On The Transfer Time Complexity of Cooperative Vehicle Routing

Kevin Spieser, Dimos V. Dimarogonas, and Emilio Frazzoli

***Abstract*— Motivated by next-generation air transportation systems, this paper investigates the relationship between traffic volume and congestion in a multi-agent system, assuming that the agents can communicate their intentions with one another. In particular, we consider n independent mobile agents, each assigned an origin and a destination point, and study how the minimum time necessary to safely transfer all agents from their origin to their destination scales with the number of agents n. We provide an algorithm for which the transfer time scales logarithmically in n. This is an improvement over previous results that rely on more conservative conflict models because they do not leverage inter-agent cooperation to the same degree, resulting in transfer times scaling as $\sqrt{n}$.**

I. Introduction

The current air transportation system (ATS) is being forced to operate ever closer to its critical capacity, leading to increases in both the frequency and duration of delays. This strain is expected to worsen alongside a projected two- to three- fold increase in the demand for air travel [1]. It is widely acknowledged the current ATS lacks the operational scalability to meet this demand, and that there is an urgent need for sweeping change in how the National Airspace (NAS) is managed. To achieve improvements in system efficiency, while safely increasing the capacity of the NAS, there has been a growing movement to embrace the notion of free-flight and afford aircraft greater autonomy in planning routes and during flight [1]. Such a move would represent an epoch-defining shift from predominantly centralized, preplanned operations under the close supervision of human air-traffic controllers, to more flexible, primarily autonomous operations relying on inter-agent communication and coordination to maintain a well-functioning airspace.

While an autonomous airspace holds the promise to address many of the issues that plague the current ATS, the task of instituting new regulatory policies is complicated by the need to ensure safety on a system-wide level, and by the wide assortment of feasible regulatory policies. In order to make informed decisions about which next-gen policies to put in place, it is necessary to have a precise understanding of the performance that is and, perhaps more importantly, is not possible in such systems. In other words, we must establish quantifiable relationships between key metrics that measure how efficiently the airspace is used, such as delay and throughput, and environmental parameters, such as traffic intensity, navigational uncertainty, and the onset of inclement weather, that may degrade during operation.

Research efforts aimed at coordinating the motion of multiple autonomous, mobile agents have yielded an array of techniques for moving agents between specified points in a shared workspace, while avoiding collisions (see, for example, [2]–[9]). Often, it is prudent to consider the complexity of these methods from various perspectives, including, for example, the computational complexity of the algorithm, for which a number of results have been developed (see [10] for a timeline). Recently, researchers have begun to address other facets of complexity, including the communication complexity (the amount of information that must be gathered and or shared among agents) and time complexity (the actual time required to complete the task) [11]. In [10], the authors investigate how the time required to transfer n agents between specified origin and destination points, without conflicts, scales as the number of agents gets large. They consider various conflict models and assume agents do not communicate with one another.

In many applications, including the next generation of air transportation systems, the agents that make up the system will have the ability to communicate with one another. In this paper, we show that, given this added functionality, there exists a transfer policy that is free of conflicts and scales logarithmically in the number of agents. It is worth reinforcing that although we are interested in determining performance limitations associated with the traffic density in an autonomous airspace, the discussion to follow is probably also applicable to other domains in which large numbers of autonomous agents navigate in a shared environment. For this reason, we use the terms agent and aircraft interchangeably throughout.

This paper is organized in sections. Section II provides the formulation of the problem we consider, as well as a brief account of related work that has been reported in the literature. Section III provides a new way to model conflicts among agents operating in a shared environment. In Section IV, we provide an algorithm capable of safely transferring all agents, and, by developing bounds on the transfer time, conclude our algorithm is optimal in an asymptotic sense. Simulation results in Section V demonstrate the essential features of our algorithm. Finally, Section VII closes with concluding remarks.

II. The Transfer Problem

We consider the problem of transferring n agents between arbitrary source and destination points in a manner that is free of conflicts and scales favorably as n gets large [10]. Such a scenario is representative of operations in a so-called super-density traffic environment. We label the

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, 02139. {kspieser,ddimar,frazzoli}@mit.edu

agents arbitrarily as $\mathcal{A}_1,\ldots,\mathcal{A}_n$. The workspace of interest is denoted by $\mathcal{W}$ and is assumed to be of fixed area A. The position and velocity of $\mathcal{A}_i$ at time t are denoted by $q_i(t) \in \mathbb{R}^2$ and $v_i(t) \in \mathbb{R}^2$, respectively, for $i = 1,\ldots,n$. It proves advantageous to conduct the analysis to follow in terms of a polar coordinate frame. To this end, we express q_i in terms of the radial and angular component of $\mathcal{A}_i$ as $q_i(t) = (r_i(t), \theta_i(t))$, where $\theta_i(t) \in [0, 2\pi)$.

