2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

A Reactive Inverse PN Algorithm for Collision Avoidance among Multiple Unmanned Aerial Vehicles

Joel George and Debasish Ghose

***Abstract*— While performing a mission, multiple Unmanned Aerial Vehicles (UAVs) need to avoid each other to prevent collisions among them. In this paper, we design a collision avoidance algorithm to resolve the conflict among UAVs that are on a collision course while flying to their respective destinations. The collision avoidance algorithm consist of each UAV that is on a collision course reactively executing a maneuver that will, as in 'inverse' Proportional Navigation (PN), increase Line of Sight (LOS) rate between them, resulting in a 'pulling out' of collision course. The algorithm is tested for high density traffic scenarios as well as for robustness in the presence of noise.**

I. INTRODUCTION & PROBLEM STATEMENT

Research in Unmanned Aerial Vehicles (UAVs) is growing at a fast pace due to various capabilities that they offer for military and civilian sectors. The increasing popularity of UAVs can be attributed to, among others, the reduced cost, portability and absence to human risk. The multiple UAV missions give rise to many emergent subproblems that need to be tackled effectively. For any mission involving multiple UAVs, a common subproblem is that the UAVs may collide with each other. The problem assumes great importance in a high density multiple UAV traffic scenario. In this paper, we design a collision avoidance algorithm to resolve conflict among UAVs that are on collision course in a highly cluttered environment.

We consider the following problem in this paper. We assume that several UAVs are flying from different bases to their respective destinations. These UAVs need to avoid mid-air collision with other UAVs on their path. For safety reasons, 'miss distance' or the minimum separation between any two UAVs at any time of flight is desirable to be greater than a specified value. This has to be achieved preferably in a decentralized manner. The collision avoidance maneuvers need to be realistic and efficient in the sense that the deviation of a UAV from its nominal path owing to collision avoidance maneuver should be minimal so as to minimize the late arrival at its destination. The problem is to find an algorithm that, when executed by every UAV, results in no collisions and minimum number of 'near misses'. Although it is desirable to have zero near misses, it might be impossible to achieve this in high density air traffic scenarios like the ones considered in this paper. Moreover, in the case of UAVs, where there is no human risk, such a high level of safety requirement may not be necessary. So we emphasize on reducing the near misses while allowing some

Joel George and Debasish Ghose are with the Guidance, Control, and Decision Systems Laboratory, Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560012, India, {joel, dghose}@aero.iisc.ernet.in

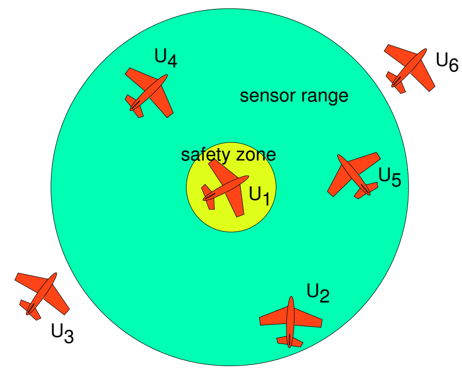


Fig. 1. An example situation requiring collision avoidance

in trade-off for efficiency. Each UAV needs to achieve the above objective with the limited information of positions and velocities of only the neighboring UAVs that are within its sensor range. An example scenario of the problem that we tackle is depicted in Fig. 1 where mid-air encounter of 6 UAVs during flight to their respective destinations is shown. In the figure, the inner and the outer discs around UAV U_1 show the desired safety zone around the UAV and its sensor range. The same applies to all other UAVs.

We assume that a UAV has limited sensor range and knows exactly the positions and velocities of all UAVs within its sensor range. For a group of cooperating UAVs, if the communication is possible, this information can be acquired through sharing. We consider UAVs with a kinetic constraint of minimum radius of turn. We assume that all UAVs fly at a same constant speed and have same minimum radius of turn. Thus, in this paper, we are dealing with a homogeneous group of UAVs although the algorithm we develop is easily extendable to a heterogeneous group. The kinematics of the UAV U_i with position (x_i, y_i), velocity V, and heading ψ_i is

$$\begin{aligned} \dot{x}_i &= V\cos(\psi_i) \\ \dot{y}_i &= V\sin(\psi_i) \\ \dot{\psi}_i &= u_i \end{aligned} \qquad (1)$$

where u_i is the commanded angular velocity issued by the collision avoidance guidance algorithm. If R_{min} is the minimum radius of turn, then $|u_i| \leq V/R_{min}$.

