Trajectory Planning for Multiple Robots in Bearing-Only Target Localisation

Cindy Leung, Shoudong Huang, Gamini Dissanayake
ARC Centre of Excellence for Autonomous Systems
Faculty of Engineering
University of Technology, Sydney, Australia
{cleung, sdhuang, gdissa}@eng.uts.edu.au

Tomonari Furukawa
ARC Centre of Excellence for Autonomous Systems
School of Mechanical and Manufacturing Engineering
University of New South Wales, Australia
t.furukawa@unsw.edu.au

***Abstract* - This paper provides a solution to the optimal trajectory planning problem in target localisation for multiple heterogeneous robots with bearing-only sensors. The objective here is to find robot trajectories that maximise the accuracy of the locations of the targets at a prescribed terminal time. The trajectory planning is formulated as an optimal control problem for a nonlinear system with a gradually identified model and then solved using nonlinear Model Predictive Control (MPC). The solution to the MPC optimisation problem is computed through Exhaustive Expansion Tree Search (EETS) plus Sequential Quadratic Programming (SQP). Simulations were conducted using the proposed methods. Results show that EETS alone performs considerably faster than EETS+SQP with only minor differences in information gain, and that a centralised approach outperforms a decentralised one in terms of information gain. We show that a centralised EETS provides a near optimal solution. We also demonstrate the significance of using a matrix to represent the information gathered.**

Index Terms – bearing only target localisation, multi-robot optimal trajectory planning, Extended Information Filter, Model Predictive Control, Sequential Quadratic Programming

I. Introduction

Cooperation among a team of distributed agents enables tasks such as target search, localisation, tracking, engagement, etc. [1]-[4], to be performed with greater efficiency than a single robot. The accuracy of information available about the targets determines which task is to be performed. If the targets are stationary and the locations of the targets are partially known, the primary task of the robots becomes target localisation or sometimes referred to as geolocation.

When the sensor on board the robot measures the relative bearing of the target without any range information, as in the case of a camera, the robot trajectory has a significant influence on the accuracy of the target location. Bearing-only target localisation has a wide range of applications in both military and civilian areas [3][5].

The optimal robot trajectory for bearing only localisation can be computed off-line when the exact locations or the trajectories of the targets are known, e.g. [6] and [7] for single robot/target. However, in real applications, the location of the targets needs to be estimated and only the estimates are available for computing the robot trajectories. With bearing-only localisation, the initial estimates may be fairly inaccurate. Undoubtedly these estimates are gradually updated as more measurements are acquired. Thus the development of optimal real time trajectory planning algorithms for multi-robot target localisation is an important research topic.

Model Predictive Control (MPC) and Sequential Quadratic Programming (SQP), have previously been applied in trajectory planning for multi-agent formations [8][9], as well as target tracking and engagement [10][9], as they incorporate constraints in the planning process and have a capability of allowing feedback at each planning horizon. In our other current work [11], MPC is applied to trajectory planning in Simultaneous Localisation and Mapping (SLAM).

In this paper, we consider the trajectory planning for target localisation using heterogeneous robots with bearing only sensors. The objective of the trajectory planning is to maximise the estimation accuracy about the locations of the targets [5]. We assume that all the targets are stationary and the precise locations of the robots are known at any time. It is also assumed that collision between the robots will not occur (for example a set of UAVs flying at different altitudes). We first show that the trajectory planning problem is an optimal control problem for a gradually identified nonlinear system. Later on, the multi-step optimisation problem used in MPC is formulated and two optimisation strategies are proposed to solve it. An efficient solution using one optimisation strategy, Exhaustive Expansion Tree Search (EETS) is provided. Then we show another strategy using SQP which refines the solution from EETS by using the latter as an initial guess to a non-linear constrained optimisation problem. It is demonstrated that EETS alone provides a near-optimal solution.

The paper is organised as follows. In Section II, the information accumulation process in multi-robot target localisation is stated using Extended Information Filter (EIF). In Section III, the multi-step trajectory planning problem is formulated. In Section IV, nonlinear MPC and the multi-step prediction methods are introduced. In Section V two multi-step optimisation strategies are presented. In Section VI simulation results are provided to demonstrate the optimality and effectiveness of the proposed methods. In Section VII the computational complexity of these methods are addressed. Finally Section VIII concludes the paper.

