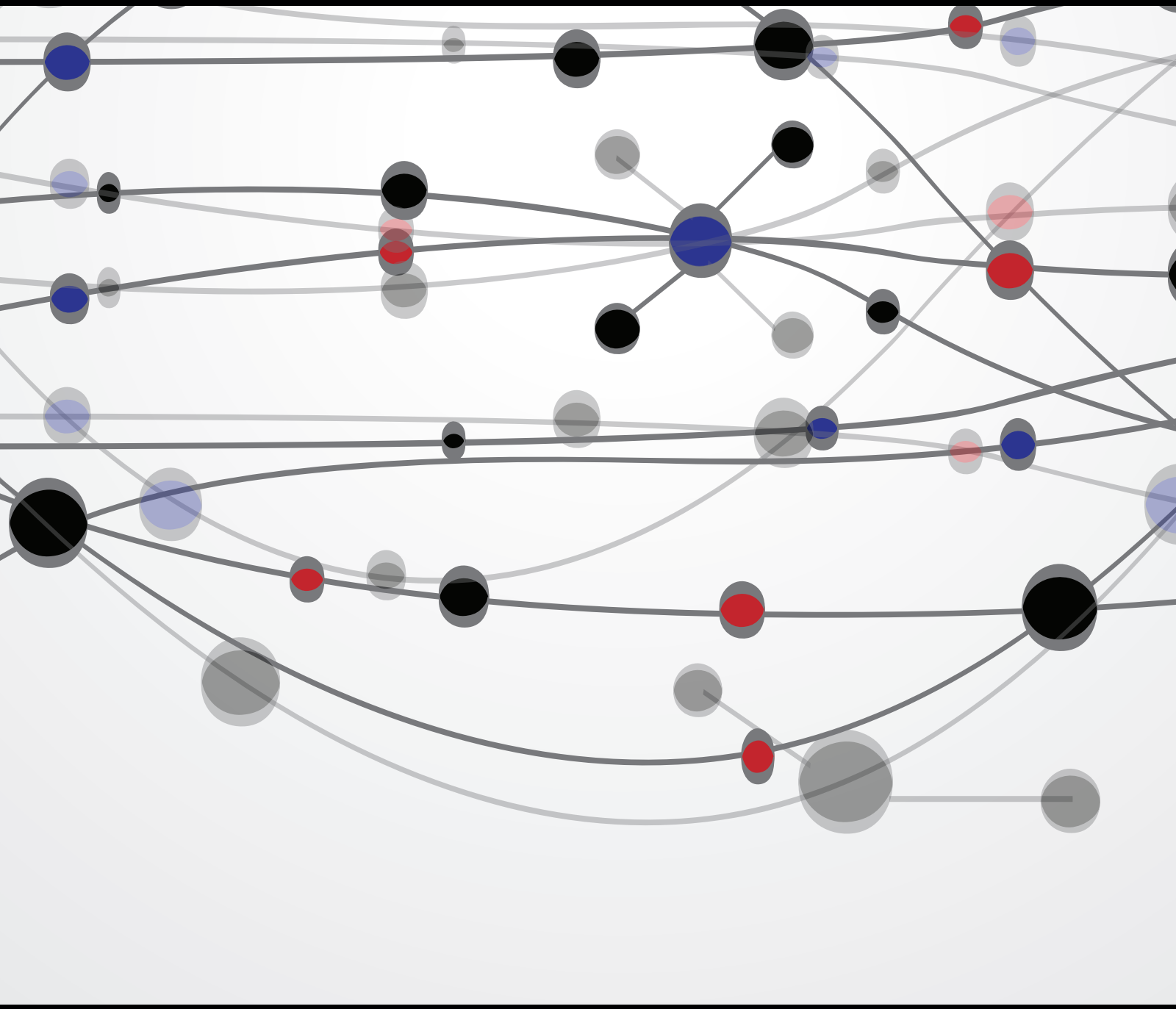
The Scientific World Journal

Modeling, Control, and Optimization in Aeronautical Engineering

Guest Editors: Zheng Zheng, Kemao Peng, Wenbo Du, and Guangquan Zhang



Modeling, Control, and Optimization in Aeronautical Engineering

The Scientific World Journal

Modeling, Control, and Optimization in Aeronautical Engineering

Guest Editors: Zheng Zheng, Kemao Peng, Wenbo Du, and Guangquan Zhang

Contents

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 979107, 2 pages
http://dx.doi.org/10.1155/2015/979107

Editorial

Modeling, Control, and Optimization in Aeronautical Engineering

Zheng Zheng,[1] Kemao Peng,[2] Wenbo Du,[3] and Guangquan Zhang[4]

[1] *School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China*
[2] *Temasek Laboratories, National University of Singapore, Singapore*
[3] *School of Electronic and Information Engineering, Beihang University, Beijing 100191, China*
[4] *Centre for Quantum Computation and Intelligent Systems, University of Technology, Sydney, NSW 2007, Australia*

Correspondence should be addressed to Zheng Zheng; zhengz@buaa.edu.cn

Received 3 May 2015; Accepted 3 May 2015

The last decade has witnessed the rapid development of aeronautical engineering including all branches of applied sciences and technology dealing with aircraft and their support systems, which brings forward the higher request of safety, efficiency, and environmental protection. Modeling, control, and optimization of aeronautical engineering have played an increasingly important role in meeting aeronautical requirements, and they have drawn widespread attention from communities including control theory, intelligent optimization, system science, real-time distributed computing, electronic information engineering, and aeronautical engineering industry. Driven by such motivations, the main focus of this special issue is on the new theories, new technologies, and their applications in modeling, control, and optimization for aeronautical engineering systems.

The special issue is a collection of original research articles, whose authors and editors belong to academic or research institutions of five different countries from Asia, Europe, and Australia. The full papers in this issue can be broadly organized into two main categories: (i) modelling and optimization and (ii) control techniques.

Papers in category (i) are mainly concerned with network optimization of air transportation system and terminal-area operation. Recently, people come to realize that the air transportation system can be modeled as complex networks. Along this line, K. Cai et al. in "A Novel Biobjective Risk-Based Model for Stochastic Air Traffic Network Flow Optimization Problem" and X. Guan et al. in "An Airway Network Flow Assignment Approach Based on an Efficient Multiobjective Optimization Framework" modeled the flight operation as network flow optimization problem and proposed novel algorithms, respectively. Terminal-area operation is the bottleneck of the air transportation system. Y. Yang et al. investigated the aircraft intent inference approach and proposed an online trajectory clustering method, which will be meaningful to the efficiency and safety of busy terminal area.

Papers in category (ii) are focusing on the topics of rotary-wing aircrafts and robots. The rotary-wing aircraft is one of the popular platforms in the control community such as the helicopter and quadrotor. T. Oktay and F. Sal in "Helicopter Control Energy Reduction Using Moving Horizontal Tail" attempted to improve the flight duration by reducing consumption of the energy. X. Xu et al. in "MRAC Control with Prior Model Knowledge for Asymmetric Damaged Aircraft" intended to explore control of the wing-damaged aircraft, which is one of advanced topics to improve survivability of the fighters. Z. Li and Y. Wang in "Coordinated Control of Slip Ratio for Wheeled Mobile Robots Climbing Loose Sloped Terrain" attempted to explore the control of mobile robots in 3D environments. J. López et al. in "A Robust H_∞ Controller for an UAV Flight Control System" implemented and validated a robust H_∞ controller for an UAV to track different types of manoeuvres in the presence of noisy environment.

It should be mentioned that a special issue simply provides a snapshot of the field taken at a particular point in time. Due to the standard page limitations of a journal volume,

it can only include a relatively small number of papers. As a result, its coverage is by no means complete despite our best efforts.

Acknowledgments

Many individuals contributed to the success of this special issue. We take this opportunity to thank all the authors for their submissions. We are also indebted to a small army of referees who have put in the hard work and the long hours to review each paper in a timely and professional way. Editorial Assistant Kun Qiu provided valuable assistance. Last but not least, we are indebted to this journal for offering us this opportunity and for patiently waiting for the completion of the special issue.

Zheng Zheng
Kemao Peng
Wenbo Du
Guangquan Zhang

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 671360, 11 pages
http://dx.doi.org/10.1155/2015/671360

Research Article

Terminal-Area Aircraft Intent Inference Approach Based on Online Trajectory Clustering

Yang Yang, Jun Zhang, and Kai-quan Cai

School of Electronic and Information Engineering, Beihang University, Xueyuan Road No. 37, Beijing 100191, China

Correspondence should be addressed to Kai-quan Cai; caikq@ee.buaa.edu.cn

Received 11 November 2014; Accepted 15 December 2014

Academic Editor: Kemao Peng

Terminal-area aircraft intent inference (T-AII) is a prerequisite to detect and avoid potential aircraft conflict in the terminal airspace. T-AII challenges the state-of-the-art AII approaches due to the uncertainties of air traffic situation, in particular due to the undefined flight routes and frequent maneuvers. In this paper, a novel T-AII approach is introduced to address the limitations by solving the problem with two steps that are intent modeling and intent inference. In the modeling step, an online trajectory clustering procedure is designed for recognizing the real-time available routes in replacing of the missed plan routes. In the inference step, we then present a probabilistic T-AII approach based on the multiple flight attributes to improve the inference performance in maneuvering scenarios. The proposed approach is validated with real radar trajectory and flight attributes data of 34 days collected from Chengdu terminal area in China. Preliminary results show the efficacy of the presented approach.

1. Introduction

Terminal-area aircraft intent inference (T-AII) consists in predicting the future intended route for aircraft flying in the terminal airspace and can hence support conflict detection and resolution (CD&R) operations [1]. Indeed, in order to address the flight safety challenges posed by high traffic density operation in the terminal area, automatic CD&R tool resting on the aircraft intent information is taken as the main resolution strategy in the next generation air traffic management system master plans, such as, European SESAR [2] and US NextGen [3]. Recently, the AII problem has been widely studied in both academic and industrial contexts [4–9]. Additionally, the prototype systems for maintaining aircraft separation in terminal area based on aircraft intent have also been developed and tested [10–12].

Due to the uncertainties affecting the terminal-area operations, T-AII is a more difficult task than en route AII. This is because terminal-area air traffic control operates in a low altitude airspace, which can be strongly affected by severe meteorological conditions and may be subject to special use airspace restrictions. The route assigned to each aircraft is determined by the real-time airspace conditions and, as a result, no detailed flight plan can be provided in advance to pilots for the terminal flight phase. Furthermore, to maintain safe separation in a confined space, air traffic controllers issue clearances, such as vectoring, hold-on, and direct-to clearances, to direct aircraft, and, hence, pilots must implement maneuvers frequently. Clearances are oral and transmitted by a voice communication system, which aggravates the unpredictability of T-AII. These uncertainties of air traffic situation characterizing the terminal area bring new challenges for the AII problem.

The state-of-the-art AII approaches can be divided into two main classes depending on the kind of information that is used to infer the aircraft future intended route. In the first class of approaches, pilot actions are supposed to be available and are matched to the intended route as extracted from the en route flight plan [13, 14]. Various methods were then proposed to solve this best matching problem, such as those based on expert systems [15], plan recognition [16], and event tracking [17]. However, since pilot actions are hard to collect and transmit, these approaches are not widely used in practice. In the second class of approaches, an air traffic controller point of view is adopted and AII is realized through best fitting of the aircraft state observations

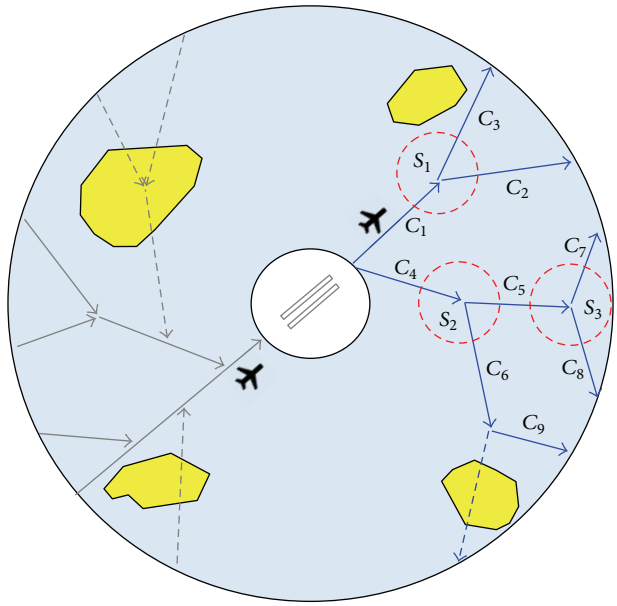


Figure 1: Terminal-area aircraft intent inference scenarios.

to intent models [4, 5]; for example, heading angle is used to infer horizontal intent. Enhanced versions of this idea have been developed in [6–8], where both spatial and temporal aircraft information are adopted to improve intent inference performance. Obviously, the second class of approaches is much easier to be implemented and demonstrated.

Both classes of approaches are characterized by the following two properties, which make them not suitable for addressing the T-AII problem: (1) they rely on predefined flight plans (for example, in [4], 22 intent models based on the flight plan are investigated); and (2) they are all based on best fitting of observations to the intent model, which can be effective only in an en route scenario where maneuvers are quite limited.

In order to solve the specific challenges in the T-AII problem, this paper proposes a two-step T-AII approach where in the first step an intent model is derived and then, in the second step, intent inference is performed based on the identified intent model.

First of all, intent modeling is formulated as a trajectory clustering problem which is motivated by the idea of terminal path library [18, 19]. The intended routes are represented by the cluster centroids extracted from the trajectory data without the input of flight plans. However, the existing trajectory clustering approaches in aviation domain are offline in nature [20–27] with the aim of evaluating the operational performance, which are relying on the complete trajectory information obtained after operations. Hence, an online trajectory clustering approach is designed so as to recognize the real-time intended routes in which the cluster centroids are dynamically updated and reorganized based on the new input information of trajectory. Here, a flexible binary-tree structure [28, 29] is adopted.

In the second step, the T-AII is implemented with a probabilistic scheme integrating the multiple flight attributes. The attributes, such as, heading angle and route relevance, can be obtained through the trajectory data. Note that the route relevance expressing the relation of two connected routes is offered by the online trajectory clustering procedure. Other attributes, such as flight call sign, destination airports, aircraft size, and aircraft approach category, can be obtained from air traffic flow management system. Then, based on the current trajectory and flight attributes, the intended route with maximum probability is identified as the inferred intent at the time instant.

The proposed approach is verified on the radar trajectory and flight attributes data of 34 days including 8995 departure flights collected from Chengdu terminal area in China. Based on intended routes recognized from the online trajectory clustering procedure, the efficacy of T-AII is investigated in three intent inference scenarios with aircraft maneuvers.

The rest of this paper is organized as follows. Section 2 briefly outlines the T-AII problem with some examples. Section 3 introduces the proposed two-step T-AII approach and prescribes in detail the online trajectory clustering procedure and the probabilistic T-AII with multiple flight attributes. Some experimental results are presented in Section 4. Finally, Section 5 draws some conclusions.

2. Problem Description

The configuration of terminal area is normally shown as a circular region centered at the airport shown in Figure 1, where departure routes (blue lines) and arrival routes (grey lines) are always spatially separated for the sake of flight safety. The solid lines represent the available routes while the dashed lines represent the closed routes. Severe meteorological or special use airspace conditions are represented by yellow

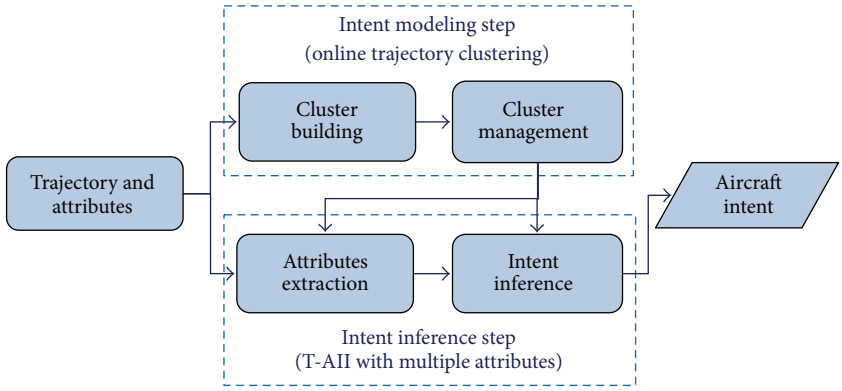


Figure 2: Block diagram of the proposed T-AII approach.

clouds. Here, the problem is presented with reference to the 2-dimensional case but can be generalized to the more general 3-dimensional case.

Take departure T-AII scenario in Figure 1 as an example. When an aircraft is flying along the route denoted by C_1 inside scenario S_1 (represented by red dashed circle), then the T-AII should determine the most likely intended route (either route C_2 or route C_3) for this aircraft based on the available information collected in route C_1. Another two successive scenarios S_2 and S_3 are shown in Figure 1. For simplicity, two possible intended routes are considered in each scenario, which however can be easily extended to multiple intended routes case.

3. T-AII Approach Based on Online Trajectory Clustering

Figure 2 gives the block diagram of the T-AII approach including two successive steps that are intent modeling and intent inference. The intent modeling step is divided into two parts: cluster building and cluster management. Radar trajectory at each time instant is applied to update the cluster centroids and dynamically reshape the structure of clusters based on the cluster control rules and operators, such as updating, splitting, merging, and deleting operators. In the intent inference step, several flight attributes are considered and the corresponding probabilities are extracted from both trajectory data and air traffic flow management system. The candidate intended route with the maximum probability summation of all attributes in logarithmic format is identified as the inferred intent. The detailed steps of online trajectory clustering procedure and T-AII with multiple attributes are introduced as follows.

3.1. Online Trajectory Clustering. In this section, we present the online trajectory clustering procedure. Let us denote by $\mathbf{C} = \{C_m\}_{m=1}^{M}$ the cluster set with M elements organized in binary-tree structure. Each cluster is represented by the cluster centroids $C_m = \{\hat{x}_m^k, \hat{y}_m^k, \hat{\sigma}_m^k\}_{k=1}^{K_m}$ along the time horizon $[1, K_m]$, where $\{\hat{x}_m^k, \hat{y}_m^k\}$ represents the horizontal position at discrete time instant k and $\hat{\sigma}_m^k$ represents variance of cluster in the corresponding position. The trajectories are denoted by $\mathbf{T} = \{T_i^j\}$, $i = 1, \ldots, N$, $j = 1, \ldots, n$, where $T_i^j = \{x_i^j, y_i^j\}$ represents the position of trajectory i at time instant j. N represents the number of aircraft trajectories and n represents the number of points for each trajectory. The detailed steps of the procedure are summarized as cluster building and cluster management.

(1) Cluster Building. Initially, set the first cluster $C_1 \in \mathbf{C}$ as the same positions of forepart α points of the trajectory $\{T_1^j\}_{j=1}^{n}$. Then, new trajectory $\{T_i^j\}_{j=1}^{\alpha}$ is used to match any existing cluster $C_m \in \mathbf{C}$ based on the trajectory-cluster similarity (TCS) defined in the following:

$$\left\{\mathrm{TCS}\left(T_i^j, C_m\right)\right\}_{j=1}^{\alpha} = \left\{\min_{k\in[1,K_m]} \frac{1}{\hat{\sigma}_m^k}\sqrt{\left(x_i^j - \hat{x}_m^k\right)^2 + \left(y_i^j - \hat{y}_m^k\right)^2}\right\}_{j=1}^{\alpha}. \quad (1)$$

If there is any $C_{m^*} \in \mathbf{C}$, let the similarity summation $\sum_{j=1}^{\alpha} \mathrm{TCS}(T_i^j, C_{m^*})$ satisfy the predefined similarity threshold; then the trajectory is assigned the specific cluster. Otherwise, a new cluster is created based on the trajectory information. Cluster set $\mathbf{C}$ is updated.

Clusters are dynamically manipulated through formulating the cluster set into the binary-tree structure based on the following cluster control rules and cluster building operators. The subsequent points $\{T_i^j\}_{j=\alpha+1}^{n}$ of each trajectory are the input information for the online process. In order to handle the trajectory point at time instant j, we also need to reserve the subsequent $\alpha - 1$ trajectory points. Note that an array of trajectory points is used to match the cluster instead of a single point so as to avoid the mismatch due to some irregular trajectory data. As a result, the TCS between the trajectory points $\{T_i^\tau\}_{\tau=j}^{j+(\alpha-1)}$ and cluster C_m are denoted as an array $\{\mathrm{TCS}(T_i^\tau, C_m)\}_{\tau=j}^{j+(\alpha-1)}$, where τ is the time instant. Here, some notations are given: reserving array is denoted as *reserving*, the lower and upper similarity thresholds are denoted as S_l and S_u, and the lower and upper thresholds of array length are denoted as L_l and L_u.

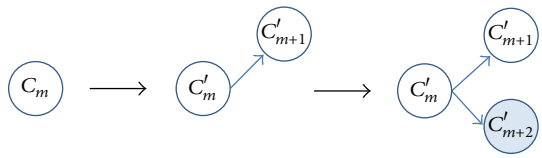


Figure 3: Cluster splitting operator.

The detailed cluster control rules are given as follows. Note that the trajectory point is determined as the outlier if it does not satisfy any of the following rules.

(1) If the trajectory point $T_i^\tau, \tau = j, \ldots, j+(\alpha-1)$, satisfies $\mathrm{TCS}(T_i^\tau, C_m) \leq S_l$, the updating operator is active. Otherwise, the trajectory point $T_i^\tau, \tau = j, \ldots, j + (\alpha - 1)$, is inserted into the array *reserving*, and the updating and splitting operators are inactive. Then, trajectory is shifted to the next time instant $\tau + 1$.

(2) Among the array *reserving*, if there exists a subarray $\{\mathrm{TCS}(T_i^\tau, C_m)\}_{\tau=k}^{k+l-1}$ with length $l < L_l$ in which values of all elements are larger than S_l and the similarity at time instant $k + l$ satisfies $\mathrm{TCS}(T_i^{k+l}, C_m) \leq S_l$, the updating operator is active. Then, the cluster C_m is updated with trajectory information $\{\mathrm{TCS}(T_i^\tau, C_m)\}_{\tau=k}^{k+l}$ and it empties the array *reserving*.

(3) Among the array *reserving*, if there exists a subarray with length $l \geq L_u$ in which values of all elements are larger than S_l, the splitting operator is active. Then, the cluster C_m is split into two separate clusters and it empties the array *reserving*.

(4) Among the array *reserving*, if there exists a subarray with length $L_l \leq l \leq L_u$ in which values of all elements are larger than S_u, the splitting operator is active. Then, the cluster C_m is split into two separate clusters and it empties the array *reserving*.

Then, the two cluster building operators are described as follows.

(1) Updating operator: when cluster C_m is updated to the new cluster C'_m based on the trajectory point information $T_i^\tau = (x_i^\tau, y_i^\tau)$, the corresponding centroid of $C_m = \{\hat{x}_m^{k^*}, \hat{y}_m^{k^*}, \hat{\sigma}_m^{k^*}\}$ is updated based on (2), where k^* is the index of centroid gained with the corresponding minimum TCS value and $\eta_1 \in [0, 0.5]$:

$$\begin{aligned}
\left(\hat{x}_m^{k^*}\right)' &= \eta_1 \hat{x}_m^{k^*} + (1-\eta_1)\, x_i^\tau, \\
\left(\hat{y}_m^{k^*}\right)' &= \eta_1 \hat{y}_m^{k^*} + (1-\eta_1)\, y_i^\tau, \\
\left(\hat{\sigma}_m^{k^*}\right)' &= \eta_1 \hat{\sigma}_m^{k^*} + (1-\eta_1)\left[\left(x_i^\tau - \hat{x}_m^{k^*}\right)^2 + \left(y_i^\tau - \hat{y}_m^{k^*}\right)^2\right].
\end{aligned} \tag{2}$$

(2) Splitting operator: when the cluster C_m is split at time instant k^*, C_m is separated into two new clusters C'_m and C'_{m+1}. Then, a new cluster C'_{m+2} with default variance is created based on the subsequent trajectory points after time instant k^*; see Figure 3.

(2) Cluster Management. In the cluster building step, a tree structure based cluster set is generated. However, each cluster is followed by multiple subsequent clusters due to the frequent use of splitting operator. In order to recognize the main clusters, the clusters are managed through deleting and merging operators. Note that the number of trajectory points belonging to each cluster is counted in the cluster building step. And the cluster-cluster similarity (CCS) is defined in the following:

$$\begin{aligned}
\mathrm{CCS}(C_m, C_n) &= \frac{1}{K}\sqrt{\sum_{k=1}^{K}\left(\left(\hat{x}_m^k - \hat{x}_n^k\right)^2 + \left(\hat{y}_m^k - \hat{y}_n^k\right)^2\right)}, \\
K &= \min\left(K_m, K_n\right).
\end{aligned} \tag{3}$$

The cluster management operators are described as follows.

(1) Deleting operator: if the number of trajectory points associated to the cluster is less than a threshold, the corresponding cluster is deleted. These trajectory points are classified as outliers.

(2) Merging operator: when CCS of clusters C_m and C_n satisfies the predefined threshold, these clusters are merged and the new cluster is computed based on (4), where η_2 is defined as the proportion of number of trajectory points associated to the cluster C_m in the total number of trajectory points associated to clusters C_m and C_n:

$$\begin{aligned}
\left(\hat{x}_m^k\right)' &= \eta_2 \hat{x}_m^k + (1-\eta_2)\, \hat{x}_n^k, \\
\left(\hat{y}_m^k\right)' &= \eta_2 \hat{y}_m^k + (1-\eta_2)\, \hat{y}_n^k, \\
\left(\hat{\sigma}_m^k\right)' &= \eta_2 \hat{\sigma}_m^k + (1-\eta_2)\, \hat{\sigma}_n^k, \\
k &= 1, \ldots, \min\left(K_m, K_n\right).
\end{aligned} \tag{4}$$

Two kinds of merging operators are shown in Figures 4 and 5. For parallel merging operator, the clusters C_m and C_n are divided into 3 new clusters separately denoted by C'_m, C'_{m+1}, C'_{m+2} and C'_n, C'_{n+1}, C'_{n+2}. The corresponding clusters C'_m and C'_n are then merged into a new cluster C''_m. For cascaded merging operator,

Table 1: Notation and definition of flight attributes.

Notation	Definition	Category	Data
CS	Call sign (CS) is the identification of each flight, such as ABW285 and CAO1023.	Static	Discrete
DA	Destination airport (DA) is divided into several groups based on geographical location of destination airport, such as northeast and southwest.	Static	Discrete
AS	Aircraft size (AS) refers to the wake turbulence category, such as heavy, medium, and light.	Static	Discrete
AC	Aircraft approach category (AC) is defined based on the speed category on the approach phase, such as A, B, C, D, and E.	Static	Discrete
RR	Route relevance (RR) refers to the relation of the connected intended routes.	Dynamic	Discrete
HA	Heading angle (HA) is defined as the angular difference between the aircraft heading direction and the intended routes.	Dynamic	Continuous

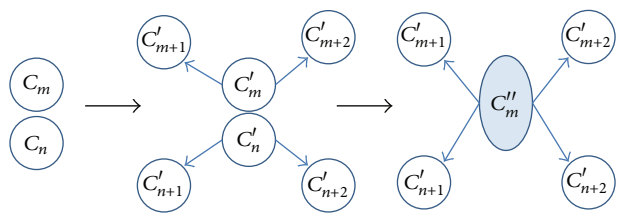


Figure 4: Parallel merging operator.

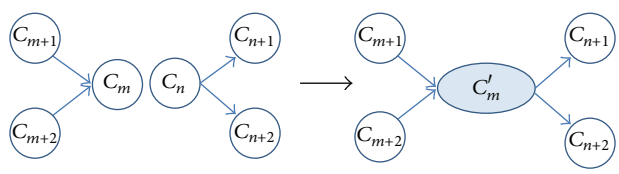


Figure 5: Cascaded merging operator.

the cluster C_m without any subsequent clusters and cluster C_n without any preceding clusters are merged into a new cluster C'_m.

3.2. T-AII with Multiple Attributes. Based on the online trajectory clustering procedure, the intended routes and relevance of routes can be derived from the cluster set. Then, intent inference step is implemented through comparing the probability summation of all flight attributes referencing to each candidate intended route. The notation and definition of flight attributes are given in Table 1. Based on the temporal property of the data, the attributes can be classified into static and dynamic categories. Moreover, the attributes can also be labeled as discrete data and continuous data.

For the discrete attributes including CS, DA, AS, AC, and RR, the probability is derived from its frequency in the data set defined as follows:

$$P_i^k = \frac{\text{count}\left(A_i^k\right)}{\text{sum}_j\left(\text{count}\left(A_i^j\right)\right)}, \tag{5}$$

where P_i^k is the probability of attribute i when it takes value of k. The number of trajectory points associated to the attribute is derived from function count. Note that the probability of RR is defined as a conditional probability in (6), where the subsequent and preceding intended routes are denoted by C_m and C_n separately. The number of trajectory points associated to the clusters is counted based on the online trajectory clustering procedure:

$$P_{\text{RR}}^{m,n} = \frac{\text{count}\left(C_m, C_n\right)}{\text{count}\left(C_n\right)}. \tag{6}$$

For the continuous attribute HA, the Gaussian random variable with zero mean and standard variance in [6] is adopted here, where φ_k is the angular difference between aircraft heading direction and direction to ending point of intended route denoted by C_k:

$$P_{\text{HA}}^k = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{\varphi_k^2}{2\sigma^2}\right). \tag{7}$$

In order to determine the intended route for trajectory point T_i^j of trajectory i at time instant j, the intended route set as the subset of cluster set $\mathbf{C}_I \subset \mathbf{C}$ are investigated. For each intended route $C_k \in \mathbf{C}_I, k = 1, \ldots, I_k$, the corresponding probabilistic summation is computed through the following

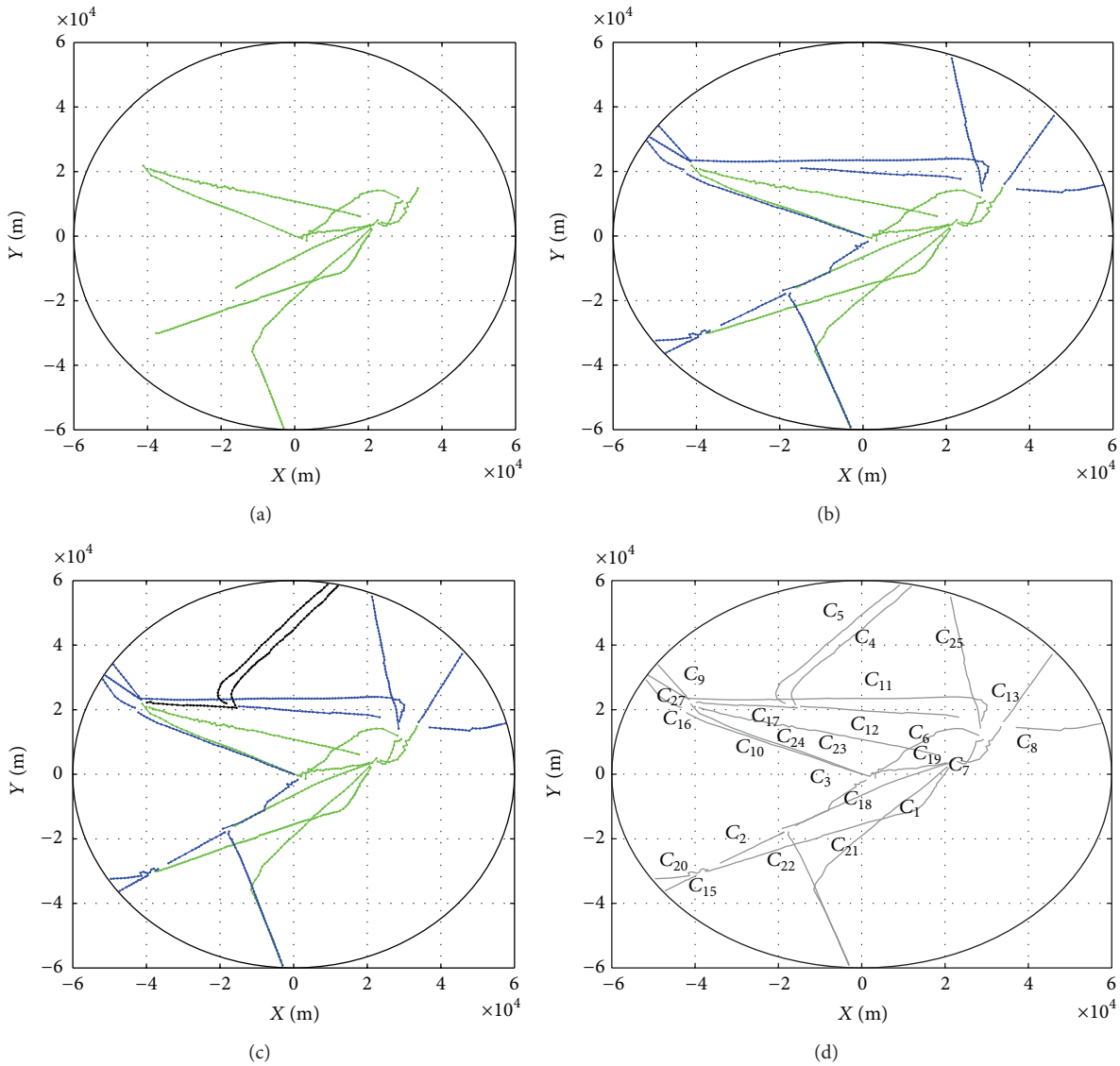


Figure 6: The evolution of clustering process for departure trajectories.

logarithmic format defined in (8), where $\omega_1, \ldots, \omega_6$ are the weight factors of attributes:

$$\begin{aligned} P\left(C_k \mid T_i^j\right) &= \ln\left[\left(P_{\mathrm{CS}}\right)^{\omega_1} \cdot \left(P_{\mathrm{DA}}\right)^{\omega_2} \cdot \left(P_{\mathrm{AS}}\right)^{\omega_3} \cdot \left(P_{\mathrm{AC}}\right)^{\omega_4} \cdot \left(P_{\mathrm{RR}}\right)^{\omega_5} \cdot \left(P_{\mathrm{HA}}\right)^{\omega_6}\right] \\ &= \omega_1 \ln P_{\mathrm{CS}} + \omega_2 \ln P_{\mathrm{DA}} + \omega_3 \ln P_{\mathrm{AS}} + \omega_4 \ln P_{\mathrm{AS}} + \omega_5 \ln P_{\mathrm{RR}} + \omega_6 \ln P_{\mathrm{HA}}. \end{aligned} \tag{8}$$

Finally, the inferred intended route for trajectory point denoted by k^* is recognized based on

$$k^* = \arg\max_{k=1,\ldots,I_k} P\left(C_k \mid T_i^j\right). \tag{9}$$

4. Experimental Results

4.1. Data Set. In the following experiments, radar trajectory and flight attribute data of 34 days from December 18, 2012, to January 20, 2013, collected from Chengdu terminal area are used. The radar trajectory data are limited in a circular area with radius 60 km centered in the position of Chengdu Shuangliu International Airport. The data consist of flight date and time, call sign, longitude, latitude, altitude, and ground speed with an update rate ranging from 1 to 5 seconds. Total 20854 primitive departure and arrival trajectories are collected. Based on the data preprocessing to eliminate some missing and fault data, 8995 departure valid trajectories with more than 250 trajectory points of each trajectory are used to verify the performance of the proposed approach. Note that rectangular coordinate system is applied in our experiments originated in the position of airport center. It is also worth pointing out that the same data set should be applied in the

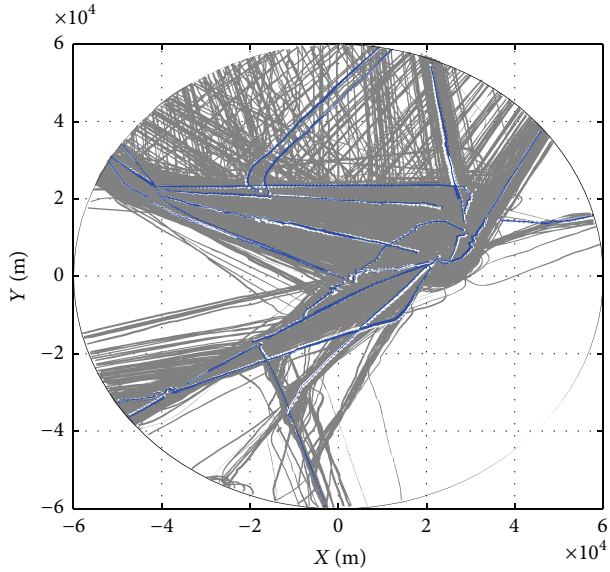


Figure 7: Clustering results for departure trajectories are represented by blue lines. The radar trajectories are marked with grey lines.

whole validation process due to the continuity of the intent modeling and intent inference steps.

Flight attribute data are derived from the air traffic flow management system in Chengdu terminal area. The information contains flight date and time, call sign, destination airports, and aircraft type. Flight time and call sign are the keys to correlate the radar trajectory data. Here, 552 departure call signs are considered. The destination airports are divided into 7 categories based on the airport code and geographic information including northeast, north, northwest, southeast, east, southwest, and middle. Note that the international flights are classified based on their entering positions in the terminal area. Two kinds of aircraft size (heavy and medium) and three kinds of aircraft approach categories (B, C, and D) are considered.

4.2. Online Trajectory Clustering. Figure 6 reports the evolution of clustering process for departure trajectories. Plot (a) in Figure 6 shows the centroids of clusters starting from the runway represented by green lines. Subsequent clusters represented by blue and black lines are shown in plots (b) and (c). Plot (d) gives the final clustering results where the cluster is denoted by the original index, such as C_1 and C_{27}. Note that clusters C_{14} and C_{26} have been deleted during the clustering process so that total 25 clusters are maintained. Furthermore, the clustering results are compared with the radar trajectories shown in Figure 7. In general, the clusters reflect the main routes. Note that some trajectories in the upper part of the figure are determined as outliers due to the low frequency in the data set.

We then investigate the available routes in different air traffic situations with 5 experiments shown in Figure 8. For each experiment, 2000 trajectories are selected randomly among the departure trajectories as the input to the clustering procedure. Clustering results are presented in plots (a), (b), (c), (d), and (e). Note that plot (b) shows the air traffic situation in which routes to the positive direction of X coordinate are rarely used. The situation shown in plot (d) reflects that flights are distributed in all directions.

4.3. T-AII with Multiple Attributes. By using the same trajectory data set and intended routes obtained in Section 4, eight main departure routes as shown in Figure 9 are selected as intended route set $\mathbf{C}_I$ to verify the T-AII approach, including C_{18}, C_{24}, C_2, C_{22}, C_9, C_{27}, C_{15}, and C_{20}. These routes are organized in a binary-tree structure consisting of three intent inference scenarios shown in Figure 10. When aircraft takes off from the runway, the phase is marked as START state. The T-AII is triggered when the aircraft enters either route C_{18} or route C_{24}. For example, based on trajectory data and flight attributes information collected in route C_{18}, the first T-AII is to infer the likely intended routes (C_2 or C_{22}). Then, the second T-AII is followed if the aircraft is flying along route C_2 in which C_{15} and C_{20} are the possible intended routes. The third T-AII scenario refers to the case determining the possible route (C_9 or C_{27}) for aircraft in route C_{24}.

Table 2 reports the intent inference accuracy of T-AII approach in three different cases (with only dynamic attributes, with only static attributes, and with both dynamic and static attributes), through adopting the following weight factors: $\{0, 0, 0, 0, 0.5, 0.5\}$, $\{0.25, 0.25, 0.25, 0.25, 0, 0\}$, and $\{0.17, 0.17, 0.17, 0.17, 0.17, 0.17\}$ separately. Note that, in the maneuvering scenarios (turning), such as "C_{18} to C_{22}", "C_{24} to C_9", and "C_2 to C_{20}," the T-AII with only dynamic attributes results in quite low inference accuracy due to the ineffectiveness of dynamic attributes. This is because the dynamic attributes only contain short-term motional feature of aircraft. On the contrary, T-AII approach with only static attributes gains very high inference accuracy. One of the possible reasons is that static attributes consist of the long-term periodical characteristics for commercial flights, which are more efficient in the maneuvering scenario.

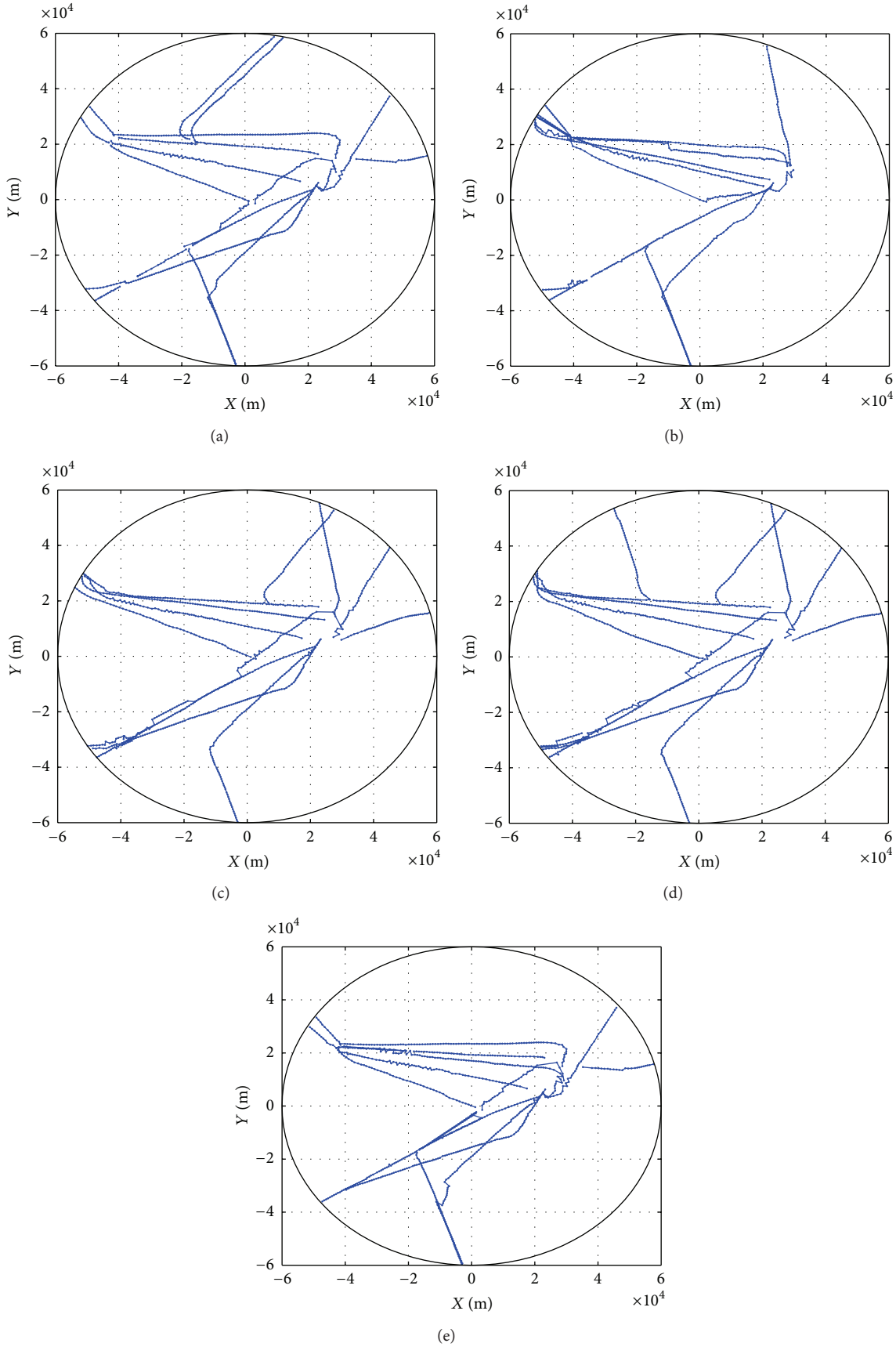


Figure 8: Clustering results for different air traffic situations.

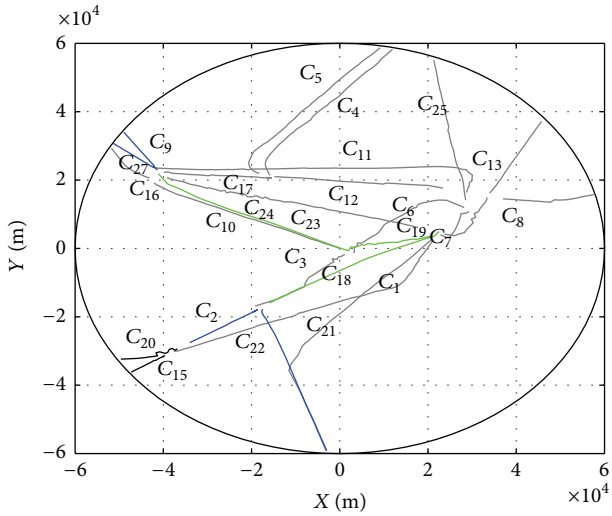


Figure 9: Intended route set represented by green, blue, and black lines.

