On Dynamic Capacity-Demand Balance in the Terminal Area Using Multi-objective Co-evolutionary Computational Red Teaming

Author:
Zhao, Wenjing

Publication Date:
2012

DOI:
https://doi.org/10.26190/unsworks/15796

On Dynamic Capacity-Demand Balance in the Terminal Area Using Multi-objective Co-evolutionary Computational Red Teaming

Wenjing Zhao

B.Eng.(Engineering Management), Shandong Institute of Business and Technology, China

A thesis submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy at the
School of Engineering and Information Technology
University of New South Wales
Australian Defence Force Academy

Abstract

Future air traffic management (ATM) systems are expected to handle the increasingly heavy demand on air traffic, especially in the highly constrained terminal area (TeA). However, the realizable capacity of current TeA is a challenge for future air transportation development. This is due to the limitations in accommodating safe and efficient travel under the highly limited airspace configuration strategies and pre-defined terminal trajectories. Therefore, making the TeA resources flexible and available corresponding to different traffic scenarios is the key to enhance the practical ATM efficiency in future TeAs.

Improving the TeA airspace configuration to balance capacity and demand is a challenging task, since a TeA system inherently involves high uncertainties and multiple interactions among many different components. The inherent complexity of the TeA necessitates a system-level analysis approach, in which each component is investigated through modelling the complex interactions among other parts of the environment in which it operates. Hence, the process of understanding (through modelling), evaluating and dynamically designing TeA airspace configurations, while considering dynamic constrained ground resources, is becoming crucial for enhancing the practical ATM efficiency in future TeAs.

In this thesis, an air traffic simulation system with a novel representation of an integrated TeA that considers the air-ground collaboration and arrival-departure cooperation is presented for system-level modelling of TeA concepts. Then a simulation-based co-evolutionary computational environment – Co-evolutionary Computational Red Teaming (CCRT) – is developed for evaluating advanced TeA airspace concepts and understanding the TeA system-level vulnerabilities. The interactions between traffic distributions and constrained ground resources (including runways, taxiways and gates) are co-evolved with each other and considered from the perspective of identifying inefficiencies, with the integration of arrival and departure operations. By evaluating these interactions, we are able to reveal "improvement opportunities" in the implementation of future TeA airspace concepts and, thereby, understand major bottlenecks which cause system inefficiencies.

A multi-objective co-operative co-evolutionary methodology is then proposed as a new optimization search engine of the CCRT framework, in order to solve complex TeA problems with multiple conflicting objectives.

A novel TeA airspace design concept for capacity-demand balancing including a measure of collision risks derived from the probabilistic nature of aircraft's performance is proposed. Then, an air traffic simulator, originally representing the novel TeA airspace design concept while considering the interactions among dynamic ground events, is presented. The multi-objective CCRT is applied to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce dynamic trajectories while maintaining ATM efficiency and aircraft safety. The multi-objective CCRT also provides an analyst with the trade-off between these two air traffic control priorities - efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand.

In summary, the contributions of this thesis are:

- A methodology to evaluate advanced TeA airspace concepts and understand TeA system vulnerabilities.
- A multi-objective co-operative co-evolutionary methodology proposed for co-evolving solutions towards the efficient set of trade-offs effectively, while maintaining diversity of the solution set.
- A methodology to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety.

keywords

Terminal Area, Future Airspace Design, Collision Risk, Ground-air Network, Arrival-Departure Integration, Co-evolutionary Algorithms, Multi-objective Optimization, Computational Red Teaming

Acknowledgement

No words of gratitude are enough to thank my supervisor Prof. Hussein A. Abbass, whose untiring support, motivation, supervision and guidance in all aspects of my work provide me great help on various topics.

I also wish to acknowledge my co-supervisor, Dr. Axel Bender (DSTO) and Dr. Sameer Alam, for their guidance and support throughout my thesis. I owe my sincere thanks to my colleagues at the Computational Intelligence, Modelling, and Cognition Lab. (CIMC), Dr. Lam T. Bui, Dr. Jiangjun Tang, Prof. Jing Liu, Dr. Weicai Zhong, Fan Qi, Van Viet Pham, Shir Li Wang, Amr Ghoneim, George Leu, whom I enjoyed my research collaboration. I am also grateful to my dear colleagues and friends: Qiang Li, Theam Foo Ng, Nizami Jafarov.

I would like to extend my hearty appreciation to the anonymous reviewers for my journal and conference papers whose valuable suggestions were instrumental in improving the research work.

I would like to acknowledge, with gratitude and appreciation, financial support from the China Scholarship Council and the University of New South Wales at the Australian Defense Force Academy, Canberra.

Finally, to my mother Xiaobo Wang and father Juquan Zhao, for their constant encouragement, support and love during this endeavor.

Wenjing Zhao
Australia, 2012

Certificate of Originality

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by colleagues, with whom I have worked at UNSW or elsewhere, during my candidature, is fully acknowledged.

I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.

Wenjing Zhao

Contents

List of Figures

List of Tables

List of Acronyms

ACE	Automated Co-Evolution
ADI	Arrival-Departure Integration
ART	Automated Red Teaming
ATA	Actual Time of Arrival
ATC	Air Traffic Control
ATD	Actual Time of Departure
ATFM	Air Traffic Flow Management
ATOMS	Air Traffic Operations & Management Simulator
ATM	Air Traffic Management
BADA	Eurocontrol's Aircraft Database
BI	Best Individual
CAD	Continuous Ascent Departure
CC	Competitive Co-evolution
CCEA	Co-operative Co-evolutionary Algorithm
CCGAs	Co-operative Co-evolutionary Genetic Algorithms
CCMOEA	Multi-objective Co-operative Co-evolutionary Algorithms
CCRT	Co-operative Co-evolutionary Red Teaming
CDA	Continuous Descent Approach
CEA	Co-Evolutionary Algorithm
CeC	Co-evolutionary Computation
CoC	Co-operative Co-evolution
CoCA	Co-operative Co-evolutionary Algorithm
CLOU	Cooperative Local Resource Planner
CP	Choosing Pool
CR	Collision Risk
CRT	Computational Red Teaming
DAC	Dynamic Airspace Configuration
EA	Evolutionary Algorithm
EC	Evolutionary Computation
ETA	Estimated Time of Arrival
ETE	Estimated Time of Execution
ETD	Estimated Time of Departure
FAA	Federal Aviation Adminstration
FAF	Final Arrival Fix
FCFA	First Come First Assigned
FMS	Flight Management System

GAs	Genetic algorithms
GD	Generation Distance
IAF	Initial Arrival Fixe
IFR	Instrument Flying Rules
ILS	Instrument Landing System
JPDO	Joint Planning and Development Office
MAS	Multi-Agent Systems
MEBRA	Multi-objective Based Risk Assessment
MO	Multi-Objective
MOBNET	Multi-objective Co-operative Networks
MOCCA	Multi-Objective Co-operative Co-evolutionary Algorithm
MOEA	Multi-Objective Evolutionary Algorithm
MS	Maximum Spread
NASA	National Aeronautics and Space Administration
NextGen	Next Generation Air Transportation System
NPGA	Niched Pareto Genetic Algorithm
NSCCGA	Non-dominated Sorting Co-operative Co-evolutionary Algorithm
NSGAII	Non-dominated Sorting Genetic Algorithm II
OM	Outer Marker
OPDs	Optimized Profile Descents
POS	Set of Pareto-Optimal Solutions
RNAV	Terminal Area Navigation
RNP	Required Navigation Performance
RI	Random Individual
RT	Red Teaming
SDO	Super-Density Operations
SEBRA	Single-objective Based Risk Assessment
SESAR	Single European Sky ATM Research
SID	Standard Instrument Departure Procedure
SO	Single-Objective
SPEA2	Strength Pareto Evolutionary Algorithm 2
STAR	Standard Terminal Arrival Route
SUA	Special Use Airspace
TA	Terminal Airspace
TAPSS	Terminal Area Precision Scheduling and Spacing System
TAR	Transition Airspace Radius
TARGETS	Terminal Area Route Generation, Evaluation, and Traffic Simulation Tool
TAs	Tailored Arrivals
TeA	Terminal Area
TOC	Top of Climb
TOD	Top of Decent
TOOWiLD	Trajectory-Oriented Operations with Limited Delegation

List of Publications

Peer-reviewed publications arising from research work conducted in this thesis are listed chronologically below (latest to earliest):

Journal Publications

1. W. Zhao, S. Alam, H. A. Abbass, "Evaluating Ground-Air Network Vulnerability in an Integrated Terminal Maneuvering Area using Co-evolutionary Computational Red Teaming", Transportation Research Part C, Elsevier, (Conditionally Accepted) 2012.
2. S. Alam, W. Zhao, J. Tang, C. Lokan, H. A. Abbass, M. Ellejmi, S. Kirby, "Discovering Delay Patterns in Arrival Traffic with Dynamic Continuous Descent Approaches using Co-Evolutionary Red Teaming", Air Traffic Control Quarterly, Air Traffic Control Association Institute, Inc., vol. 20, no. 1, pp.47-72, 2012.
3. W. Zhao, S. Alam, H. A. Abbass, "S-MOCCA: A Systemic Multi-objective Cooperative Co-evolutionary Algorithm", submitted to Applied Soft Computing, 2012.
4. W. Zhao, S. Alam, M. Ellejmi, H. A. Abbass, "Computational Red Teaming for Dynamic Airspace Design to Reduce Delays and Risk", submitted to AIAA Journal of Aircraft, 2012.

Conference Publications

1. W. Zhao, J. Tang, S. Alam, A. Bender, H. A. Abbass, "Evolutionary-Computation Based Risk Assessment of Aircraft Landing Sequencing Algorithms", *The World Computer Congress, Brisbane, Australia, Sep*, 2010.
2. W. Zhao, J. Liu, H. A. Abbass and A. Bender, "A Multi-objective Risk-Based Approach for Airlift Task Scheduling Using Stochastic Bin Packing", *IEEE Congress on Evolutionary Computation (IEEE-CEC), Barcelona, Spain, Jul*, 2010.

Chapter 1

Introduction

1.1 Overview

Air transport demand continues to grow faster than the available system capacity, leading to a gradual increase of the number of flight delays in the global air transportation system (Nolan, 2004). Future air traffic management (ATM) requires maximizing potential air traffic capacity to alleviate the growing capacity/demand imbalance, especially in the highly constrained terminal area TeA. However, the realizable capacity of a current TeA is ultimately limited to its capability to accommodate safe and efficient travel under current highly limited airspace configuration strategies and pre-defined terminal trajectories. Therefore, designing safe and efficient TeA airspace configuration strategies given different traffic scenarios is the key to solve the practical ATM issues in future TeAs.

A typical TeA naturally involves multiple interactions among many different components – such as the inherent uncertainty of the availability of ground resources (including runways, taxiways and gates), interactions between air traffic distributions (spatial and temporal) and constrained ground events, incoming arrivals and outgoing departures concurrently passing through the TeA airspace, and an airport competing for the same resources (e.g., runways and taxiways). To date, the majority of work has focused on solving the ATM problem in a TeA in a highly isolated manner, in which each component is investigated without modelling the complex interactions among other parts of the environment in which it operates. Little discussion has been dedicated to the role of systematic approaches, an important factor that directly affects ATM performance. The inherent complexity of a TeA necessitates a system-level analysis to understand its overall system behavior and vulnerabilities, in order to facilitate effective design of efficient and safe TeA configuration strategies, in the presence of dynamic constrained ground resources.

This thesis first considers a TeA system which integrates arrival and departure operations and combines air and ground resources. This system-level modelling of a TeA helps us to understand the behavior and complexity of the entire TeA system. It paves a way for discovering major bottlenecks which cause system inefficiencies and evaluating advanced TeA airspace concepts. Furthermore, a concept of dynamic airspace design for TeA airspace is also introduced in this thesis. A major component of this concept is to optimize the scenario-specific TeA airspace design strategies to maximize the TeA efficiency and safety, given the interdependencies from dynamic constrained ground resources. A novel way to measure the collision risk, which is derived from the probabilistic nature of aircraft's performance, is presented and implemented in the proposed concept of TeA airspace design.

The scenario space of an integrated TeA comprises interweaving scenarios which are correlated in time and space. As a result, causes and effects are networked and the dynamics of system's components become complex. This level of complexity necessitates a simulation-based approach and requires more sophisticated quantitative approaches capable of analyzing the network of interdependencies and evaluating system-level vulnerability (Abbass *et al.*, 2009).

Computational Red Teaming (CRT) (Abbass *et al.*, 2011), is a computational environment that integrates computational intelligence techniques and multi-agent systems. It offers the means to search massive spaces of possibilities governed by uncertainty and complex networked dynamics quickly and find possible and relevant solutions in these spaces. The basic philosophy of CRT is to provide a computational environment in which competition between the system under investigation and scenarios representing situations of risk or uncertainty can be modelled, and superior strategies can be identified (Abbass *et al.*, 2011). CRT has been used successfully to evaluate risk in safety-critical operations (Abbass *et al.*, 2011).

In an integrated TeA system, the air-side and ground-side subcomponents are highly co-adapted as each subcomponent itself is changing and evolving. A performance evaluation of each subcomponent depends on the reciprocal interactions with other components in the system and any change in context can be well captured by a co-evolutionary algorithm (CEA) which is a biologically-inspired population-based search technique. Because a CEA provides an effective means of handling large and complex problems via problem decomposition, it seems natural to use it in a problem domain in which solutions can be evolved through interactions of co-adapted subcomponents, rather than by hand tuning or pre-scripting how scenarios should change.

A simulation-based co-evolutionary computational environment – Co-evolutionary Computational Red Teaming (CCRT) – is developed for evaluating advanced TeA airspace concepts and understanding the TeA system vulnerabilities. Interactions between traffic distributions and constrained ground resources (including runways, taxiways and gates) are co-evolved with each other and considered from the perspective of identifying inefficiencies, with the integration of arrival and departure operations. By evaluating these interactions, we are able to reveal "improvement opportunities" in the implementation of future TeA airspace concepts and, thereby, understand major bottlenecks which cause system inefficiencies.

Although the proposed CCRT was originally designed for single objective problems, most ATM problems naturally involve multiple conflicting objectives, such as efficiency versus safety. Contrary to a single-objective CEA, the multi-objective co-operative co-evolutionary algorithm (MOCCA) does not have a single solution that optimizes all criteria concerned, but a set of trade-off solutions, known as Pareto-optimal solutions. Any Pareto-optimal solution is optimal in the sense that no improvement can be made in one criterion without degradation of at least another criterion. Since none of the solutions in the Pareto-optimal set is absolutely better than any other, anyone of them is an acceptable solution. Which solution should be chosen depends on the decision-makers, preferences and various problem-related factors. Hence, a decision-maker is typically interested in knowing as many Pareto-optimal solutions as possible (Deb and Kalyanmoy, 2001).

A MOCCA is proposed in this thesis to co-evolve individuals toward the true global Pareto front effectively and maintain a high diversity of the solution set. The multi-objective CCRT is applied to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety. The multi-objective CCRT also provides an analyst of the trade-off between these two ATC priorities - efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand.

The Air Traffic Operations and Management Simulator (ATOMS) (Alam *et al.*, 2008) provides a high-fidelity simulation and modelling environment for the exploration, development and evaluation of advanced ATM concepts. In this thesis, ATOMS is modified and extended in two ways. Firstly, the function of integrating arrivals and departures using a shared ground-air network, which can implement different TeA airspace models, such as the present day's arrival and departure procedure models (STARs and SIDs) and an advanced ATM concept known as dynamic Continuous Descent Approaches (CDA) (Alam *et al.*, 2010a) are included. Secondly, the execution of a novel TeA airspace design concept considering the dynamic ground events for given arrival traffic scenarios is embedded into ATOMs. As a case study, an assumed airport model inspired by Sydney's Kingsford-Smith Airport (hereafter referred to as Sydney Airport) is used for the analysis conducted in this paper.

In this thesis, the uncertainties in ATM are captured from three main perspectives: 1) The initial air traffic demand data is generated based on space and time uncertainties (as Section 6.3), the space uncertainty is modeled as Gaussian distribution, while the time uncertainty as Poisson process. 2) The route followed by a flight may differ from the desired or ideal route due to a variety of reasons such as uncertainty in aircraft performance, navigation system error or flight technical error. To capture this type of uncertainty, the probabilistic nature of an aircraft's position is modeled by a Normal distribution (as Section 6.5). 3) The unexpected disruptions to the ground resources on the airport are also considered in this thesis. Event type, activation time and duration are assumed as uniform distributions (as Section 6.7.2). Above uncertainties are major restrictions for system-level capacity in a terminal area. The situation where aircrafts appear suddenly is normally considered in more practical situation. For simplification, it is not included in the problem scope of this thesis.

In particular, we are interested in determining how to achieve dynamic capacity-demand balance in a TeA while considering ground events/uncertainties. We believe that the resultant methodology will be a benchmark methodology for decision-making in future ATM research.

1.2 Motivation

As the critical interface between a relatively 'unconstrained' en route airspace and high density airport complex, TeAs are expected to handle an increasingly higher volume of air traffic in the future (Eurocontrol, 2010). However, the realizable capacity of a current TeA is ultimately limited to its capability to accommodate safe and efficient travel under highly limited airspace design strategies and pre-defined terminal trajectories. Thus, future ATM will require safe and effective TeA airspace configuration strategies to maximize potential TeA capacities, and to facilitate alleviation of the growing capacity/demand imbalance.

A TeA system inherently involves high uncertainties and multiple interactions among many different components. The complexity determines that a TeA system is very sensitive to any changes in traffic, procedures or meteorological factors. Changes to any element, whether static or dynamic, can influence the state of the system which is naturally dynamic.

To handle such a complex TeA system, current ATM manages air traffic in a highly distributed manner, with arrival traffic in TeA airspace, departure traffic in TeA airspace, arrival ground movements and departure ground operations controlled independently by corresponding air traffic controllers with different responsibilities and objectives. From a system perspective, most of these objectives are local (compared with the global objective of the entire system) and could have adverse effects upon each other. As a result, it is difficult to achieve truly high efficiency for air transportation under current ATM operations which targets at local-optima.

In addition, most work on new ATM procedural developments has typically been accomplished without considering the complex interactions among all parts of the operational environment; for example, the studies in (Kuster and Jannach, 2006; Green and Vivona, 2001; Barmore *et al.*, 2004) focus on one specific TeA element through which they are able to improve efficiency in their own functional component. However, they potentially fail to support a systematic approach towards modelling dynamical behavior over time by not taking into account the complex interactions of a highly distributed ATM.

Our motivation for this thesis stems from the fact that the process of understanding, evaluating and dynamically designing TeA airspace configurations in a systematic manner is primarily to maximize a TeA's potential capacity, and to facilitate alleviation of the growing capacity/demand imbalance. A better understanding of the behavior and vulnerability of a TeA system considering the interactions between both arrivals and departures and shared air-ground resources, can pave the way towards achieving the potential benefits of advanced TeA airspace configurations as well as gaining additional ATM system capacity. Making the TeA resources flexible and available to correspond to different traffic scenarios and dynamic constrained ground resources is the key to enhancing the practical ATM efficiency in future TeAs.

1.3 Research Question and Hypothesis

The TeA is a complex and adaptive system in which it is difficult to achieve truly high efficiency for air transportation under current ATM operations which target at local-optima. However, the challenge lies in how to accomplish system-optimization considering the complex interactions among all parts of the operational environment. This thesis argues that a systematic approach of understanding, through modelling, and unraveling the congestion in a TeA is primary to alleviating the growing capacity/demand imbalance in future ATM. These are the focal points of the research reported in this thesis, that is, it aims to address the following question.

How to understand, evaluate and dynamically design TeA airspace configurations in a systematic manner?

Our hypothesis is that a simulation-based co-evolutionary computational environment will be suitable for understanding, evaluating and resolving ATM issues in a highly constrained TeA, especially when considering coupled arrivals and departures with shared ground-air resources. Hence, our main research objective is to prove or disprove the proposal that a co-evolutionary computational environment based on simulation can search massive spaces of possibilities governed by uncertainty and complex networked dynamics quickly, leading to the identification of system vulnerability as well as the evaluation and design of an advanced TeA concept.

To address this question we build a simulation environment which combines air and ground subsystems and provide a proper operational environment for processing arrivals and/or departures. We then develop a co-evolutionary computational environment – CCRT – to evaluate advanced TeA concepts and identify system-level risks in a TeA. Afterwards, a multi-objective co-operative co-evolutionary methodology is proposed as a new optimization search engine of the CCRT framework, in order to solve complex TeA problems with multiple conflicting objectives. A novel TeA airspace design concept for capacity-demand balancing including a measure of collision risks derived from the probabilistic nature of aircraft's performance is proposed. Then, an air traffic simulator, originally representing the novel TeA airspace design concept while considering the interactions among dynamic ground events is presented. Subsequently by applying the multi-objective CCRT, we design and investigate scenario-specific TeA airspace design strategies, which can effectively and safely make TeA resources flexible and available while considering the ground events/uncertainties.

In order to answer the main research question, the following related sub-questions also need to be investigated.

1. **How to evaluate advanced TeA airspace concepts in the integrated TeA system and understand system-level vulnerabilities ?**

   To design new airspace configuration in a TeA, we naturally start from analysis of the newly developed TeA airspace concepts which are expected to enhance ATM efficiency. However, most advanced TeA concepts (such as dynamic CDA (Alam *et al.*, 2010a)) were developed and evaluated in a traditional manner, where complex interactions among other parts of the environment in which they operate are not considered. Hence, evaluating advanced TeA concepts in an integrated TeA, which involves interactions between system components, paves the way to design airspace configuration in a TeA. By evaluating these interactions, which are considered from the perspective of identifying inefficiencies, we will be able to reveal "improvement opportunities" in the implementation of future TeA airspace concepts and, thereby, it may significantly improve understanding and decision-making as well as help achieving system-level objectives.

2. **How to design an efficient multi-objective co-operative co-evolutionary algorithm which can co-evolve solutions towards the efficient set of trade-offs effectively, while maintaining diversity of the solution set?**

   Most real-world ATM issues naturally involve multiple conflicting objectives, such as efficiency versus safety. Multi-objective optimization aims to optimize several components of a vector of objective functions simultaneously. Contrary to single-objective optimization, the multi-objective problem usually does not have a single solution that optimizes all criteria concerned, but a set of solutions (known as Pareto-optimal solutions). None of the solutions in the Pareto-optimal set is absolutely better than any other, thus anyone of them is an acceptable solution. Which solution should be chosen depends on the decision-makers, preferences and various problem-related factors. Hence, a decision-maker is typically interested in knowing as many Pareto-optimal solutions as possible (Deb and Kalyanmoy, 2001). A multi-objective co-operative co-evolutionary algorithm, which can approximate the Pareto-front effectively, will provide solutions representing a fine trade-off between different ATC priorities - efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand.

3. **How to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety?**

   The interactions from ground events and the uncertainties in the air traffic performance are two main factors that affect the decision-making while identifying effective TeA airspace design strategies. Therefore, given a certain set of arrival traffic, both efficiency and safety need to be considered when judging on the quality of a candidate TeA airspace configuration. With more thought given to the trade-off between efficiency and safety, TeA airspace design choices may significantly enhance the potential TeA capacity and facilitate alleviation of the growing capacity/demand imbalance. Meanwhile, by involving the interdependencies between constrained ground resources and air traffic, an airport and its TeA airspace are collaborated as a whole system; and the output will be a fine trade off between different local objectives – ground and air. The cooperation between ground and air manages to support a systematic approach towards achieving system-level objectives while designing TeA airspace configurations.

1.4 Organization of Thesis

This thesis has seven chapters and is organized as follows.
In Chapter 1, an introduction, which provides an overview of the research field, the motivation for, and research questions raised by this study, an outline of the thesis and the scientific contributions stemming from this research, are presented.

In Chapter 2, a background is provided for the research conducted into an integrated TeA system. First, a background to the integrated TeA system is provided and the need for a system-level approach is emphasized. Existing TeA airspace design procedures are surveyed along with some advanced TeA concepts. A survey of the literature on the computational red teaming methodology is reported. Also, a summary of co-operative co-evolutionary algorithms is given. The multi-objective co-operative co-evolutionary algorithms are then presented, with a discussion of several research issues involved in implementing co-operative co-evolutionary algorithms for multi-objective optimization problems.

In Chapter 3, the design and development of an integrated TeA simulation system is explained. Firstly, a description of the simulation environment is given, followed by the input and output to the simulator. The data structure issue of the major components is discussed next. Then, the network representation for modelling the ground-air resources is presented along with the modelling of a queue manager. Both the advanced TeA airspace models derived from the dynamic CDA and the conventional TeA model (fixed STARs and SIDs) are implemented in the simulator. A thorough investigation of the results collected from various measures and metrics, which provide evidences of the improvement of the dynamic CDA model and illustrate problem spaces where the TeA system is vulnerable in terms of efficiency, is given. Then, the design principle and system architecture are provided, followed by the procedures for the queue manager and safety separation. The arrival and departure managers are discussed next and, finally, the arrival-departure integration principle is presented.

In Chapter 4, a CCRT methodology for identifying the TeA system vulnerabilities and evaluating advanced TeA airspace concepts is designed and presented. A cooperative co-evolutionary search mechanism with a single objective is developed to search problem spaces of possibilities governed by uncertainty and complex networked dynamics, induced by the air-ground integration and arrival-departure combination. Air traffic scenarios and constrained ground event scenarios are automatically generated and co-evolved with each other under the co-evolutionary pressure guided by the designed fitness function. The proposed methodology is then validated for a variety of experimental parameters and scenarios. Finally, the results are analysed and discussed based on various measures and metrics such as computational efficiency, efficiency in the dynamic CDA model, major air traffic flow constraints in the TeA and scenario patterns with higher delays.

In Chapter 5, a multi-objective co-operative co-evolutionary algorithm (MOCCA) is proposed and designed in order to tackle the TeA issues with multiple conflicting objectives. Firstly, among existing co-operative co-evolutionary algorithms for multi-objective optimization, the most competitive and robust one – CCEA which is proposed by (Tan *et al.*, 2006), is introduced as the basis and reference point, followed by a discussion of three issues (fitness assignment mechanism, niching strategy and archiving updating scheme) identified as potential weaknesses of CCEA. A summary of existing fitness assignment mechanisms and niching strategies available in the multi-objective optimization research is then given. A variety of fitness assignment strategies and niching strategies are proposed for potential use in the proposed MOCCA. Two rounds of experiments are then carried out for evaluating the performance of different algorithms – all candidate algorithms are examined on a variety of benchmark test cases with different performance metrics. In the end, the results distill the most competitive and robust algorithm, and the proposed MOCCA is then presented.

In Chapter 6, we present and develop a TeA system for airspace design. The system environment is firstly described, followed by its input and output. A scenario generation methodology for air traffic with different temporal and spatial distributions is explained. The data structure of the major components is discussed next. Then the objective function design is presented, in which for the safety objective, we break away from traditional methods of defining collision risk and introduce a novel collision risk concept derived from the probabilistic nature of aircraft performance. The architecture of the system is discussed next and, finally, the arrival manager is presented.

Afterwards, the MOCCA proposed in Chapter 5 is employed as the new search engine in the CCRT framework. The multi-objective CCRT is then applied to investigate prior TeA airspace configuration strategies with less flight delays and lower collision risks; which corresponds to dynamic constrained ground resources given different traffic scenarios. For a specific traffic scenario, TeA airspace design scenarios and constrained ground event scenarios are automatically generated and co-evolved with each other using collaborative guidance from two objective functions. Then, the proposed algorithm is validated for a variety of experimental parameters and scenarios. Finally, the results demonstrate a resealable trade-off between two ATC priorities: efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand.

In Chapter 7, the main findings from this thesis are summarized. The chapter concludes the thesis with a discussion of possible future research directions.

1.5 Original Contributions

A list of the scientific contributions arising from this thesis is given below.

- A simulation-based co-evolutionary computational environment (CCRT) is developed for evaluating advanced TeA airspace concepts and understanding the TeA system vulnerabilities (Chapter 4). Compared with previous approaches for assessing ATM concepts, which focused on using an isolated evaluation for each sub-system, a co-evolutionary computational environment that integrates computational intelligence techniques and multi-agent systems offers the means to search massive problem spaces of possibilities governed by uncertainty and complex networked dynamics quickly, and find the spaces representing system vulnerabilities. An air traffic simulation system with a novel representation of an integrated TeA considering air-ground collaboration and arrival-departure cooperation is presented for a system-level understanding of TeA concepts (Chapter 3). The dynamic CDA model is evaluated against the fixed STARs and SIDs model as comparison. A thorough investigation into results collected from various measures and metrics is given, providing evidence of the superior performance of the dynamic CDA model and illustrating the problem spaces in which the TeA system is vulnerable in terms of efficiency. By co-evolving the air-side and ground-side subcomponents, evaluating and understanding advanced TeA airspace concepts in an integrated manner may significantly improve understanding and decision-making as well as help achieve system-level objectives.

- A multi-objective co-operative co-evolutionary algorithm (MOCCA), which can co-evolve solutions towards the true global Pareto-front effectively and maintaine a high diversity of the solution set, is proposed and designed for solving the TeA issues with multiple conflicting objectives (Chapter 5). Although the existing cooperative co-evolutionary algorithm (CCEA) for multi-objective optimization proposed by (Tan *et al.*, 2006) is the most competitive and robust algorithm so far, three issues are identified as weaknesses of CCEA. The MOCCA presented in this work involves an improved fitness assignment strategy which effectively avoids the situation that individuals dominated by same archive members have identical fitness values; an enhanced niching strategy which requires no user-defined parameter and prevents boundary solutions being removed during archive truncation process; and a new globalized scheme for archive updating so that the solutions which resides in less populated regions of the global Pareto front are saved. After being validated on various benchmark test cases on different performance metrics, results demonstrates that the proposed approach is capable of evolving solutions towards the true global Pareto-front more effectively while maintaining a higher diversity of the solution set, comparing to CCEA.

- A simulation-based co-evolutionary computational environment for multiple objectives is proposed, in order to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety (Chapter 6). TeA airspace design, specially when considering its inter-dependency with dynamic ground events, has always been a challenging area given the inherent system uncertainty and complexity. An air traffic simulator representing an original novel TeA airspace design concept for capacity-demand balancing including a measure of collision risks derived from the probabilistic nature of aircraft's performance is proposed. With more thought given to the balance between safety and efficiency, the TeA airspace design choices in an air-ground integration level may significantly enhance the potential TeA capacity and facilitate alleviation of the growing capacity/demand imbalance. The advantage of this methodology is that it co-evolves air- and ground-side subcomponents and, investigates superior TeA airspace configuration strategies considering both efficiency and safety in an integrated manner. The multi-objective approach also provides an analyst with the trade-off between these two ATC priorities - efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand. This highly accessible method of capturing a TeA scenario's complexity with multiple objectives may have significant implications not only within the scope of the TeA domain but also across the much wider spectrum of the ATM research field.

Chapter 2

Background

2.1 Capacity-Demand Balance

In the ATM domain, the capacity of an airspace or airport normally represents its ability to safely handle a number of aircrafts per unit of time. Capacity depends on many factors, such as the configuration of an airspace, layout of airport ground infrastructure, ATM operations and procedures, capability and availability of air traffic control, and capacity and availability of element resources in the airspace or airport. Some of these factors are inherently dynamic (e.g. disturbance to the availability of an element resource in the airport due to adverse weather), and any change in such factors can influence the capacity. Air traffic demand is normally measured by the number of flights per unit of time serving an airspace or airport. It is usually represented by the space and time information about the aircraft fleet mix. Any variation in these factors can affect air traffic demand. When the amount of aircrafts allocated to an airspace or airport mismatches the number of aircraft it can safely handle, capacity and demand imbalance occurs.

In current operational concepts, as implemented in the United States and Europe, air traffic flow management (ATFM) plays a central role in maintaining capacity-demand balancing by adjusting traffic flows according to the "declared capacity" of an airport or air traffic control sector. Current ATFM in the United

States has focused mainly on congestion at major airports or in the terminal airspace around them, whereas in Europe, ATFM has to deal with congestion in the en route airspace as well, since European ATM system is composed of many different national ATM systems and provides lower flexibility in handling en route traffic (Lulli and Odoni, 2007).

Each airport and air traffic control sector declares its maximum capacity. When the declared capacity is exceeded, ATFM approaches are taken to reduce the traffic demand. There are two popular ATFM strategies: holding patterns and ground delay programs. When an air traffic control unit reaches its capacity, arriving aircrafts are directed towards holding patterns where they circle until their time to land. Since this way of delaying a flight is inefficient and costly, delayed flights prefer to stay with engines off on the ground of their departing airports, saving considerable amounts of fuel. This is called a ground delay program. Although absorbing most delays on the ground instead of in the air is economically efficient, it depends on precise calculations of flight time and traffic flow as a whole, which requires sophisticate methodologies and techniques.

Odoni (1987) is the first to formulate the ground delay problem for a single airport in mathematical terms as a deterministic process. Richetta and Odoni (1993); Ball *et al.* (2003) addressed this problem as a probabilistic process. Vranas *et al.* (1994) studied the ground delay problem for a multi-airport scenario. Instead of being focused on airports, some techniques were developed to deal with the whole ATFM problem, including capacity restrictions of en route airspace. More ATFM strategies were involved and formulated except for ground delay, such as airborne holding (holding patterns), speed control and rerouting. Such ATFM problems have been thoroughly studied in (Bertsimas and patterson, 1998; Bertsimas and Patterson, 2000; Sridhar *et al.*, 2002; Lulli and Odoni, 2007; Myers and Kierstead, 2008; Bertsimas *et al.*, 2008; Rathinam *et al.*, 2009; Bertsimas *et al.*, 2011).

It is well known that the air transport demand has been experienced an enormous growth (U.S. Department of Transporation, 2011; EUROCONTROL, 2012). Although the continuous growth of air traffic demand is a major contributor to economic expansion, it has placed an enormous strain on the capacity of the air traffic system (Bertsimas *et al.*, 2011). In 2011, approximately 21% of all flights in the United States were delayed on arrival, 19% on departure, and 3% were cancelled (U.S. Department of Transporation, 2011). It has been estimated by Air Transport Association (2011) that delays raised $7.7 billion in direct aircraft operating costs for U.S. airlines in 2011. In addition, delayed flights also cost air travelers billions of dollars in the form of lost productivity, wages and goodwill (Air Transport Association, 2011). By 2010, the average ATFM delays in Europe has reached 75620 minutes per day, and the ATFM delay per flight is 2.9 minutes on average for all flights (EUROCONTROL, 2012).

This trend shows that current ATM systems in the United States and Europe are already stretched to the limit and starting to get over-passed. Towards addressing capacity-demand imbalances, a large research effort is now underway to improve ATM system's efficiency while ensuring safety.

In U.S., the Next Generation Air Transportation System (NextGen) Concept of Operations (ConOps) was developed by the Joint Planning and Development Office (JPDO) to modernize the U.S. air transportation system (Joint Planning and Development Office, 2007). This future ATM system will be a trajectory based, performance specified air traffic operating environment, which involves a robust, automated and integrated digital system (Joint Planning and Development Office, 2011). The overall philosophy driving the delivery of the NextGen is to adjust ATM system capability to satisfy forecast demand, rather than constraining demand to match available capacity (Joint Planning and Development Office, 2011). A number of ATM changes and capabilities are needed to be performed, such as collaborative capacity, collaborative flow contingency, trajectory, and separation management (Joint Planning and Development Office, 2011). These capabilities describe at a high level vision for managing the anticipated growth in air traffic demand by maximizing the use of available ATM resources (Joint Planning and Development Office, 2011).

In NextGen (Joint Planning and Development Office, 2011), ATM resources are managed to maximize utility to flight operators, in order to address short-term capacity-demand imbalances. For instance, the ATM system could adjust airspace structure and boundaries, apply known procedures, or reallocate personnel to meet forecasted demand. For expected long-term capacity-demand imbalances, the ATM system may implement major changes to airspace design, significant airport infrastructure improvements, or develop new flight operational procedures and automation systems. If the ATM system can not meet the resulting demand with its maximum capacity, ATFM strategies are imposed to ensure that safe levels of traffic demand are not exceeded when capacity limits are reached (Joint Planning and Development Office, 2011).

As NextGen proceeds to be deployed, NASA (National Aeronautics and Space Administration) is conducting a new operational paradigm for increasing airspace flexibility and capacity – Dynamic Airspace Configuration (DAC) (Kopardekar *et al.*, 2007, 2008). It aims at migrating away from the current structured, static airspace to a future airspace which is flexible, dynamic and capable of adapting to traffic demand while meeting constraints of weather and equipage (Kopardekar *et al.*, 2007). There are three major components contained in DAC concept: 1) restructuring the airspace to take advantage of advanced technologies, 2) adaptable airspace based on fluctuating traffic demand, and 3) generic airspace promoting interchangeability of controller resources within and across facilities.(Kopardekar *et al.*, 2008)

In Europe, the SESAR (Single European Sky ATM Research) programme is transforming the current European ATM into a more efficient and safer system (SESAR Consortium, 2007). Like NextGen, this future ATM system will also be a trajectory based, performance specified air traffic operating environment involving advanced information sharing environment, automated tools, and improved surveillance capabilities. The combination of the improved information serves with increased automation will enable the ATM system to be managed in a collaborative manner, in which the decision making among stakeholders will be shared and the potential impacts of decisions will be better assessed. The collaboration in ATM decisions is critical for increasing ATM capacity in order to address the expected capacity-demand imbalance.

Dynamic Demand Capacity Balancing (dDCB) is a SESAR project, which is proposed to improve ATM safety and capacity through reducing traffic complexity and streamlining air traffic controller workload (EUROCONTROL, 2011). Motivated by avoiding systematically regulations when demand does not significantly exceed available capacity, the dDCB concept targets at exploiting buffer of capacity while managing the ATC workload and traffic complexity. With the help of accurate prediction of air traffic demand, the dDCB helps to transform capacity management to minute-based streamlined actions at sector level, from the current global hour-based traffic limitations. Given a detected congestion, a set of short-term ATFM measures (such as short ground delays, flight level caps or minor re-routing) are implemented to reduce the demand to a safe level.

All the reviewed works were drove by the same overall philosophy to solve the imbalance between capacity and demand – adjusting one of them by assuming the other one is fixed. For instance, given a predicted capacity shortfall, ATM decision makers decrease the demand to match the capacity by reducing/regulating traffic flows; whereas given an anticipated level of demand, they increase the capacity to meet the demand by using the current resource more efficient, designing new airspace, or building new runways and/or airport. Without assuming either capacity or demand is fixed or can be precisely forecasted, the dynamic capacity-demand balance problem in this thesis is studied by involving uncertainties in both of them, in order to improve the efficiency and safety in an ATM system.

2.2 TeA System

In the air traffic management (ATM) domain, a typical flight trip is partitioned into a set of phases: pushing back from the gate, taxiing onto the runway, taking off, following a departure route, cruising along an airway, following an approach route, landing, taxiing off the runway and stopping at the gate (as demonstrated in Figure 2.1). Each flight goes through these phases as it complete an air traffic task.

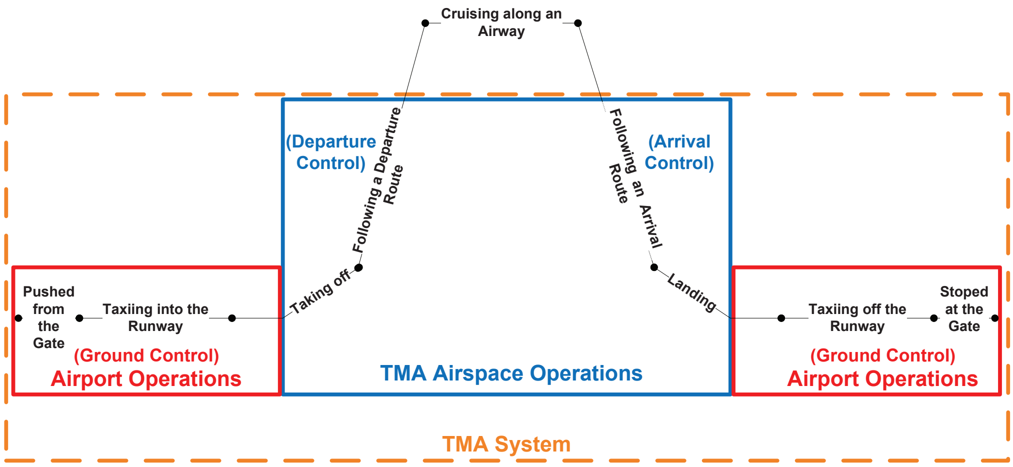


Figure 2.1: Phases of a Typical Flight Trip

ATM operations in each individual phase impact the ATM efficiencies in both its preceding and succeeding phases. Currently ATM is operated in a highly distributed manner where different air traffic controllers are responsible for air traffic in different phases; for instance, approach controllers in Terminal Control Areas are responsible for inbound approaching traffic, departure controllers for outbound departing traffic, and ground controllers in the air traffic control tower for ground movements in the airport.

Except for the cruise phase, all other flight phases operate in an airport and the TeA airspace in its vicinity (normally 20-30 nm around the airport). The TeA airspace, which is also referred to as the terminal airspace (TA), is defined as the en route airspace that transitions inbound and outbound flights to and from an airport (Jani and Toi, 1982). It forms the critical interface between the relatively 'unconstrained' en route airspace and the high-density airport complex. A variety of ATC procedural constraints (routes, altitudes and speeds) are needed to facilitate the safe and orderly monitoring of arrivals and departures. In addition, ATM constraints related to airport capacity limitations (e.g., miles-in-trail or sequencing) have a significant impact on air traffic efficiency in this environment (Nolan, 2004).

An airport is an essential element of an air transportation system as it is "either an intermediate or terminal point of an aircraft on the air portion of a trip (Ashford *et al.*, 1997)". Traditionally, its operations are divided into two parts: air-side and land-side functions. The former consist of all the 'traffic' operations on the airport's infrastructure (runways, taxiways and gates) while the latter include the handling of passengers, processing of payload and crew, and servicing and maintaining aircraft (Ashford *et al.*, 1997). In this thesis, from ATM perspective, airport operations refer to air-side operations. In any busy airport, even one with a fairly simple airport topology, managing ground operations and dealing with all the incoming and outgoing airport ground traffic can be a very challenging task since an airport represents a highly dynamic and volatile environment with continuously changing resources' availabilities (Kuster and Jannach, 2006). Tower controllers are required to provide ground operations for both arrivals and departures to facilitate safe and smooth ground traffic flows within uncertain constrained ground events.

In this work, the TeA system is defined as an environment combining an airport's ground transportation network (runways, taxiways and gates) and its TeA airspace, as indicated by the dashed line in Figure 2.1. It is a highly tactical environment involving a variety of ATM operations for accomplishing capacity-demand balance and ATC restrictions which facilitate the safe and orderly handling of arriving and departing aircraft. The complex interactions between arrivals and departures with constrained air and ground resources using different ATM operations have a significant impact on flight efficiency in this environment.

2.2.1 TeA System Complexity

A TeA system is a highly complicated environment involving different static elements (such as current fixed airspace geometry and airport infrastructure) and dynamic elements (such as air traffic and ground events). This complexity determines that a TeA system is very sensitive to any changes in traffic, procedures or meteorological factors which, whether static or dynamic, can influence the state of the system which is naturally dynamic. Recognizing the nature and state of each element is a primary step towards attempting to understand the TeA system complexity. Figure 2.2 captures a number of components which influence the state of a

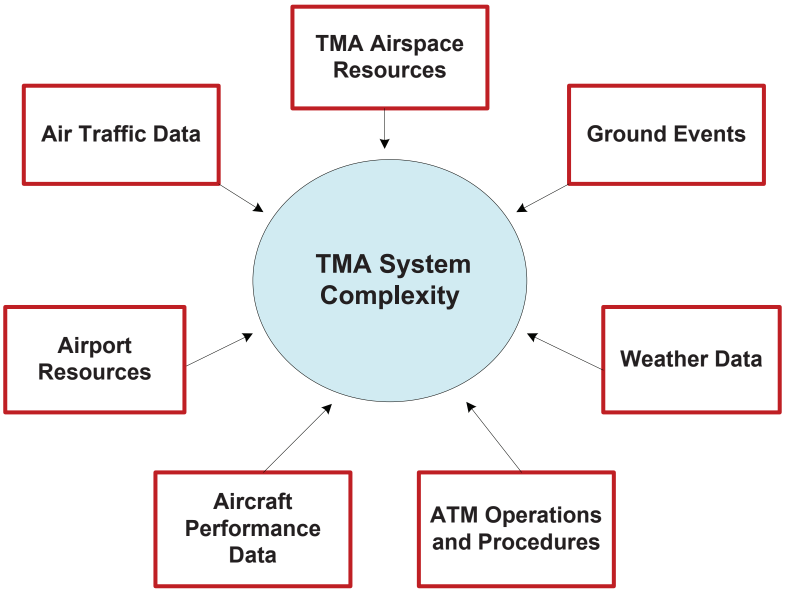


Figure 2.2: Components Influencing TeA system's State and Complexity

TeA system and contribute to its complexity.

Air traffic in a TeA system usually has the following characteristics (Tosic and Netjasov, 2003):

- arrival traffic converges from, and departure traffic diverges to, different directions;
- as traffic density during operational times is not constant, it requires the flexible use of TeA resources; and
- as different aircraft of different sizes and velocities are mixed in a fleet, they require appropriate separation rules.

Air traffic demand data refers to the space-time information about the aircraft fleet mix servicing an airport within its TeA system. Since it embraces both a spatial- and time-based nature, it is normally represented by the statistical characteristics of air traffic's spatial and temporal distributions (Netjasov *et al.*, 2011). Regarding time, a snapshot analysis of a number of traffic movements through a TeA system over a certain time period is typically a feasible approach for understanding traffic density.

From a space perspective, it is necessary to determine the spatial distribution of the air traffic by identifying the traffic flows at the entry points to a TeA system.

Information about a TeA airspace includes its way points, the numbers and lengths of arrival and departure trajectories, its special use airspace (SUA), and the availability and capacity of its resources. Its ground configuration refers to its airport air-side infrastructure (runways, taxiways and gates), and the capacity and availability of each ground resource. ATM operations and procedures comprise arrival, departure and surface movement operations, safe separation rules, and aircraft sequencing and routing procedures.

A constrained ground event is a disruption to the availability of an airport's surface resources, including all its runways, taxiways and gates; for example, snow, ice, slush or water on a runway can reduce aircraft braking and directional control and increase runway occupancy time. An increase in runway occupancy time leads to reductions in airport arrival and departure acceptance rates (throughput) due to the need for increased inter-aircraft spacing, which are aggravated by the closure of runways and certain runways being unusable. Also, disruptive airport ground movements can propagate elsewhere in a TeA system (e.g., air traffic in the transition airspace) and cause flight delays and system inefficiency. The inherent uncertainty of ground events and their interactions with other TeA system components heavily influence an airport's capacity and raise the complexity of its TeA system.

A TeA is a complex and adaptive system, involving high uncertainties and multiple interactions among many different components which needs to respond quickly to any changes. This high degree of variability and adaptation exists not only between its various components but within most of their elements. This interweaving-component organization is summarized in Figure 2.3.

To handle such a complex TeA system, current ATM manages the air traffic in a highly distributed manner, with arrival/departure traffic in the TeA airspace, and arrival/departure ground operations controlled independently by their respective air traffic controllers who have different responsibilities and objectives. From a system

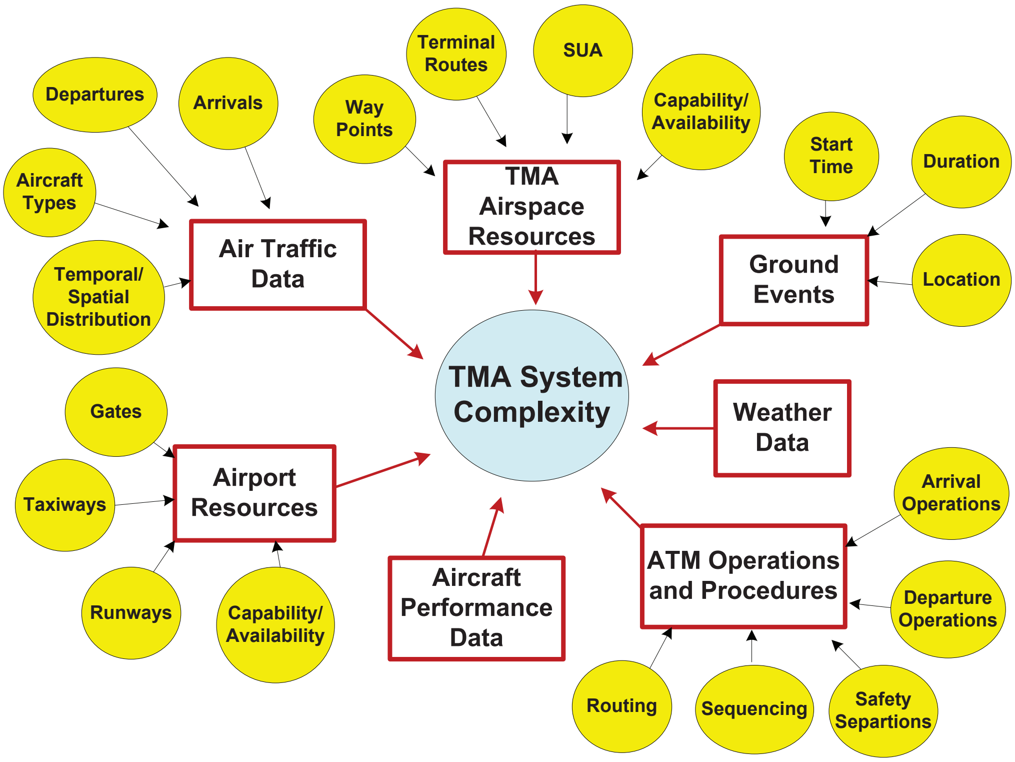


Figure 2.3: TeA system's Complexity

perspective, most of these objectives are local (compared with a global objective of the entire system) and could adversely affect each other. As a result, it is difficult to achieve truly high efficiency for air transportation under current ATM operations which targets local-optima.

2.2.2 Air-ground Collaboration

This section introduces historical research which has made an effort to achieve collaboration between air- and ground-subsystems in a TeA.

Korn *et al.* (2006) introduced a concept for air-ground co-operation which connects the on-ground arrival manager to the on-board flight management system, to optimize approach trajectories by considering all present arrival aircraft and ground constraints. This concept enables reductions in both flight times in the TeA and controller workload providing a highly reliable data link is available. The discussed technology level of the negotiations between ATC and arrival aircraft is primarily

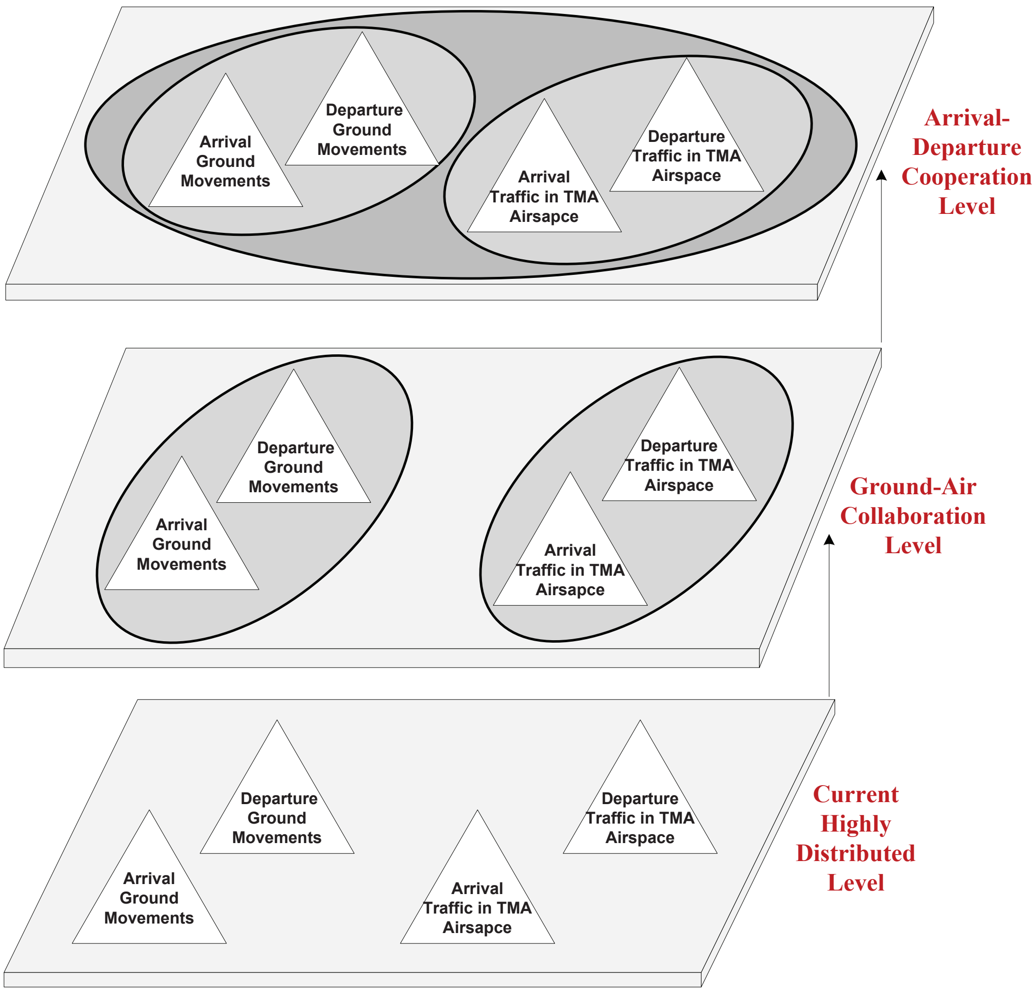


Figure 2.4: Different Integration Levels in a TeA system

for the trajectory based traffic management.

Prevot *et al.* (2007) conducted an air/ground simulation based on a site-specific implementation of the Trajectory-Oriented Operations with Limited Delegation (TOOWiLD) concept. Their purpose is to assess the alignment between near-term evolutionary progress and far-term potential transformation of an air traffic system. The former's major initiatives include airborne spacing, trajectory-oriented arrival management, CDAs, decision support tools for ATC and controller-pilot data link communication. However, although air/ground co-operation lies in the data connections between flight crews and controllers, runway scheduling is the only ground-side operation they consider.

Haraldsdottir *et al.* (2007) took a near-term step towards transitioning to trajectory-based operations in future ATM by proposing an operational concept of arrival management which increases the capacity of a TeA by integrating advanced navigation performance capabilities and ground-based decision support tools. However, although this integration lies in the connection between its airborne and ground-side automation tools, runway assignments and scheduling at runway thresholds are the only ground operations considered.

Oberheid and Söffker (2008) presented a colored petri net model which simulates a future arrival planning process which includes co-operation between the air and ground sides. It formally models co-operative arrival management processes considering the combinatorial aspects of the sequence planning problem. This model is designed to understand how the behaviors of individual aircraft within this co-operative arrangement affect the planned arrival sequence. However, it places a strong focus on co-operation between the information supplied by airborne arrival aircraft and the ground-side ATC, but does not consider ground traffic situations or trajectory generations.

The US NextGen (Next Generation Air Transportation System) presented an operational concept referred to as Super-Density Operations (SDO), which envisions the combination of advanced ground and flight deck automation, optimized 3D and/or 4D trajectory profiles and appropriate separation management to achieve higher performances in terms of robust terminal airspace and airport utilization. This leads to the increased safety, efficiency and reliability of future ATMs, as specified in the NextGen goals (Isaacson *et al.*, 2010). However, the concern for air-ground collaboration in SDO concept is limited to the air-ground information exchanging. The operational domain on which SDO focuses is the TeA airspace while, airport surface operations and configuration management are not considered.

Swenson *et al.* (2011) developed a terminal area precision scheduling and spacing system (TAPSS), which is a strategic and tactical planning tool. It integrates a set of trajectory-based automation tools, including 4-D trajectory prediction, arrival runway balancing, aircraft scheduling with constrained separation, traffic flow visualization and trajectory-based advisories. It is designed to provide air traffic controllers with the capability to efficiently meter, sequence and space arrival traffic. Nevertheless, except for runway constraints and allocations, changes in the ground traffic situation and airport configuration in other part of the airport are not considered.

Uebbing-Rumke and Temme (2011) developed a ground-based decision support system for mixed approaching traffic combining a negotiated CDA and a conventional approaching traffic method which is able to assist controllers in calculating suitable landing sequences and times in mixed traffic scenarios. Its working principle is to take into account results from its own trajectory prediction engine, which calculates possible profiles for conventional approaches, and the information about CDA flights, the landing times of which are negotiated in an air-ground protocol. Although this air-ground negotiation for a CDA is based on the implementation of advanced air-ground communication concepts, no actual ground resources are considered.

Most reviewed technologies are driven by considering how to match currently under-utilized airborne and ground-side automation systems in order to minimize investment risk and accomplish a smooth ATM transition from current operations to an envisioned system. However, the air and ground subsystems in those work are not considered from a system-level perspective. For instance in (Korn *et al.*, 2006), approach trajectories are optimized with considering present arrival aircraft and ground constraints, yet ground operations for the optimized arrival traffic are not considered. In another words, the existing research on air-ground co-operation is managing air traffic (in air or on ground) by considering constraints from other parts of the environment, but still with the managed section's own isolated objective.

Air-ground integration in our work is carried out from a system-level perspective. It manages air traffic operations in TeA by involving the interactions between the air traffic and constrained ground resources of an entire air-side airport configuration, including – runways, taxiways and gates. All resources in its TeA airspace and airport are included in an air-ground network which represents a complete model of the TeA system, thereby providing the possibility of analysing system-level risks. This level of co-operation treats the TeA as a whole system, and the output will be a fine trade off between these local objectives: approach management and ground control; or ground control and departure management.

2.2.3 Arrival-departure Co-operation

This section introduces historical research which has attempted to process arrival and departure traffic in a TeA system in a co-operative manner.

Gilbo (1993) studied the estimation and optimization of an airport's capacity. His approach is not to consider each flight separately but, rather, to look at the total number of arrivals and departures over periodic intervals and use optimization to dynamically determine the capacities to be allocated to them over a time horizon. Gilbo (1997) extended these results to a case in which an airport's capacity is stochastic.

Anderson *et al.* (2000) developed an integrated ground-operations model which capture the dynamics of ground operations at congested hub airports. It consists of an arrival model for the taxi-in process, a ground model for the taxi-out process and a departure model for the turnaround process. This work also proposes an application of the integrated model to improve predictions of aircraft movement times on the ground. The purpose of such predictive improvement is to identify anticipated congestion periods based on the captured dynamics of the airport and, thereby, improve overall airport efficiency. This integrated model may also be implemented to evaluate decision support tools for alternative control strategies.

Mayer and Swancy (2005) compared and evaluated potential benefits of candidate procedures for arrivals and departures which have intersecting runway operations. All these operational procedures consider correlations between arrival and departure traffic on the ground with respect to the simultaneous occupancy of the runway system. An airport model is developed to quantify and visualize ground operations for arrivals and departures. However, the major concern of this work is to maximize the use of intersecting runways and the capacities of other ground resources are not considered.

Bohme *et al.* (2007) presented a concept for coordinating arrival traffic in a TeA with the departure situation on the ground which is based on fuzzy rules expressing expertise considering, in particular, operational and implementation issues. It is generic for any arrival and departure managers which meet a set of well-defined requirements. However, this concept is designed for evolutionary rather than revolutionary changes in today's ATM, which is currently organized in a de-centralized manner, and the "priority of arrivals" is maintained. Therefore, coordination is focused more on inbound traffic than on arrivals and departure sharing TeA resources.

