Flow-Capacity-Maintaining, Decoupled Conflict Resolution Procedures for Air Traffic Control

Jeff D. Yoo

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2014

Reading Committee:

Santosh Devasia, Chair

Brian C. Fabien

Joseph Garbini

Aslaug Haraldsdottir

Program Authorized to Offer Degree:
Mechanical Engineering

University of Washington

Abstract

Flow-Capacity-Maintaining, Decoupled Conflict Resolution Procedures for Air Traffic Control

Jeff D. Yoo

Chair of the Supervisory Committee:
Professor Santosh Devasia
Mechanical Engineering

This research addresses the design of automated provably-safe en-route Conflict Resolution Procedures (CRP) for Air Traffic Control (ATC). The main issue in ATC today is to deal with the increasing workload on air traffic controllers. Automation can be widely used to reduce controller workload for controllers and to improve performance, e.g, to avoid capacity loss causing delays due to adverse weather. However, proving safety of automated CRPs has been challenging due to substantial computational effort and complexity with large number of aircraft and conflicts in the ATC system. Towards a provably-safe CRP, this thesis develops automated CRPs that satisfy conditions for guaranteed solution. In particular, this thesis addresses three issues: (i) account for aircraft turn dynamics, (ii) resolve conflicts on non-perpendicularly intersecting routes with different speeds, and (iii) account for on-demand activation and deactivation of the CRP.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to University of Washington, and the Department of Mechanical Engineering where he has had the opportunity to work with great faculty and students with numerous assitants on research. My sincere appreciation and deep gratitude to my advisor, Professor Santosh Devasia, for his dedication, guidance, and support throughout my graduate studies, help from Dr. D. Iamratanakul with data extraction for simulation, and the Boeing Company for tuition assitance, and encouragement to pursue my degree

.

DEDICATION

to my dear wife, Soojin, my daughter, Hannah, and all my friends and family

Chapter 1
INTRODUCTION

This research addresses the design of automated provably-safe en-route Conflict Resolution Procedures (CRP) for Air Traffic Control (ATC). The main issue in ATC today is to deal with the increasing workload on air traffic controllers. Automation can be widely used to reduce controller workload [1, 2] for controllers and to improve performance, e.g, to avoid capacity loss causing delays due to adverse weather [3, 4]. However, proving guaranteed solution of automated CRPs has been challenging due to substantial computational effort and complexity with large number of aircraft and conflicts in the ATC system [5]. Towards a provably-safe CRP, this thesis develops automated CRPs that satisfy conditions for guaranteed solution. In particular, this thesis addresses three issues: (i) account for aircraft turn dynamics, (ii) resolve conflicts on non-perpendicularly intersecting routes with different speeds, and (iii) account for on-demand activation and deactivation of the CRP.

This chapter starts with a motivational severe-weather example, followed by an overview of ATC, a description of the challenges in ATC, and the main research problems.

1.1 Motivational Example - Capacity Loss Causing Delays

Consider the rerouting of aircraft trajectories (routes) to avoid adverse weather conditions in some region of the National Airspace (NAS). Standard reroutes to accommodate severe weather are done by merging affected flows, and then routing them around the flight restricted area [4]. To illustrate, consider the West Watertown procedure from [4] in Figure 1.1 where the routes from Helena, Sacramento, and Bryce

Canyon to Boston, La Guardia, and Dulles are affected by the occurrence of adverse weather in the middle of these three routes as in Figure 1.2. Then the West Watertown procedure merges the affected routes, and reroutes them around the severe weather zone as shown in Figure 1.2. This merging type method is based on simplicity and

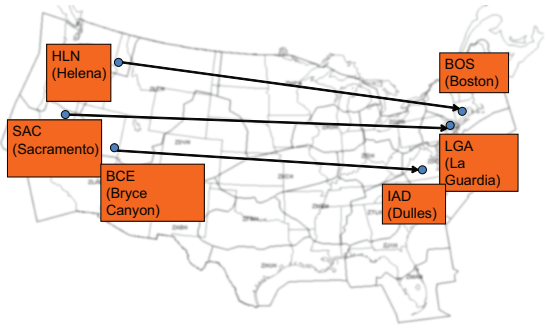

Figure 1.1: West Watertown example (adapted from [4]). Three routes (Helena to Boston, Sacramento to La Guardia, Bryce Canyon to Dulles) from west to east in the U.S. are shown.

ease of adoption by the human controllers [4, 6]. However, merged reroutes result in loss of capacity. To illustrate, each route, prior to merging can have the maximum capacity, where aircraft are separated by the lateral minimum required spacing D_{sep} (where $D_{sep} = 5$ nautical miles from [7]) to avoid conflict as shown in Figure 1.3. The capacity of the single merged route (between point A and point B in Figure 1.2) is also limited by this same maximum flow capacity of a single route as shown in Figure 1.3. Therefore, merging cannot happen with aircraft at maximum capacity in each route before the merge. As a result, the routes prior to merge need to reduce the pre-merge capacity to less than one third of maximum capacity for this example (assuming all flows have constant spacing) since three routes are merging into one route as shown in Figure 1.4 leading to capacity loss. To accommodate the pre-merge loss in capacity, holds or delays have to be placed on aircraft departing from airports. Thus, playbooks with merges are simple, but lead to loss in capacity and delays.

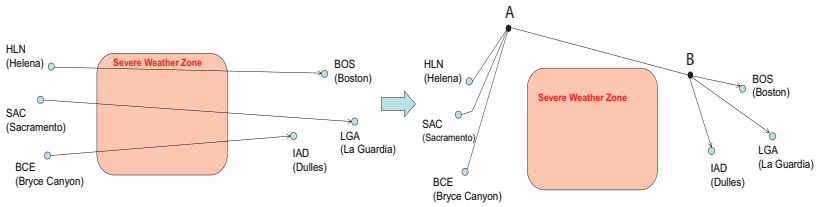

Figure 1.2: From example in Figure 1.1 severe weather occurs (left) reroutes are done by merging affected routes and reroute around the severe weather zone (right).

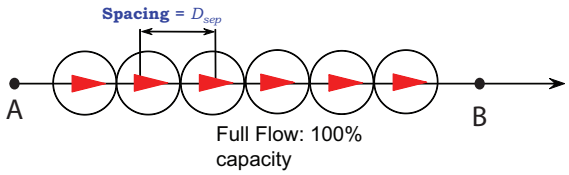

Figure 1.3: The full capacity of the merged route between Point A and Point B from Figure 1.2. All aircraft are separated by the minimum spacing requirement distance D_{sep} within a flow at full capacity. The circles around all aircraft represent the lateral protection zone of diameter of D_{sep}. If an aircraft's the protection zone overlaps with another aircraft's protection zone, the two aircraft have conflict.

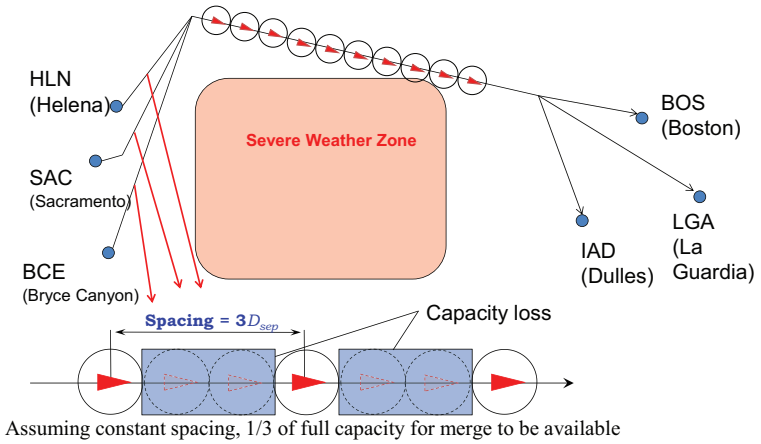

Figure 1.4: West Watertown example of capacity loss due to merging; Each route prior to merging originally with full capacity need to reduce to less than1/3 of full capacity due to reroute merging.

In order to prevent delays due to capacity loss, an alternative, rerouting around an adverse weather area without merging (as shown in Figure 1.5) is preferred. The rerouting method without merges would enable each route to maintain its original capacity but can lead to more complex ATC. Consider the case where there is another flow that intersects with the three routes that need to reroute as shown in Figure 1.6(a) and (b). Due to rerouting, the distance between the intersections are closer compared to the original intersections before the reroute. Close intersections can fall within the domain of a single controller, which implies an increase in workload (to resolve the accompanying conflicts) for the controller.

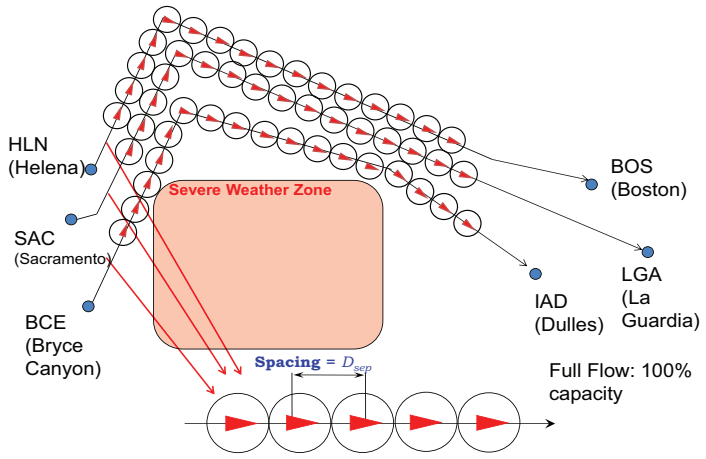

Figure 1.5: Non-merging reroute alternative that does not result in capacity loss because each route does not merge with other routes into a single flow, and could keep its original capacity.

1.2 An Overview of Air Traffic Control

An overview of air traffic control is presented in this section to place the thesis in context with other efforts in ATC and how it can help in future applications.

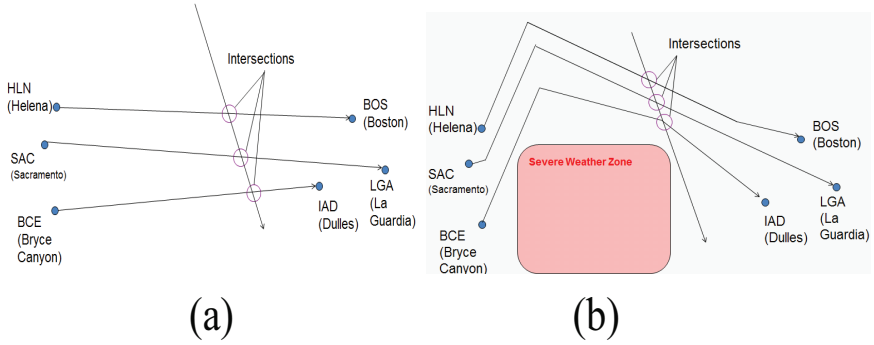

Figure 1.6: A route intersects with the three routes from Figure 1.1. (a) When the three routes do not reroute due to weather. (b) When the three routes reroute due to weather. The intersections are closer when rerouting around severe weather.

1.2.1 Overview of current ATC roles by phase of flight

To begin, the overall human resources during a flight cycle of an airplane are shown to demonstrate how many air traffic controllers are generally involved for an airplane flight from departure to destination.

Preflight and Airport ATC for aircraft at the airport terminal is managed by the Airport Traffic Control Tower (ATCT) [8, 9]. Controllers involved are the ground controller, the local controller, and the flight data controller. The pilot is required to file a flight plan to the ATCT at least 30 minutes prior to pushing back from the gate [9]. Once the flight plan is received, the flight data controller reviews the flight plan, the weather conditions, and enters the information to the FAA database. The ground controller provides clearance of pushing back from the gate, and then issues taxi instructions to the runway for takeoff. Once the aircraft is at the runway, the local controller issues takeoff clearance, and maintains separation between departure aircraft. The local controller also provides the latest weather and field conditions.

Terminal Departure The Terminal Radar Approach Control (TRACON) takes over after take off [8]. TRACON is responsible for near terminal area aircraft (approximately 30 - 50 miles radius from the terminal [9]). During near terminal depar-

ture, the departure controller from TRACON, assigns heading directions and altitude to the aircraft. Additionally, the departure controller maintains safe distance between ascending aircraft. Once the aircraft reaches the en-route phase, the control is transferred to the Air Route Traffic Control Center (ARTCC).

En-route The ARTCC manages aircraft in the en-route phase with long range radars beyond the TRACON coverage areas (sectors) [8, 10–13]. The three main controllers in the ARTCC are the radar controller, the radar associate controller, and the radar associate flight data controller [8, 10–13]. For each sector, the radar controller ensures safe separation between aircraft and coordinates with other sector centers. The radar controller monitors the aircraft until it leaves the corresponding sector, and then transfers control to the next sector. The radar associate controller assists the radar controller, and the radar associate flight data controller assists both the radar controller and the radar associate controller by reviewing the flight plan and also monitoring the radar for heavy traffic.

Terminal Arrival As mentioned for the terminal departure phase, near terminal control is done by TRACON. Terminal approaching and descending aircraft are managed by the arrival controller. The arrival controller assigns heading and altitude for final arriving aircraft and maintains safe separation for descending aircraft [8, 10–13].

Arrivals and Airport Air traffic is managed by ATCT for aircraft that have landed. The local controller provides landing clearance and provides updated weather and field conditions. Also, the local controller maintains arrival separation. After landing on the runway, the ground controller provides safe taxi guidance and clearance to pull up to the gate [8, 10–13].

Overall, the number of controllers involved for an aircraft flight cycle is seven from terminal and near-terminal control, and three controllers per en-route zones the aircraft travels, and additional controllers from the command center that overviews the whole flight cycle as shown in Figure 1.7. Thus, current ATC is human controller centric.

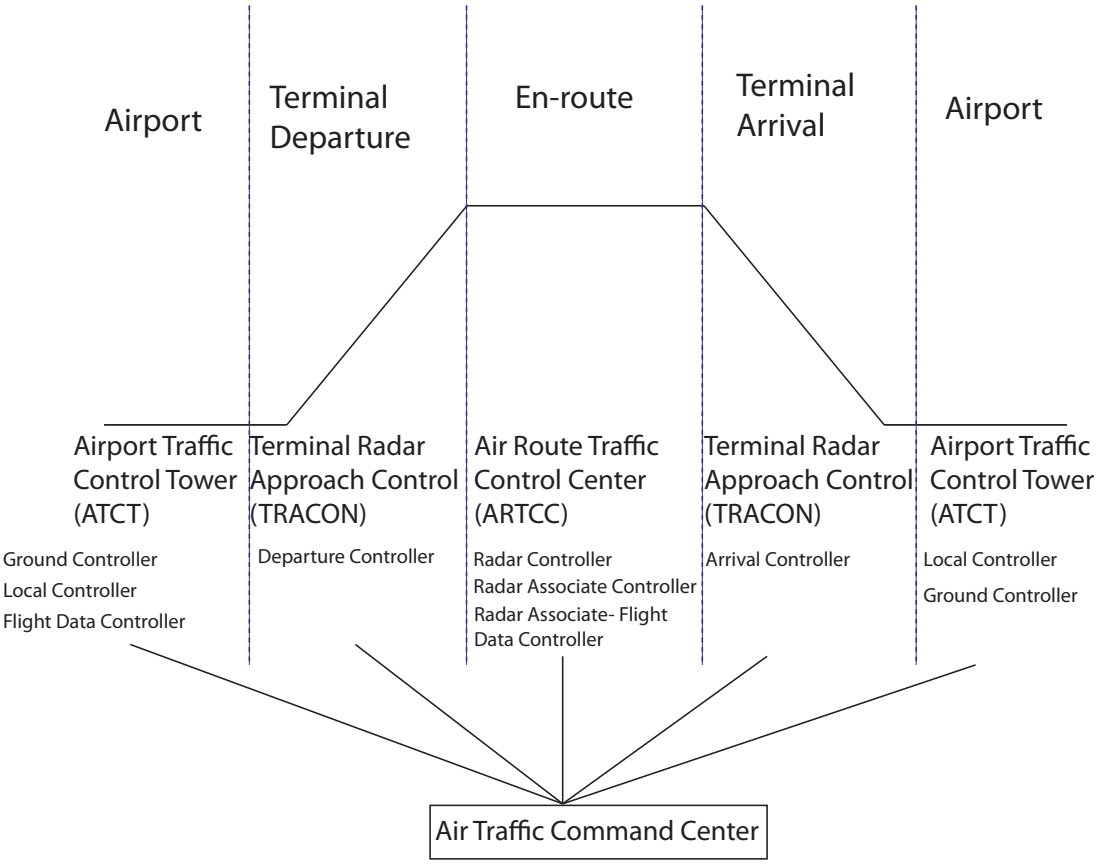

Figure 1.7: Air traffic controllers involved in an aircraft's flight cycle per [8, 10–13].

1.2.2 The trends in current ATC and the need for Automation

Staffing in ATC today The FAA staffing for human resources are based on "staffing to traffic" [8]. The demand for flights have increased since the 1960s as shown in Figure 1.8 from the statistical data [14–16]. Therefore, the controller workforce has been increasing, and the controller headcount for 2009-2012 was around 15,000 [8, 10–13]. However, the turnover rate due to retirements (with the majority retirement age being 56 which is less compared to most industries), resignations, removals, promotions, transfers, and developmental or academy attrition is about 9 percent (1,400 loss over

15,000 total [8]). The turnover rate for ATC workforce is higher than the total industry turnover rate, which is about 3.1 percent for 2012 [17]. Part of this turnover is due to high workload and the associated stress. Additionally, increasing traffic results in continued reduction in sector size, which increases the number of flight controllers in ATC which is unsustainable [18]. Overall, automation is needed to reduce workload for ATC (to accommodate for increasing demands), and to reduce the relatively high turnover rate for air traffic control.

Technology Development Lack of sufficient technology development relative to the growth in traffic also contributes to increase of workload. The following shows the chronological order of ATC technology development associated to workload. The first powered flight started with the Wright brothers in 1903. Until the early 1920s, the skies had so few aircraft that there was no rules and regulations from the government [19]. Simple rules applied for aircraft safety were the rules of the road (i.e., aircraft keep to the right as they fly), which were executed by pilots [20]. For en-route ATC, early methods include using bonfires for night vision to guide pilots [19, 21]. Aircraft were mostly operating under good weather conditions and flew visually and pilots fended for themselves by flying under a seen or be-seen basis [20, 22]. For better safety, as radio and telephone communications developed, more controllers were involved from airports and airlines to keep track of the position of planes moving along the airways with the help of maps and blackboards [19, 20, 22]. Eventually, beacons were developed for ATC, where a network of radio beacons were used to guide through low-frequency radio ranges laid out to connect principal cities in the United States [20, 21]. Further safety rules and regulations were implemented and tracked by air traffic controllers with Air Route Traffic Control Centers (ARTCC) and control towers with recognition of safety of air travel by the federal government with increasing demand [19, 20, 22–25].

A major milestone in development in ATC occurred with the development of radars with the ability to track aircraft more accurately and provide speed and altitude data.

In addition, with advanced communication technology, and advanced aircraft, more sophisticated control was evolved with a control center and direct-to-pilot communication [22]. This evolution milestone allowed better situational awareness between air traffic controllers and for each aircraft. Additionally, this allowed growth in the numbers for air travel and the resulting increase in workload for air traffic controllers [1].

An additional development milestone is the use of the GPS (Global Positioning Satellite System) and the development of NextGen (Next Generation Air Transportation System) [26]. The GPS system allows better precision in tracking aircraft. With better precision, more optimal flight paths (where multiple flight paths can form a network that are closer) can be developed [26]. Additionally, airliners today would prefer using these optimal paths for better flight efficiency due to increase in fuel costs. This provides more workload to controllers due to substantial computational efforts, and also for monitoring closer, and often changing optimal flight paths. Also, innovative concepts such as free-flight, where the operator has the freedom to select aircraft path and speed in real time [27–29] can lead to increased work load.

Future applications Future applications include the integration of unmanned aerial vehicles (UAVs) in civilian airspace [30, 31], e.g., if UAV routes need to cut across established routes for commercial aircraft. The increase in use of the NAS (National Airspace System) involving applications such as UAVs can also increase controller workload. Overall, the increase in traffic with the current human resources, the development of new technology, and future application increase workload. Automation could help to reduce this workload.

1.2.3 Current State of the Art in Automation in ATC

The majority of today's ATC automation is applied for near terminal-area traffic. Current efforts in automation are in efficient departure and landing at the runways with the goal of reducing delays due to the increasing traffic [1, 32–37]. For example, at Dallas/Fort Worth International airport, the FAA has applied the NextGen

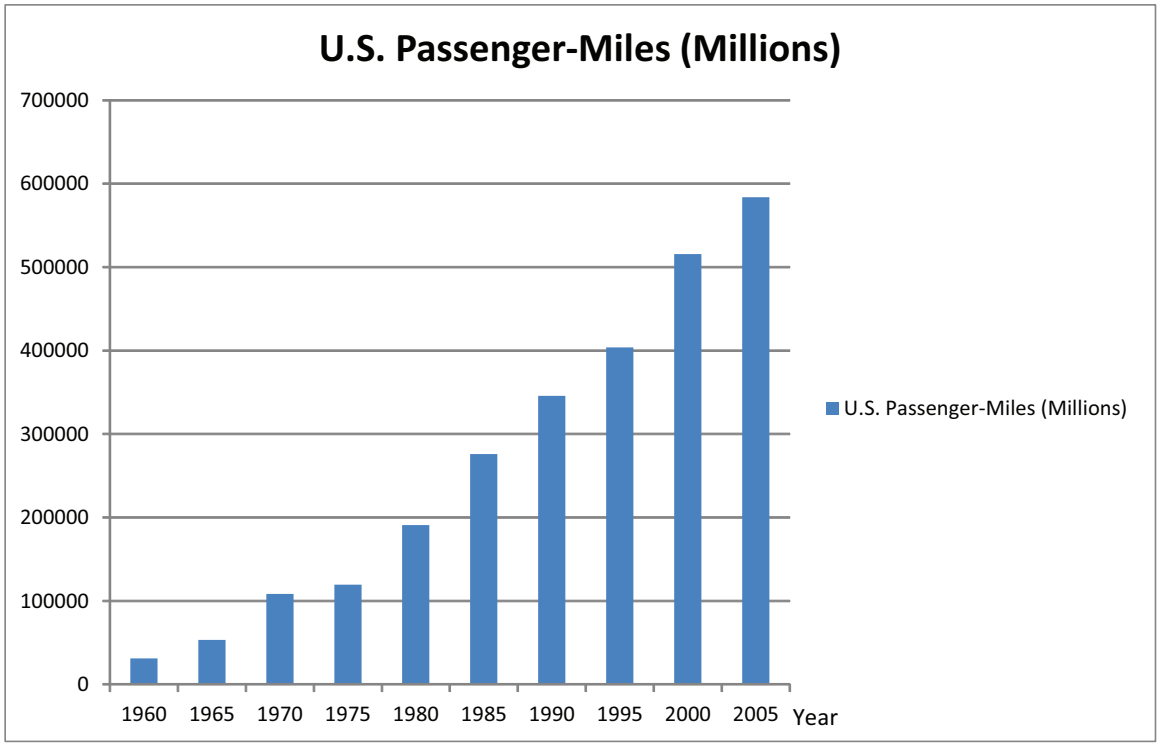

Figure 1.8: The increase in air travel passengers as time passes per [14–16].

procedure, which allows less separation distance between departing aircraft compared to the conventional procedures [38]. Thus, increasing the number of departures, and saving costs by reducing delays.

1.2.4 Near-term Trends for En-route Automation in ATC

Like the current state where automation is applied at the terminal area due to increase of traffic, the en-route region will also need to deal with the increasing workload due to the increasing traffic. With more increase in demand for air travel, not only will numbers in aircraft increase near terminal area, but also on en-route areas as well. Therefore, automation will be needed for en-route air traffic control. This en-route ATC automation (CRP design) is the focus of the current thesis.

1.3 Challenges in En-route ATC Automation (CRP Design)

There are three main challenges in the design of automated en-route CRP: (i) avoiding the *domino* effect [39], (ii) guaranteeing a solution for distributed CRP, and (iii) guaranteeing stability.

First, the *domino* effect is where a resolution of a conflict leads to a new conflict, whose resolution leads to another conflict, and so on [39]. To illustrate, consider an example of the *domino* effect as shown in Figure 1.9. Consider the case as in Figure 1.9(a) where aircraft $R1$ and $B1$ has conflict at the intersection point P_1. Then the conflict between aircraft $R1$ and $B1$ is resolved by laterally shifting aircraft $R1$ as shown in Figure 1.9(b). However, consider the case as in Figure 1.9(c) where aircraft $B2$ is located behind aircraft $B1$. Here, due to the lateral shift operation of aircraft $R1$, a new conflict occurs between aircraft $R1$ and $B2$ at intersection point P_2. In order to resolve the conflict between aircraft $R1$ and $B2$, aircraft $B2$ is laterally shifted as in Figure 1.9(d). Finally, if aircraft $R2$ is located behind aircraft $R1$ as in Figure 1.9(e), due to the lateral shift operation of aircraft $B2$, a new conflict occurs between aircraft $R2$ and $B2$ at intersection point P_3. As shown in this example, the resolution of one conflict results in causing a new conflict, which resolution causes a new conflict and so on, causing the *domino* effect, which is a challenge in ATC that needs to be avoided by the automated CRP design.

Next, conflict resolution procedures (CRPs) need to be distributed (spatially-and-temporally) because of the substantial increase in computational and modeling complexity with a centralized CRP when the number of aircraft (and conflicts) are large, e.g., over the entire U.S. airspace. Additionally, centralized controllers become inefficient, over a large region, because the need to handle large uncertainty in the aircraft trajectories [40, 41]. To illustrate, consider an example route in Figure 1.10

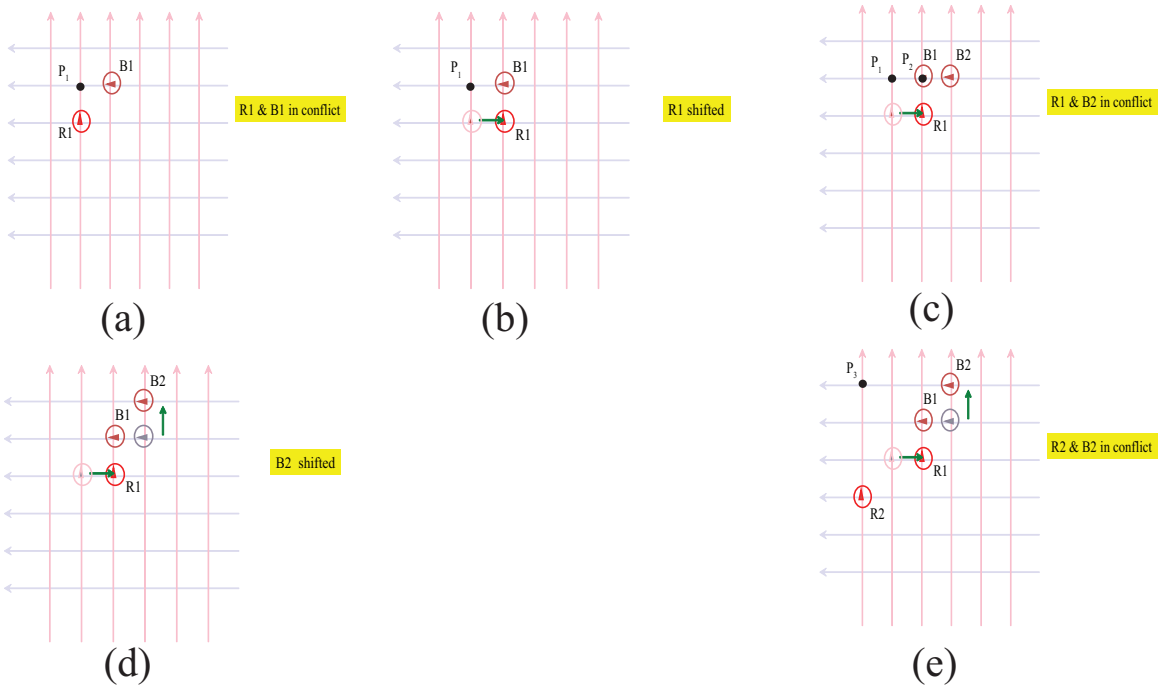

Figure 1.9: Example of the *domino* effect in conflict resolution where one's resolution (using lateral shift operations) causes a new series of conflict. (a) Aircraft $R1$ and $B1$ in conflict at intersection P_1. (b) Conflict is resolved between aircraft $R1$ and $B1$ from (a) by laterally shifting aircraft $R1$. (c) Aircraft $R1$ and $B2$ are in conflict if aircraft $B2$ is located in a course for conflict where the two aircraft route intersects at point P_2. (d) Conflict is resolved between $R1$ and $B2$ from (c) by laterally shifting aircraft $B2$. (e) Aircraft $R2$ and $B2$ are in conflict if aircraft $R2$ is located in a course for conflict where the two aircraft route intersects at point P_3.

where aircraft $a1$ crosses the U.S. from east-to-west. Typically, an aircraft takes 4-5 hours to cross the U.S. from east to west. However, unexpected severe weather can develop in 2-3 hours, and last for long durations of time [42] that affect aircraft routes. Consider a severe weather occurring in the middle of the east-to-west route from Figure 1.10 as in Figure 1.11(a). Then, the east-to-west route needs to be rerouted around the severe weather, e.g., as in Figure 1.11(b). The reroute delays aircraft $a1$ and alters the potential of conflicts aircraft such as aircraft $b1$ on an intersecting route

due to uncertainty as of when aircraft $a1$ arrive at the intersection point o (where the position of aircraft $a1$ is uncertain and a larger area is considered for conflict resolution as in Figure 1.12). Therefore, the CRP needs to be locally decoupled spatially and temporarily in order to reduce uncertainties and improve efficiency. The challenge then is to have distributed CRPs provide conflict resolution to the global system.

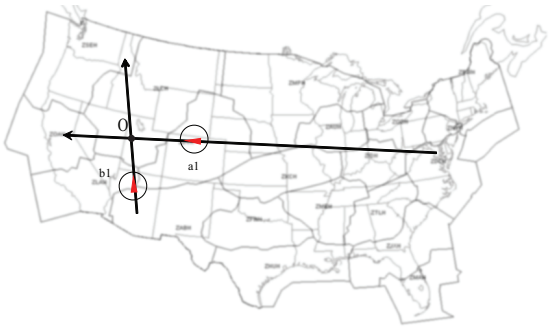

Figure 1.10: Example intersecting routes crossing the U.S.

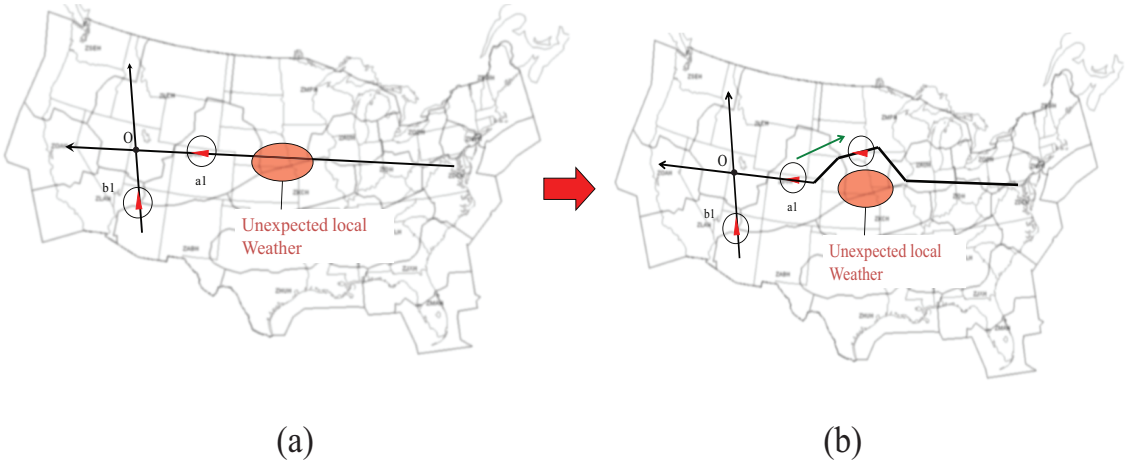

Figure 1.11: (a) Unexpected severe weather occurring in the east-to-west route from example in Figure 1.10. (b) The severe weather forces to reroute aircraft $a1$ around the severe weather. Uncertainty of arrival at the intersection increases with distance from the intersection, e.g., due to potential reroutes.

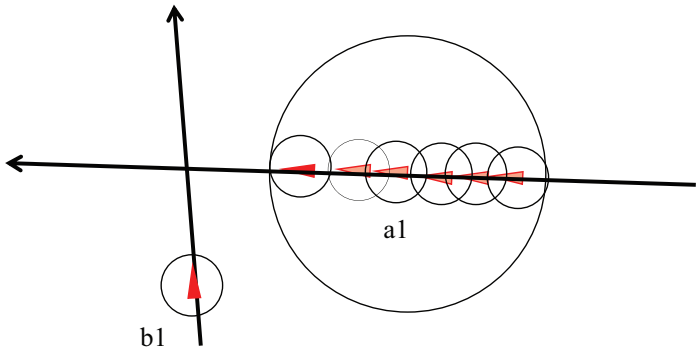

Figure 1.12: Standard rerouting around severe weather delays aircraft $a1$ and increases uncertainty (circular area) of aircraft position of $a1$. The increase in uncertainty of position is inefficient for conflict resolution because of the need to handle larger set of possible conflicts.

The third challenge is to guarantee stability of CRPs. A CRP is defined to be stable if the applied CRP has the maximum deviation (from the original route) be a finite bounded distance [43–45]. The problem of guaranteed conflict resolution in a stable manner remains challenging for automated CRPs. An example is provided to illustrate when a CRP is not stable. In this example, lateral shifting operations are performed for resolving conflicts as in Figure 1.13.

The example for when three routes intersect are shown in Figure1.14(a). Each aircraft from one route has conflict with aircraft of their intersecting routes. When applying the lateral shift operations for conflict resolution (as in Figure 1.13), aircraft arriving later (aft) require more lateral shifting distance (d_{dev}) as in Figure 1.14(b). Thus, aircraft deviation distance from the original route increases more for later arriving aircraft such that the maximum deviation is unbounded. Therefore stability is not automatic for a given conflict resolution procedure and needs to be established.

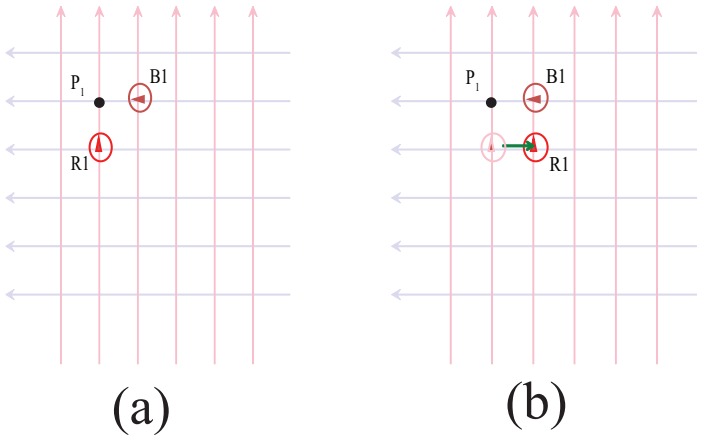

Figure 1.13: Conflict resolution by lateral shifting operations. (a) Aircraft $R1$ and $B1$ have conflict at intersection point P_1. (b) Conflict is resolved by laterally shifting aircraft $R1$.

1.4 How to address the challenges

The CRP design concept proposed in this thesis, shown in Figure 1.15, can address the challenges of avoiding the *domino* effect, the need of decoupled CRP, and guaranteed stability if conflicts (route intersections) are finite, and sufficiently sparse (sufficiently separated from each other) and has the following properties:

1. The local CRP is bounded in space and time.

2. The aircraft's arrival and exiting sequence are the same as they go through the local resolution zone.

3. When aircraft exit the local CRP zone, aircraft would have returned back to the original destined route and separated by the minimum required distance.

4. All aircraft entering the local CRP would have traveled an equal resolution path

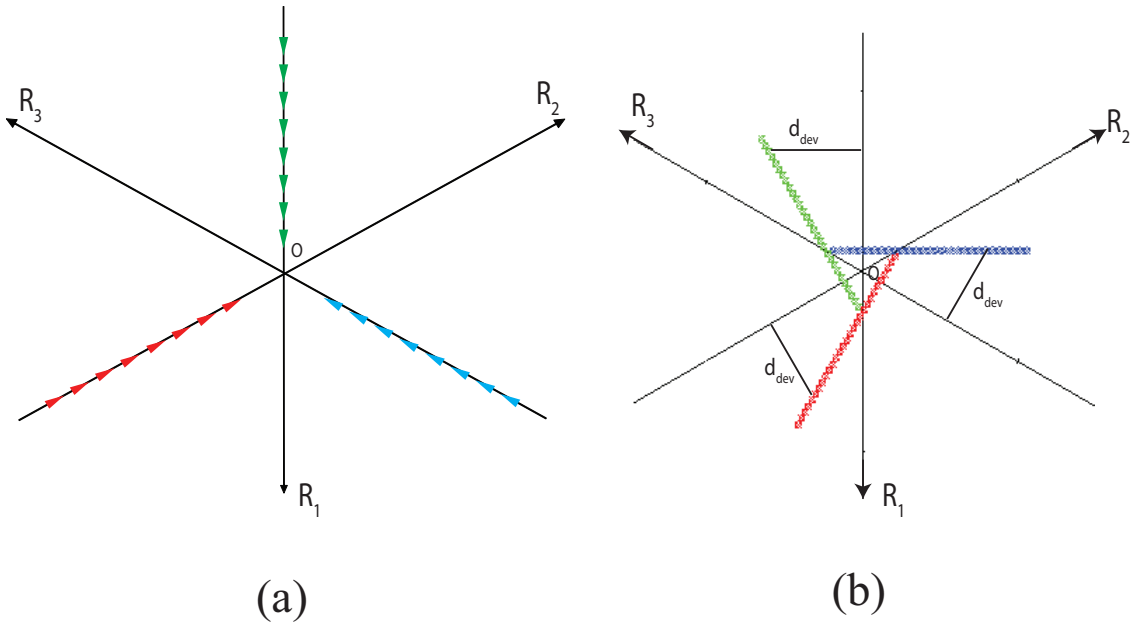

Figure 1.14: Stability example adapted from [43]. (a) Three routes with aircraft intersect at point O. (b) Lateral shifting operations are performed for conflict resolution. Later arriving aircraft has increasing resolution deviation distance (d_{dev}) and therefore it is not stable.

length and equal deviation distance from the nominal route.

The following shows how the properties of the proposed CRP in this thesis overcome the challenging issues in en-route CRP.

1) How the proposed CRP design concept solves the *domino* effect?

For the given airspace as shown in Figure 1.15, all conflict resolution is performed within local CRP regions. In addition, all aircraft that have passed through a local CRP zone would have returned back to its original destined flow. So, solving one conflict would not lead to a new series of other conflicts, and therefore, there would be no *domino* effects (provided the local CRP areas are disjoint and sufficiently sparse such that each local CRP intersection would be independent from each other).

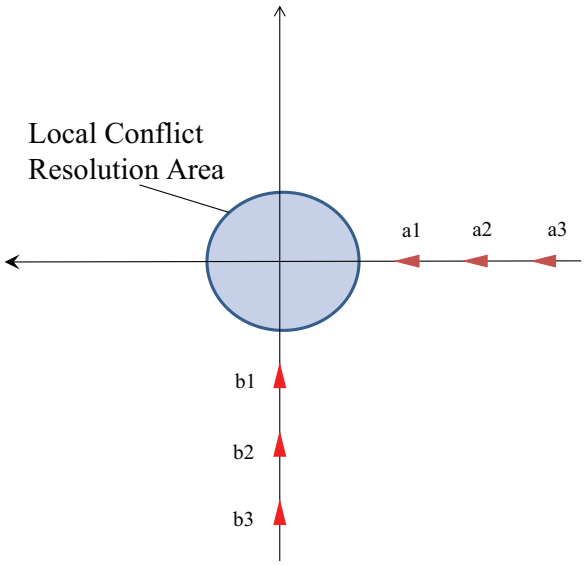

Figure 1.15: Local conflict resolution zone bounded around an intersection.

2) How the proposed CRP design concept is distributed?

When aircraft in a route pass through their first local CRP, the aircraft are back on their original route. Additionally, aircraft are in the same sequence before they have entered the first local CRP (provided that aircraft spacing is synchronized inside the local conflict resolution zone as in Figure 1.15). Therefore, the first local CRP does not affect the next CRP (as in Figure 1.15). Thus, the CRP is local in space and time resulting in decoupled CRPs.

3) How the proposed CRP design concept guarantees stability?

If each local CRP (which is bounded in space and time) only has a finite number of conflicts to resolve and there are no new conflicts, the local CRP is stable. Since all aircraft entering the local CRP would have traveled on the same resolution path with equal deviation distance from the nominal route, the CRP is stable locally. As a result, with each local CRP being stable, the global CRP would also be stable, provided that there is a finite numbers of intersections.

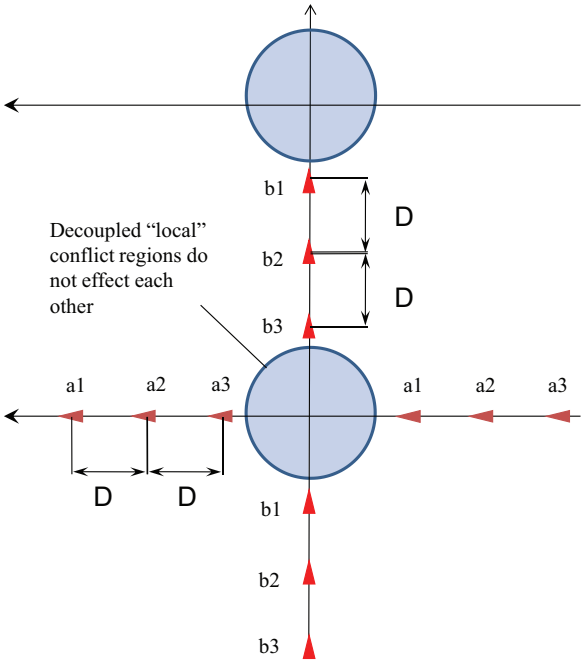

Figure 1.16: Locally decoupled CRP properties; The arrival and exiting sequence of aircraft are kept and aircraft passing through the local CRP would separate aircraft by at least the minimum distance D, thus not affecting the next CRPs.

1.5 The proposed local CRP

Given a case such as in Figure 1.17, with two intersecting flows (one with heavy traffic with less separation distance between aircraft), it would be difficult for the aircraft on the intersecting flow to pass the intersection without having conflicts due to lack of space. Therefore, sufficient spacing is needed from the intersecting route with heavy traffic (south-to-north route in Figure 1.17). Additionally, timing of aircraft traveling from each route is important. With precise timing, where an aircraft can be centered at the intersection and in between aircraft on the intersecting flow, and with sufficient spacing (as in Figure 1.18), there would be no conflict. However, if spacing is not sufficient, then route spacing is created by aircraft traveling an alternating split pattern on routes as shown in Figure 1.19. Timing at the intersections created by the

split is also kept where aircraft are centered and in between the intersection route. By applying this split method, the separation distance between aircraft on the split route would have increased by the number of splits applied when intersecting with the other route. The general local CRP path is shown in Figure 1.20, with splits applied to all incoming flows to the local conflict regions (intersections).

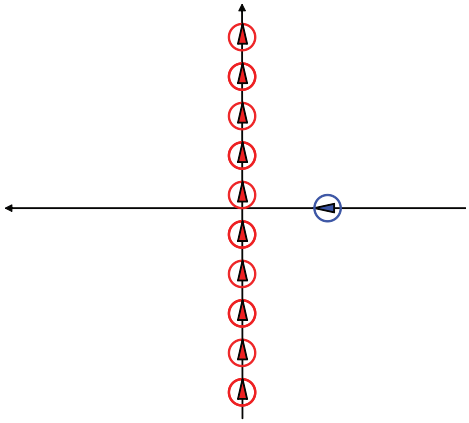

Figure 1.17: Two intersecting flows with heavy traffic. There is not enough space for aircraft to intersect without conflicts.

1.6 Research Problems

The main contribution of this research is to address the following three topics in the design of automated en-route decoupled CRP.

- Accounting for aircraft turn dynamics.
- CRP for non-perpendicular and compound intersections.
- On-demand activation and deactivation of CRPs.

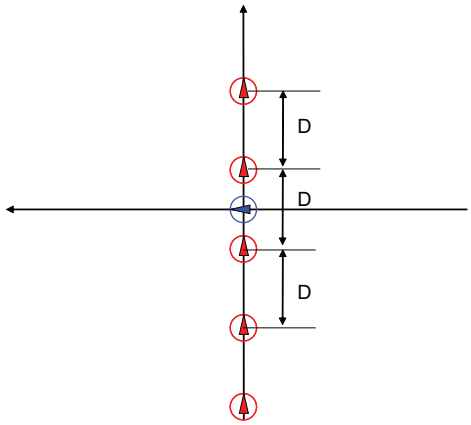

Figure 1.18: Safe travel through intersection without conflict. With correct timing where an aircraft is centered and in between aircraft spacing of the intersecting route, and with sufficient spacing, aircraft travel at the intersection is safe.

