OPTIMAL CONTROL THROUGH BIOLOGICALLY-INSPIRED PURSUIT [1]

Cheng Shao * and D. Hristu-Varsakelis **

** Department of Mechanical Engineering and Institute for System Research, University of Maryland, College Park, MD 20742, USA*
*** Department of Applied Informatics, University of Macedonia, Thessaloniki 54006, Greece*

Abstract: Inspired by the process by which ants gradually optimize their foraging trails, this paper investigates the cooperative solution of a class of free-final time, partially-constrained final state optimal control problems by a group of dynamic systems. A cooperative, pursuit-based algorithm is proposed for finding optimal solutions by iteratively optimizing an initial feasible control. The proposed algorithm requires only short-range, limited interactions between group members, and avoids the need for a "global map" of the environment on which the group evolves. The performance of the algorithm is illustrated in a series of numerical experiments.

Keywords: Co-operative control, Optimization, Agents, Group work, Trajectories

1. INTRODUCTION

In recent years, problems in cooperative control are increasingly capturing the attention of researchers, fueled by the development of decentralized control systems with cost and performance advantages. The rising interest in deploying cooperative systems also stems from the fact that such systems have the potential to perform tasks that are not feasible for individuals. Examples include remote exploration and information gathering by swarms of small autonomous robots (Brooks and Flynn, 1989), and satellite arrays, to name a few. Members of such "engineered collectives" usually have limited sensing, communication and computing capabilities. This suggests that each member can only perform relatively simple tasks. On the other hand, the limitations of individuals can often be overcome by cooperation, if one can identify an effective way to organize the group into "more than the sum of its parts". This is often demonstrated – rather impressively – by biological collectives. For example, a school of fish can coordinate their movement in a tight formation; worker honey bees share information by "dancing" and distribute themselves among nectar sources in accordance with the profitability of each source; ants are known to utilize pheromone secretions for recruiting nest-mates and for optimizing their foraging trails (Camazine, 2001). Observations of such activities in nature have already seeded a variety of research, from modeling of animal group behaviors (Camazine, 2001; Bruckstein, 1993; Jadbabaie *et al.*, 2003), to distributed collective covering and searching (Wagner *et al.*, 1999), cooperative estimation (Roumeliotis and Bekey, 2002) and biologically-motivated optimization (Dorigo *et al.*, 1996).

[1] This work was supported by NSF Grant No. EIA0088081 and by ARO ODDR&E MURI01 Grant No. DAAD19-01-1-0465, (Center for Communicating Networked Control Systems, through Boston University). Corresponding author: D. Hristu-Varsakelis, Tel: +30-2310-891-721, Fax: +30-2130-891-721, e-mail: dcv@uom.gr

One of the earliest optimization methods inspired by trail formation in ants was presented in (Bruckstein, 1993), where it was shown that ants that "pursued" one another on $\mathbb{R}^2$ (each pointing its velocity vector towards a predecessor) had the effect of producing progressively "straighter" trails. That idea was later extended to path optimization problems involving kinematic vehicles in non-Euclidean environments (Hristu-Varsakelis, 2000). The last two works were restricted exclusively to the "discovery" of geodesics, meaning that the autonomous system-members of the group had very simple dynamics with no drift terms. This paper shows that the earlier work can be generalized to a much broader class of optimal control problems, and collectives whose members have non-trivial dynamics. The proposed algorithm is based on "local pursuit" (to use the term coined in (Bruckstein, 1993)) and guides members of a group towards a solution of a free final time, partially-constrained final state optimal control problem, this time in a broader and more intricate setting. Under "continuous local pursuit", as this iterative, decentralized algorithm is termed, agents do not need a global map of their environment or even an agreed-upon common coordinate system. The algorithm is most useful in trajectory optimization problems which are easier to solve when boundary conditions are "close" to one another (because of, for example, the members' computational or sensing limitations), with the term "close" taken to include not only geographical separation but also distance on the manifold on which copies of a dynamical system evolve.

