International Journal of Control
Vol. 80, No. 7, July 2007, 1169–1179

Autonomous and decentralized mission planning for clusters of UUVs

L. GIOVANINI, J. BALDERUD* and R. KATEBI

Industrial Control Centre, University of Strathclyde, Glasgow, Graham Hill Building,
50 George St., Glasgow G1 1QE, UK

(*Received 1 June 2006; in final form 31 January 2007*)

This paper proposes an algorithm for autonomous strategic mission planning of missions where multiple unhabitated underwater vehicles (UUVs) cooperate in order to solve one or more mission tasks. Missions of this type include multi-agent reconnaissance missions and multi-agent mine sweeping missions. The mission planning problem is posed as a receding horizon mixed–integer constrained quadratic optimal control problem. This problem is subsequently partitioned into smaller subproblems and solved in a parallel and decentralized manner using a distributed Nash-based game approach. The paper presents the development of the proposed algorithm and discusses its properties. An application example is used to further demonstrate the main characteristics of the proposed method.

1. Introduction

This paper considers the problem of allocating resources and assigning tasks in a multi-agent system. Systems of this type are characterized by their requirement for coordination and cooperation between the agents (Liu *et al*. 2003). However, whilst coordination and cooperation between agents are desirable, it can be complicated to implement in practice. To perform missions that exhibit these features, acceptable algorithms must be solved in real time, taking into account the need for task precedence and coordination, timing constraints and feasible trajectories (Tews and Wyeth 2000). One of the most difficult features of a cooperative control problem is complexity, which results from the size of the problem and interactions between agents and tasks (Chandler *et al*. 2002a).

To solve these complex planning problems different classes of methods have been proposed. These include: mixed integer linear programming (*MILP*) methods (Richards *et al*. 2002, Schumacher *et al*. 2004), capacitated transshipment methods (Schumacher *et al*. 2002a) and iterative capacitated transshipment methods (Chandler *et al*. 2002a). Due to the special characteristics of the problem and the requirement for a tractable solution, all of the proposed algorithms are suboptimal in some sense. For example the *MILP* algorithm of Richards *et al*. (2002) uses Euclidean distances while that of Schumacher *et al*. (2004) uses piecewise trajectories between targets and hence both do not take into account of the need for feasible trajectories. The single task assignment algorithm (Schumacher *et al*. 2002a) is only optimal for the current task and do not take into account tasks that will be carried out when the current task is completed. Although a variation on the single task assignment algorithm (that utilize in essence a greedy solver) (Schumacher *et al*. 2002b) provides a solution to the multiple task assignment problem, it is heuristic in nature and therefore is not optimal. Note also that for most problems these algorithms generally take a long time to set up and execute (Kang *et al*. 2001).

Several cooperative control algorithms have also been proposed, implemented, and simulated (Murphy 1999, Nygard *et al*. 2001, Schumacher *et al*. 2001, Chandler *et al*. 2002b, Guo and Nygard 2002, Alighanbari *et al*. 2003), but due complexity issues these have been heuristic in nature. Many of these control algorithms also do not meet all of the requirements of the

*Corresponding author. Email: balderud@eee.strath.ac.uk

International Journal of Control
ISSN 0020–7179 print/ISSN 1366–5820 online © 2007 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/00207170701253363

assignment problem, i.e., assignment coordination, task precedence, and feasible trajectories. An exception is the tree generation algorithm (Rasmussen *et al.* 2003), which produces optimal solutions to the assignment problem based on piecewise optimal trajectories. The algorithm generates a tree of feasible assignments and then by exhaustive search finds the optimal assignment. During generation of the tree all of the requirements of the mission are met, but since enumeration of all of the feasible assignments is needed, direct use of this approach is only practical for relatively low dimensional problems and off-line applications. For on-line applications a branch and bound algorithm has been proposed (Rasmussen *et al.* 2004). This deterministic search method has desirable qualities such as providing feasible solutions, that monotonically improves and, eventually, converges to the optimal solution.

This paper proposes an algorithm for strategic mission planning of missions where multiple unhabitated underwater vehicles (UUVs) cooperate in order to solve one or more mission tasks. The planning problem is posed as a receding horizon mixed-integer constrained optimal control problem, in which the mission tasks are represented using a set of integer variables and set of objective functions and in which the mission environment is represented using constraints.

Existing mission planning algorithms typically employ centralized solution approaches (e.g., Schumacher *et al.* (2002a) and Shima *et al.* (2005)). The proposed algorithm by contrast employs a distributed Nash-based solution approach (Neck and Dockner 1987, Waslander *et al.* 2003), in which the planning problem is divided into a set of subproblems that are distributed between the mission agents. A partitioning scheme for the planning problem is furthermore proposed that reduces the interactions and dependence between the subproblems.

The subproblems, when solved using the proposed Nash-based solution approach yields a globally optimal solution to the strategic planning problem. However, each subproblem can also be solved independently, in which case a sub-optimal solution is obtained. This feature subsequently provides improved operational robustness and autonomy for the system.

The paper is organized as follows: in § 2 the mission planning problem is posed as a receding horizon mixed-integer constrained optimal control problem; § 3 discusses a distributed solution approach for the planning problem and outlines a distributed algorithm for the solution of the mission planning problem; the proposed algorithm is benchmarked in § 4 using an application case study; conclusions are presented in § 7.

2. Optimal control problem

A strategic mission planning system, responsible for coordination and supervision, is typically required when several independent agents cooperate in order to solve one or more mission tasks. The strategic mission planning problem can in generic terms be characterized by:

- a set of tasks and the requirements necessary to complete the tasks;
- a set of objectives, constrains how the tasks are solved (e.g., the most efficient way or the quickest way);
- a set of models that describes the behaviour of the agents and their initial conditions;
- a representation of the environment where the mission takes place.

Algorithms for strategic mission planning must consequently be able to accommodate representations of these characteristics and yield planning strategies that simultaneously satisfies the task requirements, meets the objectives and does not violate the constraints.

The mission planning problem can be posed as a discrete- and finite-time mixed-integer constrained quadratic optimal control problem, where the objectives and tasks are represented by a set of objective functions and a set of binary variables. The state of the binary variables indicate the assignment of a particular task to a particular agent. For instance, if task n is assigned to agent i the binary variable denoted $I_{i,n}$ is set to 1.