Delgado *et al.* (2009) imagined the future civil application of an Unmanned Aerial System in an integrated TeA airspace covering arrival and departure operations. This application is assessed and its problems analyzed and potential solutions proposed. However, as its focus is on the concept of actual operations based on predesigned trajectories, no future improvements in the integration between arrival and departure operations is considered.

In (Rehwald *et al.*, 2010), a prototype called the Co-operative Local Resource Planner (CLOU) is viewed from flow management perspective. It is a pre-tactical local planning system which considers the interaction between in- and outbound traffic at an airport. With the help of a holistic view of traffic processing at the airport, CLOU provides suggestions for a runway-use strategy and assists in negotiations between the tower and approach controllers regarding the prioritization of arrival or departure traffic. Obviously in CLOU, the runway is the main resource shared by arrival and departure traffic.

Although many efforts have been made to process arrival and departure traffic in a more co-operated manner than current state, none of these research activities consider the arrivals and departures as a whole system, and manage the air traffic with system-level objectives such as TeA system efficiency. In this thesis, the arrival and departure will be managed by competing for the same TeA system resources, and the output will be a fine balance of conflicting metrics in a highly efficient TeA system.

2.3 TeA Airspace Design

A TeA airspace represents the transitional airspace between the en route airspace and the airport. Today's TeA airspace has a fixed structure, such as way points, navigational aids, pre-defined arrival and departure trajectories and SUA. The entry points to, and exit points from, this airspace are defined by radio-navigational aids (Netjasov *et al.*, 2011), with inbound traffic following the arrival trajectories converging towards the airport and outbound traffic flying along the departure trajectories diverging from it.

In current ATM, the following three categories of trajectories operate between the high-altitude air routes and the airport surface.

- Terminal Radar vectoring routes: are a series of vectors consisting of course, points and altitudes over specific way points, which are executing instructions issued by a corresponding air traffic controllers.
- Standard Terminal Arrival Routes (STARs) and Standard Instrument Depar-

ture routes (SIDs): are specified terminal routes defined by courses, distances and altitude constraints between ground-based navigation aids.

- Terminal Area Navigation (RNAV) or Required Navigation Performance (RNP) routes: are 2D (latitude and longitude), 3D (latitude, longitude and altitude) or 4D (latitude, longitude, altitude and speed) trajectories defined by RNAV (which is a navigation method enabling aircrafts flying point-to-point routes based on a pre-programmed profile, without requiring a track directly to or from any specific ground-based navigation aid (Nakamura, 2000; Becher and Formosa, 2000)) or RNP (which is a RNAV operation permitting more complex routes with on-board navigation performance monitoring and alerting (Nakamura, 2000)) terminal procedures.

Current ATC operations (Nolan, 2004) in a TeA airspace are based mainly on controllers issuing vectoring instructions to facilitate the proper separation, merging or spacing of aircrafts, especially in the terminal airspace without operational use of published routes. These vectoring operations are able to provide controllers with high levels of controllability and flexibility. However, this method can cause high workloads for controllers due to numerous control interventions as well as pilots inducing frequent controller-pilot communications. If an aircraft could follow a predetermined route, the need for many specific instructions from the controllers could be reduced, thus leading to reductions in their workload.

STARs and SIDs (Nolan, 2004) are commonly used ATM operations published as typical ATC-coded procedures. They are fixed and defined standard terminal routes usually involving a set of way points, each of which is designated with altitude and speed restrictions that accommodate a wide variety of aircraft types across a range of expected weather conditions. After an aircraft is cleared into a TeA airspace, the terminal controller files it as having a standard route (STAR or SID) along which it is guided so as to transition towards the next portion of its flight plan, following exactly the turn radius and ground path of the coded procedures.

Since RNAV and RNP terminal procedures (Becher and Formosa, 2000; MITRE

CAASD, 2007) allow navigation independent of the physical location of ground-based navigation aids, RNAV routes can be designed in a more flexible manner. The proposed RNAV and RNP routes are normally defined based on the overlays of current STAR or SID routes and existing flight vectoring paths (statistical) derived from historical aircraft tracking data, and subsequently developed and published for public implementation (Becher and Formosa, 2000). Although some automation techniques have been developed to facilitate the definition of a route structure, such as Terminal Area Route Generation, Evaluation, and Traffic Simulation Tool (TARGETS) (Becher and Formosa, 2000; MITRE CAASD, 2007), the current process of developing RNAV and RNP routes is mostly manual. Consequently, it can be time-consuming, expensive and is usually limited by the quality, quantity and availability of the collected data, and bounded by the experience and knowledge of the participant planners.

Using current procedures, ATM in a transition airspace is a challenging task for terminal controllers, especially under high workload conditions, for two reasons. Firstly, as the amount of air traffic increases, terminal controllers are required to issue headings, speeds and altitudes to guide more flights to track standard terminal routes within a variety of ATM operational rules and restrictions. Secondly, a controller's ability to efficiently manage such situations is limited by the tactical nature of current techniques for merging flights to standard routes and the lack of supporting automation. In Central Flow Management Unit (2009), it is shown that the proportion of delays related to the terminal area and airport has increased significantly and now accounts for 40% of total delays.

Researchers have been evaluating ways of alleviating the consequences of congestion by introducing many automated supports for ATC, such as ASAS 2 in EUROCONTROL/FAA (2001), a spacing algorithm developed for integration in a flight deck tool in Abbott (2002) and AMSTAR in Barmore *et al.* (2004). Although current procedures may be supported by automation and yield some benefits, a fundamental change is needed to significantly increase controller productivity and the accommodation of user preferences (flexibility) (Green and Vivona, 2001). Continued growth in traffic congestion will require air traffic to be managed more dynamically than it is today with fewer ATC restrictions.

On the other hand, according to current procedures, arriving aircrafts fly powered constant-altitude (level) segments through typical ‘dive and drive’ procedures after their initial descent from an en route cruise altitude (RobinsonIII and Kamgarpour, 2010). This step-descent approach with level segments is generally inefficient due to the increased fuel burn, noise pollution and greenhouse gas emissions. Efforts to develop methods for increasing vertical profile efficiency are also considered as one of the building blocks for SESAR and NextGen.

The CDA concept (EUROCONTROL, 2007) is one of such methods to boost the aircraft’s vertical file efficiency. In CDAs, the aircraft descends without level altitude segments from cruise to touchdown with engines at or near idle. The goal of a CDA is to keep the aircraft at a high altitude for as long as possible and its thrust as low as possible. Operationally, two types of CDA have been implemented: Tailored Arrivals (TAs) (Coppenbarger *et al.*, 2007) and Optimized Profile Descents (OPDs) (IMG, 2010).

TAs can enable continuous descents under traffic or noise constrained airspace conditions by integrating advanced air and ground automation through data links. This is a procedure where trajectories are dynamically optimized for each aircraft to permit a fuel-efficient, low-noise descent that will provide separation assistance and meet arrival sequencing requirements and other airspace constraints. Although, compared with a traditional arrival path, executing a CDA realizes a more efficient fuel burn during the transition phase of a flight, its trajectory generation still relies largely on the pre-defined terminal route structure. As a result, the application of these CDA initiatives requires either periods of low traffic density or specialized protocols that limit throughput.

OPDs are arrival procedures designed with altitude restrictions which allow aircraft to descend continuously until interrupted by air traffic controllers. Although an improvement on the vertical profile, OPDs are still statically defined and clearly published procedures developed based on STARs.

The point merge (Boursier *et al.*, 2007; Ivanescu *et al.*, 2009) is a centralized mean for merging and spacing of arrival flows in a terminal area to facilitate more efficient arrival sequences. Its principle is to integrate arrival flows into one sequence with the desired separation by each aircraft flying on one of the vertically spaced sequencing legs. This procedure is based on a pre-defined route structure consisting of a merge point and a set of sequencing legs at iso-distance from the merge point. As this type of route structure determines that the merging of arrival flows relies on route modifications, it does not provide enough flexibility in heavy traffic situations.

The dynamic CDA concept (Alam *et al.*, 2010a) is a methodology which can generate aircraft-specific CDA routes that are both laterally and vertically optimized in terms of given objectives (noise, emission and fuel) in real time. The use of a real-time aircraft position and performance envelope leads to inherently safe CDA routes. Any two arrivals in the same approach airspace can follow two different trajectories, that is, there is no standard terminal arrival route. In practice, each aircraft receives its dynamic CDA by a data link before starting its descent.

However, this approach has its challenges (Alam *et al.*, 2011) because, on the operational side, it may lead to reduced controller-pilot communications for level-off segments, but may increase them in terms of speed constraints. To implement the dynamic CDA, further co-ordination between the terminal and tower control, especially in a traffic-constrained environment, will be a challenging task. Also, there may be a lack of flexibility on the ATC side since clearance for an approach has to be given well prior to an initial arrival fix (IAF). However, with increased onboard computing power, advances in digital data transmission and a proposed real-time data link between controllers and pilots, up-linking and down-linking of trajectories is possible. This makes the realization of real-time CDA route generation a near possibility. The development of dynamic CDA procedures is generally considered to be a key step in the modernization of air traffic operations.

All the tactical techniques currently available to improve a TeA airspace's efficiency are based on its limited underlying graph structure and a set of clearly pre-defined and predictable procedures (Krozel *et al.*, 2004). As a result, air traffic controllers have highly limited flexibility to reconfigure a TeA airspace, thereby restricting their productivity in terms of airspace configuration strategies. The problem in this work is motivated by the desire to optimize the airspace configuration strategies in a TeA system.

The concept of a dynamic TA design strives to remove current rigid structure of navigation aids, pre-defined trajectories and SUA to provide traffic coordinators with more flexibility to reconfigure it, in order to address the growing capacity/demand imbalance and meet fluctuations in user demand. A major component of this concept is to optimize the TeA airspace design strategies to maximize TeA efficiency and safety, given the interdependencies from dynamically constrained ground resources.

2.4 Computational Red Teaming

Red Teaming (RT) is a vulnerability assessment tool, which assists in decision making and understanding competition in military operational tactics (Yang *et al.*, 2006; Abbass *et al.*, 2011). There are two teams formed - Red and Blue. The motivations and interests of the defending force are represented by the Blue Team, while opposing force are charged with challenging and attacking the defenses and represented by the Red Teaming (Yang *et al.*, 2006; Abbass *et al.*, 2011). Although RT is originated in the context of military conflict simulation or wargaming, it can actually be applied to understand any entity that has the potential to influence a system or organization and its decision making (Yang *et al.*, 2006; Abbass *et al.*, 2011). In such case, the opposing force is an entity whose objectives compete with the system's and that takes actions impeding the system (Abbass *et al.*, 2011). As explained in Abbass *et al.* (2011): "The primary focus of RT is on 'Red': how to represent it and how to reproduce its behavioral patterns".

Traditionally, RT has been performed manually and, typically, requires close collaboration from a group of experts relevant to the targeted system and in charge of assessing the system's vulnerability (Chua *et al.*, 2008; Ranjeet *et al.*, 2011). However, this manually intensive process can be time-consuming, expensive and is usually bounded by the perspectives of humans "thinking inside the box" (Chua *et al.*, 2008; Xu *et al.*, 2009; Ranjeet *et al.*, 2011), particularly when considering a complicated system or organization with a multi-sided nature (Yang *et al.*, 2006; Xu *et al.*, 2009). As a result of practical constraints, only a limited range of scenarios can be explored. Hence, computer-based simulations have been used as a promising solution to conquer the 'human' limitations inherent in manual RT (Yang *et al.*, 2006; Ranjeet *et al.*, 2011; Hingston and Preuss, 2011).

For some simple RT scenarios which can be precisely described by a set of equations, relatively little computational effort is required to obtain exact solutions through traditional analytical optimization methods; however, many real-world situations require definitions or descriptions which can capture dynamic details and interdependencies among various elements. An agent-based simulation system is often a good choice to model such a complicated scenario as a whole system, where many low-level interdependencies and correlations can lead to macro-phenomena (Yang *et al.*, 2006; Hingston and Preuss, 2011).

Since computer-based simulation (e.g., an agent-based simulation model) is much cheaper and less time-consuming, by applying advanced computing technologies (such as search-based optimization methods and machine learning methods), search mechanisms for RT become achievable. These methods are usually able to expose surprises and shocks in a system or organization since they do not suffer from 'blind spots' as humans do (Upton and McDonald, 2003; Upton *et al.*, 2004; Yang *et al.*, 2006; Hingston and Preuss, 2011). Evolutionary Computation (EC) is one of such algorithms and is able to assist ART especially in decision support for two reasons: firstly, as a population-based algorithm, it is capable of maintaining groups of good strategies instead of one single best strategy, so the decision-maker can then obtain an overview of all optional strategies and make a decision based on practical bias not represented in the simulation; and secondly, it can handle multiple criteria problems in a natural way by employing a multi-objective evolutionary mechanism.

One example of the combination of agent-based simulation and EC optimization methods is Automated Red Teaming (ART) (Upton and McDonald, 2003; Upton *et al.*, 2004; Yang *et al.*, 2006; Choo *et al.*, 2007; Xu *et al.*, 2009). It is a concept which complements the manual RT effort by analyzing a scenario through automatically searching for critical weaknesses in the investigated system by simulation. In essence, a typical ART task is to fix the Blue (targeted system) strategy, and then search for a Red strategy which is able to exploit some weakness in the predetermined Blue strategy and 'defeat' it; the system's weakness found by RT can then be used to either improve or assist the manual CR effort and improve the Blue's strategy, in a more focused way. The typical ART task described could be viewed as the traditional ART task and the ART approaches applied in this situation could be named as one-sided ART.

In a one-sided ART procedure (Upton and McDonald, 2003; Upton *et al.*, 2004; Choo *et al.*, 2007), only the parameter values which define the behavior or capabilities of the Red Team are evolved by EC algorithms, in order to optimize its efficiency (in terms of evaluation metrics) against the Blue. Although the one-sided ART promises to automatically uncover weaknesses in Blue, a manual analytical process is still necessary so that these identified weaknesses are resolved and the Blue's strategy improved.

However in practice, a new weakness may arise during the process when a Blue strategy is adjusting by taking account of a particular Red strategy, and it may need to be investigated by some other Red strategy. In another word, the improvement course, which is primary to fulfill a traditional ART task, is not guaranteed to converge. To resolve the local optima issue, an alternative approach named Automated Co-Evolution (ACE) is applied to search for good Red and Blue strategies simultaneously by co-evolving strategies for both in an agent-based simulation system (Lauren *et al.*, 2009; Seng *et al.*, 2009; Hingston and Preuss, 2011).

The co-evolutionary search engine wrapped around the simulation environment employs mainly an 'All versus Best' approach where all optimal strategies for Red Team are evolved against the best Blue strategy, and vice versa (Abbass *et al.*, 2011).

The parameter values which define the behavior or capabilities of both Red and Blue are co-evolved. The ACE approaches complement the one-sided ART in the manner of automating the analytical course required to enhance the Blue's strategy against the corresponding Red (Decraene *et al.*, 2010; Abbass *et al.*, 2011). The extension of one-sided ART to ACE increases the search spaces significantly, which permits exploring more diverse simulation systems; whereby one could devise more effective and strong defense plans against adaptive adversaries.

Although ACE is more advanced than the one-sided ART, it still has the same limitation as the one-sided RT does: in any implementation of automated red teaming, the representation of Red's behaviors or capabilities is a set of pre-defined parameter values imposing considerable constraints on gaining a complete understanding of Red's motivation, objective, attitudes and behavior in real world (Abbass *et al.*, 2011). For the case where Red Team has higher adaptability and system complexity, and its behavior and capacity can not be easily captured in a elegant set of parameter values, one needs more sophisticated model to define and represent Red system. Since "Multi-Agent Systems (MAS) are a natural representation of systems (Abbass *et al.*, 2011)", employing the MAS to define and model Red is a promising approach.

Computational Red Teaming (CRT) (Abbass, 2009; Abbass *et al.*, 2011), is a computational environment that integrates computational intelligence techniques and multi-agent systems. It offers the means to search massive spaces of possibilities governed by uncertainty and complex networked dynamics quickly and find possible and relevant solutions in these spaces. The basic philosophy of CRT is to provide a computational environment in which competition between the system under investigation and scenarios representing situations of risk or uncertainty can be modeled, and superior strategies can be identified (Abbass *et al.*, 2011). In CRT framework, "Multi-agent simulation systems use simulation as a computational device to reproduce the range of behaviors that the underlying MAS exhibits (Abbass *et al.*, 2011)". CRT has been used successfully to evaluate risk in safety-critical operations (Abbass, 2009; Alam *et al.*, 2009; Abbass *et al.*, 2010b,a; Alam *et al.*, 2010b; Abbass *et al.*, 2011).

The principle of CRT is to develop a framework, such as a multi-objective evolutionary-based risk assessment (MEBRA) (Abbass *et al.*, 2009, 2011), where we can identify and evaluate system-level risks in all sorts of situations that are of concern to decision makers. The framework has four generic building blocks: scenario representation, scenario generation, scenario evaluation and scenario mining. Scenario representation is to appropriately design the chromosome representation of scenarios which can be a high level description of the problem domain; scenario generation is to automatically generate scenarios which represent a spectrum of events that can affect the objectives; scenario evaluation is to identify risk patterns in scenario space to quantify system-level risk assessments; scenario mining is to mine through the common patterns and trends existing among scenarios to system vulnerabilities by data-mining techniques (Abbass *et al.*, 2009, 2011).

By iterating through these four building blocks, MEBRA offers a novel way of modelling, identifying and understanding risks that are present in complex systems characterized by cause-effect networks and intricate interdependencies (Abbass *et al.*, 2009, 2011). In the blocks of scenario generation and scenario evaluation, it utilizes a search engine to search over the problem space; whereby reciprocal interactions between the blue and red systems are modeled. In ATM, blue is normally the system under investigation and red the scenarios representing situations of risk which create vulnerability in the system and throughout the search engine, their populations play against each other. The framework of single-objective based risk assessment (SEBRA) (Zhao *et al.*, 2010; Abbass *et al.*, 2011) is a special case of MEBRA. It is applied when the system has a single objective only – and not multiple objectives as in the case of MEBRA.

So far, CRT has been mainly working in non-cooperative environment. Non-cooperative CRT has been used successfully to evaluate risk in safety-critical operations. One of the most famous examples is the MEBRA.

CRT aims at seeking as much bottlenecks as possible in advance by exploring areas of vulnerability, thereby assists in the process of discovering the situations which have not been encountered before and cuts down future damage recovery treatment efforts. However, we have to accept that not all areas of vulnerability can be found; after all, we only deal with models not with reality itself. In practice, many chromosomes - especially in the first generations - are not constrained enough to generate many failures. Only as the evolution converges to better areas of vulnerability, more bottlenecks appear in the scenario.

2.5 Co-operative Co-evolutionary Algorithm

2.5.1 Single-objective Co-operative Co-evolutionary Algorithms

Genetic algorithms (GAs) are population-based evolution-guided stochastic search techniques inspired by natural selection and natural genetics (Holland, 1975; Goldberg, 1989). They are highly simplified and abstract computational models which evolves one homogeneous population of individuals representing a global solution. In a GA, a pre-defined fitness function is applied to an individual, and individuals are evaluated immediately after their birth (Paredis, 1995). The reason that GAs perform better than other traditional optimization and search techniques is twofold: firstly, their searches for improvement are computationally simple yet powerful; and secondly, restrictive assumptions about the search space are not necessary (Goldberg, 1989). GAs have been successfully applied to many numerical and combinatorial optimization problems in recent years (Sarker *et al.*, 2002).

However, as most real world problems originate from a complex environment, which is not only influenced by an individual's own actions but also by other individuals interacting with each other, when applied to large and complex problems, classical GAs often lose their effectiveness and advantages (Paredis, 1995). In order to tackle such problems where traditional GAs tend to be difficult to apply or perform poorly, Paredis (1995) proposed a universal framework, named as co-evolutionary computation (CeC), to boost the performance of a genetic search through problem decomposition.

Contrary to its traditional, non-coevolutionary counterpart, CeC evolves more than one population (termed sub-populations), which represent specific parts of the global solution (Paredis, 1995; Dorronsoro *et al.*, 2011). All these sub-populations evolve simultaneously and presumably improve the global solution. Also in CeC, a more partial but continuous fitness evaluation is utilized, for adaptation to the complex environment (Paredis, 1995). The fitness of an individual is evaluated in collaboration with those of other individuals from other sub-populations. This collaboration can be either positive or negative, leading to co-operative or competitive co-evolution respectively (Paredis, 1995).

The Competitive Co-evolution (CC) model is often compared to predator-prey relations, where prey has a evolutionary pressure to defend themselves better, responding to that predators in future generations develop better attacking abilities (Paredis, 1995). This idea can be summarized as *arms races*, where "success on one side is felt by the other side as failure to which one must respond in order to maintain one's chances of survival" (Paredis, 1995). CC algorithms then apply this idea into optimization problem. Usually, two sub-populations are evolved and the fitness of an individual is determined by competitions with other individuals. The losing sub-population adapts to compete the winning sub-population aiming to become the new winner (Goh and Tan, 2009). Although a discussion about the CC algorithm is beyond the scope of this work, there are many relevant works available for the interested reader (Hillis, 1990; Angeline *et al.*, 1993; Rosin and Belew, 1997; Lohn *et al.*, 2002).

The Co-operative Co-evolution (CoC) model is inspired by symbiotic interactions where different species live together in a mutually beneficial relationship (Yang *et al.*, 2008a; Goh and Tan, 2009). It is regarded as an automatic approach to implement the *divide-and-conquer* strategy: the objective system is decomposed into many smaller modules, each of which is assigned to a species evolved mostly separately, and then they are combined together to form the whole system (Yang *et al.*, 2008a; Goh and Tan, 2009). Generally, a number of insolated sub-populations are evolved co-operatively; whereby forming global solutions to solve complex computational problems. The fitness of an individual is determined by co-operations with other individuals by utilizing a positive fitness feedback mechanism, where the winning on one side enhances the survival possibility of the other (Yang *et al.*, 2008a; Goh and Tan, 2009).

Implementations of CoC algorithms can be classified into two basic categories – single-level and two-level co-evolutions – based on the interactions between individuals and populations (Khare *et al.*, 2004). In single-level co-evolution models (Keerativuttitumrong *et al.*, 2002; Maneeratana *et al.*, 2004; Iorio and Li, 2004; Tan *et al.*, 2004; Dorronsoro *et al.*, 2011), the sub-components are evolved in genetically independent sub-populations, and the fitnesses of individuals are assigned in collaboration with those of the individuals in the remaining sub-populations, while, in two-level co-evolution models (Moriarty and Miikkulainen, 1997; Barbosa and Barreto, 2001; Garcia Pedrajas *et al.*, 2005), global solutions (systems) and local-solutions (sub-components) are co-evolve simultaneously in the form of one population evolving all sub-components and another evolving systems. The contribution of one sub-component in the first population to different systems in the other population is used to evaluate the sub-component. (Khare *et al.*, 2004)

The three main basic ingredients consisted in a typical CoC algorithm are: firstly, a decomposition method by which the complex system is divided into many sub-components; secondly, a collaboration mechanism in which one individual from a certain sub-population is evaluated in combination with individuals from the other sub-populations; and thirdly, an optimization operation which includes crossover, mutation, and other evolutionary operators designed for each sub-component evolving in its own sub-population (Chen *et al.*, 2010). A conventional CoC framework processes each sub-population is a round-robin fashion, where one sub-population is being evolved while all the remaining sub-populations are held fixed (Sofge *et al.*, 2002). The optimization of all individuals in one separate sub-population is called a *phase*, while one iteration over all sub-populations called a *cycle*. The CoC paradigm was initially proposed for single-objective (SO) optimizations, the most prominent of which are listed hereafter.

Potter and De Jong (1994) officially introduced the initial CoC model for function optimization, named co-operative co-evolutionary genetic algorithms (CCGAs). Its idea is to evolve complex complete solutions in the form of interacting co-adapted sub-components (Potter and De Jong, 1994). The main procedures the proposed model are: firstly, the targeted problem is divided into sub-components based on the decision variables (one-dimensional based strategy); secondly, each sub-population evolves a part of sub-components using a standard GA; and thirdly, the complete solutions are evaluated through exchanging individuals co-operatively among all sub-populations. Although the CCGAs choose the GA as the method to evolve the sub-populations, the authors also pointed out that any other evolutionary algorithm (EA) could potentially be applied.

Potter and De Jong (2000b) later proposed a generalized architecture for evolving co-adapted sub-components without human involvement, which provides computational extensions to their previous CoC model (Potter and De Jong, 1994). Such extensions can enable the emergence of an appropriate number of interacting sub-components, evolve them to a reasonable level of generality, and facilitate their adaptation as other sub-components change over time (Potter and De Jong, 2000b). Performance improvements brought by such dynamically evolutionary pressure are demonstrated as this approach has the ability to scale up to larger and more complex problems than is possible using standard EAs.

Wiegand *et al.* (2001) and Wiegand *et al.* (2002) made further efforts towards selecting collaborators for individual's evaluation. An empirical analysis of various collaboration mechanisms are presented in (Wiegand *et al.*, 2001), and advice about how to choose an appropriate one for a particular problem are offered. The foremost lesson provided is that, generally, an optimistic approach is the best method for collaboration credit assignment (Wiegand *et al.*, 2001). Subsequently, a further analysis of the effects of representational bias on collaboration mechanism choices is provided in (Wiegand *et al.*, 2002). The results demonstrate that the standard 'greedy' collaboration method in which the single best individual is selected from each of the other sub-populations, is reasonably robust across two kinds of representational bias analyzed, in most situations.

Apart from implementing various collaboration mechanisms to improve the algorithmic performance, Sofge *et al.* (2002) proposed a blended population approach to change the search procedures in the existing CoC algorithms. The idea is to combine the advantages of both CoC algorithms and traditional EAs, by allowing individuals in sub-populations to shift to a common population during the evolutionary cycle. This procedure encourages greater mixing of sub-components to better handle epistatic interactions between them as the evolution progresses.

There are also some studies on the problem decomposition strategies in the numerical optimization domain (Yang *et al.*, 2008b; Chen *et al.*, 2010). These works are motivated by the desire to tackle nonseparable problems which consist of tight interacted decision variables. Yang *et al.* (2008b) first proposed a CoC-based algorithm implementing a random grouping scheme and adaptive weighting for problem decomposition, and a differential evolutionary algorithm for optimizing each sub-component. However, this algorithm needs a pre-defined problem-dependent parameter – group size – which is difficult to determine in practice.

Yang *et al.* (2008b) later adopted a multilevel strategy to overcome this limitation. Based on the problem under investigation and the stage of the evolution, a set of group sizes are selected to construct a problem decomposer pool. Different decomposers indicate different group size, enabling the production of sub-components with different interaction levels. Although this framework is able to self-adapt between different group sizes according to their historical performance, the determination of an appropriate decomposer pool is difficult due to the unknown relationships among interacting variables (Chen *et al.*, 2010).

Chen *et al.* (2010) then presented a new framework combining previous CoC model with variable interaction learning mechanism. It is a two-step approach: learning step and optimization step. The learning step is initially consider all variables as independent, each of which evolved in a sub-population. Such that interactions among the decision variables are discovered iteratively. The second step is to optimize these sub-populations with traditional CoC model.

All the above reviewed works have proven that co-operative co-evolution is a promising area for identifying solutions for tackling large and complex problems.

2.5.2 Multi-objective Co-operative Co-evolutionary Algorithms

Most real-world search and optimization problems naturally involve multiple objectives. Multi-objective optimization aims to optimize several components of a vector of objective functions simultaneously. Contrary to single-objective optimization, the multi-objective problem usually does not have a single solution that optimizes all criteria concerned, but a set of solutions, known as the POS — set of Pareto-optimal solutions. Any Pareto-optimal solution is optimal in the sense that no improvement can be made in one criterion without degradation of at least another criterion. Since none of the solutions in the Pareto-optimal set is absolutely better than any other, anyone of them is an acceptable solution. Which solution should be chosen depends on the decision-makers, preferences and various problem-related factors. Hence, a decision-maker is typically interested in knowing as many Pareto-optimal solutions as possible (Deb and Kalyanmoy, 2001).

A straightforward way to solve multi-objective optimization problems is to take the weighted sum of the objectives. The weighted sum approach transforms the problem to a single objective problem; allowing the use of single objective optimization methods. There are two profound drawbacks of this method. First, the obtained solutions are highly sensitive to the weight vector. Second, the method can only generate the whole non-dominated set — through repeated call of the optimization algorithm with different weights — if the Pareto-front is convex; otherwise many solutions will be missed.

Because a Evolutionary Algorithm (EA) deals with a population of candidate solutions, it seems natural to use it in multi-objective optimization problems to find a group of Pareto-optimal solutions simultaneously. The main characteristic of EA, which differs from traditional optimization techniques or heuristic methods such as simulated annealing and tabu search, is the simultaneous evaluation of many solutions. This enables a parallel search and potentially avoids convergence to a local optimum that does not coincide with the global one.

In practice, a multi-objective (MO) optimization algorithm is expected to find a set of non-dominated solutions, which belong to the optimal Pareto front or as close as possible to it, spreading uniformly along it. Therefore, a MO algorithm must achieve two primary goals: guiding the search towards the global Pareto front and maintaining the population diversity throughout the Pareto front. Among the most well-known and state-of-the-art MO evolutionary algorithms are the Niched Pareto Genetic Algorithm (NPGA) (Horn *et al.*, 1994), the strength Pareto evolutionary algorithm 2 (SPEA2) (Zitzler *et al.*, 2001), and non-dominated sorting genetic algorithm II (NSGAII) (Deb *et al.*, 2002).

The NPGA (Horn *et al.*, 1994) works as a Pareto-based technique, which demonstrates that MO evolutionary algorithm is able to approximate the optimal set in a single simulation run. It employs a tournament selection, where a pair of randomly selected individuals are compared with each other, to select individuals for reproduction. Then a niche count is calculated based on the fitness sharing procedure, in order to measure the distance between two individuals in the objective space. However, the importance of elitism has not been recognized and incorporate explicitly in NPGA.

The SPEA2 (Zitzler *et al.*, 2001) is an elitist multi-objective evolutionary algorithm. Its working principle is to employ an strength raw fitness assignment strategy which provides a niching mechanism based on the Pareto dominance; while in the case where most individuals do not dominate each other, a density estimation strategy is incorporated to get the most promising solutions among those having identical raw fitness values. The strength raw fitness and density estimation value are then summarized to be assigned to each individual as its fitness value. The density estimation strategy is based on calculating the distances to the k-th nearest neighbor, where boundary solutions are always retained.

The NSGA-II (Deb *et al.*, 2002) is an elitism-based approach which is the most well-known and referenced algorithm in the multi-objective literature (Dorronsoro *et al.*, 2011). It is characterized by its selection operators which preserves elitism through fast-non-dominated-sort, and makes use of a new parameterless measure of density of solutions during neighborhood search (crowding-distance-assignment). The fast-non-dominated-sort strategy creates a mating pool by combining the parent and offspring populations; then the individuals in the mating pool are sorted according to their rank values, and best solutions are selected based on their fitness (rank values) to create a new population. In addition, the crowding-distance-assignment is used to discriminate between individuals having identical rank value, ensuring diversity is maintained among non-dominated solutions.

Designing of co-operative co-evolutionary algorithms for MO optimization is challenged with respect to many issues that are caused by the interaction with the MO optimization, such as individual evaluation (representative selection), fitness (credit) assignment, incorporation of various elitist, niching strategies and selection mechanisms.

Individual evaluation and fitness assignment strategies affect the algorithmic performance with respect to the accuracy and diversity (Zitzler *et al.*, 2000; Goh and Tan, 2009). In SO optimization, an individual in a sub-population is evaluated by composing a complete solution with the best individuals from all the other sub-populations, and the fitness assignment for the evaluated individual is straightforward; while in MO optimization, there will be more than one best individuals in each sub-population and each individual is associated with more than one objective values. Therefore, appropriate representative selection and fitness assignment mechanisms are necessary for identifying the solutions that approximate the true Pareto-optimal front.

Elitism and niching strategies are two primary operators for converting a SO algorithm to a MO algorithm. Elitism is able to significantly speed up the algorithmic performance by enhancing the convergence properties and preventing the loss of good solutions once they are found (Zitzler *et al.*, 2000; Deb *et al.*, 2002). Niching is a significant diversity-preservation mechanism, which ensures diversity in a population in order to maintain a wide variety of equivalent solutions. Under the effect of niching, the population of solutions is dynamically stable under the selection pressure (Mahfoud, 1995; Horn, 2003). Apart from elitist selection (elitism), there are various other selection operations available in the literature, such as roulette-wheel selection, stochastic universal sampling, truncation selection, tournament selection (Back, 1994; Schmitt, 2004). Along with fitness assignment mechanism, selection procedures guide the search towards the true global Pareto front.

Keerativuttitumrong *et al.* (2002) presented a Multi-objective Co-operative Co-evolutionary Algorithm (MOCCGA), which integrates the co-operative co-evolutionary effect in (Potter and De Jong, 1994) and the search mechanisms utilized in multi-objective GA(Fonseca and Fleming, 1993). In each subpopulation, the objective values for each individual evaluated twice in collaboration with either the best or a random individual from the other sub-populations. The fitness value of each individual is assigned based on the rank in the local Pareto front in its own sub-population. The selection operator is stochastic universal sampling method. However, the performance of MOCCGA is limited due to the local Pareto optimality perception and the lack of elitism.

Maneeratana *et al.* (2004) proposed an extension of the MOCCA by incorporating elitism in the form of a fixed size archive in each sub-population to store the nondominated solutions. The individual is evaluated once in combination with the Non-dominated individual with the best degree of crowding. The fitness value is assigned based on the rank in its own sub-population. The niching strategy is based on the crowding distance. However, like MOCCGA, the algorithm also suffers from the limited performance due to the localized perception of nondominance.

Iorio and Li (2004) presented an extension of NSGAII (Deb *et al.*, 2002) – non-dominated sorting co-operative co-evolutionary algorithm (NSCCGA). Individuals are evaluated in collaboration with random individuals from the best non-dominated set in the other sub-populations. The fitness of the individual is assigned based on the non-dominated solutions set via non-dominated sorting. The elitist solutions from the previous generation are reinserted into the subpopulations in the next generation. Tournament selection based on the non-dominated level and crowding distance value as in (Deb *et al.*, 2002). The local non-dominance also limits the performance of NSCCGA.

Tan *et al.* (2004) proposed the co-operative co-evolutionary algorithm (CCEA), which makes use of the co-operative co-evolutionary mechanism specially designed for multi-objective optimization problem. The individual is evaluated in the co-operation with two individuals from each subpopulation, and the better one is kept for the objective values. The fitness is assigned based on the individual's rank against the nondominated solutions stored in an external archive. Tournament selection is based on the rank and niche count. This algorithm is the first co-operative co-evolutionary approach that employs a globalized perception of elitist.

After being introduced in (Tan *et al.*, 2004), CCEA was compared with various MOEAs on more benchmark test problems, with respect to more performance indicators in (Tan *et al.*, 2006). It showed that the CCEA produced competitive and robust results in finding the non-dominated solutions.

Contrary to the trend of developing multi-objective co-operative co-evolutionary algorithms for a general problem domain, there are some efforts have been made to some specific problem domain.

Garca-Pedrajas *et al.* (2002) proposed the multi-objective co-operative networks (MOBNET), which is a multi-objective co-operative co-evolutionary algorithm for artificial neural networks. The individual's fitness is assigned based on the competitive fitness paradigm as in (Ficici and Pollack, 2001). The best $p\%$ of each sub-population is replicated in the next generation and individuals selected from the best $p\%$ by roulette selection refill the rest $1 - p\%$ of each sub-population.

Xing *et al.* (2006) presented a multi-objective co-operative co-evolutionary algorithm to construct fuzzy classification system. A single scalar fitness function is used to combine three objectives. The individual is evaluated in collaboration with the best and another two random individuals from the other sub-populations, and the minimum value is assigned as the fitness to the evaluated individual. The best individual in each sub-population is reinserted into the next generation. Tournament selection is used as the selection operator. However, the co-operative co-evolutionary mechanism is actually suitable for single objective problem and no non-dominated solutions is considered and stored.

Xing *et al.* (2007) later introduced a non-dominated sorting collaboration mechanism based on the NSGAII (Deb *et al.*, 2002). The non-dominated set with respect to all the search space is selected and stored for next generation. The selection is based on the non-dominated level and crowding distance sorting. Although this algorithm is advanced regarding to the globalized perception of non-dominance, the weighted objective function, which transfers the multiple objectives to a single objective, is not a good indicator for optimality.

Dorronsoro *et al.* (2011) proposed three multi-objective co-operative co-evolutionary algorithms (CCMOEA) for continuous and combinatorial optimization based on NSGAII (Deb *et al.*, 2002), SPEA2 (Zitzler *et al.*, 2001) and MOCell (Nebro *et al.*, 2007). The main contributions in this paper are threefold: firstly, a generic framework is presented, allowing building different CCMOEAs; secondly, distributed archive management in the form of one archive per sub-population is introduced and a global non-dominance is generated at the final process; finally, the individual is evaluated in co-operation with a set of randomly selected non-dominated solutions from the other sub-populations.

Table 2.1 demonstrates the comparison of different strategies which have been employed in historical research on multi-objective co-operative co-evolutionary algorithms. The strategies we are interested in are fitness assignment, evaluation mechanism, selection method, elitism strategy, niching Strategy.

All the Multi-objective Co-operative Co-evolutionary Algorithms reviewed in this section have two common features with the one proposed in this thesis: firstly, the problem is manually decomposed into various sub-components as a priori (static population management); secondly, each sub-component is represented by a genetically isolated sub-population (single-level co-evolution).

2.5.3 Evolution of the Reviewed Literature

To better understand previous works and their relationships with this thesis, we synthesize an overall picture of how the reviewed literature evolved until now (see Figure 2.5 and 2.6).

Demand Management with Capacity Constraints	*Odoni (1987); Richetta and Odoni (1993); Vranas et al. (1994).*	*Bertsimas and Patterson (1998); Bertsimas and Patterson (2000).*	*Sridhar et al. (2002); Ball et al. (2003).*	*Lulli and Odoni (2007); Myers and Kierstead (2008); Bertsimas et al. (2008); Rathinam et al. (2009).*	*Bertsimas et al. (2011).*
Capacity Management with Demand Constraints	*N/A*	*N/A*	*N/A*	*Joint Planning and Development Office (2007); Kopardekar et al. (2007); SESAR Consortium (2007); Kopardekar et al., (2008).*	*EUROCONTROL (2011).*
Dynamic Capacity-Demand Balance	*N/A*	*N/A*	*N/A*	*N/A*	***This Thesis***
Air-Ground Information Exchange	*N/A*	*N/A*	*N/A*	*Korn et al. (2006); Prevot et al. (2007); Haraldsdottir et al. (2007); Oberheid and Soffker (2008); Isaacson et al.(2010).*	*Swenson et al. (2011); Uebbing-Rumke and Temme (2011).*
Air-Ground Collaboration from System-level Perspective	*N/A*	*N/A*	*N/A*	*N/A*	***This Thesis***
Arrival Management with Departure Constraints	*N/A*	*N/A*	*N/A*	*Bohme et al. (2007); Delgado et al. (2009).*	*N/A*
Air-Departure Cooperation at some Ground Resources	*Gilbo (1993).*	*Gilbo (1997); Anderson et al. (2000).*	*Mayer and Swancy (2005).*	*Rehwald et al. (2010).*	*N/A*
Air-Departure Cooperation from System-level Perspective	*N/A*	*N/A*	*N/A*	*N/A*	***This Thesis***

Earlier 1995 2000 2005 2010 Now

Figure 2.5: Evolution of the Reviewed Literature (Part I)

Fixed TMA Airspace Design	*STARs and SIDs.*	*Becher and Formosa (2000).*	*Krozel et al. (2004).*	*MITRE CAASD, 2007; EUROCONTROL (2007); Coppenbarger et al. (2007); Boursier et al. (2007); Ivanescu et al. (2009); IMG (2010); Alam et al. (2010a).*	*Alam et al. (2011).*
Dynamic TMA Airspace Design	*N/A*	*N/A*	*N/A*	*N/A*	***This Thesis***
Computational Red Teaming	*N/A*	*N/A*	*Upton and McDonald (2003); Upton et al. (2004).*	*Yang et al. (2006); Choo et al. (2007); Chua et al. (2008); Abbass (2009); Alam et al. (2009); Xu et al. (2009); Lauren et al. (2009); Seng et al. (2009); Decraene et al. (2010); Abbass et al. (2010a); Abbass et al. (2010b); Alam et al. (2010b); Zhao et al. (2010).*	*Abbass et al. (2011); Ranjeet et al. (2011); Hingston and Preuss (2011).*
Single-objective Co-operative Co-evolutionary Algorithm	*Potter and De Jong (1994).*	*Moriarty and Miikkulainen (1997); Potter and De Jong (2000).*	*Barbosa and Barreto (2001); Wiegand et al. (2001); Sofge et al. (2002); Wiegand et al. (2002); Khare et al. (2004).*	*Yang et al. (2008a); Yang et al. (2008b); Goh and Tan (2009); Chen et al. (2010).*	*N/A*
Multi-objective Co-operative Co-evolutionary Algorithm	*N/A*	*N/A*	*Keerativuttitumrong et al. (2002); Garca-Pedrajas et al. (2002); Maneeratana et al. (2004); Iorio and Li (2004); Tan et al. (2004).*	*Tan et al. (2006); Xing et al. (2006); Xing et al. (2007).*	*Dorronsoro et al. (2011).*
Single-objective Co-evolutionary Computational Red Teaming	*N/A*	*N/A*	*N/A*	*N/A*	***This Thesis***
Multi-objective Co-evolutionary Computational Red Teaming	*N/A*	*N/A*	*N/A*	*N/A*	***This Thesis***

Earlier 1995 2000 2005 2010 Now

Figure 2.6: Evolution of the Reviewed Literature (Part II)

Table 2.1: Multi-objective Co-operative Co-evolutions Comparison

Historical Work	Fitness Assignment	Evaluation Mechanism	Selection Method	Elitism Strategy	Niching Strategy
Keerativuttitumrong et al. (2002)	Rank-based Fitness Assignment	Maxi{with 1 best, with 1 random}	Stochastic Universal Sampling	N/A	Fitness Sharing in Objective Space
Garca-Pedrajas et al. (2002)	Rank-based Fitness Assignment	With Best	Roulette Selection	Best P% in sub-populations	Competitive Fitness Paradigm
Maneeratana et al. (2004)	Rank-based Fitness Assignment	With Best	Stochastic Universal Sampling	Non-dominated Set in Sub-populations	Crowding Distance
Iorio and Li (2004)	Rank-based Fitness Assignment	With 1 Randomly from Best Non-dominated Set	Tournament Selection	Non-dominated Set in Sub-populations	Crowding Distance
Tan et al. (2004)	Rank-based Fitness Assignment	With Best	Tournament Selection	Non-dominated Set in the entire population	Fitness Sharing in Objective Space
Tan et al. (2006)	Rank-based Fitness Assignment	Maxi{with 1 best, with 1 random}	Tournament Selection	Non-dominated Set in the entire population	Fitness Sharing in Objective Space
Xing et al. (2006)	Equals Min{with 1 best, with 1 random, with 1 random}	Minwith 1 best, with 1 random, with 1 random	Elite Selection; Tournament Selection	Best Ind in Sub-populations	N/A
Xing et al. (2007)	Weighted Objectives	With Best & Random	Tournament Selection	External Archive	Crowding Distance
Dorronsoroet al. (2011)	Objectives	With a set of randomly selected non-dominated set	Binary Tournament Selection	Non-dominated Set in Sub-populations	Crowding Distance

Chapter 3

Proposed Integrated TeA System

To evaluate TeA airspace concepts in an integrated TeA system and understand their vulnerabilities, a systematic modelling of system behaviors and interactions among subcomponents is needed. In this chapter, we introduce a TeA system model which integrates arrival and departure operations and combines air- and ground-side resources. It is developed as a part of this thesis to evaluate advanced TeA airspace concepts and discover system-level vulnerabilities through a simulation-based computational environment. Although other air traffic simulation systems are presented in the literature, no study to our knowledge has attempted to model an integrated TeA system in a systematic manner in which the interdependency between traffic distributions and the dynamics of ground resources are modelled, and arrival-departure cooperation can be implemented in a fast-time simulation environment. This system is also designed to be able to model and implement different types of ATM procedures: the present day's arrival and departure procedures (STARs and SIDs) and an advanced ATM concept known as dynamic CDA. The objective is to provide more insight into the complex interactions of various TeA subsystems and pave the way for later investigations into TeA airspace configuration design. This chapter describes the proposed integrated TeA system in details: input and output, data structure, design, functionality and how to achieve the arrival-departure integration.

3.1 Overview of the Proposed Integrated TeA System

The fundamental goal underlying the proposed integrated TeA system is to develop a simulation and modelling environment where the arrivals and departures are integrated, while considering the interdependencies between the traffic distributions and the dynamics of ground resources. The main simulation engine of the integrated TeA system embraces an arrival manager and a departure manager which models the basic air traffic and ATM operational features on the airport surface and the transition airspace surrounding it (e.g. TeA airspace, waypoints, aircrafts and trajectory generation) that are essential for evaluating any air traffic concept. Other modules which implement queue management, air-ground resources models and arrival-departure cooperation are built around the core engine.

The integrated TeA system is defined as an environment combining the airport's ground transportation network (runways, taxiways and gates) and its surrounding transition airspace, whose radius is set to 25 nm in this work. Hence the modelling domain for the integrated TeA system is comprised of the air (transition airspace) resources model and the ground resources model. However, in the simulation, a larger simulating scope is necessary as the kernel generating the flights needs certain time to be processed. Therefore, an extra 225 nm flying distance is given to inbound traffic before they start to execute the transition air routes and ground routes; while extra 1 minute time window is designed for outbound traffic prior to their departure.

In any busy airport, even with a fairly simple airport layout, managing ground operations and dealing with all incoming and outgoing airport ground traffic can be a very challenging task. An airport represents a highly dynamic environment with continuously changing resources (Kuster and Jannach, 2006). Tower controllers are required to provide ground operations for both arrivals and departures to facilitate safe and smooth ground traffic flows within uncertain constrained ground events. A constrained ground event is a disruption to the availability of surface resources, including all runways, taxiways and gates, in the airport. These include snow, ice, slush or water on a runway, which can reduce aircraft braking and directional control and increase runway occupancy time. An increase in runway occupancy time leads to a reduction in airport arrival and departure acceptance rates (throughput) due to the need for increased inter-aircraft spacing. This reduction is aggravated by the closure of runways and certain runways being unusable. Disruptive ground movements could propagate elsewhere in a TeA system (e.g., air traffic in the transition airspace) and cause flight delays and system inefficiency. Thus, the inherent uncertainty of ground events and their interactions with other TeA system components heavily influence airport capacity and potentially cause a rise in the complexity of a TeA system.

The Air Traffic Operations and Management Simulator (ATOMS) Alam *et al.* (2008) provides a high-fidelity simulation and modelling environment for exploration, development and evaluation of advanced ATM concepts. The integrated TeA system presented here will be blended in ATOMS, in order to implement the integration of arrivals and departures using a shared ground-air network. As a case study, an assumed airport model inspired by Sydney's Kingsford-Smith Airport (hereafter referred to as Sydney Airport) is used for designing the air-ground resource models.

It is noted that there are some assumptions in abstracting or representing the integrated TeA system, which might lead to errors in the absolute results which should be as small as possible. Although the abstraction of the real world TeA system operations may not faithfully reflect all details in reality, it is argued that the simulation system model developed in this chapter comprises the key features which are necessary to reveal arising macro-phenomena at the system level which are typical of an integrated TeA system in the real world (Decraene *et al.*, 2010). Therefore, the developed model of a TeA system can expose the emerging behavior or phenomena of interest without the burden of simulating unnecessary detailed features. In addition, in the following chapter the focus and objective of implementing the simulation is on relative investigation and evaluation on different scenarios and the absolute results are less important.

3.2 Input/Output

The input to the integrated TeA system consists of both static and random factors. The static factors include TeA air resource model, ground resource model and capacity for each resource. The dynamic factors contain the time and space context of arrival flight plans, departure flight plans and ground events. The human factors such as ATC controller is excluded from explicit consideration. The detailed information that each item contains is listed as follows.

- TeA Air Resource Model: there are two types of air networks representing either the fixed STARs and SIDs configurationS or the dynamic CDA structure;
- Ground Resource Model: there are two ground networks representing the airport ground resources (runways, taxiways and gates), one for arrival and the other for departure. The interdependencies arising from overlapping resources are captured by the ground resource record;
- Resource Capacity: this is a user defined factor and in this work, it is assumed that each resource has a capacity of 1;
- Arrival Flight Plan: Aircraft Name, Aircraft Type, Estimated Time of Activation, Activation Point, Outer Marker Point, Designated Runway, Array of Designated Taxiways and Designated Gate;
- Departure Flight Plan: Aircraft Name, Aircraft Type, Estimated Time of Activated, Designated Gate, Array of Designated Taxiways and Designated Runway; and
- Ground Event: Event Location, Event Name, Start Time and Duration.

The output from the integrated TeA system is modified arrival (or departure) flight plans and a total flight delay value, listed as follows:

- Modified Arrival Flight Plan: Aircraft Name, Aircraft Type, Estimated Time of Activated, Activation Point, Outer Marker Point, Modified Runway, Modified Array of Taxiways and Modified Gate;

- Modified Departure Flight Plan: Aircraft Name, Aircraft Type, Estimated Time of Activated, Designated Gate, Modified Array of Taxiways and Modified Runway;

- Total flight delay for an arrival: The difference between the ETA of an arrival flight from its OM to the requested gate and its actual time of arrival (ATA) at the actual gate from its respective OM; and

- Total flight delay for a departure: The difference between the ETA of a departure flight from its requested gate to the OM and its ATA to the assigned OM from its respective gate.

3.3 Data Structure

This section discusses the major data structure of the integrated TeA system including air resources models (dynamic CDA and fixed STARs & SIDs models), ground resources model, air resources records and ground resources records.

3.3.1 Air Resources Model

3.3.1.1 Dynamic CDA

Finding dynamic CDA trajectories in the transition airspace can be modelled as a problem of path planning in three dimensions. In ATM, this problem takes on unique dimensions due to aircraft performance constraints that are imposed on it in the approach phase, such as limited maneuverability (low thrust), speed restrictions and altitude constraints. Apart from the hard safety constraints, the other competing objectives are to minimize noise, emissions and fuel consumption. For

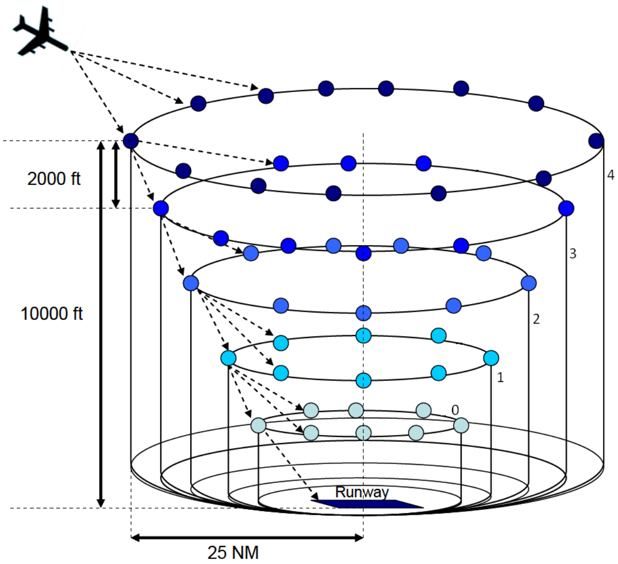


Figure 3.1: Conceptual representation of transition airspace divided into concentric circles and wedges acting as trajectory change points for dynamic CDA trajectory generation Alam *et al.* (2010a).

details of the dynamic CDA procedures, please refer to paper Alam *et al.* (2011). In this section, we present only the transition airspace model used in this work and its implementation for arrivals and extension to departures.

As demonstrated in Alam *et al.* (2011), the problem search space (transition airspace) is equally divided with a 5 nm safety separation defined as a set of five concentric cylinders with a runway (touchdown point) at the center, as illustrated in Figure 3.1. The height of the transition airspace is set to 10,000 ft and the radius to 25 nm (transition airspace radius (TAR)). The outermost cylinder (denoted as Ring 4) has a radius of 25 nm and the inner cylinders (Rings 3, 2, 1 and 0) radii of 20, 15, 10 and 5 nm respectively, as calculated by Equation 3.1. The outermost cylinder's height is 10,000 ft which corresponds to the starting altitude of the CDA and the inner cylinders have heights of 8000, 6000, 4000 and 2000 ft respectively. Thus, the transition airspace is divided into 5 levels, with each level divided into 2000 ft to give a typical jet aircraft sufficient vertical height to maneuver given a low thrust setting.

$$RingRadius = \frac{TAR \times \ (Ring\,Number + 1)}{5} \tag{3.1}$$

Each cylinder has wedges which represent transition points from one level to another and are spaced 1.5 nm apart to provide safe separations between approaching aircraft (Spence, 2003). The number of wedges for a given cylinder is calculated as:

$$Number\,of\,Wedges = \frac{2\pi \times Ring\,Radius}{Separation\,Distance} \tag{3.2}$$

The angle between the wedges is calculated as:

$$Wedge\,Angle = \frac{2\pi}{Number\,of\,Wedges} \tag{3.3}$$

A transition airspace radius of 25 nm and a separation distance of 1.5 nm give the number of wedge points as 104, 83, 62, 41 and 20 for Rings 4, 3, 2, 1 and 0 respectively. As such, the transition airspace is discretely partitioned into concentric cylinders with artificial waypoints, so that CDA trajectories are generated from one waypoint to another to identify all possible air routes.

The cylindrical airspace model comes from the fact that the typical Terminal airspace is cylindrical in shape. The distance between five rings is considered for safe maneuverability vertically and that between wedges provides safe separation between approaching aircraft. This simplified modelling helps in assigning and reserving the shared resources in a simpler fashion. The radius of the cylinder represent the typical IAF distance for an aircraft at approximately 10,000 ft and the FAF distance (5 nm from runway) is a typical representation of the same in an operational environment.

For the purpose of implementing interactions between arrivals and departures, the modelling designed in the dynamic CDA is also used for departures in this chapter. More precisely, the artificial trajectory points (also referred to as wedge points) modeled in Figure 3.1 are implemented with sufficient generality to allow for arrivals and departures passing concurrently through the transition airspace with shared resources (concentric cylinders with artificial waypoints) rather than being restricted to only arrivals. By searching in the same problem search space, 'CDA-like' routes are generated in a similar way as they are in the CDA. They are computed in real-time and both laterally and vertically optimized for given objectives. Any two departures in the same departure airspace can follow two different trajectories: there is no standard ascent departure route. The major difference is that air routes for departures are generated from the innermost cylinder (final approach fix) to the outermost cylinder (initial approach fix) instead of in the opposite direction as they are in CDA. For distinction, in this chapter, we refer to this departure procedure as the dynamic continuous ascent departure (CAD) and the air routes for departures obtained from it as CAD routes.

It is assumed that the capacity of each wedge point is 1, which means each wedge point can only be used by the single aircraft to which it is assigned for a certain duration. For instance, if an arrival is in a wedge point for a particular duration, this point is blocked and cannot be used by any other arrival or departure during that time window.

3.3.1.2 Fixed STARs and SIDs

Current ATM operations are implemented by filing an aircraft's standard route (STAR or SID) after it is cleared into a TeA airspace. Using the current STAR and SID procedures, the arrival and departure flows are processed separately in the TeA airspace around Sydney Airport. Figure 3.2 shows a typical STAR chart – GALGA SIX ARRIVAL for Sydney Airport (Air Services Australia, 2012), while Figure 3.3 demonstrates a typical SID chart – RWY 16L KEVIN THREE (Air Services Australia, 2012).

