Communication-Aware Navigation Functions for Cooperative Target Tracking

Alireza Ghaffarkhah and Yasamin Mostofi

Abstract— **In this paper we consider a team of mobile nodes that are in charge of cooperative target tracking. We propose communication-aware navigation functions that allow the nodes to perform their task while maintaining their connectivity to a fixed base station and avoiding obstacles. More specifically, we show how to incorporate measures of link qualities in the navigation functions. We consider both centralized and decentralized scenarios. We furthermore explore the impact of stochastic channels and channel estimation error on the overall performance.**

I. INTRODUCTION

Recently, there has been considerable interest in cooperative mobile sensor networks. Such networks have a variety of applications from environmental monitoring, surveillance and security to target tracking and military systems. Communication plays a key role in the overall performance of mobile networks as each sensor relies on improving its status by processing the information received from others. Most of the current research in this area, however, assumes ideal or over-simplified communication links. For instance, it is common to assume links that are perfect within a certain radius of a node. Such simple models, however, will not work well for realistic communication scenarios as there are many factors that can degrade the performance of wireless communication in an indoor or outdoor environment. Studying the impact of realistic communication channels on estimation/control performance of wireless sensor networks is a new and emerging area of research. Authors in [1]-[7] have looked at the impact of some aspects of a communication link, like noise, quantization, fading, medium access and packet loss, on wireless control of a mobile sensor network.

In [4]-[6], Mostofi et al. introduced communication-aware motion planning strategies, using an information-fusion approach, and considered the impact of distance-dependent path loss and fading on decentralized motion-planning and data fusion in mobile networks. An extension of [6], with a modification of the cost function, appeared in [7]. In [8], we proposed building communication-aware navigation functions for centralized cooperative target tracking scenarios, where we proved convergence to the optimum configuration. Navigation functions [9] are special types of artificial potential fields and have been extensively used in motion planning

literature [10]. In [8], we modified the definition of the classical navigation functions to include realistic communication and sensing objectives. The resulting navigation functions did not have all the properties of the classical navigation functions introduced by Rimon and Koditschek [9]. For example, it had multiple time-varying minima and a time-varying structure. In this paper, we extend our previous work to both centralized and decentralized target tracking cases. We show how to build novel communication-aware navigation functions that allow the vehicles to perform their task efficiently while staying connected and avoiding obstacles. We consider realistic communication environments that can experience path loss, shadowing and fading. We furthermore explore the impact of the uncertainty in the estimation of the SNR map on the performance of the proposed framework.

The rest of the paper is organized as follows. In Section II, we formulate the cooperative target tracking problem and show how to model realistic stochastic communication links. In Section III, we show how to build both centralized and decentralized communication-aware navigation functions by incorporating link quality information in the classical navigation function framework. We furthermore explore the impact of the uncertainty in the estimation of the SNR map on the cooperative operation. We conclude in Section IV.

II. PROBLEM FORMULATION

Consider a team of mobile robots that are cooperating to track a moving target jointly. The robots are equipped with sensing devices to measure their own positions as well as the position of the target. Then they communicate their acquired information to a fixed base station. The overall goal is for the base station to have the best estimate of the target position. Fig. 1 shows a schematic of the problem considered in this paper.

We consider a spherical workspace $\mathcal{W} = \{q \in \mathbb{R}^2 \mid \|q\| \leq R\} \subset \mathbb{R}^2$ punctured by M disjoint disc-shaped obstacles and N disc-shaped robots. The assumption of spherical workspace, which is common in navigation function framework, is not limiting as long as we can find a diffeomorphism that properly translates the given workspace into a spherical one. But even if such diffeomorphism cannot be found, any non-spherical workspace can be conservatively approximated with a spherical one. Therefore, in this paper we present our results for a spherical workspace.

The robots and obstacles are specified by the following sets:

$$\mathcal{R}_j = \{q \in \mathbb{R}^2 \mid \|q - q_j\| \leq r_j\}, \ 1 \leq j \leq M + N, \quad (1)$$
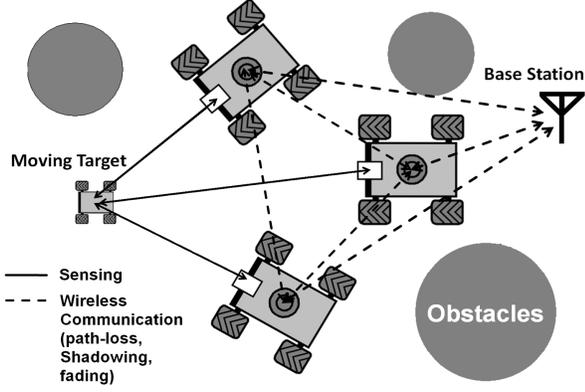
Fig. 1. Cooperative target tracking by a group of mobile robots.