The objective functions describe the cost incurred by a particular agent when solving a particular set of tasks. The structure and parameters of the objective functions therefore depends on the values of the integer variables. For instance, the objective function associated with the ith agent, denoted $J_i(\cdot)$, depends on the values of $I_{i,1}$, $I_{i,2}, \ldots$. For a given combination of tasks a particular objective function is uniquely identified. It is assumed that this objective function, when minimized, ensures that the tasks it represents are solved in the desired way. The optimal solution, i.e., the planning strategy, obtained by solving the optimal control problem is expressed in terms of a set of control actions for each of the agents.

The combinatorial nature of the problem means that for missions that involves N agents and M tasks, $N \times M$ binary variables and $N \times M \times M$ objective functions are required in order to fully characterize all possible combinations of agent–task pairings.

It is assumed in the succeeding mathematical derivation that the state dynamics associated with the ith agent can be characterized by a model of the form,

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k). \tag{1}$$

It is furthermore assumed that the mission constraints can be represented as a set of linear in-equality constraints. Under these assumptions the optimal control problem is defined by

$$\min_{\Omega(k)} \sum_{i=1}^{N} J_i\big(I_{i,1}, \ldots, I_{i,M}, X(k), U(k)\big) \tag{2a}$$

s.t.

$$x_i(k+j+1) = A_i x_i(k+j) + B_i u_i(k+j), \quad \forall i = 1 \cdots N, \quad \forall j = 0 \cdots H-1, \tag{2b}$$

$$\sum_{i=1}^{N} \sum_{n=1}^{M} I_{i,n} = M \tag{2c}$$

$$\sum_{i=1}^{N} I_{i,n} = 1, \quad \forall n = 1 \cdots M \tag{2d}$$

$$G(X(k), U(k)) \leq 0, \tag{2e}$$

where H is the time horizon and where,

$$X(k) = \big[x_1(k)^T, \ldots, x_1(k+H)^T, \ldots, x_N(k)^T, \ldots, x_N(k+H)^T\big]^T \tag{3}$$

$$U(k) = \Big[u_1(k)^T, \ldots, u_1(k+H-1)^T, \ldots, u_N(k)^T, \ldots, \times u_N(k+H-1)^T\Big]^T \tag{4}$$

$$\Omega(k) = \big[I_{1,1}, \ldots, I_{1,M}, \ldots, I_{N,1}, \ldots, I_{N,M}, U(k)^T\big]^T. \tag{5}$$

The constraint function, $G(X(k), U(k))$ is assumed to be linear. The constraints (2c) and (2d) ensures that all tasks will eventually be completed and that responsibility for completing each task rests with a single agent. Finally note that the optimization problem is minimized with respect to the control actions, $U(k)$, and the binary variables, $I_{1,1}, \ldots, I_{1,M}, \ldots, \mathrm{I}_{N,1}, \ldots, \mathrm{I}_{N,M}$. The solution obtained by solving the control problem is therefore not only optimal with respect to the control actions but also optimal with respect to how the tasks are assigned to the agents.

The objective functions, $J_i(\cdot)$, quantify the cost of allocating a combinations of tasks to a particular agent, i. It includes the cost associated with each task and the transition cost between successive tasks. The objective functions are normally functions of an agent's states (position and speed) and the control inputs, but may occasionally depend on other agents' states and control inputs. The minimization of the objective function, possibly subjected to constraints, should always lead to a solution that satisfactory solves the tasks assigned to the agent. An example of an objective function is one that correspond to the task of minimum distance navigation to a pre-defined target (single task),

$$\begin{aligned} J_i(X(k), U(k)) = \sum_{j=1}^{H} &\big\| x_T - x_i(k+j) \big\|_{Q_i}^2 \\ &+ \big\| (x_i(k+j) - x_i(k+j-1) \big\|_{Q_i}^2 \\ &+ \big\| u_i(k+j-1) \big\|_{R_i}^2 \end{aligned} \tag{6}$$

where x_T defines the destination point for the agent and where H is the time horizon.

The optimization problem (2) distributes the tasks between the mission agents and selects the control actions for the agents such that the overall cost is minimized within the constraints. If the optimization problem is solved at time k, the solution obtained is optimal with respect to the mission conditions at time k. However, because of changing mission conditions, this solution may no longer be optimal at time $k+1$. In order to incorporate new information more effectively the optimization is repeated at each sample instance. This strategy is also referred to as receding horizon optimal control.

3. Distributed receding horizon optimal control

The design requirements for a strategic mission planning system depends on a wide variety of mission related factors, and can therefore vary between different mission scenarios. However, it is usually desired to design the mission planning system such that it exhibit some degree of autonomy and some level of operational robustness. Autonomy here refers to the ability of the system to adjust to disturbances whilst robustness refers to the ability of the system to function in the event of failures.

Adequate system autonomy can generally be achieved by ensuring that new information is taken into account when it becomes available and that the planning strategy continously adjusted according to this information. A receding horizon planning strategy, such as the one in §2, therefore generally exhibit good autonomy characteristics.

The robustness of the planning system is considered both at the system level (coordination and supervision) as well as the subsystem level (sensor and actuator failures). Satisfactory robustness against subsystem failures is within the context of constrained optimal control achieved through the use of constraints (IAEA 2004). These constraints serves to ensure that the optimal controller selects a control strategy that does not rely on the unavailable sensors and actuators.

Satisfactory robustness against failures at the coordination and supervision level can be achieved by adding redundancy to the system. However, adding redundancy at this level unfortunately often means that multiple instances of the same mission planning problem must be solved simultaneously. Hence the computational demand for the system may dramatically increase as a result. This paper therefore proposes a distributed approach, where the mission planning problem is divided into smaller subproblems. This paper further argues that by partitioning the planning problem and distributing the subproblems between the mission agents such that the interactions between subproblems and the coupling between agents and subproblems are minimized, the impact of failures can be decreased and the overall robustness of the system improved.

The objective function in the optimization problem (2) is defined as the sum of a set of functions that each represents the cost associated with a particular set of tasks being carried out by a particular mission agent. Contextually, each function thus have a strong link to a particular mission agent, and it is therefore reasonable to assume that the associated cost strongly depends on the actions of this agent. Consequently, for optimization problems structured as problem (2) interactions and couplings can be minimized by employing a agent-centric partitioning approach.