There has been some research over the last decade on aircraft conflict and collision avoidance both from the multiple UAV and the air traffic control points of view. Most of the algorithms developed for air traffic management are

978-1-4244-4524-0/09/$25.00 ©2009 AACC

those that guarantee safe trajectories in a very low density traffic with a few aircraft ([1], [2], [3]). In literature, we find protocol based conflict resolution algorithms ([4], [5]) that require instantaneous changes in position, velocity and/or heading of aircraft which are unrealistic. Also, these are not applicable in highly dynamic environments considered in our paper.

There has also been some work in the area of collision avoidance where size and shape of objects are explicitly taken into account ([6], [7], [8]). These algorithms need the *a priori* knowledge of the trajectories of obstacles.

Archibald *et al.* [9] use satisficing game theory to address the problem of collision avoidance in multiple UAVs. They do simulations involving test cases with high density air traffic scenarios to test their algorithm. This algorithm, apart from being computationally intensive, requires constant communication between neighboring UAVs which may not always be possible. We call this algorithm Satisficing Game Theory based Algorithm (SGTA) for collision avoidance and use it for comparative studies with the algorithm that we develop in our paper.

A broad overview of our solution for the multiple UAV collision avoidance while the UAVs fly from their bases to respective destinations is as follows. When a UAV envisages a collision with another UAV, it 'reacts' by switching to a collision avoidance mode. The collision avoidance mode comprises of a maneuver of doing a turn until the projected trajectories of UAVs are safe with a minimum required separation between them. The turn taken is one which ensures an increase in the LOS rate between the concerned UAVs. The collision avoidance maneuver would have taken the UAV out of its nominal path. We require that the arrival of a UAV at its destination has to be as quick as possible. The UAV therefore takes a Dubins path [10], which is a minimum time path, from its current location to the destination.

The organization of the rest of the paper is as follows. In Section II, we develop the algorithm for collision avoidance. We analyze the performance of this algorithm through various test cases and discuss the results in Section III. Section IV gives conclusions.

II. THE ALGORITHM

Proportional Navigation (PN) guidance law [11] is very robust and elegant, and its variants are therefore most widely used for missile guidance. The PN guidance law applies a lateral acceleration to the missile so as to nullify the rate of rotation of Line of Sight (LOS) or the LOS rate between the missile and the target, and brings the missile into a collision course with the target. As far as collision avoidance is concerned, a lateral acceleration which takes a vehicle away from the collision course is what is desirable. We propose an algorithm that employs a kind of 'inverse' PN for collision avoidance. The algorithm consist of applying lateral accelerations to increase the LOS rate between UAVs that are on a collision course.

There are a few papers in the literature that have used PN based algorithms for collision and obstacle avoidance ([12], [13], [14]). These algorithms are usually not suitable for large number of aircraft. Our proposed algorithm resembles the work in above papers, extended to multiple UAV case by considering pairs of UAVs with the highest chance of predicted collision.

A. Collision prediction

The first step toward employing such a collision avoidance rule for a UAV is to calculate the envisaged miss distance or the Zero Effort Miss (ZEM) with all other UAVs which are within its sensor range. We assume that every UAV has the capability to locate other UAVs in its sensor range and measure their velocities exactly.