Associated with each $\mathcal{A}_i$ is an origin point O_i and a destination point D_i. At some point during transfer, $\mathcal{A}_i$ travels along a path that starts at O_i and terminates at D_i. For a particular conflict-free transfer policy, $\mathcal{P}$, we define the *transfer time*, $T_{\mathcal{P}}(n)$, to be the total amount of time required to transfer each $\mathcal{A}_i$ from O_i to D_i. We define the *optimal transfer time*, $T^*(n)$, to be the minimum amount of time needed to transfer each $\mathcal{A}_i$ from O_i to D_i without conflicts. Note that $T^*(n) \leq T_{\mathcal{P}}(n)$ for any $\mathcal{P}$. In this paper, we are interested in how $T_{\mathcal{P}}(n)$ and $T^*(n)$ scale as n gets large for various distributions of origin and destination points. Unsurprisingly, the answer depends on how a conflict is defined, which in turn is tied to how we relate interactions between agents in the workspace to our notion of what constitutes safe operation.

In [10], the authors consider various conflict models for agents that do not share their intentions with one another, and investigate the associated transfer time as a function of n. A traditional approach to model conflicts in multi-agent systems is to assign to each $\mathcal{A}_i$ a safety disc, $\mathcal{M}_i(t)$, of radius $r_i > 0$ centered about $q_i(t)$. A system is said to be free of conflicts if $\mathcal{M}_i(t) \cap \mathcal{M}_j(t) = \emptyset$ for $i, j = 1,\ldots,n$, $i \neq j$, $\forall t$. The question of defining safety then reduces to one of defining r_i. In [10], the authors show that if the safety radius is lower bounded by a constant, i.e., $r_i \geq r_o$, then $T^*(n) = \Theta(n)$. In the same work, the authors consider a safety radius with affine dependence on v_i, the velocity of $\mathcal{A}_i$. They show that if $r_i = r(n) + \zeta|v_i|$ where $\zeta > 0$ is a constant, $r(n) = O(1/\sqrt{n})$ from above, then $T^*(n) = \Theta(\sqrt{n})$.

In the next section, we argue that, in many cases, it is the relative, rather than the absolute, velocity of agents that is most important when describing safety, and define the relative-velocity conflict model that we use in the remainder of this paper.

III. A System Model

For each $\mathcal{A}_i$, in addition to $\mathcal{A}_i$'s position and velocity, at each time instant, we associate $\mathcal{A}_i$ as being in either an *inactive* or *active* state. We exclude inactive agents when considering the conflict condition. Therefore, it is only among the active agents that conflicts must be avoided. We believe this formalism makes sense in a number of settings. For example, in air traffic control, airplanes docked in hangars or parked at terminal gates on the ground (i.e, in an inactive state) do not pose a safety risk to planes already in the sky until they take off (at which time they become active). To fuse the notion of a relative-velocity-based separation distance with the affine dependence on speed model in [10], we propose the following generalized conflict model.

Definition 1 *A transfer policy is free of conflicts, in the general sense, if for every pair of active agents, $\mathcal{A}_i$, $\mathcal{A}_j$,*

$$|q_i(t) - q_i(t)| \geq r_o + \zeta|v_i| + \kappa\left|v_i(t) - v_j(t)\right|, \qquad (1)$$

$\forall i, j \in 1,\ldots,n, \forall t$, *where $\zeta > 0$ and $\kappa > 0$ are constants.* ◇

In systems where agents are able to communicate with one another, as in the proposed next-gen ATS, it is the relative, rather than the absolute, velocity of agents that has the greatest impact on safety. For these systems, it is appropriate for a first-order analysis to base the minimum separation distance between two agents entirely on the difference in their velocities.

Definition 2 *A transfer policy is free of conflicts, in the relative-velocity sense, if for every pair of active agents, $\mathcal{A}_i$, $\mathcal{A}_j$,*

$$|q_i(t) - q_i(t)| \geq \kappa\left|v_i(t) - v_j(t)\right|, \forall i, j \in 1,\ldots,n, \ \forall t, \qquad (2)$$

where $\kappa > 0$ is a constant. ◇

We use the relative-velocity conflict model throughout the remainder of this paper.

Presentation of our algorithm requires we define the *dispersion* associated with a set of points. Let $\mathcal{S} = \{O_1,\ldots,O_n,D_1,\ldots,D_n\}$ denote the set of all origin and destination points in $\mathcal{W}$. The dispersion, denoted r_{disp}, is defined to be the radius of the largest circle that can be inscribed in $\mathcal{W}$ and does not contain any of the points in $\mathcal{S}$ [12]. Mathematically, the dispersion can be expressed as

$$r_{\text{disp}} = \max_{q \in \mathcal{W}}\{\min_{p \in \mathcal{S}} \|q - p\|_2\}. \qquad (3)$$

We also refer to the circle with radius r_{disp} as the dispersion circle and denote its center by q_C. Moreover, for large n (i.e., the super-density case we are interested in), the dispersion can be shown to satisfy the asymptotic bound $r_{\text{disp}} = \Omega(1/\sqrt{n})$ [12].