In order to partition problem (2) the dependence of the objective function on the decision variables associated with each of the mission agents must first be established. To better highlight this dependence the objective function is expressed on the following equivalent form,

$$\begin{aligned} J\big(I_{*,*}, X(k), U(k)\big) &= \sum_{i=1}^{N} J_i\big(I_{i,*}, X(k), U(k)\big) \\ &= \sum_{i=1}^{N} J_i\big(I_{i,*}, X_i(k), U_i(k), U_{\bar{i}}(k), X_{\bar{i}}(k)\big), \end{aligned} \tag{7}$$

where the variables, $X_i(k)$ and $U_i(k)$, denote the states and decision variables associated with agent i. The notation $I_{i,*}$ denote $\{I_{i,1}, \ldots, I_{i,M}\}$, $I_{*,*}$ denote $\{I_{1,*}, \ldots, I_{N,*}\}$, $U_i(k)$ and $X_i(k)$ denote $[u_i(k)^T, \ldots, u_i(k+H-1)^T]^T$ and $[x_i(k)^T, \ldots, x_i(k+H-1)^T]^T$ respectively and $U_{\bar{i}}(k)$ and $X_{\bar{i}}(k)$ denote $[U_1(k)^T, \ldots, U_{i-1}(k)^T, \quad U_{i+1}(k)^T, \ldots, U_N(k)^T]^T$ and $[X_1(k)^T, \ldots, X_{i-1}(i)^T, \quad X_{i+1}(k)^T, \ldots, X_N(k)^T]^T$ respectively.

The original optimization problem (2) can then be decomposed into N subproblems, each with the following structure,

$$\min_{U_i(k),\, I_{i,*}} J_i\big(I_{i,*}, X_i(k), U_i(k), U_{\bar{i}}(k), X_{\bar{i}}(k)\big) \tag{8a}$$

s.t.

$$x_i(k+j+1) = A_i x_i(k+j) + B_i u_i(k+j), \quad \forall j = 0 \cdots H-1,$$

$$\sum_{i=1}^{N}\sum_{n=1}^{M} I_{i,n} = M \tag{8b}$$

$$\sum_{i=1}^{N} I_{i,n} = 1, \quad \forall n = 1 \cdots M \tag{8c}$$

$$G_i\big(X_i(k), U_i(k), U_{\bar{i}}(k), X_{\bar{i}}(k)\big) \le 0. \tag{8d}$$

The ith subproblems is solved with respect to the decision variables associated with the ith mission agent. Thus, by solving the N subproblems an approximation of the solution to (2) is obtained. Under certain conditions the approximation can be gradually refined by iteratively solving the N subproblems and between each iteration exchange the solutions to the N subproblems (thereby updating $U_{\bar{i}}(k)$ and $X_{\bar{i}}(k)$ between iterations). If sufficient iterations are carried out the solution to (2) is eventually obtained. The conditions under which this method indeed converges to the solution to (2) is governed by the Nash optimality principle (Nash 1951).

Definition 1: A group of decision variables, $U_i^q(k), I_{i,*}^q$ for $\forall i = 1 \ldots N$, given by the solutions to a set of optimization problems of the form (8) is said to converge toward a Nash optimal solution when,

$$\begin{aligned} &J_i\Big(I_{i,*}^q, X_i^q(k), U_i^q(k), U_{\bar{i}}^q(k), X_{\bar{i}}^q(k)\Big) \\ &\quad \le J_i\Big(I_{i,*}^{q-1}, X_i^{q-1}(k), U_i^{q-1}(k), U_{\bar{i}}^{q-1}(k), X_{\bar{i}}^{q-1}(k)\Big) \end{aligned} \tag{9}$$

for $\forall i = 1 \cdots N$, where q denotes the iteration count. The group of decision variables, $U_i^q(k), I_{i,*}^q$ for $\forall i = 1 \cdots N$, is said to have converged to a Nash optimal point when,

$$\begin{aligned} \Big\| &J_i\Big(I_{i,*}^q, X_i^q(k), U_i^q(k), U_{\bar{i}}^q(k), X_{\bar{i}}^q(k)\Big) \\ &- J_i\Big(I_{i,*}^{q-1}, X_i^{q-1}(k), U_i^{q-1}(k), U_{\bar{i}}^{q-1}(k), X_{\bar{i}}^{q-1}(k)\Big)\Big\| \le \gamma_f \end{aligned} \tag{10}$$

$$\big\| U_i^q(k) - U_i^{q-1}(k) \big\| \le \gamma_U \tag{11}$$

$$\big\| I_{i,*}^q(k) - I_{i,*}^{q-1}(k) \big\| \le \gamma_I \tag{12}$$

for $\forall i = 1 \cdots N$ and for sufficiently small values of γ_f, γ_U and γ_I.

The definition above essentially states that if, at each iteration, the solutions associated with each of the subproblems are selected such that condition (9) holds, the solutions generated by iteratively solving the

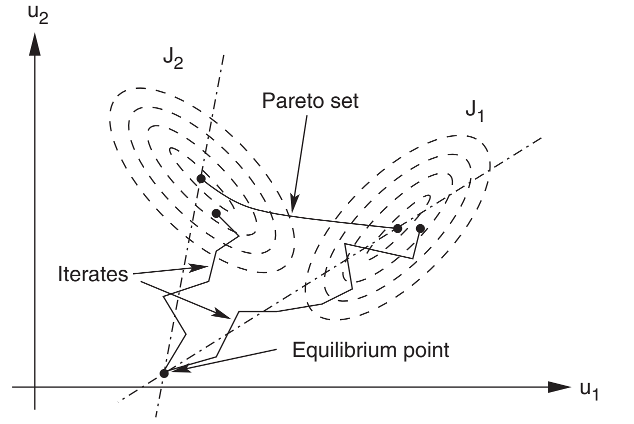

Figure 1. Trajectories in decision space traced by two subsystems.

subproblems converge toward the solution of (2). A geometric representation of the relation between the optimal solution to each subproblem and the iterations toward the Nash equilibrium point is illustrated in figure 1.

The proposed Nash-based distributed receding horizon constrained optimal control algorithm for strategic mission planning purposes can be summarized by the following steps,

Algorithm 1

Step 1: At sampling time instant k, the subproblems are solved by the agents, possibly using information from time instant $k-1$, such that an initial estimate of the decision variables can distributed amongst the agents. The iterative index, q, is set to $q=0$.

