[Type here]

University of Southampton Research Repository

Thesis: Hang Nguyen (2019) "Crew Scheduling for Vietnam Airlines", University of Southampton, Southampton Business School, PhD Thesis, pagination.

University of Southampton

Faculty of Social Sciences

Southampton Business School

Crew Scheduling for Vietnam Airlines

By

Hang Thu Nguyen

Thesis for the degree of Doctor of Philosophy

August 2019

University of Southampton

Abstract

Faculty of Social Sciences

Southampton Business School

Thesis for the degree of Doctor of Philosophy

Crew scheduling for Vietnam Airlines

Hang Thu Nguyen

An airline crew scheduling problem is one of the largest-scale optimization problems for the airline industry. The crew scheduling problem plays an important role in airline operations and is an interesting problem for the application of operations research. The objective of this problem is the optimal allocation of crews to flights.

Because of the large size of the airline industry and the complexity of safety rules and regulations, as well as employment agreements, the crew scheduling problem is divided into two sub-problems: crew-pairing and crew-rostering. For crew-pairing, all pairings are formed in order to exactly cover every flight of the schedule, and then the crew-rostering process assigns these pairings to individual crew-members and generates the monthly rosters.

The objective criteria of the crew scheduling problem typically require a reduction of the number of unassigned flights, the minimization of the number of crews needed to cover the duties and fairness of crew assignment, as well as the preferences of special working-patterns of crews.

In this research, we decompose the problems in many stages and solve them in a day-by-day rolling manner for the crew pairing problem, with two heuristics and exact method combinational algorithms, and a crew-by-crew approach for the crew rostering problem. We combine heuristics with a new mathematic formulation in several algorithms to solve the problems.

Specifically, we apply techniques of constraint programming, such as domain creation, local consistency, bound consistency, local search and constraints propagation, in order to design effective heuristic algorithms for the crew rostering problem to generate rosters, thus gaining a good quality solution and reducing computational time significantly. In addition to the airline regulations being encoded by several constraints, we impose additional constraints to reduce the domains of variables. The resulting domain reductions are propagated to other constraints, which additionally reduces the search space. Numerical results based on the data for Vietnam Airlines are presented and demonstrate the potential of our approach.

Table of Contents

Table of Tables

Table of Figures

Research Thesis: Declaration of Authorship

Print name:	HANG THU NGUYEN

Title of thesis:	Crew Scheduling of Vietnam Airlines

I declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. None of this work has been published before submission.

Signature:		Date:	

Acknowledgements

I would like to thank my supervisory team for their guidance, including Professor Chris Potts for his instruction and correction of my models, Professor Tolga Bektas for his support and sympathy for my weaknesses in the research field and Dr Antonio Martinez-Sykora for his code checking and directing of my second paper, as well as his questions for further research.

I also thank Dr Yue Wu for not hesitating to take me on as a PhD student and for supervising me through my first year of study. Especially, I wish to thank Professor André Bergsten Mendes, who taught me research skills and mathematical models in the optimization field during the first year of my PhD.

I wish to thank Professor Hoai V. Nguyen, Professor Man Nguyen and Dr. Nguyen T. Hoang at HoChiMinh University of Technology for their teaching and consultancy when I had just started my research and at any time I needed advice. I want to thank Dr. Du Nguyen and Dr HaiHang Nguyen at the Vietnam Academy of Aviation for their concerns.

Special thanks go to my young friends Nghia Nguyen, KieuMy Nguyen, Huy Nguyen and Le Nguyen for their help and Java teaching.

Last but not least, I would like to thank my parents for their support and encouragement and my brothers for understanding me.