IV. The spiral algorithm

In this section, we propose a new transfer algorithm, show it is safe under the relative-velocity conflict model, and investigate how its transfer time scales as n gets large. The proposed algorithm, called the SPIRAL ALGORITHM, is composed of two phases: a SPIRAL-IN phase, followed by a SPIRAL-OUT phase. The basic idea is to have all of the agents congregate inside the dispersion circle during the SPIRAL-IN phase and then send them off to their respective destination points during the SPIRAL-OUT phase. Before discussing each phase in detail, assume, for notational simplicity, but without loss of generality, that the workspace is a circle of radius R, and the dispersion circle is centered at the origin. The SPIRAL-IN and SPIRAL-OUT phases are described below.

A. Phase I: SPIRAL-IN

At the beginning of the SPIRAL-IN phase, all aircraft are in an inactive state. This is consistent with aircraft being on the ground at their outset airports, such that no aircraft poses a safety risk to another aircraft. During the SPIRAL-IN phase, $\mathcal{A}_i$ is active over the interval $[-t_{i,1}, 0]$, where $-t_{i,1} < 0$ is

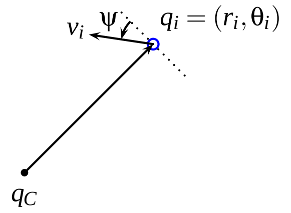

Fig. 1. Geometry of $\mathcal{A}_i$ in $\mathcal{W}$.

the time $\mathcal{A}_i$ is activated and leaves its origin point. Once activated, $\mathcal{A}_i$ travels inward from O_i along a logarithmic spiral trajectory emanating from q_C. Along this path, the polar coordinates of $\mathcal{A}_i$ evolve according to the dynamics

$$\begin{aligned} \dot{r}_i &= -\alpha r_i & (4)\\ \dot{\theta}_i &= \omega, & (5) \end{aligned}$$

where $\alpha > 0$ and $\omega > 0$ are constants. That is, $\mathcal{A}_i$ travels with a speed proportional to its distance from q_C and at a fixed angle inward from the vector tangent to its direction of motion (see Figure 1). If we let $O_i = (r_{i,o}, \theta_{i,o})$, then solving (4) and (5) gives $q_i(0) = (r_i(0), \theta_i(0))$, where

$$\begin{aligned} r_i(0) &= r_{i,o} e^{-\alpha t_{i,1}} & (6)\\ \theta_i(0) &= \theta_{i,o} + \omega t_{i,1}. & (7) \end{aligned}$$

B. Phase II: SPIRAL-OUT

The SPIRAL-OUT phase begins at $t = 0^+$, just after the SPIRAL-IN phase has ended. During the SPIRAL-OUT phase, $\mathcal{A}_i$ is active over the interval $[0, t_{i,2}]$, where $t_{i,2} > 0$ is the time $\mathcal{A}_i$ reaches its destination. We adopt the convention that $\mathcal{A}_i$ is deactivated (i.e., returned to an inactive state) once it reaches its destination. In the case of aircraft, this is indicative of an aircraft reaching its destination and landing. The SPIRAL-OUT phase is analogous to the SPIRAL-IN phase, with two important exceptions. First, all agents begin the SPIRAL-OUT phase at time zero, but, in general, are deactivated at different times. Second, the radial coordinate of $\mathcal{A}_i$ increases, rather than decreases throughout the SPIRAL-OUT phase until the time at which an agent reaches its destination point and is deactivated. In this case, the polar coordinates of $\mathcal{A}_i$ evolve according to the dynamics

$$\begin{aligned} \dot{r}_i &= \alpha r_i & (8)\\ \dot{\theta}_i &= \omega. & (9) \end{aligned}$$

Note the heading angle of $\mathcal{A}_i$ continues to increase during the SPIRAL-OUT phase, just as it did during the SPIRAL-IN phase. Were we to use $\dot{\theta}_i = -\omega$ during the SPIRAL-OUT phase, instead of (8), $\mathcal{A}_i$ would retrace the route it took from S_i, making it impossible to transfer $\mathcal{A}_i$ to destination points not residing on this path.

If we let $D_i = (r_{i,d}, \theta_{i,d})$, then solving (8) and (9) gives

$$\begin{aligned} r_{i,d} &= r_i(0) e^{\alpha t_{i,2}} & (10)\\ \theta_{i,d} &= \theta_i(0) + \omega t_{i,2} - 2\pi k, \text{ for some } k \in \mathbb{Z}^+. & (11) \end{aligned}$$

Combining (6) with (10) to eliminate $r_i(0)$, and (7) with (11) to eliminate $\theta_i(0)$ gives

$$\begin{aligned} r_{i,d} &= r_{i,o} e^{\alpha\left(t_{i,2} - t_{i,1}\right)} & (12)\\ \theta_{i,d} &= \theta_{i,o} + \omega(t_{i,2} + t_{i,1}) - 2\pi k, \text{ for some } k \in \mathbb{Z}^+. & (13) \end{aligned}$$