The remainder of this paper is organized as follows: Section 2 describes the optimal problems to be addressed and proposes an iterative algorithm that is appropriate for a group of cooperative dynamic systems. Section 3 discusses the main results concerning the performance of the proposed algorithm. A pair of illustrative numerical examples are given in Section 4.

2. A BIO-INSPIRED ALGORITHM FOR OPTIMAL CONTROL

This paper explores the solution of optimal control problems using a group of cooperating "agents". The term "agent" refers to a member of a group of dynamical systems, each taken to be a "copy" of:

$$\dot{x}_k = f(x_k, u_k),\ x_k(t) \in \mathbb{R}^n, u_k(t) \in \Omega \subset \mathbb{R}^m \quad (1)$$

for $k = 0, 1, 2 \ldots$. Physically, each copy of (1) could stand for a robot, UAV or other autonomous system. The problem of interest is as follows:

Problem 1. Find a trajectory $x^*(t)$, a final time $\Gamma^* > 0$ and a final state $x^*(\Gamma^*)$ that minimize

$$J(x, \dot{x}, t_0) = \int_{t_0}^{t_0+\Gamma} g(x, \dot{x})dt + G(x(t_0 + \Gamma)) \quad (2)$$

subject to the constraints $x(t_0) = x_0$ and $Q(x(t_0+\Gamma)) = 0$.

Here it is assumed that $g(x(t), \dot{x}(t)) \geq 0$, $G(x(t_0+\Gamma)) \geq 0$ and that $Q(\cdot)$ is an algebraic function of the state.

Definition 1. Given the final state constraint $Q(x) = 0$, the *constraint set* of x is

$$S_Q = \{x | Q(x) = 0\}.$$

Now the function $G(x)$ in (2) can be replaced by

$$G(x) = \begin{cases} F(x) & \text{if } x \in S_Q \\ 0 & \text{if } x \notin S_Q \end{cases}$$

with $F(x) \geq 0, \forall x \in S_Q$. Problem 1 involves optimal control with free final time, partially-constrained final state. Fixed final state problems, where S_Q is a single state (Shao and Hristu-Varsakelis, 2004), are special cases of what are considered here.

For any pair of fixed states $a, b \in \mathbb{D} \subset \mathbb{R}^n$, suppose the optimal trajectory from a to b with free final time (minimizing J with respect to x, Γ only) is denoted by $x^*(t)$ and that the corresponding optimal final time is $\Gamma^*(a, b)$. The cost of following x^* is denoted as:

$$\begin{aligned} \eta(a, b, t_0) &\triangleq \int_{t_0}^{t_0+\Gamma^*} g(x^*, \dot{x^*})dt + G(x^*(t_0 + \Gamma^*)) \\ &= \min_{x,\Gamma} J(x, \dot{x}, t_0) \end{aligned} \quad (3)$$

subject to $x(t_0) = a$, $x(t_0+\Gamma) = b$. Now, let $x^*(t)$ be the optimal trajectory from an initial state a to the constraint set S_Q, and let the corresponding optimal final time from a to S_Q be $\Gamma^*_Q(a, S_Q)$. The cost of following x^* is denoted by

$$\begin{aligned} \eta_Q(a, t_0) &\triangleq \int_{t_0}^{t_0+\Gamma^*_Q} g(x^*, \dot{x^*})dt + G(x^*(t_0 + \Gamma^*_Q)) \\ &= \min_{x,\Gamma_Q} J(x, \dot{x}, t_0) \end{aligned} \quad (4)$$

subject to $x(t_0) = a, Q(x(t_0 + \Gamma_Q)) = 0$. The cost of following a generic trajectory $x(t)$ of (1) during $[t_0, t_0 + \sigma)$ is denoted by:

$$C(x, t_0, \sigma) \triangleq \int_{t_0}^{t_0+\sigma} g(x, \dot{x})dt + G(x(t_0 + \sigma)) \quad (5)$$