Step 2: Using the initial estimate of the decision variables, each agent solves their corresponding subproblem (8) whilst ensuring that the following condition hold,

$$J_i\Big(I^q_{i,*}, X^q_i(k), U^q_i(k), U^q_{\bar{i}}(k), X^q_{\bar{i}}(k)\Big) \\ \leq J_i\Big(I^{q-1}_{i,*}, X^{q-1}_i(k), U^{q-1}_i(k), U^{q-1}_{\bar{i}}(k), X^{q-1}_{\bar{i}}(k)\Big).$$

Step 3: Each agent checks if its terminal iteration condition is satisfied

$$\Big\| J_i\Big(I^q_{i,*}, X^q_i(k), U^q_i(k), U^q_{\bar{i}}(k), X^q_{\bar{i}}(k)\Big) \\ - J_i\Big(I^{q-1}_{i,*}, X^{q-1}_i(k), U^{q-1}_i(k), U^{q-1}_{\bar{i}}(k), X^{q-1}_{\bar{i}}(k)\Big)\Big\| \leq \gamma_f$$

$$\left\| U^q_i(k) - U^{q-1}_i(k)\right\| \leq \gamma_U$$

$$\left\| I^q_{i,*}(k) - I^{q-1}_{i,*}(k)\right\| \leq \gamma_I.$$

If the above conditions are satisfied for all agents the algorithm jumps to Step 5; otherwise it jumps to Step 4.

Step 4: Each agent distributes the solution obtained at iteration q to the other agents. The iteration count is updated to $q=q+1$ and the decision variables are updated according to,

$$U^q(k) = U^{q-1}(k).$$

The algorithm returns to Step 2.

Step 5: Each agent applies the control actions corresponding to time instant k.

Step 6: The agents wait for the next sample instant, $k=k+1$ and thereafter returns to Step 1.

In distributed systems, each subsystem can work independently to achieve its local objective, but cannot accomplish the global objective on its own. For this purpose, the subsystems must communicate, coordinate and negotiate with each other. The proposed algorithm is therefore reliant upon the existence of a communication network, which may be viewed as a limitation of the proposed algorithm. However, for online multi-agent mission planning systems communication is inevitable. Both distributed and centralized solution approaches have a need to communicate the solution to the planning problem to the mission agents. The existence of a communication network is therefore a general requirement for online multi-agent mission planning systems, regardless of the algorithm employed.

It should be noted that the dependence on the reliability and performance of the communication network is high for the proposed algorithm. Unless the communication network can provide high speed communication the iterative nature of the proposed algorithm means that the algorithm will converge slowly toward the Nash optimal solution. The reliability of the communication network can also impact the length of time required to reach the Nash optimal solution.

A discussion on the stability and convergence (feasibility due to condition (9)) properties of the proposed algorithm can be found in Giovanini and Balderud (2006).

4. Simulations and results

In this section the main features and characteristics of the proposed algorithm is demonstrated by applying the algorithm to a typical mission planning problem. The problem under consideration features a cluster of UUVs deployed on a search and exploration mission in an environment where there are non-penetrable obstacles and hidden hazards. The UUVs are expected to advance along a pre-defined search path that crosses this environment whilst searching the area in the vicinity of the search path. Under this mission scenario the objective for the strategic mission planning system is to ensure that the UUVs avoid the obstacles and that the area searched by the UUVs is maximized.

4.1 *Assumptions*

In order to obtain a mission planning problem that more clearly demonstrates the main features and characteristics of the proposed algorithm it is assumed that the UUVs advance along the pre-defined search path in a line formation and that the formation line remains orthogonal to the direction of the search path. It is furthermore assumed that the UUV formation advances at a constant speed of one unit of distance per unit of time. As a result of these assumptions the degrees of freedom of the mission planning problem is significantly reduced. This subsequently reduces the mission planning problem to the relatively simple problem of ensuring that the UUVs, at all times, remain optimally spaced along the formation line.

It is also assumed that each UUV is equipped with an on board navigational control system that responds sufficiently fast to set-point changes such that the position $(x_i(k), y_i(k))$, of the ith UUV can be modelled using a low order dynamic model of the form,

$$x_i(k+1) = x_i(t) + u_{x_i}(k) + v_{x_i}(k) \tag{13}$$

$$y_i(k+1) = y_i(k) + u_{y_i}(k) + v_{y_i}(k), \tag{14}$$

where $u_{x_i}(k)$ and $u_{y_i}(k)$ denote the set-points for the navigational control system, $v_{x_i}(k)$ and $v_{y_i}(k)$ denote zero mean stochastic disturbances and where the positions, $(x_i(k+1), y_i(k+1))$ and $(x_i(k), y_i(k))$, are expressed in terms of 2-D Cartesian coordinates. The depth of the UUVs are assumed constant.

In order for the on board navigational control system to provide sufficient tracking capabilities it is assumed that the set-points must satisfy the following set of constraints,

$$\left|u_{x_i}(k)\right| \leq 1 \tag{15}$$

$$\left|u_{y_i}(k)\right| \leq 1 \tag{16}$$

$$\left|u_{x_i}(k)\right| + \left|u_{y_i}(k)\right| \leq 1.2, \tag{17}$$

where the above constraints can be interpreted to denote conservative estimates of the maximum distance the UUVs can travel each unit of time.

It is also assumed about the UUVs that the UUVs sensor capability for search and exploration purposes reaches 0.5 units of distance in all directions and that the UUVs sensor capability for detecting non-penetrable obstacles reaches 20 units of distance in all directions.

It is finally assumed that the pre-defined search path is a straight line that starts at $(x_s, y_s) = (0, 0)$ and extends to $(x_e, y_e) = (\infty, 0)$, that the starting positions, at time $k = 0$, for the UUVs are $(x_i(0), y_i(0)) = (0, 0)$ and that the formation line remain centred on the search path.

