y_1 r_{31} - 2x_3 y_2 r_{32} - 2x_3 y_3 r_{33}
-2x_1 t_1 - 2x_2 t_2 - 2x_3 t_3
+2y_i \sum r_{i1} t_1 + 2y_j \sum r_{i2} t_2 + 2y_3 \sum r_{i3} t_3 + \sum i^2 t_i
\]

If we regard each summand on the right side of (4) as a product of known values \(-2x_1 y_1, -2x_1 y_2, \ldots, 1\) we introduce the index relevant to the MLE calculation of independent unknowns \( r_{11}, r_{12}, \ldots, \sum t_i^2 \), one can place all the unknowns on the right side of the equation:

\[
\sum i^2 t_i
\]

This index is evidently quite different from that of the SDP seek to solve the set of equations with a sufficient number of measurements. Generally, 16 measurements are sufficient solution. This is where the outcome of SDP relaxation analysis in the above solution process, we (temporarily) regard each sis, developed in this section, becomes relevant. The gradient summand in (4) as a product of a known value and an inde

flow is nonstandard, in that the orthogonality constraint on the "pendent unknown. This therefore leaves out the consideration rotation matrix needs to be preserved by the flow. Subsection of the nonlinear constraints on the unknowns (though recall III-D reviews situations where agent trajectories are non that (4) only arises after use of a number of such constraints), generic while Subsection III-E explains how to handle timing These constraints, if accounted for, reduce the number of issues arising from lack of synchronization of clocks between measurements required and improve the estimation accuracy the UAVs.
Define \( \Theta = [\theta_1, \theta_2, \ldots, \theta_{16}]^T \) to be the 16-vector of \( \Theta \) shown later in the paper. In addition, because the rotation unknowns \( r_{11}, r_{12}, \ldots, \sum r_{i3} t, \sum t_i^2 \) in (4) and \( A[k] = 42 \) matrix entries should satisfy \(-1 \leq r_{ij} \leq 1\), one can also consider \([-a_k, 1, a_k, 2, \ldots, a_k, 16]\) to be the row vector of known values \( a_k \) sider including Reformulation-linearization-technique (RLT) and all the largest singular value to \( 42a \), see Section 4.3 of [25]. In fact, it is generally a very accurate approximation because the solution of the relaxed problem is already close to rank 1. (In a number of numerical simulations, the largest singular value of the solution turned out to be generally \( 10^2 \sim 10^5 \) times larger than the second largest singular value.)

The solution of the above SDP can then be used as an initial condition to the gradient optimization presented below. Note there are altogether 10 independent constraints as listed in Appendix I. An additional 4 measurements used in the semidefinite programming method four constraints are listed which are not independent of the 10 just mentioned.

We now show how to solve (5), using semidefinite programming. Define

\[
X = \begin{bmatrix} \Theta & 0 \\ -1 & -1 \end{bmatrix}^T
\]

Solving for \( \Theta \) is equivalent to solving for \( X \) with the constraints that

1. \( X \) is positive semidefinite (denoted by \( X \succeq 0 \))
2. \( \text{rank}(X) = 1 \)
3. The bottom right corner element \( X_{17,17} = 1 \)
4. Suppose that \( \langle u, v \rangle \) denote the inner product of two matrices \( u \) and \( v \), i.e. \( \langle u, v \rangle = \text{trace}(u^\top v) \), and define
5. \( P = [A \ b]^\top [A \ b] \)
6. The objective function \( \frac{1}{2} ||A\Theta - b||^2 \) can be written as \( \frac{1}{2} \langle P, X \rangle \)

Similarly, the constraints in (19) can be written in terms of \( X \)

\[
\langle Q_i, X \rangle = q_i, i = 1, \ldots, 10
\]

where \( Q_i \) are 17 by 17 symmetric matrices and \( q_i \) are 10 scalars. Now the optimization problem becomes

\[
\begin{align*}
\text{argmin}_\Theta & \langle P, X \rangle \\
\text{subject to} & \langle Q_i, X \rangle = q_i, i = 1, \ldots, 10 \\
& X_{17,17} = 1 \\
& X \succeq 0 \\
& \text{rank}(X) = 1
\end{align*}
\]

A naive semidefinite programming relaxation is given by dropping the rank constraint. After dropping the rank \( \text{rank}(X) \) diagonal being \(-1 \) and all other entries on the diagonal being \( C_t = 0 \) no longer imply \( C_t = 1 \). The solution of this constrained version of the Orthogonal Procrustes problem is

\[
\bar{R} = U \Sigma V^* 
\]