In a $2D$ engagement scenario, consider two UAVs, U_1 and U_2. Let the initial position of U_1 be $\mathbf{p_1} = (x_1, y_1)$ and that of U_2 be $\mathbf{p_2} = (x_2, y_2)$. The UAVs have velocities of magnitude V. Let $\mathbf{d_1} = (l_1, m_1)$ and $\mathbf{d_2} = (l_2, m_2)$ be direction cosines of the respective headings of U_1 and U_2. At any time t, if the separation between the two UAVs is $\mathcal{S}$, then

$$\begin{aligned}\mathcal{S}^2 =&(\mathbf{p_1}-\mathbf{p_2})\cdot(\mathbf{p_1}-\mathbf{p_2}) + 2(\mathbf{p_1}-\mathbf{p_2})\cdot(\mathbf{d_1}-\mathbf{d_2})(Vt)\\ &+ (\mathbf{d_1}-\mathbf{d_2})\cdot(\mathbf{d_1}-\mathbf{d_2})(Vt)^2. \end{aligned} \tag{2}$$

Solving for t from $\frac{d\mathcal{S}}{dt} = 0$ will give the time at which a minimum separation between UAVs will occur; we call this the time-to-go (t_{go}) which is given as

$$t_{\text{go}} = -\frac{(\mathbf{p_1}-\mathbf{p_2})\cdot(\mathbf{d_1}-\mathbf{d_2})}{V(\mathbf{d_1}-\mathbf{d_2})\cdot(\mathbf{d_1}-\mathbf{d_2})}. \tag{3}$$

If $t_{\text{go}} > 0$, then there exists a time at which closest approach between two UAVs occur. The t_{go} calculated as above can take negative values which means that the vehicles are on a diverging path (that is, their paths extrapolated backwards lead to a minimum distance between them and thus the negative t_{go}). Substituting t_{go} into the expression for $\mathcal{S}$ will give the predicted miss distance or ZEM which is the distance of closest approach. We do not give the explicit analytical expression for this here for sake of brevity. If the obtained miss distance is less than the desired separation, then a collision avoidance maneuver has to be performed.

A UAV prepares a list of UAVs in its neighborhood with which its miss distance is less than the minimum required separation. From this list, the UAV chooses that UAV with which it has minimum positive t_{go}, which means that they are on a collision course, and does a collision avoidance maneuver.

B. Collision avoidance maneuver

The collision avoidance maneuver that we propose involve UAVs pulling out of the collision course. To pull out of a collision course, a UAV has to apply a lateral acceleration in such a way so as to increase the rate of rotation of the LOS connecting the two UAVs. This is illustrated in Fig. 2. Let r be the LOS separation. If $\theta_2 > \theta_1$ (refer Fig. 2), then the LOS rate is

$$\dot{\theta} = \frac{V\sin\theta_2 - V\sin\theta_1}{r}. \tag{4}$$

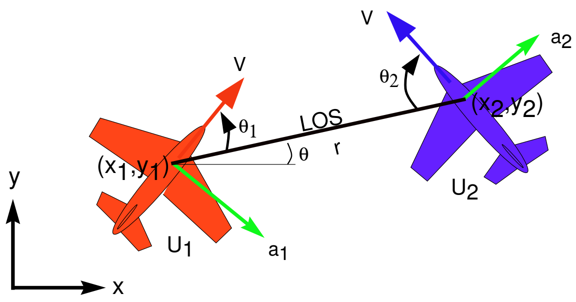


Fig. 2. Collision avoidance rule for a $2D$ engagement

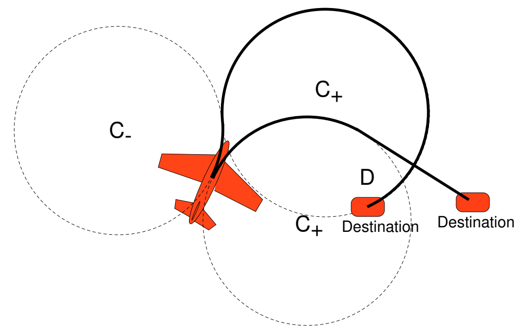


Fig. 3. Dubins paths to destination. Two possible scenarios where the destination is inside and outside the circle ($\mathcal{C}_+$) of minimum radius of turn (case of free terminal constraint).

