Title

Tim Woodbury

I. Rigid body cooperative carry

The problem is assumed to consist of three rigid bodies; a naturally inert body, labelled body 0, that is to be transported, and quadrotors, which are labelled i and j. It is assumed that each quadrotor is tethered to body 0, and that the tether has no slack, although its orientation is not necessarily fixed relative to body 0. If the relative orientations of the tethers and the position of the anchor point on the central body for each tether are known, then the position of each quadrotor relative to the central body is also. The attitudes of the quadrotors are assumed to be unconstrained by the tether. The system has 16 degrees-of-freedom overall; the attitudes of all three bodies (9), the inertial position of any single body (3), and two DOF relating the orientation of each tether to the body (4).

It has been stated that the inertial position of any one body, combined with the knowledge of the tether orientations, is sufficient to localize all three bodies. It is not yet clear whether it is preferable to use the inertial position of the individual quadrotor or the central body, as the former is more natural for vehicle state estimation, but the latter may be more simpler in terms of representing the position of all three bodies. For now, the central body position will be used.

Define the following reference frames: body-fixed reference frames o^+, i^+, and j^+ attached to the central body and to quadrotors i and j. These frames are assumed to be collocated with the body centers of gravity. In addition, define frames a_i^+ and a_j^+; these frames are defined by transformation from o^+ such that the 1 axis of a_i^+ and a_j^+ lies along the vector from the anchor point on the central body to quadrotors i and j, respectively. The following values are assumed known: $[\mathbf{r}_{ai/o}]_o\ [\mathbf{r}_{aj/o}]_o$, the anchor point of the tethers for vehicles i and j relative to the central body center of mass in the central body coordinate system, which is fixed; and the tether lengths l_i and l_j. The key position-level variables are:

- $\mathbf{r}_o$: the inertial position of body 0

- $\mathbf{q}_o$: the attitude transformation from the inertial to the body 0 reference frames

- $\mathbf{q}_{ak}$: the attitude transformation from the body 0 reference frame to the tether k frame, $k \in \{i, j\}$

- $\mathbf{q}_k$: the attitude transformation from the inertial to the body k reference frames, $k \in \{i, j\}$

The subsequent paragraphs develop, in vector form, the governing kinematic and dynamic equations. $\boldsymbol{\omega}_k$ is used to indicate the angular velocity of frame k^+ with respect to the inertial reference frame and its time rate in frame k^+ is given by $^k\frac{d\boldsymbol{\omega}_k}{dt} = \dot{\boldsymbol{\omega}}_k$.

In writing the forces acting on each body, a generic quadrotor model with forces and moments $\mathbf{f}^q$ and $\mathbf{l}^q$ is assumed for each vehicle. The tension forces acting on the central body from tethers i and j are denoted $\mathbf{t}_i$ and $\mathbf{t}_j$ and are given in the tether frames by, e.g., $\mathbf{t}_i = t_i\hat{\mathbf{a}}_{i_1}$. The tension forces are ideal constraint forces and may be omitted in a Kanian or LaGrangian problem formulation. The forces acting on each body are given by:

$$\mathbf{f}_0 = \mathbf{t}_i + \mathbf{t}_j - m_0 g\hat{\mathbf{n}}_1 \tag{1}$$
$$\mathbf{f}_i = -\mathbf{t}_i + \mathbf{f}_i^q \tag{2}$$
$$\mathbf{f}_j = -\mathbf{t}_j + \mathbf{f}_j^q \tag{3}$$

The moments are given by:

$$\mathbf{l}_0 = \mathbf{r}_{ai/0} \times \mathbf{t}_i + \mathbf{r}_{aj/0} \times \mathbf{t}_j \tag{4}$$

$$\mathbf{l}_i = \mathbf{l}_i^q \tag{5}$$

$$\mathbf{l}_j = \mathbf{l}_j^q \tag{6}$$

Generally, the position vector to each body is given by:

$$\mathbf{r}_0 = \mathbf{r}_0 \tag{7}$$

$$\mathbf{r}_i = \mathbf{r}_0 + \mathbf{r}_{ai/0} + \mathbf{r}_{i/ai} \tag{8}$$

$$\mathbf{r}_j = \mathbf{r}_0 + \mathbf{r}_{aj/0} + \mathbf{r}_{j/aj} \tag{9}$$

The time rates of the position vectors are:

$$\frac{d\mathbf{r}_0}{dt} = \mathbf{v}_0 \tag{10}$$

$$\mathbf{v}_i = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_{ai/o} + \boldsymbol{\omega}_{ai} \times \mathbf{r}_{i/ai} \tag{11}$$

$$\mathbf{v}_j = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_{aj/o} + \boldsymbol{\omega}_{aj} \times \mathbf{r}_{j/aj} \tag{12}$$