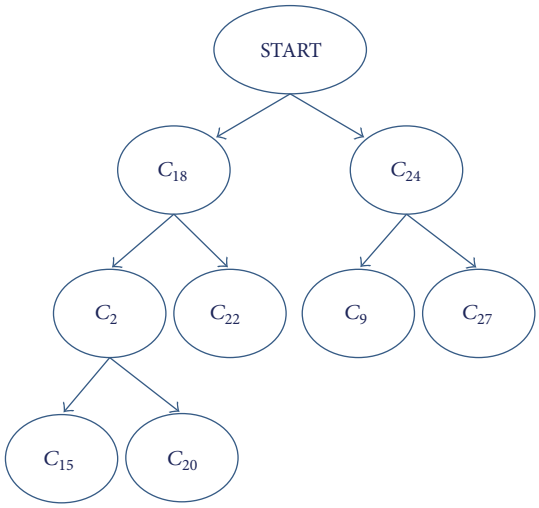


Figure 10: Structure of intended routes.

Furthermore, the impact of each static attribute on inference accuracy is compared by independently applying the T-AII approach with each static attribute. Take the first scenario including "C_{18} to C_2" and "C_{18} to C_{22}" as an example. For the case "C_{18} to C_2," the lowest and the highest inference accuracy are 98.42% and 99.89% obtained from T-AII with DA and AS attributes, respectively. It can be seen that there is no obvious difference for each static attribute. However, the inference results of case "C_{18} to C_{22}" are 87.40%, 20.18%, 1.30%, and 1.30% for the T-AII with CS, DA, AS, and AC, respectively. Here, the approaches with attributes AS and AC can gain the same accuracy level as that obtained with dynamic attributes shown in Table 2. On the contrary, the results by using attributes CS and DA are much better. One of the possible reasons is that the geographically related attributes DA and CS strongly affect the T-AII results in the turning maneuver case such as "C_{18} to C_{22}."

Table 2: Inference accuracy of AII approach in three different cases.

Scenarios	T-AII with only dynamic attributes	T-AII with only static attributes	T-AII with both static and dynamic attributes
1			
C_{18} to C_2	99.89%	99.86%	99.87%
C_{18} to C_{22}	1.30%	87.18%	87.16%
2			
C_{24} to C_9	0.04%	84.02%	78.65%
C_{24} to C_{27}	99.94%	99.60%	99.60%
3			
C_2 to C_{15}	99.78%	79.75%	79.75%
C_2 to C_{20}	0.45%	80.84%	80.78%

Table 3: Confusion matrix in scenario 1.

Scenario 1	C_2 (inferred)	C_{22} (inferred)
T-AII with only dynamic attributes		
C_2 (actual)	294147	319
C_{22} (actual)	13488	177
T-AII with only static attributes		
C_2 (actual)	294069	397
C_{22} (actual)	1752	11913
T-AII with both static and dynamic attributes		
C_2 (actual)	294069	397
C_{22} (actual)	1755	11910

Then, confusion matrix in the first scenario is further investigated as shown in Table 3. It can be seen that most of trajectory points are associated with the straight route from C_{18} to C_2 resulting in the high probability values for the two dynamic attributes (RR and HA) associated to "C_{18} to C_2." As a result, 13488 trajectory points are incorrectly determined to C_2 rather than C_{22} based on T-AII approach with only dynamic attributes. However, when the static attributes are considered, the corresponding number of incorrect inferences is decreased to 1752.

5. Conclusion

This paper has introduced a two-step T-AII approach that integrated intent modeling and intent inference steps so as to address the uncertainties of air traffic situation in terminal area, especially the undefined flight routes and frequent maneuvers. The online trajectory clustering procedure is designed to recognize the real-time available routes merely based on radar trajectory rather than the predefined flight plan. Then, the T-AII approach with both dynamic and static attributes is proposed in which the corresponding inference accuracy in maneuvering scenario is increased. Although the results of the proposed approach are promising based on the verification with real radar trajectory and flight attributes data of 34 days collected from Chengdu terminal area, some

of the directions need to be further addressed in the future. For example, the outliers in the clustering procedure should be investigated in the perspective of air traffic controller. Also, more attributes related to flight should be discussed under the context of T-AII.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Science Foundation for Young Scientists of China (Grants no. 61401011 and no. 61201314), the National ATC Scientific Research Project (Grant no. GKG201404004), and the National Key Technologies R&D Program of China (Grant no. 2012BAG04B01).

References

[1] J. K. Kuchar and L. C. Yang, "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 4, pp. 179–189, 2000.

[2] EUROCONTROL, "SESAR consortium D3 the ATM target concept," 2007.

[3] Joint Planning and Development Office, "Concept of operations for the Next Generation Air Transportation System," version 2.0, 2007.

[4] J. Krozel and A. Dominick, "Intent inference and strategic path prediction," in *Proceedings of the AIAA Guidance, Control, and Dynamics Conference and Exhibit*, San Francisco, Calif, USA, August 2005.

[5] J. Krozel and A. Dominick, "Intent inference with path prediction," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 2, pp. 225–236, 2006.

[6] J. L. Yepes, I. Hwang, and M. Rotea, "New algorithms for aircraft intent inference and trajectory prediction," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 2, pp. 370–382, 2007.

[7] I. Hwang and C. Eng Seah, "Intent-based probabilistic conflict detection for the next generation air transportation system," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 2040–2059, 2008.

[8] W. Liu and I. Hwang, "Probabilistic trajectory prediction and conflict detection for air traffic control," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 6, pp. 1779–1789, 2011.

[9] I. Lymperopoulos and J. Lygeros, "Sequential Monte Carlo methods for multi-aircraft trajectory prediction in air traffic management," *International Journal of Adaptive Control and Signal Processing*, vol. 24, no. 10, pp. 830–849, 2010.

[10] H. Tang, J. Robinson, and D. Denery, "Tactical conflict detection in terminal airspace," in *Proceedings of the 10th AIAA Aviation Technology, Integration, and Operations Conference (ATIO '10)*, Fort Worth, Tex, USA, September 2010.

[11] H. Tang, J. E. Robinson, and D. G. Denery, "Tactical conflict detection in terminal airspace," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 2, pp. 403–413, 2011.

[12] T. Kozon, A. Farrahi, S. Verma, H. Tang, and D. Ballinger, "Phase-2 evaluation of a tactical conflict detection tool in the terminal area," in *Proceedings of the 31st Digital Avionics Systems Conference*, Williamsburg, Va, USA, October 2012.

[13] B. Hoshstrasser and N. Geddes, "OPAL: operator intent inferencing for intelligent operator support systems," in *Proceedings of the IJCAI-89 Workshop on Integrated Human-Machine Intelligence in Aerospace Systems*, 1989.

[14] N. D. Geddes, "A model for intent interpretation for multiple agents with conflicts," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 3, San Antonio, Tex, USA, 1994.

[15] D. A. Thurman, A. R. Chappell, and C. M. Mitchell, "An enhanced architecture for OFMspert: a domain-independent system for intent inferencing," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 1, pp. 955–960, October 1998.

[16] H. Kautz and J. Allen, "Generalized plan recognition," in *Proceedings of the 5th National Conference on Artificial Intelligence*, vol. 1, pp. 32–37, 1986.

[17] M. Tambe and P. S. Rosenbloom, "Event tracking in a dynamic multiagent environment," *Computational Intelligence*, vol. 12, no. 3, pp. 499–521, 1996.

[18] T. G. Reynolds, J. M. Histon, H. J. Davison, and R. J. Hansman, "Structure, intent & conformance monitoring in ATC," in *Proceedings of the Air Traffic Management Workshop on ATM System Architectures and CNS Technologies (ATM '02)*, Capri, Italy, September 2002.

[19] J. DeArmon, A. Mahashabde, C. Shiotsuki, and K. Amefia, "Developing a terminal path library for system-wide environmental impact assessment of aviation activity," in *Proceedings of the 12th AIAA Aviation Technology, Integration, and Operations (Atio) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Indianapolis, Ind, USA, September 2012.

[20] C. Jesse, H. Liu, E. Smart, and D. Brown, "Analysing flight data using clustering methods," in *Proceedings of the 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, Part I*, pp. 733–740, 2008.

[21] A. Eckstein, "Automated flight track taxonomy for measuring benefits from performance based navigation," in *Proceedings of the Integrated Communications, Navigation and Surveillance Conference*, Valencia, Spain, April 2009.

[22] F. Rehm, "Clustering of flight tracks," in *Proceedings of the AIAA Infotech@Aerospace*, Atlanta, Ga, USA, April 2010.

[23] M. Gariel, A. N. Srivastava, and E. Feron, "Trajectory clustering and an application to airspace monitoring," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1511–1524, 2011.

[24] K. Leiden and S. Atkins, "Trajectory clustering for metroplex operations," in *Proceedings of the 11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, Va, USA, September 2011.

[25] E. Smart, *Detecting abnormalities in aircraft flight data and ranking their impact on the flight [Ph.D. thesis]*, University of Portsmouth, 2011.

[26] R. Annoni Jr. and C. H. Q. Forster, "Analysis of aircraft trajectories using Fourier descriptors and kernel density estimation," in *Proceedings of the 15th International IEEE Conference on Intelligent Transportation Systems (ITSC '12)*, pp. 1441–1446, Anchorage, Alaska, USA, September 2012.

[27] M. Enriquez, "Identifying temporally persistent flows in the terminal airspace via spectral clustering," in *Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar (ATM '13)*, Chicago, Ill, USA, June 2013.

[28] C. Piciarelli and G. L. Foresti, "Toward event recognition using dynamic trajectory analysis and prediction," in *Proceedings of the IEE International Symposium on Imaging for Crime Detection and Prevention (ICDP '05)*, pp. 131–134, 2005.

[29] C. Piciarelli and G. L. Foresti, "On-line trajectory clustering for anomalous events detection," *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1835–1842, 2006.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 302615, 9 pages
http://dx.doi.org/10.1155/2015/302615

Research Article

An Airway Network Flow Assignment Approach Based on an Efficient Multiobjective Optimization Framework

Xiangmin Guan,[1,2,3] Xuejun Zhang,[1,2] Yanbo Zhu,[1,2] Dengfeng Sun,[4] and Jiaxing Lei[1,2]

[1] *School of Electronic and Information Engineering, Beihang University, Beijing 100191, China*
[2] *National Key Laboratory of CNS/ATM, Beijing 100191, China*
[3] *Beijing Key Laboratory for Cooperative Vehicle Infrastructure Systems and Safety Control, Beijing 100191, China*
[4] *School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907-2023, USA*

Correspondence should be addressed to Xuejun Zhang; zhxj@buaa.edu.cn

Received 23 July 2014; Revised 7 September 2014; Accepted 18 September 2014

Academic Editor: Zheng Zheng

Considering reducing the airspace congestion and the flight delay simultaneously, this paper formulates the airway network flow assignment (ANFA) problem as a multiobjective optimization model and presents a new multiobjective optimization framework to solve it. Firstly, an effective multi-island parallel evolution algorithm with multiple evolution populations is employed to improve the optimization capability. Secondly, the nondominated sorting genetic algorithm II is applied for each population. In addition, a cooperative coevolution algorithm is adapted to divide the ANFA problem into several low-dimensional biobjective optimization problems which are easier to deal with. Finally, in order to maintain the diversity of solutions and to avoid prematurity, a dynamic adjustment operator based on solution congestion degree is specifically designed for the ANFA problem. Simulation results using the real traffic data from China air route network and daily flight plans demonstrate that the proposed approach can improve the solution quality effectively, showing superiority to the existing approaches such as the multiobjective genetic algorithm, the well-known multiobjective evolutionary algorithm based on decomposition, and a cooperative coevolution multiobjective algorithm as well as other parallel evolution algorithms with different migration topology.

1. Introduction

With the development of civil aviation, the permanently increasing air traffic and the limited airspace resource have resulted in more and more serious congestion and flight delay [1–3]. Meantime, heavy congestion challenges the airspace safety and the flight delay costs the airline industry billions of dollars every year [4]. Hence, how to safely accommodate high levels of demand and to maximize the use of capacity-limited airspace and airport resources has become a major concern to both researchers and practitioners in air traffic management (ATM).

In recent years, FAA proposed the 4D-trajectory (4DT) operation concept in next generation air transportation system (NextGen), which includes three dimensions of position and a time description of the flight. Under the 4DT environment, flights can be accurately planned in both space and time. The airway network flow assignment (ANFA) approach can provide solutions from a global point of view with the aim of alleviating airspace congestion and reducing time cost by optimizing the 4D trajectories of all flights considered in the entire airspace. Therefore, it has become a focus of research interests.

However, the ANFA problem is a large-scale combinatorial optimization problem with complicated constraints as well as tightly coupled variables, which is difficult to solve in general. For instance, there are more than ten thousand flights flying over China every day on the air route network with more than one thousand waypoints, which generates a large number of tightly coupled decision variables and constraints. Hence, in order to get optimal flight plans for all flights, the ANFA problem has a very high computational complex. Besides, with consideration of reducing congestion and minimizing the induced delay, the problem usually has

multiple objectives which are all nondifferentiable or even noncontinuous.

Due to the importance of the ANFA problem, it has drawn a mass of attention of researchers [5]. Earlier work has focused on the slot-time allocation, known as ground holding in a single or multiple airport setting, in which delays propagate through the network [6, 7]. Besides, the flight level allocation has been considered to the flights in this problem [8, 9]. In addition, Bertsimas et al. [10, 11] considered the time slots and routes assignment simultaneously through an efficient deterministic approach. From a stochastic optimization point of view, Delahaye and Odoni proposed a genetic algorithm to solve the problem [12]. However, it often falls into the local optimum because of the large-scale decision variables. More recently, a cooperative coevolution (CC) algorithm was introduced into the resolution of the ANFA problem [13]. It adopted the divide-and-conquer strategy to divide the complex problem into several low-dimensional subproblems which becomes easier to deal with.

So far these works took the minimization of the airspace congestion or the flight delay as the sole objective [14, 15]. However, it might be more appropriate to consider both airspace congestion and extra flight cost and try to seek a good trade-off between them. Hence, Daniel et al. [16] formulated the problem into a biobjective optimization problem and solved it with the multiobjective genetic algorithm (MOGA). However, the application of existing multiobjective evolutionary algorithms (MOEAs) to the ANFA problem might lead to poor performance due to the scale of the problem. To the best of our knowledge, few works could effectively solve complicated multiobjective optimization problems with thousands of variables.

With the consideration of the reduction of the airspace congestion and the flight delay simultaneously, this paper formulates the ANFA problem as a multiobjective optimization model. Then, an efficient multiobjective optimization framework is presented to solve it. This framework employs the parallel computation and population diversity adjustment techniques to improve the optimization capability. Firstly, an effective multi-island parallel evolution algorithm (PEA) with multiple evolution populations is adopted. Besides, one-way ring migration topology is applied to exchange individuals among populations to improve the efficiency of the cooperation of populations. Secondly, the multiobjective evolutionary algorithm NSGA2 is used to optimize each population. In addition, a cooperative coevolution algorithm is adapted to divide the ANFA problem into several low-dimensional biobjective optimization problems which are easier to deal with. Finally, in order to maintain the diversity of solutions and avoid prematurity, a dynamic adjustment operator based on solution congestion degree is specifically designed. It can greatly improve the distribution of the nondominated solutions in the archive and maintain the population diversity by inheriting the nondominated solutions with low congestion degree in a high probability. Simulation results using the real traffic data from China air route network and daily flight plans demonstrate that the proposed approach can improve the solution quality effectively, showing superiority to the existing approaches such as the multiobjective genetic algorithm, the well-known multiobjective evolutionary algorithm based on decomposition (MOEA/D), and a CC-based multiobjective algorithm as well as other parallel evolution algorithms with different migration topology.

The rest of this paper is organized as follows. Section 2 introduces the formulation of the investigated biobjective ANFA problem. Section 3 describes the proposed multiobjective optimization framework in detail. Experimental study is presented in Section 4 to evaluate the effectiveness of our algorithm. Finally, Section 5 concludes this paper and discusses directions for further research.

2. Problem Formulation

The airway network can be modeled as a directed graph including the waypoints as nodes and segments as edges. Each flight can be considered as a single commodity on the network with a defined pair of origin-destination nodes. Besides, each flight has its predefined departure time slots and the optional routes. In this paper, with the consideration of safety and efficiency, the ANFA problem is formulated as a biobjective problem to reduce the airspace congestion and the total flight delay simultaneously.

Each flight (flight k) is associated with a pair of decision variables (r_k, τ_k), where r_k is a possible route and τ_k is a feasible departure time slot. In addition, they are subject to the following:

(1) $r_k \in \text{Path}_k = \{r'_1, r'_2, \ldots, r'_{\max}\}$, where Pathk is the set of all possible paths of flight k;

(2) $t_k^{\min} \leq \tau_k \leq t_k^{\max}$, where $t_k^{\max}$ is the latest departure time slot and $t_k^{\min}$ is the earliest departure time slot.

We can see that constraint (1) denotes a feasible route of each flight from a defined route set, while constraint (2) enforces the flight to depart at a predefined departure time slot.

The first objective function is the minimization of the airspace congestion. Here the workload of the sectors is used to indicate the airspace congestion, which is defined as follows.

It is generally known that the workload of a sector mainly depends on the monitoring workload and the coordination workload. Hence, the total workload of a sector S_k at time t can be roughly expressed by the following [16]:

$$W_{S_k}^t = W_{\text{mo}S_k}^t + W_{\text{co}S_k}^t, \tag{1}$$

where $W_{\text{mo}S_k}^t$ is the monitoring workload and can be numerically estimated by

$$W_{\text{mo}s_k} = \begin{cases} 1 + M_{S_k}^t - C_{mS_k}^t, & \text{if } M_{S_k}^t > C_{mS_k}^t \\ 0, & \text{else,} \end{cases} \tag{2}$$

where $M_{S_k}^t$ is related to the number of aircraft in sector S_k at time t and $C_{mS_k}^t$ is the monitoring critical capacity of the sector at time t.

```
BEGIN
Initialize M populations of size ps each. g = 0. Archive = NULL
WHILE g < max gen
    FOR population j = 0 : (M − 1)
        Divide the problem into n subcomponents based on cooperative co-evolution
        FOR each subcomponent
            Use the NSGA2 framework and differential evolution as the evolutionary
            algorithm to solve the subcomponent
        END FOR
        Archive = Obtain and update the non-dominated solutions based on the dynamic
        adjustment operator for solutions congestion degree
    END FOR
    IF (migration condition meet)
        Exchange individuals selected among populations based on a migration topology
    END IF
    g = g + 1
END WHILE
END
```

Algorithm 1: The framework of the proposed algorithm.

Besides, $W^t_{\mathrm{co}S_k}$ is the coordination workload and can be numerically estimated by

$$W_{\mathrm{co}S_k} = \begin{cases} 1 + C^t_{S_k} - C^t_{cS_k}, & \text{if } C^t_{S_k} > C^t_{cS_k} \\ 0, & \text{else,} \end{cases} \tag{3}$$

where $C^t_{S_k}$ is related to the flights passing through the boundary of sector S_k at time t and $C^t_{S_k}$ is related to the number of aircraft passing the boundaries of sector S_k at time t.

The goal of the optimization is to minimize the airspace congestion via spreading the congestion over several sectors. Hence, the objective is defined by the following [16]:

$$\min AC = \sum_{k=1}^{k=n_S} \left(\left(\sum_{t\in T} W^t_{S_k} \right)^{1-\varphi} \times \left(\max_{t\in T} W^t_{S_k} \right)^{\varphi} \right), \tag{4}$$

where $\varphi, (1-\varphi) \in [0,1]$ indicates the relative importance of the maximum congestion and the average congestion.

The second objective is the minimization of the extra flight cost (EFC) which includes the departure delay and the airborne delay caused by choosing a longer path than the shortest one. Then the second objective can be expressed as follows [16]:

$$\min \mathrm{EFC} = \sum_{i\in F} \left(\left| \tau_i - t^{\mathrm{prf}}_k \right| + \left(t_{r_i} - t_{r^{\mathrm{prf}}_i} \right) \right)^2, \tag{5}$$

where t_{r_i} and $t_{r^{\mathrm{prf}}_i}$ denote the flight time of r_i and the preferred shortest path, respectively, and t^{prf}_k indicates the preferred departure time. The first part is the airborne delay, and the second part is the ground delay.

3. Optimization Method

It can be seen that the ANFA problem is a large-scale combinatorial optimization problem, and the objective functions are nonlinear and nondifferentiable. Black-box optimization methods such as EAs appeared to be promising to deal with this kind of problem [15–19]. Given the multiobjective model of the ANFA problem, basically any existing MOEA for discrete multiobjective optimization can be readily utilized. However, the ANFA problem involves thousands of tightly coupled decision variables which are difficult to solve. In addition, few existing MOEAs have been extensively investigated on problems of such a large scale while the scalability of EAs with respect to the number of decision variables is in general deemed to be poor; therefore a direct application of existing MOEAs might not obtain satisfactory solutions.

With the aim of avoiding prematurity and improving the convergence rate of this complex problem, a new multiobjective optimization framework is proposed. Firstly, an effective multi-island parallel evolution algorithm with multiple evolution populations is employed and one-way ring migration topology for exchange individuals among populations is applied to improve the efficiency of the cooperation of populations. Secondly, the multiobjective evolutionary algorithm NSGA2 is used to optimize each population. In addition, this paper introduces the idea of cooperative coevolutionary (CC) into the resolution of the multiobjective ANFA problems. The main idea is to divide the high-dimensional problem into low-dimensional subcomponents. The subcomponents work cooperatively to obtain better solutions. Finally, all the nondominated solutions are stored in an archive. As the ANFA problem is of such a large scale with thousands of variables, the main difficulty is how to maintain the diversity of the population to get better solutions and to improve the distribution of the solutions in the archive. Hence, in order to maintain the diversity of solutions and to avoid prematurity, a dynamic adjustment operator based on solution congestion degree is specifically designed for the ANFA problem. The framework is presented in Algorithm 1.

3.1. Multi-Island Parallel Evolution Algorithm. As parallel computers become more commonplace in scientific computing, it becomes more feasible to harness their power for use with evolutionary algorithms (EAs) [20]. Multi-island PEAs consist of several populations, which can optimize simultaneously to avoid premature convergence. They have been successfully applied to find acceptable solutions to problems in different engineering domains [21, 22].

Suppose that there are M islands and N flights. Then each population can be denoted as

$$\mathrm{pop}_i = \left\{idiv_{i1}, idiv_{i2}, \ldots, idiv_{ips}\right\}, \quad 1 \le i \le M, \tag{6}$$

where $idiv_{ij}$ is defined by

$$idiv_{ij} = \left(r_{ij1}, \tau_{ij1}, r_{ij2}, \tau_{ij2}, \ldots, r_{ijN}, \tau_{ijN}\right), \quad 1 \le j \le \mathrm{ps}, \tag{7}$$

where ps is the size of population.

3.2. Migration Topology. The migration topology is a key feature of the island model which determines the destination of the migrants, and it could greatly affect the quality of the solutions and the efficiency of algorithms. If two populations rarely communicate with each other, it is difficult for the best solution to spread which may prevent populations finding better solutions.

Currently, the main migration topologies are the one-way ring topology and the random topology [22]. The random topology delivers the migrants to a randomly selected population. However, it may result in inefficient communication. After many times of migration under the random topology, the best solution cannot be spread effectively among populations. In the one-way ring topology, populations are numbered, and the worst individual of a population is replaced by the best individual of the next population. The one-way ring topology can provide sufficient communication among populations and maintain the population diversity, which can effectively avoid premature convergence and improve the optimization capability.

3.3. Cooperative Coevolution for Each Population. Though the multi-island PEA uses several populations simultaneously, in fact each population is hard to avoid falling into a local optimum because the ANFA problem involves large-scale tightly coupled decision variables as presented in the previous sections. Hence, we introduce the cooperative coevolution algorithm for the optimization of each population to further improve the solution quality. The cooperative coevolution (CC) algorithm, adopting the divide-and-conquer strategy, divides the complex problem into several low-dimensional subproblems [23–25]. There are two critical issues in this approach.

(1) Decomposition Strategy. The decomposition strategy is a key feature of the cooperative coevolution framework which can greatly affect the capability and the efficiency of algorithms. In this work, the random grouping strategy is used which has been both theoretically and experimentally proved to be effective for the large-scale complex problem [23]. At each generation, each population is randomly divided into ns disjoint subpopulations with the same population size:

$$\begin{gathered}
\mathrm{pop}_i = \left\{\mathrm{sp}_i^1, \ldots, \mathrm{sp}_i^{ns}\right\}, \quad 1 \le i \le M \\
\mathrm{sp}_i^j = \left\{sid_i^{j1}, sid_i^{j2}, \ldots, sid_i^{j\mathrm{ps}}\right\}, \quad 1 \le j \le ns \\
sid_i^{jk} = \left(r'_{jk1}, \tau'_{jk1}, r'_{jk2}, \tau'_{jk2}, \ldots, r'_{jk(N/ns)}, \tau'_{jk(N/ns)}\right), \\
1 \le j \le \mathrm{ps},
\end{gathered} \tag{8}$$

where sid denotes the individual of each subpopulation and sp_i^j indicates the subpopulation j of population i.

(2) Subpopulation Optimization. Another critical point is optimization for each subpopulation. In this paper, the well-known multiobjective evolutionary algorithm NSGA2 is employed by each subpopulation [26]. Besides, differential evolution (DE) [27] is used in the framework to generate new solutions, because it is a simple yet effective algorithm for global optimization. DE is a randomized parallel searching algorithm. It begins with a random population, according to specific rules, for example, selection, crossover, and mutation. An optimized resolution is reached by retaining good individuals and discarding bad individuals. Compared with other optimization algorithms, DE has the advantages in global optimization as well as easy operation. Its operators are described below.

Suppose that $f(x)$ is the objective function and the goal is to minimize it.

In mutation, if current chromosome is $x_{i,G}$, then choose three different chromosomes from current generation population named $x_{r1,G}$, $x_{r2,G}$, and $x_{r3,G}$. The mutation operator is defined by

$$V_{i,G+1} = x_{r1,G} + F \cdot \left(x_{r2,G} - x_{r3,G}\right), \quad F \in [0, 2], \tag{9}$$

where F is a parameter which decides the scale of mutation. A new chromosome is generated by the crossover of $V_{i,G+1}$ and $x_{i,G}$ as follows:

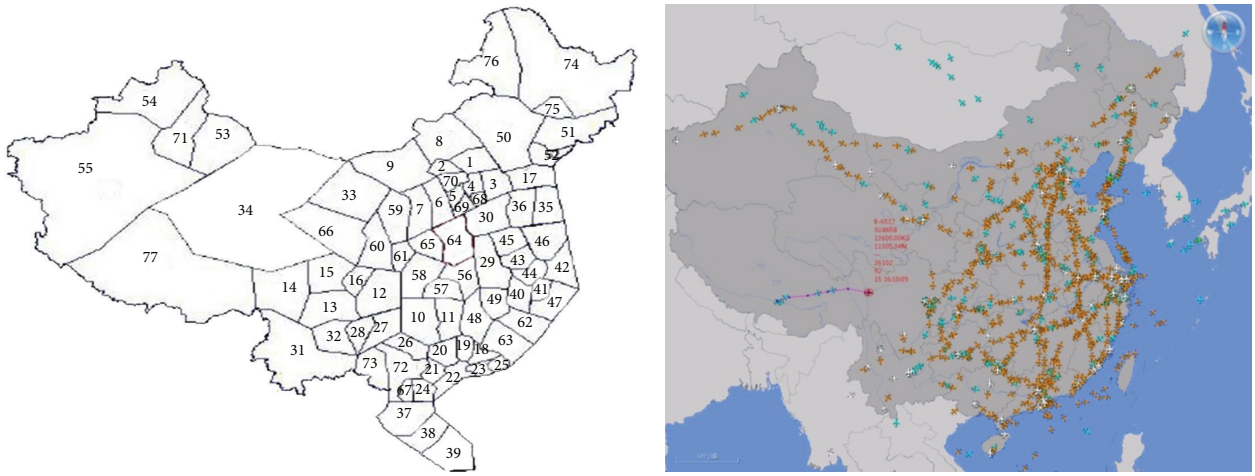
$$\begin{gathered}
U_{i,G+1} = \left(u_{1i,G+1}, u_{2i,G+1}, \ldots, u_{Ni,G+1}\right) \\
u_{j,i,G+1} = \begin{cases} v_{ji,G+1}, & \mathrm{rand}\left(j\right) \le CR \vee j = \mathrm{rnbr}\left(i\right) \\ x_{ji,G}, & \text{else}, \end{cases}
\end{gathered} \tag{10}$$

where rnbr(i) is a random integer number between 1 and N, which ensures that $U_{i,G+1}$ gets at least one component from $v_{i,G+1}$ component from $V_{i,G+1}$ and rand(j) is a uniformly distributed random number between 0 and 1.

After the evaluation of each chromosome, the chromosomes of next generation are chosen according to the following rule:

$$x_{i,G+1} = \begin{cases} u_{i,G+1}, & f\left(u_{i,G+1}\right) \le f\left(x_{i,G}\right) \\ x_{i,G}, & \text{else}. \end{cases} \tag{11}$$

3.4. Dynamic Adjustment Operator Based on Solution Congestion Degree. As the optimization generation increases,



(a) (b)

Figure 1: (a) Airspace sectors in China. (b) Flights operation in China.

the population diversity decreases rapidly for more solutions congested at a local searching space [28, 29]. On the one hand, too crowded solutions will cause premature convergence. On the other hand, some local searching space with sparse solutions is not explored enough and needs more attention for better solutions. Hence, the more evenly the nondominated solutions distribute, the better the optimization is. In this paper, we propose a dynamic adjustment operator to improve the distribution of solutions and maintain the population diversity based on the congestion degree of solutions.

(1) Congestion Degree of the Nondominated Solutions in the Archive. The distance between solution i and j in archive is defined by

$$d_{i,j} = \frac{1}{2}\sqrt{\left(P_i^{\text{fit}[1]} - P_j^{\text{fit}[1]}\right)^2 + \left(P_i^{\text{fit}[2]} - P_j^{\text{fit}[2]}\right)^2}, \tag{12}$$

where $p_i^{\text{fit}[1]}$ and $p_i^{\text{fit}[2]}$ denote the value of the first and the second objective functions of solution i, respectively. Besides, the average distance of all solution pairs in the archive is described by

$$s = \frac{2}{n(n-1)}\sum_{i=1}^{i=n}\sum_{j=i+1}^{j=n} d_{i,j}, \tag{13}$$

where n is the number of the solutions in archive.

Then the congestion degree of a nondominated solution can be expressed as

$$d_i = \frac{1}{n-1}\sum_{j=1, j\neq i}^{j=n} \delta_{i,j}, \tag{14}$$

where

$$\delta_{i,j} = \begin{cases} 1 & \text{if } d_{i,j} \leq \gamma \cdot s \\ 0 & \text{if } d_{i,j} > \gamma \cdot s, \end{cases} \tag{15}$$

where γ is a parameter which can be predefined. We can see that the larger d_i is, the closer to other solutions solution i is.

(2) Update the Archive. The solutions in the archive will be constantly updated during the evolution of the subpopulations. When the number of the nondominated solutions exceeds the maximum size of the archive, the solutions i will be moved out from the archive in a probability

$$p_i = \frac{d_i^{\alpha}}{\sum_{j=1}^{n} d_j^{\alpha}}, \tag{16}$$

where α is a positive regulation factor. Equation (16) indicates that the higher d_i is, the higher probability it will be moved out from the archive.

4. Experimental Studies

4.1. Database and Experimental Setup. The national route network of China consists of 1706 airway segments, 940 waypoints, and 150 airports. Note that the takeoff and landing phases of flights are truncated within a given radius (usually 10 NM) around airports. The traffic around airports is managed with specific procedures by the terminal control area (TCA) control services in these zones. The airspace is divided into many sectors, and Figure 1(a) shows the sectored airspace in China. The air traffic data was extracted from flight schedule database (FSD) of the summer in 2009 released by Civil Aviation Administration of China (CAAC). In order to better describe the difference between the algorithms' performances, we consider two scenarios: 960 flights (the busiest one hour) and 1664 flights (the busiest three hours).

The parameters are set as follows: the number of populations $M = 5$, $\phi = 0.9$, $\varphi = 0.1$, $r = 0.3$, $\gamma = 1/25$, and $ns = 10$. The mutation probability and the crossover probability of DE are 0.15 and 0.85.

The algorithms, such as our proposed method, MOGA, MOEA/D [30], and cooperative coevolution based algorithm,

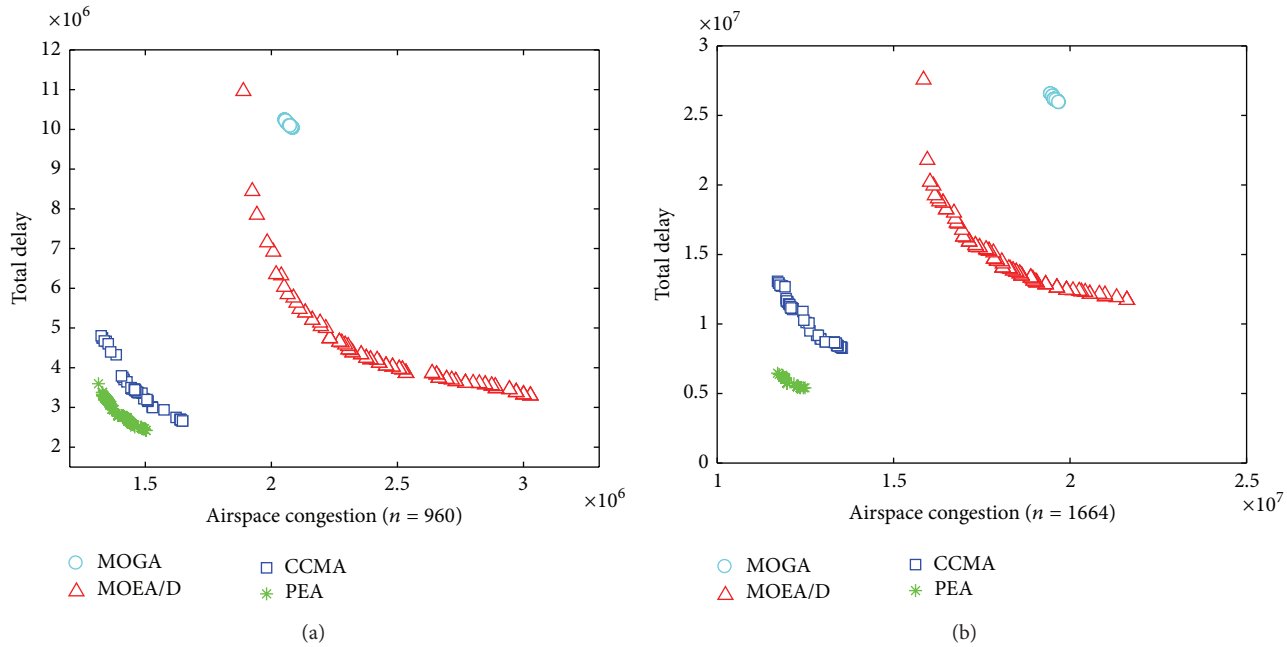


Figure 2: Comparison of different algorithms for 960 flights (a) and 1664 flights (b).

Table 1: Parameters of the experiments.

Parameters	Description	MOGA	MOEA/D	CCMA	PEA
ps	Population size	100	100	100	20 ∗ 5 (IM = 5)
max gen	Max generation	150	150	150	150
p_c	Crossover probability	0.9	0.9	0.9	0.9
p_m	Mutation probability	0.1	0.1	—	—

in this work were implemented in C++, and the simulations were performed on a server with an E5620 2.4 GHz CPU with 12 GB RAM. For each algorithm, the results were collected and analyzed on the basis of 15 independent runs. Besides, the proposed approach was realized by multithreaded programming. Then, the optimization of all islands and all subcomponents of each population can proceed separately and simultaneously which can reduce the computation time.

The parameters used in all experiments are listed in Table 1.

In order to evaluate the performance of the solutions obtained by each of the algorithms, three typical metrics are adopted: the convergence metric (γ) [26], the spread metric (Δ) [31], and the hypervolume metric I_H [32, 33]. γ suggests the average Euclidean distance from the obtained nondominated solution set to the actual Pareto front. Note that it is difficult to find the actual Pareto front for most real-world optimization problems; so we use the best solution set obtained by these algorithms in 15 runs. Δ indicates the diversity of solutions along the Pareto front. I_H can evaluate the convergence and the extent of spread of the solutions simultaneously without the real Pareto front. The smaller the first two indexes are, the better the algorithm is. On the contrary, the larger the third index is, the better the algorithm is.

Table 2: Comparison of different algorithms for 960 flights (I_H, γ, Δ).

Algorithm	I_H	γ	Δ
MOGA	$1.1156e+13$	$6.6865e+06$	1.0078
MOEA/D	$1.9054e+13$	$1.8715e+06$	1.2554
CCMA	$3.1758e+13$	$3.3631e+05$	1.0540
PEA	**3.2575e + 13**	**3.1237e + 04**	**1.0047**

4.2. Comparison with the Existing Methods. In order to test the effectiveness of the proposed multi-island PEA framework, in this part, we will compare it with some existing algorithms, including the classical multiobjective genetic algorithm (MOGA), multiobjective evolutionary algorithm based on decomposition (MOEA/D), and a CC-based multiobjective algorithm (CCMA).

Tables 2 and 3 show the average value of I_H, I_D, and Δ over 15 independent runs of the algorithms for the two scenarios, respectively. In each row of the table, the best value is highlighted in boldface. It can be seen from the tables that PEA outperforms the other three algorithms in terms of I_H, I_D, and Δ. Moreover, when the number of flights increases, PEA performs much better. It can be concluded that PEA has superiority to solve this large-scale problem.

Figure 2 shows the nondominated solutions obtained by the four algorithms. Specifically, the nondominated solutions of each algorithm were obtained over 15 runs. From Figure 2, it can be concluded that PEA performs the best because its solutions can dominate those obtained by other

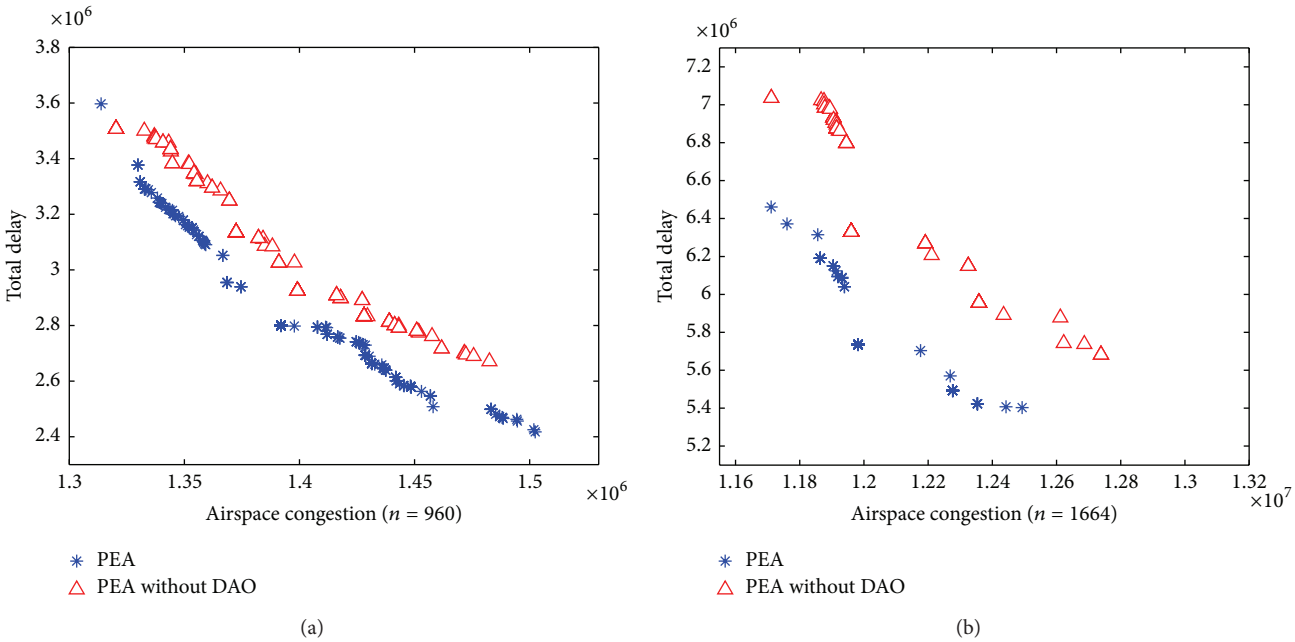


Figure 3: Comparison of PEA and PEA without DAO for 960 flights (a) and 1664 flights (b).

Table 3: Comparison of different algorithms for 1664 flights (I_H, γ, Δ).

Algorithm	I_H	γ	Δ
MOGA	3.3979e + 12	2.1876e + 07	1.0113
MOEA/D	7.4733e + 13	1.1361e + 07	1.0452
CCMA	1.8763e + 14	4.7800e + 06	0.9897
PEA	**2.1909e + 14**	**2.6247e + 05**	**0.8029**

algorithms. Besides, it can be seen that MOGA has the worst performance, and CCMA performs better than MOEA/D, but MOEA/D has good performance in terms of diversity.

From the experimental results, we find that PEA performs better than the other three methods for the two scenarios. It adopts an effective multi-island parallel evolution framework which can improve the optimization capability. Besides, the one-way ring migration strategy can further avoid premature. In addition, this paper introduces the cooperative coevolutionary (CC) into each population optimization via dividing the high-dimensional problem into low-dimensional subcomponents. The subcomponents work cooperatively to obtain better solutions. With the help of parallel computation, the computation is just about 30 minutes which is feasible for the ANFA problem.

4.3. Investigation of the Effectiveness of the Dynamic Adjustment Operator. In the previous section, the first set of experiments has justified the superiority of PEA to existing methods. The next experiment is designed to further investigate whether the dynamic adjustment operator (DAO) based on solution congestion degree contributes to the success of PEA.

Table 4: Comparison of different algorithms for 960 flights (I_H, I_D, Δ).

Algorithm	I_H	I_D	Δ
PEA without DAO	9.2026e + 10	7.0467e + 04	1.2471
PEA	**1.3937e + 11**	**3.3957e + 04**	**1.0050**

Table 5: Comparison of different algorithms for 1164 flights (I_H, I_D, Δ).

Algorithm	I_H	I_D	Δ
PEA without DAO	7.3890e + 6	3.4212e + 05	0.9395
PEA	**1.3107e + 7**	**1.2861e + 05**	**0.7091**

Tables 4 and 5 show the results obtained by PEA and PEA without DAO in terms of the values of the metrics over 15 independent runs of the algorithms when the number of flights is 996 and 1664, respectively. It is seen from the tables that PEA always outperforms the other algorithm in terms of I_H, I_D, and Δ.

Furthermore, like the first experiment, the nondominated solutions of the algorithms are shown in Figure 3. It shows that PEA performs much better than the other method and its nondominated solutions can dominate the solutions obtained by PEA without DAO. For the scenario of 1664 flights, PEA has the most nondominated solutions and spreads nicely in the objective space.

As the optimization generation increases, more solutions will be congested in some local searching space which may cause premature convergence. The dynamic adjustment operator based on the congestion degree can effectively improve the distribution of solutions by inheriting the nondominated

solutions with low congestion degree in a high probability. Hence, it can avoid decreasing the diversity of all populations.

5. Conclusion and Future Work

With the aim of reducing the airspace congestion and the flights delay simultaneously, this paper formulates the airway network flow assignment (ANFA) problem into a multiobjective optimization model and presents a new multiobjective optimization framework to solve it. Firstly, an effective multi-island parallel evolution algorithm is employed to solve the problem by multiple evolution populations. Besides, one-way ring migration topology is applied to improve the efficiency of the cooperation of populations by exchanging individuals among populations. Secondly, the multiobjective evolutionary algorithm NSGA2 is used to optimize each population. In addition, a cooperative coevolution algorithm is adapted to improve the optimization capability by dividing the ANFA problem into several low-dimensional biobjective optimization problems. Finally, in order to maintain the diversity of solutions and avoid prematurity, a dynamic adjustment operator based on solution congestion degree is specifically designed. Simulation results using the real traffic data from the China air route network and daily flight plans demonstrate that the proposed approach can improve the solution quality effectively, showing superiority to the existing approaches such as the multiobjective genetic algorithm, the well-known multiobjective evolutionary algorithm based on decomposition, and a CC-based multiobjective algorithm as well as other parallel evolution algorithms with different migration topology. For future research, the ANFA problem with the influence of severe weather will be considered.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National High Technology Research and Development Program of China (Grant no. 2011AA110101), the National Basic Research Program of China (Grant no. 2010CB731805), and the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (Grant no. 61221061).