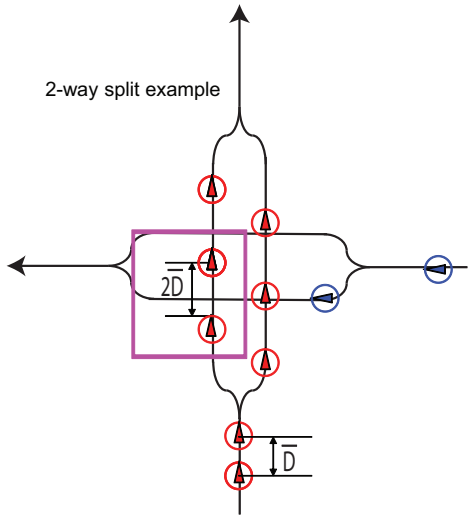

Figure 1.19: Local CRP with split paths; aircraft separation on the split path increases by the number of splits applied from the original path.

1.6.1 Accounting for aircraft turn dynamics

This work shows the importance of the need in having aircraft turn dynamics with bounded heading turn rates while aircraft perform a turn on a turn route of the CRP. Previous work [3, 46] dealt with CRPs with turn path designs that do not

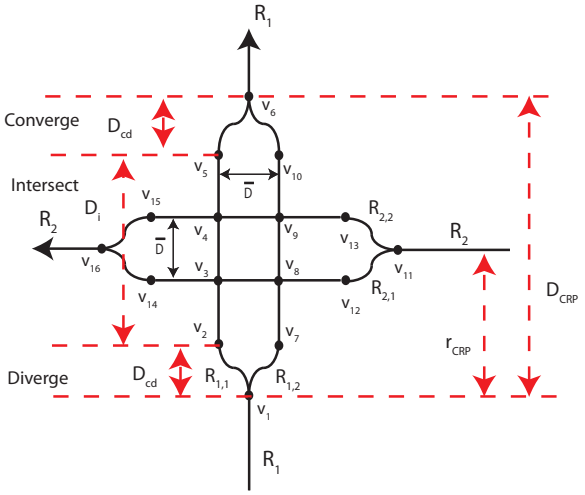

Figure 1.20: Local CRP with a 2-way split on each incoming route.

take in account of realistic aircraft turn maneuvers; it was assumed that once an aircraft reaches its turn route, the aircraft's heading angle would instantaneously change towards the turn route meaning that the heading change rate for turns are not bounded which are not realistic. This research shows that compared to previous work with instantaneous turns [3,46], applied rate-limited turn dynamics changes the conflict avoidance requirements with respect to aircraft arrival spacing. In addition, robustness issues such as uncertainties affecting an aircraft position while traveling on the CRP path is studied.

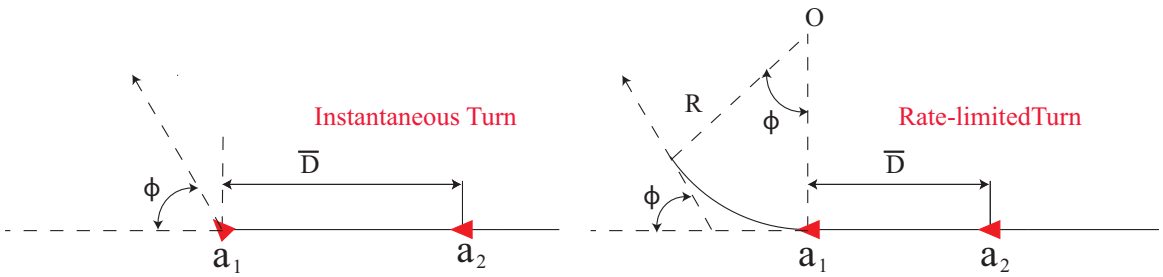

Figure 1.21: Aircraft performing instant heading changes (left) and bounded rate turns (right).

1.6.2 CRP for non-perpendicular and compound intersections

This work removes two limitations of the CRP design by: (i) developing procedures for non-perpendicular intersections, and (ii) removing the requirement that aircraft on each intersecting route have the same speed. Conditions are developed to guarantee conflict resolution, and the CRP is illustrated through an example en-route intersection. This is important since the safety conditions changes from previous work [3, 47] where the conflict conditions are directly related to the intersection orientation angle and route speed ratio. Additionally, with this design of the CRP, additional rerouting efforts can be eliminated since there is no need to force the intersecting routes to meet perpendicularly to apply the CRP design from previous work [3, 47]. In addition, different route speeds used in the design was considered such that aircraft speed affected by wind or other effects can be observed. Additionally, this CRP design is applied to compound intersections to demonstrate the decoupled distributed CRP guarantees solution as in the example shown in Figure 1.22.

1.6.3 On-demand activation and deactivation of CRPs

Previously, the provably-safe CRP design was always active, even in the absence of conflicts at the intersection - which would be acceptable if conflicts occur frequently at the intersection. Therefore, the always-on CRP leads to unwanted CRP maneuvers (when there are no conflicts at the intersection) resulting in increased travel time, travel distance, and required fuel, and is not efficient for intersections where conflicts are rare, e.g., because one of the routes only has intermittent traffic. This work removes the inefficiency of always-on CRPs by developing provably-safe CRPs that can be activated on-demand (when conflicts appear) to accommodate an impending conflicts, and the CRP is then deactivated when not needed as shown in Figure 1.23.

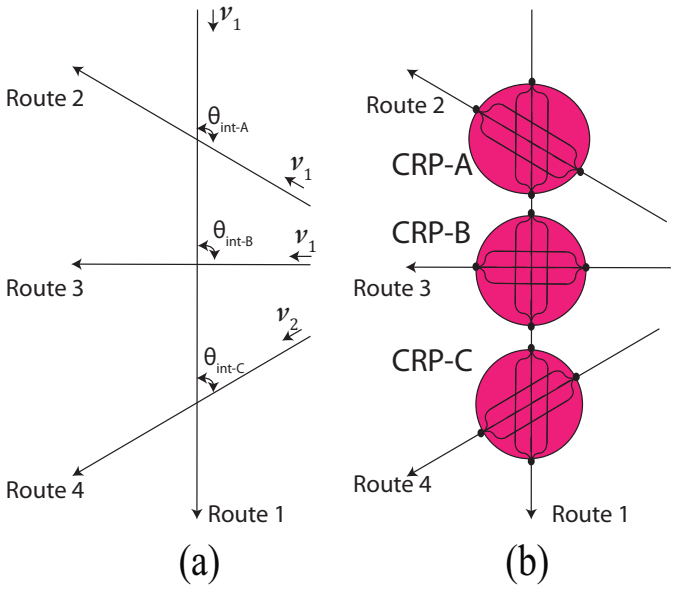

Figure 1.22: (a) Example of multiple intersections with perpendicular and non-perpendicular orientation and same or different route speeds. (b) CRPs applied at each intersection to illustrate the decoupled property.

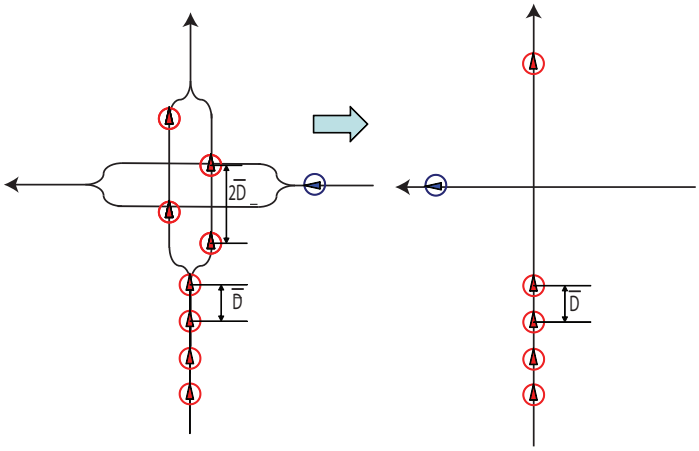

Figure 1.23: Example case of On-demand activation and deactivation of the CRP on two intersecting routes; The CRP is activated (left) for impending conflict and deactivated (right) as it is safe with no incoming intersecting flow aircraft.

1.7 Structure of the Dissertation

The heading rate-limited turn dynamics implementation in the design of CRP is discussed in Chapter 2, which is from [47]. Chapter 3 demonstrates the CRP solution for non-perpendicular intersecting routes and compound intersections which is based from [48]. The on-demand activation and deactivation of the CRP is provided in Chapter 4, which is from [49]. Conclusions are provided in Chapter 5 followed by the Appendix that provides simulation code and flow charts for examples of CRP design shown in Chapters 2-4.

Chapter 2

PROVABLY-SAFE CONFLICT RESOLUTION WITH BOUNDED TURN-RATE FOR AIR TRAFFIC CONTROL

2.1 Introduction

This chapter presents a procedure for resolving conflicts between aircraft along intersecting routes. A challenge in the design of conflict resolution procedures (CRP) is to guarantee the overall safety and efficiency of a route network with multiple intersections — where each CRP acts locally, in space and in time. Recent work [3] has identified necessary and sufficient local conditions on such CRPs to ensure global safety. However, this initial work [3] on the existence of such decoupled CRPs (that can be implemented in a decentralized manner) did not bound the aircraft turn-rate, i.e., the heading angle was assumed to change instantly. While the roll dynamics (that leads to an aircraft turn) is relatively fast and can be ignored in the CRP design, the turn dynamics places an upper bound on the turn-rate [50], which could affect the CRP design. The main contribution of this thesis is to extend the design of provably-safe CRPs by including the bounded-turn-rate limitation. The current chapter provides detailed proofs of the main results, clarifying the importance of including the turn dynamics, and illustrates issues in the applicability of the proposed CRP.

CRPs tend to be decoupled (spatially-and-temporally) because of a substantial increase in computational and modeling complexity with a single global CRP (due to increased number of aircraft and conflicts) when compared to decoupled CRPs, e.g., [5]. Additionally, a global CRP over a large airspace is inefficient (with lower overall capacity [51]) because of the need to handle the larger uncertainty [52]. The

uncertainty tends to be larger, e.g., because of ground-speed sensitivity to wind and temperature that depend, in turn, on forecasts of dynamic weather conditions with substantial uncertainties over time [40]. Therefore, decoupled CRPs are needed to manage the complexity and uncertainty in air traffic control (ATC).

It is challenging to guarantee overall stability in a route network with multiple decoupled CRP. In particular, a challenge is to ensure that modifications of flight trajectories, for resolving a local conflict, do not lead to a *domino* effect; i.e., resolution of a conflict should not lead to new conflicts, whose resolution again leads to additional conflicts, and so on [39]. Moreover, for guaranteeing safety, the procedure should always lead to a solution of the conflict resolution problem. Developing such provably-safe, decoupled CRPs remains a challenging problem in ATC — this is addressed in this thesis.

Previous works on CRPs range from non-local probabilistic approaches that handle uncertainties [53] to local deterministic approaches that resolve conflicts in a collaborative manner [44,54]. Analytical issues such as proving local safety of conflict resolution were studied in, e.g., [55]. Moreover, conditions for CRP stability were studied, for two and three intersecting routes, in [43, 44]. The main difficulty is that conflict resolution at one route-intersection will interact with the conflict resolution at the next intersection along a route; stable solutions to the resulting coupled problem require centralized solutions [43]. In contrast, the current work seeks decoupled procedures that guarantee conflict resolution with multiple conflicts (intersections) by using decoupled CRPs — the cost of this guarantee is time delay in the flow, however, with known bounds. An advantage of the proposed decoupled approach is that the CRP at the next intersection (along a route) can be designed independent of the current CRP. Moreover, the proposed CRP does not require a reduction of the flow capacity in each route (as long as aircraft spacing is greater than D_{sep}); therefore, it can aid in increasing the efficiency of en-route ATC, e.g., in the design of capacity-maintaining protocols for adverse weather re-routing [3].

This chapter begins with an overview of the proposed CRP in Section 2.2, followed by the development of design conditions to ensure safety in Section 2.3. The importance of including turn-rate bounds in the CRP design is investigated in Section 2.4. The proposed approach is illustrated with an example application in Section 2.5 followed by conclusions in Section 2.6

2.2 The Conflict Resolution Procedure (CRP)

2.2.1 Airspace and Conflict Description

The CRP is studied for perpendicularly intersecting aircraft routes, which can occur, e.g., in *highway-like* route structures [56, 57]. These routes, if sufficiently dense in the airspace and variable over time [26], could provide the flexibility needed for accommodating varying weather patterns, missed connections, and traffic congestion by choosing different flight segments in a Free-flight-like setting while maintaining a structure for the aircraft routes. The intersections and routes are assumed to be at a fixed altitude (planar flight) as illustrated in Fig 2.1.

Moreover, the intersections are assumed to be spatially sparse leading to a sufficiently-large local region L, around each intersection (conflict point CP), where the local region L is conflict free from all other routes and other CPs in the airspace. The CRP can use this local region L to resolve conflicts at the intersection without potentially causing additional conflicts as long as the route modifications due to CRP procedures are contained within the local region L. Aircraft along the nominal routes (R_1 and R_2) arrive into this local region L at arrival points A_1, A_2 with a fixed nominal speed v_{sp} and exit at E_1, E_2 as shown in Fig. 2.1. It is assumed that aircrafts, arriving at the local region L are separated by at least distance $\overline{D}$ at the arrival points A_1, A_2, where the minimal arrival spacing $\overline{D}$ is greater than the minimum required separation distance D_{sep} to avoid conflicts.

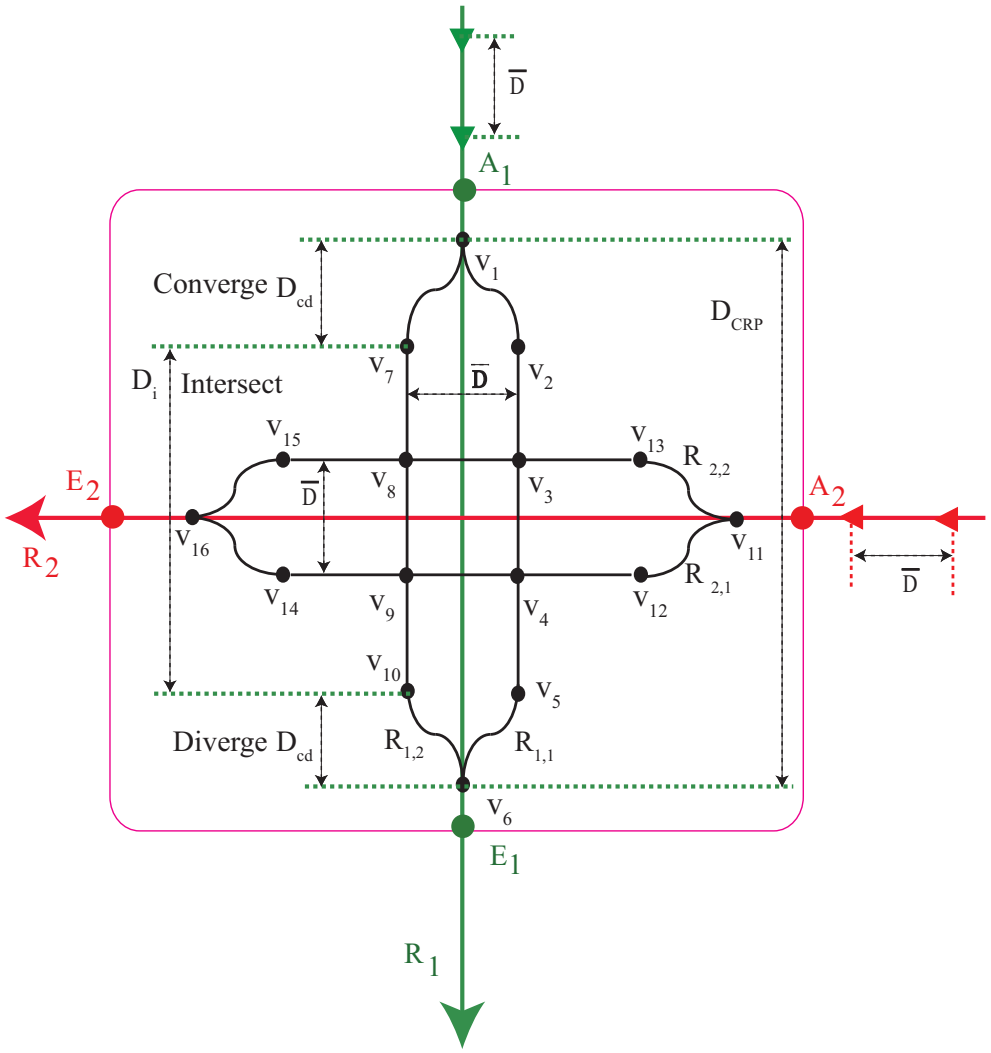

Figure 2.1: Local region L of the airspace around two perpendicularly intersecting routes, R_1, R_2, the corresponding arrival points A_1, A_2, and exit points E_1, E_2. The CRP includes: (i) diverge; (ii) intersect; and (iii) converge.

2.2.2 The Conflict Resolution Problem

The requirements on each decentralized CRP, to enable decoupled designs, are stated below.

Definition 1. *[CRP Decoupling Conditions] The problem is to find a CRP using heading change maneuvers (with bounds on the rate of heading change) such that*

conflicts are avoided between aircraft and the following CRP-decoupling conditions are satisfied:

1. local intent: *aircraft on each route (R_1, R_2) exit along the same route at the corresponding exit point E_1, E_2;*

2. local liveness: *aircraft on each route exit the local region L within a specified bounded maximum time $T < \infty$;*

3. local fairness: *the passage through the local region L is first-come-first-served (FCFS) within each route; and*

4. local exit spacing: *aircraft exiting the local region L (at each point E_1, E_2) are separated by at-least distance $\overline{D}$.*

Remark 1. *The local liveness condition implies that aircraft will not be stuck in the airspace (e.g., in a loop) and the fairness (FCFS) condition facilitates acceptance of the CRP since the FCFS scheduling of aircraft through an airspace is considered as the canonical, fair schedule in ATC.*

Remark 2. *In general, conflict resolution can be achieved using maneuvers that change the heading, speed and altitude. However, heading changes are preferred over speed changes, which require additional fuel for accelerating and decelerating the aircraft. Similarly, heading changes are preferred over altitude changes, which tend to incur passenger discomfort.*

2.2.3 Proposed CRP

When aircraft are closely spaced in the intersecting routes, sufficient space might not be available for aircraft to pass through the intersection point without conflicts. In this scenario, the proposed CRP splits aircraft in each route route (using diverge

procedures) into multiple paths — with increased spacing between aircraft in each path. Aircraft in these paths (with sufficiently large spacing between aircraft) can then pass through the intersection without conflicts as shown in Ref. [3]. After the intersections, aircraft in the different paths are merged back to the original routes.

Remark 3. *The current work addresses perpendicular intersections. However, the main concept of splitting the route to increase aircraft spacing, followed by a merge, could be extended to non-perpendicular intersections (e.g., as in Lemma 5 of Ref. [3]). Moreover, the current CRP can be applied to non-perpendicular cases, if the routes can be re-arranged to create a perpendicular intersection. This might not be possible, if the intersecting routes are close to being parallel in which case a redesign of the overall route structure might be needed.*

An example CRP (with a two-path split) is shown in Fig. 2.1 — it consists of splitting of each route (R_1, R_2) into two equal-length paths and choosing one of the paths for each arriving aircraft. In particular, the two paths $\{R_{1,i}\}_{i=1}^{i=2}$ for route R_1 (shown in Fig. 2.1) are described by a set of way points (v_i):

$$\begin{aligned} R_{1,1} &= \{v_1, v_2, v_3, v_4, v_5, v_6\} \\ R_{1,2} &= \{v_1, v_7, v_8, v_9, v_{10}, v_6\} \end{aligned} \tag{2.1}$$

and the two paths $\{R_{2,i}\}_{i=1}^{i=2}$ for route R_2 are

$$\begin{aligned} R_{2,1} &= \{v_{11}, v_{12}, v_4, v_9, v_{14}, v_{16}\} \\ R_{2,2} &= \{v_{11}, v_{13}, v_3, v_8, v_{15}, v_{16}\}\,. \end{aligned} \tag{2.2}$$

Definition 2. *[*Cyclic, Path-Assignment Procedure*] Let the scheduled time of arrival (STA) of aircraft at the initial way-points (v_1 for route R_1 and v_{11} for route R_2 in Fig. 2.1) be at discrete time instants*

$$t_k = k\left(\frac{\overline{D}/2}{v_{sp}}\right) = kT_{\overline{D}/2} \tag{2.3}$$

where the index k is a nonnegative integer. Then, assign paths $R_{1,1}$ and $R_{2,1}$ (for routes R_1 and R_2, respectively) if k is odd, and paths $R_{1,2}$ and $R_{2,2}$ if k is even.

Remark 4. *The time difference, $2T_{\overline{D}/2}$ in Eq. (2.3), between two scheduled time of arrivals (STAs) on a single route, corresponds to the time needed to travel (with nominal speed v_{sp}) the minimum separation distance $\overline{D}$ between aircraft arriving in each route.*

Remark 5 (Decentralized Implementation)**.** *Although the design of the route structure and the CRPs and the availability of a global clock are centralized, aircraft can self-select the CRP path based on the arrival time index t_k. In this sense, the CRP can be implemented in a decentralized manner — this is similar to automobiles following traffic-light rules in a decentralized fashion. Tight control over the space-time trajectories of the aircraft should be maintained during the CRP, similar to such control of aircraft near airports, e.g., when scheduling landing [37].*

Remark 6. *Synchronization procedures needed to achieve a desired STA has been well studied in the literature, e.g., to schedule arrivals at airports [32,37]. Such approaches can be adapted to manage asynchronous arrivals and achieve STAs for the CRP, in a decentralized manner based on the arrival time, as shown in Ref. [3].*

2.2.4 CRPs are Decoupled

The path-split-based CRP satisfies decoupling conditions as shown in [3]. Briefly, equal-length paths (used in the CRP) enable the local procedure to satisfy the decoupling conditions provided: (i) the aircraft arrival can be synchronized to the discrete time instants; and (ii) the CRP avoids conflicts. The CRP does not change the sequence of aircraft in each route and maintains a minimal separation of $\overline{D}$ (i.e., the route-flow capacity for which the CRP is designed) at the exit. Therefore, if aircraft in one of the routes (R_1 or R_2) reaches another conflict point then the CRP at the second intersection point does not have to depend on the procedures used at the first CRP provided the conflict points are sufficiently separated from each other, i.e., the associated local regions needed for conflict resolution are disjoint. Thus, the design of

the proposed distributed CRPs (that only uses local information of each route) can be decoupled from each other, without domino-type stability problems if the conflict points are sufficiently sparse in the airspace.

2.3 Conditions for Conflict Avoidance

The design of the CRP in Fig. 2.1 to ensure conflict avoidance between aircraft is studied in two parts: (i) the diverge procedure; and (ii) the intersect procedure. The converge procedure is the same as the diverge procedure backwards in time, and hence shares the same conflict avoidance issues — therefore, the converge procedure is not discussed to avoid repetition.

2.3.1 Diverge Procedure

The different aspects of the diverge procedure (single turn, consecutive turns, and path splitting) are shown to be conflict-free below.

Potential for Conflicts during Turns

A critical concern is that the spacing between aircraft along a route could decrease during a turn (when compared to the arrival spacing $\overline{D}$ along each straight route) and result in the loss of minimum separation. Turns occur during the diverge procedure, e.g., in the path $R_{2,2}$ from Fig. 2.1, the route segment from v_{11} to v_{13} (diverge) consists of two consecutive turn paths each with constant radius R, and each curved path results in a heading angle change of ϕ as shown in Fig. 2.2. The potential for reduction in aircraft spacing during a turn places a lower limit on the initial spacing ($\overline{D}$) to achieve a turn ϕ without conflict. These restrictions can become more stringent when making consecutive turns. Conditions for conflict-free turns are studied below for two cases: (i) single turn; and (ii) consecutive turns.

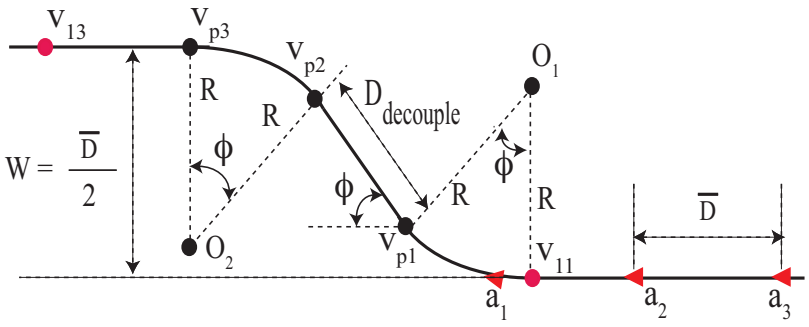

Figure 2.2: Detail of path $R_{2,2}$ (v_{11} to v_{13} from Fig. 2.1) showing two turns.

Restriction on Heading-Change Rate

Although the roll dynamics is relatively fast and can be ignored in the CRP path design, the aircraft turn dynamics should be included in terms of an upper bound on the heading-change rate $\dot{\theta}$ [50]. In particular, from the free body diagram of the aircraft (in the vertical plane) shown in Fig. 2.3, the lift L and the bank angle μ are related by

$$L\cos(\mu) = mg, \quad L\sin(\mu) = mv_{sp}^2/R, \tag{2.4}$$

where v_{sp} is the nominal speed, m is the aircraft mass and g is the gravitational acceleration. Then, the turn radius is given by $R = v_{sp}^2/[g\tan(\mu)]$. With a bank angle limit of $\mu \leq \mu_{max}$ (where $\mu_{max} = 30^\circ$ for passenger safety and comfort in commercial aircraft [50]), the minimum turn radius R_{min} is

$$R_{min} = \frac{v_{sp}^2}{g\tan(\mu_{max})}. \tag{2.5}$$

The above lower bound on the turn radius, i.e., $R > R_{min}$, leads to an upper bound on the maximum heading-change rate. Let the aircraft make a heading change of $\theta \leq \phi$ in time t along a circular arc of radius R and length $R\theta = v_{sp}t$, as shown in Fig. 2.4 (a). Then, the upper bound on the heading-change rate is obtained as

$$\dot{\theta} = \frac{v_{sp}}{R} \quad \leq \quad \frac{v_{sp}}{R_{min}} = \frac{g\tan(\mu_{max})}{v_{sp}} = \Omega_{max}. \tag{2.6}$$

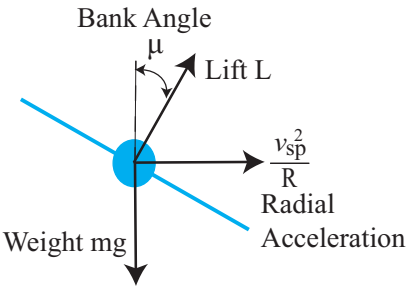

Figure 2.3: Free body of aircraft performing a banked turn; the restriction on the acceptable bank angle limits the maximum heading-change rate.

Thus, the limit on the acceptable bank angle implies that the turn rate should be bounded by a maximum value of Ω_{max}. Note that the rate of heading change $\dot{\theta}$ (i.e. the turn rate) can be kept below the upper bound of Ω_{max} by choosing a sufficiently large turn radius R, i.e.,

$$R \geq R_{min} = \frac{v_{sp}}{\Omega_{max}}. \tag{2.7}$$

Single Turn Analysis

Consider a single turn in a route where aircraft start from a straight section, then move along a circular arc of length $R\phi$, and finally continue along a straight line — both the straight line segments are tangential to the circular arc, and the angle between the two straight segments correspond to the turn angle ϕ as shown in Fig. 2.4 (a).

Lemma 1 (*Single Turn*). *To avoid conflict between aircraft on the route with the single curved turn (as in Fig. 2.4 (a) with $R > R_{min}$) and a maximum heading change of $\phi < \pi/2$, the minimum separation distance $\overline{D}$ between arriving aircraft (on the straight segment) should satisfy*

$$\overline{D} \geq \overline{D}_{\phi} = \frac{D_{sep} - 2R\sin(\frac{\phi}{2})}{\cos(\frac{\phi}{2})} + R\phi \tag{2.8}$$

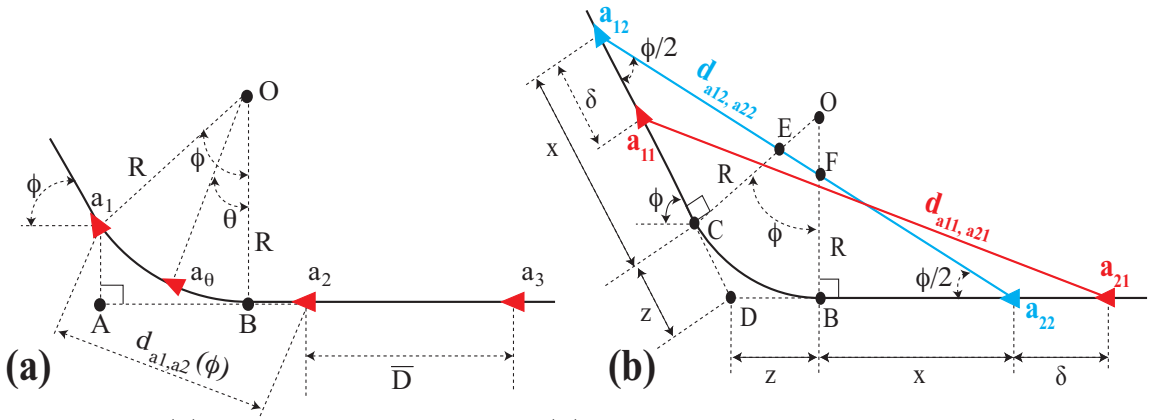

Figure 2.4: (a) Single curved-turn case. (b) Distance between aircrafts during a single turn when only the forward aircraft has passed the curved path of the turn. Distance between the aircraft, $d_{a_{11},a_{21}}$, represents the non-equidistant case, and distance $d_{a_{12},a_{22}}$ represents the symmetric, equidistant case.

if $\phi \leq \phi^* = \frac{\overline{D}}{R}$, *and*

$$\overline{D} \quad \geq \overline{D}_{\phi^*} \quad = 2R\sin^{-1}\left(\frac{D_{sep}}{2R}\right) \quad = \frac{(\phi^*/2)D_{sep}}{\sin(\phi^*/2)} \tag{2.9}$$

otherwise, i.e., when $\phi \geq \phi^*$. *The two conditions are equivalent when the turn angle is at the critical angle, i.e.,* $\phi = \phi^*$. *Turn angle* ϕ *less than or equal to the critical angle* ϕ^* *will be referred to as scenario 1 and turn angle* ϕ *greater than the critical angle* ϕ^* *will be referred to as scenario 2.*

Proof. The proof is divided into four steps.

Step 1: Distance decreases during the turn The distance between the forward aircraft a_1 and an aft aircraft a_2 that were separated by distance $\overline{D}$ in the straight portion decreases as aircraft a_1 starts turning along the circular arc, while the aft aircraft a_2 is still on the straight path as in Fig. 2.4 (a). To show this, consider the distance d_{a_1,a_2} between the two aircraft when the heading change of $\phi \leq \phi^*$ is completed by the first aircraft a_1 as in Fig. 2.4 (a) given by

$$d_{a_1,a_2}(\phi) = \sqrt{(\overline{D} - R\phi + R\sin\phi)^2 + (R - R\cos\phi)^2} \tag{2.10}$$

if $\phi \leq \phi^*$, which follows from the right angled triangle a_1Aa_2 because the length of the arc Ba_1 is $R\phi$, path distance between the two aircraft is $\overline{D}$ due to the constant speed assumption, and thereby, the distance $d_{B,a2}$ is $\overline{D} - R\phi$. If the turn angle θ completed by the aircraft a_1 satisfies $0 < \theta < \phi \leq \phi^*$, then the same expression holds with ϕ replaced by θ, i.e.,

$$d_{a_1,a_2}(\theta) = \sqrt{(\overline{D} - R(\theta - \sin\theta))^2 + (R - R\cos\theta)^2} \tag{2.11}$$

with the derivative of the distance-squared given by

$$\frac{d}{d\theta}[d_{a_1,a_2}(\theta)]^2 = -2R\left(\overline{D} - R\theta\right)(1 - \cos\theta) \quad < 0 \tag{2.12}$$

if $\theta \leq \phi < \phi^*$ and $\phi < \pi/2$ because $\overline{D} = R\phi^* \geq R\theta$ — see definition of ϕ^* in Eq. (2.8). Thus, the distance d_{a_1,a_2} between the aircraft is decreasing as θ increases until $\theta = \phi \leq \phi^*$.

Step 2: Minimum distance for scenario 2 The rate of change in distance with turn angle (the derivative in Eq. 2.12) becomes zero if $\theta = \phi^*$ which can occur if the desired, heading-angle-change ϕ is greater than ϕ^*. The minimal distance, reached when the turn angle becomes $\theta = \phi^*$, is given by (when $\phi \geq \phi^*$)

$$\begin{aligned} d_{a_1,a_2}(\phi^*) &= \sqrt{(R\sin\phi^*)^2 + (R - R\cos\phi^*)^2} \\ &= 2R\sin\tfrac{\overline{D}}{2R} \quad = 2\tfrac{\overline{D}}{\phi^*}\sin\tfrac{\phi^*}{2} \end{aligned} \tag{2.13}$$

It is noted that scenario 2 will be considered only if the critical angle ϕ^* is strictly less than $\pi/2$ to ensure the Lemma's condition that the turn angle ϕ satisfies $\phi < \pi/2$. Furthermore, under scenario 2, when both aircraft are on the curved path, the distance between the aircraft is the minimum distance.

The minimum distance between the aircraft during the turn needs to be larger than the minimum required spacing D_{sep} to avoid conflict. Therefore, under scenario 2, from Eq. (2.13),

$$D_{sep} \leq d_{a_1,a_2}(\phi^*) = 2R\sin\tfrac{\overline{D}}{2R} = 2\tfrac{\overline{D}}{\phi^*}\sin\tfrac{\phi^*}{2} \tag{2.14}$$

that can be rewritten as Eq. (2.9) with arrival spacing $\overline{D} = D_{\phi^*}$ when the equality holds, i.e., $D_{sep} = d_{a_1,a_2}(\phi^*)$.

Step 3: Location for minimum distance for scenario 1 The following shows that the minimum distance between two aircraft occurs when both aircraft are on the straight-line segment, equidistant from the curved path. To show this, the distance $d_{a_{11},a_{21}}$ between aircraft a_{11} and a_{21} (general non-equidistant case — see Fig. 2.4 (b)) is compared to the distance $d_{a_{12},a_{22}}$ between aircrafts a_{12} and a_{22} (symmetric, equidistant case) where the distance between a_{12} and point C as well as a_{22} and point B are the same length x.

By geometry, the angle $\angle CDB$ (shared by triangles $\triangle a_{11}Da_{21}$ and $\triangle a_{12}Da_{22}$) is $\pi - \phi$ since two of the angles in the quadrilateral $CDBO$ are $\pi/2$. Then, the distances between the aircraft, $d_{a_{11},a_{21}}$ (non-equidistant case) and $d_{a_{12},a_{22}}$ (symmetric, equidistant case), can be found by using the law of cosines as shown below.

$$\begin{aligned} d^2_{a_{11},a_{21}} &= d^2_{D,a_{11}} + d^2_{D,a_{21}} - 2d_{D,a_{11}}d_{D,a_{21}}\cos(\pi-\phi) \\ &= (x+z-\delta)^2 + (x+z+\delta)^2 \\ &\quad -2(x+z-\delta)(x+z+\delta)\cos(\pi-\phi) \\ &= 2(x+z)^2(1+\cos(\phi)) + 2\delta^2(1-\cos(\phi)) \end{aligned} \tag{2.15}$$

$$\begin{aligned} d^2_{a_{12},a_{22}} &= d^2_{D,a_{12}} + d^2_{D,a_{22}} \\ &\quad - 2d_{D,a_{12}}d_{D,a_{22}}\cos(\pi-\phi) \\ &= (x+z)^2 + (x+z)^2 \\ &\quad -2(x+z)(x+z)\cos(\pi-\phi) \\ &= 2(x+z)^2\left[1+\cos(\phi)\right] \end{aligned} \tag{2.16}$$

The differences between the two distances, the non-equidistant case and the symmetric equidistant case, can be found by subtracting Eq. (2.15) and Eq. (2.16) to obtain

$$d^2_{a_{11},a_{21}} - d^2_{a_{12},a_{22}} = 2\delta^2(1-\cos\phi) \geq 0. \tag{2.17}$$

Therefore, from Eq. (2.17) the symmetric, equidistant case has the smallest distance for aircraft pairs during the single-turn case under scenario 1.

Step 4: Minimum-distance expression for scenario 1 For the symmetric, equidistant case in Fig. 2.4 (b), the minimum distance during the turn $d_{a_{12},a_{22}}$ (between aircraft positions a_{12} and a_{22}) can be found by adding the three distances $d_{a_{12},E}$, $d_{E,F}$, and $d_{a_{22},F}$, which are found below. Note that, by symmetry, distance $d_{a_{12},E}$ is the same as distance $d_{a_{22},F}$.

Note that the travel distance between a_{12} and a_{22} is $\overline{D}$. Since the arc length of the turn, between point B and point C in Fig. 2.4 (b), is $R\phi$, the distance x is found to be $\frac{1}{2}(\overline{D} - R\phi)$. Due to symmetry in geometry, $\triangle OEF$ is an isosceles triangle. Since $\angle EOF$ is the turn angle ϕ, the angles $\angle a_{12}EC$, and $\angle Ea_{12}C$ are found as $\pi/2 - \phi/2$ and $\phi/2$, respectively. From the right-angled triangle $\triangle Ea_{12}C$,

$$d_{C,E} = d_{B,F} = \frac{1}{2}\tan(\phi/2)(\overline{D} - R\phi) \tag{2.18}$$

$$d_{a_{12},E} = d_{a_{22},F} = \frac{1}{2}\frac{\overline{D} - R\phi}{\cos(\phi/2)}. \tag{2.19}$$

Since the distances $d_{O,C}, d_{O,B}$ are the turn radius R, from Eq. (2.18),

$$d_{O,E} = d_{O,F} = R - d_{C,E} = R - \frac{\tan(\frac{\phi}{2})(\overline{D} - R\phi)}{2} \tag{2.20}$$

Next, from $\triangle OEF$, the distance $d_{E,F}$ between points E and F can be found by using the law of sines as

$$\frac{d_{E,F}}{\sin\phi} = \frac{d_{O,E}}{\sin(\pi/2 - \phi/2)}. \tag{2.21}$$

Substituting for distance $d_{O,E}$ from Eq. (2.20) into the above equation leads to

$$\begin{aligned} d_{E,F} &= \left[R - \tfrac{1}{2}\tan(\phi/2)(\overline{D} - R\phi)\right]\frac{sin(\phi)}{\sin(\pi/2-\phi/2)} \\ &= 2R\sin(\phi/2) - \tfrac{\sin^2(\phi/2)}{\cos(\phi/2)}(\overline{D} - R\phi) \end{aligned} \tag{2.22}$$

Then, the minimal spacing between aircraft can be found as (from Eqs. 2.19,2.22)

$$\begin{aligned} d_{a_{12},a_{22}} &= d_{a_{12},E} + d_{E,F} + d_{a_{22},F} \\ &= (\overline{D} - R\phi)\cos(\phi/2) + 2R\sin(\phi/2) \end{aligned} \tag{2.23}$$

which results in the condition (in Eq. 2.8) for scenario 1 of the Lemma by setting $d_{a_{12},a_{22}} \geq D_{sep}$ in Eq. (2.23) and rearranging the terms. □

Conflict-free Consecutive Turns

A general conflict resolution algorithm could have multiple consecutive turns, e.g., during the diverge/converge procedure. The conflict-free single-turn analysis can be used to show existence of conflict-free multiple turns provided each turn is sufficiently separated from each other. In particular, sufficient spacing between the turns allows conflict-free consecutive turns leading to an offset in the aircraft path during the converge and diverge procedure as shown in Fig. 2.2.

Definition 3. *[*Path Offset Maneuver*] The offset maneuver, shown in Fig. 2.2, consists of two consecutive turns of a route where aircraft start from a straight section, then move along a circular arc (with arc length of $R\phi$, $0 < \phi < \pi/2$), followed by a straight decoupling section, a turn again in the opposite direction (with arc length of $R\phi$), and finally a turn into a straight section. The initial and final straight-line sections are parallel leading to a path-offset maneuver. Moreover, all the straight segments are tangential to the curved paths to avoid large heading-change rates.*

Lemma 2 (*Path Offset Maneuver is Conflict Free*)**.** *If conditions of the single turn Lemma 1 are satisfied for each of the turns inside the consecutive turn maneuver shown in Fig. 2.2, then the offset maneuver (with consecutive turns) is conflict free as long as the straight, decoupling section — in-between the two turns — is sufficiently large.*

Proof. The proof follows from Lemma 1 because a sufficiently-large straight line segment $\overline{v_{p1}, v_{p2}}$ (e.g., of length $D_{decouple} > \overline{D}$) in the middle of the consecutive turn (in Fig. 2.2) implies that there will be no conflicts between aircraft on the two different curved segments (in Fig. 2.2). Therefore, the straight segment in the middle decouples potential conflicts in the maneuver into those for two decoupled single turns, which are conflict free from Lemma 1. □

Remark 7. *The straight line section of length $D_{decouple}$ in Fig. 2.2, between the turns,*

is only used to establish the existence of a conflict-free solution. In practice, the length of this straight line section can be reduced to decrease the space needed for the path-offset maneuver.

Remark 8. *The above Lemma can be used to establish arrival rate criteria for other CRP procedures (e.g., [44]) that use such offset maneuvers.*

Conflict-free Path Splitting

Lemma 3. *Without splitting, let there be no conflicts along a path that is to one side of the initial straight line segment as in Fig. 2.5. Then, there are no conflicts with the path splitting if it is symmetric about the axis of the initial straight line segment.*

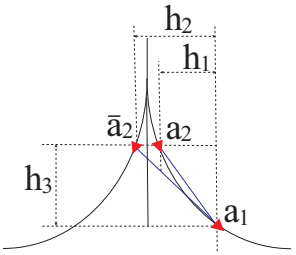

Figure 2.5: Potential conflict between aircraft during path splitting.

Proof. Since there is no conflict between aircraft without the path splitting, the distance between aircraft locations a_1, a_2 in Fig. 2.5 is greater than the minimal separation distance, i.e., $d_{a_1,a_2} \geq D_{sep}$. This implies that there are no conflicts even if the two aircraft were on diverging paths, e.g., between aircraft locations $a_1, \bar{a}_2$ in Fig. 2.5, because

$$d_{a_1,\bar{a}_2} = \sqrt{h_3^2 + h_2^2} \geq \sqrt{h_3^2 + h_1^2} = d_{a_1,a_2} \geq D_{sep} \tag{2.24}$$

where $h_2 \geq h_1$ because aircraft locations a_2 and $\bar{a}_2$ are on two sides of the axis of the initial straight line segment. □

This completes the arguments to show that the different parts of the diverge procedure are conflict free.

2.3.2 Flow Intersect Procedure

The minimum spacing between aircraft needed to enable a single conflict-free intersection is considered next.

Lemma 4. *Let aircraft arrive at a perpendicular intersection evenly spaced in each path at distance $d_{\pi/2}$. Moreover, when an aircraft from one of the paths is at the intersection, it is centered between the aircraft from the other path with a minimal distance of $0.5d_{\pi/2}$. Then, for a conflict-free perpendicular intersection, the minimal spacing in each path is*

$$d_{\pi/2} = 2\sqrt{2}D_{sep}. \tag{2.25}$$

Proof. Consider the case in Fig 2.6 (a) where aircraft a_1 from route $R_{1,1}$ has advanced a distance x from the intersection. Then, the separation distance $d_{a1,b2}$ between aircraft a_1 (on path $R_{1,1}$) and b_2 (on path $R_{2,1}$) is given by

$$\begin{aligned} d_{a1,b2}^2 &= (\tfrac{d_{\pi/2}}{2} - x)^2 + x^2 \\ &= \left[\sqrt{2}x - \tfrac{d_{\pi/2}}{2\sqrt{2}}\right]^2 + \tfrac{d_{\pi/2}^2}{8}. \end{aligned} \tag{2.26}$$

The minimal distance $d_{a1,b2,min}$ between aircraft $d_{a1,b2}$ (over all x) needs to be larger than the minimal separation distance of D_{sep}, i.e., $d_{a1,b2,min}^2 = d_{\pi/2}^2/8 \geq D_{sep}^2$

which results in the condition (Eq. 2.25 in the Lemma) for the minimal spacing between aircraft in each path. □

Splitting the routes into n paths increases the spacing between aircraft from $\overline{D}$ in each route to $n\overline{D}$ in each path, which enables conflict-free intersections from Lemma 4.