The time rates of the velocity vectors are:

$$\frac{d\mathbf{v}_0}{dt} = {}^0\frac{d\mathbf{v}_0}{dt} + \boldsymbol{\omega}_0 \times \mathbf{v}_0 \tag{13}$$

$$\frac{d\mathbf{v}_i}{dt} = \frac{d\mathbf{v}_0}{dt} + \dot{\boldsymbol{\omega}}_0 \times \mathbf{r}_{ai/o} + \boldsymbol{\omega}_0 \times (\boldsymbol{\omega}_0 \times \mathbf{r}_{ai/o}) + \dot{\boldsymbol{\omega}}_{ai} \times \mathbf{r}_{i/ai} + \boldsymbol{\omega}_{ai} \times (\boldsymbol{\omega}_{ai} \times \mathbf{r}_{i/ai}) \tag{14}$$

$$\frac{d\mathbf{v}_j}{dt} = \frac{d\mathbf{v}_0}{dt} + \dot{\boldsymbol{\omega}}_0 \times \mathbf{r}_{aj/o} + \boldsymbol{\omega}_0 \times (\boldsymbol{\omega}_0 \times \mathbf{r}_{aj/o}) + \dot{\boldsymbol{\omega}}_{aj} \times \mathbf{r}_{j/aj} + \boldsymbol{\omega}_{aj} \times (\boldsymbol{\omega}_{aj} \times \mathbf{r}_{j/aj}) \tag{15}$$

The attitudes of the three bodies are parameterized independently of one another:

$$\dot{\mathbf{q}}_0 = [B(\mathbf{q}_0)]\boldsymbol{\omega}_0 \tag{16}$$

$$\dot{\mathbf{q}}_i = [B(\mathbf{q}_i)]\boldsymbol{\omega}_i \tag{17}$$

$$\dot{\mathbf{q}}_j = [B(\mathbf{q}_j)]\boldsymbol{\omega}_j \tag{18}$$

$$\dot{\mathbf{q}}_{ai} = [B(\mathbf{q}_{ai})]\boldsymbol{\omega}_{ai} \tag{19}$$

$$\dot{\mathbf{q}}_{aj} = [B(\mathbf{q}_{aj})]\boldsymbol{\omega}_{aj} \tag{20}$$

$$\mathbf{I}_0\dot{\boldsymbol{\omega}}_0 + \boldsymbol{\omega}_0 \times (\mathbf{I}_0\boldsymbol{\omega}_0) = \mathbf{l}_0 \tag{21}$$

$$\mathbf{I}_i\dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times (\mathbf{I}_i\boldsymbol{\omega}_i) = \mathbf{l}_i \tag{22}$$

$$\mathbf{I}_j\dot{\boldsymbol{\omega}}_j + \boldsymbol{\omega}_j \times (\mathbf{I}_j\boldsymbol{\omega}_j) = \mathbf{l}_j \tag{23}$$

A minimal set of position coordinates for this problem is: $[\mathbf{r}_0]_n$, $\mathbf{q}_0$, $\mathbf{q}_i$, $\mathbf{q}_j$, $\mathbf{q}_{ai}$, and $\mathbf{q}_{aj}$. (Assumes that each body's attitude is given by a three-parameter set and that $\mathbf{q}_{ai}$ and $\mathbf{q}_{aj}$ are given by two parameters each such as spherical coordinates.)

I.A. Planar simplification

Simpler equations of motion can be obtained by considering the planar subproblem. In this case, the position of the mass center of body 0 is given in the inertial frame as $[\mathbf{r}_0]_n = \begin{bmatrix} r_{0x} & r_{0y} & 0 \end{bmatrix}^T$. The location of each

anchor k is given by $[\mathbf{r}_{ak/0}]_0 = \begin{bmatrix} r_{akx} & r_{aky} & 0 \end{bmatrix}^T$, and $[\mathbf{r}_{k/ak}]_n = l_k \begin{bmatrix} \cos\theta_k & \sin\theta_k & 0 \end{bmatrix}^T$. We can write inertial-frame position vectors for each body as:

$$[\mathbf{r}_0]_n = \begin{bmatrix} r_{0x} & r_{0y} & 0 \end{bmatrix}^T \tag{24}$$

$$[\mathbf{r}_i]_n = \begin{bmatrix} r_{0x} \\ r_{0y} \\ 0 \end{bmatrix} + [C_{0/n}^T]\begin{bmatrix} r_{a1x} \\ r_{a1y} \\ 0 \end{bmatrix} + l_i \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \\ 0 \end{bmatrix} \tag{25}$$

Translational velocity expressions can be derived as follows:

$$[\mathbf{v}_0]_n = \begin{bmatrix} \dot{r}_{0x} & \dot{r}_{0y} & 0 \end{bmatrix}^T \tag{26}$$

$$[\mathbf{v}_i]_n = \begin{bmatrix} \dot{r}_{0x} \\ \dot{r}_{0y} \\ 0 \end{bmatrix} + [C_{0/n}^T][^\times\boldsymbol{\omega}_0]\begin{bmatrix} r_{a1x} \\ r_{a1y} \\ 0 \end{bmatrix} + l_i\omega_i \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \\ 0 \end{bmatrix} \tag{27}$$