II. Multi-Robot Targets Localisation Algorithm

In this section, we state how the information about the targets evolves in the multi-robot target localisation. Here we make use of the Extended Information Filter (EIF) based multi-robot target localisation algorithm [5][12].

Suppose there are n robots and m targets. The target localisation problem is to estimate the locations of the m targets using the information obtained from a sequence of observations made by the n robots. The sensors on the robots have limited viewing range, scope and precision. Accuracy of the estimation is characterised by the information matrices.

Discrete-time models are used throughout this paper. We assume that the observations are made at each time step and consider a finite time horizon [0, T] where T is a given integer.

A. Notations

k – k=0, 1,…,T – observation and planning time step

t – t= k, $k+{}^{1}/_{\Delta}$,…, $k+{}^{\Delta-1}/_{\Delta}$ – robot motion time step, where Δ is a positive integer

$\mathbf{x}_{r_i}(t) = (x_{r_i}(t), y_{r_i}(t), \phi_{r_i}(t))^{\mathrm{T}}$ i=1,…,n, t=k,…,k+1 – location and heading of the n robots at time t

$\mathbf{u}_i(k) = (v_i(k), \omega_i(k))^{\mathrm{T}}$, i=1,…,n – control vector (velocity v and turn-rate ω) of the n robots at time k

$\mathbf{x}_{f_j} = (x_{f_j}, y_{f_j})^{\mathrm{T}}$ j=1,…,m – true locations of the m targets

$\hat{\mathbf{x}}_{f_j}(k)$ j=1,…,m – estimated locations of the m targets at time k (after the observation and update made at time k)

$\mathbf{I}_j(k)$, j = 1,…, m – information matrix (2 × 2) about the location for the m targets at time k (after the observation and update made at time k), equivalent to the inverse of the covariance matrix $\mathbf{P}_j(k)^{-1}$

$z_{ij}(k)$ – observation of the robot i observing target j at time k

R_i, i = 1, …, n – covariance of the observation of the n robots

$\mathbf{I}_{ij}(k)$, i = 1,…, n, j = 1, …, m – the information (matrix) obtained from the observation of the ith robot observing the jth target at time step k

B. The Process and Observation Model

Suppose the discrete-time process model of the ith robot is

$$\mathbf{x}_{r_i}(t+\tfrac{1}{\Delta}) = \mathbf{f}_i(\mathbf{x}_{r_i}(t), \mathbf{u}_i(k)) \tag{1}$$

where $\mathbf{f}_i$ is a nonlinear function which depends on the type and the dynamic model of the heterogeneous robots. This process model, $\mathbf{f}_i$, is iteratively evaluated from k to $k+1$, with the same control, $\mathbf{u}_i(k)$, using the smaller time step, t, in increments of ${}^{1}/_{\Delta}$ to generate a smoother trajectory.

The observation model (at time $k+1$) of the ith robot (observing the jth target) is

$$z_{ij}(k+1) = h_i(\mathbf{x}_{r_i}(k+1), \mathbf{x}_{f_j}) + w_i \tag{2}$$

where w_i is a zero-mean Gaussian measurement noise with covariance matrix R_i and h_i is a nonlinear function which depends on the model of the sensor equipped on the ith robot.

C. Information Obtained from the Observations

By the EIF approach [5][12][13], the information obtained from the observation $z_{ij}(k+1)$ is a matrix

$$\mathbf{I}_{ij}(k+1) = \mathbf{H}_{ijk}^{T} R_i^{-1} \mathbf{H}_{ijk} \tag{3}$$

where $\mathbf{H}_{ijk}$ is the Jacobian of h_i, with respect to the feature state $\mathbf{x}_{f_j}$, evaluated at $(\mathbf{x}_{r_i}(k+1), \hat{\mathbf{x}}_{f_j}(k))$. That is

$$\mathbf{H}_{ijk} = \nabla_{\mathbf{x}_{f_j}} h_i \,|_{(\mathbf{x}_{r_i}(k+1), \hat{\mathbf{x}}_{f_j}(k))}. \tag{4}$$

If target j is out of the sensor range of robot i at time $k+1$ then $\mathbf{I}_{ij}$ would simply be a zero matrix. This means that no new information about target j is obtained from the observation taken by robot i. Since n independent observations (from the n robots) are made for target j at time $k+1$, the new information obtained about target j at time $k+1$ is (5). The total information about target j at time $k+1$ (after the observations), $\mathbf{I}_j(k+1)$, can be calculated by (6), where $\mathbf{I}_j(k)$ is the total information obtained about the target j at time k.

