

# A Decentralized Control Framework for Energy-Optimal Goal Assignment and Trajectory Generation

Logan E. Beaver, *Student Member, IEEE*, Andreas A. Malikopoulos, *Senior Member, IEEE*

**Abstract**—This paper proposes a decentralized approach for solving the problem of moving a swarm of agents into a desired formation. We propose a decentralized assignment algorithm which prescribes goals to each agent using only local information. The assignment results are then used to generate energy-optimal trajectories for each agent which have guaranteed collision avoidance through safety constraints. We present the conditions for optimality and discuss the robustness of the solution. The efficacy of the proposed approach is validated through a numerical case study to characterize the framework’s performance on a set of dynamic goals.

## I. INTRODUCTION

### A. Motivation

Complex systems are encountered in many applications, including cooperative autonomous agents, sensor fusion, and biological systems. Referring to something as complex implies that it consists of interconnected agents which adapt and respond to their local and global environment. As we move towards increasingly complex systems [1], new control approaches are needed to optimize the impact on system behavior of the interaction between its entities [2], [3].

Robotic swarm systems can exhibit complex behavior and have attracted considerable attention in many applications, e.g., transportation [4]–[6], construction [7], [8], and surveillance [9]. A common requirement for swarms is to move into a desired formation. However, due to cost constraints imposed on individual agents in a swarm, e.g., limited computation capabilities, battery capacity, and sensing abilities, any efficient control approach must take into account energy consumption. The task of moving in a specified formation has been explored in the literature [10]–[12]. However, achieving formations with minimum energy consumption during operation has not yet been thoroughly investigated.

Several approaches to building cohesive formations in robotic systems have been proposed, such as formations built from triangular sub-structures [13], [14], where a scalable formation is achieved through the construction of a series of isosceles triangles. Methods inspired by crystal growth [15] and lattices structures [16] have also shown promise. Other control methods using only scalar, bearing, or distance measurements were presented by Swartling et al. [17]. This approach was generalized to include the case where only a

single leader agent was able to make distance or bearing measurements.

The problem of generating a desired formation was solved via scheduling by Turpin et al. [18], where an initial assignment is achieved using a scheduling-based heuristic run on a central computer with global information. A significant amount of work, e.g., Wang and Xin [19], Sun and Cassandras [20], Xu and Carrillo [21], and Rajasree and Jisha [22], used optimization techniques in their solutions. However, these methods optimized the position of each agent in a virtual potential field and did not consider energy consumption by individual agents.

The contribution of this paper is an assignment and trajectory generation algorithm which uses only local information for each agent. Other approaches, such as those by Turpin et al. [23], Morgan et al. [24], or Rubenstein et al. [25], required global information in terms of a priori assignment, characteristics about the communication network size, or specifically oriented seed agents, respectively. Our proposed formulation is valid for any feasible initial and final conditions, requiring only that the initial and final positions be non-overlapping. In addition, the formulation does not rely on potential fields [19]–[21], and instead produces energy-optimal trajectories which use proactive steering to avoid collisions.

The remainder of this paper proceeds as follows. In Section II, we formulate the decentralized optimal control problem for each agent. In Section III, we provide the problem formulation and solution approach of the assignment and trajectory generation and discuss implications on robustness. We present a numerical case study in Section IV, which shows the behavior of the proposed method. Finally, we draw concluding remarks and discuss some ideas about future work in Section V.

## II. PROBLEM FORMULATION

We consider the set  $\mathcal{A} = \{1, \dots, N\}$ ,  $N \in \mathbb{N}_{>0}$ , to index a system of autonomous agents in  $\mathbb{R}^2$ . The agents are moving into a desired formation indexed by a set of  $\mathcal{F} = \{1, \dots, M\}$ ,  $M \in \mathbb{N}_{>0}$ , goals. We consider the case where  $N \leq M$ , i.e., no redundant agents are brought to fill the formation, as shown in Fig. 1. This requirement can be relaxed by defining a behavior for excess agents, such as idling [23].

Each agent,  $i \in \mathcal{A}$ , is modeled as a double integrator

$$\dot{\mathbf{p}}_i(t) = \mathbf{v}_i(t), \quad (1)$$

$$\dot{\mathbf{v}}_i(t) = \mathbf{u}_i(t), \quad (2)$$

This research was supported in part by ARPAE’s NEXTCAR program under the award number DE-AR0000796 and by the Delaware Energy Institute (DEI).