4.2 *Mathematical characterization of the mission planning problem*

In order to maximize the area searched by the UUVs the mission planning system must ensure that the UUVs remain optimally spaced on the formation line. By considering N UUVs, each with a search range of L units of distance, and by planning the movements of these UUVs H time steps into the future, the UUVs optimal positions relative to the centre point of the formation line, $p_1(k+1)\cdots p_1(k+H)\cdots p_N(k+1)\cdots p_N(k+H)$, can be obtained by solving the following optimization problem,

$$\min_{\substack{p_n(k+m),\ \forall n=1\cdots N \\ \forall m=1\cdots H}} \sum_{j=1}^{H}\left[\beta_1\left(\sum_{i=1}^{N-1}(2L-(p_i(k+j)-p_{i+1}(k+j)))^2\right) + \beta_1\left(\sum_{i=1}^{N} p_i(k+j)\right)^2 + \beta_2\sum_{i=1}^{N}(p_i(k+j)-p_i(k+j-1))^2\right] \tag{18a}$$

s.t.

$$\hat{A}[p_1(k+1)\cdots p_1(k+H)\cdots p_N(k+1)\cdots p_N(k+H)]^T \leq \hat{b}, \tag{18b}$$

where the first term of the cost-function above ensures that the UUVs are spaced $2L$ units of distance apart during optimal conditions, where the second term ensures that the UUVs are symmetrically positioned around the centre point of the formation line and where the third term ensures that the resulting search trajectory for each UUV remain smooth. The optimization variables, $p_1(k+1)\cdots p_1(k+H)\cdots p_N(k+1)\cdots p_N(k+H)$, denote the UUVs positions relative to the angle and position of the formation line at time instants $k+1\cdots k+H$. The scalar constants β_1 and β_2 denote tuning parameters. The constraints in the optimization problem serves to ensure that the optimal trajectories for the UUVs are selected such that the UUVs avoids non-penetrable obstacles and such that the UUVs on-board navigational control system can adequately track the trajectories.

Since the optimal trajectories obtained by solving (18) are expressed in terms of coordinates that are relative to the angle and position of the formation line a method for converting between relative coordinates and Cartesian coordinates is needed—particularly for the purpose of defining the optimization constraints. By denoting the position (the centre point) of the formation line $(x_f(k), y_f(k))$, and the angle of the formation line $\Theta(k)+\pi/2$ the Cartesian coordinates, $(x_{c_i}(k), y_{c_i}(k))$, corresponding to a point on the formation line, $p_i(k)$, can be computed using,

$$x_{c_i}(k) = x_f(k) - \sin(\Theta(k))p_i(k) \tag{19}$$

$$y_{c_i}(k) = y_f(k) + \cos(\Theta(k))p_i(k). \tag{20}$$

Since it is in this instance assumed that (i) the formation line is centred on the search path, (ii) the search path is a straight line extending from (0, 0) to (∞, 0) and (iii) the formation advances one unit of distance per unit of time along the search path, the position of the formation line can be modelled using,

$$x_f(k+1) = x_f(k) + 1, \quad x_f(0) = 0 \tag{21}$$

$$y_f(k+1) = y_f(k) + 0, \quad y_f(0) = 0. \tag{22}$$

The angle of the formation line remains constant at $\pi/2$. By solving the above system of difference equations and combining the results with equations (19) and (20) the following relations are obtained,

$$x_{c_i}(k) = k \tag{23}$$

$$y_{c_i}(k) = p_i(k) \tag{24}$$

The next step is to define the constraints that ensures that the optimal trajectories produced by (18) can be tracked by the UUVs on-board navigational control system. To this end, note that the set-points delivered to the UUVs on-board navigational control system can be expressed as

$$u_{x_i}(k) = (k+1) - x_i(k) \tag{25}$$

$$u_{y_i}(k) = p_i(k+1) - y_i(k), \tag{26}$$

where $(x_i(k), y_i(k))$ denote the measured or predicted position of the ith UUV at time instant k. Since the set-points must satisfy the constraints defined by (15)–(17) the optimal positions obtained by solving (18) must satisfy the following set of constraints,

$$|p_i(k) - y_i(k-1)| \le 1 \tag{27}$$

$$|(k) - x_i(k-1)| + |p_i(k) - y_i(k-1)| \le 1.2, \tag{28}$$

where $(x_i(k-1), y_i(k-1))$ denote the measured or predicted position of the ith UUV at time instant $k-1$. By combining equations (25)–(26) and (13)–(14) the predicted position of the ith UUV at time instant $k+1$ can be expressed as,

$$x_i(k+1) = (k+1) \tag{29}$$

$$y_i(k+1) = p_i(k+1). \tag{30}$$

Consequently, over the full length of the prediction horizon, H, the optimal positions for the ith UUV, $p_i(k+1)\cdots p_i(k+H)$ must satisfy the following set of constraints,

$$|p_i(k+1) - y_i(k)| \le 1 \tag{31}$$

$$|(k+1) - \hat{x}_i(k)| + |p_i(k+1) - \hat{y}_i(k)| \le 1.2 \tag{32}$$

$$|p_i(k+1+j) + p_i(k+j)| \le 1 \quad \forall j = 1\cdots H-1 \tag{33}$$

$$|p_i(k+1+j) - p_i(k+j)| \le 1.2 - |(k+1+j) - (k+j)| \quad \forall j = 1\cdots H-1, \tag{34}$$

where $\hat{x}_i(k)$ and $\hat{y}_i(k)$ denote measurements of the position of the ith UUV at time instant k. Note that the constraint defined by (33) is made redundant by the constraint defined by (34). Note also that the right hand side of the constraint defined by (34) simplifies to 0.2.

The remaining constraints, i.e., those that ensure that the optimal trajectories produced by (18) does not cross path with the obstacle boundaries, are defined by employing the following procedure.

(i) At time instant k, prior to solving the optimization problem defined by (18) collect measurements of the positions of the N UUVs.
(ii) Then, for each UUV, carry out the following steps.
   (a) Based on the position of the UUV and the range of the UUVs sensor capability construct a virtual mission environment populated with the obstacle boundaries that the UUV can "see".
   (b) Since the angle and position of the formation line is assumed known at future time instants (recall that the angle and position is given by pre-defined search trajectory) the formation line can be projected onto the virtual mission environment at time instants $(k+1)\cdots(k+H)$.
   (c) By recording the intersections between the projected formation lines and the obstacle boundaries at the different time instants the necessary constraints can be constructed. For instance, if, for the ith UUV, the projected formation line corresponding to time instant

$k+n$ intersects with obstacle boundaries at γ^+ and γ^- the following constraints results:

$$p_i(k+n) \leq \gamma^+$$
$$-p_i(k+n) \leq \gamma^-.$$

Note that the intersection points should be expressed in terms coordinates relative to the position and angle of the formation line.