$$\sum_{i=1}^{n} \mathbf{I}_{ij}(k+1) \tag{5}$$

$$\mathbf{I}_j(k+1) = \mathbf{I}_j(k) + \sum_{i=1}^{n} \mathbf{I}_{ij}(k+1) \tag{6}$$

D. Update of the Estimation

At time k and $k+1$, the estimation of the location of target j are $\hat{\mathbf{x}}_{f_j}(k)$ and $\hat{\mathbf{x}}_{f_j}(k+1)$, respectively. The update of the estimation takes the form

$$\hat{\mathbf{x}}_{f_j}(k+1) = \hat{\mathbf{x}}_{f_j}(k) + \mathbf{I}_j^{-1}(k+1) \sum_{i=1}^{n} \mathbf{i}_{ij}(k+1) \tag{7}$$

where $$\mathbf{i}_{ij}(k+1) = \mathbf{H}_{ijk}^{\mathrm{T}} R_i^{-1} \mu_{ij}(k+1), \tag{8}$$

here $\mathbf{H}_{ijk}$ is given by (4) and $\mu_{ij}(k+1)$ is the innovation defined by

$$\mu_{ij}(k+1) = z_{ij}(k+1) - h_i(\mathbf{x}_{r_i}(k+1), \hat{\mathbf{x}}_{f_j}(k)). \tag{9}$$

These EIF equations are slightly different from the typical EIF equations using the information vector [12][13], but they are equivalent.

III. The Trajectory Planning Problem

A. Problem Statement

For bearing only target localisation it is critical for the robots to observe the targets from different angles, so the robot trajectories play an important role in the information gathering. In this paper, we consider the trajectory planning problem on a finite time horizon [0, T]. The objective is to maximise the total information (estimation accuracy) at time step T by choosing suitable control actions for the n robots. The trajectory planning problem can be stated as follows.

Problem: *Suppose at time step* 0, *the positions of the robots are* $\mathbf{x}_{r_i}(0)$, *the initial estimate of the positions of the m targets are* $\hat{\mathbf{x}}_{f_j}(0)$ *and the information on the targets are* $\mathbf{I}_j(0)$. *Decide how to choose the control actions,* $\mathbf{u}_i$, *for the n robots from time k*=0 *to time k*=T – 1,

$$\mathbf{u}_i(0), \mathbf{u}_i(1), \ldots, \mathbf{u}_i(T-1), \quad i = 1, \ldots, n, \tag{10}$$

such that $$\min_{1\le j\le m} \min(eig(\mathbf{I}_j(T))) \tag{11}$$

is maximised, where $\mathbf{I}_j$ *(T) is the information matrix of the jth target at time T which is obtained by* (12)

$$\mathbf{I}_j(T) = \mathbf{I}_j(0) + \sum_{k=0}^{T-1} \sum_{i=1}^{n} \mathbf{I}_{ij}(k+1) \tag{12}$$

and $\mathbf{I}_{ij}(k+1)$ *is given by* (3).

We use the scalar from (11) as an indication of the inverse of the maximum uncertainty of the *m* targets. Uncertainty can be represented as an ellipse and the longest axis of the ellipse represents the greatest uncertainty. If we maximise (11), the maximal axis of the uncertainty ellipses is minimised.

Generally speaking, the initial target locations can be obtained from other available sources of information before the target localisation starts, if not, methods such as particle filters or nonlinear least squares can be applied to obtain an initial estimation of the targets.

B. *An Optimal Control Problem for System with Gradually Identified Model*

At first glance, the trajectory planning problem is a finite horizon optimal control problem for a nonlinear control system. The model of the "control system" is the compilation of equations (1), (3), (4), (6), (7), (8), and (9).

The objective is to choose the control (10) in order to maximise the performance measure (11).

Notice that in (3) and (6), $\mathbf{H}_{ijk}$ is needed to calculate $\mathbf{I}_j(k+1)$, and in (4) $\hat{\mathbf{x}}_{f_j}(k)$ is needed to calculate $\mathbf{H}_{ijk}$, but when we update $\hat{\mathbf{x}}_{f_j}(k+1)$ in (7), (8), and (9), the observation $z_{ij}(k+1)$ is required. Because the observation $z_{ij}(k+1)$ is not available until time $k+1$, at time 0, we can not obtain a clear relationship between the control actions (10) and the performance measure (11). Hence the above model is a **nonlinear control system with a gradually identified model**.