The authors are with the Department of Mechanical Engineering at the University of Delaware in Newark, DE 19716, USA (emails: lebeaver@udel.edu; andreas@udel.edu)

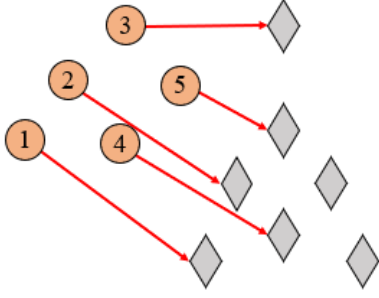


Fig. 1. A group of  $N = 5$  agents entering a formation consisting of  $M = 7$  goals in  $\mathbb{R}^2$ .

where  $\mathbf{p}_i(t) \in \mathbb{R}^2$  and  $\mathbf{v}_i(t) \in \mathbb{R}^2$  are the time-varying position and velocity vectors respectively, and  $\mathbf{u}_i(t) \in \mathbb{R}^2$  is the control input over time  $t \in [0, T_i]$ , where  $T_i \in \mathbb{R}_{>0}$ , is the arrival time for agent  $i$  to its assigned goal. Each agent's velocity and control input are bounded, namely,

$$v_{\min} \leq \|\mathbf{v}_i(t)\| \leq v_{\max}, \quad (3)$$

$$u_{\min} \leq \|\mathbf{u}_i(t)\| \leq u_{\max}, \quad (4)$$

where  $\|\cdot\|$  denotes the Euclidean norm, and  $v_{\min}$ ,  $v_{\max}$  and  $u_{\min}$ ,  $u_{\max}$  are the minimum and maximum allowable speed and control input respectively for each agent  $i \in \mathcal{A}$ . The state of each agent is the time-varying vector

$$\mathbf{x}_i(t) = \begin{bmatrix} \mathbf{p}_i(t) \\ \mathbf{v}_i(t) \end{bmatrix}. \quad (5)$$

Our objective is to develop a framework for the  $N$  agents to optimally, in terms of energy, create any feasible formation of  $M$  points while avoiding collisions between agents. The energy consumption of each agent  $i \in \mathcal{A}$  is given by

$$\dot{E}_i(t) = \frac{1}{2} \|\mathbf{u}_i(t)\|^2. \quad (6)$$

By minimizing the  $L^2$  norm of the control input (acceleration/deceleration) we will have direct benefits in energy consumption.

**Definition 1.** The *desired formation* is the set of time-varying goals  $\mathcal{G}(t) = \{\mathbf{p}_k(t) : \mathbb{R} \rightarrow \mathbb{R}^2 \mid k \in \mathcal{F}\}$ . The set  $\mathcal{G}$  can be prescribed offline, i.e., by a human designer, or online by a high-level planner.

Next, we present our modeling framework, which outlines the approach and assumptions used to solve the minimum energy desired formation problem.

#### A. Modeling Framework

In this framework, the agents can communicate with each other. The maximum sensing and communication range,  $h \in \mathbb{R}_{>0}$ , is used to define a neighborhood for each agent.

**Definition 2.** The *neighborhood* of agent  $i \in \mathcal{A}$  is defined as the time-varying set

$$\mathcal{N}_i(t) = \{j \in \mathcal{A} \mid \|\mathbf{p}_i(t) - \mathbf{p}_j(t)\| \leq h\}.$$

An agent  $i \in \mathcal{A}$  is able to measure the relative position of any neighboring agent  $j \in \mathcal{N}_i$ . This leads to a natural definition of the scalar separating distance.

**Definition 3.** The *scalar separating distance* is defined as

$$r_{ij}(t) = \|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|.$$

Each agent  $i \in \mathcal{A}$  occupies a closed disk of radius  $R$ . To guarantee no collisions between any agents  $i, j \in \mathcal{A}$ , we impose the following conditions on the system

$$r_{ij}(t) > 2R, \quad t \in \mathbb{R}_{>0}, \quad (7)$$

$$h \gg 2R. \quad (8)$$

To ensure each goal in the formation is feasible, the following condition should hold

$$\min_{\mathbf{p}(t), \mathbf{q}(t) \in \mathcal{G}} \{\|\mathbf{p}(t) - \mathbf{q}(t)\|\} < 2R, \quad t \in \mathbb{R}_{>0}. \quad (9)$$

In our modeling framework we impose the following assumptions:

**Assumption 1.** The state  $\mathbf{x}_i(t)$  for each agent  $i \in \mathcal{A}$  is perfectly observed and there is negligible communication delay between the agents.

Assumption 1 is required to evaluate the idealized deterministic performance of the generated optimal solution.

**Assumption 2.** All agents are homogeneous, and any agent may fill any goal in the formation.

This assumption simplifies the trajectory generation and assignment problems, and it can generally be relaxed by adding goal types as a constraint on the goal assignment.

**Assumption 3.** The energy cost of communication is negligible; the only energy consumption is in the form of (6).

The strength of this assumption is application dependent. For cases with long-distance communications or high data rates, the trade-off for communication cost can be controlled by the selection of  $h$ .

Under this framework, the energy-optimal desired formation problem can be solved. This problem can be decomposed into two coupled subproblems: (1) goal assignment and (2) trajectory generation. Both of these problems are described in the following section, with emphasis on the goal assignment.