There are 5 STAR and 14 SID routes serving Sydney Airport. To capture the bottlenecks in a TeA system, we model all STAR or SID way points which could affect its capacity as a network. The interconnections among all way points, and how one leads to another, are modeled as networks shown in Figure 3.4 and Figure 3.5

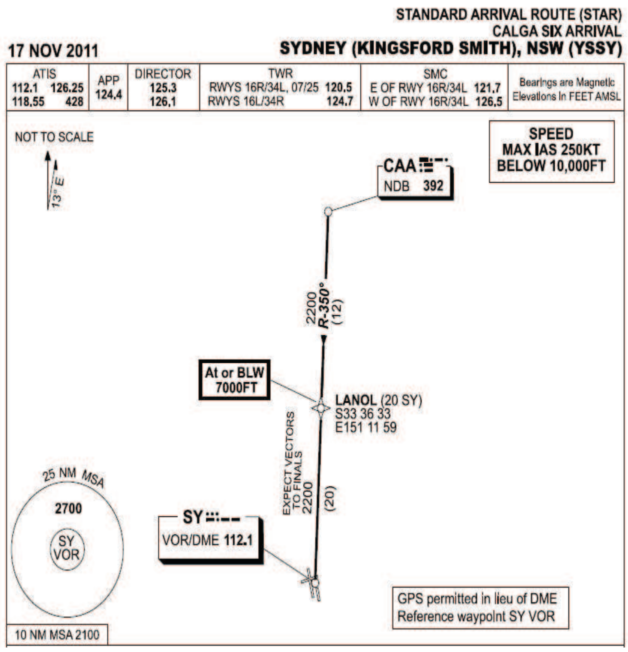


Figure 3.2: STAR example: GALGA SIX ARRIVAL for Sydney's Kingsford-Smith Airport, Air Services Australia (2012)

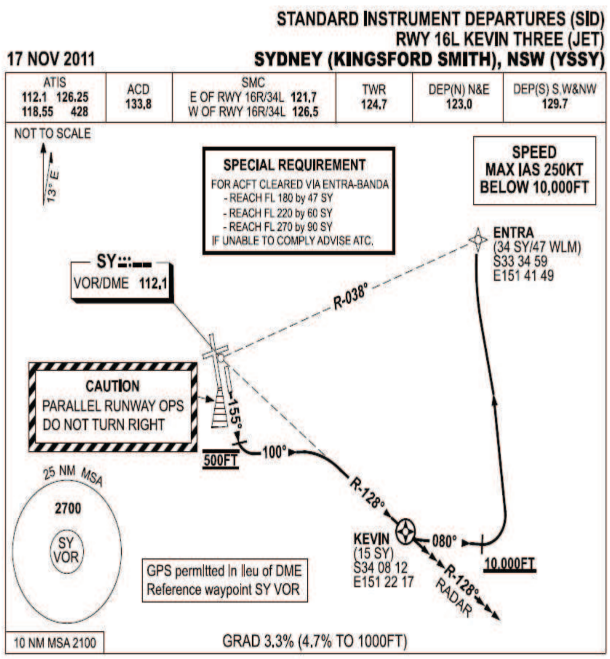


Figure 3.3: SID example: RWY 16L KEVIN THREE for Sydney's Kingsford-Smith Airport, Air Services Australia (2012)

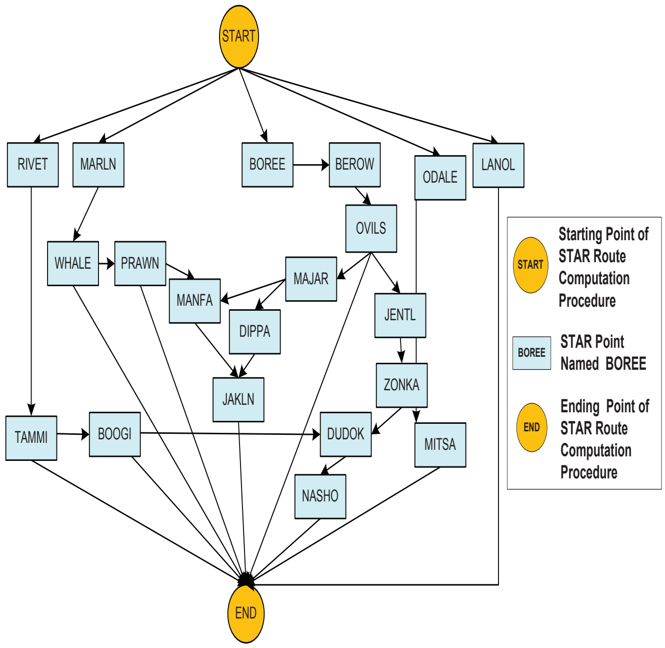


Figure 3.4: The Fixed STARs network for Sydney Airport

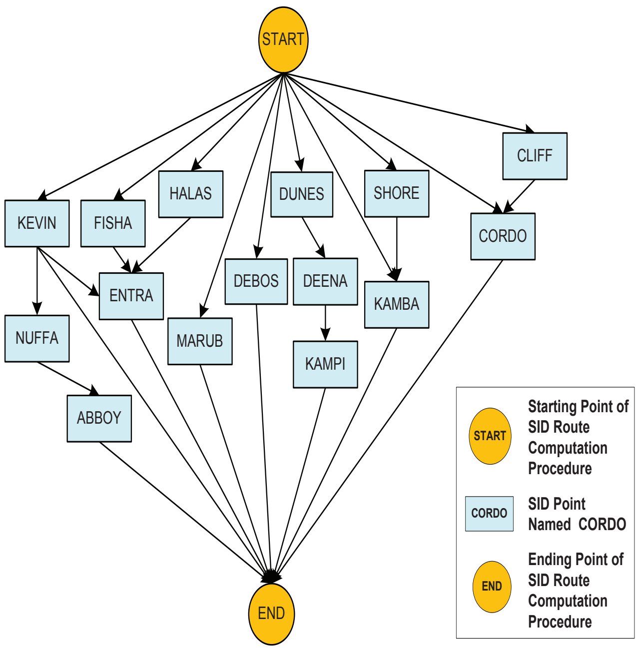


Figure 3.5: The Fixed SIDs network for Sydney Airport

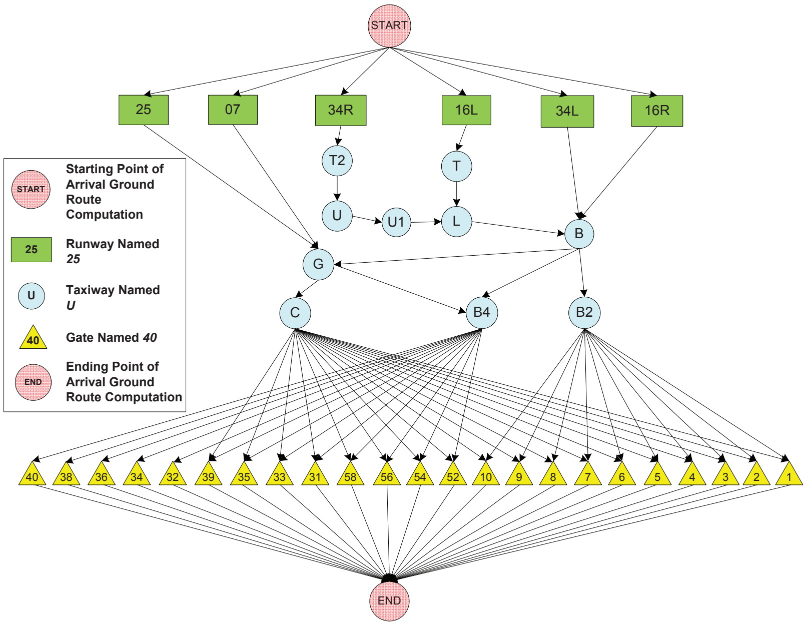


Figure 3.6: Ground arrival network for Sydney Airport

and store all possible STAR and SID routes. It is assumed that the capacity of each resource is 1, which means each STAR (or SID) way point can only be used by the arrival (or departure) to which it is assigned for a certain duration. An arrival (or departure) is assigned a STAR (or SID) route at the time of air route computation based on resource availability. The START and END points in Figures 3.4 and 3.5 represent the starting and ending points of the STAR or SID route computational procedure.

3.3.2 Ground Resources Model

For the sake of simplicity, only the domestic section of the Sydney Airport's terminal area is used to design the ground model which contains medium-spaced parallel runways in the north-south (16/34) direction (3,962m and 2,438m) and an intersecting/cross runway in the east-west (07/25) direction (2,530m). In total, there are 6 runways, 16 taxiways and 23 gates. Each runway is treated separately, i.e., runway 16 and runway 34 are independent of each other. The interconnections between these resources, and how one leads to another, is modeled as two networks - arrival and departure ground models, as shown in Figure 3.6 and Figure 3.7. The

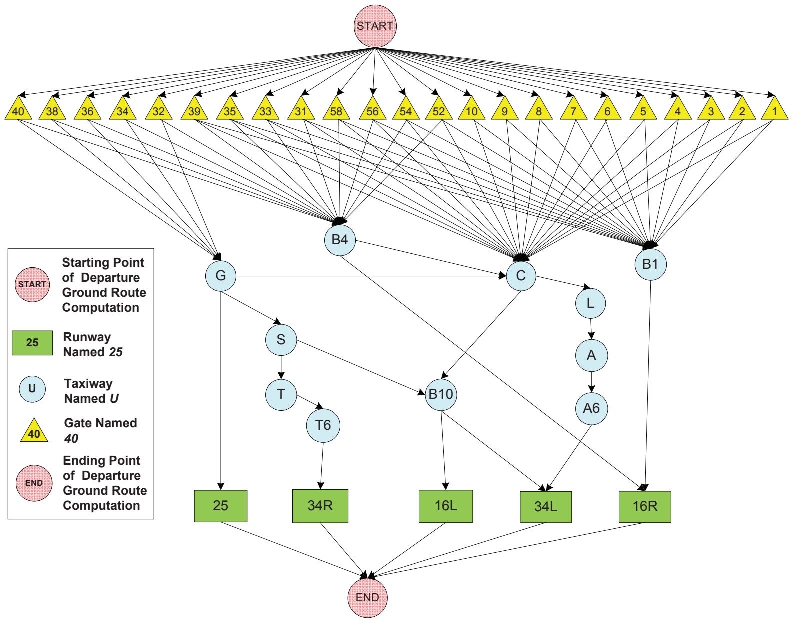


Figure 3.7: Ground departure network for Sydney Airport

former network data structure stores all possible connections from the runways to different taxiways, and from the taxiways to different gates. The latter stores all connections from the gates to different taxiways, and from the taxiways to different runways; for example an arriving aircraft is assigned a particular gate, the network stores the links for the various networks available to route the aircraft to that gate.

The data structure for each ground point in the ground resource network consist of: ID, Capacity, Aircraft Types, Latitude, Longitude, Altitude and Speed, each of which is explained as follows.

- ID: is the name of the ground point;
- Capacity: is the number of aircrafts this ground point can handle simultaneously;
- Aircraft Type: represents which types of aircraft this ground point can handle;
- Latitude: is the latitude value of this ground point;
- Longitude: is the longitude value of this ground point;
- Altitude: is the altitude value of this ground point; and

- Speed: is the speed constraint for the ground point.

As the air resources model does, it is assumed that the capacity of each ground point is 1 and each ground resource can only be used by an aircraft (either arrival or departure) to which it is assigned for a certain duration.

An arrival (or departure) aircraft is assigned its runway, taxiway and allocated gate at the arrival (or departure) airport when its initial flight plan, which is used as an input to the simulator, is generated. However, this may change at the time of its ground route computation depending on resource availability and/or other ground events, such as snow on the runway, gate closure, etc. The flight ground route modification process works in the following manner. If an aircraft is assigned a new runway/taxiway/gate due to the unavailability of a certain resource, the aim is to meet the scheduled surface route while minimizing changes to the existing route. If there is no possible way of leading an aircraft from its assigned runway to its assigned gate, an alternative gate is selected. If the runway is completely closed, the search is performed using the next available runway and the aircraft ground route is updated accordingly.

The START and END points in figures 3.6 and 3.7 represent the starting and ending points of the ground route computational procedure. When combining the TeA airspace procedures with the ground procedures, the END of the STARs network is connected to the START of the the ground arrival network, while the END of the ground departure network is connected to the START of the SIDs network. For arrivals, when the air route computational process starts, the on-computing arrival aircraft is at the START point in Figure 3.4. Once a proper STAR route is chosen for this arrival by going through all possible STAR routes stored in the STARs network, it comes to the END point in Figure 3.4. Thereafter, the arrival moves to the START point in Figure 3.6 which means its ground route computation starts. When an acceptable ground route is chosen for this arrival, it moves forward to the END point in Figure 3.6. Departures start the ground route computation by standing at the START point in Figure 3.7. After searching through the ground departure network, the departure obtains a proper ground route which satisfies all constraints and moves to the END point in Figure 3.7. Then, it goes to the START point in Figure 3.5 to begin the SID route computation and moves to the END point in Figure 3.5 after being assigned an acceptable SID route.

3.3.3 Air Resources Records

To accomplish the integration of arrivals and departures using share air-ground resources and assess the impact of dynamic ground events on arrival and departure traffic distributions, a data structure which records the occupancies of air resources is necessary. As a result, an array named *Transition Airspace Resources Records* is designed, whose elements are presented as follows.

- Resource Name: represents the name of the air resource;
- Start Time: represents the time when the aircraft starts to occupy the air resource;
- Duration: represents how long the aircraft will occupy the air resource; and
- Aircraft Type: the category of the aircraft – heavy, large or small.

When an aircraft is assigned an appropriate air route (as in Section 3.4.6 and Section 3.4.7), any way point (air resource) in this route is recorded in the *Transition Airspace Resources Records*, along with its name, time window for the aircraft to stay and the aircraft's type.

3.3.4 Ground Resources Records

As can be seen, the arrival and departure ground models share a large proportion of ground resources, such as runways, gates and most of the taxiways. To integrate the arrivals and departures, a data structure which records the occupancies of ground resources is designed. It is an array named *Ground Resources Records* whose elements are presented as follows.

- Resource Name: represents the name of the ground resource;

- Start Time: represents the time when the aircraft starts to occupy the ground resource;

- Duration: represents how long the aircraft will occupy the ground resource; and

- Aircraft Type: the category of the aircraft – heavy, large or small.

When an aircraft is assigned an appropriate ground route (as in Section 3.4.6 and Section 3.4.7), any way point (ground resource) in this route is recorded in the *Ground Resources Records*, along with its name, time window for the aircraft to stay and the aircraft's type. When computing the ground route for an aircraft, we check the possible ground resource availability by searching through the record table; for example, if an arrival is about to be assigned to gate 40 at time T, but that gate is occupied by another aircraft (arrival or departure) at the same time according to the record table, then gate 40 is not available for this new arrival at that time.

In addition, the *Ground Resources Records* has to include another type of occupancy – the ground events. When a ground event is about to occur in a ground resource, the name of the resource, the start time and duration of the event and what type of aircraft the event will effect will be recorded in the *Ground Resources Records*.

3.4 Integrated TeA Simulation

This section discusses the architecture and design principles of the integrated TeA system including queue manager, arrival manager, departure manager and arrival-departure integration.

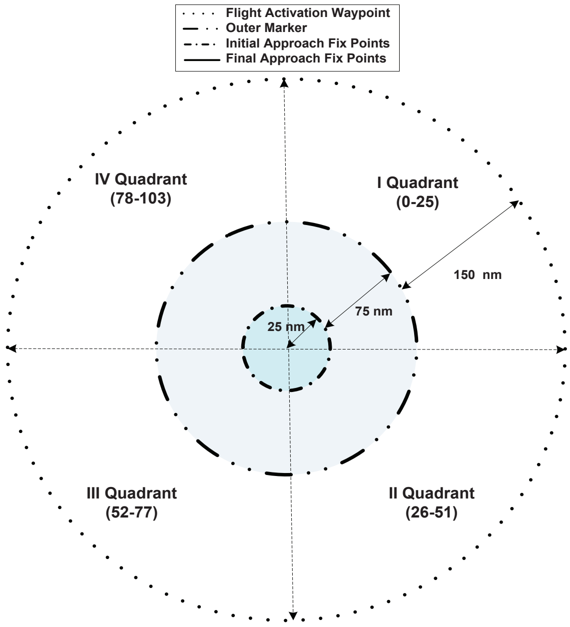


Figure 3.8: TeA-ATOMS simulation scope

3.4.1 Integrated TeA System Design

The overall simulation scope in the TeA-ATOMS simulation is larger then the integrated TeA system, as depicted in Figure 3.8. It is a region of radius of 250 nm around an airport, comprised of part of the en route airspace, the transition airspace and the airport. The starting points (104 in number) for arrivals are set to 150 nm from the outer marker (OM) (consisting 104 way points as well). The circular grid is divided into 4 equal regions of 90 degrees each which we call quadrants, each of which contains 26 starting points. The OM is 75 nm away from the initial approach fix which is the entry to the transition airspace, the radius of which is 25 nm. Arrival aircraft come from their designated starting points based on the traffic distribution. The transition airspace resources model for dynamic CDA or fixed SIDs and STARs (Section 3.3.1) lies between the Initial Approach Fix Points circle and the Final Approach Fix Points circle. The ground resources model (Section 3.3.2) is in the airport.

Any inbound aircraft gets activated at a Flight Activation Waypoint, and flies directly to an OM point based on the aircraft's performance data. An aircraft's activation waypoint and OM point are pre-defined in its flight plan as input and present which direction the aircraft comes from. The simulator calculates a Top of Decent (TOD) point for this aircraft according its initialized flight plan and aircraft's performance data. This TOD point is then inserted into the flight plan as part of the flight route. Before OM point, the arrival is positioned in a conceptual queue (Section 3.4.3) according to its time to reach the OM point. The transition airspace and ground route calculation (Section 3.4.6) for the arrival starts when it reaches the OM point. An entry point to the transition air space model is selected based on the closest distance and minimum variability in heading from the aircraft's current position. Once the arrival is filed a transition air route and a ground route, the way points in these routes will be added to the initialized flight plan. After the arrival reaches the last way point in transition air route, we assumed there is a direct descent to its runway. Then the arrival passes along the airport according to assigned the ground route. When it reaches the gate, the simulation of the arrival flight terminates.

On the other hand, a departure gets activated one minute before being ready to be pushed back from its assigned gate. It then is positioned in the conceptual queue (Section 3.4.3) based on its time to departure from its designated gate. The ground route and transition airspace calculation (in Section 3.4.7) is followed by the modification of the initialized flight plan. When the departure is at the end of its runway, it is assumed there is a direct ascent to the first way point in its transition airspace route. When the departure reaches the last way point in the air route, a OM point is assigned to it based on the closest distance and minimum variability in heading from the aircraft's current position. After it reaches its OM point, the simulation of the departure is finished.

3.4.2 Architecture of the Integrated TeA Simulation

The architecture of integrated TeA simulation consists of the following key components (see Figure 3.9):

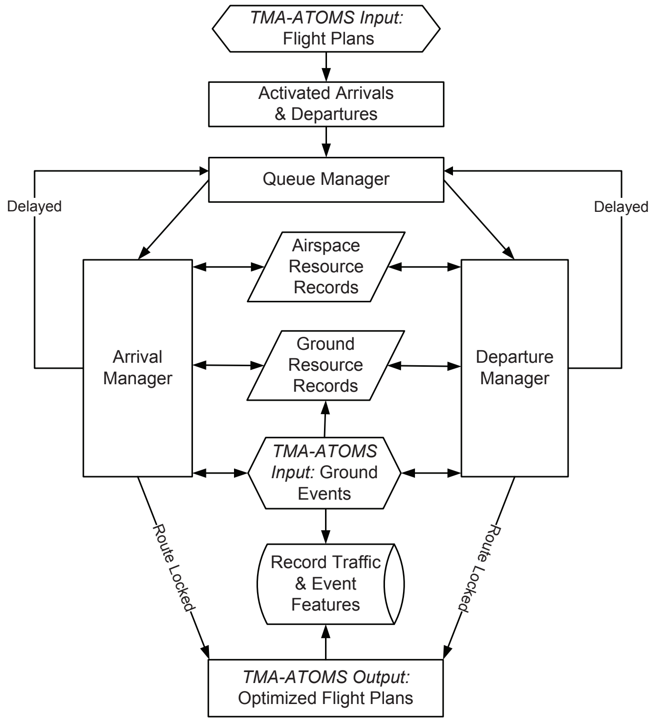


Figure 3.9: Architecture Design of Integrated TeA Simulation

This architecture was designed to be modular and flexible enough to incorporate new air-ground network and ATM operational constraints. Starting from the top:

Given a set of flight plans, we separate them into two conceptual waiting queues – one for arrival and the other for departure – based on their activation time. Once an arrival reaches 75nm before the TeA (OM point), or a departure is one minute prior to being ready to be pushed back from the gate, it becomes active. All active aircraft (arrivals and departures) are processed by a Queue Manager in which they are positioned in a conceptual queue according to their times to be at the OM (for the arrival) or gate (for the departure). Based on the first come first assigned principle, the first aircraft (arrival or departure) is selected to go through the optimization process of the Arrival Manager or Departure Manager accordingly. These managers interact with each other by sharing the same information about the availability of air-side and ground-side resources. If the optimized route for an aircraft is found, the result is output and recorded, otherwise the delayed aircraft is positioned back to the Queue Manager to wait for another optimization process. The procedural details of the Queue, Arrival and Departure Managers are described in the following subsections. After all flights are scheduled, the total number of delays for all flights and delays caused by each potential resource bottleneck are calculated and recorded.

For the sake of simplicity, it is assumed that any Flight Activation Waypoint and OM point is able to handle any type of arriving or departing flight, as long as it is operationally allowed according to the aircraft's performance data. Thereby, any aircraft can fly from one way point directly to another based on the flight route in the flight plan and the aircraft's performance data. Additionally, any way point at the end of air resources modelling (see Section 3.3.1) has a direct descent line to any runway, and vice versa.

Algorithm 3.1 demonstrates the procedure of the integrated TeA simulation:

3.4.3 Queue Manager

Given a set of flight plans, the objective of the queue manager is to sequence the flight plans, and output a set of sequenced flight plans. The data structure of the flight queue is represented by an variable-length array which is comprised by a number of *Flight-Element*. Each *Flight-Element* contains: the sequencing number of the flight, the Estimated Time of Arrival (ETA) of the flight at the OM (for the arrival) or gate (for the departure), name of the flight, aircraft type, the OM point, the designated runway, the array of designated taxiways and the designated gate.

First Come First Served (FCFS) is a prominent scheduling algorithm in Sequencing Theory (Pinedo, 2002). It is the most straightforward and commonly used sequencing algorithm that generates efficient aircraft processing sequences. The basis of this method is the ETA of aircraft at the OM (for the arrival) or gate (for the departure). In FCFS, the aircraft's flight plans get optimized in order of their scheduled arrival times. FCFS is straightforward and favoured by airlines for its fairness and by ATC for its simplicity that puts little demands on ATC workloads.

Note that contrary to the typical aircraft landing algorithm, the minimum time separation between two landing aircrafts for safety purpose is not considered in

Algorithm 3.1 Pseudo Code of Arrival-Departure Integration in TeA-ATOMS Simulation

```
ArrivalFlightPlans; DepartureFlightPlans; GroundEvents; AirRecords;
GroundRecords; FlightQueue;

FlightPlans = ArrivalFlightPlans ⋃ DepartureFlightPlans;

t = 0;

while (FlightPlans ⋃ FlightQueue is not empty) do

  for each flight in FlightPlans do
    if flight is active then
      FlightPlans = FlightPlans - flight;
      FlightQueue = FlightQueue + flight;
    end if
  end for

  FlightQueue = Sorting( FlightQueue);

  firstflight = QueueManager( FlightQueue);

  if ETA(firstflight) <= t then
    if firstflight is arrival then
      firstflight = ArrivalManager (firstflight);
    else
      firstflight = DepartureManager (firstflight);
    end if
    if RouteLocked (firstflight) then
      Update (AirRecords);
      Update (GroundRecords);
      FlightQueue = FlightQueue - firstflight;
    else
      UpdateETA (firstflight);
      FlightQueue = Sorting (FlightQueue);
    end if
  end if

  t ++;

end while
```

the FCFS algorithm. The reason are twofold: firstly, to integrate the arrival and departure traffic, the Queue Manager has to produce a combined sequence of both arrival and departure flights; secondly, different flights will pass through different resources (OM points for arrivals and gates for departures). This safety separation issue will be taken care of in the following section (3.4.4).

3.4.4 Occupancy Time

As one of those elements in a piece of Air Resource Record or Ground Resource Record, the occupancy time indicates how long it takes for an arrival or departure aircraft to pass through a resource. A time window is employed to block the resource for a specific time interval so as to no other aircraft can access to the same resource. For the simplicity, it is assumed that the occupancy time for aircraft staying at each resource is solely based on the type of the aircraft and has same value for air resources as well as ground ones.

The FAA divides aircraft into three weight classes, based on the maximum take-off weight capability. These classes are:

1. *Heavy Aircraft* are capable of having a maximum takeoff weight of 255,000 lbs or more.
2. *Large Aircraft* can have more than 41,000 lbs and up to 255,000 lbs maximum takeoff weight.
3. *Small Aircraft* are incapable of carrying more than 41,000 lbs takeoff weight.

A matrix of the minimum time window is shown in Table 3.1, based on the three category system utilized in todays system.

3.4.5 Parameter Source

The parameters used during the modelling process are discussed below:

Table 3.1: Time window for different aircraft category

Aircraft Category	*Time Window (sec)*
Heavy	120
Large	90
Small	60

- Arrival Ground Network, Domestic section of Sydney Airport, Departure Ground Network, Domestic section of Sydney Airport, STARs Network, All STAR way points, SIDs Network, and All SID way points are from Air Services Australia (2004).
- STAR or SID Way Point Occupancy Capacity is 1. A natural value since two aircraft cannot pass through the same STAR or SID node at the same time.
- Runway Occupancy Capacity is 1. A typical value used in the literature (Gotteland *et al.*, 2001).
- Taxiway Occupancy Capacity is 1. Different values used in the literature such as "One aircraft per Taxiway", "many aircraft per Taxiway separated by 60 m", and "One aircraft per Taxiway segment".
- Gate Occupancy Capacity is 1. A natural value since two aircrafts cannot dock at the same gate at the same time.
- Occupancy Time of Runway is 120, 90, and 60 sec for heavy, medium and light aircraft respectively. The occupancy time of runway depends on the length of runway and several other factors. This is a user-defined parameter which can be adopted to different values. One example of values used in the literaturep Gotteland *et al.* (2001) is: 180, 120, and 60 sec for heavy, medium and light aircraft respectively.
- Hold Pattern is 60 secs. There are four types of typical hold patterns:
    1. Full Circle (360 degrees turn): With 3 deg/sec rate of turn (source: Eurocontrol's Aircraft Database) aircraft takes 120 secs.

2. One Half Circle Turn (180 degree turn): With 3 deg/sec rate of turn (source: Eurocontrol's Aircraft Database) aircraft takes 60 secs.
3. Two Half Circle Turns (180 deg + 180 deg): With 3 deg/sec rate of turn (source: Eurocontrol's Aircraft Database) aircraft takes 120 secs.
4. Path Extension (ATC Manoeuvres): Depends upon the path elongation (can be from 20 secs (metering time matching) in terminal areas to 2 hrs enroute (bad weather)).

- Connection between FAFs and Runway in Dynamic CDA Procedure assumes that any FAF point has access to any runway similar to Alam *et al.* (2011).
- The error between aircraft leaving the gate and arriving at the runway is not considered in this thesis.
- Connection between Exit Points of STARs Network and Runway in Fixed STARs Procedure assumes that any exit points of STARs network has access to any runway. Every STAR may have a specific runway assigned to it. However, if the STAR route is designated as RNAV (area navigation) route. Then that means after last point in STAR route, ATC can vector the aircraft to any runway. Therefore, we have made the assumption: any exit points of STARs network has access to any runway.
- Minimum Time Separation between Landing or Departing Aircrafts is taken as the same value for Occupancy Time of Taxiway and Occupancy Time of Gate as a natural way to stress-test the system. Normally time at a Gate is large but with Stands and Aprons, the flow is larger than the number of available gates.

3.4.6 Arrival Manager

Figure 6.5 presents the procedural flowchart for the Arrival Manager for which the operational details of implementing the dynamic CDA or fixed STARs & SIDs are as follows.

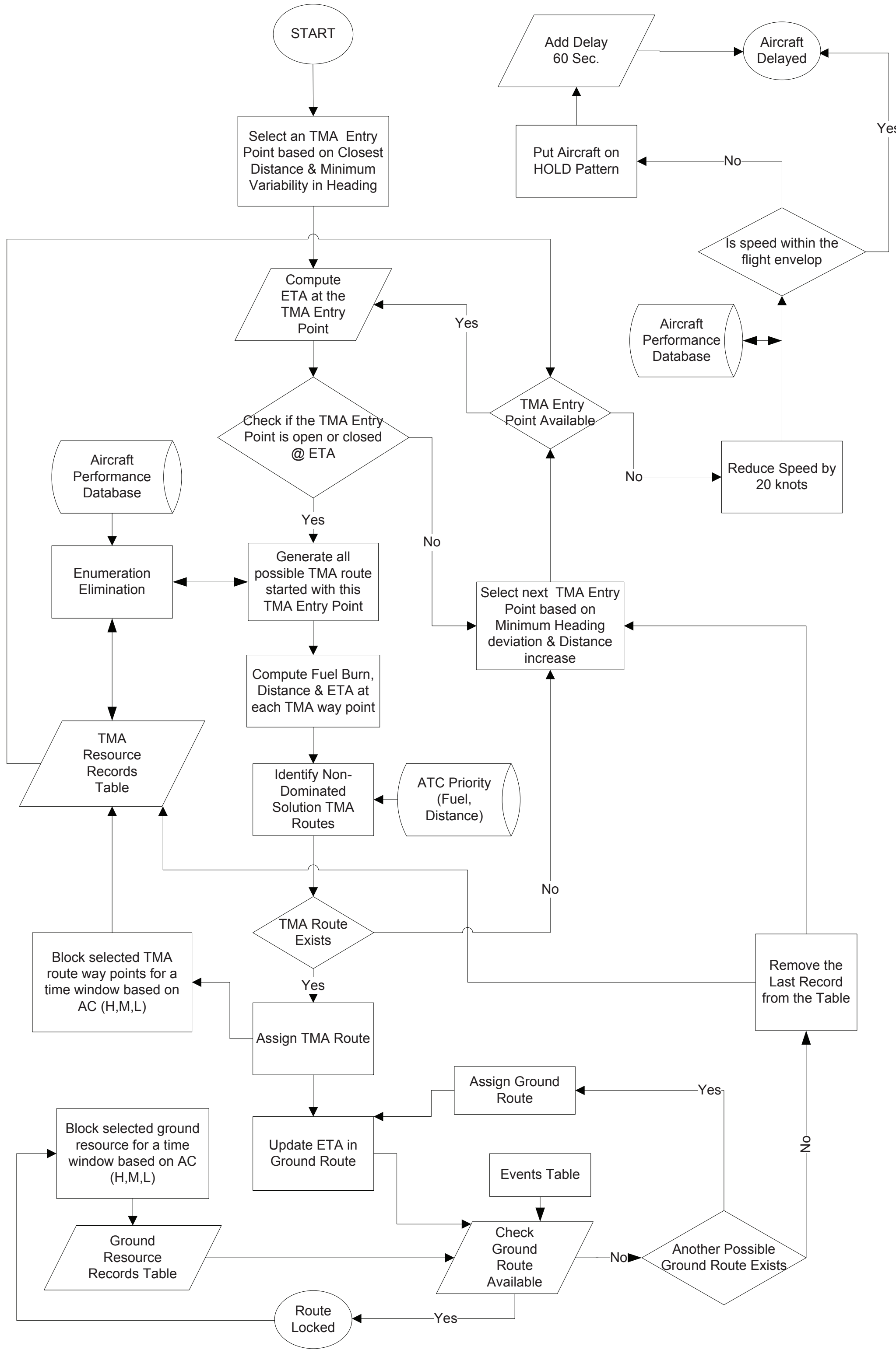


Figure 3.10: Arrival Manager

The Arrival Manager starts from generating the air route for arrivals in transition airspace. Firstly, an entry point of the air route which is closest to the aircraft's current physical position and needs minimum variability in its heading is chosen for the arrival. Then the estimated time of arrival (ETA) to this entry point is computed to check whether it is open or closed at the ETA according to the *Airspace Resource Records*. If yes, a full enumeration of the search space is performed to generate all possible routes from this entry point to the final point of the air route according to the airspace model, followed by eliminating those links that violate the aircraft performance constraints derived from Eurocontrol's Aircraft Database (BADA); if no, the next entry point which meets the objectives (closest distance and minimum variability in heading) is selected, followed by computing the ETA, checking whether the entry point is available at the ETA according to the *Airspace Resource Records* and generating all possible routes again. After searching, if there is no available entry point, the arrival has to be delayed (delay process), either by putting it into a HOLD pattern (adding a delay of 60 seconds) or reducing its speed by 20 knots if the aircraft's performance allows. Considering the ATC priority (by calculating the fuel and distance values) for each air route, we identify a set of non-dominated solution air routes based on the air model. (The concept of "set of non-dominated solutions" is generally used to describe a set of solutions of which any solution is optimal in the sense that no improvement can be made on an objective without degradation of at least another objective. None of the solutions in the set is absolutely better than any other; thereby anyone of them is not dominated by any other.) By updating the ETA in the air route, we check whether all the way points in the air route are available at their ETA times by checking the *Airspace Resource Records*. The first approved air route is selected and all the way points in this route are blocked for a time window based on the aircraft type (as shown in Table 3.1), and their ETAs and occupancy times are recorded in the *Airspace Resource Records*. This ensures that no two flights (arrivals or departures) occupy the same way point in a given time window. If no air route is approved, the arrival needs to find a next available entry point which meets the objectives. Now, we start to process the ground route which is pre-defined in the flight plan. The ETA is updated for each way point on the ground according to the designated ground route. In line with the events table and *Ground Resource Records Table*, if any ground way point (runway, taxiway or gate) is unavailable due to a certain ground event or is simply occupied by another aircraft, the aim is to meet the designated ground route while minimizing changes to the existing route. If there is no possible way of leading an arrival from its assigned runway to its assigned gate, an alternative gate is selected. If the runway is completely closed, the search is performed using the next available runway and the aircraft's ground route is reassigned (if there is another possible ground route) by searching through the ground resource network as shown in Figure 3.6. The ETA for each ground resource is updated, each resource is blocked for a time window based on the aircraft type (as shown in Table 3.1) and the *Ground Resource Records Table* is updated accordingly. However, if there is no available ground route, the arrival aircraft has to find a next available entry point to the TeA which meets the objectives and the last record in the *Airspace Resource Records* is removed. In the end, the available air and ground routes are combined as the full arrival route for this flight.

The difference between dynamic CDA and fixed STARs operations lies mainly in the air-route generation part. The dynamic CDA trajectory generation is based on the transition airspace model in Figure 3.1, while STAR route is generated by searching through the STARs network in Figure 3.4. It is assumed that ends of all air routes (final approach fixes) have accesses to any runway.

3.4.7 Departure Manager

Figure 3.11 presents the procedural flowchart for the Departure Manager for which the operational details implementing the dynamic CAD or fixed STARs & SIDs are as follows.

The Departure Manager starts from generating the ground route for departures in the airport. Firstly, we analyze the ground route which is pre-defined in the

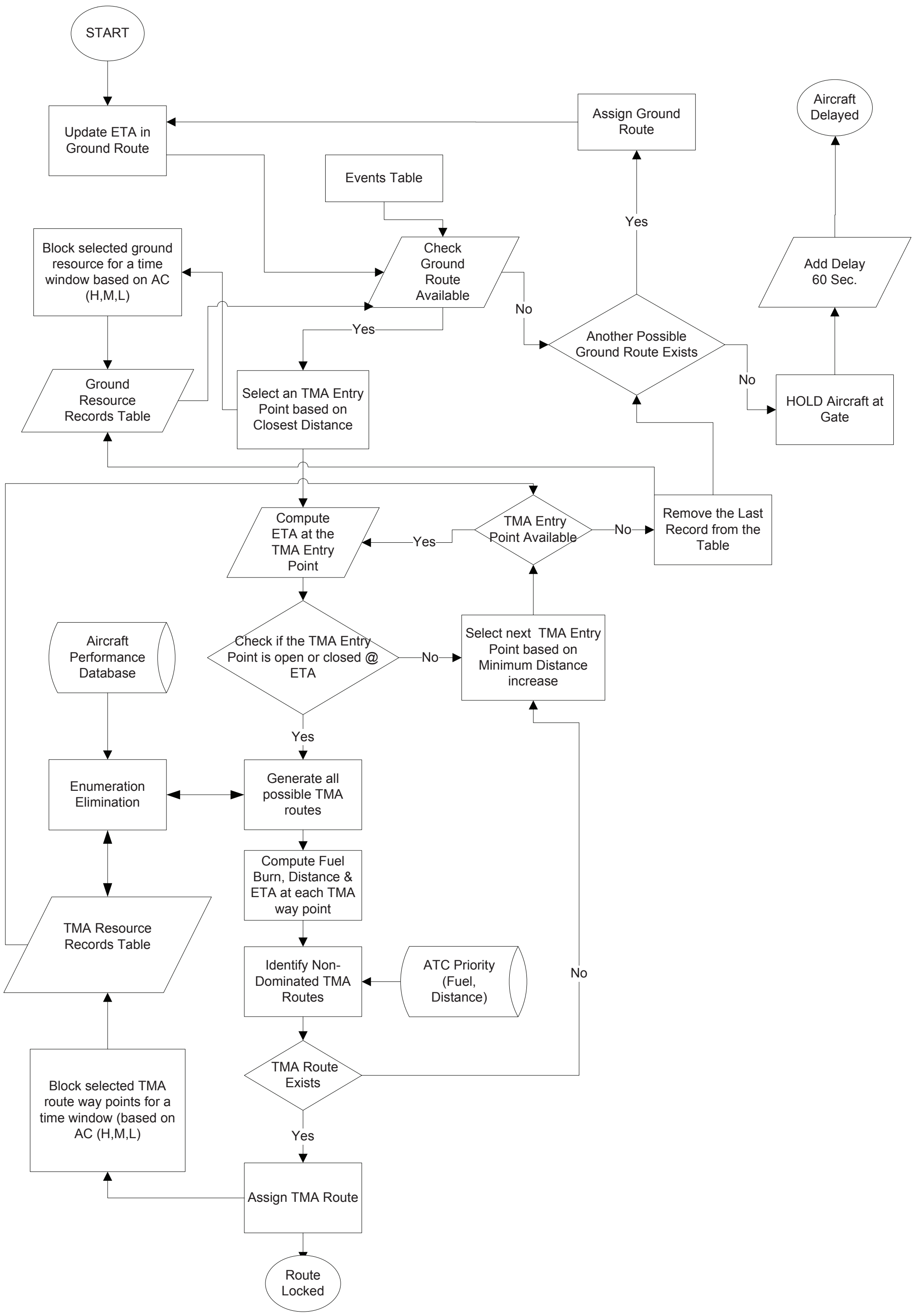


Figure 3.11: Departure Manager

flight plan and update the ETA for each way point on the ground according to the scheduled ground route. In line with the events table and *Ground Resource Records Table*, if any ground way point is unavailable due to a certain ground event or is simply occupied by another aircraft, the aim is to meet the scheduled ground route while minimizing changes to the existing route. If there is no possible way leading a departure from its assigned gate to its assigned runway, an alternative runway is selected and the aircraft ground route is reassigned (if there is another possible ground route) by searching through the ground resource network as shown in Figure 3.7. The ETA for each ground resource is updated, each resource is blocked for a time window based on the aircraft type (as shown in Table 3.1) and the *Ground Resource Records Table* is updated accordingly. However, if there is no available ground route, the departure aircraft has to be delayed by waiting at the gate for 60 seconds (delay process). Then, the air route in the transition airspace is processed. An entry point which is closest to the aircraft's taking-off point is chosen for the departure. The ETA to this entry point is computed to check whether it is open or closed at the ETA according to the *Airspace Resource Records*. If yes, a full enumeration of the search space is performed to generate all possible routes from this entry point to the final point of the air route, followed by eliminating those routes that violate the aircraft performance constraints derived from BADA; if no, the next entry point which meets the objective (closest distance) is selected and is followed by computing the ETA, checking whether the entry point is available at the ETA according to the *Airspace Resource Records* and generating all possible routes again. After searching, if there is no available entry point, the departure needs to find another possible ground route. Considering the ATC priority (by calculating the fuel and distance values) for each route, we identify a set of non-dominated solution CAD routes based on the transition airspace model. By updating the ETA in the air route, we check if all the way points in this air route are available at their ETA times by checking the *Airspace Resource Records*. The first approved CAD route is selected and all the way points in this route are blocked for a time duration based on the aircraft type (as shown in Table 3.1), and their ETAs and occupancy times are recorded in the *Airspace Resource Records*. This ensures that no two flights (arrivals or departures) occupy the same way point in a given time window. If no air route is approved, the departure needs to find another possible ground route. Finally, the available ground and air routes are combined as the full departure route for this flight.

For Departure Manager, the dynamic CDA and fixed STARs operations differ with each other mainly in the air-route generation part. The dynamic CDA trajectory generation is based on the transition airspace model in Figure 3.1, while SID route is generated by searching through the SIDs network in Figure 3.5.

The delay time (60 seconds) in Section 3.4.6 and 3.4.7 for a delay process is a user defined parameter and can be experimented with different values. A different delay time will not affect our relative analysis to different scenarios in the following Chapter (4).

3.4.8 Arrival-Departure Integration

When applying mixed mode operations at an airport, an appropriate arrival-departure integration (ADI) concept must consider the overall traffic situation. Thus the concept must provide a good compromise between the needs of air and ground resources for the arrival traffic on one hand and those for departure traffic on the other.

The coordination concept developed here (as Figure 3.12) takes into account both the arrival traffic situation and the departure situation in the TeA airspace and on the ground. The management of the arrival-departure integration is organized in a centralized manner. With the help of the shared information about the air and ground resources availability, the ADI concept allows the dynamic coordination of an arbitrary arrival and departure manager. For the need of the coordination, the ADI must provide sufficient and timely information exchange and the communication between the connected components is demonstrated in Figure 3.13.

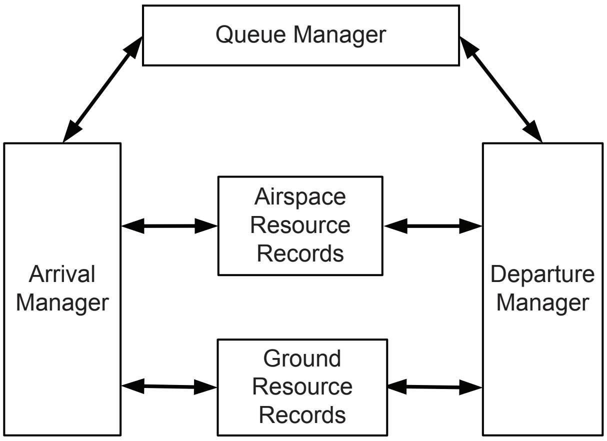


Figure 3.12: Architecture for Arrival-Departure Integration Concept

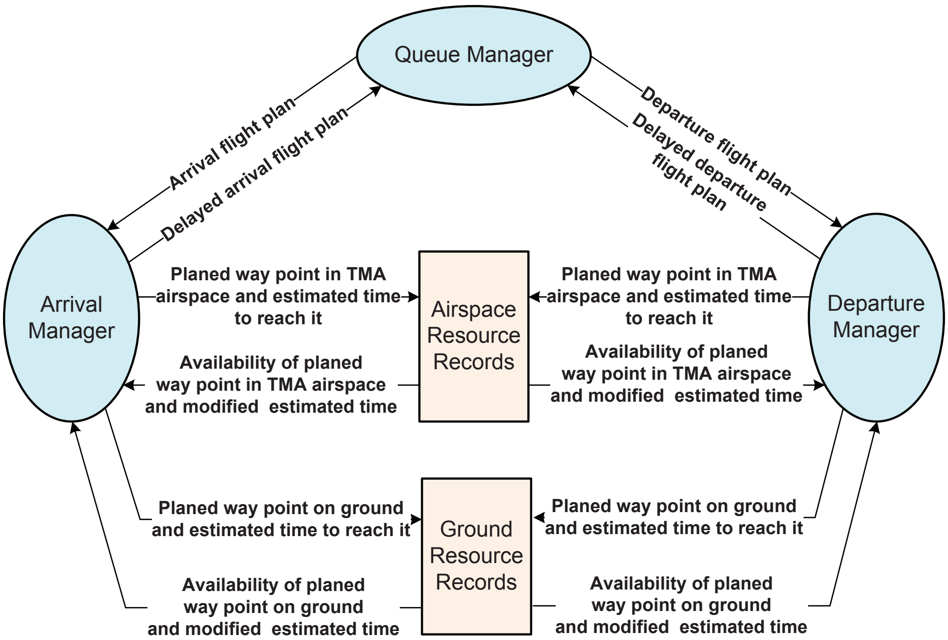


Figure 3.13: Information Exchange between the Connected Components

3.5 Chapter Summary

In this chapter, an air traffic simulation system with a novel representation of an integrated TeA, considering air-ground collaboration and arrival-departure cooperation, is presented for a system-level understanding of integrated TeA concepts. This simulation environment provides more insight into complex interactions of various TeA subsystems and pave the way for developing a simulation-based computational environment, in order to evaluate advanced TeA airspace concepts and understand TeA system vulnerabilities in the next chapter.

To be specific, in this chapter we have presented the design, architecture and various functionalities of the integrated TeA system. The inputs and outputs of this system are also shown. Air- and ground-side resources modelling is explained and validated. Arrivals and departures integration is illustrated. Two types of ATM procedures: the present day's arrival and departure procedures (STARs and SIDs) and an advanced ATM concept known as dynamic CDA are modeled and implemented. Finally queue manager, arrival manager and departure manager for flight management in the integrated TeA are presented.

Chapter 4

Co-Evolutionary Computational Red Teaming

The research question that needs to be resolved in this chapter is: How to evaluate advanced TeA airspace concepts in the integrated TeA system and understand system-level vulnerabilities? In the previous chapter, we developed an integrated TeA system which can provide a simulation environment for co-operating the arrivals and departures, while considering the interdependency between the traffic distribution and the dynamics of ground resources. Two TeA airspace models – dynamic CDA and fixed STARs/SIDs – were embedded in the simulator. In this chapter, the first objective is to provide quantitative evidences that, when implemented in the integrated TeA, dynamic CDA model is advanced than fixed STARs and SIDs model, and the second objective is to understand system-level vulnerabilities in the integrated TeA.

We propose a methodology using the Computational Red Teaming (CRT) framework to identify ground-air network bottlenecks by exploring areas of vulnerability in the integrated TeA. The search engine in CRT relies on single-objective co-operative co-evolutionary search which evolves reciprocal interaction of traffic distributions and ground events (including runways, taxiways and gates). These interactions are considered from the perspective of identifying inefficiencies, while considering the integration of arrival and departure operations. By evaluating these interactions, we are also able to identify "improvement opportunities" in the implementation of two different TeA airspace concepts and, thereby, understand and work-around major bottlenecks which cause system inefficiencies. The advancement of the dynamic CDA model over the fixed STARs and SIDs model in the worst case can be captured quantitatively; thus facilitating later investigations into TeA airspace configuration design based on the dynamic CDA model.

4.1 Overview of the Co-Operative Co-Evolutionary Red Teaming

The scenario space of an integrated TeA comprises interweaving scenarios which are correlated in time and space. As a result, causes and effects are networked and the dynamics of system components become complex. This level of complexity necessitates a simulation-based approach and requires more sophisticated quantitative approaches capable of analyzing the network of interdependencies and evaluating system-level vulnerabilities (Abbass *et al.*, 2009).

Traffic distributions (spatial and temporal) and constrained ground resources (including runways, taxiways and gates) are two main entities which can significantly influence the efficiency of the integrated TeA system. In order to identify the system-level bottlenecks, it is primary to understand the interdependency between these two elements. In addition, the air-side and ground-side subcomponents are highly co-adapted as each subcomponent itself is changing and evolving, in an incorporated TeA system. The performance evaluation of each subcomponent depends on the reciprocal interactions with other components in the system. The change of context can be well captured by a Co-evolutionary algorithm (CEA), which is a biologically-inspired population-based search technique. Because CEA provides an effective means of handling large and complex problems via problem decomposition, it seems natural to use CEA in the problem domain where solutions can be evolved through co-adapted subcomponents interactions, rather than by hand tuning or pre-scripting how scenarios should change.

The Co-evolutionary Computational Red Teaming (CCRT) proposed here is to break a way from the classical approach of evolving one population of integrated TeA scenarios in which the interactions between air traffic and ground events are already pre-scripted as a simple CRT would do, and use Co-operative Co-evolutionary algorithm (CoC) instead to separate them into two populations and make them co-evolve with each other. Thus, the objective of the CoC algorithm in the CCRT framework is to co-evolve increasingly complex air traffic and ground events scenarios so that the integrated TeA can incur maximum failure (in terms of the evaluation metric – total flight delay here).

4.2 Co-Evolutionary Computational Red Teaming Framework

When implementing the CCRT framework to understand system-level vulnerabilities and evaluate advanced ATC concepts in the integrated TeA system, the Blue Team is evidently the developed TeA simulation system under investigation, representing the system's objectives and interests. There are two teams: traffic distributions (spatial and temporal) and constrained ground resources (including runways, taxiways and gates), representing situations of risk which create vulnerability in the system. Throughout the search method, their populations co-evolved with each other, and their behavioral patterns are reproduced, towards the areas of higher vulnerability.

We should note that, what matters in the CCRT is the characteristics of air and ground event scenarios in the area of high vulnerability and how they interact. As such, we are not interested in optimising per se. Our main interest is to use co-evolutionary algorithm as a population-based method, guided by an appropriate fitness function, to explore as many areas of bottlenecks as possible.

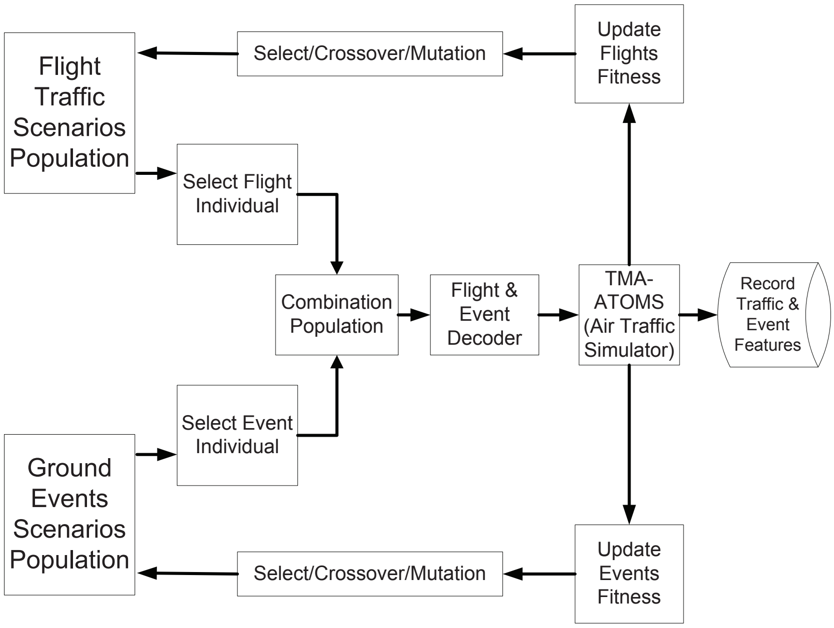


Figure 4.1: Co-operative Co-evolutionary flowchart

4.2.1 Search Mechanism

The search engine employed in the CCRT is co-operative co-evolution technique. One of the simplest approaches for applying co-operative co-evolutionary algorithms (CCAs) is to identify a natural decomposition of the environment (problem to be solved) into two components. Each component involves a species, such that individuals in a given species represent potential components of the environment. Each species is evolved simultaneously in its own population, in isolation from each other, and adapts to the environment through the repeated application of EAs. To evaluate individuals from one species, collaborators are selected with representatives (best individuals from each species) from each of the other species to form a complete solution.

In our problem, there are two naturally decomposed species: air events (flight traffic) and ground events (dynamically constrained ground resources). The challenge lies in how to identify and represent each species, provide an environment in which they can interact and co-adapt, and apportion credit to them for their contributions to the problem-solving activity such that their evolutions proceed without human involvement Potter and De Jong (2000a).

Figure 4.1 illustrates the co-operative co-evolutionary framework in which the following two populations of partial solutions evolve together:

- flight traffic scenarios; and
- ground events scenarios.

Firstly, each population (flight and event) is initialized and random flight and event scenarios are generated and encoded in the chromosome representation. Each population is processed by selecting each individual and combining it with a randomly selected member of the other population. These two-combined individuals form a set of complete candidate solutions for a new population.

Secondly, each new population is evaluated in TeA-ATOMS one by one and the fitness value is calculated and assigned back to the individual undergoing evaluation. Once the entire population has been evaluated, the best individual(s) is selected to be used in evaluating the individuals in the other population. Evolutionary operators (mutation and crossover) are then applied to produce a new generation of this population.

Thirdly, the entire process is repeated for the pre-determined number of generations, which is one of the parameters of the co-evolutionary algorithm.

4.2.2 Scenario Generation

Since our methodology focus on searching on problem spaces rather than solution spaces, the problem chromosome is supposed to describe a circumstance or a scenario that might actually occur. Therefore the scenario generation is primary to automatically generate scenarios which represent a spectrum of events that can affect the objectives. The initialized populations are realistic but not real data. The chromosome representations of the traffic and ground scenarios are high level descriptions of the problems domain; hence they could understand and capture the impact of uncertainty in the problem space on objectives.

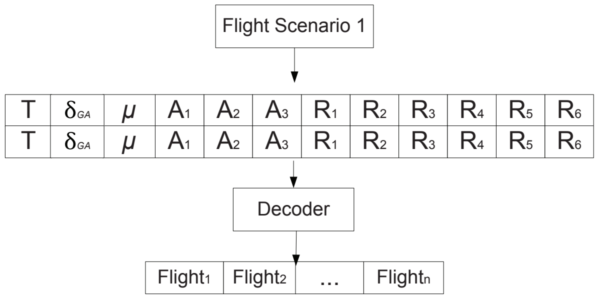


Figure 4.2: Traffic chromosome design showing genomes which encode spatial and temporal distributions for flight in scenarios.

4.2.2.1 Population Design of Flight Traffic Scenarios

To implement the interaction between arrivals and departures, we design a 2-dimensional chromosome to represent an individual in the flight population. Information about the temporal and spatial distributions of flights in the airspace, the aircraft type (light, medium, heavy) and the designated runway at the destination airport are encoded in one chromosome. As shown in Figure 6.6, there are 12 genes in each chromosome, with each gene containing two independent numbers, one for arrivals and the other for departures. The gene values are represented as follows.

- Time (T) is the parameter for the inter-aircraft time distribution which varies in a pre-defined time interval.
- σ_{GA} represents the selection probability of a flight activation point on the OM, the value of which is selected uniformly from the interval [0, 1] and is not used in departures.
- μ controls the distribution around the flight activation point [0, N] and is not used in departures.
- (A_n) represents the three genes that determine the aircraft type (light, medium, heavy) and it is uniformly selected from the interval [0, 1].
- R_n represents the six genes that determine the runway selection probability and is uniformly selected from the interval [0, 1].

T controls the activation time (temporal dimension) of a flight. The flight activation times for two successive flights are based on the inter-aircraft time distribution which is generated using a Poisson process with the value of T being derived from a uniform random number generator in a pre-defined time interval ([45, 120]). An inter-aircraft time distribution of 50 implies the next flight activation time is Poisson-distributed with a mean of 50 seconds.

μ and σ_{GA} are the parameters which control the flight activation point (spatial dimension). Activation point σ is a normal random variable constructed by a standard normal random variable Z (as in Equation 4.2) with a specific mean μ and variance σ_{db}^2 (as in Equation 4.3). μ is derived from a uniform random number generator among all activation points on the outermost circle (see Figure 3.8). Based on the '3-sigma rule', the standard deviation σ_{db} is obtained from the selection probability σ_{GA} and the mean μ (as in Equation 4.1). If N represents the maximum number of activation points on the outer most circle, which is 103 in this work (see Figure 3.8), the activation point σ of a flight is generated by the following equations.

$$\sigma_{db} = \frac{\sigma_{GA} \times min\{N - \mu, \mu\}}{3} \tag{4.1}$$

$$Z = Gaussian[0 \quad 1] \tag{4.2}$$

$$\sigma = \sigma_{db} \times Z + \mu \tag{4.3}$$

We use the fitness proportionate selection (roulette-wheel) to select the aircraft type and runway, according to A_n and R_n respectively. To evaluate this chromosome, each representation needs to be transformed into a desired number (100 in this work) of flight plans, representing a set of valid real-world inputs to a simulation environment. This is done in a decoder through the use of aircraft performance parameters, the structure and characteristics of the airspace, and the airport configuration to generate flight plans in a traffic scenario.

Since we are studying a snapshot of the traffic situation in the integrated TeA system, the statistical nature (temporal and spatial distribution) of the arrival and departure traffic is modeled separately. Algorithm 4.1 demonstrates the procedure of the decoder for arrival flights and algorithm 4.2 demonstrates that for departure flights.

Algorithm 4.1 Pseudo Code of Decoder for Arrival Flights

Input:
T; σ_{GA}; μ; R_n; A_n; *ArrivalNetwork*;
Num: is the desired number for arrival flights;

ActivationTime = 60;

n = 0;

while (n < *Num*) **do**

  ActivationTime = ActivationTime + Poisson(T);

  while ($\sigma < 0$ or $\sigma > 103$) **do**

    $\sigma_{db} = \frac{\sigma_{GA} \times min\{N-\mu,\mu\}}{3}$
    $Z = Gaussian[0 \quad 1]$
    $ActivationPoint = \sigma_{db} \times Z + \mu$

  end while

  FlightType = RouletteSelection(A_n);
  FlightRunway = RouletteSelection(R_n);
  FlightTaxiway = AssignPath(*ArrivalNetwork*,FlightRunway);
  FlightGate = AssignPath(*ArrivalNetwork*,FlightRunway);

  n ++;

end while

4.2.2.2 Population Design of Ground Event Scenarios

An event population represents a set of event scenarios. Each scenario contains 10 constrained ground events. To encode an event scenario into a chromosome, we first develop an 'Event-Table' data structure. All the ground resources (runway, taxiway and gate) along with all the possible events that can be associated with them are included as illustrated in Table 4.1. Each combination of a surface resource with an event is given a unique event ID. For each resource, there are the following six

Algorithm 4.2 Pseudo Code of Decoder for Departure Flights

Input:
T; σ_{GA}; μ; R_n; A_n; *DepartureNetwork*;
Num: is the desired number for departure flights;

ActivationTime = 60;

n = 0;

while (n < *Num*) **do**

ActivationTime = ActivationTime + Poisson(T);

FlightType = RouletteSelection(A_n);
FlightRunway = RouletteSelection(R_n);
FlightTaxiway = AssignPath(*ArrivalNetwork*,FlightRunway);
FlightGate = AssignPath(*ArrivalNetwork*,FlightRunway);

n ++;

end while

Table 4.1: Event-Table

Event ID	*Event Location*	*Event Name*
1	Runway 16R	E1
2	Runway 16R	E2
3	Runway 16R	E3
...	...	...
377	Gate 40	E5
378	Gate 40	E6

possible events:

- E1: Resource unavailable for heavy and medium aircraft;
- E2: Resource unavailable for heavy and light aircraft;
- E3: Resource unavailable for heavy aircraft;
- E4: Resource unavailable for light aircraft;
- E5: Resource unavailable for medium aircraft; and
- E6: Resource unavailable for medium and light aircraft.

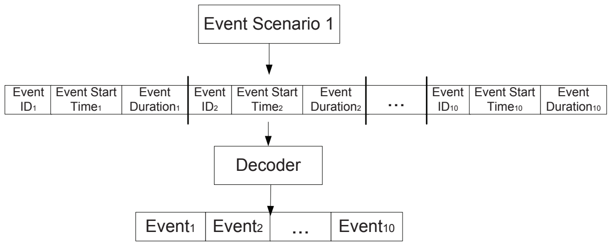


Figure 4.3: Event chromosome design showing genomes which encode event ID, event activation time and duration

Each individual (scenario) in the event population has 10 chromosome-blocks each representing for 1 event, and each chromosome-block has 3 genes as illustrated in Figure 4.3. The first gene in the chromosome-block is the event-ID which is randomly sampled in the interval [0, 1]. Its value is then used in the decoding process to select the event-ID value from the event table. The second gene is the event-start-time which represents the activation time of this event. The third gene is event-duration-time which is a duration for which the event will be active. The event-start-time and event-duration-time are both randomly sampled in the interval [0, 1] and then be used in the decoder to assign a real value in a pre-defined time interval. Algorithm 4.3 demonstrates the procedure of the decoder for ground events.

As being illustrated in Figure 4.1, each event individual and each flight individual has to be integrated to be a combined individual. This individual is then decoded to 10 events and 100 flight plans which will be feed into the simulator for evaluation. Thus for any flight scenario (100 flights) decoded from one flight individual is related with one event scenario (10 events) decoded from one event individual.

4.2.3 Fitness Function Design

The individual evaluation is based on the total number of delays for all flights, both arrivals and departures which is a measure of the induced delay due to the unavailability of air and ground resources. Since we attempt to discover the vulnerability of a TeA system, the co-evolutionary process seeks to maximize the total

Algorithm 4.3 Pseudo Code of Decoder for Ground Events

Input:
Num: is the number of chromosome blocks;
ID: is the value of the event-ID gene;
ST: is the value of the event-start-time gene;
IntervalST: is the pre-defined time interval for event-start-time;
DT: is the value of the event-duration-time gene;
IntervalDT: is the pre-defined time interval for event-duration-time;
Event-Table: contains event ID, event location and event name;

n = 0;

while (n < Num) **do**

    Length = LengthOf*Event-Table*;
    Index = *ID* × Length;
    EventLocation = *Event-Table*(Index);
    EventName = *Event-Table*(Index);
    EventStartTime = *ST* × *IntervalST*;
    EventDurationTime = *DT* × *IntervalDT*;

    n ++;

end while

number of delays in the system by evolving combinations of air traffic distribution and ground events which rank more highly (generated a higher number of delays) in the co-evolutionary process.

A total arrival flight delay in a TeA system is defined as: the difference between the ETA of a flight from its OM to the requested gate and its actual time of arrival (ATA) at the actual gate from its respective OM (as in Equation 4.4).

A total departure flight delay in a TeA system is defined as: the difference between the ETA of a flight from its requested gate to the OM and its ATA to the assigned OM from its respective gate (see Equation 4.5).

The objective - flight delay is then calculated by averaging the summation of the total numbers of arrival and departure flight delays by the number of flights in a given scenario (see Equation 4.6).

$$TotalArrivalFlightDelay = [\Sigma_{i=1}^{M}(ATA_i - ETA_i)]_G^{OM} \tag{4.4}$$

$$TotalDepartureFlightDelay = [\Sigma_{i=1}^{N}(ATA_i - ETA_i)]_{OM}^{G} \tag{4.5}$$

$$Delay = \frac{\alpha TotalArrivalFlightDelay + (1-\alpha)TotalDepartureFlightDelay}{M+N} \tag{4.6}$$

where

- M = number of arrivals;
- N = number of departures;
- ATA = actual time of arrival;
- ETA = estimated time of arrival;
- OM = outer marker;
- G = gate; and
- α = weight factor used for arrival- and departure-related delays $\alpha \in (0, 1)$; in this work, we make $\alpha = 0.5$.

