Decentralized Flight Trajectory Planning of Multiple Aircraft

Nobuhiro Yokoyama[*]
National Defense Academy, Yokosuka , Kanagawa, Japan

Abstract: Conventional decentralized algorithms for optimal trajectory planning tend to require prohibitive computational time as the number of aircraft increases. To overcome this drawback, this paper proposes a novel decentralized trajectory planning algorithm adopting a constraints decoupling approach for parallel optimization. The constraints decoupling approach is formulated as the path constraints of the real-time trajectory optimization problem based on nonlinear programming. Due to the parallelization and no-redundancy, the computational time for one cycle in the proposed algorithm is not so sensitive to the number of aircraft as the conventional algorithms. Several results of numerical simulations are presented to demonstrate the effectiveness of the proposed algorithm.

I. Introduction

Autonomous flight trajectory planning will play an important role in free flight operations [1] in which each aircraft is capable of dynamically updating its flight plan and assuming responsibility for maintaining conflict-free trajectory (i.e. maintaining sufficient vertical or horizontal distance from own aircraft's trajectory to the other aircraft's trajectory). Thus far, many algorithms have been proposed for trajectory planning with conflict prevention and resolution [2-12]. These algorithms can roughly be characterized as follows: 1) Protocol-based or optimization-based: Although the optimization-based algorithms [6-12] tend to require substantial computational time, they offer desirable trajectory which minimizes specified criterion (e.g., arrival time delay, fuel consumption, etc.). The advantage of the protocol-based algorithms [3-5] is their simplicity. 2) Applicable to only a pair of aircraft or multiple aircraft [2]: From a practical viewpoint, applicability to multiple aircraft is more desirable. 3) Centralized or decentralized: Centralized algorithms [9-12] are well-suited to conventional ground-based air traffic controls, while decentralized algorithms [3-8] may be more suited to free flight operations because of their distributed nature.

This paper covers the optimization-based decentralized algorithms for trajectory planning of multiple aircraft, because they are most advantageous in view of safety and efficiency. This class of algorithms can further be distinguished between sequential algorithms [8] and simultaneous algorithms [6,7]. In the sequential algorithms, during the computation and update of an aircraft's trajectory, the computations and updates of the other aircraft's trajectories are stopped. Thus, the computations and updates of the trajectories are executed in one by one manner, and the computational time for one cycle (i.e., time to update all the relevant aircraft's trajectories) grows prohibitively as the number of aircraft increases. On the other hand, the simultaneous algorithms optimize each aircraft's trajectory at the same time in a decentralized fashion. The simultaneous algorithms use the computers distributed in each aircraft more efficiently than the sequential algorithms in the sense that there are no sleeping times for the computers in the simultaneous algorithms. Nevertheless, the conventional simultaneous algorithms usually require prohibitive computational time in multiple aircraft cases. For example, the approach by Bicchi et al. [7] requires the optimization of not only the own aircraft's trajectory but also the other aircraft's trajectories.

[*] Research Associate, Department of Aerospace Engineering, 1-10-20 Hashirimizu, Yokosuka.

Therefore, the computational time for this approach grows combinatorially as the number of aircraft increases. A game-theoretic approach by Tomlin, et al. [8] does not include this type of redundancy. However, the algorithmic complexity of this approach also grows combinatorially as the number of aircraft increases, and hence it may become computationally prohibitive for more than three aircraft.

In order to overcome this drawback, this paper proposes a simultaneous decentralized algorithm using constraints decoupling approach for parallel optimization. The constraints decoupling approach is formulated as the path constraints of the real-time trajectory optimization problem based on nonlinear programming. Due to the parallelization and no-redundancy, the computational time for one cycle in the proposed algorithm is not so sensitive to the number of aircraft as the conventional decentralized algorithms. The effectiveness of the proposed algorithm is evaluated through several numerical simulations.

II. Trajectory planning algorithm

A. Description of aircraft's motion and trajectory

The state equations are based on the kinematics of an aircraft. For simplicity, only the horizontal motion is covered in this study. Let us define M as the number of aircraft and $j\ (=1,\cdots,M)$ as the aircraft index. The state equations of an aircraft j are described as follows,

$$\dot{X}_j(t) = V_j(t)\cos\psi_j(t) \tag{1}$$

$$\dot{Y}_j(t) = V_j(t)\sin\psi_j(t) \tag{2}$$

$$\dot{\psi}_j(t) = \omega_j(t) \tag{3}$$

$$\dot{V}_j(t) = a_j(t) \tag{4}$$

where t : the time, X_j, Y_j : the position of the aircraft, V_j : the horizontal velocity, ψ_j : the heading angle, ω_j : the heading angular rate caused by the aircraft's bank, a_j : the horizontal acceleration. In this simple kinematic model, the state variables $\boldsymbol{x}_j(t)$ and the control variables $\boldsymbol{u}_j(t)$ are described as

$$\boldsymbol{x}_j(t) = [X_j(t), Y_j(t), \psi_j(t), V_j(t)]^T, \quad \boldsymbol{u}_j(t) = [\omega_j(t), a_j(t)]^T \tag{5}$$

In addition, the motion of the aircraft is constrained by the following inequalities,

$$(V_j)_{\min} \le V_j(t) \le (V_j)_{\max} \tag{6}$$

$$-(\omega_j)_{\max} \le \omega_j(t) \le (\omega_j)_{\max} \tag{7}$$

$$-(a_j)_{\max} \le a_j(t) \le (a_j)_{\max} \tag{8}$$

where $(V_j)_{\min}, (V_j)_{\max}, (\omega_j)_{\max}, (a_j)_{\max}$ are the constants.