The following facts can be derived easily from the properties of optimal trajectories and are helpful in the sequel:

Fact 1. Let η, η_Q, C as defined in (3)-(5), and $x_k(t)$ be a generic trajectory of (1). Then, the following hold:

(1) $\eta(a,b,t_0) \le C(x_k,t_0,\Gamma)$ *for any* $x_k(\cdot)$ *with* $x_k(t_0)=a,\ x_k(t_0+\Gamma)=b$.
(2) $\eta(a,c,t_0) \le \eta(a,b,t_0)+\eta(b,c,t_0+\sigma)$, *with* $\sigma = \Gamma^*(a,b)$.
(3) $\eta_Q(a,t_0) \le \eta(a,b,t_0)$ *for any* $b \in S_Q$.

Assume that there is available an initial feasible control/trajectory pair $(u_{feas}(t), x_{feas}(t))$ (but sub-optimal) for (1), obtained through a combination of a-priori knowledge about the problem and/or random exploration. Inspired by the idea in (Bruckstein, 1993), the agents are scheduled to leave the initial state x_0 sequentially and pursue one another in a way which will be made precise shortly. The sequence is initiated with the first agent following x_{feas} to the set S_Q. Each agent will attempt to intercept its predecessor – along optimal trajectories defined by (3) – if the predecessor has not reached the final state. If the predecessor has already reached the constraint set S_Q, then the pursuer evolves along the optimal trajectory defined by (4). The precise rules that govern the movement of each agent are:

Algorithm 1. (Continuous Local Pursuit): Identify the starting state x_0 on $\mathbb{D}$ and the constraint set S_Q. Let $x_0(t)$ $(t \in [0,T_0])$ be an initial trajectory satisfying (1) with $x_0(0)=x_0,\ Q(x_0(T_0))=0$. Choose the pursuit interval Δ such that $0 < \Delta \le T_0$.

(1) For $k=1,2,3\ldots$, let $t_k = k\Delta$ be the starting time of k^{th} agent. Let $u_k(t)=0, x_k(t)=x_0$ for $0 \le t \le t_k$.
(2) For all $t \ge t_k$, calculate $u_t^*(\tau)$ for all $t \in [t_k, t_k+T_k]$ such that $f(\hat{x}_k(\tau), u_t^*(\tau)) = \dot{\hat{x}}_t(\tau)$, and $\hat{x}_t(\tau)$ achieves
$$\begin{cases} \eta(x_k(t), x_{k-1}(t), t), & \text{if } x_{k-1}(t) \notin S_Q \\ \quad (\tau \in [t, t+\Gamma^*(x_k(t), x_{k-1}(t))]) & \\ \eta_Q(x_k(t), t), & \text{if } x_{k-1}(t) \in S_Q \\ \quad (\tau \in [t, t+\Gamma_Q^*(x_k(t), S_Q)]) & \end{cases}$$
(3) Apply $u_k(t) = u_t^*(0)$ to the k^{th} agent.
(4) Repeat from step 2, until the k^{th} agent reaches S_Q.

Under continuous local pursuit (CLP), each agent continuously updates its own trajectory at every $t \in [t_k, t_k+T_k]$. It is possible to alter the algorithm so that each agent only performs a finite number of trajectory optimizations as it evolves from x_0 to S_Q. The resulting "sampled local pursuit" algorithm is detailed in (Shao and Hristu-Varsakelis, 2004).

The $(k-1)^{th}$ agent is designated as the "leader" and the k^{th} agent as the "follower" during pursuit. As Step 2 of the algorithm indicates, there are two types of followers' movements, "free running" and "catching up", depending on whether the leader has reached the final constraint S_Q or not. The former type lets agents "learn" from their leaders, while the "free running" stage enables them to find the optimal final state within S_Q. Both stages are essential for the cooperative solution to an optimization problem with partially-constrained final state.