Suppose \( J \) is a diagonal matrix with the last entry on the by dropping the rank constraint. After dropping the rank \( \text{rank}(X) \) diagonal being \(-1 \) and all other entries on the diagonal being \( C_t = 0 \) no longer imply \( C_t = 1 \). The solution of this constrained version of the Orthogonal Procrustes problem is

\[
\tilde{R} = \begin{cases} UV^* & \text{if } \det(UV^*) = 1 \\ UV & \text{if } \det(UV^*) = -1 \end{cases}
\]

Note that \( \det(\Sigma - I) \) can be used as an error measure.
C. Maximum Likelihood Estimation and Gradient Flow on a Manifold

In principle, in the presence of noise, the best choices for \( R \) and \( T \) are the maximum likelihood estimates of those quantities. As in many estimation problems, closed-form solutions expressions rarely exist, and minimizing the performance index often leads to convergence to a local minimum. Our approach is to use the solution of the SDP relaxation to initialize a gradient descent algorithm. As examples later show, this proves highly effective.

The entries of \( R \) and \( T \) constitute the variables being estimated. A somewhat non-standard descent algorithm (gradient descent on a manifold) is required to ensure that the orthogonality constraint is preserved.

Suppose the measurement distance, \( z[k] \), is contaminated by a Gaussian noise with zero mean and standard deviation \( \sigma \); i.e. the measurement sensor delivers \( \tilde{z}[k] = z[k] + \xi, \xi \sim N(0, \sigma^2) \). We assume in this section that \( p_x \) and \( p_y \) can be obtained without noise and that the measurement noise values at different times are independent.

Now we obtain \( \tilde{z}[k] - \| R p_y[k] + T - p_x[k] \| \sim N(0, \sigma^2) \). The likelihood function is

\[
\mathcal{L}(p_x, p_y, z|R, T) = \frac{1}{\sigma \sqrt{2\pi}} \prod_{k=1}^{N} \exp\left[ -\frac{(\tilde{z}[k] - \| R p_y[k] + T - p_x[k] \|)^2}{2\sigma^2} \right]
\]

or equivalently,

\[
\log \mathcal{L}(p_x, p_y, z|R, T) = \sum_{k=1}^{N} \left[ -\frac{(\tilde{z}[k] - \| R p_y[k] + T - p_x[k] \|)^2}{2\sigma^2} \right] - \log(\sigma \sqrt{2\pi})
\]

Therefore the maximum likelihood estimate is given by solving the optimization problem

\[
\hat{R}, \hat{T} = \arg \min_{R,T} \sum_{k=1}^{N} (\tilde{z}[k] - \| R p_y[k] + T - p_x[k] \|)^2
\]

subject to \( RR^T = 1 \) and \( \det(R) = 1 \)

As in many MLE estimation problems, the index is not convex and minimization is not necessarily straightforward. A particular problem with using gradient descent on any nonconvex function is to find an initialization within the capture region of the global minimum, and this is where the calculations of the previous subsections become relevant, if not critical. We use the result from linear processing (with the number of measurements being 16) or preferably the SDP approach of the previous subsections (with the number of measurements being greater than or equal to 7 and the solution adjusted if necessary via the Procrustes algorithm) to initialize a gradient descent algorithm aimed at finding the minimum.

It is useful for this purpose to know how to compute the gradient of a function of a special orthogonal matrix on the manifold of special orthogonal matrices. Consider \( f: SO_3 \rightarrow \mathbb{R} \), mapping special orthogonal matrices to the reals. Suppose we want to compute the gradient, reflecting the orthogonal property.

The general idea (technically a consequence of the fact that \( SO_3 \) is a Riemannian manifold and so inherits a metric from the Euclidean space in which it is embedded [28]) is: first we consider a point \( R \in \mathbb{R}^{3 \times 3} \) on the \( SO_3 \) manifold, and compute the gradient \( \partial f / \partial R \) in the standard way, then we project it onto the tangent space of \( SO_3 \) at the point \( R \). For this purpose, we need to have the tangent space and normal space of \( SO_3 \) at some point \( R \), and also the projection of a vector to the tangent space.

Lemma 1. i. The tangent space of \( SO_3 \) at \( R \in SO_3 \) is the set of \( P \) such that [28]

\[
P^T R + R^T P = 0
\]

or equivalently

\[
T_{SO_3}(R) = \{ RQ, Q + Q^T = 0 \}
\]

and the normal space is

\[
N_{SO_3}(R) = \{ RS, S - S^T = 0 \}
\]

where \( T_{SO_3}(R) \) and \( N_{SO_3}(R) \) denote the tangent and normal spaces at \( R \) respectively.

ii. Furthermore, suppose at a point \( R \in SO_3 \), the gradient of \( f \) in \( \mathbb{R}^{3 \times 3} \) is \( M = \partial f / \partial R \). Then the projection of \( M \) on the tangent space \( T_{SO_3}(R) \) is given by

\[
M_T = \frac{1}{2} M - \frac{1}{2} R M^T R
\]

Proof. The proof is given in Appendix II.