4.3 Experimental Design

4.3.1 Experimental Scenarios and Parameters

Though we use the Sydney airport as the base for site-model, our methodology will be able to evaluate any advanced ATM concepts in the TeA. To accomplish the purpose of identifying the vulnerability of the integrated TeA system and compare the performance of dynamic CDA operations over fixed STARs and SIDs, we designed the 2x3 repeated-measures Table 6.1 which yields the following six different experimental scenarios.

Table 4.2: Experimental Design

	Fixed STARs & SIDs	*Dynamic CDA*
Arrivals Only	**Fixed Arrivals**	**dCDA Arrivals**
Arrivals & Departures	**Fixed Mixed**	**dCDA Mixed**
Departures Only	**Fixed Departures**	**dCDA Departures**

- *Scenario **Fixed Arrivals*** consists of 100 arrival flights with fixed STARs as there are no departures.
- *Scenario **Fixed Departures*** consists of 100 departure flights with fixed SIDs as there are no arrivals.
- *Scenario **Fixed Mixed*** consists of 50 arrival flights and 50 departure flights with fixed STARs and SIDs.
- *Scenario **dCDA Arrivals*** consists of 100 arrival flights using the dynamic CDA as there are no departures.
- *Scenario **dCDA Departures*** consists of 100 departure flights using dynamic CAD (derived from the dynamic CDA) as there are no arrivals.
- *Scenario **dCDA Mixed*** consists of 50 arrival flights using the dynamic CDA and 50 departure flights using the dynamic CAD (derived from dynamic CDA).

The two populations (traffic scenarios and event scenarios) co-evolve co-operatively and are represented by fixed-length real-valued genomes. The traffic scenario population size is 24. Each scenario consists of 100 flights in a time window of about 3 hours, either landing at, or taking off from, Sydney Airport. The pre-defined time interval for activation-time-control (temporal distribution) parameter, T, is $[45, 120]$ seconds which means that all 100 flights are processed in about 3 hours. Parameter μ, which controls the flight's spatial distribution, is uniformly initialized around all activation points on the outermost circle (as shown in Figure 3.8) which implies that flights can come from any direction. The event scenario population size is also 24 and each scenario consists of 10 events. The pre-defined time interval for the event-Start-Time is $[0, 12000]$ seconds and that for the event-duration-time is $[0, 1800]$ which means each event can be encountered by all 100 flights and solved in 30 minutes.

We run each of the 6 experimental scenarios 30 times using different seeds and apply tournament selection by elitism, single-point crossover with a probability of 1.0 and uniform mutation with a probability of 0.3. These parameters are chosen carefully after a number of sample runs to make sure the methodology work well and the best solutions do not change significantly in a reasonable time. We allow a sufficient number of objective evaluations in each run for its evolution to become stable (the best solution does not change significantly).

4.3.2 Measures and Metrics

4.3.2.1 Worst Case Analysis

Co-operative co-evolution is used as the search methodology for evolving complex scenarios through incremental feedback from the simulation system. Firstly, we evaluate whether this evolution procedure works efficiently. To explore the system's vulnerability, our approach is to analyze the scenarios according to the worst case, that is, those with the highest delays. After ranking the fitness values of all individuals in each population, we record the fitness data of the individual which has the most significant overall delay. Then, we analyze the evolutionary progress of this individual over all generations.

4.3.2.2 Efficiency of Dynamic CDA Model

As an advanced ATM procedure, the dynamic CDA is supposed to benefit the TeA system through better management, sustained and improved system capacity and minimized delays. However, if this new procedure is developed and assessed focusing on only the transition airspace without considering its interaction with ground resources/events, one cannot be sure the TeA system will derive maximum benefits from it in the real-world. By analyzing the highest total number of flight delays at the end of an evolution (the same data recorded from the previous metric), we determine how much real improvement can be achieved through implementing the dynamic CDA in the worst case.

4.3.2.3 Air Traffic Flow Constraints

To evaluate the ground-air network vulnerability, system flow constraints and their causalities in the arrival and departure processes, which are primarily responsible for generating TeA system inefficiencies and delays, need to be identified.

First, we look at how the dynamics of constrained ground resources can affect system capacity by causing higher number of flight delays (**total delay analysis**). The goal of this analysis is to identify the airport ground component's bottlenecks which could cause higher TeA system delays. We record the number of total flight delays by combining the TeA airspace and ground delays induced by dynamic ground events happening at each ground resource.

Since the complex TeA system is characterized by being highly structured with variations and involving multiple interactions among many different components, the major delays and inefficiencies observed due to some downstream resources probably propagate back and block the traffic flow at some upstream resources. For this purpose, we analyze all the system's resources, including those of the TeA airspace and ground, to ascertain the frequencies of holding delays caused by each (**holding delay analysis**).

In the above two analyses, the reason for a total delay/holding delay may be the unavailability of a set of resources; for example, if we assume an aircraft is about to be assigned to '*Runway 07, Taxiway G, Taxiway B2, Gate 3*' at time '*T1, T2, T3, T4*', but *Taxiway B2* and *Gate 3* are blocked at *T3* and *T4*, to avoid overstating system delays, in this work, we record only the first resource – *Taxiway B2* – as being the one which induces the delay.

4.3.2.4 Scenario Patterns with Higher Delays

In order to derive maximum benefits from the advanced dynamic CDA procedures and balance the capacity-demand of flight traffic with constrained surface operations, we use a heuristic methodology to search in a large scenario's search space to find complex traffic patterns and constrained ground events which can cause higher number of TeA delays in the presence of the advanced ATM procedures. This analysis aims to demonstrate the relationships implicit in the scenario patterns that lead to the higher delays.

Firstly, we look at how the spatial and temporal distributions affect the system's capacity by analyzing the spatial and temporal distribution parameters of the best solutions in the last generations of each experimental scenario. Then, we calculate the frequencies of each event appearing in the best solutions for each trial to look at the constrained ground events which can cause higher TeA delays.

4.4 Results and Discussion

4.4.1 Worst Case Analysis

Figure 4.4, 4.5 and 4.6 show the progress in the total delays induced by the best rerouting solutions over 100 generations averaged over all seeds. Despite some fluctuations due to the inherent stochastic nature of EAs and the problem, the fitness of the best solution does not change significantly. These figures reveal that, for all experimental scenarios, co-operative co-evolution is capable of evolving ground events for resources which can maximize delays.

4.4.2 Efficiency of Dynamic CDA Model

In Figure 4.4, 4.5 and 4.6, we can see the fitness values of the best individuals which have the highest total number of delays at the end of the evolutions for all experimental scenarios. We note that scenarios with mixed traffic (50 arrivals

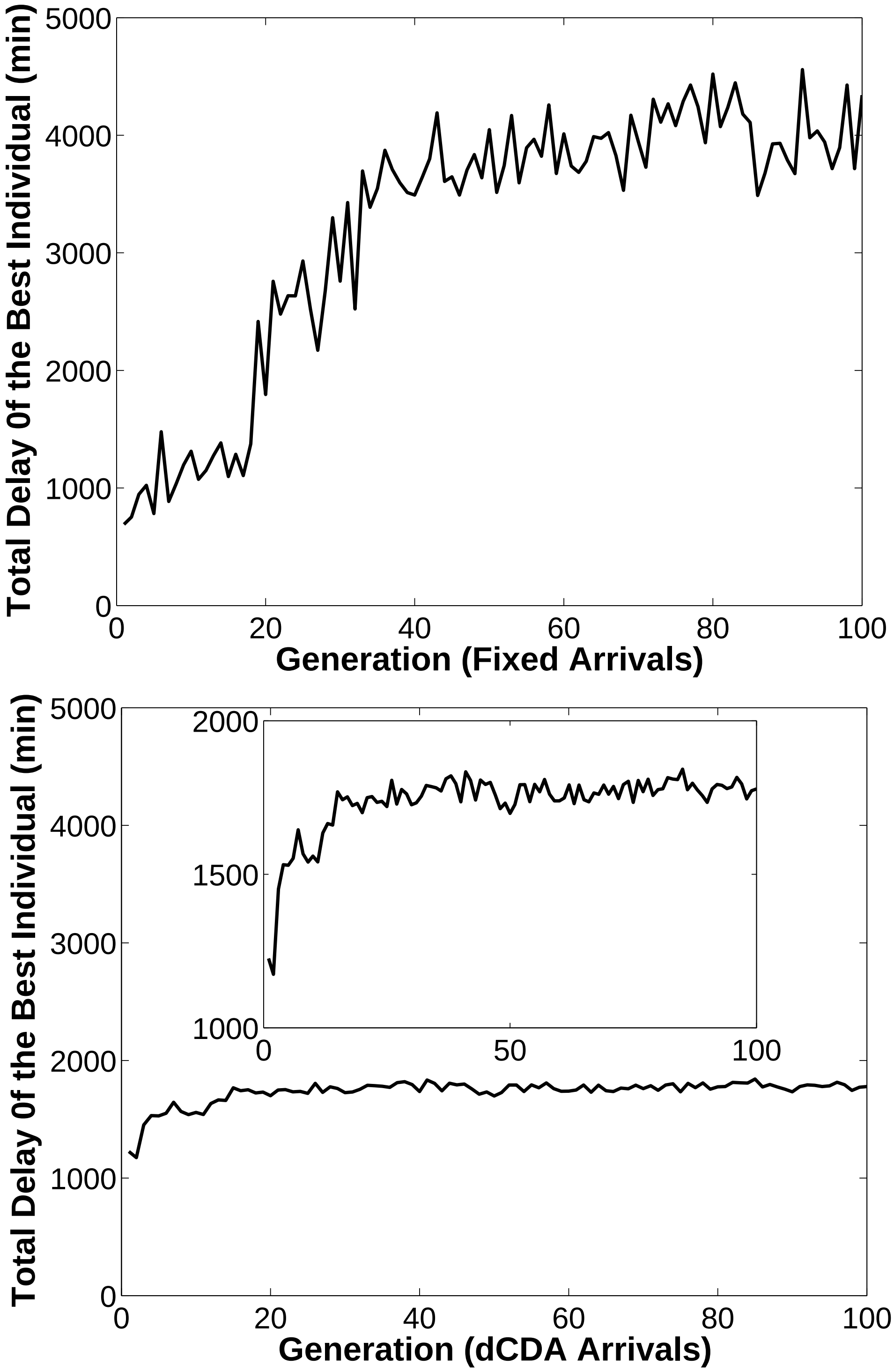


Figure 4.4: Delays induced by best individuals for *Arrivals Only* in each generation averaged over 30 seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

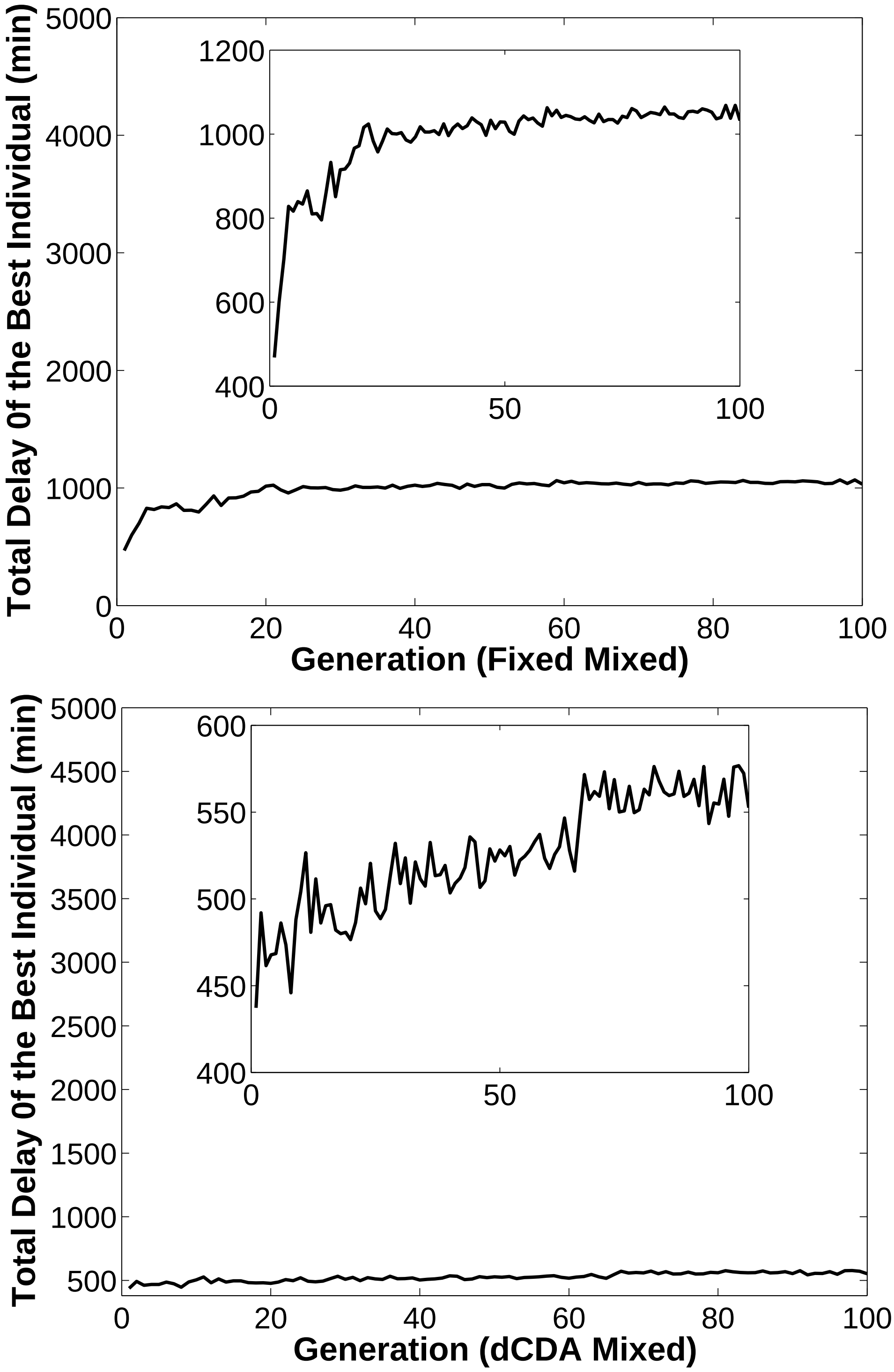


Figure 4.5: Delays induced by best individuals for *Mixed Traffic* in each generation averaged over 30 seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

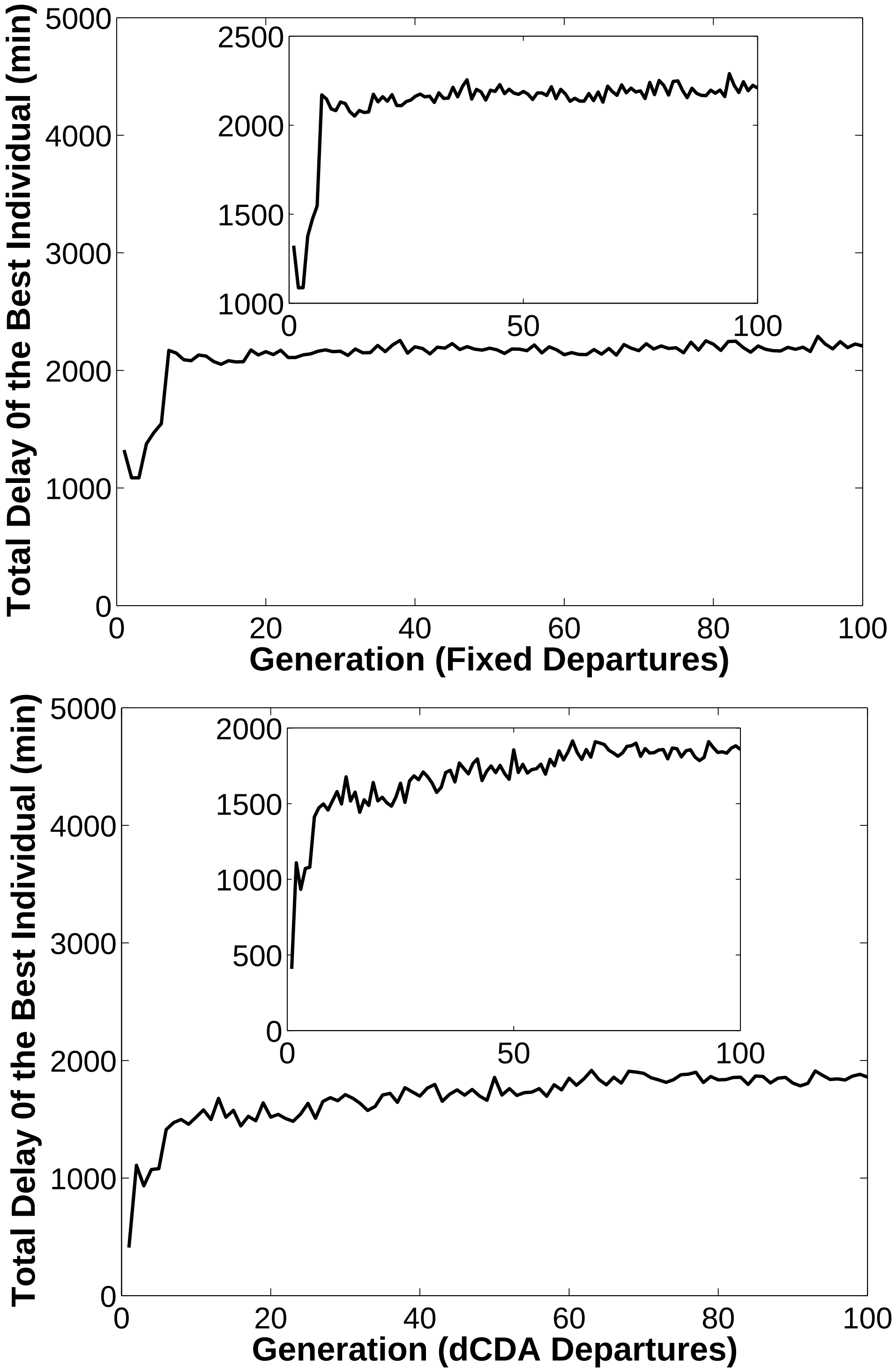


Figure 4.6: Delays induced by best individuals for *Departures Only* in each generation averaged over 30 seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

and 50 departures) have lower flight delays than those with arrivals only (100) or departures only (100). The reason is that, given the same time interval, 'arrivals only' or 'departures only' over-stress the arrival or departure sub-systems compared with 'mixed traffic' which can make use of the integration between them. However in the dynamic CDA approach, arrivals and departures sharing the same airspace model makes stress on this sub-system unreasonable. Our hypothesis is that different scenarios cause different bottlenecks in the sub-systems of TeA system. Hence the stress on airspace model in dynamic CDA is stressing on different sets of bottlenecks. This hypothesis will be inspected by results in next section.

As can be seen, the best fitness values (highest overall flight delays) for all cases using the dynamic CDA are significantly lower than for all those using the fixed STARs and SIDs. This reveals that, for those scenarios which involve the dynamically constrained ground resources, the new procedures – the dynamic CDA – have considerable advantage over the existing operations – the fixed STARs and SIDs. The main reason for this advantage is that, despite using the same ground model and dealing with the same ground events, air route computations for flights in the dynamic CDA scenarios are processed by searching in a much larger space ($104 + 83 + 62 + 41 + 20 = 310$ artificial way points, as shown in Figure 3.1) than the STAR and SID way points in the fixed STARs and SIDs scenarios ($18 + 15 = 32$ STAR and SID way points, as shown in figures 3.4 and 3.5). In other words, the greater flexibility of the dynamic CDA procedures is capable of improving a TeA system's efficiency.

Overall, the results provide support for the shift from current operations of controlling aircraft along fixed route structures towards the more dynamic and flexible management of flight trajectories with fewer ATC restrictions – the dynamic CDA. It could address the local demand-capacity imbalance by providing the flexibility in adjusting traffic flows in response to changing conditions and enable an air traffic system to cope more effectively with local disruptions, such as constrained ground events caused by unexpected bad weather.

4.4.3 Major Air Traffic Flow Constraints

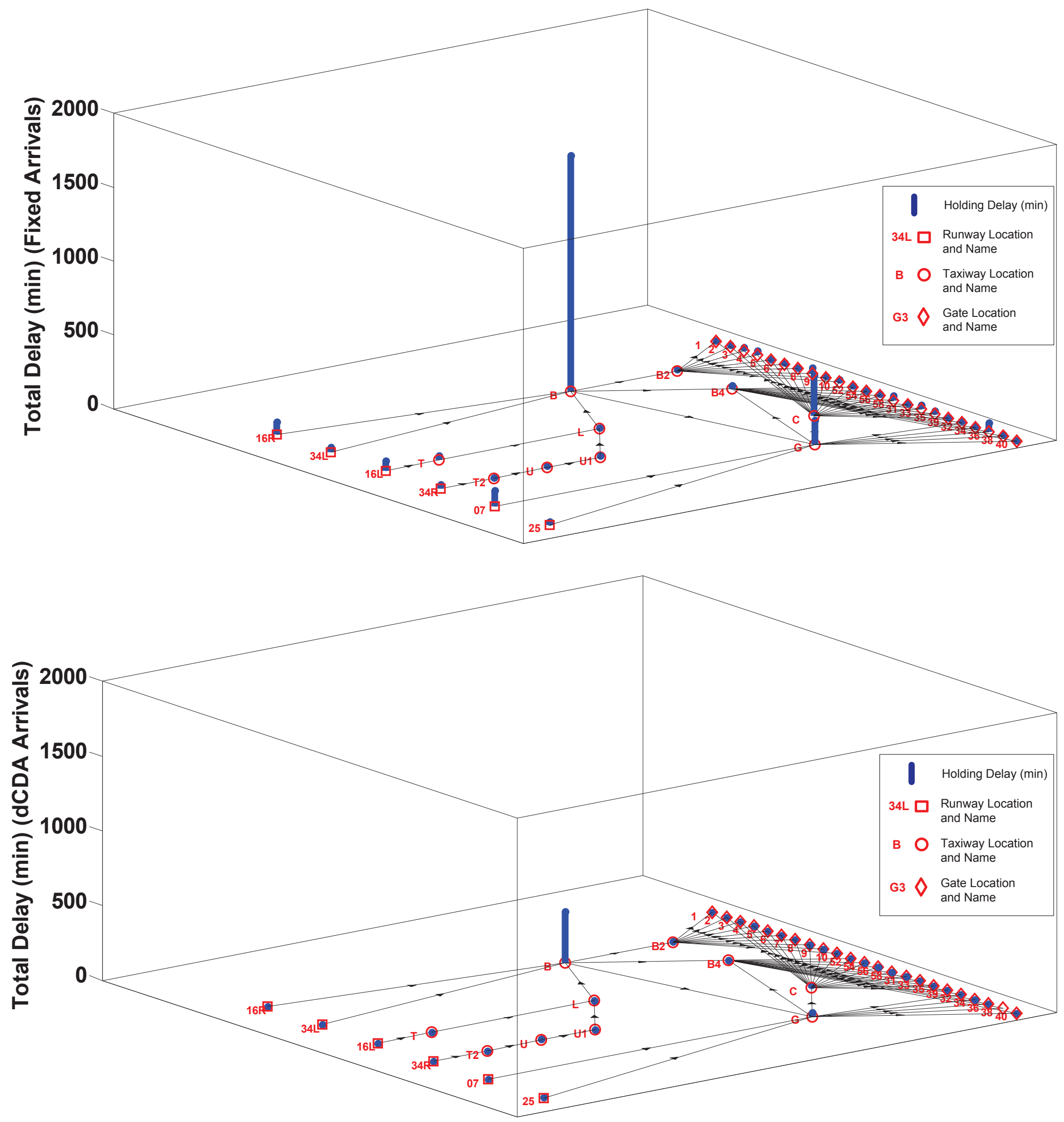


Figure 4.7: Total flight delays caused by individual ground resources in best solutions (individuals in last generation) for *Arrivals Only* averaged over all seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

The total number of flight delays combining TeA airspace delays and ground delays induced by dynamic ground events are recorded and presented in Figure 4.7, 4.8 and 4.9. The network on the surface represents the ground model used in each experimental scenario; for instance, as the top graph in Figure 4.7 presents results from the 'fixed arrivals' scenario, the network on its surface is the arrival ground model. The data is collected from the best solutions (individuals in the last

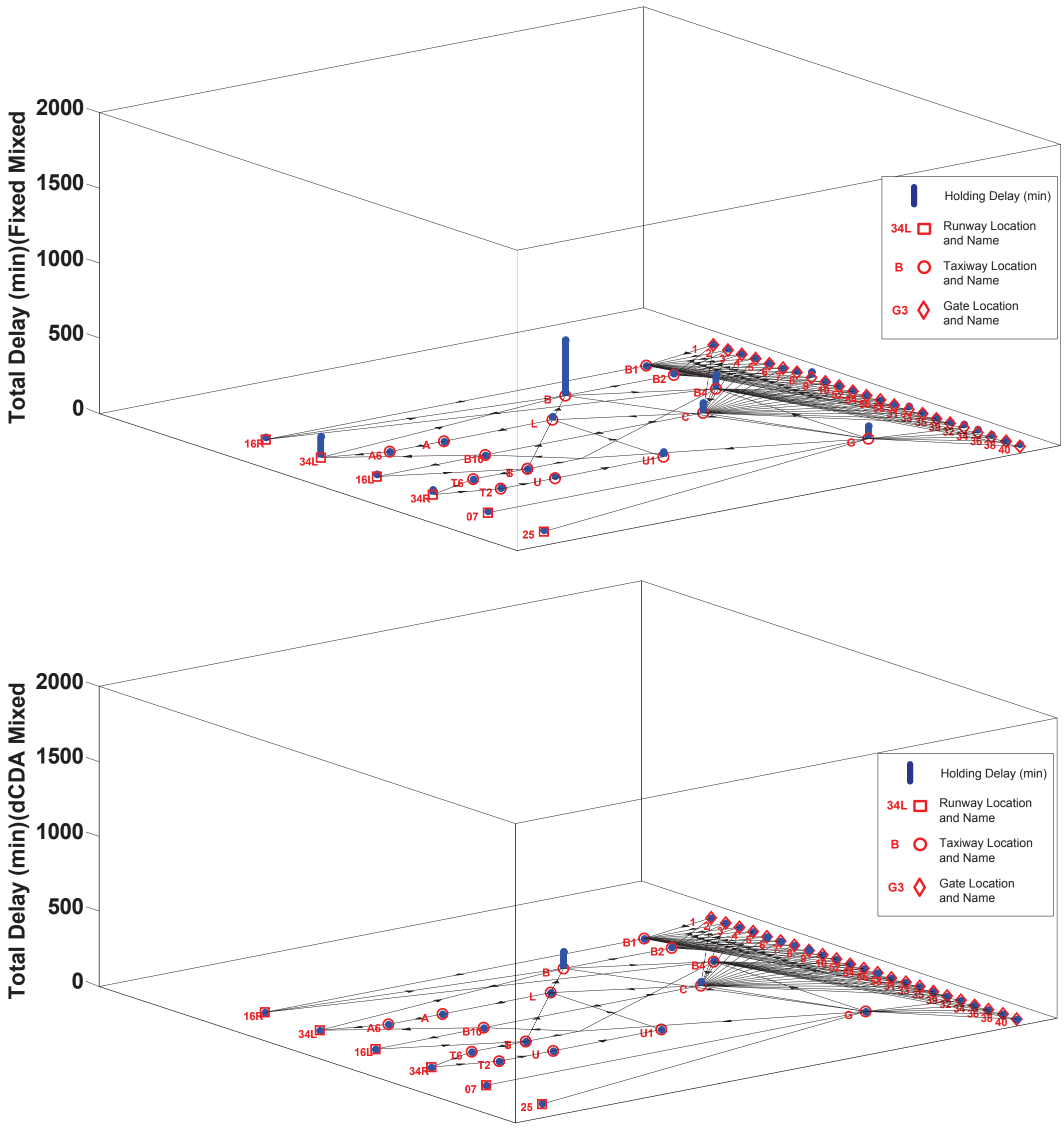


Figure 4.8: Total flight delays caused by individual ground resources in best solutions (individuals in last generation) for *Mixed Traffic* averaged over all seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

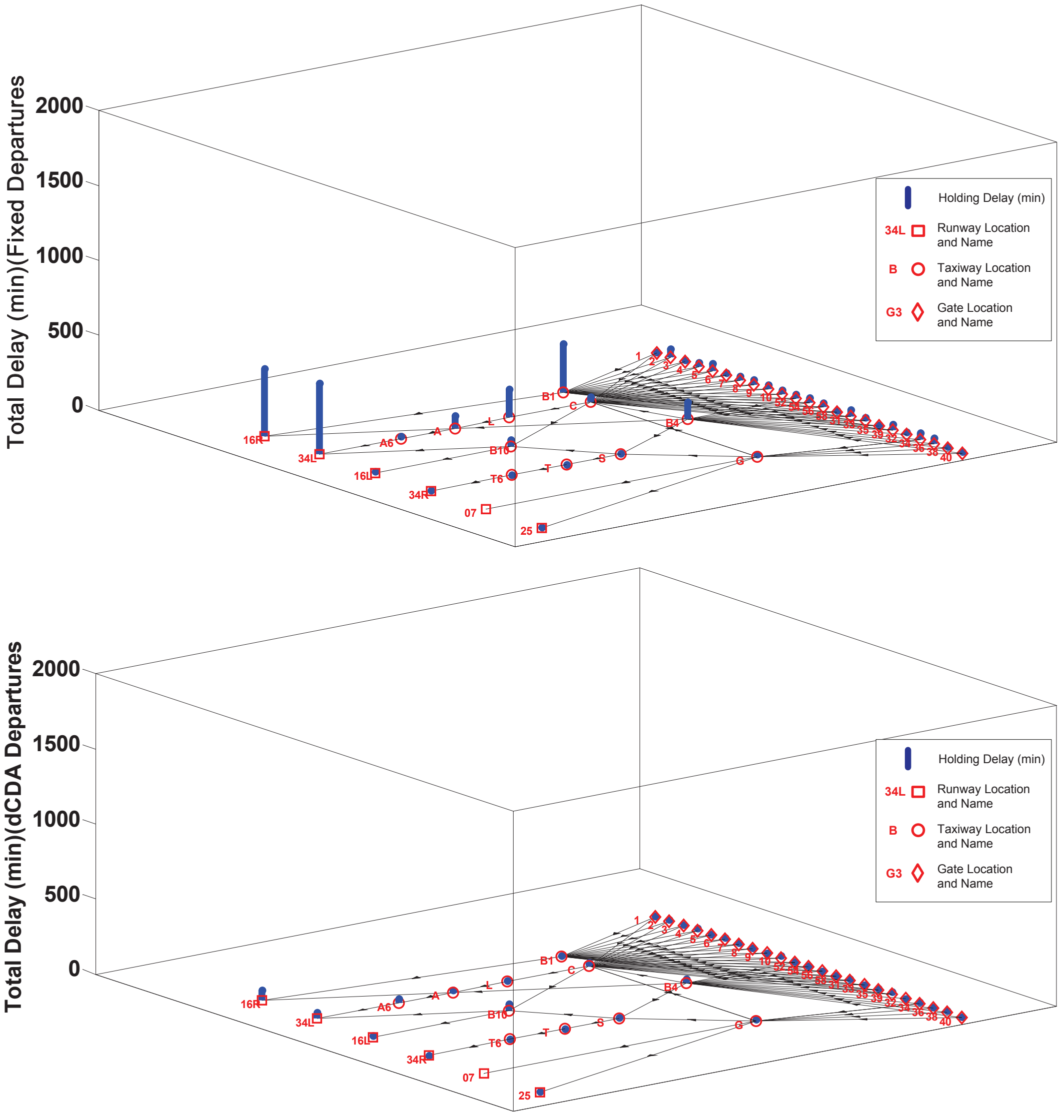


Figure 4.9: Total flight delays caused by individual ground resources in best solutions (individuals in last generation) for *Departures Only* averaged over all seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

generation) averaged over all seeds. It can be seen that the ground resources which encounter disturbance and contribute to the highest total number of flight delays vary from trial to trial, which proves our earlier hypothesis in Section 4.4.2: different scenarios cause different bottlenecks in the sub-systems of TeA system.

In the fixed STARs and SIDs case, the bottlenecks with 'arrivals only' are taxiways *B* and *G*, and runway *07*, with 'mixed traffic' taxiways *B* and *B4* and runway *34L*, and with 'departures only' runways *16R* and *34L* and taxiway *B1*. For the dynamic CDA case, the bottlenecks with 'arrivals only' are taxiways *B*, *G* and *C*, with 'mixed traffic' taxiways *B*, *C* and *B4*, and with 'departures only' runways *16R* and *34L* and taxiway *B10*. These bottlenecks represent vulnerabilities of the airport ground component. Disturbances occurring in these resources are more likely to cause TeA system congestion and affect its efficiency. We note that taxiway B contributes a large number of delays in most experimental scenarios due mainly to it being a critical node resource in both the arrival and departure ground model networks (Figure 3.6 and 3.7 respectively).

Figure 4.10, 4.11 and 4.12 show the holding delays caused by each resource of individuals in the last (100th) generation. Apart from the ground resource model, the air-side model (CDA model, STAR network, SID network) used in each experimental scenario is projected on the surface; for instance, as the top graph in Figure 4.10 presents results from the 'fixed arrivals' scenario, the network on its surface is the STAR and arrival ground network model. We can see that the bottleneck resources contributing to the highest number of holding delays are different from those obtained when we focused on the constrained ground resources independently as shown in Figure 4.7, 4.8 and 4.9.

As shown in Figure 4.10, 4.11 and 4.12, when ATM operations in the TeA airspace have insufficient flexibility using fixed terminal routes, taxiway B contributes major delays when processing arrivals only since it is a critical node resource in the arrival ground model network (Figure 3.6). The majority of delays are absorbed at the gates for both departures only and mixed traffic, one of the main reasons being that a departing aircraft occupies its designated gate once it

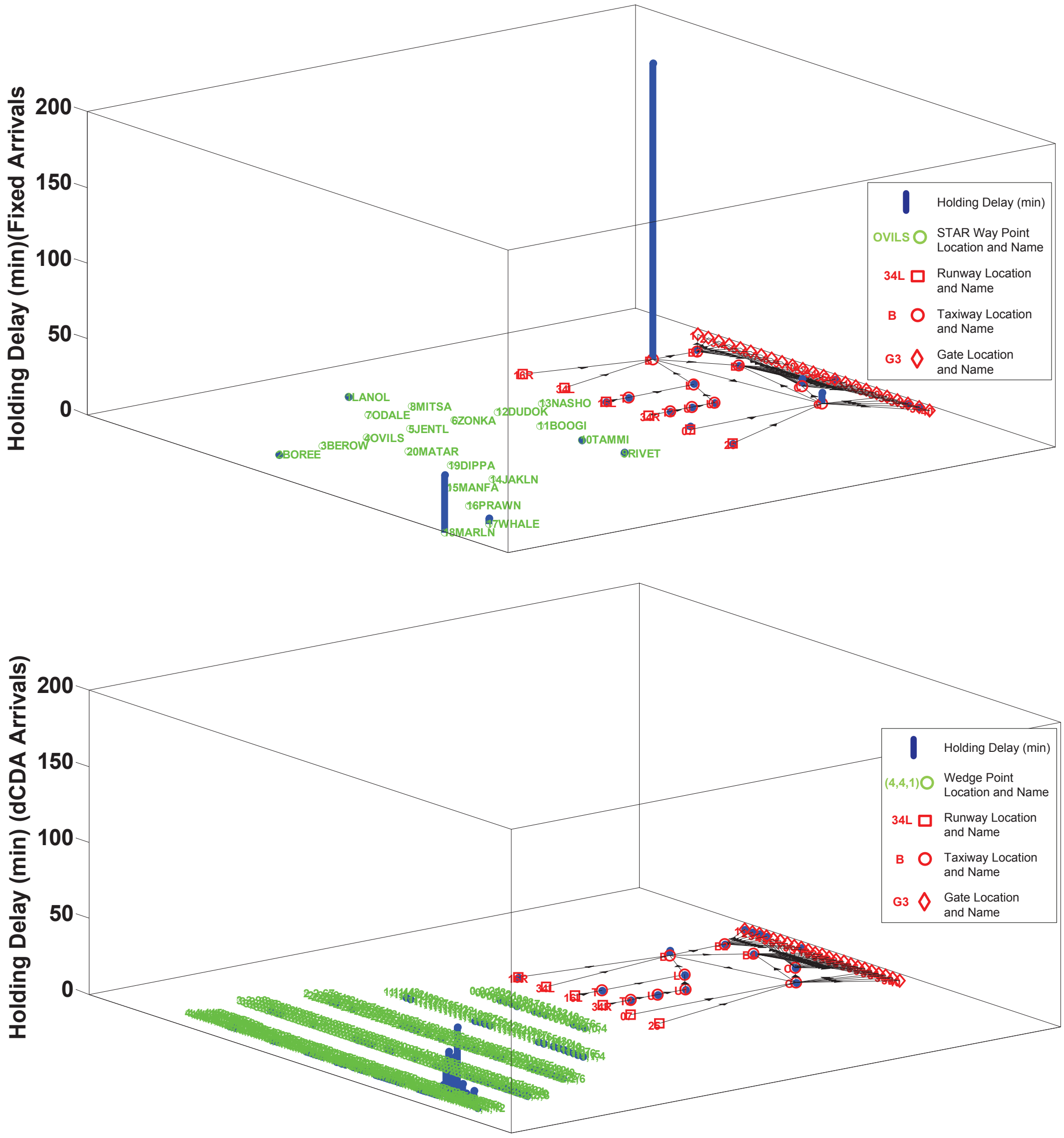


Figure 4.10: Holding delays caused by each air-side and ground-side resource in best solutions (individuals in last generation) for *Arrivals Only* averaged over all seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

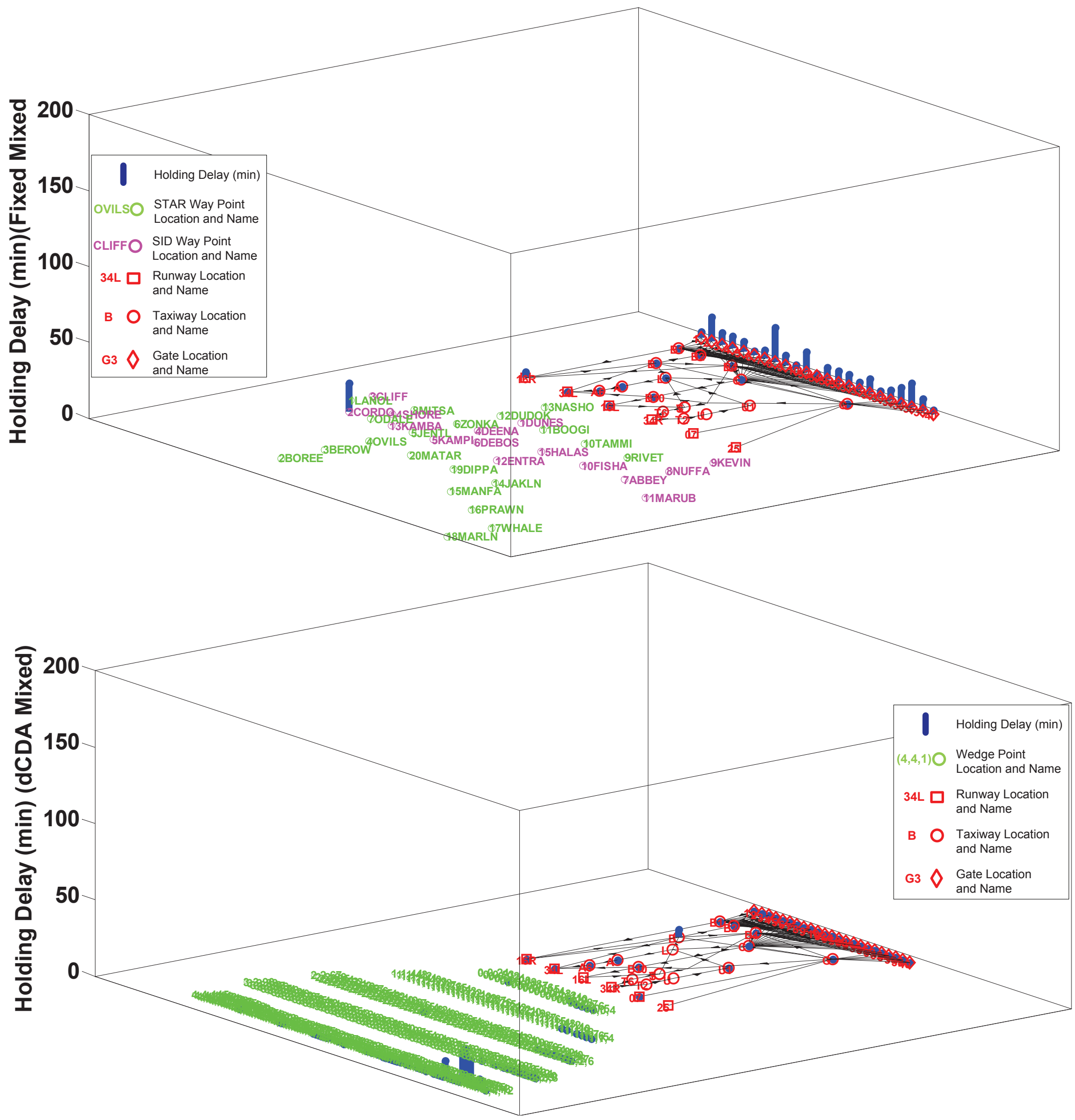


Figure 4.11: Holding delays caused by each air-side and ground-side resource in best solutions (individuals in last generation) for *Mixed Traffic* averaged over all seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

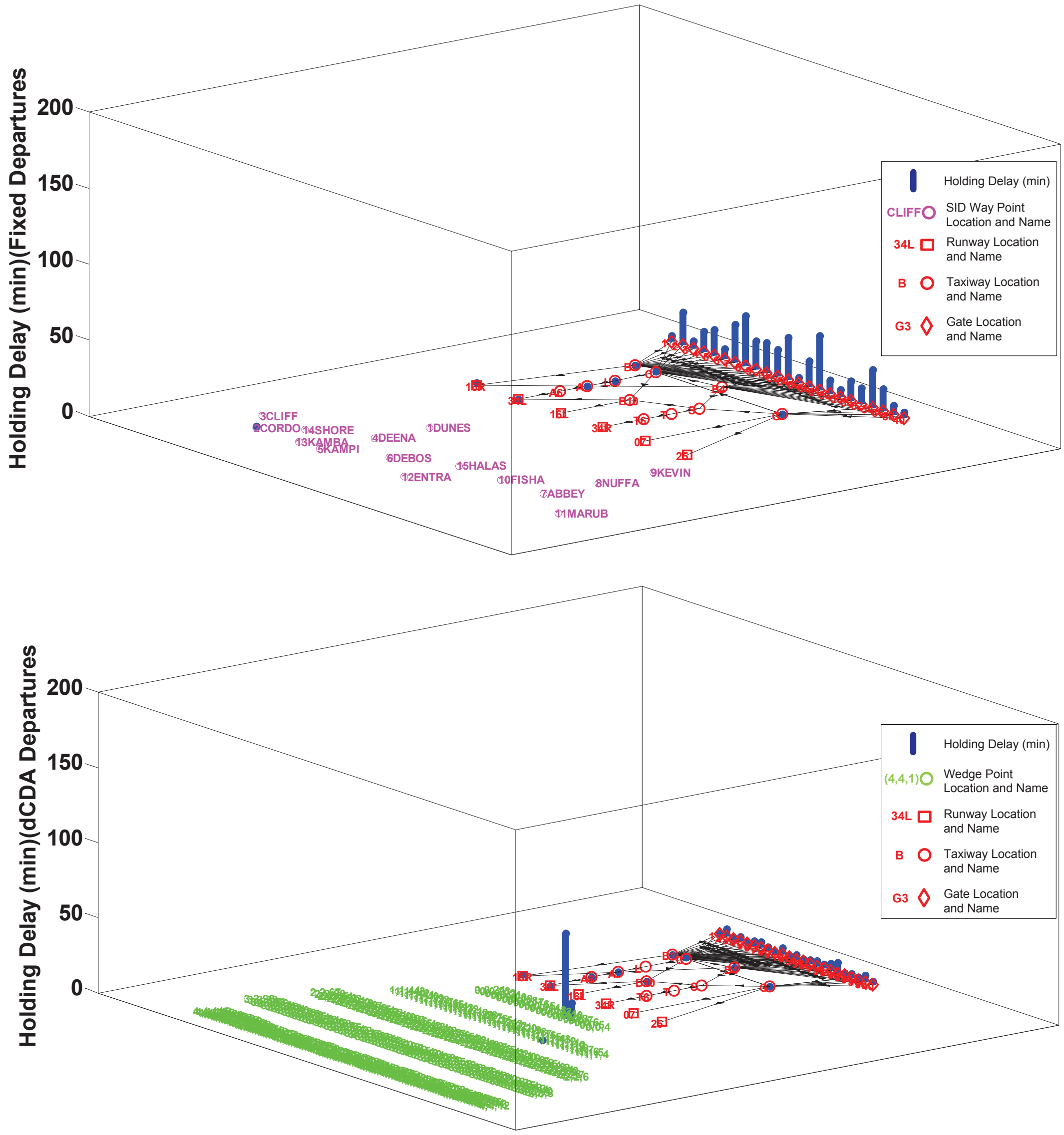


Figure 4.12: Holding delays caused by each air-side and ground-side resource in best solutions (individuals in last generation) for *Departures Only* averaged over all seeds (The figure on top for *fixed STARs/SIDs* and that on bottom for *dynamic CDA*)

Table 4.3: Top three constrained resources from total delay analysis

	Fixed STARs & SIDs	**Dynamic CDA**
Arrivals Only	B, G, 07	B, G, C
Arrivals & Departures	B, 34L, B4	B, C, B4
Departures Only	16R, 34L, B1	16R, B10, 34L

Table 4.4: Top three constrained resources from holding delay analysis

	Fixed STARs & SIDs	**Dynamic CDA**
Arrivals Only	B, MARLN, G	(4, 22), (4, 19), (4, 20)
Arrivals & Departures	G8, CORDO, G2	(4, 19), (4, 20), (4, 17)
Departures Only	G31, G8, G54	(0, 8), (0, 7), G39

becomes active. Therefore, this gate remains unavailable until an acceptable ground route starting from this gate is assigned to this aircraft. When the TeA airspace has greater flexibility using the dynamic CDA, the air-side resources absorb more delays than the ground resources and reduce system delays.

For comparison, a summary of the top three major constrained resources of each experimental scenario in each analysis is presented in Table 4.3 and 4.4. One interesting observation from these tables is that, when implementing interactions between arrivals and departures, the ground bottleneck resources found previously in the dynamic CDA case are taxiways *B*, *C* and *B4* while those of their counterparts are wedge points (4,19), (4,20) and (4,17) which shift totally to resources in the transition airspace. However, in the fixed STARs and SIDs case, the shifting is not that thorough. The reason for this is that, under the STARs and SIDs procedures, the arrival and departure paths are procedurally separated, unlike under the dynamic CDA procedures in which arrivals and departures share and compete for the same resources (the same set of artificial wedge points) in the transition airspace. Therefore, heavy interactions between arrival and departure flows are also identified in the terminal area in the dynamic CDA scenarios.

One of the main insights gained from the comparison of the above two analyses is the highly interactive nature of a TeA system's components which indicates that, when attempting to solve ATM issues, one cannot separate the air-side complex from the ground-side complex, or divide the arrival process from the departure process. Therefore, advanced ATM procedures should not be evaluated as stand-alone tools that deal with each aspect without considering interactions from other parts of the environment in which the system operates; this is one of the main problems with current automation tools. Rather, newly developed technologies should be integrated, in either a central or distributed fashion, in order to better manage, sustain and improve a TeA system's capacity and minimize delays.

Another interesting finding is that the shifting of major constrained resources reveals the potential transfer of major delays in each system component – runways, taxiway, gates, TeA airspace; for instance, gates are obviously a critical bottleneck for the 'fixed departures only' scenario in the holding delay analysis since three major constraint resources are gates *G31*, *G8* and *G54*. However, in the 'dCDA departures only' trial, only one of the top three major constraints is in the gates component while wedge points in the TeA airspace tend to cause more constraints in the system. To shed more light on this observation, we then look at the sources of constraints in the TeA system's components which could increase congestion. Figure 4.13 and 4.14 show the summarized number of total delays and holding delays for each system component – runways, taxiways, gates and TeA airspace from all trials.

The summarized results from the total delay analysis explore how the dynamics of constrained ground resources affect the total number of flight delays which represent the TeA system's efficiency. Figure 4.13 shows that, under either the fixed STARs and SIDs or dynamic CDA conditions, the events happening in taxiways are the major source of delays as long as the airport is not processing only the departures. For departures only, the gates contribute to the majority of total delays in the TeA system under the dynamic CDA procedures while disturbances at the runways cause major system congestion under the fixed STARs and SIDs procedures. An observation gained from this comparison is that the current major congestion in Sydney Airport's TeA system (when arrivals and departures are processed con-

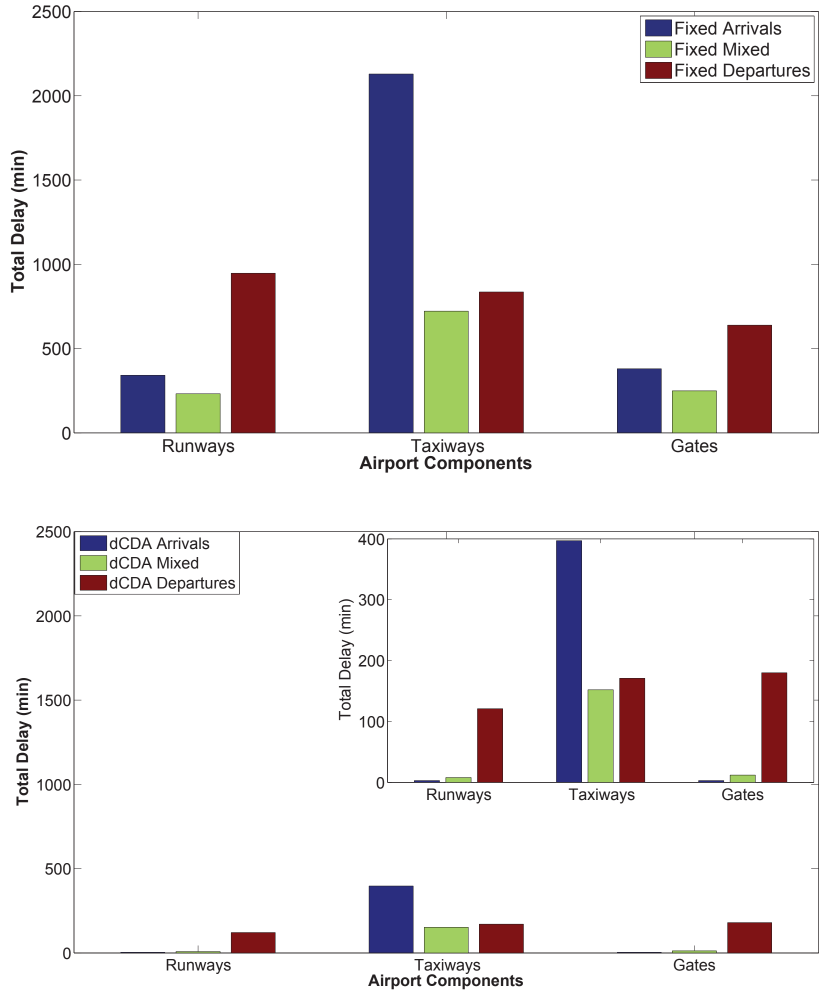


Figure 4.13: Summarized delays caused by each sub-system: runways, taxiways and gates (Figures depict data collected from *Total Delay Analysis*; the figure on top under the *fixed STARs/SIDs* condition and that on bottom under *dynamic CDA* procedures)

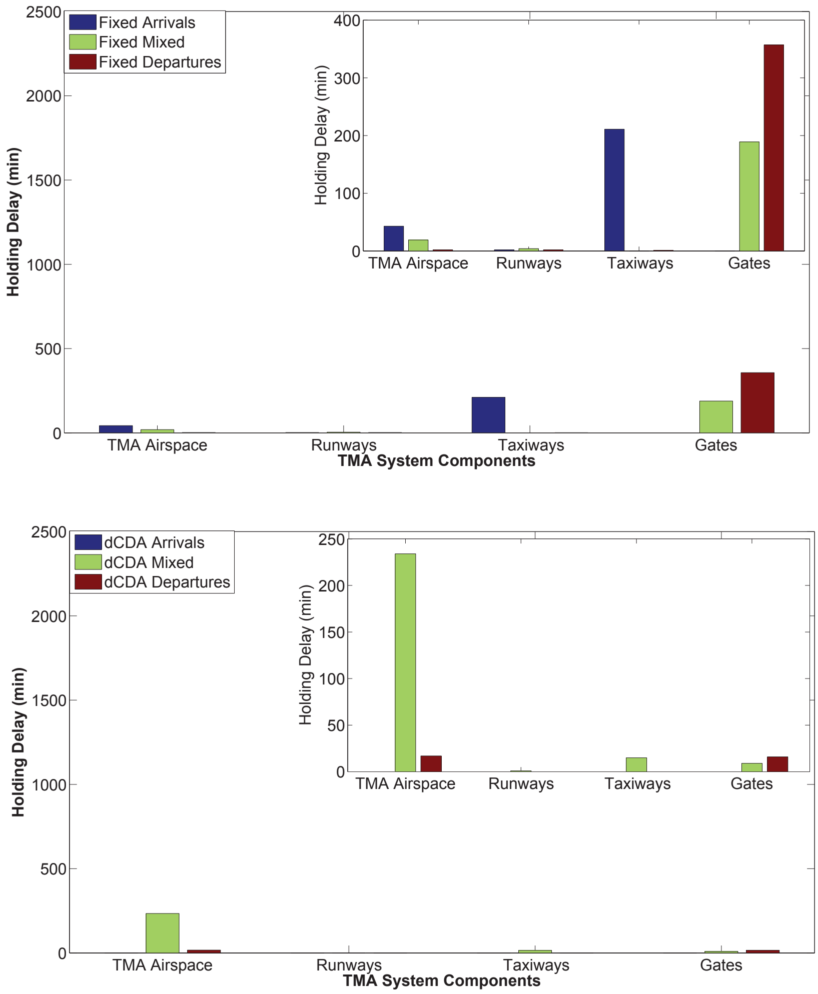


Figure 4.14: Summarized delays caused by each sub-system: runways, taxiways, gates and TeA airspace (Figures depict data collected from *Holding Delay Analysis*; the figure on top under the *fixed STARs/SIDs* condition and that on bottom under *dynamic CDA* procedures)

currently) is caused by dynamic ground events existing in the taxiways which will continue to be critically congested with the implementation of the dynamic CDA procedures in the future. The main reason for this is that the taxiways component contains some highly critical resources, as shown for taxiway B in Figure 4.7, 4.8 and 4.9.

The summarized results from the holding delay analysis explore the possible bottlenecks in the entire TeA system which are the main contributors to system congestion. Figure 4.14 reveals that, under the current TeA procedures, the gates cause most of the holding delays either when processing departures only or handling mixed traffic whereas, if there are only arrivals, the taxiways are the major contributors to holdings delays. On the other hand, under the dynamic CDA, of all the airport components, the taxiways are the bottleneck contributing most to holding delays when handling mixed traffic while the gates are the major constraint when there are departing flights only. We can see that, using the current ATC procedures, the traffic flow bottlenecks in Sydney Airport are the gates. However, if the dynamic CDA is adopted in the future, the taxiways would become the major bottleneck in terms of system capacity. We note that the dynamic CDA concepts will improve the TeA system's efficiency by absorbing delays from the airport's ground resources using a more flexible TeA airspace.

4.4.4 Scenario Patterns with Higher Delays

Figure 4.15 and 4.16 plot the spatial and temporal distribution parameters of the best solutions in the last generations of each scenario. While the inter-arrival times, activation points and distribution parameters are all uniformly initialized, at the end of the co-evolutionary run, the population of rerouting strategies converges to an inter-arrival time of $[45, 65]$ seconds, and the activation points to 20 ± 8 in the 'dynamic CDA arrivals' and 'dynamic CDA mixed' respectively. These parameter values produce higher numbers of delays for the given traffic scenario and associated events than earlier generations. The results show that the flights which have

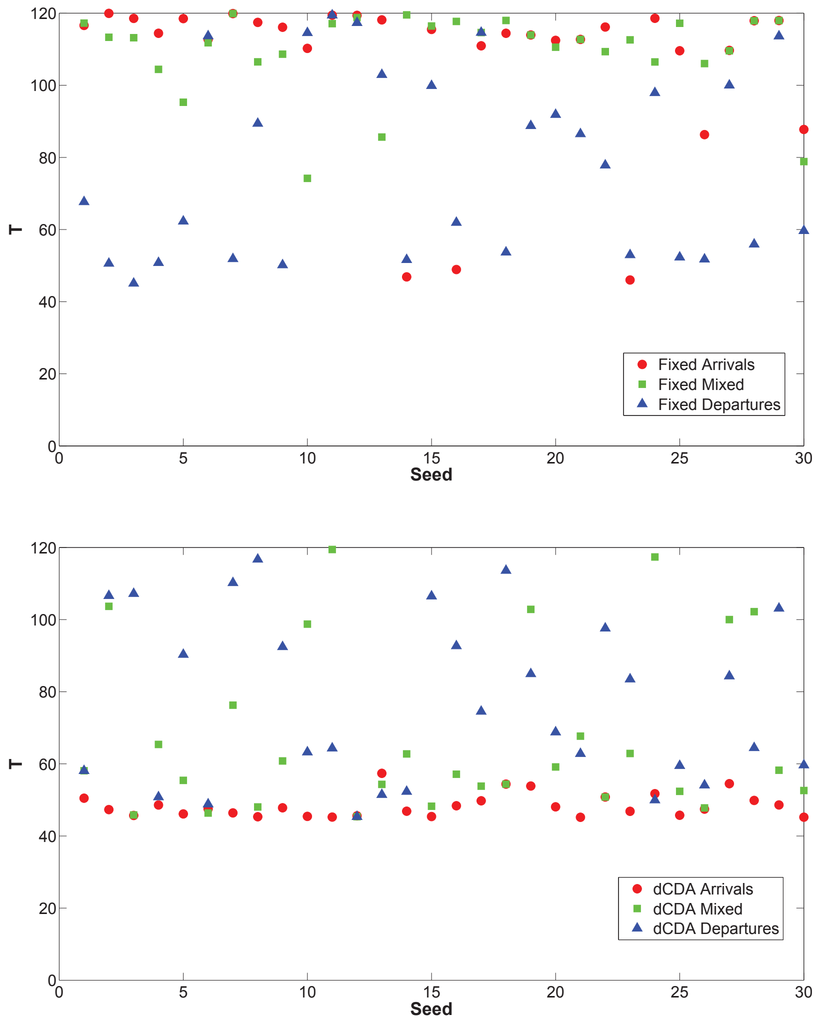


Figure 4.15: The temporal distribution parameter of best solutions in last generations of each scenario (The figure on top plot temporal distribution parameter T for *Fixed* scenario and the figure on bottom for *dCDA* scenario)

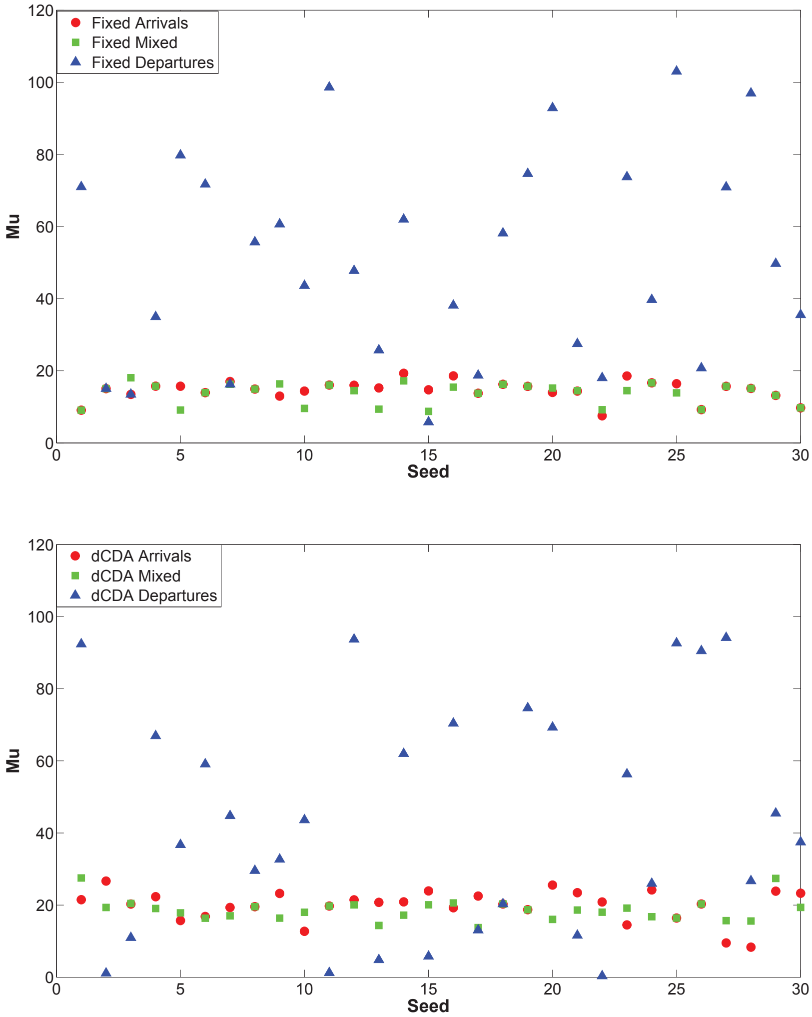


Figure 4.16: The spatial distribution parameter of best solutions in last generations of each scenario (The figure on top plot spatial distribution parameter μ for *Fixed* scenario and the figure on bottom for *dCDA* scenario)

average inter-arrival times of 50 seconds and which are spatially distributed around the boundary between the I quadrant (north-east) and II quadrant (south-east) (see Figure 3.8) of the search airspace can cause significant delays in the dynamic CDA scenarios when processing either arrival traffic only or mixed traffic. However under the current fixed STARs & SIDs operations, the inter-flight times of the best solutions for 'mixed traffic' and 'arrivals only' trials are 110±10 seconds and the activation points distributed mainly in the range of $[0, 20]$ respectively (I quadrant in Figure 3.8). Since spatial and temporal distribution parameters are not designed to work in departure traffic (see Section 4.2.2.1), both parameters do not converge for 'departures only' scenarios.