### III. SOLUTION APPROACH

The decentralized desired formation problem is solved by decomposing it into the coupled goal assignment and trajectory generation subproblems. To decouple these problems the minimum energy objective in the assignment problem is approximated by the minimum Euclidean distance. Prior work, [23], [24], has shown that this approximation is generally sufficient. This enables the assignment problem to be solved independently, which results in the endpoint constraints for the minimum-energy trajectory generation.

### A. Assignment Problem

The objective of the assignment problem is to assign each agent to a goal such that the total distance traveled by all agents is minimized. In the decentralized case, each agent  $i \in \mathcal{A}$  only has information about the positions of its neighbors,  $j \in \mathcal{N}_i$ , and the available goals,  $\mathcal{G}$ . A local assignment can be realized with the use of a local assignment matrix,  $\mathbf{A}_i$ ,

$$\begin{pmatrix} \mathbf{p}_1(T_1) \\ \mathbf{p}_2(T_2) \\ \vdots \\ \mathbf{p}_n(T_n) \end{pmatrix} = \mathbf{A}_i \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ g_M \end{pmatrix}, \quad (10)$$

where  $\mathbf{p}_j(T_j)$ ,  $j \in \mathcal{N}_i$  are the final positions of each agent being assigned,  $g_k$ ,  $k \in \mathcal{F}$ , are the indices for each goal, and the elements  $a_{jk} \in \mathbf{A}_i$  are binary assignment variables. Each agent  $i \in \mathcal{A}$  can solve (10) independently as a linear program, and use the solution to select the prescribed goal.

**Definition 4.** For each agent  $i \in \mathcal{A}$  the *prescribed goal*,  $\mathbf{p}_i^a(t)$ , is defined as the goal assigned to agent  $i$  for which

$$\mathbf{p}_i^a(t) \in \{\mathbf{p}_k \in \mathcal{G} \mid k \in \mathcal{F}, a_{ik} = 1, a_{ik} \in \mathbf{A}_i\}, \quad (11)$$

where the right hand side is a singleton set.

It is possible for multiple agents to have the same prescribed goal. This occurs when two agents  $i \in \mathcal{A}$ ,  $j \in \mathcal{N}_i$ , have different neighborhoods and use conflicting information to solve their assignment problem. This conflict is resolved by introducing the *banned goal* set, defined next.

**Definition 5.** For any agent  $i \in \mathcal{A}$ , the *banned goal* set is defined as the set  $\mathcal{B}_i \subset \mathcal{G}$  which consists of all goals that agent  $i$  is *permanently banned from* when solving the goal assignment (10).

The following definitions and algorithm are presented for some agent  $i \in \mathcal{A}$ . However, all steps are performed *simultaneously by all agents*. For this agent  $i$ , the banned goal set is initially empty. Goals may be added to this set whenever the following condition is not satisfied

$$\mathbf{p}_i^a(t) \neq \mathbf{p}_j^a(t), \quad \forall j \in \mathcal{N}_i(t). \quad (12)$$

In the case that (12) is not satisfied, some agent(s) must be *permanently banned* from the conflict goal, defined for agent  $i$  as

$$\mathbf{p}_c(t) := \mathbf{p}_i^a(t). \quad (13)$$

Banning is achieved by sequential application of ‘‘tiebreaker’’ heuristics which compare:

- 1) the size of each agent’s neighborhood,
- 2) the distance between each agent and the goal, and
- 3) the index of each agent.

Since the metrics of criteria 1, 2, and 3 are perfectly measurable (Assumption 1), it follows that all agents must agree on the tiebreaker resolutions. The tiebreaker hierarchy allows the banned goal set to be broken into three partitions,

$$\mathcal{B}_i(t) = \mathcal{B}_i^1(t) \cup \mathcal{B}_i^2(t) \cup \mathcal{B}_i^3(t), \quad (14)$$

where superscripts 1, 2, and 3 refer to the three tiebreakers, respectively. The tiebreakers are performed by all agents in the set of competing agents, defined next.

**Definition 6.** The set of *competing agents* for agent  $i \in \mathcal{A}$  is defined as

$$\mathcal{C}_i(t) = \{k \in \mathcal{N}_i(t) \mid \mathbf{p}_a^k(t) = \mathbf{p}_c(t)\}.$$

When  $|\mathcal{C}_i| > 1$  there are at least two agents,  $i, j \in \mathcal{N}_i$  assigned to  $\mathbf{p}_c$ . Similarly to (14), the set of competing agents can be split into three decreasing subsets,

$$\mathcal{C}_i^3 \subseteq \mathcal{C}_i^2 \subseteq \mathcal{C}_i^1 = \mathcal{C}_i \quad (15)$$

where the superscripts 1, 2, and 3 correspond to the agents which are comparing the three tiebreaker heuristics.