The position-level coupling between the three bodies introduces considerable complexity into the resulting expressions. Translational acceleration can be written as:

$$[\mathbf{a}_0]_n = \frac{d[\mathbf{v}_0]_n}{dt} \tag{28}$$

$$[\mathbf{a}_i]_n = \frac{d[\mathbf{v}_0]_n}{dt} + [C_{0/n}^T][^\times\boldsymbol{\omega}_0][^\times\boldsymbol{\omega}_0]\begin{bmatrix} r_{a1x} \\ r_{a1y} \\ 0 \end{bmatrix} + [C_{0/n}^T][^\times\dot{\boldsymbol{\omega}}_0]\begin{bmatrix} r_{a1x} \\ r_{a1y} \\ 0 \end{bmatrix} + l_i\dot{\omega}_i \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \\ 0 \end{bmatrix} + l_i\omega_i^2 \begin{bmatrix} -\cos\theta_i \\ -\sin\theta_i \\ 0 \end{bmatrix} \tag{29}$$

The rotational velocity level states are straightforwardly expressed as $[\boldsymbol{\omega}_k]_n = \begin{bmatrix} 0 & 0 & \dot{\lambda}_k \end{bmatrix}^T$ for $k \in \{0, i, j\}$, and the angular momentum equations yield $\frac{d[\mathbf{h}_k]_n}{dt} = \begin{bmatrix} 0 & 0 & I_k\ddot{\lambda}_k \end{bmatrix}^T$.

It is assumed that the quadrotor forces and moments may be modelled in the body-fixed frames as:

$$[\mathbf{f}_k^q]_k = f_k\hat{\mathbf{k}}_1 - m_k g\hat{\mathbf{n}}_1 \tag{30}$$

$$[\mathbf{l}_k^q]_k = l_k\hat{\mathbf{k}}_3 \tag{31}$$

As presented here, neither a LaGrangian or Kanian approach to obtaining the governing equations of motion offers a simple expression for the development of the translational variables.

3

II.   Cooperative localization of moving target

In this problem, three agents move independently of each other; two cooperating agents and a third, "target" agent. The goal of the cooperating agents is for a preselected agent of the pair to reach a specified radius of the target agent. (Aside: this represents nearly identically one of the low-level problems associated with Dr. Rogers's and Clark's coalition research.)

The dynamics for any individual agent are essentially trivial, and depend only on the choice of model for each. For now, it is assumed that all agents are quadrotors and may move in three dimensional space, but one of the cooperating agents or the target may be restricted to planar motion. The assumed "goals" of each agent are:

- "Lead" cooperator: reach a specified radius of the target

- "Helper" cooperator: position itself to provide "good" or "optimal" estimates to the lead

- "Target": avoid the lead vehicle for a pre-specified (but possibly unknown) time

It is assumed that, although the cooperating vehicles share some or all measurements, a full centralized filter cannot practically be implemented, and each agent must run its own estimator. The key states for estimation for any agent are assumed to be the relative position and velocity vectors of the other two agents. (Depending on the types of vehicles, possible nonholonomic motion constraints may dictate that relative attitude is also highly relevant.)

At maximum, each vehicle may need to estimate its own position, velocity, and attitude, as well as that of each other vehicle, an 18 DOF problem. The "optimal" parameterization will no doubt depend on the sensor measurements available to each vehicle; i.e. a depth sensor such as a laser scanner or RGBD camera provides different information about the target than an RGB camera, and it may be useful in some cases to couple the estimates of other states with the states of the estimating vehicle.

The "lead" vehicle may reasonably be expected to experience the greatest difficulty localizing the target if range-to-target data are not available (e.g. monoSLAM-like scenario), where the helper vehicle can place itself to increase its parallax to the target and/or the lead agent.

II.A.   2D subproblem

A restricted subproblem is considered in which all agents move in a plane. This problem is considered for the sake of setting up a representative estimator and governing equations.

Each vehicle is assumed to be constrained such that its velocity vector is aligned with the vehicle 1 axis at all times. Each agent estimates is own position and heading relative to an arbitrary inertial frame. (Agents need not use the same inertial frame.) Each agent is assumed to have velocity-level controls for speed and heading angle, allowing the velocity-level states to be treated as controls. To remain as general as practical, standard Cartesian coordinates are used for parameterization. Each agent estimates the position vectors of the other two agents, also parameterized in the inertial frame. For the other cooperating agent, it is assumed that odometry data are shared, enabling filter propagation of the position for that agent. For the target agent, the only available measurements can be relative range and/or bearing. Each agent estimates eight states: position and heading for each cooperating agent, and position of the target agent.