where $q_j \in \mathbb{R}^2$ is the position of the jth robot (or the jth obstacle) and r_j is its radius. The first N sets specify the robots and the rest specify the obstacles, which are considered stationary robots. The overall state of the system is denoted by $\mathbf{q} = [q_1^T \cdots q_N^T]^T$. The position of the base station is also denoted by $q_b \in \mathbb{R}^2$. We assume that the base station does not impose any constraints on the movement of the robots. Furthermore, we assume holonomic robots with the following dynamics:

$$q_j[k+1] = q_j[k] + u_j[k], \quad 1 \le j \le N, \qquad (2)$$

where $u_j \in \mathbb{R}^2$ is the control input to the jth robot.

Each robot measures its own position as well as the position of a moving target. Then, the objective is to navigate the whole team to positions that give the best estimate of target position at the base station.[1] Let $x \in \mathbb{R}^2$ represent the position of the target. We consider the following dynamical model for the moving target:

$$x[k+1] = Ax[k] + w[k], \qquad (3)$$

where $w \in \mathbb{R}^2$ is a zero-mean Gaussian noise with $Q[k] = \mathbb{E}\{w[k]w^T[k]\}$ representing its covariance matrix. It should be noted that the dynamics of the moving target should be slow enough in order to guarantee the stability of the whole system [8]. Let $z_j[k]$ represent the measurement of the jth robot of x at time k. We have

$$z_j[k] = x[k] + v_j[k], \quad 1 \le j \le N, \qquad (4)$$

where v_j is a zero-mean Gaussian observation noise with $R_j[k] = \mathbb{E}\{v_j[k]v_j^T[k]\}$ representing its covariance. We assume identical sensors and adopt the position-based sonar model introduced in [11] for the observation noise covariance:

$$R_j[k] = T^T\big(\phi_j[k]\big)\, \tilde{R}_j[k]\, T\big(\phi_j[k]\big) \qquad (5)$$

with

$$\tilde{R}_j[k] = \begin{bmatrix} \Xi\big(\|q_j[k] - x[k]\|\big) & 0 \\ 0 & \lambda\Xi\big(\|q_j[k] - x[k]\|\big) \end{bmatrix},$$
$$(6)$$

[1] While in this paper we focus on target tracking problems, the proposed framework is applicable to other multi-agent motion planning applications.

where $\phi_j[k]$ is the angle between the vector $q_j[k] - x[k]$ and x-axis in a global frame, $\Xi(.)$ is a real positive analytic function representing the range noise variance, $\lambda > 0$ is a scaling constant and T is the 2D rotation matrix given by

$$T(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}. \qquad (7)$$

We choose the following quadratic form for $\Xi(.)$:

$$\Xi(r) = \alpha\big(r - r_s\big)^2 + \varepsilon, \qquad (8)$$

where α and ε are positive constants and r_s is the sweet spot radius which gives the best sensing quality. As for the positions of the robots, we assume that each robot can estimate its own position perfectly. Our framework can be easily extended to consider such estimation errors.

In this paper we consider both centralized and decentralized target tracking scenarios. In the centralize case, there is no inter-agent communication and the robots send their raw measurements to the base station, which is in charge of data fusion and motion control. The base station uses an Information Filter (IF) [12] to fuse its received data and decides where each robot should go next in order to minimize its estimation error covariance (or alternatively maximize its information). It can be easily confirmed that the overall estimation error covariance at the base station is minimized when $\sum_{j=1}^{n} R_j^{-1}[k]$ is maximized. Therefore, the motion control objective at the base station is to navigate the robots to the positions which maximizes $\sum_{j=1}^{n} R_j^{-1}[k]$ while avoiding collisions and maintaining connectivity to the base station. The base station then sends the control signals to the robots. If the connection between a robot and the base station is lost, that particular robot will not contribute to the fusion process. Consequently, it will not receive a control signal from the base station. It will then search in random motions until it restores its connection.

In the decentralized case, on the other hand, each robot runs its own local information filter and sends its filtered information to the base station, which will then fuse the received information to estimate the position of the target. Similar to the centralized case, if the base station loses its connection with a number of robots, those robots cannot contribute to the fusion process. But in this case the control signals are generated at the robots instead of the base station. The inter-agent communication is crucial in this case as the robots need the positions of their neighbors for obstacle avoidance. However, the robots will not try to optimize connectivity with any other robot. They simply use the information of whoever they can talk to for obstacle avoidance.

In this paper, we explore the impact of packet drop and communication channel estimation errors on our proposed communication-aware navigation framework for both centralized and decentralized cases.

