Proceedings of the 41st IEEE
Conference on Decision and Control
Las Vegas, Nevada USA, December 2002

A Receding Horizon Approach for Solving Some Cooperative Control Problems[1]

Christos G. Cassandras and **Wei Li**
Dept. of Manufacturing Engineering
Boston University
Brookline, MA 02446
cgc@bu.edu, wli@bu.edu

Abstract

We consider a setting where multiple vehicles form a team cooperating to visit multiple target points to collect rewards associated with them. The team objective is to maximize the total reward accumulated over a given time interval. Complicating factors include uncertainties regarding the locations of target points and the effectiveness of collecting rewards, differences among vehicle capabilities, and the fact that rewards are time-varying. We propose a Receding Horizon (RH) control scheme which dynamically determines vehicle trajectories by solving a sequence of optimization problems over a *planning* horizon and executing them over a shorter *action* horizon. The properties of the resulting cooperative controller are tested in a simulated environment and seen to match a reward upper bound with high probability. This exploratory work has also helped identify several issues for further research.

1 Introduction

Cooperative control refers to settings which involve multiple control agents cooperating toward a common objective. The information based on which an agent takes control actions may differ from that of other agents, complicating such problems and placing them in the context of team theory [1],[2]. In addition, when the control agents operate in an uncertain environment, one needs to explicitly model the sources and effects of uncertainty and to provide the capability to react to unexpected events by appropriately adjusting all feasible control actions. Cooperative control has recently emerged as a framework for establishing teams of autonomous vehicles whose task it is to perform a mission with a common goal [3].

In this paper, we consider a setting which involves a team $\mathcal{A}$ of N *vehicles* indexed by $j = 1, \ldots, N$ and a set $\mathcal{T}$ of M *target points* indexed by $i = 1, \ldots, M$ in a 2-dimensional space. Associated with the ith target point is a reward R_i. A *mission* is defined as the process of controlling the movement of the vehicles and ultimately assigning them to target points so as to maximize the total reward collected by visiting points in the set $\mathcal{T}$ within a given mission time T. The problem is complicated by several factors: (i) Target point rewards may be a time-dependent, typically decreasing in time; thus, the order in which target points are visited by vehicles may be critical, (ii) Different vehicles have different capabilities in terms of collecting the reward associated with a target point; thus, assigning specific vehicles to specific points can also be critical, (iii) The exact location of target points may not always be known, (iv) There may be *obstacles* (also known as *threats*) in the 2-dimensional space (referred to as the *mission space*), which constrain the feasible trajectories of vehicles or may cause their elimination when they are encountered, (v) The collection of information about the state of the mission space is an important aspect of the problem, since knowledge of target point and obstacle locations clearly affects the ultimate objective of the mission.

This setting gives rise to a complex stochastic optimal control problem. In principle, one can use dynamic programming as a solution approach, but this is computationally intractable even for relatively simple mission control settings [4],[5]. Moreover, developing stochastic models for the various sources of uncertainty requires parameters which are generally unavailable and hard to estimate. A first step for dealing with such problems is to analyze them at different levels – from detailed control of the vehicles (e.g., [6],[7]) to higher-level path planning and assignment of vehicles to target points before considering the detailed vehicle dynamics. In this paper, we pursue the latter direction in formulating an optimal control problem based on simple vehi-

[1]Work supported in part by the National Science Foundation under Grants ACI-98-73339 and EEC-00-88073, by AFOSR under contracts F49620-01-0056 and F49620-01-1-0348, and by the Air Force Research Laboratory under contract F30602-99-C-0057.

cle dynamics, aiming to determine trajectories which cooperatively guide vehicles to target points so as to maximize the total reward the team can collect as a result of its mission. The problem framework incorporates time-dependent rewards, different vehicle capabilities, and the possibility of unknown target points. We initially assume, however, that there are no obstacles in the mission space; obstacles can be included by defining target points with negative rewards that the vehicles, therefore, try to avoid. We also assume that all information is shared by the vehicles.