Airline Crew Scheduling Terminology

- ***Block time:*** the time from the moment the aircraft door closes at departure of a flight until the moment the aircraft door opens at the arrival gate following its landing. Block hours are the industry-standard measure of aircraft utilization.
- ***Briefing time/ Reporting time***: a period of time before the start of each duty that is spent on instructions and reports for any duty.
- ***Credited flying time in a duty***: the active flying time, plus a specific percentage of deadhead flying time.
- ***Deadhead***: a flight leg in which a crew travel as a passenger and sits in the passenger areas for relocation purposes.
- ***Deadhead crew / Positioning crew***: a crew travels as a passenger and sits in the passenger areas for relocation purposes.
- ***Debriefing time/ Post-flight time***: a period of time at the end of each duty that gives the crew-member an understanding of the events that occurred and their implications.
- ***Double crew***: consists of two single crew complements.
- ***Duty period***: is the full working period that a pilot is on duty in a 24 hours day, from the moment a crew-member reports to duty as required by an operator to the moment the crew-member is free from all duties.
- ***Flight leg*** : a nonstop flight that has only one take-off and one landing. Each flight leg has five features, namely the flight number, the original airport, the destination airport, the departure time and the arrival time.
- ***Flight time:*** refers to the moments of the duty time when the aircraft first moves under its own power for the purpose of taking-off, until the moment at which it comes to rest after landing, whereas the 'flight duty period' is the part of the duty period that includes both flight time, pre- and post-flight duties, and positioning or other duties at the beginning of the duty period.
- ***Home base/ Main base***: the place nominated by the Company to the crew-member from where the crew-member normally starts and ends a duty period and at which place, under normal conditions, the Company is not responsible for accommodation of the crew-members. Hanoi and Hochiminh city are designated as the 2 home/main bases of Vietnam Airlines.
- ***Pairing***: a sequence of duty periods and rest periods that starts and ends at a home base. Typically, pairings last from 1 to 5 days.
- ***Rest period***: is a period of time (an overnight stop) between duties that typically lasts for at least 11 hours. Rest period is on duty time away from base.

- ***Route***: a sequence of flight legs flown by a specific aircraft.
- ***Single crew***: a complement of two pilots, one Captain and one First Officer (FO).
- ***TAFB (Time Away From Base)***: is the time that a crew-member is on duty away from the home base.

I.INTRODUCTION

I. 1. Background

The airline industry plays a significant part in the international and domestic transportation markets of most countries, and is the one of the most competitive industries nowadays. Air transport expenditure is high, since the cost of fuel and the cost of crew, including salaries, benefits, and expenses, form a major component in the financial statement of the carrier (Anbil *et al.*, 1991). As a result, airline planning attracts a lot of research from academia and practitioners to reduce cost and increase efficient and effective activities (Barnhart *et al.*, 2003; Bazargan, 2010).

Unlike the fuel cost, a large portion of flight crew expenses is controllable, due to good scheduling (Anbil et al., 1991). Because of the enormous effect on the total expenses of carriers, the airline crew-scheduling problem has received considerable attention from both academia and the industry (Barnhart et al., 2003; Bazargan, 2010). Airline crew-scheduling is one-step in the whole process of airline planning problems, which are typically solved sequentially due to their large size and complexity.

Airline planning is very complex and involves the airline and many other parties, such as passengers, ground handling staff, aircraft maintenance workers, crew and so on. Figure 1 shows the diagram of the stages of the airline planning procedure, which are inter-related: the output of the previous stage is the input of the following stage (Barnhart et al., 2003; Kasirzadeh, Saddoune and Soumis, 2015).

In the first planning step, based on market demands for flight routings, a schedule generation problem is solved, to determine flights that are going to be operated during a given time period; therefore, this flight schedule is also called a *commercial schedule*.

The next step is the fleet assignment problem. Typically, this determines what type of aircraft (such as Boeing 767, 727, etc.) is allocated to each *flight leg or flight segment* that has only one take-off and one landing. The routing and the estimation of potential passenger numbers on each flight leg are the main factors affecting the decision about the aircraft type allocation. Because the revenue of a flight depends on the passenger demand for the flight and the capacity of the aircraft used to operate that flight, appropriate fleet type allocation is vital to ensure a profit for each flight.