3. MAIN RESULTS

The CLP algorithm defines an ordered sequence of trajectories $\{x_k(t)\}$. This section will first investigate the convergence of the trajectories' cost, and then will show that the trajectories themselves converge to a local optimum. It will be convenient to distinguish between the *planned trajectories*, denoted by $\hat{x}(t)$, that a follower computes at every point in time, and the *realized trajectories*, denoted by $x(t)$, which the follower actually evolves along.

Lemma 1. Consider a leader-follower pair and a pursuit interval Δ *defined in continuous local pursuit. Let the leader's trajectory be* $x_{k-1}(t)$ $(t \in [t_{k-1}, t_{k-1}+T_{k-1}])$ *and* $\lambda \in [0, T_{k-1})$. *Suppose the follower updates its trajectory only once during* $[t_k, t_k+T_k]$ *as described next:*

- *If* $\lambda < T_{k-1}-\Delta$, *the follower moves along the optimal trajectory joining* $x_k(t_k+\lambda)$ *and* $x_{k-1}(t_k+\lambda)$ *(in the sense of (3)) with optimal final time* $\Gamma = \Gamma^*(x_k(t_k+\lambda), x_{k-1}(t_k+\lambda))$. *During other times, the follower replicates the leader's trajectory, i.e.*
$$\begin{cases} x_k(t) = x_{k-1}(t-\Delta) & t \in [t_k, t_k+\lambda] \\ x_k(t) = x_{k-1}(t-\Gamma) & t \in [t_k+\lambda+\Gamma, t_k+T_k] \end{cases}$$
- *If* $\lambda \ge T_{k-1}-\Delta$, *the follower evolves along the optimal trajectory from* $x_k(t_k+\lambda)$ *to the constraint set* S_Q *(in the sense of (4)). Similarly, during other times*
$$x_k(t) = x_{k-1}(t-\Delta) \quad t \in [t_k, t_k+\lambda]$$

Then the cost along the follower's trajectory will be no greater than the leader's.

PROOF. First, choose $\lambda < T_{k-1}-\Delta$. Starting at time $t_k+\lambda$ and during $t \in [t_k+\lambda, t_k+\lambda+\Gamma]$), the follower moves on the locally optimal trajectory $x_k(t)$. The cost along x_k is

$$\begin{aligned} & C(x_k, t_k, T_k) \\ &= C(x_k, t_k, \lambda) + C(x_k, t_k+\lambda+\Gamma, T_k-\lambda-\Gamma) \\ &\quad +\eta(x_k(t_k+\lambda), x_{k-1}(t_k+\lambda), t_k+\lambda) \\ &\le C(x_{k-1}, t_{k-1}, \lambda) + C(x_{k-1}, t_{k-1}+\lambda, \Delta) \\ &\quad +C(x_{k-1}, t_{k-1}+\lambda+\Delta, T_{k-1}-\lambda-\Delta) \\ &= C(x_{k-1}, t_{k-1}, T_{k-1}) \end{aligned} \tag{6}$$

where $\Gamma = \Gamma^*(x_k(t_k+\lambda), x_{k-1}(t_k+\lambda))$. If $\lambda \geq T_{k-1} - \Delta$, the cost along x_k is

$$
\begin{aligned}
& C(x_k, t_k, T_k) \\
&= C(x_k, t_k, \lambda) + \eta_Q(x_k(t_k+\lambda), t_k+\lambda) \\
&\leq C(x_{k-1}, t_{k-1}, \lambda) + C(x_{k-1}, t_{k-1}+\lambda, T_{k-1}-\lambda) \\
&= C(x_{k-1}, t_{k-1}, T_{k-1})
\end{aligned}
$$

Therefore the cost along the follower's trajectory is no greater than the leader's. □

Now the cost of the iterative trajectories can be shown to converge under CLP:

Lemma 2. (Convergence of Cost) If the agents of (1) evolve under CLP, the cost of the iterated trajectories converges.