(d) Repeat the above steps for time instants $(k+1)\cdots(k+H)$ and organize the constraints on the form,

$$A_i[p_i(k+1)\cdots p_i(k+H)]^T \leq b_i.$$

(iii) Construct a matrix A and a vector b such that,

$$A = \begin{bmatrix} A_1 & & 0 \\ & \ddots & \\ 0 & & A_N \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix}$$
$$A[p_1(k+1)\cdots p_1(k+H)\cdots p_N(k+1)\cdots p_N(k+H)]^T \leq b \tag{35}$$

By combining the constraint definitions, (31)–(34) and (35), and the problem (18) the following optimization problem finally results,

$$\min_{p_n(k+m),\, \substack{\forall n=1\cdots N \\ \forall m=1\cdots H}} \sum_{j=1}^{H} \left[\beta_1 \left(\sum_{i=1}^{N-1} (2L - (p_i(k+j) - p_{i+j}(k+1)))^2 \right) + \beta_1 \left(\sum_{i=1}^{N} p_i(k+j) \right)^2 + \beta_2 \sum_{i=1}^{N} (p_i(k+j) - p_i(k+j-1))^2 \right] \tag{36a}$$

s.t.

$$A[p_1(k+1)\cdots p_1(k+H)\cdots p_N(k+1)\cdots p_N(k+H)]^T \leq b \tag{36b}$$

$$|p_i(k+1+j) - p_i(k+j)| \leq 0.2 \quad \forall i = 1\cdots N, \quad \forall j = 1\cdots H-1 \tag{36c}$$

$$|p_i(k+1) - y_i(k)| \leq 1 \quad \forall i = 1\cdots N \tag{36d}$$

$$|(k+1) - x_i(k)| + |p_i(k+1) - y_i(k)| \leq 1.2 \quad \forall i = 1\cdots N. \tag{36e}$$

The optimization problem defined by (36) represents a centralized quadratic optimization problem that can be solved by a quadratic programming approach. In this paper, however, the centralized problem is decomposed into a set of smaller subproblems and solved using the proposed iterative approach. By employing a UUV centred decomposition scheme the following subproblem results:

$$\min_{p_i(k+m), \forall m=1\cdots H} \sum_{j=1}^{H} \left[\beta_1 (2L - (p_i(k+j) - p_{i+j}(k+1)))^2 + \beta_1 \left(\sum_{i=1}^{N} p_i(k+j) \right)^2 + \beta_2 \sum_{i=1}^{N} (p_i(k+j) - p_i(k+j-1))^2 \right] \tag{37a}$$

s.t.

$$A_i[p_i(k+1)\cdots p_i(k+H)]^T \leq b_i \tag{37b}$$

$$|p_i(k+1+j) - p_i(k+j)| \leq 0.2 \quad \forall j = 1\cdots H-1 \tag{37c}$$

$$|p_i(k+1) - y_i(k)| \leq 1 \tag{37d}$$

$$|(k+1) - x_i(k)| + |p_i(k+1) - y_i(k)| \leq 1.2. \tag{37e}$$

By letting the UUVs iteratively solve the problem above a solution that is equivalent to that obtained by solving the centralized problem results.

4.3 *Simulation scenarios*

Three slightly different simulation scenarios, demonstrating slightly different aspects of the proposed algorithm, are considered. These can be briefly summarized by scenarios 1–3.

- *Scenario 1.* The purpose of the first simulation scenario is to demonstrate the main characteristics of the proposed algorithm. This simulation scenario is used in particular to demonstrate the ability of the proposed algorithm to yield trajectories that avoids obstacles and maximizes the area searched by the UUVs and to demonstrate the number of iterations required for the algorithm to reach an optimal solution at each sample instance. The mission environment considered is rectangular shaped and stretches from $(0, -5)$ (lower left corner) to 150, 5 (upper right corner). It includes 3 non-penetrable obstacles that the UUVs must negotiate.
- *Scenario 2.* The purpose of the second simulation scenario is to demonstrate that the search trajectories generated when the proposed algorithm is prevented from performing more than 1 iterations at each sample instance is only marginally different from when the algorithm is allowed to perform unlimited number of iterations. The mission environment considered is equivalent to that considered in Scenario 1.
- *Scenario 3.* The purpose of the third simulation scenario is to demonstrate the ability of the proposed

algorithm to adapt and compensate for failures. At time instant $k = 150$ one of the UUVs suddenly fails and can no longer move. The mission environment considered is near equivalent to that considered in Scenario 1, however, in this case the environment is assumed to stretch from $(0, -5)$ (lower left corner) to 200, 5 (upper right corner).

The following set of simulation parameters have been used (for all simulations),

$$N = 4, \quad H = 20, \quad L = 0.5, \quad \beta_1 = 0.1, \quad \beta_2 = 10.$$

The disturbance sequences, $v_{x_i}(k)$ and $v_{y_i}(k)$, are assumed to be zero mean white noise sequences with a standard deviation of 0.018.

When carrying out the simulations the solution obtained at time k have been used as an initial guess of the solution at time $k + 1$.

4.4 *Simulation results*

The obstacle filled mission environment in which the UUVs operate is depicted in figure 2. The mission environment includes 3 non-penetrable obstacles that the UUVs must negotiate as they advance along the search path. The search path, which is not shown in the figure, extends from $(0, 0)$ to $(\infty, 0)$.

The optimal search trajectories shown in figure 2 corresponds to those computed by the proposed algorithm when the algorithm is allowed to iterate at each sample instance until its solution have converged. At optimal search conditions the distance in the y-direction between the UUVs should be 1. As shown in figure 2, the algorithm achieves these optimal conditions in between the mission obstacles. The non-smooth nature of the trajectories is due to the influence from the disturbance sequences $v_{x_i}(k)$ and $v_{y_i}(k)$.

The number of iterations performed at each sample instance by the proposed algorithm is shown in figure 3. Figure 3 shows that the number iterations performed varies widely and depends on the amount of information the algorithm needs to take into account of. The two peaks shown in figure 3 corresponds to the set of sample instants when the UUVs negotiates the obstacles in the mission environment.

In general, the number of iterations performed also depends on the problem size. Accordingly, if more UUVs are deployed number of iterations increase.