The maintenance routing problem is the third phase involved in the selection of a particular aircraft being assigned to a specific flight in order to fit the routine maintenance check timetable of each aircraft. For safety reasons, all aircraft must have maintenance checks after flying for a given time, with parks at the home base overnight for the routine maintenance tasks.

After having completed these three tasks, the crew-scheduling problem is the last stage, whereby each individual crew member is allocated to a specific flight (Barnhart et al., 2003; Gopalakrishnan and Johnson, 2005). Barnhart *et al.* (2003) define crew scheduling "as the problem of assigning a group of workers (a crew) to a set of tasks" (p. 515). In the other words, airline crew scheduling is the combining of individual flight legs into a sequence and allocating them to individual crew members (Bazargan, 2010).

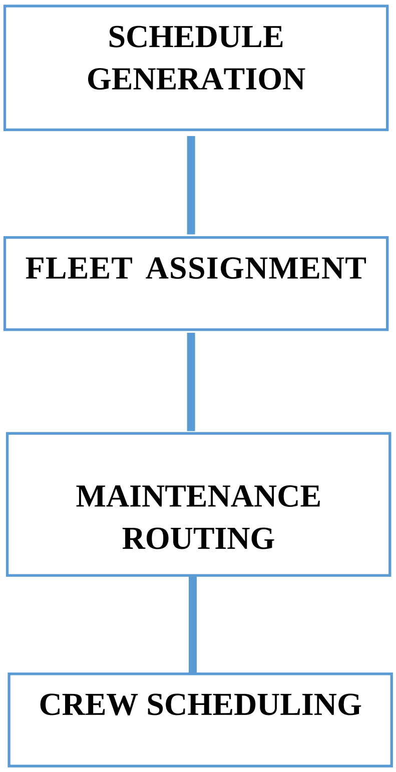


Figure 1: Scheduling Planning (Barnhart et al., 2003)

In a crew-scheduling problem, international and domestic operations are scheduled differently because of their different frequencies and routing characteristics. The international flight networks are characterized by point-to-point networks and normally fewer than the domestic ones, leading to a timetable operated on a weekly schedule. In contrast, the domestic flight networks are hub-and-spoke networks, with a higher frequency of flights operating daily schedules (Barnhart *et al.*, 2003).

As a lecturer of Vietnam Aviation Academy – the only Aviation Academy in Vietnam, I have seen that currently, the crew-scheduling of Vietnam Airlines (VNA) is still solved semi-manually by personnel in the crew-scheduling department and this approach is almost certainly inefficient and could lead to unfair scheduling of cockpit crew, with rosters those can give some cockpit crew very tight schedules, whilst others are given lighter ones. The reason that VNA continues to schedule crew semi-manually, even though there was a period of testing a commercial scheduling software, is that the application of the professional software was not successful as it was ineffective at scheduling. Thus, I would like to research these problems to find out appropriate methods and procedure to solve the problem.

I. 2. Vietnam Airlines

Vietnam airlines, the flag air carrier of Vietnam, was founded in 1956 under the name Vietnam Civil Aviation before becoming established in 1989 as a state-owned enterprise, with its headquarters in Hanoi – the capital of Vietnam. Since 2016, the two hubs have been Noi Bai International Airport in Hanoi (HAN) and Tan Son Nhat International Airport in Ho Chi Minh City (SGN), with Da Nang International Airport (DAD) being the secondary hub. The airline flies to 52 destinations in 17 countries . Passenger transport is the core activity of VNA and VNA has become a member of Sky Team in June 2010.