IV. Nonlinear Model Predictive Control

Seeing as the model is gradually identified, Nonlinear Model Predictive Control (MPC) is a natural and optimal strategy [11] to solve the trajectory planning problem (MPC is a strategy where *N*-steps are predicted ahead of time to obtain *N* control actions for *N*-steps and only the first control action is applied). This strategy is repeated at each time step *k* until $k = T$.

An important step for MPC is to predict the maximal information gain after *N*-steps.

A. *Prediction in N-step Optimisation*

At step 0, $\mathbf{x}_{r_i}(0)$, $\hat{\mathbf{x}}_{f_j}(0)$ and $\mathbf{I}_j(0)$ are known. By (1), (3), and (4), $\mathbf{x}_{r_i}(1)$ are functions of the control input at time 0 – $\mathbf{u}_i(0)$, and hence $\mathbf{H}_{ij0}$ are functions of $\mathbf{u}_i(0)$. So $\mathbf{I}_j(1)$ are functions of $\mathbf{u}_i(0)$. (Notice that $\mathbf{I}_j(1)$ does not depend on the observations $z_{ij}(1)$ though $\hat{\mathbf{x}}_{f_j}(1)$ does).

Since the observation $z_{ij}(1)$ is not available at time 0, it seems impossible to predict more than one-step ahead. However, before the observation $z_{ij}(1)$ is made, we can assume that the innovation, $\mu_{ij}(1) = z_{ij}(1) - h_i(\mathbf{x}_{r_i}(1), \hat{\mathbf{x}}_{f_j}(0))$, is a random variable with zero mean (because in the EIF implementation, we always assume that the distribution of the true target location, $\mathbf{x}_{f_j}$, is Gaussian with mean $\hat{\mathbf{x}}_{f_j}(0)$).

Actually, at time 0, this is the only possible assumption if we want to look multi-steps ahead because we have no idea how much the estimated target location deviates from the true target location. The assumption needed for the *N*-step optimisation is the following:

Assumption I. The innovations at any time are zero. i.e.

$$\mu_{ij}(k+1) = z_{ij}(k+1) - h_i(\mathbf{x}_{r_i}(k+1), \hat{\mathbf{x}}_{f_j}(k)) = 0 \tag{13}$$

for all $i = 1, \ldots, n, j = 1, \ldots, m, k = 0, \ldots, N-1$.

B. *N-step Optimisation Problem*

Under the above assumption, the *N*-step optimisation problem becomes the following

***N*-step optimisation problem.** Given $\mathbf{x}_{r_i}(0)$, $\hat{\mathbf{x}}_{f_j}(0)$ and $\mathbf{I}_j(0)$, choose control (10) to maximise measure (11) where $\mathbf{I}_j(N)$ is given by

$$\begin{aligned}
&\mathbf{I}_j(k+1) = \mathbf{I}_j(k) + \sum_{i=1}^{n} \mathbf{I}_{ij}(k+1) \\
&\mathbf{I}_{ij}(k+1) = \mathbf{H}_{ijk}^T R_i^{-1} \mathbf{H}_{ijk}, \\
&\quad \mathbf{H}_{ijk} = \nabla_{\mathbf{x}_{f_j}} h_i \,|_{(\mathbf{x}_{ri}(k+1), \hat{\mathbf{x}}_{f_j}(k))}, \\
&\mathbf{x}_{r_i}(t + \tfrac{1}{\Delta}) = \mathbf{f}_i(\mathbf{x}_{r_i}(t), \mathbf{u}_i(k))
\end{aligned} \tag{14}$$

for $i=1,\ldots, n, j=1,\ldots, m, k=0, 1,\ldots, N-1, t= k, k+{}^{1}/_{\Delta},\ldots, k+{}^{\Delta-1}/_{\Delta}$.

The above system is a deterministic system and the optimisation problem can be solved by different optimisation techniques. Two of those are described below.

V. Optimisation Techniques

A. *Exhaustive Expansion Tree Search (EETS)*

One solution to this optimisation problem is to apply EETS to conduct a coarse exhaustive search. In an EETS, the system is fed N_ω discrete control actions, this is the number of possible options the robot can take at each time step *k*. Using (14), each robot *i* can move to N_ω different positions $\mathbf{x}_{r_i}(k+1)$ at time $k+1$ if they were to apply N_ω separate controls $\mathbf{u}_i(k+1)$ over the period $t = k$ to $t = k+1$. At the N_ω different positions $\mathbf{x}_{r_i}(k+1)$, N_ω different $\mathbf{I}_{ij}(k+1)$ matrices are evaluated using (14).