Let us define the reference time as t_0 and the horizon interval as T_h. Discretizing the time domain $[t_0, t_0 + T_h]$ into N uniform intervals, let us express the nodal time as $t_0, t_1, \cdots, t_N (= t_0 + T_h)$, i.e.,

$$t_i = t_0 + i(\Delta T) \ \ (i = 0,\cdots,N), \ \ \Delta T = \frac{T_h}{N} \tag{9}$$

The trajectory at time $t \in [t_0, t_0 + T_h]$ is described by the nodal values of the state variables $\boldsymbol{x}_j(t_i)$ as well as the control variables $\boldsymbol{u}_j(t_i)$. In addition, the control variables are assumed to be piecewise constant at each interval

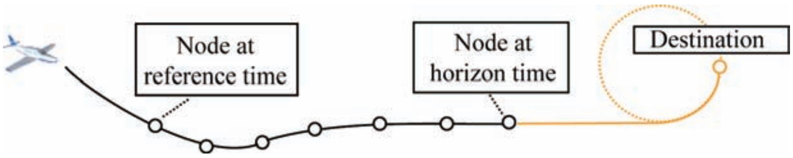


Fig. 1 Trajectory Description

$[t_i, t_{i+1}]$. As shown in Fig. 1, the trajectory beyond the horizon time ($t = t_0 + T_h$) is described by the straight line and the circular arc that connect the node at the horizon time with the destination point.

B. Configurations of algorithm

A unit cycle of the algorithm completes in the time interval ΔT. This unit cycle is split into three phases - the conflict reduction (CR) phase, the trajectory optimization (TO) phase, and the trajectory broadcast (TB) phase. It is assumed that the clocks of all the aircraft are synchronous. At the beginning of the cycle, the currently planned trajectory is modified in the CR phase to reduce the degree of conflicts with the other aircraft's trajectories. Then the trajectory modified in the CR phase is updated in the TO phase to minimize a given objective function. After the TO phase, the updated trajectory is broadcasted as the aircraft's latest trajectory. It should be noted that the role of the CR phase is to offer appropriate initial solution to the optimization algorithm in the TO phase. The details of the CR phase and the TO phase are described in subsection C and D, respectively.

Let us define t_B as the time when the current cycle starts. The reference time t_0 is set to be $t_0 = t_B + 2\Delta T$. It is assumed that the aircraft is always controlled to track the trajectory at $t \in [t_B, t_B + \Delta T]$, which is fixed in the current cycle of the algorithm.

C. Conflict reduction (CR) phase

In the CR phase, a kind of force field method [3, 4] is adopted to reduce the degree of conflicts of the aircraft's trajectories. The algorithm in the CR phase is described below.

Step 1: Conflict Search For $j = 1, \cdots, M$, search the time $(t_C)_j$ when the first conflict of the aircraft j with any other aircraft occurs based on the currently planned trajectories. Then, for $j = 1, \cdots, M$, obtain the starting time of the trajectory modification $(t_S)_j$ by the following equation.

$$(t_S)_j = \max[t_0, (t_C)_j - 2\Delta T] \tag{10}$$

Step 2: Time Initialization Set the inner simulation time τ and the index i to t_0 and zero, respectively.

Step 3: Control Calculation For $j = 1, \cdots, M$, if $\tau \geq (t_S)_j$, modify the control variables $\boldsymbol{u}_j^P(t_i)$ of the current trajectory based on the following procedure. (The superscript P denotes the component of currently planned trajectory.)

Using the force field method, generate the control command vector composed of a repulsive element for steering the aircraft away from the other aircraft and an attracting element for steering the aircraft to its destination [3]. Figure 2 shows the schematic view of the force field method. The repulsive element is obtained by using the necessary separation vector at the closest point of approach (CPA), which is determined by the linear extrapolation of the position in the direction of the aircraft's destination. Let us define t_{ijk}^A as the time of the CPA with regard to aircraft j and k determined by the linear extrapolation of the positions at time $\tau = t_i$. In