References

[1] W. Liu and I. Hwang, "Probabilistic trajectory prediction and conflict detection for air traffic control," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 6, pp. 1779–1789, 2011.

[2] W. B. Du, Z. X. Wu, and K. Q. Cai, "Effective usage of shortest paths promotes transportation efficiency on scale-free networks," *Physica A. Statistical Mechanics and Its Applications*, vol. 392, no. 17, pp. 3505–3512, 2013.

[3] X.-M. Guan, X.-J. Zhang, D.-F. Sun, and S.-T. Tang, "The effect of queueing strategy on network traffic," *Communications in Theoretical Physics*, vol. 60, no. 4, pp. 496–502, 2013.

[4] S. Oussedik and D. Delahaye, "Reduction of air traffic congestion by genetic algorithms," in *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, 1998.

[5] P. Wei, Y. Cao, and D. Sun, "Total unimodularity and decomposition method for large-scale air traffic cell transmission model," *Transportation Research Part B: Methodological*, vol. 53, pp. 1–16, 2013.

[6] P. Vranas, D. Bertsimas, and A. Odoni, "Dynamic ground-holding policies for a network of airports," *Transportation Science*, vol. 28, pp. 275–291, 1994.

[7] M. Ball and G. Lulli, "Ground delay programs: optimizing over included flight set based on distance," *Air Traffic Control Quarter*, vol. 12, pp. 1–25, 2004.

[8] N. Barnier and P. Brisset, "Graph coloring for air traffic flow management," in *Proceedings of the CPAIOR*, pp. 1–15, 2002.

[9] A. M. Abad and J.-P. B. Clarke, "Using tactical flight level allocation to alleviate airspace corridor congestion," in *Proceedings of the AIAA 4th Aviation Technology, Integration, and Operations Forum (ATIO '04)*, pp. 1–9, September 2004.

[10] D. Bertsimas and S. Patterson, "The air traffic flow management problem with enroute capacities," *Operations Research*, vol. 46, no. 3, pp. 406–422, 1998.

[11] D. Bertsimas, G. Lulli, and A. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Operations Research*, vol. 59, no. 1, pp. 211–227, 2011.

[12] D. Delahaye and A. Odoni, "Airspace congestion smoothing by stochastic optimization," in *Proceedings of the Evolutionary Programming Conference*, 1997.

[13] L. Huaxian, Z. Yanbo, C. Kaiquan, and P. Qingge, "Route Network Flow Assignment in the new generation of aviation by cooperative co-evolution," in *Proceedings of the IEEE 5th International Conference on Cybernetics and Intelligent Systems (CIS '11)*, pp. 175–180, September 2011.

[14] A. M. Bayen, R. L. Raffard, and C. J. Tomlin, "Adjoint-based control of a new Eulerian network model of air traffic flow," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 5, pp. 804–818, 2006.

[15] Z. Zheng, Z. Guo, Y.-N. Zhu, and X.-Y. Zhang, "A critical chains based distributed multi-project scheduling approach," *Neurocomputing*, vol. 143, pp. 282–293, 2014.

[16] D. Daniel, S. Oussedik, and P. Stephane, "Airspace congestion smoothing by multi-objective genetic algorithm," in *Proceedings of the 20th Annual ACM Symposium on Applied Computing*, pp. 907–912, March 2005.

[17] Y. Zhu, Z. Zheng, X. Zhang, and K. Cai, "The r-interdiction median problem with probabilistic protection and its solution algorithm," *Computers & Operations Research*, vol. 40, no. 1, pp. 451–462, 2013.

[18] C. Liu, W.-B. Du, and W.-X. Wang, "Particle swarm optimization with scale-free interactions," *PLoS ONE*, vol. 9, no. 5, Article ID e97822, 2014.

[19] X. Zhang, Z. Zheng, Y. Zhu, and K. Y. Cai, "Protection issues for supply systems involving random attacks," *Computers and Operations Research*, vol. 43, no. 1, pp. 137–156, 2014.

[20] J. Tang, M. H. Lim, and Y. S. Ong, "Diversity-adaptive parallel memetic algorithm for solving large scale combinatorial optimization problems," *Soft Computing*, vol. 11, no. 9, pp. 873–888, 2007.

[21] J. Tang, M. H. Lim, Y. S. Ong, and M. J. Er, "Study of migration topology in island model parallel hybrid-GA for large scale quadratic assignment problems," in *Proceedings of the*

8th International Conference on Control, Automation, Robotics and Vision (ICARCV '04), pp. 2286–2291, Kunming, China, December 2004.

[22] J. Tang, M. H. Lim, and Y. S. Ong, "Parallel memetic algorithm with selective local search for large scale quadratic assignment," *Journal of Innovative Computing, Information and Control*, vol. 2, pp. 1399–1416, 2006.

[23] Z. Yang, K. Tang, and X. Yao, "Large scale evolutionary optimization using cooperative coevolution," *Information Sciences*, vol. 178, no. 15, pp. 2985–2999, 2008.

[24] M. Potter and K. D. Jong, "A cooperative coevolutionary approach to function optimization," in *Proceedings of the 3rd Conference on Parallel Problem Solving from Nature*, pp. 249–257, 1994.

[25] F. Zhu and S.-U. Guan, "Cooperative co-evolution of GA-based classifiers based on input decomposition," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 8, pp. 1360–1369, 2008.

[26] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[27] R. Storn and K. Price, "Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[28] W. Liu, Z. Zheng, and K.-Y. Cai, "Bi-level programming based real-time path planning for unmanned aerial vehicles," *Knowledge-Based Systems*, vol. 44, pp. 34–47, 2013.

[29] Z. Zheng, L. Shumin, G. Ze, and Z. Yueni, "Resource-constraint multi-project scheduling with priorities and uncertain activity durations," *International Journal of Computational Intelligence Systems*, vol. 6, no. 3, pp. 530–547, 2013.

[30] Q. Zhang and H. Li, "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[31] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, "Performance assessment of multiobjective optimizers: an analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.

[32] M. Fleischer, "The measure of pareto optima applications to multi-objective metaheuristics," in *Evolutionary Multi-Criterion Optimization: Proceedings of the 2nd International Conference, EMO 2003, Faro, Portugal, April 8–11, 2003*, vol. 2632 of *Lecture Notes in Computer Science*, pp. 519–533, Springer, Berlin, Germany, 2003.

[33] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, pp. 80–83, 1945.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 523914, 10 pages
http://dx.doi.org/10.1155/2015/523914

Research Article

Helicopter Control Energy Reduction Using Moving Horizontal Tail

Tugrul Oktay and Firat Sal

College of Aviation, Erciyes University, 38039 Kayseri, Turkey

Correspondence should be addressed to Tugrul Oktay; tugruloktay52@gmail.com

Received 7 November 2014; Revised 14 February 2015; Accepted 1 April 2015

Academic Editor: Zheng Zheng

Helicopter moving horizontal tail (i.e., MHT) strategy is applied in order to save helicopter flight control system (i.e., FCS) energy. For this intention complex, physics-based, control-oriented nonlinear helicopter models are used. Equations of MHT are integrated into these models and they are together linearized around straight level flight condition. A specific variance constrained control strategy, namely, output variance constrained Control (i.e., OVC) is utilized for helicopter FCS. Control energy savings due to this MHT idea with respect to a conventional helicopter are calculated. Parameters of helicopter FCS and dimensions of MHT are simultaneously optimized using a stochastic optimization method, namely, simultaneous perturbation stochastic approximation (i.e., SPSA). In order to observe improvement in behaviors of classical controls closed loop analyses are done.

1. Introduction

Traditionally in order to control helicopters, collective and cyclic (i.e., longitudinal and lateral) rotor blade pitches are used. Presently almost all of the helicopters employ a swashplate mechanism and pitch links (it consists of two circular plates and a ball bearing arrangement separating them; see [1] for more details) to transmit two cyclic and collective pitch commands to the blade root. However, this mechanism is heavy and complex and also causes important drag during high flight speeds. Throughout history some other control methods have been considered in order to avoid these drawbacks and also for some other reasons such as reduction of control energy and redundancy in case of failures. Some of these alternatives are using trailing edge flaps (TEFs) with (see [2–4]) and without (see [5–7]) classical swashplate mechanism, passive (see [8–10]) and active (see [11–13]) helicopter morphing, and MHT (see [14–18]). For example, in [2] TEFs were integrated into blades for the case of a failure of the pitch link making the blade free float in pitch. By this method catastrophic results of a pitch link failure were corrected. In [5] TEFs were replaced with a conventional swashplate mechanism. Via eliminating swashplate mechanism and using TEFs, important reduction in weight, drag, and cost and also improvement in rotor performance were obtained. Moreover, in [8] passive morphing was used in order to reduce helicopter FCS energy. In that study many blade parameters (e.g., blade length and blade chord length) were simultaneously optimized with helicopter FCS parameters in order to save FCS energy. Substantial reduction in helicopter FCS energy was obtained using passive morphing idea. In [11] active morphing was used to save helicopter FCS energy. In that study actively morphing parameters were blade chord length, blade length, blade twist, and main rotor angular speed. The main difference between this and previous study (i.e., passive morphing) was that for the active case the helicopter design parameters are able to change (except helicopter FCS) during flight, but in prescribed interval. Using active morphing idea significant reduction in helicopter FCS energy was obtained.

MHT idea was firstly studied in [14] in 1953. In this study differential control of each side of a canted horizontal tail was permitted. In [15] collective control of horizontal tail was mechanically achieved. In this study it was claimed that a fixed horizontal tail is advantageous in order to improve longitudinal stability of helicopters in forward flight, but it is not enough during gliding and climbing flights. More recently in [16] a moveable horizontal tail to give the desired attitude at different flight speed for UH-60 was designed. Recently

in [17, 18] a moveable horizontal tail was designed in order to reduce TEF deflection for swashplateless helicopters since the stroke capacity of existing smart material actuators is not enough for the required TEFs inputs. Using MHT TEFs deflections were relieved.

Numerous helicopter FCS design methods have been studied throughout the years, in historical sequence classical pole placement techniques (see [19, 20]), simple feedback control methods (see [21, 22]), and modern control approaches depend on linear matrix algebra such as linear quadratic regulator (LQR) and linear quadratic Gaussian (LQG) techniques (see [23, 24]), H_∞ control synthesis (see [25, 26]), and model predictive control (MPC) (see [27, 28]). In this paper, a modern constrained control method, namely, OVC, is chosen for the design of helicopter FCS. OVC has many advantages with respect to the other control strategies existing in the literature. First, these controllers are modified LQG controllers and they benefit from Kalman filters as state estimators. Second, variance constrained controllers apply second-order information (i.e., state covariance matrix; see [29, 30] for details) and this kind of information is very beneficial during multivariable control system design because all stabilizing controllers are parameterized in relation to the physically meaningful state covariance matrix. Last, for large and strongly coupled multi-input, multi-output (MIMO) systems as in air vehicles control and especially in this paper, variance constrained control methods give guarantees on the transient behavior of independent variables by enforcing upper limits on the variance of these variables.

Variance constrained controllers have been used for many aerospace vehicles (e.g., helicopters, see [8, 11, 31–36]; tiltrotor aircraft, see [37]; Hubble space telescope, see [38]; tensegrity structures, see [39]) in last thirty years. For instance, in [32] variance constrained controllers were used for helicopter FCS during maneuvers, specifically level banked turn and helical turn. In that paper, performance of them was also considered during failures of some helicopter sensors. Reasonable consequences (meaning that variance constraints on outputs/inputs were satisfied and also closed loop systems were exponentially stabilized) were found in terms of helicopter FCS. Robustness of the closed loop systems (obtained via integration of linearized helicopter model and FCS) with respect to some modeling uncertainties (i.e., variation of flight conditions and all helicopter inertial parameters) was also studied and it was found that these controllers have stability robustness with respect to modeling uncertainties.

In this paper, MHT is for the first time simultaneously designed with helicopter FCS. For this purpose, a specific variance constrained controller OVC is also used for the first time for FCS. It is important to note that when MHT is integrated with classical helicopter, the number of controls increases. This causes an important result. The number of trim unknowns increases with additional MHT controls. Nevertheless, there are no additional trim equations. Therefore, in order to solve the resulting nonlinear trim equations, a useful optimization algorithm is required. For its solution, a stochastic optimization method specifically simultaneous perturbation stochastic approximation (i.e., SPSA) (see [40, 41] for brief description of SPSA) is for the first time applied for the simultaneous trimming and FCS design problem since it is computationally cheap and effective during solving constrained optimization problems when it is impossible to compute derivatives such as gradients and Hessians, analytically as in the situation herein. This paper first presents helicopter models used for simultaneous MHT and FCS design. Second, MHT is illustrated and motions of it are described. Then, definition of applied FCS (i.e., OVC) is given briefly. After that, trimming the system (i.e., the one obtained via integration of helicopter, MHT, and FCS) via simultaneous trimming and FCS design idea is explained. Then, the specific optimization method, namely, SPSA, applied in order to trim the system is summarized. Finally, this simultaneous design idea is applied for Puma SA 330 helicopter and closed loop responses of classical helicopter and helicopter with MHT are compared.

2. Helicopter Model

The modeling approach of used helicopter models in this paper is presented in detail in [31, 42]. The essential modeling assumptions are given next. First of all, multibody system approach was used to include all helicopter components: fuselage, horizontal tail, tail rotor hub and shaft, landing gear, and fully articulated main rotor with 4 rigid blades with blade flapping and lagging hinges. Secondly, a static inflow formulation (i.e., Pitt-Peters formulation) was applied for helicopter main rotor downwash. Thirdly, linear incompressible aerodynamics was used for the main rotor blades, but an analytical formulation was applied for the modeling of fuselage.

The modeling procedure requires using physics principles and because of the assumptions described in the previous paragraph it directly led to helicopter dynamic models that consisted of finite sets of ordinary differential equations (ODEs). This mathematical structure is fairly beneficial for control system design since it assists the direct use of modern control theory, which relies on state space representations of the system's dynamics, easily obtained from ODEs.

The modeling methodology summarized above was applied in Maple and it led to a nonlinear helicopter model in implicit form:

$$f\left(\dot{x}_n, x_n, u_n\right) = 0, \tag{1}$$

where $f \in \mathbb{R}^{28}$, $x_n \in \mathbb{R}^{25}$, and $u_n \in \mathbb{R}^4$. Here x_n and u_n are nonlinear state and control vectors, respectively, and $\mathbb{R}^*$ represents the linear space of $*$-dimensional real vectors, where "$*$" can be 28, 25, or 4. It should be noted that the inconsistency between the size of $f(28)$ and the size of $x_n(25)$ is due to the three static downwash equations. The 28 nonlinear equations in (1) are categorized as follows: 9 fuselage equations, 8 blade flapping and 8 blade lead-lagging equations, and 3 static main rotor downwash equations. The helicopter models obtained have too many terms, making its use in fast computation impractical. For that reason, a systematic model simplification technique, named ordering scheme, was applied to reduce the number of terms in the nonlinear ODEs. The ordering scheme iteratively deletes terms from an equation depending on their relative magnitude with respect to the other terms in that equation. Each term's magnitude is

Table 1: Flight dynamics modes comparison.

Mode (rad/s)	V_A = hover		V_A = 40 kts		V_A = 80 kts	
	Padfield	Our model	Padfield	Our model	Padfield	Our model
1st	$0.2772 \pm 0.5008i$	$0.2215 \pm 0.5966i$	$-0.1543 \pm 0.9181i$	$-0.0434 \pm 1.0846i$	$-0.1854 \pm 1.0546i$	-0.1736 ± 2.0642
2nd	$-0.0410 \pm 0.5691i$	$0.0587 \pm 0.3589i$	$0.0275 \pm 0.3185i$	$-0.0143 \pm 0.3253i$	$-0.0085 \pm 0.2074i$	$-0.0138 \pm 0.1674i$
3rd	-0.2697	-0.1449	-0.0976	-0.0703	-0.1358	-0.04786
4th	-0.3262	-1.1944	-0.9817	-0.7140	-1.5163	-0.7587
5th	$-1.2990 \pm 0.2020i$	$-0.6536 \pm 0.3536i$	$-1.0394 \pm 0.2798i$	$-0.6125 \pm 1.1300i$	$-0.9252 \pm 1.0503i$	-0.6759 ± 1.8865

(Padfieldresults are taken from [43]).

guessed depends on expected values that the state and control variables can take during helicopter flight (see [31, 42]). It is significant to note that the ordering scheme does not change the number or type of equations generated using physics principles; it just shortens the equations by retaining the dominant terms.

The model found after using the ordering scheme is still reasonably complex (i.e., with a total of 28 nonlinear equations). In this paper, for FCS design the nominal trajectories considered are straight level flights. When the straight level flight conditions were applied for the nonlinear equations of motion, 17 trim equations were found (i.e., $0 = 0$ equations were deleted). These equations were solved using MATLAB for different straight level flight speeds. After trimming, the model was linearized using Maple, yielding continuous linear time-invariant (LTI) systems:

$$\dot{x}_p = A_p x_p + B_p u_p. \tag{2}$$

Here x_p and u_p are the perturbed state and perturbed control vectors. Matrices A_p and B_p are of size 25×25 and 25×4. The state vector consists of 9 fuselage states, 8 blade flapping states, and 8 blade lead-lagging states. The control vector includes 3 main rotor controls (collective, θ_0, longitudinal cyclic, θ_c, and lateral cyclic, θ_s, blade pitch angles) and 1 tail rotor control (collective, θ_T).

Puma SA 330 helicopter (see [31, 43]) was used to validate the models used in this paper. These models are leading to acceptable agreement on trim values, flight dynamics modes, and qualitatively similar flapping and lead-lagging mode behavior (see [43]). In Table 1 and Figure 1 some validation results show how the models correctly capture the dynamics of Puma SA 330 helicopter (see [31] for more validation data). For instance, most of the flight dynamics modes (linearized system eigenvalues) of the models for hover and straight level flights (i.e., 40 kts and 80 kts) match well the results reported in [43]. The mode displaying the largest discrepancy is the 4th mode (it is important to note that this is due to modeling discrepancy between the models used and [43]); nevertheless, the qualitative behavior is similar (they are both exponentially stable modes).

The qualitative behaviors of the blade flapping and lead-lagging modes are also identical with the ones given in [43] that the blade flapping modes are much farther away from the imaginary axis with respect to the blade lead-lagging modes and the magnitude of the frequency bound for the blade flapping modes is larger than the one for the blade lead-lagging modes (see Figure 2).

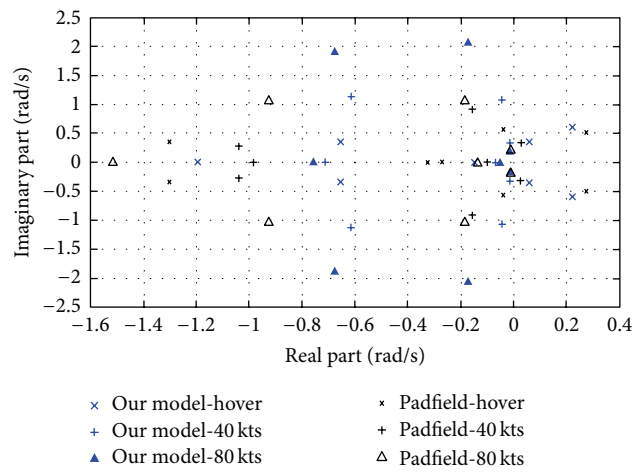


Figure 1: Loci of flight dynamics modes.

It is also required to note that all trim results obtained using our model also showed good correspondence with data given in the literature (see [31] for more details). For instance, the trim values for straight level flight at 40 kts were

$$\begin{aligned} {}^{40\,\text{kts}}x_0 &= \Bigg[\underbrace{0.2753, 0.0370, -0.0908, 0.4857}_{\theta_{0_0}, \theta_{c_0}, \theta_{s_0}, \theta_{T_0}}, \underbrace{-0.0456, 0.0272}_{\phi_{A_0}, \theta_{A_0}}, \\ &\quad \underbrace{0.0795, 0.0592, 0.0252, 0}_{\beta_{0_0}, \beta_{c_0}, \beta_{s_0}, \beta_{d_0}}, \underbrace{0.0218, 0.0010, -0.0082, 0}_{\zeta_{0_0}, \zeta_{c_0}, \zeta_{s_0}, \zeta_{d_0}}, \\ &\quad \underbrace{1.2523, 6.2236, 9.2217}_{\chi_0, \lambda_{0_0}, \lambda_{c_0}} \Bigg]^T. \end{aligned} \tag{3}$$

Here $\{\theta_{0_0}, \theta_{c_0}, \theta_{s_0}, \theta_{T_0}\}$, $\{\beta_{0_0}, \beta_{c_0}, \beta_{s_0}, \beta_{d_0}\}$, $\{\zeta_{0_0}, \zeta_{c_0}, \zeta_{s_0}, \zeta_{d_0}\}$, and $\{\phi_{A_0}, \theta_{A_0}\}$ vectors are trim values of conventional helicopter controls, blade flapping angles, blade lead-lagging angles, and Euler angles, respectively, and all are given in unit of radians. The trim vector of linear downwash is $\{\chi_0, \lambda_{0_0}, \lambda_{c_0}\}$ where λ_{0_0}, λ_{c_0} are trims of collective and longitudinal cyclic downwash in m/s and χ_0 is the trim of wake skew angle given in radians.

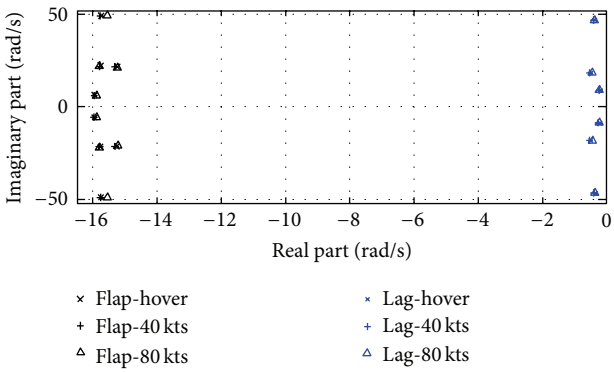


Figure 2: Loci of flapping and lead-lagging modes.

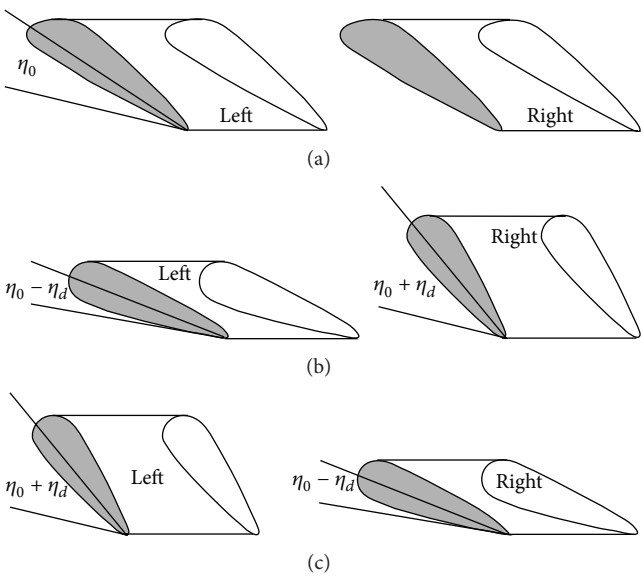


Figure 3: (a) Collective MHT angle, (b) left negative and right positive differential MHT angles, and (c) right negative and left positive differential MHT angles.

3. Illustration of MHT

MHT angles (i.e., collective and differential) are illustrated in Figure 3. Collective motion refers to the movement of left and right horizontal tails in the same direction and magnitude simultaneously. On the other hand, differential motion refers to the movement of them in the opposite direction and the same magnitude simultaneously.

Angle of attack for left and right horizontal tails is calculated using

$$\begin{aligned} \alpha_{tp_r} &= \alpha_{tp} + \eta_0 + \eta_d, \\ \alpha_{tp_l} &= \alpha_{tp} + \eta_0 - \eta_d, \end{aligned} \tag{4}$$

where α_{tp} is the angle of attack of classical fixed horizontal tail and α_{tp_r} and α_{tp_l} are angle of attack for moving right and left horizontal tails, respectively. Thelast MHT control (i.e., l_0) is the control parameter of distance between helicopter center of gravity (cg) and horizontal tail (HT). The distance between cg and HT is found to be multiplying distance control parameter with the classical helicopter cg-HT distance.

4. Flight Control System (FCS)

For FCS, a variance constrained controller specifically output variance constrained control (OVC) is chosen. The OVC problem's description is given next.

For a given continuous linear time invariant (LTI) system

$$\dot{x}_p = A_p x_p + B_p u_p + w_p, \qquad y = C_p x_p, \qquad z = M_p x_p + v \tag{5}$$

and a positive definite input penalty matrix $R > 0$, find a full order dynamic controller

$$\dot{x}_c = A_c x_c + F z, \qquad u_p = G x_c \tag{6}$$

to solve the problem

$$\min_{A_c, F, G} J = E_\infty u_p^T R u_p = \operatorname{tr}\left(R G \Phi G^T\right) \tag{7}$$

subject to

$$E_\infty y_i^2 \le \sigma_i^2, \quad i = 1, \ldots, n_y, \tag{8}$$

where z represents sensor measurements, w_p and v are zero-mean uncorrelated Gaussian white noises with intensities W and V, respectively, σ_i^2 is the upper bound imposed on the ith output variance, and n_y is the number of outputs. The quantity $J = E_\infty u_p^T R u_p$ is referred to as the control energy (or cost) and Φ is the state covariance matrix computed using the OVC algorithm (see [44, 45]). Here $E_\infty \triangleq \lim_{t\to\infty} E$ and E is the expectation operator. The solution to the OVC problem is obtained from a linear quadratic Gaussian (LQG) problem by choosing appropriately the output penalty $Q > 0$. Specifically, Q is dictated by the constraints imposed on the output variances (i.e., σ_i^2 in (8)) and it can be obtained using the iterative algorithm described in [44, 45]. After the algorithm converges and Q is found, the OVC parameters are computed using

$$\begin{aligned} A_c &= A_p + B_p G - F M_p, \\ F &= X M_p^T V^{-1}, \\ G &= -R^{-1} B_p^T K, \end{aligned} \tag{9}$$

where X and K are obtained by solving the following two algebraic Riccati equations:

$$0 = X A_p^T + A_p X - X M_p^T V^{-1} M_p X + W, \tag{10a}$$

$$0 = K A_p + A_p^T K - K B_p R^{-1} B_p^T K + C_p^T Q C_p. \tag{10b}$$

Clearly, compared to LQG where the penalties are selected ad hoc, OVC has the advantage that the penalty Q is selected such that output variance constraints are satisfied.

Table 2: MHT parameters, constraints, and optimum points using SPSA.

MHT parameters	Nominal trim value	Lower bound $\Delta x_i/x_i$	Upper bound $\Delta x_i/x_i$	Optimum trim value	Change $\Delta x_i/x_i$
η_0	0 rad	−0.05	0.05	0.5022 rad	—
η_d	0 rad	−0.05	0.05	−0.5037 rad	—
l_0	3.80 m	−0.05	0.05	2.9317 m	−0.2285

5. Trimming and Simultaneous MHT and FCS Design

Now it is required to define the simultaneous trimming and FCS design problem for the helicopter with MHT. This problem makes use of the extra number of trim unknowns (i.e., the 3 MHT control trims) and the ability to create helicopter linearized state-space models in terms of these MHT control trims. Let $x = \{\eta_0, \eta_d, l_0\}$ be the set of MHT control trims. The problem of finding optimum trim values for MHT controls can be obtained via changing the traditional OVC design problem summarized in Section 4 if the dependencies $A_p(x)$, $B_p(x)$ are considered. It is important to note that here x denotes the MHT controls trim values. During the control problem, formulation u_p represents perturbedcontrol vector and includes the MHT controls. The FCS energy in (7) and the expected values (i.e. $E_\infty y_i^2$, $i = 1, \ldots, n_y$) in (8) are now function of these MHT control trims also, in addition of the control matrices (A_c, F, G). Therefore, the following optimization problem is created:

$$\min_{A_c, F, G, x} J = E_\infty u_p^T R u_p \tag{11}$$

subject to (5), (6), and (8). Furthermore, the components of x are constrained (i.e., $x_{i_{\min}} \le x_i \le x_{i_{\max}}$, see Table 2). This new optimization problem is much more complicated than traditional OVC design and how to solve it is discussed next.

6. Simultaneous Perturbation Stochastic Approximation (SPSA)

The problem of finding the optimum values of the MHT control trims during the simultaneous trimming and FCS design problem summarized in Section 5 is much more difficult than the traditional OVC design due to the introduction of the additional MHT trim optimization variables and the associated constraints on them. Since there is complex dependency between J and expected values of outputs of interest, computation of their derivatives with respect to these variables is analytically impossible. This recommends the application of certain stochastic optimization techniques. In order to solve it, a stochastic optimization method, namely, SPSA, is chosen. This method was successfully used in similar complex constrained optimization problems (see [8, 11, 31, 41]) before. SPSA has many advantages. First, SPSA is inexpensive because it uses only two evaluations of the objective to estimate the gradient (see [40]). It is also successful in solving constrained optimization problems (see [8, 11, 31, 41, 46]). Moreover, under certain conditions (see [41]) strong convergence of SPSA was theoretically proved. Its short summary is given next.

Let x denote the vector of optimization variables. For the classical SPSA, if $x_{[k]}$ is the estimate of x at kth iteration, then

$$x_{[k+1]} = x_{[k]} - a_k g_{[k]}, \tag{12}$$

where

$$g_{[k]} = \left[\frac{\Gamma_+ - \Gamma_-}{2 d_k \Delta_{[k]1}} \cdots \frac{\Gamma_+ - \Gamma_-}{2 d_k \Delta_{[k]p}} \right]^T, \tag{13}$$

a_k and d_k are gain sequences, $g_{[k]}$ is the estimate of the objective's gradient at $x_{[k]}$, $\Delta_{[k]} \in R^p$ is a vector of p mutually independent mean-zero random variables $\{\Delta_{[k]1} \cdots \Delta_{[k]p}\}$ satisfying certain conditions (see [47, 48]), and Γ_+ and Γ_- are estimates of the objective evaluated at $x_{[k]} + d_k \Delta_{[k]}$ and $x_{[k]} - d_k \Delta_{[k]}$, respectively. An adaptive algorithm considering the requirement that the optimization variables must be between lower and upper bounds was previously developed and combined with OVC to solve the simultaneous actively and passively morphing helicopter and FCS design problem (see [8, 11]). The adaptation is via the gain sequences, a_k and d_k, and they are

$$a_k = \min\left\{ \frac{a}{(S+k)^\lambda}, 0.95 \min_i \left\{ \min\left(\varphi_{l_i}\right), \min\left(\varphi_{u_i}\right) \right\} \right\},$$
$$d_k = \min\left\{ \frac{d}{k^\Theta}, 0.95 \min_i \left\{ \min\left\{\vartheta_{l_i}\right\}, \min\left\{\vartheta_{u_i}\right\} \right\} \right\}, \tag{14}$$

where ϑ_l and ϑ_u are vectors whose components are $(x_{[k]i} - x_{\min_i})/\Delta_{[k]i}$ for each positive $\Delta_{[k]i}$ and $(x_{\max_i} - x_{[k]i})/\Delta_{[k]i}$ for each negative $\Delta_{[k]i}$, respectively. Similarly, φ_l and φ_u are vectors whose components are $(x_{[k]i} - x_{\min_i})/g_{[k]i}$ for each positive $g_{[k]i}$ and $(x_{[k]i} - x_{\max_i})/g_{[k]i}$ for each negative $g_{[k]i}$, respectively, and d, a, λ, Θ, and S are other SPSA parameters. The reader interested in the details of this algorithm is referred to [8, 11, 31].

In order to solve the simultaneous trimming and control design problem for optimal MHT trim values, the following algorithm is used in this paper.

Step 1. Set $k = 1$ and choose initial values for the optimization parameters, $x = x_{[k]}$, and a specific flight condition (e.g., V_A = 40 kts straight level flight).

Step 2. Compute A_p and B_p, design the corresponding OVC using (9), (10a), and (10b), and find the current value of the objective, Γ_k using (11); note that $\Gamma_k = J_k$ for OVC.

Step 3. Perturb $x_{[k]}$ to $x_{[k]} + d_k \Delta_{[k]}$ and $x_{[k]} - d_k \Delta_{[k]}$ and solve the corresponding OVC problems to find Γ_+ and Γ_-,

respectively. Then compute the approximate gradient, $g_{[k]}$, using (13) with d_k given by (14).

Step 4. If $\|a_k g_{[k]}\| < \delta x$, where a_k is given by (14) and δx is the minimum allowed variation of x, or $k+1$ is greater than the maximum number of iterations allowed, exit, else calculate the next estimate of x, $x_{[k+1]}$, using $x_{[k+1]} = x_{[k]} - a_k g_{[k]}$, set $k = k+1$, and return to Step 2.

7. Results

7.1. Helicopter FCS Energy Saving. It is first required to note that, for all of the numerical results reported in this paper, the sensor measurements (z in (5)) were helicopter linear velocities, angular velocities, and Euler angles. The outputs of interest (y in (5)) were the helicopter Euler angles. The tolerance used for all of the OVC designs was 10^{-7}. Firstly, the nonlinear helicopter model including MHT was trimmed using simultaneous trim and FCS design idea. The output variance constraints on helicopter Euler angles were $\sigma^2 = 10^{-4}\begin{bmatrix}1 & 1 & 0.1\end{bmatrix}$ while the inputs of interest were all traditional helicopter controls (i.e., 3 main rotor controls and 1 tail rotor control) and the additional MHT controls (i.e., collective and differential control and distance control parameter). The helicopter FCS energy obtained after simultaneous trimming and FCS design is labeled as J_r. Secondly, for the same flight condition, the same outputs and inputs of interest and the same constraints OVC were redesigned for the helicopter without MHT. The resulting helicopter FCS energy is labeled as J_n. In order to see the benefits of using MHT on helicopters, the relative variation of the helicopter FCS energy, $\%J$, was computed using $\%J = 100(J_n - J_r)/J_n$.

The adaptive SPSA algorithm summarized in Section 6 was applied in order to solve the simultaneous trimming and design problem using the SPSA parameters of $S = 5$, $\lambda = 0.602$, $a = 500$, $d = 20$, and $\Theta = 0.101$ via MATLAB software. For this design problem the algorithm was very effective in rapidly decreasing the helicopter FCS energy, J, converging quickly to a stable value, as seen in Figure 4 (see Table 2 for optimum MHT control trim values). Moreover, the FCS energy corresponding to the system obtained using simultaneous trimming and design was 59.4% lower than the FCS energy of system obtained using classical helicopter and traditional OVC (meaning that $\%J = 59.4\%$). The vector of trim values obtained after applying simultaneous trimming and design situation was

$$
\begin{aligned}
{}_{\mathrm{MHT}}{}^{40\,\mathrm{kts}}x_0 &= \Bigg[\underbrace{0.2830, 0.0041, -0.1675, 0.4826}_{\theta_{0_0},\theta_{c_0},\theta_{s_0},\theta_{T_0}}, \\
&\quad \underbrace{-0.0620, 0.070}_{\phi_{A_0},\theta_{A_0}}, \underbrace{0.0849, 0.1348, -0.0027, 0}_{\beta_{0_0},\beta_{c_0},\beta_{s_0},\beta_{d_0}}, \\
&\quad \underbrace{0.08634, -0.0014, -0.0123, 0}_{\zeta_{0_0},\zeta_{c_0},\zeta_{s_0},\zeta_{d_0}}, \underbrace{1.1942, 6.6766, 9.3000}_{\chi_0,\lambda_{0_0},\lambda_{c_0}}\Bigg]^T.
\end{aligned}
\tag{15}
$$

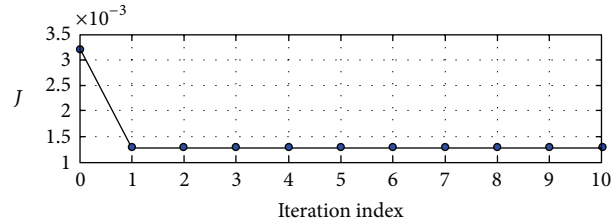


Figure 4: Cost optimization via SPSA.

7.2. Closed Loop Simulations. In order to better evaluate the influence of MHT on helicopter performance, closed loop performance of classical helicopter and helicopter with MHT are compared. For this purpose, helicopter linearized state-space model obtained after simultaneous trimming and control design is used. For the discussions given next, closed loop system, which is obtained via integration of classical helicopter and OVC designed for it, is referred to as the 1st closed loop system. Similarly, closed loop system, which is obtained via integration of helicopter with MHT and OVC designed for it using simultaneous trimming and control design, is referred to as the 2nd closed loop system. In the figures given next, degrees are used to better show the behaviors of certain variables. The labels of classical and MHT are referring to the classical helicopter and helicopter with MHT, respectively. It is also significant to note that because linearized models are used, variables represent perturbations from their trim values in all the next set of figures.

In Figure 5, closed loop responses of helicopter Euler angle states are given when the 1st closed loop system (solid black line) and 2nd closed loop system (solid blue line) are both excited by white noise perturbations. From Figure 3 it can be easily seen that, for both classical helicopter and helicopter with MHT, the qualitative (i.e., shape of the response) and quantitative (i.e., magnitude of the response) behaviors of Euler angles are basically the same. This can be explained using the fact that the expected values ($E_\infty y_i^2$) of outputs of interest (i.e., helicopter Euler angles in this paper) are very close and satisfy the constraints ($E_\infty y_i^2 \le \sigma_i^2$).

In Figure 6, closed loop responses of helicopter linear and angular velocity states are given for the 1st closed loop system (solid black line) and 2nd closed loop system (solid blue line). Figure 6 shows that the linear and angular velocity states do not experience catastrophic behavior (meaning that fast and large variations do not occur). For both classical helicopter and helicopter with MHT, qualitative behaviors are similar. This nice behavior is clarified by the exponentially stabilizing effect of OVC (see [31] for more details).

In Figure 7, closed loop responses of all traditional helicopter controls (i.e., 3 main rotor and 1 tail rotor controls) are given for both classical helicopter and helicopter with MHT. The most important observation related to the traditional helicopter controls is that there is substantial reduction in the peaks of the absolute values of these controls if MHT is utilized. It can be easily seen from this figure that lateral cyclic blade pitch angle (i.e., θ_c) experiences with smallest

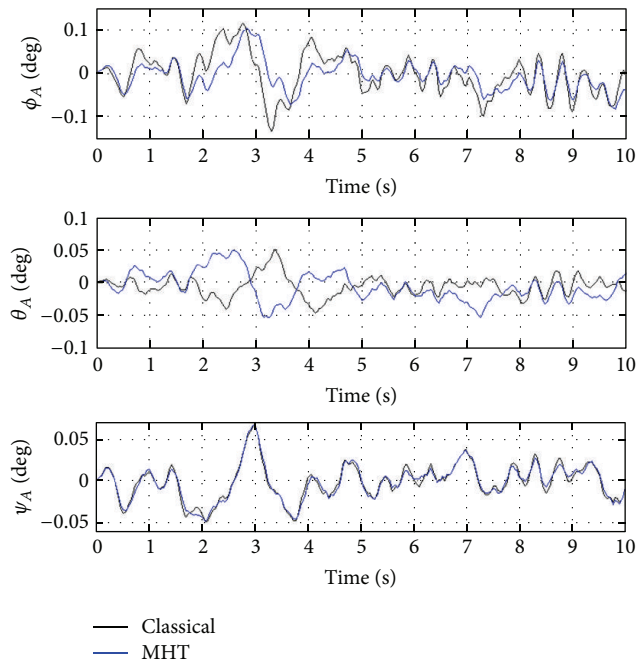


Figure 5: Responses of helicopter Euler angle states.

reduction. Moreover, the control variations are smooth and small.

In Figure 8, closed loop responses of MHT controls (i.e., collective and differential angles and distance control parameter) are given. It is clear from this figure that the peaks of the absolute values of all these additional controls are reasonable. Furthermore, they do not experience catastrophic behavior. Our extensive results also show that blade states do not experience catastrophic behavior and their qualitative behaviour is similar with/without MHT. This good behaviour is also clarified by the exponentially stabilizing effect of OVC.

8. Conclusions

Moving horizontal tail (MHT) idea is investigated in order to reduce helicopter flight control system (FCS) energy. Complex, control-oriented, physics-based nonlinear helicopter models are used for this purpose. Output variance constrained (OVC) controller is applied for helicopter FCS design. A stochastic optimization method is used in order to trim the helicopter during the simultaneous trimming and FCS design problem. Substantial FCS energy reduction (around 60%) is obtained using MHT. It is also important to note that this energy saving is obtained using small MHT control inputs. Nowadays such small changes are easily achievable and technologically feasible. It is also required to note that the FCS energy saving given in this paper is computed using linearized state-space model. In reality due to the nonlinearities it may be slightly different than this value.

Moreover, the qualitative behaviors of fuselage and blade states with/without MHT are similar and they do not display catastrophic behaviors. The outputs of interest (i.e., helicopter Euler angles) with/without MHT also display qualitatively and quantitatively similar behaviors while satisfying all of the output variance constraints. The peak values of traditional controls decrease with MHT clarifying the substantial reduction of FCS energy seen when MHT is applied.

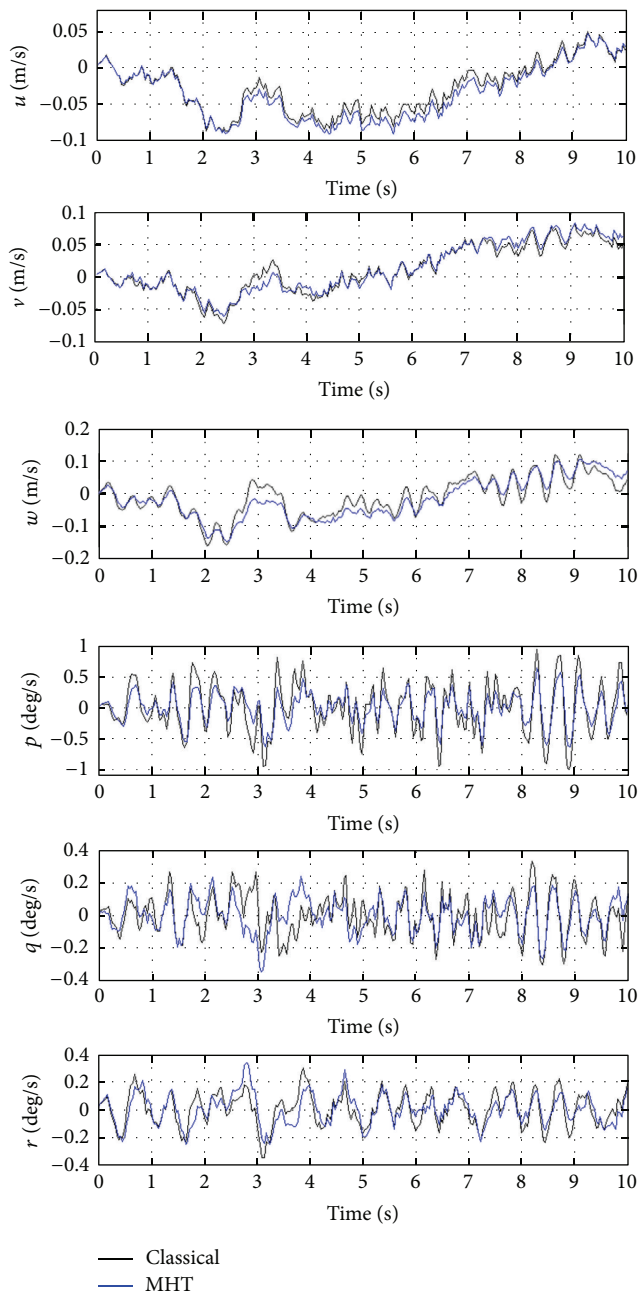


Figure 6: Responses of helicopter linear and angular velocity states.

Nomenclature

p, q, r:	Helicopter angular velocities, [rad/s]
u, v, w:	Helicopter linear velocities, [m/s]
ϕ_A, θ_A, ψ_A:	Helicopter Euler angles, [rad]
J:	Control energy, []

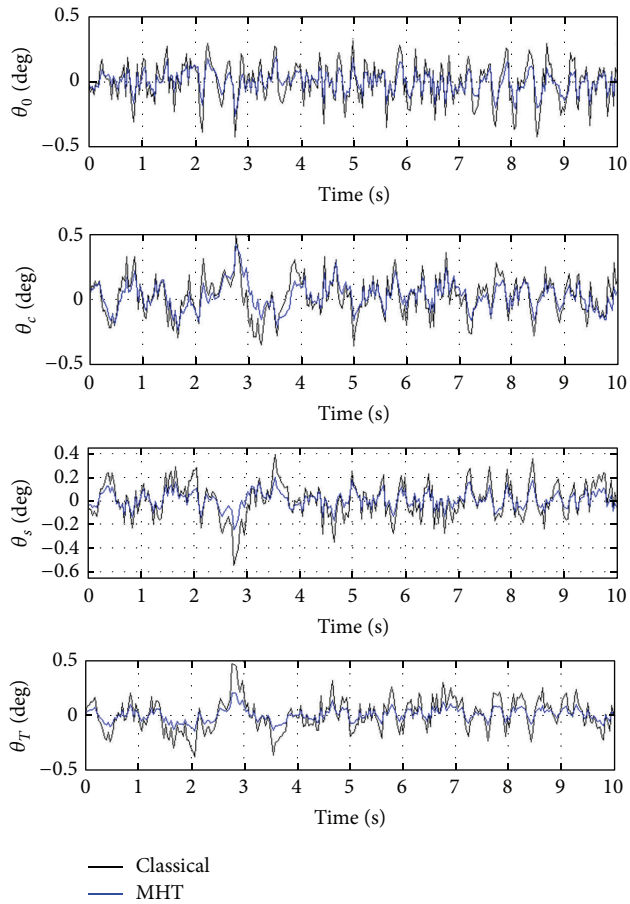


Figure 7: Responses of helicopter classical controls.