Lemma 5. *Aircraft that arrive synchronized do not have conflicts with each other in the intersection area (e.g., marked by D_i in Fig. 2.1 for Route R_1) with the use of the two-path assignment procedure in Definition 2 if the path lengths from the arrival points to the straight line segments are all equal and if there is sufficient arrival spacing ($2\overline{D} > 2\sqrt{2}D_{sep} = d_{\pi/2}$).*

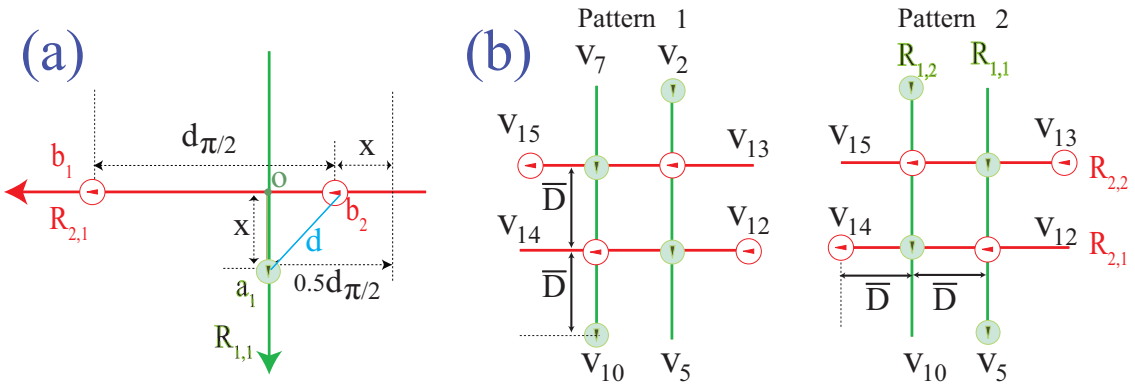

Figure 2.6: (a) Separation distance on a single intersection. (b) All possible scenarios under which an aircraft can occupy a path intersection in Fig. 2.1.

Proof. This follows from Lemma 4 and Ref. [3] since all scenarios under which an aircraft can occupy a path intersection in Fig 2.6 (b) satisfy Lemma 4 conditions. □

Remark 9. *[Number of Paths Splits] The number n of paths required is at-most* 3 *since the resulting aircraft spacing on the paths will satisfy the Lemma 4 condition, i.e., $3\overline{D} > 2\sqrt{2}D_{sep}$ as the minimal arrival spacing $\overline{D}$ is larger than the minimal separation distance D_{sep}. CRPs with more than two splits have been studied in [3]. Increasing the number of splits can increase the minimal aircraft distance in the CRP.*

This completes the discussion to show that the proposed CRP is conflict free provided the arrival spacing conditions are met for the single turn (Lemma 1) and consecutive turns are sufficiently separated (Lemma 2).

2.4 Bounded vs. Unbounded Rate Turns

The rate-bounded (continuous) turn and the rate-unbounded (instantaneous) turn, shown in Fig. 2.7, are comparatively evaluated to illustrate the importance of including the turn dynamics in the CRP design.

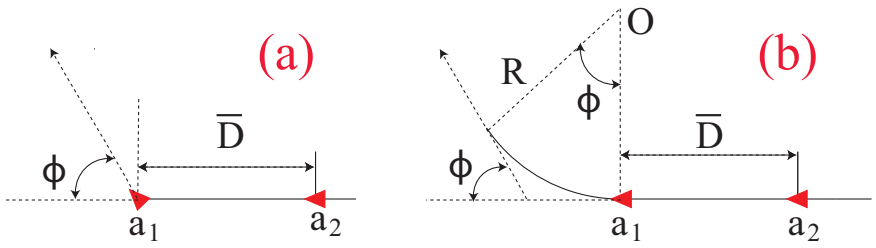

Figure 2.7: (a) Rate-unbounded (instantaneous) turn and (b) rate-bounded turns.

2.4.1 Importance of Single-turn Analysis

The arrival-spacing requirement to have a conflict-free single-turn (in Lemma 1) is the critical requirement for the proposed conflict-free CRP design. This is because conflict-free conditions for multiple turns (in the converge and diverge portions of the CRP) can be established using the single-turn condition as shown in Lemmas 2 and 3. Moreover, the flow-intersect procedure does not add requirements on the route's arrival spacing since sufficient spacing for conflict-free intersection can be achieved by splitting each route into at-the-most three paths as in Remark 9. Therefore, the arrival spacing requirement for the overall CRP can be established using Lemma 1 which develops the conditions to avoid conflicts within a single turn.

2.4.2 Comparison of Arrival Spacing

Lemma 6. *For any given turn angle $0 < \phi < \pi/2$, the lower bound (e.g., $\overline{D}_\phi$ in Eq. 2.8, Lemma 1) on the acceptable arrival-spacing distance $\overline{D}$ for a rate-bounded turn maneuver is less than the corresponding lower bound ($\overline{D}_{\phi,inst}$) on the acceptable arrival-spacing distance $\overline{D}$ for an instantaneous turn maneuver.*

Proof. The lower bound $\overline{D}_{\phi,inst}$ on the acceptable arrival-spacing distance $\overline{D}$ for an instantaneous turn is given by [3]

$$\overline{D}_{\phi,inst} = \frac{D_{sep}}{\cos(\phi/2)}. \tag{2.27}$$

For scenario 1 ($\phi \leq \phi^*$ in Lemma 1), the difference between the lower bounds for the two cases (instantaneous turn vs. rate-bounded turn) is, from Eqs. (2.8, 2.27),

$$\begin{aligned} \overline{D}_{\phi,inst} - \overline{D}_{\phi} &= \frac{D_{sep}}{\cos(\phi/2)} - \left[\frac{D_{sep} - 2R\sin(\phi/2)}{\cos(\phi/2)} + R\phi\right] \\ &= R\left[2\tan(\phi/2) - \phi\right] \quad = R\Delta_{\overline{D}}(\phi). \end{aligned} \tag{2.28}$$

The Lemma follows for scenario 1 ($\phi \leq \phi^*$) since

$$\Delta_{\overline{D}}(\phi) \;=\; \int_0^{\phi/2} \frac{d\Delta_{\overline{D}}}{d\phi}\, d\phi \;=\; \int_0^{\phi/2} \tan^2(\phi/2)\, d\phi \quad > 0 \tag{2.29}$$

where the integrand is a positive continuous function when $\phi > 0$.

For scenario 2 ($\phi > \phi^*$ in Lemma 1), the difference between the lower bounds for the two cases (instantaneous turn vs. rate-bounded turn) is, from Eqs. (2.9, 2.27),

$$\begin{aligned} \overline{D}_{\phi,inst} - \overline{D}_{\phi^*} &= \frac{D_{sep}}{\cos(\phi^*/2)} - (R\phi^*) \\ &= R\left[2\tan(\phi^*/2) - \phi^*\right] \quad \text{from Eq. (2.14)} \\ &= R\Delta_{\overline{D}}(\phi^*). \end{aligned} \tag{2.30}$$

The Lemma follows for scenario 2 ($\phi > \phi^*$) since $\Delta_{\overline{D}}(\phi^*) > 0$ from Eq. (2.29) because $\phi^* < \phi < \pi/2$. □

Remark 10. *The above Lemma also implies that a larger turn angle ϕ is possible with a given aircraft spacing $\overline{D}$ for the rate-bounded approach when compared to the instantaneous-turn case. This is to be expected because the rate-bounded turn takes more distance to complete the turn when compared to the instantaneous turn (see Fig. 2.7). Therefore, the reduction of aircraft spacing during the more-gradual, rate-bounded turn is expected to be less — leading to less potential for conflict.*

Lemma 7 (*Effect of Increasing Turn Radius*)**.** *For a given arrival spacing $\overline{D} > D_{sep}$, the rate-bounded turn can achieve any given turn angle $0 < \phi < \pi/2$ with a sufficiently-large, turn radius R.*

Proof. The rate-bounded turn becomes a scenario 2 case (i.e., $\phi^* < \phi$) for a sufficiently large turn radius R since, from Eq. (2.8), the critical turn angle ϕ^* becomes small

($\phi^* \to 0$) as the turn radius increases ($R \to \infty$). As a result, from Eq. (2.9), the minimal arrival spacing $\overline{D}_{\phi^*}$ scenario 2 tends to the limit

$$\lim_{\phi^* \to 0} \overline{D}_{\phi^*} = \lim_{\phi^* \to 0} \frac{D_{sep}\phi^*/2}{\sin(\phi^*/2)} = D_{sep} \tag{2.31}$$

as the turn radius increases ($R \to \infty$). Then, the arrival flow spacing distance $\overline{D}$ is greater than the minimal arrival spacing distance $\overline{D}_{\phi^*}$ for a sufficiently large turn radius R because

$$\begin{aligned} \lim_{R \to \infty} \overline{D} - \overline{D}_{\phi^*} &= \lim_{\phi^* \to 0} [(\overline{D} - D_{sep}) - (\overline{D}_{\phi^*} - D_{sep})] \\ &= \overline{D} - D_{sep} \ > 0 \end{aligned} \tag{2.32}$$

□

The lower bound on the minimal arrival spacing is compared in Fig. 2.8; as expected, from Lemma 6, for a given turn angle ϕ, the rate-bounded turn can be achieved with a lower arrival spacing when compared to the instantaneous case. Moreover, Fig. 2.8 shows that the maximum allowable turn angle is larger with the rate-bounded turn when compared to the single instantaneous turn. Furthermore, as the turn radius is increased, e.g., from $R = R_{min}$ to $R = 2R_{min}$, the required arrival spacing decreases. While any turn angle $0 < \phi < \pi/2$ can be achieved for any given arrival spacing $\overline{D} > D_{sep}$ with a rate-bounded turn (by choosing a sufficiently large turn radius R), the maximum turn angle ϕ is limited for the instantaneous case to $\phi < 2\cos^{-1}\left(\frac{D_{sep}}{\overline{D}}\right)$ according to Eq. (2.27).

For example, with an arrival spacing of $\overline{D} = 5.1$NM, the maximum turn angle for the instantaneous turn is $\phi = 22.73°$ as seen in Fig. 2.8. Even for the instantaneous turn case, multiple turns can be used to achieve a larger net turn $0 < \phi < \pi/2$; however, each turn would not be consistent with the aircraft turn-rate bounds. In contrast, by choosing a sufficiently large turn radius, e.g., $R = 2R_{min}$, any turn angle $0 < \phi < \pi/2$ can be achieved with the rate-bounded approach. Thus, the above development of rate-bounded turns enables the design of CRPs, which include the aircraft turn-dynamics.

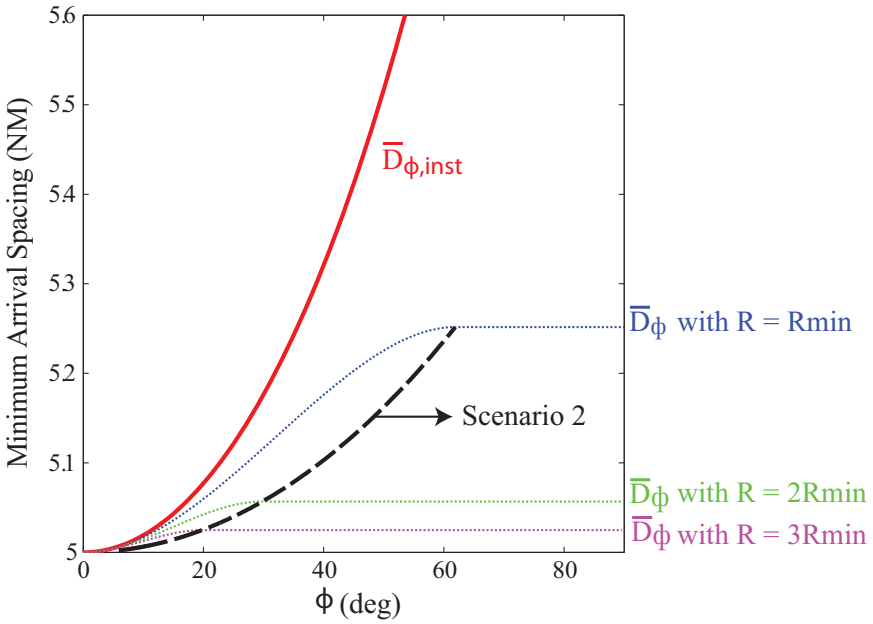

Figure 2.8: Comparison of minimum arrival spacing for: (dotted lines) rate-bounded turns $\overline{D}_{\phi}$ from Eqs. (2.8,2.9); and (solid line) instantaneous turns $\overline{D}_{\phi,inst}$ from Eq. (2.27).

2.5 Example Route Intersection

An example route intersection is used in this Section to illustrate the CRP design. Additionally, robustness of the proposed CRP is discussed.

2.5.1 Description of the Route Intersection

Conflicts are not expected in actual air traffic data (since these are already resolved) — however, existing data can be used to illustrate issues in the design of the proposed CRP. In particular, data from a perpendicularly intersecting route in the Cleveland sector ZOB59 (see Fig. 2.9) was used to: (i) quantify the minimum arrival spacing $\overline{D}_{min}$ in the routes; and (ii) identify the average aircraft velocity and its variation.

The data was obtained using the Future ATM Concepts Evaluation Tool (FACET) for May 1st, 2004 at 35000ft altitude during the time interval $[305, 65735]$ Coordinated Universal Time (UTC) seconds which corresponds to about 18 hours of data. During

this period, 35 aircraft passed through the east-to-west route, and 19 aircraft passed through the north-to-south route. The average velocity of these aircraft on the two routes from Fig 2.9 was 0.122 nautical miles per second (NM/s) with a standard deviation of 0.005 (NM/s).

The latitude and longitude of each aircraft in the data were converted into 2-D Cartesian coordinates (x,y) by using stereographic projection [58], and the averaged initial and final coordinates were used as the nominal entry and exit points for the sector for the example. The routes R_1 and R_2 were then considered to be straight lines between these averaged arrival and departure points, which were then used to identify the route intersection point, as in Fig. 2.9. Finally, the estimated time of arrival (ETA) of the aircraft at the initial waypoints were synchronized to a scheduled time of arrival (STA)

$$STA = kT_{\overline{D}} \quad \text{if} \quad ETA \in \left[\left(k - \frac{1}{2} \right) T_{\overline{D}}, \left(k + \frac{1}{2} \right) T_{\overline{D}} \right) \tag{2.33}$$

which assigns a discrete time instant $t_k = kT_{\overline{D}}$ to the arrival of each aircraft at the initial waypoint.

The minimum spacing between aircraft pairs was $\overline{D}_{min} = 9.23(NM)$ which is obtained by calculating the minimum 2-D position distance in routes R_1 and R_2 in this example data.

2.5.2 CRP Design

The design variables in the CRP design are the arrival spacing $\overline{D}$, the number of paths into which the routes are split, the turn angle ϕ, and the turn radius R in the converge/diverge procedures.

Bounds on Arrival Spacing

A smaller number of split paths (of each route) can reduce the overall area needed for the CRP. The maximum possible arrival spacing $\overline{D}$ that can be used in the CRP

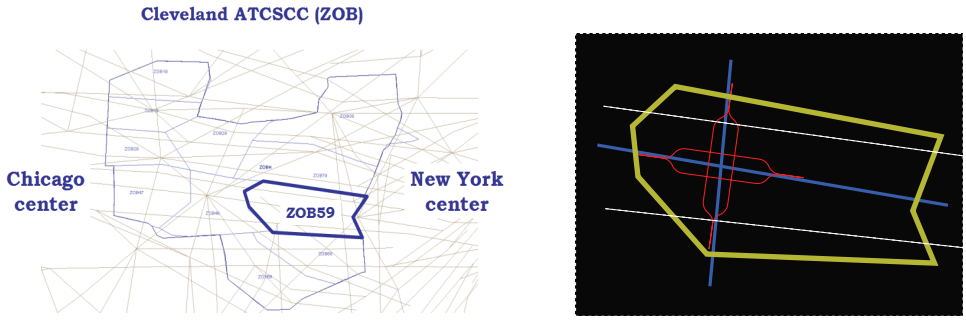

Figure 2.9: (a) Cleveland Sector ZOB59. (b) Perpendicularly intersecting routes (solid line), CRP designed with the arrival spacing of $\overline{D} = 9.23(NM)$, and example nearby routes (dashed lines).

design is the minimal aircraft spacing in the routes $\overline{D}_{min} = 9.23(NM)$, i.e., for this example,

$$\overline{D} \leq \overline{D}_{min} = 9.23(NM). \tag{2.34}$$

The maximum arrival spacing $\overline{D} = 9.23(NM)$ is still smaller than the spacing needed $2\sqrt{2}D_{sep} = 14.14(NM)$ for the conflict-free intersection condition from Eq. (2.25) in Lemma 4, where the minimal separation distance is $D_{sep} = 5(NM)$. Therefore, each route needs to be split into at-least two paths as in Fig. 2.6. With two paths, the possible choice of arrival spacing $\overline{D}$ needs to satisfy, from Lemma 4,

$$2\overline{D} \geq 2\sqrt{2}D_{sep} = 14.14(NM) \tag{2.35}$$

From Eqs. (2.34,2.35), the acceptable arrival spacing $\overline{D}$ in the CRP design should satisfy

$$\frac{2\sqrt{2}}{2}D_{sep} = 7.07(NM) \leq \overline{D} \leq \overline{D}_{min} = 9.23(NM). \tag{2.36}$$

Turn Angle and Radius

During the diverge section (and similarly during the converge section), e.g., between nodes v_1 and v_2 in Fig. 2.1, the route is offset by half the arrival spacing $\overline{D}/2$ used

in the CRP design by using the path-offset scheme shown in Fig. 2.2. Moreover, to minimize the space needed for the maneuver, we begin with the smallest possible decoupling distance, $D_{decouple} = 0$. Minimal separation between aircraft with this choice needs to be checked numerically, however,

the analysis in Section 2.3 shows that the CRP will be safe provided the decoupling distance is sufficiently large. With a zero decoupling distance $D_{decouple} = 0$, the offset requirement implies that

$$\overline{D}/2 = 2R(1 - \cos\phi) \tag{2.37}$$

which relates the turn radius R and the turn angle ϕ by

$$R = \frac{\overline{D}/4}{1 - \cos\phi}. \tag{2.38}$$

Choosing the Turn Angle

The lateral distance L_{curve} needed for the offset maneuver with bounded-rate turns (see Fig. 2.2) is given by (with zero decoupling distance $D_{decouple} = 0$)

$$L_{curve} \;=\; 2R\sin\phi \;= \frac{\overline{D}/2}{1 - \cos\phi}\sin\phi. \tag{2.39}$$

Note that the required lateral distance L_{curve} decreases with increasing turn angle ϕ (for $0 < \phi < \pi/2$) since the derivative with respect to ϕ is negative, i.e.,

$$\begin{aligned} \frac{dL_{curve}}{d\phi} &= \frac{\overline{D}}{2}\left[\frac{\cos\phi(1-\cos\phi) - \sin^2\phi}{(1-\cos\phi)^2}\right] \\ &= -\frac{\overline{D}}{2}\left[\frac{1}{(1-\cos\phi)}\right] \quad \leq 0. \end{aligned} \tag{2.40}$$

Therefore, the turn angle ϕ should be chosen as large as possible to reduce the lateral distance L_{curve} needed to accomplish the offset maneuver, which, in turn will reduce the overall space needed for the CRP. However, the required turn radius R (from Eq. 2.38), which needs to be larger than the minimum turn radius (R_{min} from Eq. 2.7), decreases with increasing turn angle ϕ for $0 < \phi < \pi/2$ since the denominator (in

Eq. 2.38) is increasing with ϕ. The maximum turn angle $\phi = \phi_{max}$, which occurs when $R = R_{min}$, can be found from Eq. (2.38) as

$$\phi_{max} = \cos^{-1}\left[1 - \frac{\overline{D}}{4R_{min}}\right]. \tag{2.41}$$

Effect of Varying the Arrival Spacing

The minimal distance $D_{cd,min}$ between aircraft during the diverge maneuver and the minimal distance $D_{i,min}$ during the intersect portion of the CRP are computed numerically and shown in Fig. 2.10 for different values of the arrival spacing $\overline{D}$ in the acceptable range from Eq. (2.36) with zero decoupling distance $D_{decouple} = 0$. The overall minimal distance in the CRP is at-least the minimal separation distance D_{sep}, and therefore, the CRP is conflict free for arrival spacing $\overline{D}$ in the acceptable range (Eq. 2.36) — even with zero decoupling distance $D_{decouple} = 0$. The minimal distance between the aircraft in the CRP increases with the arrival spacing $\overline{D}$ used in the design.

The variation of the size D_{CRP} of the CRP (see Fig. 2.1), given by

$$\begin{aligned} D_{CRP} &= 2D_{cd} + D_i \\ &= 2\tfrac{\overline{D}/2}{1-\cos\phi_{max}} \sin\phi_{max} + 3\overline{D}, \end{aligned} \tag{2.42}$$

is shown in Fig. 2.10. The space needed for the CRP (quantified by D_{CRP}) increases with the arrival spacing $\overline{D}$ used in the design.

Remark 11. *A CRP with smaller space requirement will enable, more dense route structures, e.g., see comparison of space needed in Fig, 2.10. In general, a redesign of the overall route structure will be needed to provide sufficient space when adopting the proposed CRP approach. For example, nearby routes (e.g., shown in dashed lines in Fig. 2.9) will need to be spaced further away. Alternatively, resolution of nearby conflicts can be integrated into a compound CRP [3].*

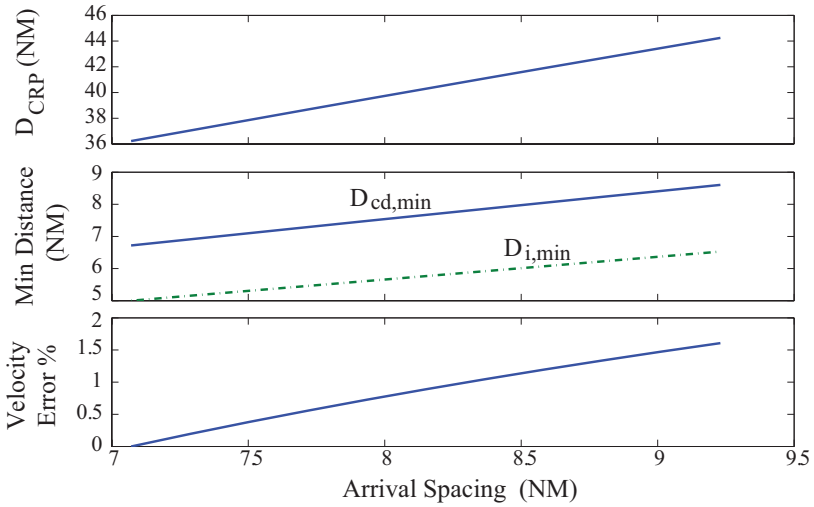

Figure 2.10: Effect of choosing different arrival spacing $\overline{D}$ in CRP design. Top plot: variation of the CRP size D_{CRP} from Eq. (2.42). Middle plot: variation of the minimal distance $D_{cd,min}$ in the converge/diverge section and the minimal distance $D_{i,min}$ in the intersect section. Bottom plot: variation of percent acceptable velocity error $E_{v,\%}$ from Eq. (2.44).

2.5.3 Simulation of Example Sector

The CRP simulation for the sector example described above along with the simulation flow chart is provided in the Appendix of this thesis. In the simulation, the arrival flow spacing distance is $\overline{D} = 9.23NM$ as described in the example description in Section 2.5.1. The turn radius R is calculated from Eq. (2.5), and the turn angle ϕ is calculated from Eq. (2.41) with zero decoupling distance $D_{decouple} = 0$ on the CRP turn paths. The simulation shows that the CRP with bounded turn rates are conflict free.

2.5.4 Size vs. Robustness

Using a smaller arrival spacing leads to a smaller CRP size (quantified by D_{CRP}); however, it also leads to lower robustness as shown below. In particular, the space needed for the CRP is minimized by choosing the smallest arrival spacing $\overline{D} = \frac{2\sqrt{2}}{2} D_{sep} =$

$7.07(NM)$, as seen in Fig. 2.10. However, this design is not robust since the minimal distance of aircraft in the CRP is equal to the minimum required aircraft separation $D_{sep} = 5(NM)$ from Fig. 2.10. For example, small variations, e.g., in the speed of the aircraft can lower distance between the aircraft below the required aircraft separation D_{sep} leading to conflicts between aircraft.

In contrast, with the larger arrival spacing $\overline{D} = 9.23(NM)$, the minimal distance between the aircraft in the CRP (which occurs in the intersection region as shown in Fig. 2.10) is given by $D_{i,min} = 9.23/\sqrt{(}2) = 6.5266(NM)$. This larger than the minimal required separation leads to robustness of the CRP, for example, due to errors caused by velocity variations. To illustrate, let the maximum path length for aircraft through the CRP be P_{CRP} given by

$$\begin{aligned} P_{CRP} &= 4R\phi_{max} + 3\overline{D} \\ &= 4\frac{\overline{D}/4}{1-\cos\phi_{max}}\phi_{max} + 3\overline{D}. \end{aligned} \tag{2.43}$$

Then, the maximum error e_{pos} in the position of an aircraft (from the expected position with the nominal speed v_{sp} during the CRP is $e_{pos} = \frac{P_{CRP}}{v_{sp}}\delta_{vmax}$ where δ_{vmax} is the maximum (absolute) deviation from the nominal velocity. This position error would be acceptable during the CRP, if it does not lead to conflict. Since each aircraft could have velocity deviations, the resulting error in position e_{pos} of each aircraft needs to be smaller than half the excess separation between aircraft in the CRP, i.e., $e_{pos} \leq \frac{D_{i,min}-D_{sep}}{2}$.

Therefore, the acceptable variation in the aircraft speed is

$$E_{v,\%} = \frac{\delta_{vmax}}{v_{sp}} * 100 \leq \frac{\bar{D}_{sep} - D_{sep}}{2P_{CRP}} * 100, \tag{2.44}$$

which increases with an increase in the the arrival spacing $\overline{D}$ used the CRP design as seen in Fig. 2.10. For the example, with the maximum arrival spacing $\overline{D} = 9.23(NM)$, the velocity error needs to be maintained below 1.6% to avoid conflicts.

Remark 12. *Even for the smallest arrival spacing of $\overline{D} = 7.07(NM)$, the CRP can be re-designed to be robust by including a factor of safety K_{fs} in the minimum separation*

distance D_{sep} between aircraft used in the CRP design, i.e., i.e., by choosing a larger minimum separation distance

$\bar{D}_{sep} = K_{fs} D_{sep}$. *The resulting excess separation* $(\overline{D}_{sep} - D_{sep})$ *can be used to ensure robustness.*

2.6 Conclusion

This chapter presented a conflict resolution procedure (CRP) that includes the effect of turn dynamics by bounding the heading-change (turn) rate. The main importance of including the turn-rate bound is when proving the existence of a safe CRP. In particular, as shown in Section 2.4, for a given arrival spacing $\overline{D}$ in the intersecting routes, the allowable turn angle (in a single turn) can be different depending on whether the turn-rate-bound is included in the analysis or not. The bounded turn analysis (with a given arrival spacing $\overline{D}$) is important to establish the existence of conflict free converge/diverge procedures in Section 2.3. Moreover, the bounded-turn-rate analysis is important to quantify the size of the converge/diverge procedure and therefore, to quantify the space needed for the CRP. The proposed CRP was illustrated with an example conflict between two routes in the Cleveland sector.

This chapter is aimed at establishing the existence of decoupled CRP, which is challenging, in general, for large distributed systems such as ATC. The next chapters consider issues such as extension of the proposed CRP for non-perpendicular intersections with variations in aircraft route speeds, and applying on-demand CRPs, which consider activating and deactivating CRPs depending on the incoming flow rate of routes.

Chapter 3

DECOUPLED CONFLICT RESOLUTION PROCEDURES FOR NON-PERPENDICULAR AIR TRAFFIC INTERSECTIONS WITH DIFFERENT SPEEDS

3.1 INTRODUCTION

This chapter presents a Conflict Resolution Procedure (CRP) for non-perpendicularly intersecting routes in en-route Air Traffic Control (ATC).

The current chapter removes two limitations in these decouple provably-safe CRP approaches by: (i) developing procedures for non-perpendicular intersections; and (ii) removing the requirement that aircraft on each intersecting route have the same speed. Conditions are developed to guarantee conflict resolution, and the proposed CRP is illustrated through an example en-route intersection in the Cleveland sector.

The current chapter generalizes the CRP design in Chapter 2, which satisfied conditions for conflict resolution, and led to decoupled CRPs for multiple en-route intersections without the domino effect. Briefly, the main ideas (same as the CRP proposed in Chapter 2) of the decoupling CRP are: (i) route-splitting into multiple paths; (ii) equal-length maneuvers; and (iii) path merging into original routes. Splitting each route in an intersection into multiple paths increases the spacing between aircraft in each of the split paths the increased spacing allows conflict-free intersections between paths. After the intersection, the paths are merged back into the original routes, which implies that new conflicts will not be generated outside of the CRP region. Therefore, domino effects can be avoided if the route intersections (and as a consequence, the CRP regions) are sufficiently sparse in the airspace. Finally, since the travel distance of the split paths are of equal length, the CRP maintains the

original minimal arrival spacing and sequence of aircraft in each route. Therefore, the state of aircraft arrival (minimal spacing and sequence) at the next intersection along the route is not adversely impacted by the current CRP, which enables the decoupled CRP design.

A problem is that the decoupling CRP in Chapter 2 was designed for perpendicularly intersecting routes with all aircraft having the same speed. While perpendicular intersections can be used to resolve non-perpendicular intersections by rearranging the routes to create a perpendicular intersection [3], the additional maneuvers needed for such rearrangements can lead to inefficiencies due to: (i) additional ATC effort, (ii) increased travel distance (fuel cost); and (iii) increased time (delay). Moreover, optimal cruise speed can be different in each of the intersecting routes due to wind effects that depend on the different direction of the routes [59, 60]. Both these issues, non-perpendicular intersections and different speeds in the different routes, are addressed in the current chapter.

3.2 The Conflict Resolution Problem (CR)

The routes are assumed to be at a fixed altitude (planar flight) intersecting at an angle θ_{int}, and each route has a different nominal aircraft velocity, i.e., route 1 has velocity v_1, and route 2 has velocity v_2, as illustrated in Fig 3.1. Moreover, the intersections are assumed to be spatially sparse leading to a sufficiently-large local region L, around each intersection (conflict point CP), where the local region L is conflict free from all other routes and other CRPs in the airspace. The CRP can use this local region L to resolve conflicts at the intersection without potentially causing additional conflicts as long as the route modifications due to CRP procedures are contained within the local region L. Aircraft along the nominal routes (Route 1 and Route 2) arrive into this local region L at arrival points A_1, A_2 and exit at E_1, E_2 as shown in Fig. 3.1. It is assumed that aircrafts, arriving at the local region L are separated by at least distance $\overline{D}_1$ for route 1 at the arrival point A_1 and, $\overline{D}_2$ for route 2 at the arrival

point A_2, where the minimal arrival spacing $\overline{D}_1$ and $\overline{D}_2$ is greater than the minimum required separation distance D_{sep} to avoid conflicts.

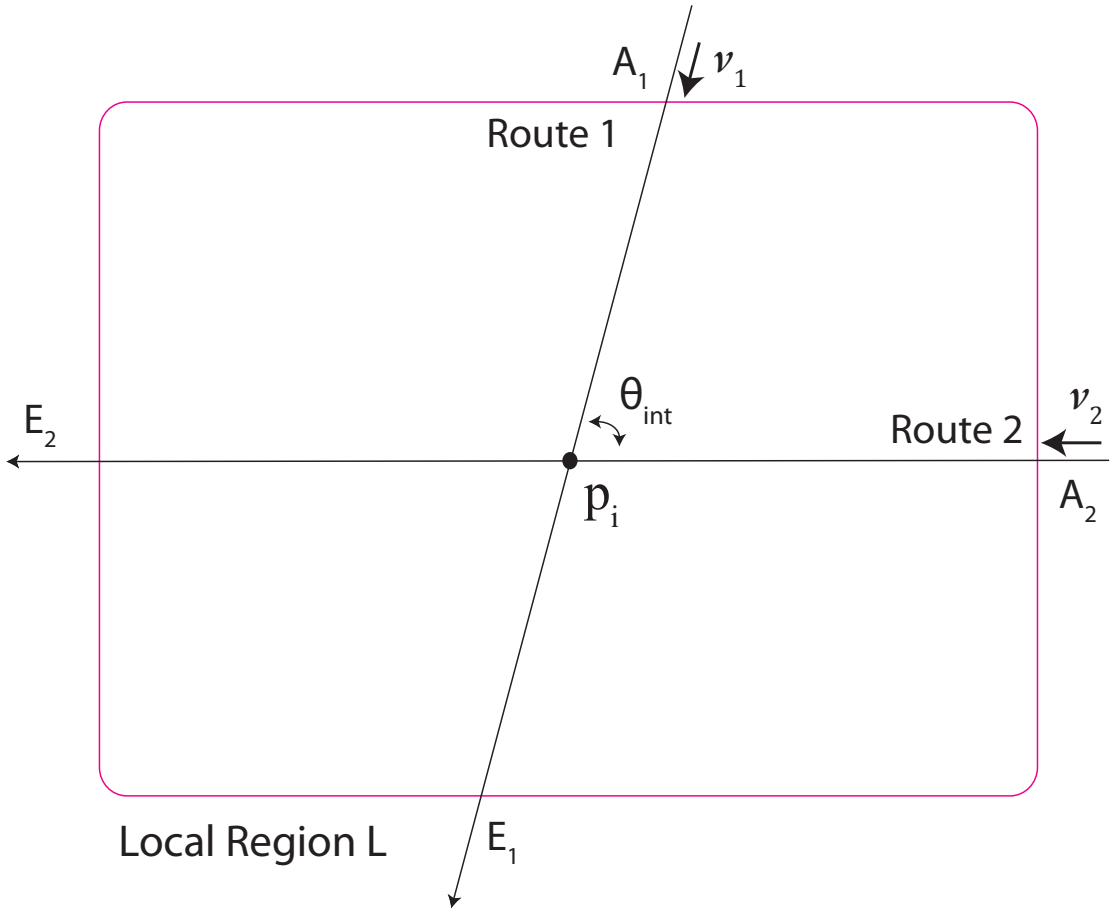

Figure 3.1: Local region L of the airspace around two intersecting routes, (route 1 and route 2) with angle θ_{int}, the corresponding arrival points A_1, A_2 into region L, and exit points E_1, E_2 from region L.

3.2.1 Path-Split CRP

An example CRP (with a two-path split) is shown in Fig. 3.2 — it consists of splitting of each route (route 1 and route 2) into two turning paths (diverge) and choosing one of the paths for each arriving aircraft. The aircraft in each path is merged back into

the original routes after the intersection.

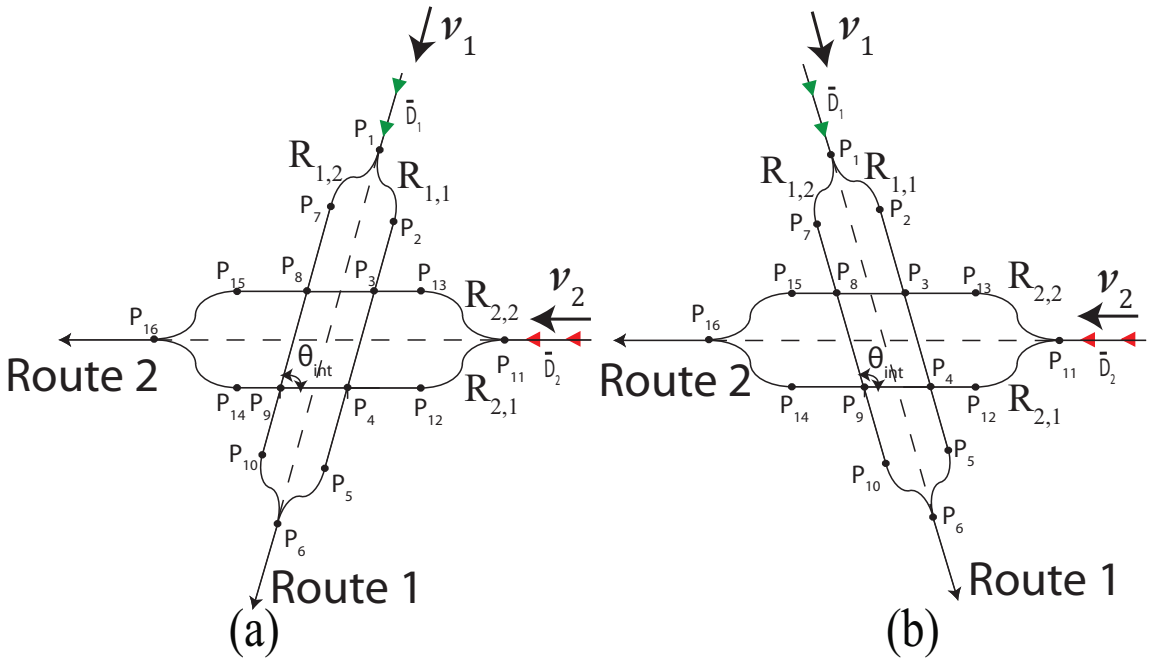

Figure 3.2: The CRP splits aircraft in each route (using diverge procedures) into multiple paths — with increased spacing between aircraft in each path. Aircraft in these paths (with sufficiently increased spacing between aircraft due to the route splitting) can then pass through the intersection without conflicts.After the intersections, aircraft in the different paths are merged back to the original routes. Each CRP path is shown for (a) when intersection angle θ_{int} is smaller than $\pi/2$ and (b) when intersection angle θ_{int} is greater than $\pi/2$

The two paths $\{R_{1,1}, R_{1,2}\}$ for route 1 (shown in Fig. 3.2) are described by a set of way points (v_i):

$$\begin{aligned} R_{1,1} &= \{p_1, p_2, p_3, p_4, p_5, p_6\} \\ R_{1,2} &= \{p_1, p_7, p_8, p_9, p_{10}, p_6\} \end{aligned} \tag{3.1}$$

and the two paths $\{R_{2,1}, R_{2,2}\}$ for route 2 are

$$\begin{aligned} R_{2,1} &= \{p_{11}, p_{12}, p_4, p_9, p_{14}, p_{16}\} \\ R_{2,2} &= \{p_{11}, p_{13}, p_3, p_8, p_{15}, p_{16}\}\,. \end{aligned} \tag{3.2}$$

Definition 4. *[*Cyclic, Path-Assignment Procedure*] Let the scheduled time of arrival (STA) of aircraft at the initial way-points (p_1 for route 1 and p_{11} for route 2 in Fig. 3.2) be at discrete time instants*

$$t_k \quad = k\left(\frac{\overline{D}_1}{v_1}\right) \quad = k\left(\frac{\overline{D}_2}{v_2}\right) \quad = kT_{\overline{D}} \tag{3.3}$$

where the index k is a non-negative integer, $\overline{D}_1$ is the arrival spacing and v_1 is the aircraft speed for route 1, while $\overline{D}_2$ is the arrival spacing and v_2 is the aircraft speed for route 2. Then, assign paths $R_{1,1}$ and $R_{2,1}$ (for route 1 and route 2, respectively) when k is odd, and paths $R_{1,2}$ and $R_{2,2}$ if k is even. The path allocation rule is cyclic and repeats after every two discrete time instants. The main concept is that for the given discrete time instance t_k, even when route 1 and route 2 with different velocities, each route have aircraft move by their respective arrival spacing distance $\overline{D}_1$ and $\overline{D}_2$. Also, from Eq. (3.3), $\overline{D}_1 = \alpha\overline{D}_2$.

Remark 13. *The time $T_{\overline{D}}$ between two scheduled time of arrivals (STAs) on a single route, in Eq. (3.3), corresponds to the time needed to travel the minimum arrival separation $\overline{D}$ between aircraft in each route.*

Remark 14. *Synchronization procedures needed to achieve a desired STA have been well studied in the literature, e.g., to schedule arrivals at airports [32, 37]. Such approaches can be adapted to manage asynchronous arrivals and achieve STAs for the CRP, in a decentralized manner based on the arrival time, as shown in Ref. [3].*

3.3 Conflict Free Conditions for the CRP

A brief summary of the conditions for the path-split CRP is provided in the following. Mainly, when compared to previous works [3,47], the proposed CRP needs to account for 2 issues; (i) the non-perpendicular intersection angle θ_{int} and (ii) the potential for different aircraft speed v_1, v_2 in each route.

3.3.1 Flow Intersection Procedure

In order to find the minimum spacing requirement within a flow, the case is considered when an aircraft from one flow is at the center of the aircraft spacing of the intersecting flow. This enables a single conflict-free intersection between two paths.

Lemma 8. *The two-path split CRP can be designed to be conflict free at the intersection if the minimal arrival spacing $\overline{D}_1$ and $\overline{D}_2$ satisfies the following conditions*

$$2\overline{D}_1 > d_{int1} = \frac{2D_{sep}\sqrt{\alpha^2 - 2\alpha\cos\theta_{int} + 1}}{\sin\theta_{int}}. \tag{3.4}$$

$$2\overline{D}_2 > d_{int2} = \frac{2D_{sep}\sqrt{\alpha^2 - 2\alpha\cos\theta_{int} + 1}}{\alpha\sin\theta_{int}}. \tag{3.5}$$

where $v_1 = \alpha v_2$.

Proof. Let the spacing between the aircraft in path $R_{1,1}$ be d_{int1}, when one of the aircraft (e.g., b_1) is at the intersection as in Fig. 3.3a. Then, the spacing $d_{a1,b1}$ between aircraft on different paths as in Fig. 3.3b, after aircraft a_1 from route $R_{1,1}$ has advanced a distance αx from the intersection and aircraft b_1 from route $R_{2,1}$ has advanced by a distance x (since airspeed $v_1 = \alpha v_2$), is given by

$$d_{a1,b1}^2 = x^2 + \left(\alpha x + \frac{d_{int1}}{2}\right)^2 - 2x\left(\alpha x + \frac{d_{int1}}{2}\right)\cos\theta_{int}. \tag{3.6}$$

The minimum spacing of $d_{a1,b1}$ occurs when the derivative of $d_{a1,b1}$ with respect to x is equal to 0. This is when

$$x = \frac{-(\alpha d_{int1} - d_{int1}\cos\theta_{int})}{2(\alpha^2 - 2\alpha\cos\theta_{int} + 1)}.$$

Substituting x into Eq. (3.6) yields the minimum distance of $d_{a1,b1}$. Thus, to avoid conflict at the intersection, the minimum spacing $d_{a1,b1}$ must be greater than D_{sep} which is given by

$$\begin{aligned} min(d_{a1,b1}^2) &= \frac{d_{int1}^2\sin^2\theta_{int}}{4(\alpha^2 - 2\alpha\cos\theta_{int} + 1)} \\ &= D_{sep}^2. \end{aligned} \tag{3.7}$$

Re-arranging this equation yields the spacing requirement in Eq.(3.4) listed in Lemma 8. A similar analysis results in the spacing requirement shown in Eq.(3.5) with an aircraft from the north-to-south route initially occupying the intersection as in Fig. 3.4, and in-between two aircraft from the east-to-west route as in Fig. 3.4(b). □

Remark 15. *The benefits in applying the CRP to non-perpendicularly intersecting routes can be decided by the amount of rerouting efforts required to meet perpendicularly and apply the CRP solution provided in Chapter 2. For instance, if the original intersecting angle θ_{int} is close to 0 or 180 degrees, the spacing distance required to avoid conflicts (d_{int1} in Eq.(3.4) and d_{int2} in Eq.(3.5)) become extremely large. For such cases, rerouting and applying the perpendicular CRP solution provided in Chapter 2 would be more suitable.*

Remark 16. *The number of path splits n needed in the CRP is determined by the spacing requirement d_{int1} and d_{int2} at the intersection from Lemma 8 and the arrival spacing in both routes ($\overline{D}_1$ and $\overline{D}_2$), i.e.*

$$n\overline{D}_1 > d_{int1} \tag{3.8}$$

$$n\overline{D}_2 > d_{int2}. \tag{3.9}$$

3.3.2 Diverge/Converge Procedure

The arrival spacing conditions to avoid conflict in the diverge and converge procedure is provided below.