It follows that

$$\begin{aligned} t_{i,2} - t_{i,1} &= \frac{1}{\alpha} \log\left(\frac{r_{i,d}}{r_{i,o}}\right) & (14)\\ t_{i,2} + t_{i,1} &= \frac{(\theta_{d,i} - \theta_{i,o} + 2\pi k)}{\omega}, \text{ for some k} \in \mathbb{Z}^+. & (15) \end{aligned}$$

The presence of the $2\pi k/\omega$ term in (11), (13), and (15) is required to resolve the fact that $\mathcal{A}_i$ may have encircled the origin one or more times (during the the transfer) with our convention that $\theta_{i,d} \in [0, 2\pi)$. As such, $t_{i,1}$ and $t_{i,2}$ may be determined by solving (14) and (15) for the smallest $k \in \mathbb{Z}^+$ such that

$$\begin{aligned} t_{i,1} &\leq (1/\alpha) \log\left(\frac{r_{\text{disp}}}{r_o}\right) \text{ and} & (16)\\ \theta_{i,o} &\leq \omega t_{i,1} + \theta_{i,d} + 2\pi k. & (17) \end{aligned}$$

The condition in (16) guarantees $\mathcal{A}_i$ make its way inside the dispersion circle, while the condition in (17) addresses the aforementioned multiple encirclements scenario.

To summarize, by solving (14) and (15), subject to (16) and (17), we can calculate the activation time and deactivation time of each agent. It is important to note that inter-agent coordination is critical to the success of this algorithm; agents must schedule their motion according to a common clock in order to ensure the SPIRAL-IN and SPIRAL-OUT phases end and begin, respectively, at time zero. In the next section, we show the algorithm is free of conflicts.

C. Safety of the SPIRAL-ALGORITHM

Concerning the safety of the SPIRAL ALGORITHM, we have the following result:

Theorem 1 *Under the relative-velocity conflict model, the* SPIRAL ALGORITHM *is free of conflicts provided* $\kappa \leq 1/\sqrt{\alpha^2 + \omega^2}$. $\diamond$

Proof: We begin by noting that during the SPIRAL-IN phase, all of the active aircraft move according to (4) and (5). During the SPIRAL-OUT phase, all of the active aircraft move according to (8) and (9). Also, the SPIRAL-IN and SPIRAL-OUT phases span disjoint intervals of time. Therefore, at any time, all of the active agents move by following a common flow field associated with either the SPIRAL-IN or SPIRAL-OUT phase.

Now consider any two aircraft, say $\mathcal{A}_i$ and $\mathcal{A}_j$, that are active at time t during the SPIRAL-IN phase, as shown in Figure 2. For convenience, we suppress the dependence on time in the relations that follow. From Figure 2, the square of the distance separating $\mathcal{A}_i$ and $\mathcal{A}_j$ is

$$\left|q_i - q_j\right|^2 = r_i^2 + r_j^2 - 2 r_i r_j \cos(\theta_i - \theta_j). \quad (18)$$

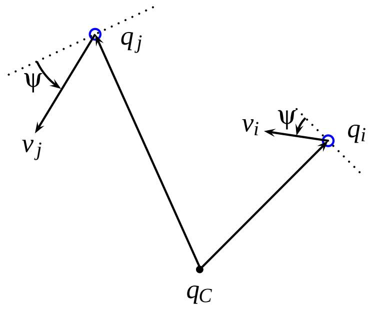

Fig. 2. Relevant geometry of $\mathcal{A}_i$ and $\mathcal{A}_j$ in $\mathcal{W}$.

Now consider the expression $\left|v_i - v_j\right|$. The speed of $\mathcal{A}_i$ is given by

$$\begin{aligned} |v_i| &= \sqrt{\left(\frac{\partial r_i}{\partial t}\right)^2 + \left(r_i \frac{\partial \theta_i}{\partial t}\right)^2} \\ &= r_i \sqrt{\alpha^2 + \omega^2}. \end{aligned}$$

It follows, again from Figure 2, that the square of the difference between the velocities of $\mathcal{A}_i$ and $\mathcal{A}_j$ is

$$\left|v_i - v_j\right|^2 = \left(\alpha^2 + \omega^2\right)\left(r_i^2 + r_j^2 - 2r_i r_j \cos\left(\theta_i - \theta_j\right)\right). \quad (19)$$

Combining (18) and (19) gives

$$\left|q_i - q_j\right| = \frac{1}{\sqrt{\alpha^2 + \omega^2}} \left|v_i - v_j\right|.$$

Comparing the above expression with the relative-velocity conflict model in (2), we see the transfer is free of conflicts provided α and ω are chosen to satisfy

$$\sqrt{\alpha^2 + \omega^2} \leq \frac{1}{\kappa}. \quad (20)$$

A similar argument applies to any two agents that are active during the SPIRAL-OUT phase. Therefore, for appropriately chosen α and ω, we conclude the SPIRAL-TRANSFER algorithm is free of conflicts. ■

We remark that, in the above, the selection of $\sqrt{\alpha^2 + \omega^2}$, and therefore α, can be done independent of n. In the next section, we use this fact to develop an upper bound on the transfer time of the SPIRAL ALGORITHM.