When interviewing VNA staff and observing them working in the crew-scheduling department, it was clear that the crew scheduling system at VNA was still being done by a group of personnel on the computers with only the support of simple Excel software. Meanwhile, VNA has been developing, the fleet and crews have been increasing significantly and, therefore, manual scheduling is inefficient and ineffective. One of the specific characteristics of Vietnam Airlines' crew scheduling problem is the existence of two separate payment methods for two different groups of pilots, namely, the Vietnamese and the Non-Vietnamese pilots. This aspect, together with other routing characteristics, prevents the effective application of commercial scheduling software in the computing system. These also make the scheduling tasks more complicated and the staff have to schedule manually to balance the workload of crew appropriately between the two groups.

I. 3. Research Contributions

We present two new models and several solution approaches for the crew scheduling problem of VNA, with specific characteristics, such as the combination of the hub-and-spoke structure with the point-to-point route system and the two payment methods for the crew complement. Our models are a daily-rolling pairing formulation for the crew pairing problem and an integer linear programming (ILP) model for the crew rostering problem. Several algorithms have been developed and implemented, based on randomization and constraint programming concepts, to meet the real requirements of crews.

The results of our research will not only provide VNA with the appropriate crew scheduling solution which will reduce expenditure, increase the efficiency of manpower (the average working hours of crew will rise), but they will also provide the exact number of required crew. Besides, the thorough analysis of crew scheduling of VNA has been presented in the Vietnam Aviation Academy project. More importantly, we applied constraint programming techniques to reduce the search space, leading to reduced computational time. Furthermore, in the current literature review, the structures of routes have not been concerned with much when analysing and solving the crew pairing problem, even though there are several studies which have integrated fleet assignment with the crew pairing problem. Therefore, our research also provides a new approach to analysing scheduling data and optimizing the problem from the business point of view.

The thesis is composed of seven themed chapters, including this introductory chapter. Chapter two begins with an overview of airline crew scheduling and the third chapter presents the current literature review. The fourth section details the analysis of the crew scheduling of VNA, with special characteristics and its data. Chapter 5 introduces two heuristics and exact combinatorial algorithms to solve the crew pairing problem of VNA, and then chapter 6 proposes a new formulation and three heuristic algorithms for the crew rostering problem, in order to produce the best quality rosters for the VNA crew complement. The final chapter summarises the entire thesis and identifies areas for further research.

II. AN OVERVIEW OF AIRLINE CREW SCHEDULING

The airline crew scheduling methodology has developed from the solely standard quantitative optimization techniques to the recent trends toward a structured planning process in order to obtain meaningful schedules representing airline operations. In addition, the large number of flights, complex rules to be consider and cost structure make the crew scheduling problem more difficult. Therefore, it is necessary to consider all components of the airline in the model and a combination of exact mathematical programming algorithms and heuristics is applied in the 'construction' and 'evaluation' of schedules (Maximilian and Dennis 1985). This chapter provides the elements of general airline crew scheduling at commercial airlines as well as the reason of crew scheduling decomposition into crew pairing problem and crew rostering problem.

II. 1. Components of airline crew scheduling

Normally, each cockpit crew or pilot is qualified to fly a specific type of aircraft or fleet type and a set of closely related fleet types, called a fleet family. For example, the aircraft Airbus A321 is a member of the Airbus A320 family of short to medium range, narrow-bodied, commercial passenger twin-engine jet airliners. Since there are different flight times on the routes of each fleet type and the health and safety regulations, the number of required crew on each flight is also different. Therefore, the crew-scheduling problem is solved separately for each group of crewmembers in different fleet types. Accordingly, the input of a crew-scheduling problem is the set of flights assigned to a specific fleet, which is the outcome of the fleet assignment problem, as shown on Table 1, below (Barnhart et al., 2003). More details must be provided for the crew-scheduling process, as seen in Table 2, below, a part of the flight schedule of Vietnam Airlines.