Each agent measures bearing to every other agent, and takes odometry measurements. Odometry, w_k, is taken as the distance travelled along the position history of the agent between sample times, and is assumed to be measured by vehicle encoders. It is approximated in simulation by summing the norm of the change in position for all numerical integration steps between sample times. It is assumed that the cooperating agents share both their bearing measurements and odometry, but not control inputs. The bearing measurement of

agent j by agent i is denoted θ_{ji}. We can write the measurement model for agent i as:

$$w_k = \sum_{j=0}^{M} \|\mathbf{r}_i(t_{j+1}) - \mathbf{r}_i(t_j)\| + v_k^w \; \forall \; \{j : t_{k-1} \le t_j \le t_k\}, v_k^w \sim N(0, \sigma_w^2) \tag{32}$$

$$\theta_{ji} = \arctan \frac{r_{jy} - r_{iy}}{r_{jx} - r_{ix}} - \psi_i \tag{33}$$

$$\theta_{ti} = \arctan \frac{r_{ty} - r_{iy}}{r_{tx} - r_{ix}} - \psi_i \tag{34}$$

The shared measurements from agent j can be incorporated by switching the indices in the previous expression. The bearing measurements are directly incorporated into the EKF, giving each agent a total of four measurements per time step. Odometry measurements are treated as inputs to the propagation equations, which allows for reasonable accuracy because of the motion constraint between velocity and heading. The complete propagation equations for agent i are written as:

$$\mathbf{r}_{k+1}^i = \mathbf{r}_k^i + \tilde{w}_k \begin{bmatrix} \cos \psi_k^i \\ \sin \psi_k^i \end{bmatrix} - v_k^w \begin{bmatrix} \cos \psi_k^i \\ \sin \psi_k^i \end{bmatrix} \tag{35}$$

$$\psi_{k+1}^i = \psi_k^i + \omega_i T_s \tag{36}$$

$$\mathbf{r}_{k+1}^j = \mathbf{r}_k^j + \tilde{w}_k^j \begin{bmatrix} \cos \psi_k^j \\ \sin \psi_k^j \end{bmatrix} - v_k^{wj} \begin{bmatrix} \cos \psi_k^j \\ \sin \psi_k^j \end{bmatrix} \tag{37}$$

$$\psi_{k+1}^j = \psi_k^j + v_k^\psi, v_k^\psi \sim N(0, T_s^2) \tag{38}$$

$$\mathbf{r}_{k+1}^t = \mathbf{r}_k^t + v_k^r, v_k^r \sim N(0, (10T_s)^2) \tag{39}$$

Note that the heading control inputs to agent j and the velocity-level inputs to the target t are assumed unknown; it is assumed that ω_j is on the order of 1, and the velocity of t is on the order of 10, leading to the variances specified in the preceding equations.

II.B. Control strategy

Controls are implemented open-loop with respect to agent estimation, using the true state values. Simple heuristic control strategies are implemented to produce reasonable but suboptimal agent motion. This motion is relevant in preliminary simulation because the estimation accuracy is highly dependent on the relative agent position and pose. Localizing the target is most difficult when flying directly towards it, which is a fair approximation of the pursuit agent's overall goal. Agent 1 denotes the agent tasked with reaching the target. Agent 2 denotes the helper agent. Agent 1's strategy is to move at its maximum speed v_{max}^1 towards the closest intercept point with the target, assuming the target moves at a known, constant velocity. At each time step, agent 1 solves for the roots of the following equation with respect to the intercept time T_{int} and the reference heading ψ_{ref}^1:

$$0 = \mathbf{r}_k^t - \mathbf{r}_k^1 + T_{int}(v_{max}^t \begin{bmatrix} v_x^t \\ v_y^t \end{bmatrix} - v_{max}^1 \begin{bmatrix} \cos \psi_{ref}^1 \\ \sin \psi_{ref}^1 \end{bmatrix}) \tag{40}$$

Depending on the current heading ψ_k^1, agent 1 will then turn at the maximum turn rate along the direction that minimizes the heading error. This control strategy is implemented in lieu of solving a free-final time optimal control problem.

Agent 2's objective is to fly relative to agent 1 and the target in a fashion that provides good parallax between both agent 1 observing 2 and agent 2 observing the target. Sequential feedback controllers for agent 2's rate of change of speed $\dot{v}_2$ and turn rate ω_2 are designed to track a reference inertial position. For preliminary investigation, the reference position is taken to be 10 metres from the target along the target's

$\pm\hat{\mathbf{b}}_2$ body-frame axis. The sign of the reference is chosen to place agent 2 on the opposite side of the target from agent 1. Reference heading and velocities are obtained by solving the following equation for v_{ref} and ψ^2_{ref}:

$$v_{ref}\begin{bmatrix} \cos \psi^2_{ref} \\ \sin \psi^2_{ref} \end{bmatrix} = -[K](\mathbf{r}^2_k - \mathbf{r}^2_{ref}) \tag{41}$$

The reference time rates of speed and heading, which are treated as controls in this scenario, are given from:

$$\begin{bmatrix} \dot{\psi}^2_{ref} \\ \dot{v}_{ref} \end{bmatrix} = -[K_v]\left(\begin{bmatrix} \psi^2_k - \psi^2_{ref} \\ v^2_k - v_{ref} \end{bmatrix} \right) \tag{42}$$

