Collaborative 3-D localization of robots with heterogeneous inter-robot relative measurements

Joseph Knuth and Prabir Barooah

***Abstract*—We propose a method for collaborative localization of multiple autonomous robots that fuses relative-pose measurements of each robot with various types of inter-vehicle relative measurements. We do not rely on the use of any maps, or the ability to recognize landmarks in the environment. Instead we assume that noisy relative measurements between pairs of robots - which can be one of any of the following: pose (position and orientation), orientation, position, bearing, distance - are intermittently available. We utilize the additional information about each robot's pose provided by these measurements to improve over self-localization (dead-reckoning) estimates. The proposed method is based on solving an optimization problem in a product manifold. A provably correct explicit gradient descent law on the manifold is provided. Unlike many previous approaches, the proposed algorithm is easily applicable to the 3-D case. The distributed algorithm we propose is also capable of handling a fully dynamic scenario where the neighbor relationships among robots are time-varying. Simulations - as well as experiments with a pair of ground robots - show that the errors in the localization estimates obtained using this algorithm are significantly lower than what is achieved when robots estimate their poses without cooperation. Additionally, the improvement of localization - over dead reckoning - for the various types of measurements considered allows us to examine cost-benefit trade-offs of the sensors required to obtain such measurements.**

I. Introduction

Localization is a crucial capability for any autonomous mobile robot. In recent years, there has been increased interest in utilizing not just one, but teams of autonomous mobile robots. Multi-robot teams are beneficial in many ways. In search and rescue operations, a group of robots can cover a larger area than a single robot. In hazardous conditions, the innate redundancy of a group of robots may be necessary to prevent catastrophic loss of mission capability.

Localization for autonomous robots can be accomplished using a variety of sensors. Some of the more common sensors include Inertial Measurement Units (IMUs), vision based sensors (cameras, LIDARs), and Global Positioning System (GPS). Though GPS provides global measurements of a robots position, GPS measurements may not be available in many situations, or may only be intermittently available. For example, a group of unmanned aerial vehicles (UAVs) operating in an urban environment may temporarily lose GPS measurements when the signal is blocked by large buildings. In such a situation, the global pose (position and orientation) can be obtained by integrating over the relative pose measurements obtained using IMUs or vision based sensors. This method of localization is referred to as *dead reckoning*.

Localization through dead reckoning can lead to a rapid growth in localization error [17]. When utilizing a team of robots, relative measurements between pairs of robots may be available. These provide additional information on the robots' pose that can be used to improve localization accuracy.

In this paper we propose a method for collaborative localization after a group of robots loses access to GPS. We assume all robots are equipped with proprioceptive sensors (vision, IMU, etc.) allowing each robot to measure its change in pose. We refer to these noisy measurements as *inter-time relative pose measurements*. Using these noisy measurements, each robot can perform localization through dead reckoning. In addition, we assume each robot is equipped with exteroceptive sensors, allowing intermittent noisy measurements of the relative pose, orientation, position, bearing or distance between pairs of robots. We refer to these measurements as *inter-robot relative measurements*. These inter-robot relative measurements provide additional information on the absolute pose of each robot. We propose both a centralized and distributed algorithm to perform collaborative localization by fusing the inter-time and inter-robot relative measurements to obtain an improved estimate of the absolute pose of every robot. In the distributed algorithm, communication is only necessary between pairs of robots for which an inter-robot relative measurement has been obtained.

A. Related Work

Collaborative localization has been considered in the context of simultaneous localization and mapping (SLAM). In one class of approaches, robots exchange local maps which are aligned and merged to improve robots' location estimates as well as to improve the maps; see [16, 2] and references therein. This requires the ability to identify common features in distinct maps generated by the robots. A method based on pose graphs is developed in [9] that does not require exchange of maps. In [6], robots exchange images and an implicit extended Kalman filter is used to update the state of each robot when a common feature is found.

Recognizing common landmarks in distinct maps is often challenging. In addition, exchanging image data or maps between robots requires high bandwidth communication. A second body of work therefore considers the collaborative localization problem as one in which only relative measurements (of pose, position, orientation etc.) between pairs of robots are obtained and used to improve accuracy over self-localization.

One of the early works on collaborative localization with relative measurements is [11], in which a stationary group of robots function as landmarks for the moving group and the group switch roles over time. Later works allow all robots to move simultaneously. Collaborative localization is often performed through Kalman filter or its extensions; see [20, 19, 8, 21] and references therein. In each case, collaborative localization was shown to improve over dead reckoning estimates. Other approaches include probabilistic methods, such as those by Fox et al. [4], Leung et al. [12], in which the pdf (probability density function) of robots' positions (or poses) are improved by fusing inter-robot measurements. Aragues et al. [3] propose a method for estimating the pose for a set of collaborating robots by decomposing the problem into different phases, each of which can be solved using linear estimation techniques. Howard et al. [5] proposed a nonlinear cost function that, when minimized, provides a maximum likelihood estimate of the pose for each robot. Knuth and Barooah [10] propose an algorithm for collaborative localization in 3-D by linearizing the relation between the pose measurements.

Most of the aforementioned papers assume relative pose measurements between robots. In many applications, these may be difficult or expensive to obtain. To that end, Rekleitis et al. [18] and Martinelli et al. [15] respectively extend the particle - and Kalman - filtering approaches for 2-D collaborative localization to include not only relative pose measurements, but also measurements of the relative orientation, position, bearing, and distance. In particular, Martinelli et al. [15] provides simulations that indicate any combination of these measurements, with the exception of distance alone, improves localization accuracy over dead reckoning alone.

Our work falls into the second category of work on collaborative localization discussed above, and is distinct from SLAM-type approaches.