Figure 3 reveals that the proposed algorithm on average performs approximately 8 iterations each sample instant. Based on this observation it is perhaps reasonable to assume that if significantly fewer iterations were to be performed at each sample instant the accuracy of the algorithm would severely suffer. However, since only small adjustments are made to the decision variables between subsequent iterations the accuracy of the algorithm remains relatively unaffected by the number of iterations actually performed. This fact is verified by figure 4 which shows the trajectories generated by the algorithm when limited to 1 iteration per sample instant. By comparing figure 2 and figure 4 it can be concluded that the errors introduced by limiting the number of iterations performed by the algorithm are of such small magnitude that the trajectories shown in figure 4 are near indistinguishable from the trajectories

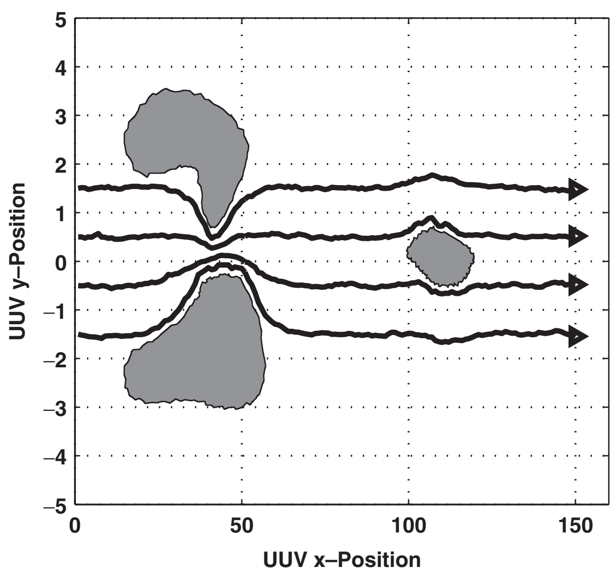

Figure 2. Simulation scenario 1: optimal search trajectories for the four UUVs.

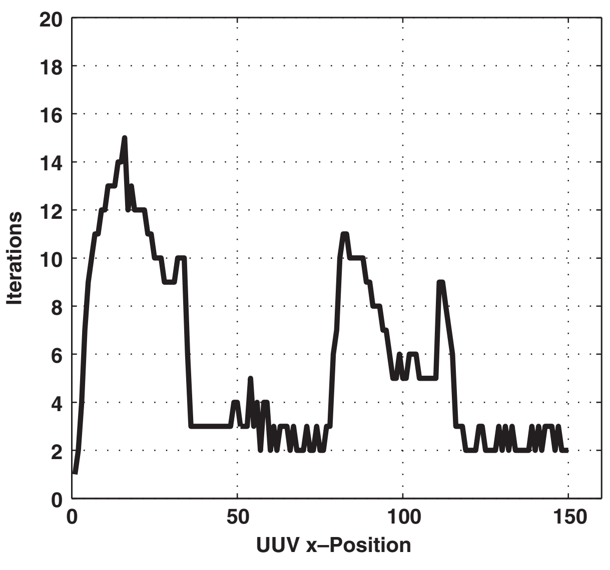

Figure 3. Simulation scenario 1: number of iterations required for the proposed algorithm to compute the optimal solution at each sample instance.

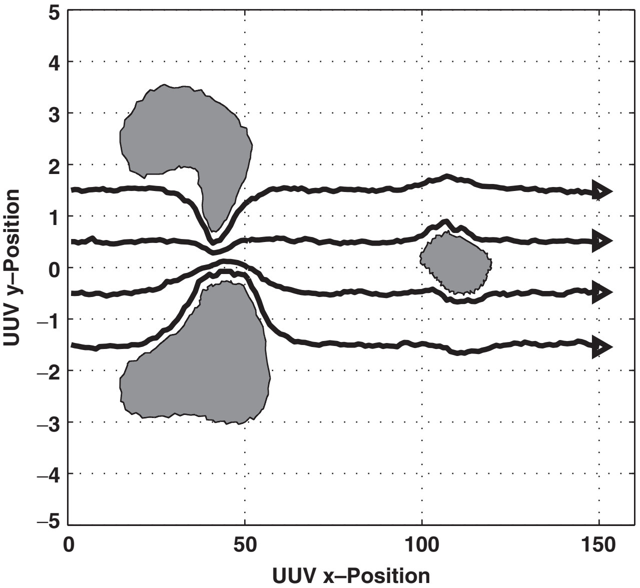

Figure 4. Simulation scenario 2: optimal search trajectories for the four UUVs.

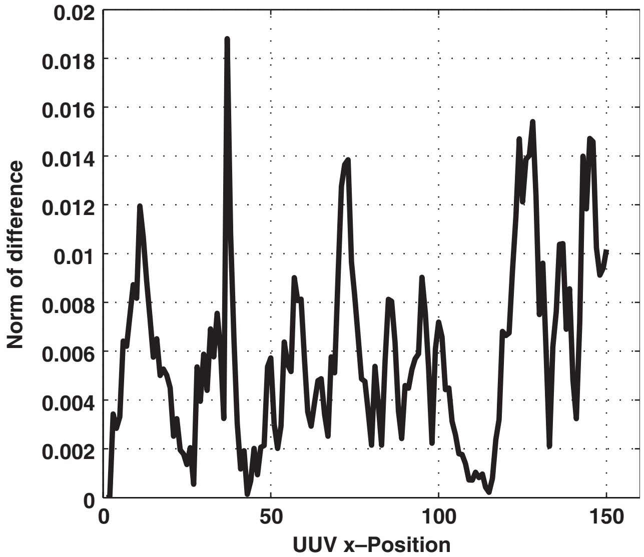

Figure 5. Simulation scenario 2: the norm of the difference between (i) the solution computed by the proposed algorithm when limited to one iteration and (ii) the solution computed by the proposed algorithm when the number of iterations are unlimited.

shown in figure 2. A better illustration of the errors is shown in figure 5, which shows the norm of the differences between the trajectories shown in figure 2 and those shown in figure 4. Figure 5 verifies that the errors indeed remain small. Figure 5 furthermore suggests that the errors are independent and does not depend on problem specific parameters, as is the case for the number of iterations shown in figure 3.

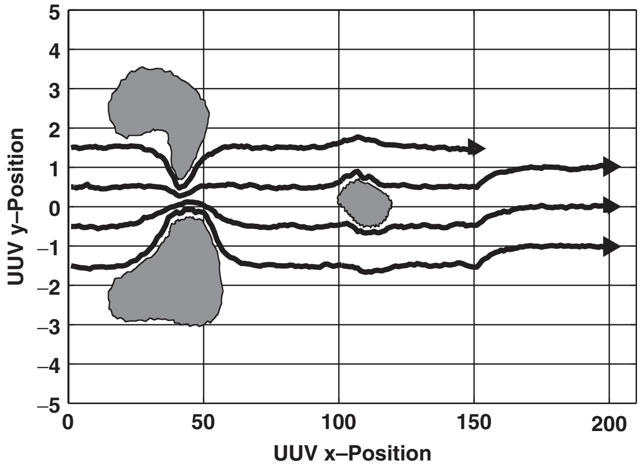

Figure 6. Simulation scenario 3: optimal search trajectories for the four UUVs under failure conditions: at time $k = 150$ one of the UUVs becomes incapacitated.