For each agent  $i \in \mathcal{A}$ , the banned goal sets partitions in (14) are defined as

$$\mathcal{B}_i^m(t) = \left\{ \bigcup_{\tau=0}^t \left( \{\mathbf{p}_i^a(\tau)\} \cap \Phi_i^m(\tau) \right) \right\}, \quad (16)$$

where  $\Phi_i^m(t)$  is given by

$$\Phi_i^m(t) = \begin{cases} \mathcal{G}, & \text{if } m = 1, i \neq \underset{j \in \mathcal{C}_i^1(t)}{\operatorname{argmax}}\{|\mathcal{N}_j|(t)\}, \\ \mathcal{G}, & \text{if } m = 2, i \neq \underset{j \in \mathcal{C}_i^2(t)}{\operatorname{argmax}}\{\|\mathbf{p}_c(t) - \mathbf{p}_j(t)\|\}, \\ \mathcal{G}, & \text{if } m = 3, i \neq \underset{j \in \mathcal{C}_i^3(t)}{\operatorname{argmin}}\{j\}, \\ \emptyset, & \text{otherwise.} \end{cases} \quad (17)$$

where  $m \in \{1, 2, 3\}$  again corresponds to the three tiebreaker heuristics.

To begin the tiebreaker process for agent  $i \in \mathcal{A}$ , consider the first conflict set  $\mathcal{C}_i^1$  with the neighborhood heuristic. Every agent  $j \in \mathcal{C}_i^1$  which satisfies

$$j = \underset{k \in \mathcal{C}_i^1}{\operatorname{argmax}}\{|\mathcal{N}_k(t)|\}, \quad (18)$$

is eligible to be assigned to goal  $\mathbf{p}_c$ . If agent  $i \in \mathcal{A}$  uniquely satisfies (18), then the conflict test is complete and  $i$  is assigned to  $\mathbf{p}_c$ . If  $i$  does not satisfy (18), then the goal  $\mathbf{p}_c$  is added to  $\mathcal{B}_i^1(t)$  as designated by (17). Finally, if agent  $i$  does not uniquely satisfy (18) then the second criteria, distance to goal, must be compared. This comparison is done over a reduced conflict set,

$$\mathcal{C}_i^2(t) = \{j \in \mathcal{C}_i^1(t) \mid |\mathcal{N}_j(t)| = |\mathcal{N}_i(t)|\}. \quad (19)$$

The second tiebreaker, maximum distance, is a minimax strategy which seeks to minimize the maximum distance traveled by any agent to the conflict goal. Again, every agent  $j \in \mathcal{C}_i^2$  which satisfies

$$j = \underset{k \in \mathcal{C}_i^2}{\operatorname{argmax}}\{\|\mathbf{p}_k - \mathbf{p}_c\|\}, \quad (20)$$

is eligible to be assigned to goal  $\mathbf{p}_c$ . If agent  $i$  uniquely satisfies (20), then the conflict test is complete and  $i$  is assigned to  $\mathbf{p}_c$ . If  $i$  does not satisfy (20), then the goal  $\mathbf{p}_c$  is added to  $\mathcal{B}_i^2$  per (17). Finally, if  $i$  satisfies (20), but not

uniquely, the final test must be taken over a further reduced conflict set, given by

$$\mathcal{C}_i^3(t) = \left\{ j \in \mathcal{C}_i^2(t) \mid \|\mathbf{p}_c - \mathbf{p}_j\| = \|\mathbf{p}_c - \mathbf{p}_i\| \right\}, \quad (21)$$

where the agent  $k$  satisfying

$$k = \min \left\{ j \in \mathcal{C}_i^3 \right\}, \quad (22)$$

is assigned to the goal, and all other agents add  $\mathbf{p}_c$  to  $\mathcal{B}_i^3$  as designated by (17). After the conflicts are resolved, if the size of  $\mathcal{B}_i$  has increased then the value of  $T_i$  must also increase to

$$T_i = t + T, \quad (23)$$

where  $t$  is the current time, and  $T$  is a system parameter. This allows agent  $i$  a sufficient amount of time to reach its new goal. Finally, for each subsequent assignment involving agent  $i \in \mathcal{A}$ , when  $\mathcal{B}_i(t) \neq \emptyset$  agent  $i$  must broadcast its banned goal set to all  $j \in \mathcal{N}_i$ .

The assignment and banning process is iterated by all  $j \in \mathcal{N}_i$  until (12) is satisfied in the entire neighborhood. The banned and restricted goal information is enforced through a constraint on the assignment problem, which follows.

**Problem 1** (Goal Assignment). Each agent assigns itself a goal independently by solving the linear minimum-distance assignment (10).