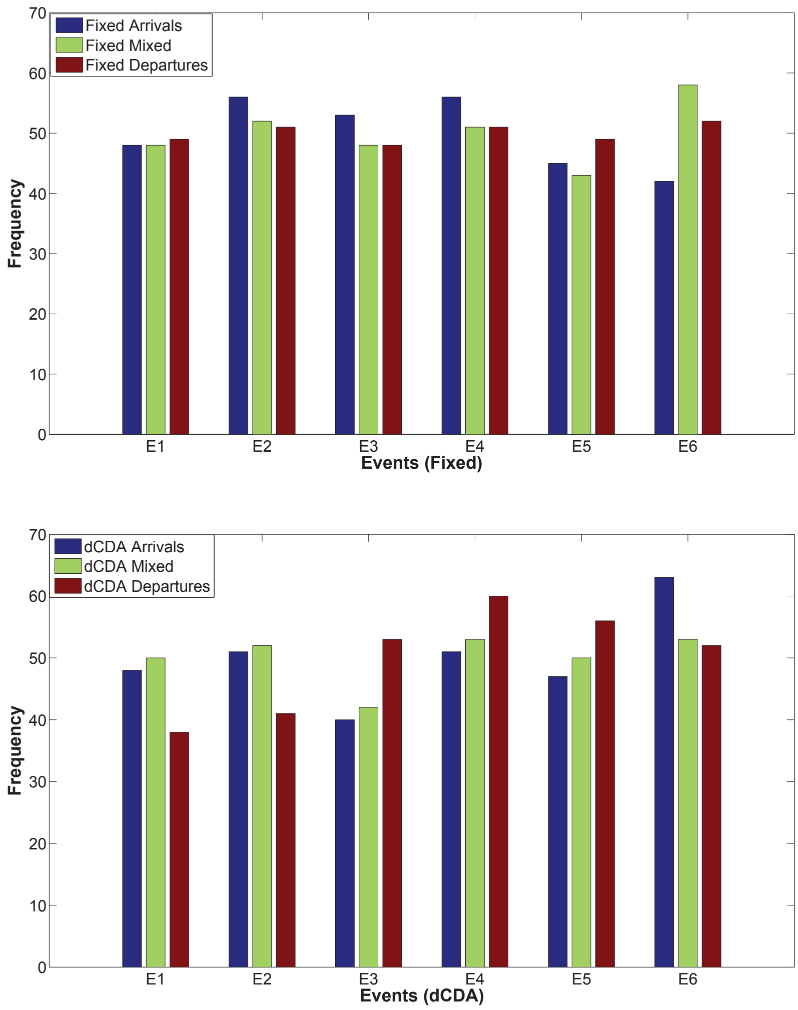


Figure 4.17: Events in best solutions at end of co-evolutionary runs

The calculated frequencies of each event appearing in the best solutions for different trials are shown in Figure 4.17. It reveals that, although the events are all uniformly initialized , at the end of the co-evolution, the population of rerouting strategies encounters more E6 (resource unavailable for medium and light aircrafts) in the 'dCDA arrivals' scenario, more E4 (resource unavailable for light aircraft) in the 'dCDA departures' scenario, and more E6 and E4 in the 'dCDA mixed' scenario. However, under the fixed STARs & SIDs procedures, E2 (resource unavailable for heavy and light aircraft) and E4 appear more frequently in 'arrivals only', and E6 more frequently in both 'mixed traffic' and 'departures only'. In short, events E6 and E4 are major contributors to delays under both ATM procedures.

4.5 Chapter Summary

In this chapter, a simulation-based co-evolutionary computational environment is developed for evaluating advanced TeA airspace concepts and understanding the TeA system vulnerabilities. A single-objective co-operative co-evolutionary algorithm was used as the search engine to evolve the reciprocal interactions of arrivals and departures using a shared ground-air network.

We conducted a series of computational experiments with different air traffic distributions (both spatial and temporal), ground resource constraints and TeA operational scenarios. The parameters impacting on the delay performances were co-evolved with our synthetic model of a TeA system (TeA airspace, runways, taxiways and gates). The results demonstrated the power of our methodology in evaluating vulnerabilities of the air-ground network with integrated arrival and departure operations.

Our analysis identified the bottlenecks of the integrated TeA system and synthesize an overall situational awareness picture that decision makers can utilize. For instance, with the fixed STARs and SIDs model, the traffic flow bottlenecks in Sydney Airport are the gates; however if the dynamic CDA model is adopted in the future, the taxiways would become the major bottleneck in terms of system capacity;

for mixed traffic with arrivals and departures, dynamic ground events which occur in the taxiways can cause significant delays using either the fixed STARs and SIDs or dynamic CDA concept. Our analysis suggested and demonstrated that, when making effort to solve ATM issues, one cannot separate the air-side complex from the ground-side complex, or divide the arrival process from the departure process.

The results indicated that, in the presence of constrained ground resources, the dynamic CDA model was able to provide controller and airspace user benefits to further improve the TeA system's throughput capacity as well as to minimize flight delays. The quantified performance evaluation increases decision maker's confidence to support this transition. The results also revealed that when the TeA airspace has greater flexibility using the dynamic CDA, the air-side resources absorb more delays than the ground resources and reduce system delays.

It is noted that some of conclusions are known, however their root causes are not. The objective of this work is to provide a generic methodology for evaluation and understanding, and while many conclusions may sound logical, the value of our proposed methodology is to provide a set of quantitative evidences.

Chapter 5

Multi-Objective Co-Operative Co-Evolution

The research question that needs to be solved in this chapter is: How to design an efficient multi-objective co-operative co-evolutionary algorithm which can co-evolve solutions towards the efficient set of trade-offs effectively, while maintaining diversity of the solution set? In the previous chapter, we presented a simulation-based co-evolutionary computational environment (CRRT) in order to evaluate advanced TeA airspace concepts and understand the TeA system vulnerabilities. Although the proposed CCRT was originally designed for single objective problem, most ATM problems naturally involve multiple conflicting objectives, such as efficiency versus safety.

Contrary to single-objective CEA, the multi-objective co-operative co-evolutionary algorithm (MOCCA) does not have a single solution that optimizes all criteria concerned, but a set of trade-off solutions, known as Pareto-optimal solutions. Since none of the solutions in the Pareto-optimal set is absolutely better than any other, anyone of them is an acceptable solution. Which solution should be chosen depends on the decision-makers, preferences and various problem-related factors. Hence, a decision-maker is typically interested in knowing as many Pareto-optimal solutions as possible (Deb and Kalyanmoy, 2001). In this chapter, we propose a Multi-

Objective Co-operative Co-evolutionary Algorithm (MOCCA) in order to tackle multi-objective ATM issues. Two round of experiments are carried out to evaluate the algorithm performance on benchmark test problems with respect to different performance metrics. The results demonstrates that the proposed approach is capable of evolving solutions toward the true global Pareto-front more effectively and maintaining a higher diversity of the solution set.

5.1 Multi-Objective Co-Operative Co-Evolution Algorithm

The design of co-operative co-evolutionary algorithms for MO optimization is challenged with respect to many issues that are caused by the interaction with the MO optimization, such as individual evaluation (representative selection), fitness (credit) assignment, incorporation of various elitist, niching strategies and selection mechanisms.

5.1.1 Overview of CCEA by KC Tan

CCEA (Tan *et al.*, 2004), an acronym for Co-operative Co-Evolutionary Algorithm, is a kind of co-operative co-evolutionary algorithm particularly designed for multi-objective optimization. It is the first CCEA approach that employs a globalized perception of elitism: the non-dominated solutions found during the co-evolution process, from any sub-population, are preserved in one external archive. The global non-dominated archive helps the co-evolution mechanism work well in multi-objective optimization. The external archive also serves as a comparison set in the fitness assignment mechanism; and the second comparison indicator during the tournament selection scheme. In addition, an extending operator was designed in order to guide the co-evolutionary search to regions which were not explored enough.

After being introduced in (Tan *et al.*, 2004), CCEA was compared with various MOEAs on more benchmark test problems, with respect to more performance indicators in (Tan *et al.*, 2006). It showed that the CCEA produced competitive and robust results in finding the non-dominated solutions. As CCEA clearly outperformed the other localized-elitism alternative methods, it is chosen as the basis for the MOCCA that we are going to propose; and the point of reference in the quantitative and qualitative performance evaluation that this work will carry out.

We give a brief summary of the algorithm here. The interested reader please refer to (Tan *et al.*, 2004, 2006) for a more detailed description.

CCEA assign m sub-populations for m-parameter problem and each sub-population optimizes only one parameter. Starting with m initialized sub-populations and an empty external archive, the m sub-populations are evolved in a sequential way. The following steps are performed in one evolution cycle. First, in order to evaluate an individual in the evolving sub-population, the evaluated individual is combined with representatives from other sub-populations to form a complete solution. Then, the archive is updated based upon the evaluation results: if any member in the archive is dominated by the evaluated individual, then the dominated member is replaced by the evaluated solution. If the evaluated individual is non-dominated with all archive members, it is added to the archive; when the size of the updated archive exceeds a predefined limit, the archive member with the greatest niche count is replaced by the evaluated solution. At the same time, the ranges of the objective space is estimated from the renewed archive.

Afterwards, the evaluated individual will be assigned a rank and its niche count in the normalized objective space will be obtained. The next step represents offspring reproduction in sub-populations, by performing the genetic operations, consisting of tournament selection (based on rank and niche count), uniform crossover, and bit-flip mutation. Once an evolution cycle is completed, the least crowed archive members are found and cloned to sub-populations. Until the terminate criterion is met, the above steps are performed per iteration.

Although CCEA performed well due to the global archiving, dynamic sharing and extending operator, there is still room for improvement. In particular, we have identified the following issues as potential weaknesses of CCEA:

Fitness assignment mechanism: The simple Pareto-based rank scheme applied in CCEA levels an individual according to how many members in the external archive dominating it. This mechanism leads to individuals dominated by the same archive members have identical values for fitness. For instance in the case where all individuals in one sub-population are dominated by all solutions in the archive: all sub-population members have the same rank regardless of the dominance relation among themselves. As a consequence, the selective pressure decreases substantially and in this particular case, we get too much exploration and not enough exploitation, leading to reduce speed and efficiency of the stochastic search in the co-evolution.

Niching strategy: Although the dynamic fitness sharing method employed in CCEA requires no prior knowledge of the optimal trade-off surface, it still involves some user-defined parameters which can effect the efficiency of the algorithm. For instance, as it will been seen in Section 5.2.1, the niche radius function in (Tan *et al.*, 2004) is different from the canonical dynamic sharing used in the author's subsequent work (Tan *et al.*, 2006), leading to different results as shown in Figure 5.1, 5.2 and 5.3.

Archive updating scheme: The CCEA updates the external archive every time after a solution is evaluated. However, until the whole population is evaluated and the external archive is updated at the end of each evolution circle, the renewed archive is only a subset of the global trade-off front. For instance, the archive which is updated after the first sub-population is evaluated, does not cover the non-dominated individuals which will be found in the other $m-1$ sub-populations; thus, the most crowded archive member which has been eliminated in the first evolution circle might turn out to be in a less populated region. It leads to a destructive loss of valuable archive members. In addition, by using the dynamic sharing scheme, CCEA can not guarantee the extreme solutions in the external archive are preserved, during the archive truncation. The conservation of the extreme points in the updated non-dominated set helps in obtaining a good spread of non-dominated solutions.

5.1.2 Fitness Assignment

When an evolutionary algorithm is applied to multi-objective optimization, the first issue needs to be addressed is: How to accomplish fitness assignment properly, in order to guide the search towards the Pareto-optimal front. Contrary to single-objective optimization, multiobjective optimisation aims to optimize several components of a vector of objective functions simultaneously. In CCEA, after a complete solution is evaluated and mapped into an objective vector by the multiple objective functions, this objective vector reflects how well the examined individual co-operates with other subpopulations to produce good solutions. However, the objective vector cannot be used as fitness in evolutionary algorithm directly, each individual needs a single scalar fitness value so that reproduction operators can proceed in the usual way.

Pareto-based fitness assignment proposed by Goldberg (1989) is the fist type of fitness assignment method which explicitly uses Pareto dominance to determine the reproduction probability of each individual. All non-dominated individuals are assigned equal chance of reproduction (rank 1) and removed from contention. A new set of non-dominated individuals from the remaining population are then found and assigned rank 2, and so forth until the whole population is ranked.

Fonseca and Fleming (1993) proposed another popular Pareto-ranking scheme where an individual's rank is defined as the number of individuals in the population by which it is dominated. Therefor, non-dominated individuals get the best rank and dominated solutions are penalized based on the population density of the corresponding part of the trade-off surface. It is accepted that, comparing to the first scheme, this method is easier to interpret and analyze mathematically (Fonseca and Fleming, 1998).

Zitzler and Thiele (1999) improved Pareto-based approaches by introducing an external archive, which is a separated, continuously updated population where nondominated solutions are preserved externally. The archive is regarded as an elitism mechanism and preserves the best solutions found so far. An individual's fitness is then evaluated according to the number of solutions dominating it in the external archive. This scheme results in the continuous improvement of quality of the archive and ensures the convergence of evolutionary optimization.

As elaborated in (Zitzler *et al.*, 2001), the truth that individuals which are dominated by the same archive members have identical fitness values could decrease the selection pressure substantially in some particular case. Therefor, Zitzler *et al.* (2001) improved the fitness assignment in two ways: first, except for the number of archive members dominating the evaluated individual, the number of population members which dominate it is involved as well; second, they incorporated density information by adopting the k-th nearest neighbor method. Comparing to their previous work, this scheme distinguishes population solutions with different fitness values more effectively.

5.1.3 Niching Strategy

Except for the selection pressure, there is another important issue in the evolution process that must be addressed: how to maintain population diversity so that premature convergence can be prevented and an equally distributed trade-off front can be achieved. It is related to the first issue: an increase in the selection pressure possibly decreases the diversity of the population, and vice versa. It is important to strike a balance between these two factors, especially in multi objective evolutionary algorithm (MOEA) where a number of solutions with different trade-offs among the multiple objectives need to be evolved simultaneously. Thus, a multiple-solution preserving mechanism is required to avoid premature convergence, by maintaining high quality diversity.

It has been widely accepted that a niching method in MOEA is able to form and maintain multiple diverse solutions and preserve them for the entire duration of the GA run. Niching induces restorative pressure (Horn *et al.*, 1993) to balance the convergence pressure of selection (Horn, 1997). Under the effect of niching, the population of solutions is dynamically stable under the selection pressure.

Fitness sharing was introduced by Goldberg and Richardson (1987) to prevent genetic drift and to distribute the evolved population equally along the Pareto-optimal front in the search space. The method creates sub-divisions in the objective domain by degrading an individual fitness upon the existence of other individuals in its neighborhood defined by a sharing distance. The niche count is calculated by summing a sharing function over all members of the population, as defined in Equation 5.1.

$$nc_i = \Sigma_{i=1}^{N} sh(d(i,j)) \tag{5.1}$$

where the distance $d(i,j)$ represents the distance between individual i and j. The sharing function is defined as

$$sh(d(i,j)) = \begin{cases} 1 - \left(\frac{d(i,j)}{\sigma_{share}}\right)^{\alpha} & \text{if } d(i,j) < \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

where the parameter α is commonly set to 1 or 2.

As shown in Equation 5.2, the fitness sharing method involves a sharing parameter σ_{share}, which specifies the neighborhood size in the objective space and is named as niche radius. The parameter denotes the largest value of Euclidean distance within which any two solutions share each others fitness. This parameter is usually set by the user, although the size and shape of the objective landscape cannot often be predefined.

Tan *et al.* (2003) propose a dynamic sharing method that is capable of adaptively computing the niche radius σ_{share} to distribute the population evenly along the Pareto-optimal front at each generation. As shown in Equation, the niche radius is dynamically calculated based upon the population distribution at each generation.

$$\sigma_{share}^{(n)} = N^{1/(1-m)} \times \frac{d^{(n)}}{2} \tag{5.3}$$

where $\sigma_{share}^{(n)}$ is the niche radius at generation n in terms of the diameter $d^{(n)}$ and the population size N. The diameter $d^{(n)}$ is the diameter of the hyper-sphere at generation n and is often estimated by the average distance between the shortest and the longest possible diameter of the trade-off curve formed by the non-dominated individuals in the objective space (Tan *et al.*, 2003).

This method requires no prior knowledge of the optimal trade-off surface. Moreover, dynamically adopting the computation of niche radius is also more appropriate and effective than the method of off-line estimation with pre-assumed objective space, since the objective landscape may be changed any time along the evolution process.

Deb *et al.* (2002) replace the fitness sharing method with a crowded-comparison approach which does not require any user-defined parameter for maintaining diversity along the nondominated front. Instead of niche count, this niching mechanism requires the calculation of the average distance of two individuals on either side of the interested individual in the objective space, in order to get an estimate of the density of individuals surrounding the evaluated individual in the population. "The crowding distance serves as an estimate of the perimeter of the cuboid formed by using the nearest neighbors as the vertices. (Deb *et al.*, 2002) "

In the MOCC, a niching strategy is applied in two ways: for archive updating to maintain the capacity and diversity of the archive; and for tournament selection to break the tie in case two individuals have the same fitness values. When the maximum archive size is reached, the most crowded archive members will be eliminated by a truncation method based on niche count. When two individuals are compared in tournament selection, fitness values will be considered first followed by niche count in order to break the tie, i.e. the one with less niche count wins in the selection.

5.2 Investigation I

5.2.1 Methods

In this section, the first round of investigation is designed to examine the potential modifications of KC Tan's CCEA from two aspects: the fitness assignment mechanism and the niching strategy.

First, five fitness assignment methods are proposed to explore the potential improvements of CCEA, which are described as follows.

1. **F1**: The fitness of each individual is assigned according to how many members in the external archive dominate that individual, as implemented in (Tan *et al.*, 2004, 2006). The comparison set in the fitness assignment of individuals in sub-populations is exclusively the updated external archive. This fitness value partially reflects the distance between the objective vector of this individual and the Pareto front.

$$F1_i = N_i + 1 \tag{5.4}$$

   where N_i is the number of members in the external archive dominating the individual i in the objective domain.

2. **F2**: The number of members in each sub-population dominate the evaluated individual and those that are dominated by this individual are also considered in the fitness assignment process. In this case, the comparison set is extended to the combination of the updated external archive and the sub-population that the evaluated individual comes from.

$$F2_i = F1_i + n_{ia} + n_{ib} \tag{5.5}$$

   where n_{ia} is the number of members in the sub-population dominating individual i in the objective space, and n_{ib} is those dominated by individual i.

3. **F3**: The individual fitness is defined exclusively based on number of members in each sub-population dominating the evaluated individual and those that are dominated by this individual. At this time, the comparison set is solely the sub-population which the evaluated individual belongs to. This fitness value reflects the dominance relationship between the evaluated individual and the other members in its own sub-population.

$$F3_i = n_{ia} + n_{ib} \tag{5.6}$$

where n_{ia} is the number of members in the sub-population dominating the individual i in the objective domain, and n_{ib} is that dominated by individual i.

4. **F4**: The individual fitness is defined by the difference of the number of members in the whole population and the archive dominating the examined individual and by those dominated by it. The comparison set for F4 is a combination of the entire population and the updated external archive.

$$F4_i = F1_i + N_{ia} + N_{ib} \tag{5.7}$$

where N_{ia} is the number of members in the whole population dominating the individual i in the objective domain, and N_{ib} is that dominated by individual i.

5. **F5**: The individual fitness is defined by the difference of the number of members in the whole population dominating the examined individual and that of those dominated by it. At this point, the comparison set is exclusively the entire population.

$$F5_i = N_{ia} + N_{ib} \tag{5.8}$$

where N_{ia} is the number of members in the whole population dominating the individual i in the objective domain, and N_{ib} is that dominated by individual i.

Then, we propose four niching strategies to investigate the possible improvements of CCEA, which are described as follows.

1. **N1**: The first niching mechanism comes from Tan *et al.* (2004), where fitness sharing is employed and the niche radius is dynamically estimated using the archive at each generation, as in Equation 5.9.

$$\sigma_{share} = 2/archive\_size \tag{5.9}$$

   The sharing function is defined as:

$$sh(d(i,j)) = \begin{cases} \left(1 - \frac{d(i,j)}{\sigma_{share}}\right)^{\alpha} & \text{if } d(i,j) < \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \tag{5.10}$$

   where the parameter α is set to 2. And the niche count function is defined as Equation 5.1.

2. **N2**: In dynamic sharing which is proposed by Tan *et al.* (2003), the sharing radius is defined as in Equation 5.3 and the sharing function and niche count function are calculated by Equation 5.2 and 5.1.

3. **N3**: Based on the dynamic sharing, the niche radius is defined as

$$\sigma^{(n)}_{share} = 2 \times N^{1/(1-m)} \times d^{(n)} \tag{5.11}$$

   where $\sigma^{(n)}_{share}$ is the niche radius at generation n in terms of the diameter $d^{(n)}$ and the population size N. The diameter $d^{(n)}$ is calculated in the same way as in dynamic sharing (Tan *et al.*, 2003).

4. **N4**: In the crowding distance mechanism which is introduced by Deb *et al.* (2002), all individuals are sorted according to each objective function value in ascending order of magnitude. The boundary individuals (for each objective) are always conserved by assigning them a maximum distance value. All the other individuals are assigned a distance value equal to the absolute normalized

difference in the objective values of nearest neighbors. Each objective function is normalized before calculating the crowding distance.

To examine the effectiveness of each strategy on the CCEA from the above two aspects, we design two sets of comparison experiment. The performance is compared among strategies in each group. The first group is listed as follows.

1. **P11**: the CCEA algorithm with F1 and N1;
2. **P21**: the CCEA algorithm with F2 and N1;
3. **P31**: the CCEA algorithm with F3 and N1;
4. **P41**: the CCEA algorithm with F4 and N1;
5. **P51**: the CCEA algorithm with F5 and N1.

And the second group is listed as follows.

1. **P11**: the CCEA algorithm with F1 and N1;
2. **P12**: the CCEA algorithm with F1 and N2;
3. **P13**: the CCEA algorithm with F1 and N3;
4. **P14**: the CCEA algorithm with F1 and N4.

5.2.2 Test Problems

In order to assess the performance of the seven combined algorithms described in the previous section, they will be validated by nine optimization test cases: the test problems of ZDT1, ZDT2, ZDT3, ZDT4, ZDT5 and ZDT6 designed by (Zitzler *et al.*, 2000), and other test problems including FON (Fonseca and Fleming, 1995), KUR (Kursawe, 1991) and DTLZ2 (Deb *et al.*, 2002). These benchmark functions are widely used in the MOEA community, such as (Zitzler *et al.*, 2000; Garca-Pedrajas *et al.*, 2002; Iorio and Li, 2004; Dorronsoro *et al.*, 2011). The reason of their popularity is that they involve characteristics which are required for examining the performance of multi-objective optimization approaches in converging to the Pareto front as well as maintaining population diversity. The features of the search space in these functions are different from each other: e.g. ZDT1 involves a convex Pareto front which is continuous and uniformly distributed, while ZDT4 has 21^9 local Pareto fronts; instead of two objectives, the problem of DTLZ2 has high-dimensional objective space (five is employed in this work). Mathematical definitions of these test problems are listed as folows (Zitzler *et al.*, 2000; Tan *et al.*, 2006):

- **ZDT1**:

$$\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where \\
&f_1(x) = x_1 \\
&f_2(x) = g(x_2, ..., x_m)h(f_1(x), g(x_2, ..., x_m)) \\
&g(x_2, ..., x_m) = 1 + 9(\Sigma_{i=2}^{m} x_i/(m-1)) \\
&h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} \\
&\text{subject to} \quad x = (x_1, ..., x_m), m = 30, x_i \in [0, 1].
\end{aligned} \tag{5.12}$$

- **ZDT2**:

$$\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where \\
&f_1(x) = x_1 \\
&f_2(x) = g(x_2, ..., x_m)h(f_1(x), g(x_2, ..., x_m)) \\
&g(x_2, ..., x_m) = 1 + 9(\Sigma_{i=2}^{m} x_i/(m-1)) \\
&h(f_1, g) = 1 - (\frac{f_1}{g})^2 \\
&\text{subject to} \quad x = (x_1, ..., x_m), m = 30, x_i \in [0, 1].
\end{aligned} \tag{5.13}$$

- **ZDT3**:

$$
\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where\\
&f_1(x) = x_1\\
&f_2(x) = g(x_2, ..., x_m)h(f_1(x), g(x_2, ..., x_m))\\
&g(x_2, ..., x_m) = 1 + 9(\Sigma_{i=2}^{m} x_i/(m-1))\\
&h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} - (f_1/g)\sin(10\pi f_1)\\
&\text{subject to} \quad x = (x_1, ..., x_m), m = 30, x_i \in [0, 1].
\end{aligned}
\tag{5.14}
$$

- **ZDT4**:

$$
\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where\\
&f_1(x) = x_1\\
&f_2(x) = g(x_2, ..., x_m)h(f_1(x), g(x_2, ..., x_m))\\
&g(x_2, ..., x_m) = 1 + 10(m-1) + \Sigma_{i=2}^{m}(x_i^2 - 10\cos(4\pi x_i))\\
&h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}}\\
&\text{subject to} \quad x = (x_1, ..., x_m), m = 10, x_i \in [0, 1],\\
&and \quad x_2, ..., x_m \in [-5, 5].
\end{aligned}
\tag{5.15}
$$

- **ZDT5**:

$$\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where \\
&f_1(x) = 1 + u(x_1) \\
&f_2(x) = g(x_2, ..., x_m)h(f_1(x), g(x_2, ..., x_m)) \\
&g(x_2, ..., x_m) = \Sigma_{i=2}^{m} v(u(x_i)) \\
&h(f_1, g) = 1/f_1 \\
&u(x_i) \quad \text{gives the number of ones in the bit vector} \quad x_i \quad (unitation), \\
&v(u(x_i)) = \begin{cases} 2 + u(x_i) & \text{if } u(x_i) < 5 \\ 1 & \text{if } u(x_i) = 5 \end{cases} \\
&\text{subject to} \quad x = (x_1, ..., x_m), m = 11, x_1 \in \{0,1\}^{30}, \\
&and \quad x_2, ..., x_m \in \{0,1\}^{5}. \qquad (5.16)
\end{aligned}$$

- **ZDT6**:

$$\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where \\
&f_1(x) = 1 - exp(-4x_1)\sin^6(6\pi x_1) \\
&f_2(x) = g(x_2, ..., x_m)h(f_1(x), g(x_2, ..., x_m)) \\
&g(x_2, ..., x_m) = 1 + 9((\Sigma_{i=2}^{m} x_i)/(m-1))^{0.25} \\
&h(f_1, g) = 1 - (\frac{f_1}{g})^2 \\
&\text{subject to} \quad x = (x_1, ..., x_m), m = 10, x_i \in [0,1]. \qquad (5.17)
\end{aligned}$$

- **FON**:

$$\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where \\
&f_1(x) = 1 - exp[-\Sigma_{i=1}^{8}(x_i - 1/\sqrt{8})^2] \\
&f_2(x) = 1 - exp[-\Sigma_{i=1}^{8}(x_i + 1/\sqrt{8})^2] \\
&\text{subject to} \quad x = (x_1, ..., x_8), x_i \in [-2,2). \qquad (5.18)
\end{aligned}$$

- **KUR**:

$$
\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x)), \quad where \\
&f_1(x) = \Sigma_{i=1}^{2}[-10exp(-0.2\sqrt{x_i^2 + x_{i+1}^2})] \\
&f_2(x) = \Sigma_{i=1}^{3}[|x_i|^{0.8} + 5\sin(x_i^3)] \\
&\text{subject to} \quad x = (x_1, x_2, x_3), x_i \in [-5, 5].
\end{aligned}
\tag{5.19}
$$

- **DTLZ2**:

$$
\begin{aligned}
&Minimize \quad T(x) = (f_1(x), f_2(x), ..., f_M(x)), \quad where \\
&f_1(x) = (1 + g(x_M))\cos(x_1\pi/2)\ldots\cos(x_{M-1}\pi/2) \\
&f_2(x) = (1 + g(x_M))\cos(x_1\pi/2)\ldots\sin(x_{M-1}\pi/2) \\
&\vdots \\
&f_M(x) = (1 + g(x_M))\sin(x_1\pi/2) \\
&g(x_M) = \Sigma_{x_i \in x_M}(x_i - 0.5)^2 \\
&\text{subject to} \quad x_M = (x_M, ..., x_{M+9}), M = 5, x_i \in [0, 1], \\
&and \quad i = 1, 2, ..., M + 9.
\end{aligned}
\tag{5.20}
$$

5.2.3 Metrics of Performance

Unlike single-objective optimization problems, in multiobjective optimization the evaluation of an algorithm has multiple dimensions. In general, we wish to achieve closeness of the obtained non-dominated set to the Pareto-optimal front (we call it convergence goal) and achieve reasonable level of spread in the obtained non-dominated solutions (we call it diversity goal). Obviously, these two goals cannot be measured adequately with one performance metric. Here, we use four quantitative performance indicators, which are widely used in the MOEA literature.

The first metric is Generation Distance (GD), which is the average distance from the resulting non-dominated solutions to the true Pareto-optimal front (Van Veldhuizen, 1999; Zitzler *et al.*, 2000; Tan *et al.*, 2004, 2006). Formally, it is defined as in Equation (5.21), where n is the number of members in the obtained trade-off solutions and d_i is the Euclidean distance (in objective space) between the ith solution and the nearest solution in the Pareto-optimal front. The smaller the value of this metric, the better the convergence towards the Pareto-optimal front. When all obtained solutions lie exactly on the Pareto-optimal front, this metric takes a value of zero. GD reflects the exploitation ability of the MOEAs.

$$GD = \frac{\sqrt{\sum_{i=1}^{n} {d_i}^2}}{n} \tag{5.21}$$

The second metric Spacing (S) (Srinivas and Deb, 1994; Tan *et al.*, 2004, 2006) measures the solution distribution of achieved non-dominated solutions. Here, we calculate the Euclidean distance d_i between consecutive solutions in the obtained set of solutions. Thereafter, we calculate the average $\bar{d}$ of these distances. Then we use the following function to calculate the uniformity in the distribution:

$$S = \frac{\frac{1}{n}\sqrt{\sum_{i=1}^{n}(d_i - \bar{d})^2}}{\bar{d}} \tag{5.22}$$

where n is the number of members in the obtained trade-off solutions and $\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$. The smaller the value of spacing is, the more evenly solutions in obtained non-dominated solutions distribution. When all resulting solutions lie evenly long the Pareto-optimal front, this metric takes a value of zero. Spacing reflects the algorithm's ability to maintain diversity.

The third metric is Maximum Spread (MS) (Zitzler *et al.*, 2000; Tan *et al.*, 2004, 2006), which measures how well the true Pareto front is covered by the resulting non-dominated solutions. It is defined as follows,

$$MS = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left\{\frac{min(f_i^{max}, F_i^{max}) - max(f_i^{min}, F_i^{min})}{F_i^{max} - F_i^{min}}\right\}^2} \tag{5.23}$$

where m is the number of objectives, f_i^{max} and f_i^{min} are respectively the maximum and minimum of the ith objective in resulting non-dominated set; and F_i^{max} and F_i^{min} are respectively the maximum and minimum of the ith objective in the true Pareto front. When the obtained trade-off set covers the entire true Pareto front, this metric takes a value of one. MS mirrors the MOEAs' ability to explore the spread of the non-dominated solutions.

The last metric is a volume-based metric – Hypervolume Ratio (HVR) (Van Veldhuizen, 1999; Tan *et al.*, 2004, 2006) – which is a ratio of the hypervolume of the resulted non-dominated set and that of the true Pareto front. It is defines as follws,

$$HVR = \frac{volume(\bigcup_{i=1}^{N} v_i)}{volume(\bigcup_{j=1}^{trueN} v_j)} \tag{5.24}$$

where N is the number of solutions in the achieved nondominated set and $trueN$ is that of members in Pareto front. Mathematically, for each solution i in the evaluated non-dominated set, a hypercube is constructed with a reference point W and the solution i as the diagonal corners of the hypercube. Simply, a W point can be found by constructing a vector of the worst objective function values. Thereafter, the hypervolume of the evaluated non-dominated set is a union of all hypercubes constructed by the non-dominated solutions and the point W. The HVR is considered as the most appropriate scalar indicator since it "combines both the distance of solutions (towards some utopian trade-off surface) and the spread of solutions (Zitzler *et al.*, 2001)" and can reduce the bias because of the normalization.

All these measurement criteria are calculated from the objective vectors of the obtained solutions. According to the definition, no correlation among the first three metrics while HVR generally correlates to all the other three metrics.

5.2.4 Simulation Results and Discussions

The performance is compared among strategies in each group based upon the test problems described in Section 5.2.2. In order to guarantee a fair comparison, all runs considered are implemented with the same binary coding scheme of 30-bit per decision variable, tournament selection, uniform crossover, and bit-flip mutation. An exception is made for the discrete problem of ZDT5, where the first variable is a 30-bit string while the rest of the variables are of 5-bit string according to the problem definition. In the simulation, 30 independent runs with random initial populations of each implementation are performed on each of the test problems in order to study the statistical performance, such as consistency and robustness of the algorithms. The number of generations for each simulation run is fixed as 200.

Figure 5.1, 5.2 and 5.3 summarize the simulation results of the algorithms for each test problem with respect to each performance metric. The distribution of simulation data for the 30 independent runs is represented in the box plot format to visualize the distribution of simulation data. Each box plot represents the distribution of a sample set where a thick horizontal line within the box encodes the median, while the upper and lower ends of the box are the upper and lower quartiles. Dashed appendages illustrate the spread and shape of distribution and dots represent the outside values.

The problems of ZDT1, ZDT2 and ZDT3 are relatively easy to solve according to their definition functions. As shown in Figure 5.1, all fitness assignment methods perform well except for F1 for GD. In terms of solution distribution, F2 and F3 have better performance than F4 and F5 on spacing. The results show that F1 performs poorly for both metrics of GD and spacing. It reveals that the comparison set in the fitness assignment exclusively depending on the updated external archive does not benefit the CCEA's exploitation ability.

Although F1 partially reflects the distance between the objective vector of the evaluated individual and the Pareto front, it can not handle the situation where most or all of the individuals are dominated by same number of archive members. F2, F3, F4 and F5 reflect the domination relation between the interested individual and other population members. They effectively distinguish the "good" individuals from normal ones, thus improve the ability of exploitation. Since P12, P13 and P14 employ the same fitness assignment funciton – F1, they suffer in identifying

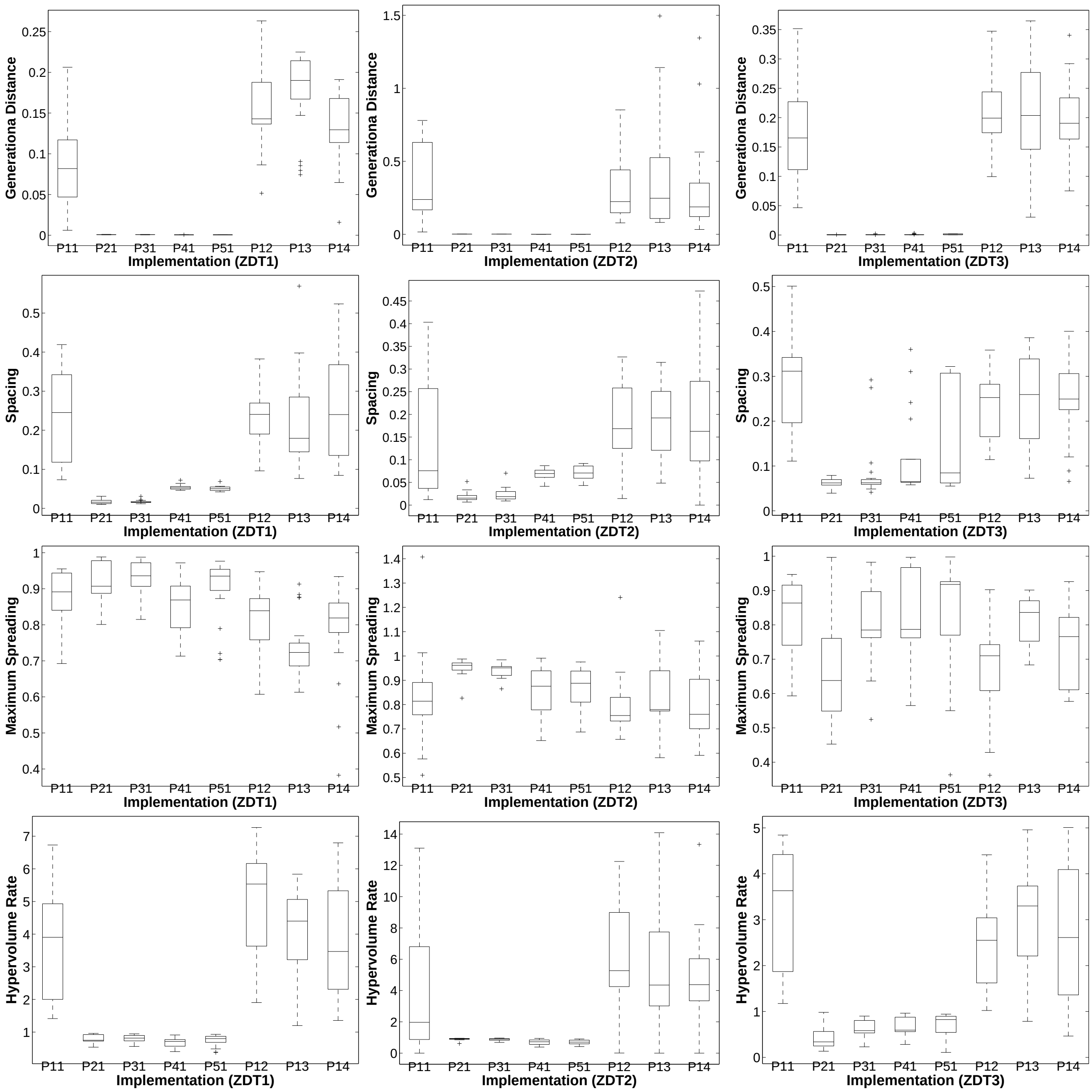


Figure 5.1: Box plots for test problems ZDT1, ZDT2 and ZDT3 in Experiment I with respect to the metrics of generation distance (GD), spacing (SP), maximum spread (MS) and hypervolume ratio (HVR).

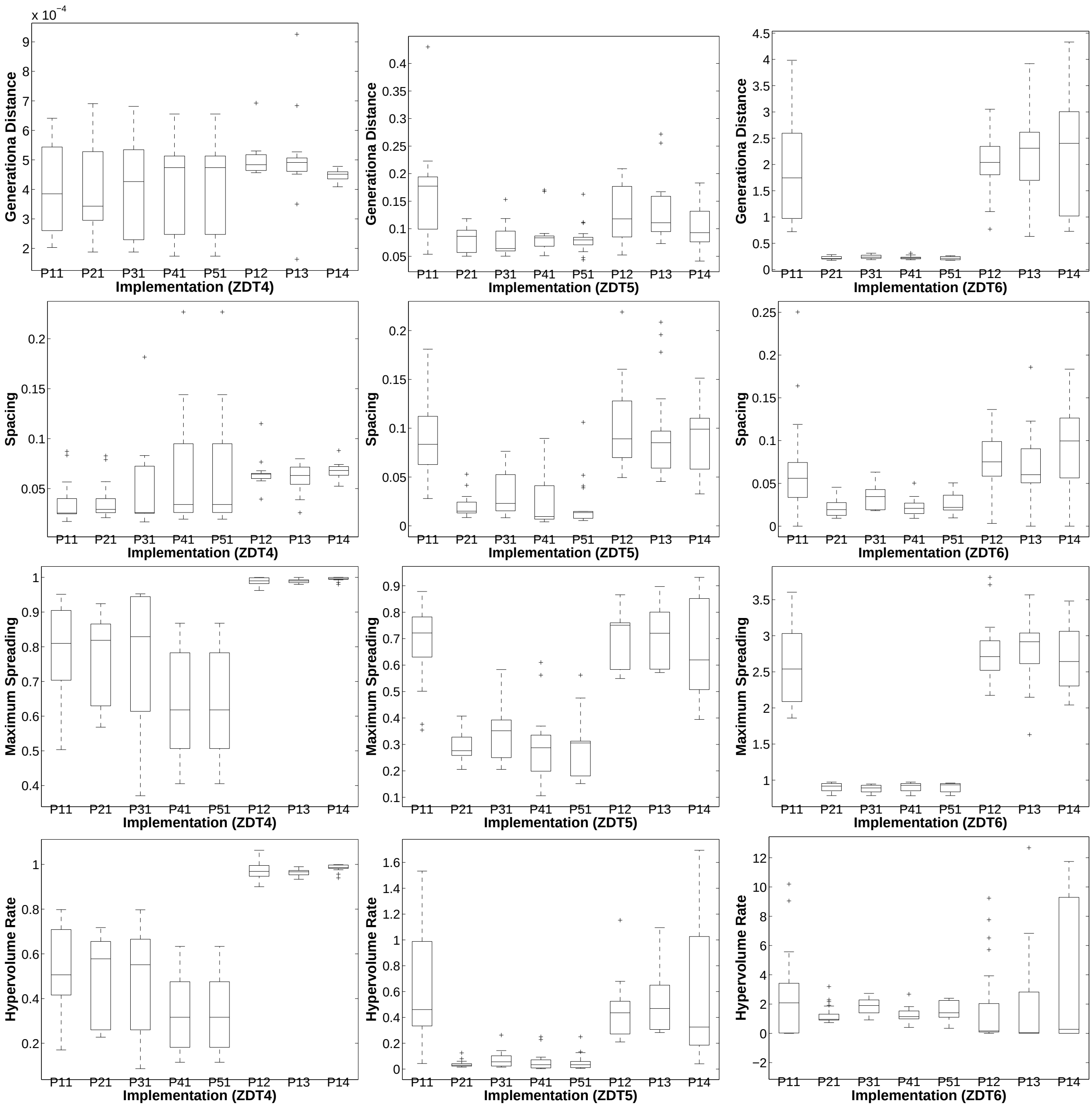


Figure 5.2: Box plots for test problems ZDT4, ZDT5 and ZDT6 in Experiment I with respect to the metrics of generation distance (GD), spacing (SP), maximum spread (MS) and hypervolume ratio (HVR).

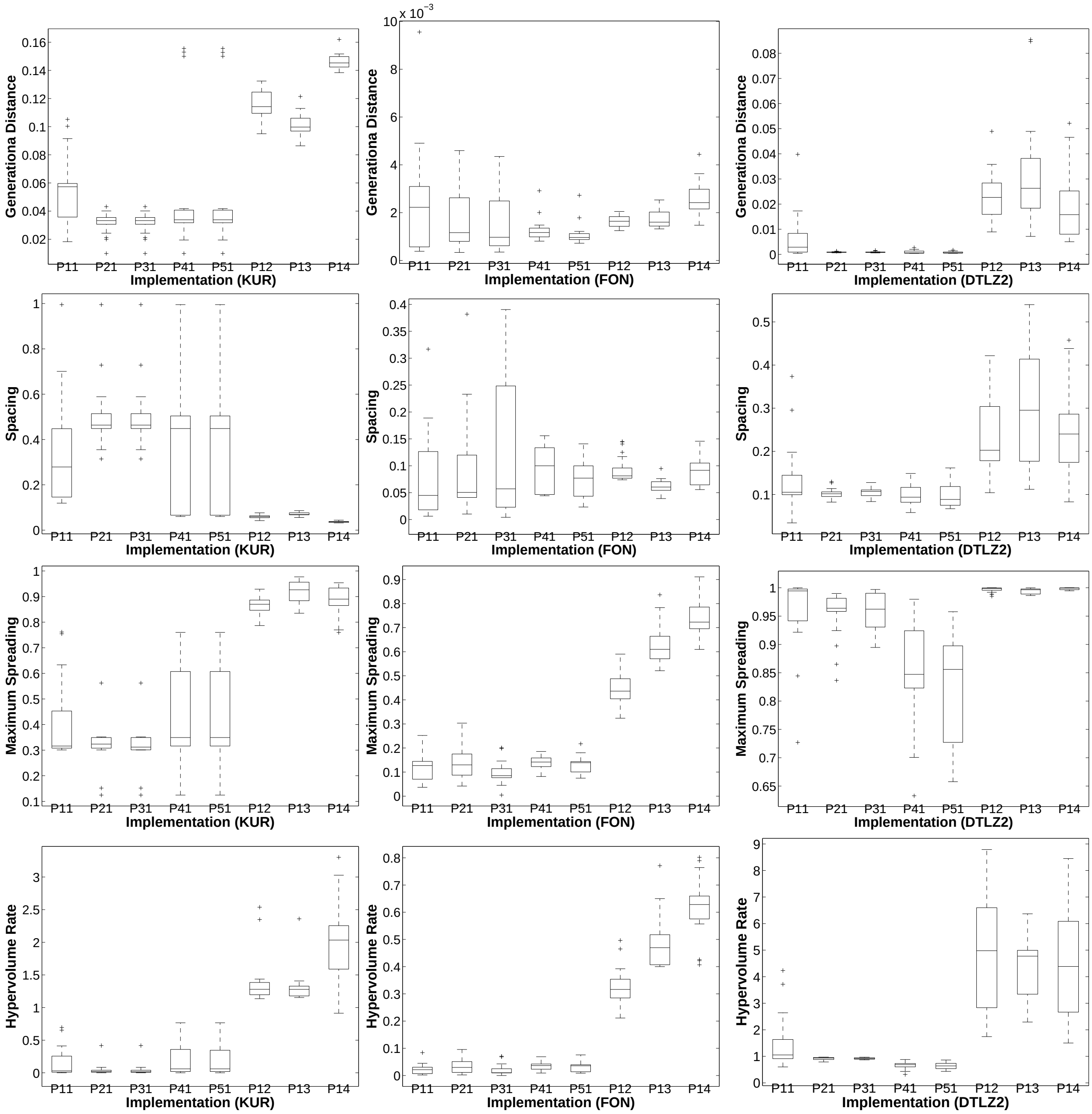


Figure 5.3: Box plots for test problems FON, KUR and DTLZ2 in Experiment I with respect to the metrics of generation distance (GD), spacing (SP), maximum spread (MS) and hypervolume ratio (HVR).

the individuals that approximate the true Pareto optimal front, leading to poor performance on metrics of MS and HVR.

The problem ZDT4 has 21^9 local Pareto front which challenge the algorithm's ability to escape from local optima. Figure 5.2 shows that all examined approaches manage to find the global Pareto front of the problem of ZDT4. The results demonstrate that all fitness assignment methods and niching strategies are able to evolve individuals towards the global Pareto front effectively for ZDT4. It is noticed that N4 (crowding distance) helps the algorithm to cover the entire global Pareto front excellent.

ZDT5 is the only discrete problem in the test suite. It is deceptive, i.e. most search methods lead to a local Pareto front while the global Pareto front is isolated. Figure 5.2 shows that all examined approaches manage to escape from the local Pareto front of the problem of ZDT5. It is observed that F2, F3, F4 and F5 perform well on GD, but poorly on MS and HVR. It reveals that an effective fitness assignment mechanism solely can not make sure the solution distribution and diversity – an efficient niching strategy is necessary for multi-objective optimization.

ZDT6 is a difficult problem to tackle by most MOEAs since it involves nonuniform distributed solutions. The F2, F3, F4 and F5 again perform better than F1, which illustrates that those fitness mechanisms are capable of evolving individuals towards the global Pareto front effectively. On the other hand, all of those niching strategies are trapped at local optima and suffer in convergence towards the optimal Pareto front for problem ZDT6.

There is a high interaction among the variables in FON and KUR, and every variable has a family of optima. For the problem FON, all the comparing strategies have good performance for the metrics of GD and spacing. In terms of MS and HVR, P11, P21, P31, P41 and P51 perform similarly on the ability of exploring the spread of non-dominated individuals since they employ the same niching strategy – N1. For the problem of KUR, F2 and F3 perform excellent on GD, but poorly on MS and HVR, while P2, P3 and P4 perform poorly on GD, but good on metrics of spacing and MS.

There are five objectives to optimize for the problem DTLZ2, which is used to evaluate the performance in producing adequate pressure for driving the evolution of individuals toward the large Pareto front in the high-dimensional objective domain. Figure 5.3 shows that CCEA scales well with F1 and F2, while F4 and F5 suffer in covering the entire global Pareto front well. Niching strategies N2, N3 and N4 work worse than N1 on GD, but excellent in terms of MS.

Generally, F2, F3, F4 and F5 produce excellent performance evolving individuals towards the global Pareto front effectively and escape from deceptive local optima. For the metric of spacing, F2 shows a distinct advantage over other fitness assignment functions, which demonstrates its capability of affecting the CCEA with respect to the accuracy as well as diversity, i.e. effectively identifying the individuals that approximate the true Pareto optimal front. For the metric of maximum spread and hyper volume ratio, F2 performs competitively in exploring the spread of nondominated individuals in most of the test problem (e.g. ZDT1, ZDT2, ZDT3, ZDT6), as shown in Figure 5.1 and 5.2. In brief, F2 is strongly competitive with other fitness assignment methods.

As for niching strategies, there is not an obviously outstanding one according to the results, i.e. although N4 produce excellent ability in exploring the spread of nondominated individuals in ZDT3, FON and DTLZ2, its competitors perform better than it in other test problem. Since it can not be concluded which niching strategy is the best choice, another set of experiment is necessary.

5.3 Investigation II

5.3.1 Experimental Design

In order to distinguish the effectiveness of each niching strategy on the CCEA, we design another set of comparison experiment, each of which employs the same fitness assignment function – F2, which is the best fitness assignment function found in previous section.

1. **P21**: the CCEA algorithm with F2 and N1;
2. **P22**: the CCEA algorithm with F2 and N2;
3. **P23**: the CCEA algorithm with F2 and N3;
4. **P24**: the CCEA algorithm with F2 and N4.

5.3.2 Simulation Results and Discussions

As the first round of the experiment, the second round simulation is also implemented based upon the nine benchmark test problems described in Section 5.2.2. All runs considered are implemented in the same way as the first one: chromosome coding, selection, crossover and mutation. Also, the number of generations in each run is fixed as 200. Each algorithm runs 30 times on each test function so as to study the statistical performance.

The box plots in Figure 5.4, 5.5 and 5.6 summarize the simulation results of each algorithm on each test problem with respect to the performance metrics, as described in Section 5.2.3. As a reference point, the CCEA with P11 (as in Section 5.2.1) is plotted in the figures as well for comparison.

The results shows that for almost all the test problems, P24 has the best performance in terms of MS and HVR, implying that P24 has a strong ability to explore the spread of non-dominated individuals and cover entire global Pareto front well. The reason is that crowding distance scheme is capable of conserving the extreme points in the Pareto set, leading to a good spread of non-dominated solutions. For the metrics of GD and spacing, P24 performs well in most of the test problems, which reflects that it is capable of evolving individuals toward the true global Pareto front effectively and maintaining a high diversity of the solution set. In brief, the combination of F2 and N4 is strongly competitive with other groups of strategies.

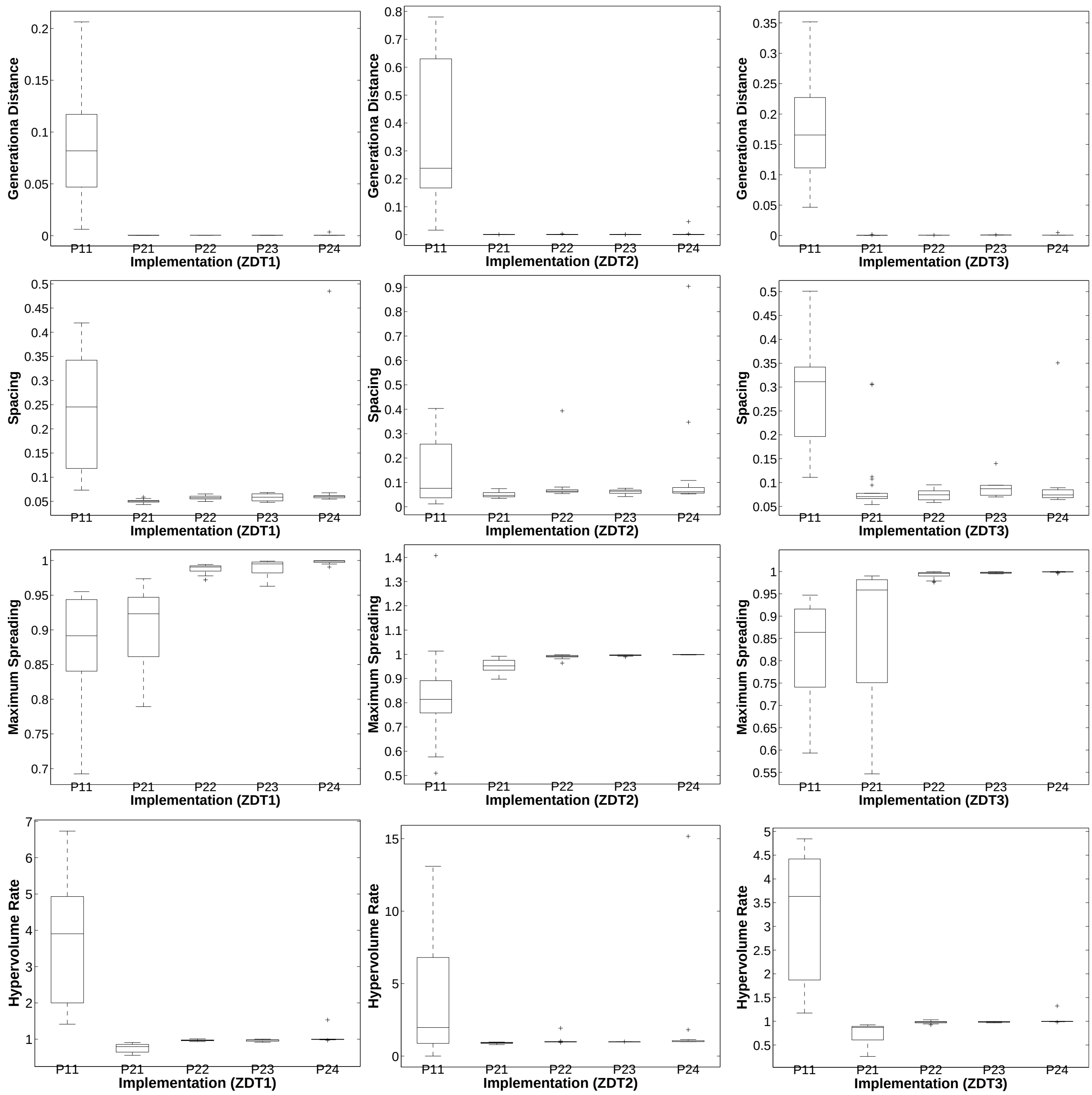


Figure 5.4: Box plots for test problems ZDT1, ZDT2 and ZDT3 in Experiment II with respect to the metrics of generation distance (GD), spacing (SP), maximum spread (MS) and hypervolume ratio (HVR).