The control strategy for the target is to drive at its maximum speed along a fixed path. Greater realism could be easily added in this assumption, but the problem as outlined provides sufficient complexity to examine estimation accuracy.
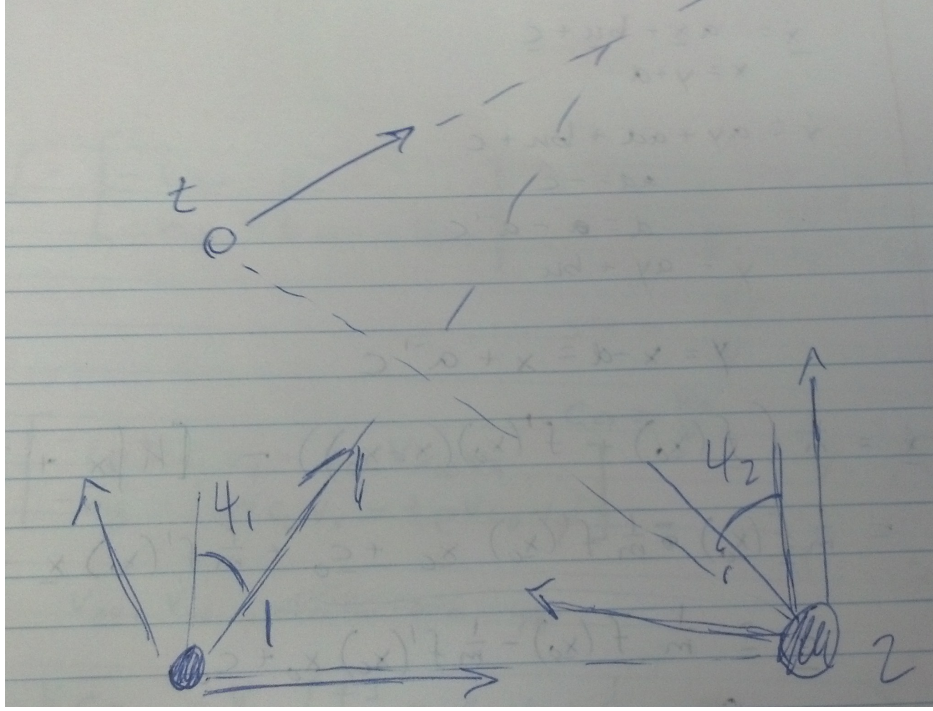

Figure 1. Problem geometry showing target t, pursuit agent 1, and assisting agent 2. $\psi_2 < 0$ as shown.

II.C. Simulation

Simulation is conducted with a 10 Hz sample rate and the following sensor variances:

$$\sigma^2_\theta = 3.385 \times 10^{-5} \tag{43}$$
$$\sigma^2_w = .00111 \tag{44}$$
$$\sigma^2_{r_t} = 1 \tag{45}$$
$$\sigma_{\dot{\psi}} = .01 \tag{46}$$

Sensor readings are unbiased. Each agent is initialized in an inertial space where Agent 1 is initially at the origin with zero heading. Agent 2 is initialized at $(5, -5)$ with heading 0.1 rad. The target is initialized at $(10, 0)$ with heading $\pi/4$ rad. The maximum speed of the target is 0.9 m/s. The maximum speed and turn rate of agent 1 is 1 m/s and 1 rad/s. Agent 2's rates are left unconstrained, to ensure it can reach its reference fast enough to provide good measurements. The maximum rates experienced by agent 2 are about 4.5 m/s and 1.3 rad/s. The trajectories are constant for all simulations. Monte Carlo analysis of estimator performance is conducted with new sensor noise and new initial conditions every simulation. The initial estimator states for each vehicle are set between 50% and 150% of the true initial states, using a uniformly distributed multiplicative error with respect to the true states.

1000 simulations are conducted both using shared measurements and not using shared measurements. Simulation time is 30 seconds each. We concern ourselves mostly with the final range-to-target error for agent 1; the bearing error is relatively low, so range errors will be the most significant source of uncertainty in trajectory generation for agent 1. In 33 simulations, the individual estimator for agent 1 has a final value of greater than 10 m, which is assumed to indicate a failed estimator. Table 1 summarizes errors for the remaining individual cases and for all cooperative cases. Cooperative estimation reduces the standard error by approximately a factor of three. The error bias seen in both cases is not surprising, as the two agents approache the target along the same trajectories in every simulation.

In examining these results, it should be kept in mind that the agent trajectories are in no way optimized, either for estimation or target tracking purposes. It is also true that a SLAM algorithm might be employed to improve localization of the agents and might reduce drift in estimated states. Further, a different choice of filter coordinates might produce better results. For instance, a parameterization of the unknown moving target inspired by the inverse-depth parameterization popular in monocular SLAM might be possible. (Aside: I have tried such a parameterization but achieved worse accuracy. I do not know if my implementation is correct.)