Lemma 9 (*Single Turn*)**.** *To avoid conflict between aircraft on the route with the single curved turn, (a) with a maximum heading change of $\phi < \pi/2$, the minimum separation distance $\overline{D}$ between arriving aircraft (on the straight segment) should satisfy*

$$\overline{D} \geq \overline{D}_\phi = \frac{D_{sep} - 2R\sin(\frac{\phi}{2})}{\cos(\frac{\phi}{2})} + R\phi \tag{3.10}$$

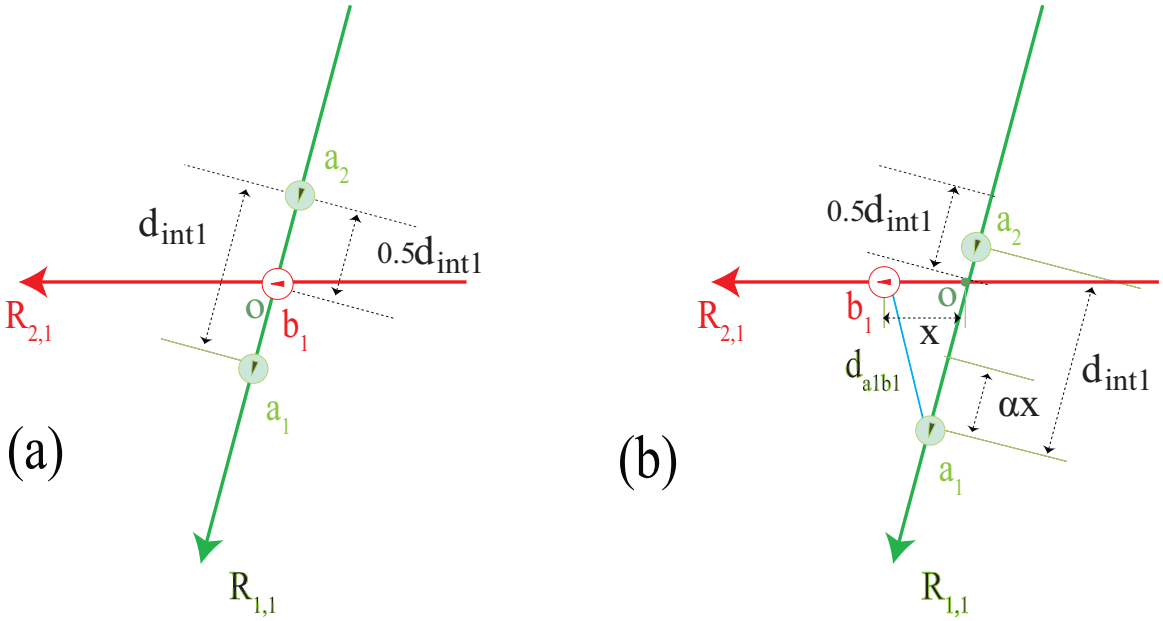

Figure 3.3: Separation distance between intersecting aircraft when aircraft b_1 occupies the center of the intersection and then aircraft from route $R_{2,1}$ advancing x distance while aircraft from route $R_{1,1}$ advance αx.

if $\phi \leq \phi^* = \frac{\overline{D}}{R}$, *and*

$$\overline{D} \quad \geq \overline{D}_{\phi^*} \quad = 2R \arcsin\left(\frac{D_{sep}}{2R}\right) \quad = \frac{(\phi^*/2) D_{sep}}{\sin(\phi^*/2)} \tag{3.11}$$

otherwise, i.e., when $\phi \geq \phi^*$. *The two conditions are equivalent when the turn angle is at the critical angle, i.e.,* $\phi = \phi^*$. *Turn angle* ϕ *less than or equal to the critical angle* ϕ^* *will be referred to as scenario 1 and turn angle* ϕ *greater than the critical angle* ϕ^* *will be referred to as scenario 2.*

Proof. The proof was presented in chapter 2, and is not presented here. To summarize the the main concept, sufficient spacing should be available between aircraft before the turn so that even with decreases in the distance between aircraft while performing a turn, the initial spacing is sufficiently large to prevent conflicts. The only difference for the non-perpendicular intersection case is that the turn paths of the intersecting

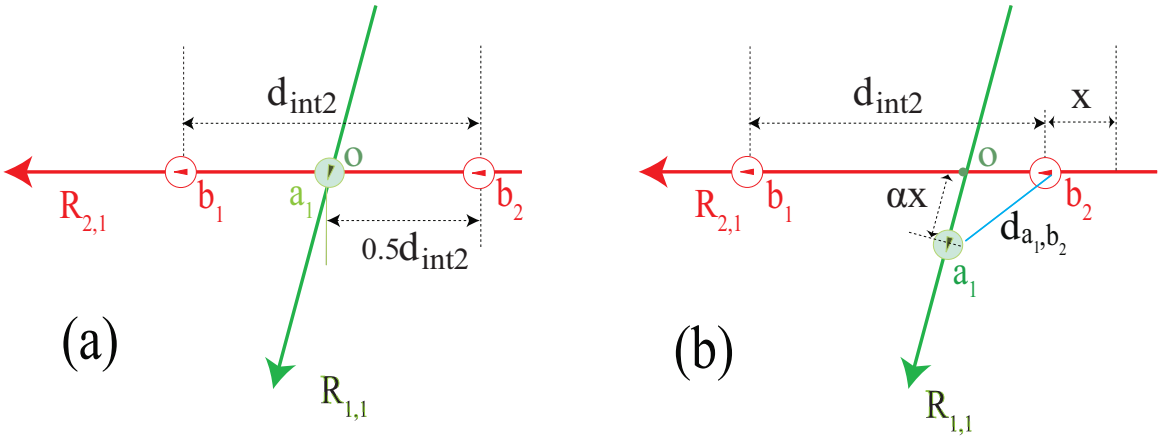

Figure 3.4: Separation distance between intersecting aircraft when aircraft a_1 occupies the center of the intersection and then aircraft from route $R_{2,1}$ advancing x distance while aircraft from route $R_{1,1}$ advance αx.

routes are not symmetric. For example, the split from route 2 into paths $R_{2,1}$ and $R_{2,2}$ each with turn radii R_2 with turn angle ϕ_2 are different to the split from route 1 into paths $R_{1,1}$ and $R_{1,2}$ each with turn radii R_1 with turn angle ϕ_1 as shown in Fig. 3.2(a) and (b). □

3.3.3 Timing Procedure on the CRP Intersections

Due to the aircraft speed difference between route 1 and route 2, the split paths from route 1 and route 2 have different path lengths and are not symmetric. However, the conflict-free intersection criterion in Lemma 8, requires the aircraft on one path to be centered (when it is at the intersection) between potential aircraft positions on the other path. This requires some care in the design to ensure precision timing of the aircraft arrivals at the intersection, for a non-perpendicular intersection. The timing procedure is divided into two cases; (i) when intersection angle θ_{int} is smaller than $\pi/2$ and (ii) when intersection angle θ_{int} is greater than $\pi/2$ due to the different path

lengths in the CRP design.

Lemma 10. *Consider the CRP case when intersection angle θ_{int} is smaller than $\pi/2$ and $\alpha \geq 1$ as in Fig. 3.2 (a). Let the distance (d_{p13p3}) between node p_{13} to p_3 on path $R_{2,2}$ be $\overline{D}_2$ for a 2-path split CRP for non-perpendicular intersection with intersection angle θ_{int} smaller than $\pi/2$ as in Fig. 3.2(a). Then, the CRP where aircraft follow the path assignment procedure from Definition 4 is conflict-free when the following conditions of the CRP path lengths are satisfied.*

1. *The turn path distance between node p_1 and p_2 as well as turn path distance between node p_1 and p_7 are both $2R_1\phi_1$ where R_1 is the turn radius and ϕ_1 is the turn angle of the turn paths.*

2. *The turn path distance between node p_{11} and p_{12} as well as turn path distance between node p_{11} and p_{13} are both $2R_2\phi_2$ where R_2 is the turn radius and ϕ_2 is the turn angle of the turn paths.*

3. *The node distance(d_{p12p4}) between p_{12} and p_4 prior to the intersection is $\overline{D}_2+\Delta_2$ (where Δ_2 is the extra path length to reach the intersection node from the end of the turn path, i.e., difference between node distance d_{p13p3} and d_{p12p4}).*

4. *The node distance(d_{p2p3}) between p_2 and p_3 prior to the intersection is $2\alpha R_2\phi_2+\overline{D}_1-2R_1\phi_1$ from route $R_{1,1}$.*

5. *The node distance(d_{p7p8}) between p_7 and p_8 prior to the intersection is $d_{p2p3}+\Delta_1$ (where Δ_1 is the extra path length to reach the intersection node from the end of the turn path, i.e., difference between node distance d_{p7p8} and d_{p2p3}).*

6. *The node distance at the intersection (d_{p3p4}) between p_3 and p_4 as well as the node distance (d_{p8p9}) between p_8 and p_9 are $\overline{D}_1+\alpha\Delta_2$.*

7. *The node distance at the intersection (d_{p3p8}) between p_3 and p_8 as well as the node distance (d_{p4p9}) between p_4 and p_9 are $\overline{D}_2 + \frac{\Delta_1}{\alpha}$.*

Proof. Let aircraft a_1 arrive at initial-way point p_1 at time t_1 aircraft b_1 arrive at initial-way point p_{11} at time t_1. Then aircraft a_2 arrives at initial-way point p_1 at time $t_2 = t_1 + T_{\overline{D}}$ and b_2 arrives at initial-way point p_{11} also at t_2. The proof is provided below by the travel time to reach each intersection node (p_3, p_4, p_8, and p_9) between aircraft a_1 and b_1 from the initial-way point or a_2 and b_2 from $\overline{D}$ before the respective initial-way point with the CRP design conditions in Lemma 10.

Case 1: Travel time to intersection node p_3 Consider the case when aircraft a_1 traveling from the initial way point (node p_1) to node p_3 which is the center of the intersection between paths $R_{2,2}$ and $R_{1,1}$, while aircraft b_2 which travels from $\overline{D}_2$ prior to the initial way point (node p_{11}) to node p_3. The travel time for aircraft a_1 to travel from initial way point (p_1) to point p_3 is given by

$$t_{a1p3} = \frac{2R_1\phi_1 + d_{p2p3}}{v_1} \tag{3.12}$$

while the travel time for aircraft b_2 to travel from $\overline{D}_2$ prior to initial way point (p_{11}) to point p_3 is given by

$$t_{b2p3} = \frac{2R_2\phi_2 + d_{p13p3} + \overline{D}_2}{v_2}. \tag{3.13}$$

Then, the difference in travel time between aircraft b_2 and a_1 is

$$t_{b2p3} - t_{a1p3} = T_{\overline{D}}, \tag{3.14}$$

by using the conditions in the current Lemma 10, and using the relationship between the velocities ($v_1 = \alpha v_2$). Since the travel time difference is $T_{\overline{D}}$ from Eq. 3.3, there is no conflict at intersection point p_3 between aircraft a_1 and b_2 as in Fig. 3.5(a) since when a_1 is at p_3, b_2 would be $\overline{D}_2$ distance away.

Case 2: Travel time to intersection node p_4 Consider the case when aircraft a_1 traveling from the initial way point (node p_1) to node p_4 which is the center of the

intersection between paths $R_{2,1}$ and $R_{1,1}$, while aircraft b_1 which travels from the initial way point (node p_{11}) to node p_4. The travel time for aircraft a_1 to travel from initial way point (p_1) to point p_4 is given by

$$t_{a1p4} = \frac{2R_1\phi_1 + d_{p2p3} + d_{p3p4}}{v_1} \tag{3.15}$$

while the travel time for aircraft b_1 to travel from initial way point (p_{11}) to point p_4 is given by

$$t_{b1p4} = \frac{2R_2\phi_2 + d_{p12p4}}{v_2}. \tag{3.16}$$

Then, the difference in travel time between aircraft b_1 and a_1 is

$$t_{b1p4} - t_{a1p4} = T_{\overline{D}}, \tag{3.17}$$

by using the conditions in the current Lemma 10, and using the relationship between the velocities ($v_1 = \alpha v_2$). Since the travel time difference is $T_{\overline{D}}$ from Eq. 3.3, there is no conflict at intersection point p_4 between aircraft a_1 and b_1 as in Fig. 3.5(b) since when b_1 is at p_4, a_1 would be $\overline{D}_1$ distance away prior to p_4.

Case 3: Travel time to intersection node p_8 Consider the case when aircraft a_2 traveling from the initial way point (node p_1) to node p_8 which is the center of the intersection between paths $R_{2,2}$ and $R_{1,2}$, while aircraft b_2 which travels from the initial way point (node p_{11}) to node p_8. The travel time for aircraft a_2 to travel from initial way point (p_1) to point p_8 is given by

$$t_{a2p8} = \frac{2R_1\phi_1 + d_{p7p8}}{v_1} \tag{3.18}$$

while the travel time for aircraft b_2 to travel from initial way point (p_{11}) to point p_8 is given by

$$t_{b2p8} = \frac{2R_2\phi_2 + d_{p13p3} + d_{p3p8}}{v_2}. \tag{3.19}$$

Then, the difference in travel time between aircraft b_2 and a_2 is

$$t_{b2p8} - t_{a2p8} = T_{\overline{D}}, \tag{3.20}$$

by using the conditions in the current Lemma 10, and using the relationship between the velocities ($v_1 = \alpha v_2$). Since the travel time difference is $T_{\overline{D}}$ from Eq. 3.3, there is no conflict at intersection point p_4 between aircraft a_2 and b_2 as in Fig. 3.5(c) since when a_2 is at p_8, b_2 would be $\overline{D}_2$ distance away prior to p_4.

Case 4: Travel time to intersection node p_9 Consider the case when aircraft a_2 traveling from $\overline{D}_1$ prior to the initial way point (node p_1) to node p_9 which is the center of the intersection between paths $R_{2,1}$ and $R_{1,2}$, while aircraft b_1 which travels from the initial way point (node p_{11}) to node p_9. The travel time for aircraft a_2 to travel from $\overline{D}_1$ prior to the initial way point (p_1) to point p_9 is given by

$$t_{a2p9} = \frac{\overline{D}_1 + 2R_1\phi_1 + d_{p7p8} + d_{p8p9}}{v_1} \tag{3.21}$$

while the travel time for aircraft b_1 to travel from initial-way point (p_{11}) to point p_9 is given by

$$t_{b1p9} = \frac{2R_2\phi_2 + d_{p12p4} + d_{p4p9}}{v_2}. \tag{3.22}$$

Then, the difference in travel time between aircraft a_2 and b_1 is

$$t_{a2p9} - t_{b1p9} = T_{\overline{D}}, \tag{3.23}$$

by using the conditions in the current Lemma 10, and using the relationship between the velocities ($v_1 = \alpha v_2$). Since the travel time difference is $T_{\overline{D}}$ from Eq. 3.3, there is no conflict at intersection point p_9 between aircraft a_2 and b_1 as in Fig. 3.5(d) since when b_1 is at p_9, a_2 would be $\overline{D}_1$ distance away. □

Lemma 11. *Consider the CRP case when intersection angle θ_{int} is greater than $\pi/2$ and $\alpha \geq 1$ as in Fig. 3.2 (b). Let the distance (d_{p12p4}) between node p_{12} to p_4 on path $R_{2,1}$ be $\overline{D}_2$ for a 2-path split CRP for non-perpendicular intersection with intersection angle θ_{int} greater than $\pi/2$ as in Fig. 3.2(b). Then, the CRP where aircraft follow the path assignment procedure from Definition 4 is conflict-free when the following conditions of the CRP path lengths are satisfied.*

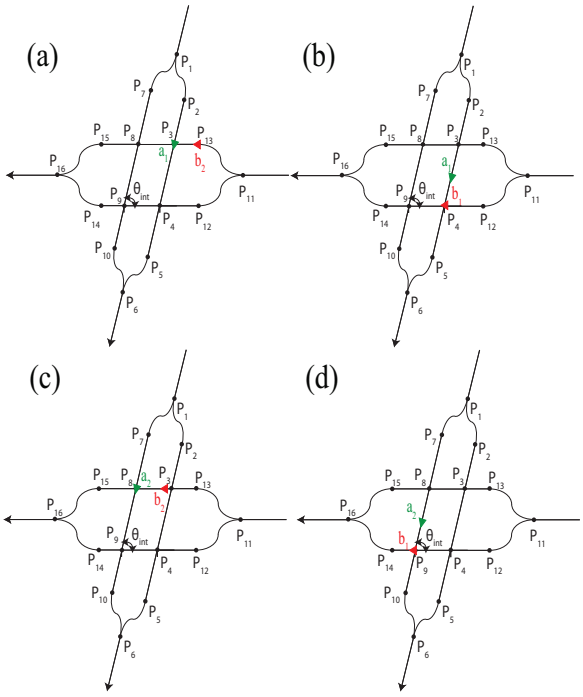

Figure 3.5: The timing procedure shows when an aircraft occupies an intersection node where (a) aircraft a_1 is at p_3, b_2 is $\overline{D}_2$ distance away. (b) aircraft b_1 is at p_4, a_1 is $\overline{D}_1$ distance away. (c) aircraft a_2 is at p_8, b_2 is $\overline{D}_2$ distance away. (d) aircraft b_1 is at p_9, a_1 is $\overline{D}_1$ distance away

1. *The turn path distance between node p_1 and p_2 as well as turn path distance between node p_1 and p_7 are both $2R_1\phi_1$ where R_1 is the turn radius and ϕ_1 is the turn angle of the turn paths.*

2. *The turn path distance between node p_{11} and p_{12} as well as turn path distance between node p_{11} and p_{13} are both $2R_2\phi_2$ where R_2 is the turn radius and ϕ_2 is the turn angle of the turn paths.*

3. *The node distance(d_{p13p3}) between p_{13} and p_3 prior to the intersection is $\overline{D}_2+\Delta_2$ (where Δ_2 is the extra path length to reach the intersection node from the end of the turn path, i.e., difference between node distance d_{p13p3} and d_{p12p4}).*

4. *The node distance(d_{p2p3}) between p_2 and p_3 prior to the intersection is $d_{p7p8}+\Delta_1$.*

5. *The node distance(d_{p7p8}) between p_7 and p_8 prior to the intersection is $2\alpha R_2\phi_2+\alpha\Delta_2+\overline{D}_1-2R_1\phi_1-\Delta_1$.*

6. *The node distance at the intersection (d_{p3p4}) between p_3 and p_4 as well as the node distance (d_{p8p9}) between p_8 and p_9 are $\overline{D}_1-\alpha\Delta_2$.*

7. *The node distance at the intersection (d_{p3p8}) between p_3 and p_8 as well as the node distance (d_{p4p9}) between p_4 and p_9 are $\overline{D}_2-\frac{\Delta_1}{\alpha}$.*

Proof. The proof is similar to the proof of Lemma 10 and is not shown here. The main concept is with the path conditions in the current Lemma 11 to find the travel time difference from the initial-way point or one flow spacing distance prior to the initial-way point to the intersection point, where the travel time difference between aircraft from the intersecting flow is $T_{\overline{D}}$ from Eq. 3.3 which ensures that no conflict occurs since when one aircraft is at the intersection point, the intersectng aircraft is one flow spacing distance away. □

3.4 Results and Discussion

An example is presented below to illustrate the proposed CRP. Nominal values for the example (such as aircraft arrival rates, speed, and spacing) are based on aircraft data from a perpendicularly intersecting route in the Cleveland sector ZOB59 (see Fig. 3.6). The data was obtained using the Future ATM Concepts Evaluation Tool (FACET) for May 1st, 2004 at 35000ft altitude during the time interval $[305, 65735]$ Coordinated Universal Time (UTC) seconds which corresponds to about 18 hours of data. During this period, 35 aircraft passed through the east-to-west route, and 19 aircraft passed through the north-to-south route. The average of aircraft speeds

for each route in the date were used as the nominal aircraft speed, where the north-to-south route (route 1) velocity $v_1 = 0.123$ Nautical Miles (NM) per second (or 443 knots), while the east-to-west route (route 2) velocity $v_2 = 0.121$ Nautical Miles (NM) per second (or 437 knots) in the example. The velocity ratio between route 1 and route 2, $\alpha = \frac{v_1}{v_2} = 1.014$. Moreover, the minimal arrival spacing from the data including both routes is $\overline{D}_{min} = 9.23$ NM. For the example , the spacing for the east-to-west route (route 2) was considered to be the minimum arrival spacing after synchronization $\overline{D}_2 = \overline{D}_{min} = 9.23$NM, while the north-to-south route (route 1) spacing is $\overline{D}_1 = \alpha\overline{D}_2 = 9.36$NM, as discussed from Definition 4. With both arrival spacing $\overline{D}_1$ and $\overline{D}_2$, a 2-way split CRP provides sufficient spacing between aircraft in the split paths for conflict free intersections of the paths, i.e., $n = 2$ in Eqs. (3.8), (3.9) of Remark 16. Example cases are presented using these data. The first example case demonstrates the local CRP design with different speeds between each routes and considers the non-perpendicular intersection ($\theta_{int} = 60° < 90°$). The simulation and flow chart of this example is provided in the Appendix of this thesis. The second example demonstrates multiple intersections with local CRPs applied at each intersection with different properties (i.e, non-perpendicular intersections with general route speeds or perpendicular intersection with the same route speeds).

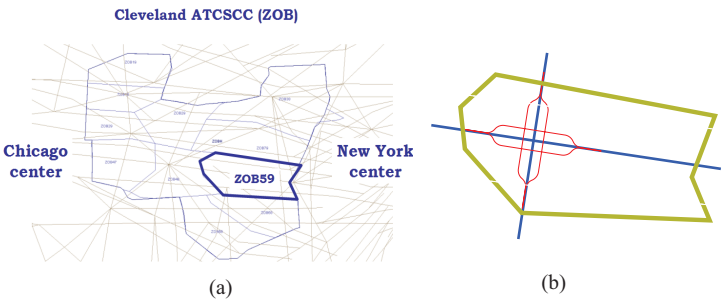

Figure 3.6: Example perpendicularly intersecting routes in Cleveland Sector.

3.4.1 Example non-perpendicular local intersection with different speeds

With the intersection angle in this section chosen as $\theta_{int} = 60°$, the available variables in the CRP design are the arrival spacing $\overline{D}_1$, $\overline{D}_2$, the number of paths into which the routes are split, Δ_1 (difference between node distance d_{p7p8} and d_{p2p3}) and Δ_2 (difference between node distance d_{p13p3} and d_{p12p4}) , the turn angle ϕ_1, ϕ_2, and the turn radius R_1,R_2 in the converge/diverge procedures.

Bounds on Arrival Spacing for Single Flow Intersection

The number of split paths are determined by the arrival spacing from each route and ultimately, using a smaller number of splits can reduce the CRP area. The arrival spacing for each route as described above is $\overline{D}_1 = \alpha\overline{D}_2 = 9.36$NM for route 1 and $\overline{D}_2 = \overline{D}_{min} = 9.23$NM for route 2. With the given arrival spacing for each route, and the known value of the speed ratio between the two routes α, and the minimum separation distance $D_{sep} = 5$NM, the required intersection distances d_{int1} and d_{int2} from Lemma 8 Eq. (3.4) and (3.5) are $d_{int1} = 14.24$NM, and $d_{int2} = 14.05$NM. Therefore, the arrival spacing from each route satisfies the conflict-free condition at the intersection as per Lemma 8 when each route splits into at-least two paths as per Remark 16.

Turn Angle and Radius

The turn radius and turn angle have geometrical dependents on the height distance (H_1 and H_2) as in Fig. 3.7, where the turn path from the diverging split path are symmetric (same turn radius and turn angle for the turn paths). The turn radius, however, has limitations due to roll dynamics as discussed in chapter 3. The free body diagram of the aircraft (in the vertical plane) shown in Fig. 3.8, the lift L and

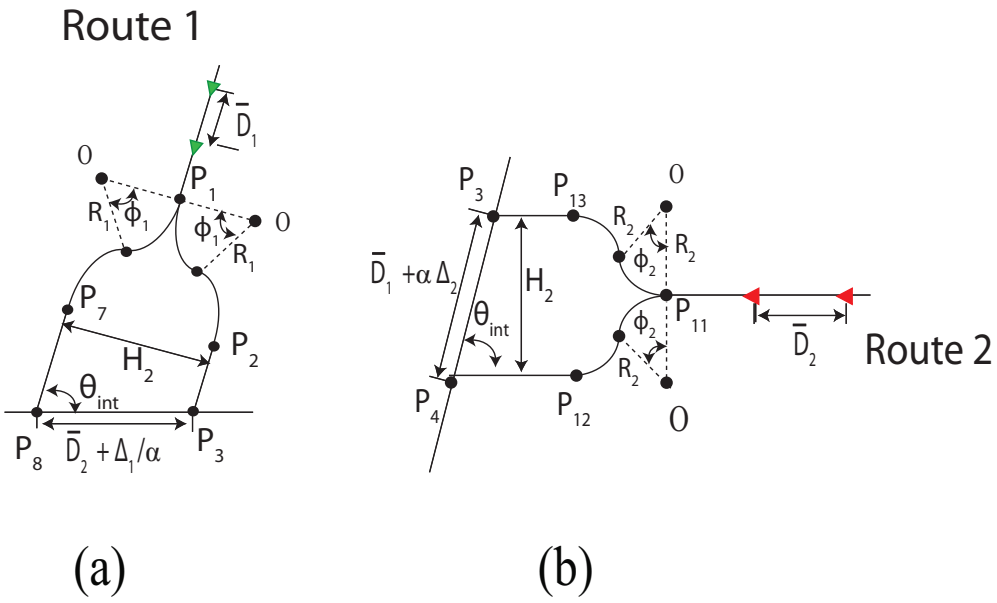

Figure 3.7: The diverge turn paths of the CRP. The turn path parameters are geometrically related to the intersection angle θ_{int}, and the intersection distance d_{p3p8} and d_{p3p4}. (a) The turn paths from route 1 and intersection distance d_{p3p8} . (b) The turn paths from route 2 and intersection distance d_{p3p4}.

the bank angle μ are related by

$$L\cos(\mu) = mg, \quad L\sin(\mu) = mv_{sp}^2/R, \tag{3.24}$$

where v_{sp} is the nominal speed, m is the aircraft mass and g is the gravitational acceleration. Then, the turn radius is given by $R = v_{sp}^2/(g\tan(\mu))$. With a bank angle limit of $\mu \leq \mu_{max}$ (where $\mu_{max} = 30°$ for passenger safety and comfort in commercial aircraft [50]), the minimum turn radius R_{min} is

$$R_{min} = \frac{v_{sp}^2}{g\tan(\mu_{max})}. \tag{3.25}$$

In order to have the CRP use less lateral area, and to satisfy the banked angle turn limit from Eq. (3.25), the turn radius for paths $R_{1,1}$ and $R_{1,2}$ are

$$R_1 = R_{min1} = \frac{v_1^2}{g\tan(\mu_{max})} \tag{3.26}$$

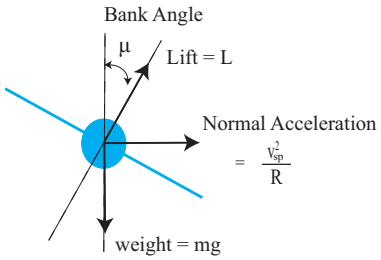

Figure 3.8: Free body of aircraft performing a banked turn; the restriction on the acceptable bank angle limits the maximum heading-change rate.

while the turn radius for paths $R_{2,1}$ and $R_{2,2}$ are

$$R_2 = R_{min2} = \frac{v_2^2}{g \tan(\mu_{max})}. \tag{3.27}$$

The turn angles are determined by the lateral distance of the split intersection H_1 and H_2, and the lateral path length differences Δ_1 and Δ_2 from Fig. 3.7. From the intersection distance d_{p3p4} and d_{p3p8} (from Lemma 10), the intersection angle θ_{int}, and by geometry, the lateral path length differences Δ_1 and Δ_2 are

$$\Delta_1 = (\overline{D}_2 + \frac{\Delta_1}{\alpha}) \cos\theta_{int} \tag{3.28}$$

$$\Delta_2 = (\overline{D}_1 + \alpha\Delta_2) \cos\theta_{int}. \tag{3.29}$$

Rearranging Eq. 3.28 and 3.29 yields

$$\Delta_1 = \frac{\overline{D}_2 \cos\theta_{int}}{1 - \frac{\cos\theta_{int}}{\alpha}} \tag{3.30}$$

$$\Delta_2 = \frac{\overline{D}_1 \cos\theta_{int}}{1 - \alpha\cos\theta_{int}}. \tag{3.31}$$

Additionally, the turn angles are derived from the following split path heights:

$$H_1 = (\overline{D}_2 + \frac{\Delta_1}{\alpha}) \sin\theta_{int} = 4R_1(1 - \cos\phi_1). \tag{3.32}$$

$$H_2 = (\overline{D}_1 + \alpha\Delta_2) \sin\theta_{int} = 4R_2(1 - \cos\phi_2) \tag{3.33}$$

Rearranging Eq. 3.32 and 3.33 yields the turn angles as

$$\phi_1 = \arccos\left(1 - \frac{(\overline{D}_2 + \frac{\Delta_1}{\alpha})\sin\theta_{int}}{4R_1}\right) \tag{3.34}$$

$$\phi_2 = \arccos\left(1 - \frac{(\overline{D}_1 + \alpha\Delta_2)\sin\theta_{int}}{4R_2}\right). \tag{3.35}$$

Bounds on Arrival Spacing for the CRP Turn Paths

The arrival spacing requirement also needs to satisfy the turn path requirement of the CRP. With the turn path parameters $(R_1, R_2, \phi_1, \phi_2)$ obtained from the previous section, the arrival spacing requirement from Eq. 3.40 needs to be satisfied from both route 1 and route 2. The arrival spacing for each route as described above is $\overline{D}_1 = \alpha\overline{D}_2 = 9.36$NM for route 1 and $\overline{D}_2 = \overline{D}_{min} = 9.23$NM for route 2. With the given arrival spacing for each route, and the known value of the minimum separation distance $D_{sep} = 5$NM, the required arrival spacing distances $\overline{D}_\phi$ (from Eq. 3.40) are 5.15NM for both route 1 and route 2 with $R_1 = 4.95$NM, $R_2 = 4.82$NM, $\phi_1 = 78.25°$,$\phi_2 = 81.52°$. Therefore, the arrival spacing for each route are greater than the turn requirement arrival spacing and therefore, satisfies the conflict-free condition for the turn paths as per Lemma 9.

CRP Intersection Path and Timing Procedure

As discussed from the timing procedure in Section 3.3, the path lengths at the intersection are from the conditions listed in Lemma 10 with $\theta_{int} = 60°$. With the given arrival flow spacing from each route ($\overline{D}_1$ and $\overline{D}_2$), the turn radius for each turn path (R_1 and R_2), the turn angles (ϕ_1 and ϕ_2) and the lateral path length differences (Δ_1 and Δ_2), the CRP intersection paths from the conditions listed in Lemma 10 are as the following:

1. The node distance $d_{p13p3} = \overline{D}_2 = 9.36$NM.

2. The node distance $d_{p12p4} = \overline{D}_2 + \Delta_2 = 18.72$NM.

3. The node distance $d_{p2p3} = 2\alpha R_2\phi_2 + \overline{D}_1 - 2R_1\phi_1 = 9.73$NM.

4. The node distance $d_{p7p8} = d_{p2p3} + \Delta_1 = 18.84$NM.

5. The node distance $d_{p3p4} = d_{p8p9} = \overline{D}_1 + \alpha\Delta_2 = 18.97$NM.

6. The node distance $d_{p3p8} = d_{p4p9} = \overline{D}_2 + \frac{\Delta_1}{\alpha} = 18.21$NM.

In addition, due to the decoupling property of the CRP where aircaft after passing the final-way point (p_6 for route 1 and p_{16} for route 2) maintain the original spacing distance ($\overline{D}_1$ for route 1 and $\overline{D}_2$ for route 2), the total travel distance of each split path have to be the same (i.e., total travel length of CRP path $R_{1,1}$ is the same as $R_{1,2}$, and the travel length of CRP path $R_{2,1}$ is the same as $R_{2,2}$). Therefore, the path lengths d_{p2p3} (distance between p_2 and p_3) and d_{p9p10} (distance between p_9 and p_{10}) are the same as well as path lengths d_{p7p8} (distance between p_7 and p_8) and d_{p4p5} (distance between p_4 and p_5) for split paths $R_{1,1}$ and $R_{1,2}$. In addition, the turn paths from route 1 are all symmetric (i.e., paths from p_1 to p_2 and to p_7 are the same as well as paths from p_5 to p_6 and p_{10} to p_6 are also the same). For the CRP paths $R_{2,1}$ and $R_{2,2}$, the path lengths d_{p13p3} and d_{p9p14} are the same as well as path lengths d_{p12p4} and d_{p8p15} are the same. In addition, the turn paths from route 2 are all symmetric (i.e., paths from p_{11} to p_{12} and to p_{13} are the same as well as paths from p_{15} to p_{16} and p_{14} to p_{16} are also the same). With the obtained CRP path lengths that satisfy the timing procedure, and with aircraft following the given path assignment from Definition 4, the non-perpendicular conflict-free CRP design is complete.

3.4.2 Example CRPs on Compound Intersections

An example is presented where local CRPs are applied on compound intersections as shown in Fig. 3.9 to illustrate the decoupling property of the CRP and the comparison

of the local CRPs designed from chapter 3 (i.e, CRP-B from Fig. 3.9(b) where the intersection angle is perpendicular, and the speed of each route are the same) and current work (i.e, CRP-A and CRP-B from Fig. 3.9(b) where the intersection angle is non-perpendicular and each route have same or different speeds) from this paper.

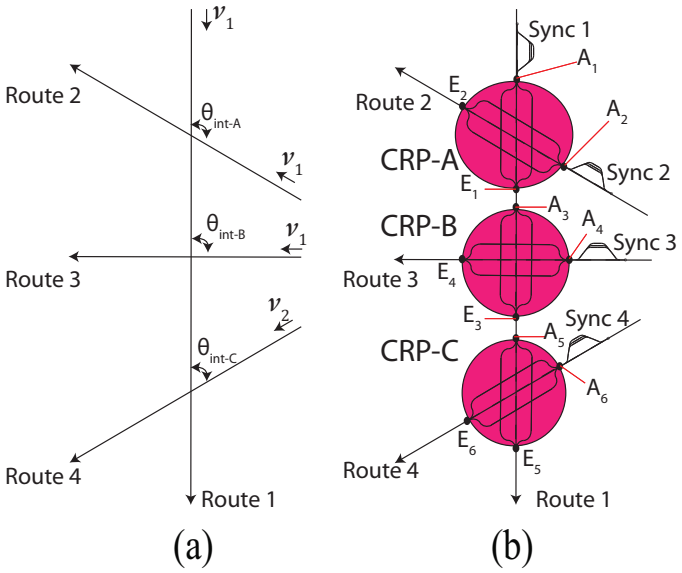

Figure 3.9: (a) Multiple routes (route 2, 3, and 4) intersecting with route 1. (b) Local CRPs are applied at each intersection (CRP-A between route 1 and 2, CRP-B between route 1 and 3, CRP-C between route 1 and 4).

CRP intersection properties

The data from the Cleveland sector ZOB 59 mentioned previously are utilized to demonstrate various local CRPs in this example.

CRP-A: Intersection between route 1 and route 2 CRP-A is applied on the intersection between route 1 and route 2 from Fig. 3.9(a). CRP-A represents the case with route 1 and route 2 having the same aircraft speed on a non-perpendicular intersection. For this example, aircraft speed for route 1 and route 2 are both $v_1 = 0.123$

Nautical Miles (NM) per second. The intersection angle between route 1 and route 2 is $\theta_{int-A} = 110°$ (in Fig. 3.9(a)). Since the intersection angle θ_{int-A} is greater than 90°, the paths and geometry of CRP-A is designed based on Lemma 11 with $\alpha = 1$ (same speeds), turn radius R_1 (from Eq. 3.27), and turn angle ϕ (derived similarly from the local CRP presented in the previous subsection). With all the geometric parameters, and given a minimal spacing distance of $\overline{D} = \overline{D}_1 = \overline{D}_2 = 9.23$ Nautical Miles (NM) for each route (since each route has the same aircraft speed) that satisfies the spacing requirements from Eqs. (3.5) and (3.40) for all routes such that local CRP-A would be conflict free.

CRP-B: Intersection between route 1 and route 3 CRP-B is applied on the intersection between route 1 and route 3 from Fig. 3.9(a). CRP-B represents the case with route 1 and route 3 having the same aircraft speed at a perpendicular intersection. For this example, data from Section 2.5 is used where aircraft speed for route 1 and route 2 are both $v_1 = 0.123$ Nautical Miles (NM) per second. The intersection angle between route 1 and route 3 is $\theta_{int-B} = 90°$ (in Fig. 3.9(a)). Since the intersection angle θ_{int-B} is 90°, the paths and geometry of CRP-B is designed based on Section 2.5, given a minimal spacing distance of $\overline{D} = \overline{D}_2 = 9.23$ Nautical Miles (NM) for each route (since each route has the same aircraft speed) and with turn radius R, and turn angle ϕ chosen from Section 3.5.2 for all routes such that local CRP-B would be conflict free.

CRP-C: Intersection between route 1 and route 4 CRP-C is applied on the intersection between route 1 and route 4 from Fig. 3.9(a). CRP-B represents the case with route 1 and route 4 having different aircraft speed at a non-perpendicular intersection. For this example, data from Section 3.4.1 is used where aircraft speed for route 1 is $v_1 = 0.123$ Nautical Miles (NM) per second and aircraft speed for route 4 is $v_2 = 0.121$ Nautical Miles (NM) per second. The intersection angle between route 1 and route 3 is $\theta_{int-C} = 60°$ (in Fig. 3.9(a)). Since the intersection angle θ_{int-C} is smaller than 90°,

the paths and geometry of CRP-C is designed the same as the previous subsection of the local CRP example presented in Section 3.4.1 which is conflict free.

Conflict-free Compound Intersections

The global section with CRPs applied on each intersection shown in Fig. 3.9(b) is conflict-free provided: (i) the local CRPs of each intersection are conflict free (i.e, CRP-A, CRP-B, and CRP-C as shown above); (ii) aircraft route entering the CRP is synchronized to the intersecting incoming route to the CRP to apply the cyclic path assignment procedure given in Definition 4; and (iii) the synchronization path is conflict-free. For the example in Fig. 3.9(b), the synchronization for route 1 and route 2 for CRP-A are performed by path Sync 1 for route 1 and path Sync 2 for route 2 to ensure aircraft arrivals at each route's CRP initial-way points (point A_1 for route 1 and A_2 for route 2 at the scheduled time of arrival (STA) to apply the cyclic path assignment procedure given in Definition 4 for CRP-A. Similarly, the synchronization for route 1 and route 3 for CRP-B are performed by path Sync 3 for route 3. In this case, route 1 does not need an additional synchronization path between CRP-A and CRP-B since aircraft on route 1 after CRP-A are already synchronized in aircraft spacing. The main synchronization is needed for the spacing and timing for route 3 which would be done in synchronization path Sync 3 such that the STA at point A_3 for route 1 and point A_4 for route 3 on the precise STA for the cyclic path assignment procedure given in Definition 2 for CRP-B. Finally, the synchronization for route 1 and route 4 follow similar arguments such that synchronization path Sync 4 for route 4 would have aircraft precise STA at point A_5 for route 1 and A_6 for route 4 for the cyclic path assignment procedure given in Definition 4 for CRP-C. A general synchronization path design and how it is conflict-free is shown below.

Conflict-free Synchronization Path

An example synchronization path design and process is presented from synchronization path Sync 1 from Fig. 3.9(b) in this section. Given an expected arrival time (ETA) at time t at the initial-way point (point A_1 for CRP-A on route 1), the synchronization procedure assigns an STA time t_k (from Eq. (3.3)) to the initial-way point. In particular, the STA t_k is chosen to be the closest discrete integer time instant to ETA t with integer k such that

$$ETA\ at\ A_1\ \in [t_k - t_{0.5}, t_k + t_{0.5}) \rightarrow STA\ =\ t_k \tag{3.36}$$

where $t_{0.5}$ is the aircraft travel time for traveling half of the minimum arrival spacing (i.e, $\overline{D}/2$). The potential STAs are separated by one discretized integer time point t_k as defined in Definition 4 Eq. (3.3) such that the arrival time only needs to be adjusted by a maximum of $t_{0.5}$ discrete time from Eq. (3.37). Towards this arrival time adjustment, the distance traveled by an aircraft needs to be changed by a maximum of $\pm\overline{D}/2$ from the nominal travel distance during synchronization to arrive at the initial way point at the correct STA. The following synchronization procedure achieves the STA by using offset maneuvers. The nominal path length (between point S_1 to point A_1 on path Sync 1) in Fig. 3.10 is changed by δ_x, where

$$\delta_x\ =\ (STA - ETA)v_{sp} \tag{3.37}$$

where v_{sp} is the speed of the aircraft (for Sync 1, $v_{sp} = v_1$).

The increase in path length by changing the second turn point of the synchronization path in Fig. 3.10 at point x_1 through points x_3, x_4, x_5, x_6, and x_2 to reach point S_9 compared to the second turn path to be at point S_4 to reach point S_9 through points S_{13}, S_{14}, S_{15}, and S_{16} is given as

$$\begin{aligned} \Delta_{turn} = \\ 2(x + R_s\phi_s + (x - \Delta x)\cos\phi_s) - 2(R_s\phi_s + \Delta x\cos\phi_s) = \\ 2x(1 - \cos\phi_s)\ = \delta_x. \end{aligned} \tag{3.38}$$

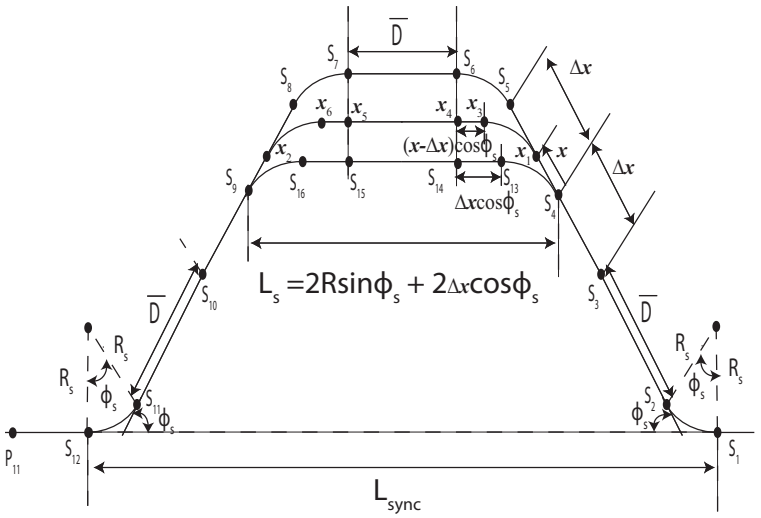

Figure 3.10: Synchronization procedure and path design example of Sync 1 from Fig. 3.9(b). Distance x between waypoints s_4 and x_1 is given by Eq. (3.39). The maximum value Δx of x corresponds to $\delta_x = \overline{D}/2$. All turn paths have the same turn radii R_s and turn angle ϕ_s. Distance d_{s13s14} (between way points S_{13} and S_{14}) and d_{x3x4} (between way points x_3 and x_4) is found by geometry of the turn path where the lateral distance L_s between waypoint S_4 and S_9 are the same for all aircraft paths.

Re-arranging Eq. (3.38), the distance x from waypoint S_4 in Fig. 3.10 to the second turn path at point x_1 is given by

$$x \;=\; \frac{\delta_x}{2[1-\cos\phi_s]}. \tag{3.39}$$

Thus, the synchronization procedure with δ_x from Eq. (3.37) achieves the time difference between STA and ETA at initial way point A_1. The synchronization path itself is conflict-free given that the chosen turn radius R_s and turn angle ϕ_s used in the turns of the synchronization satisfy the single turn requirement of

$$\overline{D} \;\geq\; \overline{D}_\phi \;= \frac{D_{sep} - 2R\sin(\frac{\phi}{2})}{\cos(\frac{\phi}{2})} + R\phi \tag{3.40}$$

from Lemma 1 of Chapter 2.

Two cases are considered to show that there are no conflicts during synchronization: (i) when the path changes δ_x are the different for two aircraft. For each case, the first turn path from S_1 to S_2 has a decoupling straight segment of length $\overline{D}$ and the final turn path between S_{11} and S_{12} also has a decoupling straight segment of length $\overline{D}$ such that the single turn requirement from Eq. (3.40) is sufficient to show that there are no conflicts at the initial and final turn paths in the synchronization path in Fig. 3.10. In addition, it is sufficient to show that there are no conflicts before the third turn onto $(\overline{S_8 S_{11}})$ because, by symmetry, reversing the flow would lead to no conflicts until the second turns from $(\overline{S_3 S_5})$ and with a minimum spacing of $\overline{D}$. For aircraft traveling on the same second turn path (i.e., same x; all aircraft travel through points $x_1, x_3, x_4, x_5, x_6, x_2$ from Fig. 1), there are no conflicts because the single turn spacing requirement from Eq. (3.40) ensures that there are no conflicts on the turn between points x_1 and x_3.

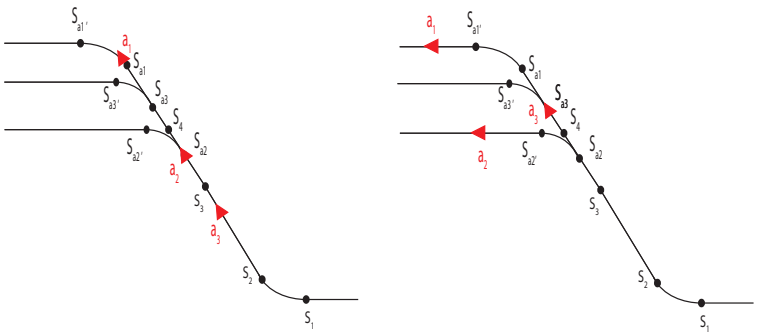

Figure 3.11: Example of Synchronization path of possible conflict between aircraft between a_2 and a_3. Aircraft a_2 performs the second turn at S_{a2} and aircraft a_3 performs its second turn at S_{a3}. (a) When aircraft a_2 is at S_{a2}. (b) When aircraft a_2 has passed $S_{a2}\prime$ and aircraft a_3 has passed S_{a2}.