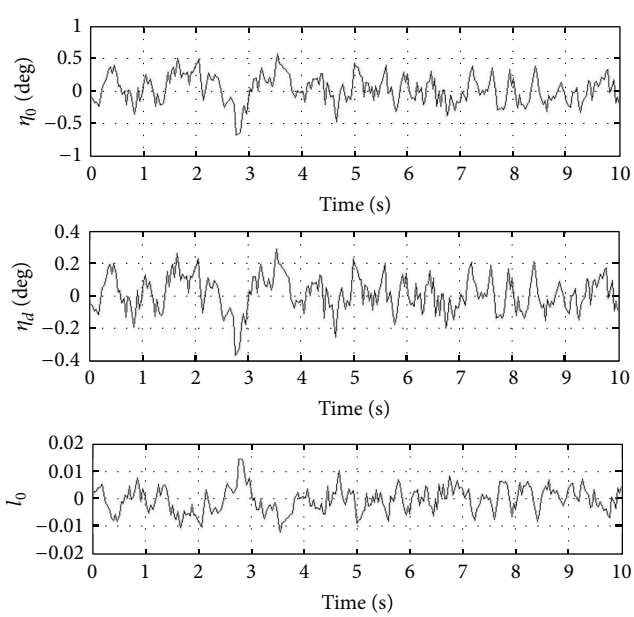


Figure 8: Responses of helicopter MHT controls.

l_0: Control parameter of distance between helicopter center of gravity and horizontal tail, []

$\beta_0, \beta_c, \beta_s$: Collective and two cyclic blade flapping angles, [rad]

$\zeta_0, \zeta_c, \zeta_s$: Collective and two cyclic blade lagging angles, [rad]

$\theta_0, \theta_c, \theta_s$: Collective and two cyclic blade pitch angles, [rad]

θ_T: Collective tail rotor angle, [rad]

η_0, η_d: Collective and differential moving horizontal tail angles, [rad]

FCS: Flight control system

MHT: Moving horizontal tail

SPSA: Simultaneous perturbation stochastic approximation.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by Research Fund of the Erciyes University, Project no. FBA-2013-4179.

References

[1] W. J. Wagtendonk, *Principles of Helicopter Flight*, Aviation Supplies & Academics, 2nd edition, 2006.

[2] R. Celi, "Stabilization of helicopter blades with severed pitch links using trailing-edge flaps," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 4, pp. 585–592, 2003.

[3] O. Dieterich, B. Enenkl, and D. Roth, "Trailing edge flaps for active rotor control aeroelastic characteristics of the ADASYS rotor system," in *Proceedings of the American Helicopter Society 62nd Annual Forum*, pp. 965–986, Phoenix, Ariz, USA, May 2006.

[4] B. Glaz, P. P. Friedmann, and L. Liu, "Vibration reduction and performance enhancement of helicopter rotors using an active/passive approach," in *Proceedings of the 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Material Conference*, Schaumburg, Ill, USA, 2008.

[5] J. Shen and I. Chopra, "Swashplateless helicopter rotor with trailing-edge flaps," *Journal of Aircraft*, vol. 41, no. 2, pp. 208–214, 2004.

[6] C. Malpica and R. Celi, "Simulation-based bandwidth analysis of a swashplateless rotor helicopter," in *Proceedings of the 63rd American Helicopter Society International Annual Forum*, pp. 2021–2042, Virginia Beach, Va, USA, May 2007.

[7] J. Falls, A. Datta, and I. Chopra, "Integrated trailing-edge flaps and servotabs for helicopter primary control," *Journal of the American Helicopter Society*, vol. 55, no. 3, Article ID 032005, 5 pages, 2010.

[8] T. Oktay and C. Sultan, "Simultaneous helicopter and control-system design," *Journal of Aircraft*, vol. 50, no. 3, pp. 911–925, 2013.

[9] D. Fusato and R. Celi, "Multidisciplinary design optimization for aeromechanics and handling qualities," *Journal of Aircraft*, vol. 43, no. 1, pp. 241–252, 2006.

[10] R. Ganguli, "Optimum design of a helicopter rotor for low vibration using aeroelastic analysis and response surface methods," *Journal of Sound and Vibration*, vol. 258, no. 2, pp. 327–344, 2002.

[11] T. Oktay and C. Sultan, "Flight control energy saving via helicopter rotor active morphing," *Journal of Aircraft*, vol. 51, no. 6, pp. 1784–1804, 2014.

[12] H. Kang, H. Saberi, and F. Gandhi, "Dynamic blade shape for improved helicopter rotor performance," *Journal of the American Helicopter Society*, vol. 55, no. 3, Article ID 32008, 11 pages, 2010.

[13] S. Barbarino, F. Gandhi, and S. D. Webster, "Design of extendable chord sections for morphing helicopter rotor blades," *Journal of Intelligent Material Systems and Structures*, vol. 22, no. 9, pp. 891–905, 2011.

[14] J. Sherry, "Helicopter stabilizer," U.S. Patent 2,630,985, March 10, 1953.

[15] J. Stuart, "Horizontal tail plane for helicopters," U.S. Patent 2,979,286, 1961.

[16] J. Howlett, "UH-60 Black Hawk engineering simulation program," NASA Contractor Report 166309, 1981.

[17] J. E. Bluman and F. S. Gandhi, "Reducing trailing edge flap deflection requirements in primary control with a movable horizontal tail," *Journal of the American Helicopter Society*, vol. 56, no. 3, Article ID 032005, 12 pages, 2011.

[18] J. E. Bluman, *Reducing trailing edge flap deflection requirements in primary control with a moveable horizontal tail [M.S. thesis]*, 2008.

[19] D. Fusato, G. Guglieri, and R. Celi, "Flight dynamics of an articulated rotor helicopter with an external slung load," *Journal of the American Helicopter Society*, vol. 46, no. 1, pp. 3–13, 2001.

[20] V. Sahasrabudhe, M. Tischler, R. Cheng, A. Stumm, and M. Lavin, "Balancing CH-53K handling qualities and stability margin requirements in the presence of heavy external loads," in *Proceedings of the American Helicopter Society 63rd Forum*, Virginia Beach, Va, USA, May 2007.

[21] P. Apkarian, C. Champetier, and J.-F. Magni, "Design of a helicopter output feedback control law using modal and structured-robustness techniques," *International Journal of Control*, vol. 50, no. 4, pp. 1195–1215, 1989.

[22] C. M. Ivler, M. B. Tischler, and J. D. Powell, "Cable angle feedback control systems to improve handling qualities for helicopters with slung loads," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Portland, Ore, USA, August 2011.

[23] Z. Jiang, J. Han, Y. Wang, and Q. Song, "Enhanced LQR control for unmanned helicopter in hover," in *Proceedings of the 1st International Symposium on Systems and Control in Aerospace and Astronautics*, pp. 1438–1443, January 2006.

[24] L. Yi-Bo, L. Wan-Zhu, and S. Qi, "Improved LQG control for small unmanned helicopter based on active model in uncertain environment," in *Proceedings of the International Conference on Electronics, Communications and Control (ICECC '11)*, pp. 289–292, Zhejiang, China, September 2011.

[25] C. C. Luo, R. F. Liu, C. D. Yang, and Y. H. Chang, "Helicopter H_∞ control design with robust flying quality," *Aerospace Science and Technology*, vol. 7, no. 2, pp. 159–169, 2003.

[26] R. Kureemun, D. J. Walker, B. Manimala, and M. Voskuijl, "Helicopter flight control law design using H_∞ techniques," in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference (CDC-ECC '05)*, pp. 1307–1312, Seville, Spain, December 2005.

[27] C. L. Bottasso and L. Riviello, "Rotor trim by a neural model-predictive auto-pilot," in *Proceedings of the 31st European Rotorcraft Forum*, pp. 1–12, Florence, Italy, September 2005.

[28] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, "Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 4, pp. 818–831, 2011.

[29] R. E. Skelton, *Dynamic Systems Control: Linear Systems Analysis and Synthesis*, chapter 8, John Wiley & Sons, 1987.

[30] R. E. Skelton, T. Iwasaki, and K. M. Grigoriadis, *A Unified Algebraic Approach to Linear Control Design*, The Taylor & Francis Systems and Control Book Series, chapter 4, Taylor & Francis, London, UK, 1998.

[31] T. Oktay, *Constrained control of complex helicopter models [Ph.D. dissertation]*, Virginia Tech, 2012.

[32] T. Oktay and C. Sultan, "Variance-constrained control of maneuvering helicopters with sensor failure," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 227, no. 12, pp. 1845–1858, 2013.

[33] T. Oktay and C. Sultan, "Modeling and control of a helicopter slung-load system," *Aerospace Science and Technology*, vol. 29, no. 1, pp. 206–222, 2013.

[34] T. Oktay and C. Sultan, "Variance constrained control of maneuvering helicopters," in *Proceedings of the 68th American Helicopter Society International Annual Forum*, Fort Worth, Tex, USA, May 2012.

[35] T. Oktay and C. Sultan, "Integrated maneuvering helicopter model and controller design," in *AIAA Guidance, Navigation, and Control Conference*, Minneapolis, Minn, USA, August 2012.

[36] T. Oktay and C. Sultan, "Robustness of variance constrained controllers for complex, control oriented helicopter models," in *Proceedings of the American Control Conference (ACC '13)*, pp. 794–799, IEEE, Washington, DC, USA, June 2013.

[37] T. Oktay, "Performance of minimum energy controllers on tiltrotor aircraft," *Aircraft Engineering and Aerospace Technology*, vol. 86, no. 5, pp. 361–374, 2014.

[38] R. E. Skelton and M. DeLorenzo, "Space structure control design by variance assignment," *Journal of Guidance, Control, and Dynamics*, vol. 8, no. 4, pp. 454–462, 1985.

[39] R. E. Skelton and C. Sultan, "Controllable tensegrity: a new class of smart structures," in *Smart Structures and Materials 1997: Mathematics and Control in Smart Structures*, vol. 3039 of *Proceedings of SPIE*, San Diego, Calif, USA, March 1997.

[40] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Transactions on Automatic Control*, vol. 37, no. 3, pp. 332–341, 1992.

[41] C. Sultan, "Proportional damping approximation using the energy gain and simultaneous perturbation stochastic approximation," *Mechanical Systems and Signal Processing*, vol. 24, no. 7, pp. 2210–2224, 2010.

[42] T. Oktay and C. Sultan, "Constrained predictive control of helicopters," *Aircraft Engineering and Aerospace Technology*, vol. 85, no. 1, pp. 32–47, 2013.

[43] G. D. Padfield, *Helicopter Flight Dynamics*, AIAA Education Series, AIAA, 2007.

[44] C. Hsieh, R. E. Skelton, and F. M. Damra, "Minimum energy controllers with inequality constraints on output variances," *Optimal Control Applications and Methods*, vol. 10, no. 4, pp. 347–366, 1989.

[45] G. Zhu and R. E. Skelton, "Mixed L_2 and L_∞ problems by weight selection in quadratic optimal control," *International Journal of Control*, vol. 53, no. 5, pp. 1161–1176, 1991.

[46] C. Sultan, "Decoupling approximation design using the peak to peak gain," *Mechanical Systems and Signal Processing*, vol. 36, no. 2, pp. 582–603, 2013.

[47] Y. He, M. C. Fu, and S. I. Marcus, "Convergence of simultaneous perturbation stochastic approximation for nondifferentiable optimization," *IEEE Transactions on Automatic Control*, vol. 48, no. 8, pp. 1459–1463, 2003.

[48] P. Sadegh and J. C. Spall, "Optimal random perturbations for stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Transactions on Automatic Control*, vol. 43, no. 10, pp. 1480–1484, 1998.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 403236, 11 pages
http://dx.doi.org/10.1155/2015/403236

Research Article

A Robust H_∞ Controller for an UAV Flight Control System

J. López,[1] R. Dormido,[2] S. Dormido,[2] and J. P. Gómez[1]

[1] *Dynamic Systems Research Group, Universidad Politécnica de Madrid (ETSIA/EUITA), Plaza Cardenal Cisneros 3, 28040 Madrid, Spain*
[2] *Department of Computer Sciences and Automatic Control, UNED, Calle Juan del Rosal 16, 28040 Madrid, Spain*

Correspondence should be addressed to R. Dormido; raquel@dia.uned.es

Received 27 November 2014; Revised 14 January 2015; Accepted 19 February 2015

Academic Editor: Zheng Zheng

The objective of this paper is the implementation and validation of a robust H_∞ controller for an UAV to track all types of manoeuvres in the presence of noisy environment. A robust inner-outer loop strategy is implemented. To design the H_∞ robust controller in the inner loop, H_∞ control methodology is used. The two controllers that conform the outer loop are designed using the H_∞ Loop Shaping technique. The reference vector used in the control architecture formed by vertical velocity, true airspeed, and heading angle, suggests a nontraditional way to pilot the aircraft. The simulation results show that the proposed control scheme works well despite the presence of noise and uncertainties, so the control system satisfies the requirements.

1. Introduction

There is a considerable and great interest in using unmanned air vehicles (UAVs) to perform a multitude of tasks [1]. UAVs are gaining more powerful skills to accomplish a wide range of missions with high efficiency and high accuracy rate. They are becoming vital warfare and homeland security platforms because they significantly reduce both the costs and the risk to human life and first-responder capabilities. UAVs have many typical applications such as intervention in industrial plants, natural disasters intervention, cooperation with other ground robots in demining operations, through aerial mapping, remote environmental research, pollution assessment and monitoring, fire-fighting management, security, for example, border monitoring, law enforcement, scientific missions, agricultural and fisheries applications, oceanography, or communications relays for wideband applications. Due to their numerous benefits, it would be nice to decrease the global cost of this type of aircraft. In this sense, the flight control design problem for low cost UAV still requires significant efforts, being the control and dynamic modeling of UAVs which is an attractive field of research.

The control of UAVs is not an easy task as the UAV is a multi-input multioutput (MIMO), under actuated, unstable, and highly coupled system. Many traditional control strategies have been used over the years for the control of UAVs, such as linear quadratic regulator (LQR) [2, 3].

Robust techniques have also been applied to design controllers to achieve robust performance and simultaneously guarantee stability when system deviates from its nominal design condition and/or is subjected to exogenous disturbances. In particular, robust H_∞ control method by Zames [4, 5] has been used in flight control systems for both lateral and longitudinal dynamics of aircraft [6–8].

In this work, an inner-outer loop control architecture applied to the longitudinal and lateral flight motions is implemented using the H_∞ Loop Shaping Design procedure [9, 10] to synthesize the inner-loop controller. The technique decouples the longitudinal and lateral dynamics and minimizes the cross effects involved. The feasibility of the controller is analyzed.

The control scheme is implemented on a 6-DOF nonlinear simulation model. Different simulation results are presented to show the robustness of the proposed control architecture. The paper is structured as follows. Section 2 presents the aircraft model and its linearization. Section 3 describes the control problem, presenting the control objectives and the control scheme. Design results are analyzed in Section 4. Flight test results are presented in Section 5.

Figure 1: KUAV scale model.

2. Aircraft Model

2.1. Fully Nonlinear Dynamic Model. The UAV is a 1/3 scaled down model of a Diamond Katana DA-20 shown in Figure 1.

The main characteristics of the aircraft are as follows:

(i) span 3.9 m,

(ii) wing surface 1.47 square meters,

(iii) mean aerodynamic chord 0.39 m,

(iv) mass 18–30 kg,

(v) cruise velocity 130 km/h,

(vi) maximum velocity 200 km/h,

(vii) engine power 8 HP,

(viii) centre of gravity between 15 and 31% of mean aerodynamic chord.

Aircraft dynamics is described as a full 6-degree-of-freedom (DOF) 13-state high fidelity UAV nonlinear model. The nonlinear model has been developed in standard body axes centered at the aircraft center of gravity where x points forward, through the aircraft noise, y is directed to the starboard (right), and z is directed through the belly of the aircraft.

Using the notation given by Stevens and Lewis [11], the flight dynamic model that describes the rigid body motion of the aircraft is given by the following equations.

Force equations are as follows:

$$\begin{aligned} \dot{U} &= RV - QW + g_x + \frac{F_x}{m}, \\ \dot{V} &= -RU + PW + g_y + \frac{F_y}{m}, \\ \dot{W} &= QU - PW + g_z + \frac{F_z}{m}. \end{aligned} \tag{1}$$

Moment equations are as follows:

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} L \\ M \\ N \end{bmatrix} - \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \mathbf{J} \begin{bmatrix} P \\ Q \\ R \end{bmatrix}. \tag{2}$$

Kinematic equations are as follows:

$$\begin{aligned} \dot{\phi} &= P + \tan\theta \left(Q \sin\phi + R \cos\phi\right), \\ \dot{\theta} &= Q \cos\phi - R \sin\phi, \\ \dot{\psi} &= \frac{\left(Q \sin\phi + R \cos\phi\right)}{\cos\theta}. \end{aligned} \tag{3}$$

Navigation equations are as follows:

$$\begin{bmatrix} \dot{p}_N \\ \dot{p}_E \\ -\dot{h} \end{bmatrix} = \mathbf{B}^{-1} \begin{bmatrix} U \\ V \\ W \end{bmatrix}, \tag{4}$$

where m is the mass; (U, V, W) are the body axis velocity states; (P, Q, R) are the body axis rates; ϕ, θ, ψ are the roll, pitch, and yaw angles, respectively; and (p_N, p_E, h) are the north, east, and height positions. $F = (F_x, F_y, F_z)$ represents the aerodynamic force vector and $M = (L, M, N)$ represents the moment vectors. J is the aircraft inertia:

$$J = \begin{bmatrix} J_{XX} & 0 & -J_{xz} \\ 0 & J_{yy} & 0 \\ -J_{xz} & 0 & J_{zz} \end{bmatrix}; \tag{5}$$

$\mathbf{B}$ is the inertial top body transformation matrix; (g_x, g_y, g_z) is the gravity vector, which is the transformation of the $(0, 0, g)$ NED-frame gravity vector to the body axis frame, as shown below:

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = \mathbf{B} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}, \tag{6}$$

where

$$\mathbf{B} = \begin{bmatrix} \cos\theta \cos\psi & \cos\theta \sin\psi & -\sin\theta \\ -\cos\theta \sin\psi + \sin\phi \sin\theta \sin\psi & \cos\theta \cos\psi + \sin\phi \sin\theta \sin\psi & \sin\phi \cos\theta \\ \sin\theta \sin\psi + \cos\phi \sin\theta \cos\psi & -\sin\theta \cos\psi + \cos\phi \sin\theta \sin\psi & \cos\phi \cos\theta \end{bmatrix}. \tag{7}$$

The resulting model is described by a thirteen-state order model [12]. Due to the complexity and the uncertainty inherent to aerodynamic systems, the dynamic model was identified by a complete identification flight set through the full envelope. See Stevens and Lewis for details [11].

2.2. Linearized Dynamic Model. The nonlinear dynamic model described in Section 2.1. is linearized about certain trimmed operating conditions. This process is accomplished by perturbing the state and control variables from steady state.

The mathematical formulation of the dynamic system is modeled with standard continuous time invariant state space formulation given by (8). Where A is a 13×13 matrix, B a 13×4 matrix, C a 12×13 matrix, and D is a 12×4 matrix,

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du, \end{aligned} \tag{8}$$

and the state, output, and control vectors are, respectively,

$$\begin{aligned} x &= \left[V_T \;\; \alpha \;\; \beta \;\; \phi \;\; \theta \;\; \psi \;\; P \;\; Q \;\; R \;\; p_N \;\; p_E \;\; h \;\; \text{pow}\right]^T, \\ y &= \left[a_x \;\; a_y \;\; a_z \;\; P \;\; Q \;\; R \;\; \text{lon} \;\; \text{lat} \;\; h \;\; \dot{p}_N \;\; \dot{p}_E \;\; \dot{h}\right]^T, \\ u &= \left[\delta_{tl} \;\; \delta_e \;\; \delta_a \;\; \delta_r\right]^T. \end{aligned} \tag{9}$$

The state vector (x) components are true airspeed (V_T), angle of attack (α), sideslip angle (β), roll angle (ϕ), pitch angle (θ), yaw angle (ψ), roll rate (P), pitch rate (Q), yaw rate (R), north position (p_N), east position (p_E), altitude (h), and power (pow).

The output vector (y) is formed by x-component of acceleration (a_x), y-component of acceleration (a_y), z-component of acceleration (a_z), roll rate (P), pitch rate (Q), yaw rate (R), longitude (lon), latitude (lat), altitude (h), north position derivative, east position derivative, and altitude derivative.

The control vector (u) is defined by throttle (δ_{tl}), elevator (δ_e), aileron (δ_a), and rudder (δ_r).

The dynamics are linearized about a representative flight condition. This nominal condition is $V_T = 30\ \text{ms}^{-1}$, centre of gravity position equal to 25% of mean aerodynamic chord, $\phi = 0$ rad, $\psi = 0$ rad, $R = 0$ rad, $P = 0$ rad, $\theta = 0$ rad, rate of climb = 0 rad, and lateral acceleration = 0 rad.

3. Control Technique

3.1. Control Objectives. The main objective is the design of a robust controller to track all types of input commands in a noisy environment. The controller has to be designed as a trade-off robustness and performance in order to fulfill the specifications described in this section.

3.1.1. Closed Loop Specifications. Stability of the aircraft, minimal overshoot, and reasonably long settling time are important constraints in the design. Translated into physical design goals, the controller must perform the following specifications:

(i) altitude response: overshoot < 5%, rise time < 5 s, and settling time < 20 s,

(ii) heading angle response: overshoot < 5%, rise time < 3 s, and settling time < 10 s,

(iii) flight path angle response: overshoot < 5%, rise time < 1 s, and settling time < 5 s,

(iv) airspeed response: overshoot < 5%, rise time < 3 s, and settling time < 10 s,

(v) cross coupling between airspeed and altitude: for a step in commanded altitude of 30 m, the peak value of the transient of the absolute error between airspeed and commanded airspeed should be smaller than $0.5\ \text{ms}^{-1}$; conversely, for a step in commanded airspeed of $2\ \text{ms}^{-1}$, the peak value of the transient of the absolute error between altitude and commanded altitude should be smaller than 5 m.

3.1.2. Gust Rejection. Second objective of the control system is to include robustness to gust effects on the aircraft. In this sense, turbulence can be considered as a stochastic process defined by its velocity spectra. For an aircraft flying at a cruise speed U, a commonly used velocity spectra for turbulence model is the Dryden spectra [13]:

$$\Phi_v = \frac{2L_v\sigma^2\left(1 + 12\left(L_v/U\right)^2 w^2\right)}{\pi U\left(1 + 4\left(L_v/U\right)^2 w^2\right)^2}, \tag{10}$$

where w is the frequency in rad s^{-1}, σ is the turbulence standard deviation, and L_v is the turbulence scale length. The turbulence parameters values for severe gust conditions are given by [14]

$$\begin{aligned} \sigma &= 0.1 + 0.00733h, \quad 300 < h < 600\ \text{m} \\ \sigma &= 3.04 + 0.00244h, \quad 600 < h < 1400\ \text{m} \\ \sigma &= 6.45\ \text{m/s}, \quad 1400 < h < 5800\ \text{m} \\ \sigma &= 8.40 - 0.000336h, \quad h > 5800\ \text{m} \\ L_v &= \frac{h}{(0.177 + 0.00274h)^{1.2}}, \end{aligned} \tag{11}$$

where h is the altitude. Our gust rejection specification is to reject all disturbances below 13 rad s^{-1}.

3.1.3. Noise Rejection. Basically, the measured variables for the lateral control are the lateral acceleration and the yaw and roll rates measured in body fixed axis. For the selected sensors, the noise is high and concentrated in the frequency range above 30 rad s^{-1}. Thus, high frequency specification is that in which all noise spectra, which normally occur above 30 rad s^{-1}, should be rejected.

3.1.4. Robustness Specifications. The controller designed has to be robust against uncertainty in the plant model. The robust specifications are defined as follows.

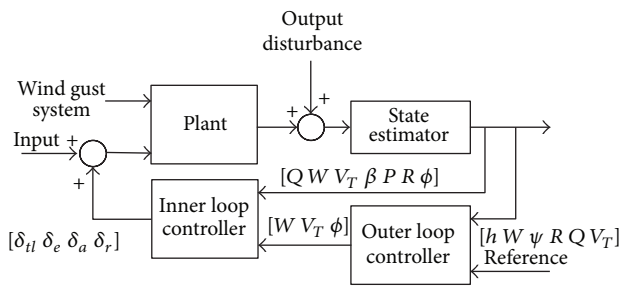


Figure 2: Controller architecture.

(i) Centre of gravity variation is as follows: stability and sufficient performance should be maintained for horizontal centre of gravity variations between 15% and 31% cbar (mean aerodynamic chord).

(ii) Vertical centre of gravity must not suffer variations: it should remain at 0% cbar.

(iii) Mass variations are as follows: stability and sufficient performance should be maintained for aircraft mass variations between 18 and 30 kg.

(iv) Time delay is as follows: stability and sufficient performance should be maintained for transport delays from 0 to 60 ms.

(v) Speed variations are as follows: stability and sufficient performance should be maintained for speed variations from 1.23VS (stall velocity) to $55\,\mathrm{m\,s^{-1}}$ (200 Km/h).

3.2. Controller Design. The control architecture is based on that proposed by Tucker and Walker [13]. As Figure 2 shows, basically, it consists of two loops: an inner-loop controller to achieve stability and robustness to expected parameter uncertainty and an outer loop for tracking reference performances.

The design of the inner loop is focused on maintaining the vertical velocity deviation, the heading angle deviation, and the airspeed deviation near zero.

Two different controllers conform to the outer loop: the altitude controller and the heading angle-lateral deviation controller. Both controllers are synthesized using the H_∞ Loop Shaping technique (see [9, 15, 16]).

Figure 3 shows the general framework used in the design process.

3.2.1. The Inner Loop Synthesis Procedure. Figure 4 shows the inner loop architecture. Its main goal is to minimize both the deviation to desired output and the control effort. $r_i \in R^3$ is the reference input vector, whose components are the vertical speed, airspeed, and the roll angle. $u \in R^4$ is the control signal. $z_1 \in R^3$ is the vector of performance outputs. $z_2 \in R^2$ is the vector of weighted control inputs. The feedback variables are the vertical speed, airspeed, the roll angle, the pitch rate, the yaw rate, the roll rate, and the sideslip.

The total plant G_{total} is formed by the plant G (the linearized UAV model), the actuators model, and the corresponding delays. These delays are modelized using the first order Pade approximations. They are used to represent plant uncertainties in the high frequency range such as modeling errors and neglected actuator dynamics. Four delays of 100 ms are included in the plant model, one in each input including the throttle.

The actuator model for δ_e, δ_a, and δ_r is given by the first-order linear approximation $10/(s+10)$ and the engine model is represented by $2/(s+2)$.

The sensor noise is represented by means of white noise model. The standard deviations of the sensor noise corresponding to the output vector are $0.1\,\mathrm{ms^{-2}}$ for accelerations, $0.005\,\mathrm{rad\,s^{-1}}$ for angular velocity, 5 m for position, and $0.5\,\mathrm{ms^{-1}}$ for velocity.

The controller K is designed using the H_∞ technique. It must guarantee the stability and follow an ideal model, the so called matching model (M). That is, the closed loop system output $y1$ is expected to match $y_m \in R^3$, the output of the ideal model M.

The matching model M, which defines the behaviour of the vertical speed, the true speed, and the heading angle, consists of the following three second-order systems:

$$M = \begin{bmatrix} \dfrac{4^2}{s^2+2\cdot 4s+4^2} & 0 & 0 \\ 0 & \dfrac{1.5^2}{s^2+2\cdot 1.5s+1.5^2} & 0 \\ 0 & 0 & \dfrac{2.25^2}{s^2+2\cdot 2.25s+2.25^2} \end{bmatrix}. \tag{12}$$

The matching model is selected to accomplish desired behaviour of the vertical speed, airspeed, and roll angle to achieve the closed loop specifications detailed in Section 3.1. The cross coupling terms are zero, thus, defining the requirement for closed loop system as decoupled.

Four weights W_i ($i = 1, \ldots, 4$) are used in the inner loop to accomplish the frequency dependent specifications on performance and robustness. They are added to maximize disturbances rejection and to minimize wind gusts effects and sensor noises.

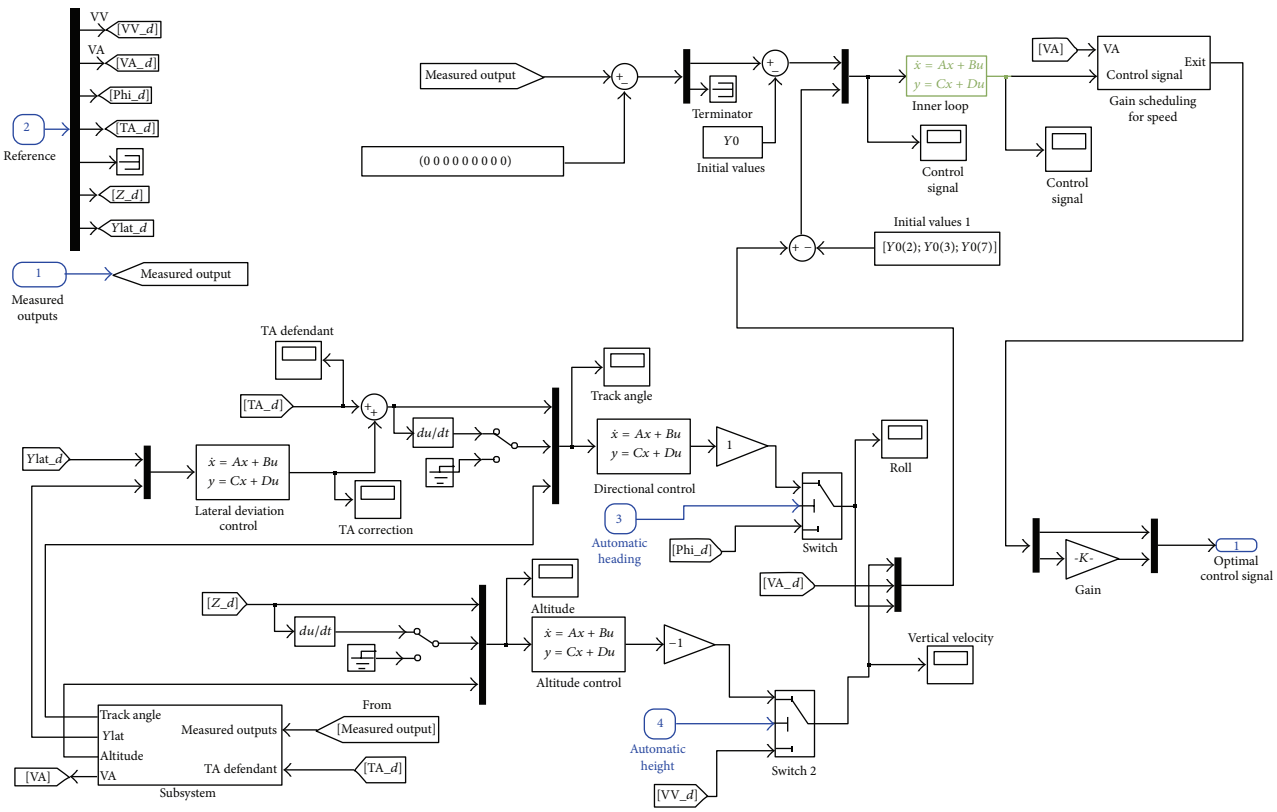


FIGURE 3: Framework to design the H_∞ controller.

W_1 is related with reference tracking. So, its elements are selected as low pass filters. The yaw rate and roll rate are selected as pass band filters.

W_2 is devoted to minimize the control effort. This is why it is selected as a high pass filter, where its gain and bandwidth are chosen to allow low frequency control effort and to minimize high frequency control effort.

W_3 and W_4 are unity matrix. They weight turbulences and output disturbances, respectively.

The controller's synthesis is accomplished using an iterative procedure. First, the weights are selected; then the controller K is synthesized and finally the resulting system performances are analysed.

After this iterative process, the weights selected are the following:

$$W_1 = \operatorname{diag}\left(\frac{3^2(s+1)}{s+2\cdot 3s+3^2}, 10\frac{500s}{s/0.001+1}, 10\frac{500s}{s/0.001+1}, 5\frac{500s}{s/0.001+1}, \frac{7^2(s+1)}{s+2\cdot 7s+7^2}, \frac{7^2(s+1)}{s+2\cdot 7s+7^2}, 8\frac{500s}{s/0.001+1}\right),$$

$$W_2 = \operatorname{diag}\left(0.5\frac{s/0.1+1}{s/0.008+1}, 0.5\frac{s/0.1+1}{s/0.008+1}, 0.5\frac{s/0.1+1}{s/0.008+1}, 0.5\frac{s/0.1+1}{s/0.008+1}\right),$$

$$W_3 = I_3,$$

$$W_4 = I_7. \tag{13}$$

After some iterations, a stabilizing controller $K(s)$ is determined. This controller minimizes the variables z_1 y z_2 (see Figure 3) which corresponds to deviation between the desired output, provided by the matching model and the real aircraft output and control effort. The subresulting suboptimal robust stability margin is $\gamma = 4.98$.

3.2.2. The Outer Loop Synthesis Procedure. Two different controllers conform to the outer loop: the altitude controller (see Figure 5) and the heading angle-lateral deviation controller (see Figure 6). The two outer-loop controllers are synthesized using the H_∞ Loop Shaping technique [16].

Figure 5 shows the first problem to be solved, where C is the controller and W_1 and W_2 are the weights used to tune the optimization. The simplified models of the plant used to synthesize these controllers are those defined in the matching model.

In the design of the altitude controller, an output integrator is used to provide height and vertical velocity outputs. An input integrator is used to improve the low frequency behaviour.

In a similar way in the heading angle-lateral deviation controller design an output integrator is used to provide yaw

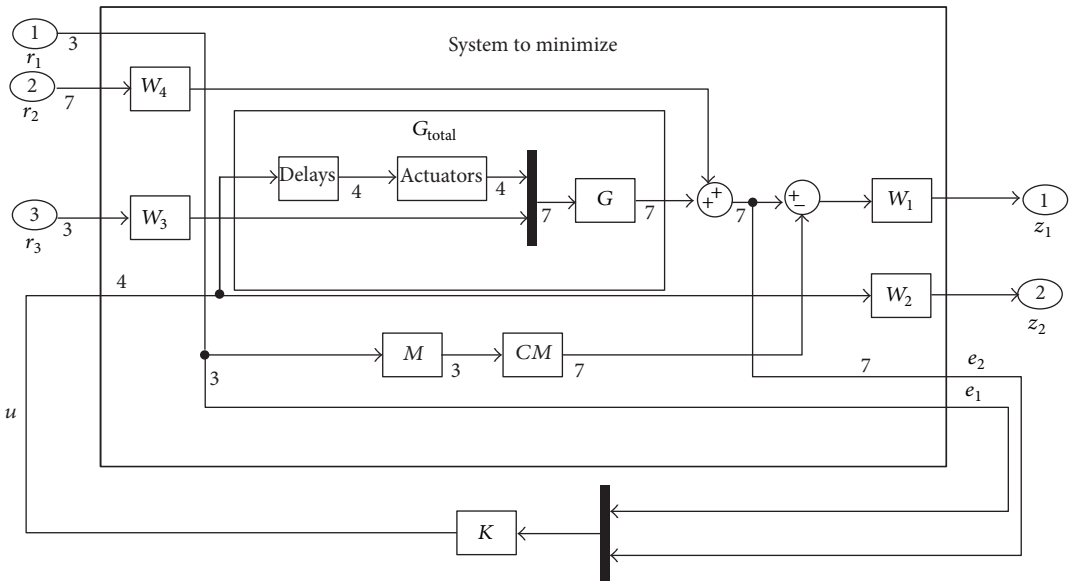


Figure 4: Inner loop architecture.

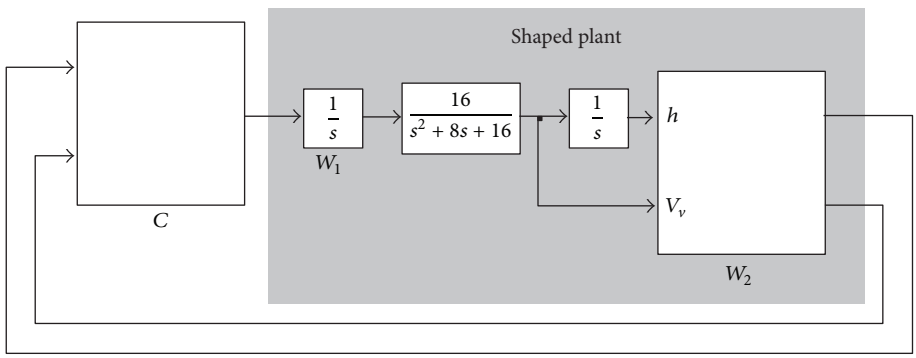


Figure 5: Outer-loop altitude command tracker.

angle and its derivative outputs. An input integrator is used to improve the low frequency behaviour.

The gamma values encountered are 3.18 and 2.5 for the altitude controller and the heading angle-lateral deviation controller, respectively.

The heading angle and lateral deviation controllers have been built together due to the hard interaction between the variables implied which motivates a tedious iterative process when individual controllers were designed. In this approach, these two controllers are synthesized jointly.

4. Design Results

The performance of a system can be represented by the sensitivity function S. The maximum singular value of S is an important boundary in this case. By using the largest singular value, we are effectively assessing the worst case scenario. Performance specification means the minimization of the sensitivity function as much as possible for low frequencies. At the same time, control effort should be small in the high frequency range.

Figure 7 shows the sensitivity function. It is easy to see that our goal of minimising the sensitivity at low frequencies has been achieved. At high frequencies, the gain is unity and around the bandwidth there is a peak in the response.

This behaviour of the sensitivity enables good tracking reference at the low frequency range and noise reduction and robustness in the high frequency range.

Figure 8 shows the control effort behaviour which is lower in the high frequency range as it was expected.

Since the H_∞ controller designed produces a 46-size state space realization, it is necessary to apply controller reduction techniques. A final state realization for the controller of dimension 27 is achieved using Hankel minimum degree approximation (MDA) without balancing reduction method [17].

This method has been applied iteratively checking the frequency and time responses every step to evaluate the performance of the proposed UAV control scheme. One example of the time response in one step of this iterative process is shown in Figure 9.

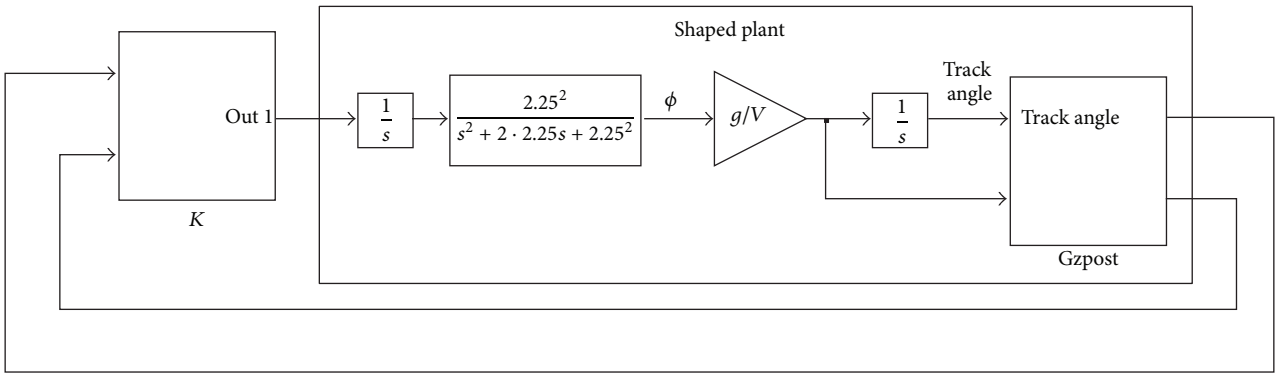


Figure 6: Outer-loop heading angle and lateral deviation controller.

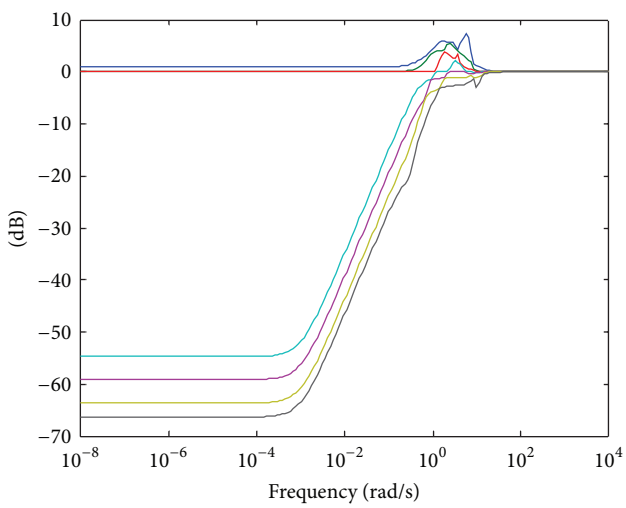


Figure 7: Singular values of the sensitivity function.

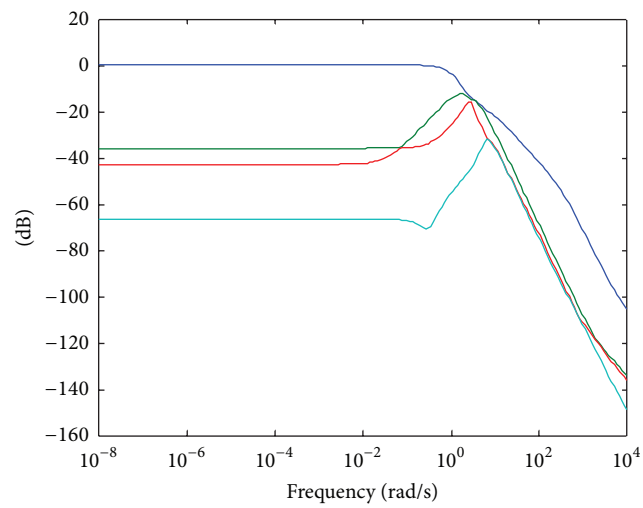


Figure 8: Singular values of the control effort.

Figure 10 shows the effect of an incorrect order reduction. This performance is obtained when an order reduction is forced and the reduced controller is not able to maintain the desired specifications.

5. Simulation Tests Results

In order to validate the controller designed, a set of test cases have been developed. Below, an experience corresponding to 45-degree heading angle step response is shown. The results allow checking the performance of the aircraft in a noisy environment along this type of manoeuvre.

The airplane desired reference is illustrated in Figure 11. The dashed line is the desired trajectory.

Figure 12 shows the airplane simulated trajectory tracked. The dashed line is the desired trajectory and the continuous line is the real one.

The controller is able to manage adequately the output and to calculate the control vector. Control variables evolutions are shown in Figure 13. The throttle varies around 2% and elevator, ailerons, and rudder present a smooth behaviour. The aileron and rudder are deflected by the controller to order the 45-degree change of direction. Immediately, a sustentation loose typical in this type of manoeuvers is suffered by the aircraft. To compensate this trend, the elevator acts to raise the noise of the aircraft and slightly increase the throttle to maintain the velocity.

Figure 13 confirms that the control variables remain far from its saturation values. The power demand is less than 40% and the elevator, aileron, and rudder demanded deflections are less than 5 degrees. In this case, if the altitude holder is not connected, in 5 s, the airplane suffers an altitude loss of 3 m and rapidly it recovers the desired altitude, in about 5 s more. The UAV quickly corrects its heading angle turning to reduce the error. In about 4.5 s, the error is null; however, the airplane continues turning. This is produced because of the lateral deviation. If the airplane stops its turning movement in 4.5 s, it would continue straight ahead along a parallel line to the desired trajectory. To reduce the lateral deviation, it must continue turning and augmenting, in a first stage, the heading angle error. Following this strategy, the controller gains its tracking heading angle and its lateral deviation reduction goal.