Table 1: Fleet assignment for Ultimate Air (Bazargan, 2010)

Flight number	**Origin**	**Destination**	**Fleet type**
111	ATL	JFK	737 -800
113	MIA	JFK	737 -800
118	BOS	JFK	737 -800
131	JFK	ATL	737 -800
136	JFK	MIA	737 -800
138	JFK	BOS	737 -800

Table 2, below, shows part of the real flight schedule of Vietnam Airlines (VNA) on 1st Jan, 2014. Each line of the schedule is the detail of a single flight, including the aircraft type (column 1), with a unique aircraft registration number (column 2) to carry flights, and the flight number (column 3) as shown on the tickets of passengers. The route of the flight in the Dep (Departure) column is indicated by the three-character code of the origin (KIX) and in the Arr (Arrival) column, containing the other three-character code of the destination (HAN). The other important information about the flight are an Estimated Date of the flight in E_Date column, an Estimated Time of the Departure (ETD) in ETD column, an Estimated Time of the Arrival (ETA) in ETA column, and a flight time (Flt Time).

Table 2: A partial flight schedule of Vietnam Airlines

AC type	Reg AC	Flt_No	Dep	Arr	E_Date	ETD	ETA	Flt_Time
32B	VNA322	VN331	KIX	HAN	1/1/14	01:30	07:15	05.45
32B	VNA322	VN7565	HAN	DLI	1/1/14	08:00	09:50	01.50
32B	VNA322	VN7564	DLI	HAN	1/1/14	10:40	12:25	01.45
32B	VNA322	VN1547	HAN	HUI	1/1/14	13:30	14:40	01.10
32B	VNA322	VN1379	HUI	SGN	1/1/14	15:30	16:50	01.20
32B	VNA322	VN420	SGN	PUS	1/1/14	17:15	21:50	04.35
32C	VNA323	VN1372	SGN	HUI	1/1/14	02:10	03:30	01.20

Each *flight leg* or *flight segment* consists of only one take-off at the ETD and one landing at the ETA, without any intervening stops. In addition, several sequential flights have been assigned to a particular aircraft, as they are listed in the schedule. For example, from line number 2 to line number 7, the aircraft VNA322 carries six flights, specifically, VN331, VN7565, VN7564, VN1547, VN1379 and VN420.

When allocating crew complements to these flights, a crew complement or a crew, including a captain and a first officer (FO), there are many rules and regulations that must be checked and adhered to in the Flight Operation Manual (FOM) of the airlines. For instance, the total flight time limits of all flights being flown by a crew member over the course of a work day must be equal to or less than 11 hours in a 24 hour day. A sequence of these flights makes a *duty period,* and the same crew members in a crew complement typically fly together throughout the duration of a duty period.

In the example of the six flights for the aircraft VNA322 above, the total flight time of these flights is 16h 25'; thus, one crew (a captain and a FO) cannot operate all of them. These flights must be separated into several duty periods and how to separate them has to be carefully calculated and comply with many rules and regulations.

The sixth flight of the VNA322 starts from SGN to go to PUS in the late evening; therefore, the crew must stay at PUS that night and fly back to SGN on the following day(s). Since a duty period can start and end at different airports, the crew are unable to return to their home base at the end of their duty period and must layover at the destination until the following day(s), when they begin another duty period back to the home base. These two flights, belonging to two duty periods of different days, create a *pairing*. Therefore, a pairing is a sequence of duty periods and stopovers and must start and finish at the same home base. Normally, crew spend anytime from one to five days in a row away from home, and stay together for all of the duties within a pairing (Barnhart et al., 2003). However, if a duty period starts and finishes at the same home base, it is a one day pairing or a one duty period pairing.