The issue is to show that there are no conflicts between aircraft traveling on a different turn path for its second turn on the synchronization path as shown in

Fig. 3.11, e.g., between aircraft a_1 and a_2 or aircraft a_2 and a_3. Aircraft a_2 and a_3 were separated by at least $\overline{D}$ when aircraft a_2 is at S_{a2} as in Fig. 3.11(a). As all aircraft advance up to the point where aircraft a_3 has arrived at S_{a2}, aircraft a_2 and a_3 would be conflict free under the single turn requirement from Eq. (3.40. When aircraft a_3 has passed S_{a2}, (as in Fig. 3.11(b)), the relative motion of aircraft a_2 with respect to a_3 is in the 1st quadrant direction in the directional orientation shown in Fig. 3.11 (since both aircraft a_2 and a_3 have the same speed magnitude) where the distance between a_2 and a_3 increases until a_3 reaches $S_{a3}\prime$, and depending on the second turn path length, if both a_2 and a_3 are on its respective turn path, both aircraft turn at the same rate such that the distance between a_2 and a_3 would not change and thus a_2 and a_3 is conflict free.

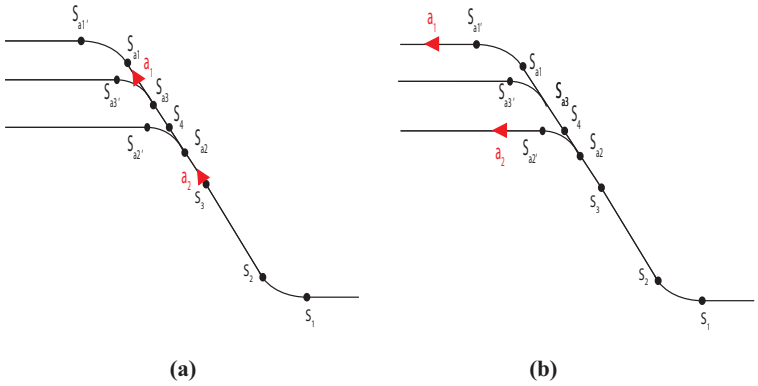

Figure 3.12: Example of Synchronization path of possible conflict between aircraft a_1,and a_2. Aircraft a_1 performs the second turn at S_{a1}, aircraft a_2 performs the second turn at S_{a2}. (a) When aircraft a_1 is at a closer distance to second turn point S_{a1} compared to the distance of aircraft a_2 to S_{a2}. (b) When aircraft a_1 and a_2 have completed travel on the second turn path.

For the case between aircraft a_1 and a_2 follow the similar arguments from the case between a_2 and a_3. Consider the case as shown in Fig. 3.12 where aircraft a_1 is closer to the initial point S_{a1} to start the second turn in the synchronization path than aircraft $a_2\prime$s distance to its second turn point S_{a2}. The opposite case of when

aircraft a_2's distance to point S_{a2} being closer than aircraft a_1's distance to its second turn point S_{a1} follows a similar argument. As shown in Fig. 3.12(a), when aircraft a_1 is at point S_{a1}, aircraft a_1 and a_2 are conflict free with the single turn requirement from Eq. (3.40) until aircraft a_2 has arrived at point S_{a2}. If each second turn paths are long enough (between S_{a1} and $S_{a1}\prime$ for aircraft a_1 and between S_{a2} and $S_{a2}\prime$ for aircraft a_2) such that both aircraft are on the second turn path, since each turn path have the same turn radii and each aircraft turn at the same rate, the distance between aircraft and a_2 does not change. Next, as aircraft a_1 finishes its second turn (passed point $S_{a1}\prime$) and aircraft a_2 is still on the second turn path, (as in Fig. 3.12(b)), the relative motion of aircraft a_2 with respect to a_1 is in the 1st quadrant direction (since both aircraft a_1 and a_2 have the same speed magnitude) where the distance between a_1 and a_2 increases until a_2 reaches $S_{a2}\prime$. Finally both aircraft a_1 and a_2 are traveling parallel after both aircraft passed the second turn path, and therefore, aircraft a_1 and a_2 are conflict free.

Remark 17. *The lateral area used from the synchronization procedure L_{sync} (from Fig. 3.10) can be quantified with respect to the arrival spacing distance $\overline{D}$. By geometry from the synchronization path in Fig. 3.10, the lateral length for all aircraft traveling on the synchronization path are the same. The lateral path length of the synchronization path $L_{sync} = 4R_s \sin\phi_s + 2\overline{D}\cos\phi_s + \overline{D} + 4\Delta_x \cos\phi_s$, with $\Delta_x = x$ from Eq. (3.39), and $\delta_x = \overline{D}/2$. For example, the synchronization path Sync 1 (from Fig. 3.10) has a lateral path length of $L_{sync} = 97.25NM$ where the turn radius R_s is obtained from Eq. (2.7), the turn angle ϕ_s is obtained from Eq. (2.41) with the given parameters from CRP-A ($\overline{D} = \overline{D}_1 = 9.23NM$, $v_1 = 0.123NM/s$).*

3.4.3 Perpendicular CRP with Same Speeds vs Non-perpendicular CRP with Different Speeds

The previous sections presented the example for a local non-perpendicular intersection cases with different speeds and how the local CRP is applied. This example demonstrates that the constraints in speed and intersection angle can be flexible compared to work in chapter 3 where all incoming routes were fixed to the same speed and the same arrival flow spacing distance. The non-perpendicular intersecting CRP design avoids extra rerouting to make the routes intersect perpendicularly as described in Chapter 2. Moreover, the non-perpendicular CRP additionally avoids extra synchronization to have both incoming routes to have the same speed and arrival flow spacing. Also another example from the previous section shows the conditions to have multiple consecutive intersections to be conflict-free when the local CRPs are applied at each intersection with the synchronization performed prior to the local CRPs. Each local CRP represented unique features: (i)non-perpendicular intersection with same route speeds; (ii) perpendicular intersection with same speeds; and (iii) non-perpendicular intersection with different speeds. The main difference comes at the intersection minimum spacing requirement d_{int}. The minimum spacing requirement for the perpendicular intersection with same speed is

$$d_{int-ps} = 2\sqrt{2}D_{sep} \tag{3.41}$$

from Eq.(2.25) of Chapter 2. The minimum spacing requirement for the non-perpendicular intersection with same speed is

$$d_{int-nps} = \frac{2\sqrt{2-2\cos\theta_{int}}D_{sep}}{\sin\theta_{int}}. \tag{3.42}$$

Finally, the intersection spacing requirement for each route for the non-perpendicular and different speed case is shown in Eqs. (3.4) and (3.5). Analytically, the main difference between the perpendicular intersection with same speeds and the non-perpendicular intersection with same speeds is the requirement for the minimum

spacing on the non-perpendicular intersection with same speeds $d_{int-nps}$ at the intersection is affected by the intersection angle θ_{int} ($\theta_{int} = 90^\circ$ in Eq. (3.42) yields Eq. (3.41)). Similarly, the main difference between the non-perpendicular intersection with same speeds and the non-perpendicular intersection with different speeds is the requirement for the minimum spacing requirement for the non-perpendicular intersection with different speeds d_{int1} and d_{int2} from Eqs. (3.4) and (3.5) where each requirement is affected by both the speed ratio between the incoming routes α and the intersection angle θ_{int} (α =1 for Eqs. (3.4) and (3.5) yields Eq.(3.42)).

3.5 Conclusion

The current chapter removed two current limitations in recently developed provably-safe CRP by: (i) developing procedures for non-perpendicular intersections; and (ii) removing the requirement that aircraft on each intersecting route have the same speed. Conditions were developed to guarantee conflict resolution, and the CRP was illustrated with an example enroute intersection in the Cleveland sector.

Chapter 4

ON-DEMAND CONFLICT RESOLUTION PROCEDURES FOR AIR TRAFFIC INTERSECTIONS

4.1 Introduction

This chapter develops a provably-safe, on-demand Conflict Resolution Procedure (CRP) for intersecting routes in en-route Air Traffic Control (ATC). The provably-safe CRP design in Chapter 2 is always-on — even in the absence of conflicts at the intersection, which would be acceptable if conflicts occur frequently at the intersection. A problem is that such an always-on CRP leads to unwanted CRP maneuvers (when there are no conflicts at the intersection) resulting in increased travel time, travel distance and required fuel, and is not efficient for intersections where conflicts are rare, e.g., because one of the routes only has intermittent traffic. The current chapter removes the inefficiency of always-on CRPs by developing provably-safe CRPs that can be activated on-demand (when conflicts appear) to accommodate an impending conflict. The CRP is then deactivated when not needed. Conditions are developed to avoid conflict for not only during conflict resolution but also during activation and deactivation of the CRP. The proposed on-demand CRP is illustrated through an example based on a route-intersection in the Cleveland sector. When conflicts are rare, the proposed on-demand CRP can lead to substantial performance improvements — only four aircraft out of the fifty four aircraft (in the example) need to use the CRP (with the on-demand CRP) as opposed to all fifty four aircraft requiring the CRP with the always-on CRP. To improve efficiency, the current chapter aims to only activate the CRPs when needed. This chapter expands on previous preliminary results in Chapter 2 by developing and providing detailed proofs of the main results and

illustrating issues in the applicability of the on-demand CRP.

4.2 CRP Background

This section provide background information on the CRP used in this chapter.

4.2.1 The Conflict Resolution Procedure

As in Chapter 2 and 3, the routes are assumed to be at a fixed altitude (planar flight) with perpendicular intersections, as illustrated in Fig 4.1. Moreover, the intersections are assumed to be spatially sparse leading to a sufficiently-large local region L, around each intersection (conflict point CP), where the local region L is conflict free from all other routes and other CPs in the airspace. The CRP can use this local region L to resolve conflicts at the intersection without potentially causing additional conflicts as long as the route modifications due to CRP procedures are contained within the local region L. Aircraft along the nominal routes (R_1 and R_2) arrive into this local region L at arrival points A_1, A_2 with a fixed nominal speed v_{sp} and exit at E_1, E_2 as shown in Fig. 4.1. It is assumed that aircrafts, arriving at the local region L are separated by at least distance $\overline{D}$ at the arrival points A_1, A_2, where the minimal arrival spacing $\overline{D}$ is greater than the minimum required separation distance D_{sep} to avoid conflicts.

Remark 18. *The current chapter addresses perpendicular intersections. However, the main concept of splitting the route to increase aircraft spacing, followed by a merge, could be extended to non-perpendicular intersections (e.g., as in Lemma 5 of [3]). Moreover, the current CRP can be applied to non-perpendicular cases, if the routes can be re-arranged to create a perpendicular intersection (e.g., as in [3], Section IV) . This might not be possible, if the intersecting routes are close to being parallel in which case a redesign of the overall route structure might be needed.*

Remark 19. *In general, conflict resolution can be achieved using maneuvers that change the heading, speed and altitude. However, heading changes are preferred over*

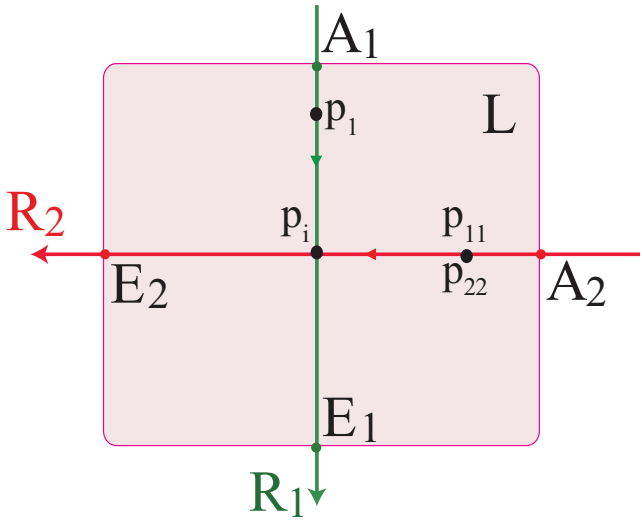

Figure 4.1: No-CRP case. Local region L of the airspace around two perpendicularly intersecting routes, R_1, R_2, the corresponding arrival points A_1, A_2 into region L, and exit points E_1, E_2 from region L. The initial waypoint for route R_1 is point p_1 and the initial waypoint for route R_2 is (i) p_{11} for a 2-path CRP; (ii) p_{22} for a 3-path CRP.

speed changes, which require additional fuel for accelerating and decelerating the aircraft. Similarly, heading changes are preferred over altitude changes, which also tend to require additional fuel, and can cause conflicts in the other altitudes.

4.2.2 Conditions for Decoupled CRP

The conditions from Definition 1 of Chapter 2 enable decoupled CRP designs for multiple en-route conflicts.

4.2.3 Path-Split CRP

An example CRP (with a two-path split) is shown in Fig. 4.2. The CRP consists of splitting of each route (R_1, R_2) into two equal-length paths (diverge maneuver) and then choosing one of the paths for each arriving aircraft. The aircraft in each path is merged back to its original route after the intersection.

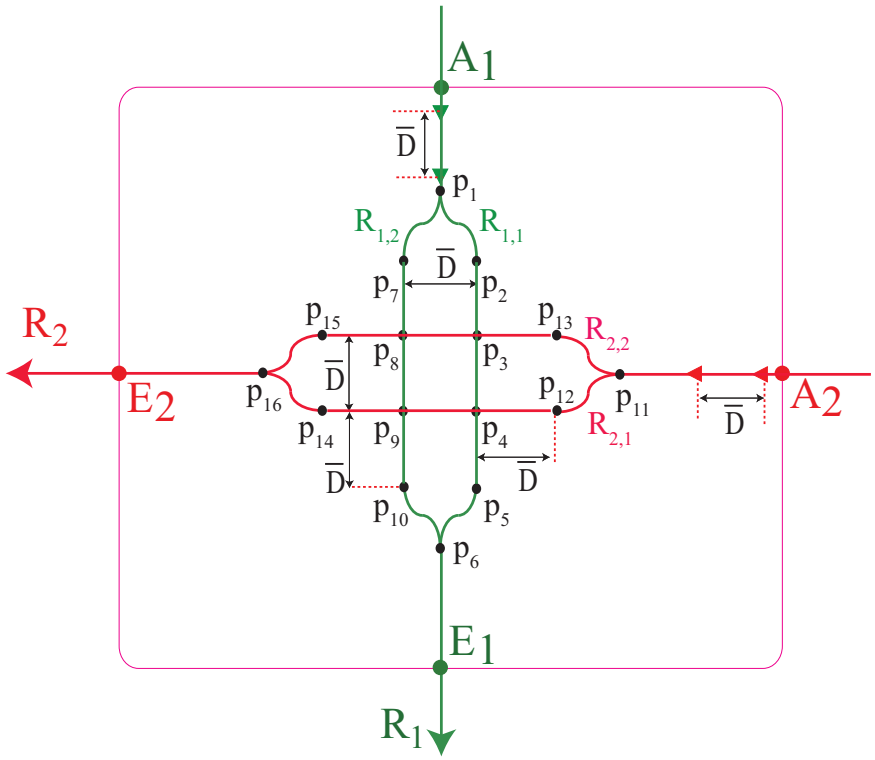

Figure 4.2: The CRP splits aircraft in each route (using diverge procedures) into multiple paths — with increased spacing between aircraft in each path. Aircraft in these paths (with sufficiently large spacing between aircraft) can then pass through the intersection without conflicts as shown in [3]. After the intersections, aircraft in the different paths are merged back to the original routes.

The two paths $\{R_{1,1}, R_{1,2}\}$ for route R_1 (shown in Fig. 4.2) are described by a set of waypoints (p_{index}):

$$\begin{aligned} R_{1,1} &= \{p_1, p_2, p_3, p_4, p_5, p_6\} \\ R_{1,2} &= \{p_1, p_7, p_8, p_9, p_{10}, p_6\} \end{aligned} \tag{4.1}$$

and the two paths $\{R_{2,1}, R_{2,2}\}$ for route R_2 are

$$\begin{aligned} R_{2,1} &= \{p_{11}, p_{12}, p_4, p_9, p_{14}, p_{16}\} \\ R_{2,2} &= \{p_{11}, p_{13}, p_3, p_8, p_{15}, p_{16}\}\,. \end{aligned} \tag{4.2}$$

Definition 5. *[*Synchronized Arrival*] The scheduled time of arrival (STA) of aircraft at the initial way-points (p_1 for route R_1 and p_{11} for route R_2 in Fig. 4.2) are at discrete time instants*

$$t_k \quad = k\left(\frac{\overline{D}}{v_{sp}}\right) \quad = kT_{\overline{D}} \tag{4.3}$$

where the index k is a nonnegative integer.

Remark 20. *Synchronization procedures needed to achieve a desired STA have been well studied in the literature, e.g., to schedule arrivals at airports [32, 37, 61, 62]. Such approaches can be adapted to manage asynchronous arrivals (provided the minimum arrival spacing is at-least $\overline{D}$) and achieve synchronized STAs for the CRP as shown in Ref. [3], and is not considered in this chapter. Rather, it is assumed that the STAs are at the discrete time instants t_k as in Eq. (4.3).*

Remark 21. *The time $T_{\overline{D}}$ between two scheduled time of arrivals (STAs) on a single route, in Eq. (4.3), corresponds to the time needed to travel (with nominal speed v_{sp}) the minimum arrival separation $\overline{D}$ between aircraft in each route.*

The following cyclic path-assignment procedure is a generalization of the one in Chapter 2 to enable on-demand CRP activation and deactivation.

Definition 6. *[*Cyclic Path-Assignment Procedure*] If the path assignments at time instant t_k are $R_{1,1}$ and $R_{2,1}$ (for routes R_1 and R_2, respectively as in Fig. 4.3a) then the path assignments at the next time instant t_{k+1} are $R_{1,2}$ and $R_{2,2}$ (for routes R_1 and R_2, respectively as in Fig. 4.3b). Similarly, assign paths $R_{1,2}$ and $R_{2,2}$ for arrivals at time instant t_{k+1} if the current paths, for arrivals at time instant t_k, are $R_{1,1}$ and $R_{2,1}$.*

Remark 22. *At the initial time instant t_0 (without aircraft arrival at the previous time instant), the cyclic path-assignment is initialized as $R_{1,1}$ and $R_{2,1}$.*

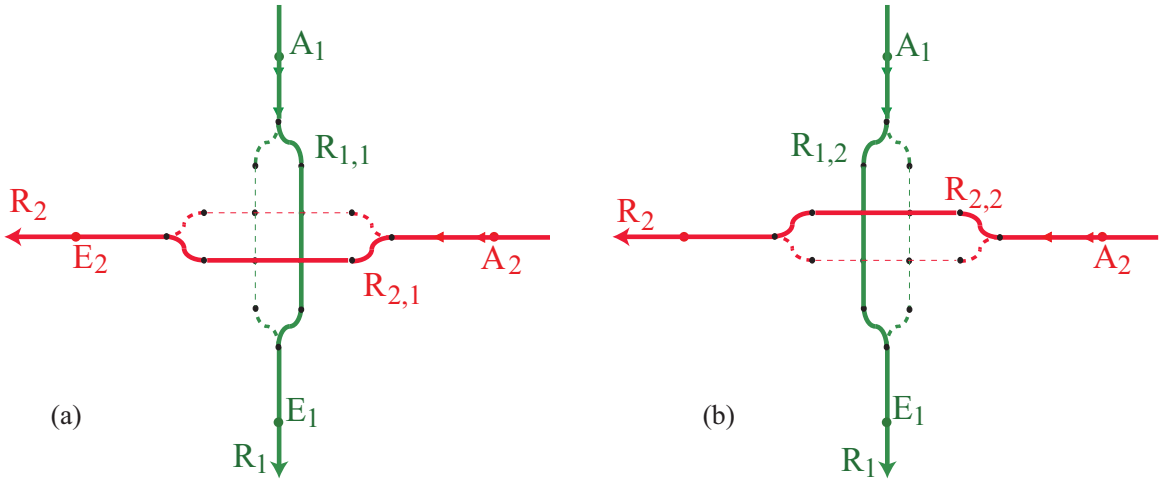

Figure 4.3: Cyclic path-assignment procedure alternates between paths $R_{1,2}$ and $R_{2,2}$ (plot a, solid lines), and paths $R_{1,2}$ and $R_{2,2}$ (plot b, solid lines).

4.2.4 CRPs are Decoupled and Conflict Free

The path-split CRP satisfies decoupling conditions and can be designed to be provably safe, as shown in [3] and in Chapter 2. Due to equal length paths, the CRP does not change the sequence of aircraft in each route and maintains a minimal separation of $\overline{D}$ (i.e., the route-flow capacity for which the CRP is designed) at the exit. Therefore, if aircraft in one of the routes (R_1 or R_2) reaches another conflict point then the CRP at the second intersection point can be designed independent of the previous CRP, provided the intersections are sufficiently separated from each other, i.e., the associated local regions needed for synchronization and conflict resolution are disjoint.

Sufficient separation distance allows conflict-free, perpendicular, route intersection. In particular, if

1. aircraft arrive synchronized at a perpendicular intersection with a minimal arrival spacing $\overline{D}$

$$\overline{D} \ > d_{\pi/2} \ = 2\sqrt{2}D_{sep}. \tag{4.4}$$

in each route

2. such that when an aircraft from one of the paths is at the intersection, the minimal distance D_{min} from aircraft on the other path be

$$D_{min} > 0.5d_{\pi/2} = \sqrt{2}D_{sep}, \tag{4.5}$$

e.g., as illustrated in Fig. 4.4

then, the perpendicular route intersection is conflict free from Lemma 4 in Chapter 2.

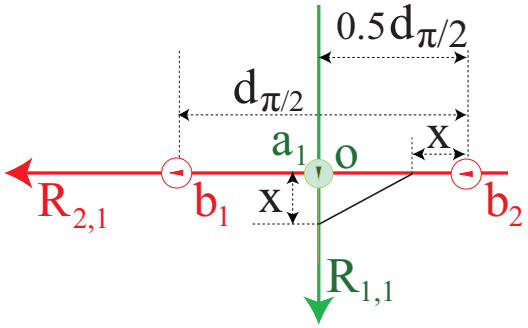

Figure 4.4: Sufficient separation distance allows conflict-free, perpendicular, route intersection as shown in, e.g., Lemma 4 in Chapter 2.

The main idea (of the proof of Lemma 4 in Chapter 2) is that sufficient initial separation distance ensures that the smallest distance remains larger than minimal separation distance D_{sep}, and thus allows a conflict-free, perpendicular, route intersection. In particular, consider the case when aircraft a_1 and b_2 have advanced a distance x from the intersection as in Fig. 4.4. Then, the separation distance $d_{a1,b2}$ between aircraft a_1 (on path $R_{1,1}$) and b_2 (on path $R_{2,1}$) needs to be larger than the minimal

separation distance D_{sep} to avoid conflict, i.e.,

$$d_{a1,b2}^2 \;=\; (\tfrac{d_{\pi/2}}{2} - x)^2 + x^2 \;\geq \tfrac{d_{\pi/2}^2}{8} \;\geq D_{sep}^2, \tag{4.6}$$

which leads to the condition in Eq. (4.5).

If the minimal arrival spacing $\overline{D}$ is smaller than the required spacing for conflict-free perpendicular intersection (in Eq. 4.4), then the CRP splits aircraft in each route (using diverge procedures) into multiple paths — with increased spacing between aircraft in each path, e.g., as in Fig. 4.2. Aircraft in these paths (with sufficiently large spacing between aircraft) can then pass through the intersection without conflicts as shown in Ref. [3].

Remark 23. *The number of path splits n needed in the CRP is determined by the spacing requirement $d_{\pi/2}$ at the intersection and the minimal arrival spacing in the route $\overline{D}$, i.e.*

$$n\overline{D} \;> d_{\pi/2} = 2\sqrt{2}D_{sep}. \tag{4.7}$$

Remark 24. *The 2-path CRP in Fig. 4.2 can be designed to be conflict free if the minimal arrival spacing $\overline{D}$ satisfies Eq. (4.7) with $n = 2$ as shown in Chapter 2, i.e.,*

$$\overline{D} \;> \sqrt{2}D_{sep}. \tag{4.8}$$

Remark 25. *The maximum number of paths required is at-most three since the resulting aircraft spacing ($3\overline{D}$) on the paths will satisfy*

$$3\overline{D} \;\geq\; 3D_{sep} \;> 2\sqrt{2}D_{sep}. \tag{4.9}$$

4.3 Provably-safe CRP Activation/Deactivation

In this section, provably-safe on-demand activation of CRP is demonstrated, and conditions for provably-safe, CRP deactivation are developed. The section begins with the 2-path-split CRP procedure.

4.3.1 Problem: Provably-safe CRP Activation/Deactivation

The goal is to activate a provably-safe CRP when needed, i.e., when a conflict is detected. Similarly, when conflicts are not anticipated, the CRP can be deactivated and aircraft should be allowed to fly along the nominal routes as illustrated in Fig. 4.1, and referred to as the no-CRP case

The main challenge is to ensure that conflicts are not triggered during the activation and deactivation of the CRP, i.e., to develop conditions for provably-safe CRP activation and deactivation. In particular, the deactivation problem is to collapse the CRP, in the absence of conflicts, into the nominal routes without the CRP, i.e., the no-CRP case. The reverse problem is to guarantee safe, on-demand, activation of the CRP whenever a conflict arises for aircraft on the two routes.

Since the CRP maintains safety (even if the aircraft in the original routes did not have a conflict), deactivation is not critical and can be done when safe deactivation conditions are met. However, there is no such flexibility (of waiting till conditions are met) for activation of the CRP in the presence of an impending conflict (in the original routes). There should be some guarantee that the CRP can always be activated, on-demand, in a safe manner. Otherwise, the CRP needs to be always on (without deactivation). This chapter develops solutions to these two problems: (a) to design CRPs that can be activated on-demand in a provably-safe manner; and (b) to identify conditions under which the CRP can be deactivated in a safe manner.

4.3.2 No-CRP Conditions

Conflict free conditions for intersections between two routes without a CRP (no-CRP case) are developed in the following Lemma.

Lemma 12. *Consider the no-CRP case for two intersecting routes (as in Fig. 4.1) with the minimal arrival spacing $\overline{D}$ in each flow satisfying the condition in Eq. (4.8). Then, there are no conflicts (without the use of a CRP) provided, at each discrete*

time instant t_k , no more than one aircraft arrives at the two initial waypoints — either at p_1 from route R_1 or at p_{11} from route R_2 in Fig. 4.2.

Proof An aircraft in a route will not have conflicts with other aircraft in the same route since arrivals at a route's initial waypoint are separated by at-least one discrete time instant, which corresponds to an arrival spacing $\overline{D}$ larger than the minimal separation D_{sep}. Moreover, since there is only one aircraft arriving at the initial waypoints (p_1 and p_{11}, which are equidistant from the intersection point p_i), only one aircraft can occupy the intersection point p_i at any time as illustrated in Fig. 4.5. For example at the instant when an aircraft from route R_1 is at the intersection point p_i (as illustrated in Fig. 4.5a), the nearest aircraft on the other route (R_2) is at least $\overline{D} > \sqrt{2}D_{sep}$ away due to only one arrival at each discrete time instant and the minimal arrival spacing condition in Eq. (4.8). This ensures that there is sufficient spacing (as in Eqs. 4.5, 4.6) to avoid conflicts with aircraft in the other route (R_2).

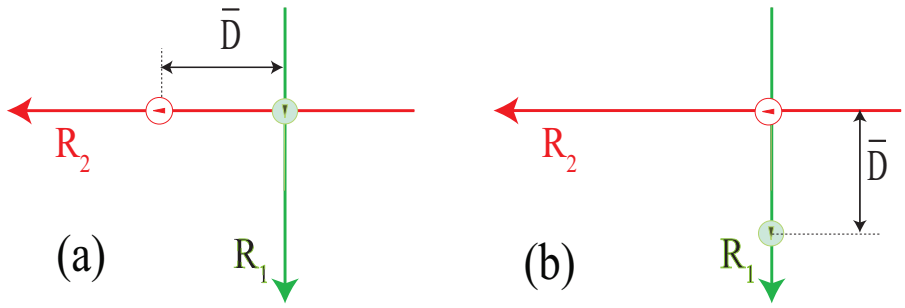

Figure 4.5: Safe no-CRP case with minimal arrival spacing $\overline{D}$ as in Eq. (4.5). Distance between aircraft is one arrival spacing $\overline{D} > \sqrt{2}D_{sep}$ when one aircraft (either from route R_1 or route R_2) is at the intersection.

4.3.3 CRP-Activation Conditions

When operating under the no-CRP scenario, if two aircraft (one from each route R_1 and R_2) arrive at the same discrete time instant t_k, then the conflict-free condition

(of no more than one arrival at each discrete time instant in Lemma 12) is violated, and a CRP needs to be activated. The following Lemma shows that the 2-path CRP (in Fig. 4.2) can be activated, on-demand, in a safe manner.

Definition 7. *[*CRP Clearance Time*] The CRP-path clearance time T_{CRP} is the time needed to travel from the initial waypoint (where the CRP path starts to deviate from the nominal route) to the final waypoint (where the CRP path returns to the nominal path) along the CRP path. The associated travel distance is defined as*

$$D_{CRP} = v_{sp} T_{CRP}. \tag{4.10}$$

For example, this would be the time needed to travel from the initial waypoint p_1 to the final waypoint p_6 along path $R_{1,1}$ or $R_{1,2}$ in Fig. 4.2.

Lemma 13. *Consider the activation, at time instant t_k, of the provably-safe CRP (illustrated in Fig. 4.2 and described in detail in Chapter 2) for the two intersecting routes (in Fig. 4.1) where the minimal arrival spacing satisfies the sufficient-spacing condition in Eq. (4.8). Furthermore, let the following CRP-activation-enabling condition be satisfied: all aircraft that arrived during the time interval $[t_k - T_{CRP}, t_k)$ satisfy the no-CRP conditions (in Lemma 12), and follow the no-CRP, nominal routes R_1 and R_2 through the intersection. Then, there are no conflicts if the CRP is activated at time instant t_k (i.e., aircraft arriving at and after time instant t_k follow the CRP paths) with the following re-initialization of the CRP path-assignment.*

1. *If there is no arrival on route R_1 at time instant t_{k-1}, then the CRP is re-initiated by assigning paths $R_{1,1}$ and $R_{2,1}$ for arrivals at time instant t_k (as illustrated in Fig. 4.6).*

2. *Otherwise, if there is no arrival on route R_2 at time instant t_{k-1}, then the CRP is re-initiated by assigning paths $R_{1,2}$ and $R_{2,2}$ for arrivals at time instant t_k .*

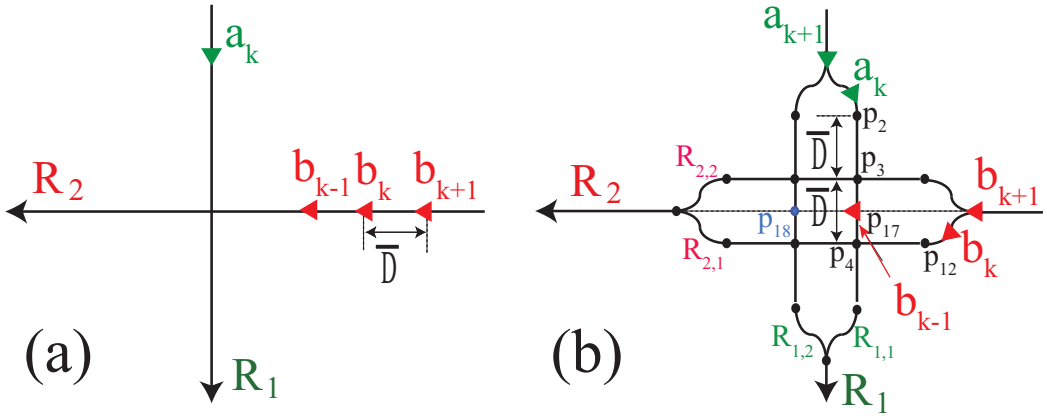

Figure 4.6: Example 2-path CRP activation. (a) Without CRP, aircraft arriving at time instant t_k have conflict with each other, i.e., between aircraft a_k on route R_1 arriving at the initial waypoint p_1 and aircraft b_k arriving at the initial waypoint p_{11} on route R_2. The subscript k indicates the arrival time instant t_k. (b) 2-path CRP is applied from time instant t_k onwards, e.g., for aircraft a_k, a_{k+1}, b_k, and b_{k+1}. Aircraft arriving earlier travel on the nominal route, e.g., aircraft b_{k-1} arriving at time t_{k-1} travels on route R_2 in this example.

Proof The proof is provided for the case when route R_1 has no aircraft arriving at time instant t_{k-1} (as in Fig. 4.6). The other case when route R_2 has no aircraft, arriving at time instant t_{k-1}, follows by a similar argument. There are no conflicts between aircraft arriving (on either route) at or after time instant t_k because the CRP is conflict free. Due to the CRP-activation-enabling condition, all aircraft remaining in the CRP region arriving before time instant t_k are no-CRP aircraft — there is no potential for conflicts between CRP-following aircraft arriving before and after time instant t_k. Therefore, the proof needs to only show that there are no conflicts between no-CRP aircraft arriving before time instant t_k on a route (e.g., b_{k-1} on route R_2) and CRP-following aircraft arriving at or after time instant t_k on: (case 1) the other route, e.g., aircraft a_k on route R_1; and (case 2) the same route, e.g., aircraft b_k on route R_2. The proof is divided into these two cases.

Case 1: No conflicts between aircraft in different routes Lack of conflicts between aircraft on route R_2 arriving before time instant t_k and aircraft on route R_1 is shown below. Lack of conflicts between aircraft on route R_1 arriving before time instant t_k and those on route R_2 follow by symmetry.

Consider the time instant t when the pre-CRP aircraft b_{k-1} along route R_2 (with initial waypoint arrival time instant t_{k-1}) is at the intersection point p_{17} between the route R_2 and the path $R_{1,1}$ as shown in Fig. 4.6b. Then, at this time instant t the distance d_{b_{k-1},a_k} between aircraft b_{k-1} and the first CRP-following aircraft a_k on route R_1 is

$$d_{b_{k-1},a_k} \ \geq \ 1.5\overline{D} \ > 1.5D_{sep} \tag{4.11}$$

because: (i) the distance along the direction of route R_2 between aircraft b_{k-1} and b_k is at-least the arrival spacing $\overline{D}$ since aircraft b_{k-1} moves straight along route R_2 while the following aircraft b_k deviates from the straight path; (ii) by a similar argument, the distance along the direction of route R_1 between aircraft a_k and waypoint p_3 (at time t) is also at-least $\overline{D}$; and (iii) the distance between the paths $R_{2,2}$ and $R_{2,1}$, placed equidistant from the original route R_2, is $\overline{D}$. Then

$$d_{b_{k-1},a_k} \ > 1.5D_{sep} \ > \sqrt{2}D_{sep} \ = 0.5d_{\pi/2} \tag{4.12}$$

and there is no conflict between between aircraft b_{k-1} and a_k since the aircraft move along perpendicular paths and are separated by sufficient spacing of more than $0.5d_{\pi/2}$, as in Eqs. (4.5, 4.6).

When aircraft b_{k-1} on route R_2 reaches the next intersection point (p_{18} after traveling a distance $\overline{D}$), aircraft a_{k+1} arriving at t_{k+1} (also spaced behind aircraft a_k by distance $\overline{D}$) would be at the same distance as d_{b_{k-1},a_k} in Fig. 4.6. Hence the lack of conflict follows from sufficient spacing as in Eqs. (4.5, 4.6). Moreover, there can be no conflicts between aircraft a_k on route R_1 and those arriving on route R_2 earlier than t_{k-1} since those aircraft will be further away from aircraft a_k than aircraft b_{k-1}, when aircraft a_k arrives at waypoint p_2.

Case 2: No conflicts between aircraft in the same route The distance between aircraft b_{k-1} and b_k at time instant t_{k-1} is $\overline{D}$ when aircraft b_{k-1} is at the initial waypoint p_{11}. This distance cannot decrease because the distance along route R_2 increases when the forward aircraft b_{k-1} follows a straight line along route R_2 while the aft aircraft b_k deviates from the straight line of route R_2. Aircraft b_{k-1} cannot have conflict with aircraft arriving later than t_k because they will be farther from the aircraft b_k arriving at t_k. Thus, with the use of CRP, the distance to the no-CRP aircraft can only increase ensuring conflict avoidance.

Remark 26. *The CRP initiation in Lemma 13, e.g., assigning paths $R_{1,1}$ and $R_{2,1}$ for arrivals at time instant t_k if there is no arrival on route R_1 at time instant t_{k-1} (as in Fig. 4.6b), is important to avoid conflicts. For example, a conflict can occur at waypoint p_{18} in Fig. 4.6 between aircraft b_{k-1} and a_k if paths $R_{1,2}$ and $R_{2,2}$ were assigned instead.*

4.3.4 CRP Deactivation Conditions

The deactivation conditions are developed below. Although CRP can be deactivated when sufficient empty space is available, conflicts can arise between aircraft on the same route because the aircraft in the no-CRP case follow straight-line routes while previous aircraft (on the CRP) follow a longer route. In particular, the distance between the CRP-following aircraft and aft, no-CRP aircraft can decrease over time. Therefore, the initial distance between the CRP-following and no-CRP aircraft is required to be sufficiently large (when deactivating the CRP): (i) to ensure conflict avoidance; and (ii) to maintain at-least the minimal arrival spacing $\overline{D}$ upon exit from the CRP (to ensure decoupling of CRP [3]).

Definition 8. *[*No-CRP Clearance Time*] The no-CRP clearance time T_{noCRP} is the time needed to travel from the initial waypoint (where the CRP path starts to deviate from the nominal route) to the final waypoint (where the CRP path returns to the*

nominal path) along the nominal route (e.g., R_1 or R_2) and the associated travel distance is defined as

$$D_{noCRP} = v_{sp} T_{noCRP}. \tag{4.13}$$

Definition 9. *[*Clearance-Time Difference*] The time difference T_δ in CRP clearance when traveling along a CRP path (e.g., $R_{2,1}$ as in Definition 7) versus traveling on a straight route (e.g., R_2) is*

$$T_\delta = T_{CRP} - T_{noCRP} > 0 \tag{4.14}$$

and the associated difference in travel distance (CRP clearance-distance difference) is (from Eqs. 4.10, 4.13)

$$\delta = D_{CRP} - D_{noCRP} > 0. \tag{4.15}$$

Lemma 14. *Consider the deactivation, at time instant t_k, of the provably-safe CRP (illustrated in Fig. 4.2 and described in detail in Chapter 2) for the two intersecting routes (in Fig. 4.1) where the minimal arrival spacing satisfies the sufficient-spacing condition in Eq. (4.8). There are no conflicts during CRP deactivation (i.e., between CRP-following aircraft before time instant t_k and the no-CRP aircraft) and during potential CRP reactivation (if needed) provided the following two conditions are satisfied.*

1. *Sufficient space condition: let there be no aircraft arriving (in either route) during the clearance-time difference (T_δ in Eq. 4.14) before the initiation of CRP deactivation, i.e., the time interval $(t_{k-1} - T_\delta, t_k)$.*

2. *CRP activation-enabling condition: let aircraft arriving with the CRP deactivated, for the time interval $[t_k, t_k + T_{CRP})$, satisfy the no-CRP conditions (in Lemma 12), and follow the no-CRP, nominal routes R_1 and R_2 through the intersection.*

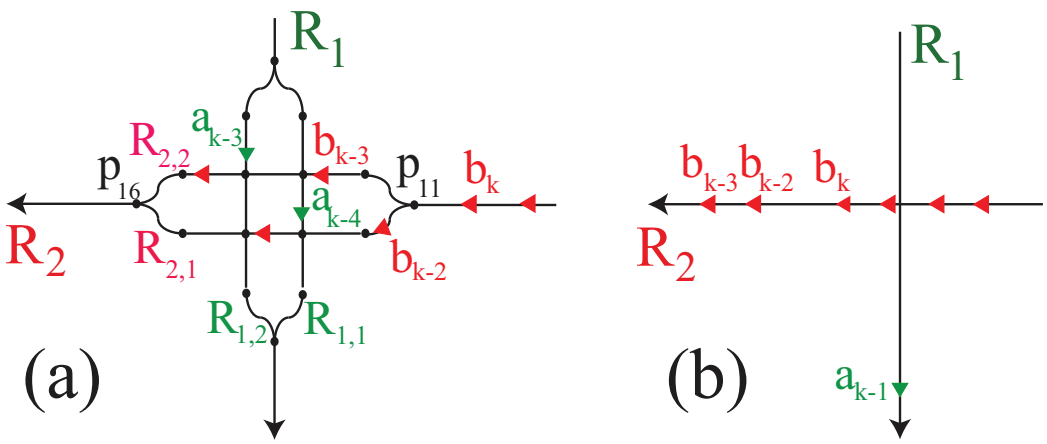

Figure 4.7: Example 2-path CRP deactivation at time instant t_k with no aircraft arriving at time instant t_{k-1}. (a) Split-path CRP is used before time instant t_k. (b) Aircraft arriving at and after time instant t_k travel on the nominal route without the CRP (e.g., b_k on R_2).

Proof As in the proof of Lemma 13, the current proof needs to only show that there are no conflicts between no-CRP-following aircraft arriving at or after time instant t_k on a route (e.g., b_k on route R_2) and CRP following aircraft arriving before time instant t_k on: (case 1) the same route, e.g., aircraft b_{k-2} on route R_2; and (case 2) the other route, e.g., aircraft a_{k-2} on route R_1. The proof is divided into these two cases.

Case 1: No conflicts between aircraft in the same route The lack of conflict follows by reversing time in the proof of Lemma 13 (Case 2) for the similar issue of conflicts between aircraft in the same route. In particular, due to the sufficient-space condition of the Lemma, the minimal space between the first no-CRP aircraft (e.g., b_k on route R_2) and a previous aircraft on the same route (say b_{k-m} with $m > 1$), when the first no-CRP aircraft (b_k) is at the final waypoint (p_{16}) at time instant t_f, is $\overline{D}$. Backwards (in time) from this time instant t_f , the distance between these aircraft (b_k, b_{k-m}) cannot decrease because the distance along route (R_2) increases (backward in time)

when the no-CRP aircraft (b_{k-m}) follows a straight line along the nominal route (R_2) while the CRP-following aircraft (b_k) deviates from the straight line of route (R_2).

Case 2: No conflicts between aircraft in different routes Note that there is no arrival at time instant t_{k-1} due to the sufficient-space condition of the Lemma and Eq. (4.14). The spacing between the last possible CRP-following aircraft b_{k-m} ($m > 1$) and the first no-CRP aircraft b_k is at-least $2\overline{D}$ when the CRP-following aircraft b_{k-m} is at the initial waypoint p_{11}. Since the final distance between b_{k-m} and b_k (when aircraft b_{k-m} is at the final waypoint p_{16}) is at least $\overline{D}$, and the only time the distance (along the route R_2 direction) can potentially decrease is when the CRP-following aircraft b_{k-m} is on the curved turns, from symmetry of the two curved-portions of the path, the distance can only reduce by a maximum of $0.5\overline{D}$ during a single curved-portion of the path. Therefore, the minimal distance (along the route R_2 direction) between aircraft b_{k-m} and b_k is greater than $1.5\overline{D}$ when aircraft b_{k-m} reaches the first intersection point, either p_3 or p_4 in Fig. 4.8. This would also be the time when the last possible CRP-following aircraft a_{k-m} ($m > 1$) on the other route is at one of the first intersection points, either p_3 or p_8 and the minimal spacing to aircraft b_k (along the route R_2 direction) is also the same — greater than $1.5\overline{D}$. Then, after each aircraft travels half the distance $\overline{D}$, the last possible CRP-following aircraft a_{k-m} is at one of the intersection points with route R_2, i.e., at waypoint p_{17} or p_{18} as illustrated in Fig. 4.8 and the minimal spacing to aircraft b_k (along the route R_2 direction) is greater than $\overline{D}$. The lack of conflict follows from sufficient spacing as in Eqs. (4.5, 4.6). The lack of conflict between aircraft a_{k-m} and b_k follows by a similar argument due to the symmetry of the CRP between the two routes.