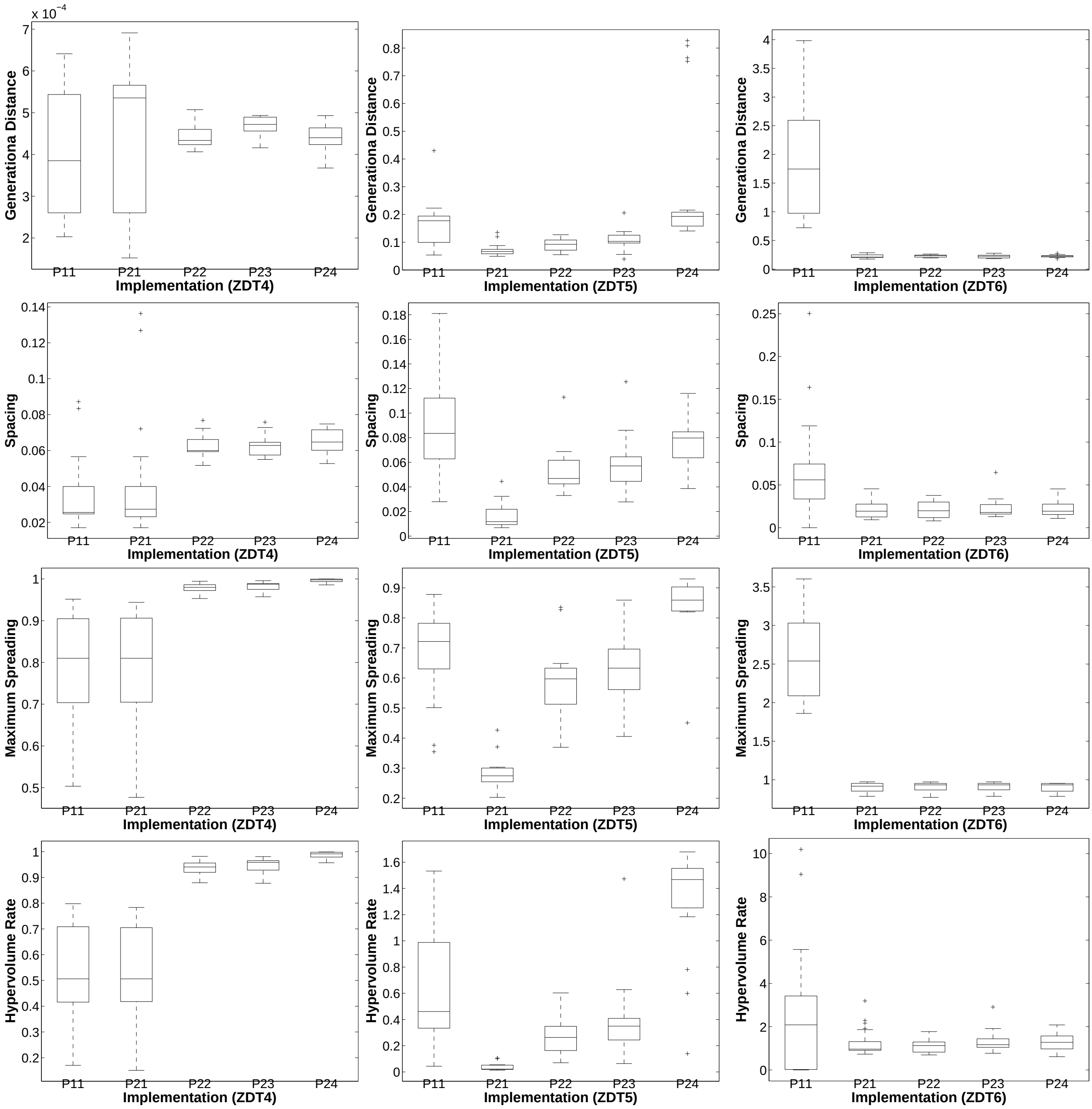


Figure 5.5: Box plots for test problems ZDT4, ZDT5 and ZDT6 in Experiment II with respect to the metrics of generation distance (GD), spacing (SP), maximum spread (MS) and hypervolume ratio (HVR).

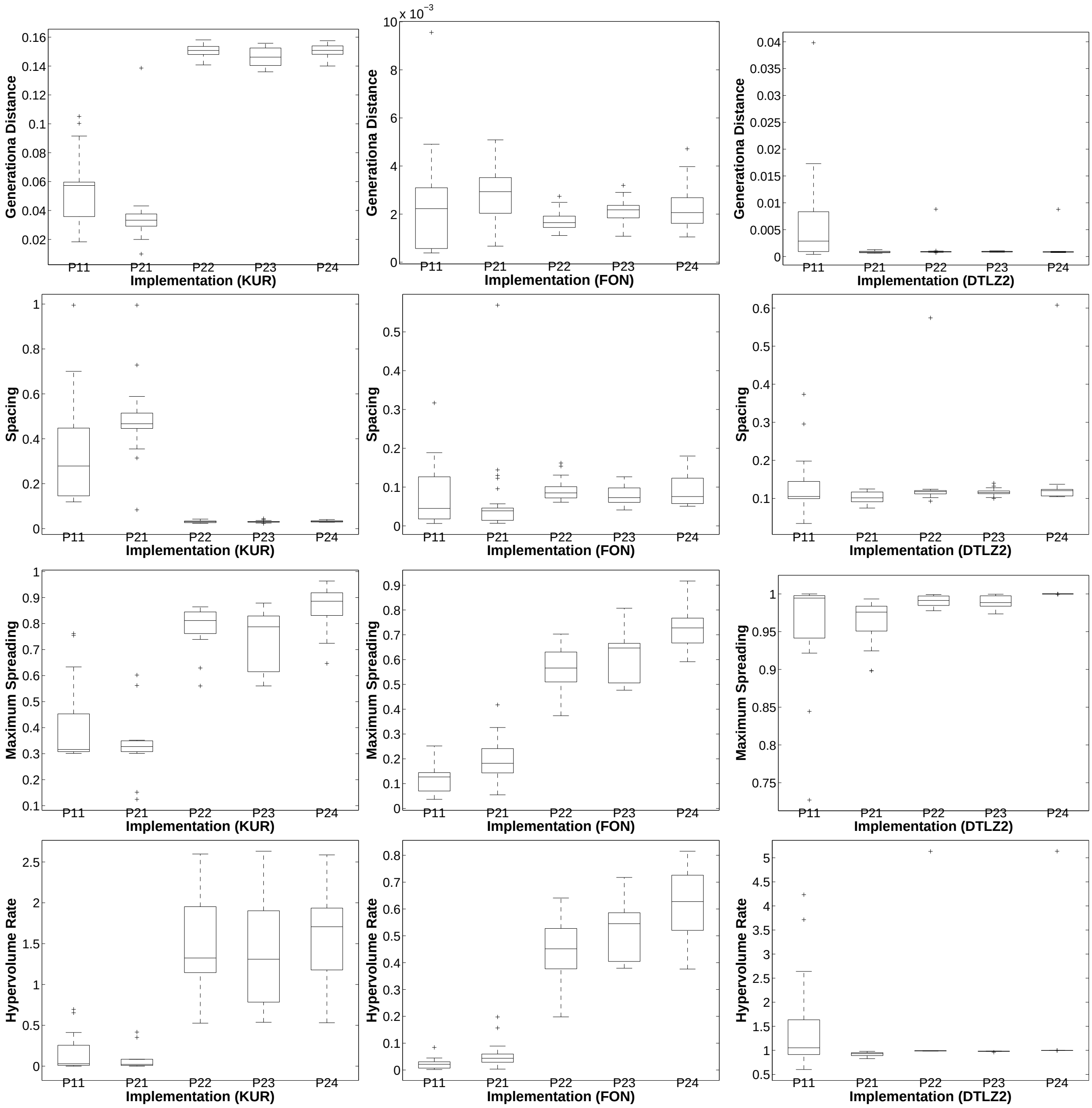


Figure 5.6: Box plots for test problems FON, KUR and DTLZ2 in Experiment II with respect to the metrics of generation distance (GD), spacing (SP), maximum spread (MS) and hypervolume ratio (HVR).

5.4 Proposed Multi-Objective Co-Operative Co-Evolution

In this section, we will describe the proposed MOCCA which addresses the issues discussed in Section 5.1.1. As shown in Algorithm 5.1, 5.2, the overall MOCCA co-evolves as follows.

Starting with M initialized sub-populations and an empty external archive, the following steps are performed in one evolution cycle, where each sub-population is performed in a sequential way. First, two individuals are chosen from the sub-population: the best individual (BI), which has the lowest fitness value and resides in a less populated region, and a random individual (RI). In order to evaluate an individual in the evolving sub-population, two types of co-operation are implemented to produce two combined solutions. The first co-operation combines the evaluated individual with the best representative from every other sub-population, while the second co-operation combines the individual with a random individual from every other sub-population.

The two combined solutions are then evaluated and unless solution of the first co-operation is dominated by the solution of second co-operation, it is retained in the choosing pool (CP). After the entire population is evaluated and according combined solutions are preserved in the CP, the nondominated solutions of CP are selected and saved in the external archive A. Then, the archive members are assigned the crowding distance (based on N4 as in Section 5.2.1). If the archive size exceeds the predefined size $\bar{N}$, the members with less crowding distance value will be eliminated.

The archive member with highest crowding distance value is copied to the sub-populations, where each part of this member is cloned into its corresponding sub-population. A fitness value is assigned to each population solution based on $F2$ as the Function 5.4, and a crowding distance value is calculated according to $N2$.

The next step represents offspring reproduction in sub-populations, by perform-

Algorithm 5.1 Multi-Objective Co-Operative Co-evolutionary Algorithm

Input: M (subpopulation number)
N (subpopulation size)
$\bar{N}$ (external archive size)
T (maximum number of generations)
Output: A (external archive)

MOCCA Main Loop:

$t = 0$;
$A = \emptyset$; $BP = \emptyset$; $RP = \emptyset$; $CP = \emptyset$;

for $i = 1, 2toM$ **do**

Initialize subpopulation $P(i)_t$;
$BP = BP \cup RandomIndividual(P(i)_t)$;
$RP = RP \cup RandomIndividual(P(i)_t)$;

end for

for $i = 1, 2toM$ **do**

for $j = 1, 2toN$ **do**

BI is a complete individual which combines representative j in $P(i)_t$ with the best representatives from the other $M - 1$ subpopulations in BP ;

RI is a complete individual which combines representative j in $P(i)_t$ with the random representatives from the other $M - 1$ subpopulations in RP ;

Evaluate $\{BI,RI\}$;

$CP = CP \cup Better\{BI, RI\}$;

end for

end for

$A =$ NondominatedSet$(CP \cup A)$;

$A =$ CrowdingDistance(A);

if Size$(A) > \bar{N}$ **then**

ArchiveTruncation(A);
$EI =$ the individual with largest crowding distance in A;

end if

for $i = 1, 2toM$ **do**

$P_t =$ ExtendingOperator$(P(i)_t)$;
$P_t =$ CalculateFitness$(P(i)_t)$;
$P_t =$ CalculateCrowdingDistance$(P(i)_t)$;

end for

Algorithm 5.2 Continued: Multi-Objective Co-Operative Co-evolutionary Algorithm

while $(t < T)$ **do**

  $t = t + 1$;
  $BP = \emptyset$; $RP = \emptyset$; $CP = \emptyset$;

  for $i = 1, 2toM$ **do**

    $BP = BP \cup BestIndividual(P(i)_t)$;
    $RP = RP \cup RandomIndividual(P(i)_t)$;
    $P(i)_t$ = TournamentSelction($P(i)_t$);
    $P(i)_t$ = UniformCrossover($P(i)_t$);
    $P(i)_t$ = Mutation($P(i)_t$);

  end for

  for $i = 1, 2toM$ **do**

    for $j = 1, 2toN$ **do**

      BI is a complete individual which combines representative j in $P(i)_t$ with the best representatives from the other $M - 1$ subpopulations in BP;

      RI is a complete individual which combines representative j in $P(i)_t$ with the random representatives from the other $M - 1$ subpopulations in RP;

      Evaluate $\{BI, RI\}$;

      $CP = CP \cup Better\{BI, RI\}$;

    end for

  end for

  A = NondominatedSet($CP \cup A$);

  A = CrowdingDistance(A);

  if Size(A) $> \bar{N}$ **then**

    ArchiveTruncation(A);
    EI = the individual with largest crowding distance in A;

  end if

  for $i = 1, 2toM$ **do**

    P_t = ExtendingOperator($P(i)_t$);
    P_t = CalculateFitness($P(i)_t$);
    P_t = CalculateCrowdingDistance($P(i)_t$);

  end for

end while

ing the genetic operations, consisting of tournament selection (based on fitness value and crowding distance), uniform crossover, and bit-flip mutation. Once an evolution cycle is completed, the least crowed archive members are found and cloned to sub-populations. Until the terminate criterion is met, the above steps are performed per iteration.

In contrast to CCEA, MOCCA uses a fitness assignment (F2 in Section 5.2.1) strategy which reflects not only the dominance relation between an evaluated individual and archive members, but also its dominance relation with other solutions in its sub-population. This difference effectively avoids the situation that individuals dominated by same archive members have identical fitness values.

Furthermore, the dynamic sharing niche count has been replaced by an alternative niching method – crowding distance (N4 in Section 5.2.1) – which was implemented in NSGA II (Deb *et al.*, 2002). This scheme improve the CCEA in two respects: 1) it requires no user-defined parameter, and 2) it prevents boundary solutions being removed during archive truncation process.

In addition, the external archive is updated in a more global perception: instead of renewing the archive every time after a complete solution is evaluated, the archive is updated after the whole population members are evaluated, i.e., the non-dominated individuals with less crowed neighborhood in the whole population will be preserved in the trade-off set. In this way, the solutions which resides in less populated regions of the global Pareto front are saved and the boundary solutions in the non-dominated set could be conserved during the archive truncation.

5.5 Chapter Summary

A multi-objective co-operative co-evolutionary algorithm (MOCCA), which can co-evolve solutions towards the efficient set of trade-offs effectively while maintaining diversity of the solution set, is proposed in this chapter. Comparing to the CCEA proposed by Tan *et al.* (2006), which is the most competitive existing co-operative co-evolutionary algorithm for multi-objective optimization, the presented MOCCA involves an improved fitness assignment strategy which effectively avoids the situation that individuals dominated by same archive members have identical fitness values; an enhanced niching strategy which requires no user-defined parameter and prevents boundary solutions being removed during archive truncation process; and a new globalized scheme for archiving updating so that the solutions which resides in less populated regions of the global Pareto front are saved. After being validated on various benchmark test cases on different performance metrics, results demonstrate that the proposed approach is capable of evolving solutions towards the true global Pareto-front more effectively while maintaining a higher diversity of the solution set, comparing to CCEA.

After examined on benchmark test problems which are well known in the literature, in next chapter we will apply the proposed approach on a real-world problem to complete the study.

Chapter 6

TeA Airspace Design

The research question that needs to be resolved in this chapter is: How to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety?

In the integrated TeA simulator developed in Chapter 3, the evaluated dynamic CDA model that has been proven to be superior is fixed, and it is assumed that the positions of approaching aircrafts passing through their designated transition points are deterministic. However, the variability in an aircraft's position which can be caused by several factors, such as weather and an aircraft's envelope, is a characteristic of ATM problems. Therefore, to judge on the quality of the TeA airspace design strategies, the probabilistic nature of an aircraft's position necessitates the inclusion of a measure of collision risk with other aircrafts passing the neighbourhood of that aircraft at the same time. Since the fixed TeA airspace model in dynamic CDA can not handle such complexity, we introduce here the concept of TeA airspace design for capacity-demand balancing including a measure of collision risks derived from the probabilistic nature of aircraft's performance. We consider in this chapter the case where an aircraft's position is a stochastic normally distributed variable.

The MOCCA proposed in the previous chapter is employed as the new search engine in the CCRT framework proposed in Chapter 4. The multi-objective CCRT is then applied to generate scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety. By involving the interdependencies between constrained ground resources and air traffic, the airport and its TeA airspace are collaborated as a whole system; and the output will be a fine trade off between different local objectives – ground and air. The cooperation between ground and air supports a systematic approach towards achieving system-level objectives while designing TeA airspace configurations.

The multi-objective CCRT will be applied to generate scenario-specific terminal airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety. The multi-objective CCRT will provide an analyst with the trade-off between these two ATC priorities - efficiency and safety; thus solutions can be selected based on the level of criticality for meeting the demand.

6.1 Overview

The fundamental goal underlying the integrated TeA system for TeA airspace design is to develop a simulation and modelling environment where a TeA airspace design concept involving the collision risk dimension can be modeled, while considering the interdependencies between the traffic distributions and the dynamics of ground resources. The main simulation engine of the proposed TeA system embraces an arrival manager which models the basic air traffic and ATM operational features on the airport surface and the transition airspace surrounding it (e.g. TeA airspace, waypoints, aircrafts and trajectory generation) that are essential for evaluating any air traffic concept. Other modules which implement queue management, safety separation control are built around the core engine.

In ideal conditions, dynamic CDA trajectories are generated based on a fixed TeA airspace model which is a set of five concentric cylinders and each cylinder has certain number of wedges that represent transition points from one level to another. It is assumed that the positions of approaching aircrafts passing through their designated transition points are deterministic. However, the variation between aircraft's assigned route and actual trajectories is a characteristic of ATM problems. The route followed by a flight differs from the desired or ideal route due to a variety of reasons such as uncertainty in aircraft performance, navigation system error or flight technical error (MITRE CAASD, 2007). Therefore, to judge on the quality of the TeA airspace configuration strategy for a traffic scenario, the probabilistic nature of an aircraft's position necessitates the inclusion of a measure of collision risk with other aircrafts passing the neighbourhood of that aircraft at the same time. Since the fixed TA model in dynamic CDA is not designed to handle such complexity, we introduce here the concept of dynamic airspace design for future transition airspaces. We consider in this work the case where an aircraft's position is a stochastic normally distributed variable.

The idea of this dynamic design is to evolve a series of static designs that meet different traffic distributions and allow for dynamic CDA. Each static design can then be used to meet a specific set of circumstances; allowing the authority to switch between slides based on demand to maximise system level efficiency while minimising risk.

It is well noted that the airport capacity constraints are major restriction for terminal airspace capacity (Leiden *et al.*, 2007); thus a disturbance in the ground resources could influences the air traffic efficiency in the TeA airspace. Constrained ground events is a concept which involves disruptions to the availability of any ground resources, including runways, taxiways and gates. The unexcepted disruptions may be caused by convective weather or equipment break-down. For instance, snow on a taxiway can lead to increased taxiway occupancy time or even temporary taxiway closure; a broken lighting installation in the runway light systems will affect the routine runway operations in the low visibility situations (Nolan, 2004). The constrained ground operations potentially leads to surface congestion in the airport, which could propagate elsewhere in the TeA environment and cause system-level inefficiency. As a result, the complex interaction between constrained ground events and arrival traffic scenarios will affect the system capacity and arrival delays.

The integrated TeA simulator developed in Chapter 3 is modified and extended to involve the TeA airspace design concept and collision risks calculation. There are some overlaps of these two simulators: both of them share similar ground resource modelling for arrivals, data structure for ground resource records and some procedures: queue manager, occupancy time and safety separation. For simplicity, we exclude those components which are alike with previous simulator.

Although in this chapter, we only implement the TeA airspace design concept with given arrival traffic scenarios in an air-ground integration manner, and do not analyze the coupling of arrival and departure; nonetheless, as demonstrated in the previous simulator developed in Chapter 3, our simulation environment is generic and easily applied to the problem of synthesizing arrival and departure traffic simultaneously. The integration in this chapter is on the Ground-Air Collaboration Level as in Figure 2.4 which is lower level, yet it can practically assist TeA terminal routes designers who are currently working on defining and developing pre-defined terminal arrival trajectories.

6.2 Input/Output

The input to the TeA System for airspace design consists of both static and random factors. The static factors includes ground resource modelling and capacity for each resource and the dynamic factors contains TeA airspace design strategy, the time and space context of arrival flight plans and ground events. Human factors such as load on ATC controller are excluded from explicit consideration. The detailed information that each item contains is listed as follows.

- Ground Resource Modelling: is the ground network representing the airport ground resources (runways, taxiways and gates) serving arrival traffic which is the same model as in Figure 3.6;
- Resource Capacity: is a user defined factor and in this work, it is assumed

that each resource has the capacity of 1;

- TeA Airspace Design Strategy: is the TeA airspace model which is used to generate terminal trajectories for arrivals;
- Arrival Flight Plan: consists of Aircraft Name, Aircraft Type, Estimated Time of Activated, Activation Point, Outer Marker Point, Designated Runway, Array of Designated Taxiways, Designated Gate;and
- Ground Event: consists of Event Location, Event Name, Start Time, Duration.

The output from the TeA System for airspace design is modified arrival flight plans, total flight delay value and collision risk, explained as follows.

- Modified Arrival Flight Plan: contains Aircraft Name, Aircraft Type, Estimated Time of Activated, Activation Point, Outer Marker Point, Modified Runway, Modified Array of Taxiways, Modified Gate;
- Total Flight Delay : is calculated by averaging the differences between the ETA of flights from their OM to requested gates and their actual time of arrival (ATA) at actual gates from their respective OM; and
- Collision Risk: is calculated by averaging the collision risks of assigned CDA trajectories in TeA airspace for all flights.

6.3 Air Traffic Scenario Generation

Air traffic demand data refer to the space-time information of the aircraft fleet mix servicing an airport within the TeA system. Since it embraces both a spatial based nature and time based nature, it is normally represented by the statistical characteristics of air traffic's spatial and temporal distributions (Netjasov *et al.*, 2011). As with regards to time, a snapshot analysis of a number of traffic movements through the TeA system over a certain time period is typically a feasible approach to

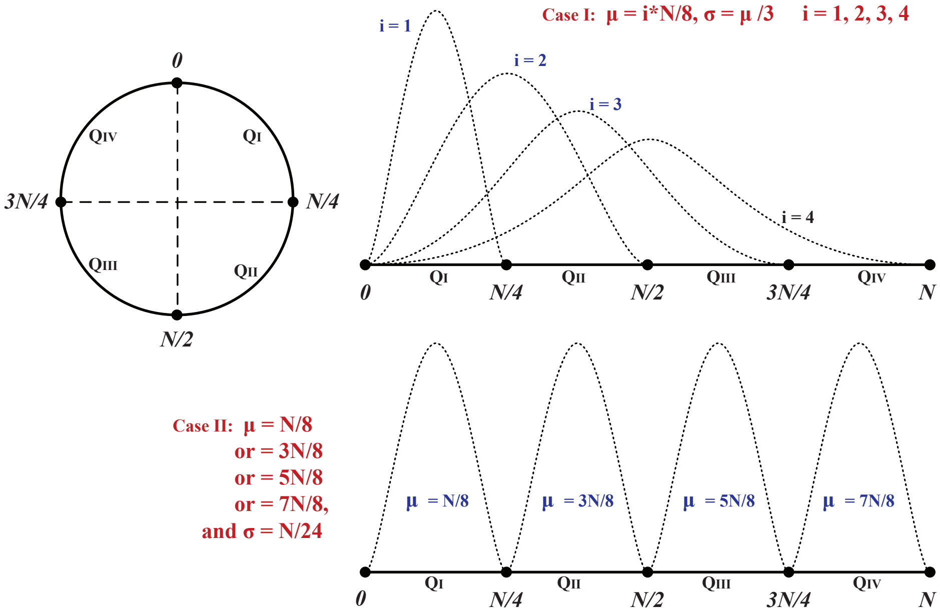


Figure 6.1: Arrival Traffic Coming from Diverse Directions

understand the traffic density. From a space perspective, it is necessary to determine the spatial distribution of the air traffic through identifying the traffic flow at entry points to the TeA system.

The flight traffic scenario generation method in Chapter 4 illustrates how to generate arrival flights coming mostly from one direction determined by parameters μ and σ_{GA}, during a specific time interval governed by factor T. The flight activation points on the outer most circle (as in Figure 3.8) are assumed to be distributed with normal distribution around the point μ. Parameter σ_{GA} is used to calculate the standard deviation according to '3-sigma rule' as illustrated in equation 4.1. The arriving flights are assumed to be activated at the outer marker circle in a Poisson process. The inter-arrival time for a pair of successive flights has an exponential distribution with parameter T, which is derived from a uniform random number generator in a pre-defined time interval.

The work in this chapter requires a variation of traffic scenarios in terms of both direction and density. Thus at this point, we explain how to extend the traffic scenario generation method in Chapter 4 for arriving aircraft coming from two or more directions with diverse traffic density.

The activation circle contains N ($N = 104$) activation points in total and each quadrant encloses $N/4$ points as illustrated in Figure 6.1. This circle can be seen as a line segment which is uniformly divided into four shorter sections and each section embraces $N/4$ activation points which are equally divided. It is assumed that arriving flights are activated at points distributed normally. The mean value of this normal distribution is μ and the standard deviation is represented by σ. According to the '3-sigma rule', it is assumed $\sigma = (\mu-S)/3whereSisthestartpointoftheclustersmaple$ at this time.

In extreme cases with $\mu = N/2$ and $\sigma = \mu/3$, arriving flights could be activated at any point on the whole circle, which means flights may originate from any direction. Figure6.1 illustrates a variety of situations where flights come from. In Case I, all flights originate from one direction (activated in consecutive quadrants) with various sample width. For instance, with all aircrafts coming from Quadrant I, μ equals $N/8$ and σ is equal to $N/24$. In Case II, flights originate from different directions (activated in discrete quadrants). For example, when aircrafts come from either Quadrant I or Quadrant III, all flights are generated randomly from two isolate normal distributions: in one case, μ equals $N/8$ and σ is equal to $N/24$, while in the other, $\mu = 5N/8$ and σ equals to $N/24$.

The aircrafts' arriving process is normally considered as a stochastic process in which a number of discrete arrivals occur in a certain time interval. It is a continuous time process that is defined as Poisson process in probability theory. The time between each pair of consecutive flights has an exponential distribution with parameter T. The value of T is derived from a uniform random number in a pre-defined time interval. An inter-aircraft time distribution of 50 implies the next flight activation time is Poisson-distributed with a mean of 50 seconds. Traffic scenarios with a variation of density differ with each other based on diverse time intervals which are pre-defined and employed to generate the uniform random number T and the number of the arrivals happened. For instance, an amount of 100 with a time interval of $T = [45, 120]$ can be used to generate 100 arriving flights in about 3 hours.

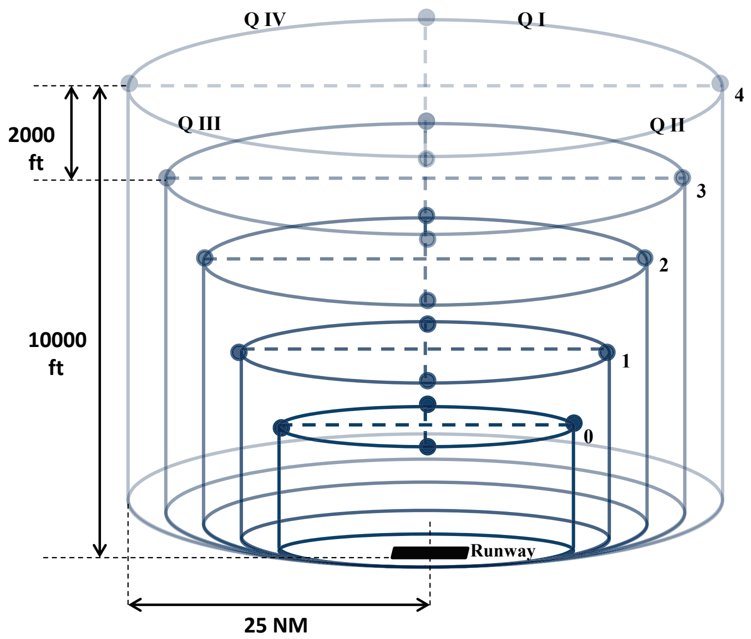


Figure 6.2: TeA Airspace model for TeA Airspace Design

6.4 Data Structure

This section discusses the major data structures of the integrated TeA system for TeA airspace design including air resources modelling and air resources records. Since ground resources modelling and ground resources records are similar with those for arrivals described in Chapter 3, we exclude explains for those components in this section for simplicity.

6.4.1 Air Resources Modelling

$$RingRadius = \frac{TAR \times \ (Ring\,Number + 1)}{5} \tag{6.1}$$

As in dynamic CDA model (Alam *et al.*, 2011), the transition airspace is equally divided with a 5 nm safety separation defined as a set of five concentric cylinders with a runway (touchdown point) at the center, as illustrated in Figure 6.2. The height of the transition airspace is set to 10,000 ft and the radius to 25 nm (transition airspace radius (TAR)). The outermost cylinder (denoted as Ring 4) has a radius of 25 nm and the inner cylinders (Rings 3, 2, 1 and 0) radii of 20, 15, 10 and 5 nm respectively, as calculated by Equation 6.1. The outermost cylinder's height is 10,000 ft which corresponds to the starting altitude of the CDA and the inner cylinders have heights of 8000, 6000, 4000 and 2000 ft respectively. Thus, the transition airspace is divided into 5 levels, with each level divided into 2000 ft to give a typical jet aircraft sufficient vertical height to maneuver given a low thrust setting.

In this work, we divide each cylinder into 4 equal regions of 90 degrees each of which we call a quadrant. Each quadrant on every cylinder has wedges which represent transition points from one level to another. These wedges are supposed to be spaced at least 1.5 nm apart to provide safe separations between approaching aircraft (Spence, 2003). When design the dynamic TA, we define the separation distance between two adjacent wedges varies from 1.5 nm to the length of the quadrant, to ensure the hard safety constraint. The maximum number of wedges for a given quadrant is calculated as:

$$Number\,of\,Wedges = \frac{0.5\pi \times Ring\,Radius}{Separation\,Distance} \tag{6.2}$$

The data structure for each wedge point consist of: ID, Capacity, Aircraft Types, Latitude, Longitude, Altitude and Speed, each of which is explained as follows.

- ID: is the name of the wedge point which contains the ring number, the quadrant number and the wedge point number;
- Capacity: is the number of aircrafts this wedge point can handle at the same time;
- Aircraft Type: represents which types of aircraft this wedge point can handle;
- Latitude: is the latitude value of this wedge point;
- Longitude: is the longitude value of this wedge point;
- Altitude: is the altitude value of this wedge point; and

- Speed: is the speed constraint for the wedge point.

It is assumed that the capacity of each wedge point is 1, which means each wedge point can only be used by the one aircraft to which it is assigned for a certain duration. For instance, if an arrival is in a wedge point for a particular duration, this point is blocked and cannot be used by any other arrival during that time window. Each wedge point has access to any wedge point on the subsequent lower ring.

6.4.2 Air Resources Records

An array named *Transition Airspace Resources Records* is designed to record occupancies of airspace resources , whose elements are presented as follows.

- Resource Name: represents the name of the air resource;
- Start Time: represents the time when the aircraft starts to occupy the air resource;
- Duration: represents how long the aircraft will occupy the air resource; and
- Aircraft Type: the category of the aircraft – heavy, large or small.

When an aircraft is assigned an appropriate air route (as in Section 3.4.6), any way point (air resource) in this route is recorded in the *Transition Airspace Resources Records*, along with its name, time window for the aircraft to stay and the aircraft's type. When computing a terminal route for another aircraft, we check the possible TeA airspace resource availability by searching through the record table; for example, if an arrival is about to be assigned to wedge point (4, 4, 6) at time T, but that wedge point is occupied by another aircraft at the same time according to the record table, then wedge point (4, 4, 6) is not available for this new arrival at that time.

6.5 Collision Risk Calculation Methodology

Collision Risk (CR) is the most commonly objective representing safety in ATM domain. Contrary to the traditional CR which is caused by a pair of conflict flights, the CR here is a result of the probabilistic nature of an aircraft's operational position when it passes through a designated way point in the TeA airspace. It is assumed that the possible position for a flight to go through one way point (W_i) is a random variable which is distributed normally ($N(\mu_i, \sigma_i)$). The position of this way point is the mean value (μ_i), and the standard deviation (σ_i) equals the aircraft's RNP (Required Navigation Performance) value divided by 2. The RNP value for each aircraft is derived from Eurocontrol's Aircraft Database (BADA). It is also assumed that any two arrivals are independent to each other to get the upper bound of the collision risk as we target at worst case study.

Algorithm 6.1 Collision Risk Calculation for Each Aircraft

Input:
$AC = \{a_1, a_2, ..., a_N\}$;
$Ring = \{r_1, r_2, ..., r_5\}$;

i = 0;
j = 0;

while (i < N) **do**
  while (j < 5) **do**
    if within the aircrafts envelope **then**
      PotentialWaypoins = $\{W_1, W_2, ..., W_5\}$;
      CR_{ij}= min $\{CR_1, CR_2, ..., CR_5\}$;
    end if
    j ++;
  end while
  i ++;
end while

The collision risk is calculated by averaging the collision risks of assigned CDA trajectories in TA airspace for all flights according to the following procedures (as in

Algorithm 6.1): Given N aircrafts 5 rings (as in the airspace model in Figure 6.2), 5 way points, one on each ring, need to be assigned to each aircraft. As long as it is allowed by the aircraft's performance envelope (according to BADA), 5 way points in a row on each ring will be considered as potential way points for this aircraft based on the minimized distance from previously chosen way point. After their CRs are calculated, the first one with the minimum CR is assigned to this aircraft. The CR for a certain traffic scenario with M arriving flights is then calculate according to equation 6.3. For each aircraft, its CR is summarized by five CR values one of which is the CR at one of its assigned TeA airspace way points.

As illustrated in Figure 6.3, the CR at each potential way point for aircraft AC_a is calculated based in such way: firstly, search in left neighborhood of 8 way points (which is the maximum number of wedges in 10 nm) of this way point W_{ij} to get the first (nearest one to the wedge W_{ij}) wedge W_{left} which is blocked by another flight AC_b at the same time T_{ij}. The mean value for AC_a is W_{ij} and its standard deviation is $RNP_a/2$; while the mean value for AC_b is W_{left} and the standard deviation is $RNP_b/2$. If there is any overlap between this two probability density functions, then assume the boundary of this overlap on W_{ij} side is value $Y_{ij|left}$, then $Y_{ij|left}$ is calculate based on the equation 6.6. Since $3\sigma_j < \mu_j - Y_i < 4\sigma_j$, to do the worst case study and get the upper bound of the collision risk, we over estimate the overlap by choosing 4. Similarly, the boundary of this overlap on W_{left} side is value Y_{left}, then Y_{left} is calculate based on the equation 6.7. Thus, the collision risk on the left side is calculated based on equation 6.5 which is derived from the cumulative distribution function. In the similar way, the CR on the right side is computed based on formula 6.8, 6.9 and 6.10. Then the CR values on both sides are summarized to obtain value CR_{ij} according to equation 6.4 and its maximum value is 1.

$$CollisionRisk = \frac{\Sigma_{i=1}^{M}(\Sigma_{j=1}^{N} CR_{ij})}{M} \tag{6.3}$$

$$CR_{ij} = min\{(CR_{ij|left} + CR_{ij|right}), 1\} \tag{6.4}$$

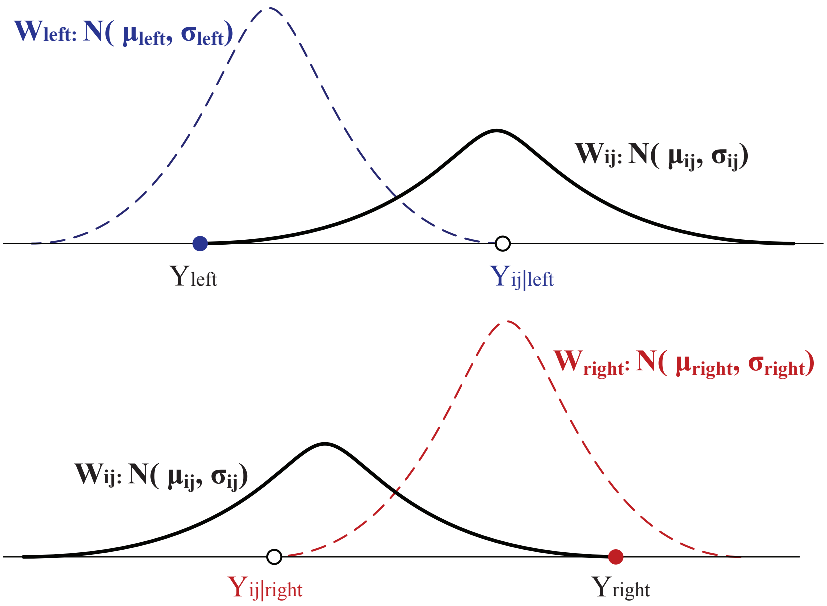


Figure 6.3: The Possible Collision Risk for A Potential Way Point

$$CR_{ij|left} = \frac{1}{2}[1 - erf(\frac{Y_{left} - \mu_{left}}{\sqrt{2}\sigma_{left}})][1 + erf(\frac{Y_{ij|left} - \mu_{ij}}{\sqrt{2}\sigma_{ij}})] \tag{6.5}$$

$$Y_{left} = \mu_{ij} - 4\sigma_{ij} \tag{6.6}$$

$$Y_{ij|left} = \mu_{left} + 4\sigma_{left} \tag{6.7}$$

$$CR_{ij|right} = \frac{1}{2}[1 - erf(\frac{Y_{ij|right} - \mu_{ij}}{\sqrt{2}\sigma_{ij}})][1 + erf(\frac{Y_{right} - \mu_{right}}{\sqrt{2}\sigma_{right}})] \tag{6.8}$$

$$Y_{right} = \mu_{ij} + 4\sigma_{ij} \tag{6.9}$$

$$Y_{ij|right} = \mu_{right} - 4\sigma_{right} \tag{6.10}$$

where

- M = number of arrival aircrafts;
- N = number of way points in one CDA trajectory route(N=5);
- CR_{ij} = collision risk of aircraft i at way point j (W_{ij});
- $CR_{ij|left}$ = collision risk of aircraft i at way point j (W_{ij}) caused by another aircraft occupying of another way point (W_{left}) on its left neighbourhood;
- $CR_{ij|right}$ = collision risk of aircraft i at way point j (W_{ij}) caused by another aircraft occupying of another way point (W_{right}) on its right neighbourhood;
- Y_{left} = value of the boundary point of the overlap between two probability density functions of standard normal distribution on W_{left} side, which leads to upper bound of CR by over estimating the overlap based on '3-sigma rule' (worst case study);
- $Y_{ij|left}$ = value of the boundary point of the overlap between two probability density functions of standard normal distribution on W_{ij} side (worst case study);
- μ_{ij} = mean value of the normal distribution of aircraft i at way point j (W_{ij});
- σ_{ij} = standard deviation of the normal distribution of aircraft i at way point j (W_{ij}), which equals half the value of the RNP of the aircraft taking this way point;
- μ_{left} = mean value of the normal distribution of way point W_{left};
- σ_{left} = standard deviation of the normal distribution of way point W_{left}, which equals half the value of the RNP of the aircraft taking this way point;
- Y_{right} = value of the boundary point of the overlap between two probability density functions of standard normal distribution on W_{right} side (worst case study);

- $Y_{ij|right}$ = value of the boundary point of the overlap between two probability density functions of standard normal distribution on W_{ij} side (worst case study);

- μ_{right} = mean value of the normal distribution of way point W_{right}; and

- σ_{right} = standard deviation of the normal distribution of way point W_{right}, which equals half the value of the required navigation performance of the aircraft taking this way point.

The collision risk in this thesis is a result of the probabilistic nature of an aircraft's operational position when it passes through a designated way point in the terminal airspace. As described in Section 6.5, it is assumed that the possible position for a flight to go through one way point is a random variable which is distributed normally. However, Algorithm 6.1 and Equation 6.3, 6.4 are generic. As long as the overlapping area in Figure 6.3 can be calculated, the methodology in this thesis can be adapted to other probabilistic descriptions of uncertainty.

6.6 TeA System for Airspace Design

This section discusses the architecture and design principles of the integrated TeA system as well as simulation engine arrival manager. Since queue manager, occupancy time and safety separation are similar with those for arrivals described in Chapter 3, we exclude explains for those components in this section for simplicity.

6.6.1 Architecture of TeA System for Airspace Design

The architecture of integrated TeA simulation consists of the following key components (see Figure 6.4):

This architecture was designed to be modular and flexible enough to incorporate new air-ground network and ATM operational constraints. Starting from the top:

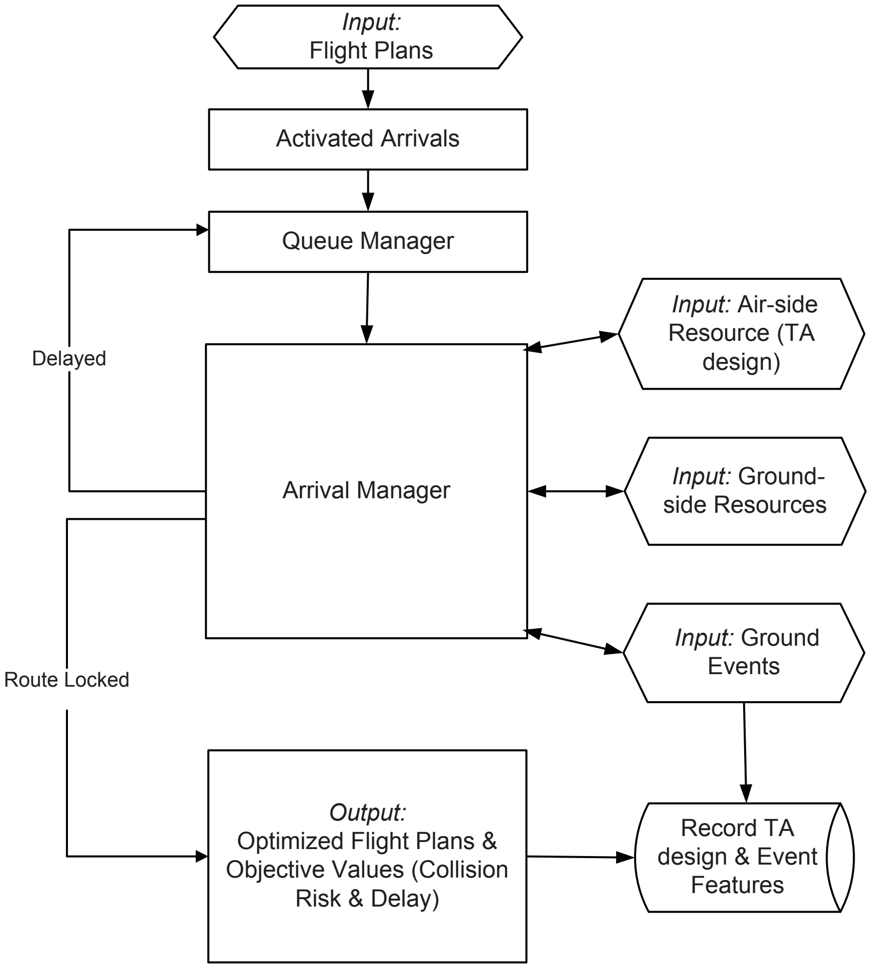


Figure 6.4: ATOMS/TeA Simulator System Flow Chart

Given a set of arrival flight plans, we put them into a conceptual waiting queues based on their activation time. Once an arrival reaches 75nm before the TeA, it becomes active. All active arrivals are processed by a Queue Manager in which they are positioned in a queue according to the activation times. Based on the first come first assigned principle, the first arrival is selected to go through the optimization process of the Arrival Manager. Arrival Manager calculates appropriate flight plans (including CDA trajectory and ground route) for each arrival with high ATC priority, according to decoded TeA airspace design model which represents air-side resources (as in Figure 6.2), the ground network (as in Figure 3.6) representing ground-side resources and decoded ground events. The ATC priority refers to those two objectives: safety (lower collision risk) and efficiency (less delay). If the optimized route for an aircraft is found, values of objectives are output and the TeA airspace design strategy and ground event features are recorded, otherwise the delayed aircraft is sent back to the Queue Manager to wait for another optimization process. The subsection 6.6.2 describes the procedural details of the Arrival Manager. After all flights are scheduled, we calculate the averaged delays and collision risks for all flights.

6.6.2 Arrival Manager

Figure 6.5 presents the procedural flowchart for the Arrival Manager of which the operational details are as follows.

Arrival management of an arrival starts from selecting 5 potential TeA entry points which are closest to the aircraft's current position and need minimum variability in their headings. Then the estimated time of arrival (ETA) to each point and collision risk of the arrival at this point are calculated. The one with minimum CR is chosen as the TeA entry point (IAF). Check whether the CR of the IAF is less than 1. If yes, a full enumeration of the search space is performed to generate all possible routes from the IAF to the final arrival fix FAF (which are CDA routes), followed by eliminating those links that violate the aircraft performance constraints

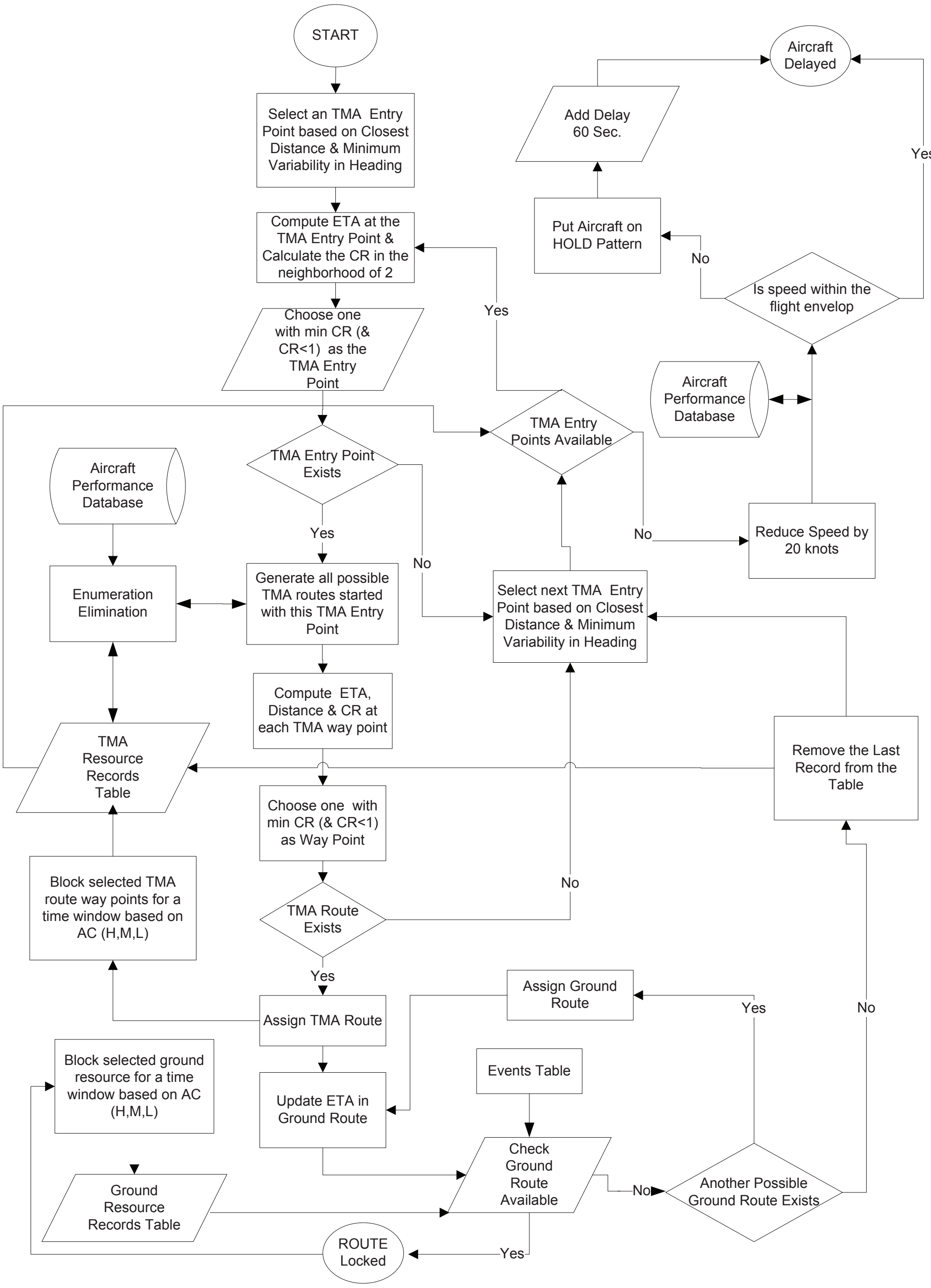


Figure 6.5: Arrival Manager Flow Chart

derived from Eurocontrol's Aircraft Database (BADA); if no, the next 5 potential TeA entry points which meet the constraints (closest distance and minimum variability in heading) are selected, followed by computing the ETA, the CR, and checking whether the CR is less than 1, then generating all possible routes again. After searching, if there is no available potential TeA entry point, the arrival has to be delayed (delay process), either by putting it into a HOLD pattern (adding a delay of 60 seconds) or reducing its speed by 20 knots if the aircraft's performance allows. After update the ETA for each way point, we calculate the CR and distance from previous way point, then identify a set of non-dominated solution CDA routes with shorter distance and lower CR. If one way point is occupied by another aircraft at its ETA according to the *Airspace Resource Records*, its CR is then defined as 1 and this way point will be eliminated. The first approved CDA route is selected and all the wedge points in this route are blocked for a time window based on the aircraft type (as shown in Table 3.1), and their ETAs and occupancy times are recorded in the *Airspace Resource Records*. This ensures that no two arrivals occupy the same wedge point in a given time window. If no CDA route is approved, the arrival needs to find next set of TeA entry points which meet the objectives.

We start to process the ground route which is pre-defined in the flight plan. The ETA is updated for each way point on the ground according to the designated ground route. In line with the events table and *Ground Resource Records Table*, if any ground way point (runway, taxiway or gate) is unavailable due to a certain ground event or is simply occupied by another aircraft, the aim is to meet the designated ground route while minimizing changes to the existing route. If there is no possible way of leading an arrival from its assigned runway to its assigned gate, an alternative gate is selected. If the runway is completely closed, the search is performed using the next available runway and the aircraft's ground route is reassigned (if there is another possible ground route) by searching through the ground resource network as shown in Figure 3.6. The ETA for each ground resource is updated, each resource is blocked for a time window based on the aircraft type (as shown in Table 3.1) and the *Ground Resource Records Table* is updated accordingly. However, if there is no available ground route, the arrival aircraft has to find next set of available TeA entry points and the last record in the Airspace Resource Records is removed. In the end, the available CDA and ground routes are combined as the full arrival route for this flight which is locked and updated to the FMS.

6.7 Methodology

Given the same set of approaching flights, the more separation distance between artificial way points in the TA model, the lower level of collision risk could be caused. Increasing separation distance leads to reducing the number of way points in TA, causing loss of throughput and growth of delay. Therefore, the problem here becomes a multi-objective problem with at least two objectives that are in conflict. The two objectives are: minimizing flight delay and minimizing collision risk. The one with smallest flight delay is the one with highest collision risk. This conflict between the two objectives makes a multi-objective representation of the problem a natural way to solve it.

The multi-objective co-operative co-evolutionary algorithm proposed in Chapter 5 is used in the framework developed in Chapter 4, the following two populations of partial solutions evolve together:

- TA design scenarios; and
- ground events scenarios.

This section, the population design of above two populations will be discussed and an appropriate fitness function will be designed for our problem.

6.7.1 Population Design of TA Design Scenarios

Therefore, when encode the TA design into a chromosome, the number of wedges for a given quadrant varies from 1 to the maximum number of wedges for it. A

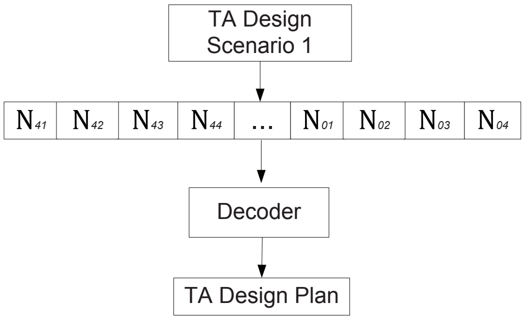


Figure 6.6: TA chromosome design showing genomes which encode TA model in scenarios.

transition airspace radius of 25 nm and a minimum separation distance of 1.5 nm gives the maximum number of wedge points as 26, 20, 15, 10 and 5 for each quadrant on Rings 4, 3, 2, 1 and 0 respectively.

Each individual in the TA design population is represented by a chromosome, which has a set of values (genes) characterizing the number of wedges in each quadrant of each ring in the TA airspace. As shown in Figure 6.6, there are 20 genes in each chromosome. The gene values are encoded as follows.

- N_{4i} is the number of wedge points in each quadrant on ring 4, the value of which is selected randomly from the interval [1, 26], $i \in \{1, 2, 3, 4\}$.
- N_{3i} is the number of wedge points in each quadrant on ring 3, the value of which is selected randomly from the interval [1, 20], $i \in \{1, 2, 3, 4\}$.
- N_{2i} is the number of wedge points in each quadrant on ring 2, the value of which is selected randomly from the interval [1, 15], $i \in \{1, 2, 3, 4\}$.
- N_{1i} is the number of wedge points in each quadrant on ring 1, the value of which is selected randomly from the interval [1, 10], $i \in \{1, 2, 3, 4\}$.
- N_{0i} is the number of wedge points in each quadrant on ring 0, the value of which is selected randomly from the interval [1, 5], $i \in \{1, 2, 3, 4\}$.

Then the total number of wedge points on each ring is the summarization of 4 quadrants. To evaluate this chromosome, each representation needs to be transformed into a TA model, representing the structure and characteristics of the TA airspace. In the simulator environment, the dynamic CDA trajectories for input flight plans are calculated according to this decoded TA model, with the help of aircraft performance parameters, and the airport configuration. The calculation process uses aircraft aerodynamic data and airspace configuration data to generate flight plans with less delay and smaller collision risk for arriving flights in a traffic scenario.

6.7.2 Population Design of Ground Event Scenarios

An event population represents a set of constrained ground events. To encode this into a chromosome, we first develop an 'event-table' data structure which contains all the ground resources (runway, taxiway and gate) along with all the possible events that can be associated with them. As illustrated in Figure 6.7, each combination of a surface resource with an event is given a unique event ID. For each resource, there are the following seven possible events:

- E0: Resource unavailable for heavy and medium aircraft;
- E1: Resource unavailable for heavy and light aircraft;
- E2: Resource unavailable for heavy aircraft;
- E3: Resource unavailable for light aircraft;
- E4: Resource unavailable for medium aircraft;
- E5: Resource unavailable for medium and light aircraft; and
- E6: Resource unavailable for heavy, medium and light aircraft.

Each individual in the event population has 10 chromosomes each representing for 1 event, and each chromosome has 3 genes. The first gene in the chromosome is the event-ID, randomly sampled in the interval [0, 1] and its value is then used to select the event-ID value from the event table in the decoding process.

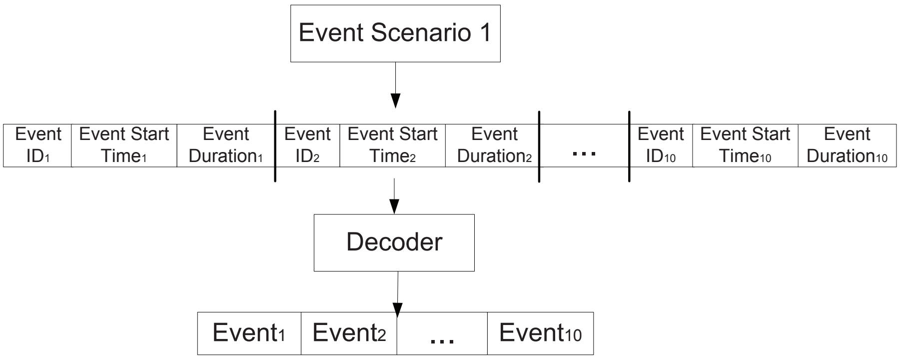


Figure 6.7: Event chromosome design showing genomes which encode event ID, event activation time and duration

Every event is also assigned an activation time and a duration for which it will be active. As illustrated in Figure 6.7, the second gene in each event chromosome is the event-start-time and the third is event-duration-time which are both randomly sampled in a pre-defined time interval. This chromosome is then translated into 10 events for a given TA design individual.

6.7.3 Fitness Function Design

With target at both efficiency and safety, the TeA airspace design evaluation is based on two objectives: averaged arrival delays for all flights, which is a measure of the induced delay due to the unavailability of air and ground resources; and averaged collision risk for all flights, which is a measure of the collision risk due to previous flights occupying wedges in their neighbourhood.

Since we attempt to identify the TA design which is able to cope better with ground events, the co-evolutionary process seeks to minimize values of both objectives in the system by evolving combinations of TA design and ground events which rank more lowly (generated a lower number of delays and smaller chance of collision) in the co-evolutionary process.

Efficiency objective – flight delay is calculated by averaging the differences between the ETA of flights from their OM to requested gates and their actual time of arrival (ATA) at actual gates from their respective OM (as in Equation 6.11).

$$ArrivalFlightDelay = \frac{[\Sigma_{i=1}^{M}(ATA_i - ETA_i)]_G^{OM}}{M} \tag{6.11}$$

where

- M = number of arrivals;
- ATA = actual time of arrival;
- ETA = estimated time of arrival;
- OM = outer marker; and
- G = gate.

The safety objective is represented by collision risk which is calculated by averaging the collision risks of assigned CDA trajectories in TA airspace for all arriving flights. The definition and calculation method of collision risk was explained in details in Section 6.5 of Chapter 5.

6.8 Experiment Design

6.8.1 Experimental Scenarios and Parameters

To get variation in arrival traffic distribution in terms of both density and direction, we designed the 3×5 repeated-measures (as Table 6.1) which yields the following 15 different experiment scenarios.

- ***Scenario 80/1Q*** consists of 80 arrivals in 3 hours coming from 1 quadrant.
- ***Scenario 100/1Q*** consists of 100 arrivals in 3 hours coming from 1 quadrant.
- ***Scenario 120/1Q*** consists of 120 arrivals in 3 hours coming from 1 quadrant.

Table 6.1: Experimental Design

	80 Arrivals in 3h	*100 Arrivals in 3h*	*120 Arrivals in 3h*
From 1 Quadrant	**Scenario 80/1Q**	**Scenario 100/1Q**	**Scenario 120/1Q**
From 2 Adjacent Quadrants	**Scenario 80/2Q**	**Scenario 100/2Q**	**Scenario 120/2Q**
From 2 Opposite Quadrants	**Scenario 80/2Q/opo**	**Scenario 100/2Q/opo**	**Scenario 120/2Q/opo**
From 3 Quadrants	**Scenario 80/3Q**	**Scenario 100/3Q**	**Scenario 120/3Q**
From 4 Quadrants	**Scenario 80/4Q**	**Scenario 100/4Q**	**Scenario 120/4Q**

- ***Scenario 80/2Q*** consists of 80 arrivals in 3 hours coming from 2 adjacent quadrants.
- ***Scenario 100/2Q*** consists of 100 arrivals in 3 hours coming from 2 adjacent quadrants.
- ***Scenario 120/2Q*** consists of 120 arrivals in 3 hours coming from 2 adjacent quadrants.
- ***Scenario 80/2Q/opo*** consists of 80 arrivals in 3 hours coming from 2 opposite quadrants.
- ***Scenario 100/2Q/opo*** consists of 100 arrivals in 3 hours coming from 2 opposite quadrants.
- ***Scenario 120/2Q/opo*** consists of 120 arrivals in 3 hours coming from 2 opposite quadrants.
- ***Scenario 80/3Q*** consists of 80 arrivals in 3 hours coming from 3 quadrants.
- ***Scenario 100/3Q*** consists of 100 arrivals in 3 hours coming from 3 quadrants.

- ***Scenario 120/3Q*** consists of 120 arrivals in 3 hours coming from 3 quadrants.
- ***Scenario 80/4Q*** consists of 80 arrivals in 3 hours coming from 4 quadrants.
- ***Scenario 100/4Q*** consists of 100 arrivals in 3 hours coming from 4 quadrants.
- ***Scenario 120/4Q*** consists of 120 arrivals in 3 hours coming from 4 quadrants.

These traffic distributions are generated using a probability distribution function. The two populations (TA design scenarios and event scenarios) co-evolve cooperatively and are represented by fixed-length real-valued genomes. The TA design scenario population size is 25. Each scenario constructs a TA model with certain number of way points on each quadrant of every ring in transition airspace. The dynamic CDA trajectories of input arrivals are generated based on this TA model. The event scenario population size is also 25 and each scenario consists of 10 events. The pre-defined time interval for the event-Start-Time is $[0, 12000]$ seconds and that for the event-duration-time is $[0, 1800]$ which means each event can be encountered by all flights in 3 hours and solved in 30 minutes. We run the experimental scenario 10 times using different seeds and apply tournament selection by elitism, single-point crossover with a probability of 1.0 and uniform mutation with a probability of 0.3. These parameters are chosen carefully after a number of sample runs. We allow a sufficient number of objective evaluations in each run for its evolution to become stable (the best solution does not change significantly).