A. Communication Model

In a realistic communication setting, such as an urban area, Line-Of-Sight (LOS) communication may not be possible

due to the existence of several blocking objects that can attenuate, reflect or refract the transmitted signal. Then the communication between the agents and the base station can be degraded due to factors such as shadowing or fading [13]. While these factors can degrade the overall performance of the network considerably, multi-agent robotic and navigation literature typically consider ideal or over-simplified communication links.

A fundamental parameter that characterizes the performance of a communication channel is the received Signal to Noise Ratio (SNR). Received Signal to Noise Ratio is defined as the ratio of the received signal power divided by the receiver thermal noise power. The instantaneous received SNR determines whether the received packet will be kept and used in the fusion and motion planning process. In general, there are three time-scales associated with the spatio-temporal changes of the channel quality and therefore the received SNR as is shown in Fig. 2 [13]. The slowest dynamic, *path loss*, is associated with the signal attenuation due to the distance-dependent power fall-off. Depending on the environment, there could be a faster variation, referred to as *shadow fading* (or shadowing), which is due to the blocking objects. Finally, multiple replicas of the transmitted signal can arrive at the receiver due to the reflection from the surrounding objects, resulting in even a faster variation in the received signal power called *multi-path fading*.
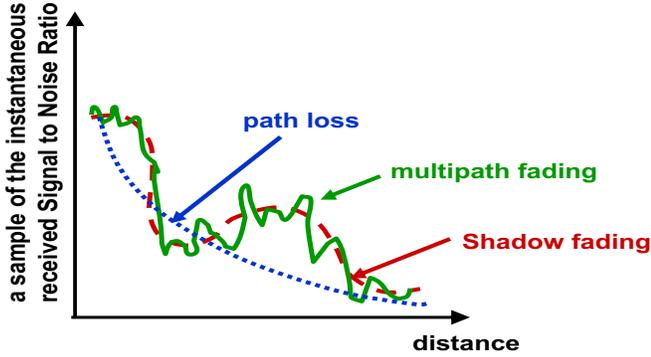

Fig. 2. A multi-scale representation of channel dynamics.

Let $\gamma_{b,j}[k]$ represent the instantaneous received SNR between the jth robot and the base station at time k.[2] We can model $\gamma_{b,j}[k]$ as a random variable whose square root is Rician distributed. Then, $\gamma_{b,j}[k]$ will have the following probability density function (pdf) [13]:

$$p_{\gamma_{b,j}[k]}(\gamma) = \frac{(1+K_b)}{\bar{\gamma}_{b,j}[k]} e^{-\frac{K_b \bar{\gamma}_{b,j}[k]+(1+K_b)\gamma}{\bar{\gamma}_{b,j}[k]}}$$
$$\times I_0\left(2\sqrt{\frac{\gamma K_b(K_b+1)}{\bar{\gamma}_{b,j}[k]}}\right), \quad (9)$$

where $\bar{\gamma}_{b,j}[k]$ is the average of $\gamma_{b,j}[k]$, $I_0(.)$ is the zeroth order modified Bessel function of the first kind and K_b is

[2]We assume the same link quality from the base station to each robot and vice versa. Our framework can be easily extended to asymmetric links.

the Rician K-factor which is defined as the ratio of the non-scattered (line of sight) part of the channel power divided by the scattered (multi-path) part of it. For $K_b = 0$, Eq. 9 reduces to the pdf of an exponentially-distributed random variable. The average of $\gamma_{b,j}[k]$ can be modeled as:

$$\bar{\gamma}_{b,j}[k] \text{ (dB)} = \gamma_0^b - 10n_p \log\left(\|q_j[k] - q_b\|\right) + s_{b,j}[k], \quad (10)$$

where $s_{b,j}[k]$ is a zero-mean Gaussian random variable representing the shadowing effect and γ_0 and n_p are positive constants. For a given configuration space, the shadowing contribution can be approximated by considering the sum of the degradations caused by the obstacles (assuming that the robots and the base station do not cause any degradation on the transmitted signal).

In order to maintain the connectivity to the base station, the exact SNR map is required. Such a map, however, will not be available in practice and can only be estimated based on online learning and/or using a model-based approach. In this paper, we explore the impact of the estimation error of the SNR map on navigation and connectivity.

Another factor that plays a key role in devising communication-aware motion planning techniques is the way the receiver of each robot or the base station handles the received packets. Based on the received SNR, the receiver can decide to either keep or drop the received data. We will then have the following two designs:

- *Packet-Dropping Receiver*: By a packet-dropping receiver, we refer to a receiver that drops all the erroneous packets. In order to do so, the receiver compares the received SNR with a pre-calculated threshold necessary for error-free reception. Any packet with the received SNR below this level will be dropped (in practice this is done by looking at the output of the decoder. However, it is equivalent to using a threshold for SNR as demonstrated in [14]). We have shown in [1] that dropping all the erroneous packets is not a suitable design for delay-sensitive control applications. However, the current literature on networked control systems typically assumes that all the erroneous packets are dropped.