In solving this problem, we adopt a Receding Horizon (RH) approach. The RH idea is an integral part of Model Predictive Control (MPC), an approach used to solve optimal control problems for which feedback solutions are extremely hard or impossible to obtain [8],[9],[10]. In such cases, one can derive implicit feedback solutions for nonlinear systems and possibly constraints in the state or control. Applying MPC is still a computationally challenging task for real-time applications and significant effort goes towards efficient ways to obtain solutions at least for some classes of problems (e.g., [11]). In our case, the use of a RH scheme is somewhat different in nature. It is related primarily to the issue of trading off long-term against short-term decisions in the presence of uncertainties by re-solving an optimization problem at appropriately selected time instants or as a result of certain (generally random) events. The main idea is that when a team of vehicles is located relatively far from any known target points, a long planning horizon is adequate to direct them towards their objective and solving a corresponding optimization problem is relatively easy. The solution to this problem is subsequently applied for an appropriately selected period of time, typically much shorter than the actual planning horizon. Thus, the overall mission problem is tackled through a sequence of optimization problems over different RH time intervals. Based on the results reported in this largely exploratory paper, it is somewhat surprising to observe that the relatively simple framework we propose yields solutions which, as will be seen in Section 3.1, are identical or near-identical to an upper bound obtained by an exhaustive search of all possible vehicle-to-target-point assignments and minimal straight line trajectories.

2 Cooperative Control Problem Formulation

We consider a 2-dimensional *mission space*, in which there is a set $\mathcal{T}$ of M *target points* indexed by $i = 1, \dots, M$. The location of target points may vary and new points may randomly emerge. However, let us initially assume that M is fixed and all locations are known with (σ_i, τ_i) being the position of the ith target point. There are also N vehicles indexed by $j = 1, \dots, N$ that define a set $\mathcal{A}$, the "cooperating team", and let $(x_j(t), y_j(t))$ denote the position of the jth vehicle at time t. The vehicle intial positions are given by (x_j^0, y_j^0), $j = 1, \dots, N$. The vehicle dynamics may be generally quite complex, but for the time scale of interest here, we set

$$\dot{x}_j = V_j \cos u_j(t), \quad x_j(0) = x_j^0 \tag{1}$$

$$\dot{y}_j = V_j \sin u_j(t), \quad y_j(0) = y_j^0 \tag{2}$$

where $u_j(t) \in [0, 2\pi]$ is the heading (i.e., the control variable) of vehicle j and V_j is the corresponding velocity, assumed fixed over a mission time interval $[0, T]$. Associated with each target point i is a value or "reward" $R_i > 0$.

The objective of the mission is to collect the maximal total reward $\sum_{i=1}^{M} R_i$ by the end of the mission at time T. Aside from the fact that T may be insufficient to allow vehicles to reach all points, there are two factors preventing the team from meeting this goal: (i) When vehicle j visits point i, there is a probability $p_{ij}(t)$ of actually collecting the reward R_i, and (ii) At the time that a vehicle visits point i, the reward R_i is generally discounted by a factor $\phi_i(t) \in [0, 1]$. If, for instance, a point i has a fixed reward over $[0, \tau]$ and ceases to have any value after τ, then the reward associated with i is given by $R_i \phi_i$ with $\phi_i = \mathbf{1}[t \in [0, \tau]]$; here $\mathbf{1}[\cdot]$ is the usual indicator function. In general, the function $\phi_i(t)$ is defined so that it is monotonically nonincreasing with $\phi_i(0) = R_i$, and $\lim_{t \to \infty} \phi_i(t) = 0$. It is worth noting that points in the mission space with $\phi_i(t)$ which is monotonically nondecreasing are also possible. Also note that we allow $p_{ij}(t)$ above to be time-dependent, since the effectiveness of a particular vehicle may change over time after one or more visits to target points prior to visiting i.

Ultimately, in the course of a mission a particular vehicle must be assigned to a target point. Given some such point i, all other factors being equal a vehicle which is closer to point i at time t is assigned to it. As vehicles move, however, the relative distances between vehicles and targets change and flexibity in target point assignment is maintained so as to have the capability to react to unexpected events. Let $d_{ij}(t) = \left[(x_j(t) - \sigma_i)^2 + (y_j(t) - \tau_i)^2\right]^{1/2}$ be the Euclidean distance between vehicle j and target point i at time t and define the *relative distance* of vehicle j with respect to point i as

$$\delta_{ij}(t) = \frac{d_{ij}(t)}{\min_{k \neq j} d_{ik}(t)} \tag{3}$$

A *normalized relative distance function* $q_{ij}(\delta_{ij})$ is any monotonically nonincreasing function of δ_{ij} such that

$$q_{ij}(0) = 1, \quad q_{ij}(1) = 1/2, \quad \lim_{\delta_{ij} \to \infty} q_{ij}(\delta_{ij}) = 0 \tag{4}$$

One can interpret $q_{ij}(\delta_{ij})$ as the probability that vehicle j is assigned to target point i when its relative distance is $\delta_{ij}(t)$.