D. An Asymptotic Upper Bound on $T_{SP}(n)$

In this subsection, we derive an upper bound on $T_{SP}(n)$, the time required for the SPIRAL ALGORITHM to transfer n agents, in the case when n is large. We have the following result:

Theorem 2 *For any distribution of origin and destination points, $T_{SP}(n) = O(\log n)$.* $\diamond$

Proof: To investigate the transfer time for large n, note that since the SPIRAL-IN and SPIRAL-OUT phases end and begin, respectively, at $t = 0$, the transfer time of the SPIRAL-ALGORITHM can be expressed as

$$T_{SP}(n) = \max_i \; t_{i,1} + \max_i \; t_{i,2}. \quad (21)$$

We begin by developing an upper bound for the leftmost term on the right-hand side of (21). Let $i = \arg\max_i t_{i,1}$ and note that the term in question represents the duration of the SPIRAL-IN phase.

When n is large, the dispersion satisfies $r_{\text{disp}} = \Omega(1/\sqrt{n})$. It follows there exists $n_o \in \mathbb{Z}^+$ and finite $c > 0$ such that for all $n \geq n_o$ the dispersion satisfies $r_{\text{disp}} \geq c/\sqrt{n}$.

For a worst case distribution of origin and destination points, $\mathcal{A}_i$ must travel from the workspace boundary ($r_i(-t_{i,1}) = R$) to reach the edge of the dispersion circle and, subsequently, rotate an additional 2π radians to reach the first point that will take it to D_i during the SPIRAL-OUT phase. From the expression in (6), we can bound $t_{i,1}$ by

$$\begin{aligned} t_{i,1} &\leq \frac{1}{\alpha} \log\left(\frac{R\sqrt{n}}{c}\right) + \frac{2\pi}{\omega} \\ &= O(\log n). \end{aligned}$$

In the above, we have made use of a point that was noted earlier, namely, that α can be chosen to provide safety independent of n. A similar worst-case scenario applies in the SPIRAL-OUT phase and leads to the bound $t_{i,2} = O(\log n)$. Combining this result with the bound on $t_{i,1}$ gives the upper bound $T_{SP}(n) = O(\log n)$. ■

For agents capable of communicating their intentions with other agents, Theorem 2 indicates the adoption of the relative velocity conflict model allows for a dramatic reduction in the transfer time. The improvement in transfer time can then be interpreted as a performance gain associated with systems in which inter-agent communication allows for coordinated trajectory planning.

E. An Asymptotic Lower Bound on $T^*(n)$

In this subsection, we consider a lower bound on $T^*(n)$. We begin by considering the transfer time complexity for a favorable distribution of origin and destination points.

Proposition 1 *Under the relative-velocity conflict model, there exists a distribution of origin and destination points for which $T^*(n) = \Omega(1)$.* $\diamond$

Proof: Consider an arrangement of origin and destination points for which $\angle(D_i - O_i) = \angle(D_j - O_j)$ for all $i, j \in 1, \ldots, n$. If all agents travel with finite speed V_{max} along straight-line paths from their origin to their destination point, then $|v_i(t) - v_j(t)| = 0$ for all i, j, ensuring there are no conflicts. Furthermore, $T^*(n) \leq c_1\sqrt{A}/V_{max}$, where c_1 is a finite constant dependent on the geometry of the workspace, indicating the transfer time is independent of n. ■

The following result considers a lower bound on $T^*(n)$ for a more challenging distribution of origin and destination points.

Theorem 3 *Under the relative-velocity conflict model, there exists a distribution of origin and destination points for which $T^*(n) = \Omega(\log n)$.* $\diamond$

Proof: Consider an arrangement of origin and destination points that has the following properties:

- all source points are distributed inside a small region such that $|S_i - S_j| \leq c_1/n^k$ for all i, j, where c_1 is a constant dependent on the geometry of $\mathcal{W}$. An appropriate value for k will be specified later.
- the destination points are distributed in a separate region of $\mathcal{W}$ such that $|D_i - D_j| \geq 2c_2/\sqrt{n}$ for all $i \neq j$, where c_2 is a constant dependent on the geometry of $\mathcal{W}$.

It is noted that the distribution of origin and destination points is well-defined even for large n provided k is sufficiently large. Now assume there is a time, say $t = 0$ without loss of generality, at which a constant fraction of the agents (i.e., $m = \alpha n$, $0 < \alpha \leq 1$) are active and inside the small region. Number these agents as $\mathcal{A}_1$ through $\mathcal{A}_m$ based on the order in which they reach their destination points, with t_i the time at which $\mathcal{A}_i$ reaches D_i. Then $[0, t_m)$ may be divided into the intervals $[0,t_1), [t_1,t_2), \ldots, [t_{m-1}, t_m)$, where $[t_{i-1}, t_i)$ is the time interval in which only agents $\mathcal{A}_i, \ldots, \mathcal{A}_m$ are active.