6.9 Results & Discussion

Multi-objective cooperative co-evolution is used as the search methodology for evolving complex scenarios through incremental feedback from the simulation system. Firstly, we evaluate the performance of this methodology. The non-dominated

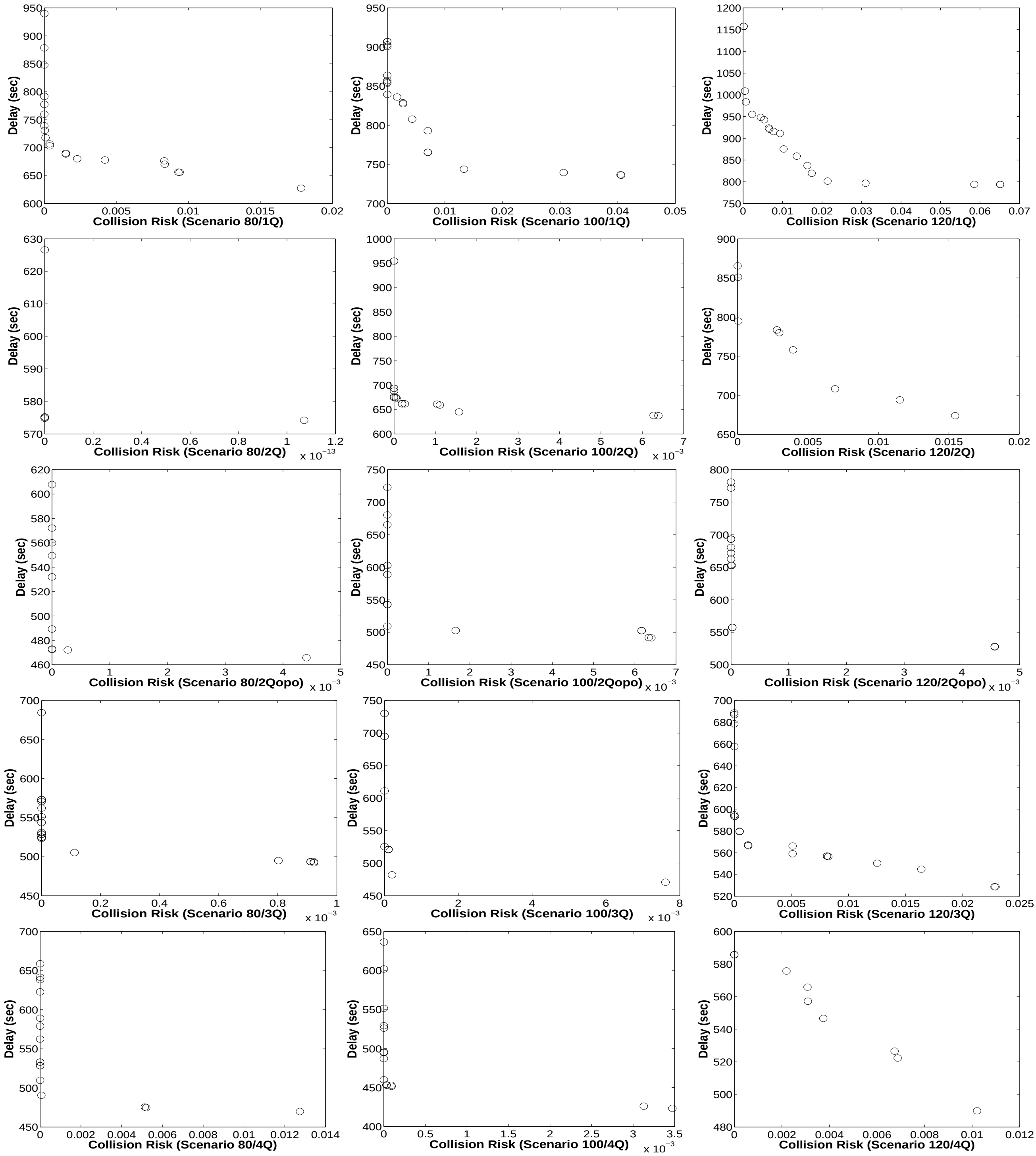


Figure 6.8: All solutions as obtained after 100 generation. The fifteen panels show results for fifteen different experimental scenarios. From left to right and top to bottom these problems are Scenario 80/1Q, Scenario 100/1Q, Scenario 120/1Q, Scenario 80/2Q, Scenario 100/2Q, Scenario 120/2Q, Scenario 80/2Q/opo, Scenario 100/2Q/opo, Scenario 120/2Q/opo, Scenario 80/3Q, Scenario 100/3Q, Scenario 120/3Q, Scenario 80/4Q, Scenario 100/4Q, Scenario 120/4Q,.

solutions of a sample run after 200 generations for each of the fifteen experimental scenarios are plotted in Figure 6.8. Despite some fluctuations due to the inherent stochastic nature of EAs and the problem, the obtained non-dominated solutions converged well and achieved a reasonable level of spread. This figure demonstrates the spread of solutions obtained in each scenario and effectiveness of our methodology to evolve ground events with better dynamic TA design to produce higher level of efficiency and safety under certain scenario. Since there is a compromise between the delay and the collision risk criticality, our multi-objective approach provides an analyst of the trade off between these two ATC priority – efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand.

In Figure 6.9, 6.10, 6.11, 6.12 and 6.13, we plot the objective values of the extreme points of the non-dominated set at each generation. These figures demonstrate how the evolutionary pressure in our algorithm works on the solutions over time. It is shown that the proposed methodology has the power to drive both safety and efficiency of solutions to a higher level.

It is also noticed that, for most cases, the solutions converged to a safer and more efficient level when the traffic density in time is lower. For instance, in Figure 6.9, the solutions for the scenario, which involves 80 arrivals in 3 hours coming from 1 quadrant, reached and maintained around 600 seconds delay before 120 generations, while those for another scenario (involving 120 arrivals in 3 hours from 1 quadrant) still suffered 800 seconds delay after 200 generations. On the other hand, the solutions, for a set of traffic coming from a narrower direction, converged to a less safe and efficient level. Take Figure 6.9 and 6.10 for example, the solutions for Scenario 80/1Q had more than 600 seconds delay after 200 generations; while those for Scenario 80/2Q had less than 600 seconds delay before 100 generations.

We then analyze the solutions at the extreme points (points corresponding to the best performance on each objective function) of each set of non-dominated solutions after 200 generations. Table 6.2 shows objective values of points with best performance on safety objective (collision risk) while Table 6.3 shows those of points with best performance on efficiency objective (delay). It is noted that

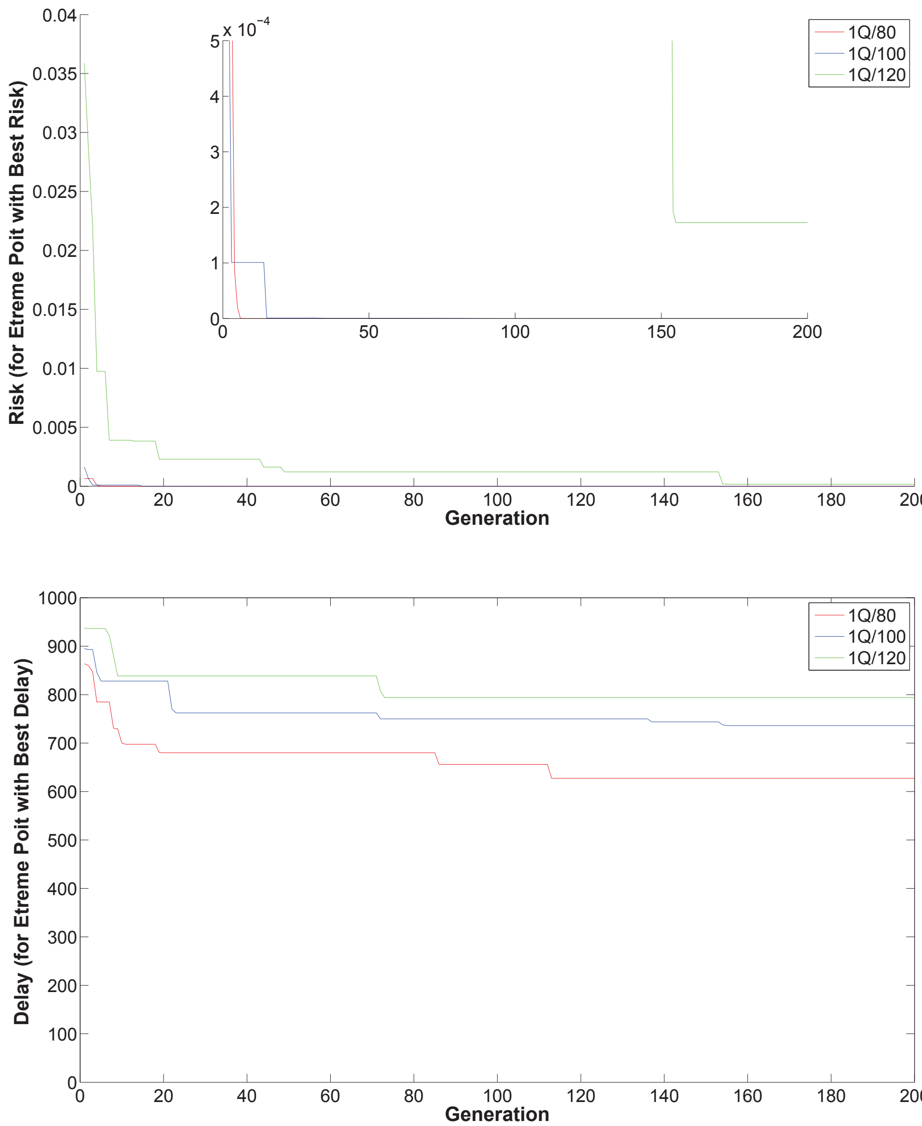


Figure 6.9: Two figures corresponding to each of extreme points evolved over time for Scenario 1Q (The figure on top is extreme points with best performance on safety objective (lowest collision risk) and the one on bottom is with best performance on efficiency objective (lowest delay))

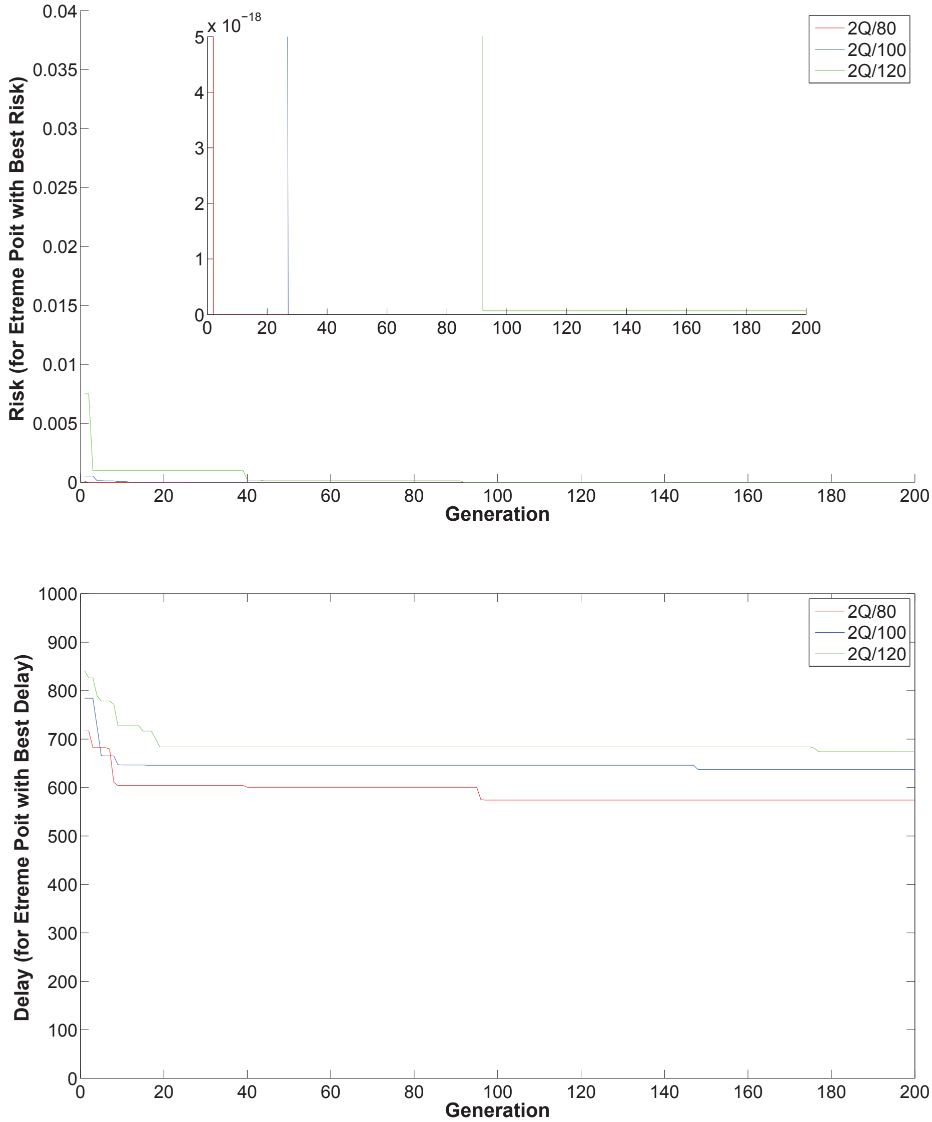
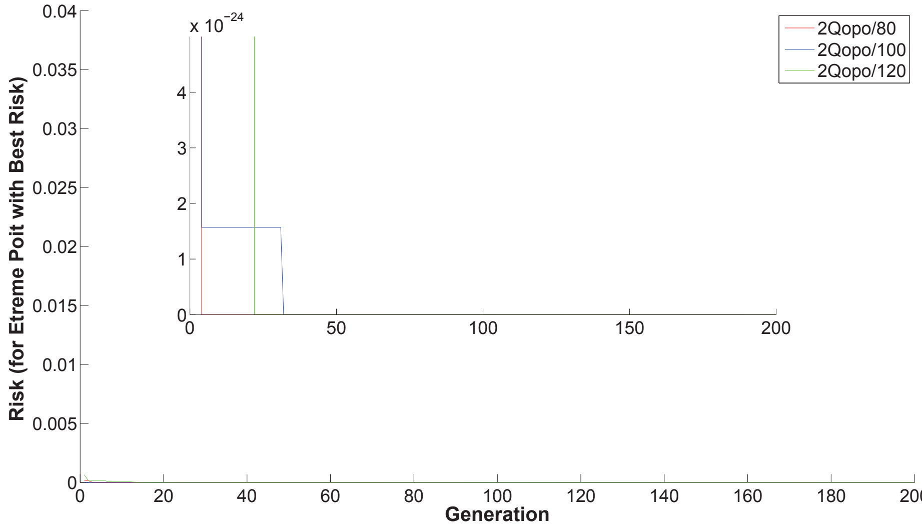


Figure 6.10: Two figures corresponding to each of extreme points evolved over time for Scenario $2Q$ (The figure on top is extreme points with best performance on safety objective (lowest collision risk) and the one on bottom is with best performance on efficiency objective (lowest delay))



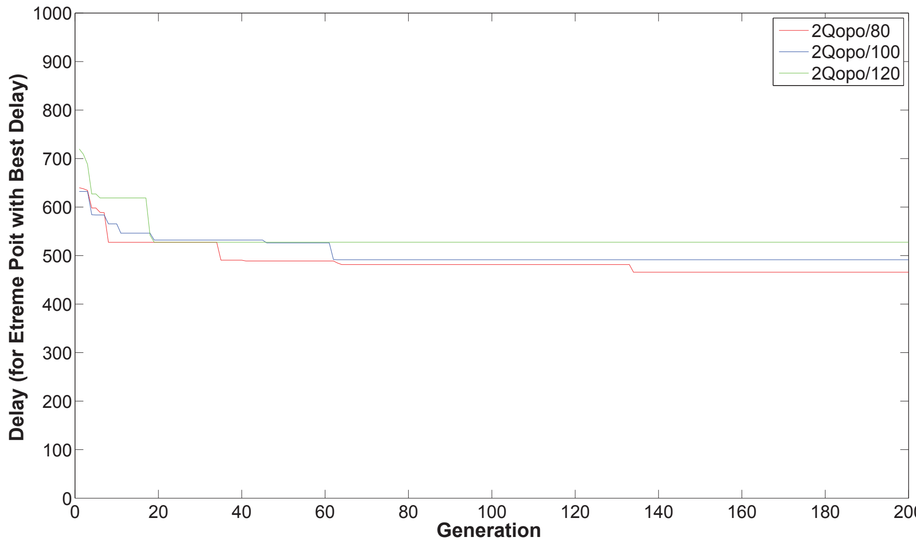


Figure 6.11: Two figures corresponding to each of extreme points evolved over time for Scenario $2Q/opo$ (The figure on top is extreme points with best performance on safety objective (lowest collision risk) and the one on bottom is with best performance on efficiency objective (lowest delay))

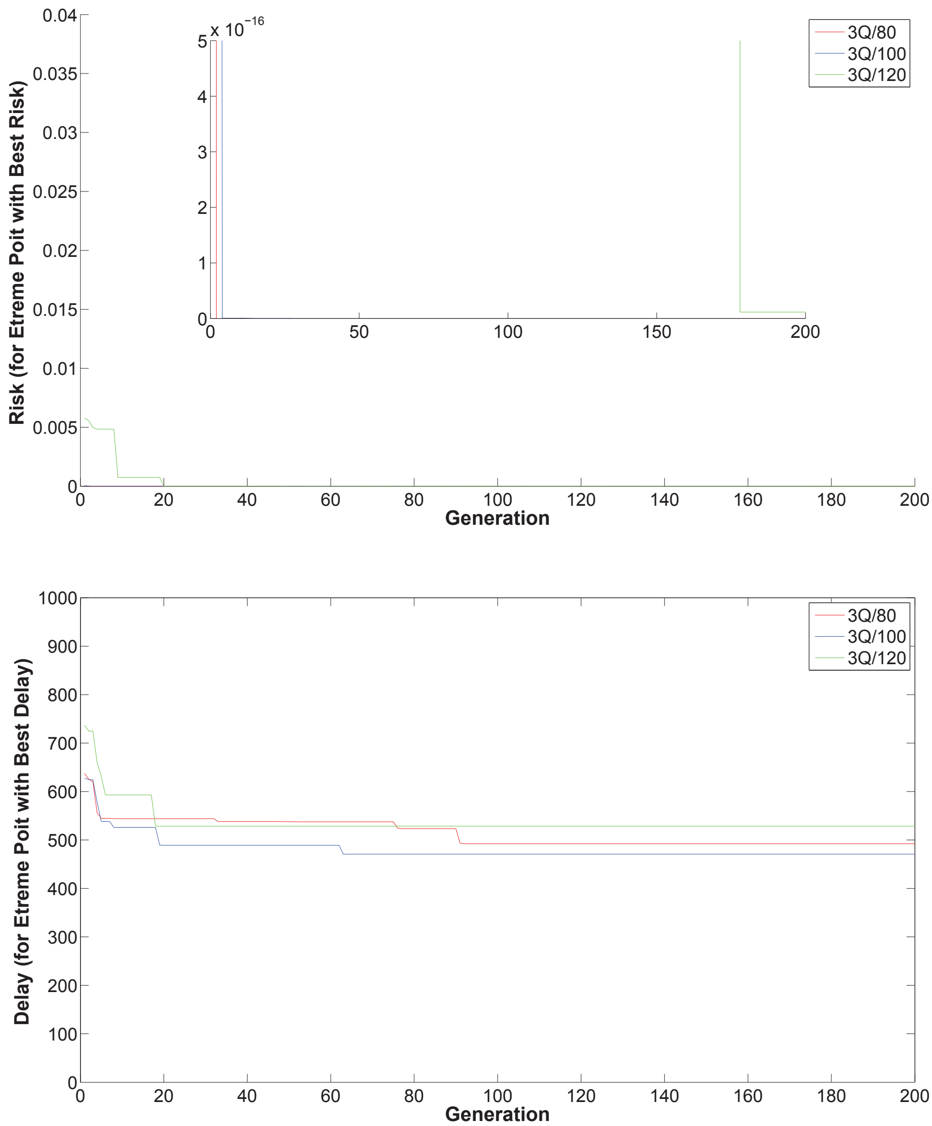
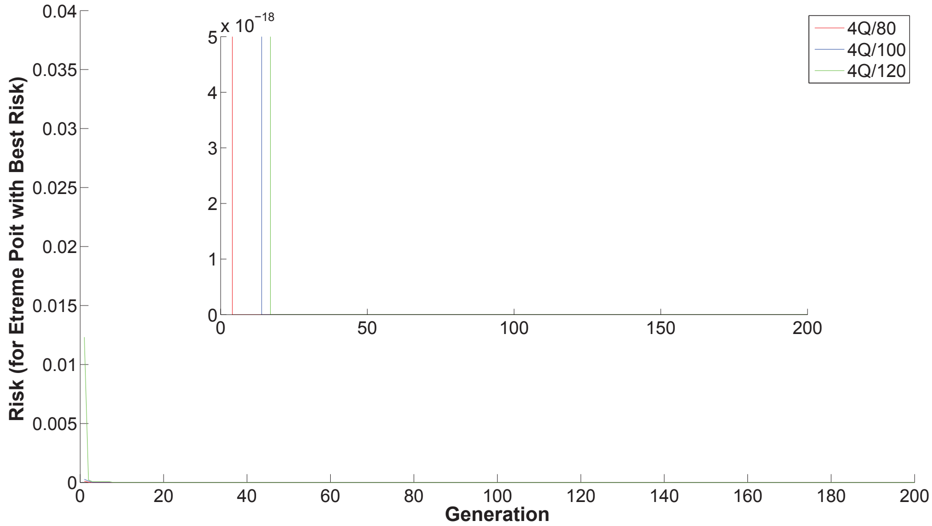


Figure 6.12: Two figures corresponding to each of extreme points evolved over time for Scenario $3Q$ (The figure on top is extreme points with best performance on safety objective (lowest collision risk) and the one on bottom is with best performance on efficiency objective (lowest delay))



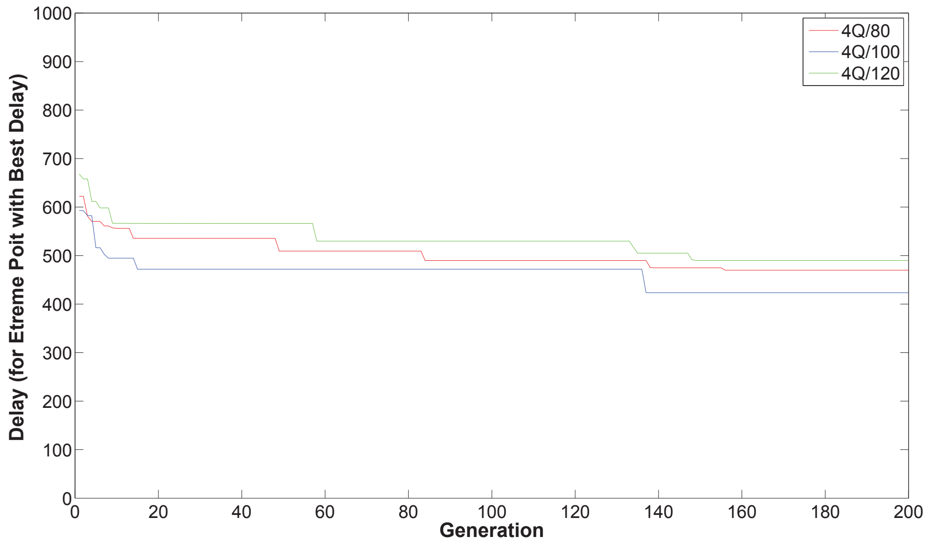


Figure 6.13: Two figures corresponding to each of extreme points evolved over time for Scenario $4Q$ (The figure on top is extreme points with best performance on safety objective (lowest collision risk) and the one on bottom is with best performance on efficiency objective (lowest delay))

Table 6.2: Objective Values $(Delay; CR)$ of Points with Best Performance on Safety Objective (Collision Risk)

$(Delay; CR)$	**80 Arrivals in 3h**	**100 Arrivals in 3h**	**120 Arrivals in 3h**
From 1 Quadrant	$(939.79; 3.99E-43)$	$(906.99; 1.81E-40)$	$(1157.76; 1.72E-04)$
From 2 Adjacent Quadrants	$(626.58; 2.16E-45)$	$(954.38; 1.04E-44)$	$(865.29; 6.57E-20)$
From 2 Opposite Quadrants	$(607.78; 5.88E-47)$	$(723.01; 1.35E-44)$	$(780.8; 7.6E-44)$
From 3 Quadrants	$(684.48; 6.5E-49)$	$(729.99; 8.2E-45)$	$(688.65; 1.16E-17)$
From 4 Quadrants	$(658.8; 7.59E-46)$	$(636.4; 3.92E-45)$	$(585.76; 1.34E-23)$

Table 6.3: Objective Values $(Delay; CR)$ of Points with Best Performance on Efficiency Objective (Delay)

$(Delay; CR)$	**80 Arrivals in 3h**	**100 Arrivals in 3h**	**120 Arrivals in 3h**
From 1 Quadrant	$(627.45; 1.78E-02)$	$(736.1; 4.01E-02)$	$(793.73; 6.51E-02)$
From 2 Adjacent Quadrants	$(574.16; 1.07E-13)$	$(637.36; 6.38E-03)$	$(673.94; 1.54E-02)$
From 2 Opposite Quadrants	$(465.72; 4.4E-03)$	$(491.36; 6.4E-03)$	$(527.51; 4.5E-03)$
From 3 Quadrants	$(492.26; 9.23E-04)$	$(470.76; 7.62E-03)$	$(528.52; 2.28E-02)$
From 4 Quadrants	$(469.83; 1.2E-02)$	$(423.4; 3.47E-03)$	$(489.94; 1.02E-02)$

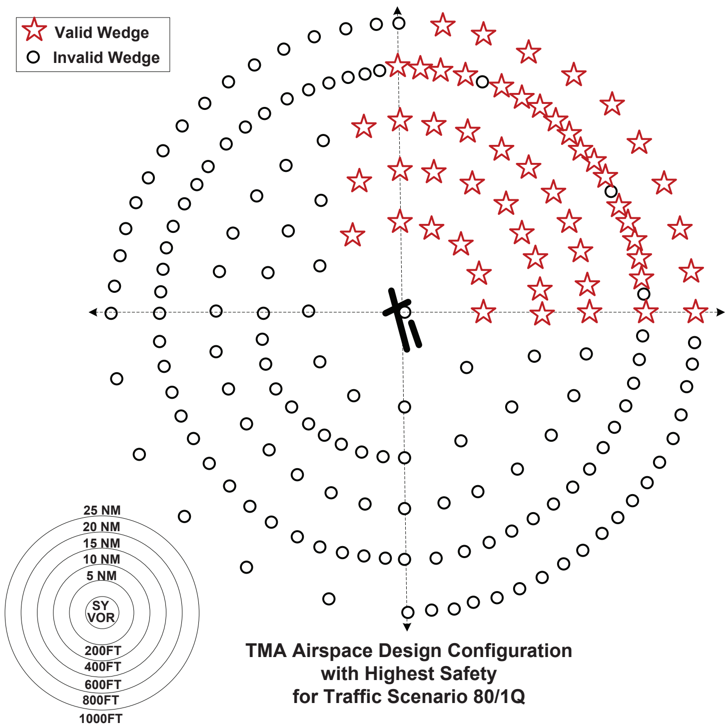
traffic scenarios with various arrival traffic distribution in terms of both density and direction have different solutions, which proves that an appropriate TA design strategy should be considered with a specific traffic scenario. Traffic scenarios with higher density experienced heavier delays (or higher collision risks) than those with lower density. This is straightforward. Traffic scenarios with narrower direction experienced heavier delays (or higher collision risks) than those with wider direction. This is also self-evident. The results also illustrate that the traffic scenario from 2 adjacent quadrants experienced heavier delays and higher collision risks than that from 2 opposite quadrants. Therefore for the sake of efficiency and safety, air traffic controllers should prioritize the TA design strategy with 2 opposite quadrants over the one with 2 adjacent quadrants.

From Figure 6.14 to Figure 6.28, the two TeA airspace design configurations corresponding to two extreme points, after 200 generations, for 15 scenarios are visualized. The red stars in each diagram represent the valid wedges which are actually used in aircrafts' trajectories; while the invalid wedges in black circle represent the wedge points which are never used by any aircraft. Based on the fixed STARs currently serving Sydney airport as shown in Figure 6.29, the proposed configuration diagrams can practically assist TeA terminal routes designers who are currently working on defining and developing pre-defined terminal arrival trajectories.

Table 6.4: Most Frequent Event Type according to Points with Best Performance on Safety Objective (Collision Risk)

$(EventType)$	**80 Arrivals in 3h**	**100 Arrivals in 3h**	**120 Arrivals in 3h**
From 1 Quadrant	Gate	Taxiway	Taxiway and Gate
From 2 Adjacent Quadrants	Gate	Taxiway	Taxiway
From 2 Opposite Quadrants	Taxiway	Taxiway	Taxiway
From 3 Quadrants	Taxiway and Gate	Taxiway	Taxiway and Gate
From 4 Quadrants	Taxiway	Gate	Taxiway

To better understand the TeA airspace design configuration proposed above, we then have a look at their corresponding ground events scenario. Table 6.4 and 6.5 show the most frequent event type happened according to the extreme points of each set of non-dominated solutions after 200 generations. According to the



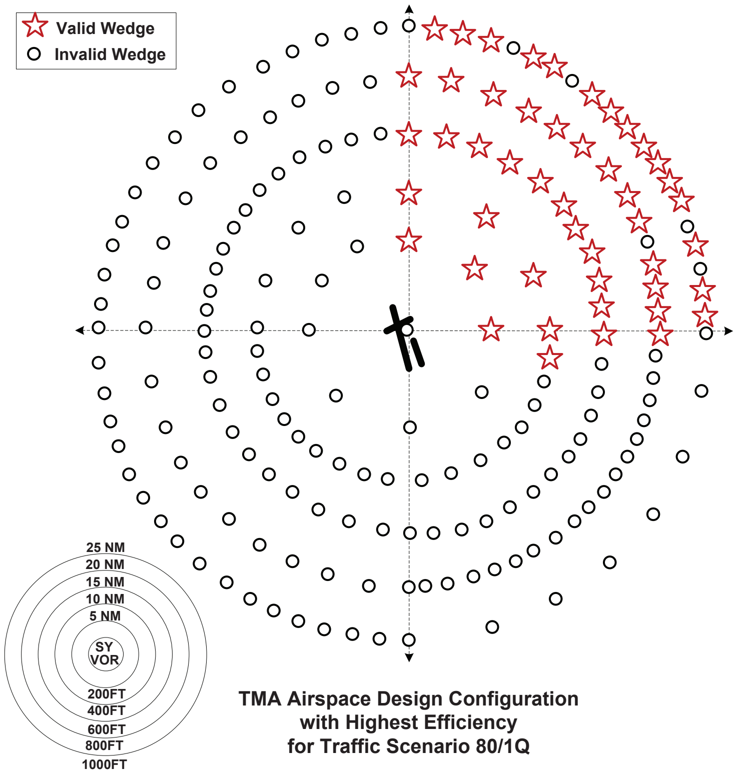
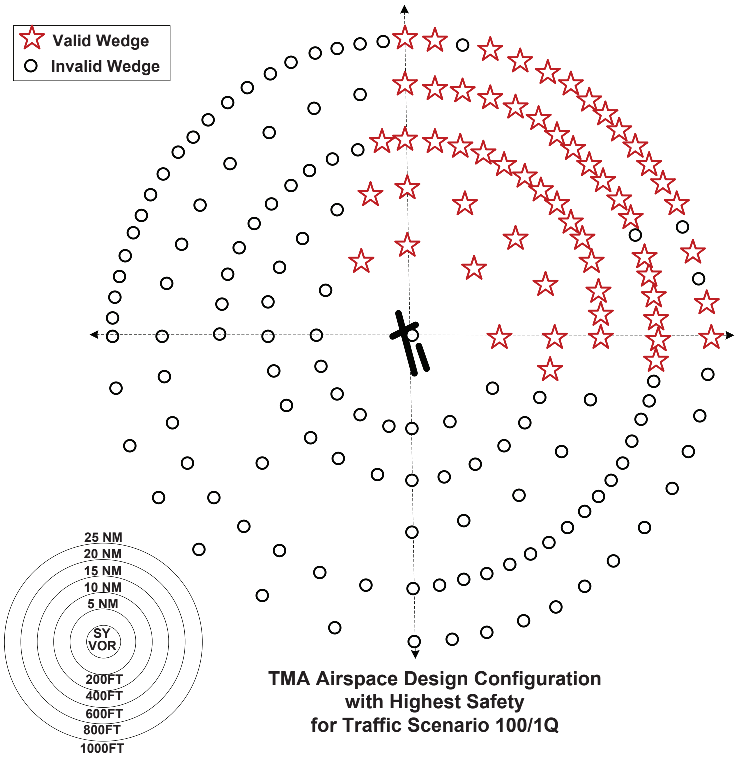


Figure 6.14: Two figures corresponding to each of the extreme points generated in one run for Scenario 80/1Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



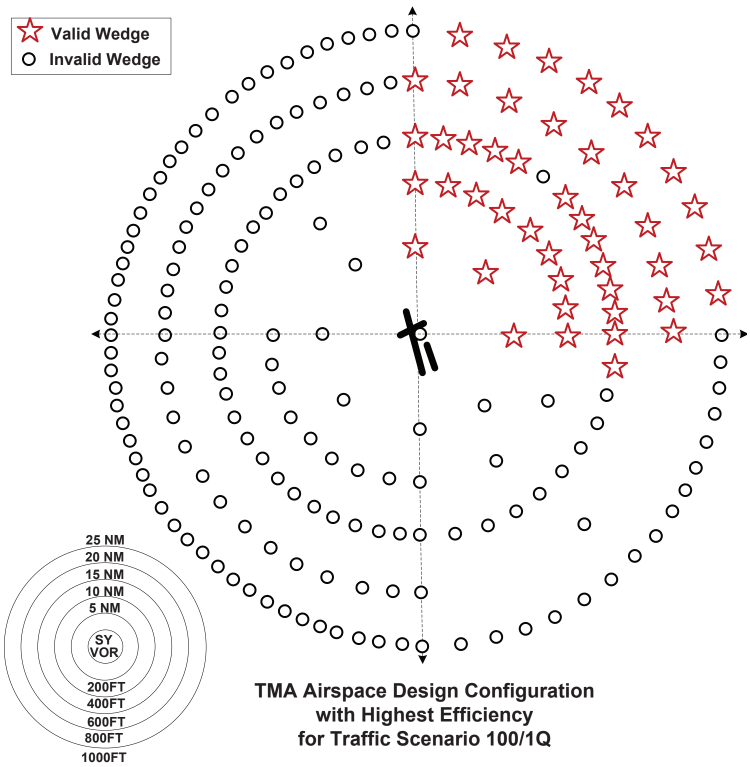


Figure 6.15: Two figures corresponding to each of the extreme points generated in one run for Scenario 100/1Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)

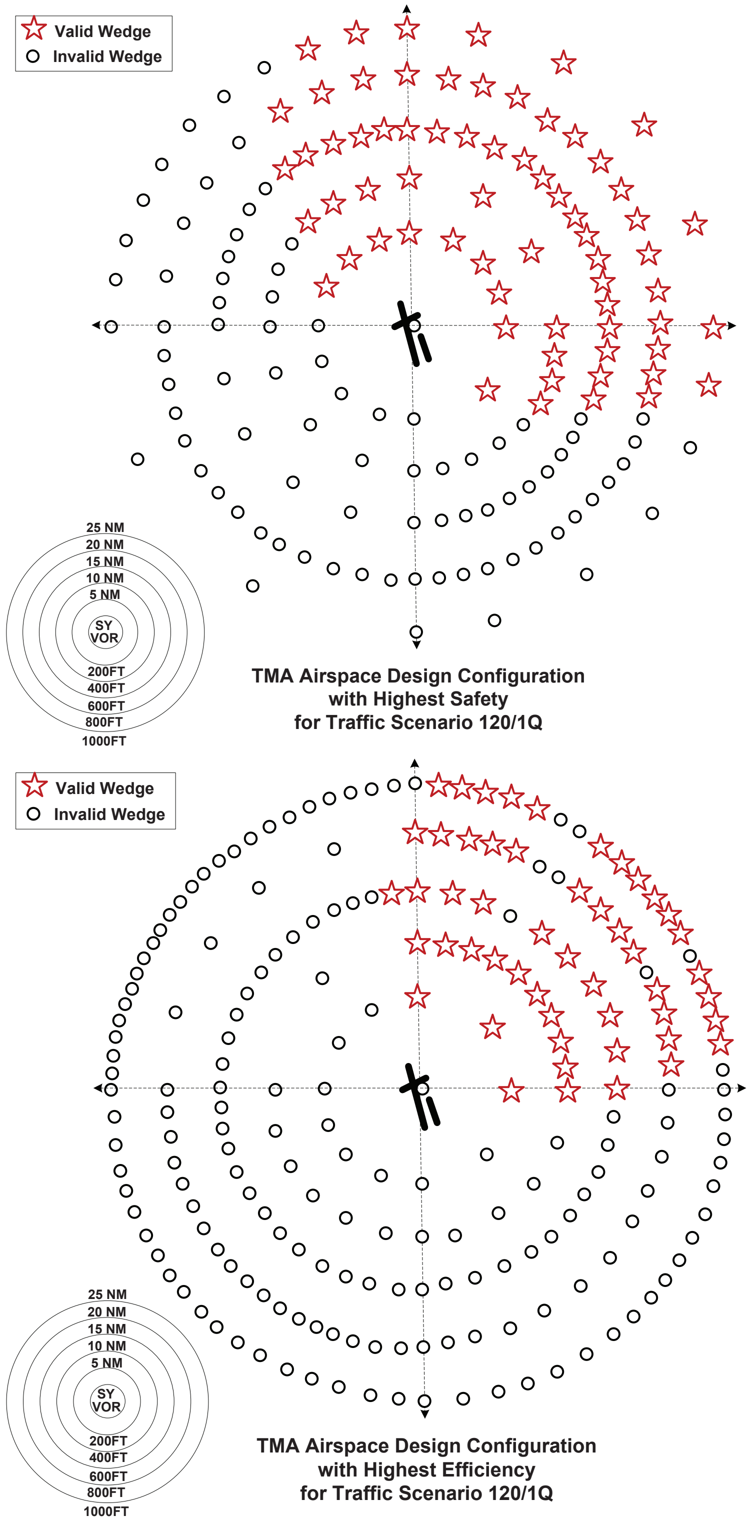
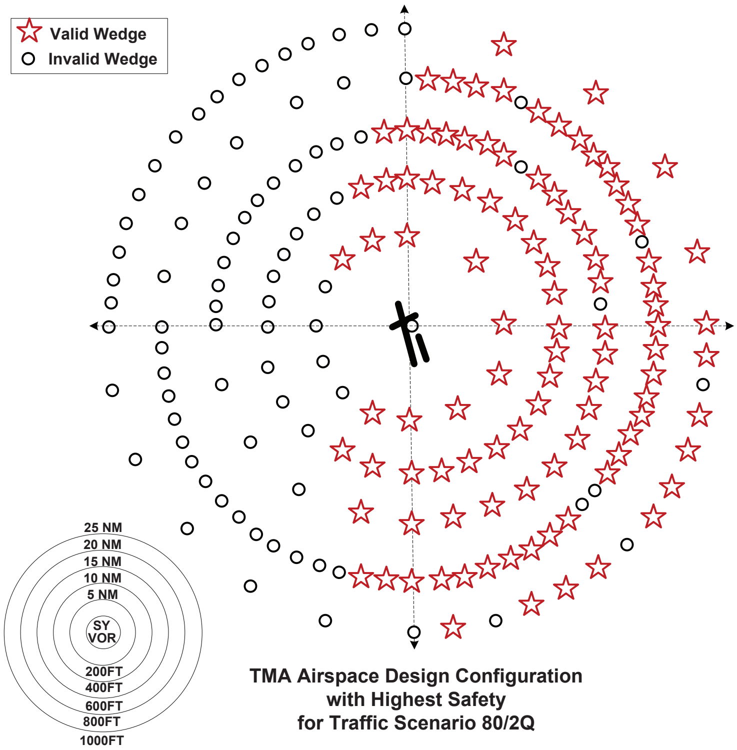


Figure 6.16: Two figures corresponding to each of the extreme points generated in one run for Scenario 120/1Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



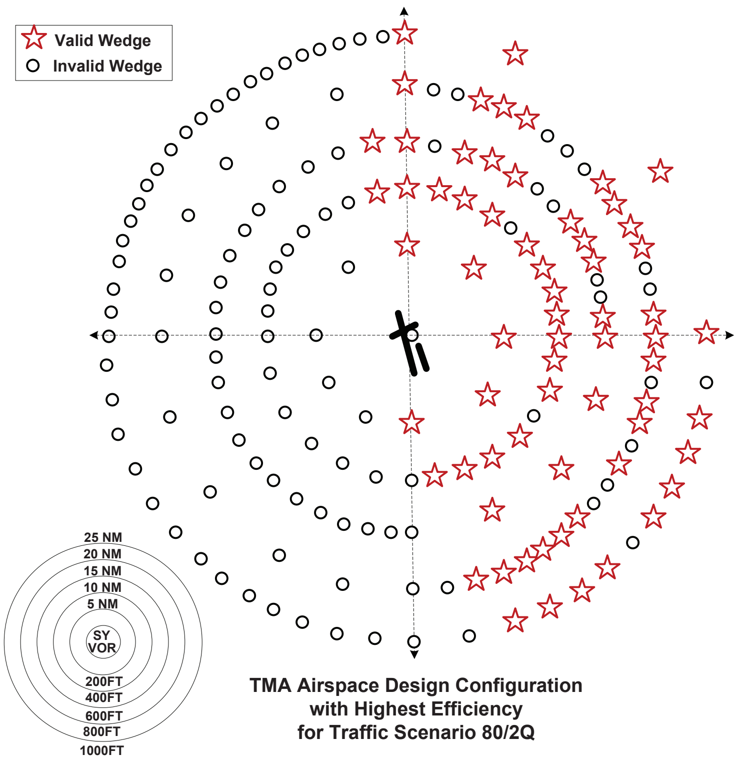
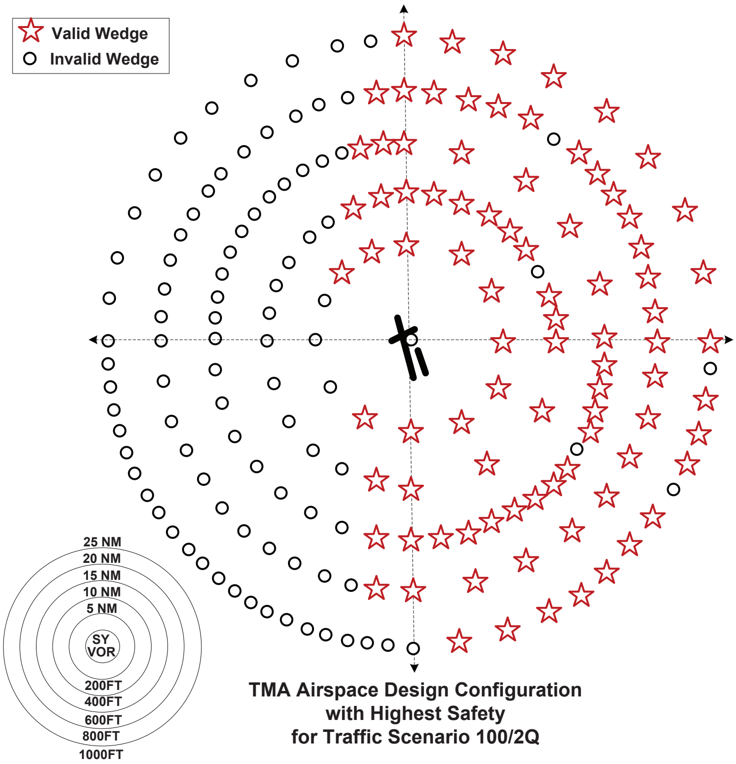


Figure 6.17: Two figures corresponding to each of the extreme points generated in one run for Scenario $80/2Q$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



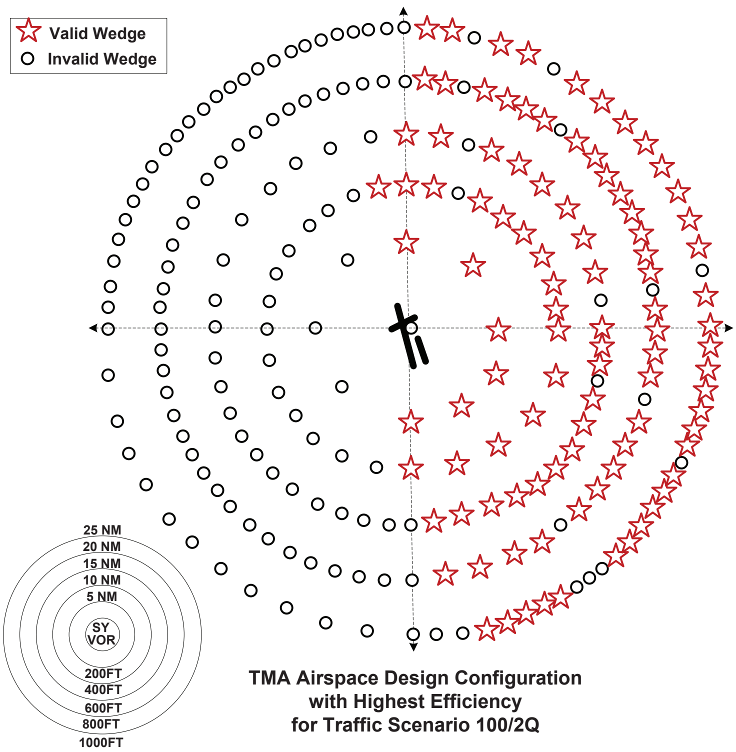
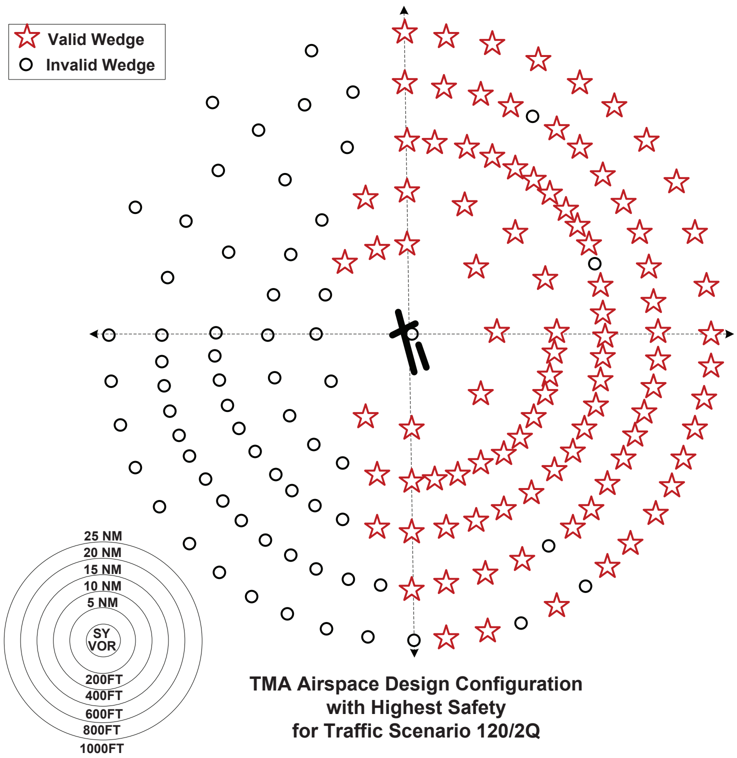


Figure 6.18: Two figures corresponding to each of the extreme points generated in one run for Scenario $100/2Q$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



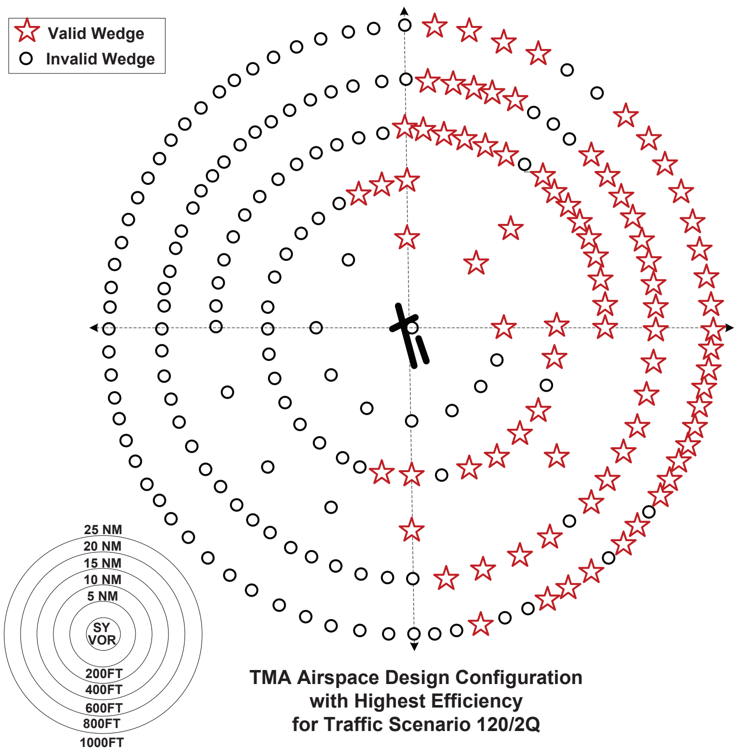
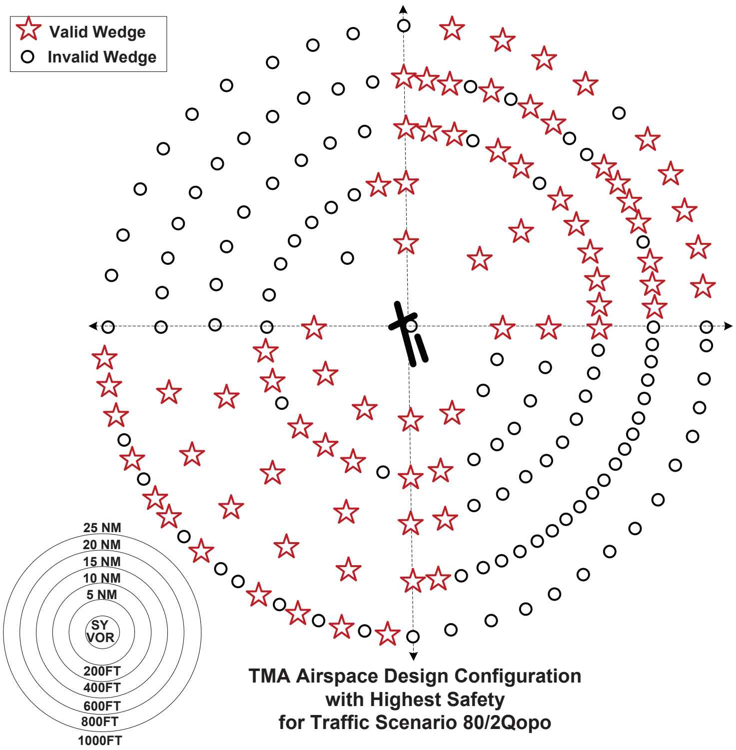


Figure 6.19: Two figures corresponding to each of the extreme points generated in one run for Scenario 120/2Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



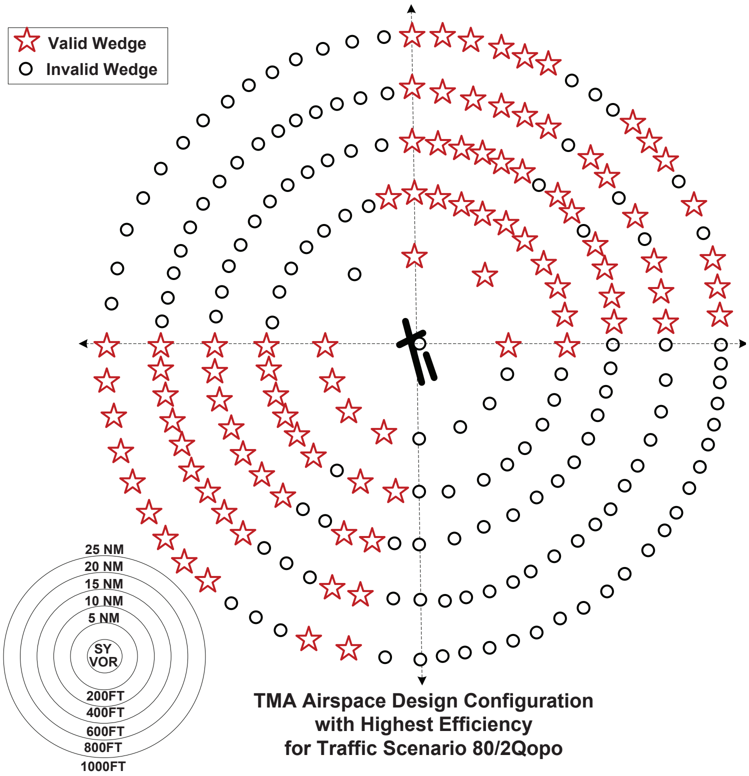


Figure 6.20: Two figures corresponding to each of the extreme points generated in one run for Scenario $80/2Q/opo$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)

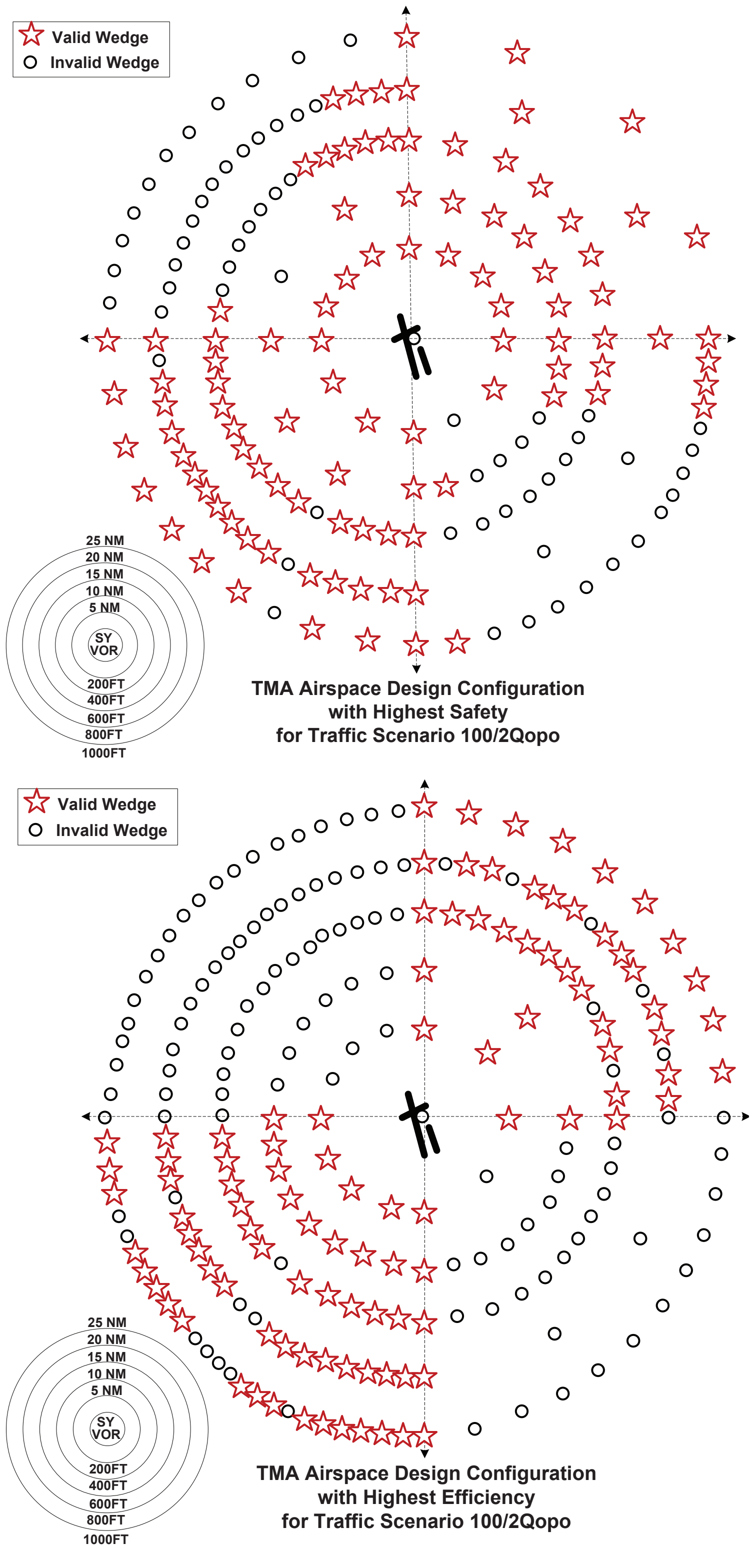
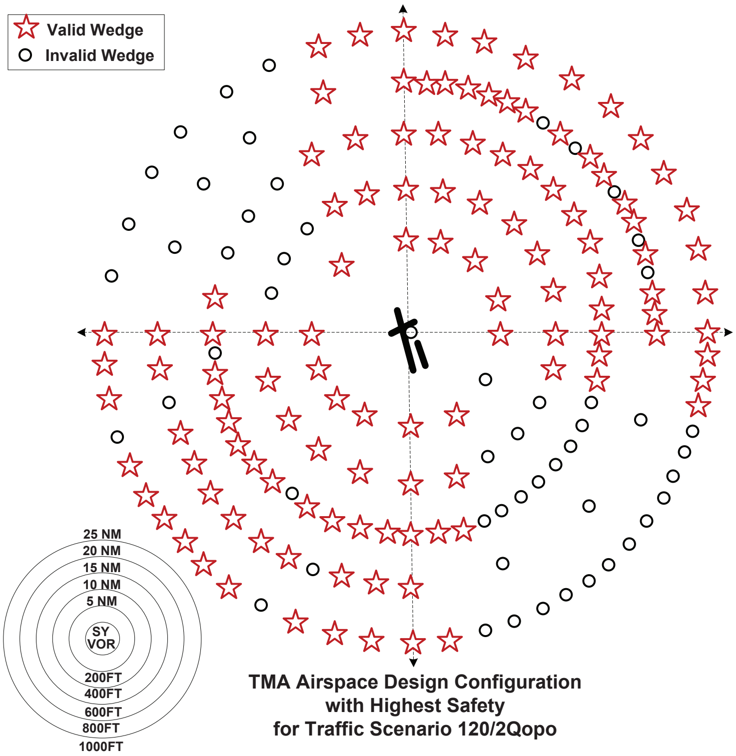


Figure 6.21: Two figures corresponding to each of the extreme points generated in one run for Scenario $100/2Q/opo$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