Let $\mathbf{u}(t) = [u_1(t) \cdots u_N(t)]$ be the control vector and $\mathbf{x}(t) = [x_1(t) \cdots x_M(t)]$, $\mathbf{y}(t) = [y_1(t) \cdots y_M(t)]$ be state vectors. An optimal control problem aimed at maximizing the total expected reward at the end of a mission may be formulated as

$$\max_{\mathbf{u}(t)} \left[\sum_{i=1}^{M} \sum_{j=1}^{N} R_i \phi_i(T) p_{ij}(T) q_{ij}(T) + \int_0^T L(\mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t)) dt \right] \quad (5)$$

s.t. (1)-(2), where the double sum in (5) is the total final expected team reward and $L(\mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t))$ is a reward function associated with the state and control during the course of a mission. At the level at which we are analyzing the mission, however, we do not consider any such pathwise rewards (e.g., maintaining some desirable vehicle separation or proximity or imposing limits on the control) and seek to solve instead a problem $\mathbf{P}_T$ as follows:

$$\max_{\mathbf{u}} \sum_{i=1}^{M} \sum_{j=1}^{N} R_i \phi_i(T) p_{ij}(T) q_{ij}(T)$$

s.t. (1)-(2), where $\mathbf{u} = [u_1 \cdots u_N]$ is a fixed control and the solution of the problem yields the final position of the vehicles specified by $x_j(T)$ and $y_j(T)$ obtained from the state equations in (1)-(2). Note that this is a nonlinear programming problem which is relatively easy to solve. However, it will frequently be characterized by multiple optima, as will be illustrated in the sequel.

3 The Receding Horizon (RH) Mission Control Scheme

Solving the problem $\mathbf{P}_T$ in (5) over the entire mission time T is of limited practical use, since its solution simply positions the vehicles so that they can maximize the total expected reward, but it does not actually assign vehicles to specific target points. Suppose, however, that the solution found is applied for a time period $h < T$. Then, problem $\mathbf{P}_T$ can be re-solved by replacing $x_j(0) = x_j^0$ and $y_j(0) = y_j^0$ by new initial conditions $x_j(h)$ and $y_j(h)$, $j = 1, \ldots, N$. Since $d_{ij}(h) \neq d_{ij}(0)$, this implies that $\delta_{ij}(h)$ has also changed, consequently $q_{ij}(T)$ is updated, leading to a new solution with more up-to-date state information. The process then repeats with problem $\mathbf{P}_T$ replaced by $\mathbf{P}_{T-h}$.

In general, however, we can replace T by some $H \leq T$, and seek an initial solution of a problem $\mathbf{P}_H$, rather that $\mathbf{P}_T$, over a shorter time horizon than the entire mission time. The choice of H depends on the state of the mission, as discussed later. With this in mind, we define $H(t)$ and $h(t)$ to be the *planning horizon* and *action horizon* respectively when problem $\mathbf{P}_H$ is formulated at time $t \in [0, T)$. Next, we discuss the choice of $H(t)$ and $h(t)$.

Let $d_{\min}(t) = \min_{j \in \mathcal{A}, i \in \mathcal{T}} d_{ij}(t)$ be the minimum distance over all vehicles and target points. The choice of the planning horizon at time t is such that $d_{\min}(t) \leq H(t) \leq T$ where $d_{\min}(t)$ can be expressed in time units since we are assuming a fixed speed for each vehicle. Note that when $d_{\min}(t)$ is small enough to identify some vehicle as being in the vicinity of a target point, then a short-term problem of the form $\mathbf{P}_H$ is solved. We shall assume that there exists a distance r defining the "range" of a target point. Thus, $d_{ij}(t) \leq r$ implies that vehicle j is close enough to point i to render it possible to collect the value $R_i \phi_i(t)$, i.e., the vehicle is effectively "at target point i".