Now suppose \( f = \sum_{k=1}^{N} (\tilde{z}[k] - \| R p_y[k] + T - p_x[k] \|)^2 \).

It is straightforward to obtain the gradient \( M_T \) and \( \partial f / \partial R \), and so we have the gradient flow

\[
\dot{R} = -M_T, \quad \dot{T} = -\frac{\partial f}{\partial T}
\]

In summary, the SDP relaxation of Section III is likely to give a good initial condition, and then a gradient method using a discretization of (13) is sufficient for solving this optimization problem. The Procrustes problem algorithm can be applied

\[2\text{In recently published work [27] studying autonomous underwater vehicle localization using distance-only measurements to a non-GPS-equipped vehicle, minimization of the same MLE index is tackled using a form of approximation for the index, with the application of a parallel projection algorithm. Proper comparison with the methods of this paper cannot be made, due to the limited details in the paper. How issues of initialization can be effectively tackled is also not clear.}]}
in each step to correct departure from orthogonality due to round-off and discretization error.

Remark 1. In the case of multiagent localization, the same algorithm can be adopted for those networks where each non-GPS equipped agent has at least 7 measurements to a GPS-equipped agent. Thus if there is only one GPS-equipped agent, the graph will be a star form; each of the GPS-denied agents (with INS) must have 7 or more distance measurements between it and the GPS-equipped agent. This allows a collection of two-agent problems to be solved, possibly in parallel. Note that this topology may well correspond to the physical situation of one agent remaining high above a building canyon, while other agents navigate within the building canyon.

Remark 2. The objective function of the constrained linear least squares problem and the objective function of the gradient descent method are different. The first one is designed to exploit as far as possible the linear occurrence of unknowns in the $z^2$ expressions while the second one is derived from a maximum likelihood estimator. As noted, the constrained least squares problem should be used to find an appropriate starting point for the gradient descent method, which then finds the maximum likelihood estimate.

D. Observability issues and nongeneric trajectories

With certain nongeneric trajectories, localization is not achievable by taking only distance measurements. To understand this phenomenon, we first study the linear estimator, as follows. In (5), if we have more than 16 distance measurements to this phenomenon, we first study the linear estimator, as achievable by taking only distance measurements. To understand this topology may well correspond to the physical situation of one agent remaining high above a building canyon, while other agents navigate within the building canyon.

To tackle the problem with biased distance measurements, we consider an initially unknown constant $s$ being added to our measurements, i.e., the obtained distance measurement is $d_o$ but the actual distance becomes $d_o + s$. We seek to estimate the unknown $s$ as well as the transformation between the two coordinate frames. Making an obvious change to equation (4), we obtain

$$d_o^2 - ||p_x||^2 - ||p_y||^2 = -2x_1y_1r_{11} - 2x_1y_2r_{12} - 2x_1y_3r_{13} - 2x_2y_1r_{21} - 2x_2y_2r_{22} - 2x_2y_3r_{23} - 2x_3y_1r_{31} - 2x_3y_2r_{32} - 2x_3y_3r_{33}$$

$$-2x_1t_1 - 2x_2t_2 - 2x_3t_3 + 2y_1 \sum r_{11}t_1 + 2y_2 \sum r_{12}t_2 + 2y_3 \sum r_{33}t_3 + \sum t_i^2 + s^2 + 2d_o s$$

Now the vector of unknowns $\Theta = [\theta_1, \theta_2, \cdots, \theta_{17}]^T$ is a 17-vector of unknowns $r_{11}, \, r_{12}, \, \cdots, \, \sum r_{33}t_3, \, \sum t_i^2 + s^2, \, s$ in (14). Following a very similar solution procedure to that of Section IIIA (either linear processing or SDP, with the minimum number of measurements being increased by 1), one can obtain estimates of $R$, $T$ and $s$. If SDP is the intended solution method, the constraints $C_9$ and $C_{10}$ in Appendix I should also change to

$$C_9' = \theta_{19}^2 + \theta_{21}^2 + \theta_{23}^2 + \theta_{25}^2 - \theta_{16} = 0$$

$$C_{10}' = \theta_{13}^2 + \theta_{14}^2 + \theta_{15}^2 + \theta_{17}^2 - \theta_{16} = 0$$