On the other hand, the trajectories are not closed-loop with respect to the sensing, no effort has been made to address sensor bias, and the agents vary their heading very little for most of the simulation. This exercise is intended to demonstrate the problem and fundamental approach.
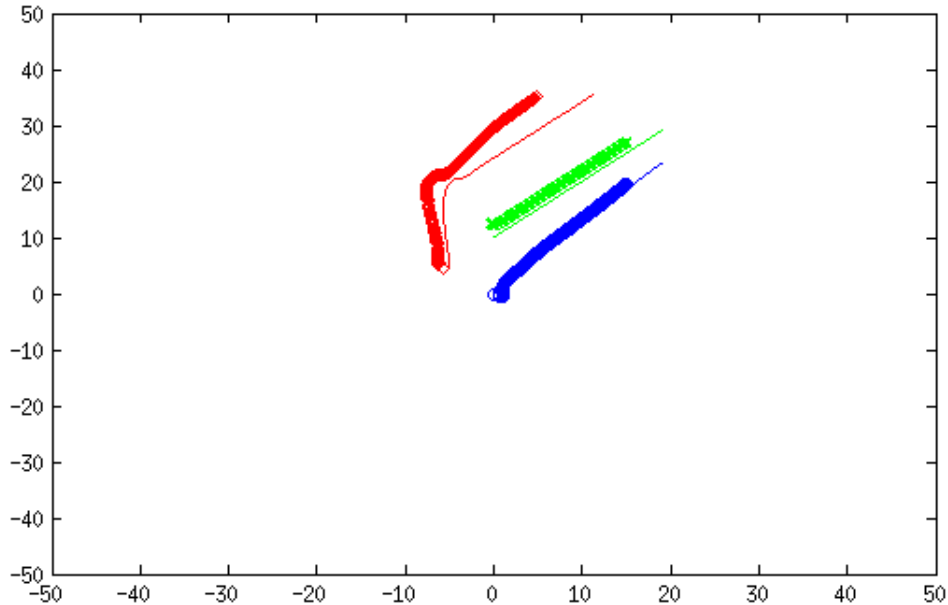

Figure 2. Sample true trajectories (lines) and estimated by agent 2 in one simulation. Target is green, agent 1 is blue, and agent 2 is red.

7

Case	$\bar{\epsilon}_{\rho_{t1}(t_f)}$ (m)	$S(\epsilon_{\rho_{t1}(t_f)})$ (m)
Cooperative	0.686	1.13
Individual	0.783	3.69

Table 1. Mean and standard error associated with agent 1 range-to-target errors at the final simulation time in the cooperative and noncooperative estimation problems.

III. Flexible body cooperative carry

In this problem, we consider two (or more) agents tasked with jointly transporting a deformable body such as a net or tarp. For simplicity, it is assumed that the deformable body's mass is negligible. The two agents can move freely relative to one another with no new constraint forces relative to normal single-agent motion. An effective kinematic constraint limiting the maximum interagent distance must be considered to avoid an undesireable failure mode. For the sake of safety, a minimum interagent distance should also be introduced to avoid collisions. The problem as posed bears resemblance to the problems of formation flying and collision avoidance in UAVs or satellites.

This problem admits several research challenges, including cooperative navigation and control in the presence of sensor noise and trajectory optimization to improve estimation accuracy, subject to state constraints. When combined with the cooperative estimation framework, a hardware implementation may be envisioned with varying levels of complexity.

For now, a planar implementation of this problem using the established cooperative estimation framework is considered. This implementation highlights some of the navigation and control challenges that must be addressed.

III.A. Agent estimator summary

The agent dynamics and estimator equations have been covered previously. For the sake of clarity, the broad structure of the estimation problem is summarized:

- Each agent possesses an IMU, used to propagate the velocity and heading states in the Kalman update.

- Each agent measures bearing to known features in the workspace, with a limited field of view and minimum and maximum range sensitivity.

- Each agent measures relative range and bearing to the other agent, irrespective of range or relative orientation.

- Control inputs that influence the velocity history are not considered explictly in the estimation. Velocity states are propagated using only accelerometer measurements.

- Each agent estimates its own state, consisting of five values: two position and velocity components, and heading. Position and velocity are coordinatized in a body-fixed reference frame.

- State histories for the agents are generated in simulation. The simulation is open-loop with respect to the estimation, using the true states for control when needed. Estimation is evaluated on the resulting state and measurement histories.

III.B. 2D subproblem

Two agents are constrained to move in a plane normal to gravity. These agents may be ground robots or UAVs. Each agent has a limited forward field of view (FOV) sensor that is used to detect features in the environment and perform SLAM or localization in the presence of known features. The agents also measure range and bearing to one another. Agents are tethered by a flexible rope of length L and have a minimum safe

distance D. Aside from the state constraints, the problem is the same as the generic cooperative localization problem previously considered.

A naïve solution is to have one agent move freely, while the second agent obeys a control law that regulates the interagent distance only. This is one of the simplest conceivable approaches; it ignores any relative heading constraints, and it does not allow admit commands that place the rope in a given orientation in inertial space. Both of these possibilities should be considered in a more realistic implementation. The naïve controller is used to establish a baseline performance, as well as to explore the research challenges associated with the larger problem.