To increase the LOS rate in this scenario ($\theta_2 > \theta_1$), we apply accelerations a_1 and a_2, which delivers a turn radius

$$R = R_{min} \exp(\lambda \times \text{ZEM}/R_{des}), \tag{5}$$

to U_1 and U_2, respectively, as shown in the figure. Here, the desired separation R_{des} and λ are tuning parameters for the algorithm, and R_{min} is the minimum radius of turn. The magnitude of applied acceleration is V^2/R and the corresponding turn rate obtained is V/R. Accelerations are applied in directions opposite to that shown in Fig. 2 for each UAV, when $\theta_2 < \theta_1$. For a predicted zero ZEM, we expect a UAV to do a tightest turn corresponding to minimum radius of turn. Whereas, for a higher predicted minimum separation, as there is a lower risk of collision it is desirable that the radius of turn of the collision avoidance maneuver of a UAV be considerably lesser to minimize deviation from the nominal path. One way of achieving this is by using an exponential function as above to command a demanded radius of turn R.

If a UAV is not in collision course with any other UAV, then it flies to its destination. It is preferable to do this in minimum time. This can be achieved if the UAV takes a Dubins path from the current location to the destination. This is explained in the following subsection.

C. Dubins path to destination

Given initial position and departure angle, and final position and arrival angle, Dubins curve or path gives the trajectory with minimum path length under the additional constraint of a maximum allowed curvature of the trajectory [10]. In the multiple UAV collision avoidance scenario, once a UAV has deviated from its nominal path after a collision avoidance maneuver, it is desirable to reach its destination in minimum time. Following a Dubins path to destination will help in achieving this because, with a constant speed assumption, a minimum length path is also a minimum time path. For arrival of a UAV at destination in minimum time, we require a special case of Dubins curve where there is no arrival angle constraint. The maximum curvature constraint takes care of the minimum radius of turn of UAV. We propose a switching algorithm that implements this special case of Dubins path.

Let $\mathcal{D}$ be the destination, $\mathcal{C}_+$ a circle, whose radius is equal to the radius of minimum turn, tangential to the velocity vector and such that following $\mathcal{C}_+$ will take UAV toward the destination, and $\mathcal{C}_-$ a similar circle such that following $\mathcal{C}_-$ will take a UAV away from the destination (refer Fig. 3). Then, a UAV employing Algorithm 1 will follow a Dubins path to the target. Fig. 3 shows the possible cases of Dubins path to a destination when there is no terminal angle constraint.

Algorithm 1 Algorithm to implement Dubins path with free terminal constraints

if UAV heading points to $\mathcal{D}$ **then**
  Head straight.
else if $\mathcal{C}_+$ encircles $\mathcal{D}$ ($\mathcal{D}$ strictly within $\mathcal{C}_+$) **then**
  Take a path along $\mathcal{C}_-$.
else
  Take a path along $\mathcal{C}_+$.
end if

We implement this algorithm with a slight modification. During simulation, a UAV follows Algorithm 1 till the difference between its desired heading, in which case the UAV's velocity vector points directly to its destination, and actual heading is less than a small quantity $\epsilon > 0$. Beyond that, the UAV uses a proportional controller of the form $\dot{\psi} = k(\psi_d - \psi)$, where ψ is the heading angle and ψ_d is the desired heading of the UAV. This avoids chattering that might occur otherwise due to switching.

Putting all this together, we give an algorithmic description of the collision avoidance algorithm, that each UAV flying to its destination implements at every time step in a multi-UAV scenario, in Algorithm 2. Since the UAV reacts to the situation of a violated desired separation and employs a maneuver in the spirit of inverse PN, we call this Reactive Inverse PN Algorithm (RIPNA).