- *Communication-Noise Receiver*: By a communication-noise receiver, we refer to a receiver that keeps all the packets, even the erroneous ones. Then the robot can build a trust factor for every reception by using the corresponding measured received SNR [1], [2], [5].

The design of communication-aware navigation functions can change drastically depending on this underlying assumption on the receiver design. In this paper, we focus on the packet-dropping receivers and leave the communication-noise case to future extensions.

III. COMMUNICATION-AWARE MOTION PLANNING

In this section, we propose centralized and decentralized communication-aware motion planning algorithms based on using navigation functions. The communication-aware navigation functions that we introduce in this section, do not have all the properties of the traditional navigation functions

as studied by Rimon and Koditschek [9]. For instance, they may have multiple minima in the free configuration space or may be dynamic when the target is moving.[3]

In the rest of the paper, we make the following assumptions which are necessary for convergence of the proposed framework:

- The probability of more than one simultaneous collision is too low to be considered.[4]
- The target is slow enough (or equivalently the robots are fast enough) to guarantee convergence. In [8], we found an upper bound for the speed of the target in order to guarantee the convergence for the centralized case. For the ease of derivations, we furthermore assume that the target is small enough such that it can be considered a point. Hence, the collision with the target is negligible. We can easily relax this assumption for non-point targets.

In the next section, we first consider building communication-aware navigation functions, for both centralized and decentralized cases, when perfect knowledge of the SNR map is available. We then proceed to explore the impact of the SNR map uncertainty on cooperative operation.

A. Centralized Navigation Functions for Packet-Dropping Receivers

A packet dropping receiver drops any reception with received SNR below a certain threshold. Let γ_t represent this threshold. Furthermore, let "safe communication region" denote the regions with SNR to the base station above γ_t. In order to enforce all the robots to remain connected to the base station, we constrain their movements to the safe communication region. We will have

$$\gamma_{b,j} > \gamma_t \Leftrightarrow \Lambda(q_b, q_j) > 0, \qquad (11)$$

where Λ is a piecewise continuous function which is zero on the boundary of the safe communication region and positive inside. Since we considered homogeneous transceivers for all the robots, Λ will not be a function of j. Generally, the boundary of the safe communication region (specified by $\Lambda = 0$) is not smooth and may intersect some of the obstacles as well as the boundary of the workspace. This can introduce a narrower workspace with a new boundary. We assume that the initial positions of the robots as well as the whole trajectory of the target are within this new workspace. We furthermore conservatively approximate the new workspace with a spherical one. Although these assumptions are limiting in some cases, they provide a good starting point for building the mathematical framework of communication-aware motion planning using navigation functions.

We next introduce our communication-aware obstacle function, an extension to the classical definition given in [9],

which embraces the impact of obstacles, moving robots and connectivity issues [8]:

$$\beta(\mathbf{q}) \triangleq \prod_{i=1}^{N} \prod_{j=i+1}^{N+M'+1} \beta_{i,j}(\mathbf{q}), \qquad (12)$$

where M' is the total number of obstacles in the new spherical workspace, after the approximation is applied to the original workspace, and

$$\beta_{i,j}(\mathbf{q}) \triangleq \begin{cases} \|q_i - q_j\|^2 - (r_i + r_j)^2 & 1 \leq i < j \leq N + M', \\ (r_c - r_i)^2 - \|q_i - q_0\|^2 & j = N + M' + 1. \end{cases} \qquad (13)$$

The constants r_c and $q_0 \in \mathbb{R}^2$ can be found based on the applied approximation.

We first consider the case where the SNR map to the base station is perfectly known. As long as the robots are within the new spherical workspace, the communication can be considered perfect and the goal is to design a communication-aware navigation strategy that can guide the robots to the optimum configurations around the target while keeping them connected to the base station and avoiding collisions. Then the base station should build an objective function whose minima are achieved at the optimum configurations. An optimum configuration is achieved when $\det\left(\sum_{j=1}^{N} R_j^{-1}\right)$ is maximized, where R_j, the observation noise covariance of the jth robot, is as defined in Eq. 5. It can be easily confirmed that the optimum configuration is reached when $\|q_j - x\| = r_s$ for all $1 \leq j \leq N$ (the robots are at the sweet spot) and the angles, ϕ_js, maximize $\sum_{i=1}^{N} \sum_{j=i+1}^{N} \sin^2(\phi_i - \phi_j)$ [15]. It can be seen that there are several possible optimum configurations all with the same cost. Furthermore, if $\lambda = 1$ or $\varepsilon = 0$, any configuration on the sweet spot radius will be optimum independent of the angles.