IV. SIMULATIONS, PERFORMANCE EVALUATION AND FLIGHT DATA EXAMPLE

A. A performance metric

In order to evaluate the performance of our localization algorithm, a performance metric is selected in this subsection. As we have already defined in the beginning of Section III, one could face a similar singularity problem. The cases of singular trajectories include, but are not restricted to: 1. each agent flies in an affine subspace of $\mathbb{R}^3$ (i.e. their trajectories are individually coplanar), or 2. the distance between agents remains constant at all time. Again, these problems can be avoided if agents’ trajectories have some oscillations.

E. Unsynchronized clocks and distance measurements bias

The most commonly used distance measurement method for UAVs is Time-of-Arrival (ToA). The ToA ranging technique usually requires that the transmitter and receiver have synchronized clocks. With synchronized clocks, the receiver can compute the signal transmission time and obtain a distance measurement. In the case of unsynchronized clocks, constant measurement bias may be added to the computed distances, and degrade the performance of a localization system.

3The assumption of a constant clock bias is essential to our solution approach, but may not apply in general. Small UAVs with cheap oscillators may have a drifting bias during the measurement interval that causes a problem with localization.
That is to say, our objective is to obtain accurate estimates of $T$. For this purpose, the error in the estimate of $R$ is irrelevant.

Also note that
\[
d(\tau_N) = \| R p_y(\tau_N) + T - p_x(\tau_N) \| = \| T \|, \tag{17}
\]
i.e., the norm of $T$ is directly measured by $d(\tau_N)$. Hence our performance metric would be a metric of directional error of $T$. A metric of directional error is defined as
\[
E_{\text{direction}} = \arccos \left( \frac{\hat{T}^\top T}{\| T \| \| T \|} \right) \tag{18}
\]
where $T$ is the true translation vector and $\hat{T}$ is its estimated counterpart.

Other metrics could be considered. However, the primary goal of this paper is the localization of a peer agent. For measuring the error, evaluation of $E_{\text{direction}}$ is enough. In addition to localization, if aligning coordinate systems is also a requirement (and this is frequently so), then we have to look at the full attitude error (in terms of $R$). Due to space limitation, the analytic evaluations of the full attitude error are not provided, but simulation data of total angular error is presented below. Generally speaking, less error in $T$ means less error in $R$, as shown in a number of simulations.

### B. Simulation and Comparison between SDP and Gradient Descent

In the simulations below, white Gaussian noise is added to inter-agent distance measurements, and there are 7 or more such measurements. Furthermore, SNR denotes the Signal-to-noise ratio in dB, defined as $\text{SNR}_{\text{dB}} = 20 \log_{10} \frac{A_{\text{signal}}}{A_{\text{noise}}}$.

- In each figure, we show the directional error of $T$ defined in (18) in radians vs. number of distance measurements.
- Figure 2 shows a typical example as used in the simulations illustrating the actual and estimated trajectory of a non-GPS equipped vehicle.
- Figure 3 depicts the noiseless case, i.e. $\text{SNR} \rightarrow \infty$. The yellow curve shows the result of SDP relaxation. With each number of measurements and each $30, \text{SNR} = 20$ and $\text{SNR} = 10$ respectively. It is seen level of SNR, each point plotted in the above figures is the well known that in practice the distance accuracy $(1-\% \text{ average of 200 simulations with random vehicle trajectories)}$ is inversely proportional to signal $(a pure random walk, whose derivative is Gaussian distributed bandwidth as well as the square root of SNR)$. For $\text{SNR}$ with zero mean and a 1 meter standard deviation in all passive RF detection problems, the SNR in free space directions) and measurements are taken every step of the integrated measurement error.
- In the case of less than or equal to 5 measurements, the gradient of the objective function of the MLE optimization is always zero on a manifold, and the final result is randomly located on the manifold. In addition, in the case of 6 measurements, there is no way to deal with the issue of disambiguating local optima.

Localization method is suitable for vehicles employing higher grade inertial navigation systems. Such vehicles include high-value platforms, long-range missiles, etc. It is important to note that the drift is only required to be insignificant within a short period of time.

- Gradient descent refinement can provide an improved result in comparison to SDP relaxation. This is partly because gradient descent refinement uses a better estimator, and partly because of the relaxation error in SDP. However, it is also notable that SDP relaxation is crucial in providing the initial condition to avoid convergence to a local minimum.
- As noted in the introductory section, the paper [13] studies a similar problem to ours, with application in robotics systems. Despite the different application scenarios, we implemented the algorithm of [13] in the context of our application and made a direct comparison of the directional error of $T$ to benchmark the performance of our algorithm.