For each agent  $i \in \mathcal{A}$ , we have

$$\min_{a_{jk} \in \mathcal{A}_i} \left\{ \sum_{k \in \mathcal{N}_i} \sum_{j \in \mathcal{G}} a_{jk} \|\mathbf{p}_k(t) - \mathbf{p}_j^*(T_k)\| \right\}, \quad (24)$$

$$\mathbf{p}_k^0 \in \mathcal{N}_i, \mathbf{p}_j^*(t) \in \mathcal{G},$$

subject to

$$\sum_{j \in \mathcal{G}} a_{jk} = 1, \quad k \in \mathcal{N}_i, \quad (25)$$

$$\sum_{k \in \mathcal{N}_i} a_{jk} \leq 1, \quad j \in \mathcal{G}, \quad (26)$$

$$a_{jk} = 0, \quad k \in \mathcal{N}_i, \mathbf{p}_j \in \mathcal{B}_k, \quad (27)$$

$$a_{jk} \in \{0, 1\}.$$

Each agent independently solves Problem 1 as a linear program and selects its assigned goal. This process is repeated by each agent,  $i \in \mathcal{A}$ , until  $|\mathcal{C}_i| = 1$ .

As the safety constraints of Problem 1 explicitly depend on the neighborhood of agent  $i \in \mathcal{A}$ , the optimization must be recalculated each time the cardinality of the neighborhood of agent  $i$  changes. Under weak assumptions about the trajectories of each agent, the assignments generated by Problem 1 is guaranteed to bring each agent to a unique goal as it is shown next.

**Lemma 1.** For every agent  $i \in \mathcal{A}$ , if  $\left| \left( \bigcup_{j \in \mathcal{N}_i} \mathcal{B}_j \right) \setminus \mathcal{G} \right| \geq |\mathcal{N}_i|$ , then the feasible region of Problem 1 is always nonempty.

*Proof.* Let the set of goals available to all agents in the neighborhood of agent  $i \in \mathcal{A}$  be denoted by the set

$$\mathcal{V}_i(t) = \{ \mathbf{p} \in \mathcal{G} \mid \mathbf{p} \notin \mathcal{B}_j(t), \forall j \in \mathcal{N}_i(t) \}. \quad (28)$$

Let the injective function  $m_i : \mathcal{N}_i(t) \rightarrow \mathcal{V}_i(t)$  map each agent to a goal. As  $|\mathcal{N}_i| \leq |\mathcal{V}_i(t)|$ , the function  $m_i$  must always exist and imposes a mapping from each agent to a unique goal.

Since  $m_i$  is injective, it satisfies (25) and (26). Likewise,  $\mathcal{V}_i \subset \mathcal{B}_j^c$  for all  $j \in \mathcal{N}_i$ , and therefore the imposed mapping satisfies (27). Therefore, the mapping imposed by the function  $m_i$  is a feasible solution to Problem 1.  $\square$

For a sufficiently large value of  $T$ , the convergence of all agents to goals is guaranteed by Theorem 1.

**Theorem 1** (Assignment Convergence). Under the assumptions of Lemma 1, for a sufficiently large value of  $T$ , and if the energy-optimal trajectories for each agent always move toward their assigned goal, then all  $i \in \mathcal{A}$  must reach an assigned goal in finite time.

*Proof.* Let  $\{g_n\}_{n \in \mathbb{N}}$  be the sequence of goals assigned to agent  $i \in \mathcal{A}$  as designated by the solution of Problem 1. From Lemma 1,  $\{g_n\}_{n \in \mathbb{N}}$  is not empty, and the elements of this sequence are integers bounded by  $1 \leq g_n \leq |\max \mathcal{F}|$ . Thus, the range of this sequence is compact and must be (1) finite, (2) convergent, or (3) periodic.

(1) For a finite sequence,  $T_i$  is bounded by  $T \cdot |\mathcal{G}|$ .

(2) Under the discrete metric, an infinite convergent sequence requires that there exists  $N \in \mathbb{N}_{>0}$  such that  $g_n = p$  for all  $n > N$  for some formation index  $p \in \mathcal{F}$ . This reduces to case 1, as  $T_i$  does not increase for repeated assignments to the same goal.

(3) By the Bolzano-Weierstrass Theorem, an infinite non-convergent sequence  $\{g_n\}_{n \in \mathbb{N}}$  must have a convergent subsequence, i.e., agent  $i$  is assigned to some subset of goals  $\mathcal{I} \subseteq \mathcal{G}$  infinitely many times with some number of intermediate assignments for each goal  $\mathbf{g} \in \mathcal{I}$ . From the construction of the banned goal set, we must have  $\mathcal{I} \cap \mathcal{B}_i(t) = \emptyset$  for all  $t \in [0, T_i]$ . This implies that, by the update method of  $T_i$ , the position of all goals,  $g(t) \in \mathcal{I}$  must only be considered at time  $T_i$ , which we denote as  $\mathbf{g}(T_i) \in \mathcal{I} = \mathbf{g} \in \mathcal{I}$ .