A Lyanpunov-based nonlinear feedback controller is developed for the follower agent. Let the agents be i and j, each with a position vector parameterized in the inertial frame:

$$[\mathbf{r}_i]_n = \begin{bmatrix} r_{ix} \\ r_{iy} \end{bmatrix} \tag{47}$$

$$[\mathbf{r}_j]_n = \begin{bmatrix} r_{jx} \\ r_{jy} \end{bmatrix} \tag{48}$$

We consider as the controlled variable the interagent distance r only. Letting the follower agent be designated i and the lead agent be j, then we can define r and the radial tracking error e in terms of a reference value r_{ref}.

$$r = \|\mathbf{r}_j - \mathbf{r}_i\| = \sqrt{(r_{jx} - r_{ix})^2 + (r_{jy} - r_{iy})^2} \tag{49}$$

$$e = r - r_{ref} \tag{50}$$

Taking the Lyapunov function to be $V = \frac{1}{2}e^2$, we obtain the condition $\dot{e} = -Ke$ as a sufficient condition for stability, $K > 0$. A constant reference $r_{ref} = \frac{L+D}{2}$ is assumed, such that the requirement becomes

$$\dot{r} = -Ke \tag{51}$$

Subsituting Eq. 49 and isolating for the controlled variables r_{ix} and r_{iy}, we obtain the expression:

$$-rKe - ((r_{jx} - r_{ix})\dot{r}_{jx} + (r_{jy} - r_{iy})\dot{r}_{jy}) = \begin{bmatrix} (r_{ix} - r_{jx}) & (r_{iy} - r_{jy}) \end{bmatrix} \begin{bmatrix} \dot{r}_{ix} \\ \dot{r}_{iy} \end{bmatrix} \tag{52}$$

Recognizing the system is underconstrained, an unweighted minimum-norm solution for $\begin{bmatrix} \dot{r}_{ix} & \dot{r}_{iy} \end{bmatrix}^T$ can be obtained. Compactly representing the system as $A\mathbf{x} = \mathbf{b}$, the minimum-norm solution for $\mathbf{x}$ is $\mathbf{x} = A^T(AA^T)^{-1}\mathbf{b}$. With this minimum-norm solution, Eq. 52 provides reference inertial-frame velocity-level commands. Note that the controller is coupled with the inertial velocity of agent j.

Similarly, we can define a velocity-level tracking error $\mathbf{e}_v = \mathbf{v} - \mathbf{v}_{ref}$. Using a similar Lyapunov function $V = \frac{1}{2}\mathbf{e}_v^T\mathbf{e}_v$, $\dot{\mathbf{e}}_v = -K_v\mathbf{e}_v$ becomes the condition for stability. Making the assumption that the reference velocity time rates obtained from Eq. 52 are much slower than the dynamics of the velocity states themselves, an acceleration-level control can be obtained in the following form:

$$\begin{bmatrix} \ddot{r}_{ix} \\ \ddot{r}_{iy} \end{bmatrix} = -K_v\mathbf{e}_v \tag{53}$$

To make the problem interesting, the following agent is modelled as having a thrust-like force in the plane of motion in the direction of its heading. The thrust magnitude and the heading time rate of change are taken as controls. Recalling that the position vectors are componetized in the inertial frame, then a reference heading ψ and thrust command f_i can be obtained:
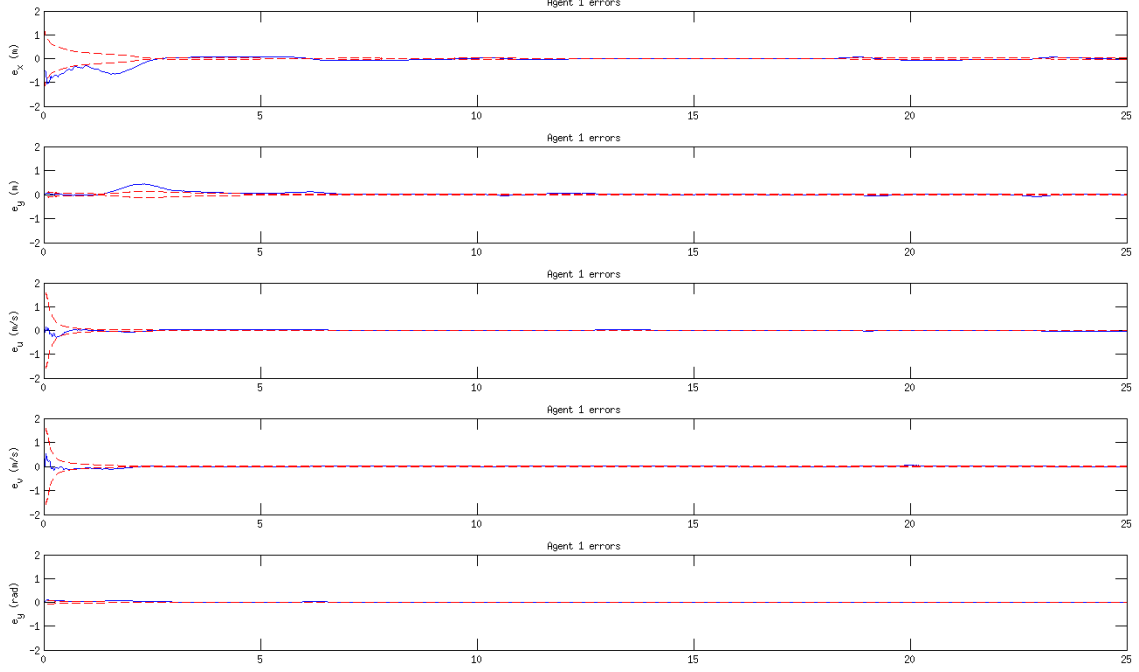
9