The nature of our cooperative target tracking problem is, therefore, different from the typical applications of navigation functions. Traditionally, the goal is to guide a robot to a fixed destination point while avoiding fixed obstacles [9]. In this paper, we need to design an appropriate well-behaved objective function in $\mathbb{R}^{2N}$ whose minima will occur at the optimum configurations, which is a considerably challenging problem. While one could use $\det\left(\sum_{j=1}^{N} R_j^{-1}\right)$ as a possible objective function to maximize, the generated navigation function will not have its minima at the desired configurations. Therefore, in this paper, we use a suboptimal objective function which reaches its minimum at any configuration on the sweet spot that does not result in collision. In this paper, we assume that the target and as a result the sweet spot radius around it does not get too close to any fixed obstacle or the boundary of the space, which will ensure that no collision with fixed obstacles or boundary occurs when the objective is minimized. We then use the following objective function to prevent inter-robot collision [8]:

$$J(\mathbf{q}, x) = \sum_{j=1}^{N} \left(\|q_j - x\| - r_s\right)^2 + \sum_{i=1}^{N} \sum_{j=i+1}^{N} g_{i,j}\left(\|q_i - q_j\|\right), \qquad (14)$$

---

[3]It can still be shown that these functions have good properties that can result in stability and collision avoidance. See [8] for more details.

[4]This assumption can be relaxed for the decentralized case. In that case, we need the probability of more than one simultaneous collision for each robot to be negligible.

where the scalar function $g_{i,j} : [0, \infty) \to [0, \infty)$ is differentiable everywhere but the origin and is defined as follows:

$$g_{i,j}(r) \triangleq \begin{cases} a(d_{i,j} - r)^3 & 0 \le r < d_{i,j} \\ 0 & r \ge d_{i,j} \end{cases} \quad (15)$$

for a positive constant a and $(r_i + r_j) < d_{i,j} < 2r_s$. The first term of Eq. 14 results in an objective function that has its minima at the sweet spot radius from the target. It should be noted that we have a ring of minima where all the points have the same cost. By adding the second term, i.e. $\sum_{i=1}^{N} \sum_{j=i+1}^{N} g_{i,j}(\|q_i - q_j\|)$, we guarantee that we only keep those points on the sweet spot radius that result in no collision (see [8] for more details). Note that if the information of the optimal angles is available at the base station, it can be used to build more sophisticated objective functions that are well-behaved with minima at the original optimal configurations (minima of $\det \left[\sum_{j=1}^{N} R_j^{-1} \right]^{-1}$).

We now propose the following centralized navigation function for the whole system, which will be calculated at the base station:

$$\varphi(\mathbf{q}, x) = \frac{J(\mathbf{q}, x)}{\left(J^{\kappa}(\mathbf{q}, x) + \beta(\mathbf{q}) \right)^{1/\kappa}}, \quad (16)$$

where κ is a tuning parameter. The control signals are then calculated as $\mathbf{u}[k] = -\mu \nabla_{\mathbf{q}} \varphi(\mathbf{q}[k], \hat{x}[k|k])$ where $\mathbf{u}[k] = \left[u_1^T[k] \ \cdots \ u_N^T[k] \right]^T$, μ is a positive gain[5] and $\hat{x}[k|k]$ is the estimate of the base station of target position using the received information. The key points that differentiate our navigation function from the traditional ones are its time-varying nature as well as the existence of multiple minima (due to the minima of J). For more details and proof of convergence see [8].

Fig. 3 demonstrates a sample centralized target tracking scenario where a team of four mobile robots, with the same radius of 2.0, track a point target in the x-y plane. The channel is generated using the path-loss and shadowing components with no multi-path fading. The empty circles/boxes and the filled ones denote the initial and final positions respectively. It can be seen that the robots converge to the sweet spot radius.

Next we consider the impact of uncertainty in the estimation of the SNR map. If the assumed map is not accurate, then some of the nodes will lose their connectivity to the base station from time to time. Let $\mathcal{N}_b[k] \triangleq \{ j \mid \gamma_{b,j}[k] > \gamma_t \}$ represent the set of nodes that are connected to the base station at time k. Then the base station will generate the time-varying obstacle and objective functions based on the
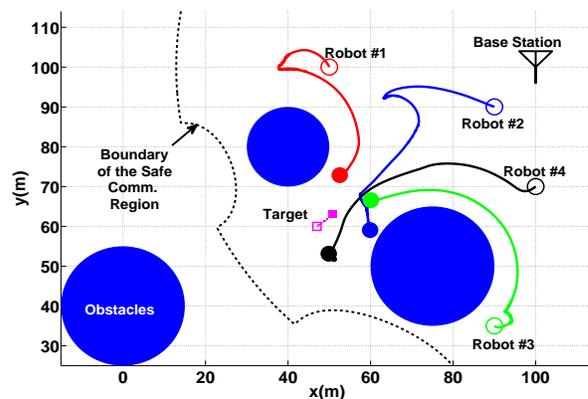