B. Contributions

Most of the work listed above are applicable to the 2-D case: a robot's pose is described by three scalars: x, y coordinate and an angle θ. Though many of them can in principle be extended to the 3-D case, the extensions are far from straightforward. The Kalman filter based approaches of [20, 19, 8, 21] will require linearization due to the non-linear relationship between absolute position and relative orientation. The belief propagation approaches [4, 5, 12] will require computing and representing conditional densities on $SO(3)^1$, which is a challenging problem by itself. In fact, no details are provided for the extension to 3D case; all simulation and experimental results in [20, 19, 8, 21, 4, 5, 12, 3] are for the 2-D case. While the method in Knuth and Barooah [10] is applicable to the 3-D case, it is based on a linearization. Thus, no analytical guarantees are provided.

In contrast to the papers listed above, the method proposed here does not rely on any specific parameterization - and does not use linearization - of $SO(3)$. The algorithm is immediately implementable, and is presented, for the general 3-D case. We provide both a centralized and distributed algorithm for collaborative localization. In the latter, each robot only uses locally available measurements and communicates only with a small number of neighbors. The complexity of the computations performed by a robot is only a function of the number of its neighbors at any given time, not the total number of robots in the group. This makes the distributed algorithm scalable to arbitrarily large groups of robots. In addition, the communication complexity of the algorithm is small. At every update, a pair of neighboring robots needs to exchange only (i) their relative measurements and (ii) their current pose estimates. Since a pose, which is an element of $SE(3)$ [2], can be represented by 6 numbers, the pair of robots have to exchange at most 12 numbers.

Additionally, the method proposed here allows fusion of heterogeneous relative measurements between robots. Any type of measurements among the following: relative pose, relative orientation, relative bearing, relative position, and distance, can be used. We extend the examination of the relative merits of various types of measurements in [18, 15] to the 3-D case.

A weakness - especially compared to the filtering-based methods - is that the proposed method only provides an estimate of the pose, but not a measure of the associated uncertainty (such as covariance).

II. Problem Statement

A. The Collaborative Localization Problem

Consider a group of r mobile robots indexed by $i = 1, \ldots, r$. Time is measured by a discrete counter $k = 0, 1, 2, \ldots$. Each robot is equipped with a local, rigidly attached reference frame. We call the reference frame attached to robot i *frame* i. Robot i is said to be *localized* when an estimate is found for the Euclidean transformation $T \in SE(3)$ that relates a robots local reference frame to some absolute reference frame. This transformation is referred to as the robot's *absolute pose*.

Measurements of a robot's global pose (position and orientation) from GPS and compass are either not available or only rarely available. Instead, we assume that each robot is equipped with sensors (such as inertial sensors or vision based sensors) such that, at every time k, the robot is able to obtain a relative pose measurement with respect to its previous pose. We refer to these measurements as *inter-time relative pose measurements*. Additionally, they need not be obtained from a sensor alone. Instead, a measurement could also be the estimate obtained by fusing raw sensor measurements with predictions of the robot's motion from a dynamic/kinematic model.

In addition to these inter-time measurements, each robot is equipped with exteroceptive sensors so that the robot is able to uniquely identify each robot it can "see", and obtain a relative measurement for each such robot with respect to itself. When

[1] $SO(3)$ denotes the set of all bounded linear operators on the Euclidean space $\mathbb{R}^3$ that preserve the length of vectors and orientation of the space. For more information about the group $SO(3)$ and its properties, see Joshi [7].

[2] For the purposes of this paper, $SE(3)$, the Special Euclidean Group, can be thought of as the set $SO(3) \times \mathbb{R}^3$.

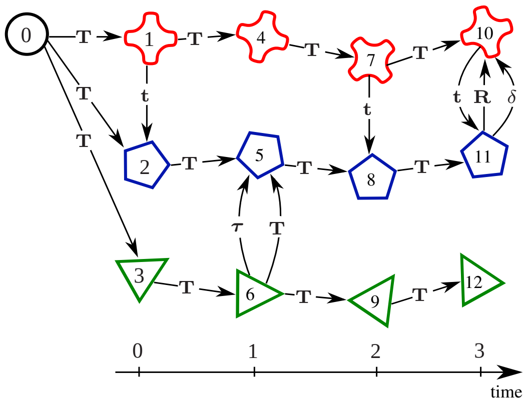

Fig. 1. A time history of three robots up to time $k = 3$ with inter-time and inter-robot relative pose measurements. Each (robot,time) pair is labeled with the corresponding node index from $\mathcal{V}_0(3)$. Arrows indicate edges, i.e., relative measurements, in $\mathcal{E}(3)$. Each edge is labeled to indicate the type of measurement. Robots 1 and 3 had GPS measurements at the initial time $k = 0$. Thereafter, no other GPS measurements were available.

robot i collects a measurement of robot j, it can be one of the following:

- **Relative pose**: The Euclidean transformation between the reference frame attached to robot i and the reference frame attached to robot j, expressed in robot i's reference frame. Denoted by the symbol $\mathbf{T}$.
- **Relative Orientation**: The element of $SO(3)$ that describes the change in orientation between frame i and frame j, expressed in frame i. Denoted by the symbol $\mathbf{R}$.
- **Relative position**: The vector in $\mathbb{R}^3$ that describes the change in position between robot i and robot j, expressed in frame j. Denoted by the symbol $\mathbf{t}$.
- **Relative bearing**: The vector of unit length that points from robot i to robot j, expressed in frame i. Denoted by the symbol $\boldsymbol{\tau}$.
- **Relative distance**: The distance between robot i and robot j. Denoted by the symbol δ.

We call these measurements *inter-robot relative measurements*.