From [12], one notes that with 6 distance measurements, the 3D geolocation problem has 40 solutions, where some of the solutions may not be real and must be discarded. The paper [15] further shows that in the absence of noise, one additional measurement can almost always disambiguate the solutions and with 7 distance-only measurements, a unique solution can be found. Although the use of SDP followed by the gradient-based MLE algorithm as proposed in this paper apparently does not require a minimum number of measurements, an implicit assumption is that the number of measurements is indeed greater than or equal to 7. In the other hand, the algorithm in [13] requires at least 10 measurements, and even with 10 measurements, the algorithm sometimes fails to converge. In our implementation, we find that the algorithm in [13] can reliably converge with 12 or more measurements. Hence we record the error data of that algorithm starting from 12 measurements.

Figures 3-6 show the change of direction error in degrees added to the distance measurements.
Evidently, our estimator is not a linear estimator; this means that (as is common for nonlinear estimators) it is not necessarily unbiased, and in fact we have found that it is biased. Therefore, as the number of useful measurements increases, the direction error does not converge to zero in Figure 4-6. The high level conclusion is that it is possible to handle a large number of measurements, and there is graceful degradation of estimation quality with increasing noise, with no apparent collapse at a certain point.

Compared to the method provided in [13], our approach in general achieves better accuracy, especially under excessive noise. The SDP processing by itself is better in all cases with noise, and the performance of SDP with gradient refinement is superior in all cases. Most importantly, our method is far more robust in three aspects. Firstly, when the noise is large, the method proposed in [13] often diverges (the highest percentage of diverging cases reported is 65%, corresponding to odometer error $\sigma_o = 0.1m$). Our method, on the other hand, despite exhibiting gradually degrading accuracy, never has this divergence issue. Furthermore, the proposed method in [13] only works with a minimum of 10 distance measurements. In the case of 7, 8 or 9 measurements, our method will still work but the one in [13] will not. Finally, our proposed SDP+MLE framework can handle system bias with only minor modifications. For example, Section III-E proposes minor algorithm modifications when there is measurement bias due to unsynchronized clocks.

Both the quaternion method in [13] and our method relies on gradient methods (e.g., interior-point method) to find solutions. Note the SDP relaxation is a way to convert non-convex optimization problems into convex optimization problems, and the solution method is the same with any standard convex optimization problems. As a result, the computational complexity is not drastically different. We do not have an opportunity to test the method in the on-board computational devices on a UAV, but the SDP+MLE method finishes within 1 second on a laptop. In general, our method may use more computational time than [13], but this should not be perceived as a major limitation.

C. Trial on real data

In order to validate the key assumptions in our model (including measurement noise level, odometer accuracy, measurement frequency, etc.), we applied our SDP+MLE method on real data. In particular, real flight data provided by Australian Defence Science and Technology Group is used in this section to evaluate the performance of the proposed method in practice. The data consists of: the true positions of UAVs in global coordinates, the positions of UAV2 in its local INS coordinates and the distance measurements between the pair of agents. The relevant numbers are summarized in Table I.

For our field test UAVs, with an air data unit the IMU will typically achieve accuracy of 0.2% of distance travelled. On the other hand, without an air data unit, it will drift about 10% of the distance travelled. For example, in 5 minutes of flight the unit will travel 9km (at 30m/s) and incur an error of approximately 18 meters with an air data unit, (and drift somewhere in the order of 500 - 1000 metres in 5 minutes without an air data unit). This all means that, in our field test data, the global coordinates data of UAV 1 can be regarded as error-free, and the local coordinates data of UAV 2 are obtained from an IMU with an air data unit (Spatial Dual from Advanced Navigation). Furthermore, the error for distance measurements is about 15m while UAVs are on average 1500m apart from each other, i.e., the SNR is roughly 40 dB. Here distance measurements are post processed distance measurements created by adding noise to the distance obtained from GPS.