The action horizon $h(t)$ is selected so that

$$h(t) = \alpha_H + \beta_H H(t), \quad \alpha_H \geq 0, \quad 0 \leq \beta_H \leq 1 \quad (6)$$

where β_H is monotonically decreasing in H. In addition, we set $\alpha_H = 0$ and $\beta_H = 1$ for any $H(t) \leq r$. Thus, if a vehicle is within the range r of a target point, the action taken is limited to the small interval $H(t)$; this is equivalent to the assignment of a vehicle to a target point. The choice of $h(t)$ in (6) allows for flexibility to select a fixed action horizon ($\alpha_H =$ constant, $\beta_H = 0$) or one that is a fraction of the planning horizon depending on its magnitude ($\alpha_H = 0$).

Let $\{t_k\}$, $k = 0, 1, \ldots$, denote the time instants when the planning horizon is updated and a new problem $\mathbf{P}_H$ is solved. Based on the above discussion, it is clear that

$$t_{k+1} = \sum_{i=0}^{k} h(t_i)$$

where $t_0 = 0$. In addition, however, this setting allows for a new planning horizon as a result of an event such as the unexpected emergence of a new target point or a change in the value assigned to a target point. In general, any change in the state of the mission space may trigger the solution of a new problem $\mathbf{P}_H$. In implementing trajectory planning, it is common to define *way points*, i.e., points on the trajectory that simplify the task of guiding a vehicle in a desirable manner. The RH scheme effectively generates way points dynamically, as solutions of problems of the form $\mathbf{P}_H$ at times $\{t_k\}$, $k = 0, 1, \ldots$

3.1 Testing Environment

In this section, we describe a testing environment created to explore the behavior and properties of this RH control scheme. This MATLAB-based environment allows for time-dependent target point values (modeled through $\phi_i(t)$), different vehicle capabilities (modeled through $p_{ij}(t)$), and random events such as a new target point or a vehicle failure. A mission problem example and its RH-based solution are shown in Fig. 1. In this example, there are 2 vehicles and 5 target points (shown as circled triangles). Each target point is assigned a reward R_i, $i = 1, \ldots, 5$ (shown connected to the circled triangles) and a "deadline" such that beyond this deadline the point has no more value to the team. The pair (a, b) associated with each target point in the figure is such that a represents the remaining reward and b is the deadline. In this case, the mission time was $T = 25$ time units and upon termination of the mission, all but one target point were visited. The total reward for each vehicle is also shown. On the top right of the figure, the notation $(c : d)$ represents the final mission time (c) and the remaining expected reward for the team (d).

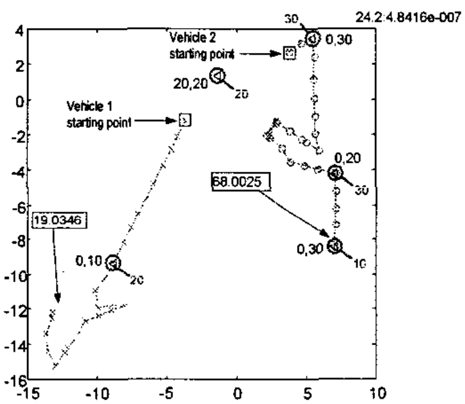

Figure 1: An example of a typical mission executed through the RH control scheme

In all examples shown, we have adopted the following. For simplicity, $p_{ij}(t) = 1$ for all $t \in [0, T]$ and all i, j. The planning horizon is set to $H(t) = d_{\min}(t)$ for all $t \in [0, T]$, and the action horizon is specified through

$$h(t) = \begin{cases} \frac{1}{3}H(t) & \text{if } H(t) > f \\ \frac{1}{2}H(t) & \text{if } H(t) \in [r, f] \\ H(t) & \text{if } H(t) < r \end{cases} \qquad (7)$$

where r is the range of all target points and f is a constant such that $f >> r$ to capture the situation where all vehicles are far from target points and the planning horizon is large relative to r. A discount function $\phi_i(t)$, associated with target point i, is selected as follows:

$$\phi_i(t) = \begin{cases} e^{\frac{\ln(1-\alpha)}{D_i}t} & \text{if } t \leq D_i \\ e^{\frac{\ln(1-\alpha)}{D_i}t} e^{-\beta(t-D_i)} & \text{if } t > D_i \end{cases}$$

where D_i is the deadline for target point i and α, β are positive constants. Note that $\phi_i(D_i) = 1 - \alpha$, so by selecting α arbitrarily small we can represent a situation where the target value experiences negligible reduction over the interval $[0, D_i]$.