Fig. 3. Trajectories of the robots in the centralized case with packet dropping receivers and perfect SNR map.

partial information available through these nodes as follows:

$$\beta(\mathbf{q}, k) = \left[\prod_{i \in \mathcal{N}_b[k]} \prod_{\substack{j \in \mathcal{N}_b[k] \\ j > i}} \beta_{i,j}(\mathbf{q}) \right]$$

$$\times \left[\prod_{i \in \mathcal{N}_b[k]} \prod_{j=N+1}^{N+M'+1} \beta_{i,j}(\mathbf{q}) \right],$$

$$J(\mathbf{q}, x, k) = \sum_{j \in \mathcal{N}_b[k]} \left(\|q_j - x\| - r_s \right)^2$$

$$+ \sum_{i \in \mathcal{N}_b[k]} \sum_{\substack{j \in \mathcal{N}_b[k] \\ j > i}} g_{i,j}(\|q_i - q_j\|). \quad (17)$$

It should be noted that the base station generates the transformed spherical workspace based on its estimate of the SNR map.

One possible scenario that can result in a mismatch in the estimation of the SNR map could arise when the base station can estimate the path loss and shadowing components (see Eq. 10) by using the information of the obstacles. However, it cannot estimate the multi-path fading component, which is considerably more challenging to estimate. As an example, Fig. 4 and 5 show such a scenario, where there is a mismatch in the estimation of SNR due to the impact of multi-path fading. Fig. 4 shows the trajectories of the robots in a scenario similar to Fig. 3 but with Rician fading with $K_b = 2$ added. The base station, however, cannot estimate the multi-path fading component and uses the same estimate of the SNR map that was used in Fig. 3. Since the robots move randomly in case they lose their connectivity, jitters can be seen along their trajectories. Furthermore, it takes more time for the robots to reach the optimal configurations in this case (if we still have convergence). Fig. 5 shows a comparison between the noisy map and its estimate which is used at the base station. It can be seen that the mismatch can be drastic. Finally, Fig. 6 shows the percentage of time that each node is connected to the base station as a function of the Rician K-factor, averaged over 20 runs. It can be seen that channel estimation error can result in a considerable loss of connectivity, specially at low K_b. By increasing K_b, the

---

[5]μ can be constant or time-varying when a gain-scheduling algorithm is used.
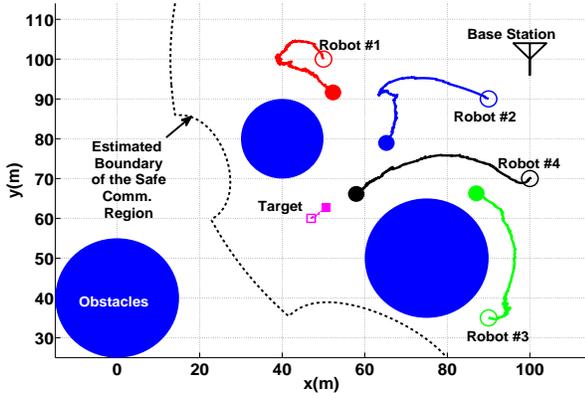
Fig. 4. Trajectories of the robots in the centralized case with packet dropping receivers and noisy SNR map.
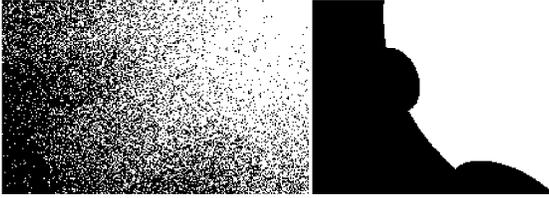

Fig. 5. (left) real SNR map and (right) its estimated version which is used at the base station. The black areas are the regions where the robots will lose their connectivity to the base station.

line of sight component increases as compared to the multi-path fading component, which results in less uncertainty in the SNR map and better communication.
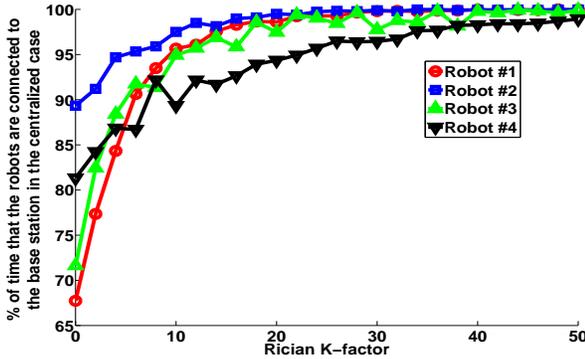

Fig. 6. Percentage of the time that the nodes are connected to the base station in the centralized case as a function of Rician K-factor.