<table>
<thead>
<tr>
<th>Time(s)</th>
<th>UAV1 Local Coordinates</th>
<th>UAV2 Local Coordinates</th>
<th>Dances measured (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2</td>
<td>349.1 -924.1 374.4</td>
<td>1038.3 680.9 311.2</td>
<td>1541.0</td>
</tr>
<tr>
<td>12.0</td>
<td>576.2 -945.0 371.9</td>
<td>1249.5 637.9 311.0</td>
<td>1516.7</td>
</tr>
<tr>
<td>22.7</td>
<td>781.0 -870.3 372.5</td>
<td>1481.7 679.0 308.5</td>
<td>1381.6</td>
</tr>
<tr>
<td>32.5</td>
<td>938.8 -712.4 373.7</td>
<td>1708.6 717.2 308.4</td>
<td>1149.9</td>
</tr>
<tr>
<td>42.3</td>
<td>1007.0 -522.7 373.3</td>
<td>1939.5 751.5 309.3</td>
<td>907.7</td>
</tr>
<tr>
<td>52.1</td>
<td>992.0 -599.4 373.5</td>
<td>2084.3 867.7 308.5</td>
<td>800.7</td>
</tr>
<tr>
<td>61.4</td>
<td>869.8 91.3 372.2</td>
<td>2040.9 1088.0 309.4</td>
<td>899.1</td>
</tr>
<tr>
<td>71.6</td>
<td>660.1 38.6 372.9</td>
<td>2004.3 1305.7 310.1</td>
<td>1120.9</td>
</tr>
<tr>
<td>80.5</td>
<td>431.4 56.6 371.1</td>
<td>1978.6 1507.0 308.5</td>
<td>1435.9</td>
</tr>
<tr>
<td>91.1</td>
<td>189.7 -49.1 373.3</td>
<td>1933.3 1723.2 310.7</td>
<td>1867.1</td>
</tr>
<tr>
<td>100.9</td>
<td>33.9 -262.2 373.6</td>
<td>1724.3 1755.4 310.5</td>
<td>2084.6</td>
</tr>
</tbody>
</table>

In this computation with real data, we take UAV2’s perspective, i.e., it obtains UAV1’s global coordinates at several time instants through the processing of distance measurements via ToA of those signals (in this experiment, both UAVs’ clocks are accurately synchronized), in conjunction with own IMU acceleration and angle rate data. Feeding those data as inputs, UAV2 computes its own positions in the global coordinates at those time instants using semidefinite programming followed by gradient MLE. In Figure 7, circles denote the true positions of UAV 1 in global coordinates and triangles denote the computed positions of UAV 2 in the global coordinates. For comparison purposes the true trajectory of UAV2 in the global coordinates is also directly measured and is displayed with a solid line. If we compute the error metric defined in (18), we find the direction error of our estimation is 0.03 rad. This result is consistent with the simulation result with 11 measurements and SNR of 30dB.

In Table II, we see that the localization algorithm achieves better accuracy in the north and east directions but poorer accuracy in height. This can be explained by the observation that the trajectories of the UAVs are very close to coplanar, and that the distance measurements shed very little information on the height dimension. Indeed, if we change the height recorded by a UAV’s inertial sensor by a few metres, a ‘mirror solution’ will be obtained. In fact, if the flying trajectories are exactly planar, then as remarked earlier it is not possible to recover bearing information from distance-only measurements due to singularity issues. However, even
TABLE II
ERROR IN X, Y AND Z COORDINATES, TOTAL POSITION ERROR, AND ANGULAR ERROR.

<table>
<thead>
<tr>
<th>Measurement No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-coordinate error (m)</td>
<td>31</td>
<td>29</td>
<td>20</td>
<td>12</td>
<td>2</td>
<td>9</td>
<td>32</td>
<td>36</td>
<td>56</td>
<td>83</td>
<td>103</td>
<td>246</td>
<td>229</td>
<td>271</td>
<td>215</td>
<td>164</td>
<td>117</td>
<td>66</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>Y-coordinate error (m)</td>
<td>14</td>
<td>6</td>
<td>0</td>
<td>9</td>
<td>31</td>
<td>4</td>
<td>41</td>
<td>57</td>
<td>68</td>
<td>90</td>
<td>124</td>
<td>239</td>
<td>200</td>
<td>241</td>
<td>275</td>
<td>216</td>
<td>167</td>
<td>118</td>
<td>67</td>
<td></td>
</tr>
<tr>
<td>Z-coordinate error (m)</td>
<td>22</td>
<td>82</td>
<td>141</td>
<td>194</td>
<td>250</td>
<td>267</td>
<td>213</td>
<td>164</td>
<td>117</td>
<td>66</td>
<td>56</td>
<td>83</td>
<td>103</td>
<td>246</td>
<td>229</td>
<td>271</td>
<td>215</td>
<td>164</td>
<td>117</td>
<td></td>
</tr>
<tr>
<td>Total error (2-norm) (m)</td>
<td>47</td>
<td>87</td>
<td>142</td>
<td>195</td>
<td>251</td>
<td>269</td>
<td>221</td>
<td>182</td>
<td>159</td>
<td>143</td>
<td>137</td>
<td>170</td>
<td>204</td>
<td>239</td>
<td>271</td>
<td>215</td>
<td>164</td>
<td>117</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>Total angular error (degrees)</td>
<td>1.72</td>
<td>3.43</td>
<td>5.73</td>
<td>9.74</td>
<td>15.47</td>
<td>18.91</td>
<td>14.32</td>
<td>9.17</td>
<td>6.30</td>
<td>4.58</td>
<td>4.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

with this close-to-singular setting, our method successfully recovered location information with reasonable accuracy.