Figure 3. Agent 1 estimate time histories for one simulation.

$$f_i \begin{bmatrix} c\psi \\ s\psi \end{bmatrix} = -K_v \mathbf{e}_v \tag{54}$$

$$f_i = \| -K_v \mathbf{e}_v \| \tag{55}$$

$$\psi = \arctan \frac{- \begin{bmatrix} 0 & 1 \end{bmatrix} K_v \mathbf{e}_v}{- \begin{bmatrix} 1 & 0 \end{bmatrix} K_v \mathbf{e}_v} \tag{56}$$

The heading rate command is simply taken as the heading error times a high fixed gain (10). Using the reference thrust and heading, agent 2 is simulated in open-loop using the true state values of agents 1 and 2.

III.C.   Simulation

The problem is simulated using two agents with bearing-only feature measurements and range and bearing measurements of one another. The feature detection field of view is taken as 130°with sensitivity between 1 and 10 m. Agent 1 follows a sinusoidal reference trajectory while Agent 2 uses the control law developed in the previous section with $K = 3.16$ and $K_v = 2.5$. In general, the estimates by Agent 1 behave as expected and are well-bounded by the 3σ covariance. For agent 2, the estimates are bounded, but generally exceed the 3σ covariance estimate.

Results for one simulation are shown in Figs. 3-4. These are typical of results seen in other simulations. No rigorous Monte Carlo analysis has been performed. The poor estimate accuracy for agent 2 is thought to be caused partly due to relatively high-frequency, large control excitations that result from the control law developed. It should be recalled that the control inputs are not explicitly accounted for in the estimation process, and might be used in a more realistic scenario to reduce some of the uncertainty associated with the vehicle acceleration.
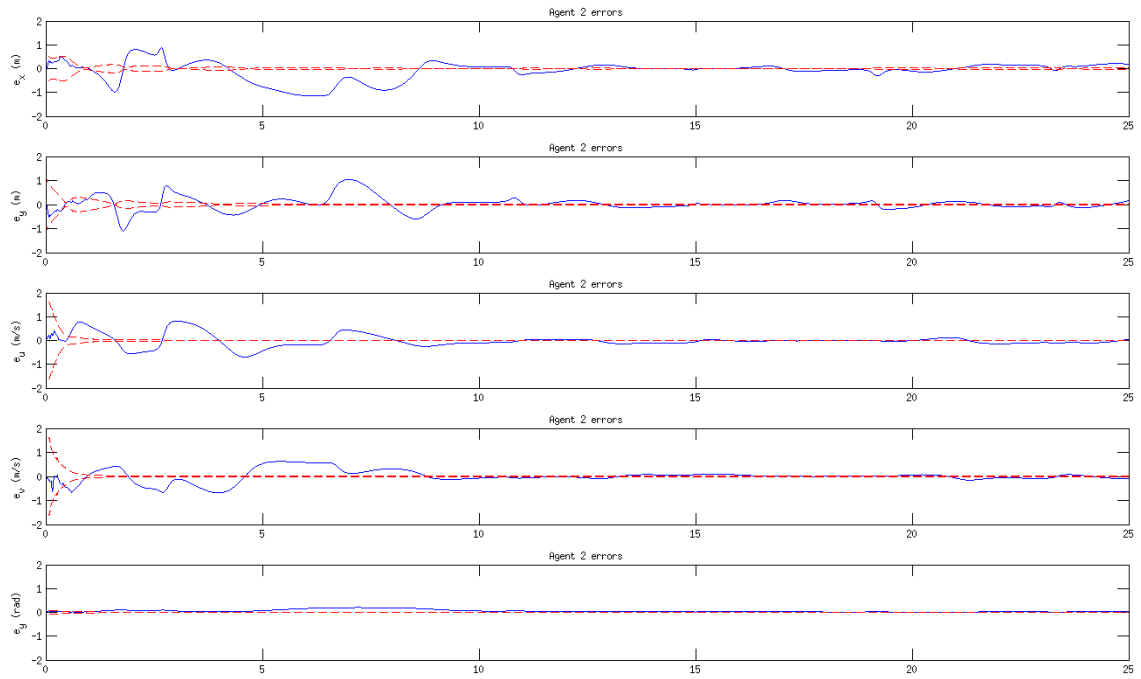
Figure 4. Agent 2 estimate time histories for one simulation.

11