B. Decentralized Navigation Functions for Packet-Dropping Receivers

In the decentralized case, unlike the centralized one, each robot runs its own local information filter and generates its own motion. It then sends its estimate as well as its corresponding covariance to the base station. The base station then fuses the received information.

The inter-agent communication is crucial in this case as the robots need the positions of their neighbors for obstacle avoidance. In this case, each robot considers the positions

of those robots that it can communicate with in its motion planner. However, it still only optimizes its connectivity to the base station. Assume that the receiver of each robot drops all the received packets if the received SNR is below γ_r. We take the links $i \rightarrow j$ and $j \rightarrow i$ to have the same quality. Then we define the time-varying neighbor set of the jth node as $\mathcal{N}_j[k] \triangleq \{i \mid \gamma_{i,j}[k] > \gamma_r\}$, where $\gamma_{i,j}[k]$ denotes the instantaneous received SNR between the ith and jth robots.

We now propose the decentralized and time-varying version of the previously defined obstacle, objective and navigation functions for the jth robot as:

$$\beta_j(\mathbf{q}, k) = \left[\prod_{i \in \mathcal{N}_j[k]} \beta_{i,j}(\mathbf{q}) \right] \left[\prod_{i=N+1}^{N+M'+1} \beta_{i,j}(\mathbf{q}) \right],$$

$$J_j(\mathbf{q}, x, k) = (\|q_j - x\| - r_s)^2 + \sum_{i \in \mathcal{N}_j[k]} g_{i,j}(\|q_i - q_j\|),$$

$$\varphi_j(\mathbf{q}, x, k) = \frac{J_j(\mathbf{q}, x, k)}{\left(J_j^\kappa(\mathbf{q}, x, k) + \beta_j(\mathbf{q}, k) \right)^{1/\kappa}}, \tag{18}$$

where $\beta_{i,j}$ and $g_{i,j}$ are as defined for the centralized case and κ is a tuning parameter. The control signal for the jth robot is then calculated as $u_j[k] = -\mu_j \nabla_{q_j} \varphi_j(\mathbf{q}[k], \hat{x}_j[k|k], k)$ where $\hat{x}_j[k|k]$ is the estimate of the jth robot of the target position at time k and μ_j is a positive gain.

Fig. 7 demonstrates the trajectories of the robots for the decentralized case, using the same parameters of Fig. 3. The map is taken to be perfect in this example. It should be noted that the performance of both centralized and decentralized cases can be improved by increasing κ.
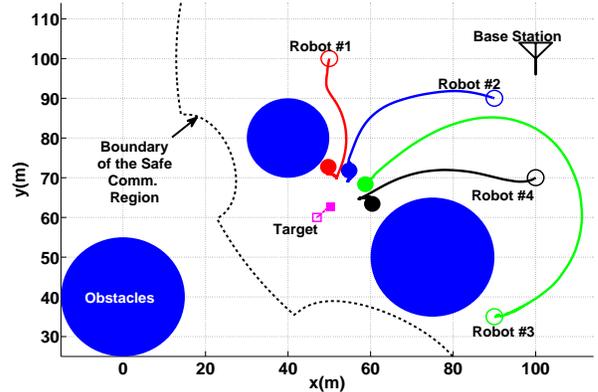

Fig. 7. Trajectories of the robots in the decentralized case with packet dropping receivers and perfect SNR map.

Fig. 8 shows the effect of SNR estimation error on the overall performance of the decentralized scenario. For this case, we added Rician fading with $K_b = 2$ to the communication with the base station as well as inter-agent communications. However, the robots cannot estimate the added multi-path fading term, which results in an estimation error in the SNR map (see Fig. 5). It can be seen that the decentralized case performs more robustly in the presence of SNR map uncertainty. This is expected as the control commands are generated locally.
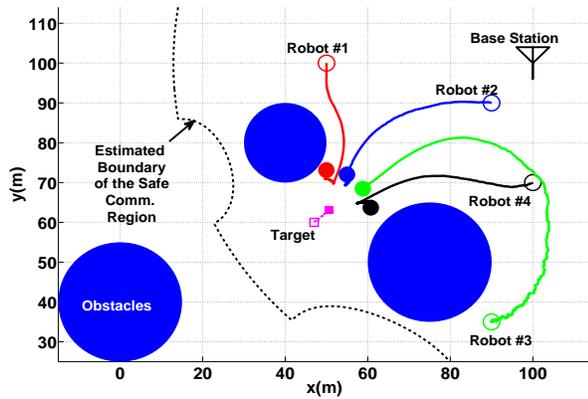
Fig. 8. Trajectories of the robots in the decentralized case with packet dropping receivers and noisy SNR map.