The data provided include 11 distance measurements. Although the result shown in Figure 7 uses all 11 measurements, we only need 7 measurements to recover the positions of UAVs in the global coordinates. Further calculations show that results obtained with fewer distance measurements have progressively decreasing accuracy as expected. Actually, this field test perhaps delivers good results because the two UAVs are close to each other. viz., the two UAVs are 1 km apart on average, and move an average of 0.2 km between taking distance measurements. Our method performs well in this setting with the above two numbers on the same scale. In real applications, however, if the two UAVs are very far apart from each other, the accuracy of localization degrades. In order to improve localization, the ‘agent geometry and mobility’ might have to be improved during distance measurement collection. This may mean that UAVs have to travel much faster and occasionally execute direction changes.

V. CONCLUSIONS

In this paper, we first proposed a novel semidefinite optimization approach for solving the problem of 3D multiagent localization of a GPS-denied agent using distance-only measurements. After that, a maximum likelihood estimator is used in a further approach to enhance the accuracy of localization, with simulations using real field test data.

Future work includes introducing a systematic treatment of the multiagent case, possibly including batch approaches, using a similar procedure to that of this paper. We are also involved in a separate study of localization of GPS-denied agents using bearing-only (azimuth and elevation) measurements. Furthermore, finding optimal trajectory for UAV-to-UAV distance-based positioning is also a future research direction as UAV trajectories have a significant effect on the accuracy of such position methods.

Because our estimator is not linear, bias will be present in our final estimates. Another direction for future research is to provide a bias correction procedure for both the pair-agent and multiagent cases.

APPENDIX I

The 10 constraints are listed as below (the first six coming from orthogonality alone). Let $C_i(\Theta)$ be the $i$th constraint; we have

\[
C_1 = \theta_1^2 + \theta_2^2 + \theta_3^2 - 1 = 0
\]

\[
C_2 = \theta_1^2 + \theta_2^2 + \theta_3^2 - 1 = 0
\]

\[
C_3 = \theta_1^2 + \theta_2^2 + \theta_3^2 - 1 = 0
\]

\[
C_4 = \theta_1^2 + \theta_2^2 + \theta_3^2 - 1 = 0
\]

\[
C_5 = \theta_1^2 + \theta_2^2 + \theta_3^2 - 1 = 0
\]

\[
C_6 = \theta_1 \theta_2 + \theta_3 \theta_4 + \theta_5 \theta_6 = 0
\]

\[
C_7 = \theta_1 \theta_2 + \theta_3 \theta_4 + \theta_5 \theta_6 - \theta_1 \theta_3 = 0
\]

\[
C_8 = \theta_2 \theta_3 + \theta_4 \theta_5 + \theta_6 \theta_7 - \theta_2 \theta_4 = 0
\]

\[
C_9 = \theta_1 \theta_2 + \theta_3 \theta_4 + \theta_5 \theta_6 = 0
\]

\[
C_{10} = \theta_1 \theta_2 + \theta_3 \theta_4 + \theta_5 \theta_6 - \theta_1 \theta_3 = 0
\]

Note there are 10 independent equality constraints and 16 independent variables, so the problem has 6 degrees of freedom. That is consistent with the fact that each of the rotation matrix $R$ and the translation matrix $T$ has three degrees of freedom. One should also note that the equation set used to express those constraints is not unique. In fact, there are 4 other constraints being dropped here, which can be derived from $C_i = 0, i = 1, \ldots, 10$. The additional constraints, of which the first 3 come from orthogonality of $R$, are

\[
C_{11} = \theta_1^2 + \theta_2^2 + \theta_3^2 - 1 = 0
\]

\[
C_{12} = \theta_1 \theta_2 + \theta_3 \theta_4 + \theta_5 \theta_6 = 0
\]

\[
C_{13} = \theta_2 \theta_3 + \theta_4 \theta_5 + \theta_6 \theta_7 - \theta_2 \theta_4 = 0
\]

\[
C_{14} = \theta_1 \theta_2 + \theta_3 \theta_4 + \theta_5 \theta_6 - \theta_1 \theta_3 = 0
\]

During the variable lifting process required in the SDP relaxation, these four constraints can no longer be inferred from the 10 constraints above, and have to be accounted for in the relaxation problem.