To predict for time $k+2$, the robots can move to N_ω positions from each of the N_ω positions at time $k+1$ in a tree spanning pattern. This means that each robot can be at $N_\omega \times N_\omega$ different positions at time $k+2$ if they had N_ω control choices at each time step *k*. Consequently at time $k+2$ each

robot would also have $N_\omega \times N_\omega$ different $\mathbf{I}_{ij}(k+2)$ matrices.

At the Nth prediction step, each robot would have $(N_\omega)^N$ different $\mathbf{I}_{ij}(k+N)$ matrices. Using (14), $(N_\omega)^{Nn}$ combinations of $\mathbf{I}_{ij}(k+N)$ are available to evaluate $\mathbf{I}_j(k+N)$. The control

$$\mathbf{u}_i(k+1), \mathbf{u}_i(k+2), \ldots, \mathbf{u}_i(k+N), \quad i = 1, \ldots, n \tag{15}$$

is chosen to maximise measure

$$\min_{1 \le j \le m} \min(eig(\mathbf{I}_j(k+N))) \tag{16}$$

Evidently computation for considering all combinations is exponential, $O(m \times (N_\omega)^{Nn})$.

Constraints of no-go-zones can be enforced by an explicit condition based on predicted target and robot locations.

Since no information is obtained when the targets are out of the robot's sensor range (as described in Section II C), it is possible that robot i cannot gain any information even after looking at all possible trajectories in N steps. Thus, it would not be able to determine which trajectory is optimal because

$$\sum_{j=1}^{m} \mathbf{I}_{ij}(k+N) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{17}$$

In the event of this, EETS is repeated using longer step, i.e. N remains the same but at every prediction step, information $\mathbf{I}_{ij}$ is evaluated at every $c \times k$ (where c is the number of times the algorithm has cycled). Thus the robots are predicted to travel longer in a time step of $c \times k$ with a single control for each step $c \times k$. As previously the control for only one step $k+1$ is applied.

Although this is a coarser search, it allows the robot to look further ahead providing it an idea of the direction to head at the next step $k+1$. This strategy is preferred than to increasing N at every repetition because doing so would severely burden the computation.

B. *EETS + Sequential Quadratic Programming (SQP)*

The benefit of an exhaustive search is that it finds the global optimum among the finite control options. However, it can be argued that a coarse exhaustive search would not obtain the optimal solution because only a few discrete options are available. We present an alternative method for the N-step optimisation. This method is also used to evaluate our results from EETS. We combine EETS with SQP because unlike EETS, SQP is continuous and has infinite solutions. In general, SQP finds the local optimum solution. Hence feeding SQP with a random guess would not result in a good performance. Instead the system is fed the control sequence (15) for each robot that was obtained from EETS. This way the SQP method is given an initial condition that is substantially close to the global optimum solution so that the coarseness of EETS can be refined.

The constraints on control actions including no-go-zones can be easily incorporated into the SQP algorithm.

VI. Simulation Results

Simulations of the complete system are conducted using 2 robots and 2 targets. Initial conditions $\mathbf{x}_{r_i}(0)$, $\hat{\mathbf{x}}_{f_j}(0)$ and $\mathbf{I}_j(0)$ are set and $\mathbf{u}_i(k)$ is selected from a 3-step look-ahead ($N = 3$). The velocity is kept constant and the turn-rate $\omega_i(k)$ varied at each step k, depending on the result of the optimisation algorithm. The process model for the robot motion was iterated $\Delta = 10$ times for every step k. There is a constraint that no robots can enter the circular no-go zones centred by the two targets with radius 2. Sensors on the robots are limited to a view of -90 to 90 degrees across the front of the robot with a range of 20.

A. *Evaluation of the EETS*

Many trials were conducted to compare the computation times and information gain using EETS with $N = 3$, and the result after passing it to the SQP algorithm for refinement. The results for 10 trials are presented in Table I.