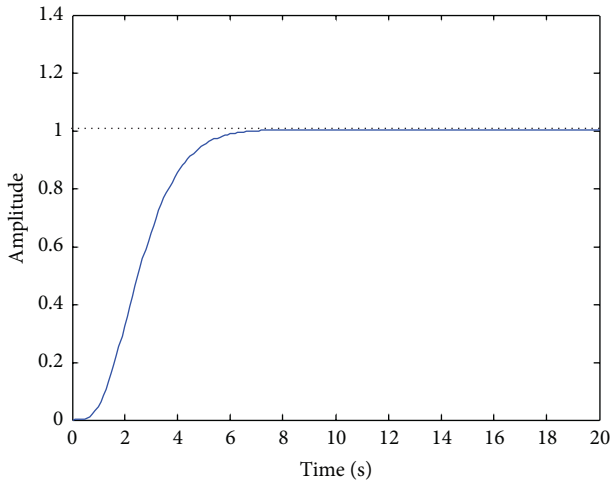


Figure 9: Lateral deviation step response (correct order reduction).

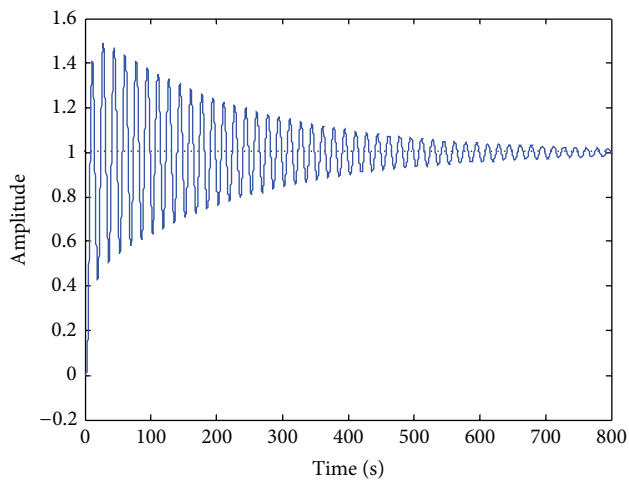


Figure 10: Lateral deviation step response (incorrect order reduction).

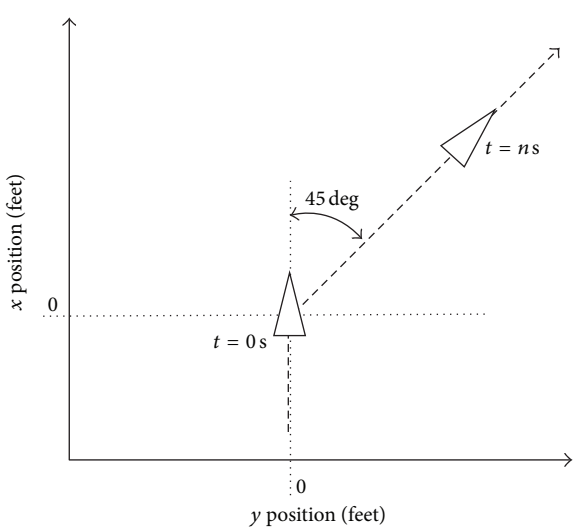


Figure 11: Airplane desired trajectory.

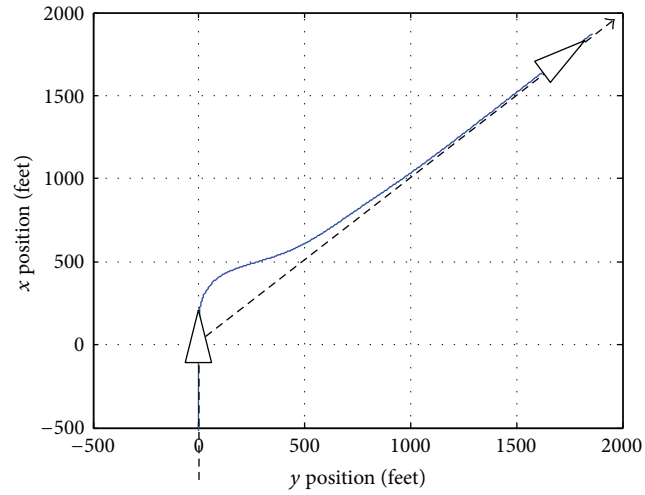


Figure 12: Airplane real trajectory.

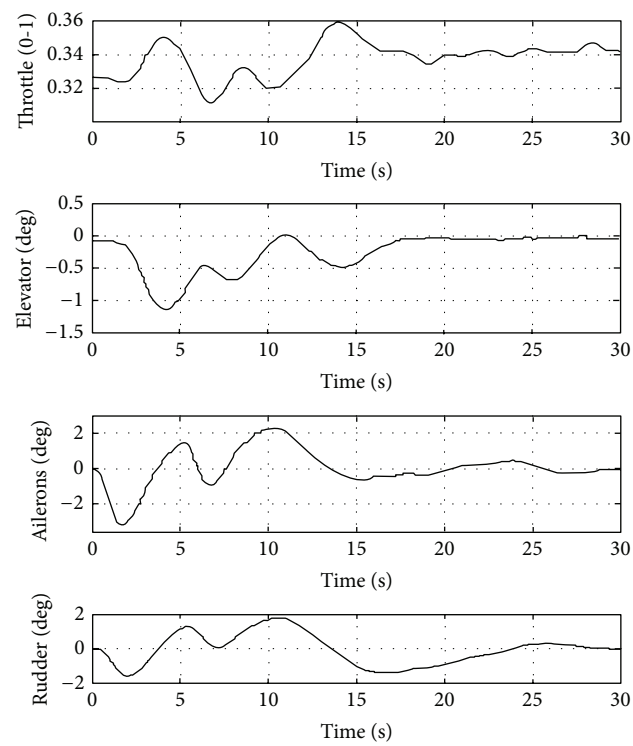


Figure 13: Control variables evolution during the 45-feet heading angle response.

6. Flight Test Results

For testing the whole system and the performance of the controller in flight, many real tests are accomplished. These tests are scheduled to validate in essence the physical design of the UAV, communications equipment, engine capabilities, and onboard software. A very important part of onboard software is the flight control system.

To manage the UAV platform, a ground station is developed (see Figure 14). It enables following the position and the attitude of the aircraft directly on a map shown in

Figure 14: Ground Control Station GUI.

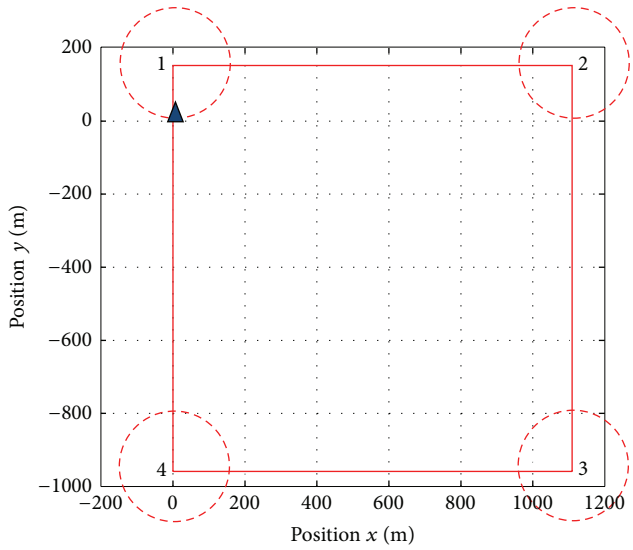


Figure 15: Waypoints and condition goals.

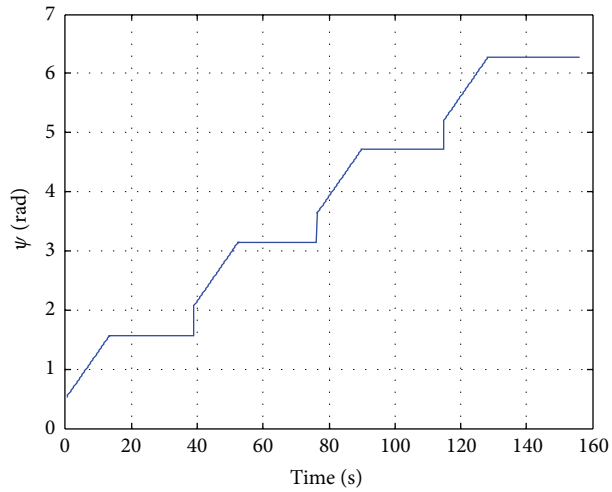


Figure 16: Smooth psi angle reference.

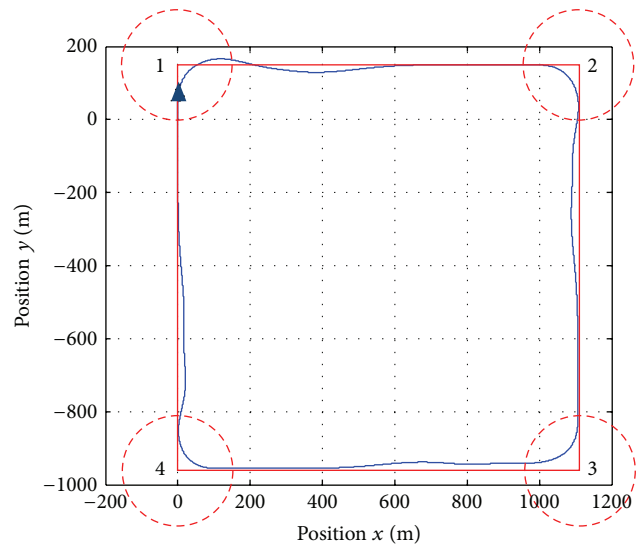


Figure 17: Real trajectory followed by the UAV.

the computer. It also allows showing the main variables of the UAV which are sent through a radio link. The ground station allows introducing a set of waypoints. The autopilot takes care of both navigation and stability of the plane. The mission is planned via waypoints, placing on a geo referred map the position of each waypoint at the beginning of the mission. This mission can be easily modified during its execution by adding/changing/removing waypoints in the map.

The system provides a user-friendly interface used to display the plane position in real time on a map during the mission and to monitor some UAV parameters such as battery levels, speed, position and orientation, or the sensors measurements. The system also provides a radio link which allows a continuous exchange of data between the plane and the control station.

In an emergency case, the aircraft can switch to a PIL (pilot in the loop) mode in which the plane can be teleoperated from the control station by using a control-stick while the onboard autopilot remains on sleep mode.

The selected test to illustrate the aptitudes of the autopilot designed is a circuit formed by four waypoints which is shown in Figure 15. The tracking reference trajectory is shaped for the waypoints labelled from one to four. The circles around the waypoint determine the instance when the reference input changes to the next waypoint (goal condition).

The reference is provided to the autopilot as a psi angle function and is built in a soft way using a combination of a step and a ramp. The reference is shown in the Figure 16.

Figure 17 shows how the UAV is capable of managing adequately the uncertainties and disturbances introduced by the modelling inaccuracies and the noisy output provided by the sensor. The response of the aircraft is not oscillating and it reaches the correct trajectory quickly when covering 600 m approximately which means 20 s at 30 ms^{-1} of mean velocity.

The entire trajectory covered is around 160 s and the psi angle and lateral deviation error are minimized satisfactorily. A desired decoupling between lateral and longitudinal dynamics is achieved.

Figure 18 shows the noisy accelerations output provided by the inertial sensors to the controller.

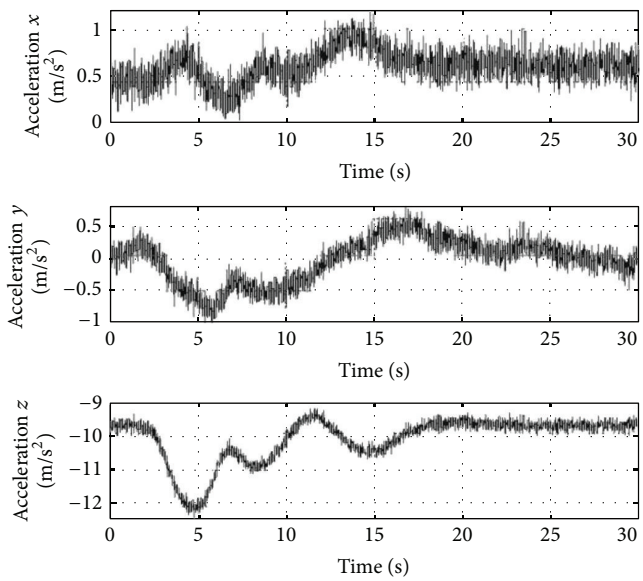


Figure 18: Accelerations measured.

Figure 19: On-board HW equipment.

Figure 20: KUAV during test cases.

Figure 19 shows the onboard equipment mounted on the UAV. In Figure 20, the UAV during the test cases is shown.

7. Conclusions

The dynamics of the UAVs are highly nonlinear and continuously vary with time. Also, it is subjected to severe external disturbances. Due to this, dynamic and parametric uncertainties arise in the mathematical model of the UAVs over different operating conditions. This paper addresses the problem of designing a robust control system for UAVs in the presence of uncertainties using H_∞ technique. The controller implemented allows the UAV to track all types of manoeuvres in the presence of noisy environment. The reference vector used, formed by vertical velocity, true airspeed, and heading angle, suggests a nontraditional way to pilot the aircraft that is based on commanding the desired reference vector and lets the controller select throttle position and surfaces deflections. This kind of pilot-machine interaction appears to be a more intuitive approximation.

The frequency domain analyses show that the proposed controller guarantees good performance, attenuating high frequency noise and also supplying suitable control signals. The tracking performance of the UAV is within the desired tracking performance range. The control efforts during soft manoeuvres are in the same way moderated. The first results obtained with the real UAV with the controller designed appeared to be very suitable.

The desired behaviour is introduced using a matching model (M). This architecture allows modifying the desired performances without varying the controller architecture.

The architecture selected to decouple the longitudinal and lateral dynamics provides very good performances. The outer-loop controller gives a very good behaviour in case of step responses, ramp responses, and combinations of these two input types. It is important to note that the outer loop shows a signal derivative at the input and this should be avoided. This signal derivative is not part of the real implementation. In this case, the inputs of the outer loop are provided directly by the GPS (height and vertical velocity). The specifications and robustness performances have been validated by mean of simulation and real tests.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work has been funded by the National Plan Project DPI2012-31303 of the Spanish Ministry of Economy and Competitiveness and FEDER funds.

References

[1] Office of the Secretary of Defense, "Unmanned aircraft systems roadmap 2005–2030," Tech. Rep., United States Department of Defence, 2005.

[2] S. Franko, "Lqr based trajectory control of full envelope, autonomous helicopter," in *Proceedings of the World Congress on Engineering*, vol. 1, London, UK, July 2009.

[3] V. G. Nair, M. V. Dileep, and V. I. George, "Aircraft yaw control system using lqr and fuzzy logic controller," *International Journal of Computer Applications*, vol. 45, no. 9, pp. 25–30, 2012.

[4] G. Zames, "Feedback and complexity, special plenary lecture addendum," in *Proceedings of the IEEE Conference on Decision and Control*, 1976.

[5] G. Zames, "Optimal sensitivity and feedback: weighted seminorms, approximate inverses, and plant invariant schemes," in *Proceedings of the 17th Allerton Conference*, pp. 744–752, 1979.

[6] M. Giacomn-Zarzar, R. Ramirez-Mendoza, P. J. Fleming, I. Griffin, and A. Molina-Cristbal, "Robust H_∞ controller design for aircraft lateral dynamics using multi-objective optimization and genetic algorithms," in *Proceedings of the 17th IFAC World Congress*, The International Federation of Automatic Control, Seoul, Republic of Korea, July 2008.

[7] R. J. Adams and S. S. Banda, "Robust flight control design using dynamic inversion and structured singular value synthesis," *IEEE Transactions on Control Systems Technology*, vol. 1, no. 2, pp. 80–92, 1993.

[8] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–847, 1989.

[9] D. McFarlane and K. Glover, "A loop shaping design procedure using H_∞ synthesis," *IEEE Transactions on Automatic Control*, vol. 37, no. 6, pp. 759–769, 1992.

[10] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice-Hall, 1996.

[11] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, Wiley-Interscience, New York, NY, USA, 1992.

[12] P. Lambrechts, S. Bennani, G. Looye, and A. Helmersson, *Robust Flight Control Design Challenge. Problem Formulation and Manual: The Research Civil Aircraft Model (RCAM)*, GARTEUR, 1997.

[13] M. R. Tucker and D. J. Walker, "RCAM design challenge presentation document: an H infinite approach," Tech. Rep. GARTEUR TP-088-21, 1997.

[14] U.S. Military Handbook MIL-HDBK-1797, *Flying Qualities of Piloted Aircraf*, 1997.

[15] J. López, R. Dormido, S. Dormido, J. P. Gómez, and I. Gómez, "An H-infinite robust tracker controller for an UAV under realistic simulated environmental effects," in *Proceedings of the 17th World Congress, International Federation of Automatic Control (IFAC '08)*, pp. 6136–6141, July 2008.

[16] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control*, Wiley, 1996.

[17] G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, *μ-Analysis and Synthesis Toolbox: User's Guide*, The MathWorks, 2001.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 742541, 10 pages
http://dx.doi.org/10.1155/2015/742541

Research Article

A Novel Biobjective Risk-Based Model for Stochastic Air Traffic Network Flow Optimization Problem

Kaiquan Cai,[1,2] Yaoguang Jia,[1,2] Yanbo Zhu,[2,3] and Mingming Xiao[1,2]

[1] *School of Electronics and Information Engineering, Beihang University, Beijing 100191, China*
[2] *National Key Laboratory of CNS/ATM, Beijing 100191, China*
[3] *Aviation Data Communication Corporation, Beijing 100191, China*

Correspondence should be addressed to Yanbo Zhu; zyb@adcc.com.cn and Mingming Xiao; buaaxmm@163.com

Received 5 December 2014; Accepted 9 January 2015

Academic Editor: Kemao Peng

Network-wide air traffic flow management (ATFM) is an effective way to alleviate demand-capacity imbalances globally and thereafter reduce airspace congestion and flight delays. The conventional ATFM models assume the capacities of airports or airspace sectors are all predetermined. However, the capacity uncertainties due to the dynamics of convective weather may make the deterministic ATFM measures impractical. This paper investigates the stochastic air traffic network flow optimization (SATNFO) problem, which is formulated as a weighted biobjective 0-1 integer programming model. In order to evaluate the effect of capacity uncertainties on ATFM, the operational risk is modeled via probabilistic risk assessment and introduced as an extra objective in SATNFO problem. Computation experiments using real-world air traffic network data associated with simulated weather data show that presented model has far less constraints compared to stochastic model with nonanticipative constraints, which means our proposed model reduces the computation complexity.

1. Introduction

With the rapid growth of air traffic, airspace congestion and flight delays are becoming serious due to demand-capacity imbalances in air traffic management (ATM) system, especially in convective weather conditions. Severe airspace congestion and flight delays not only deteriorate the service quality of airlines but also raise operational costs. According to statistics, in 2013 [1], the average punctuality rate of flights all over the world was only 80%. China endures an even lower rate, which was 73.1% for big airlines and 70.0% for small airlines. Among all factors that lead to flight delays, the impact of inappropriate flight plan and convective weather takes up a weight of 59.2%.

The network-wide air traffic flow management (ATFM), through constructing global predeparture flight plans (including departure time, flight routes, and arrival time at the waypoints or arrival airports) for all the flights planned to fly over the air traffic network, aims to balance the air traffic demand and ATM system capacity. In ATM domain, considered as one of the effective ways to alleviate airspace congestion and flight delays, ATFM becomes increasingly highlighted in the expanding air transportation system and has drawn a mass of attention to researchers. Single airport ground-holding problem was firstly proposed to minimize ground delays by optimizing departure time of a flight when its arrival airport capacity is insufficient [2]. Later on, the concerns evolved to multiairport ground-holding problem [3–5], air traffic flow management problem [6, 7], and air traffic flow management rerouting problem [8–12]. All above problems can be collectively labeled as air traffic network flow optimization (ATNFO) problem. An air traffic network [13–15] can be modeled as airports network, airspace sector network (nodes include airports and sectors), or air route network. ATNFO aims to minimize the cost of flight delays by optimizing flight plan over a given network and time horizon, taking into account the capacity limits of airports or airspace sectors.

The above ATNFO problems share a common nature; that is, the capacities of airports and airspace sectors are

predetermined. However, due to the stochastic nature of the convective weather dynamics, it is hard to predict airport/airspace capacity accurately in future 24 hours (ATNFO is usually implemented more than 24 hours earlier to provide decision-making) in practice [16]. The convective weather dynamics often lead to capacity uncertainties for the airports or airspace sectors in a network. In the presence of capacity uncertainties, ATFM actions might be noneffective or impractical when adopting the deterministic ATNFO models. Hence, in recent years, the stochastic ATNFO (SATNFO) problem with uncertainties has aroused increasing concerns. Existing models for SATNFO problem can be classified into stochastic programming [17, 18] based methods and robust optimization [19, 20] based methods.

Deterministic models for ATNFO problem assume airports and airspace sectors capacity as constant values. However, stochastic programming based method considers the uncertainties of airports and sectors capacities by establishing probabilistic distributions for them and finds a solution that is feasible and optimal in some sense. The first attempt at stochastic programming in ATFM was made by Richetta and Odoni [21, 22]. They applied scenario tree [23, 24] to model the uncertainty of capacity and proposed a dynamic model for single airport ground-holding problem. Here, scenario tree is a kind of format to describe capacity change and its probability along time horizon. Later, the utilization of scenario tree was extended to stochastic model for multiairport ground-holding problem by Ball et al. [25]. In 2012, Agustín et al. [26] dealt with SATNFO problem under weather uncertainty by using stochastic model, which is an extension of their deterministic tight mixed 0-1 model [11]. In their work, a scenario tree which models the capacity uncertainties under storm scenarios is generated to represent the deterministic equivalent model for their stochastic mixed 0-1 program. Since scenarios in reality are numerous, especially when prolonging the time horizon (such as 24 hours) of ATFM, stochastic programming becomes intractable.

Robust optimization [27, 28] captures the probabilistic property of a problem by constructing appropriate limited uncertainty sets for uncertain parameters and then solves a solution that is feasible for all outcomes of the uncertainty sets. The solution is optimized in worst-case condition. Gupta and Bertsimas [29] and Bertsimas and Goyal [30] attempted to apply the idea of robust optimization to the capacity uncertainties in ATFM rerouting problem. Unfortunately, it is likely to get a very conservative solution and cause waste of airspace and time resource when using robustness optimization based method.

Both stochastic programming and robust optimization have the same basic idea for dealing with capacity uncertainties. Firstly, generate scenarios (scenario tree or uncertainty sets) to describe capacity uncertainties; then, optimize air traffic flow by using capacity scenarios as constraints. Their objective is to find a robust plan for all scheduled flights under uncertainty. But both methods have suffered computational intractability and given highly conservative solution.

In order to solve SATNFO problem efficiently, we propose a novel mathematic model that takes into account capacity uncertainties. In this model, we introduce the concept of operational risk to evaluate the effect of capacity uncertainties in ATFM and formulate it via probabilistic risk assessment methodology. A set of flight plans with lower operational risk is deemed to a more robust solution of SATNFO problem, which means the flight plans are more insusceptible to the capacity uncertainties of airports or airspace sectors. Hence, the SATNFO problem is formulated as a weighted biobjective 0-1 integer programming model with two objectives of minimizing the cost of total delays as well as the operational risks. The contributions of our novel SATNFO model are twofold. First, all appropriate scenarios are considered to evaluate the operational risk of the solution. The introduction of biobjective, that is, operational risk and delay cost, can provide a more efficient solution with appropriate weights and thus eliminate highly conservative solution. Second, without using capacity scenarios as constraints, the number of constraints in this model will not dramatically increase with the increasing number of scenarios. Therefore, we can efficiently avoid computationally intractable problem. Using real-world air traffic network data associated with simulated weather data, computation experiments demonstrate that our proposed model has generality that is suitable for solving SATNFO problem under different capacity scenarios. Moreover, the number of constraints is far less than stochastic model with nonanticipative constraints, which means the presented model has less computation complexity. Besides, we provide an appropriate range of weight coefficients to get a better tradeoff between operational risk and flight delay cost in practical ATFM decision-making.

The remainder of the paper is organized as follows. Section 2 presents the mathematical model of our SATNFO problem, including the definition of operational risk. Section 3 presents computation experiments that evaluate the efficacy of our SATNFO model, that is, the biobjective 0-1 integer programming model. Section 4 gives conclusion and future research.

2. Mathematical Model of SATNFO

In this section, our mathematical model of SATNFO problem is introduced in detail. Firstly, the operational risk is defined via probabilistic risk assessment method to evaluate the effect of capacity uncertainties on ATFM. And then, full mathematical formulation of SATNFO problem is presented.

Before the introduction of mathematical model of SATNFO problem, two concepts should be indicated here: (1) air traffic network: airspace is divided into sectors. Each sector or airport can be represented as a node in the network. If a flight is planned to fly from one node to another, there will be a directed arc between these two nodes; (2) capacity: for each sector or airport, there are a limited number of flights that can fly within it in every time unit. It can be easily influenced by convective weather.

2.1. Definition of Operational Risk. During the modeling, we introduce the concept of operational risk into SATNFO problem to deal with the uncertainties. Generally, operational risk is defined [31] as the risk of a change in value caused by

the fact that actual losses, incurred for inadequate or failed internal processes, people, and systems, or from external events, differ from the expected losses. The operational risk in SATNFO problem can be defined as the extra flight delay cost caused by unpredictable capacity changes and inappropriate flight plan. A robust set of flight plans for all scheduled flights receives less impact resultant from capacity uncertainties and thus with lower operational risk.

In order to quantify the operational risk induced by capacity uncertainties, we adopt the probabilistic risk assessment (PRA) method [32–34]. PRA is a methodology to evaluate risks associated with a complex engineered technological entity. It has been successfully used in agrochemicals analysis [35], nuclear power industry [36], and many other complex engineering fields [37]. In PRA method, the risk is characterized by two elements: the magnitude of all possible adverse consequences (I_i, $i = 1, \ldots, N$, suppose N possible adverse consequences in total) and the likelihood of each consequence (P_i, $i = 1, \ldots, N$). And, the risk is defined as $\sum_{i=1}^{N} I_i \cdot P_i$. As for SATNFO problem, the magnitude of adverse consequence is the extra flight delay cost caused by capacity uncertainties, and the likelihood is determined by the capacity statistical distribution. According to PRA method, the steps to quantify the operational risk in SATNFO problem are as follows.

2.1.1. Capacity Statistical Distribution Modeling. In practical ATFM, the convective weather dynamics often lead to capacity uncertainties for the airports or airspace sectors in a network. A capacity statistical distribution represents possible values of capacity and its distribution and their changes with weather condition along time horizon. The modelling of capacity statistical distribution is built on the basis of the three assumptions and principles.

(1) The airspace is divided into M regions (denoted by $\{S_1, S_2, \ldots, S_M\}$), and each region consists of several sectors and airports. The time period is divided into Q time stages, and each time stage consists of certain smallest time units. When changes of weather spread in the airspace, it can only influence one of the regions during a time stage. Thus, the spreading process can be equally expressed as the capacity of this region decreasing lasting a certain time units.

(2) The extent of weather impact on capacity is divided into N degrees, and the corresponding weather modes are denoted by $\{D_1, D_2, \ldots, D_N\}$.

(3) Let scenario tree [38] describe the capacity statistical distribution. The scenario tree may be various in different time period because of various weather conditions. One node on a scenario tree represents a certain weather mode in a certain region at a certain time stage, according to which we can calculate capacity and its probability of each region (i.e., a capacity distribution of the whole airspace). We call a node as a capacity scenario.

Based on above assumptions, the steps of modelling capacity statistical distribution can be described as follows.

(1) Get the initiative scenario, that is, a weather mode and a region it influenced in time stage 1. Let a scenario be denoted by ($P_{k,l}^{\text{region}}$, $P_{i,j}^{\text{degree}}$), where $P_{k,l}^{\text{region}}$ represents the probability of the weather spreading from S_k to S_l, and $P_{i,j}^{\text{degree}}$ represents the probability of weather mode that transfers from D_i to D_j. At the initial moment, if a weather mode whose degree is D_1 influences the S_1 region, we use ($P_{0,1}^{\text{region}}$, $P_{0,1}^{\text{degree}}$) to represent this scenario.

(2) Analyze the impact of weather along time horizon, including the weather spreading modes (decide which region will be influenced) and the degree of its impact. The impact of weather is uncertain. As time stage transfers from time stage t to $t + 1$, the changes of weather may spread to different regions with a certain probability. And moreover, for a region, the impact degree of weather is also uncertain. Hence, a scenario can generate several subsequent scenarios with a certain decision probability. And, along the time horizon, a scenario tree can be established. Suppose $Q = 3$, $N = 1$, airspace is divided into 4 regions (shown in the left figure of Figure 1), weather spreading from north to south, and a scenario tree can be established as the right figure of Figure 1. In this scenario tree, the initiative scenario, that is, the root node A, is ($P_{0,1}^{\text{region}}$, $P_{0,1}^{\text{degree}}$).

(3) Based on the process of establishing scenario tree, we can get a dynamic probabilistic capacity distribution and decision probability. For example, in Figure 1, the probability of scenario G happening is

$$P(G) = P(A) \cdot P(C \mid A) \cdot P(G \mid C). \tag{1}$$

Scenario G represents a capacity distribution of the whole airspace at time stage 3. According to its information ($P_{1,1}^{\text{region}}$, $P_{1,1}^{\text{degree}}$), we can calculate the capacity for each sector or airport. And the corresponding probability $P(\text{capacity}_{it})$ is $P(G)$. Furthermore, for each sector or airport, we can obtain its probability of its possible capacity at time stage t according to all possible scenarios that may appear in stage t.

2.1.2. Extra Delay Cost Calculation. Extra flight delay cost is the magnitude of adverse consequence caused by capacity uncertainties. Flight delay may be caused if capacity decreases; that is, the extra delay cost resulting from capacity decreases. The number of flights flying in the sector or airport may exceed capacity which is decreased due to the dynamics of convective weather. Hence, some flights should execute airborne holding or ground holding, which leads to extra flight delays. And, the extra flight delay is regarded as the time for a flight to cross the sector with minimum velocity. To simplify the calculation we assume that there is no more flight coming during this waiting period, which means we take no account of the delay propagation in this model.

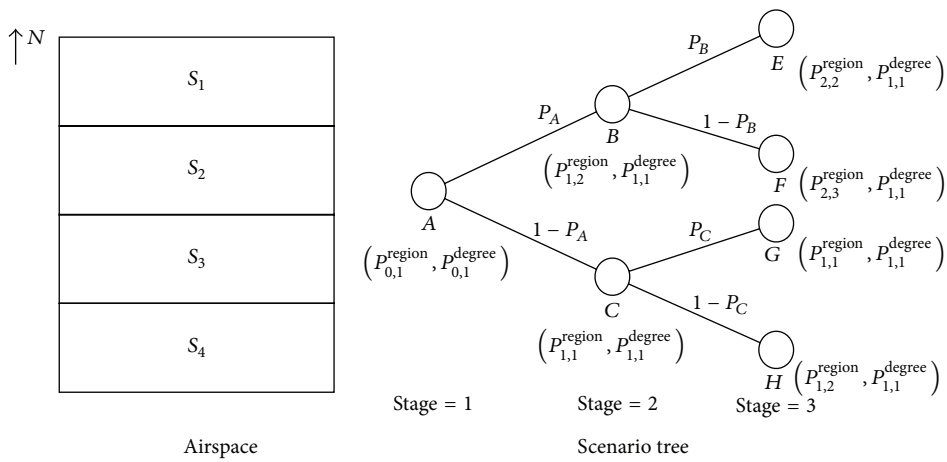


Figure 1: Demonstration of scenario tree.

When the traffic flow in sector i at time t is N_{it}, extra airborne holding time AED_{it} caused by the capacity decreases in sector i at a certain time t can be expressed as below:

$$\text{AED}_{it} = \left(N_{it} - \text{Cs}_{it}\right) \cdot \text{Pass}_i \tag{2}$$

where Cs_{it} is the capacity of sector i at time t and Pass_j is the time that a flight needed to cross sector i with minimum velocity.

Moreover, if the flow exceed the arriving capacity of airport a, it can also cause extra airborne holding:

$$\text{AED}_{at} = N_{at} - \text{Ca}_{at}. \tag{3}$$

Ca_{at} is the arriving capacity of airport a at time t. Similarly, if the flow exceeds the departure capacity of airport d, the extra ground-holding delay cost GAD_{dt} can be expressed as:

$$\text{GED}_{dt} = N_{dt} - \text{Cd}_{dt}. \tag{4}$$

Cd_{dt} is the departure capacity of airport d at time t.

2.1.3. Operational Risk Formulation. According to PRA method, the operational risk of SATNFO problem can be defined as below:

$$\text{Operational Risk} = \sum \text{extra delay} \times p\left(\text{capacity}_{it}\right). \tag{5}$$

It can also be written as

$$\text{Operational Risk} = \text{Risk}_{\text{airborne}} + \text{Risk}_{\text{ground}}. \tag{6}$$

$\text{Risk}_{\text{airborne}}$ is the risk from airborne holding and can be represented as below:

$$\begin{aligned}\text{Risk}_{\text{airborne}} = &\sum_{t,i} \left(N_{it} - \text{Cs}_{it}\right) \cdot \text{Pass}_i \cdot P\left(\text{Cs}_{it}\right) \\ &+ \sum_{t,a} \left(N_{it} - \text{Ca}_{at}\right) \cdot P\left(\text{Ca}_{at}\right),\end{aligned} \tag{7}$$

where $P(\text{Cs}_{it})$ is the probability that the capacity of sector i at time t is Cs_{it}.

$\text{Risk}_{\text{ground}}$ is the risk from ground holding that can be represented:

$$\text{Risk}_{\text{ground}} = \sum_{t,d} \left(N_{dt} - \text{Cd}_{dt}\right) \cdot P\left(\text{Cd}_{dt}\right), \tag{8}$$

where Cd_{dt} is the departure capacity of airport d at time t.

Since the cost of airborne holding is much greater than ground holding, it is necessary to be considered in the definition of the operational risk. Hence, the total risk function should be improved as

$$\begin{aligned}\text{Operational Risk} = &\, w_{\text{airborne}} \cdot \text{Risk}_{\text{airborne}} \\ &+ w_{\text{ground}} \cdot \text{Risk}_{\text{ground}},\end{aligned} \tag{9}$$

where $w_{\text{airborne}} > w_{\text{ground}}$.

2.2. Mathematical Formulation of SATNFO Problem. In our SATNFO model, the minimization of operational risk, which evaluates the effect of capacity uncertainties on ATFM, is introduced as an objective. And, the minimization cost of total flight delays is the other objective.

The air traffic network in our SATNFO model is based on deterministic ATFMO model. To our knowledge, the 0-1 integer programming model [9] proposed by Bertsimas et al. (BLO model) has high computational efficiency and also accuracy to depict the real ATFM actions such as ground holding, airborne holding, speed control, and rerouting. Thus, we build our SATNFO model based on BLO model.

2.2.1. Decision Variable. We use the same definition of decision variable in BLO model:

$$z_{i,t}^{f} = \begin{cases} 1 & \text{if flight } f \text{ arrive sector } i \text{ by time } t \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

Note that we use "by" instead of "at." This definition means once the decision variable is 1 at time unit t, the value of decision variables in the subsequent time units must be 1. It could also be applied to airports. If flight f takes off from airport i by time t, the decision variable is 1 (similarly, if flight f lands at airport j by time t, the decision variable is 1).

2.2.2. Notations. The notations in the SATFNO model which are similar to BLO model are listed as follows:

K: set of sectors,

K^f: set of sectors that could be flown by fight f,

P: set of airports,

P^f: set of airports that could be flown by fight f,

T: set of time units,

F: set of flights,

A_i^f: sets of sector i's ancestor sectors for flight f,

S_i^f: sets of sector i' subsequent sectors for flight f,

$\mathrm{Cid}_p(t)$: ideal planning departure capacity of airport p at time t,

$\mathrm{Cia}_p(t)$: ideal planning arrival capacity of airport p at time t,

$\mathrm{Cis}_i(t)$: ideal planning capacity of sector i at time t,

dep_f: departure airport of flight f,

arr_f: arrival airport of flight f,

Pass_{fij}: minimum number of time units for air crafts to pass through sector i,

$T_f^i[\underline{T_f^i}, \overline{T_f^i}]$: time window for flight f to arriving $\overline{T_f^i}$: the maximum time units for flight f to arrive at sector i (depart or arrive at airport i),

$\underline{T_f^i}$: the minimum time units for flight f to arrive at sector i (depart or arrive at airport i),

$\overline{T_f^i}$: the maximum time units for flight f to arrive at sector i (depart or arrive at airport i).

2.2.3. Objective Functions. In SATNFO model, we formulate a weighted biobjective 0-1 integer programming model. The objective consists of two parts: the operational risk and total flight delay cost. Thus, the objective function is as below:

$$\text{objective function} = (1-\beta)\cdot \text{delay cost} + \beta \cdot \mathrm{Risk}_{\text{total}}. \tag{11}$$

Note that the deterministic model is a particular case of this model when $\beta = 0$.

(1) Delay Cost. We refer to the definition in BLO model and classify the delay into two different parts: airborne holding AH and ground holding GH:

$$\begin{aligned}\mathrm{AH} = &\sum_{f\in F}\sum_{t\in T_f^{\mathrm{arr}_f}}\left(t-\underline{T_f^{\mathrm{arr}_f}}\right)\cdot\left(z^f_{\mathrm{arr}_f,t}-z^f_{\mathrm{arr}_f,t-1}\right)\\ &-\sum_{t\in T_f^{\mathrm{dep}_f}}\left(t-\underline{T_f^{\mathrm{dep}_f}}\right)\cdot\left(z^f_{\mathrm{dep}_f,t}-z^f_{\mathrm{dep}_f,t-1}\right)\end{aligned} \tag{12}$$

$$\mathrm{GH} = \sum_{f\in F}\sum_{t\in T_f^{\mathrm{dep}_f}}\left(t-\underline{T_f^{\mathrm{dep}_f}}\right)\cdot\left(z^f_{\mathrm{dep}_f,t}-z^f_{\mathrm{dep}_f,t-1}\right). \tag{13}$$

We calculate them separately and assign airborne holding a higher coefficient as it costs more resources than the other one. Then, we add up airborne holding and grounding holding to get total delay cost, which is

$$\text{delay cost} = \alpha_{\mathrm{ah}}\cdot \mathrm{AH} + \alpha_{\mathrm{gh}}\cdot \mathrm{GH}. \tag{14}$$

(2) Operational Risk. According to our definition in the previous section, the operational risk is extra delay cost caused by unpredictable capacity changes. We have obtained operational risk through (1)–(9).

2.2.4. Constraints. Consider

$$\sum_{f\in F:\mathrm{dep}_f=p}\left(z^f_{\mathrm{dep}_f,t}-z^f_{\mathrm{dep}_f,t-1}\right)\le \mathrm{Cid}_p(t)\quad \forall p\in P,\ t\in T \tag{15}$$

$$\sum_{f\in F:\mathrm{arr}_f=p}\left(z^f_{\mathrm{arr}_f,t}-z^f_{\mathrm{arr}_f,t-1}\right)\le \mathrm{Cia}_p(t)\quad \forall p\in P,\ t\in T \tag{16}$$

$$\sum_{i\in K_f:f\in F}\left(\max\left\{0, z^f_{i,t}-\sum_{i'\in S_i^f} z^f_{i',t}\right\}\right)\le \mathrm{Cis}_i(t)\quad \forall i\in K,\quad t\in T \tag{17}$$

$$z^f_{i,t-1}-z^f_{i,t}\le 0\quad \forall t\in T,\ f\in F,\ i\in K_i \tag{18}$$

$$z^f_{i,t-1}\le \sum_{i'\in A_i^f} z^f_{i',t-\mathrm{pass}_{fij}}\quad \forall t\in T,\ f\in F,\ i\in K_i \tag{19}$$

$$z^f_{i,\overline{T_f^i}}\le \sum_{i'\in S_i^f} z^f_{i',\overline{T_f^{i'}}}\quad \forall f\in F,\ i\in K^f \tag{20}$$

$$\sum_{i'\in S_i^f} z_{i',\overline{T_f^{i'}}}\le 1\quad \forall f\in F,\ i\in K^f \tag{21}$$

$$z^f_{i,t}\le 0\quad \forall f\in F,\ i\in K^f\bigcap P^f,\ t\in T: t<\underline{T_f^i} \tag{22}$$

$$z^f_{i,t}-z^f_{i,t-1}=0\quad \forall f\in F,\ i\in K^f\bigcap P^f,\ t\in T: t<\overline{T}_f^i \tag{23}$$

$$z^f_{i,t}\in 0,1\quad \forall f\in F,\ i\in K^f\bigcap P^f,\ t\in T_f^i. \tag{24}$$

Constraints (15) and constraints (16) give the limitation of departure and arrival capacity of airport p at time t. They ensure that the number of flights which take off from airport p will not exceed the departure capacity of airport p at time t and that the number of flights which land on airport p will not exceed the arrival capacity of airport p at time t. Constraints (17) stipulate that the number of flights that arrive at sector k will not exceed the capacity of sector k at time t. Closely related to the definition of decision "by," constraints (18) guarantee the time continuity of every flight. Constraints (19) state that if flight f arrives at sector i at time t, it must have been arrived at its ancestor node

before time $t - \text{pass}_{fij}$. Constraints (20) state that if flight arrives at sector i it will arrive at its subsequent sectors. Constraints (21) present that, for every flight that has arrived at sector k, it can only arrive at one of its subsequent sectors. Constraint (22) and Constraint (23) present the time-window limitation for every flight and the sectors a flight will pass by. Constraints (22) guarantee that the flight f cannot arrive at sector i before the minimum time we planned. Constraints (23) ensure that flight f should arrive at sector i before the maximum time we planned or the flight will never arrive at sector i. Finally, constraints (24) state that decision variables are Boolean values which make our model a 0-1 programming model.

3. Experiments

In this section, two sets of experiments are carried out to evaluate the efficacy of our SATNFO model, that is, the biobjective 0-1 integer programming model. In the first set, we analyze the effect of parameter setting, that is, the weight coefficient of risk (β) in our SATNFO model, and thereby provide an appropriate range of weight coefficient to get a better tradeoff between operational risk and flight delay cost in practical ATFM decision-making. In the second set, we show the generality of our mathematical model by solving a SATNFO problem under various scenarios. After that, the computational complexity of our SATNFO model is analyzed.

As for the experimental data, we extract a real air traffic network from the northern airspace of China, which consists of 25 sectors. A time horizon of two hours and 12 minutes is considered, and it is divided into 22 unites, 6 minutes each. Two sets of flight plans, consisting of 55 flights and 220 flights, respectively, are considered to be optimized for different purposes, which will be explained later. Capacity instances are generated based on simulated weather data, and the corresponding generation method will be shown in Section 3.1. All experiments are performed using optimization programming language with CPLEX 12.5.1, on a PC with Inter core i5-3470 processor, 3.20 GHz, and 12 GB RAM Microsoft Windows 7 OS.

In the following of this section, the details of capacity instances generation, two sets of experiments, and computational complexity of our SATNFO model are presented in order.

3.1. Capacity Instances Generation. As mentioned previously, a scenario tree represents possible values of capacity and its distribution and their changes with weather condition along time horizon. In this subsection, we generate 6 different capacity instances, that is, 6 different scenario trees based on simulated weather data. The steps of capacity instances are described as follows.

(1) Partition of Airspace. The airspace consisting of 25 sectors is divided into 4 regions. Figure 2 is a schematic diagram of the airspace partition, thick lines are boundary of each region, and the value in sector denotes sector number. Weather has the same impact on sectors which are in a same region. Each region contains a set of sectors as follows:

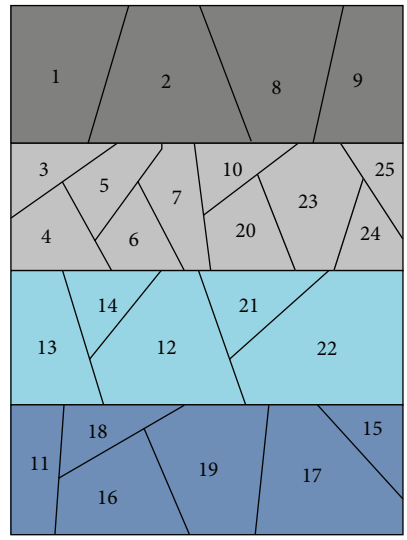


Figure 2: A schematic diagram of dividing airspace into 4 regions.

S_1: 8, 9, 1, 2;

S_2: 3, 4, 5, 6, 7, 10, 20, 23, 24, 25;

S_3: 13, 14, 12, 21, 22;

S_4: 11, 16, 18, 19, 17, 15.

(2) Definition of Impact of Weather on Capacity. We suppose that the degree of impact of weather on capacity is divided into three classes, represented by ratio between current capacity under changes of weather and the ideal capacity which is a predicted and deterministic value. Three ratio values are denoted by $\{A, B, C\}$. The values of $\{A, B, C\}$ are influenced by weather conditions. In our simulated weather data, there are two sets of them: DEG1 62.5%, 50%, and 37.5% and DEG2 50%, 37.5%, and 25%.