PROOF. Suppose the cost along the leader's trajectory $x_{k-1}(t)$ $(t \in [t_{k-1}, t_{k-1}+T_{k-1}])$ is C_{k-1}. Define a trajectory sequence $x_k^i(t)$ $(t \in [t_k, t_k+T_k^i]), i = 0, 1, 2\ldots$ whose corresponding costs and final times are C_k^i and T_k^i, as follows: let $x_k^0(t) = x_{k-1}(t)$ (i.e. the trajectory of a "leader") and x_k^i $(i > 0)$ is the trajectory of an agent that pursues x_k^{i-1} by performing only a *single trajectory update*, as described in Lemma 1 with $\lambda = (i-1)\delta, \delta > 0$ (see Fig. 1).

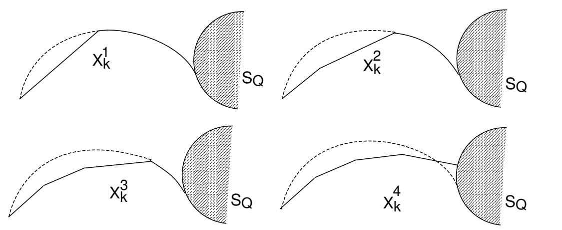

Fig. 1. Illustration of the trajectory sequence $x_k^i(t)$. Each trajectory is obtained by a single update upon its predecessor.

From Lemma 1, the cost of each follower's trajectory will be no greater than the leader's, thus $C_k^i \leq C_k^{i-1} \Rightarrow C_k^\infty \leq C_k^0 = C_{k-1}$. Let $\delta = T_{k-1}/i$, then $\delta \to 0$ as $i \to \infty$. At the limit, the trajectory $x_k^\infty(t)$ is exactly what would be obtained by an agent that pursues another evolving along x_{k-1}, using CLP. Hence the follower's cost is $C_k = C_k^\infty \leq C_{k-1}$. Since the sequence $\{C_k\}$ is non-increasing and bounded below (there exists a minimum for (2)), it must converge to a limit. □

To proceed to the main theorem, one must require that the optimal cost of (2) changes "little" for small changes to the endpoints of a trajectory:

Condition 1. Assume for a generic trajectory $x_1(t)$ there exists an $\varepsilon > 0$ such that for all $a, b_1, b_2 \in \mathbb{D}$ and all $\Delta > 0$, there exists a trajectory $x_2(t)$ such that the cost $C(x_1, 0, T)$ $(x_1(0) = a, x_1(T) = b_1)$ from a to b_1 and cost $C(x_2, 0, T)$ $(x_2(0) = a, x_2(T) = b_2)$ from a to b_2 satisfy

$$
\begin{aligned}
& \|b_1 - b_2\|_\infty < \varepsilon \\
\Rightarrow & \|C(x_1, 0, T) - C(x_2, 0, T)\|_\infty < \mathcal{L}\Delta
\end{aligned}
$$

for some constant $\mathcal{L}$ independent of Δ.

Then the next lemma holds:

Lemma 3. Let $x^*(t)$ be a trajectory of (1) such that: i) $x^*(t)$ $(t \in [0, t_1+\Delta_1])$ is optimal (in the sense of (3)) from $x^*(0)$ to $x^*(t_1+\Delta_1)$, and ii) $x^*(t)$ $(t \in [t_1, T^*])$ is optimal (in the sense of (4)) from $x^*(t_1)$ to the constraint set S_Q. Assume Condition 1 is satisfied and $0 < t_1 < t_1+\Delta_1 < T^*$. Then the trajectory $x^*(t)$ $(t \in [0, T^*])$ is a local minimum of (4) from $x^*(0)$ to S_Q.

PROOF. Pick $0 < \Delta \leq \Delta_1$. From principle of optimality, $x^*(t)$ $(t \in [0, t_1+\Delta])$ and $x^*(t)$ $(t \in [t_1, T^*])$ are each locally optimal with respect to their corresponding end points. Suppose $\|x^*(t_1+\Delta) - s\|_\infty \geq \varepsilon_1$ for any $s \in S_Q$ and that $x^*(t)$ $(t \in [0, T^*])$ is not a local minimum. There must exist a $\epsilon < \min(\varepsilon, \varepsilon_1/2)$ (ε is defined in Condition 1) and another optimum $x(t) \in \mathbb{D} \times [0, T]$ satisfying that $\|x(t) - x^*(t)\|_\infty < \epsilon$ and $C(x(t), 0, T) < C(x^*(t), 0, T^*)$.