The *collaborative localization problem* is then the problem of estimating the pose of every robot with respect to some common reference frame by utilizing the inter-time and inter-robot measurements. The situation above is best described in terms of a directed, time-varying, fully-labeled *graph* $\mathcal{G}(k) = (\mathcal{V}_0(k), \mathcal{E}(k), \ell(k))$ that shows how the noisy relative measurements relate to the absolute pose of each robot at every time step. The graph is defined as follows. For each robot $i \in \{1, \dots, r\}$ and each time $t \leq k$, a unique index (call it u) is assigned to the pair (i, t). How this indexing is done is immaterial. The set of these indices $\{1, \dots, rk\}$ define the set $\mathcal{V}(k)$ and the node set of the graph is defined as $\mathcal{V}_0(k) := \mathcal{V}(k) \cup \{0\}$. We then refer to the reference frame attached to robot i at time t as *frame* u. Node u is associated with the absolute pose of robot i at time t relative to some common reference frame, given by the Euclidean transformation between the common reference frame and frame u, expressed in the common frame. We call these poses *node variables* and denote them $\mathbf{T}_u$. The common reference frame with respect to which all node variables are expressed is associated with node 0. If the global pose of at least one robot is known at time 0, perhaps through the use of a GPS and compass, then node 0 can be associated with the global reference frame. When global pose measurements are not available, node 0 could correspond to the initial reference frame of one of the robots. In either case, estimating the node variables is equivalent to determining the robots' poses with respect to frame 0 (the reference frame associated with node 0). Node 0 is therefore called the *reference node*.

The set of *directed edges* at time k, denoted $\mathcal{E}(k)$, corresponds to the noisy inter-time and inter-robot measurements collected up to time k. That is, suppose robot i is able to measure robot j's relative pose at time $\bar{k}$, and let u, v be the nodes corresponding to robots i, j at time $\bar{k}$, respectively. Then the edge $e = (u, v)$ will be in $\mathcal{E}(k)$ for all $k \geq \bar{k}$. Similarly, each inter-time relative pose measurements of a robot also creates an edge in the graph. To delineate the type of measurement, a label from the set $\{$ $\mathbf{T}$ (pose), $\mathbf{R}$ (orientation), $\mathbf{t}$ (position), $\boldsymbol{\tau}$ (bearing), δ (distance) $\}$ is attached to each edge. The map from the set of edges to the set of labels is given by $\ell(k)$. Recall our example in which the edge $e = (u, v)$ is associated with a relative pose measurements. The label for edge e is then given by $\ell(k)(e) = \mathbf{T}$ (pose). The noisy relative measurement associated with each edge $e = (u, v) \in \mathcal{E}(k)$ is denoted by $\hat{\mathbf{T}}_{u\,v}, \hat{\mathbf{R}}_{u\,v}, \hat{\mathbf{t}}_{u\,v}, \hat{\boldsymbol{\tau}}_{u\,v},$ or $\hat{\delta}_{u\,v}$ for $\ell(k)(e) = \mathbf{T}$ (pose), $\mathbf{R}$ (orientation), $\mathbf{t}$ (position), $\boldsymbol{\tau}$ (bearing), and δ (distance) respectively.

The graph $\mathcal{G}(k)$ is called the *measurement graph* at time k. Figure 1 shows an example of the graph corresponding to the measurements collected by 3 robots up to time index 3.

To ensure at least one estimate exists for every robot at each time k, we make the following assumption.

Assumption 1. *All inter-time relative measurements are of the full relative pose and each robot has access to an estimate of its current absolute pose at time* 0.

This assumption will often hold in practice, as any robot capable of localizing itself using dead-reckoning must estimate the full relative pose between each time step. Under this assumption, an estimate for the pose of robot i at time time k (or equivalently, the value of the node variable $\mathbf{T}_u$, where node $u \in V(k)$ corresponds to the pair (i, k)) can be computed by composing the inter-time relative pose measurements found by robot i up to time k. This estimate is equivalent to robot i performing dead-reckoning.

Often many more edges are present in the graph $\mathcal{G}(k)$ than those necessary to form a single estimate of the node variables, which include the poses of the robots at time k. *The goal of the collaborative localization problem is to utilize all edges in the graph* $\mathcal{G}(k)$ *to obtain estimates of the robots' poses at time* k *that is more accurate than that possible from dead reckoning (which does not use the additional edges).* When the measurements are linearly related to the node variables, this can be accomplished by using the best linear unbiased estimator, as done in [3]. In our case, the relationship between the measurements and node variables is nonlinear.

In section III we propose a *centralized algorithm* to solve the collaborative localization problem that uses all the measurements in the graph $\mathcal{G}(k)$ to compute the "best" estimate for each node variable. In Section IV, we propose a *distributed algorithm* to compute an estimates of each robot's current pose. The distributed algorithm has the additional constraints that (i) at every time step, each robot must only be required to communicate with other robots it can "see" or that can "see" it, and (ii) memory, processor power, and communication bandwidth is limited. For the distributed algorithm, *we assume that the communication range is greater then the measurement range*. It is always possible to satisfy this assumption by dropping any measurements between robots that are unable to communicate.

III. Centralized Algorithm

In this section we present a solution to the collaborative localization problem where all the relative measurements are instantly available to a central processor at each time k. The centralized solution naturally leads to a distributed scheme, which will be described in the next section.