RIPNA handles the multiple UAV scenario by considering only that collision which is immediate to a UAV in terms of the minimum LOS rate. This approach results in a good collision avoidance algorithm with good performance as we will demonstrate through simulations. However, it is possible that by considering only the immediate collision and doing a collision avoidance maneuver may at times

Algorithm 2 Reactive Inverse PN Algorithm (RIPNA) for UAV U_i for collision avoidance

Find neighbors $\mathcal{N}_i$ that are other UAVs within the sensor range R_{sen} of U_i.
Calculate ZEM between U_i and each UAV $U_j \in \mathcal{N}_i$.
Find $\mathcal{N}_i^*$, the neighbors with which U_i has ZEM $< R_{des}$.
if $\mathcal{N}_i^*$ is not empty **then**
  Choose $U_{j*} \in \mathcal{N}_i^*$ with which U_i has least t_{go}.
  Turn with radius of turn R to increase LOS rate between U_i and U_{j*}.
else
  Take Dubins path to destination.
end if

lead the UAVs to other conflicts which are too close to avoid. Also, there is a chance of chattering when a UAV faces two nearly similar conflicts – avoiding one will lead to other and vice versa. This usually occur when engagement geometries have some sort of symmetry. In such cases where engagements are close to symmetric, it is desirable to follow a protocol which will break the symmetry and then proceed to use RIPNA. We remark that RIPNA can easily handle heterogeneous group of UAVs with different velocities and radii of turn by using tuning parameters specific to individual UAVs instead of global ones. However, to keep the situations simple, we consider only a homogeneous group of aircraft for simulations in this paper.

III. SIMULATION RESULTS AND DISCUSSION

In order to evaluate the collision avoidance algorithm, we carry out simulations. For simulations, we use a test case similar to that in [9] which was originally designed for air traffic control problems. Thus the speed of vehicles used is typical of an aircraft and not of small UAVs. Nonetheless as this suffices to test the algorithm, as in [9], we assume that the UAVs travel at a speed of 500 miles per hour (733.33 ft/s) and that they have the maximum turn rate of 5° per second.

We use the same performance metrics that are used to measure the performance of collision avoidance algorithm in [9]. These are number of near misses and efficiency, and are explained below for completeness.

Near misses: From safety point of view, UAVs are required to keep a distance of at least 5 miles between each other. Any location of two UAVs within this desired separation results in a near miss. A good collision avoidance algorithm will ensure fewer near misses.

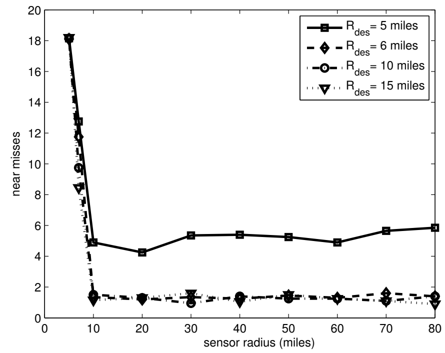
Efficiency: The UAVs are required to reach the destination in minimum possible time. The collision avoidance maneuver should be minimal so that UAVs arrive at their destinations not too late. If t_j^a is the actual flight time of j^{th} UAV and t_j^i is its ideal flight time (i.e., the time taken to reach the destination if the UAV were not to make any deviation for collision avoidance), then the efficiency for N UAVs is given as