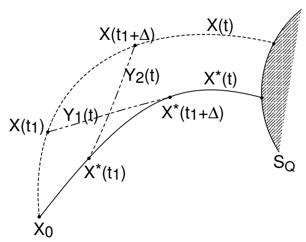

Fig. 2. Illustrating the proof of Lemma 3: overlapping optimal trajectories form a locally optimal trajectory.

Notice that $\|x(t_1+\Delta) - s\|_\infty \geq \epsilon$ for any $s \in S_Q$. Construct two trajectories $y_1(t), y_2(t)$ $(t \in [t_1, t_1+\Delta])$ that connect $x(t)$ and $x^*(t)$ (see Fig. 2) and satisfy Condition 1 (with x^* or x playing the role of x_1, and y_1, y_2 standing in for x_2). In particular, let y_1, y_2 be such that $x^*(t_1) = y_2(t_1), x^*(t_1+\Delta) = y_1(t_1+\Delta), x(t_1) = y_1(t_1), x(t_1+\Delta) = y_2(t_1+\Delta)$. Now ,Condition 1 implies that

$$
\begin{aligned}
C(y_1(t), t_1, \Delta) &< C(x(t), t_1, \Delta) + \mathcal{L}\Delta \\
C(y_2(t), t_1, \Delta) &< C(x^*(t), t_1, \Delta) + \mathcal{L}\Delta \qquad (7)
\end{aligned}
$$

Because $x^*(t)$ $(t \in [0, t_1+\Delta])$ and $x^*(t)$ $(t \in [t_1, T^*])$ are each locally optimal, the following holds:

$$C(x^*(t), 0, t_1) + C(x^*(t), t_1, \Delta) \quad (8)$$
$$< C(x(t), 0, t_1) + C(y_1(t), t_1, \Delta), \text{ and}$$

$$C(x^*(t), t_1, \Delta) + C(x^*(t), t_1 + \Delta, T^* - t_1 - \Delta) <$$
$$C(x(t), t_1 + \Delta, T - t_1 - \Delta) + C(y_2(t), t_1, \Delta) \quad (9)$$

Combining (7) with (8,9) leads to

$$C(x^*(t), 0, T) < C(x(t), 0, T) + 2\mathcal{L}\Delta \quad (10)$$

The cost $C(x(t), 0, T)$ is apparently less than $C(x^*(t), 0, T)$, but if Δ is chosen so that

$$0 < \Delta < \frac{C(x^*(t), 0, T) - C(x(t), 0, T)}{2\mathcal{L}}$$

then (10) cannot hold. This is a contradiction, because Δ could be chosen arbitrarily small. It follows that $x^*(t)$ $(t \in [0, T^*])$ must be a local minimum. □

The following lemmas also hold. The proofs (both by contradiction) can be found in (Shao and Hristu-Varsakelis, 2004) and will not be given here.

Lemma 4. If for all times during CLP, the locally optimal trajectory from follower to leader is unique, then the limiting trajectory $x_\infty(t)$ exists and is unique.

Lemma 5. Along the limiting trajectory produced under CLP, the planned trajectories $\hat{x}_k(t)$ and realized trajectories $x_k(t)$ overlap, i.e. $\hat{x}_k(t) = x_k(t)$. Furthermore, if the locally optimal trajectories obtained at every updating time are smooth, then the limiting trajectory is also smooth.

The next theorem is an immediate consequence of Lemmas $1-5$:

Theorem 1. Suppose a group of agents evolve under CLP and that at every updating time t, the locally optimal trajectories from follower to leader are unique. Then, the limiting trajectory obtained is unique and locally optimal. It is also smooth if the locally optimal trajectories calculated at every updating time are smooth.