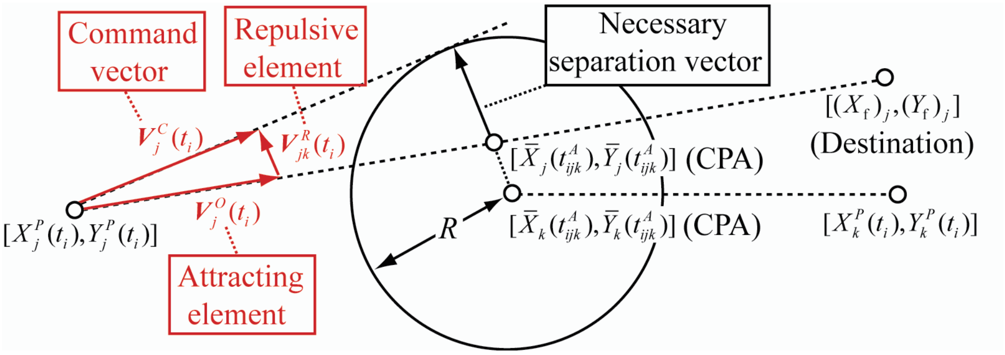

Command vector
Repulsive element
Necessary separation vector
$V_j^C(t_i)$
$V_{jk}^R(t_i)$
$[(X_f)_j,(Y_f)_j]$
(Destination)
$[\bar{X}_j(t_{ijk}^A),\bar{Y}_j(t_{ijk}^A)]$ (CPA)
$[X_j^P(t_i),Y_j^P(t_i)]$
$V_j^O(t_i)$
R
$[\bar{X}_k(t_{ijk}^A),\bar{Y}_k(t_{ijk}^A)]$ (CPA)
$[X_k^P(t_i),Y_k^P(t_i)]$
Attracting element

Fig. 2 Force field method

addition, let us define the extrapolated positions as $[\bar{X}_j(\tau),\bar{Y}_j(\tau)]$ and $[\bar{X}_k(\tau),\bar{Y}_k(\tau)]$. The control command vector $\boldsymbol{V}_j^C(t_i)$ for aircraft j is obtained by the following equations.

$$\boldsymbol{V}_j^C(t_i)=\boldsymbol{V}_j^O(t_i)+\sum_{k(\neq j)=1}^{M}\boldsymbol{V}_{jk}^R(t_i) \tag{11}$$

$$\boldsymbol{V}_j^O(t_i)=\max\left[(V_{\min})_j,\min\left[\frac{\sqrt{[(X_f)_j-X_j^P(t_i)]^2+[(Y_f)_j-Y_j^P(t_i)]^2}}{\max[1,(t_f)_j-t_i]},(V_{\max})_j\right]\right]\boldsymbol{e}_j^O(t_i) \tag{12}$$

$$\boldsymbol{e}_j^O(t_i)=\frac{[(X_f)_j-X_j^P(t_i),(Y_f)_j-Y_j^P(t_i)]}{\sqrt{[(X_f)_j-X_j^P(t_i)]^2+[(Y_f)_j-Y_j^P(t_i)]^2}} \tag{13}$$

$$\boldsymbol{V}_{jk}^R(t_i)=\frac{\max[0,R-\sqrt{[\bar{X}_j(t_{ijk}^A)-\bar{X}_k(t_{ijk}^A)]^2+[\bar{Y}_j(t_{ijk}^A)-\bar{Y}_k(t_{ijk}^A)]^2}]}{t_{ijk}^A-t_i}\boldsymbol{e}_{jk}^R(t_i) \tag{14}$$

$$\boldsymbol{e}_{jk}^R(t_i)=\frac{[\bar{X}_j(t_{ijk}^A)-\bar{X}_k(t_{ijk}^A),\bar{Y}_j(t_{ijk}^A)-\bar{Y}_k(t_{ijk}^A)]}{\sqrt{[\bar{X}_j(t_{ijk}^A)-\bar{X}_k(t_{ijk}^A)]^2+[\bar{Y}_j(t_{ijk}^A)-\bar{Y}_k(t_{ijk}^A)]^2}} \tag{15}$$

where $\boldsymbol{V}_j^O(t_i)$: the attracting element for steering the aircraft j to its destination, $\boldsymbol{e}_j^O(t_i)$: the unit vector of the attracting element, $\boldsymbol{V}_{jk}^R(t_i)$: the repulsive element for steering the aircraft j away from the other aircraft $k(\neq j)$, $\boldsymbol{e}_{jk}^R(t_i)$: the unit vector of the repulsive element, $[(X_f)_j,(Y_f)_j]$: the destination point of the aircraft j, $(t_f)_j$: the specified time to arrive at the destination, $[X_j^P(t_i),Y_j^P(t_i)]$: the currently planned position of the aircraft j at time $\tau=t_i$. If the distance of the aircraft j and k is zero at the CPA, a line passing through $[X_j^P(t_i),Y_j^P(t_i)]$ and being tangent to a circle centering on the CPA with radius R is calculated. Then $\boldsymbol{e}_{jk}^R(t_i)$ is determined as the unit vector normal to this line instead of Eq. (15).