$$\text{Efficiency} = \frac{1}{N}\sum_{j=1}^{N}\frac{t_j^i}{t_j^a}.$$

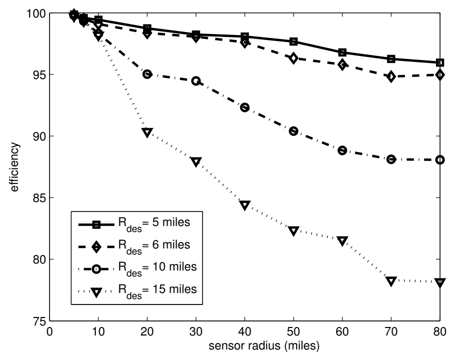
A. Test case of random flights

A test case for collision avoidance should be complex and at the same time scalable to test the high traffic density cases. We consider a test case given in [9]. The UAVs fly from random points on an outer circle, the destination being random points on an inner circle. The radii of outer and inner circles used for simulations in this paper are 120 and 100 miles, respectively. Each simulation is done for a specified aircraft density or the number of aircraft in the concerned airspace. When the simulation begins, a new aircraft is introduced at a random point on the outer circle at every 5 sec interval, which is assigned a random point on the inner circle as destination, till the required aircraft density is achieved. Whenever an aircraft reaches the destination, a new aircraft is added to keep the number of aircraft constant in the airspace. Once the airspace achieves required density, information regarding system efficiency and number of near misses is collected from simulation for a simulation time of 50 minutes. Since the test case involves random introduction of UAVs and random assignment of destination points, the simulation for a particular aircraft density is conducted 20 times and average values are taken. Since there is a randomness involved in this test case, we can expect that, if the simulations are carried out reasonably enough number of times, we would have encountered almost all the possible conflict geometries.

To achieve a good performance, right values have to be chosen for the tuning parameters of our algorithm. Towards this, we do extensive simulations with varying values of tuning parameters R_{des}, R_{sen} and λ, and observe the performance in terms of number of near misses and efficiency. One such study for a traffic density of 20 aircraft and a constant $\lambda = 0.5$ is given in Fig. 4. As seen from the figure, the choice of R_{des} and R_{sen} is a tradeoff between number of near misses and efficiency; more the efficiency, more the number of near misses. After similar studies for other aircraft densities, the values $R_{sen} = 10$ miles, $R_{des} = 6$ miles, $R_{min} = 5/\pi$ miles (equivalent to a maximum turn rate of 5° per sec) and $\lambda = 0.5$ were chosen for the tuning parameters. Fig 4(b) shows a decrease in efficiency with an increase in sensor radius. This is contrary to the intuition that more information should lead to better efficiency. Here, we remark that RIPNA is not an optimal algorithm that take complete advantage of all the available information, but a reactive one. For a UAV implementing RIPNA, higher sensor radius implies more neighbors. Then, it may so happen that in trying to avoid conflicts with this larger class of neighbors, a UAV may result in taking a roundabout path avoiding the whole neighbor set as a group while there may be a safe path through them. This behavior results in a poor efficiency. Restricting the sensor radius forces a UAV to ignore those neighbors which are far and take the risk of exploring a safe path through the neighbors. Also, we observe through



(a)



(b)

Fig. 4. Variation of near misses and efficiency with R_{sen} and R_{des} for a traffic density of 20 aircraft.

TABLE I

COMPARISON OF PERFORMANCE OF ALGORITHMS RIPNA AND SGTA: TEST CASE OF RANDOM FLIGHTS (AVERAGED OVER 20 RUNS)

Number of Aircraft	Near Misses		Efficiency	
	SGTA	RIPNA	SGTA	RIPNA
20	1.95	1.35	99.45	99.09
40	7.05	3.75	97.94	97.64
60	17.85	12.65	94.38	96.39

simulations that by a suitable choice of tuning parameters, we can adapt RIPNA to mimic a near optimal behavior.

The results for low density (20 aircraft), medium density (40 aircraft), and high density (60 aircraft) traffic are given in Table I. For the purpose of comparison, the values obtained by simulating the algorithm of Archibald *et al.* [9] are also given in the table. Archibald *et al.* [9] use satisficing game theory based approach for aircraft conflict resolution. They give two versions of the algorithm – full and simplified. We use the simplified algorithm as it is computationally much less demanding while the performance is almost same as that of the full version [9]. The simplified satisficing approach to aircraft conflict resolution which we call Satisficing Game Theory based Algorithm (SGTA). To compare the performances under similar test conditions, along with RIPNA, we implement SGTA using information available in [9]. As complete details of the implementation of SGTA is not available in [9], we handle the missing details the same way as we do it for RIPNA. For example, the same subroutines are used for both the algorithms to calculate predicted minimum separation, efficiency, number of near misses, and other required quantities. The results used for comparison are obtained from our implementation and not directly taken form the above mentioned reference. However, the results are similar in noise-free case. They do not consider the case with noisy measurements. Both the algorithms were executed and evaluated under same conditions for all test cases – for example, same sequence of random numbers were used in setting up the problem, and performances are measured in the same way. All the values in Table I are averages over 20 runs. As seen from Table I, the number of near misses are lesser for RIPNA when compared to SGTA, while the efficiencies are comparable for all the cases. A snapshot during one of the simulations with a traffic density of 40 aircraft is shown in Fig. 5.