This implies that the goals available to agent  $i$ , i.e.,  $\mathcal{I} = \mathcal{G} \setminus \mathcal{B}_i$ , must be shared between  $n > 0$  other periodic agents. This implies at some time  $t_1$  that a goal,  $\mathbf{g} \in \mathcal{I}$ , must be an optimal assignment for agent  $i$ , a non optimal assignment at time  $t_2 > t_1$  and an optimal assignment at time  $t_3 > t_2$ . This implies the distance between agent  $i$  and goal  $\mathbf{g}$  satisfy

$$\|\mathbf{p}_i(t_1) - \mathbf{g}\| < \|\mathbf{p}_i(t_1) - \mathbf{g}'\|, \quad (29)$$

$$\|\mathbf{p}_i(t_2) - \mathbf{g}'\| < \|\mathbf{p}_i(t_2) - \mathbf{g}\|, \quad (30)$$

$$\|\mathbf{p}_i(t_3) - \mathbf{g}\| < \|\mathbf{p}_i(t_3) - \mathbf{g}'\|, \quad (31)$$

for some goal  $\mathbf{g}' \in \mathcal{I}$ ,  $\mathbf{g}' \neq \mathbf{g}$ . Agent  $i$  must not increase his

distance from his assigned goal, which implies

$$|\mathbf{p}_i(t_1) - \mathbf{g}| > |\mathbf{p}_i(t_2) - \mathbf{g}|, \quad (32)$$

$$|\mathbf{p}_i(t_2) - \mathbf{g}'| > |\mathbf{p}_i(t_3) - \mathbf{g}'|, \quad (33)$$

$$(34)$$

and hence

$$|\mathbf{p}_i(t_1) - \mathbf{g}'| > |\mathbf{p}_i(t_3) - \mathbf{g}'|, \quad (35)$$

which is satisfied for all goals  $\mathbf{g}' \in \mathcal{I}$ . This is only possible if agent  $i$  simultaneously approaches all goals in  $\mathcal{I}$ , which implies they are arbitrarily close. This contradicts (9), and thus no such periodic behavior may exist.  $\square$

### B. Optimal Trajectory Generation

After the goal assignment is determined, each agent must generate a collision-free and energy-optimal trajectory to their assigned goal. The initial and final condition constraints for any agent  $i \in \mathcal{A}$  are given by

$$\left( \mathbf{p}_i(t_0) - \mathbf{p}_{i,0}, \mathbf{v}_i(t_0) - \mathbf{v}_{i,0} \right) = \left( \mathbf{0}, \mathbf{0} \right), \quad (36)$$

$$\left( \mathbf{p}_i(T_i) - \mathbf{p}_{i,f}, \mathbf{v}_i(T_i) - \mathbf{v}_{i,f} \right) = \left( \mathbf{0}, \mathbf{0} \right), \quad (37)$$

where  $\mathbf{p}_{i,f}, \mathbf{v}_{i,f}$  are the result of solving Problem 1. To resolve the coupling introduced by collision avoidance, each agent  $i \in \mathcal{A}$  predicts the optimal trajectories of its neighbors,  $j \in \mathcal{N}_i$  to select its prescribed trajectory.

**Definition 7.** The *prescribed trajectory*,  $u_i^a(t)$ , is the trajectory assigned to agent  $i$  after solving for the optimal trajectories of every agent in its neighborhood,  $j \in \mathcal{N}_i$ .

For agent  $i$  to calculate its prescribed trajectory,  $u_i^a(t)$ , the trajectory optimization problem must be solved over the set

$$\mathbf{U}_i(t) = \{u_j(t) : j \in \mathcal{N}_i\},$$

such that

$$u_i^a(t) = u_i(t) \in \mathbf{U}_i(t).$$

This can be achieved by the quadratic optimization problem given by:

**Problem 2 (Trajectory Generation).** For each agent  $i \in \mathcal{A}$ , we have

$$\min_{\mathbf{u}_i \in \mathbf{U}} \left\{ \sum_{j \in \mathcal{N}_i} \int_{\tau=t}^{T_i} \|\mathbf{u}_j(\tau)\| dt \right\}, \quad (38)$$

subject to

- Dynamic constraints (1), (2),
- State and control bounds (3), (4),
- Collision avoidance (7),
- Boundary constraints (36), (37).

Problem 2 can be solved as an iterated quadratic program with a similar conflict framework as Problem 1, where one agent fixes its trajectory and others steer to avoid it.

Problems 1 and 2 are solved sequentially at time  $t = 0$  to achieve an initial set of assignments and corresponding optimal trajectories. As both optimizations only use local information, each agent must resolve each problem whenever their neighborhood changes. This ensures that every agent is using all available information to optimize their trajectories while guaranteeing collision avoidance.