Finally, Fig. 9 shows the percentage of time that each node is connected to the base station as a function of the Rician K-factor. Similar behavior to Fig. 6 can be seen.
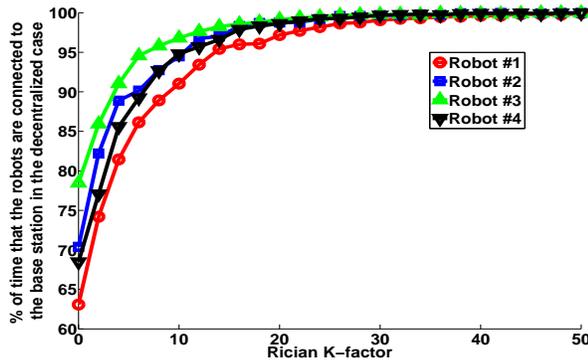

Fig. 9. Percentage of the time that the nodes are connected to the base station in the decentralized case as a function of Rician K-factor.

IV. CONCLUSIONS

In this paper we considered a cooperative target tracking scenario in which the robots need to maintain their connectivity to a fixed base station. We proposed communication-aware motion planning approaches based on using navigation functions, which enforced the nodes to maintain their connectivity while performing their task and avoiding obstacles. More specifically, we showed how to incorporate measures of link qualities in the navigation functions for both centralized and decentralized scenarios. We furthermore explored the impact of stochastic channels and channel estimation error on the overall performance.

REFERENCES

[1] Y. Mostofi and R. Murray, "Kalman Filtering over Wireless Fading Channels - How to Handle Packet Drop," to appear, *Special Issue on Control with Limited Information, International Journal of Robust and Nonlinear Control,* 2009.
[2] Y. Mostofi and R. Murray, "Effect of Time-Varying Fading Channels on the Control Performance of a Mobile Sensor Node," *Proceedings of 1st IEEE International Conference on Sensor and Adhoc Communications and Networks,* 2004.
[3] X. Liu and A. Goldsmith, "Wireless communication tradeoffs in distributed control," *42nd IEEE Conference on Decision and Control,* volume: 1, 2003, pages:688-694.
[4] Y. Mostofi, "Decentralized Communication-Aware Motion Planning in Mobile Networks: An Information-Gain Approach," in revision, *Journal of Intelligent and Robotic Systems,* 2009.
[5] Y. Mostofi, "Communication-Aware Motion Planning in Fading Environments," *IEEE International Conference on Robotics and Automation (ICRA),* May 2008.
[6] Y. Mostofi, T. Chung, R. Murray and J. Burdick, "Communication and Sensing Trade Offs in Decentralized Mobile Sensor Networks: A Cross-Layer Design Approach," *4th International Conference on Information Processing in Sensor Networks (IPSN),* April 2005.
[7] Eric W. Frew, "Information-Theoretic Integration of Sensing and Communication for Active Robot Networks," *ACM/Springer Mobile Networks and Applications Journal,* Oct. 2008.
[8] A. Ghaffarkhah and Y. Mostofi, "Communication-Aware Target Tracking using Navigation Functions - Centralized Case," to appear, *International Conference on Robot Communication and Coordination (RoboComm),* Odense, Denmark, 2009.
[9] E. Rimon and D. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation,* vol. 8, no. 5, pp. 501-518, October 1992.
[10] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos, "Nonholonomic navigation and control of multiple mobile manipulators," *IEEE Transactions on Robotics and Automation,* vol. 19, no. 1, pp. 53-64, 2003.
[11] K. Umeda and J. Ota and H. Kimura, "Fusion of Multiple Ultrasonic Sensor Data and Imagery Data for Measuring Moving ObstacleS Motion," *Intl. Conf. on Multisensor Fusion and Integration for Intelligent Systems,* December 1996, pp. 742-748.
[12] A. G.O. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*, CRC Press, 1998.
[13] W. Jakes, *Microwave Mobile Communications*, IEEE Press, 1974.
[14] D. Son, B. Krishnamachari and J. Heidemann, "Experimental Analysis of Concurrent Packet Transmissions in Low-Power Wireless Networks," *USC-ISI Technical Report ISI-TR-2005-609,* November 2005.
[15] T. H. Chung, V. Gupta, J. W. Burdick and R. M. Murray, "On a Decentralized Active Sensing Strategy using Mobile Sensor Platforms in a Network," *43rd IEEE Conf. on Decision and Control,* 2004.