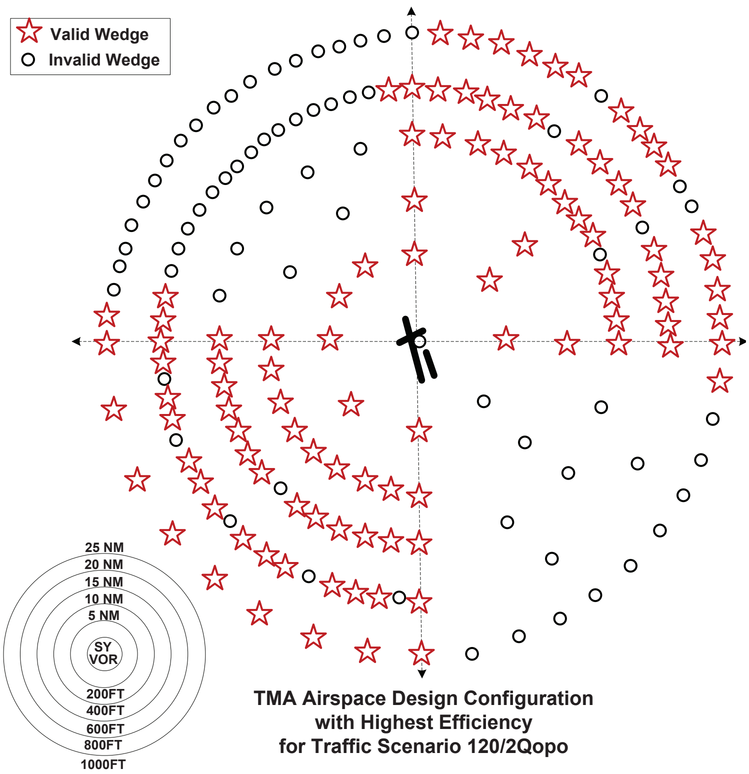


Figure 6.22: Two figures corresponding to each of the extreme points generated in one run for Scenario $120/2Q/opo$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)

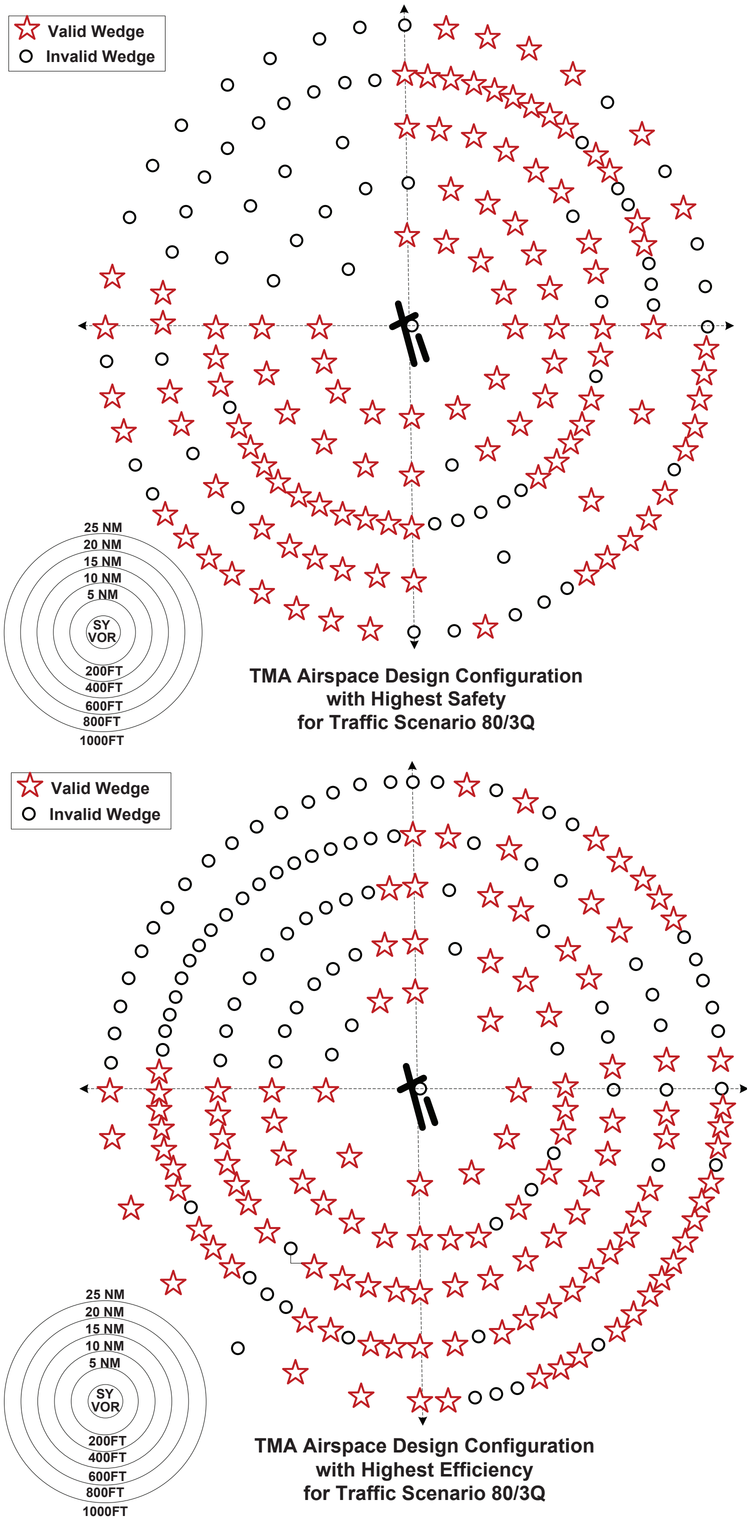


Figure 6.23: Two figures corresponding to each of the extreme points generated in one run for Scenario $80/3Q$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)

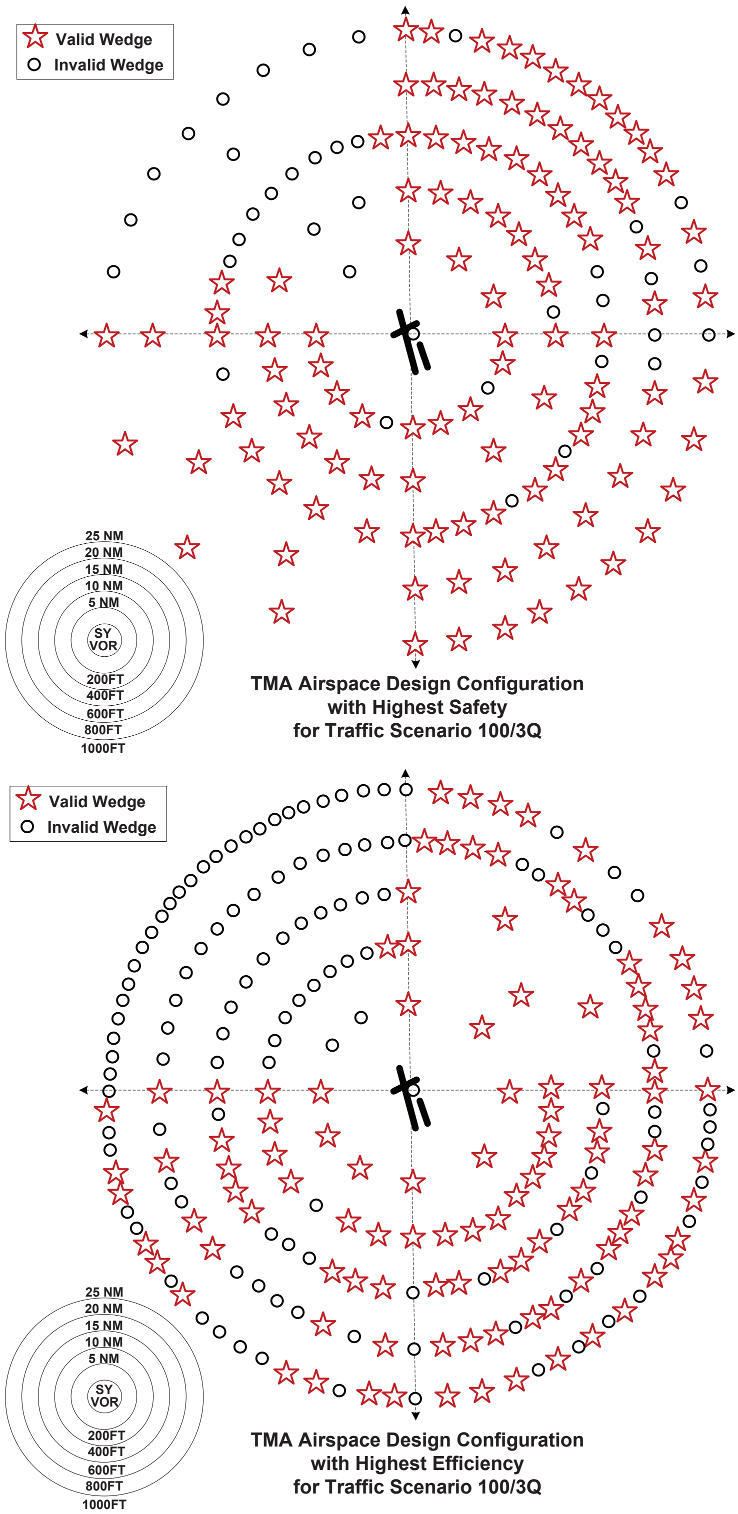


Figure 6.24: Two figures corresponding to each of the extreme points generated in one run for Scenario 100/3Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)

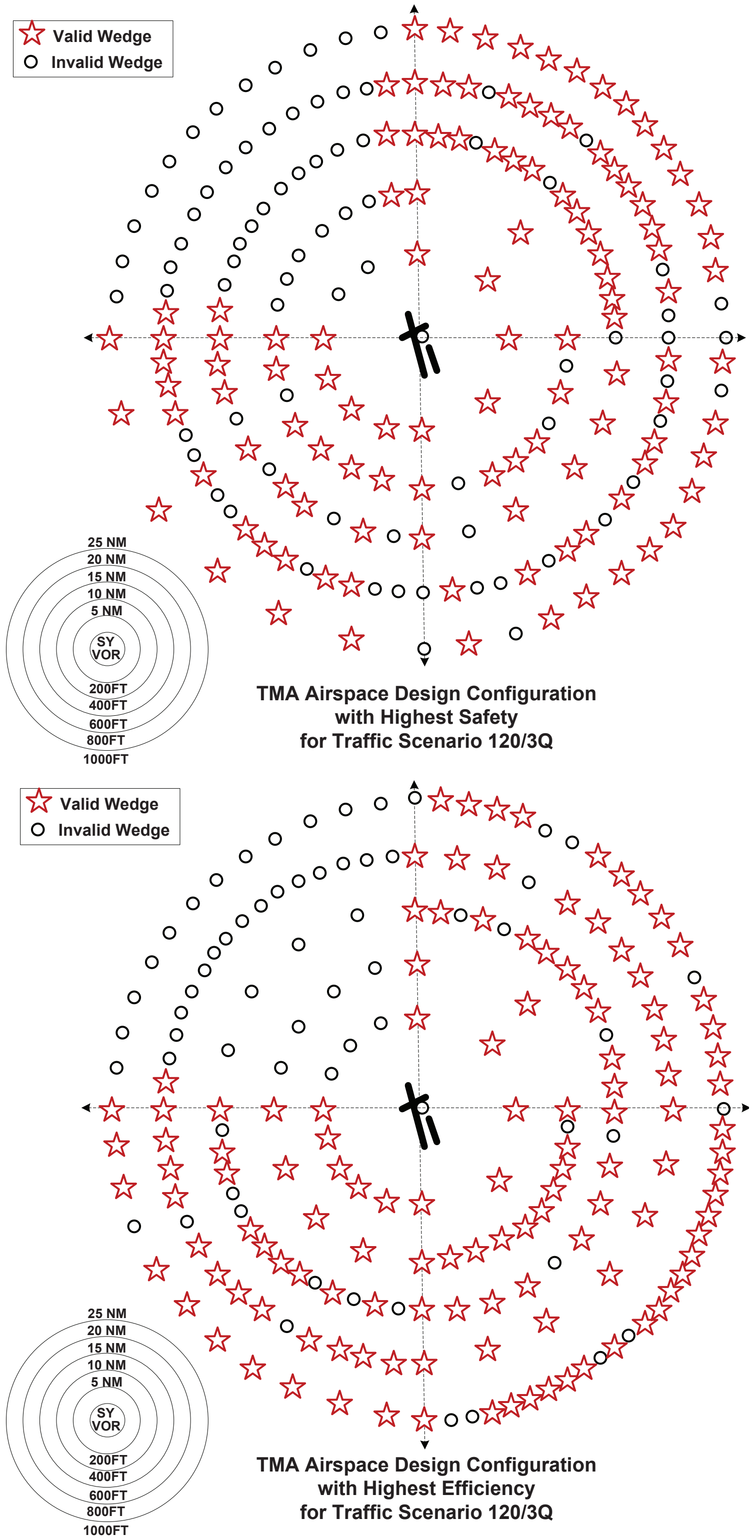
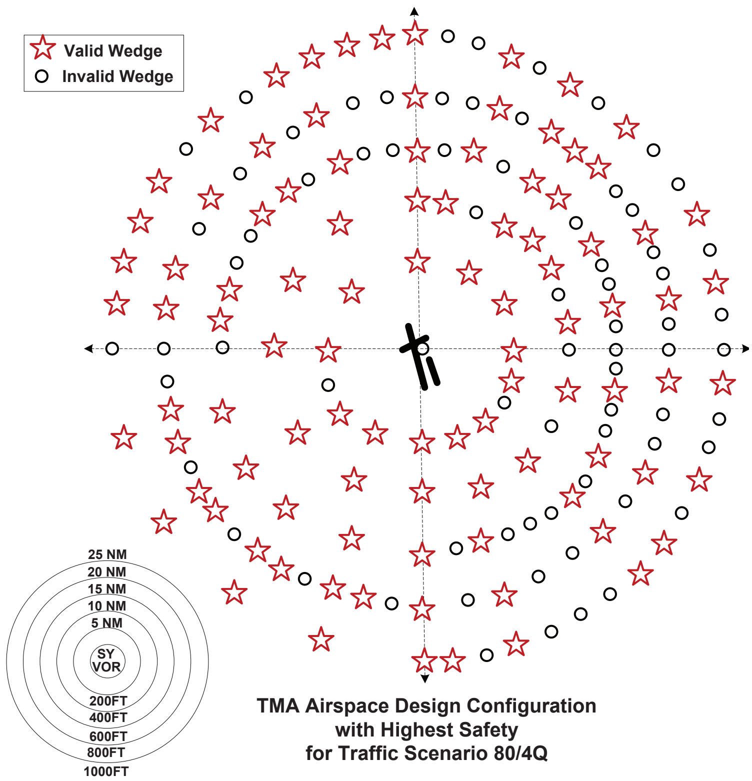


Figure 6.25: Two figures corresponding to each of the extreme points generated in one run for Scenario 120/3Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



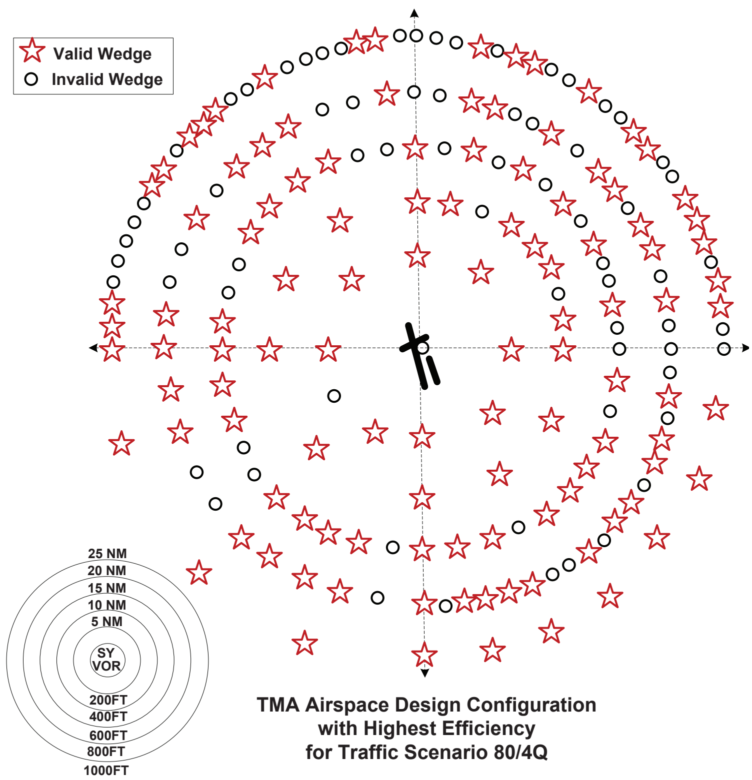
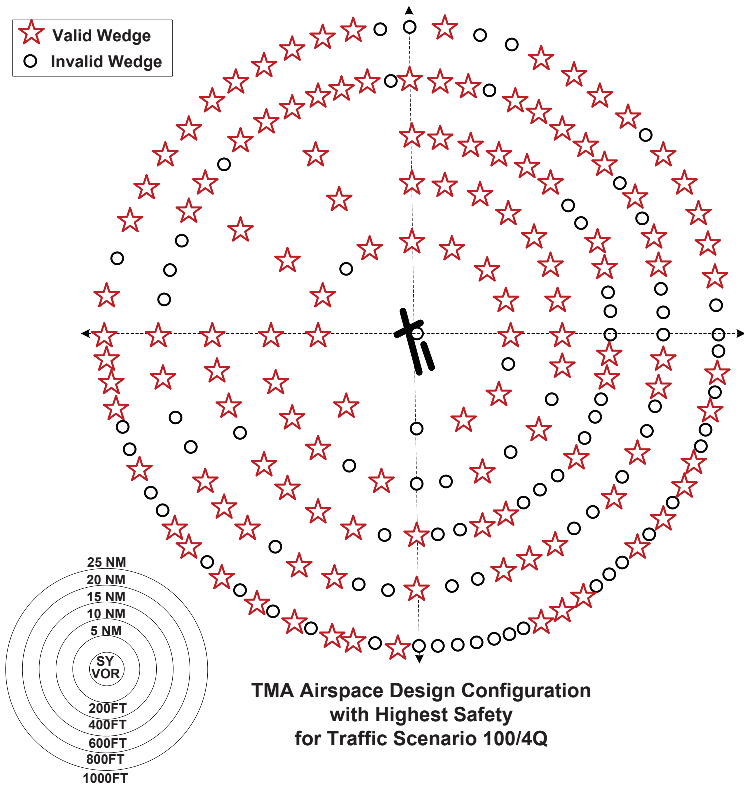


Figure 6.26: Two figures corresponding to each of the extreme points generated in one run for Scenario $80/4Q$ (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



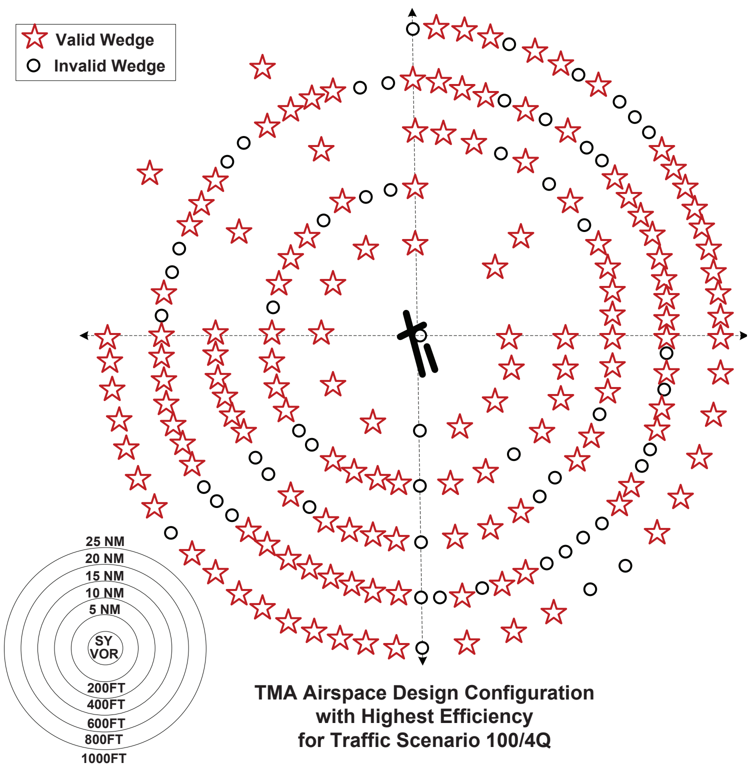
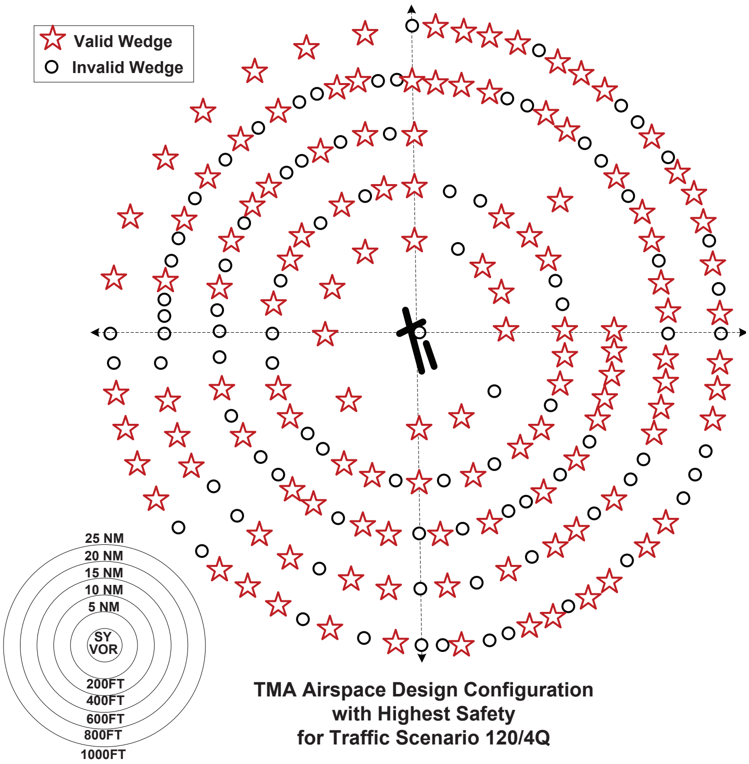


Figure 6.27: Two figures corresponding to each of the extreme points generated in one run for Scenario 100/4Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)



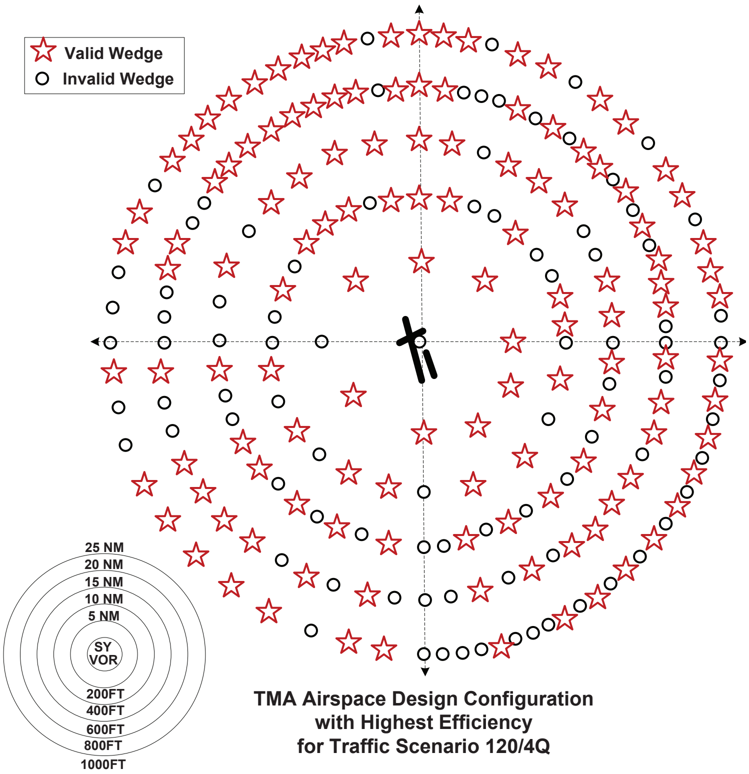


Figure 6.28: Two figures corresponding to each of the extreme points generated in one run for Scenario 120/4Q (The figure on top is the TeA airspace design configuration with highest safety and the one on bottom is the TeA airspace design configuration with highest efficiency)

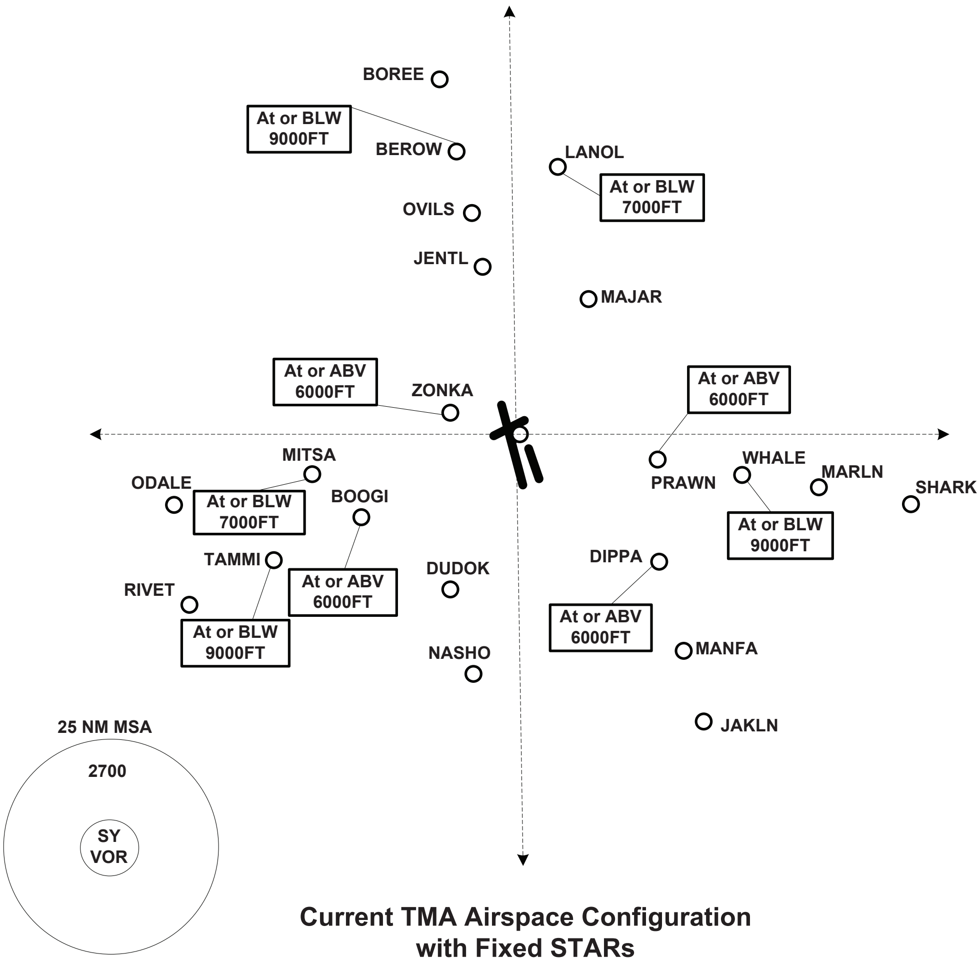


Figure 6.29: The fixed TeA airspace configuration for Sydney airport

Table 6.5: Most Frequent Event Type according to Points with Best Performance on Efficiency Objective (Delay)

(*EventType*)	**80 Arrivals in 3h**	**100 Arrivals in 3h**	**120 Arrivals in 3h**
From 1 Quadrant	Gate	Taxiway	Gate
From 2 Adjacent Quadrants	Taxiway	Taxiway	Gate
From 2 Opposite Quadrants	Taxiway and Gate	Taxiway	Taxiway
From 3 Quadrants	Taxiway	Taxiway	Taxiway
From 4 Quadrants	Runway and Gate	Taxiway and Gate	Runway and Taxiway

design of the event table (as in Table 4.1), the proportions among runway, taxiway and gate are 6 : 34 : 23. Since taxiways are the most frequent resources, it is not surprising that most of the scenarios encounter ground events on taxiways. However, as shown in the ground network model (see Figure 3.6), there are some critical bottleneck resources, such as taxiway G and B, heavily constrain capacity of the ground network. Therefore, the solutions which have least delay encounter less ground events on taxiways as shown in Table 6.5.

6.10 Chapter Summary

In this chapter, a simulation-based co-evolutionary computational red teaming environment for multiple objectives – multi-objective CCRT – is proposed, in order to generate the scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety.

An air traffic simulation system with a novel representation of a TeA airspace design concept considering the interactions from dynamic ground events is presented for system-level modelling of integrated TeA concepts. This simulation environment is inspired by the dynamic CDA model presented in a previous chapter and extend that concept to a flexible TeA airspace model which involves a novel way of defining a collision risk metric, which is derived from the probabilistic nature of an aircraft's operational position.

The multi-objective co-operative co-evolution algorithm, developed and tested in a previous chapter, was used as the optimization search engine to takes into account the impact of the dynamically constrained ground resources, in order to accomplish the ground-air integration.

We conducted a series of computational experiments with different air traffic density and direction scenarios. The parameters impacting on the delay and collision risk performances were co-evolved with our synthetic model of a TeA system for airspace design (TeA airspace, runways, taxiways and gates). Our methodology demonstrated traffic scenarios with various arrival traffic distributions in terms of both density and direction have different solutions, which proves that an appropriate TeA airspace design strategy should be considered with a specific traffic scenario.

The results also reveal that traffic scenarios from two adjacent quadrants experienced heavier delays and higher collision risks than that from two opposite quadrants. Therefore for the sake of efficiency and safety, air traffic controllers should prioritize the TA design strategy with two opposite quadrants over the one with two adjacent quadrants.

The proposed multi-objective approach also provide an analyst with the trade-off between these two ATC priorities efficiency and safety; thus solutions can be selected based on the criticality for meeting the demand with an acceptable risk level.

Chapter 7

Conclusion

7.1 Summary of Results

This thesis presented a systematic study of understanding (through modeling), evaluating and dynamically designing TeA airspace, in order to alleviate the growing capacity/demand imbalance in future ATM. An integrated air traffic simulation system with a novel representation of an integrated TeA was presented for system-level modeling of current and future TeA concepts. This simulator combines the air and ground subsystems and provide a proper operational environment for processing arrivals and/or departures. A simulation-based co-evolutionary computational environment, named as CCRT, was developed for evaluating advanced TeA airspace concepts and understanding the TeA system vulnerabilities. The interactions between traffic distributions and constrained ground resources (including runways, taxiways and gates) are co-evolved with each other and considered from the perspective of identifying inefficiencies, with the integration of arrival and departure operations. By evaluating these interactions, we are able to reveal "improvement opportunities" in the implementation of future TeA airspace concepts and, thereby, understand major bottlenecks which cause system inefficiencies.

A multi-objective co-operative co-evolutionary methodology was proposed as another search engine of the CCRT framework, in order to solve complex TeA problems with multiple conflicting objectives. An air traffic simulator representing an original novel TeA airspace design concept, while considering the interactions from dynamic ground events, was presented. The proposed TeA airspace concept involves a measure of collision risks derived from the probabilistic nature of aircraft's performance. The multi-objective CCRT was applied to generate scenario-specific TeA airspace design strategies, that were able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety. The multi-objective CCRT also provided an analyst with the trade-off between two ATC priority: efficiency and safety, so that solutions can be selected based on the criticality level of meeting the demand.

Overall, the main findings of the research introduced in the thesis can be summarized as follows:

- The proposed CCRT framework was proven to successfully evaluate advanced TeA airspace concepts and discover TeA system vulnerabilities, by evolving the reciprocal interactions of arrivals and departures using a shared ground-air network. The results demonstrated the power of this methodology in objectively evaluating TeA concepts in the integrated TeA system and synthesizing an overall situational awareness picture that decision makers can utilize.
- Our analysis suggested and demonstrated that, when making efforts to improve ATM performance on a system level, one cannot separate the air-side complex from the ground-side complex, or divide the arrival process from the departure process.
- In the presence of constrained ground resources, the dynamic CDA model was able to provide controllers and airspace users benefits to further improve the TeA system's throughput capacity as well as to minimize flight delays. The quantified performance evaluation increases decision maker's confidence to support this transition. The results also revealed that when the TeA airspace has greater flexibility using the dynamic CDA, the air-side resources absorb more delays than the ground resources and reduce system delays.

- The proposed multi-objective cooperative co-evolutionary algorithm is shown to be able to co-evolve solutions towards the true global Pareto-front effectively while maintaining a high diversity of the solution set. The advanced fitness assignment scheme increased the selection pressure while the modified archiving updating mechanism and niching strategy help a fine spread of the non-dominated solutions.

- The proposed multi-objective CCRT was successfully applied for identifying the scenario-specific TeA airspace design strategies that are able to cope better with ground events/uncertainties and produce prior trajectories to distribute demand while maintaining aircraft safety. It provides an analyst with the trade-off between these two ATC priority efficiency and safety; thus solutions can be selected based on the criticality level of meeting the demand.

- Traffic scenarios with various arrival traffic distribution in terms of both density and direction have different solutions which proves that an appropriate TeA airspace design strategy should be considered with a specific traffic scenario.

- Traffic scenarios from two adjacent quadrants experienced heavier delays and higher collision risks than that from two opposite quadrants. Therefore for the sake of efficiency and safety, air traffic controllers should prioritize the TA design strategy with two opposite quadrants over the one with two adjacent quadrants.

7.2 Future Work

There are several possibilities to extend the research on the integrated TeA system presented in this thesis.

- Multiple airport scenarios: More than one airport can be incorporated in the scenarios, so that the vulnerabilities of a flight traffic flow system can be discovered and evaluated on a state level, given its ground and air constraints at a higher air traffic system level.

- Adding more concerns in the TeA airspace design problem: Efficiency and safety are the two priorities considered in this work, other ATM concerns, e.g. fuel consumption and human factors, can be incorporated in the model.

- Evaluation of other advanced ATM concepts: The efficiency of the dynamic CDA is evaluated in the integrated TeA system, other new developed ATM concepts can be examined in our methodology before real-world utilization.

- New co-evolutionary algorithm for CCRT: The thesis has shown a great deal of advantage to use cooperative co-evolution for searching massive spaces of possibilities governed by uncertainty and complex networked dynamics. There are still possibilities to extend the framework by combining competitive co-evolution with cooperative co-evolutionary techniques.

Appendix

Bibliography

Abbass, H. (2009). Computational Red Teaming and Cyber Challenges.

Abbass, H., Bender, A., and Whitbread, P. (2010a). Computational Red Teaming.

Abbass, H., Bender, A., and Whitbread, P. (2010b). Computational Red Teaming: Unravelling the Pharaoh's Curse.

Abbass, H. A., Alam, S., and Bender, A. (2009). Application notes: MEBRA: multiobjective evolutionary-based risk assessment. *Comp. Intell. Mag.*, **4**, 29–36.

Abbass, H. A., Bender, A., Gaidow, S., and Whitbread, P. (2011). Computational Red Teaming: Past, Present and Future. *Computational Intelligence Magazine, IEEE*, **6**(1), 30 –42.

Abbott, T. (2002). Speed Control Law for Precision Terminal Area in-trail Self Spacing. Technical report, NASA Technical Memorandum 2002-211742, NASA Langley Research Center, Hampton, VA.

Air Services Australia (2004). Designated Airspace Handbook. Technical report, Air Services Australila, Canberra, Australia.

Air Services Australia (2012). Departure and Approach Procedures (DAP). Technical report, Air Services Australila, Canberra, Australia.

Air Transport Association (2011). Cost of delays.

Alam, S., Abbass, H., and Barlow, M. (2008). ATOMS: Air Traffic Operations and Management Simulator. *Intelligent Transportation Systems, IEEE Transactions on*, **9**(2), 209 –225.

Alam, S., Abbass, H., and Lokan, C. (2009). Computational Red Teaming to Investigate Failure Patterns in Medium Term Conflict Detection.

Alam, S., Nguyen, M., Abbass, H., Lokan, C., Ellejmi, M., and Kirby, S. (2010a). A Dynamic Continuous Descent Approach Methodology for Low Noise and Emission. In *Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th*, pages 1.E.5–1 –1.E.5–18.

Alam, S., Zhao, W., Tang, J., Abbass, H., Lokan, C., Ellejmi, M., and Kirby, S. (2010b). Discovering Delay Patterns in Arrival Traffic with Dynamic Continuous Descent Approaches using Co-Evolutionary Red Teaming.

Alam, S., Zhao, W., Tang, J.and Lokan, C., Abbass, H., Ellejmi, M., and Kirby, S. (2011). Discovering Delay Patterns in Arrival Traffic with Dynamic Continuous Descent Approaches using Co-Evolutionary Red Teaming. In *9th USA/Europe Air Traffic Management Research and Development Seminars*, Berlin, Germany.

Anderson, K., Carr, F., Feron, E., and Hall, W. (2000). Analysis and Modeling of Ground Operations at Hub Airports. In *3th USA/Europe Air Traffic Management Research and Development Seminars*, Napoli, Italy.

Angeline, P., Angeline, P. J., Pollack, J. B., and Pollack, J. B. (1993). Competitive environments evolve better solutions for complex tasks. In *Proceedings of the Fifth International Conference (GA93)*.

Ashford, N., Stanton, H. P. M., and Moore, C. (1997). *Airport operations*. McGraw Hill (New York).

Back, T. (1994). Selective pressure in evolutionary algorithms: a characterization of selection mechanisms. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*, pages 57 –62 vol.1.

Ball, M. O., Hoffman, R., Odoni, A. R., and Rifkin, R. (2003). A stochastic integer program with dual network structure and its application to the ground-holding problem. *Oper. Res.*, **51**(1), 167–171.

Barbosa, H. and Barreto, A. (2001). An interactive genetic algorithm with co-evolution of weights for multiobjective problems. In *Genetic Evolutionary Computation, 2001. CEC '02. Proceedings of the 2001 Congress on*, pages 203–210.

Barmore, B., Abbott, T., and Krishnamurthy, K. (2004). Airborne-managed Spacing in Mulitple Arrival Streams. In *24th International Congress of the Aeronautical Sciences*, Japan.

Becher, T. and Formosa, J. (2000). Rnav near-term terminal procedures development. In *3rd USA/Europe Air Traffic Management Research and Development Seminar*.

Bertsimas, D. and patterson, S. (1998). The air traffic flow management problem with enroute capacities.

Bertsimas, D. and Patterson, S. (2000). The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach. *Transportation Science*, **34**(3), 239–255.

Bertsimas, D., Lulli, G., and Odoni, A. (2008). The air traffic flow management problem: An integer optimization approach. In A. Lodi, A. Panconesi, and G. Rinaldi, editors, *Integer Programming and Combinatorial Optimization*, Lecture Notes in Computer Science. Springer Berlin / Heidelberg.

Bertsimas, D., Lulli, G., and Odoni, A. (2011). An integer optimization approach to large-scale air traffic flow management. volume 59, pages 211–227, Institute for Operations Research and the Management Sciences (INFORMS), Linthicum, Maryland, USA. INFORMS.

Bohme, D., Brucherseifer, R., and Christoffels, L. (2007). Coordinated Arrival Departure Management. In *7th USA/Europe Air Traffic Management Research and Development Seminars*, Barcelona, Spain.

Boursier, L., Favennec, B., Hoffman, E., Trzmiel, A., Vergne, F., and Zeghal, K. (2007). Merging Arrival Flows Without Heading Instructions. In *7th USA/Europe Air Traffic Management Research and Development Seminars*.

Central Flow Management Unit (2009). Indicators and Analysis of the ATM Network Operations Performance. Technical report, EUROCONTROL, Brussels, Network Operations Report.

Chen, W., Weise, T., Yang, Z., and Tang, K. (2010). Large-scale global optimization using cooperative coevolution with variable interaction learning. In *Proceedings of the 11th international conference on Parallel problem solving from nature: Part II*, PPSN'10, pages 300–309, Berlin, Heidelberg. Springer-Verlag.

Choo, C. S., Chua, C. L., and Tay, S.-H. V. (2007). Automated red teaming: a proposed framework for military application. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, GECCO '07, pages 1936–1942, New York, NY, USA. ACM.

Chua, C. L., Sim, C. W. C., Choo, C. S., and Tay, V. (2008). Automated red teaming: an objective-based data farming approach for red teaming. In *Proceedings of the 40th Conference on Winter Simulation*, WSC '08, pages 1456–1462. Winter Simulation Conference.

Coppenbarger, R. A., Mead, R. W., and Sweet, D. N. (2007). Field Evaluation of the Tailored Arrivals Concept for Datalink-Enabled Continuous Descent Approach. In *7th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*.

Deb, K. and Kalyanmoy, D. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc., New York, NY, USA.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, **6**(2), 182 –197.

Decraene, J., Zeng, F., Low, M. Y. H., Zhou, S., and Cai, W. (2010). Research advances in automated red teaming. In *Proceedings of the 2010 Spring Simulation Multiconference*, SpringSim '10, pages 47:1–47:8, San Diego, CA, USA. Society for Computer Simulation International.

Delgado, L., Prats, X., Ferraz, C., Royo, P., and Pastor, E. (2009). An Assessment for UAS Depart and Approach Operations . In *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Hilton Head, South Carolina.

Dorronsoro, B., Danoy, G., Bouvry, P., and Nebro, A. (2011). Multi-objective cooperative coevolutionary evolutionary algorithms for continuous and combinatorial optimization. In P. Bouvry, H. Gonzlez-Vlez, and J. Kolodziej, editors, *Intelligent Decision Systems in Large-Scale Distributed Environments*, volume 362 of *Studies in Computational Intelligence*, pages 49–74. Springer Berlin / Heidelberg.

EUROCONTROL (2007). Contiuous Descent Approach: Implementation Guidance Information. Technical report, Environmental Unit, EUROCONTROL, Brussels, Belgium.

Eurocontrol (2010). Performance review report for 2010. Technical report, Eurocontrol, Brussels, Belgium, Tech. Rep. PRR2010.

EUROCONTROL (2011). Dynamic DCB: Reducing traffic complexity and streamlining ATC workload.

EUROCONTROL (2012). Network manager annual network operations report 2011.

EUROCONTROL/FAA (2001). Principles of Operation for the Use of Airborne Separation Assurance Systems. Technical report, EUROCONTROL/FAA Cooperative R&D, Edition 7.1.

Ficici, S. and Pollack, J. (2001). Pareto optimality in coevolutionary learning. In J. Kelemen and P. Sosk, editors, *Advances in Artificial Life*, volume 2159 of *Lecture Notes in Computer Science*, pages 316–325. Springer Berlin / Heidelberg.

Fonseca, C. and Fleming, P. (1995). Multiobjective genetic algorithms made easy: selection sharing and mating restriction. In *Genetic Algorithms in Engineering Systems: Innovations and Applications, 1995. GALESIA. First International Conference on (Conf. Publ. No. 414)*, pages 45 –52.

Fonseca, C. and Fleming, P. (1998). Multiobjective optimization and multiple constraint handling with evolutionary algorithms. i. a unified formulation. volume 28, pages 26 –37.

Fonseca, C. M. and Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: Formulation discussion and generalization. In *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 416–423, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Garca-Pedrajas, N., Hervs-Martnez, C., and Muoz-Prez, J. (2002). Multi-objective cooperative coevolution of artificial neural networks (multi-objective cooperative networks). *Neural Networks*, **15**(10), 1259 – 1278.

Garcia Pedrajas, N., Hervas Martinez, C., and Ortiz Boyer, D. (2005). Cooperative coevolution of artificial neural network ensembles for pattern classification. *Evolutionary Computation, IEEE Transactions on*, **9**(3), 271 – 302.

Gilbo, E. (1993). Airport capacity: Representation, estimation, optimization. *IEEE Transactions on Control Systems Technology*, **1**(3), 144–154.

Gilbo, E. (1997). Optimization of air traffic management strategies at airports with uncertainty in airport capacity. In *Proceedings of the 8th IFAC/IFIP/IFORS Symposium*, pages 35–40.

Goh, C.-K. and Tan, K. C. (2009). A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *Evolutionary Computation, IEEE Transactions on*, **13**(1), 103 –127.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Reading, Massachusetts : Addison Wesley Longman, Inc.

Goldberg, D. E. and Richardson, J. (1987). Genetic algorithms with sharing for multimodal function optimization. In *Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application*, pages 41–49, Hillsdale, NJ, USA. L. Erlbaum Associates Inc.

Gotteland, J., Durand, N., Alliot, J., and Page, E. (2001). Aircraft ground traffic optimization. Santa Fe, NM, USA.

Green, S. M. and Vivona, R. A. (2001). En Route Descent Advisor Concept for Arrival Metering. In *AIAA Guidance, Navigation, and Control Conf.*

Haraldsdottir, A., Scharl, J., Berge, M. E., Schoemig, E. G., and Coats, M. (2007). Arrival management with required navigation performance and 3d paths. In *7th USA/Europe Air Traffic Management Research and Development Seminar*.

Hillis, W. (1990). Co-evolving parasites improve simulated evolution as an optimization procedure. *Physica D: Nonlinear Phenomena*, **42**, 228 – 234.

Hingston, P. and Preuss, M. (2011). Red teaming with coevolution. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pages 1155 –1163.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor.

Horn, J. (1997). *The nature of niching: genetic algorithms and the evolution of optimal, cooperative populations*. Ph.D. thesis, Champaign, IL, USA.

Horn, J. (2003). Niche distributions on the pareto optimal front. In C. Fonseca, P. Fleming, E. Zitzler, L. Thiele, and K. Deb, editors, *Evolutionary Multi-Criterion Optimization*, volume 2632 of *Lecture Notes in Computer Science*, pages 75–75. Springer Berlin / Heidelberg.

Horn, J., Horn, J., Nafpliotis, N., Nafpliotis, N., and Goldberg, D. E. (1993). Multiobjective Optimization Using the Niched Pareto Genetic Algorithm. Technical report, Illinois Genetic Algorithms Laboratory.

Horn, J., Nafpliotis, N., and Goldberg, D. (1994). A niched pareto genetic algorithm for multiobjective optimization. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*, pages 82 –87 vol.1.

IMG (2010). Use Optimized Profile Descent: Integration Managers Group, Federal Aviation Administration. https://nasea.faa.gov/products/oi/main/display/85 accessed 13/03/2012.

Iorio, A. and Li, X. (2004). A cooperative coevolutionary multiobjective algorithm using non-dominated sorting. In K. Deb, editor, *Genetic and Evolutionary Computation GECCO 2004*, volume 3102 of *Lecture Notes in Computer Science*, pages 537–548. Springer Berlin / Heidelberg.

Isaacson, D. R., Robinson, J. E., Swenson, H. N., and Denery, D. G. (2010). A concept for robust, high density terminal air traffic operations. *Weather*, (September), 1–23.

Ivanescu, D., Shaw, C., Tamvaclis, C., and Kettunen, T. (2009). Models of Air Traffic Merging Techniques: Evaluating Performance of Point Merge. In *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO).*

Jani, M. and Toi, V. (1982). Terminal Airspace Capacity Model. *Transportation Research Part A: General*, **16**(4), 253 – 260.

Joint Planning and Development Office (2007). Concept of operations for the next generation air transportation system. version 2.0.

Joint Planning and Development Office (2011). Concept of operations for the next generation air transportation system. version 3.2.

Keerativuttitumrong, N., Chaiyaratana, N., and Varavithya, V. (2002). Multi-objective co-operative co-evolutionary genetic algorithm. In J. Guervs, P. Adamidis, H.-G. Beyer, H.-P. Schwefel, and J.-L. Fernndez-Villacaas, editors, *Parallel Problem Solving from Nature PPSN VII*, volume 2439 of *Lecture Notes in Computer Science*, pages 288–297. Springer Berlin / Heidelberg.

Khare, V., Yao, X., and Sendhoff, B. (2004). Credit assignment among neurons in co-evolving populations. In X. Yao, E. Burke, J. Lozano, J. Smith, J. Merelo-Guervs, J. Bullinaria, J. Rowe, P. Tino, A. Kabn, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *Lecture Notes in Computer Science*, pages 882–891. Springer Berlin / Heidelberg.

Kopardekar, P., Bilimoria, K., and Sridhar, B. (2007). Initial concepts for dynamic airspace configuration. In *7th AIAA Aviation Technology, Integration and Operations Conference, Belfast, Northern Ireland*.

Kopardekar, P., Bilimoria, K., and Sridhar, B. (2008). Airspace configuration concepts for the next generation air transportation system. *Air Traffic Control Quarterly*, **16**(4).

Korn, B., Helmke, H., and Kuenz, A. (2006). 4D Trajectory Management in the Extended TMA: Coupling AMAN and 4D FMS for Optimized Approach Trajectories. In *25th International Congress of the Aeronautical Sciences*.

Krozel, J., Penny, S., Prete, J., and Mitchell, J. S. (2004). Comparison of Algorithms for Synthesizing Weather Avoidance Routes in Transition Airspace. In *AIAA Guidance, Navigation, and Control Conf.*

Kursawe, F. (1991). A variant of evolution strategies for vector optimization. In H.-P. Schwefel and R. Mnner, editors, *Parallel Problem Solving from Nature*, volume 496 of *Lecture Notes in Computer Science*, pages 193–197. Springer Berlin / Heidelberg.

Kuster, J. and Jannach, D. (2006). Handling Airport Ground Processes Based on Resource-Constrained Project Scheduling. In M. Ali and R. Dapoigny, editors, *Advances in Applied Artificial Intelligence*, volume 4031 of *Lecture Notes in Computer Science*, pages 166–176. Springer Berlin / Heidelberg.

Lauren, M., Chong, N., McDonald, M., Pickl, S., and Sanchez, R. (2009). Maritime force protection study using mana and automatic co-evolution (ace). In *Proc. Scythe, Issue 6, Workshop 18*.

Leiden, K., Kamienski, J., and Kopardekar, P. (2007). Initial implications of automation on dynamic airspace configuration. In *Proc of Aviation Technology Integration and Operations Forum*.

Lohn, J., Kraus, W., and Haith, G. (2002). Comparing a coevolutionary genetic algorithm for multiobjective optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02.* , volume 2, pages 1157 –1162.

Lulli, G. and Odoni, A. (2007). The european air traffic flow management problem. *Transportation Science*, **41**(4), 431–443.

Mahfoud, S. W. (1995). *Niching methods for genetic algorithms*. Ph.D. thesis, Champaign, IL, USA. UMI Order No. GAX95-43663.

Maneeratana, K., Boonlong, K., and Chaiyaratana, N. (2004). Multi-objective optimisation by co-operative co-evolution. In X. Yao, E. Burke, J. Lozano, J. Smith, J. Merelo-Guervs, J. Bullinaria, J. Rowe, P. Tino, A. Kabn, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *Lecture Notes in Computer Science*, pages 772–781. Springer Berlin / Heidelberg.

Mayer, R. and Swancy, H. (2005). Analysis of potential benefits of arrival-departure procedures proposed for ord. In *Digital Avionics Systems Conference, 2005. DASC 2005. The 24th*, volume 1, pages 3.A.5 – 31–13 Vol. 1.

MITRE CAASD (2007). Terminal Area Route Generation, Evaluation, and Traffic Simulation. $http : //www.mitrecaasd.org/work/project_details.cfm?item_id =$ 152.

Moriarty, D. E. and Miikkulainen, R. (1997). Forming neural networks through efficient and adaptive coevolution. *Evol. Comput.*, **5**(4), 373–399.

Myers, T. and Kierstead, D. (2008). Network model to address capacity/demand imbalance in the national airspace system.

Nakamura, D. (2000). General information on the functional and technical aspects of required navigation performance (rnp) area navigation (rnav) and applications.

Nebro, A. J., Durillo, J. J., Luna, F., Dorronsoro, B., and Alba, E. (2007). Mocell: A cellular genetic algorithm for multiobjective optimization. *International Journal of Intelligent Systems*, **24**, 726–746.

Netjasov, F., Janic, M., and Tosic, V. (2011). Developing a generic metric of terminal airspace traffic complexity. *Transportmetrica*, **7**(5), 369–394.

Nolan, M. S. (2004). *Fundamentals of Air Traffic Controls*. 4th ed. Belmont, CA 94002-3098, USA.

Oberheid, H. and Söffker, D. (2008). Cooperative arrival management in air traffic control - a coloured petri net model of sequence planning. In *Proceedings of the 29th international conference on Applications and Theory of Petri Nets*, PETRI NETS '08, pages 348–367, Berlin, Heidelberg. Springer-Verlag.

Odoni, A. R. (1987). The flow management problem in air traffic control.

Paredis, J. (1995). Coevolutionary Computation. *Artificial Life*, **2**, 355–375.

Pinedo, M. (2002). *Scheduling:theory,algorithms,and systems*. Prentice-Hall, New Jersey.

Potter, M. and De Jong, K. (1994). A cooperative coevolutionary approach to function optimization. In Y. Davidor, H.-P. Schwefel, and R. Mnner, editors, *Parallel Problem Solving from Nature PPSN III*, volume 866 of *Lecture Notes in Computer Science*, pages 249–257. Springer Berlin / Heidelberg.

Potter, M. and De Jong, K. (2000a). Cooperative Coevolution: An Architecture for Evolving Coadapted Subcomponents. *Evolutionary Computation*, **8**(1), 1–29.

Potter, M. A. and De Jong, K. A. (2000b). Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evol. Comput.*, **8**(1), 1–29.

Prevot, T., Callantine, T., Homola, J., Lee, P., and Mercer, J. (2007). Air/Ground Simulation of Trajectory-Oriented Operations with Limited Delegation. In *7th USA/Europe Air Traffic Management Research and Development Seminars*, Barcelona, Spain.

Ranjeet, T. R., P., H., Lam, C., and Masek, M. (2011). Analysis of key installation protection using computerized red teaming. In *34th Australian Computer Science Conference (ACSC 2011)*.

Rathinam, S., Wood, Z., Sridhar, B., and Jung, Y. (2009). A generalized dynamic programming approach for a departure scheduling problem. American Institute of Aeronautics and Astronautics, 1801 Alexander Bell Dr., Suite 500 Reston VA 20191-4344 USA,.

Rehwald, E., Gaus, U., Kocks, S., Hecker, P., Schmand, C., and Schwanke, S. (2010). Flow management as a contribution to airport's efficiency and pollution control. In *Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th*, pages 2.A.4–1 –2.A.4–5.

Richetta, O. and Odoni, A. R. (1993). Solving optimally the static ground-holding policy problem in air traffic control. In *Transportation Science*.

RobinsonIII, J. and Kamgarpour, M. (2010). Benefits of Continuous Descent Operations in High-Density Terminal Airspace Under Scheduling Constraints. In *American Instituate of Aeronautics and Astronautics*.

Rosin, C. and Belew, R. (1997). New methods for competitive coevolution. *Evolutionary Computation*, **5**(1), 1–29.

Sarker, R., Mohammadian, M., and Yao, X. (2002). *Evolutionary Optimization*. Kluwer Academic Publishers, MA, USA.

Schmitt, L. M. (2004). Theory of genetic algorithms ii: models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling. *Theoretical Computer Science*, **310**, 181 – 231.

Seng, C. C., Lian, C. C., Spencer, L. K. M., and Darren, O. W. S. (2009). A co-evolutionary approach for military operational analysis. In *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, GEC '09, pages 67–74, New York, NY, USA. ACM.

SESAR Consortium (2007). Single European Sky ATM Research Consortium (SESAR) ATM Target Concept.

Sofge, D., De Jong, K., and Schultz, A. (2002). A blended population approach to cooperative coevolution for decomposition of complex problems. In *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*, volume 1, pages 413 –418.

Spence, C., E. (2003). FAA Aeronautical Information Manual/Federal Aviation Regulation. Technical report, McGraw-Hill Companies.

Sridhar, B., Chatterji, G., Grabbe, S., and Sheth, K. (2002). Integration of traffic flow management decisions. In *AIAA Guidance, Navigation, and Control Conference*, pages 1–9. Monterey, California.

Srinivas, N. and Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. volume 2, pages 221–248. MIT Press.

Swenson, H., Thipphavong, J., Sadovsky, A., and Chen, L. (2011). Design and evaluation of the terminal area precision scheduling and spacing system. In *9th USA/Europe Air Traffic Management Research and Development Seminar*.

Tan, K., Lee, T., Yang, Y., and Liu, D. (2004). A cooperative coevolutionary algorithm for multiobjective optimization. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 2, pages 1926 – 1931 vol.2.

Tan, K., Yang, Y., and Goh, C. (2006). A distributed cooperative coevolutionary algorithm for multiobjective optimization. *Evolutionary Computation, IEEE Transactions on*, **10**(5), 527 –549.

Tan, K. C., Khor, E. F., Lee, T. H., and Sathikannan, R. (2003). An evolutionary algorithm with advanced goal and priority specification for multi-objective optimization. *J. Artif. Int. Res.*, **18**(1), 183–215.

Tosic, V. and Netjasov, F. (2003). Terminal airspace traffic complexity metric. *5th EURO/INFORMS Joint International Meeting, Istanbul, Turkey*.

Uebbing-Rumke, M. and Temme, M., M. (2011). Controller aids for integrating negotiated continuous descent approaches into conventional landing traffic. In *9th USA/Europe Air Traffic Management Research and Development Seminar*.

Upton, S. and McDonald, M. (2003). Automated red teaming using evolutionary algorithm. In *WG31 – Computing Advances in Military OR, 2003*.

Upton, S., Johnson, S., and McDonald, M. (2004). Breaking blue: Automated red teaming using evolvable simulations. In *Proceding of the Genetic and Evolutionary Computation Conference (GECCO 2004)*.

U.S. Department of Transporation (2011). Bureau of Transportation Statistics, Airline on-time tables.

Van Veldhuizen, D. A. (1999). *Multiobjective evolutionary algorithms: classifications, analyses, and new innovations*. Ph.D. thesis, Wright Patterson AFB, OH, USA. AAI9928483.

Vranas, P. B., Bertsimas, D. J., and Odoni, A. (1994). The multi-airport groundholding problem in air traffic control. In *OPERATIONS RESEARCH*, volume 42.

Wiegand, R. P., Liles, W. C., and Jong, K. A. D. (2001). An empirical analysis of collaboration methods in cooperative coevolutionary algorithms. In *In Proceedings from the Genetic and Evolutionary Computation Conference*, pages 1235–1242. Morgan Kaufmann.

Wiegand, R. P., Liles, W. C., and Jong, K. A. D. (2002). The effects of representational bias on collaboration methods in cooperative coevolution. In *Proceedings of the 7th International Conference on Parallel Problem Solving from Nature*, PPSN VII, pages 257–270, London, UK, UK. Springer-Verlag.

Xing, Z., Hou, Y., Zhang, Y., Jia, L., and Hou, Y. (2006). A multi-objective cooperative coevolutionary algorithm for constructing accurate and interpretable fuzzy systems. In *Fuzzy Systems, 2006 IEEE International Conference on*, pages 1404 –1410.

Xing, Z., Zhang, Y., Hou, Y., and Jia, L. (2007). On generating fuzzy systems based on pareto multi-objective cooperative coevolutionary algorithm. *International Journal of Control, Automation, and Systems*, **5**(4).

Xu, Y., Low, M. Y. H., and Choo, C. S. (2009). Enhancing automated red teaming with evolvable simulation. In *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, GEC '09, pages 687–694, New York, NY, USA. ACM.

Yang, A., Abbass, H., and Sarker, R. (2006). Characterizing warfare in red teaming. volume 36, pages 268 –285.

Yang, Z., Tang, K., and Yao, X. (2008a). Large Scale Evolutionary Optimization Using Cooperative Coevolution. *Information Sciences*, **178**(15), 2985 – 2999.

Yang, Z., Tang, K., and Yao, X. (2008b). Multilevel cooperative coevolution for large scale optimization. In *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, pages 1663 –1670.

Zhao, W., Tang, J., Alam, S., Bender, A., and Abbass, H. A. (2010). Evolutionary-computation based risk assessment of aircraft landing sequencing algorithms. In *DIPES/BICC'10*, pages 254–265.

Zitzler, E. and Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. volume 3, pages 257 –271.

Zitzler, E., Deb, K., and Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: empirical results. *Evolutionary Computation*, **8**(2), 173–195.

Zitzler, E., Laumanns, M., and Thiele, L. (2001). Spea2: Improving the strength pareto evolutionary algorithm. Technical Report 103, Eidgenssische Technische Hochschule Zrich (ETH), Institut fr Technische Informatik und Kommunikationsnetze (TIK).