The problem of estimating the robots' current poses at time k is embedded in the problem of estimating all the node variables of the measurement graph $\mathcal{G}(k)$

$$\{\mathbf{T}\}_{\mathcal{V}(k)} := \{\mathbf{T}_i \in SE(3) : i \in \mathcal{V}(k)\} \tag{1}$$

using the robots' past noisy relative measurements (both inter-time and inter-robot), given by the set

$$\{\hat{M}\}_{\mathcal{E}(k)} := \left\{\hat{M}_e : e = (i,j) \in \mathcal{E}(k)\right\}. \tag{2}$$

where $\hat{M}_e$ is an element of $SE(3)$, $SO(3)$, $\mathbb{R}^3$, $\mathbb{S}^2$, or $\mathbb{R}$ depending on the type of measurement on the edge e. The so-called 2-sphere $\mathbb{S}^2$ is the unit sphere in $\mathbb{R}^3$. We estimate the node variables by minimizing a cost function of these variables that measures how well a given set of node variables (absolute poses) explains the noisy measurements. The initial condition for each node variable $\mathbf{T}_i$, $i \in \mathcal{V}(k)$ is given by the dead-reckoning estimate guaranteed by Assumption 1.

The cost function we propose is of the form

$$f(\{\mathbf{T}\}_{\mathcal{V}(k)}) := \frac{1}{2} \sum_{(i,j)=e \in \mathcal{E}(k)} g_e(\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{t}_j) \tag{3}$$

where $g_e(\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{t}_j)$ is designed to be a measure of how well the noisy relative measurement associated with the edge $e = (i,j)$ fit estimates of the relevant node variables $\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{t}_j$. Here $\mathbf{R}_i \in SO(3)$ and $\mathbf{t}_i \in \mathbb{R}^3$ denote the rotation and translation components that make up the node variable $\mathbf{T}_i$. For an illustrative example, consider the relative rotation measurement $\hat{\mathbf{R}}_{i\,j}$ associated with the edge $e = (i,j)$. By definition, $\hat{\mathbf{R}}_{i\,j}$ is a measurement of the rotation between frame i and frame j, expressed in frame i. This same rotation is also equal to $\mathbf{R}_j^T \mathbf{R}_i$, where $\mathbf{R}_i^T$ is the adjoint of the operator $\mathbf{R}_i$. Therefore, when no noise is present in the measurement, $\hat{\mathbf{R}}_{i\,j}$ is equal to $\mathbf{R}_j^T \mathbf{R}_i$, but not when measurements are noisy. The distance between these two expressions, measured by a suitable metric on $SO(3)$, provides a measure of how well any given estimates for $\mathbf{R}_i$ and $\mathbf{R}_j$ fit the measurement. Therefore, a suitable cost function for a relative orientation measurement $\hat{\mathbf{R}}_{i\,j}$ associated with the edge $e = (i,j)$ is given by

$$g_e(\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{R}_i) = d^2(\hat{\mathbf{R}}_{i\,j}, \mathbf{R}_i^T \mathbf{R}_j), \tag{4}$$

where the distance $d(\,\cdot\,,\,\cdot\,)$ in $SO(3)$ is given by the Riemannian distance

$$d(A,B) = \sqrt{-\frac{1}{2}\operatorname{Tr}\left(\log^2(A^T B)\right)}, \quad A, B \in SO(3). \tag{5}$$

Using arguments similar to the one presented above for orientation measurements, appropriate cost functions for all measurement types are constructed:

$$g_e(\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{t}_j) = \begin{cases} d^2(\hat{\mathbf{R}}_{i\,j}, \mathbf{R}_i^T \mathbf{R}_j) + \|\hat{\mathbf{t}}_{i\,j} - \mathbf{R}_i^T(\mathbf{t}_j - \mathbf{t}_i)\|^2 & \text{if } \ell(k)(e) = \mathbf{T} \\ d^2(\hat{\mathbf{R}}_{i\,j}, \mathbf{R}_i^T \mathbf{R}_j) & \text{if } \ell(k)(e) = \mathbf{R} \\ \|\hat{\mathbf{t}}_{i\,j} - \mathbf{R}_i^T(\mathbf{t}_j - \mathbf{t}_i)\|^2 & \text{if } \ell(k)(e) = \mathbf{t} \\ \|\left(\hat{\boldsymbol{\tau}}_{i\,j}\|\mathbf{t}_j - \mathbf{t}_i\|\right) - \mathbf{R}_i^T(\mathbf{t}_j - \mathbf{t}_i)\|^2 & \text{if } \ell(k)(e) = \boldsymbol{\tau} \\ \|\left(\hat{\delta}_{i\,j} - \|\mathbf{t}_j - \mathbf{t}_i\|\right)\|^2 & \text{if } \ell(k)(e) = \delta \end{cases} \tag{6}$$

If the relative measurements were completely error free, the minimum value of the cost function f would be 0. By minimizing the cost function, we expect to find an improved estimate for the absolute pose of each robot over what can be obtained through dead reckoning alone.

Finding the minimum of a function defined over a vector space has been studied extensively. However the function $f(\,\cdot\,)$ in (3) is defined on a curved surface, specifically, the product *Riemannian Manifold* $(SO(3) \times \mathbb{R}^3)^{n(k)}$ where $n(k) = |\mathcal{V}(k)|$, the cardinality of the set $\mathcal{V}(k)$. One option for this optimization is to use a parameterization of the rotations using, say, 3×3 rotation matrices or unit quaternions, and then embedding the manifold in an vector space of higher dimension. Optimization techniques applicable to vector spaces can then be used, with the constraints on the parameterization of rotations appearing as Lagrange multipliers. This however, leads to an unnecessary increase in the dimensionality of the problem, and will lead to needlessly complicated and time consuming calculations to carry out the optimization. Even when a parameterization is chosen that doesn't lead to an increase in dimensionality, such as Euler angles or Hopf Coordinates [22], the optimization step still requires more computation then what is possible when the geometry of $SO(3)$ is utilized. Additionally, any such parameterization will encounter problems (failing to be bijective) near its boundaries. Our goal is to find a provably correct algorithm that utilizes the geometry of the space without relying on any particular parameterization. We accomplish this through use of a gradient descent algorithm on the product manifold.