It can be shown that the safety condition in (2) implies

$$|q_i(t_2) - q_j(t_2)| \leq e^{(t_2 - t_1)/\kappa} |q_i(t_1) - q_j(t_1)|, \qquad (22)$$

for all i, j, t_1, and $t_2 : t_1 \leq t_2 \leq \min(t_i, t_j)$. Noting that $T^*(n)$ is the minimum time required to transfer all agents, we proceed by considering two cases.

CASE 1: There exists a pair of agents, say $\mathcal{A}_i$ and $\mathcal{A}_j$, such that $|q_i(t) - q_j(t)| \geq c_2/\sqrt{n}$ for $t \in [0, \min(t_i, t_j))$. Then since $|S_i - S_j| \leq c_1/n^k$, it follows that $T^* \geq \log\left((c_2/c_1)\, n^{k-0.5}\right)$ and for $k > 0.5$, we have $T^*(n) = \Omega(\log n)$.

Now assume CASE 1 does not occur. It must be that we have the following case:

CASE 2: The agents, $\mathcal{A}_1, \ldots, \mathcal{A}_m$ travel as a "pack", between the destination points D_1 through D_m, with any two active agents always separated by less than $c_2/\sqrt{n}$. However, since $|D_i - D_j| \geq 2c_2/\sqrt{n}$ for all $i \neq j$, the transfer of agents is done one agent at a time, implying the duration of each of the intervals $[0,t_1), [t_1,t_2), \ldots, [t_{m-1}, t_m)$ is bounded from below by $c_2/(V_{max}\sqrt{n})$. The total transfer time is then bounded from below by $(\alpha c_2 \sqrt{n})/V_{max}$. However, since $T_{SP}(n) = O(\log n)$, we are in contradiction with our definition of $T^*(n)$ as the minimum transfer time for a specific arrangement of source and destination points. Therefore, CASE 1 must always hold and $T^* = \Omega(\log n)$. ■

This result implies that for certain arrangements of origin and destination points the SPIRAL ALGORITHM is to within a constant factor of the optimal transfer policy. The following theorem summarizes this result formally.

Theorem 4 *Under the relative-velocity conflict model, for any distribution of origin and destination points,* $T^*(n) = \Theta(\log n)$. ◇

Proof: Given $T^*(n) \leq T_{SP}(n)$, the result follows directly from the upper bound in Theorem 2 and the lower bound in Theorem 3. ■

It is noted that while $T^*(n) = \Omega(1)$ for certain arrangements and $T^*(n) = \Omega(\log n)$ for others, it remains to categorize the transfer time for a stochastic distribution of origin and destination points. This is the subject of ongoing work.

V. Simulation Results

This section demonstrates the functionality of the SPIRAL ALGORITHM for a transfer of $n = 20$ agents. Figure 3 provides six snapshots that have been evenly spaced throughout the time spanning the activation of the first agent to the deactivation of the last agent, inclusively. For each of the n origin/destination points, the radial and angular component were selected by sampling uniformly over $[0.2, 0.75]$ and $[0.2\pi)$, respectively. The remaining parameters used are $\alpha = 0.3$ and $\omega = 2$, so that the transfer is free of conflicts for $\kappa \leq 0.4975$. The parameter values used were selected so the dispersion circle is sufficiently large and the trajectories suitably distinguishable from one another to ensure readability.

VI. A Revised Conflict Model

Although we have argued that the relative-velocity conflict model is appropriate for many applications, in some cases, it is prudent to consider both the relative velocity and relative position of agents when specifying a minimum separation. In this section, we show that under one such conflict model, the spiral algorithm remains free of conflicts.

Definition 3 *A transfer policy is free of conflicts, in the spatial, relative-velocity sense, if for every pair of active agents,* $\mathcal{A}_i$, $\mathcal{A}_j$,

$$|q_i - q_j| \geq \kappa \frac{(q_j - q_i)^T}{|q_j - q_i|} (v_i - v_j), \qquad (23)$$

$\forall i, j \in 1, \ldots, n$, $\forall t$, *where* $\kappa > 0$ *is a constant.* ◇

Under this conflict model, the minimum separation between $\mathcal{A}_i$ and $\mathcal{A}_j$ is mandated to be positive only if $\mathcal{A}_i$ and $\mathcal{A}_j$ are heading "toward" each other. We have the following results concerning the safety of the SPIRAL ALGORITHM:

Proposition 2 *Under the spatial, relative-velocity conflict model, the* SPIRAL ALGORITHM *is free of conflicts for* $\kappa\alpha \leq 1$. ◇