Subsequently, convert the control command vector $\boldsymbol{V}_j^C(t_i)$ to the control variables $\omega_j^P(t_i),a_j^P(t_i)$ by the following equations,

$$\omega_j^P(t_i)=\max\left[-(\omega_j)_{\max},\min\left[\frac{\psi_j^C(t_{i+1})-\psi_j^P(t_i)}{\Delta T},(\omega_j)_{\max}\right]\right] \tag{16}$$

$$\psi_j^C(t_{i+1}) = \tan^{-1}\left[\boldsymbol{V}_j^C(t_i)_y \big/ \boldsymbol{V}_j^C(t_i)_x \right] \tag{17}$$

$$a_j^P(t_i) = \max\left[(a_j)_{\min}, \min\left[a_j^C(t_i), (a_j)_{\max} \right] \right] \tag{18}$$

$$a_j^C(t_i) = \frac{\max\left[(V_j)_{\min}, \min\left[\| \boldsymbol{V}_j^C(t_i) \|, (V_j)_{\max} \right] \right] - V_j^P(t_i)}{\Delta T} \tag{19}$$

Step 4: Integration of the State Equations For $j = 1, \cdots, M$, integrate the state equations (1)-(4) from $\tau = t_i$ to $\tau = t_{i+1}$, and update the state variables of the current trajectory $\boldsymbol{x}_j^P(t_{i+1})$ to the results obtained by the integration. Set τ to t_{i+1}, and increase i by one. If $\tau < t_N$, go back to Step 3. Otherwise, terminate the CR phase and go to the TO phase.

It should be noted that the trajectories modifications of all the aircraft are executed in each aircraft, i.e., the same computations are executed simultaneously in all the aircraft in the CR phase. Thus, the trajectories modified in the CR phase are consistent and their conflicts are substantially reduced. Although the computational time for this approach grows as the number of aircraft increases, it is still negligible compared to that of the TO phase.

D. Trajectory optimization (TO) phase

In the TO phase, the trajectory modified in the CR phase is updated to the optimal trajectory, which minimizes the squared error between the specified time $(t_f)_j$ and the time to arrive at the destination. The separation constraints on the trajectories of aircraft j and k are usually described as the following coupled form.

$$\sqrt{[X_j(t_i) - X_k(t_i)]^2 + [Y_j(t_i) - Y_k(t_i)]} - R \ge 0 \; (i = 1, \cdots, N) \tag{20}$$

In order to parallelize the computation in the TO phase, this inequality condition is decoupled as follows. As shown in Fig. 3, a feasible region for update of each node of each aircraft's trajectory is introduced based on the trajectories modified in the CR phase. The feasible region for update of the node $[X_j(t_i), Y_j(t_i)]$ of aircraft j relative to the aircraft k is expressed as follows,

$$a_{jk}(t_i)[X_j(t_i) - p_{jk}(t_i)] + b_{jk}(t_i)[Y_j(t_i) - q_{jk}(t_i)] - \frac{R}{2} \ge 0 \; (i = 1, \cdots, N) \tag{21}$$

$$a_{jk}(t_i) = \frac{X_j^P(t_i) - p_{jk}(t_i)}{d_{jk}(t_i)}, b_{jk}(t_i) = \frac{Y_j^P(t_i) - q_{jk}(t_i)}{d_{jk}(t_i)} \tag{22}$$

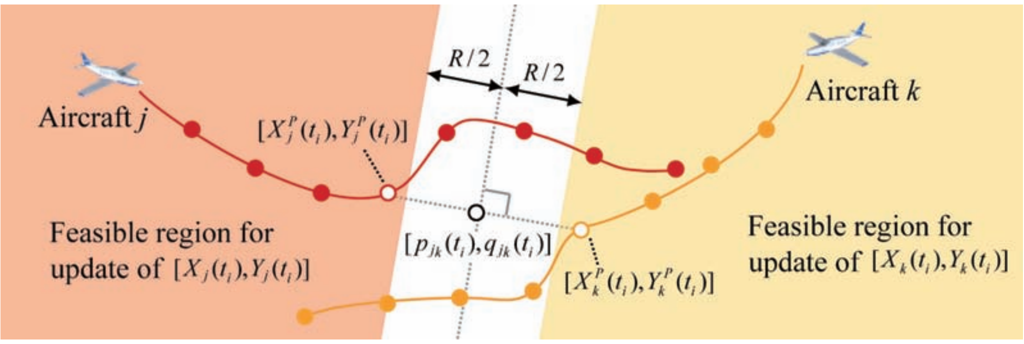

R/2
R/2
Aircraft j
Aircraft k
[X_j^P(t_i),Y_j^P(t_i)]
Feasible region for update of [X_j(t_i),Y_j(t_i)]
[p_jk(t_i),q_jk(t_i)]
[X_k^P(t_i),Y_k^P(t_i)]
Feasible region for update of [X_k(t_i),Y_k(t_i)]

Fig. 3 Feasible region for update of node

$$d_{jk}(t_i) = \sqrt{[X_j^P(t_i) - p_{jk}(t_i)]^2 + [Y_j^P(t_i) - q_{jk}(t_i)]} \tag{23}$$

where the point $[p_{jk}(t_i), q_{jk}(t_i)]$ denotes the midpoint of $[X_j^P(t_i), Y_j^P(t_i)]$ and $[X_k^P(t_i), Y_k^P(t_i)]$. It should be noted that the sufficient separation between $[X_j(t_i), Y_j(t_i)]$ and $[X_k(t_i), Y_k(t_i)]$ is assured as long as the update of each node is constrained within the assigned feasible region. Thus, the parallel computation is enabled by the inequality constraint (21). Considering all the other aircraft's nodes at time t_i, the eventual feasible region for update of $[X_j(t_i), Y_j(t_i)]$ is described as the product set of Eq. (21) with $k = 1, 2, \cdots M$ $(k \neq j)$.

