

Notes on decentralized coordination of multi-robot networks

I. PROBLEM FORMULATION

In the model there is a set of N robots \mathcal{V} , whose state vector is $\mathbf{s}_i = [\mathbf{q}_i, \theta_i]$, where $\mathbf{q}_i = [x_i, y_i]$ is the position of the robot on a plane and θ_i the orientation, and a set of $M \geq N$ targets \mathcal{M} with positions \mathbf{x}_j . The state depends on the control signal that consists of the velocity and orientation of the robot $\mathbf{u}_i = [v_i, \theta_i]$, both bounded, and can be represented in discrete time as follows:

$$\mathbf{s}_i(k+1) = \mathbf{f}(\mathbf{s}_i(k), \mathbf{u}_i(k)), \quad |\mathbf{u}_i(k)| \leq \mathbf{b}, \mathbf{a}_1 \leq \mathbf{s}_i(k+1) \leq \mathbf{a}_2 \quad (1)$$

which represent the individual robot's constraints, $\chi_i(k) = [\mathbf{s}_i(k), \mathbf{u}_i(k)] \in \mathcal{X}_i$.

The mobile robots know their position and through their cameras, for those that fall in their field $\mathcal{S}_i(k)$ of view, they can compute an estimate $\hat{\mathbf{x}}_{ij}(k)$ of the position and velocity of the targets $\mathbf{x}_j(k) = [x_j, y_j, v_{xj}, v_{yj}]$, relative to a reference system centered at the robot i such that $\mathbf{x}_j(k) \in \mathcal{S}_i(k)$. The targets are assumed to move in the reference system centered at the robot according to linear Gaussian dynamics and through sensors the robots' observation is also linear and corrupted by Gaussian noise:

$$\mathbf{x}_j(k+1) = \mathbf{F}\mathbf{x}_j(k) + \mathbf{w}(k), \quad \mathbf{w}(k) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}), \quad (2)$$

$$\mathbf{z}_{i,j}(k) = \begin{cases} \mathbf{H}\mathbf{x}_j(k) + \mathbf{v} & \mathbf{x}_j(k) \in \mathcal{S}_i(k) \\ \mathbf{0} & \text{else} \end{cases} \quad (3)$$

where $\mathbf{H} = [\mathbf{I}_2, \mathbf{0}_2]$ means that the sensors only observe a noisy position but not the velocity. Given the model, the Kalman filter provides the optimum estimate of the future state of the robots $\hat{\mathbf{x}}_{ij}(k+1)$. The robots can communicate through a meshed network represented by the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that forms a strongly connected graph and that can help them coordinate their trajectories in pursuit of the task of monitoring the targets. The goal is to assign to each robot a distinct set of targets and achieve a consensus on the division of tasks. This is an integer problem, and it is not a matching problem because a robot will have to pursue multiple targets, since the targets outnumber the robots. However, assuming that each robot can at most pursue P targets, we can build a model in which individual robots are operating as equivalently P of them, so that the total $PN \geq M$ in general, transforming the problem into a matching problem.

A. Robot assignment through maximum utility maximization

Suppose one has a worker/job assignment problem, with edge weight w_{nj} is the *competence* of the worker n to perform the job j . The problem of finding a matching that minimizes the cost or maximizes the average competency is the so-called *linear sum assignment problem*; its integer formulation is as follows:

$$\max_{\mathbf{c}} \mathbf{w}^\top \mathbf{c} \quad (4)$$

$$\text{subject to } \mathbf{c} \leq \mathbf{1}, \quad (5)$$

$$\mathbf{B}^u \mathbf{c} \leq \mathbf{1}, \quad (6)$$

$$c_i \in \{0, 1\} \quad (7)$$

where $c_i \in \{0, 1\}$ is the integrality constraint, \mathbf{B}^u is the unoriented incidence matrix of the bipartite graph and the constraint $\mathbf{B}^u \mathbf{c} \leq \mathbf{1}$ ensures that at most one edge is picked for each node. It is well known that the trick to remove the integrality constraints is to introduce two fictitious nodes, a source a connected to the workers and a destination b connected to the jobs and solve the following problem:

$$\max_{\mathbf{c}} \mathbf{w}^\top \mathbf{c} \quad (8)$$

$$\text{subject to } \mathbf{0} \leq \mathbf{c} \leq \mathbf{1}, \quad (9)$$

$$\mathbf{B}^u \mathbf{c} \leq \mathbf{1}, \quad (10)$$

$$[\mathbf{B}\mathbf{c}]_a = -[\mathbf{B}\mathbf{c}]_b = \min(N, M), \quad (11)$$

$$[\mathbf{B}\mathbf{c}]_u = 0, \quad u \neq \{a, b\} \quad (12)$$

which, if the weights are integer numbers, has a solution that is such that $c_i \in \{0, 1\}$, even though the constraint has not been explicitly imposed.

An estimate of the ease for the robot i to pursue the target j , which we call utility, can be based on estimate of the target minimum future relative position, for instance:

$$U_{ij}(k) = \begin{cases} \frac{1}{\min_{\mathbf{u}_i(k)} \|\hat{\mathbf{x}}_{ij}(k+1)\|}, & \hat{\mathbf{x}}_{ij}(k) \in \mathcal{S}_i(k) \\ 0 & \text{else} \end{cases}. \quad (13)$$

In order to solve the matching problem the weights have to be integer values we can define the weight:

$$w_{ij}(k) = \lceil U_{ij}(k) \kappa \rceil \quad (14)$$

where κ is a large enough constant. Here we do not want each robot to only have one target to follow, so the idea is to create for each robot an allocation vector $\mathbf{c}_i, i \in \mathcal{V}$ of size P chosen so that the expected $M \leq NP$. Let $\mathbf{c} = [\mathbf{c}_1^\top, \dots, \mathbf{c}_N^\top]^\top$, the problem formulation is:

$$\max_{\mathbf{c}_i} \sum_{i \in \mathcal{V}} \mathbf{w}_i^\top \mathbf{c}_i \quad (15)$$

$$\text{subject to } \mathbf{0} \leq \mathbf{c}_i \leq \mathbf{1}, \quad i \in \mathcal{V} \quad (16)$$

$$\mathbf{B}^u \mathbf{c} \leq \mathbf{1}, \quad (17)$$

$$[\mathbf{B}\mathbf{c}]_a = -[\mathbf{B}\mathbf{c}]_b = M, \quad (18)$$

$$[\mathbf{B}\mathbf{c}]_u = 0, \quad u \neq \{a, b\} \quad (19)$$