Given a smooth real valued function f defined on a manifold M, the gradient of f at $p \in M$, denoted $grad\, f(p)$, is a vector in the tangent space of M at p, denoted T_pM. Just as in Euclidean Space, $grad\, f(p)$ points in the direction of greatest rate of increase of f. Using linearity of the gradient operator, finding the gradient of (3) reduces to finding the gradient of (6) for each of the 5 measurement types. As an illustrative example, the gradient for the edge cost of a pose measurement $(\hat{\mathbf{R}}_{i\,j}, \hat{\mathbf{t}}_{i\,j})$ associated with the edge $e = (i, j)$ at a point $p = (\mathbf{R}_1, \mathbf{t}_1, \ldots, \mathbf{R}_n, \mathbf{t}_n) \in \left(SO(3) \times \mathbb{R}^3\right)^n$ is given by

$$\begin{aligned} grad\ g_e(p) = \Big(& grad\ g_e(\mathbf{R}_1), grad\ g_e(\mathbf{t}_1), \ldots, \\ & grad\ g_e(\mathbf{R}_n), grad\ g_e(\mathbf{t}_n)\Big) \end{aligned} \tag{7}$$

where

$$grad\, g_e(\mathbf{R}_k) = \begin{cases} -2\mathbf{R}_k\Big(\log(\mathbf{R}_k^T\mathbf{R}_j\hat{\mathbf{R}}_{i\,j}^T) \\ \quad +\mathbf{R}_k^T(\mathbf{t}_j - \mathbf{t}_i)\hat{\mathbf{t}}_{i\,j}^T - \hat{\mathbf{t}}_{i\,j}(\mathbf{t}_j - \mathbf{t}_i)^T\mathbf{R}_k\Big) & \text{if } k = i \\ -2\mathbf{R}_k \log(\mathbf{R}_k^T\mathbf{R}_i\hat{\mathbf{R}}_{i\,j}) & \text{if } k = j \\ 0 & \text{o.w.} \end{cases}$$

$$grad\ g_e(\mathbf{t}_k) = 2I_{ij}(k)(\mathbf{t}_i + \mathbf{R}_i\hat{\mathbf{t}}_{i\,j} - \mathbf{t}_j)$$

for $I_{ij}(k)$ equals 1 if $k = i$, -1 if $k = j$, and 0 otherwise. Due to space constraints, the full gradient is omitted, but can be derived using techniques found in [13].

Minimizing a function f using gradient descent requires that during each iteration, the current estimate must be updated by moving in the direction of $-\eta\, grad\, f$ for some appropriate scalar η. On a Riemannian manifold, this requires the notion of *parallel transport*. The parallel transport map at a point $p = (\mathbf{R}_1, \mathbf{t}_1, \ldots, \mathbf{R}_n, \mathbf{t}_n) \in (SO(3) \times \mathbb{R}^3)^n$, denoted by $\exp_p$, is given by

$$\begin{aligned} \exp_p(\xi) = (\mathbf{R}_1 \exp(\mathbf{R}_1^T\xi_{\mathbf{R}_1}), \mathbf{t}_1 + \xi_{\mathbf{t}_1}, \ldots, \\ \mathbf{R}_n \exp(\mathbf{R}_n^T\xi_{\mathbf{R}_n}), \mathbf{t}_n + \xi_{\mathbf{t}_n}) \end{aligned} \tag{8}$$

where $\xi = (\xi_{\mathbf{R}_1}, \xi_{\mathbf{t}_1}, \ldots, \xi_{\mathbf{R}_n}, \xi_{\mathbf{t}_n})$ is an element of the *tangent space* $T_p\left[(SO(3)\times\mathbb{R}^3)^n\right] = T_{\mathbf{R}_1}SO(3)\times\cdots\times T_{\mathbf{t}_n}\mathbb{R}^3$, and the $\exp(\,\cdot\,)$ function appearing in the right hand side of (8) is the Lie-group exponential map [13].

The gradient descent update law is then given by

$$p_{t+1} = \exp_{p_t}(-\eta_t grad\, f(p_t)), \quad t = 0, 1, \ldots, \tag{9}$$

where $\eta_t \geq 0$ is chosen to be the *Armijo step size*. More detail on gradient descent algorithms on manifolds, and on the Armijo step size in particular, can be found in [1].

Using the update law given in (9), a gradient descent is performed, terminating when the norm of the gradient falls below some user specified threshold. Theorem $4.3.1$ in [1] guarantee that this algorithm converges to a critical point of the cost function f defined in (3).

It should be noted that the algorithm presented above is independent of the parameterization used to represent rotations. One could use unit quaternions, 3×3 rotation matrices, etc.

IV. Distributed Algorithm

We now propose an algorithm for collaborative localization that each robot executes. This distributed algorithm only requires measurements that the robot can collect with on-board sensors and communication with nearby robots. It also requires a limited memory and processor power.

For each robot i, let $N_i^{(+)}(k)$ denote the set of all robots $j \in \{1, \ldots, r\}$ such that, at time k, robot i can measure its relative pose, orientation, position, bearing, or distance with respect to j. Let $N_i^{(-)}(k)$ denote the set of all robots $j \in \{1, \ldots, r\}$ such that, at time k, robot j can measure its relative pose, orientation, position, bearing, or distance with respect to robot i. The *neighbors* of robot i at time k are then given by the set $N_i(k) = N_i^{(+)}(k) \cup N_i^{(-)}(k)$. *We assume that each robot can communicate with its neighbors during each time step.*