TABLE I
Simulation Results

Trial	– Optimal Control Strategy – Information Gain (Computation Time (s))			
	EETS N_ω=3	EETS (N_ω=3) + SQP	EETS N_ω=9	EETS (N_ω=3) Decentralised
1	**3.5947** (0.12)	**3.5947** (0.26)	**3.5947** (39.72)	2.0748 **(0.07)**
2	**2.2478** (0.13)	**2.2478** (0.27)	**2.2478** (39.76)	1.4044 **(0.08)**
3	**7.1881** (0.07)	**7.1881** (0.22)	**7.1881** (8.95)	6.8754 **(0.05)**
4	**5.2361** (0.09)	**5.2361** (0.24)	**5.2361** (25.16)	4.2059 **(0.06)**
5	0.3866 (0.12)	**0.3869** (3.06)	0.3868 (36.96)	0.2681 **(0.07)**
6	1.3465 (0.12)	**1.3696** (5.53)	1.3680 (36.88)	0.2652 **(0.07)**
7	1.4133 (0.11)	**1.4569** (4.50)	1.4530 (29.42)	0.9449 **(0.07)**
8	0.9559 (0.08)	**1.0025** (4.63)	0.9929 (11.89)	0.5803 **(0.05)**
9	1.3003 (0.20)	**1.3109** (4.95)	1.3082 (74.59)	0.7745 **(0.08)**
10	2.0831 (0.08)	**2.1418** (0.37)	2.1405 (21.10)	1.5059 **(0.06)**

From Table I the combination EETS+SQP performs slightly better in terms of information gain than EETS alone, as shown in Trials 5-10. There are also instances where the information gains are equivalent, as seen in Trials 1-4. There are two main explanations for this occurrence. 1) From Trial 1, the difference in turn-rate for one of the time steps after executing EETS+SQP is only 0.001rad/s less than EETS alone. The differences in information gain from bearing-only observations from very similar positions are not discernable. 2) In Trial 3, SQP did not make any adjustments to the control actions from EETS. They were all set to the maximum turn-rate. In the case of bearing-only localisation, often the highest information gain is when the observation is taken at an angle that is perpendicular to the previous observation. So to maximise the information gain, the robots would often be driven at the maximum turn-rate to go around the targets. In 200 trials, control actions were evaluated to be the maximum turn-rate 70 percent of the time.

Although EETS+SQP has the best information gain (on average 1.3% better than EETS N_ω=3), it has a significantly longer computation time. The time increases depending on

how far the initial condition is from the optimal. Hence it may require more iterations to find the solution, as in Trials 5-10.

We also conducted trials to compare EETS with nine options (i.e. N_ω=9) instead of N_ω=3. This results in a finer search. As seen in Table I, the computation time has increased significantly (approx. 300 times greater) without gaining much in terms of information.

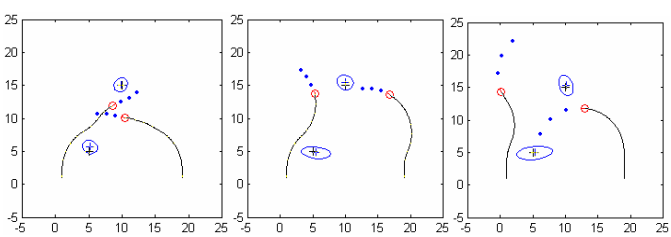
B. *Decentralised vs. Centralised*

The algorithms considered so far are all centralised. We were interested to see how a decentralised approach compares to a centralised one. The last column of Table I contains the results for a decentralised approach where the information about the targets are shared but the possible trajectory or positions of the other robot are not taken into account in the planning. It can be seen that the computation time for the decentralised approach is the fastest of all the approaches. However as seen in Table I, the performance by combining individual optimal controllers results in inferior global performance. The information gain is consistently the lowest, on average 35% less information than centralised EETS.

C. *Significance of Information Represented as a Matrix*

To demonstrate the significance of information represented as a matrix, we simulate EETS given three sets of initial information matrices $\mathbf{I}_j(0)$ all with the same eigenvalues ($eig(\mathbf{I}_1(0))$=0.4384, 4.5616; $eig(\mathbf{I}_2(0))$=0.6972, 4.3028) and determinants ($det(\mathbf{I}_1(0))$=2, $det(\mathbf{I}_2(0))$=3).

Simulations are conducted with a terminal time of T = 10. Fig. 1 shows the different trajectories resulting from different initial information matrices.