Proof: To begin, note the safety condition can be rewritten as

$$|q_i(t) - q_j(t)|^2 \geq \kappa (q_j - q_i)^T (v_i(t) - v_j(t)). \qquad (24)$$

Consider agents $\mathcal{A}_i$ and $\mathcal{A}_j$ that are active during the SPIRAL-IN phase. Noting $\dot{r}_i = -\alpha r_i$ and $\dot{\theta}_i = \omega$, expanding the right-hand side of (24) gives

$$\begin{aligned}
&\kappa (q_j - q_i)^T (v_i - v_j) \\
&= -\kappa (r_i \cos\theta_i - r_j \cos\theta_j, r_i \sin\theta_i - r_j \sin\theta_j) \times \\
&\quad (\dot{r}_i \cos\theta_i - \omega r_i \sin\theta_i - \dot{r}_j \cos\theta_j + \omega r_j \sin\theta_j, \dot{r}_i \sin\theta_i + \\
&\quad \omega r_i \cos\theta_i - \dot{r}_j \sin\theta_j - \omega r_j \cos\theta_j)^T \\
&= -\kappa (r_i \dot{r}_i - 2\alpha r_i r_j \cos\theta_i \theta_j + r_j \dot{r}_j - 2 r_j r_j \alpha \sin\theta_i \sin\theta_j) \\
&= \kappa \left(\alpha \left(r_i^2 + r_j^2\right) - 2\alpha r_i r_j \cos(\theta_i - \theta_j)\right) \\
&= \kappa\alpha |q_i - q_j|^2.
\end{aligned}$$

Therefore, the transfer is safe for $\kappa\alpha \leq 1$. An similar analysis applies to agents that are active during the SPIRAL-OUT phase; in this case, $\dot{r}_i = \alpha r_i$, implying $\kappa (q_j - q_i)^T (v_i - v_j) = -\kappa\alpha |q_i - q_j|^2$, indicating motion is safe for any $\alpha, \omega, \kappa > 0$.