To facilitate the description of the distributed algorithm, consider the time varying *local* measurement graph $\mathcal{G}_i(k) = (\mathcal{V}_i(k), \mathcal{E}_i(k), \ell_i(k))$ of robot i, whose node set is simply the neighbors of i at time k along with the reference node 0 and i itself: $\mathcal{V}_i(k) = N_i(k) \cup \{0, i\}$. The edges of $\mathcal{G}_i(k)$ correspond to the inter-robot measurements at time k between i and its neighbors, along with an edge $e = (0, j)$ for each $j \in \mathcal{V}_i(k)$. Thus if robot i can "see" robot j at time k, then $e = (i, j) \in \mathcal{E}_i(k)$. Similarly, if j can see i, $e = (j, i) \in E_i(k)$. Thus each edge $(p, q) \in \mathcal{G}_i(k)$ (where $i = p$ or q) is associated with the noisy inter-robot relative measurement between robots p and q at time k. The additional edges $e = (0, j)$, $j \in \mathcal{V}_i(k)$ are associated with the initial estimate for each robot's absolute pose, denoted $\hat{\mathbf{T}}_{0\,j}$. Each robot j obtains $\hat{\mathbf{T}}_{0\,j}$ at time k by concatenating the robot's pose estimate obtained at time $k-1$ with the noisy inter-time relative pose measurement describing the robots motion from $k-1$ to k. This estimate is then used as the measurement associated with edge $e = (0, j)$. The edges $e = (0, j) \in \mathcal{E}_i(k)$ for $j \in \mathcal{V}_i(k)$ ensures that $\mathcal{G}_i(k)$ satisfies Assumption 1.

The distributed algorithm works as follows. At each time k, every robot $i \in \{1, \ldots, r\}$ forms an initial estimate of its absolute pose $\hat{\mathbf{T}}_{0\,i}$ as described above and measures the inter-robot relative pose, orientation, position, bearing, or distance (or some combination there in) for each robot $j \in N_i^{(+)}(k)$. Robot i then transmits to each robot $j \in \mathcal{V}_i(k)$ the estimated absolute pose $\hat{\mathbf{T}}_{0\,i}$ along with all inter-robot relative measurements found by robot i between robots i and j at time k. Robot i receives in turn robot j's estimate of its current absolute pose $\hat{\mathbf{T}}_{0\,j}$. Similarly, for each robot $j \in N_i^{(-)}(k)$ robot i will receive $\hat{\mathbf{T}}_{0\,j}$ along with the inter robot relative measurements between robots i and j found by robot j and transmit to robot j the estimate $\hat{\mathbf{T}}_{0\,i}$. Robot i then executes the centralized algorithm described in the previous section to the local measurement graph $\mathcal{G}_i(k)$. After the computation, only the estimated value for the node variable $\mathbf{T}_i$ is retained. No other value need be stored in robot i's local memory. Note that if robot i has no neighbors at time k, the distributed collaborative localization algorithm is equivalent to performing self-localization from

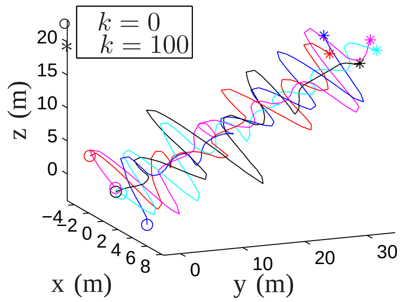

Fig. 2. The 3D trajectories for robots 1 through 5, used in all simulations.

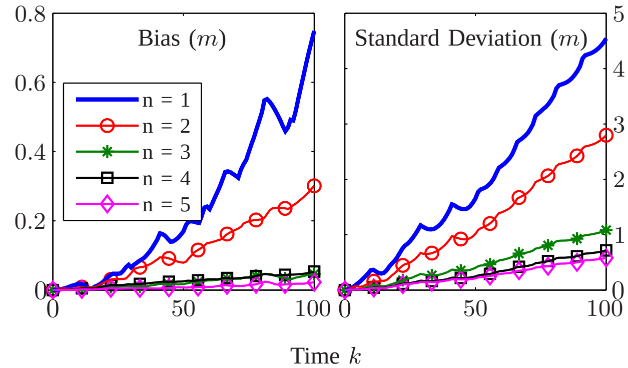

Fig. 3. Simulation: The bias and standard deviation in position error for robot 1 when the distributed algorithm is applied to a group of $1, 2, 3, 4$ or 5 robots. All inter-robot measurements are of relative pose. $n = 1$ implies robot 1 was localized using dead reckoning alone.

inter-time relative measurements. Since the distributed algorithm is simply the centralized algorithm applied to a local measurement graph, it inherits the correctness property of the centralized algorithm as well.

V. Simulation Results

In this section we present simulations studying the improvement in accuracy when the distributed algorithm is used. For each scenario discussed blow, the bias and variance in position error is estimated through the use of a Monte Carlo simulation with $1,000$ samples. The distributed, rather then the centralized, algorithm was chosen because it is more practical to implement in a real-world application.

A. With relative pose measurements

We first study how the number of robots affects localization accuracy. A group of robots was simulated traveling along the distinct zig-zag paths in 3-D space, shown in Figure 2, so that all three translational and rotational coordinates varied along time for each robot. Two robots can obtain relative pose measurements at time k if the Euclidean distance between them at that time is less than $7\ m$. Furthermore, 25% of these potential measurements were dropped, simulating random failure and insuring the measurement graph would not be symmetric. The rotation measurements for each relative pose (both inter-robot and inter-time) were corrupted by independent identically distributed (i.i.d.) unit quaternions drawn from a Von Mises-Fisher distribution [14] centered around the zero-rotation quaternion and with a concentration parameter of $10,000$. Noise in the relative translation measurements was simulated by adding i.i.d zero-mean normal random variables with covariance matrix $I_{3\times 3} \times 10^{-6}\ m^2$.