(a) $\mathbf{I}_1$ = diag[0.4384, 4.5616] (b) $\mathbf{I}_1$ = [3 2; 2 2] (c) $\mathbf{I}_1$ = diag[4.5616, 0.4384]
$\mathbf{I}_2$ = diag[4.3028, 0.6972] $\mathbf{I}_2$ = [1 1; 1 4] $\mathbf{I}_2$ = diag[0.6972, 4.3028]

Fig. 1 Trajectories of the two robots, with different $\mathbf{I}_j(0)$.

Given the dimension of one information matrix is a 2 × 2 symmetric matrix instead of a scalar representation of information, trajectories can be planned based on detailed knowledge of the uncertainty.

D. *Following the Constraints*

With a 3-step look-ahead, the robots are prevented from moving to a position where violating a constraint is unavoidable in the next step. This may occur with a one-step look-ahead where the robots move close to the target to gain more information but not considering they might be trapped in the following step. However this is subject to the accuracy of the estimations of the target location. In Fig. 2 the no-go-zones are depicted by the circles around the targets.

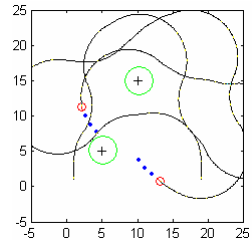


Fig. 2 Trajectories outside the no-go-zone after k=80 time steps

VII. Online Computation

In this section, we justify the necessity of the proposed method and show how the special features of the target localisation problem make the near optimal solution tractable.

A. *N-step look-ahead and MPC*

By planning a few steps ahead, the near optimal route can be chosen by assessing the maximum predicted future reward. Use of the MPC strategy allows for continual feedback of new target information to be incorporated into the planning system. Increasing the number of steps look-ahead, N, provides a solution closer to the global optimum, but the computational complexity increases exponentially with respect to N. Using a small N maintains reasonably fast computation times.

In the target localisation problem, the frequency of the observations may not be very high. Especially, when the velocities of the robots are large (such as UAVs) and/or the processing of the sensor data of the observations is time consuming (e.g. when the observations are made by cameras), thus only a few time steps N look-ahead are needed.

Furthermore, the assumption in N-step optimisation may not always hold. When the innovations are far from zero, (see Assumption I in Section IV A), the control action obtained by the N-step optimisation strategy may be far from optimal. Thus using a large N may not provide any guaranteed rewards.

B. *Searching Possible Control Options*

Search algorithms are necessary to systematically select possible control actions that are within the constraints to obtain the optimal solution. Since there is no perfect nonlinear optimization algorithm available to obtain a global optimal solution, EETS and EETS+SQP are proposed to choose the control actions.

Although the computational complexity of EETS grows exponentially with respect to the number of options N_ω, keeping N_ω small reduces the computational cost while having only a minor tradeoff in information gain, as seen from Table I. This is due to the nature of the target localisation problem; if two observations are made at points very close to each other, then the information obtained from the observations will be similar.

Fig. 3 shows the information gain as a function of the control actions (turning-rates) is relatively smooth, (the steep drop is due to the control actions moving the robots so that the targets are out of the sensor's field of view).

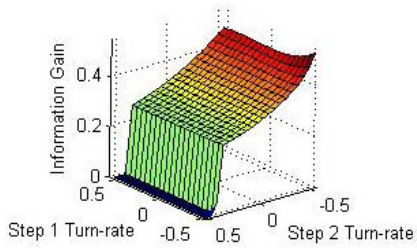


Fig. 3 Graph of Information gain vs. Turn-rates

We also conducted 34 trials with one robot, one target and N=2, to compare N_ω=129 and N_ω=3, samples of the results are shown in Table II. There was only a 4% improvement in information gain with N_ω=129. Hence, in the trajectory planning problem, it is not necessary to consider all possible control actions, it is appropriate to select only a small number of possible control actions which result in significantly different observing points in the next time step.

TABLE II
EETS

N_ω	Information Gain						
3	0.2346	1.3621	0.8192	1.5429	1.1056	1.7024	1.7184
129	0.2596	1.3621	0.8192	1.5435	1.1056	1.7178	1.7425

C. *Centralised Control*

Despite that the computational complexity for centralised control increases exponentially with respect to the number of robots *n,* decentralised control without coordination may perform poorly compared to centralised control. An extreme case is when the two robots start from similar positions, trajectories obtained from optimal decentralised control will intuitively result in similar trajectories for each robot, which is far from the global optimal, where the robots will separate. Better performance may be achieved by decentralised control combined with negotiation of robots (e.g. [5]) but the negotiation process is more complicated. Hence, if centralised control is not used in the trajectory planning there is no guarantee that the near optimal solution can be achieved.