PROOF. From Lemma 4, the limiting trajectory is unique. It follows that $x_{k-1}(t-\Delta) = x_k(t)$ if $x_{k-1}(t) = x_\infty(t - t_{k-1})$. Choose δ_1, δ_2 such that $0 < \delta_1 < \delta_2 < \Gamma$ for all optimal final times Γ of planned trajectories $\hat{x}_k$ generated during CLP. The limiting trajectory x_∞ is piecewise smooth and locally optimal for $t \in [t_k + i\delta_1, t_k + i\delta_1 + \delta_2], i = 0, 1, 2\ldots$ because it coincides with the planned trajectories $\hat{x}(t)$. From Lemma 3 – in this case the S_Q is a single point – it can be concluded that $x_k(t)$ $(t \in [t_k, t_k + \delta_1 + \delta_2])$ is optimal because it is the composition of two overlapping locally optimal trajectories, $x_k(t)$ $(t \in [t_k, t_k + \delta_2])$ and $x_k(t)$ $(t \in [t_k + \delta_1, t_k + \delta_1 + \delta_2])$. Successive applications of this argument $(i = 2, 3, \ldots)$ lead to the result that $x_\infty(t)$ is locally optimal. The smoothness is also proved "piece by piece". □

CLP is a cooperative, decentralized algorithm for learning optimal controls/trajectories, starting from a feasible solution. Each agent is only required to calculate optimal trajectories from its own state to that of its nearby leader. Because agents are separated by Δ time units as they leave x_0, each agent relies only on local information in order to follow its predecessor, and requires no knowledge of the global geometry. There is no need for agents to exchange or "fuse" local maps that they obtain individually. Agents do not need to communicate their choice of coordinate systems as they evolve, nor do they need to know the coordinates of x_f. While it is possible that a group of agents could disperse and construct a global map from local information, such an approach might require significantly more computation and communication than CLP. The latter solves the optimal control problem in many "short pieces", which makes it appropriate for systems with short-range sensors (for example, in the case of a swarm of robots exploring unknown terrain), and optimal control problems which are easier to solve over "short" distances.

4. EXAMPLES

Consider the minimum-time control of the second-order system $\ddot{x} = u,\ |u| \le 1$, where the cost to be minimized is $J(x, \dot{x}, 0) = T$, with the boundary conditions $x(0) = \pi,\ x(T) = \dot{x}(0) = \dot{x}(T) = 0$. Here the constraint set S_Q is a single point in the state space. It is well known that the optimal control for this problem is "bang-bang":

$$u^*(t) = \begin{cases} -1 \text{ if } t \in [0, T/2) \\ \ \ 1 \text{ if } t \in [T/2, T] \end{cases}$$

Under CLP, the trajectory of sixth agent was optimal. It is interesting to note that in this case, optimality was achieved after a finite number of iterations. Some of the iterated trajectories are shown in Fig. 3.

A second example demonstrates the solution of a "geodesic discovery" problem on the plane. The agents were governed by $\dot{x}(t) = u(t),\ x(t),\ u(t) \in \mathbb{R}^2$. The constraint set was a circle with radius 10, centered at the origin. All agents departed from the point $(30, 30)$ and moved with a constant