APPENDIX II

This appendix presents the proof of Lemma 1.

**Proof.** The first part of the proof regarding the tangent and normal spaces of $SO_3$ (i.) is given in [28], and we simply give an outline here. Any infinitesimal perturbation $\delta R$ of an orthogonal matrix $R$ which preserves orthogonality, i.e. $R^T R = I$, necessarily satisfies $R^T \delta R + (\delta R)^T R = 0$. This identifies the set $P$ and the tangent space description is immediate. Further, if $N$ is a matrix in the normal space, there must hold by definition

\[
\text{trace}(N^T P) = 0 \forall P \in T_{SO_3}(R).
\]

One can readily check that any matrix of the set $N_{SO_3}(R)$ satisfies this requirement, and, with slightly more work, that this set exhausts all such matrices.

We now prove that (ii.) the projection of $M$ on $T_{SO_3}(R)$ is given by $M_T = \frac{1}{2} M - \frac{1}{2} R M R^T$.

First, let $N_M = \frac{1}{2} M + \frac{1}{2} R M R^T$. Observe that

\[
R^T M_R = \frac{1}{2} R^T M + \frac{1}{2} M R^T R.
\]
is symmetric; therefore, $M_N \in N_{SO_3}(R)$ is normal to the tangent space $T_{SO_3}(R)$ at $R$. Furthermore, $\begin{bmatrix} R^T M T - M^T R \end{bmatrix} = 0$ is skew-symmetric; thus $M_T \in T_{SO_3}(R)$. Further because $M_P + M_N = M$, $M_T$ is the projection of $M$ on $T_{SO_3}(R)$.

### References


**Bomin Jiang** is a PhD candidate in Institute for Data, Systems, and Society, Massachusetts Institute of Technology (MIT). He received the B.E. degree with First Class Honours in Mechatronic Systems Engineering from The Australian National University in August 2014. He was awarded with a University Medal from the Australian National University upon completing his B.E. degree. In the Australian National University, the University Medals are not awarded in every disciplines every year; they are reserved only for students who complete Honours Degrees with outstanding achievements in both coursework and research. In September 2017, he was awarded with the MIT Presidential Fellowship. His current research interests include Networked Dynamical Systems, Game Theory and Econometrics.

**Brian D. O Anderson** (M’66–SM’74–F’75–LF’07) was born in Sydney, Australia, and educated at Sydney University in mathematics and electrical engineering, with PhD in electrical engineering from Stanford University in 1966. He is an Emeritus Professor at the Australian National University (having retired as Distinguished Professor in 2016), Distinguished Professor at Hangzhou Dianzi University, and Distinguished Researcher in National ICT Australia. His awards include the IEEE Control Systems Award of 1997, the 2001 IEEE James H Mulligan, Jr Education Medal, and the Bode Prize of the IEEE Control System Society in 1992, as well as several IEEE and other best paper prizes. He is a Fellow of the Australian Academy of Science, the Australian Academy of Technological Sciences and Engineering, the Royal Society, and a foreign member of the US National Academy of Engineering. He holds honorary doctorates from a number of universities, including Université catholique de Louvain, Belgium, and ETH Zürich. He is a past president of the International Federation of Automatic Control and the Australian Academy of Science. His current research interests are in distributed control, social networks and econometric modelling.
Fig. 1. Demonstration of the elementary two-agent localization problem

Fig. 2. An example of the actual and estimated trajectories used in the simulations

Hatem Hmam received his Ph.D. degree in electrical engineering from the University of Cincinnati, Ohio, in 1992. He joined the Defence Science and Technology Group, Australia, in late 1996 and is currently a senior research scientist in the Cyber and Electronic Warfare Division. His current research interests include signal processing, alternative positioning and navigation, and optimal sensor placement.

Fig. 3. Performance comparisons between algebraic method, SDP relaxation alone, and SDP with gradient descent (noiseless)

Fig. 4. Performance comparisons between algebraic method, SDP relaxation alone, and SDP with gradient descent (SNR=30)

Fig. 5. Performance comparisons between algebraic method, SDP relaxation alone, and SDP with gradient descent (SNR=20)
Fig. 6. Performance comparisons between algebraic method, SDP relaxation alone, and SDP with gradient descent (SNR=10)

Fig. 7. 3D plot of the UAV positions in global coordinate systems