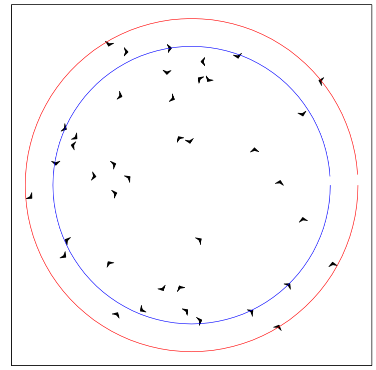


Fig. 5. A snapshot during simulation of random flights where UAVs fly from random points on outer to inner circle for a traffic density of 40 aircraft.

TABLE II

COMPARISON OF COMPUTATION TIMES OF RIPNA AND SGTA: TEST CASE OF RANDOM FLIGHTS (AVERAGED OVER 20 RUNS)

Number of Aircraft	Time Taken (sec)	
	SGTA	RIPNA
20	673	68
40	1711	200
60	3009	396

RIPNA has a great computational advantage over the algorithm in [9]. The average computational time taken in the simulations with various air traffic densities is presented in Table II. Clearly, RIPNA is about 8–10 times faster.

B. Simulations with noise

To study the robustness of the collision avoidance algorithm, we introduce noise in the presence of measurement of UAV positions and observe the degradation of performance of the algorithm. In previous simulations, we assumed that the positions of neighboring UAVs which are within the sensor range of a particular UAV are exactly known to it. We

TABLE III

COMPARISON OF PERFORMANCE OF RIPNA AND SGTA: TEST CASE OF RANDOM FLIGHTS WITH NOISE IN MEASUREMENT OF POSITION

Std. Dev. of Noise	Near Misses		Efficiency	
	SGTA	RIPNA	SGTA	RIPNA
0	1.95	1.35	99.45	99.09
0.1	8.65	1.35	99.58	98.99
0.2	12.50	1.55	99.79	99.08
0.3	14.10	1.95	99.44	99.02

TABLE IV

COMPARISON OF PERFORMANCE OF RIPNA AND SGTA: TEST CASE OF RANDOM FLIGHTS WITH NOISE IN MEASUREMENT OF HEADING

Std. Dev. of Noise	Near Misses		Efficiency	
	SGTA	RIPNA	SGTA	RIPNA
0	1.95	1.35	99.45	99.09
5	10.40	1.80	99.92	99.01
10	13.30	1.85	99.96	99.08
15	14.60	2.25	99.99	99.06

relax this assumption and hold that the x and y coordinates of other UAVs are measured with a noise which is normally distributed with a standard deviation of σ in miles. The performance of collision avoidance algorithms, both RIPNA and SGTA, for the case of random flights with an aircraft density of 20 for various values of σ is tabulated in Table III. From the table we observe that as the standard deviation of noise in position measurement increases the performance of the collision avoidance algorithm developed in this paper degrades gracefully when compared to the sudden loss of performance as observed for the algorithm of Archibald *et al.* [9].

We also study the effect of noise in the measurement of heading of neighboring aircraft on the performance of our collision avoidance algorithm. During simulation, we introduce noise of various standard deviations to actual heading of neighboring aircraft to make the measurement noisy. Collision avoidance algorithm is run with this noisy measurement. Performance results obtained are presented in Table IV where the standard deviations are in degrees. Even in this case we observe a very low degradation in performance of RIPNA in terms of number of near misses when compared to SGTA.