Remark 27. *[*Activation-Enabling Condition*] The CRP activation-enabling condition in Lemma 14 (for arrivals after the CRP deactivation at time instant t_k) is not necessary for safe deactivation; however, it is needed to ensure that the CRP can be safely activated, i.e., that conditions in Lemma 13 will be met to resolve potential*

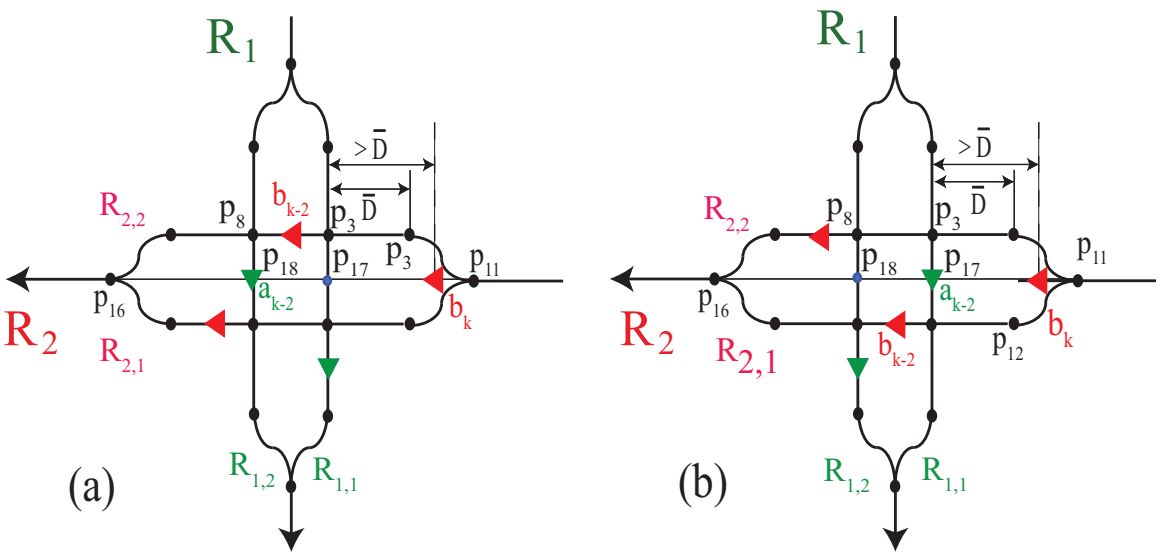

Figure 4.8: Example scenarios during 2-path CRP deactivation when aircraft are not present on route R_1 at and after time instant t_k. (a) Aircraft a_{k-2} is at the intersection (p_{18}) of path $R_{1,2}$ and route R_2. (b) Aircraft a_{k-2} is at the intersection (p_{17}) of path $R_{1,1}$ and route R_2.

conflicts for aircraft arriving after the enforced, no-CRP time interval $[t_k, t_k + T_{CRP})$. *In particular, the activation-enabling condition in Lemma 13 ensures that there is no potential for conflict between CRP-following aircraft arriving before and after CRP activation. A tighter bound on the activation-enabling condition (in Lemmas 13 and 14) can be developed for the 2-path CRP, as shown in the following Lemma.*

Lemma 15. *Consider the 2-path-split , provably-safe CRP (illustrated in Fig. 4.2 and described in detail in Chapter 2) for the two intersecting routes (in Fig. 4.1) where the minimal arrival spacing satisfies the sufficient-spacing condition in Eq. (4.8). Let the CRP clearance-distance difference* δ *in Eq. (4.15) satisfy*

$$\delta \leq m\overline{D} \quad \text{where} \quad m \geq 1 \text{ is an integer.} \tag{4.16}$$

Then, the following CRP activation and de-activation are conflict free with the CRP

active at initial time instant t_0.

1. *Deactivation at time instant t_k ($k > 1$) if: (i) there is less than two aircraft arrivals at the initial waypoints at time t_k; and (ii) there are no aircraft arrivals at the initial waypoints in the time interval (t_{k-1-m}, t_k).*

2. *Activation at time instant t_k ($k > 1$) if there are two aircraft arriving at the initial waypoints at time instant t_k using the CRP re-initialization as in Lemma 13.*

Proof The proof follows from those for Lemma 13 and Lemma 14.

Activation: The main difference is the lack of the CRP activation-enabling condition from Lemma 13 in the current Lemma 15, which can result in potential conflicts between CRP-following aircraft arriving before CRP activation, and after CRP activation at time instant t_k. For example, CRP-following aircraft arriving at of after time instant t_k cannot have conflicts with no-CRP aircraft arriving at time instant t_{k-1} from arguments similar to those in the proof of Lemma 13. Nevertheless, there could be conflict between CRP-following aircraft arriving at of after time instant t_k and CRP-following aircraft arriving before time instant t_{k-1}. Standard arguments for no conflicts between CRP-following aircraft (with the arrival spacing requirement in Eq. 4.8) cannot be used because CRP-following aircraft arriving before time instant t_k and CRP-following aircraft arriving after time instant t_k do not share the same path assignment procedure due with a single initialization (or a single re-initialization).

The lack of conflict between CRP-following aircraft arriving before and at-or-after time instant t_k can be attributed to sufficient spacing between them. In particular, due to the deactivation condition (ii) in the current Lemma 15, there is at-least one time instant without aircraft arrival before time instant t_{k-1}. Hence the nearest CRP-following aircraft arrival to time instant t_k is at time instant t_{k-3}. There can be no conflict between CRP-following aircraft on the same route arriving at time instants t_k and t_{k-3} because there is no conflict between aircraft on the same route with arrivals

at time instants t_k and t_{k-1} (due to Lemma 14) and the distance along the route they follow is larger for aircraft arriving at time instants t_k and t_{k-3} when compared to the distance along the route being followed by aircraft arriving at time instants t_k and t_{k-1}.

The other possibility is conflict between CRP-following aircraft arriving in different routes at-or-before time instant t_{k-3} and at-or-after time instant t_k. Again, sufficient spacing ensures that by the time the aircraft (with arrival at t_k) reaches intersections between two paths in the CRP, the previous CRP-following aircraft are sufficiently far away from these path intersections. For example, consider the case when the CRP is reinitialized with aircraft on paths $R_{1,1}$ and $R_{2,1}$. When a CRP-following aircraft a_k arriving at time instant t_k from route R_1 reaches the first CRP-path-intersection waypoint p_3, the CRP-following aircraft b_{k-3} arriving at time instant t_{k-3} on route R_2 has gone past waypoints p_{15} or p_{14}, and are at-least $\overline{D}$ away from the CRP paths $R_{1,1}$ and $R_{1,2}$. The lack of conflict follows from sufficient spacing as in Eqs. (4.5, 4.6).

Moreover, at this same time, aircraft arriving on route R_2 at time instant t_k will be at waypoint p_4 and the CRP-following aircraft on route R_1 (a_{k-3} arriving at time instant t_{k-3}) will be at waypoints p_5 or p_{10} which are at-least $\overline{D}$ away from the CRP paths $R_{2,1}$ and $R_{2,2}$ — the lack of conflicts follows from the intersection separation conditions represented by Eqs. (4.5, 4.6). By symmetry, the argument follows for the case when the CRP is reinitialized with aircraft on paths $R_{1,2}$ and $R_{2,2}$.

De-activation: Since the CRP activation-enabling condition is not needed to ensure safety during de-activation as clarified in Remark 27, there can be no conflicts between no-CRP aircraft arriving at time instant t_{k-1} and CRP-following aircraft arriving before time instant t_{k-1} from arguments similar to those in the proof of Lemma 14. Also, there can be no conflicts between no-CRP aircraft arriving at time instant t_{k-1} and before time instant t_{k-1} from Lemma 12 as there are no concerns with CRP re-initialization when evaluating for conflicts between no-CRP aircraft.

4.3.5 CRP with Three Path Splits

Conflict free activation and deactivation of the 3-path-split CRP (which is the maximum number of path splits needed as in Remark 25) is studied here. An example 3-path-split CRP is shown in Fig. 4.9. – it consists of splitting of each route (R_1, R_2) into three equal-length paths (diverge) and choosing one of the paths for each arriving aircraft. The aircraft in each path is then merged back into the original route after the intersection. The main difference between a 2-path-split and a 3-path-split CRP is the cyclic path assignment procedure. As a result, the on-demand activation and deactivation approach slightly differs for the 3-path-split CRP. Nevertheless, the main CRP concepts (such as diverge, intersection and converge) and approaches to avoid conflicts (during CRP activation and deactivation) remain similar.

The three paths $\{R_{1,1}, R_{1,2}, R_{1,3}\}$ for route R_1 (shown in Fig. 4.9) are described by a set of waypoints (p_{index}):

$$\begin{aligned} R_{1,1} &= \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9\} \\ R_{1,2} &= \{p_1, p_2, p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_9\} \\ R_{1,3} &= \{p_1, p_{16}, p_{17}, p_{18}, p_{19}, p_{20}, p_{21}, p_{15}, p_9\} \end{aligned} \tag{4.17}$$

and the three paths $\{R_{2,1}, R_{2,2}, R_{2,3}\}$ for route R_2 are

$$\begin{aligned} R_{2,1} &= \{p_{22}, p_{23}, p_{25}, p_6, p_{13}, p_{20}, p_{28}, p_{31}, p_{33}\} \\ R_{2,2} &= \{p_{22}, p_{24}, p_{27}, p_4, p_{11}, p_{18}, p_{30}, p_{32}, p_{33}\} \\ R_{2,3} &= \{p_{22}, p_{23}, p_{26}, p_5, p_{12}, p_{19}, p_{29}, p_{32}, p_{33}\} \end{aligned} \tag{4.18}$$

Definition 10. *[*Cyclic, Path-Assignment Procedure for 3-path CRP*] Let the scheduled time of arrival (STA) of aircraft be at discrete time instants*

$$t_k = k\left(\frac{\overline{D}/2}{v_{sp}}\right) = kT_{\overline{D}} \tag{4.19}$$

at the initial way-points (in Fig. 4.9) — p_1 on route R_1 for even values of k, and p_{22} on route R_2 for odd values of k. The path allocation rule is cyclic and repeats after

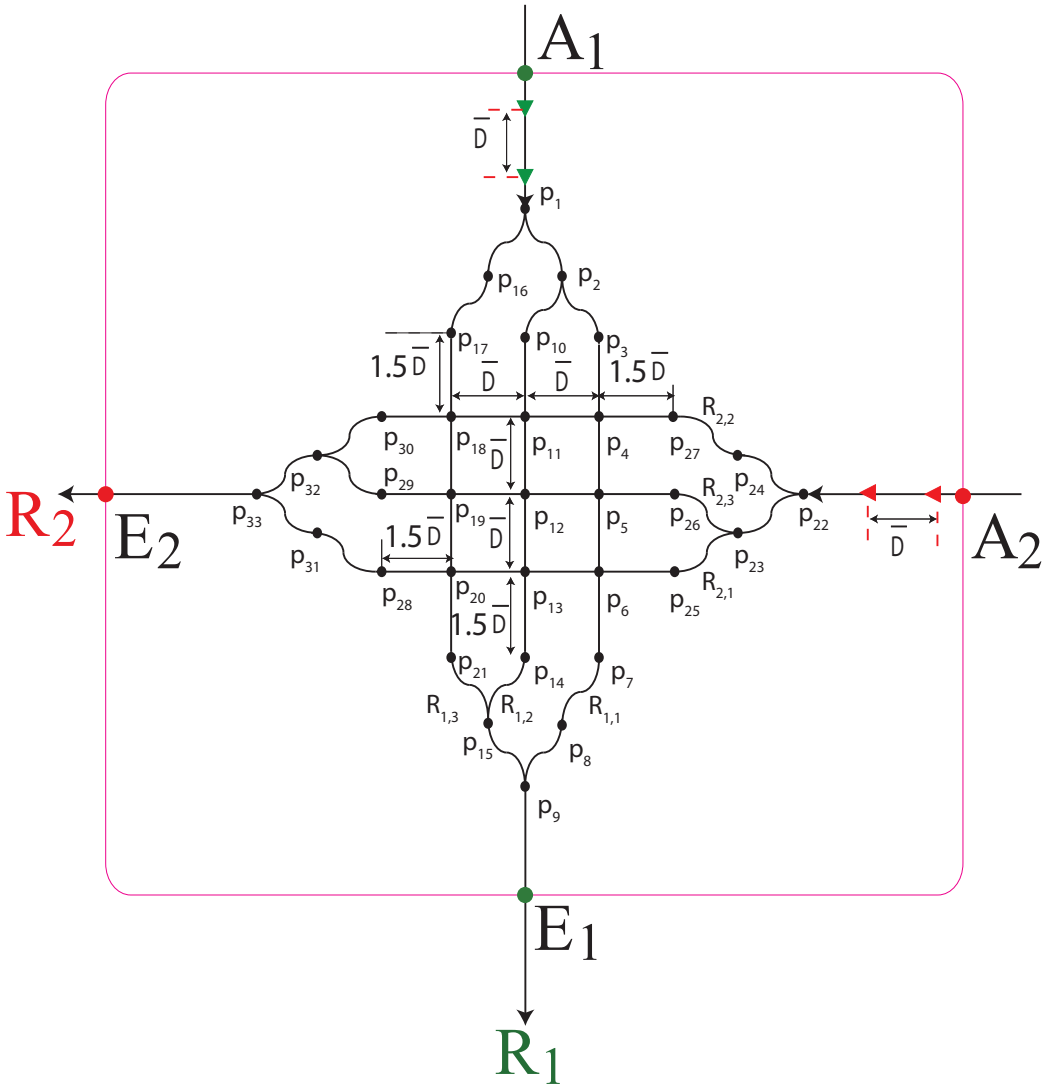

Figure 4.9: A conflict-free 3-path CRP design from [3].

every six discrete time instants as in Fig. 4.10 if the CRP stays activated. If needed, at time instant t_k, the cyclic path-assignment is initialized with CRP paths $R_{1,1}$ or $R_{2,2}$ depending on whether there is an aircraft on route R_1 (even k) or R_2 (odd k), respectively.

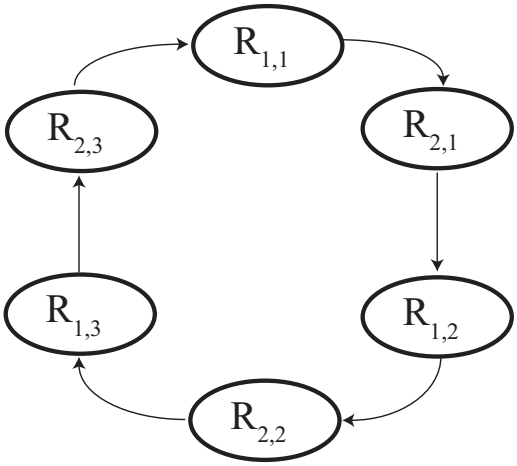

Figure 4.10: Cylic path assignment procedure for the 3-path CRP where the assigned path switches at every increase in the discrete time instant t_k .

Remark 28. *The 3-path-split CRP has an aircraft spacing of* $3\overline{D}$ *on the straight sections of its paths, as the number of splits discussed as in Remark 25. Therefore, the safe distance between aircraft, when one aircraft is at the intersection (say* p_{18} *on path* $R_{1,3}$*) and the other aircraft is on a different path (say on route* $R_{2,2}$*), is (*$1.5\overline{D}$*), which is not an integer multiple of the arrival spacing of* $\overline{D}$*. Such spacing is achieved by using: (i) separation between arrival time instants* t_k *for the 3-path case (Eq. 4.19) that is one half of the one for the 2-path case (Eq. 4.3); and (ii) an offset between the scheduled time of arrivals (STAs) of aircraft as in Definition 10 — arrivals for route* R_1 *at initial waypoint* p_1 *for even* k *and along route* R_2 *at initial waypoint* p_{22} *for odd* k*.*

Provable-safe activation and deactivation of the 3-path CRP are developed below, which is considered to be safe if always-on as shown in Chapter 2, and formally stated in the following Lemma.

Lemma 16. *[Provably-safe 3-path CRP design] Consider two intersecting routes (as*

in Fig. 4.1) with synchronized arrival of aircraft along route R_1 (at initial waypoint p_1 for even k) and along route R_2 (at initial waypoint p_{22} for odd k). Then, a 3-path-split CRP with the cyclic path-assignment procedure in Definition 10 can be designed to be provably safe (conflict free) if the arrival spacing $\overline{D}$ is larger than the minimal separation distance D_{sep}, i.e., $\overline{D} > D_{sep}$.

Proof The proof follows from Chapter 2.

No-CRP Conditions (3-path CRP)

Conflict free conditions for intersections between two routes without the CRP (no-CRP case) are developed in the following Lemma.

Lemma 17. *Consider the no-CRP case for two intersecting routes (as in Fig. 4.1) with synchronized arrival of aircraft along route R_1 (at initial waypoint p_1 for even k) and along route R_2 (at initial waypoint p_{22} for odd k). Then, there are no conflicts with previous arriving aircraft (without the use of the CRP) if whenever an aircraft from one route arrives at its initial waypoint at time instant t_k, then aircraft from the other intersecting route does not arrive at its initial waypoint at the previous time instant t_{k-1}.*

Proof The proof is similar to that of Lemma 12 for the 2-path CRP. However, since the arrival spacing $\overline{D}$ can be smaller for the 3-path CRP (compared to the 2-path CRP), the 3-path CRP requires a minimal separation of $1.5\overline{D}$ or three time-instants separation (from Eq. (4.19)) at an intersection as illustrated in Fig. 4.11 since $1.5\overline{D} > 1.5D_{sep} > \sqrt{2}D_{sep}$.

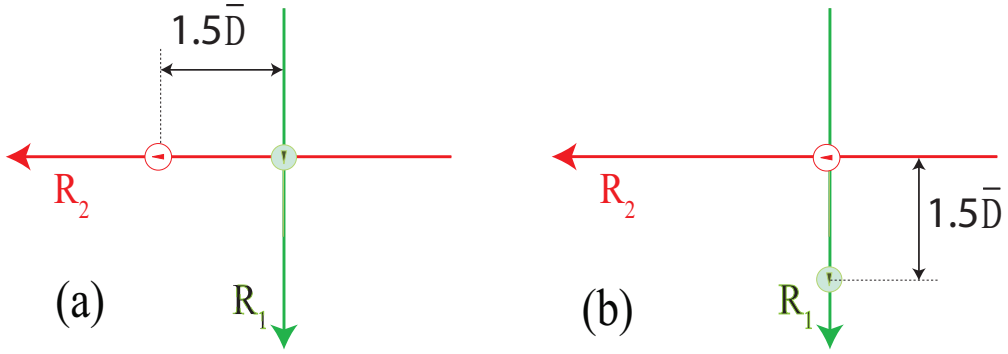

Figure 4.11: Safe no-CRP case under 3-path-CRP arrival conditions. Distance between aircraft is one and a half arrival spacing $1.5\overline{D} > \sqrt{2}D_{sep}$ when one aircraft (either from route R_1 or route R_2) is at the intersection.

Safe Activation/Deactivation for 3-path CRP

The approach for activation and deactivation is similar to the 2-path-split CRP shown in Sections 4.3.2-4.3.4, as shown in the Lemma below.

Lemma 18. *Consider a provably-safe 3-path-split CRP design for two intersecting routes (as in Fig. 4.1) with synchronized arrival of aircraft along route R_1 (at initial waypoint p_1 for even k) and along route R_2 (at initial waypoint p_{22} for odd k).*

Let the CRP clearance-distance difference δ in Eq. (4.15) satisfy

$$\delta \leq m\overline{D} \quad \text{where} \quad m \geq 1 \text{ is an integer.} \tag{4.20}$$

Then, the following CRP activation and de-activation are conflict free with the CRP active at initial time instant t_0.

1. *Deactivation at time instant t_k ($k > 1$) if there are no aircraft arrivals at the initial waypoints in the time interval $(t_{k-2(m+2)-1}, t_k)$.*

2. *Activation at time instant t_k ($k > 1$) if there is an aircraft arrival on the intersecting route's initial waypoint at time instant t_{k-1} traveling on the no-CRP path, where the CRP re-initialization is as in Definition 10.*

Proof The proof is similar to the proof of Lemma 15 for the 2-path CRP case, and is not provided here for brevity. The main difference is the deactivation condition — there is an increase in the time interval without arrivals needed for safe CRP deactivation. This is due to the increased distance needed to pass through all potential intersection waypoints for the 3-path CRP when compared to the 2-path CRP.

4.3.6 On-demand CRP for Non-perpendicular Intersections with Different Speeds

The on-demand features can be implemented to the non-perpendicular CRP with different speeds discussed in Chapter 3. This section discusses the implementation and shows safe activation and deactivation conditions. The on-demand features are applied to the CRP provided in Fig. 3.2(a), with path lengths from Lemma 10 for intersection angle smaller than 90 degrees (a similar argument can be done for intersection angle greater than 90 degrees, which is not shown in this section). The on-demand features of the non-perpendicular CRP with different speeds follows the path assignment provided in Definition 6, and aircraft synchronization procedures provided in Definition 4 (from Chapter 3). First, the no-CRP case is discussed.

Lemma 19. *Consider the no-CRP case for two intersecting routes (as in Fig. 3.1) with the minimal arrival spacing ($\overline{D}_1$ for Route 1, and $\overline{D}_2$ for Route 2) satisfying the condition in Eq. (3.8), and Eq. (3.9) with $n = 1$ for the single flow intersection. Then, there are no conflicts (without the use of a CRP) provided, at each discrete time instant t_k, no more than one aircraft arrives at the two initial waypoints either at p_1 from Route 1 or at p11 from Route 2 in Fig. 3.2(a).*

Proof An aircraft in a route will not have conflicts with other aircraft in the same route since arrivals at a route's initial waypoint are separated by at-least one discrete time instant, which corresponds to an arrival spacing ($\overline{D}_1$ for Route 1, and $\overline{D}_2$ for Route 2) larger than the minimal separation D_{sep}. Moreover, since there is only one aircraft arriving at the initial waypoints (p_1 and p_{11}, which are equal in route travel

time for each route to the intersection point p_i), only one aircraft can occupy the intersection point p_i at any time as illustrated in Fig. 4.12. For example, at the instant when an aircraft from Route 1 is at the intersection point p_i (as illustrated in Fig. 4.12(a)), the nearest aircraft on the other route (Route 2) is at least $\overline{D}_2 \geq d_{int2} = \frac{2D_{sep}\sqrt{\alpha^2-2\alpha\cos\theta_{int}+1}}{\alpha\sin\theta_{int}}$ away due to only one arrival at each discrete time instant and the minimal arrival spacing condition in Eq. (3.9) with $n = 1$. This ensures that there is sufficient spacing to avoid conflicts with aircraft in the other route (Route 2).

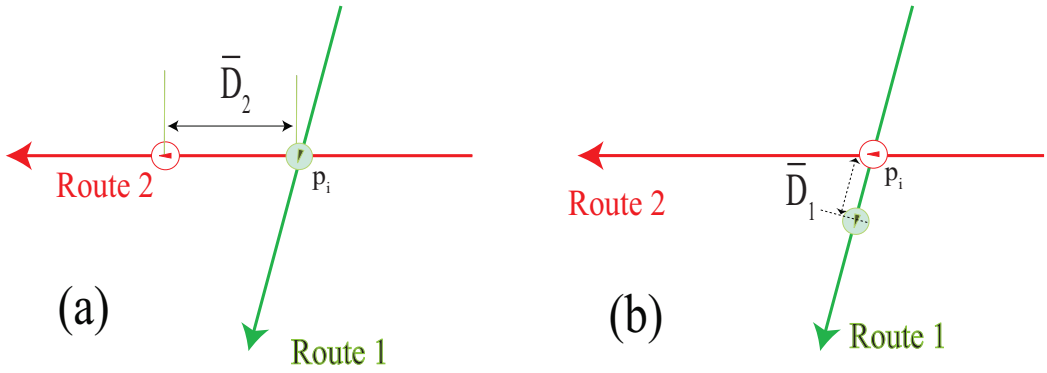

Figure 4.12: Safe no-CRP case under non-perpendicular CRP with different speeds arrival conditions. Distance between aircraft is one arrival spacing $\overline{D}_1 > d_{int1}$ for Route 1 and $\overline{D}_2 > d_{int2}$ for Route 2 from Eq. (3.8) and (3.9) with $n = 1$ when one aircraft is at the intersection.

Next, the activation conditions are considered in the following lemma.

Lemma 20. *Consider the activation, at time instant t_k, of the provably-safe CRP (illustrated in Fig. 3.2(a) and described in detail in Chapter 3, with path lengths defined in Lemma 10) for the two intersecting routes (in Fig. 3.2(a)) where the minimal arrival spacing satisfies the sufficient-spacing condition in Eq. (3.4), and Eq. (3.5). Furthermore, let the following CRP-activation-enabling condition be satisfied: all aircraft that arrived during the time interval $[t_k - T_{CRP}, t_k)$ satisfy the no-CRP conditions (in Lemma 19), and follow the no-CRP, nominal routes Route 1 and Route*

2 through the intersection. Then, there are no conflicts if the CRP is activated at time instant t_k (i.e., aircraft arriving at and after time instant t_k follow the CRP paths) with the following re-initialization of the CRP path-assignment.

1. *If there is no arrival on Route 1 at time instant t_{k-1}, then the CRP is re-initiated by assigning paths $R_{1,1}$ and $R_{2,1}$ for arrivals at time instant t_k (as illustrated in Fig. 4.13).*

2. *Otherwise, if there is no arrival on Route 2 at time instant t_{k-1}, then the CRP is re-initiated by assigning paths $R_{1,2}$ and $R_{2,2}$ for arrivals at time instant t_k .*

Proof The proof is provided for the case when Route 1 has no aircraft arriving at time instant t_{k1} (as in Fig. 4.13). The other case when route Route 2 has no aircraft, arriving at time instant t_{k1}, follows by a similar argument. There are no conflicts between aircraft arriving (on either route) at or after time instant t_k because the CRP is conflict free. Due to the CRP-activation-enabling condition, all aircraft remaining in the CRP region arriving before time instant t_k are no-CRP aircraft there is no potential for conflicts between CRP-following aircraft arriving before and after time instant t_k. Therefore, the proof needs to only show that there are no conflicts between no-CRP aircraft arriving before time instant t_k on a route (e.g., b_{k1} on Route 2) and CRP-following aircraft arriving at or after time instant tk on: (case 1) the other route, e.g., aircraft a_k, a_{k+1} on Route 1; and (case 2) the same route, e.g., aircraft b_k on Route 2. The proof is divided into these two cases.

Case 1: No conflicts between aircraft in different routes Lack of conflicts between aircraft on Route 2 arriving before time instant t_k and aircraft on Route 1 is shown below. Lack of conflicts between aircraft on Route 1 arriving before time instant t_k and those on Route 2 follow by symmetry.

Consider the time instant at $t = t_{k-1}$ as in Fig. 4.13(a), when aircraft b_{k-1} along Route 2 (which will travel on the no-CRP path) has arrived at the initial way point

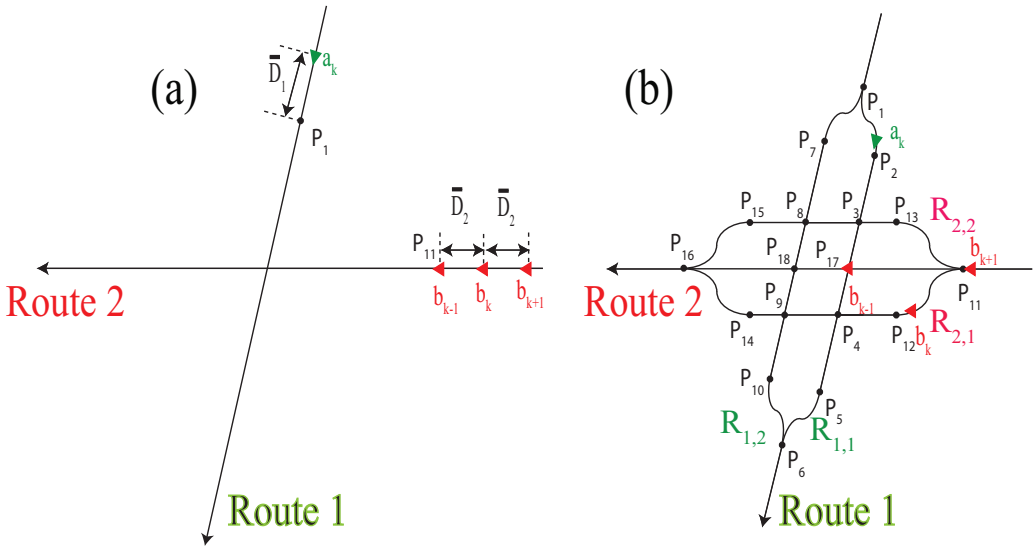

Figure 4.13: Example CRP activation for non-perpendicular intersection and different speeds. (a) Without CRP, aircraft arriving at time instant t_k have conflict with each other, i.e., between aircraft a_k on Route 1 arriving at the initial waypoint p_1 and aircraft b_k arriving at the initial waypoint p_{11} on Route 2. The subscript k indicates the arrival time instant t_k. (b) 2-path CRP is applied from time instant t_k onwards, e.g., for aircraft a_k, a_{k+1}, b_k, and b_{k+1}. Aircraft arriving earlier travel on the nominal route, e.g., aircraft b_{k1} arriving at time t_{k1} travels on route Route 2 in this example.

p_{11}. At this time instant, aircraft a_k would be located $\overline{D}_1$ distance prior to the initial way point p_1 on Route 1, and aircraft a_{k+1} would be $2\overline{D}_1$ distance prior to the initial way point p_1. Then, when all aircraft advance such that aircraft b_{k-1} arrives at the intersection point p_{17} (between the no-CRP path Route 2 and CRP path route $R_{1,1}$) as in Fig. 4.13(b). Then, at this moment, the distance $d_{bk-1,ak}$ between aircraft b_{k1} and the first CRP-following aircraft a_k on Route 1 is

$$d_{b_{k-1},a_k} \geq 1.5\overline{D}_1 > 1.5D_{sep} \tag{4.21}$$

because the travel time difference for aircraft a_k and b_{k-1} from the initial position

from Fig. 4.13 to the intersection point p_{17} is

$$T_{a_k} - T_{b_{k-1}} = \frac{1.5\overline{D}_1 + 2\alpha R_2(\phi_2 - \sin\phi_2)}{v_1} > 1.5T_{\overline{D}} \tag{4.22}$$

, where the CRP path lengths are defined in Lemma 10 from Chapter 3, and $T_{\overline{D}}$ is defined from Definition 4 in Chapter 3. Because the time difference of $1.5T_{\overline{D}}$ represents a travel distance of $1.5\overline{D}_1$ on Route 1, when aircraft b_{k-1} is at point p_{17}, aircraft a_k is at least $1.5\overline{D}_1$ away, as in Eq. (4.21). When aircraft b_{k1} on Route 2 reaches the next intersection point (p_{18}), by similar arguments, b_{k-1} would be at least $1.5\overline{D}_1$ distance away from aircraft a_{k+1}. Thus, activation is safe.

Case 2: No conflicts between aircraft in the same route The proof for this case is the same as the proof for Lemma 13 on no conflicts between aircraft in the same route because the speed for aircraft on the same route are the same. Thus this case is conflict-free.

Finally, the deactivation condition is when all aircraft has passed the final way point (p_6 for Route 1. and p_{16} for Route 2), and there are no simultaneous arrivals at the initial way point of each route, then the CRP is deactivated to the original no-CRP routes.

4.3.7 On-demand CRP without synchronized arrival on one route

The on-demand CRP can be extended to the case for non-synchronized arrival on one route, and the intersecting route with rare arrivals as illustrated in Fig. 4.14. Here, the CRP is considered such that one route does not need synchronization on Route 2 from Fig. 4.14 where the arrival spacing distance between aircraft is not constant. Aircraft a_k from Route 1, however needs to arrive synchronized with aircraft b_k such that when b_k arrives at p_5, a_k arrives at p_1. The CRP is activated for on Route 2 for aircraft b_k and aft of b_k (b_{k-1} travels on no-CRP path, b_k travels on route $R_{2,2}$, and b_{k+1} travels on route $R_{2,1}$ as in Fig. 4.14(b)). Then, the CRP in Fig. 4.14(b) needs the following path lengths in the following definition to be conflict-free.

Definition 11. *[*CRP Path length without synchronization from one route*]*

1. *Initial way points* p_1 *and* p_5 *are equal in route travel time for each route to the intersection point* p_i

2. *The route speed ratio* α *is* $\frac{v_1}{v_2}$.

3. *Distance between points* p_2 *and* p_7 (d_{p2p7}) *is* d_{min}, *where* $d_{min} = min(d_1, d_2, d_3)$.

4. *Distance between points* p_3 *and* p_6 (d_{p3p6}) *is* $d_{min} + \Delta$.

5. *Distance between points* p_3 *and* p_8 (d_{p3p8}) *is* d_{min}.

6. *Distance between points* p_2 *and* p_9 (d_{p2p9}) *is* $d_{min} + \Delta$.

7. *Distance between points* p_2 *and* p_3 (d_{p2p3}) *is* $\alpha(d_{min} + \Delta)$.

8. *Difference in distance* $\Delta = d_{p3p6} - d_{p2p7} = d_{p2p3} \cos\theta_{int}$.

The CRP can be activated safe with the conditions provided above if d_{min} is sufficient and satisfies the spacing requirements provided in Eq. (3.8) and (3.9) with $n = 1$. This can be proved by the same method to prove the on-demand CRP shown in Lemma 20. Considering the travel time from all aircraft from the initial position in Fig. 4.14(a)) to the position when no-CRP aircraft a_k arrives at the intersection points p_2, p_i, and p_3, the distance to b_k, b_{k+1}, and b_{k+2} respectively will each be at least d_{min}. Deactivation is done when all aircraft has passed the final way point (p_4 for Route 1. and p_{10} for Route 2).

4.4 Application of CRP

4.4.1 Example Intersection

An example is presented below to illustrate the proposed CRP activation and deactivation. Conflicts are not expected in existing aircraft data (since these are already

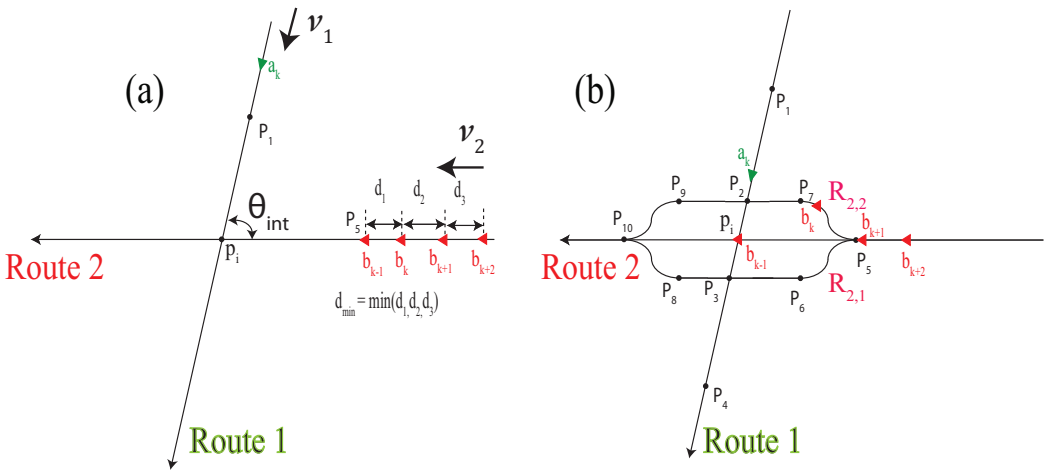

Figure 4.14: Example CRP activation for non-synchronized arrival for Route 2. (a) Without CRP, aircraft arriving at time instant t_k have conflict with each other, i.e., between aircraft a_k on Route 1 arriving at the initial waypoint p_1 and aircraft b_k arriving at the initial waypoint p_5 on Route 2. (b) 2-path CRP is applied from for aircraft b_k, b_{k+1}, and b_{k+2}.

resolved) — however, existing data can be used to illustrate typical issues that arise in the design of the proposed CRP. In particular, for the example below, data from a perpendicularly intersecting route in the Cleveland sector ZOB59 (see Fig. 4.15) was used to: (i) quantify the arrival spacing $\overline{D}$ in the routes; and (ii) identify the average aircraft velocity v_{sp}. Additionally, simulation of the on-demand CRP for random arriving aircraft is provided in the Appendix of this thesis.

4.4.2 Data Generation

Nominal values, such as aircraft arrival rates, speed, and spacing are based on aircraft data from the example Cleveland sector ZOB59 using the Future ATM Concepts Evaluation Tool (FACET) for May 1st, 2004 at 35000ft altitude during the time interval $[305, 65735]$ Coordinated Universal Time (UTC) seconds which corresponds to about 18 hours of data. During this period, 35 aircraft passed through the east-to-

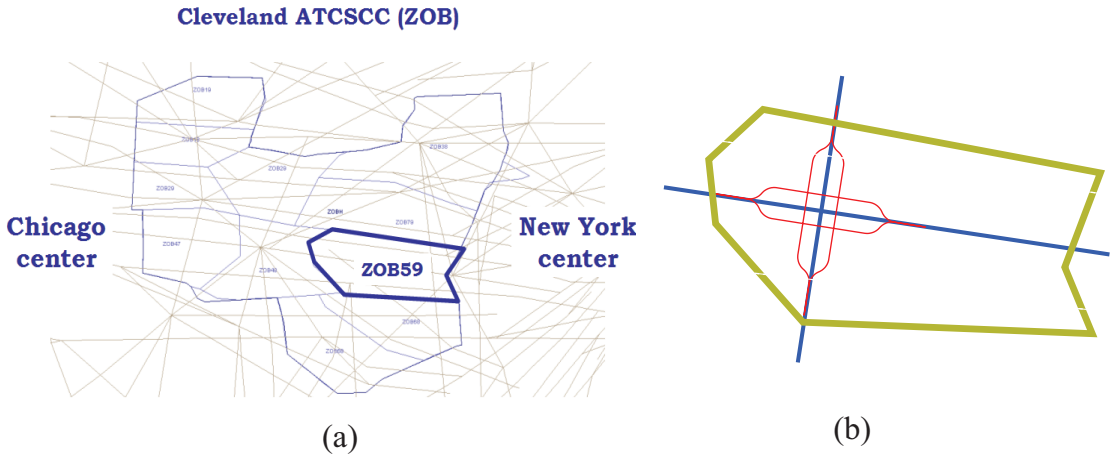

Figure 4.15: (a) Cleveland Sector ZOB59. (b) Example perpendicularly intersecting routes in Cleveland Sector and the proposed CRP.

west route, and 19 aircraft passed through the north-to-south route. The average of all aircraft speeds in the date were used as the nominal aircraft speed $v_{sp} = 0.122$ Nautical Miles (NM) per second in the example. Moreover, the minimal arrival spacing, $\overline{D}_{min} = 9.23$ NM was considered to be the synchronized arrival spacing $\overline{D} = \overline{D}_{min}$. The latitude and longitude of each aircraft in the data were converted into 2-D Cartesian coordinates (x,y) by using stereographic projection [58], and the averaged initial and final coordinates were used as the nominal entry and exit points for the sector for the example. The routes R_1 and R_2 were then considered to be straight lines between these averaged arrival and departure points, which were then used to identify the route intersection point, as in Fig. 4.15. Finally, the estimated time of arrival (ETA) of the aircraft at the initial waypoints were synchronized to a scheduled time of arrival (STA)

$$STA = kT_{\overline{D}} \quad \text{if} \quad ETA \in \left[\left(k - \frac{1}{2}\right) T_{\overline{D}}, \left(k + \frac{1}{2}\right) T_{\overline{D}}\right) \tag{4.23}$$

which assigns a discrete time instant $t_k = kT_{\overline{D}}$ to the arrival of each aircraft at the initial waypoint. The resulting expected and scheduled arrival times (ETAs, STAs,

and k corresponding to arrival time instant t_k) for all aircraft in the data are presented in Tables 4.1 and 4.2.

4.4.3 CRP Activation and Deactivation

A provably-safe always-on CRP for this example was designed in Chapter 2. Briefly, the arrival spacing of $\overline{D} = 9.23$ NM satisfies the CRP number of split condition in Eq. (4.7), Remark 23 to maintain a minimal separation distance $D_{sep} = 5$ NM with a 2-path CRP. Additionally, the diverge and merge portions of the CRP design were composed of circular segments of radius $R = 4.86$NM and turn angle $\phi = 58.31$ degrees. Additional details of the provably-safe, always-on, 2-path CRP design are provided in Chapter 2.

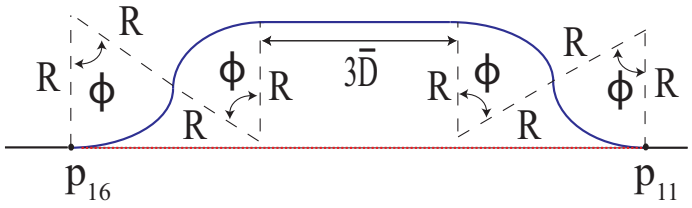

Figure 4.16: Travel distance on CRP path.

A requirement for deactivation is that there should be sufficient spacing between aircraft on the same route as in Eq. (4.16), Lemma 15. For the 2-path CRP with path design shown in Fig. 4.16, the clearance-distance difference δ (see Eq. (4.15)) of

$$\delta = 4R(\phi - \sin(\phi)) = 3.24\text{NM}$$

satisfies Eq. (4.16) with $m = 1$ since the clearance-distance difference $\delta = 3.24$ NM is less than the arrival spacing $\overline{D} = 9.23$. Therefore, the 2-path CRP can be deactivated safely whenever there is less-than-two (i.e., zero or one) arrivals at time instant t_k if there is no aircraft arrival in the previous time instant t_{k-1} from Lemma 15. Moreover,

Table 4.1: Expected time of arrival (ETA), scheduled time of arrival (STA), and k corresponding to discretized time of arrival t_k for North-to-south route R_1. If the CRP is used then the CRP path is provided; otherwise, the aircraft follows the nominal route R_1

#	Aircraft #	ETA (s)	k	STA (s)	CRP Path
1	31	3010	40	3028	-
2	117	37780	499	37774	-
3	130	39010	515	38985	-
4	136	39580	523	39590	-
5	145	40810	539	40802	-
6	196	44440	587	44435	-
7	203	45370	599	45344	-
8	221	46180	610	46176	-
9	222	46240	611	46252	-
10	227	46480	614	46479	-
11	296	51700	683	51702	-
12	298	51880	685	51854	$R_{1,1}$
13	301	52180	689	52156	-
14	314	53080	701	53065	-
15	316	53380	705	53368	$R_{1,1}$
16	328	55180	729	55184	-
17	351	57040	754	57077	-
18	431	64180	848	64193	-
19	428	64240	849	64268	-

Table 4.2: Expected time of arrival (ETA), scheduled time of arrival (STA), and k corresponding to discretized time of arrival t_k for East-to-west route R_2. If the CRP is used then the CRP path is provided; otherwise, the aircraft follows the nominal route R_2

#	Aircraft #	ETA (s)	k	STA (s)	CRP Path	#	Aircraft #	ETA (s)	k	STA (s)	CRP Path
1	2	908	12	908	-	21	189	45518	601	45495	-
2	7	1118	15	1135	-	22	265	49658	656	49658	-
3	9	1478	20	1514	-	23	262	49868	659	49886	-
4	10	1568	21	1590	-	24	277	50558	668	50567	-
5	18	2498	33	2498	-	25	287	51818	685	51854	$R_{2,1}$
6	21	2678	35	2649	-	26	300	53138	702	53141	-
7	32	3728	49	3709	-	27	302	53258	704	53292	-
8	40	5408	71	5375	-	28	303	53348	705	53368	$R_{2,1}$
9	44	5618	74	5602	-	29	327	56018	740	56017	-
10	47	5918	78	5905	-	30	340	57548	760	57531	-
11	50	7238	96	7267	-	31	342	57758	763	57758	-
12	51	7418	98	7418	-	32	358	58988	779	58969	-
13	55	7808	103	7797	-	33	425	64748	855	64723	-
14	87	18278	241	18243	-	34	432	65648	867	65631	-
15	111	38318	506	38304	-	35	435	65858	870	65858	-
16	120	38678	511	38682	-						
17	129	39848	526	39818	-						
18	164	42938	567	42921	-						
19	180	44978	594	44965	-						
20	192	45428	600	45419	-						

Lemma 15 also shows that the CRP can always be safely activated (whenever needed), i.e., whenever there are two arrivals at any time t_k and the CRP is deactivated when there are no arrivals at the previous time instant t_{k-1} and less than two arrivals at the current time instant t_k. These conditions for the transitions between no-CRP and the 2-path CRP are illustrated in Fig. 4.17.

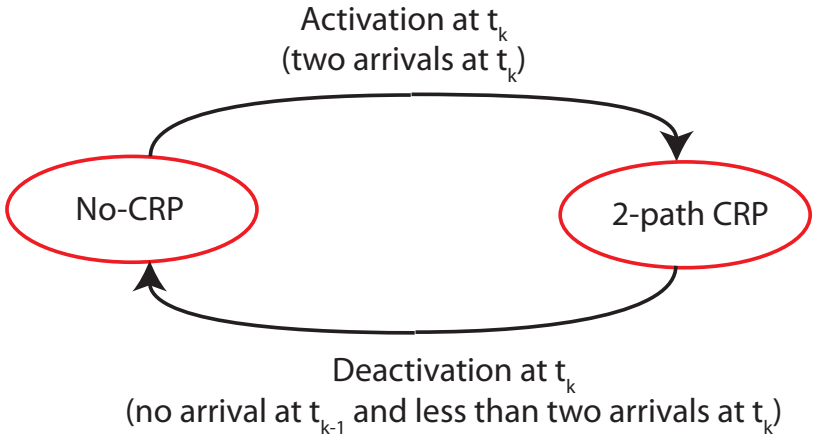

Figure 4.17: Conditions for transitions between no-CRP and the 2-path CRP for the example intersection.

4.4.4 Results

There were two cases of conflicts — between aircraft # 298 from R_1 and aircraft # 287 from R_2 at time instant t_{685}, and between aircraft # 316 from R_1 and aircraft # 303 from R_2 at time instant t_{705} — from Table 4.2 and Table 4.1. For the first conflict case, there are no aircraft arrivals at the previous time instant, i.e., at time instant t_{684}, while the second conflict case does have an aircraft arrival on route R_2 at the previous time instant t_{704}. Therefore, from Lemmas 13 and 15, the CRP is activated with paths $R_{1,1}$ and$R_{2,1}$ in both instances. There are no aircraft arrivals at the time instants t_{686} and t_{706} immediately after the CRP activation at time instants

t_{685} and t_{705} respectively — therefore, the CRP can be deactivated at the following time instants t_{687} and t_{707} as per Lemma 15.