Let us define the state variables and the control variables to be optimized as

$$(\boldsymbol{x}_j)_i = [(X_j)_i, (Y_j)_i, (\psi_j)_i, (V_j)_i]^T, \quad (\boldsymbol{u}_j)_i = [(\omega_j)_i, (a_j)_i]^T \tag{24}$$

where the subscript i denotes the value at time t_i. The trajectory optimization problem is defined as follows.

- The objective function to be minimized:

$$\phi_j = [t_N + (T_e)_j - (t_\mathrm{f})_j]^2 + \rho_j \delta_j \tag{25}$$

where $(T_e)_j$: the time interval between the horizon time t_N and the planned time to arrive at the destination, ρ_j : the positive penalty parameter, δ_j : the constraint relaxation variable which is explained later. The calculation of $(T_e)_j$ is based on the line and circular arc trajectory beyond the horizon.

- The initial condition:

$$(\boldsymbol{x}_j)_0 = \boldsymbol{x}_j^P(t_0) \tag{26}$$

- The terminal condition:

$$\sin\left[\frac{\psi_j(t_N + T_e) - (\psi_\mathrm{f})_j}{2}\right] = 0 \tag{27}$$

where $(\psi_\mathrm{f})_j$ denotes the specified heading angle at the destination, and $\psi_j(t_N + T_e)$ is calculated by the line and circular arc trajectory beyond the horizon. Although Eq. (27) is equivalent to $\psi_j(t_N + T_e) - (\psi_\mathrm{f})_j = 2n\pi$ (n is an arbitrary integer), it is adopted due to its suitability to the numerical optimization.

- The integration condition of the state equations:

$$(\boldsymbol{x}_j)_{i+1} - \int_{t_i}^{t_{i+1}} [\boldsymbol{f}(\boldsymbol{x}_j(\tau), \boldsymbol{u}_j(\tau)) \,|\, \boldsymbol{x}_j(t_i) = (\boldsymbol{x}_j)_i, \boldsymbol{u}_j(\tau) = (\boldsymbol{u}_j)_i] d\tau = \boldsymbol{0}, (i = 0, \cdots, N-1) \tag{28}$$

where $\boldsymbol{f}$ denotes the right hand side of Eqs. (1)-(4). Since the control variable $\boldsymbol{u}_j(\tau)$ is piecewise constant vector $(\boldsymbol{u}_j)_i$ in the interval $[t_i, t_{i+1}]$, the second term of the above equation can be analytically integrated.

- The path constraints: Instead of the separation constraint (21), the following constraints are enforced.

$$a_{jk}(t_i)[(X_j)_i - p_{jk}(t_i)] + b_{jk}(t_i)[(Y_j)_i - q_{jk}(t_i)] - \left[\frac{R}{2} + \left(d_{jk}(t_i) - \frac{R}{2}\right)\delta_j\right] \geq 0, [i = 1, \cdots, N, k = 1, \cdots, M(k \neq j)] \tag{29}$$

$$0 \leq \delta_j \leq 1 \tag{30}$$

Even if it is impossible to obtain conflict-free solutions due to the geometric relationships among the currently planned trajectories, the introduction of the relaxation variable δ_j makes it substantially easier to obtain conflict-reduced solutions. On the other hand, strong relaxation of Eq. (29) by large δ_j is penalized as the deterioration of the objective function (25).

The optimization problem described above is simultaneously solved in each aircraft j $(= 1, \cdots, M)$. The variables to be optimized for the aircraft j are $(\boldsymbol{x}_j)_0, (\boldsymbol{x}_j)_1 \cdots, (\boldsymbol{x}_j)_N, (\boldsymbol{u}_j)_0, (\boldsymbol{u}_j)_1, \cdots, (\boldsymbol{u}_j)_{N-1}$ and δ_j. The trajectory modified in the CR phase is used as the initial solution of the optimization algorithm, and the initial solution of δ_j is set to be zero. In the optimization problem described above, the Jacobian of the constraints and the Hessian of the Lagragian have special sparse structure. Thus, the sparse sequential quadratic programming (SQP) [13] is adopted as the optimization algorithm. It is well known that the sparse SQP can compute the optimal solution substantially faster than the conventional dense matrix approach. In this way, the real-time optimization is enabled in the proposed algorithm.