**Remark 1.** The solutions of Problems 1 and 2 reduce to the centralized case as  $h \rightarrow \infty$ .

Remark 1 relies on the fact that, as  $h \rightarrow \infty$ , it must be true that  $\mathcal{N}_i(t_0) = \mathcal{A}$  and  $\mathcal{B}_i(t_0) = \emptyset \forall i \in \mathcal{A}$ . Hence, problems 1 and 2 simply reduce to each agent solving the centralized problem individually.

## IV. SIMULATION CASE STUDY

To give insight into the behavior of the agents a series of simulations were performed in Matlab. Each simulation lasted for 20 s or until all agents reach their assigned goal, whichever was longer. The centroid of the formation moved with a fixed velocity, while the leftmost and rightmost three goals included additional periodic motion

The minimum separating distance between agents, total energy consumed, and maximum velocity for the unconstrained solutions to Problem 2 are given as a function of the horizon in Table I. A graph of each agent's position over time for two cases is given in Fig. 2 and 3.

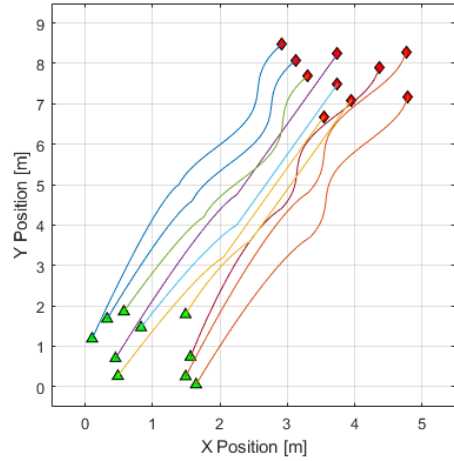


Fig. 2. Agent trajectories for the centralized case, here the assignment globally minimizes distance travelled and trajectories are evenly spaced.

The results in Table I generally show no correlation between energy consumption and sensing horizon. In fact, the minimum energy consumption occurs near  $R = 1.3$  m rather than the centralized case. This is likely a result of the minimum distance approximation, which does not account for the required change in velocity for a dynamic formation with moving goals.

## V. CONCLUSION

In this paper, we proposed an approach for solving the desired formation problem of a group of autonomous agents.

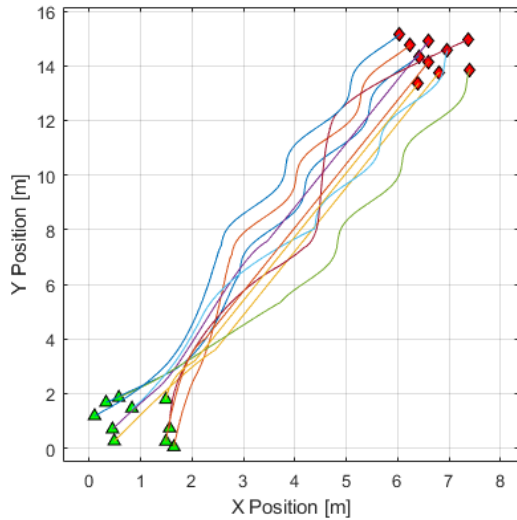


Fig. 3. Agent trajectories for  $R = 0.5\text{m}$ , one grid cell in diameter. Agents use very little information and tend to group at the nearest goal position.

$h$ [m]	min. separation [cm]	$E$ [kJ/kg]	$t_f$ [s]
inf	16.5	63.77	20
1.60	0.82	83.76	27
1.50	1.21	56.43	20
1.40	0.38	140.7	41
1.30	5.25	52.26	20
1.10	0.32	96.13	34
0.95	0.54	41.61	20
0.75	0.60	227.7	42
0.50	2.41	140.1	39

TABLE I

NUMERICAL RESULTS FOR  $N=M=10$  AGENTS AND GOALS, WITH A TIME PARAMETER OF  $T = 10$  S AND VARIOUS SENSING DISTANCES.

We presented a formulation of the formation reconfiguration problem and introduced a concept of prescribed goals and trajectories. The robustness and convergence properties of the system were discussed, and the performance was characterized relative to the centralized approach. A numerical solution was presented for  $N = M = 10$  agents and goals, and the system performance metrics were compared relative to the sensing radius.

Future areas of research include: relaxing the assumptions on Lemma 1 to characterize when solutions exist, incorporating information from outside the neighborhood into goal assignment, analyzing the effect on sensing radius on communication cost versus convergence and propulsion energy, reducing the computational load in calculating Problem 2, and characterizing the optimality of the tiebreaker heuristics.