4.4.5 Advantage of proposed on-demand CRP

Overall, the Cleveland sector example shows that the use of the on-demand CRP can improve performance when compared to the always-on CRP. In particular, the number of on-demand activations are two and also the number of deactivations that could be safely be done is two. Only four aircraft out of the fifty four aircraft (in the data for this example) need to use the CRP. In contrast, with the always-on CRP all fifty four aircraft would use the CRP. While safety can be guaranteed with the always-on CRP, the proposed activation/deactivation procedures avoid unwanted CRP maneuvers, and associated delays and effort, for the other 33 aircraft on route R_2 and 17 aircraft on route R_1. Thus, the use of the proposed on-demand CRP (with provably-safe activation and deactivation) can lead to substantial performance improvement over the case with an always-on CRP.

4.4.6 Implementation Issues

Implementation of the proposed on-demand CRP is also affected by non perpendicularity of the route intersections and the need to ensure robustness of the CRP design in the presence of aircraft-speed and arrival-time variations caused by uncertainties. Although not studied in the current chapter, the proposed notion of deactivation when sufficient space is available and ensuring the ability to activate the CRP in a safe manner can also be extended to non-perpendicular CRP designs or cases when the nominal speed of the aircraft are different along the two routes, e.g., in CRP designs studied in Chapter 3. Alternatively, non-perpendicularly intersecting routes could be turned to generate a perpendicular intersection before implementing the current on-demand CRP approach. Density differences in the two routes could also be handled in the proposed approach by using different number of split-paths in each

route to ensure conflict-free intersection of the path, i.e., to satisfy the CRP number of split condition in Remark 23.

Timing is critical to maintain the minimal spacing needed to avoid conflict in the proposed CRP approach and the activation and deactivation procedures. The robustness of the proposed CRP can be increased by using larger (minimal) separation distances in the CRP design as discussed, e.g., in Chapter 3) to create additional buffer space and thereby, allow for uncertainty/error in the aircraft arrival timing as well as aircraft velocities — at the cost of increased space needed for the CRP.

The use of on-demand CRP increases the need for information sharing. For example, with the always-on 2-path CRP, an aircraft arriving at the initial waypoint can choose a CRP path based on the arrival time, such as path 1 for even arrival times and path 2 for odd arrival times. The choice of the path does not depend on previous arrivals on the same route or the other route in the intersection. In contrast, with the on-demand CRP additional information about aircraft in both routes is needed to make the decision to activate or deactivate the CRP. Thus, the increase in efficiency of the on-demand CRP (compared to the always-on CRP) is at the cost of increased need for information-of and co-ordination-between aircraft in the two routes.

4.5 Conclusion

This chapter presented a decoupled conflict resolution procedure (CRP) that includes on-demand activation of the split-path CRP, and deactivation by collapsing the split-path CRP when not needed. The proposed on-demand CRP was shown to be provably-safe and was illustrated with an example route intersection in the Cleveland sector. When conflicts are rare, the proposed on-demand CRP can lead to substantial performance improvements. Results from the example showed that only four aircraft out of the fifty four aircraft (in the data) needed to use the CRP (with the on-demand CRP) as opposed to all fifty four aircraft using the CRP with the always-on CRP.

Chapter 5

CONCLUSIONS

This thesis presented automated provably-safe en-route CRPs for ATC. The main purpose for using automated en-route CRPs are to reduce workload and improve performance for air traffic controllers. The provably-safe CRP design in this thesis dealt with the challenges in automated en-route CRPs by proposing locally decoupled CRPs. The CRP design in the thesis addresses three issues: (i) accounts for turn dynamic, (ii) resolves conflicts for non-perpendicular intersections with different speeds, and (iii) enables on-demand activation and deactivation of CRPs.

The main importance of including the turn-rate bound for CRPs are when proving the existence of a safe CRP. In particular, for a given arrival spacing $\overline{D}$ in the intersecting routes, the allowable turn angle (in a single turn) is more flexible compared to CRP designs with instant turns. Moreover, the bounded-turn-rate analysis is important to quantify the size of the converge/diverge procedure and therefore, to quantify the space needed for the CRP.

The CRP design was generalized by applying the CRP to non-perpendicular intersections and allowing different route speeds. Conditions were developed to guarantee safety and were illustrated with an example route intersection in the Cleveland sector.

Finally, the thesis presented a decoupled conflict resolution procedure (CRP) that includes on-demand activation of the split-path CRP, and deactivation by collapsing the split-path CRP when not needed. The proposed on-demand CRP was shown to be provably-safe and was illustrated with an example route intersection in the Cleveland sector. When conflicts are rare, the proposed on-demand CRP can lead to substantial performance improvements.

Overall, the provably-safe automated CRP design will help contribute in ATC for en-route airspace, where traffic will increase in the near future.

Future work is needed to optimize the CRP, e.g., to minimize the time delay generated by the CRP or minimize the area used by the CRP. Such optimization could be done with the design parameters such as the turn radius (R in Figure 2.2 and R_s in Figure 3.10) or turn angle (ϕ in Figure 2.2 and ϕ_s in Figure 3.10) used in the CRP turn paths and the synchronization path. Moreover, the automation procedures need methods to interface with human controllers for better situational awareness (for dealing with cases such as failure in automation). Additionally, issues such as dealing with daily changing traffic should be addressed.

Appendix A

SIMULATION OVERVIEW

This section of the Appendix explains the conflict resolution simulation algorithm of the proposed CRP design in this thesis. The simulation is programmed in Matlab software version 7.4.0. There are three main simulations that are presented. First is the always-on CRP for two perpendicularly intersecting routes with same speeds (referred as Simulation 1). In this case, each route is assumed to have synchronized arrival from STA of $k = 1$ to $k = 20$ (per Definition 2 in Chapter 2). The second simulation is the always-on CRP for two non-perpendicularly intersecting routes with different speeds (referred as Simulation 2). In this case, each route is assumed to have synchronized arrival from STA of $k = 1$ to $k = 20$ (per Definition 4 in Chapter 4) . Finally, the third simulation is the on-demand CRP for two perpendicularly intersecting routes with same speeds (referred as Simulation 3). For this simulation, aircraft arrival to the intersection area is random. The flow chart of each simulation is presented, followed by the files of each simulation and the description for each simulation files.

A.1 Flow Chart of Simulations

Fig. A.1 shows the flow chart of both Simulation 1, always-on CRP for perpendicular intersecting routes with same speeds, and Simulation 2, always-on CRP for non-perpendicular intersecting routes with different speeds.

Fig. A.2 shows the flow chart of both Simulation 3, on-demand CRP for perpendicular intersecting routes with same speeds.

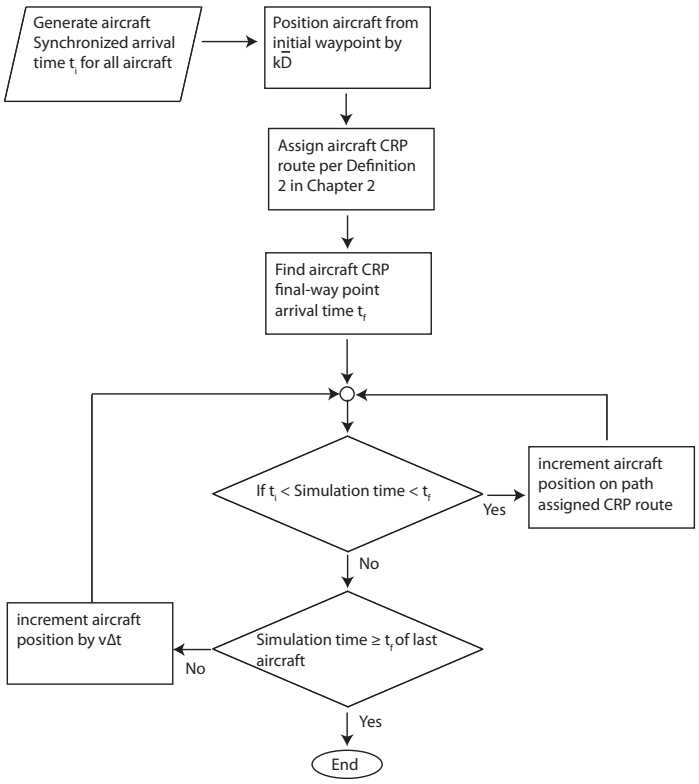

Figure A.1: Flow chart of simulation for CRP for perpendicularly intersecting route with same speeds and non-perpendicularly intersecting routes with different speeds.

A.2 Description for simulation file of CRP that is always-on for perpendicularly intersecting route with same speeds

The CRP simulation that is always-on for perpendicularly intersecting route with same speeds is programmed in Matlab file $'crp-perpend-twosplit-always-on.m'$. The program runs the CRP for aircraft on routes R_1 and R_2 from Chapter 2, Fig.2.1. The given inputs are the arrival spacing distance $\overline{D}$, the aircraft speed v. The process to find the turn path parameters are described in Section 2.5. The program first checks that the arrival spacing satisfies the requirements from Eq. (2.8), (2.9) and Lemma 4 with the CRP path design parameters. The program defines the paths of the CRP with the design parameters. Aircraft from each route are assigned synchronized

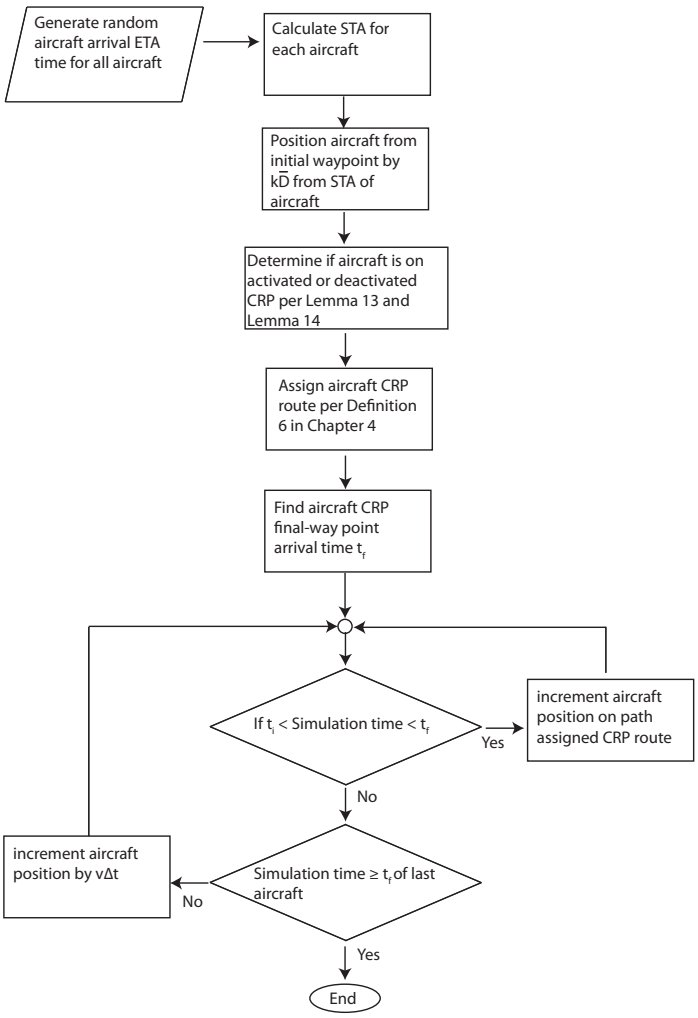

Figure A.2: Flow chart of simulation for on-demand CRP for perpendicularly intersecting routes with same speeds.

arrival time to the initial-way point of the CRP (discrete STA of $k = 1$ to $k = 20$). Aircraft are all positioned initially by $k\overline{D}$ from the initial-way point of the CRP for each route. Then, the program assigns the CRP route for each aircraft per Definition 2, depending on odd or even discrete STA. Once all aircraft are assigned to the CRP path to travel on, the simulation runs by incrementing the aircraft position by the assigned path depending on aircraft location which is as the following:

1. If an aircraft has not arrived at the initial-way point, increment toward the initial-way point by the increment time step.

2. If an aircraft has passed the initial-way point, and has not exited the CRP path, increment position on the assigned path.

3. If an aircraft has passed the final-way point, increment away from the final-way point by the increment time step.

A.3 Description for simulation file of CRP that is always-on for non-perpendicularly intersecting route with different speeds

The CRP simulation that is always-on for non-perpendicularly intersecting route with different speeds is programmed in Matlab file $'crp-nonperpend-twosplit-always-on.m'$. The program runs the CRP for aircraft on routes Route 1 and Route 2 from Chapter 3, Fig.3.2. The given inputs are the arrival spacing distance $\overline{D}_1$ for $Route1$, the arrival spacing distance $\overline{D}_2$ for $Route2$, the aircraft speed v_1 for $Route1$, the aircraft speed v_2 for $Route2$, and the route intersection angle θ_{int}. The process to find the turn path parameters are described in Section 3.4. The program first checks that the arrival spacing for each route satisfies the requirements from Eq. (3.8), (3.9),(3.10) and (3.11) with the CRP path design parameters. Other than having each route having its own arrival spacing distance and speed, the program algorithm is the same as the simulation for the CRP for perpendicularly intersecting routes with same speeds.

A.4 Description for simulation files of on-demand CRP for perpendicularly intersecting route with same speeds

The CRP simulation that is on-demand for perpendicularly intersecting route with same speeds is programmed in Matlab files $'CRP-2path-input-generator.m'$, and

$'CRP - OnDemand - Example.m'$. This simulation applied the CRP for aircraft on R_1 and R_2 from Chapter 4, Fig.4.2. $'CRP - 2path - input - generator.m'$ generates random arrival times as ETAs for aircraft on each route. The ETAs for all aircraft are stored from the Matlab workspace as $'CRP - 2path - input.mat'$.
$'CRP - OnDemand - Example.m'$ imports $'CRP - 2path - input.mat'$. The given inputs are the arrival spacing distance $\overline{D}$, the aircraft speed v. The process to find the turn path parameters are described in Section 2.5. The program first checks that the arrival spacing satisfies the requirements from Eq. (2.8), (2.9) and Lemma 4 with the CRP path design parameters. The program defines the paths of the CRP with the design parameters. Aircraft from each route are assigned synchronized arrival time from the ETAs imported from $'CRP - 2path - input.mat'$ per Eq. (3.36) to the initial-way point of the CRP. Aircraft are all positioned initially by $k\overline{D}$ from the initial-way point of the CRP for each route. Each aircraft are determined if the CRP is activated for the respective aircraft per Lemma 13 and Lemma 14. Then all aircraft are assigned to the path that the aircraft will travel per Definition 6. The program simulates the CRP by incremented aircraft position by the path the aircraft travels on.

Appendix B

PROGRAM FILE FOR ALWAYS-ON CRP FOR PERPENDICULAR INTERSECTING ROUTES WITH SAME SPEEDS

This Appendix section gives example simulation of the CRP design concept in Chapter 2.

B.1 ***Matlab file*** *'crp − perpend − twosplit − always − on.m'*

```
%general data
Dsep = 5;
Dbar= 9.23; %Cleveland aircraft spacing distance
%v_avg=438.95/3600; %NM/s
v_avg = 400/3600;

g= 0.0052969762419; % NM/s^2 gravitational acceleration
mu=30*pi/180; %bank angle limit
Rmin = v_avg^2/(g*tan(mu)); %min turn radius requirement for banked angle turn
dint = 2*sqrt(2)*Desp %min spacing requirement on intersection

phi= acos(1-Dbar/(4*R)); %turn angle
omega = v_avg/R; %angular velocity

R= Dbar/(4*(phi-sin(phi))); % turn radius of turn paths on CRP
dturn1 = (Dsep-2*R*sin(phi/2))/cos(phi/2)+R*phi; %spacing distance requirement
%for turns
```

```
%discrete time increment
K = Dbar/v_avg;
delta = 100;
tstep = K/delta;
t=0;
theta1= 1.5*pi; %angles to define turn path
theta2= pi/2-phi;

%check if turn radius satisfies banked turn requirement
if R<Rmin
    error('turn radius is too small')
end

%check if arrival spacing satisfies turn requirement
if Dbar<dturn1
    error('Arrival spacing is too small for turns')
end

%check if arrival spacing satisfies intersection requirement
if 2*Dbar<dint
    error('Arrival spacing is too small for 2 path split intersection')
end
%define CRP split paths
%turn path center of curvature coordinates
E0x = 1.5*Dbar+2*R*sin(phi);
E0y = R;
E1x = 1.5*Dbar;
E1y = 0.5*Dbar-R;

turn_index = round(phi/(omega*tstep)); %number of points
%to represent phi angle turn arc

%increment position algorithm
```

```
for kk=1:turn_index+1
    %clock
    t(kk+1) = t(kk) + tstep;
    %turn arc for route R2,2 1st turn
        theta1(kk+1) = theta1(kk)-omega*tstep;
        E0x(kk+1)= E0x(1)+R*cos(theta1(kk));
        E0y(kk+1)= E0y(1)+R*sin(theta1(kk));
    %turn arc for route R2,2 2nd turn
        theta2(kk+1) = theta2(kk)+omega*tstep;
        E1x(kk+1)= E1x(1)+R*cos(theta2(kk));
        E1y(kk+1)= E1y(1)+R*sin(theta2(kk));

end
% 1st turn of Route R2,2
R221x = E0x(2:kk);
R221y = E0y(2:kk);
%figure(3);
%plot(E1x,E1y)
% 2nd turn of Route R2,2
R222x = E1x(2:kk);
R222y = E1y(2:kk);
% 3rd turn of Route R2,2
R223x = -R222x;
R223y = R222y;
R223x = fliplr(R223x);
R223y = fliplr(R223y);
% 4th turn of Route R2,2
R224x = -R221x;
R224y = R221y;
R224x = fliplr(R224x); %reverse order for correct node sequence
R224y = fliplr(R224y);

%1st turn of R2,1
```

```
R211x = R221x;
R211y = -R221y;
%2nd turn of R2,1
R212x = R222x;
R212y = -R222y;
% 3rd turn of Route R2,1
R213x = -R212x;
R213y = R212y;
R213x = fliplr(R213x);
R213y = fliplr(R213y);
% 4th turn of Route R2,1
R214x = -R211x;
R214y = R211y;
R214x = fliplr(R214x);
R214y = fliplr(R214y);
%1st turn of R1,1
R111x = R221y;
R111y = R221x;
%2nd turn of R1,1
R112x = R222y;
R112y = R222x;
% 3rd turn of Route R1,1
R113x = R112x;
R113y = -R112y;
R113x = fliplr(R113x);
R113y = fliplr(R113y);
% 4th turn of Route R1,1
R114x = R111x;
R114y = -R111y;
R114x = fliplr(R114x);
R114y = fliplr(R114y);

%1st turn of R1,2
```

```
R121x = -R111x;
R121y = R111y;
%2nd turn of R1,2
R122x = -R112x;
R122y = R112y;
% 3rd turn of Route R1,2
R123x = R122x;
R123y = -R122y;
R123x = fliplr(R123x);
R123y = fliplr(R123y);
% 4th turn of Route R1,2
R124x = R121x;
R124y = -R121y;
R124x = fliplr(R124x);
R124y = fliplr(R124y);

%straight sections
xst1 = 1.5*Dbar:-R*omega*tstep:-1.5*Dbar;
xstEW=xst1;
yst1 =  0.5*Dbar*ones(size(xst1));
yst2 = -yst1;
R21xst = xstEW; R21yst = yst2;
R22xst = xstEW; R22yst = yst1;
yst3 = xstEW; xstNS = yst1; xstNS2 = yst2;
R11xst = xstNS; R11yst = yst3;
R12xst = xstNS2; R12yst = yst3;

%initial path
xEWinitial = 1.5*Dbar+2*R*sin(phi)+Dbar/delta:Dbar/delta:,...
1.5*Dbar+2*R*sin(phi)+2*Dbar;
xEWinitial = fliplr(xEWinitial);
yEWinitial = zeros(size(xEWinitial));
yNSinitial = 1.5*Dbar+2*R*sin(phi)+Dbar/delta:Dbar/delta:,...
```

```
1.5*Dbar+2*R*sin(phi)+2*Dbar;
yNSinitial = fliplr(yNSinitial);
xNSinitial = zeros(size(yNSinitial));

%final path
xEWfinal = -1.5*Dbar-2*R*sin(phi)-Dbar/delta:-Dbar/delta:-1.5*Dbar-2*R*sin(phi)-2*Dbar;
yEWfinal = zeros(size(xEWfinal));
xNSfinal = yEWfinal;
yNSfinal = xEWfinal;

%nominal path
xEWnom = 3*Dbar+2*R*sin(phi)+Dbar:-Dbar/delta:-3*Dbar-2*R*sin(phi)-Dbar;
yEWnom = zeros(size(xEWnom));
xNSnom = yEWnom;
yNSnom = xEWnom;
%route definition in correct sequence
R22x = [R221x R222x R22xst R223x R224x];
R22y = [R221y R222y R22yst R223y R224y];
R21x = [R211x R212x R21xst R213x R214x];
R21y = [R211y R212y R21yst R213y R214y];
R12x = [R121x R122x R12xst R123x R124x];
R12y = [R121y R122y R12yst R123y R124y];
R11x = [R111x R112x R11xst R113x R114x];
R11y = [R111y R112y R11yst R113y R114y];


crp_path_length1 = (length(R11x))*tstep*v_avg; %route 1 CRP path length
crp_path_length2 = (length(R21x))*tstep*v_avg; %route 2 CRP path length
ini_point_R2x = 1.5*Dbar+2*R*sin(phi);
ini_point_R1y = 1.5*Dbar+2*R*sin(phi);
%Route 2 Aircraft (E-W)
for EW_AP=1:20
    %index number of aircraft
```

```
    AP_EW(EW_AP,1)=EW_AP;
    %synchronize aircraft by discrete time index "k"
    AP_EW(EW_AP,2)= EW_AP;
    %x-position
    AP_EW(EW_AP,3)=AP_EW(EW_AP,2)*Dbar + ini_point_R2x;
    %y-position
    AP_EW(EW_AP,4)=0;
    %ti
    AP_EW(EW_AP,5)=AP_EW(EW_AP,2)*Dbar/v_avg;
    %tf
    AP_EW(EW_AP,6)=AP_EW(EW_AP,5)+crp_path_length2/v_avg;
    %Route
    if rem(AP_EW(EW_AP,2),2) == 1
        AP_EW(EW_AP,7)=21;
    elseif rem(AP_EW(EW_AP,2),2) == 0
        AP_EW(EW_AP,7)=22;
    end
    %pointer
    AP_EW(EW_AP,8)=1;
end

%Route 1 Aircraft (NE-SW)
for NS_AP=1:20
    %index number of aircraft
    AP_NS(NS_AP,1)=NS_AP;
    %synchronize aircraft by discrete time index "k"
    AP_NS(NS_AP,2)= NS_AP;
    %x-position
    AP_NS(NS_AP,3)= 0;
    %y-position
    AP_NS(NS_AP,4)=AP_NS(NS_AP,2)*Dbar+ ini_point_R1y;
    %ti
    AP_NS(NS_AP,5)=AP_NS(NS_AP,2)*Dbar/v_avg;
```

```
    %tf CRP exit time;
    AP_NS(NS_AP,6)=AP_NS(NS_AP,5)+crp_path_length1/v_avg;
    %Route
    if rem(AP_NS(NS_AP,2),2) == 1
        AP_NS(NS_AP,7)=11;
    elseif rem(AP_NS(NS_AP,2),2) == 0
        AP_NS(NS_AP,7)=12;
    end
    %pointer
    AP_NS(NS_AP,8)=1;
end

%Number of airplanes in each route
EWindex = size(AP_EW);
NSindex = size(AP_NS);

%simulation
sp_index = 1; %speed of simulation (increment interval)
deltaT= tstep;
finalT = round((max(AP_EW(end,6),AP_NS(end,6))+5*Dbar/v_avg)/(sp_index*deltaT));

figure(6);
sim_movie = avifile('crp_perp_same_speed.avi','quality',100);
for tindex = 1:finalT
    % simulation time
    t= tindex * deltaT * sp_index;
    % route 2 aircraft
    for EW=1:EWindex(1)
        if AP_EW(EW,5)> t %if t>ti (before initial way point) increment x position
            AP_EW(EW,3) = AP_EW(EW,3)-deltaT*v_avg*sp_index;
            AP_EW(EW,4) = 0;

        elseif AP_EW(EW,5)<= t && AP_EW(EW,8)< length(R21x)
```

```
        AP_EW(EW,8) = AP_EW(EW,8)+sp_index;    %increment pointer
%if route is R21 increment position in R21
        if AP_EW(EW,7)==21
            AP_EW(EW,3) = R21x(AP_EW(EW,8));
            AP_EW(EW,4) = R21y(AP_EW(EW,8));
%if route is R22 increment position in R22
        elseif AP_EW(EW,7)==22
            AP_EW(EW,3) = R22x(AP_EW(EW,8));
            AP_EW(EW,4) = R22y(AP_EW(EW,8));
        end


    elseif AP_EW(EW,8) >= length(R21x)
        AP_EW(EW,3) = AP_EW(EW,3)-deltaT*v_avg*sp_index;
        AP_EW(EW,4) = 0;
    end
end
%R1 aircraft, same algorithm as R2 aircraft above
for NS=1:NSindex(1)
    if AP_NS(NS,5)> t
        AP_NS(NS,3) = 0;
        AP_NS(NS,4) = AP_NS(NS,4)-deltaT*v_avg*sp_index;

    elseif AP_NS(NS,5)<= t && AP_NS(NS,8) < length(R11x)
        AP_NS(NS,8) = AP_NS(NS,8)+sp_index;
        if AP_NS(NS,7)==11
            AP_NS(NS,3) = R11x(AP_NS(NS,8));
            AP_NS(NS,4) = R11y(AP_NS(NS,8));
        elseif AP_NS(NS,7)==12
            AP_NS(NS,3) = R12x(AP_NS(NS,8));
            AP_NS(NS,4) = R12y(AP_NS(NS,8));

        end
```

```
        elseif AP_NS(NS,8) >= length(R11x)
            AP_NS(NS,3) = 0;
            AP_NS(NS,4) = AP_NS(NS,4)-deltaT*v_avg*sp_index;
        end
    end

    plot(AP_EW(:,3),AP_EW(:,4),'r<',AP_NS(:,3),AP_NS(:,4),'bv')
    hold on
    plot(R11x,R11y,'k',R12x,R12y,'k',R21x,R21y,'k',R22x,R22y,'k',xEWnom,yEWnom,'k',...
xNSnom,yNSnom,'k')

        for EWN =1:EWindex(1)
            line(AP_EW(EWN,3)+Dsep/2*cos([0:pi/50:2*pi]),...
            AP_EW(EWN,4)+Dsep/2*sin([0:pi/50:2*pi]),'Color','r');
        end
        for NSN =1:NSindex(1)
            line(AP_NS(NSN,3)+Dsep/2*cos([0:pi/50:2*pi]),...
            AP_NS(NSN,4)+Dsep/2*sin([0:pi/50:2*pi]),'Color','b');
        end

     axis([-35 35 -35 35])
     axis equal
     axis manual
          drawnow
     hold off
     M_get = getframe;
     sim_movie = addframe(sim_movie,M_get);
end
```

Appendix C

PROGRAM FILE FOR CRP ON NON-PERPENDICULAR INTERSECTING ROUTES WITH DIFFERENT SPEEDS

This Appendix section gives example simulation of the CRP design concept in Chapter 3.

C.1 Matlab file $'crp - nonperpend - twosplit - always - on.m'$

```
%intersection angle smaller than 90 degrees, different speeds

%general data
Dsep = 5;
 v1=443/3600; %NM/s
 v2=437/3600;
alpha = v1/v2;

Dbar2= 9.23; %Aircraft spacing distance for route 2
Dbar1 = alpha*Dbar2;


g= 0.0052969762419; % NM/s^2 gravitational acceleration
mu=30*pi/180; %bank angle limit
Rmin1 = v1^2/(g*tan(mu)); %min turn radius of route 1 (north to south)
Rmin2 = v2^2/(g*tan(mu)); %min turn radius of route 2 (east to west)
theta_intc = 60*pi/180; %intersection angle

%intersection requirement
```

```
%Dbar1_min requirement for intersection
dint1c = 2*Dsep*sqrt(alpha^2-2*alpha*cos(theta_intc)+1)/(sin(theta_intc));
 %Dbar2_min requirement for intersection
dint2c=2*Dsep*sqrt(alpha^2-2*alpha*cos(theta_intc)+1)/...
    (alpha*sin(theta_intc));

% extra straight length on route 2's longer path split
delta2c = Dbar1 * cos(theta_intc) / (1-alpha*cos(theta_intc));
%intersecting flow node to node path length for route 1 (N-S)
dp3p4c = Dbar1+ alpha.*delta2c;
H2c = dp3p4c * sin(theta_intc); %total height of split for route 2
r2c = Rmin2; %turn radius of route 2 (EW route)
phi2c = acos(1-H2c/(4*r2c)); %turn angle of route 2
% extra straight length on route 1's longer path split
delta1c = Dbar2 * cos(theta_intc) / (1-cos(theta_intc)/alpha);
%intersecting flow node to node path length for route 2 (E-W)
dp3p8c = Dbar2+ delta1c/alpha;
H1c = dp3p8c * sin(theta_intc); %total height of split for route 1
r1c = Rmin1; %turn radius of route 1 turn
phi1c = acos(1-H1c/(4*r1c)); %turn angle of route 1

%spacing requirement due to turn
Dreq1 = (Dsep-2*r1c*sin(phi1c/2))/(cos(phi1c/2))+r1c*phi1c
Dreq2 = (Dsep-2*r2c*sin(phi2c/2))/(cos(phi2c/2))+r2c*phi2c
%straight distance after diverge turn
dp13p3c = Dbar2; %Assume route 2 first straight section is Dbar2
dp2p3c = alpha*(2*r2c*phi2c)-2*r1c*phi1c+Dbar1;
%remaining path lengths
dp12p4c = dp13p3c + delta2c;
dp7p8c = dp2p3c + delta1c;
%check if design parameters are valid
phi1c_deg = phi1c*180/pi;
phi2c_deg = phi2c*180/pi;
```

```
%turn rates
omega1c = v1/r1c; %angular velocity
omega2c = v2/r2c; %angular velocity
%discrete time increment
K = Dbar1/v1; %same as Dbar2/v2
%K2 = Dbar2/v2;
%delta = 100; %number of discrete points of 1 diverge/converge turn

%delta = 373;
%tstep = K/delta;
tstep = 0.5; %time step = 0.1 sec
tc=0; ttc=0;

%CRP node coordinates
p3xc = dp3p8c/2*cos(pi/2-theta_intc);
p3yc = dp3p4c/2+dp3p8c/2*sin(pi/2-theta_intc);
p13xc = p3xc + dp13p3c*cos(pi/2-theta_intc);
p13yc = p3yc + dp13p3c*sin(pi/2-theta_intc);
p4xc = p3xc;
p4yc = p3yc - dp3p4c;
p11xc = p13xc+H2c/2*sin(pi/2-theta_intc)+...
    2*r2c*sin(phi2c)*cos(pi/2-theta_intc);
p11yc = p13yc - H2c/2*cos(pi/2-theta_intc)+2*r2c*sin(phi2c)*...
    sin(pi/2-theta_intc);
p12xc = p4xc + dp12p4c*cos(pi/2-theta_intc);
p12yc = p4yc + dp12p4c*sin(pi/2-theta_intc);
p8xc = p3xc - dp3p8c*cos(pi/2-theta_intc);
p8yc = p3yc - dp3p8c*sin(pi/2-theta_intc);
p9xc = p8xc;
p9yc = p8yc - dp3p4c;
p15xc = p8xc - dp12p4c*cos(pi/2-theta_intc);
p15yc = p8yc - dp12p4c*sin(pi/2-theta_intc);
```

```
p2xc = p3xc;
p2yc = p3yc + dp2p3c;
p7xc = p8xc;
p7yc = p8yc + dp7p8c;
p5xc = p3xc;
p5yc = p4yc - dp7p8c;
p10xc = p8xc;
p10yc = p9yc - dp2p3c;
p16xc = p15xc + H2c/2*cos(theta_intc)-2*r2c*sin(phi2c)*cos(pi/2-theta_intc);
p16yc = p15yc - H2c/2*sin(theta_intc)-2*r2c*sin(phi2c)*sin(pi/2-theta_intc);
p14xc = p9xc - dp13p3c*cos(pi/2-theta_intc);
p14yc = p9yc - dp13p3c*sin(pi/2-theta_intc);
p1xc = 0;
p1yc = p2yc + 2*r1c*sin(phi1c);
p6xc = 0;
p6yc = p5yc - 2*r1c*sin(phi1c);

%%%initial way point coordinate
ini_point_R2x = p11xc; %initial arrival of route 2 x coordinate
ini_point_R2y = p11yc; %initial arrival of route 2 y coordinate
ini_point_R1xc = p1xc;
ini_point_R1yc = p1yc;

%define CRP split paths
%center of curvature coordinates (turns)
E111_xshiftc = p1xc + r1c;    %R_{1,1} first turn center x coordinate
E111_yshiftc = p1yc; %R_{1,1} first turn center y coordinate
E112_xshiftc = p2xc - r1c; %R_{1,1} second turn center x coordinate
E112_yshiftc = p2yc; %R_{1,1} second turn center y coordinate
E113_xshiftc = p5xc - r1c;    %R_{1,1} third turn center x coordinate
E113_yshiftc = p5yc; %R_{1,1} third turn center y coordinate
E114_xshiftc = p6xc + r1c;    %R_{1,1} fourth turn center x coordinate
E114_yshiftc = p6yc; %R_{1,1} fourth turn center y coordinate
```

```
E121_xshiftc = p1xc - r1c;    %R_{1,2} first turn center x coordinate
E121_yshiftc = p1yc; %R_{1,2} first turn center y coordinate
E122_xshiftc = p7xc + r1c; %R_{1,2} second turn center x coordinate
E122_yshiftc = p7yc; %R_{1,2} second turn center y coordinate
E123_xshiftc = p10xc + r1c;    %R_{1,2} third turn center x coordinate
E123_yshiftc = p10yc; %R_{1,2} third turn center y coordinate
E124_xshiftc = p6xc - r1c;    %R_{1,2} fourth turn center x coordinate
E124_yshiftc = p6yc; %R_{1,2} fourth turn center y coordinate
%number of points to represent phi angle turn arc
turn_index1c = round(phi1c/(omega1c*tstep));

E221_xshiftc = p11xc-r2c*cos(theta_intc);%R_{2,2} first turn center x-coord
E221_yshiftc = p11yc+r2c*sin(theta_intc);%R_{2,2} first turn center y-coord
E222_xshiftc = p13xc + r2c*cos(theta_intc); %R_{2,2} second turn center x
E222_yshiftc = p13yc - r2c*sin(theta_intc); %R_{2,2} second turn center y
E223_xshiftc = p15xc+r2c*cos(theta_intc);    %R_{2,2} third turn center x
E223_yshiftc = p15yc-r2c*sin(theta_intc); %R_{2,2} third turn center y
E224_xshiftc = p16xc - r2c*cos(theta_intc); %R_{2,2} fourth turn center x
E224_yshiftc = p16yc + r2c*sin(theta_intc); %R_{2,2} fourth turn center y

E211_xshiftc = p11xc+r2c*cos(theta_intc);    %R_{2,1} first turn center x
E211_yshiftc = p11yc-r2c*sin(theta_intc); %R_{2,1} first turn center y
E212_xshiftc = p12xc-r2c*cos(theta_intc); %R_{2,1} second turn center x
E212_yshiftc = p12yc+r2c*sin(theta_intc); %R_{2,1} second turn center y
E213_xshiftc = p14xc-r2c*cos(theta_intc); %R_{2,1} third turn center x
E213_yshiftc = p14yc+r2c*sin(theta_intc); %R_{2,1} third turn center y
E214_xshiftc = p16xc+r2c*cos(theta_intc); %R_{2,1} fourth turn center x
E214_yshiftc = p16yc-r2c*sin(theta_intc); %R_{2,1} fourth turn center y
turn_index2c = round(phi2c/(omega2c*tstep)); %number of points to represent
                                             %phi angle turn arc

%first turn path of route R_{1,1}
```

```
theta1_R11c = pi:omega1c*tstep:pi+phi1c; %Angle from the circle and range
E111xc = r1c*cos(theta1_R11c)+ E111_xshiftc;
E111yc = r1c*sin(theta1_R11c)+ E111_yshiftc;
%second turn path of route R_{1,1}
theta2_R11c = phi1c:-omega1c*tstep:0; %Angle from the circle and range
E112xc = r1c*cos(theta2_R11c)+ E112_xshiftc;
E112yc = r1c*sin(theta2_R11c)+ E112_yshiftc;
%first turn path of route R_{1,2}
theta1_R12c = 0:-omega1c*tstep:-phi1c; %Angle from the circle and range
E121xc = r1c*cos(theta1_R12c)+ E121_xshiftc;
E121yc = r1c*sin(theta1_R12c)+ E121_yshiftc;
%second turn path of route R_{1,2}
theta2_R12c = pi-phi1c:omega1c*tstep:pi; %Angle from the circle and range
E122xc = r1c*cos(theta2_R12c)+ E122_xshiftc;
E122yc = r1c*sin(theta2_R12c)+ E122_yshiftc;
%third turn path of route R_{1,1}
theta3_R11c = theta1_R12c; %Angle from the circle and range
E113xc = r1c*cos(theta3_R11c)+ E113_xshiftc;
E113yc = r1c*sin(theta3_R11c)+ E113_yshiftc;
%fourth turn path of route R_{1,1}
theta4_R11c = theta2_R12c; %Angle from the circle and range
E114xc = r1c*cos(theta4_R11c)+ E114_xshiftc;
E114yc = r1c*sin(theta4_R11c)+ E114_yshiftc;
%third turn path of route R_{1,2}
theta3_R12c = theta1_R11c; %Angle from the circle and range
E123xc = r1c*cos(theta3_R12c)+ E123_xshiftc;
E123yc = r1c*sin(theta3_R12c)+ E123_yshiftc;
%fourth turn path of route R_{1,2}
theta4_R12c = theta2_R11c; %Angle from the circle and range
E124xc = r1c*cos(theta4_R12c)+ E124_xshiftc;
E124yc = r1c*sin(theta4_R12c)+ E124_yshiftc;

%first turn path of route R_{2,1}
```

```
theta1_R21c = pi-theta_intc:omega2c*tstep:pi-theta_intc+phi2c; %Angle from
                                                        %the circle and range
E211xc = r2c*cos(theta1_R21c)+ E211_xshiftc;
E211yc = r2c*sin(theta1_R21c)+ E211_yshiftc;
%second turn path of route R_{2,1}
theta2_R21c = 2*pi-theta_intc+phi2c:-omega2c*tstep:2*pi-theta_intc;
E212xc = r2c*cos(theta2_R21c)+ E212_xshiftc;
E212yc = r2c*sin(theta2_R21c)+ E212_yshiftc;

%first turn path of route R_{2,2}
theta1_R22c = -theta_intc:-omega2c*tstep:-theta_intc-phi2c; %Angle from the
                                                      %circle and range
E221xc = r2c*cos(theta1_R22c)+ E221_xshiftc;
E221yc = r2c*sin(theta1_R22c)+ E221_yshiftc;
%second turn path of route R_{2,2}
theta2_R22c = pi-theta_intc-phi2c:omega2c*tstep:pi-theta_intc;
E222xc = r2c*cos(theta2_R22c)+ E222_xshiftc;
E222yc = r2c*sin(theta2_R22c)+ E222_yshiftc;

%third turn path of route R_{2,1}
theta3_R21c = theta1_R22c; %Angle from the circle and range
E213xc = r2c*cos(theta3_R21c)+ E213_xshiftc;
E213yc = r2c*sin(theta3_R21c)+ E213_yshiftc;
%fourth turn path of route R_{2,1}
theta4_R21c = theta2_R22c; %Angle from the circle and range
E214xc = r2c*cos(theta4_R21c)+ E214_xshiftc;
E214yc = r2c*sin(theta4_R21c)+ E214_yshiftc;

%third turn path of route R_{2,2}
theta3_R22c = theta1_R21c; %Angle from the circle and range
E223xc = r2c*cos(theta3_R22c)+ E223_xshiftc;
E223yc = r2c*sin(theta3_R22c)+ E223_yshiftc;
%fourth turn path of route R_{2,2}
```

```
theta4_R22c = theta2_R21c; %Angle from the circle and range
E224xc = r2c*cos(theta4_R22c)+ E224_xshiftc;
E224yc = r2c*sin(theta4_R22c)+ E224_yshiftc;

% 1st turn of Route R2,2, delete repeated initial point
R221xc = E221xc(2:length(E221xc));
R221yc = E221yc(2:length(E221yc));

% 2nd turn of Route R2,2
R222xc = E222xc(2:length(E222xc));
R222yc = E222yc(2:length(E222yc));

% 3rd turn of Route R2,2
R223xc = E223xc(2:length(E223xc));
R223yc = E223yc(2:length(E223yc));

% 4th turn of Route R2,2
R224xc = E224xc(2:length(E224xc));
R224yc = E224yc(2:length(E224yc));

%1st turn of R2,1
R211xc = E211xc(2:length(E211xc));
R211yc = E211yc(2:length(E211yc));
%2nd turn of R2,1
R212xc = E212xc(2:length(E212xc));
R212yc = E212yc(2:length(E212yc));
% 3rd turn of Route R2,1
R213xc = E213xc(2:length(E213xc));
R213yc = E213yc(2:length(E213yc));
% 4th turn of Route R2,1
R214xc = E214xc(2:length(E214xc));
R214yc = E214yc(2:length(E214yc));
```

```
% 1st turn of Route R1,1
R111xc = E111xc(2:length(E111xc));
R111yc = E111yc(2:length(E111yc));
% 2nd turn of Route R1,1
R112xc = E112xc(2:length(E112xc));
R112yc = E112yc(2:length(E112yc));
% 1st turn of Route R1,2
R121xc = E121xc(2:length(E121xc));
R121yc = E121yc(2:length(E121yc));
% 2nd turn of Route R1,2
R122xc = E122xc(2:length(E122xc));
R122yc = E122yc(2:length(E122yc));
% 3rd turn of Route R1,1
R113xc = E113xc(2:length(E113xc));
R113yc = E113yc(2:length(E113yc));
% 4th turn of Route R1,1
R114xc = E114xc(2:length(E114xc));
R114yc = E114yc(2:length(E114yc));
% 3rd turn of Route R1,2
R123xc = E123xc(2:length(E123xc));
R123yc = E123yc(2:length(E123yc));
% 4th turn of Route R1,2
R124xc = E124xc(2:length(E124xc));
R124yc = E124yc(2:length(E124yc));

%route 1 split straight sections
R11ystc = p2yc:-r1c*omega1c*tstep:p5yc;
R11xstc = H1c/2*ones(size(R11ystc));
R12ystc = p7yc:-r1c*omega1c*tstep:p10yc;
R12xstc = -H1c/2*ones(size(R12ystc));

% route 2 split straight paths
m1c = tan(pi/2-theta_intc); %slope of route 2
```

```
b1c = p3yc - m1c*p3xc; %intercept of linear line of R_{2,2}
b2c = p4yc- m1c*p4xc; %intercept of linear line of R_{2,1}
R21xstc = p12xc:-r2c*omega2c*tstep*cos(pi/2-theta_intc):p14xc;
R21ystc = m1c*R21xstc+b2c;
R22xstc = p13xc:-r2c*omega2c*tstep*cos(pi/2-theta_intc):p15xc;
R22ystc = m1c*R22xstc+b1c;
%original routes
R2xc = p11xc+2*Dbar2:-r2c*omega2c*tstep*cos(pi/2-theta_intc):p16xc-2*Dbar2;
R2yc = m1c*R2xc;
R1yc = 2*Dbar1+p1yc:-r1c*omega1c*tstep:p6yc-2*Dbar1;
R1xc = zeros(size(R1yc));
%route definition in correct sequence
R22xc = [R221xc R222xc R22xstc R223xc R224xc];
R22yc = [R221yc R222yc R22ystc R223yc R224yc];
R21xc = [R211xc R212xc R21xstc R213xc R214xc];
R21yc = [R211yc R212yc R21ystc R213yc R214yc];
R12xc = [R121xc R122xc R12xstc R123xc R124xc];
R12yc = [R121yc R122yc R12ystc R123yc R124yc];
R11xc = [R111xc R112xc R11xstc R113xc R114xc];
R11yc = [R111yc R112yc R11ystc R113yc R114yc];

%Data for aircraft in route
%col 1- index, col2-k, col3- x, col4- y, col5- ti, col6- tf, col7- route,
%col8-pointer
crp_path_length1c = length(R11xc)*tstep*v1;
crp_path_length2c = length(R21xc)*tstep*v2;
%Route 2 Aircraft (E-W)
for EW_APc=1:20
    %index number of aircraft
    AP_EWc(EW_APc,1)=EW_APc;
    %synchronize aircraft by discrete time index "k"
    AP_EWc(EW_APc,2)= EW_APc;
    %x-position
```

```
    AP_EWc(EW_APc,3)= p11xc + EW_APc*Dbar2*cos(pi/2-theta_intc);
    %y-position
    AP_EWc(EW_APc,4)= p11yc + EW_APc*Dbar2*sin(pi/2-theta_intc);
    %ti
    AP_EWc(EW_APc,5)=AP_EWc(EW_APc,2)*Dbar2/v2;
    %tf
    AP_EWc(EW_APc,6)=AP_EWc(EW_APc,5)+crp_path_length2c/v2;
    %Route
    if rem(AP_EWc(EW_APc,2),2) == 1
        AP_EWc(EW_APc,7)=21;
    elseif rem(AP_EWc(EW_APc,2),2) == 0
        AP_EWc(EW_APc,7)=22;
    end
    %pointer
    AP_EWc(EW_APc,8)=1;
end

%Route 1 Aircraft (NE-SW)
for NS_APc=1:20
    %index number of aircraft
    AP_NSc(NS_APc,1)=NS_APc;
    %synchronize aircraft by discrete time index "k"
    AP_NSc(NS_APc,2)= NS_APc;
    %x-position
    AP_NSc(NS_APc,3)= 0;
    %y-position
    AP_NSc(NS_APc,4)=AP_NSc(NS_APc,2)*Dbar1+ p1yc;
    %ti
    AP_NSc(NS_APc,5)=AP_NSc(NS_APc,2)*Dbar1/v1;
    %tf
    AP_NSc(NS_APc,6)=AP_NSc(NS_APc,5)+crp_path_length1c/v1;
    %Route
    if rem(AP_NSc(NS_APc,2),2) == 1
```

```
        AP_NSc(NS_APc,7)=11;
    elseif rem(AP_NSc(NS_APc,2),2) == 0
        AP_NSc(NS_APc,7)=12;
    end
    %pointer
    AP_NSc(NS_APc,8)=1;
end
%Number of airplanes in each route
EWindexc = size(AP_EWc);
NSindexc = size(AP_NSc);

%simulation
sp_index = 1; %speed of simulation (increment interval)
deltaT= tstep;
finalT = round((max(AP_EWc(end,6),AP_NSc(end,6))+5*Dbar1/v1)/...
    (sp_index*deltaT));
%Data for aircraft in route
%col 1- index, col2-k, col3- x, col4- y, col5- ti, col6- tf, col7- route,
%col8-pointer

figure(6);
sim_movie = avifile('crp_ondemand_final.avi','quality',100);
for tindex = 1:finalT
    % simulation time
    t= tindex * deltaT * sp_index;
    % route 2 aircraft
    for EWc=1:EWindexc(1)
        if AP_EWc(EWc,5)> t %if t>ti increment x position
            AP_EWc(EWc,3) = AP_EWc(EWc,3)-deltaT*v2*...
                sp_index*cos(pi/2-theta_intc);
            AP_EWc(EWc,4) = AP_EWc(EWc,4)-deltaT*v2*sp_index*...
                sin(pi/2-theta_intc);
```

```
    elseif AP_EWc(EWc,5)<= t && AP_EWc(EWc,8)< length(R21xc)-sp_index
        AP_EWc(EWc,8) = AP_EWc(EWc,8)+sp_index;    %increment pointer
        if AP_EWc(EWc,7)==21 %if route is R21 increment position in R21
            AP_EWc(EWc,3) = R21xc(AP_EWc(EWc,8));
            AP_EWc(EWc,4) = R21yc(AP_EWc(EWc,8));
        elseif AP_EWc(EWc,7)==22 %if route is R22 increment position in
                                 %R22
            AP_EWc(EWc,3) = R22xc(AP_EWc(EWc,8));
            AP_EWc(EWc,4) = R22yc(AP_EWc(EWc,8));

        end
    elseif AP_EWc(EWc,8) >= length(R21xc)-sp_index
        AP_EWc(EWc,3) = AP_EWc(EWc,3)-deltaT*v2*sp_index*...
            cos(pi/2-theta_intc);
        AP_EWc(EWc,4) = AP_EWc(EWc,4)-deltaT*v2*sp_index*...
            sin(pi/2-theta_intc);
    end
end
%R1 aircraft, same algorithm as R2 aircraft above
for NSc=1:NSindexc(1)
    if AP_NSc(NSc,5)> t
        AP_NSc(NSc,3) = 0;
        AP_NSc(NSc,4) = AP_NSc(NSc,4)-deltaT*v1*sp_index;

    elseif AP_NSc(NSc,5)<= t && AP_NSc(NSc,8) < length(R11xc)-sp_index
        AP_NSc(NSc,8) = AP_NSc(NSc,8)+sp_index;
        if AP_NSc(NSc,7)==11
            AP_NSc(NSc,3) = R11xc(AP_NSc(NSc,8));
            AP_NSc(NSc,4) = R11yc(AP_NSc(NSc,8));
        elseif AP_NSc(NSc,7)==12
            AP_NSc(NSc,3) = R12xc(AP_NSc(NSc,8));
            AP_NSc(NSc,4) = R12yc(AP_NSc(NSc,8));
        end
```

```
        elseif AP_NSc(NSc,8) >= length(R11xc)-sp_index
            AP_NSc(NSc,3) = 0;
            AP_NSc(NSc,4) = AP_NSc(NSc,4)-deltaT*v1*sp_index;
        end
    end

    plot(AP_EWc(:,3),AP_EWc(:,4),'r<',AP_NSc(:,3),AP_NSc(:,4),'bv')
    hold on
    plot(R11xc,R11yc,'k',R12xc,R12yc,'k',R21xc,R21yc,'k',R22xc,R22yc,...
        'k',R2xc,R2yc,'k',R1xc,R1yc,'k')
        for EWNc =1:EWindexc(1)
            line(AP_EWc(EWNc,3)+Dsep/2*cos([0:pi/50:2*pi]),...
            AP_EWc(EWNc,4)+Dsep/2*sin([0:pi/50:2*pi]),'Color','r');
        end
        for NSNc =1:NSindexc(1)
            line(AP_NSc(NSNc,3)+Dsep/2*cos([0:pi/50:2*pi]),...
            AP_NSc(NSNc,4)+Dsep/2*sin([0:pi/50:2*pi]),'Color','b');
        end

     axis([-45 45 -45 45])
     axis equal
     axis manual
          drawnow
     hold off
     M_get = getframe;
     sim_movie = addframe(sim_movie,M_get);
end
```

Appendix D

PROGRAM FILE FOR ON-DEMAND CRP ON TWO PERPENDICULARLY INTERSECTING ROUTES WITH SAME SPEEDS

This Appendix section gives example simulation of the CRP design concept in Chapter 4.

