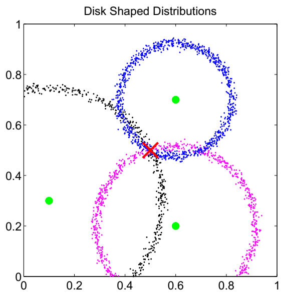

Fig. 2. Samples from the distributions of each of three beacons estimations of a robot's location, resulting in three ring-shaped distributions

robots which receive distance readings from robot s at time τ as the neighbors of s, and denote them by $\Gamma(s,\tau)$.

Our objective is to determine position estimates $\hat{x}_{s,\tau}$ which minimize the mean squared error of each robot's location estimates, averaged over all time steps:

$$L(x,\hat{x}) = \frac{1}{nT}\sum_{\tau=1}^{T}\sum_{s=1}^{n}||\hat{x}_{s,\tau} - x_{s,\tau}||^2 \quad (1)$$

The NBP algorithm provides non-parametric distributions at each time step that can be used to quantify confidence based on variance. A multi-modal distribution can be resolved in a variety of ways, as discussed in the Future Work section.

B. Modeling System Properties

The state of each robot s at time τ is represented by its position and velocity $\chi_{s,\tau} = (x_{s,\tau}, \dot{x}_{s,\tau})$. We model errors in the estimated distance between robots s and t as follows:

$$d_{st,\tau} = ||x_{s,\tau} - x_{t,\tau}|| + \nu_{st,\tau} \qquad \nu_{st,\tau} \sim \mathbb{N}(0,\sigma_\nu^2) \quad (2)$$

To unambiguously localize a robot, inter-distance readings from at least three other localized robots or beacons are required. Therefore, for localization at the first timestep, the network must contain at least three beacons and all robots must have at least three inter-distance readings. We demonstrate three ring-shaped distributions from exact beacons, used to localize a single robot in Fig. 2.

Following [20], we model the decay in a robot's probability P_o of measuring the distance to another robot as follows:

$$P_o(x_{s,\tau}, x_{t,\tau}) = \exp\left\{-\frac{1}{2}||x_{s,\tau} - x_{t,\tau}||^2/R^2\right\} \quad (3)$$

Here, R is a parameter specifying the range of the transmitter used for distance estimation. More elaborate models could be used to capture environmental factors or multi-path effects.

Each robot moves over time according to a dynamics model $P(\chi_{s,\tau+1} \mid \chi_{s,\tau})$ defined as follows:

$$x_{s,\tau+1} = x_{s,\tau} + \dot{x}_{s,\tau+1} \quad (4)$$

$$\dot{x}_{s,\tau+1} = \dot{x}_{s,\tau} + \omega_{s,\tau} \qquad \omega_{s,\tau} \sim \mathbb{N}(0,\sigma_\omega^2) \quad (5)$$

In practice, these velocities are often unobserved variables used to explain the typically smooth character of real robotic motion. Alternatively, sensors such as accelerometers could provide more precise dynamics information.

IV. Background

A. Undirected Graphical Models

We model the relationships among the random variables in our tracking problem using an undirected graphical model, or pairwise Markov random field (MRF) [28]. This model is specified via an undirected graph G, with vertex set $V = \{1,\ldots,n\}$ and edge set E. Each node $s \in V$ is associated with a random variable x_s. For notational simplicity, we focus here and in Sec. V on static localization, so that node variables are robot positions x_s. For the tracking problems considered in Sec. VI, they become temporal states $\chi_{s,\tau}$.

The graph G specifies a factorization of the joint distribution of these random variables into a product of local, non-negative compatibility functions:

$$p(x) = \frac{1}{Z}\prod_{s\in V}\psi_s(x_s)\prod_{(s,t)\in E}\psi_{st}(x_s,x_t) \quad (6)$$

Here, Z is a normalization constant, and the compatibility functions encode the information provided by dynamics models and observed inter-distances.

B. Belief Propagation

Belief propagation (BP) [3] is a general inference algorithm for graphical models. Let $\Gamma(s) = \{t \in V \mid (s,t) \in E\}$ denote the set of neighbors of node s. In BP, each node s iteratively solve the global inference problem by integrating information via local computation, and then transmitting a summary message to each neighbor $t \in \Gamma(s)$. In general, this message $m_{st}(x_t)$ is a function encoding sufficient statistics needed for node t to perform the next round of computation. For tree-structured graphs, BP is a dynamic programming algorithm which exactly computes posterior marginal distributions $p_s(x_s)$ for all nodes $s \in V$. However, the same update equations are widely applied to graphs with cycles, in which case they provide (often good) approximate marginal estimates $\hat{p}_s(x_s)$ [3].

The BP algorithm begins by initializing all messages $m_{st}(x_t)$ to constant vectors, and then updates the messages along each edge according to the following recursion:

$$m_{st}(x_t) \leftarrow \int_{x_s}\psi_s(x_s)\psi_{st}(x_s,x_t)\prod_{u\in\Gamma(s)\setminus t}m_{us}(x_s)\,dx_s \quad (7)$$

This sum-product algorithm is then iterated until the set of messages converges to some fixed point. The order in which the messages are updated is a design parameter, and various schedules (parallel, sequential, etc.) exist. Upon convergence, the messages can be used to compute approximations to the marginal distributions at each node:

$$\hat{p}_s(x_s) \propto \psi_s(x_s)\prod_{t\in\Gamma(s)}m_{ts}(x_s) \quad (8)$$

C. Nonparametric BP: Distributions as Samples

For inter-distance tracking and localization problems, it is computationally infeasible to discretize the state space as required by the standard sum-product algorithm. Thus, we employ nonparametric belief propagation (NBP) [4],