III. Examples of simulation

Some example scenarios were considered to evaluate the effectiveness of the proposed algorithm. In all the scenarios, the separation minimum was specified as *R* = 9260 [m]. The performance of each aircraft was assumed to be the same, and the constants were specified as follows.

$$(V_j)_{\min} = 51.44\,[\text{m/s}],\ (V_j)_{\max} = 72.02\,[\text{m/s}],\ (\omega_j)_{\max} = 0.02401\,[\text{rad/s}],\ (a_j)_{\max} = 1.0\,[\text{m/s}^2] \tag{31}$$

A numerical simulation environment, which was composed of multiple PCs (personal computers) connected by LAN (local area network), was developed for evaluating the proposed algorithm. In this environment, the PCs corresponded to the aircraft, and the LAN corresponded to the airborne data link system. The period of the unit cycle and the number of nodes were specified as ΔT = 10.0 [sec] and 61, respectively.

Five aircraft, whose initial and terminal states are listed in Table 1, were assumed. In each aircraft, the time to arrive at the destination was specified as $(t_\mathrm{f})_j = 1000\,[\text{sec}]$, and the initial horizontal velocity was specified as $61.73\,[\text{m/s}]$. The aircraft's trajectories before applying the proposed algorithm were supposed to be straight. On the basis of these specifications, the following five conflict scenarios were considered: 1) aircraft A and B, 2) aircraft A and C, 3) aircraft A and E, and 4) aircraft A, C, and E, 5) aircraft A, B, C, and D. In each scenario, the initial straight trajectories converged to a point at the same time, i.e., the initial trajectories were assumed to be severely conflicted.

Figure 4 shows the final trajectories tracked by the aircraft in these scenarios. In this figure, the CPAs in the final trajectories are plotted as the circles. It can be observed that the final trajectories computed by the proposed algorithm are conflict-free and have reasonableness as the near-optimal solution, because the distance of every pairs of CPAs is equal to or slightly larger than the separation minimum *R* (=9260 [m]). As an example of planned trajectories obtained by the proposed algorithm, Fig. 5 shows the trajectories planned at the cycle of $t_0 = 200\,[\text{sec}]$. The CPAs detected in the planned trajectories are also plotted as the circles in this figure. It was observed that the distance of every pairs of CPA is equal to or slightly larger than the separation minimum *R*. Thus, the reasonableness of the planned trajectories can also be confirmed.

With respect to the computational speed, the proposed algorithm showed sufficient performance, because the time to compute a feasible solution was substantially smaller than the cycle period ΔT (= 10 [sec]) in each scenario (in the worst case, it took 0.38 [sec]). Table 2 shows the average of the computational time per iteration of the sparse

Table 1 Initial and terminal states of the aircraft

Aircraft	A	B	C	D	E
Initial position [m]	(0, 36000)	(0, -36000)	(36000, 0)	(–36000, 0)	(25456, 25456)
Initial heading angle [rad]	$-\pi/2$	$\pi/2$	π	0	$-3\pi/4$
Terminal position [m]	(0, –36000)	(0, 36000)	(–36000, 0)	(36000, 0)	(-25456, -25456)
Terminal heading angle [rad]	$-\pi/2$	$\pi/2$	π	0	$-3\pi/4$

Table 2 Comparison of the computational time

Scenario	(1)	(2)	(3)	(4)	(5)
Number of aircraft	2	2	2	3	4
Average computational time per iteration of sparse SQP [sec]	0.20	0.20	0.16	0.l4	0.37