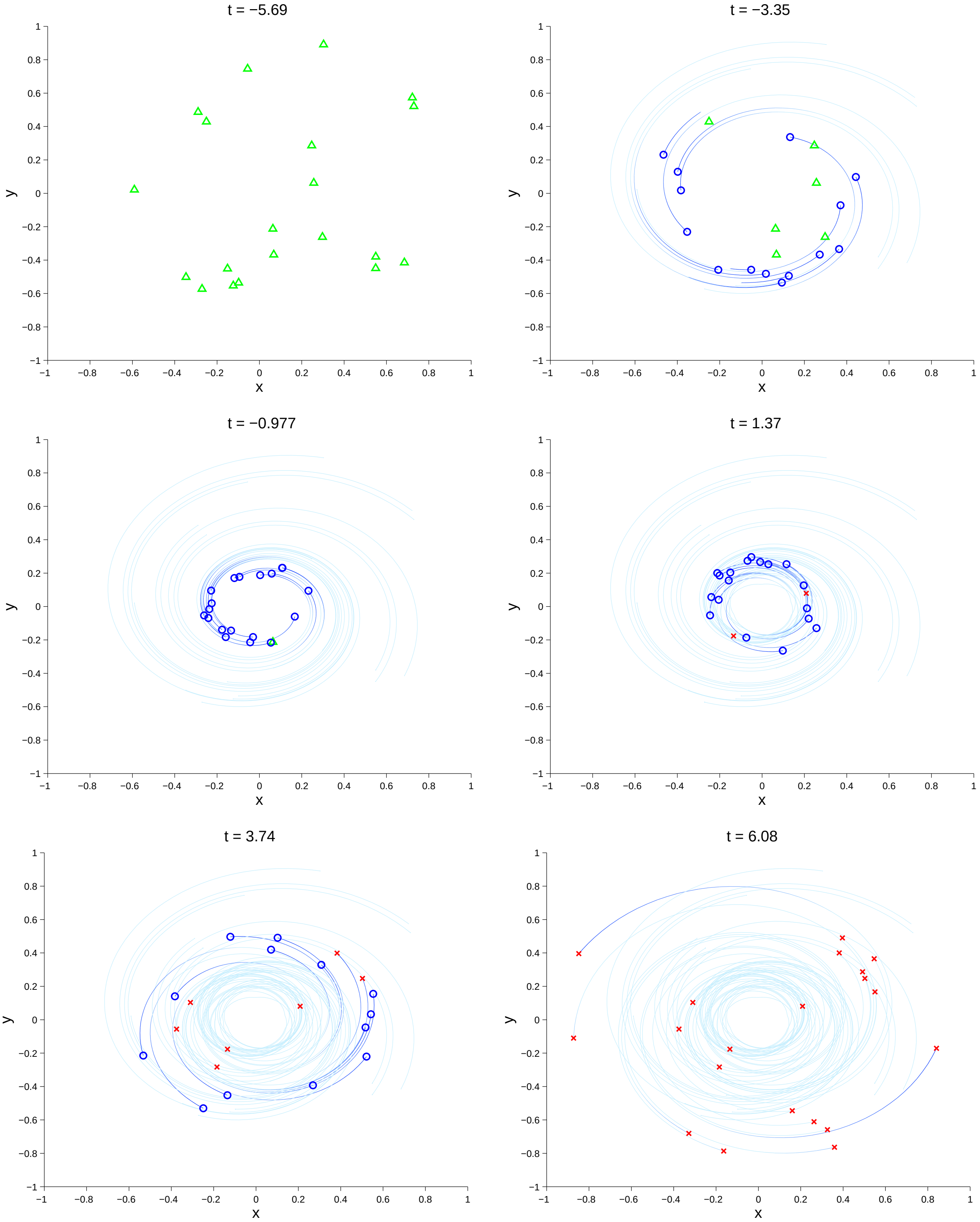

Fig. 3. Progression of agent trajectories during transfer. Green triangles denote agents that have not left their origin point. Red crosses denote agents that have reached their destination point. Blue circles denote agents in the process of being transferred. The time instant corresponding to each snapshot is shown at the top center of each plot. The blue lines represent agent trajectories. To provide a sense for how fast the agents move, the last 0.3s of each agent's trajectory is shown in varying shades of blue; the darker the shade, the more recently the point was visited.

Therefore, we are limited by safety considerations imposed during the SPIRAL-IN phase and we conclude the spiral transfer is safe for $\kappa\alpha \leq 1$. ■

Given our interpretation of (23), the above result makes sense; namely, since active agents never travel “toward” one another during the SPIAL-OUT phase, there is no issue with safety during this phase. It is noted that the safety condition that must be satisfied during the SPIAL-IN phase does not depend on ω, and we are free to choose this quantity to fulfill other criteria.

VII. Conclusions and Future Directions

We introduced a new approach to model conflicts among aircraft operating in a shared airspace. An algorithm was presented that transfers aircraft between origin and destination points, is free of conflicts, and whose transfer time scales logarithmically in the number of aircraft was presented. The ideas reported have a number of natural extensions. Presently, we are investigating transfer schemes in which the velocity of aircraft deviates only minimally from a constant value (rather than scaling linearly with distance as in our algorithm), as this is more representative of the velocity profile that is desired during flight. We are also interested in a dynamic version of the transfer problem in which requests by agents to move between points in the workspace arrive stochastically on an ongoing basis. Finally, it remains to quantify how closures in portions of the workspace (due, for example, to inclement weather) and navigational uncertainty have on the transfer time in super-density traffic environments.

References

[1] NASA, “Nasa and the next generation air transportation system,” NASA, Technical Report, June 2007.

[2] D. V. Dimarogonas and K. J. Kyriakopoulos, “Decentralized navigation functions for multiple robotic agents with limited sensing capabilities,” *Journal of Intelligent and Robotic Systems*, vol. 48, no. 3, pp. 411–433, March 2007.

[3] G. P. Roussos, D. V. Dimarogonas, and K. J. Kyriakopoulos, “Distributed 3D navigation and collision avoidance for nonholonomic aircraft-like vehicles,” in *Proceedings of the 2009 European Control Conference*, Budapest, Hungary, 2009.

[4] L. Pallottino, V. G. Scordio, A. Bicchi, and E. Frazzoli, “Decentralized cooperative policy for conflict resolution in multivehicle systems,” *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1170–1183, December 2007.

[5] L. Pallottino, E. M. Feron, and A. Bicchi, “Conflict resolution problems for air traffic management systems solved with mixed integer programming,” *IEEE Transactions on Intelligent Systems*, vol. 3, no. 1, pp. 3–11, March 2002.

[6] S. Mastellone, D. Stipanovic, C. Graunke, K. Intlekofer, and M. Spong, “Formation control and collision avoidance for multi-agent nonholonomic systems: Theory and experiments,” *International Journal of Robotics Research*, vol. 27, no. 1, pp. 107–126, 2008.

[7] H. Tanner and A. Kumar, “Towards decentralization of multi-robot navigation functions,” *2005 IEEE International Conference on Robotics and Automation*, pp. 4143–4148, 2005.

[8] H. Yamaguchi and J. W. Burdick, “Asymptotic stabilization of multiple nonholonomic mobile robots forming group formations,” *1998 IEEE International Conference on Robotics and Automation*, 1998.

[9] C. Tomlin, G. Pappas, and S. Sastry, “Conflict resolution for air traffic management: A study in multiagent hybrid systems,” *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 509–521, 1998.

[10] V. Sharma, M. Savchenko, E. Frazzoli, and P. Voulgaris, “Transfer time complexity of conflict-free vehicle routing with no communications,” *The International Journal of Robotics Research*, vol. 26, no. 3, pp. 255–271, 2007.

[11] S. Martinez, F. Bullo, J. Cortés, and E. Frazzoli, “On synchronous robotic networks - part i: Models, tasks and complexity notions,” *IEEE Transactions on Automatic Control*, vol. 52, pp. 1835–1841, 2007.

[12] S. LaValle, *Planning Algorithms*. New York, NY: Cambridge University Press, 2007.