IV. CONCLUSIONS

In this paper we developed a reactive inverse PN algorithm to achieve collision avoidance among UAVs in a multi-UAV scenario while UAVs are flying to their respective destinations. This algorithm tries to achieve a collision avoidance by increasing the LOS rate between the UAVs on a collision course. This is inspired from and opposite to what is done in the popular PN in missile guidance. Through simulations, we demonstrated the performance of this collision avoidance algorithm. We also showed that the degradation of the performance of the algorithm is graceful in presence of noise which points to the robustness of the algorithm. The devised algorithm is decentralized as each UAV implements it with its limited information of surroundings gathered through its sensors, and is simple to implement as it contains the spirit of PN guidance law. The algorithm is computationally less demanding. However, there is still scope for a lot of improvement. A lack of efficiency can arise due to 'over doing' the collision avoidance maneuver because of the myopic behavior of UAVs implementing the present algorithm – only the most threatful UAV is considered. It can so happen that by the time one UAV is avoided, another one which was not 'seen' earlier is encountered. Algorithms which take into account all the UAVs in the sensor radius and even model those outside it are expected to perform better. Future work in this topic will be directed along these lines.

REFERENCES

[1] C. Tomlin, G.J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: a case study in multi-agent hybrid systems," *IEEE Transactions on Automatic Control,* vol. 43, no. 4, pp. 509–521, Apr. 1998.

[2] E. Frazzoli, L. Pallottino, V. Scordio, and A. Bicchi, "Decentralized coopperative conflict resolution for mutiple nonholonomic vehicles," *Proceedings of the AIAA Guidance, Navigation and Control Conference,* San Francisco, CA, Aug. 2005, Paper AIAA-2005-6048.

[3] A. Bicchi and L. Pallottino, "An optimal cooperative conflict resolution for air traffic management systems," *IEEE Transactions on Intelligent Transportation Systems,* vol. 1, no. 4, pp. 221–232, Dec. 2000.

[4] I. Hwang and C. Tomlin, "Protocol-based conflict resolution for finite information horizon," *American Control Conference,* Anchorage, AK, May 2002, vol. 1, pp. 748–753.

[5] Z-H. Mao, E. Feron, and K. Bilimoria, "Stability of intersecting aircraft flows under decentralized conflict avoidance rules," *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* Denver, CO, Aug. 2000, Paper AIAA-2000-4271.

[6] A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: a collision cone approach," *IEEE Transactions on Systems, Man and Cybernetics, Part A,* vol. 28, no. 5, pp. 562–574, Sep. 1998.

[7] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *International Journal of Robotics Research,* vol. 17, no. 7, pp. 760–772, 1998.

[8] Z. Shiller, F. Large, and S. Sekhavat, "Motion planning in dynamic environments: obstacles moving along arbitrary trajectories," *Proceedings of the IEEE International Conference on Robotics and Automation,* Seoul, Korea, May 2001, pp. 3716–3721.

[9] J.K. Archibald, J.C. Hill, N.A. Jepsen, W.C. Stirling, and R.L. Frost, "A satisficing approach to aircraft conflict resolution," *IEEE Transactions on System, Man, and Cybernetics – Part C: Applications and Reviews,* vol. 38, no. 4, pp. 510–521, Jul. 2008.

[10] L.E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics,* vol. 79, no. 3, pp. 497-516, 1957.

[11] S.N. Ghawghawe and D. Ghose, "Pure proportional navigation against time-varying target maneuvers." *IEEE Transactions on Aerospace and Electronic Systems,* vol. 32, no. 4, pp. 1336–1347, Oct. 1996.

[12] B. Ajith Kumar and D. Ghose, "A proportional navigation based collision avoidance/guidance strategy for low-altitude flight," *Proceedings of the 3rd Asian Control Conference,* Shanghai, China, Jul. 2000.

[13] P.A. Wilson and C.J. Harris, "A line of sight counteraction navigation algorithm for ship encounter collision avoidance," *The Journal of Navigation,* vol. 56, no. 1, pp. 111–121, Jan. 2003.

[14] D.E. Chang and J.E. Marsden, "Gyroscopic forces and collision avoidance with convex obstacles," in *New Trends in Nonlinear Dynamics and Control,* LNCIS 295, W. Kang *et al.,* Eds. Berlin: Springer-Verlag, 2003, pp. 145–159.