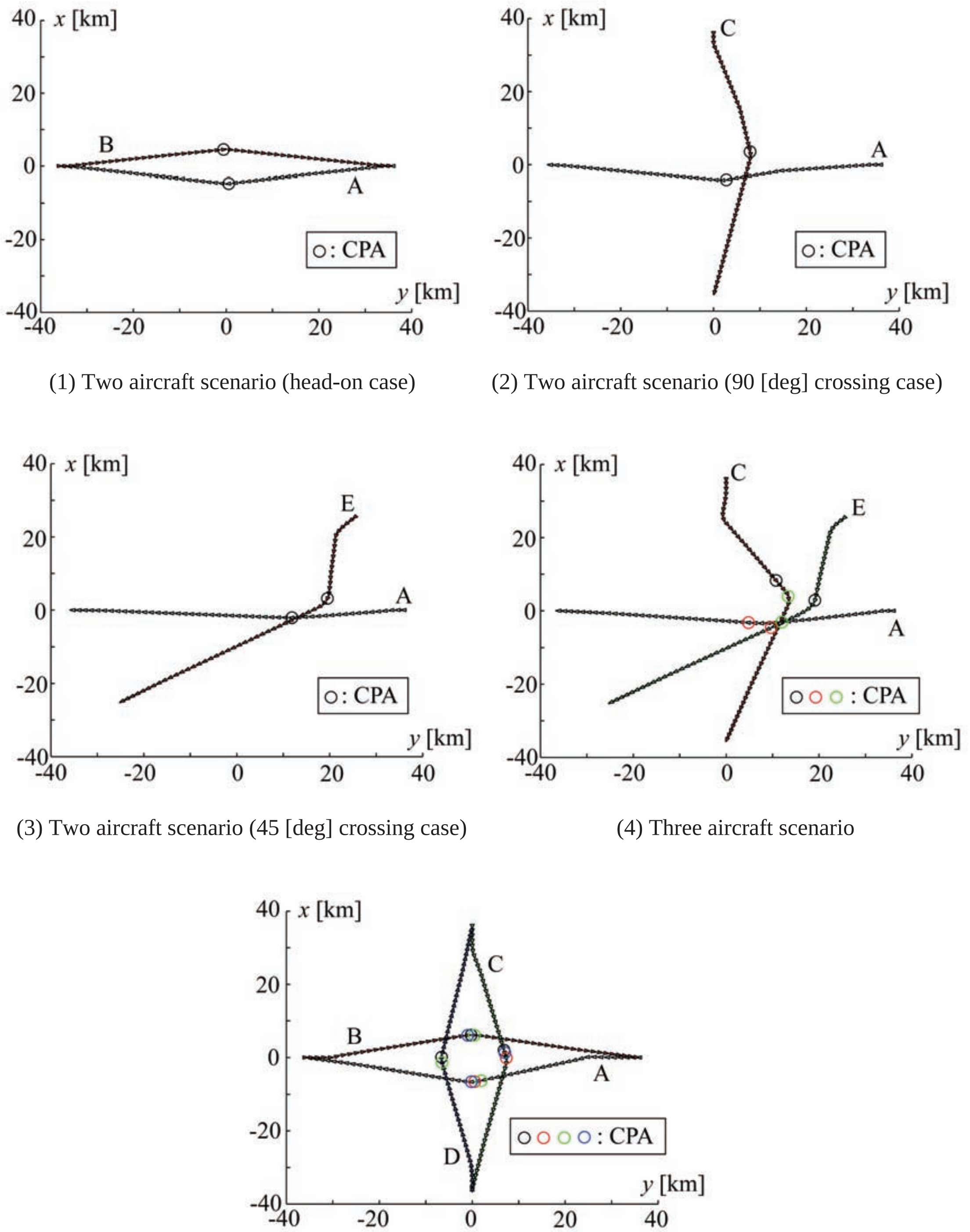
SQP in the TO phase. In the conventional simultaneous algorithms, this criterion is sensitive to the number of aircraft, and it generally grows combinatorially. Although the number of scenarios is not sufficient, it can be seen that the proposed algorithm is not so sensitive to the number of aircraft as the conventional algorithms.

IV. Conclusions

A decentralized algorithm for optimal trajectory planning of multiple aircraft was presented in this paper. The novelty of the proposed algorithm lies in its decoupling of the separation constraints for parallel optimization. Through the application to some numerical simulations, it was observed that the proposed algorithm could compute reasonable conflict-free trajectories with sufficient computational speed. Our future work will include extensive evaluations and convergence assurance of the algorithm, as well as the algorithm enhancement for the three-dimensional problem.

References

[1] RTCA, “Final Report of RTCA Task Force 3: Free Flight Implementation,” RTCA Inc., Washington, D.C., 1995.
[2] Kuchar, J. K. and Yang, L. C., “A Review of Conflict Detection and Resolution Modeling Methods,” IEEE Transactions on Intelligent Transportation Systems, Vol. 1, No. 4, 2000, pp.179–189.
[3] Eby, M. S., “A Self-organizational Approach for Resolving Air Traffic Conflicts,” Lincoln Laboratory Journal, Vol. 7, No. 2, 1994, pp.239–254.
[4] Hoekstra, J. M., van Grent, R. N. H. W., and Ruigrok, R. C. J., “Designing for Safety: the ‘Free Flight’ Air Traffic Management Concept,” Reliability Engineering and System Safety, Vol. 75, 2002, pp.215–232.
[5] Frazzoli, E., Pallottino, L, Scordio, V., and Bicchi, A., “Decentralized Cooperative Conflict Resolution for Multiple Nonholonomic Vehicles,” AIAA Paper 2005–6048, 2005.
[6] Tomlin, C., Pappas, G., and Sastry, S., “Conflict Resolution for Air Traffic Management: A Study in Multi-Agent Hybrid Systems,” IEEE Transactions on Automatic Control, Vol. 43, No. 4, 1998, pp.509–521.
[7] Bicchi, A. and Pallottino, L., “On Optimal Cooperative Conflict Resolution for Air Traffic Management Systems,” IEEE Transactions on Intelligent Transportation Systems, Vol. 1, No. 4, 2000, pp.221-232.
[8] Schouwenaars, T., How, J., and Feron, E., “Decentralized Cooperative Trajectory Planning of Multiple Aircraft with Hard Safety Guarantees,” AIAA Paper 2004–5141, 2004.
[9] Menon, P. K., Sweriduk, G. D., and Sridhar, B., “Optimal Strategies for Free-Flight Air Traffic Conflict Resolution,” Journal of Guidance, Control, and Dynamics, Vol. 22, No. 2, 1999, pp.202–211.
[10] Frazzoli, E., Mao, Z. H., Oh, J. H., and Feron, E., “Resolution of Conflicts Involving Many Aircraft via Semidefinite Programming,” Journal of Guidance, Control, and Dynamics, Vol. 24, No. 1, 2001, pp.79–86.
[11] Hu, J., Prandini, M., and Sastry, S., “Optimal Coordinated Maneuvers for Three-dimensional Aircraft Conflict Resolution,” Journal of Guidance, Control, and Dynamics, Vol. 25, No. 5, 2002, pp.888–900.
[12] Raghunathan, A. U., Gopal, V., Subramanian, D., Biegler, L. T., and Samad, T., “Dynamic Optimization Strategies for Three-dimensional Conflict Resolution of Multiple Aircraft,” Journal of Guidance, Control, and Dynamics, Vol. 27, No. 4, 2004, pp.586–594.
[13] Betts, J. T., “Practical Methods for Optimal Control Using Nonlinear Programming,” SIAM, Philadelphia, 2001.