## REFERENCES

[1] A. A. Malikopoulos, "Centralized stochastic optimal control of complex systems," in *Proceedings of the 2015 European Control Conference*, 2015, pp. 721–726.  
 [2] A. A. Malikopoulos, V. Maroulas, and J. Xiong, "A multiobjective optimization framework for stochastic control of complex systems," in *Proceedings of the 2015 American Control Conference*, 2015, pp. 4263–4268.

[3] A. A. Malikopoulos, "Equilibrium Control Policies for Markov Chains," in *50th IEEE Conference on Decision and Control and European Control Conference*, 2011, pp. 7093–7098.  
 [4] A. A. Malikopoulos, C. G. Cassandras, and Y. J. Zhang, "A decentralized energy-optimal control framework for connected automated vehicles at signal-free intersections," *Automatica*, vol. 93, no. April, pp. 244–256, 2018.  
 [5] L. Zhao and A. A. Malikopoulos, "Enhanced mobility with connectivity and automation: A review of shared autonomous vehicle systems," *IEEE Intelligent Transportation Systems Magazine*, 2020.  
 [6] W. Ren and E. Atkins, "Distributed multi-vehicle coordinated control via local information exchange," *International Journal of Robust and Nonlinear Control*, 2007.  
 [7] Q. Lindsey, D. Mellinger, and V. Kumar, "Construction with quadrotor teams," *Autonomous Robots*, vol. 33, no. 10, pp. 323–336, 2012.  
 [8] J. Werfel, K. Petersen, and R. Nagpal, "Designing Collective Behavior in a Termite-Inspired Robot Construction Team," *Science*, vol. 343, pp. 754–758, 2014.  
 [9] J. Cortes, "Global formation-shape stabilization of relative sensing networks," in *Proceedings of the American Control Conference*, 2009.  
 [10] H. Oh, A. R. Shirazi, C. Sun, and Y. Jin, "Bio-inspired self-organising multi-robot pattern formation: A review," *Robotics and Autonomous Systems*, no. 91, pp. 83–100, 2017.  
 [11] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, no. 53, pp. 424–440, 2015.  
 [12] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: A review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.  
 [13] J. Guo, Z. Lin, M. Cao, and G. Yan, "Adaptive control schemes for mobile robot formations with triangularised structures," *IET Control Theory & Applications*, vol. 4, no. 9, pp. 1817–1827, 2010.  
 [14] Y. Hanada, G. Lee, and N. Y. Chong, "Adaptive Flocking of a Swarm of Robots Based on Local Interactions," in *IEEE Swarm Intelligence Symposium*, 2007, pp. 340–347.  
 [15] Y. Song and J. M. O'Kane, "Forming repeating patterns of mobile robots: A provably correct decentralized algorithm," in *IEEE International Conference on Intelligent Robots and Systems*, vol. 2016–Novem, 2016, pp. 5737–5744.  
 [16] G. Lee and Y. C. Nak, "Self-configurable mobile robot swarms with hole repair capability," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2008.  
 [17] J. O. Swartling, I. Shames, K. H. Johansson, and D. V. Dimarogonas, "Collective Circumnavigation," *Unmanned Systems*, vol. 02, no. 03, pp. 219–229, 2014.  
 [18] M. Turpin, K. Mohta, N. Michael, and V. Kumar, "Goal Assignment and Trajectory Planning for Large Teams of Aerial Robots," *Proceedings of Robotics: Science and Systems*, vol. 37, pp. 401–415, 2013.  
 [19] J. Wang and M. Xin, "Integrated Optimal Formation Control of Multiple Unmanned Aerial Vehicles," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1731–1744, 2013.  
 [20] X. Sun and C. G. Cassandras, "Optimal Dynamic Formation Control of Multi-Agent Systems in Environments with Obstacles," in *IEEE Conference on Decision and Control*, 2015, pp. 2359–2364.  
 [21] H. Xu and R. G. C. Carrillo, "Distributed Near Optimal Flocking Control for Multiple Unmanned Aircraft Systems," in *International Conference on Unmanned Aerial Vehicles*. Denver, CO, USA: ICUAS, 2015, pp. 879–885.  
 [22] R. Rajasree, "Optimal Formation Control of Unmanned Aerial Vehicles with Obstacle Avoidance," in *International Conference on Control, Instrumentation, Communication and Computational Technologies*, 2010.  
 [23] M. Turpin, N. Michael, and V. Kumar, "CAPT: Concurrent assignment and planning of trajectories for multiple robots," *International Journal of Robotics Research*, vol. 33, no. 1, pp. 98–112, 2014.  
 [24] D. Morgan, G. P. Subramanian, S.-J. Chung, and F. Y. Hadaegh, "Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming," *International Journal of Robotics Research*, vol. 35, no. 10, pp. 1261–1285, 2016.  
 [25] M. Rubenstein, C. Ahler, and R. Nagpal, "Kilobot: A low cost scalable robot system for collective behaviors," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3293–3298, 2012.