(3) Definition of Weather Spreading Modes. In our simulated weather data, there are three weather spreading modes for describing changes of weather.

MODE 1: North-to-South Spreading. Initially, changes of weather influence S_1 with a certain degree m ($m \in A, B, C$), and its probability is P_m. In the following period, extreme weather leaves S_1, so the capacity in S_1 recovers. At the same time, changes of weather influence S_2 and the degree of capacity decrease transfers from m to degree n ($n \in A, B, C$). The transfer probability can be expressed as P_{mn}. According to this rule, the changes of weather transfer to S_3, S_4 sequentially.

MODE 2: Expanding from Center. In this mode, changes of weather influence S_2 first. In the next time period, changes of weather may spread to the north S_1 or spread to the south S_3. Then, if the change of

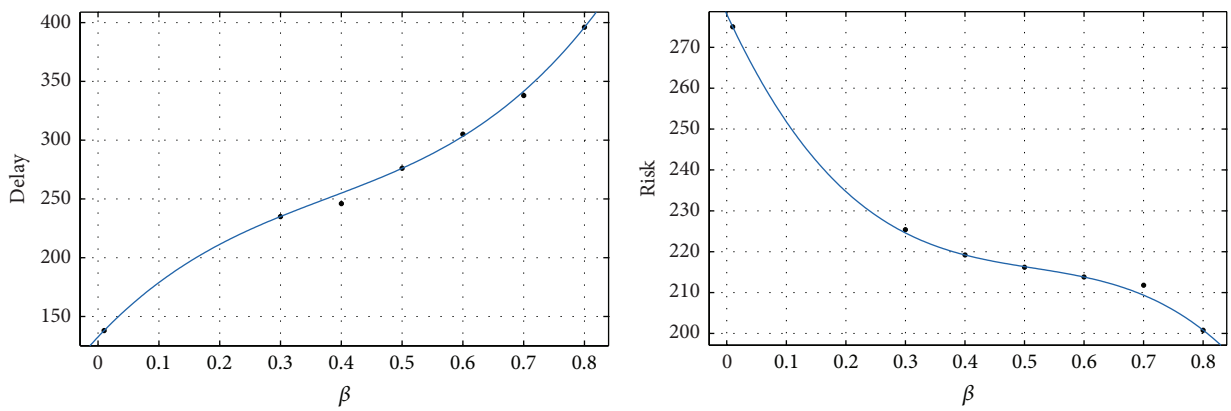


Figure 3: Demonstration of delay cost and risk with β.

weather is in S_3, it would spread to S_4. Otherwise, the changes of weather would leave this airspace and all capacities recover straight after. The definition of transfer probability is similar to mode 1.

MODE 3: South-to-North Spreading. In this mode, changes of weather influence S_4 first and then spread to S_3, S_2, and S_1 sequentially. The definition of transfer probability is similar to mode 1 and mode 2.

(4) Capacity Instances. Having gotten above three spreading modes and two sets of impact degrees of weather on capacity, we can get 6 capacity instances, denoted by the following:

Instance 11 (MODE1, DEG1), Instance 12 (MODE1, DEG2),

Instance 21 (MODE2, DEG1), Instance 22 (MODE2, DEG2),

Instance 31 (MODE3, DEG1), Instance 32 (MODE3, DEG2).

According to each instance, we can generate its corresponding scenario tree based on the given weather data.

3.2. Risk Weight Coefficient Analysis. In this experiment, we analyze the effect of parameter setting, that is, the weight coefficient of risk β, in our SATNFO model. Hence, the relationship between β and total delay and the relationship between β and operational risk are analyzed concretely. For this experiment, a set of flight plans which consists of 55 flights is optimized under capacity instance "Instance 21."

Table 1 shows corresponding delay and risk with various β. Figure 3 shows the relationship between β and delay (left) and the relationship between β and risk (right).

From Figure 3, we find that operational risk decreases and total delay increases with the increase of β. It means that when we add risk as one objective of SATNFO problem, the optimized flight plans are relatively conservative, that is, allowing more delays for safety operation. Further, the optimized flight plans become more conservative with the increase of risk weight coefficient. Thus, decision-makers can choose their appropriate risk coefficient to optimize flight plans for all scheduled flights using our SATNFO model. According to their experience and the real condition, they can obtain an acceptable tradeoff between optimization and robustness.

Table 1: β versus delay and risk.

β	Delay	Risk
0.01	138	275
0.3	235	225.4
0.4	246	219.2
0.5	276	216.2
0.6	305	213.8
0.7	338	211.8
0.8	396	200.8

More important, the analysis of Figure 3 can provide information that helps decision-makers to give an appropriate value of β and get a better tradeoff. In the left figure, we can find that the delay cost increases almost linearly with the increase of β. In the right figure, the pace of risk decreasing with β is fast at first and then slows down. When β is less than 0.4, the risk will decrease obviously while the increase of delay cost is relatively slow; when β is more than 0.4, the risk decreases become slow while the increase of delay cost is dramatic. Therefore, adjusting β below 0.4 will help us to gain a small risk with a relatively lower delay cost.

3.3. Generality Analysis of Our SATNFO Model. In this experiment, we use our 6 generated capacity instances to test the generality of our SATNFO model. Moreover, we have increased the number of flights to 220. The value of β is set to 0.4.

In each instance, capacity distribution changes according to its weather spreading mode and impact degree of weather on capacity. Table 2 illustrates the risk in deterministic model and our SATNFO model. The first column represents different capacity instances. The second and third columns list corresponding delay and operational risk obtained by our SATNFO model for each capacity instance. The last column is the operational risk if we do not consider changes of

Table 2: Delay, risk in SATNFO model, and risk in deterministic model.

Instance	Delay-uncertain	Risk-uncertain	Risk-determine
Instance 11	2260	170.3	512.0
Instance 12	2151	63.6	296.0
Instance 21	2415	387.2	581.7
Instance 22	2318	240.8	338.7
Instance 31	2454	340.8	500.8
Instance 32	2365	252.3	285.4

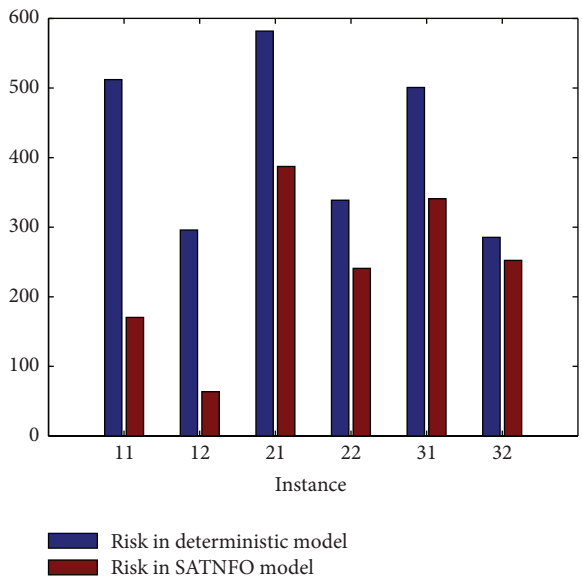


Figure 4: Demonstration of risk in deterministic model and our SATNFO model.

weather (the optimization result of the deterministic model). Figure 4 shows the comparison between the operational risk under deterministic model and our SATNFO model. In all 6 different capacity instances, our SATNFO model can obtain flight plans with less risk.

Note that our model can efficiently reduce operational risk under different capacity instances. Moreover, it can demonstrate the superiority of our SATNFO model on aspect of generality, since we can solve SATNFO problem efficiently with increased number of flights (from 55 flights to 220 flights), no matter what kind the capacity uncertainty is.

Furthermore, we find out that the results we obtain reflect the real situation. This also illustrates the rationality of our proposed model. In our capacity instances, south part of this area refers to Shanghai and its surrounding areas; the center part refers to Beijing (the capital of China) and its surrounding areas; and north part of this area refers to Inner Mongolia. The airline operation is far busier in Beijing and Shanghai than that in Inner Mongolia. So it is reasonable that capacity distribution changes which start from center and south (Instance 21, Instance 22, Instance 31, and Instance 32) will lead to more risk than those which start from north (Instance 11, Instance 12). Our experiment exactly shows this difference. From this perspective, our SATNFO model is consistent with the reality, which illustrates its rationality.

Table 3: Numbers of constraints of SATNFO model and SPNAC model.

Model	Flight	Time stage	Sector	Degradation degree	Constraints
SPNAC	55	4	25	3	494971
SATNFO	55	4	25	3	42374
SATNFO	220	4	25	3	165530

3.4. Computational Complexity Analysis. If the numbers of sectors, flights, and time units are denoted by S, F, and T, respectively, then the dimension of decision variable is $|T| \cdot |F| \cdot |S|$. The upper bound of constraints is $2 \times |F| \times |T| \times |F| \times |S| + 2 \times |F| \times |T|$.

Table 3 shows the numbers of constraints in our proposed SATNFO model and in stochastic-programming with nonanticipative constraints (denoted by SPNAC model hereafter) [26]. From the table, we found that to solve a SATNFO problem with the same scale, the constraints in our proposed model are far more less than in the SPNAC model.

4. Conclusion and Future Research

In this paper, we successfully provide a novel weighted biobjective 0-1 integer programming model to deal with the uncertainty in SATNFO problem. In this model, two objectives are the minimization of the cost of total flight delays and the minimization of the operational risk. In order to evaluate the effect of capacity uncertainties on ATFM, the operational risk is specifically defined via probabilistic risk assessment method. By assigning different weight coefficients on risk and delay cost, some efficiency is compromised to make our flight schedule more robust. This model allows decision-makers to adjust risk weight coefficients, so that they can make tradeoff between risk and delay costs. Moreover, because we introduce the operational risk function into objective instead of using capacity constraints under uncertainties, we avoid the dramatic increase of the number of constraints and thus avoid suffering the computational intractability.

Through computation experiment, we find out an appropriate range of risk weight coefficient that enables decision-makers to gain a small risk with a relatively lower delay cost. Successful tests under different capacity distribution instances confirm the generality of our SATNFO model. Moreover, result shows the number of constraints is far less than nonanticipative constraints in stochastic programming, which illustrates the high computational quality of this model.

Our model could be further improved in the future. First, various kinds of uncertainty sources could be considered besides capacity changes. For example, the flight demand is also an uncertain element in air traffic flow management problem. Taking various kinds of uncertainties can make our model closer to the real-world application. Second, the choice between risk and delay cost is totally decided by setting risk

weight, which is an artificial decision in this model. Although it is flexible, to some extent, the unreliability is inevitable. It is possible to introduce a feedback mechanism in future studies that the risk weight could be automatically adjusted according to the risk value during the optimization. In this case, we can avoid the unreasonable decision and make our model more reliable.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Science Foundation for Young Scientists of China (Grant nos. 61401011 and 61201314), the National ATC Scientific Research Project (Grant no. GKG201404004), and the National Key Technologies R&D Program of China (Grant no. 2012BAG04B01).

References

[1] Civil Aviation Administration of China, "Civil Aviation Industry Statistical Report 2013," 2014-06-23.

[2] G. Andreatta, A. R. Odoni, and O. Richetta, "Models for the ground holding problem," in *Large Scale Computation and Information Processing in Air Traffic Control*, pp. 125–168, Springer, Berlin, Germany, 1993.

[3] P. B. Vranas, D. Bertsimas, and A. R. Odoni, "Dynamic ground-holding policies for a network of airports," *Transportation Science*, vol. 28, no. 4, pp. 275–291, 1994.

[4] L. Brunetta, G. Guastalla, and L. Navazio, "Solving the multi-airport ground holding problem," *Annals of Operations Research*, vol. 81, pp. 271–288, 1998.

[5] O. Richetta, "Optimal algorithms and a remarkably efficient heuristic for the ground-holding problem in air traffic control," *Operations Research*, vol. 43, no. 5, pp. 758–770, 1995.

[6] D. Bertsimas and S. S. Patterson, "The air traffic flow management problem with enroute capacities," *Operations Research*, vol. 46, no. 3, pp. 406–422, 1998.

[7] P. Dell'Olmo and G. Lull, "A dynamic programming approach for the airport capacity allocation problem," *IMA Journal of Management Mathematics*, vol. 14, no. 3, pp. 235–249, 2003.

[8] G. Lulli and A. Odoni, "The European air traffic flow management problem," *Transportation Science*, vol. 41, no. 4, pp. 431–443, 2007.

[9] D. Bertsimas, G. Lulli, and A. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Operations Research*, vol. 59, no. 1, pp. 211–227, 2011.

[10] D. Bertsimas and S. S. Patterson, "The traffic flow management rerouting problem in air traffic control: a dynamic network flow approach," *Transportation Science*, vol. 34, no. 3, pp. 239–255, 2000.

[11] A. Agustín, A. Alonso-Ayuso, L. F. Escudero, and C. Pizarro, "On air traffic flow management with rerouting. Part I: deterministic case," *European Journal of Operational Research*, vol. 219, no. 1, pp. 156–166, 2012.

[12] A. M. Churchill, D. J. Lovell, and M. O. Ball, "Evaluating a new formulation for large-scale traffic flow management," in *Proceedings of the 8th USA/Europe Air Traffic Seminar (ATM '09)*, vol. 310, 2009.

[13] K.-Q. Cai, J. Zhang, W.-B. Du, and X.-B. Cao, "Analysis of the Chinese air route network as a complex network," *Chinese Physics B*, vol. 21, no. 2, Article ID 028903, 2012.

[14] W.-B. Du, Z.-X. Wu, and K.-Q. Cai, "Effective usage of shortest paths promotes transportation efficiency on scale-free networks," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 17, pp. 3505–3512, 2013.

[15] C. Liu, W.-B. Du, and W.-X. Wang, "Particle swarm optimization with scale-free interactions," *PLoS ONE*, vol. 9, no. 5, Article ID e97822, 2014.

[16] M. Steiner and J. Krozel, "Translation of ensemble-based weather forecasts into probabilistic air traffic capacity impact," in *Proceedings of the IEEE/AIAA 28th Digital Avionics Systems Conference (DASC '09)*, pp. 2.D.61–2.D.67, IEEE, October 2009.

[17] A. Prékopa, *Stochastic Programming*, Springer, 1995.

[18] D. P. Heyman and M. J. Sobel, *Stochastic Models in Operations Research: Stochastic Optimization*, Courier Dover Publications, Mineola, NY, USA, 2003.

[19] S. Verdu and H. V. Poor, "On minimax robustness: a general approach and applications," *IEEE Transactions on Information Theory*, vol. 30, no. 2, pp. 328–340, 1984.

[20] H.-G. Beyer and B. Sendhoff, "Robust optimization—a comprehensive survey," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 33-34, pp. 3190–3218, 2007.

[21] O. Richetta and A. R. Odoni, "Solving optimally the static ground-holding policy problem in air traffic control," *Transportation Science*, vol. 27, no. 3, pp. 228–238, 1993.

[22] O. Richetta and A. R. Odoni, "Dynamic solution to the ground-holding problem in air traffic control," *Transportation Research Part A: Policy and Practice*, vol. 28, no. 3, pp. 167–185, 1994.

[23] K. Frauendorfer, "Barycentric scenario trees in convex multistage stochastic programming," *Mathematical Programming*, vol. 75, no. 2, pp. 277–293, 1996.

[24] N. Gröwe-Kuska, H. Heitsch, and W. Römisch, "Scenario reduction and scenario tree construction for power management problems," in *Proceedings of the IEEE PowerTech Conference*, vol. 3, pp. 152–158, Bologna, Italy, June 2003.

[25] M. O. Ball, R. Hoffman, A. R. Odoni, and R. Rifkin, "A stochastic integer program with dual network structure and its application to the ground-holding problem," *Operations Research*, vol. 51, no. 1, pp. 167–172, 2003.

[26] A. Agustín, A. Alonso-Ayuso, L. F. Escudero, and C. Pizarro, "On air traffic flow management with rerouting. Part II: stochastic case," *European Journal of Operational Research*, vol. 219, no. 1, pp. 167–177, 2012.

[27] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*, Princeton University Press, 2009.

[28] D. Bertsimas, D. B. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Review*, vol. 53, no. 3, pp. 464–501, 2011.

[29] S. Gupta and D. J. Bertsimas, "Multistage air traffic flow management under capacity uncertainty: a robust and adaptive optimization approach," in *Proceedings of the 51st AGIFORS Annual Proceedings: Annual Symposium and Study Group Meeting*, pp. 692–721, Antalya, Turkey, October 2011.

[30] D. Bertsimas and V. Goyal, "On the power of robust solutions in two-stage stochastic and adaptive optimization problems," *Mathematics of Operations Research*, vol. 35, no. 2, pp. 284–305, 2010.

[31] I. I. Basel, *Revised International Capital Framework*, Basel Committee on Banking Supervision, Basel, Switzerland, 2004.

[32] G. W. Parry, "The characterization of uncertainty in probabilistic risk assessments of complex systems," *Reliability Engineering & System Safety*, vol. 54, no. 2-3, pp. 119–126, 1996.

[33] G. E. Apostolakis, "How useful is quantitative risk assessment?" *Risk Analysis*, vol. 24, no. 3, pp. 515–520, 2004.

[34] H. A. Abbass, S. Alam, and A. Bender, "MEBRA: multiobjective evolutionary-based risk assessment," *IEEE Computational Intelligence Magazine*, vol. 4, no. 3, pp. 29–36, 2009.

[35] K. Solomon, J. Giesy, and P. Jones, "Probabilistic risk assessment of agrochemicals in the environment," *Crop Protection*, vol. 19, no. 8–10, pp. 649–655, 2000.

[36] R. R. Fullwood and R. E. Hall, *Probabilistic Risk Assessment in The Nuclear Power Industry*, Pergamon Press, New York, NY, USA, 1988.

[37] E. J. Henley and H. Kumamoto, *Probabilistic Risk Assessment and Management for Engineers and Scientists*, IEE Press, New York, NY, USA, 2002.

[38] J. van Ryzin, "Quantitative risk assessment," *Journal of Occupational Medicine*, vol. 22, no. 5, pp. 321–326, 1980.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 247301, 10 pages
http://dx.doi.org/10.1155/2015/247301

Research Article

MRAC Control with Prior Model Knowledge for Asymmetric Damaged Aircraft

Xieyu Xu,[1] Lingyu Yang,[2] and Jing Zhang[2]

[1] *CNIGC Institution of Navigation and Control, Chedaogou No. 10, Haidian District, Beijing 100191, China*
[2] *Department of Automation Science and Electrical Engineering, Beihang University, Haidian District, Beijing 100191, China*

Correspondence should be addressed to Xieyu Xu; xudavid501@163.com

Received 16 November 2014; Accepted 8 December 2014

Academic Editor: Kemao Peng

This paper develops a novel state-tracking multivariable model reference adaptive control (MRAC) technique utilizing prior knowledge of plant models to recover control performance of an asymmetric structural damaged aircraft. A modification of linear model representation is given. With prior knowledge on structural damage, a polytope linear parameter varying (LPV) model is derived to cover all concerned damage conditions. An MRAC method is developed for the polytope model, of which the stability and asymptotic error convergence are theoretically proved. The proposed technique reduces the number of parameters to be adapted and thus decreases computational cost and requires less input information. The method is validated by simulations on NASA generic transport model (GTM) with damage.

1. Introduction

It has been shown that the vehicle damage to airframe and engines had led to quite a few aircraft accidents and fatalities in recent years [1]. Asymmetric aircraft structural damage results in abrupt deterioration on flight performance and handling quality, which impairs safety. Researches to recover aircraft stability and handling qualities, like fault-tolerant flight control, reconfigurable control method, and so forth, have already become a hot spot in related academic fields.

Lots of studies have been conducted to recover control performance under aircraft failures or damages. NASA has launched several aircraft safety projects like IFCS [2] and AvSP [3], which innovate flight control algorithms and experiment on aircraft testbeds. Various flight control laws like the adaptive control designs [4–9], linear quadratic regulator (LQR) designs [6, 10, 11], and robust linear designs [12] were proposed and evaluated by flights. The pilot Cooper-Harper ratings showed that the proposed methods could retain the handling qualities under certain failure cases [4, 13].

Generally, the adaptive control algorithms are quite popular in this field. Reference [14] used an adaptive artificial neural network (ANN) control method to recover the handling qualities of an F-18 model. The method uses a pretrained ANN to provide a model inversion block with aerodynamics and handling characteristics and another online-adjusted ANN to compensate modeling error and failure conditions. Reference [15] extended the result of [14]. It changes the pretrained ANN to an online learning network, which adjusts the inversion block adaptively, resulting in a hybrid direct and indirect adaptive ANN control scheme. Reference [16] designed an angular rate controller using adaptive single hidden layer ANN method for an unmanned airborne vehicle. With a classic linear guidance law, several successful automated landings were made in flight experiments with 25% left wing loss.

Except for ANN adaptive laws, multivariable MRAC with various structures were also studied in this field. Reference [9] combined a direct MRAC method with a parameter estimator using gradient algorithm, obtaining a combined MRAC structure. References [6, 17] proposed a derivative-free MRAC using delayed weight update law. Reference [7] utilized the L_1 adaptive law [18] for a flight controller design. Reference [19] analyzed the plant characteristics by LDS decomposition of the high frequency gain matrix (HFGM) of the damaged aircraft. Realizing the signs of the leading

principle minors of the HFGM does not change before and after damage. Along with several other results and assumptions, an MRAC law with output feedback structure is proposed. The method is validated through simulations on a high-fidelity GTM model with left wing tip damage. A similar design can be found in [20], which uses a state feedback structure with adaptive gains instead of the original output feedback structure.

The methods mentioned above do not need to take the characteristics of damage conditions into account. As long as the assumptions hold, these methods can be applied to various damage cases and allow for relatively large parameter changes. However, the range of admissible parameter changes can be so large, in a way that is far enough to cover all damage cases. If proper descriptions can be used to model the concerned damage cases, it is possible to design a controller with parameters adjusted in a smaller region, reducing computation workload and improving performance.

Based on the analysis above, this paper models the structural damaged aircraft with polytope LPV models and model reference controllers (MRCs) are given, which explicitly integrates plant characteristics. The parameters updated by adaptive laws are the polytope interpolation coefficients, instead of control gains in traditional MRAC designs. The number of the adaptive parameters is therefore reduced. The paper is organized as follows. Section 1 introduces the background and design philosophy of the proposed method. Section 2 models and analyzes a damaged aircraft, namely, GTM. A polytope linear model is introduced to describe the concerned damage cases. Section 3 designs the control laws and adaptive laws, as well as Lyapunov stability and error convergence proof. Section 4 validates the design by simulations on a 6DOF nonlinear GTM model with left outboard wing tip damage. Section 5 concludes the whole paper.

2. Modeling of an Asymmetric Damaged Aircraft with Prior Knowledge

2.1. Nonlinear 6DOF Modeling. This paper studies the GTM from IRAC project in AvSP program carried out by NASA. Due to the limitation of modeling data, this paper only discusses the damage case of left outboard wing tip loss. With data and methods from [21], the characteristics of aerodynamics, mass, inertia, center of gravity (CG) movement, and so forth, can be modeled. With methods from [22], the rigid body dynamic equations are deduced. It should be noted that the sensors remain unmoved before and after damage, so that the point which the dynamic equations describe remains unchanged. The dynamic equation can be written as follows:

$$\begin{bmatrix} F \\ M \end{bmatrix} = \begin{bmatrix} m\mathbb{I} & -m\Delta^{ss} \\ m\Delta^{ss} & I \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} m\omega^{ss} & -m\omega^{ss}\Delta^{ss} \\ m\omega^{ss}\Delta^{ss} & \omega^{ss}I - mV^{ss}\Delta^{ss} \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix}. \tag{1}$$

F and M refer to the total force and moment relevant to original CG, respectively. m, I, and Δ are the total mass, inertia in original CG, and the offset of CG reference to its origin location. $V = [u, v, w]^T$, $\omega = [p, q, r]^T$ are the line speed and angular speed described in the origin body frame of the aircraft. Superscript ^{ss} is the skew-symmetric matrix form of a vector, $(x^{ss})y = x \times y$.

The line speed of a specific point generally changes with its location, only except that it moves parallel to the rotation axis of the rigid body. The calculation of the line acceleration has to take the centripetal and the Coriolis force into account, as long as the point is not located on the rotation axis. In short, due to the CG movement by the structural damage, the calculation of line speed becomes quite complicated, coupling with angular speed and angular acceleration. However, if the body frame is selected parallel to the original one, the direction of each axis does not change, and the values of the angular speed are the same wherever the frame is. Thus, the angular speed can be modeled in the body frame at the new CG without changing its value, while the line speed has to be modeled at the same point. Rewrite (1) as follows:

$$\begin{bmatrix} m\mathbb{I} & -m\Delta^{ss} \\ 0 & \overline{I} \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -m\omega^{ss} & m\omega^{ss}\Delta^{ss} \\ 0 & -\omega^{ss}\overline{I} \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} + \begin{bmatrix} F \\ \overline{M} \end{bmatrix}. \tag{2}$$

Since the value of angular speed remains unchanged, the symbol ω is used as before. $\overline{I}$ and $\overline{M}$ are the inertia and moment represented in the new body frame located at CG after damage. The kinetic equations are the same as those of normal aircrafts. $[\phi, \theta, \psi]$ is the Euler angle. $[x_g, y_g, z_g]$ is the position vector relative to the ground:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix},$$
$$\begin{bmatrix} \dot{x}_g \\ \dot{y}_g \\ \dot{z}_g \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix}. \tag{3}$$

With the sensors unmoved, the definitions of airspeed V, angle of attack α, and sideslip β are the same as before. The equations are as follows:

$$\alpha = \arctan \frac{w}{u}, \quad \beta = \arcsin \frac{v}{V}, \quad V = \|V\|_2. \tag{4}$$

2.2. Polytope Linear Model. Generally, the nonlinear equations of an aircraft can be described as

$$\dot{x}(t) = f(x(t), u(t)). \tag{5}$$

x, u represent the state and input vectors, respectively. An operation point, or trim point, should be selected before linearization. It should be noted that the actual damage cases are unknown, which renders the trim point unavailable. A general solution is to select a common operation point for all damage cases, for example, the normal point (x_0, u_0) [20]. Because the normal point generally is not the trim point for damaged cases, an extra term f_i is added to the state-space equations, as follows [23]:

$$\begin{aligned} \Delta\dot{x}(t) &= \sum A_i \chi_i(t) \Delta x(t) + \sum B_i \chi_i(t) \Delta u(t) \\ &\quad + \sum f_i \chi_i(t) \\ \chi_i(t) &= \begin{cases} 1 & \text{if } (A(t), B(t), f(t)) = (A_i, B_i, f_i) \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \tag{6}$$

where $A_i \in \mathbb{R}^{n\times n}$, $B_i \in \mathbb{R}^{n\times m}$ are unknown constant matrices, representing the state transition matrices and control matrices of different patterns and degrees of structural damage, respectively. $f_i \in \mathbb{R}^n$ is the unknown constant disturbance. Various control methods can be designed with (6) [20, 23]. However, A_i, B_i, f_i are unknown constants with weak constraints like the signs of leading principle minors of high frequency gain matrix that do not change, which could lead to relatively large parameter variations. The actual damage cases may not cover all these variations. This paper believes that if proper definitions of different damage cases are introduced, the parameter variations can be limited, which facilitates the controller designs.

Based on the above analysis, the model of damaged aircraft is rewritten in a polytope form:

$$\begin{aligned} \Delta\dot{x} &= A(p)\Delta x + B(p)\Delta u + f(p), \\ [A(p), B(p)] &\in \Omega = \mathrm{Co}\{[A_1^*, B_1^*], \ldots, [A_l^*, B_l^*]\}, \end{aligned} \tag{7}$$

where $[A_i^*, B_i^*]$, $i = 1, \ldots, l$ are known constant matrices, the vertices of polytope Ω. There exist nonnegative coefficients $\alpha_i(p)$; thus, $[A(p), B(p)]$ can be calculated by

$$\begin{aligned} [A(p), B(p)] &= \sum_{i=1}^{l} \alpha_i(p) [A_i^*, B_i^*], \\ \sum_{i=1}^{l} \alpha_i(p) &= 1, \quad 0 \le \alpha_i(p) \le 1. \end{aligned} \tag{8}$$

Parameter p represents different damage patterns and degrees. For any specific damage case, it is assumed that a certain p^* can describe the model of the damaged aircraft. $f(p)$ is the extra unknown constant term, for the aircraft is untrimmed under damage. $f(p)$ also captures other effects from structural damages and is not modelled in the polytope form. Finally the model is written as

$$\Delta\dot{x} = \sum_{i=1}^{l} \alpha_i(p) A_i^* \Delta x + \sum_{i=1}^{l} \alpha_i(p) B_i^* \Delta u + f(p). \tag{9}$$

Compared to (6), the only unknown variables in (9) are $\alpha = [\alpha_1, \ldots, \alpha_l]^T \in \mathbb{R}^l$ and $f(p) \in \mathbb{R}^n$. Different damage cases simply differs in weighting coefficient α and constant $f(p)$, thus vastly reduces the number of unknown variables.

2.3. Vertex Calculation by High Order SVD Method. As stated before, the less the number of vertices is, the less complicated the controller is. It is crucial to balance between the accuracy and the complexity of the polytope aircraft model with damage of various patterns and degrees. The high order SVD (HOSVD) method is a decomposition method for high order arrays or data tensors. By representing the polytope LTI models into a tensor-product (TP) form, the HOSVD method can be used to reduce the number of vertices [24].

First, a parameter grid is generated on the concerned damage cases. For the left outboard wing tip damage case, a grid of k-element vector is used. Each node of the grid represents a different wing tip loss ratio to the semispan. The GTM model with damage is linearized at each node of the grid, written in the system matrix form

$$S_i = \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix}. \tag{10}$$

For every specific case in the concerned damage cases, the model $S(p)$ can be written as the interpolation of the linearized models. That is,

$$S(p) = \sum \alpha_i(p) S_i, \tag{11}$$

wherein function $\alpha_i(p)$ is the interpolation coefficient function. The interpolation calculation can be rewritten in a compact TP form. Stack all the linearized models to make the model tensor $\mathcal{S}$ and all the coefficients to make the parameter tensor $\mathcal{U}$. Thus,

$$S(p) = \sum \alpha_i(p) S_i = \mathcal{S} \otimes \mathcal{U}(p). \tag{12}$$

To accurately describe the original model, a relatively intense grid has to be generated. The algorithm based on HOSVD method from [24] is adopted in this paper to simplify the grid. By omission of small singular values, the number of models and corresponding coefficients can be reduced, without losing accuracy to the original ones. With this algorithm, the preliminary TP model can be approximated to

$$S(p) = \mathcal{S} \otimes \mathcal{U}(p) \approx \mathcal{S}^* \otimes \mathcal{U}^*(p). \tag{13}$$

New models S_i^* and interpolation coefficients $\alpha_i^*(p)$, $i = 1, \ldots, l$, can be extracted from the system tensor $\mathcal{S}^*$ and the

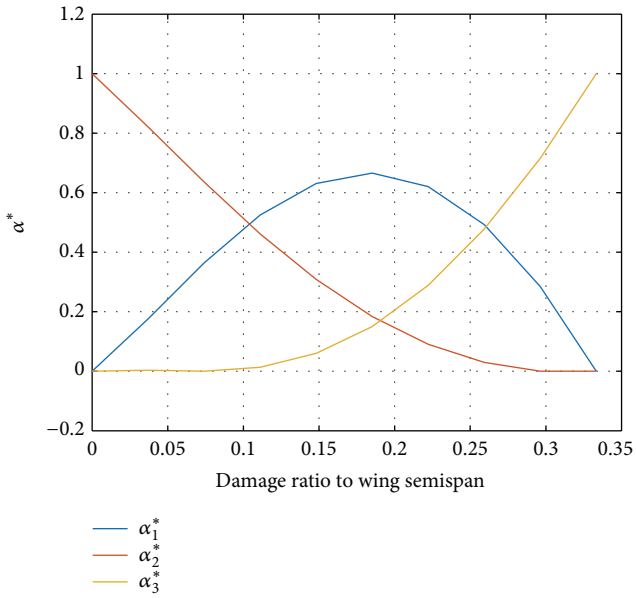


FIGURE 1: The curves of the interpolation coefficients to wing tip loss ratios.

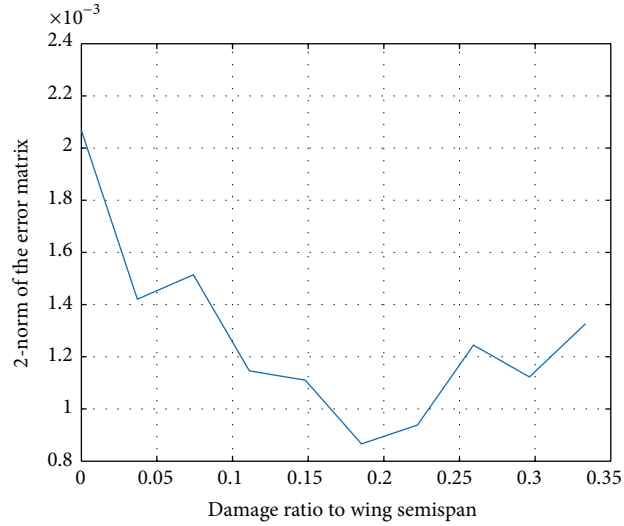


FIGURE 2: The accuracy of processed model.

coefficient tensor $\mathscr{U}^*$. The algorithm automatically processes the coefficients to build up a polytope model, constrained by $\sum \alpha_i^* = 1, \alpha_i^* > 0$.

The numerical results are shown as follows. The left outboard wing tip loss of GTM ranges from 0% to 33% semispan. A grid size of 10 is selected. State-space models are obtained by linearizing nonlinear GTM models of 1 normal case and 9 damage cases equally spaced along the loss ratios to wing span. After rewriting the models into TP form, the singular values calculated by HOSVD method are given as follows in sorted sequence:

$$\sigma = \left[166.4366 \quad 0.1842 \quad 0.0035 \quad 0.0034 \quad \cdots\right]. \tag{14}$$

It can be seen that the first 2 singular values are relatively large. By omitting the rest of singular values, the algorithm simplifies the 10 models to 3 vertex models. Thus, any model from the original 10 models, S_i, for example, can be approximated by interpolation of the obtained 3 models as follows:

$$S_i \approx \widehat{S}_i = \sum_{j=1}^{3} \alpha_j^*(i) S_j^*. \tag{15}$$

$\alpha_j^*(i)$ is the interpolation coefficient. The output models are in accordance with the properties of polytope models. The curves of the interpolation coefficients to wing tip loss ratios are shown in Figure 1.

The accuracy of the processing procedure can be measured by the 2-norm of the matrix error $\widetilde{S}_i = S_i - \widehat{S}_i$, as shown in Figure 2.

It can be shown that the error norm is relatively small, which concludes that the resulting polytope model is close to the original ones. The subsequent designs are based on the obtained polytope model.

The control algorithm proposed in this paper is not of LMI designs in [24]. The interpolation parameters satisfy $\sum_j \alpha_j(i) = 1$ only, which means extrapolation is admissible in the algorithm. The parameters are treated independent of each other, without restrictions in Figure 1. That is, the 3 parameters can vary freely on the plane $\sum_j \alpha_j(i) = 1$ in the 3D space, while Figure 1 restricts the parameters to a curve. As a result, it is expected that the polytope model can describe more damage cases by relaxing the restrictions on the parameters.

2.4. GTM Model Simplification. Assume the polytope model of GTM is written as

$$\dot{x}_{\mathrm{GTM}} = A(p) x_{\mathrm{GTM}} + B(p) u + f_{\mathrm{GTM}}(p), \tag{16}$$

wherein

$$\begin{aligned} x_{\mathrm{GTM}}(t) &= \left[q, p, r, V, \alpha, \theta, \beta, \phi\right]^T, \\ u(t) &= \left[\delta_e, \delta_a, \delta_r\right]^T. \end{aligned} \tag{17}$$

The angular rates are to be controlled. Rewrite the state-space equations to treat angular rates as states, while the other signals related to angular rates are treated as measurable disturbances. Dividing the original states into two vectors yields

$$\begin{aligned} x_{\mathrm{GTM}}^T &= \left[x^T, z^T\right], \\ x &\triangleq \left[q, p, r\right]^T, \\ z &\triangleq \left[V, \alpha, \theta, \beta, \phi\right]^T. \end{aligned} \tag{18}$$

Reformulate the state-space equation as follows:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} &= \begin{bmatrix} A_{xx}(p) & A_{xz}(p) \\ A_{zx}(p) & A_{zz}(p) \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} \\ &\quad + \begin{bmatrix} B_x(p) \\ B_z(p) \end{bmatrix} u + \begin{bmatrix} f_x(p) \\ f_z(p) \end{bmatrix}. \end{aligned} \tag{19}$$

Finally, the polytope model with angular rates as states is shown below:

$$\dot{x} = A_{xx}(p)x + A_{xz}(p)z + B_x(p)u + f_x(p). \tag{20}$$

The model is explained as follows. The states are defined as the angular rates. The state transition matrix is A_{xx}. The input of the model is separated into 2 parts, the first part of which is the remaining state z with input matrix A_{xz}. The second part is the original input u with input matrix B_x, which describes the effects of control surfaces. Because the damaged aircraft is not trimmed, a constant disturbance $f_x(p)$ is added to the equation. For the convenience of the following discussions, (20) is rewritten as

$$\dot{x} = A(p)x + H(p)z + B(p)u + f(p). \tag{21}$$

For the specific case of GTM, the linear model of the undamaged case is shown below:

$$\begin{aligned}\dot{x} = &\underbrace{\begin{bmatrix} -3.0362 & 0 & 0 \\ 0 & -4.6576 & 0.5481 \\ 0 & -0.3932 & -1.0176 \end{bmatrix}}_{A} x \\ &+ \underbrace{\begin{bmatrix} 0 & -29.0039 & 0 & 0 & 0 \\ 0 & 0 & 0 & -75.2928 & 0 \\ 0 & 0 & 0 & 24.8998 & 0 \end{bmatrix}}_{H} z \\ &+ \underbrace{\begin{bmatrix} -43.2662 & 0 & 0 \\ 0 & 39.5686 & 11.1409 \\ 0 & 3.2818 & -28.537 \end{bmatrix}}_{B} u \\ &+ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{f_0}.\end{aligned} \tag{22}$$

The linear model of 33% semispan left outboard wing tip loss is written as follows:

$$\begin{aligned}\dot{x} = &\underbrace{\begin{bmatrix} -3.004 & -0.1346 & 0.0062 \\ -0.5033 & -3.2188 & 0.4941 \\ -0.0508 & -0.2771 & -1.0288 \end{bmatrix}}_{A} x \\ &+ \underbrace{\begin{bmatrix} 2.2889 & -20.815 & 0 & -0.7196 & 0 \\ -34.4495 & -158.3492 & 0 & -75.2928 & 0 \\ -3.7582 & -21.7562 & 0 & 25.8689 & 0 \end{bmatrix}}_{H} z \\ &+ \underbrace{\begin{bmatrix} -43.3124 & 2.6631 & -0.0442 \\ 0.6455 & 22.0466 & 11.3155 \\ 0.0211 & 1.8451 & -28.7231 \end{bmatrix}}_{B} u \\ &+ \underbrace{\begin{bmatrix} 53.0318 \\ -797.3925 \\ -86.6825 \end{bmatrix}}_{f_0}.\end{aligned} \tag{23}$$

It can be seen from the above results that the attitude angles, namely, θ and ϕ, have no effect on the angular rates, which can be omitted in the model. The omission is reasonable because the moment of aircraft can be generated only by aerodynamic moment and thrust, and the attitude angles have nothing to do with it. Thus, the state z is simplified to $z = [V, \alpha, \beta]^T$.

Although the input matrix B changes with damage, the characteristics of B, like the domination and signs of the diagonal elements, remain unchanged. The numeric changes of nondiagonal elements relative to diagonal ones can also be ignored. Based on the above conclusions, the B matrix is treated as a constant before and after wing tip damage. In short,

$$B(p) \approx B. \tag{24}$$

For the convenience of design, the B matrix adopted in polytope model is the undamaged one.

3. The MRAC Design

Model reference scheme with state-tracking method is adopted to recover the control performance after the aircraft structural damage. The reference model is designed as follows:

$$\dot{x}_m = A_m x_m + B_m r. \tag{25}$$

Hurwitz state transition matrix A_m and input matrix B_m reflect the desired dynamic performance. It is known that given symmetric positive real matrix Q, there exists a symmetric positive real matrix P, satisfying

$$A_m^T P + P A_m + Q = 0. \tag{26}$$

Considering the angular motion of an aircraft can be directly altered by control surfaces, input matrix B will have full rank to be invertible. The model can be rewritten as

$$\begin{aligned}\dot{x} = &A_m x + B \\ &\times \left(u + B^{-1}\left(\sum \alpha_i A_i^* - A_m\right)x + B^{-1}H(p)z + B^{-1}f\right).\end{aligned} \tag{27}$$

Define

$$\begin{gathered}\mu_i \triangleq \left[B^{-1}(A_i^* - A_m) \quad B^{-1}H_i^*\right]\begin{bmatrix} x \\ z \end{bmatrix}, \\ \overline{f} \triangleq B^{-1} f.\end{gathered} \tag{28}$$

Thus,

$$\dot{x} = A_m x + B\left(u + \left[\mu_1 \ \cdots \ \mu_l\right]\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_l \end{bmatrix} + \overline{f}\right). \tag{29}$$

Rewrite the polytope constraint $\sum_{i=1}^{l}\alpha_i = 1$ to $\alpha_1 = 1 - \sum_{i=2}^{l}\alpha_i$ and substitute

$$\begin{aligned}\dot{x} &= A_m x + B \\ &\quad\times\left(u + \mu_1 + \begin{bmatrix}\mu_2 - \mu_1 & \cdots & \mu_l - \mu_1\end{bmatrix}\begin{bmatrix}\alpha_2\\ \vdots \\ \alpha_l\end{bmatrix} + \overline{f}\right) \\ &= A_m x + B\left(u + \mu_1 + \overline{M}\overline{\alpha} + \overline{f}_0\right).\end{aligned} \tag{30}$$

It is assumed that every state of the aircraft can be measured. Matrix $\overline{M}$ is known, since signal μ_i can be measured and calculated with known signals. The parameter vector $\overline{\alpha}$ concatenated from α_2 to α_l and the constant offset $\overline{f}$ are unknown in actual damage cases. The control law can be designed as follows to alter the plant dynamic characteristics to reference model precisely, if every piece of parameter is known:

$$u = B^{-1}B_m r - \mu_1 - \overline{M}\overline{\alpha} - \overline{f}. \tag{31}$$

However, $\overline{\alpha}$ and $\overline{f}$ are unknown after damage. With certain-equivalence design principle [25], the control law is designed as

$$u = B^{-1}B_m r - \mu_1 - \overline{M}\widehat{\overline{\alpha}} - \widehat{\overline{f}}. \tag{32}$$

$\widehat{\overline{\alpha}}$ and $\widehat{\overline{f}}$ are estimates of $\overline{\alpha}$ and $\overline{f}$ by adaptive laws, respectively. Define the parameter errors as $\widetilde{\overline{\alpha}} \triangleq \widehat{\overline{\alpha}} - \overline{\alpha}$, $\widetilde{\overline{f}} \triangleq \widehat{\overline{f}} - \overline{f}$ and substitute the control law into (30), resulting in

$$\dot{x} = A_m x + B_m r - B\overline{M}\widetilde{\overline{\alpha}} - B\widetilde{\overline{f}}. \tag{33}$$

The tracking error is defined as $e \triangleq x - x_m$, the dynamics of which can be obtained as

$$\dot{e} = A_m e - B\overline{M}\widetilde{\overline{\alpha}} - B\widetilde{\overline{f}}. \tag{34}$$

The adaptive laws are designed using Lyapunov direct method. The Lyapunov function V is chosen to be

$$V = e^T P e + \widetilde{\overline{\alpha}}^T \Gamma_\alpha^{-1}\widetilde{\overline{\alpha}} + \widetilde{\overline{f}}^T \Gamma_f^{-1}\widetilde{\overline{f}}. \tag{35}$$

Γ_α, Γ_f are positive symmetric matrices. The derivative of Lyapunov function V along the dynamics of system is

$$\begin{aligned}\dot{V} &= \dot{e}^T P e + e^T P \dot{e} + 2\dot{\widetilde{\overline{\alpha}}}^T \Gamma_\alpha^{-1}\widetilde{\overline{\alpha}} + 2\dot{\widetilde{\overline{f}}}^T \Gamma_f^{-1}\widetilde{\overline{f}} \\ &= -e^T Q e - 2e^T PB\overline{M}\widetilde{\overline{\alpha}} - 2e^T PB\widetilde{\overline{f}} \\ &\quad + 2\dot{\widetilde{\overline{\alpha}}}^T \Gamma_\alpha^{-1}\widetilde{\overline{\alpha}} + 2\dot{\widetilde{\overline{f}}}^T \Gamma_f^{-1}\widetilde{\overline{f}}.\end{aligned} \tag{36}$$

The adaptive laws are selected as

$$\begin{aligned}\dot{\widetilde{\overline{\alpha}}}^T &= e^T PB\overline{M}\Gamma_\alpha, \\ \dot{\widetilde{\overline{f}}}^T &= e^T PB\Gamma_f.\end{aligned} \tag{37}$$

Thus, the derivate $\dot{V}$ becomes

$$\dot{V} = -e^T Q e. \tag{38}$$

The design process of the adaptive law in this paper is quite similar to the classical ones using the Lyapunov method. Properties like stability and asymptotic error convergence can also be proved using same procedures. All states in the system are bounded due to the positivity of V and seminegativity of $\dot{V}$, so that the system is stable in the Lyapunov sense. The boundedness of states guarantees that $\ddot{V}$ is bounded, which means $\dot{V}$ is uniformly continuous. Using Barbalat's lemma, it can be concluded that $\dot{V}$ converges to zero asymptotically; that is, the tracking error $e \to 0$. In summary, the properties of the proposed method are stated as follows.