In the target localisation problem, the number of actively cooperating robots may be small in general. The cooperative control problem for a large number of robots can often be decoupled into two sub-problems; a task assignment problem and an actively cooperative control problem for only a small group of robots.

VIII. Conclusion and Further Work

In this paper, trajectory planning problem with heterogeneous robots and bearing only target localisation has been investigated. It has been shown that this is an optimal control problem for a nonlinear system with a gradually identified model and a nonlinear MPC strategy is proposed.

Simulations were presented comparing the information gain and computation times of different optimisation strategies, using two robots and two targets with 3-step look-ahead MPC. The results show that on average EETS ($N_\omega = 3$) is 20 times faster than EETS+SQP with similar gains in information. Simulation results also demonstrated that centralised control obtained, on average, 35% higher information gain than decentralised control.

According our analysis and simulation results, we make the following suggestions to obtain a solution for high performance: 1) Information matrices and EETS should be used to plan the trajectory to give a near optimal performance in terms of time and information gain. 2) Only a small number of possible control actions are needed to be considered to reduce the computational complexity. 3) Keep the planning horizon N relatively short to reduce the computation. 4) Use centralised control for optimal robot coordination.

Further research work includes how to quantitatively decide the number of possible control actions, the number of prediction steps and the limit of the number of robots that can be accommodated within a given computational capacity. The extension of the proposed techniques with the possibility of robot collision and time-delays in the robot communications [2] can be taken into account.

Acknowledgment

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

References

[1] F. Bourgault, T. Furukawa and H. F. Durrant-Whyte, "Decentralized Bayesian Negotiation for Cooperative Search." *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Sendai, Japan. 28 vol.3 pp. 2681 - 2686 Sept.-2 Oct. 2004

[2] K. Passino, *et al*. "Cooperative Control for Autonomous Air Vehicles." *Proc. of the Cooperative Control Workshop*, Chapter 1, FL. Dec, 2000.

[3] B. Ristic, and M. Sanjeev Arulampalam, "Tracking a Manoeuvring Target using Angle-only Measurements: Algorithms and Performance," *Signal Processing*, vol. 83, pp. 1223-1238, 2003.

[4] T. Furukawa, G. Dissanayake, and H. F. Durrant-Whyte, "Time-optimal Cooperative Control of Multiple Robot Vehicles," *Proc. 2003 International Conference on Robotics and Automation*, Taipei, pp. 944 – 950, vol.1, 14-19 Sept., 2003.

[5] B. Grocholsky, "Information-Theoretic Control of Multiple Sensor Platforms" PhD Thesis, The University of Sydney.

[6] Y. Oshman and P. Davidson. "Optimization of Observer Trajectories for Bearings-only Target Localization." *IEEE Trans. on Aerospace and Electronic Systems*, 35(3):892-902, 1999.

[7] J.M. Passerieux and D. V. Cappel. "Optimal Observer Maneuver for Bearings-only Tracking." *IEEE Trans. on Aerospace and Electronic Systems*, 34(3):777-788, 1998.

[8] K. Wesselowski, R. Fierro, **"**A Dual-mode Model Predictive Controller for Robot Formations", *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 4, pp. 3615-3620, 9-12 Dec. 2003

[9] L. Feng-Li, R. Murray, **"**Cooperative Task Planning of Multi-robot Systems with Temporal Constraints"; *Proc. IEEE Int. Conf. on Robotics and Automation,* vol. 2, pp. 2504 – 2509, 14-19 Sept. 2003

[10]J.E. Tierno, "Distributed Autonomous Control of Concurrent Combat Tasks" *Proc. American Control Conf.,* vol. 1, pp. 37-42, 25-27 June 2001

[11]S. Huang, N. M. Kwok, G. Dissanayake, Q.P. Ha and G. Fang. "Multi-Step Look-Ahead Trajectory Planning in SLAM: Possibility and Necessity," *2005 IEEE Int. Conf. on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 1103 – 1108.

[12]P. Maybeck, "Stochastic Models, Estimation, and Control", Vol.1, Academic, New York, 1979.

[13]S. Thrun, *et al*., "Simultaneous Localization and Mapping with Sparse Extended Information Filters," *International J. of Robotics Research*, vol. 23, no. 7-8, pp. 693-716, 2004