Simulations for robot teams of size $1, 2, 3, 4$ and 5 were carried out. When only one robot is present in the team, collaborative localization is equivalent to self-localization without the aid of any inter-robot relative pose measurement.

The position estimation error of robot i is $\mathbf{e}_i(k) := \hat{\mathbf{t}}_i(k) - \mathbf{t}_i(k)$, where $\mathbf{t}_i(k)$ is its global position at time k and $\hat{\mathbf{t}}_i(k)$ is the estimate of this position. The bias in the position estimation error of robot i is defined as $\mathbb{E}\left[\|\mathbf{e}_i(k)\|\right]$, where $\|\cdot\|$ refers to 2-norm, and the standard deviation as $\sqrt{\mathrm{Tr}\left(Cov(\mathbf{e}_i(k), \mathbf{e}_i(k))\right)}$. The bias/std. dev for the robots are shown in Figure 3, both of which show significant improvement with distributed collaborative localization over self-localization. This is evident even for a team of only two robots. As the number of robots in the team increases, the localization error of robot 1 decreases. The improvement in accuracy however, shows a diminishing return with increasing team size.

B. With various relative measurements

In this section, in addition to inter-robot relative pose measurements, we also consider inter-robot relative measurements of orientation, position, bearing, and distance. To examine the effect that measurement type has on localization accuracy, we let all inter-robot relative measurements be of the same type. In each simulation, 5 robots travel along distinct zig-zag paths. Errors in the pose, rotation and position measurements are induced as in section V-A. Noise in the bearing only measurements is induced by rotating the true bearing through the application of a unit quaternion generated from an i.i.d. Von Mises-Fisher distribution. The noise in the distance measurements is normally distributed. The bias and standard deviation in position error for robot 1 for each simulation are reported in figure 4. A similar trend was seen in orientation error, but not presented here due to lack of space.

Improvement over single robot localization in both bias and standard deviation occurs for all measurement types, with the exception of distance only measurements. While inter-robot distance measurements improve the standard deviation of the position estimates, they have little effect on the bias. As expected, full relative pose provides the most benefit to localization accuracy. However, other measurement types also leads to improvement over self-localization. For the scenario and time duration considered, the three types of relative measurements - bearing, position and orientation - seem to have somewhat similar degree of benefit.

VI. Experimental results

In this section we present results for an experiment conducted using the two Pioneer P3-DX robots shown in Figure 5. Each robot was equipped with a calibrated monocular Prosillica EC 1020 camera and wheel odometers. Measurements

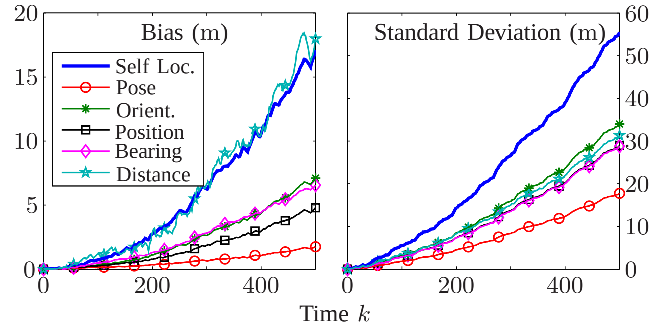

Fig. 4. Simulation: The bias and standard deviation in position error for robot 1 when the distributed algorithm is applied to a group of 5 robots utilizing various types of inter-robot relative measurements. The label "Self Loc." refers to a robot using dead reckoning to localize, without the use of any inter-robot measurements. The labels "Pose", "Orient." (Orientation), "Position", "Bearing" and "Distance" indicate the group of robots used inter-robot measurements of the respective types to perform collaborative location using the distributed algorithm.

from these sensors were fused to find the noisy inter-time relative pose measurements. Each robot is additionally equipped with a target allowing the on-board cameras to measure the inter-robot relative pose by exploiting the known geometry of each target. The true pose of each robot was determined using an overhead camera capable of tracking each robot's target. Noisy inter-robot relative pose measurements were obtained every 0.2 seconds.

All robots moved in straight lines with their paths approximately parallel. Six different pose estimates of the robots were obtained. The first estimate was obtained from dead reckoning. The remaining 5 estimates utilized the distributed collaborative localization algorithm with the inter-robot noisy relative pose measurements projected such that all were measurements of one of the following: full pose, orientation only, position only, bearing only, or distance only. The resulting global position estimates, along with the true positions, for robot 1 are reported in Figure 6. Simulations presented in section V indicate that we should see a significant improvement in localization accuracy even in this small team, and the experimental results are consistent with that conclusion. A distinct improvement in localization accuracy is seen when collaborative localization is performed for all but the distance only measurements. Because figure 6 is only a single realization of the robot's motion, the bias in position estimation is visible, while the variance is not.

The experimental results thus reinforce the trends seen in section V-B. Specifically, when the collaborative localization algorithm is used in conjunction with inter-robot relative measurements that are only of distance, the bias in the position estimation error is similar to that seen when the robot localizes without collaboration. For all other types of relative measurements, there is improvement with collaborative localization over self localization.

Fig. 5. Two Pioneer P3-DX robots equipped cameras and targets. Robot 1 is shown on the left, while robot 2 is on the right.

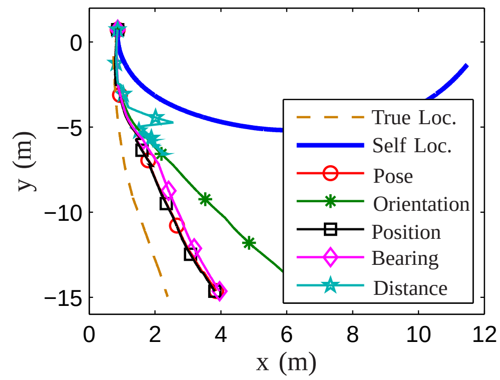

Fig. 6. Experimental: A plot of the location of robot 1 in the overhead camera reference frame when both robots move in a straight line. The true path (found using the overhead camera), estimated path using self localization, and estimated path using the distributed collaborative localization algorithm are all reported.