(1) Two aircraft scenario (head-on case)

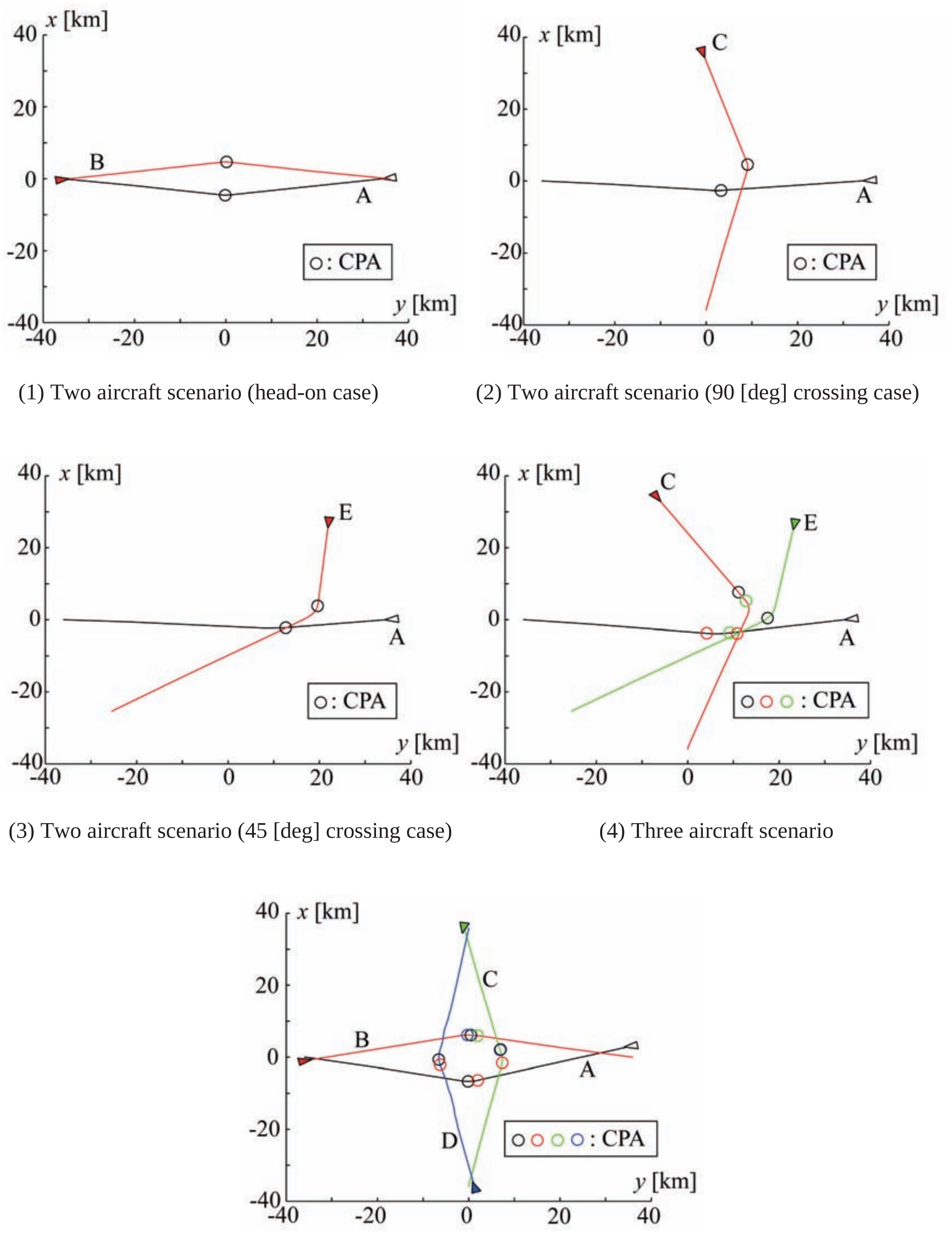
(2) Two aircraft scenario (90 [deg] crossing case)

(3) Two aircraft scenario (45 [deg] crossing case)

(4) Three aircraft scenario

(5) Four aircraft scenario

Fig. 4 Final trajectories tracked by the aircraft



(1) Two aircraft scenario (head-on case)

(2) Two aircraft scenario (90 [deg] crossing case)

(3) Two aircraft scenario (45 [deg] crossing case)

(4) Three aircraft scenario

(5) Four aircraft scenario

Fig. 5 Trajectories planned at the cycle of $t_0 = 200$ [sec]