A normalized relative distance function $q_{ij}(\delta_{ij})$ is constructed as follows. Let $\tilde{\delta}_{ij} = \frac{d_{ij}}{\sum_{j=1}^{N} d_{ij}}$ and define

$$q_{ij}(\tilde{\delta}_{ij}) = \begin{cases} 1 & \text{if } \tilde{\delta}_{ij} \leq \Delta \\ \frac{1}{2}(\cos\frac{(\tilde{\delta}_{ij}-\Delta)\pi}{1-2\Delta} + 1) & \text{if } \Delta < \tilde{\delta}_{ij} \leq 1 - \Delta \\ 0 & \text{if } \tilde{\delta}_{ij} > 1 - \Delta \end{cases}$$

which satisfies the conditions set in (4). The parameter Δ is selected so that $\Delta \geq r$, i.e., some threshold value greater than the range of any target point.

The most noteworthy, perhaps, feature of the RH scheme is illustrated in Fig. 1, where we observe that vehicles are always ultimately assigned to target points, despite the fact that, as formulated, problem P_H does not explicitly involve any assignment procedure. The RH control approach is based on guiding vehicles towards points in the mission space that maximize the expected rewards conditioned on the current mission state. As deadlines approach and relative distances vary, vehicles are attracted to the vicinity of target points and are eventually assigned to them by virtue of their proximity. The fact that vehicles do not generally follow straight-line trajectories to the various target points is a potential advantage in a setting where significant uncertainty is present: Vehicles tend to wait as long as possible before committing to points by following paths allowing them to alter their destination in the event that the state unexpectedly changes.

3.2 Comparison to a Reward Upper Bound

An upper bound to the mission optimization problem described above is provided by considering a fully deterministic environment and performing an exhaustive search over all possible trajectories from given initial vehicle positions, assuming straight-line paths between all target points. In other words, the trajectory of vehicle j is fully specified by a sequence of target points to be visited by j. Such a search obviously becomes infeasible for large values of N and M. In what follows, we limit ourselves to $N = 2$ vehicles and $M = 6$ target points and obtain results for 15 randomly generated mission spaces (i.e., target point positions, rewards, and deadlines, as well as initial vehicle locations) within a square area defined by $x \in [-10, 10]$ and $y \in [-10, 10]$. These results are compared to the total reward obtained under the proposed RH scheme, as shown in Table 1 (ES stands for 'Exhaustive Search'). In some cases there are multiple target point assignment sequences with the same optimal value; in this case, we report the one that yields the shortest mission

Case	Same Seq. (Y/N)	ES Value (U.B.)	RH Value	ES Time (L.B.)	RH Time
1	N	110	100	22.8648	29.7260
2	Y	140	140	17.2791	17.2791
3	Y	110	110	10.6540	13.0039
4	Y	130	130	16.5697	21.0438
5	N	100	80	19.4185	27.2128
6	Y	140	140	13.7395	18.4356
7	Y	80	80	10.2367	12.6853
8	Y	110	110	19.0695	26.2326
9	Y	130	130	17.7994	23.4521
10	Y	100	100	15.4430	15.4430
11	Y	120	120	15.2512	19.4545
12	Y	140	140	19.4528	22.3243
13	Y	120	120	14.4911	17.7579
14	Y	120	120	17.1673	17.4501
15	Y	90	90	16.0619	24.7739

Table 1: Exhaustive Search Upper Bound vs. RH Control

time. Rather than specifying a mission time, we let the mission end as soon as there is no more positive reward left to collect (because all target point rewards have either been collected or have expired). Clearly, the ES mission time defines a lower bound.