Theorem 1. *The MRAC theme with control law (32) updated by adaptive law (37), when applied to the plant (30), guarantees the closed-loop signal boundedness and asymptotic state-tracking $e = x - x_m \to 0$.*

Remark 2. Thanks to the popularity of the standard Lyapunov design procedure, lots of modifications can be applied to the control law, such as robust modifications [26], predictor-based or reference model modifications to improve transient performance [27–30], and adaptive law modifications [31].

Remark 3. The proposed method updates the interpolation coefficients of the feedback gains. The gains are designed beforehand for the vertices of the polytope model, which integrates prior information of different cases of damage into the control law. With prior knowledge of various damage cases, the performance of the design is expected to be improved. The HOSVD and model simplification process reduce the number of vertices and the amount of parameters. Less stimulation is required for the input signal [25].

Remark 4. The linearization of the original 6DOF nonlinear model and the vertex simplification using HOSVD method result in an approximated damaged aircraft model. There always exists modeling error between the real model of various damage cases and the developed polytope model. Similar to common adaptive algorithms, the update law has to be modified to ensure closed-loop stability [25]. Modifications like the projection operator have to be applied to the adaptive law in real cases.

Remark 5. It should be noted that the maximum deflection of control surfaces is finite in real aircrafts, which poses a constraint on u in (32) and is not taken into consideration in MRAC design process. It can be seen in (23) that some values in matrix H and vector f_0 are quite large, which might saturate u and cause the aircraft to be uncontrollable in some cases. Actually, if all control surfaces are limited within $\pm 30^\circ$, which is even larger than that in [21], it is impossible to maintain a steady cruise in the operation point of (23) in 33% left wing tip loss case. The steady cruise means that no rotation is allowed; thus, the angular speed $[p, q, r]$ keeps zero, and the operation point means the other states in the

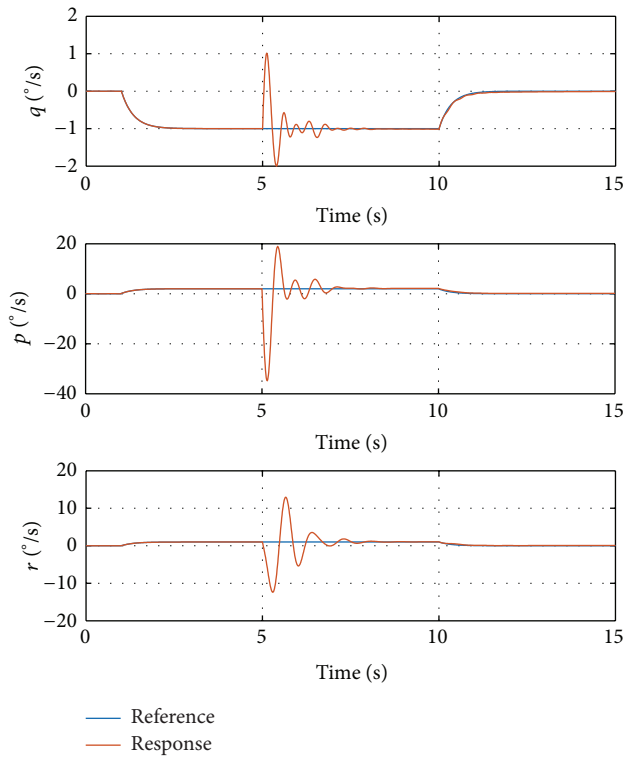


Figure 3: Commands and responses in simulation with step input signal.

linearization model (23), namely, $[V, \alpha, \beta]$, are equal to zero, resulting in

$$\underbrace{\begin{bmatrix} -43.3124 & 2.6631 & -0.0442 \\ 0.6455 & 22.0466 & 11.3155 \\ 0.0211 & 1.8451 & -28.7231 \end{bmatrix}}_{B} u + \underbrace{\begin{bmatrix} 53.0318 \\ -797.3925 \\ -86.6825 \end{bmatrix}}_{f_0} = 0, \tag{39}$$

$$u = \begin{bmatrix} -3.4640 & -36.4141 & 0.6762 \end{bmatrix}^T.$$

It can be seen that the second element in u exceeds -30°, which means the aileron is saturated, concluding that it is impossible to balance the constant offset f_0 solely by input u.

However, it is feasible to maintain a straight cruise under some attack angle α and sideslip β, and it is possible to saturate the control surfaces under improper α and β. Due to the left wing tip loss, if the damaged aircraft is flown with relative large lift force, the left roll aerodynamic moment will be too large to be compensated by ailerons. Detailed discussions of flight performance under various damage cases may be found in [32, 33] and are beyond the scope of this paper.

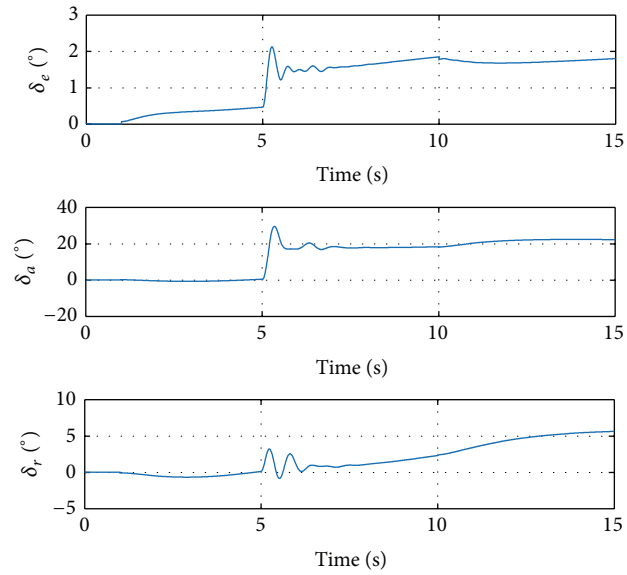


Figure 4: Control surface deflections in simulation with step input signal.

4. Simulation Study

In this section, the developed adaptive law under the obtained linear polytope model is validated by simulations on a NASA GTM nonlinear model with damage. GTM is a 5.5% dynamically scaled twin-turbine powered aircraft model, designed and manufactured in the NASA AvSP program, dedicated to flight testing of research control laws in adverse flight conditions [34].

The damage case with loss of 20% semispan of the left wing tip is selected. Simulations are conducted on the nonlinear 6DOF GTM model, with maximum deflection of the control surfaces limited within ±30 degrees. The aircraft is flying normally at the beginning, injected with wing tip damage at 5 s during simulation. The simulation results with minor step commands are shown in Figure 3.

The step input starts at 1 second with value $[-1, 1, 1]$. Figure 3 shows that the response of the undamaged aircraft tracks the reference command from 0 to 5 seconds, which benefits from the proper selection of the initial values of α and f_0. The damage is injected at 5 seconds, resulting in some transient oscillations during 5–8 seconds. The magnitudes of the oscillations are relatively small and admissible. After the oscillations, the angular rate responses track the reference signals again. Another step input is injected at 10 seconds. The response signal tracks the reference consistently, which proves the performance of the proposed method. The deflections of control surfaces are shown in Figure 4.

Figure 4 shows that all the control surfaces are not saturated; thus, the angular rates are controllable. More explanations can be found in the responses of state z in (21), shown in Figure 5.

It can be seen that the responses of the other states are relatively small, especially for the angle of attack α. As in Remark 5, (23) indicates that positive α with offset f_0 can saturate the

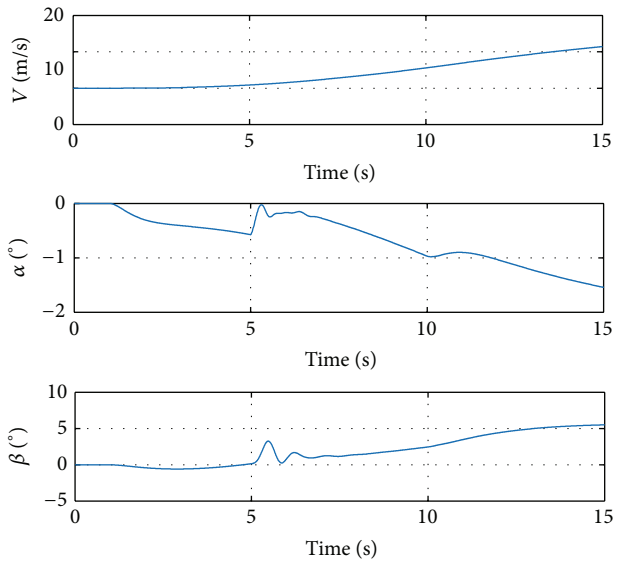


Figure 5: Response of the other states in simulation with step input signal.

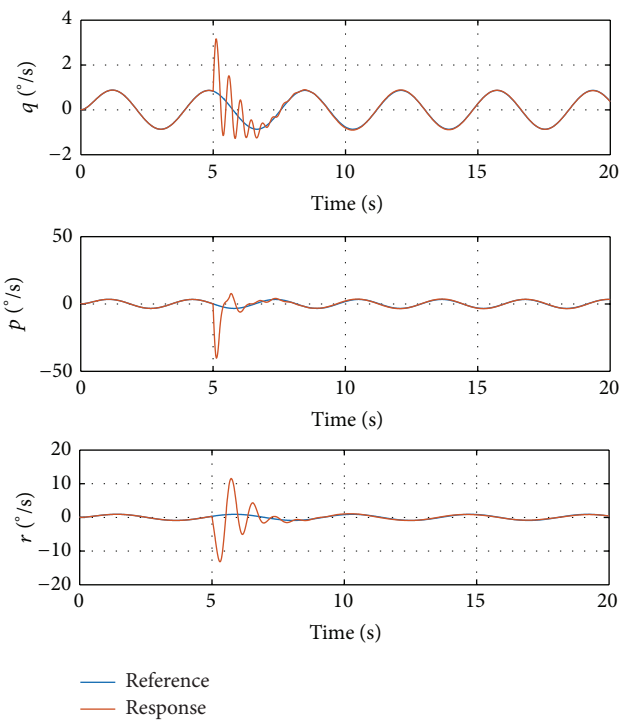


Figure 6: Commands and responses in simulation with sinusoidal input signal.

ailerons easily, meaning that inappropriate command might cause loss of stability. If the input is selected as $[q, p, r] = [1, 2, 0]$, simulation shows that positive α will be generated, leaving the ailerons saturated, the response divergent, and the angular rates uncontrollable. Thus, the pitch angular rate command in this case is selected as $-1^{\circ}/\mathrm{s}$ to obtain the negative α response, which helps reduce the deflections of control surfaces and results in a controllable aircraft.

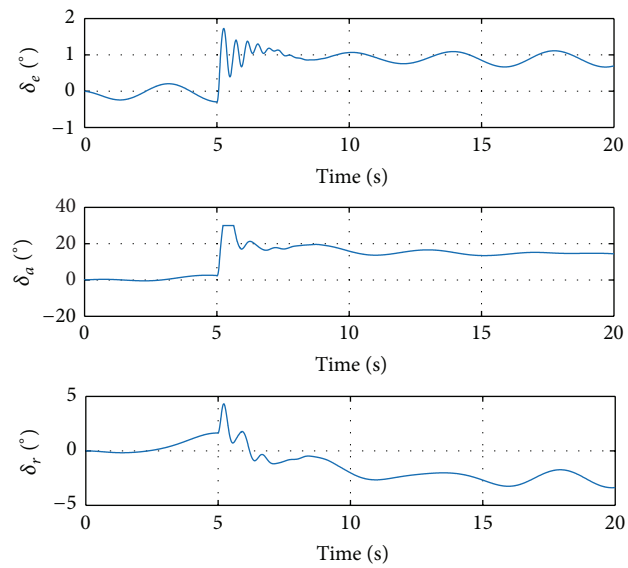


Figure 7: Control surface deflections in simulation with sinusoidal input signal.

Further demonstration of tracking performance can be shown in sinusoidal command signals, as in Figure 6.

The sinusoidal inputs are arbitrarily chosen as $q_c = \sin\sqrt{3}t$, $p_c = 4\sin 2t$, and $r_c = \sin\sqrt{2}t$. The damage is injected as 20% left wing tip loss at 5 seconds. As in the step input case, the angular rates track the reference signals before damage, which validates the initial control parameters and the control structure. The transient oscillation lasts from 5 to 8 seconds in the simulation, the magnitude of which is relatively small and admissible. The transient ends at about 8 seconds and the response signals track the reference again. It can be concluded that the proposed MRAC method can restore the handling qualities after the structural damage.

The deflections of control surfaces are shown in Figure 7.

It can be seen that the ailerons saturate shortly in the oscillations. However, the ailerons come back to about 18 degrees after the transient, and the saturation does not affect the overall performance of the design. The responses of z are shown in Figure 8.

It can be seen that the airspeed and air flow angles vary in the same sinusoidal pattern as that in the angular rates. The overall magnitude of state z is relatively small, which helps reduce the deflections of control surfaces, making it possible to track the reference signals.

In summary, the simulations show that the proposed MRAC method guarantees error convergence and stability in both step input and sinusoidal input. It can be concluded that the proposed method is validated by simulation.

5. Conclusions

In this paper, an MRAC method with prior model knowledge for asymmetric damaged aircraft has been developed. The method is designed for a series of linearized GTM models with the same operation points but with different left

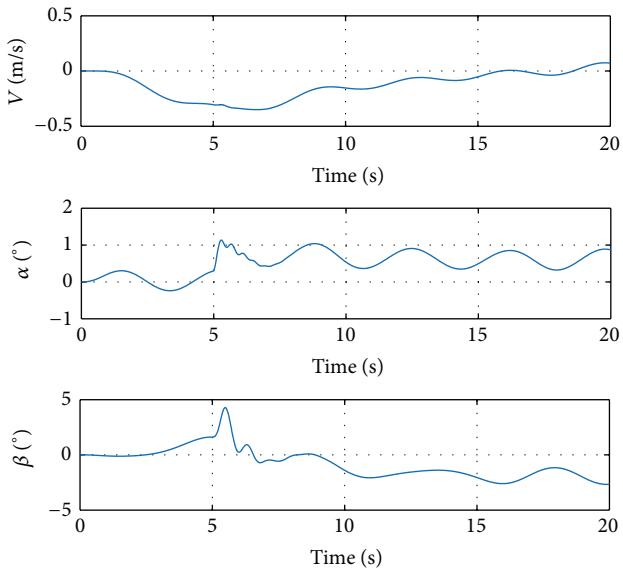


Figure 8: Response of the other states in simulation with sinusoidal input signal.

outboard wing tip damage degrees. A polytope model for damaged aircraft with small amount of vertices is obtained first. By representing the vast models of various damage cases in a compact tensor-product form, the number of vertices is reduced by omission of smaller singular values using HOSVD algorithm. An extra constant offset term to the derivatives of states is introduced since the actual trimming values of damaged aircraft are unknown. By interpolating the vertex controllers, the model reference control can be achieved. An adaptive law is deduced to update the interpolation coefficients and the trimming values, finalizing the controller design. The method is theoretically proved with closed-loop stability and asymptotic tracking error convergence. Simulations on a nonlinear GTM with damage validate the design.

Some problems remain unsolved in this paper. It was assumed that the B matrix is constant before and after damage, but actually it varies a bit. The result is a small tracking error in the simulation of sinusoidal inputs. Future research will address the various assumptions above. The performance of the design is expected to be improved.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This research was funded by National Natural Science Foundation of China (Grant no. 61273099).

References

[1] C. M. Belcastro and J. V. Foster, "Aircraft loss-of-control accident analysis," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Toronto, Canada, August 2010.

[2] NASA, *NASA Dryden Fact Sheet—Intelligent Flight Control System*, 2006.

[3] NASA, *Aviation Safety Program Fact Sheet*, 2011.

[4] J. T. Bosworth and P. S. Williams-Hayes, "Stabilator failure adaptation from flight tests of NF-15B intelligent flight control system," *Journal of Aerospace Computing, Information and Communication*, vol. 6, no. 3, pp. 187–205, 2009.

[5] J. Kaneshige and J. Burken, "Enhancements to a neural adaptive flight control system for a modified F-15 aircraft," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, Hawaii, USA, August 2008.

[6] T. Yucelen and A. J. Calise, "Derivative-free model reference adaptive control," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 4, pp. 933–950, 2011.

[7] I. M. Gregory, C. Cao, E. Xargay, N. Hovakimyan, and X. Zou, "L1 adaptive control design for NASA AirSTAR flight test vehicle," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 2009.

[8] I. Gregory, R. Gadient, and E. Lavretsky, "Flight test of composite model reference adaptive control (CMRAC) augmentation using NASA AirSTAR infrastructure," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2011.

[9] E. Lavretsky, "Combined/Composite model reference adaptive control," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2009.

[10] L. G. Crespo, M. Matsutani, and A. M. Annaswamy, "Design of a model reference adaptive controller for an unmanned air vehicle," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Toronto, Canada, August 2010.

[11] R. Gadient, J. Levin, and E. Lavretsky, "Comparison of model reference adaptive controller designs applied to the NASA Generic Transport Model," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Toronto, Canada, August 2010.

[12] A. Dorobantu, A. Murch, and G. Balas, "H-infinity robust control design for the NASA AirSTAR flight test vehicle," in *Proceedings of the 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, pp. 1–14, Nashville, Tenn, USA, January 2012.

[13] K. Cunningham, D. E. Cox, D. G. Murri, and S. E. Riddick, "A piloted evaluation of damage accommodating flight control using a remotely piloted vehicle," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Portland, Ore, USA, August 2011.

[14] B. S. Kim and A. J. Calise, "Nonlinear flight control using neural networks," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 1, pp. 26–33, 1997.

[15] N. Nguyen, K. Krishnakumar, J. Kaneshige, and P. Nespeca, "Flight dynamics and hybrid adaptive control of damaged aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 3, pp. 751–764, 2008.

[16] G. Chowdhary, E. N. Johnson, R. Chandramohan, M. S. Kimbrell, and A. Calise, "Guidance and control of airplanes under actuator failures and severe structural damage," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 1093–1104, 2013.

[17] T. Yucelen and A. J. Calise, "Robustness of a derivative-free adaptive control law," *Journal of Guidance, Control, and Dynamics*, vol. 37, pp. 1583–1594, 2014.

[18] C. Cao and N. Hovakimyan, "Design and analysis of a novel L1 adaptive control architecture with guaranteed transient

performance," *IEEE Transactions on Automatic Control*, vol. 53, pp. 586–591, 2008.

[19] Y. Liu, G. Tao, and S. M. Joshi, "Modeling and model reference adaptive control of aircraft with asymmetric damage," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 5, pp. 1500–1517, 2010.

[20] J. Guo, G. Tao, and Y. Liu, "Multivariable adaptive control of NASA generic transport aircraft model with damage," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 5, pp. 1495–1506, 2011.

[21] J. A. Ouellette, *Flight dynamics and maneuver loads on a commercial aircraft with discrete source damage [M.S. thesis]*, Virginia Polytechnic Institute and State University, Blacksburg, Va, USA, 2010.

[22] B. J. Bacon and I. M. Gregory, "General equations of motion for a damaged asymmetric aircraft," in *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, pp. 63–75, August 2007.

[23] S. Qian, T. Gang, and G. Jiaxing, "Multivariable state feedback for output tracking MRAC for piecewise linear systems with relaxed design conditions," in *Proceedings of the American Control Conference (ACC '14)*, pp. 703–708, Portland, Ore, USA, June 2014.

[24] P. Baranyi, "TP model transformation as a way to LMI-based controller design," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 2, pp. 387–400, 2004.

[25] P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Prentice Hall, 1995.

[26] N. Nguyen, K. Krishnakumar, and J. Boskovic, "An optimal control modification to model-reference adaptive control for fast adaptation," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008.

[27] V. Stepanyan and K. Krishnakumar, "Adaptive control with reference model modification," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 4, pp. 1370–1374, 2012.

[28] E. Lavretsky, R. Gadient, and I. M. Gregory, "Predictor-based model reference adaptive control," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 4, pp. 1195–1201, 2010.

[29] T. E. Gibson, A. Annaswamy, and E. Lavretsky, "On adaptive control with closed-loop reference models: transients, oscillations, and peaking," *IEEE Access*, vol. 1, pp. 703–717, 2013.

[30] T. Yucelen, G. de la Torre, and E. N. Johnson, "Improving transient performance of adaptive control architectures using frequency-limited system error dynamics," *International Journal of Control*, vol. 87, pp. 2383–2397, 2014.

[31] T. Yucelen and W. M. Haddad, "Low-frequency learning and fast adaptation in model reference adaptive control," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 1080–1085, 2013.

[32] D. Asadi, M. Sabzehparvar, E. M. Atkins, and H. A. Talebi, "Damaged airplane trajectory planning based on flight envelope and motion primitives," *Journal of Aircraft*, vol. 51, pp. 1740–1757, 2014.

[33] G. Yi, J. Zhong, E. M. Atkins, and C. Wang, "Trim state discovery with physical constraints," *Journal of Aircraft*, pp. 1–17, 2014.

[34] A. Murch, "A flight control system architecture for the NASA AirSTAR flight test infrastructure," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 396382, 13 pages
http://dx.doi.org/10.1155/2014/396382

Research Article

Coordinated Control of Slip Ratio for Wheeled Mobile Robots Climbing Loose Sloped Terrain

Zhengcai Li and Yang Wang

School of Mechanical Instrumental Engineering, Harbin Institute of Technology, Harbin 150001, China

Correspondence should be addressed to Zhengcai Li; 12b908067@hit.edu.cn

Received 25 July 2014; Accepted 11 August 2014; Published 3 September 2014

Academic Editor: Kemao Peng

A challenging problem faced by wheeled mobile robots (WMRs) such as planetary rovers traversing loose sloped terrain is the inevitable longitudinal slip suffered by the wheels, which often leads to their deviation from the predetermined trajectory, reduced drive efficiency, and possible failures. This study investigates this problem using terramechanics analysis of the wheel-soil interaction. First, a slope-based wheel-soil interaction terramechanics model is built, and an online slip coordinated algorithm is designed based on the goal of optimal drive efficiency. An equation of state is established using the coordinated slip as the desired input and the actual slip as a state variable. To improve the robustness and adaptability of the control system, an adaptive neural network is designed. Analytical results and those of a simulation using Vortex demonstrate the significantly improved mobile performance of the WMR using the proposed control system.

1. Introduction

Wheeled mobile robots (WMRs) used in outdoor applications, such as for planetary exploration, often have to navigate loose sloped terrain. WMRs traversing on loose terrain will inevitably encounter the problem of slip between the rigid wheels and the loose soil [1]. An increase in this slip beyond a certain degree will cause the WMR to deviate from the predetermined trajectory and the power consumption to increase and possibly cause failures. Such adverse incidents have been reported for WMRs traversing on loose terrain. For example, the 2005 Opportunity Rover relapsed into loose sand dunes of Purgatory and took five weeks to escape [2]. Similarly, the 2009 Spirit Rover was jammed into the sandy soil of the Home Plate plateau of Mars, owing to which its "mobile" life ended and it became a fixed research platform [3]. The conventional control mechanism of a WMR is designed mainly for navigation in indoor environments, typically by assuming the wheel-terrain interaction as being rigid, that is, without considering the slip or considering it an external disturbance [4]. Indubitably, this assumption of rigidity is reasonable when the velocity and acceleration of WMRs are small or the surface of the traversing path is hard (both of which correspond to a large coefficient of friction) [5]. However, a WMR's wheel-soil mechanics is substantially different in outdoor applications such as planetary exploration [6, 7], and, hence, the conventional rigidity-based control mechanism could cause loss of control of the WMR [8]. Thus, the control of WMRs on loose terrain is faced with unique challenges.

With this background, many researchers have focused on wheel-soil mechanics and considered it to be an important factor for achieving proper control of WMRs traversing on loose terrain and therefore conducted extensive research in this direction. For example, Iagnemma and Dubowsky verified through numerical analysis and tests that the rigidity-based conventional motion planning and control algorithm can cause severe slip and sinkage problems in WMRs [1]; they then modeled wheel-soil mechanics based on terramechanics [9] and also established the multiple-physical-based control mechanism for the navigation of WMRs on flat, loose terrain [9]. Further, Yoshida et al. extended the multiple-physics-based approaches and proposed a traction control method for reducing the slip ratio to a small value [10]. The method was verified through tests on dry sand, thus confirming that the wheel-slip-based control mechanism is effective in preventing severe sinkage and energy waste [10]. Furthermore,

in order to avoid incoordination among the wheels, Baumgartner et al. proposed a velocity synchronization algorithm [11]; they verified through experiments using field integrated design and operation rovers that the proposed method can reduce the required power and wheel slippage [11]. Ding et al. proposed an online soil parameter estimation method using the linear least-squares method [12] and a modified, simplified model for wheel-terrain interaction [13], and they compared the simplified model with the original one by numerical analysis [14] to proof the precision of the modified, simplified model. Now, with the development of wheel-soil interaction terramechanics for WMRs, the influence of the wheel slip ratio on the energy consumption and traction efficiency of WMRs is well understood [15]. Moreover, it has become feasible to develop more efficient control algorithms for both minimizing the energy consumption and compensating for the traction efficiency loss of WMRs as caused by wheel slip. However, current research is mainly focused on flat, loose terrain [16–19], and research on the control of WMRs traversing loose sloped terrain is rare. The presently available wheel-soil interaction mechanics models are too complex for control design [9]. The wheel-soil interaction model developed previously [10] involves numerous unknown dynamic time-varying parameters, which easily causes computation delay and control error. Therefore, the design of an effective control mechanism based on wheel-soil mechanics is considered a key research issue in the control of WMRs on loose terrain.

To this end, the present study investigates longitudinal slip-ratio-coordinated control of WMRs while they are climbing up a loose slope, through the planning and tracking of the slip based on an analysis of the wheel-soil interaction mechanism. First, experimental motion analysis is conducted to establish a model for the wheel-soil interaction mechanism. Second, an online planning algorithm for slip ratio that is based on the goal of optimizing drive efficiency is built. Third, a dynamic model is developed for a six-wheel mobile robot climbing loose sloped terrain for the design of its control mechanism. Furthermore, a tracking control method is proposed, wherein individual wheels' slip ratios are used as state variables and the planning slip ratios obtained by the drive efficiency optimization algorithm are used as the desired input. To improve the robustness and adaptability of the proposed tracking control method, an adaptive neural network designed with a weight error in its weight rate is used. Finally, the stability of the control law is proved using the Lyapunov method. Full-scale simulations are performed in real time using the simulation platform RoSTDyn for a six-wheel lunar rover climbing loose sloped terrain to demonstrate the effectiveness of the proposed control scheme.

The rest of this paper is organized as follows. Section 2 presents the development of a slope-based wheel-soil dynamic model and experimental motion analysis is conducted to amend the model for the wheel-soil interaction mechanism. Section 3 describes the establishment of a slip planning algorithm based on the goal of drive efficiency optimization. Section 4 presents a simplified model used for the mobility control of a planetary rover. Section 5 describes the establishment of an equation of state using the optimal slip as the desired input and the real slip as a state variable; a coordinated slip tracking system is designed using an adaptive neural network. Section 6 demonstrates the effectiveness of the developed control system via full-scale simulations performed using a six-wheel robot climbing sloped deformable terrain. Finally, Section 7 concludes the study.

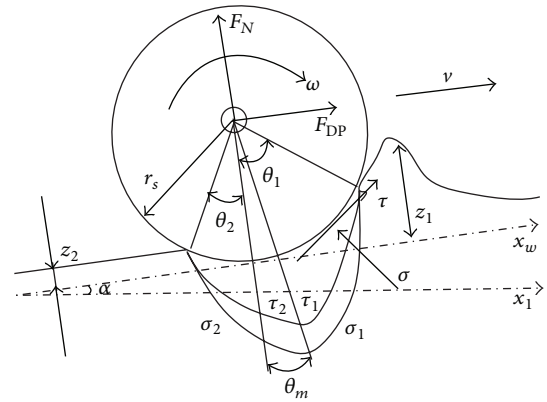


Figure 1: Wheel-soil interaction mechanism on loose sloped terrain.

2. Slope-Based Wheel-Soil Dynamic Interaction Model

When WMRs traverse loose sloped terrain, the wheels may undergo rolling, slip, and sliding movements. The slip ratio is usually used to describe wheel slip and sliding and is expressed as follows:

$$s = \frac{r_s\omega - v}{v_l} \quad v_l = \begin{cases} r_s\omega, & \forall r\omega \geq v \\ v, & \forall r\omega < v, \end{cases} \tag{1}$$

where s is the slip ratio, which means the wheel slip ratio in our study; ω is the actual angular velocity of the wheel; r_s is the effective radius of the wheel; v_l is the theoretical translational velocity of the wheel axis; and v is the actual translational velocity of the wheel axis. From (1), one can see that $-1 \leq s \leq 1$. The condition $s = 0$ indicates pure rolling, wherein the wheel touches the ground at the wheel's instantaneous velocity center; the relationship $v = r\omega$ is satisfied in this condition. The condition $s = 1$ indicates that the wheels are in a pure slip state, that is, corresponding to time when $v = 0$. When $-1 < s < 1$, the wheel is in the slip state, and, at this time, $v < r\omega$. When $-1 \leq s < 0$, the wheel is in the sliding state.

When the robot is climbing loose sloped terrain, the wheel-soil interaction is as shown in Figure 1. Here, F_N is the normal force, F_{DP} is the draw pull force, and T is the driving torque. Further, θ_1 denotes the wheel-soil interaction entry angle, θ_2 denotes the departure angle, and θ_m denotes the angle of maximum normal stress. τ and σ are the shear stress and normal stress, respectively, when the wheel-soil

interaction occurs at any point on the surface; z is the wheel sinkage, and α is the slope angle.

On the basis of the Reece formula [15], the normal stress is expressed as

$$\sigma = (k_1 + k_2 b)\left(\frac{h}{b}\right)^n, \tag{2}$$

where k_1, k_2 are soil bearing characteristic parameters, h is the sinkage, n is the soil deformation index, and b is the wheel width.

Considering the influence of slope angle on the distribution of stress on the surface, the normal stress model for a robot climbing a loose slope is expressed as follows:

$$\begin{aligned}\sigma_1(\theta) &= \left(\frac{k_1}{b} + k_2\right)(r\cos\alpha)^{n'}(\cos\theta - \cos\theta_1)^{n'} \\ &\qquad (\theta_m \le \theta \le \theta_1), \\ \sigma_2(\theta) &= \left(\frac{k_1}{b} + k_2\right)(r\cos\alpha)^{n'} \\ &\quad \times \left\{\cos\left[\theta_1 - \frac{\theta - \theta_2}{\theta_m - \theta_2}(\theta_1 - \theta_m)\right] - \cos\theta_1\right\}^{n'} \\ &\qquad (\theta_2 \le \theta \le \theta_m).\end{aligned} \tag{3}$$

θ_m is calculated by the empirical formula in (4). n' is the modified index for the condition of a robot climbing loose sloped terrain. These variables are calculated as

$$\theta_m = (c_1 + c_2 s)\theta_1, \tag{4}$$

$$\theta_2 = c_3\theta_1, \tag{5}$$

$$n' = c_1 + c_2 + c_3. \tag{6}$$

In (6), c_1, c_2 are constants determined by soil properties. Generally, $c_1 \approx 0.35$, $0 \le c_2 \le 0.25$. c_3 is the correction coefficient and is usually in the range $0 \le c_3 \le 0.075$ [15].

On the basis of the Janosi formula [15], we can determine the tangential stress, when the wheel is climbing the loose sloped terrain, as follows:

$$\begin{aligned}\tau(\theta) &= [c + \sigma(\theta)\tan\varphi]\left\{1 - \exp\left[-\frac{j(\theta)}{K}\right]\right\}, \\ j(\theta) &= r[(\theta_1 - \theta) - (1 - s)(\sin\theta_1 - \sin\theta)].\end{aligned} \tag{7}$$

In (7), c denotes the soil cohesion coefficient, φ denotes the soil friction angle, and K denotes the soil shear modulus of deformation.

Through the analysis of the wheel-soil interaction on loose sloped terrain, the correction equation for the concentrated force or torque of the wheel for WMRs climbing such a terrain can be established as

$$F_N = rb\int_{\theta_2}^{\theta_1}\sigma(\theta)\cos\theta\,d\theta + rb\int_{\theta_2}^{\theta_1}\tau(\theta)\sin\theta\,d\theta = G\cos\alpha. \tag{8}$$

In the horizontal direction, the balance equation is expressed as follows:

$$F_{DP} = F_H - F_R = m\dot{v}. \tag{9}$$

Here, F_H is the soil thrust force and F_R is the resistance force, and these are expressed, respectively, as

$$F_H = rb\int_{\theta_2}^{\theta_1}\tau(\theta)\cos\theta\,d\theta, \tag{10}$$

$$F_R = rb\int_{\theta_2}^{\theta_1}\sigma(\theta)\sin\theta\,d\theta + G\sin\alpha. \tag{11}$$

Then, the drawbar pull force can be obtained by the summation of F_H and F_R as

$$F_{DP} = -rb\int_{\theta_2}^{\theta_1}\sigma(\theta)\sin\theta\,d\theta - G\sin\varphi + rb\int_{\theta_2}^{\theta_1}\tau(\theta)\cos\theta\,d\theta. \tag{12}$$

According to the balance of Euler equations for each wheel, we get

$$T_i - T_{Ri} = I_{Wi}\dot{\omega}_{Wi}. \tag{13}$$

Here, T_i is the motor driving torque and T_{Ri} is the resistance moment, expressed as

$$T_{Ri} = r^2 b\int_{\theta_2}^{\theta_1}\tau(\theta)\,d\theta. \tag{14}$$

3. Slip Planning for Six-Wheel Robot Climbing Loose Sloped Terrain

3.1. Relationship between Wheel Slip and Key Performance Indexes of WMR. The key performance indexes of a WMR include its traction efficiency (see Figure 18), thrust coefficient and traction coefficient, and drive efficiency. These indexes are indicators of the mobile performance of a WMR. From the literature [20], a linear combination of draw-pull force and the wheel supporting force approximately gives the equivalent driving torque as

$$T = F_{DP}\cdot l + F_N\cdot e \approx F_{DP}\cdot r_s + F_N\cdot e, \tag{15}$$

$$F_{DP} \approx \frac{T}{r_s} - F_N\cdot\frac{e}{r_s}. \tag{16}$$

Here, $F_t = T/r_s$ is the thrust force, which is produced by the soil deformation caused by the rotation of the motor-driven wheel. $F_N\cdot e/r_s$ is the resistance of the soil, and e/r_s is the wheel resistance coefficient that reflects the resistance of the soil to prevent the rotation of the wheel. The wheel traction is the drawbar pull force minus the soil resistance generated by the traction. The wheel-soil interaction generates draw-pull force that drives the movement of the robot.

Equation (16) can be transformed as

$$\frac{F_{DP}}{F_t} + F_N\cdot\frac{e_s}{rF_t} \approx 1. \tag{17}$$

Table 1: Geometrical parameters of wheels.

Wheel	Radius r (mm)	Width b (mm)	Lug height h (mm)
Wh1	157.4	165	15
Wh2	135	110	15
Wh3	135	165	15
Wh4	135	165	10
Wh5	135	165	10

Let $\mathrm{PE} = F_{\mathrm{DP}}/F_t = F_{\mathrm{DP}} r_s/T$ denote the traction efficiency. It indicates how much driving force transforms into the effective draw-pull force. From (17), it is understood that the force generated by the motor rotation and thrust force is partly used to generate traction force and partly to overcome the resistance of the soil. Dividing (16) by the normal load F_N gives

$$\frac{F_{\mathrm{DP}}}{F_N} \approx \frac{T}{(F_N r_s)} - \frac{e}{r_s}. \tag{18}$$

Let $\mathrm{PC} = F_{\mathrm{DP}}/F_N$ be the traction coefficient, which indicates the draw-pull force for providing traction under a unit load, and let $\mathrm{TC} = T/(F_N r_s)$ be the thrust coefficient, which indicates the motor-provided thrust force under a unit load. Then, we have

$$\begin{gathered} \mathrm{PC} \approx \mathrm{TC} - \frac{e}{r_s}, \\ \mathrm{PE} = \frac{(F_{\mathrm{DP}}/F_N)}{(T/F_N r_s)} = \frac{\mathrm{PC}}{\mathrm{TC}}. \end{gathered} \tag{19}$$

TE is an important index that denotes the wheel's drive efficiency. Combining with (1), we can express TE as

$$\mathrm{TE} = F_{\mathrm{DP}} \cdot \frac{v}{T\omega} = F_{\mathrm{DP}} \cdot \frac{r_s(1-s)}{T} = \mathrm{PE}(1-s). \tag{20}$$

From the above analysis, we can say that the variables PE, PC, TC, and TE reflect the relationship between the WMR's mobile performance and the slip ratio.

To confirm the influence of the slip ratio on the WMR's mobile performance, experiments on the wheel-soil interaction were conducted for the scenario of a WMR climbing a sloped test bed developed at the State Key Laboratory of Robotics and System at Harbin Institute of Technology, China. In the experiment, five kinds of wheels with different dimensions and wheel lugs were used on different loose soils. Tables 1 and 2 list the parameters of the wheels and of the terrain mechanics, respectively, where r is the wheel radius, b the wheel width, and h the lug height. The travelling velocity of the WMR was 10 mm/s.

A six-wheel robot was considered an example rover and was made to climb loose sloped terrain angled at 0°, 5°, 10°, 15°, 20°, and 25° to study the relationship between the slip ratio and the key performance indexes (Figure 2). Figure 3 shows the relationship of the performance indexes PE, PC, TC, and TE with the wheel slip ratio in this climbing scenario, with different ground mechanics parameters and different wheel parameters.

Table 2: Parameters of terrain mechanics of sand and lunar soil [14].

Terrain parameter	Sand	Lunar soil
n	1.10	1.0
k_c (kPa/m^{n-1})	0.95	1.4
k_φ (kPa/m^n)	1528.43	820
c (kPa)	1.04	0.52 (0.1~2.7)
φ (°)	28	42 (25~50)
K (m)	0.0254	0.0178

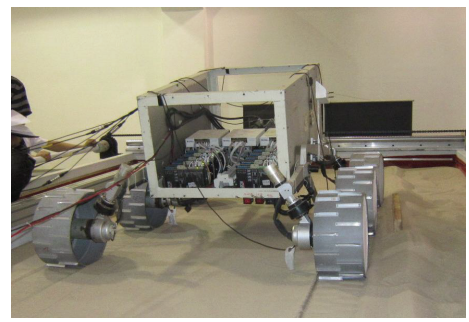
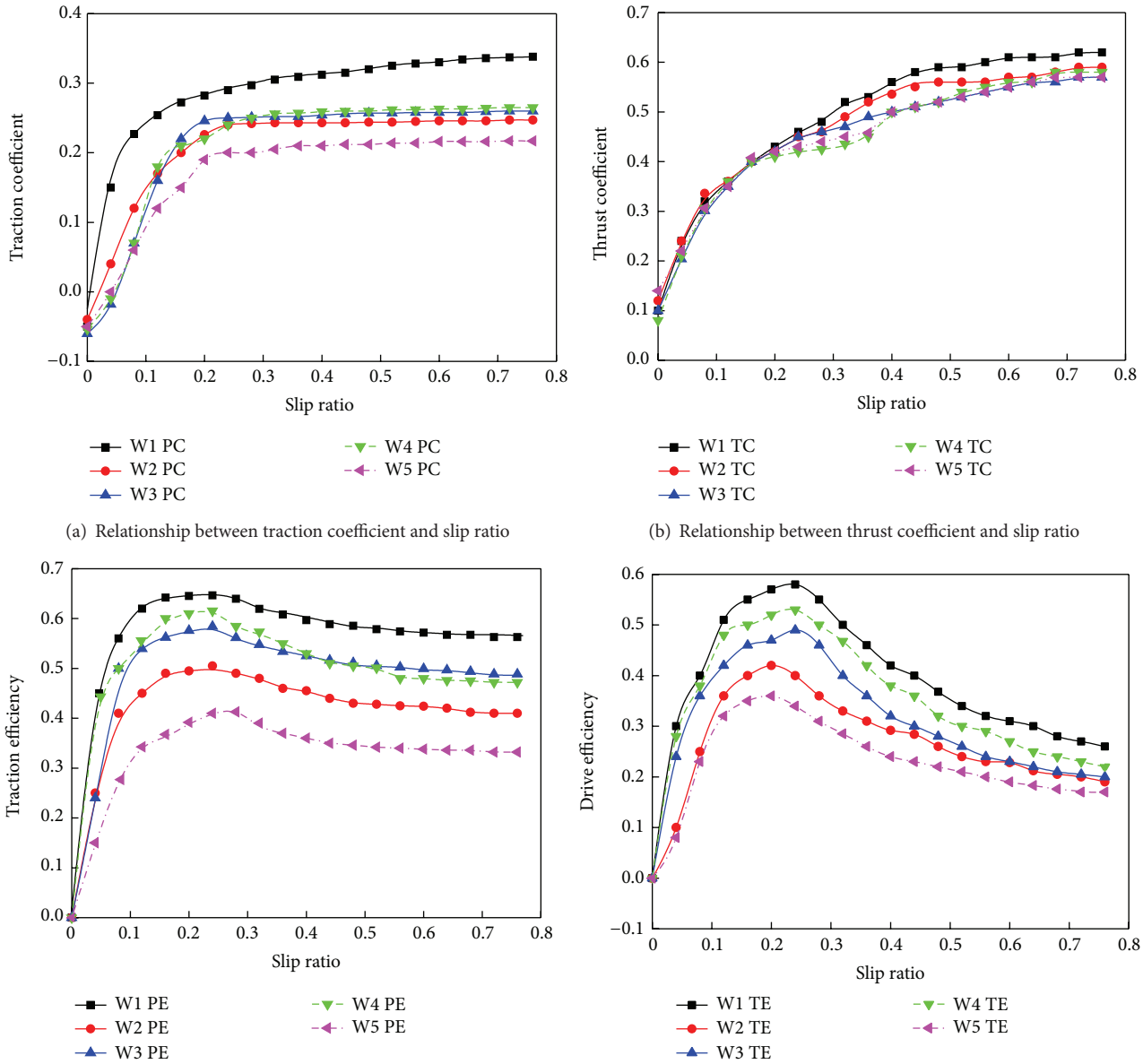


Figure 2: Photograph of six-wheel robot climbing loose sloped terrain in an experiment.

From Figure 3, we can see that the key performance indexes of the WMR are optimal at a wheel slip ratio in the range of 0.1 to 0.4. The index TE is a function of PE, PC, and TC; therefore, if we use an optimization algorithm for TE, we can perform dynamic programming at each wheel slip ratio to ensure the least possible wheel sinkage and thus obtain maximal TE of the WMR.

3.2. Slip Ratio Planning Algorithm Based on Optimal Drive Efficiency. When robots climb loose sloped terrain with constant angular control (CAC), each parameter of the wheel-soil mechanics is likely to be different, implying that the wheel slip ratios would also possibly be different. This would inevitably lead to "incoordination" between wheels. Therefore, we aim to optimize the drive efficiency for achieving the desired wheel slip ratio and to control the drive torque of all wheels in a coordinated manner, to be able to track the desired slip ratio and reduce the wheel sinkage. Through tracking of the desired slip ratio for coordinating the energy distributions of each wheel, the drive efficiency of all the wheels can be ensured to be maximal. If the wheel width, radius, payload, and contact angle of the wheel with the terrain are known for a particular WMR, it is possible to determine the desired wheel slip ratio for climbing loose sloped terrain through drive efficiency optimization. The slip ratio planning goal based on drive efficiency optimization is as follows: minimum energy consumption and maximum drive efficiency for a unit running distance of a WMR. The following factors/conditions need to be considered to meet this goal: (1) the force equilibrium equation, (2) maintenance of contact of each of the wheels with the ground and a normal force greater than zero, (3) a soil resistance moment less than the maximum motor torque, and (4) wheel-soil mechanics.