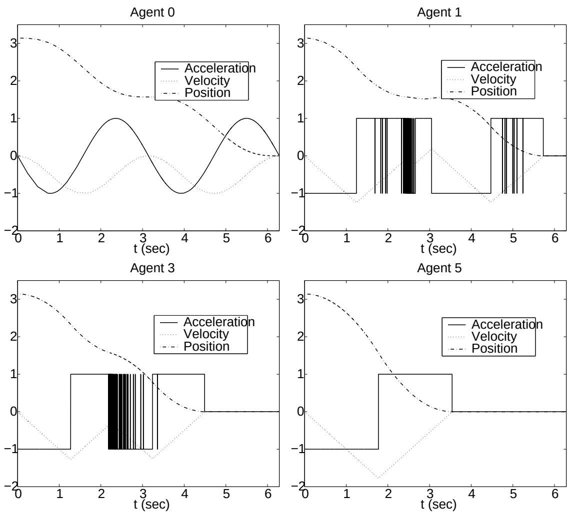

Fig. 3. Iterated trajectories for minimum time control problem through continuous local pursuit with $\Delta = 0.3\pi$, $T_0 = \pi$. The initial trajectory was produced using $u_0 = -\sin(2t)$.

speed $\|\dot{x}\| = 1$. The time separation Δ between agents was 20 seconds. Each agent followed a straight line toward its predecessor before the predecessor reached the circle, and moved on a straight line perpendicular to the circle, once its predecessor had reached the constraint set. As illustrated in Fig. 4, the trajectories converge to the optimum.

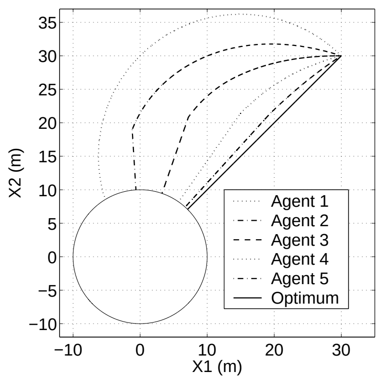

Fig. 4. Iterated trajectories generated by continuous local pursuit for geodesic discovery problems on a plane with constrained final state.

5. CONCLUSIONS AND ONGOING WORK

This paper explored a biologically-inspired cooperative strategy (termed "Continuous Local Pursuit") for solving a class of optimal control problems with free final time and partially-constrained final state. The proposed algorithm generalizes previous models that mimic the foraging behavior of ant colonies and allows a collective to discover optimal controls, starting from an initial suboptimal solution. Members of the collective are only required to obtain local information on their environment and to calculate optimal trajectories to their nearby neighbors. The CLP algorithm relies on cooperation to perform a task which would be difficult or impossible for a single system to perform, namely solving an optimal control problem with limited information (in terms of coordinate systems that describe the environment or the coordinates of the final state) and short-range sensing.

There are several natural extensions of this work, including investigating the algorithm's convergence rate, as well as its ability to lead to global (as opposed to local) optimum by choice of the algorithm's parameters. It would also be interesting to explore a "sampled" version of local pursuit as a numerical method for computing optimal controls.

REFERENCES

Brooks, R.A. and A.M. Flynn (1989). Fast, cheap and out of control: a robot invasion of the solar system. *Journal of the British Interplanetary Society* **42**, 478–485.

Bruckstein, A.M. (1993). Why the ant trails look so straight and nice. *The Mathematical Intelligencer* **15(2)**, 59–62.

Camazine, S. (2001). *Self-Organization in Biological Systems*. Princeton University Press.

Dorigo, M., V. Maniezzo and A. Colorni (1996). Ant systems: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B* **26(1)**, 29–41.

Hristu-Varsakelis, D. (2000). Robot formations: Learning minimum-length paths on uneven terrain. In: *Proceedings of the 8th IEEE Mediterranean Conference on control and Automation.*

Jadbabaie, A., J. Lin and A.S. Morse (2003). Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control* **48**, 988–1001.

Roumeliotis, S.I. and G.A. Bekey (2002). Distributed multi-robot localization. *IEEE Transactions on Robotics and Automation* **18(5)**, 781–795.

Shao, C. and D. Hristu-Varsakelis (2004). Biologically inspired algorithms for optimal control. Technical Report TR2004-29. Institute for Systems Research. University of Maryland, College Park, MD 20742.

Wagner, I.A., M. Lindenbaum and A.M. Bruckstein (1999). Distributed covering by ant-robots using evaporating traces. *IEEE Transactions on Robotics and Automation* **15(5)**, 918–933.