D.1 ***Matlab file*** $'CRP-2path-input-generator.m'$

```
%general data
Dsep = 5;
Dbar= 9.23; %Cleveland aircraft spacing distance
v_avg=438.95/3600; %NM/s


Tfinal = 50; %final time index
TDbar = (Dbar)/v_avg; %time interval for synchronization
                      %from 0 to Dbar distance
filter_EW=0; %filter index for aircraft arrival existence
filter_NS=0;

%random arrival generator
%EW - Route 2, NS- Route 1
for tindex=1:Tfinal
    %the time index arrival per Dbar/2 distance
    EW_mult = tindex+ ( -0.5 + (0.5-(-0.5))*rand(1));
    NS_mult = tindex+ (0.5*rand(1));
```

```
    %non-filtered STA arrival
    %(non-zero for aircraft existence, zero for empty spot)
    AP_EW_T(tindex,1) = TDbar* EW_mult * round(rand(1));
    AP_NS_T(tindex,1) = TDbar* NS_mult * round(rand(1));

    %filter out empty arrivals
    if AP_EW_T(tindex) > 0
        filter_EW = filter_EW+1;
        AP_EW_T_raw(filter_EW,1) = AP_EW_T(tindex,1);
        %AP_EW_k(filter_EW,1)= round(AP_EW_T_raw(filter_EW,1)/(TDbar));
    end


    if AP_NS_T(tindex) > 0
        filter_NS = filter_NS+1;
        AP_NS_T_raw(filter_NS,1) = AP_NS_T(tindex,1,1);
        %AP_NS_k(filter_NS,1)= round(AP_NS_T_raw(filter_NS,1)/TDbar);
    end
end
```

D.2 Matlab file $'CRP - OnDemand - Example.m'$

```
%load time arrival input and general data
load CRP_2path_input.mat;

%general data
g= 0.0052969762419; % NM/s^2 gravitational acceleration
mu=30*pi/180; %bank angle limit
phis = [0:0.01:90]*pi/180;
KK= 0; %intersection length index, if KK=0, non-sync'd spacing at the end

% % code for increased intersection to sync spacing at the end
% for k=1:length(phis)
```

```
%     f1(k) = phis(k)-sin(phis(k))- (1-cos(phis(k))).*(2*KK+1);
%     if k>1
%         if (f1(k-1) < 0 && f1(k) >0) || (f1(k-1) > 0 && f1(k) <0)
%             index = k
%             break
%         end
%     end
% end

%phi = phis(index) %for sync'd arrival at the end
%R = Dbar/(4*(phi-sin(phi))); %Turn radius for increased intersection
Rmin = v_avg^2/(g*tan(mu)); %minimum turn radius
R=Rmin;
if R<Rmin
    error('turn radius is too small')
end
%phi= acos(1-Dbar/(4*R)); %turn angle

phi = acos(1-(2*KK+1)*Dbar/(4*R)); %turn angle for minimum turn radius
omega = v_avg/R; %angular velocity
%discrete time increment
K = Dbar/v_avg;%T_{\overline{D}} in paper
delta = 100; %number of discrete points to divide T_{\overline{D}} in paper
tstep = K/delta;
t=0;
theta1= 1.5*pi; %angle to create turn route
theta2= pi/2-phi; %angle to create turn route

%%%initial way point coordinate
ini_point = Dbar+(2*KK+1)/2*Dbar+2*R*sin(phi);

%define CRP turn paths
%center of arc in turn paths
```

```
E0x = ini_point; %turn 1 center x coordinate
E0y = R; %turn 1 center y coordinate
E1x = Dbar+(2*KK+1)/2*Dbar; %turn 2 center x coordinate
E1y = (2*KK+1)/2*Dbar-R; %turn 2 center y coordinate
turn_index = round(phi/(omega*tstep)); %number of points to represent phi
                                       %angle turn arc
%create discrete points for turns
for kk=1:turn_index+1
    %clock
    t(kk+1) = t(kk) + tstep;
    %turn arc for route R2,2 1st turn
        theta1(kk+1) = theta1(kk)-omega*tstep;
        E0x(kk+1)= E0x(1)+R*cos(theta1(kk));
        E0y(kk+1)= E0y(1)+R*sin(theta1(kk));
    %turn arc for route R2,2 2nd turn
        theta2(kk+1) = theta2(kk)+omega*tstep;
        E1x(kk+1)= E1x(1)+R*cos(theta2(kk));
        E1y(kk+1)= E1y(1)+R*sin(theta2(kk));

end

% Create all CRP turns by shifting or mirroring first two turns created
% above
% 1st turn of Route R2,2
R221x = E0x(2:kk);
R221y = E0y(2:kk);

% 2nd turn of Route R2,2
R222x = E1x(2:kk);
R222y = E1y(2:kk);

% 3rd turn of Route R2,2
R223x = -R222x;
```

```
R223y = R222y;
R223x = fliplr(R223x);
R223y = fliplr(R223y);
% 4th turn of Route R2,2
R224x = -R221x;
R224y = R221y;
R224x = fliplr(R224x); %reverse order for correct node sequence
R224y = fliplr(R224y);

%1st turn of R2,1
R211x = R221x;
R211y = -R221y;
%2nd turn of R2,1
R212x = R222x;
R212y = -R222y;
% 3rd turn of Route R2,1
R213x = -R212x;
R213y = R212y;
R213x = fliplr(R213x);
R213y = fliplr(R213y);
% 4th turn of Route R2,1
R214x = -R211x;
R214y = R211y;
R214x = fliplr(R214x);
R214y = fliplr(R214y);
%1st turn of R1,1
R111x = R221y;
R111y = R221x;
%2nd turn of R1,1
R112x = R222y;
R112y = R222x;
% 3rd turn of Route R1,1
R113x = R112x;
```

```
R113y = -R112y;
R113x = fliplr(R113x);
R113y = fliplr(R113y);
% 4th turn of Route R1,1
R114x = R111x;
R114y = -R111y;
R114x = fliplr(R114x);
R114y = fliplr(R114y);

%1st turn of R1,2
R121x = -R111x;
R121y = R111y;
%2nd turn of R1,2
R122x = -R112x;
R122y = R112y;
% 3rd turn of Route R1,2
R123x = R122x;
R123y = -R122y;
R123x = fliplr(R123x);
R123y = fliplr(R123y);
% 4th turn of Route R1,2
R124x = R121x;
R124y = -R121y;
R124x = fliplr(R124x);
R124y = fliplr(R124y);

%straight sections
xst1 = -((2*KK+1)/2+1)*Dbar+Dbar/delta:Dbar/delta:((2*KK+1)/2+1)*...
    Dbar-Dbar/delta;
xstEW = fliplr(xst1);
yst1 =  (2*KK+1)/2*Dbar*ones(size(xst1));
yst2 = -yst1;
R21xst = xstEW; R21yst = yst2;
```

```
R22xst = xstEW; R22yst = yst1;
yst3 = xstEW; xstNS = yst1; xstNS2 = yst2;
R11xst = xstNS; R11yst = yst3;
R12xst = xstNS2; R12yst = yst3;

%final path
xEWfinal = -((2*KK+1)/2+1)*Dbar-2*R*sin(phi)-Dbar/delta:-Dbar/delta:...
    -ini_point-Dbar;
yEWfinal = zeros(size(xEWfinal));
xNSfinal = yEWfinal;
yNSfinal = xEWfinal;

%nominal path
xEWnom = ini_point-Dbar/delta:-Dbar/delta:-ini_point;
yEWnom = zeros(size(xEWnom));
xNSnom = yEWnom;
yNSnom = xEWnom;

%route definition in correct sequence
R22x = [R221x R222x R22xst R223x R224x xEWfinal];
R22y = [R221y R222y R22yst R223y R224y yEWfinal];
R21x = [R211x R212x R21xst R213x R214x xEWfinal];
R21y = [R211y R212y R21yst R213y R214y yEWfinal];
R12x = [R121x R122x R12xst R123x R124x xNSfinal];
R12y = [R121y R122y R12yst R123y R124y yNSfinal];
R11x = [R111x R112x R11xst R113x R114x xNSfinal];
R11y = [R111y R112y R11yst R113y R114y yNSfinal];
R2x = [xEWnom xEWfinal];
R2y = [yEWnom yEWfinal];
R1x = [xNSnom xNSfinal];
R1y = [yNSnom yNSfinal];
figure(5);
plot(R11x,R11y,'k',R12x,R12y,'k',R21x,R21y,'k',R22x,R22y,'k',...
```

```
    xEWnom,yEWnom,'k',xNSnom,yNSnom,'k')

%Data for aircraft in route
%col 1– index, col2–k, col3– x, col4– y, col5– ti, col6– tf, col7– route,
%col8–pointer
crp_path_length = 4*R*phi + (2*KK+3)*Dbar + Dbar–Dbar/delta;
nom_path_length = 4*R*sin(phi) + (2*KK+3)*Dbar + Dbar–Dbar/delta;

%Route 2 Aircraft (E–W)
for EW_AP=1:length(AP_EW_T_raw)
    %index number of aircraft
    AP_EW(EW_AP,1)=EW_AP;
    %synchronize aircraft by discrete time index "k"
    %AP_EW(EW_AP,2)=round(round(AP_EW_T_raw(EW_AP)/Tdbar_2)/2);
    AP_EW(EW_AP,2)= round(AP_EW_T_raw(EW_AP)/(TDbar));
    %x–position
    AP_EW(EW_AP,3)=AP_EW(EW_AP,2)*Dbar + ini_point;
    %y–position
    AP_EW(EW_AP,4)=0;
    %ti
    AP_EW(EW_AP,5)=AP_EW(EW_AP,2)*Dbar/v_avg;
    %tf
    %AP_EW(EW_AP,6)=AP_EW(EW_AP,5)+crp_path_length/v_avg;
    AP_EW(EW_AP,6)=AP_EW(EW_AP,5)+nom_path_length/v_avg;
    %Route
    AP_EW(EW_AP,7)=2;
    %pointer
    AP_EW(EW_AP,8)=0;
end
%Route 1 Aircraft (N–S)
for NS_AP=1:length(AP_NS_T_raw)
    %index number of aircraft
    AP_NS(NS_AP,1)=NS_AP;
```

```
    %synchronize aircraft by discrete time index "k"
    AP_NS(NS_AP,2)= round(AP_NS_T_raw(NS_AP)/(TDbar));
    %x-position
    AP_NS(NS_AP,3)=0;
    %y-position
    AP_NS(NS_AP,4)=AP_NS(NS_AP,2)*Dbar + ini_point;
    %ti
    AP_NS(NS_AP,5)=AP_NS(NS_AP,2)*Dbar/v_avg;
    %tf
    AP_NS(NS_AP,6)=AP_NS(NS_AP,5)+nom_path_length/v_avg;
    %Route
    AP_NS(NS_AP,7)=1;
    %pointer
    AP_NS(NS_AP,8)=0;
end

%Number of airplanes in each route
EWindex = size(AP_EW);
NSindex = size(AP_NS);

%check if time of arrival input is valid
for EWAC = 1:EWindex
    if AP_EW_T_raw(EWAC) <= 0
        error('Arrival time of route 2 (EW) cannot be 0 or less')
    end
    if EWAC > 1
        if AP_EW(EWAC-1,2) > AP_EW(EWAC,2)
            error('Aircraft in route 2 (EW) is not in chronological order')
        elseif AP_EW(EWAC-1,2) == AP_EW(EWAC,2)
            error('Aircraft in route 2 (EW) has less than min  spacing')
        end
    end
end
```

```
for NSAC = 1:NSindex
    if AP_NS_T_raw(NSAC) < 0
        error('Arrival time of route 1 (NS) cannot be negative')
    end
    if NSAC > 1
        if AP_NS(NSAC-1,2) > AP_NS(NSAC,2)
            error('Aircraft in route 1 (NS) is not in chronological order')
        elseif AP_NS(NSAC-1,2) == AP_NS(NSAC,2)
            error('Aircraft in route 1 (NS) has less than min spacing')
        end
    end
end

% %CRP Path Assignment
NSpointer= 1; %pointer for route 1
EWpointer= 1; %pointer for route 2
k_final = max(max(AP_EW(:,2)),max(AP_NS(:,2))); %final k time arrival data
Arrival_stat = zeros(3,k_final); %initialize arrival matrix
CRP_stat = zeros(3,k_final); %initialize CRP status matrix
Path_matrix = zeros(3,k_final); %initialize path matrix
% Arrival Status for each time index k
for ktime =1:k_final
    %first row of arrival status matrix is k
    Arrival_stat(1,ktime) = ktime;
    %check if R1 has aircraft at k
    if AP_NS(NSpointer,2) ==ktime
        Arrival_stat(2,ktime) = 1;
        NSpointer = NSpointer+1;
    else
        Arrival_stat(2,ktime) = 0;
    end
    %check if R2 has aircraft at k
    if AP_EW(EWpointer,2) ==ktime
```

```
        Arrival_stat(3,ktime) = 1;
        EWpointer = EWpointer+1;
    else
        Arrival_stat(3,ktime) = 0;
    end
end

% CRP Status (Active or Not active) for each time index k
for ktime2 =1:k_final
    CRP_stat(1,ktime2) = ktime2;
%number of aircraft at k
    current_stat = Arrival_stat(2,ktime2)+Arrival_stat(3,ktime2);

    if ktime2 == 1
        if current_stat == 2
            CRP_stat(2,ktime2) = 1; %Active CRP on R1
            CRP_stat(3,ktime2) = 1; %Active CRP on R2
        else
            CRP_stat(2,ktime2) = 0; %No CRP on R1
            CRP_stat(3,ktime2) = 0; %No CRP on R2
        end
    elseif ktime2 >1
        %number of aircraft at k-1
        prev_stat = Arrival_stat(2,ktime2-1)+Arrival_stat(3,ktime2-1);
        if prev_stat == 0 && current_stat < 2
            CRP_stat(2,ktime2) = 0; %No CRP on R1
            CRP_stat(3,ktime2) = 0; %No CRP on R2
        elseif current_stat==2
            CRP_stat(2,ktime2) = 1; %Active CRP on R1
            CRP_stat(3,ktime2) = 1; %Active CRP on R2
            %if CRP active at k-1 and k arrival only at R1
        elseif CRP_stat(2,ktime2-1)+CRP_stat(3,ktime2-1)>=1 &&...
                Arrival_stat(2,ktime2) == 1 && current_stat ==1
```

```
            CRP_stat(2,ktime2) = 1; %Active CRP on R1
            CRP_stat(3,ktime2) = 0; %No CRP on R2
            %if CRP active at k-1 and k arrival only at R2
        elseif CRP_stat(2,ktime2-1)+CRP_stat(3,ktime2-1)>=1 &&...
                Arrival_stat(3,ktime2) == 1 && current_stat ==1
            CRP_stat(2,ktime2) = 0; %No CRP on R1
            CRP_stat(3,ktime2) = 1; %Active CRP on R2
        end
    end
end

% Path Assignment

for ktime3 =1:k_final
    Path_matrix(1,ktime3) = ktime3;
    if ktime3 == 1  %if k=1
        %for R1 aircraft
        if Arrival_stat(2,ktime3) == 1 %if there is R1 arrival at k
            if CRP_stat(2,ktime3) == 0 %if R1 aircraft is not on CRP
                Path_matrix(2,ktime3) = 1; %assign R1
            elseif CRP_stat(2,ktime3) == 1 %if R1 aircraft is on CRP
                Path_matrix(2,ktime3) = 11; %assign R11
            end
        end
        %for R2 aircraft
        if Arrival_stat(3,ktime3) == 1 %if there is R2 arrival at k
            if CRP_stat(3,ktime3) == 0 %if R2 aircraft is not on CRP
                Path_matrix(3,ktime3) = 2; %assign R2
            elseif CRP_stat(3,ktime3) == 1 %if R2 aircraft is on CRP
                Path_matrix(3,ktime3) = 21; %assign R21
            end
        end
    elseif ktime3 > 1 %if k > 1
```

```
%for R1 aircraft
if Arrival_stat(2,ktime3) == 1 %if there is R1 arrival at k
    if CRP_stat(2,ktime3) == 0 %if R1 aircraft is not on CRP
        Path_matrix(2,ktime3) = 1; %assign R1
    elseif CRP_stat(2,ktime3) == 1 %if R1 aircraft is on CRP
        if Arrival_stat(2,ktime3-1)==0 && ...
                Arrival_stat(3,ktime3-1) == 0 %if no arrival at k-1
            Path_matrix(2,ktime3) = 11; %assign R11
        elseif Arrival_stat(2,ktime3-1) == 1 && ...
                Arrival_stat(3,ktime3-1) == 0 %if k-1 arrival on R1
            if Path_matrix(2,ktime3-1) == 1 %if k-1 arrival was
                %assigned to R1
                Path_matrix(2,ktime3) = 12; %assign k arrival to
                %R12
            elseif Path_matrix(2,ktime3-1) == 11 %if k-1 arrival
                %was assigned to R11
                Path_matrix(2,ktime3) = 12; %assign k arrival to
                %R12
            elseif Path_matrix(2,ktime3-1) == 12 %if k-1 arrival
                %was assigned to R12
                Path_matrix(2,ktime3) = 11; %assign k arrival to
                %R11
            end
        elseif Arrival_stat(2,ktime3-1) == 0 &&...
                Arrival_stat(3,ktime3-1) == 1 %if k-1 arrival on
            %R2 only
            if Path_matrix(3,ktime3-1) == 2 %if R2 k-1 arrival
                %was assigned to R2
                Path_matrix(2,ktime3) = 11; %assign R1 k arrival to
                %R11
            elseif Path_matrix(3,ktime3-1) == 21 %if R2 k-1 arrival
                %was assigned to R21
                Path_matrix(2,ktime3) = 12; %assign R1 k arrival to
```

```
                %R12
            elseif Path_matrix(3,ktime3-1) == 22 %if R2 k-1 arrival
                %was assigned to R12
                Path_matrix(2,ktime3) = 11; %assign R1 k arrival to
                %R11
            end
        elseif Arrival_stat(2,ktime3-1)==1 &&...
                Arrival_stat(3,ktime3-1) == 1 %if both R1 and R2
            %arrival at k-1
            if Path_matrix(2,ktime3-1) == 11 %if k-1 arrival was
                %assigned to R11
                Path_matrix(2,ktime3) = 12; %assign k arrival to
                %R12
            elseif Path_matrix(2,ktime3-1) == 12 %if k-1 arrival
                %was assigned to R12
                Path_matrix(2,ktime3) = 11; %assign k arrival to
                %R11
            end
        end
    end
end
%for R2 aircraft
if Arrival_stat(3,ktime3) == 1 %if there is R2 arrival at k
    if CRP_stat(3,ktime3) == 0 %if R2 aircraft is not on CRP
        Path_matrix(3,ktime3) = 2; %assign R2
    elseif CRP_stat(3,ktime3) == 1 %if R2 aircraft is on CRP
        if Arrival_stat(2,ktime3-1)==0 &&...
                Arrival_stat(3,ktime3-1) == 0 %if no arrival at k-1
            Path_matrix(3,ktime3) = 21; %assign R11
        elseif Arrival_stat(2,ktime3-1) == 1 &&...
            Arrival_stat(3,ktime3-1) == 0 %if k-1 arrival on R1only
            if Path_matrix(2,ktime3-1) == 1 %if R1 k-1 arrival was
                %assigned to R1
```

```
            Path_matrix(3,ktime3) = 22; %assign R2 k arrival to
                %R22
        elseif Path_matrix(2,ktime3-1) == 11 %if R1 k-1 arrival
                %was assigned to R11
            Path_matrix(3,ktime3) = 22; %assign R2 k arrival to
                %R22
        elseif Path_matrix(2,ktime3-1) == 12 %if R1 k-1 arrival
                %was assigned to R12
            Path_matrix(3,ktime3) = 21; %assign R2 k arrival to
                %R21
        end
    elseif Arrival_stat(2,ktime3-1) == 0 &&...
            Arrival_stat(3,ktime3-1) == 1 %if k-1 arrival on R2
        %only
        if Path_matrix(3,ktime3-1) == 2 %if R2 k-1 arrival was
                %assigned to R2
            Path_matrix(3,ktime3) = 21; %assign R2 k arrival to
                %R21
        elseif Path_matrix(3,ktime3-1) == 21 %if R2 k-1 arrival
                %was assigned to R21
            Path_matrix(3,ktime3) = 22; %assign R2 k arrival to
                %R22
        elseif Path_matrix(3,ktime3-1) == 22 %if R2 k-1 arrival
                %was assigned to R12
            Path_matrix(3,ktime3) = 21; %assign R2 k arrival to
                %R11
        end
    elseif Arrival_stat(2,ktime3-1)==1 &&...
            Arrival_stat(3,ktime3-1) == 1 %if both R1 and R2
        %arrival at k-1
        if Path_matrix(3,ktime3-1) == 21 %if k-1 arrival was
                %assigned to R21
            Path_matrix(3,ktime3) = 22; %assign k arrival to
```

```
                    %R22
                elseif Path_matrix(3,ktime3-1) == 22 %if k-1 arrival
                    %was assigned to R22
                    Path_matrix(3,ktime3) = 21; %assign k arrival to
                    %R21
                end
            end
        end
    end
end

%storing back to original data
NS_store =1; %storing index for R1
EW_store =1; %storing index for R2
for ktime4 =1:k_final
    if Path_matrix(2,ktime4) ~= 0
        AP_NS(NS_store,7) = Path_matrix(2,ktime4);
        if Path_matrix(2,ktime4) == 11 || Path_matrix(2,ktime4) ==12
            AP_NS(NS_store,6) =  AP_NS(NS_store,5)+crp_path_length/v_avg;
            %change exit time to CRP path time
        end
        NS_store = NS_store+1; %increment R1 pointer
    end
    if Path_matrix(3,ktime4) ~= 0
        AP_EW(EW_store,7) = Path_matrix(3,ktime4);
        if Path_matrix(3,ktime4) == 21 || Path_matrix(3,ktime4) ==22
            AP_EW(EW_store,6) =  AP_EW(EW_store,5)+crp_path_length/v_avg;
            %change exit time to CRP path time
        end
        EW_store = EW_store+1; %increment R2 pointer
    end
end
```

```
%simulation
sp_index = 1; %speed of simulation (increment interval)
deltaT= Dbar/delta/v_avg;
finalT = round(max(AP_EW(end,6),AP_NS(end,6))/(sp_index*deltaT));

%Data for aircraft in route
%col 1– index, col2–k, col3– x, col4– y, col5– ti, col6– tf, col7– route,
%col8–pointer
%col 4 and col5 initial way point based on 2Dbar away of initial turn
figure(6);
sim_movie = avifile('crp_ondemand_final.avi','quality',100);
for tindex = 1:finalT
    % simulation time
    t= tindex * deltaT * sp_index;
    % route 2 aircraft
    for EW=1:EWindex(1)
        if AP_EW(EW,5)> t %if t>ti (before initial way point) increment x
            AP_EW(EW,3) = AP_EW(EW,3)–deltaT*v_avg*sp_index;
            AP_EW(EW,4) = 0;
        elseif AP_EW(EW,5)<= t && AP_EW(EW,6)>t %if ti<t<tf
            AP_EW(EW,8) = AP_EW(EW,8)+sp_index;    %increment pointer
            if AP_EW(EW,7)==21              %if route is R21 increment
                %position in R21
                AP_EW(EW,3) = R21x(AP_EW(EW,8));
                AP_EW(EW,4) = R21y(AP_EW(EW,8));
            elseif AP_EW(EW,7)==22          %if route is R22 increment
                %position in R22
                AP_EW(EW,3) = R22x(AP_EW(EW,8));
                AP_EW(EW,4) = R22y(AP_EW(EW,8));
            elseif AP_EW(EW,7)==2          %if route is R2 increment x
```

```
            %position
            AP_EW(EW,3) = R2x(AP_EW(EW,8));
            AP_EW(EW,4) = 0;
        end

    elseif AP_EW(EW,6) < t  %if t> tf (after final way point)
        AP_EW(EW,3) = AP_EW(EW,3)-deltaT*v_avg*sp_index;
        AP_EW(EW,4) = 0;
    end
end
%R1 aircraft, same algorithm as R2 aircraft above
for NS=1:NSindex(1)
    if AP_NS(NS,5)> t
        AP_NS(NS,3) = 0;
        AP_NS(NS,4) = AP_NS(NS,4)-deltaT*v_avg*sp_index;
    elseif AP_NS(NS,5)<= t && AP_NS(NS,6)>t
        AP_NS(NS,8) = AP_NS(NS,8)+sp_index;
        if AP_NS(NS,7)==11
            AP_NS(NS,3) = R11x(AP_NS(NS,8));
            AP_NS(NS,4) = R11y(AP_NS(NS,8));
        elseif AP_NS(NS,7)==12
            AP_NS(NS,3) = R12x(AP_NS(NS,8));
            AP_NS(NS,4) = R12y(AP_NS(NS,8));
        elseif AP_NS(NS,7)==1      %if route is R1 increment y position
            AP_NS(NS,3) = 0;
            AP_NS(NS,4) = R1y(AP_NS(NS,8));
        end

    elseif AP_NS(NS,6)<t
        AP_NS(NS,3) = 0;
        AP_NS(NS,4) = AP_NS(NS,4)-deltaT*v_avg*sp_index;
    end
end
```

```
    plot(AP_EW(:,3),AP_EW(:,4),'r<',AP_NS(:,3),AP_NS(:,4),'bv')
    hold on
    plot(R11x,R11y,'k',R12x,R12y,'k',R21x,R21y,'k',R22x,R22y,'k',...
        xEWnom,yEWnom,'k',xNSnom,yNSnom,'k')

        for EWN =1:EWindex(1)
            line(AP_EW(EWN,3)+Dsep/2*cos([0:pi/50:2*pi]),...
            AP_EW(EWN,4)+Dsep/2*sin([0:pi/50:2*pi]),'Color','r');
        end
        for NSN =1:NSindex(1)
            line(AP_NS(NSN,3)+Dsep/2*cos([0:pi/50:2*pi]),...
            AP_NS(NSN,4)+Dsep/2*sin([0:pi/50:2*pi]),'Color','b');
        end

     axis([-35 35 -35 35])
     axis equal
     axis manual
          drawnow
     hold off
     M_get = getframe;
     sim_movie = addframe(sim_movie,M_get);
end
sim_movie = close(sim_movie);
```

BIBLIOGRAPHY

[1] Y. Zhao and R. Slattery. Capture conditions for merging trajectory segments to model realistic aircraft descents. *Journal of Guidance, Control and Dynamics*, 19(2):453 – 460, 1996.

[2] X. Xidong and I. Wilson. Automation stage and level in future air traffic management. *Integrated Communications Navigation and Surveillance Conference (ICNS)*, pages 1–19, 2010.

[3] S. Devasia, D. Iamratanakul, G. Chatterji, and G. Meyer. Decoupled conflict-resolution procedures for decentralized air traffic control. *Proceedings of the Conference of Control Applications, IEEE Trans. on Intelligent Transportation Systems*, 12(2):422–437, June 2011.

[4] B. Sridhar, S. R. Grabbe, and A. Mukherjee. Modeling and optimization in traffic flow management. *Proc. of the IEEE*, 96(12):2060–2080, Dec. 2008.

[5] M. Prandini, L. Piroddi, S. Puechmorel, and S.L. Brzdilova. Toward air traffic complexity assessment in new generation air traffic management systems. *IEEE Trans. Intel. Trans. Sys.*, 12(3):809–818, 2011.

[6] S. W. Evans, D. E. Antony, and H. R. Idris. Benefits assessment of improved decisions for rerouting flights around flow constraint areas. *AIAA Guidance, Navigation, and Control Conference and Exhibit, Providence, Rhode Island*, pages 1–17, AIAA–2004–5319, August 2004.

[7] J. K. Kuchar and L. C. Yang. A review of conflict detection and resolution modeling methods. *IEEE Trans. on Intelligent Transportation Systems*, 1(4):179–189, December 2000.

[8] Air Traffic Organization U.S. Federal Aviation Administration. A plan for the future 10-year strategy for the air traffic control workforce 2013-2022. *U.S. Department of Transportation, Federal Aviation Administration*, 2013.

[9] Air Traffic Organization U.S. Federal Aviation Administration. Aeronautical information manual. *U.S. Department of Transportation, Federal Aviation Administration*, 2012.

[10] Air Traffic Organization U.S. Federal Aviation Administration. A plan for the future 10-year strategy for the air traffic control workforce 2012-2021. *U.S. Department of Transportation, Federal Aviation Administration*, 2012.

[11] Air Traffic Organization U.S. Federal Aviation Administration. A plan for the future 10-year strategy for the air traffic control workforce 2011-2020. *U.S. Department of Transportation, Federal Aviation Administration*, 2011.

[12] Air Traffic Organization U.S. Federal Aviation Administration. A plan for the future 10-year strategy for the air traffic control workforce 2010-2019. *U.S. Department of Transportation, Federal Aviation Administration*, 2010.

[13] Air Traffic Organization U.S. Federal Aviation Administration. A plan for the future 10-year strategy for the air traffic control workforce 2009-2018. *U.S. Department of Transportation, Federal Aviation Administration*, 2009.

[14] Bureau of Transportation Statistics U.S. Department of Transportation. Air carrier summary : T1: U.s. air carrier traffic and capacity summary by service class. `http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=264&DB_Short_Name=Air%20Carrier%20Summary`. [Online; accessed February-2014].

[15] Civil Aeronautics Board. Handbook of airline statistics 1969. *Civil Aeronautics Board Technical Report*, pages part III, table 2, 1970.

[16] Civil Aeronautics Board. Handbook of airline statistics 1973. *Civil Aeronautics Board Technical Report*, pages part III, table 2, 1974.

[17] Bureau of Labor Statistics U.S. Department of Labor. Job opening and labor turnover - january 2013. *Bureau of Labor Statistics News Release*, (USDL-13-0422), January 2013.

[18] A. Majumdar, Ochieng W. Y., G. McAuley, J. M. Lenzi, and C. Lepadatu. The factors affecting airspace capacity in europe: A framework methdology based on cross sectional time-series analysis using simulated controller workload data. *Journal of Navigation*, 57(3):385–405, September 2004.

[19] National Air Traffic Controllers Association. NACTA A History of Air Traffic Control. `http://www.natca.com/ULWSiteResources/natcaweb/Resources/file/Media%20Center/ATCHistory.pdf`, 2011. [Online; accessed November-2011].

[20] G. A. Gilbert. Historical development of the air traffic control system. *IEEE Trans. on Communications*, (5):364–375, May 1973.

[21] U.S. Federal Aviation Administration. Fact sheet air traffic control management: 75 years and counting, 1926-1996. `http://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=12904&print=go/`, 2011. [Online; accessed February-2014].

[22] L. Warren. Historical development of the air traffic control system. *AIAA 6th Annual Meeting and Technical Display, Anaheim, CA*, (69-1051), October 1969.

[23] U.S. Centennial of Flight Commission. Air Traffic Control. `http://www.centennialofflight.gov/essay/Government_Role/Air_traffic_control/POL15.htm`, 2011. [Online; accessed November-2011].

[24] R. E. Bilstein. Orders of Magnitude: A History of NACA and NASA, 1915-1990. Washington: National Aeronautics and Space Administration. 1989.

[25] U.S. Federal Aviation Administration. FAA historical chronology, 1926-1996. `http://www.faa.gov/about/media/b-chron.pdf/`, 2011. [Online; accessed November-2011].

[26] J. Prete, J. Krozel, J. Mitchell, J. Kim, and J. Zou. Flexible, performance-based route planning for super-dense operations. *AIAA Guidance, Navigation and Control Conference, Honolulu, Hawaii, paper number 2008-6825*, pages 1–14, August 18-21 2008.

[27] V. Lund. Automated conflict resolution in air traffic management. Master's thesis, The university of Utah, August 2000.

[28] RTCA. Final report of radio technical commisions for aeronautics task force 3: Free flight implementation. *RTCA Technical Report*, October 1995.

[29] H. Erzberger. Design principles and algorithms for automated air traffic management. *AGARD Lecture Series No. 200, Knowledge-Based Functions in Aerospace Systems, San Francisco*, November, 1995.

[30] W. H. Johnson. Automation and attention management considerations in naval uav control systems. *AIAA's 1st Technical Conference and Workshop on Unmanned Aerospace Vehicles, S, Portsmouth, VA, May 20 - May 23, 2002*, 2002.

[31] I. McManus and R. Walker. A multidisciplinary approach to uav planning in civilian airspace. *AIAA Guidance, Navigation, and Control Conference and Exhibit, Providence, RI, May 16 - May 19, 2004*, 2004.

[32] Y. Eun, I. Hwang, and H. Bang. Optimal arrival flight sequencing and scheduling using discrete airborne delays. *IEEE Trans. on Intelligent Transportation Systems*, 11(2):359–373, March 2010.

[33] C. Quinn and Zelenka R. Air traffic control/air carrier collaborative arrival planning. *Air Transportation Systems Engineering, AIAA*, pages 177–189, 2001.

[34] T. Weidner and Green S. Modeling air traffic management automation metering conformance benefits. *Air Transportation Systems Engineering, AIAA*, pages 367–382, 2001.

[35] A. Michelin, M. Idan, and J.-L. Speyer. Merging of air traffic flows. *JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS, January-February*, 34(1), 2011.

[36] M. M. Poulose. Microwave landing system modeling with application to air traffic control automation. *JOURNAL OF AIRCRAFT, May-June*, 29(3), 1992.

[37] H. Erzberger and W. Nedell. Design of automated system for management of arrival traffic. *NASA Tech. Memorandum 102201*, June, 1989.

[38] U.S. Federal Aviation Administration. Off the ground and into the air faster with nextgen at dfw. `http://www.faa.gov/nextgen/snapshots/stories/?slide=28&cid=TW202`, 2014. [Online; accessed May-2014].

[39] K. D. Bilimoria, K. S. Seth, H. Q. Lee, and S. R. Grabbe. Performance evaluation of airborne separation assurance for free flight. *AIAA Guidance Navigation and Control Conference, August 14-17, Denver, CO*, pages 1–9, AIAA 2000–4269, 2000.

[40] W. B. Cotton. Formulation of the air traffic system as a management problem. *IEEE Trans. on Communications*, 21(5):375–382, May 1973.

[41] G. Chatterji, B. Shridar, and K. Bilimoria. En-route flight trajectory prediction for conflict avoidance and traffic management. *AIAA Guidance Navigation and Control Conference, July 29-31, San Diego, CA*, pages 1–11, 1996.

[42] C. B. Lockleae. A severe weather climatology for the wfo raleigh, nc county warning area. *National Weather Service Weather Forecast Office, Raleigh North Carolina*, 2008. [Online; accessed April-2014].

[43] Z.-H. Mao, D. Dugail, and E. Feron. Space partition for conflict resolution of intersecting flows of mobile agents. *IEEE Trans. on Intelligent Transportation Systems*, 8(3):512–527, September 2007.

[44] Z.-H. Mao, D. Dugail, E. Feron, and K. Bilimoria. Stability of intersecting aircraft flows using heading-change maneuvers for conflict avoidance. *IEEE Trans. on Intelligent Transportation Systems*, 6(4):357–369, December 2005.

[45] Z.-H. Mao, E. Feron, and K.D. Bilimoria. Stability of intersecting aircraft flows under decentralized conflict avoidance rules. *AIAA Guidance, Navigation, and Control Conference and Exibit, Denver, CO*, August 2000. Paper No. 2000-4271.

[46] D. Iamratanakul, G. Meyer, G. B. Chatterji, and S. Devasia. Quantification of airspace sector capacity using decentralized conflict resolution procedures. *Proc. of the Conf. on Decision and Control, Paradise Island, Bahamas*, pages 2003 – 2009, Dec. 14-17 2004.

[47] J. Yoo and S. Devasia. Provably safe conflict resolution with bounded turn rate for air traffic control. *IEEE Trans. on Control Systems Technology*, 21(6):2280–2289, November 2013.

[48] J. D. Yoo and S. Devasia. Decoupled conflict resolution procedures for non-perpendicular air traffic intersections with different speeds. *Conference on Decision and Control, Florence, Italy*, December 2013.

[49] J. D. Yoo and Devasia S. On-demand conflict resolution procedures for air traffic intersections. *appearing in IEEE Trans. on Intellegent Transportation Systems*, PP:1–12, February 2014.

[50] J. Krozel, C. Lee, and J. S. B Mitchell. Turn-constrained route planning for avoiding hazardous weather. *Air Traffic Control Quarterly*, 14(2):159–182, 2006.

[51] Aslaug Haraldsdottir, Robert W. Schwab, and Monica S. Alcabin. Air traffic management capacity-driven operational concept through 2015. *2nd International Air Traffic Management R&D seminar ATM-98 Sponsored by The US Federal Aviation Administration and EUROCONTROL*, Orlando,1st - 4th December 1998.

[52] Y. Wan and S. Roy. A scalable methodology for evaluating and designing coordinated air-traffic flow management strategies under uncertainty. *IEEE Trans. on Intelligent Transportation Systems*, 9(4):644–656, Dec. 2008.

[53] A. L. Visintini, W. Glover, J. Lygeros, and J. Maciejowski. Monte carlo optimization for conflict resolution in air traffic control. *IEEE Trans. on Intelligent Transportation Systems*, 7(4):470–482, Dec. 2006.

[54] L. Pallottino, E. M. Feron, and A. Bicchi. Conflict resolution problems for air traffic management systems solved with mixed integer programming. *IEEE Trans. on Intelligent Transportation Systems*, 3(1):3–11, Mar. 2002.

[55] C. Tomlin, I. Mitchell, and R. Ghosh. Safety verification of conflict resolution maneuvers. *IEEE Trans. on Intelligent Transportation Systems*, 2(2):110–120, June 2001.

[56] S. Devasia and G. Meyer. Automated conflict resolution procedures for air traffic management. *Proceedings of the 36th Conference on Decision and Control, Phoenix, AZ*, pages 2456–2462, December 1999.

[57] A.M. Bayen, R.L. Raffard, and C.J. Tomlin. Adjoint-based control of a new eulerian network model of air traffic flow. *IEEE Trans. on Control Systems Technology*, 14(5):804–818, September 2006.

[58] R. G. Mullholland and D.-W. Stout. Stereographic projection in the national airspace system. *IEEE Trans. on Aerospace and Electronic Systems*, AES-18:48–57, 1982.

[59] M.-S. Refai and R. Windhorst. Impact of tactical and strategic weather avoidance on separation assurance. *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, September 20-22, Virginia Beach, VA*, 2011.

[60] R.-A. Paielli. Tactical conflict resolution using vertical maneuvers in en route airspace. *JOURNAL OF AIRCRAFT, November-December*, 45(6), 2008.

[61] F. C. Holland, R. A. Rucker, and B. M. Horowitz. Structure of the airspace. *IEEE Trans. on Communications*, 21(5):382–398, May 1973.

[62] J. E. Robinson III and D. R. Isaacson. A concurrent sequencing and deconflicting algorithm for terminal area air traffic control. *AIAA Guidance Navigation and Control Conference, August 14-17, Denver, CO*, pages 1–11, AIAA 2000–4473, 2000.