(a) Relationship between traction coefficient and slip ratio

(b) Relationship between thrust coefficient and slip ratio

(c) Relationship between traction efficiency and slip ratio

(d) Relationship between drive efficiency and slip ratio

Figure 3: Key performance indexes of robot climbing loose sloped terrain versus wheel slip ratio.

Based on the factors/conditions of optimal TE, the expected slip ratio planning algorithm for a target WMR, that is,

$$\min \quad J(s_i) = \sum_{i=1}^{6} \left| \frac{d\left(F_{\mathrm{DP}i} r_s (1 - s_i) / T_i\right)}{ds_i} \right|, \tag{21}$$

should meet the following constraints:

$$\begin{aligned} \text{s.t.} \quad & T_i = M_{Ri} \le T_{i\max} \\ & |s_i| < 1 \\ & F_{\mathrm{DP}i} = F_{\mathrm{DP}i}(\theta_{1i}, \theta_{2i}) \\ & F_{Ni} = F_{Ni}(\theta_{1i}, \theta_{2i}) \\ & T_i = T_i(\theta_{1i}, \theta_{2i}) \quad i = 1, \ldots, 6. \end{aligned} \tag{22}$$

When the ground mechanics parameters are known, the flowchart of the planning algorithm for the optimal-TE-based desired slip ratio is as shown in Figure 4.

4. Dynamics Modeling of Six-Wheel Robot Climbing Loose Sloped Terrain Based on Wheel-Soil Mechanics

An optimal-TE-based slip ratio can ensure minimum energy consumption and maximum driving efficiency. To design the tracking control law for the optimal-TE-based desired slip ratio that would be in agreement with the actual slip ratio, we first perform dynamics modeling of climbing of WMRs on loose sloped terrain based on wheel-soil mechanics. Here,

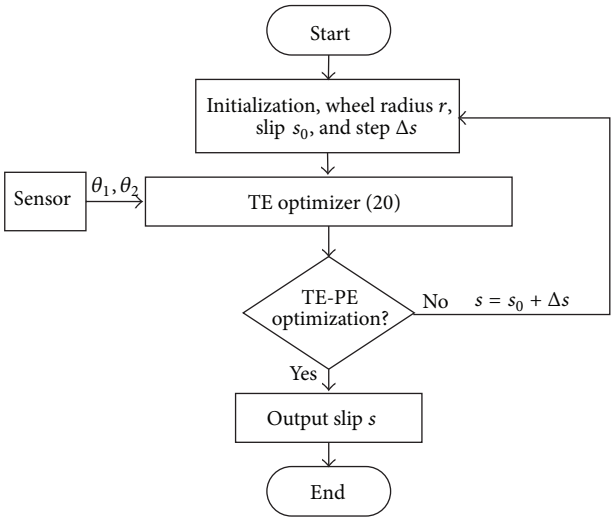


FIGURE 4: Flowchart of desired slip ratio based on optimal drive efficiency.

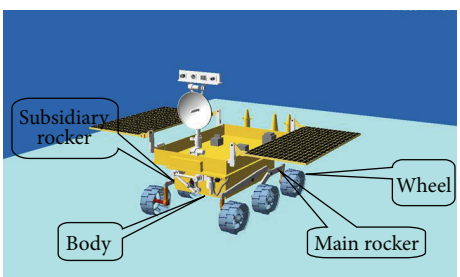


FIGURE 5: Schematic of six-wheel rocker-type mobile robot.

again, a six-wheel mobile robot is the research object; owing to its six independent drives and rocker mechanism, it can passively adapt to loose sloped terrain. Figure 5 shows the schematic of the six-wheel rocker-type mobile robot.

This mobile robot has the following features when climbing loose terrain: (a) when the robot is in longitudinal motion, the system dynamics model is a linear combination of the wheel-soil mechanics; (b) the robot forces on the left and right sides influence each other; (c) sinkage of wheels caused by wheel slip causes the tilting of the robot body.

Figure 6 shows the dynamics model based on the wheel-soil mechanics in the case of the six-wheel robot climbing loose sloped terrain.

Equation (23) represents the control-based wheel-soil dynamics model for a six-wheel robot while climbing up deformable slopes with longitudinal slip; it was obtained by the simplification of (9), (12), and (13):

$$m\dot{v} = \sum_{i=1}^{6} F_{\mathrm{DP}i}(\theta_{1i}, \theta_{2i}) - G\sin(\alpha + \Delta\alpha),$$
$$I_W \dot{\omega}_{Wi} = T_i - T_{Ri}(\theta_{1i}, \theta_{2i}). \tag{23}$$

$\Delta\alpha$ is the sinkage incline angle, $|\Delta\alpha| \le \zeta$, and ζ is a constant. A small ζ implies linearized derivation, $\sin(\alpha + \Delta\alpha) = \sin\alpha + \sin\Delta\alpha$, and, using (11)–(13),

$$\dot{s}_i = -\frac{(1 - s_i)\left[\sum_{i=1}^{6} F_{\mathrm{DP}i}(\theta_{1i}, \theta_{2i}) - G\sin(\alpha + \Delta\alpha)\right]}{Mv} + \frac{r_s(1 - s_i)^2\left[T_i - T_{Ri}(\theta_{1i}, \theta_{2i})\right]}{I_W v}. \tag{24}$$

Next, we define the output function as

$$\mathbf{h}(\mathbf{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{bmatrix}^T = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \end{bmatrix}^T. \tag{25}$$

Combining (23) and (24), we can obtain the standard form of affine nonlinear systems as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} + \mathbf{d}(\mathbf{x}),$$
$$\mathbf{h}(\mathbf{x}) = \mathbf{x}, \tag{26}$$

where

$$\mathbf{x} = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \end{bmatrix}^T,$$
$$\mathbf{u} = \begin{bmatrix} T_1 & T_2 & T_3 & T_4 & T_5 & T_6 \end{bmatrix}^T,$$
$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(x) & f_2(x) & f_3(x) & f_4(x) & f_5(x) & f_6(x) \end{bmatrix}^T,$$
$$f_i(x) = -\frac{(1 - s_i)\left[\sum_{i=1}^{n=6} F_{\mathrm{DP}i}(\theta_{1i}, \theta_{2i}) - G\sin\alpha\right]}{Mv} - \frac{r_s(1 - s_i)^2 T_{Ri}(\theta_{1i}, \theta_{2i})}{I_{Wi} v},$$
$$\mathbf{g}(x) = \begin{bmatrix} g_1 & \cdots & g_6 \end{bmatrix} \qquad g_i(x) = \frac{r_s(1 - s_i)^2}{I_{Wi} v},$$
$$\mathbf{d}(\mathbf{x}) = \begin{bmatrix} d_1(x) & d_2(x) & d_3(x) & d_4(x) & d_5(x) & d_6(x) \end{bmatrix}^T,$$
$$d_i(x) = \frac{(1 - s_i) G\sin\Delta\alpha}{Mv}. \tag{27}$$

Then, (26), that is, an equation for a multiple-input, multiple-output (MIMO) system, can be decomposed into (28) for six single-input, single-output (SISO) subsystems. Further, we can design a controller for a SISO subsystem:

$$\dot{x}_i = f_i(x) + g_i(x)u_i + d_i(x), \quad i = 1, \ldots, 6. \tag{28}$$

5. Radial Basis Function-Based Adaptive Sliding Tracking Control for Desired Slip Ratio

5.1. Approximation Properties of Radial Basis Function Network. A radial basis function (RBF) network is a kind of

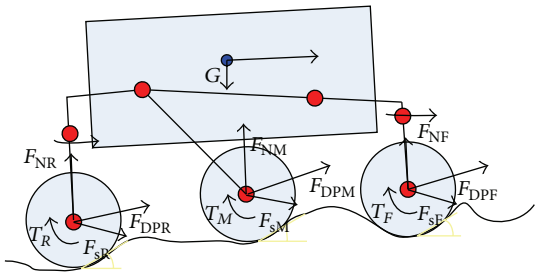


Figure 6: Lateral view of dynamics model based on wheel-soil mechanics during climbing of WMR on loose sloped terrain.

a three-layer forward neural network. The first layer is the input, made up of source nodes. The input vector map is directly connected to the second hidden layer, and the third layer is the output. The hidden layer is a transformation function unit, called a Gaussian function. Generally, an RBF network is expressed as the excitation function of a hidden unit as follows:

$$\phi_i(\kappa) = \exp\left[-\frac{1}{2b_i^2}\|\kappa - \xi_i\|^2\right], \quad i = 1, 2, \ldots, n. \tag{29}$$

Here, n is the hidden unit number, κ is the network input vector, and ξ_i and b_i are the ith basis functions for the center and radius, respectively. The output layer node is the ith for nodes in the hidden layer output linear weighted sum:

$$y_i = \sum_{j=1}^{m} w_{ij}\phi_j(x), \tag{30}$$

where w_{ij} is the weight between ϕ_j and y_i.

For any continuous nonlinear function $f(x)$, one can use the approximate representation of the RBF network as

$$f(x) = \mathbf{W}^T\phi(x) + \varepsilon(x), \tag{31}$$

where $\mathbf{W}$ is the weight matrix of the RBF network and $\varepsilon(x)$ is the reconstruction error of the neural circuits. The literatures [21, 22] have proved that the RBF network can approximate any continuous nonlinear function of arbitrary precision.

5.2. RBF-Based Adaptive Sliding Tracking Control Law. For the system given in (28), if f_i and g_i are precisely known, one can design the general sliding mode control system for tracking the desired slip ratio. When WMRs climb loose sloped terrain, the optimal slip ratio tracking error for the ith wheel is $e_i = s_i - s_{id}$, where s_i is the ith actual feedback wheel slip ratio and s_{id} is the wheel's desired slip ratio.

We express the sliding-mode surface as

$$S_i = \rho_i e_i = \rho_i(s_i - s_{id}), \tag{32}$$

where ρ_i is a proportional coefficient, generally a normal number, and is used to determine the attenuation speed of the tracking error. The ideal sliding mode can be expressed as

$$\dot{S}_i = \rho_i(\dot{s}_i - \dot{s}_{id}) = 0. \tag{33}$$

In order to meet the arrival condition of sliding mode variable structure control and in the shortest time to reach the sliding mode surface, to ensure high robustness of the system, the following is the improved index-based control law:

$$\dot{S}_i = -\varepsilon_i|S_i|\operatorname{sgn}(S_i) - k_iS_i. \tag{34}$$

Here, $\varepsilon_i > 0$, $K_i > 0$, and $\operatorname{sgn}(S_i)$ is a symbolic function. Upon substituting (28) into (33), we obtain the system control law as

$$u_i = \frac{1}{\rho_i g_i(x, v)}\left[\rho_i f_i(x, v) + \varepsilon_i|S_i|\operatorname{sgn}(S_i) + k_iS_i - \rho_i s_{di}\right]. \tag{35}$$

At this point, consider the following Lyapunov function candidate:

$$V = \sum_{i=1}^{6} V_i, \quad V_i = \frac{1}{2}S_i^2. \tag{36}$$

Differentiating it gives

$$\begin{aligned}\dot{V}_i &= S_i\dot{S}_i = S_i(-\varepsilon_i|S_i|\operatorname{sgn}(S_i) - k_iS_i)\\ &= -\varepsilon_iS_i|S_i|\operatorname{sgn}(S_i) - k_iS_i^2\\ &= -\varepsilon_iS_i^2 - k_iS_i^2 < 0,\end{aligned} \tag{37}$$

where $V \geq 0$ and $\dot{V} < 0$ are guaranteed to be negative, implying $V \to 0$ and also $S \to 0$ and $\dot{S} \to 0$ as $t \to \infty$. Therefore, global stability is guaranteed by the Lyapunov theorem.

However, in actual circumstances, f_i and g_i contain many unknown parameters, some of which are dynamic time-varying and therefore cannot be known precisely. Hence, this problem imposes a major limitation on the ideal sliding mode controller given in (35). An RBF neural network possesses the property of an approximate arbitrary nonlinear function, which is not applicable for precision parameters. Therefore, we consider using neural networks with dynamic approximate values $\tilde{f}_i$ and $\tilde{g}_i$ of variables f_i and g_i with unknown parameters, based on the RBF neural network function approximation theory.

Let us assume that $\tilde{f}_i$, $\tilde{g}_i$ have approximation weights w_{fi}, w_{gi} and that the f_i, g_i approximation errors are ε_{fi} and ε_{gi}, respectively. That is,

$$\tilde{f}_i(x) = \tilde{f}(x \mid w_{fi}) = w_{fi}^T\phi(x) + \varepsilon_{fi}, \tag{38}$$

$$\tilde{g}_i(x) = \tilde{g}(x \mid w_{gi}) = w_{gi}^T\phi(x) + \varepsilon_{gi}. \tag{39}$$

Because the RBF neural network is used to approximate an unknown nonlinear function in the system, an approximation error inevitably exists. The neural network fitting error should be reduced as possible, so we define $\tilde{w}_{fi}$ and $\tilde{w}_{gi}$ as the optimal weights to estimate w_{fi} and w_{gi}:

$$\tilde{w}_{fi} = \arg\min_{w_{fi}\in\Omega_{fi}}\left[\sup_{x\in\Omega_x}\left|\tilde{f}_i(x \mid w_{fi}) - f_i(x)\right|\right], \tag{40}$$

$$\tilde{w}_{gi} = \arg\min_{w_{gi}\in\Omega_{gi}}\left[\sup_{x\in\Omega_x}\left|\tilde{g}_i(x \mid w_{gi}) - g_i(x)\right|\right]. \tag{41}$$

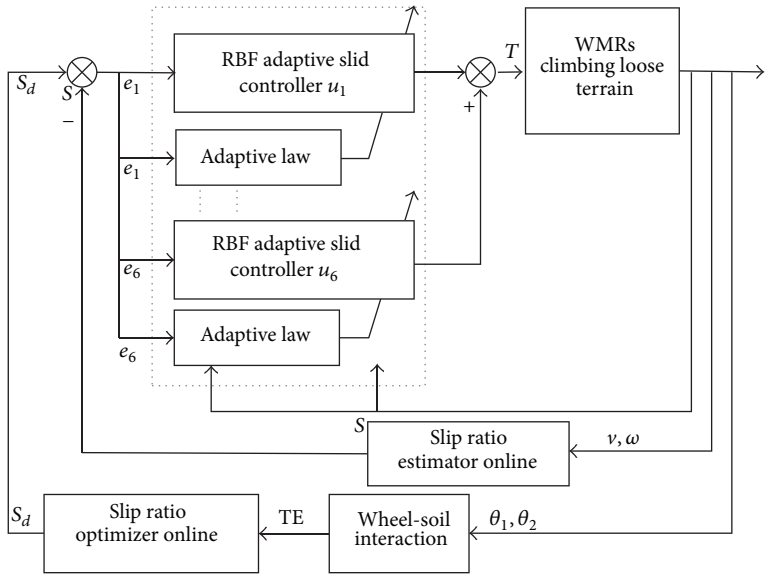


Figure 7: Flowchart for optimal slip ratio tracking control based on RBF neural network adaptive sliding mode.

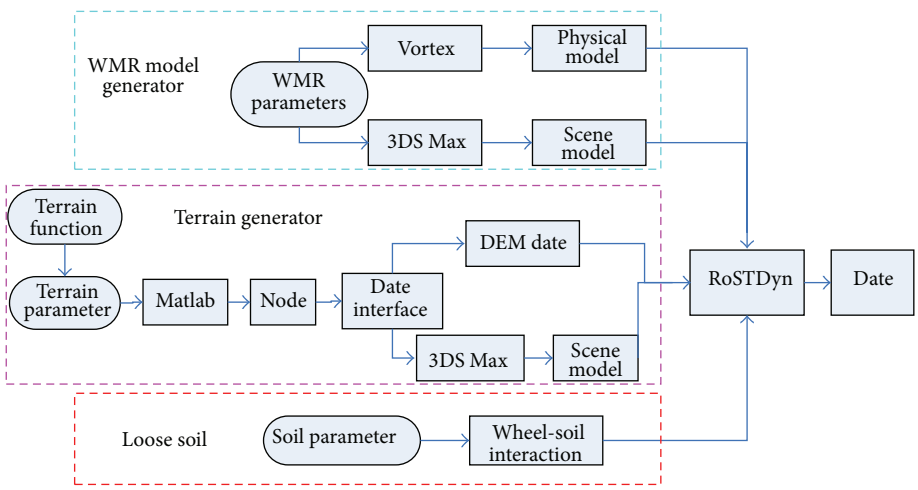


Figure 8: Schematic of simulation using RoSTDyn system.

In (40) and (41), $\Omega_x \subset R^n$, Ω_{fi} and $\Omega_{gi} \subset R^m$. Ω_x, Ω_{fi}, and Ω_{gi} all are compact sets further and n and m are the input layer and hidden layer nodes, respectively. As a result, $\widetilde{f}_i(x \mid \widetilde{w}_{fi})$ and $\widetilde{g}_i(x \mid \widetilde{w}_{gi})$ are the optimal approximations of $f_i(x)$ and $g_i(x)$, respectively. Then, we define the weights of the neural network error as

$$\eta_{fi} = \widetilde{w}_{fi} - w_{fi}, \qquad \eta_{gi} = \widetilde{w}_{gi} - w_{gi}, \tag{42}$$

where $\dot{\eta}_{fi} = \dot{\widetilde{w}}_{fi} - \dot{w}_{fi}$, $\dot{\eta}_{gii} = \dot{\widetilde{w}}_{gi} - \dot{w}_{gi}$.

Taking the weights of the neural network, w_{fi} and w_{gi}, as the adaptive control law, we get

$$\dot{w}_{fi} = -\gamma_{fi} S_i \phi_{fi}, \qquad \dot{w}_{gi} = \gamma_{gi} S_i \phi_{gi} u_i. \tag{43}$$

Based on (35), (39), (40), and (43), we get the slip ratio tracking control law using RBF neural network adaptive sliding method, expressed as

$$u_i = \frac{1}{\rho_i \widetilde{g}_i(x)} \left[\rho_i \widetilde{f}_i(x) + \varepsilon_i |S_i| \operatorname{sgn}(S_i) + k_i S_i - \rho_i s_{di} \right]. \tag{44}$$

5.3. Stability Analysis. Let us define the minimum approximation error of a neural network as

$$\delta_i = \left[\widetilde{f}_i\left(x \mid \widetilde{w}_{fi}\right) - f_i(x) \right] + \left[\widetilde{g}_i\left(x \mid \widetilde{w}_{gi}\right) - g_i(x) \right] u_i. \tag{45}$$

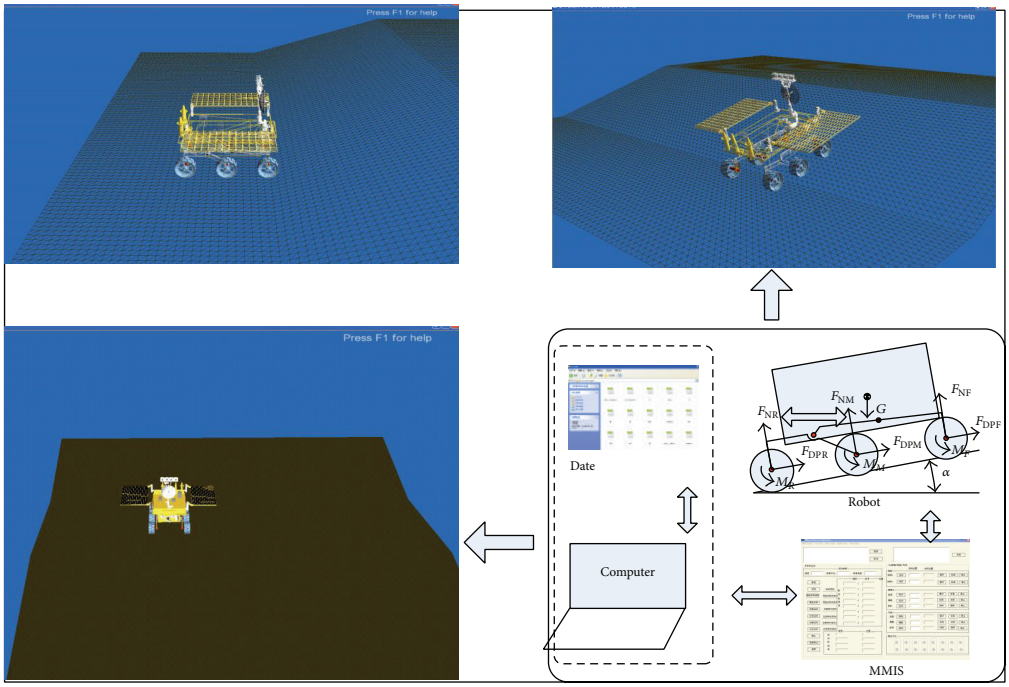


Figure 9: Six-wheel lunar rover climbing sloped terrain in RoSTDyn 3D simulation.

Then, substituting (44) and (45) into (37) gives

$$\begin{aligned}\dot{S}_i &= \rho_i\left(f_i(x)+g_i(x)u_i\right)-\rho_i\dot{s}_{di}\\ &= \rho_i\left(f_i(x)+\tilde{g}_i\left(x\mid w_{gi}\right)u_i+g_i(x)u_i\right)\\ &\quad-\tilde{g}_i\left(x\mid w_{gi}\right)u_i-\rho_i\dot{s}_{di}\\ &= f_i(x)-\tilde{f}_i\left(x\mid w_{fi}\right)+\rho_i\dot{x}_{di}-k_i\operatorname{sign}S_i+g_i(x)u_i\\ &\quad-\tilde{g}_i\left(x\mid w_{gi}\right)u_i-\rho_i\dot{s}_{di}\\ &= f_i(x)-\tilde{f}_i\left(x\mid w_{fi}\right)+\left[g_i(x)u_i-\tilde{g}_i\left(x\mid w_{gi}\right)u_i\right]\\ &\quad-k_i\operatorname{sgn}S_i\\ &= \tilde{f}_i\left(x\mid \tilde{w}_{fi}\right)-\tilde{f}_i\left(x\mid w_{fi}\right)\\ &\quad+\left[\tilde{g}_i\left(x\mid \tilde{w}_{gi}\right)-\tilde{g}_i\left(x\mid w_{gi}\right)\right]u_i-\delta_i-k_i\operatorname{sgn}S_i\\ &= \left(\tilde{w}_{fi}^T-w_{fi}^T\right)\phi_{fi}(x)+\left(\tilde{w}_{gi}^T-w_{gi}^T\right)\phi_{gi}(x)u_i\\ &\quad-\delta_i-k_i\operatorname{sgn}S_i\\ &= \eta_{fi}^T\phi_{fi}(x)+\eta_{gi}^T\phi_{gi}(x)u_i-\delta_i-k_i\operatorname{sgn}S_i.\end{aligned} \tag{46}$$

Consider the following Lyapunov function candidate:

$$\begin{aligned}V &= \sum_{i=1}^{6}V_i,\\ V_i &= \frac{1}{2}\left(S_i^2+\frac{1}{\gamma_{fi}}\eta_{fi}^T\eta_{fi}+\frac{1}{\gamma_{gi}}\eta_{gi}^T\eta_{gi}\right).\end{aligned} \tag{47}$$

Differentiating it gives

$$\begin{aligned}\dot{V}_i &= S_i\dot{S}_i+\frac{1}{\gamma_{fi}}\eta_{fi}^T\dot{\eta}_{fi}+\frac{1}{\gamma_{gi}}\eta_{gi}^T\dot{\eta}_{gi}\\ &= S_i\left(\eta_{fi}^T\phi_{fi}(x)+\eta_{gii}^T\phi_{gii}(x)u_i-\delta_i-k_i\operatorname{sgn}(S_i)\right)\\ &\quad+\frac{1}{\gamma_{fi}}\eta_{fi}^T\dot{\eta}_{fi}+\frac{1}{\gamma_{gi}}\eta_{gi}^T\dot{\eta}_{gi}\\ &= S_i\eta_{fi}^T\phi_{fi}(x)+\frac{1}{\gamma_{fi}}\eta_{fi}^T\dot{\eta}_{fi}+S_i\eta_{gi}^T\phi_{gi}(x)u_i\\ &\quad+\frac{1}{\gamma_{gi}}\eta_{gi}^T\dot{\eta}_{gi}-\delta_iS_i-S_ik_i\operatorname{sgn}(S_i)\\ &= \frac{1}{\gamma_{fi}}\eta_{fi}^T\left(\dot{\eta}_{fi}+\gamma_{fi}S_i\phi_{fi}(x)\right)\\ &\quad+\frac{1}{\gamma_{gi}}\eta_{gi}^T\left(\dot{\eta}_{gi}+\gamma_{gi}S_i\phi_{gi}(x)u_i\right)-\delta_iS_i-k_i|S_i|^2\\ &= -\delta_iS_i-k_i|S_i|^2.\end{aligned} \tag{48}$$

Based on the approximation theory of the RBF neural network, as long as numerous hidden layer nodes exist, the adaptive RBF neural network can give an infinitesimal value of the approximation error. Then,

$$\dot{V}_i\le 0,\qquad \dot{V}\le 0. \tag{49}$$

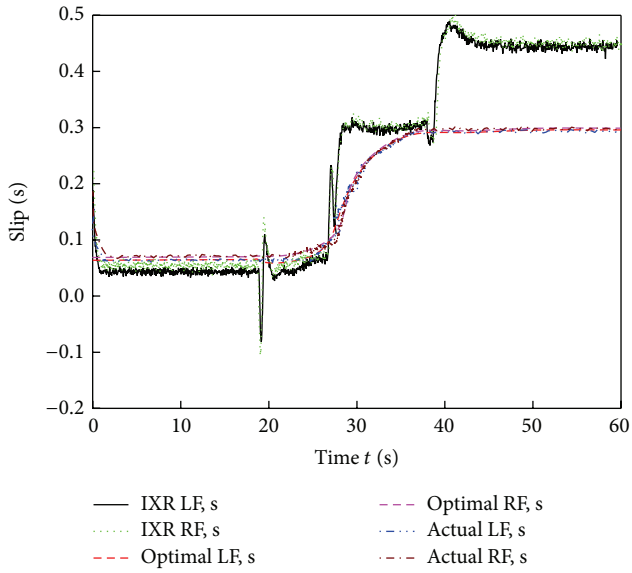


Figure 10: Front-wheels slip ratio comparison of different control strategy.

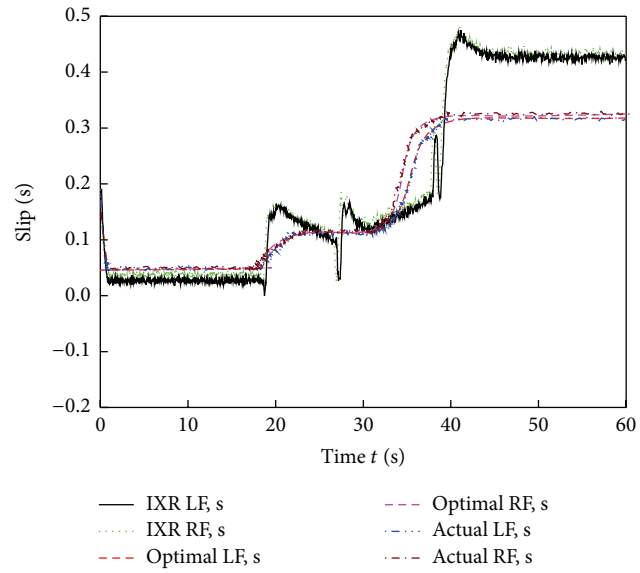


Figure 11: Middle-wheels slip ratio comparison of different control strategy.

Then, to reduce chattering that occurs usually, boundary layer methods are used, wherein a saturation function is employed instead of a signum function:

$$\operatorname{sat}(S_i, \phi_i) = \begin{cases} 1, & S_i > \phi_i \\ k_i S_i, & |S_i| \le \phi_i, k_i \phi_i = 1 \\ -1, & S_i < -\phi_i, \end{cases} \tag{50}$$

where ϕ_i is the normal to the boundary layer thickness.

The flowchart for optimal slip ratio tracking control based on the RBF neural network adaptive sliding mode in the case of a WMR climbing loose sloped terrain is shown in Figure 7.

6. Simulation Experiment Based on RoSTDyn Platform

6.1. RoSTDyn. RoSTDyn is a multibody dynamics simulation platform for WMRs moving on loose sloped terrain; it uses a Vortex engine and VC++ and was developed by the RCAMC Laboratory. Here, VC++ was used to generate the WMR system, terrain module, and wheel-soil interaction mechanics model, and the kinetic function and scene function provided by the Vortex engine were used for calculations and 3D realization. The basic simulation outline is as shown in Figure 8. From the literature [23], it is known that the RoSTDyn robot provides precise simulations on loose sloped terrain through comprehensive testing.

6.2. Simulation of Control of Six-Wheel Lunar Rover Climbing Loose Sloped Terrain. Consider a six-wheel lunar rover as an example. When it is climbing a slope on the moon, the most influential factor is the slip ratio of the wheels. Using the proposed control algorithm (RBFAS) to adjust the drive torque to track the desired slip ratio, the lunar rover can effectively climb up a slope with the least possible wheel sinkage and maximum drive efficiency of all the wheels.

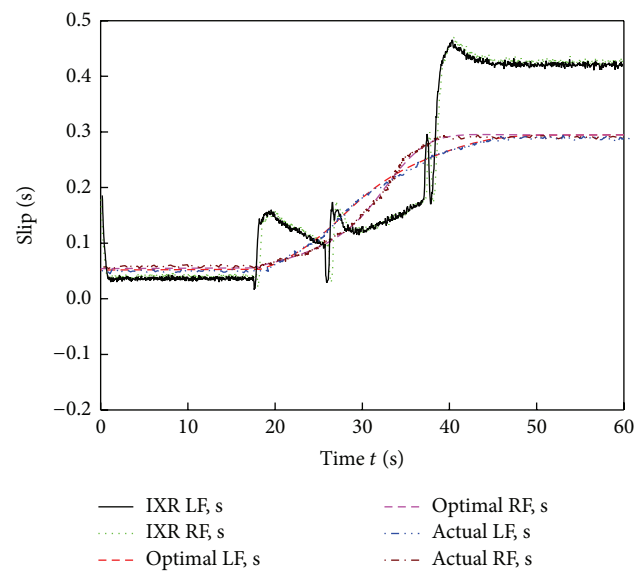


Figure 12: Behind-wheels slip ratio comparison of different control strategy.

This simulation is performed as follows. The lunar rover is started from its initial stationary state. Then, the desired slip ratio based on the optimal TE is input into the slip ratio tracking algorithm. The wheel drive moment is adjusted to control the wheel slip ratio and obtain the optimal slip ratio.

The system parameters for the simulation are set as follows.

(a) Robot parameters are the following: $M = 120\,\text{kg}$, $m_w = 1.75\,\text{kg}\cdot\text{m}^2$, $r = 0.15\,\text{m}$, $b = 0.15\,\text{m}$, and $h = 0.01\,\text{m}$.

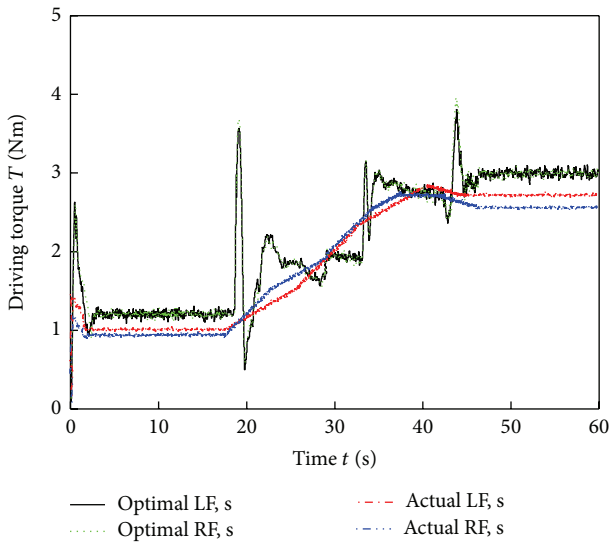


FIGURE 13: Front-wheels driving torque comparison of different control strategy.

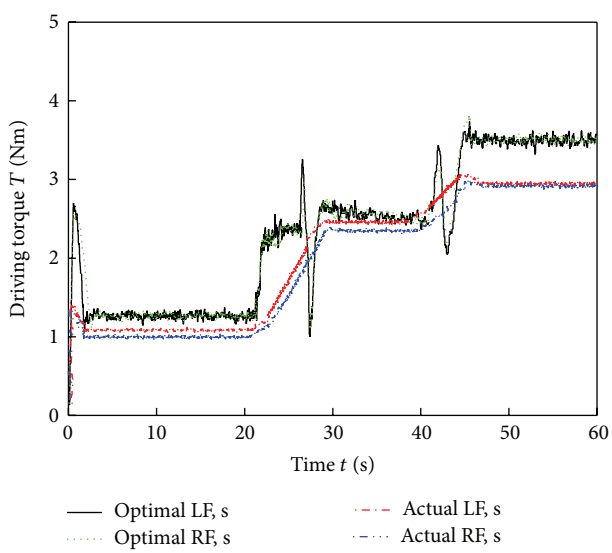


FIGURE 14: Middle-wheels driving torque comparison of different control strategy.

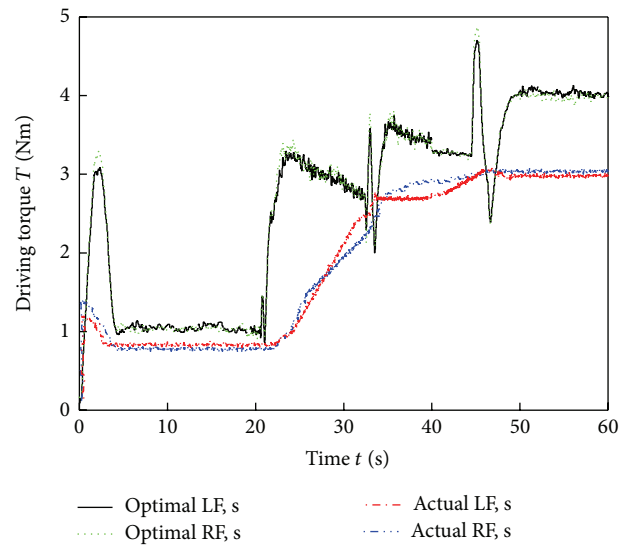


FIGURE 15: Behind-wheels driving torque comparison of different control strategy.

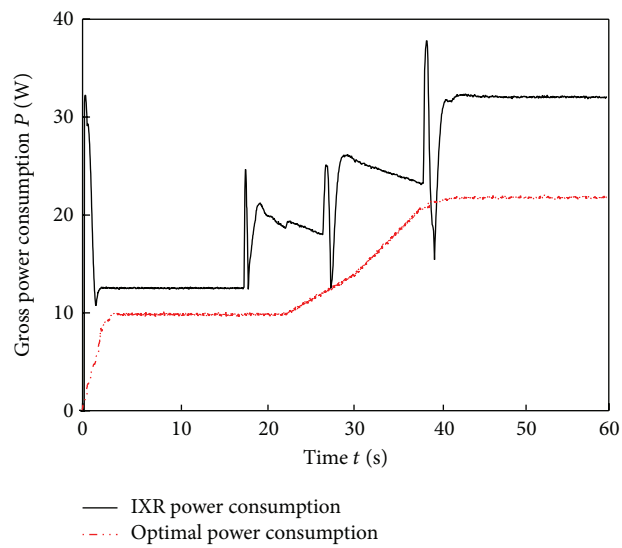


FIGURE 16: Power consumption comparison of different control strategy.

(b) Sandy soil characteristic parameters are the following: $k_1 = 1800$, $k_2 = 820000$, $g = 1.6333\,\mathrm{m/s^2}$, $c = 520$, $\varphi = 42^\circ$, $K = 0.01732$, $c_1 = 0.35$, $c_2 = 0.042$, and $c_3 = 0.012$.

(c) Control parameters are the following: $\rho_i = 10$, $\sigma_i = 0.17$, $b_i = [10\ \ 10\ \ 10\ \ 10\ \ 10]^T$, $\gamma_{fi} = 5$, $\gamma_{gii} = 1$, and $w_{fi} = w_{gi} = [6\ \ 6\ \ 6\ \ 6]$.

Figure 9 shows images of the six-wheel lunar rover climbing the sloped terrain in the RoSTDyn 3D simulation platform. Figures 10, 11, 12, 13, 14, 15, and 16 show the key performance indices of WMR comparisons climbing up a 25° slope under RBFAS and CAC control strategies.

The simulation results in Figures 10–15 show that the RBFAS control algorithm given in (45) is superior to those using CAC. Figure 17 shows that the RBFAS control algorithm in (45) significantly reduces the energy consumption and drive efficiency of the six-wheel lunar rover.

7. Conclusion

In this study, we analyzed wheel-soil interaction based on traditional terrain mechanics for WMRs climbing loose sloped terrain and determined the influence of key performance indexes of the WMRs on the wheel slip ratio. We developed an online slip ratio planning algorithm based on the optimal drive efficiency (TE) of the WMRs. Next, using

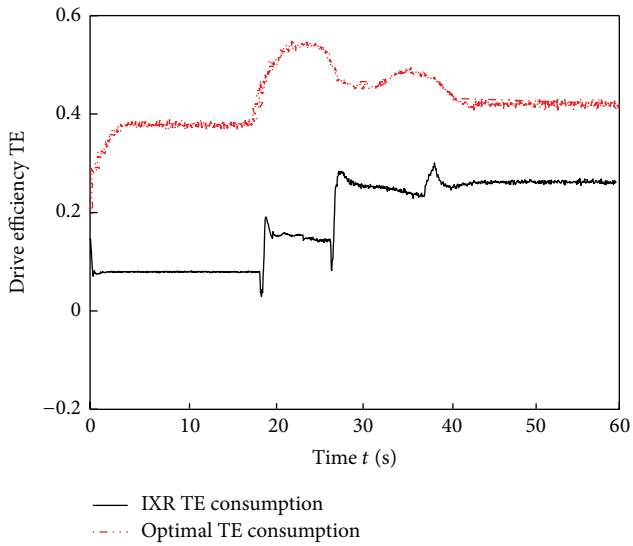


Figure 17: Drive efficiency comparison of different control strategy.

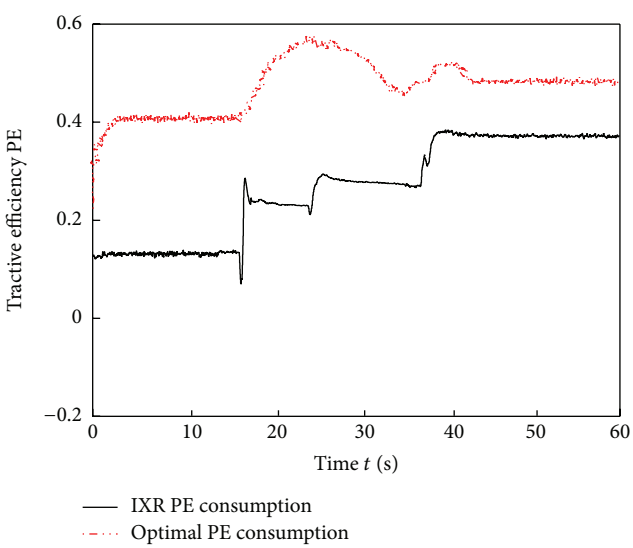


Figure 18: Tractive efficiency comparison of different control strategy.

the optimal-TE-based slip ratio as the input, the actual slip ratio as the state variable, and the wheel drive moment as the control input, we established a tracking system for the optimal-TE-based slip ratio using the method of nonlinear decoupling design. This was done with the aim of improving the robustness and adaptability of the tracking system. An adaptive neural network was used and a weight error in the weight rate was introduced in this network. The control stability of the system was confirmed using the Lyapunov method. Finally, full-scale simulations were performed to verify that the proposed control scheme not only retains the stability of the system but also improves the robot's mobile performance significantly.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was financially supported by the Natural Science Foundation of China (nos. 50975059 and 61005080), the National Postdoctoral Fund (no. 20100480994), the Postdoctoral Fund in Heilongjiang Province, and the "111" Project (no. B07018).

References

[1] K. Iagnemma and S. Dubowsky, "Traction control of wheeled robotic vehicles in rough terrain with application to planetary rovers," *International Journal of Robotics Research*, vol. 23, no. 10-11, pp. 1029–1040, 2004.

[2] M. McKee, *One Mars Rover Gets Stuck In, the Other Chills Out*, 2006, http://www.newscientist.com/article/dn9253-one-mars-rover-gets-stuck-in-the-other-chills-out.html#.U_Hk5uOSxrM.

[3] G. Webster and D. Brown, "Now a Stationary Research Platform, NASA's Mars Rover Spirit Starts a New Chapter in Red Planet Scientific Studies," 2010, http://www.nasa.gov/mission_pages/mer/news/mer20100126.html.

[4] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi, "A stable tracking control method for an autonomous mobile robot," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 384–389, Washington, DC, USA, May 1990.

[5] D. Kim and J. Oh, "Tracking control of a two-wheeled mobile robot using input-output linearization," *Control Engineering Practice*, vol. 7, no. 3, pp. 369–373, 1999.

[6] G. Oriolo, A. de Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: design, implementation, and experimental validation," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 6, pp. 835–852, 2002.

[7] J. E. Normey-Rico, I. Alcalá, J. Gómez-Ortega, and E. F. Camacho, "Mobile robot path tracking using a robust PID controller," *Control Engineering Practice*, vol. 9, no. 11, pp. 1209–1214, 2001.

[8] K. D. Do and J. Pan, "Global output-feedback path tracking of unicycle-type mobile robots," *Robotics and Computer-Integrated Manufacturing*, vol. 22, no. 2, pp. 166–179, 2006.

[9] H. Shibly, K. Iagnemma, and S. Dubowsky, "An equivalent soil mechanics formulation for rigid wheels in deformable terrain, with application to planetary exploration rovers," *Journal of Terramechanics*, vol. 42, no. 1, pp. 1–13, 2005.

[10] K. Yoshida, T. Watanabe, N. Mizuno, and G. Ishigami, "Slip traction control, and navigation of a lunar rover," in *Proceedings of the 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space (i-SAIRAS '03)*, pp. 3566–3578, Nara, Japan, 2003.

[11] E. T. Baumgartner, H. Aghazarian, and A. Trebi-Ollennu, "Rover localization results for the FIDO rover," 2001, http://www.researchgate.net/publication/252911108_Overview_of_MEMSNEMS_technology_development_for_space_at_NASAJPL.

[12] L. Ding, Z. Deng, H. Gao, K. Nagatani, and K. Yoshida, "Planetary rovers' wheel-soil interaction mechanics: new challenges

and applications for wheeled mobile robots," *Intelligent Service Robotics*, vol. 4, no. 1, pp. 17–38, 2011.

[13] H. Gao, Z. Deng, L. Ding, and M. Wang, "Virtual simulation system with path-following control for lunar rovers moving on rough terrain," *Chinese Journal of Mechanical Engineering*, vol. 25, no. 1, pp. 38–46, 2012.

[14] L. Ding, H. Gao, Z. Deng, and J. Tao, "Wheel slip-sinkage and its prediction model of lunar rover," *Journal of Central South University of Technology*, vol. 17, no. 1, pp. 129–135, 2010.

[15] J. Y. Wong, *Theory of Ground Vehicles*, Wiley, New York, NY, USA, 4th edition, 2008.

[16] D. Wang and C. B. Low, "Modeling and analysis of skidding and slipping in wheeled mobile robots: control design perspective," *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 676–687, 2008.

[17] C. B. Low and D. Wang, "GPS-based path following control for a car-like wheeled mobile robot with skidding and slipping," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 340–347, 2008.

[18] M. Michałek, P. Dutkiewicz, M. Kiełczewski, and D. Pazderski, "Trajectory tracking for a mobile robot with skid-slip compensation in the vector-field-orientation control system," *International Journal of Applied Mathematics and Computer Science*, vol. 19, no. 4, pp. 547–559, 2009.

[19] K. Gu and M. Zhao, "Dynamic modeling and simulation of driving control for wheeled mobile robot on sand," *Journal of System Simulation*, vol. 20, no. 18, pp. 5035–5039, 2008.

[20] K. R. Xia, L. Ding, H. B. Gao, and Z. Q. Deng, "Motion-control-based analytical model for wheel-soil interaction mechanics of lunar rover," in *Proceedings of the 6th International Forum on Strategic Technology (IFOST '11)*, pp. 333–338, Harbin, China, August 2011.

[21] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, New York, NY, USA, 3rd edition, 2005.

[22] Y. Lu and J. K. Liu, "Actuator nonlinearities compensation using RBF neural networks in robot control system," in *Proceedings of the IMACS Multiconference on Computational Engineering in Systems Applications*, vol. 1, pp. 231–238, Beijing, China, October 2006.

[23] H. Gao, W. Li, L. Ding, Z. Deng, and Z. Liu, "A method for on-line soil parameters modification to planetary rover simulation," *Journal of Terramechanics*, vol. 49, no. 6, pp. 325–339, 2012.