The results shown in figure 6 corresponds to simulation scenario 3 and illustrates the ability of the proposed algorithm to adapt and maintain optimal search conditions despite failures in individual UUVs.

5. Conclusions

A distributed algorithm for strategic mission planning have been developed. This has been achieved by posing the strategic mission planning problem as a receding horizon mixed-integer constrained quadratic optimal control problem. This problem have subsequently been partitioned into smaller subproblems and a distributed iterative solution approach have been derived.

The proposed algorithm sports many tractable features. For instance, its receding horizon behaviour means that it operates completely autonomously once given its set of tasks. Moreover, the inherent constraint handling features can be exploited to improves the robustness against sensor and actuator failures. The distributed nature of the algorithm finally means that it can be easily reconfigured on-line.

The main characteristics of the algorithm have been demonstrated by applying the algorithm in the context of a typical mission scenario. For the selected mission scenario the algorithm require an average of 8 iterations each sample instance to compute the solution to the planning problem. However, the simulation results also show that the loss of optimality is minimal if the iterations are limited to 1.

Acknowledgements

The authors are grateful for the financial support for this work provided by the Engineering and Physical

Science Research Council (EPSRC) Platform Grant Industrial Non-linear Control and Applications GR/R04683/01.

References

M. Alighanbari, Y. Kuwata and J. How, "Coordination and control of multiple UAVs with timing constraints and loitering", in *American Control Conference*, Denver, Colorado, USA, pp. 5311–5316, 2003.

P. Chandler, M. Pachter, D. Swaroop, J. Fowler, J. Howlet, S. Rasmussen, C. Schumacher and K. Nygard, "Complexity in UAV cooperative control", in *American Control Conference*, Anchorage, Alaska, USA, pp. 1831–1836, 2002a.

P. Chandler, M. Pachter, S. Rasmussen and C. Schumacher, "Multiple task assignment for a UAV team", in *AIAA Guidance, Navigation, and Control Conference*, Anchorage, Alaska, USA, pp. 4587–4591, 2002b.

L. Giovanini and J. Balderud, "Decentralized control via Nash bargaining", in *International Control Conference*, pp. 234–242, 2006.

W. Guo and K. Nygard, "Combinatorial trading mechanism for task allocation", in *13th International Conference on Computer Applications in Industry and Engineering*, San Diego, California, USA, pp. 175–180, 2002.

IAEA, "Managing modernization of nuclear power plant instrumentation and control systems", Technical report, *IAEA-TECHDOC-1389, International Atomic Energy Agency*, Vienna, Austria, 2004, pp. 22–25.

W. Kang, A. Sparks and S. Banda, "Coordinated control of multi-satellite systems", *J. Guid., Contr. Dyn.*, 24, pp. 360–368, 2001.

Y. Liu, M. Simaan and J. Cruz, "An application of dynamic Nash task assignment strategies to multi-team military air operations", *Automatica*, 39, pp. 1469–1478, 2003.

R. Murphy, "An approximate algorithm for a weapon target assignment stochastic program", in *Approximation and Complexity, in Numerical Optimization: Continuous and Discrete Problems*, Dordrecht, Netherlands: Kluwer Academic Publishers, 1999, pp. 39–53.

J. Nash, "Non cooperative games", *Ann. Math.*, 54, pp. 286–295, 1951.

R. Neck and E. Dockner, "Conflict and cooperation in a model of stabilization policies: a differential game approach", *J. Econ. Dyn. Cont.*, 11, pp. 153–158, 1987.

K. Nygard, P. Chandler and M. Pachter, "Dynamic network flow optimization models for air vehicle resource allocation", in *American Control Conference*, Arlington, Virginia, USA, pp. 1853–1858, 2001.

A. Richards, J. Bellingham, M. Tillerson and J. How, "Coordination and control of multiple UAVs", in *AIAA Guidance, Navigation, and Control Conference*, Monterey, California, USA, pp. 68–75, 2002.

S. Rasmussen, P. Chandler, J. Mitchell, C. Schumacher and A. Sparks, "Optimal vs. heuristic assignment of cooperative autonomous unmanned air vehicles", in *AIAA Guidance, Navigation, and Control Conference*, Austin, Texas, USA, pp. 11–14, 2003.

S. Rasmussen, T. Shima, J. Mitchell, A. Sparks and P. Chandler, "State-space search for improved autonomous UAVs assignment algorithm", in *IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, pp. 2911–2916, 2004.

C. Schumacher, P. Chandler and S. Rasmussen, "Task allocation for wide area search munitions via network flow optimization", in *AIAA Guidance, Navigation, and Control Conference*, Montreal, Quebec, Canada, pp. 4147–4152, 2001.

C. Schumacher, P. Chandler and S. Rasmussen, "Task allocation for wide area search munitions", in *American Control Conference*, Anchorage, Alaska, pp. 1917–1922, 2002a.

C. Schumacher, P. Chandler and S. Rasmussen, "Task allocation for wide area search munitions via iterative network flow optimization", in *AIAA Guidance, Navigation and Control Conference*, Monterey, California, USA, pp. 4586–4590, 2002b.

C. Schumacher, P. Chandler, M. Pachter and L. Pachter, "Constrained optimization for UAV task assignment", in *AIAA Guidance, Navigation, and Control Conference*, Providence, Rhode Island, USA, pp. 5352–5357, 2004.

A. Tews and G. Wyeth, "Maps: a system for multi-agent coordination", *Adv. Rob.*, 14, pp. 37–50, 2000.

T. Shima, S. Rasmusen and A. Sparks, "UAV cooperative multiple task assignments using genetic algorithms", in *American Control Conference*, Portland, Oregon, USA, pp. 2989–2994, 2005.

S. Waslander, G. Inalhan and C. Tomlin, "Decentralized optimization via Nash bargaining", in *Conf. on Cooperative Control and Optimization*, Destin, Florida, USA, pp. 2063–2068, 2003.