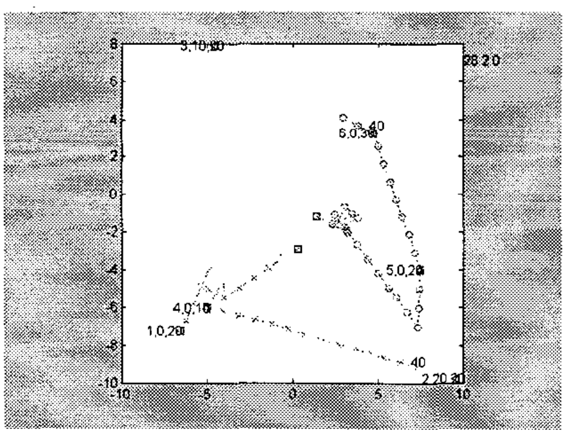

Figure 2: A 2-vehicle, 6-target point example (Case 5 in Table 1)

It is interesting to observe that in all but two cases (Cases 1 and 5) the RH control scheme yields the same total value as the ES optimal solution. In order to gain more insight into the two exceptions, the RH-based vehicle trajectories for case 5 are shown in Fig. 2. Vehicle 1 (marked by $\times$) visits target points 4 and 1, which is identical to the ES optimal solution. Vehicle 2 follows the sequence 5,6, whereas the ES solution turns out to be 6,5,2. Note in Fig. 2 that the vehicle control initially oscillates between directions pointing towards points 6 and 5, before ultimately heading towards an area between points 5 and 2. As previously mentioned, the fact that vehicles do not always follow straight-line trajectories to target points is a potential advantage in the presence of uncertainty, at the expense of mission time (as reflected in Table 1). However, this feature can also lead to instabilities in control actions such as the one seen in Fig. 2. The source of such instability is the presence of multiple local minima in solving problem $\mathbf{P}_H$, which is further discussed in the next section, where we revisit Cases 1 and 5.

4 Control Issues

The exploratory research we have pursued thus far has revealed a number of issues which are characteristic of the mission space setting. In this section we identify and briefly discuss two such issues.

Multiple Local Optima and Instabilities. As already mentioned, it is common for the solution of problem $\mathbf{P}_T$ as formulated in (5) ro exhibit multiple local optima. This is true even for relatively simple examples as we shall illustrate next. Consider a mission space with 2 vehicles initially located at the origin and three target points at $(10,0)$, $(5,5)$, and $(0,10)$. By keeping the rewards fixed but varying the deadline parameters D_1, D_2, D_3 and evaluating the resulting value of the objective function in (5) under multiple values of the controls u_1 and u_2, we can generate response surfaces, a typical one shown in Fig. 3, where we can clearly see that there are multiple local optima and their respective rewards may not be even be close to each other.

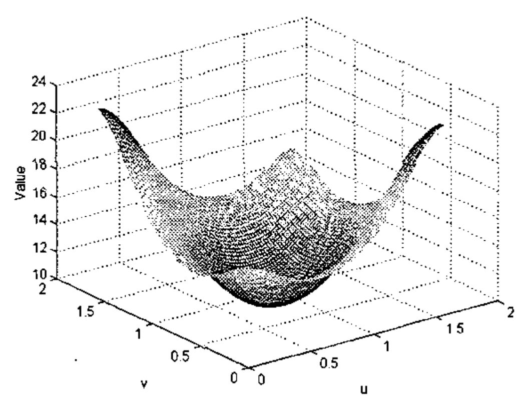

Figure 3: Total Reward response surface for a 2-vehicle, 3-target mission space with $D_1 = D_2 = 10$, $D_3 = 60$

Case	Same Seq. (Y/N)	ES Value (U.B.)	RH Value	ES Time (L.B.)	RH Time
1	N	110	100	22.8648	29.7260
1 mod.	N	110	100	22.8648	27.3045
5	N	100	80	19.4185	27.2128
5 mod.	Y	100	100	19.4185	27.4041

Table 2: Exhaustive Search Upper Bound vs. RH Control after adding a direction change cost

This type of behavior is frequently exhibited in solving problem P_H at different steps of our RH scheme. Returning to Fig. 2, note that vehicle 2 initially heads for target point 5, then switches direction towards point 6, and the process repeats for several iterations. In so doing, it is assumed that the vehicle is free to arbitrarily change directions at no cost. Introducing such a cost results in a tradeoff between changing direction and the incremental expected benefit of such a control action. This tradeoff can be formalized by introducing a *direction change cost function* $\Delta(u, u')$, where u is the current heading of a vehicle and u' is a new heading, determined as the solution of a problem of the form $\mathbf{P}_H$ at some point in time. Let $\mathbf{P}_H(k)$ denote the kth step in our RH scheme when problem $\mathbf{P}_H$ of the form (5) is solved at time t_k. Let $u^*_{j,k-1}$ denote the control for vehicle j after solving $\mathbf{P}_H(k-1)$, and $(x^*_{j,k-1}(t), y^*_{j,k-1}(t))$ be the position of the vehicle corresponding to this control at time $t \geq t_{k-1}$. We can now modify (5) to define the following problem, denoted by $\tilde{\mathbf{P}}_H(k)$:

$$\max_{\mathbf{u}} \sum_{i=1}^{M}\sum_{j=1}^{N} R_i\phi_i(t_k + H_k)p_{ij}(t_k + H_k) \quad (8)$$

$$q_{ij}(t_k + H_k) - \sum_{j=1}^{N}\Delta_j(u^*_{j,k-1}, u_j)$$

$$\text{s.t.}\quad \dot{x}_j = V_j \cos u_j,\ \ x_j(t_k) = x^*_{j,k-1}(t_k),\ \ j = 1,\ldots,N$$
$$\dot{y}_j = V_j \sin u_j,\ \ y_j(t_k) = y^*_{j,k-1}(t_k),\ \ j = 1,\ldots,N$$

Returning to the results shown in Table 1, we have implemented this modified approach by using $\Delta(u, u') = \frac{C|u-u'|}{\pi}$. Table 2 shows the two exception cases from Table 1 before and after the incorporation of the direction change cost, which results in eliminating the oscillatory behavior seen in Fig. 2.

Selecting the Planning and Action Horizon Values. As mentioned in Section 3, the planning horizon $H(t)$ is selected so that $d_{\min}(t) \leq H(t) \leq T$. We believe that setting $H(t) = d_{\min}(t)$ can guarantee the property that a vehicle always eventually gets assigned to a target point, but have not yet established a proof of this conjecture. The choice of acting horizon is easier to deal with, since it is clear that ideally a small value of $h(t)$ is desirable, forcing the controller to re-evaluate control options at a high rate, thus incorporating all new state information. Naturally, this imposes serious computational requirements to be traded off against accuracy.

References

[1] Y. Ho and K. Chu, "Team Decision Theory and Information Structures in Optimal Control Problems - Part 1," *IEEE Trans. Automatic Control*, vol. AC-17, pp. 15–22, 1972.

[2] J. Marschak and R. Radner, *Economic Theory of Teams*. Yale University Press, 1972.

[3] P. Chandler, M. Pachter, and S. Rasmussen, "UAV Cooperative Control," in *Proc. of 2001 American Control Conference*, 2001.

[4] J. Wohletz, D. Castanon, and M. Curry, "Closed-Loop Control for Joint Air Operations," in *Proc. of 2001 American Control Conf.*, 2001.

[5] M. Curry, J. Wohletz, D. Castanon, and C. Cassandras, "Modeling and Control of a Joint Air Operations Environment with Imperfect Information," in *Proc. of SPIE 16th Annual Intl. Symposium*, 2002.

[6] J. Hu and S. Sastry, "Optimal Collision Avoidance and Formation Switching on Riemannian Manifolds," in *Proc. of 40th IEEE Conf. on Decision and Control*, 2001.

[7] R. Prasanth, J. Boskovic, S. Li, and R. Mehra, "Initial Study of Autonomous Trajectory Generation for Unmanned Aerial Vehicles," in *Proc. of 40th IEEE Conf. on Decision and Control*, 2001.

[8] D. Mayne and L. Michalska, "Receding Horizon Control of Nonlinear Systems," *IEEE Trans. Automatic Control*, vol. AC-35, no. 7, pp. 814–824, 1990.

[9] J. Cruz, M. Simaan, A. Gacic, and Y. Liu, "Moving Horizon Game Theoretic Approaches for Control Strategies in a Military Operation," in *Proc. of 40th IEEE Conf. on Decision and Control*, 2001.

[10] L. Singh and J. Fuller, "Trajectory Generation for a UAV in Urban Terrain Using Nonlinear MPC," in *Proc. of 2001 American Control Conf.*, 2001.

[11] A. Bemporad, F. Borrelli, and M. Morari, "The Explicit Solution of Constrained LP-Based Receding Horizon Control," in *Proc. of 39th IEEE Conf. on Decision and Control*, 2000.