VII. Conclusion and Future Work

We introduced a distributed algorithm for estimating the full 3-D pose of multiple robots with time-varying neighbor relationships in which only communication between neighboring robots is required. The novel contributions of this work compared to much of the earlier work on collaborative localization are (i) ability to perform 3-D localization, (ii) ability to handle a time-varying network of robots, and (iii) ability to simultaneously fuse various types of inter-robot relative measurements (pose, orientation, bearing, position, and distance). While some prior works addresses each of these issues separately, we are not aware of any existing algorithm that has all three features.

Simulations as well as experiments show a significant increase in localization accuracy with the distributed collaborative localization algorithm over self-localization. The improvement is significant even for a small team of robots, with diminishing returns for an increasing number of robots. Experimental results verify that indeed significant accuracy improvement can be achieved even with two robots. Improvements were largest when all inter-robot relative measurements are of the relative pose; but significant improvements were observed also with other types of measurements. Only distance measurements were observed to not have much benefit, especially in the bias. This is consistent with the trend observed for the 2-D case in [15].

An interesting observation from the simulations and experiments is that relative measurements of bearing leads to an localization accuracy similar to that from relative measurements of orientation or position. Though further studies are required to establish if is true in general, the implications for cost vs. performance trade-off is obvious if true. While only a single camera is necessary to measure relative bearing of a neighboring robot, where stereo vision is necessary to measure the full relative orientation or position, and that too is susceptible to large measurement noise.

A study we could not present here due to space limitations is when the type of inter-robot relative measurement between the same pair of robots changes with time. An important future research direction is to develop a method that can provide not only estimates of the pose but also a measure of the associated uncertainty. Future studies will also address issues such as identification of the neighboring robots, and rejection of outliers in the relative measurements.

REFERENCES

[1] P. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008.

[2] Lars A. A. Andersson and J. Nygards. C-SAM : Multi-robot SLAM using square root information smoothing. In *IEEE International Conference on Robotics and Automation*, pages 2798–2805, 2008.

[3] R. Aragues, L. Carlone, G. Calafiore, and C. Sagues. Multi-agent localization from noisy relative pose measurements. In *IEEE International Conference on Robotics and Automation*, 2011.

[4] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous robots*, 8:325–344, 2000.

[5] A. Howard, M.J. Matark, and G.S. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002.*, 2002.

[6] Vadim Indelman, Pini Gurfil, Ehud Rivlin, and Hector Rotstein. Distributed vision-aided cooperative localization and navigation based on three-view geometry. In *IEEE Aerospace Conference*, 2011.

[7] A. W. Joshi. *Elements of Group Theory for Physicists*. John Wiley & Sons Inc., third edition, 1982.

[8] Nadir Karam, Frederic Chausse, Romuald Aufrere, and Roland Chapuis. Localization of a group of communicating vehicles by state exchange. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[9] Been Kim, M. Kaess, L. Fletcher, J. Leonard, A. Bachrach, N. Roy, and S. Teller. Multiple relative pose graphs for robust cooperative mapping. In *IEEE International Conference on Robotics and Automation*, 2010.

[10] J. Knuth and P. Barooah. Distributed collaborative localization of multiple vehicles from relative pose measurements. In *47th Annual Allerton Conference on Communication, Control, and Computing*, pages 314 –321, Allerton, IL, October 2009. doi: 10.1109/ALLERTON.2009.5394785.

[11] R. Kurazume, S. Nagata, and S. Hirose. Cooperative positioning with multiple robots. In *IEEE International Conference on Robotics and Automation*, pages 1250–1257, 1994.

[12] Keith Y.K. Leung, T.D. Barfoot, and H. Liu. Decentralized localization of sparsely-communicating robot networks: A centralized-equivalent approach. *IEEE Transactions on Robotics*, 26(1):62 –77, 2010.

[13] Yi Ma, Jana Košecká, and Shankar Sastry. Optimization criteria and geometric algorithms for motion and structure estimation. *International Journal of Computer Vision*, 44:219–249, 2001. ISSN 0920-5691.

[14] Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. Wiley series in probability and statistics. Wiley, 2000.

[15] A. Martinelli, F. Pont, and R. Siegwart. Multi-robot localization using relative observations. In *IEEE International Conference on Robotics and Automation*, 2005.

[16] K. Ni, D. Steedly, and F. Dellaert. Tectonic SAM: Exact, out-of-core, submap-based SLAM. In *IEEE International Conference on Robotics and Automation*, page 16781685, 2007.

[17] Clark F. Olson, Larry H. Matthies, Marcel Schoppers, and Mark W. Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4): 215–229, June 2003.

[18] I.M. Rekleitis, G. Dudek, and E.E. Milios. Multi-robot cooperative localization: a study of trade-offs between efficiency and accuracy. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2690 – 2695, 2002.

[19] S.I. Roumeliotis and G.A. Bekey. Distributed multirobot localization. *IEEE Transactions on Robotics and Automation*, 18(5):781 – 795, oct 2002.

[20] C. Sanderson. A distributed algorithm for cooperative navigation among multiple mobile robots. *Advanced Robotics*, 12(4):335 – 349, 1998.

[21] R. Sharma and C. Taylor. Cooperative navigation of mavs in gps denied areas. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2008.

[22] Anna Yershova, Swati Jain, Steven LaValle, and Julie Mitchell. Generating uniform incremental grids on SO(3) using the hopf fibration. *The International Journal of Robotics Research*, 29, 2010.