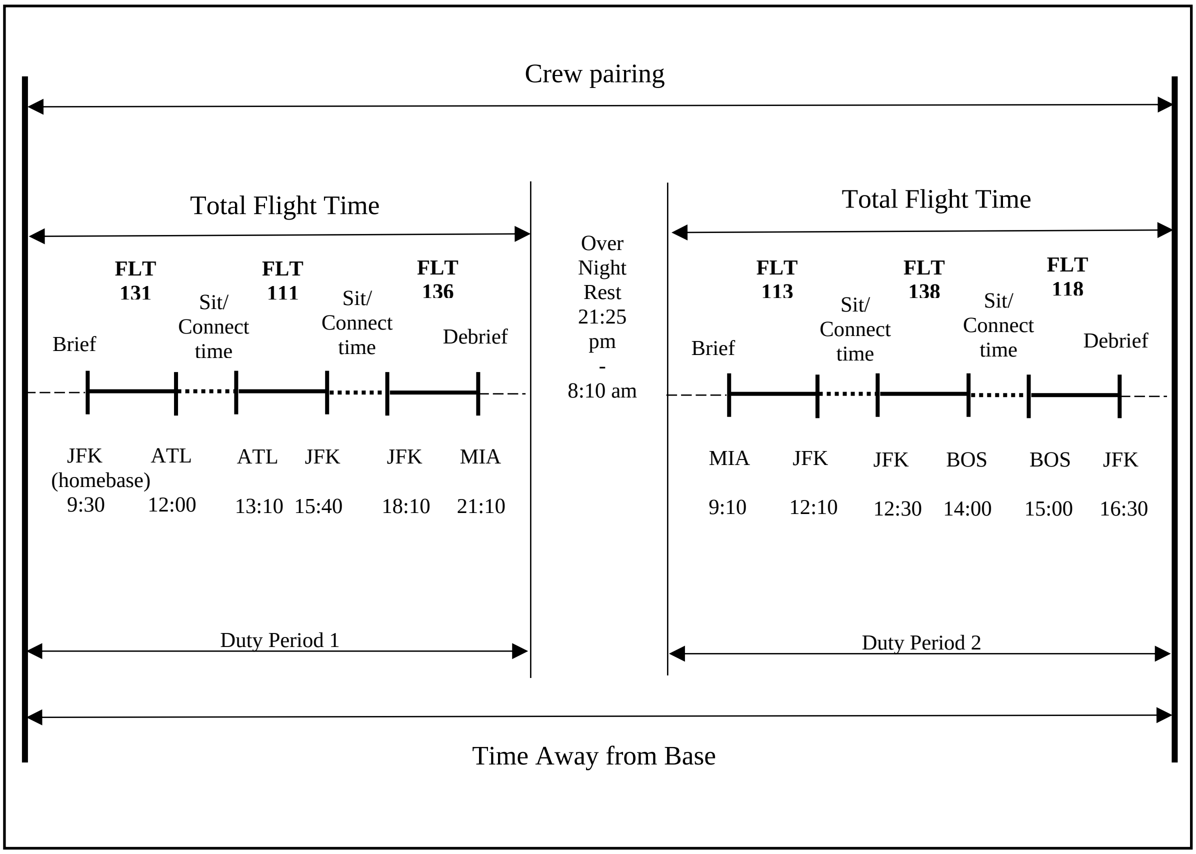


Figure 2: A typical pairing with duty periods, sits within duty periods, overnight rest, and sign-in and sign-out times (Bazargan, 2010).

Consequently, normally the crew-scheduling problem will initially involve grouping several short-haul (short flight time) flight legs of different routes to form a duty period of a working day for a crew, which consists of a sequence of flights with short rest periods or connect times or sits (American for breaks) separating them, as in Figure 2, above (Barnhart et al., 2003; Abdelghany and Abdelghany, 2010). At the beginning and end of each duty period are brief and debrief periods, respectively, which are included in the duty period (Vance et al., 1995). The duty period generation of an airline must follow the rules and regulations of the Civil Aviation Authority or Federal Aviation Administration of a country. Thus, VNA must adhere to the Civil Aviation Authority of Vietnam (CAAV)'s rules and regulations. A pairing may combine two, or several, duty periods together, but each pairing must begin and end at the home base of the crew (Vance et al., 1995). Finally, monthly schedules are made up of multiple pairings, with time off in between, and airline schedules are generally designed for the interval of six monthly periods.

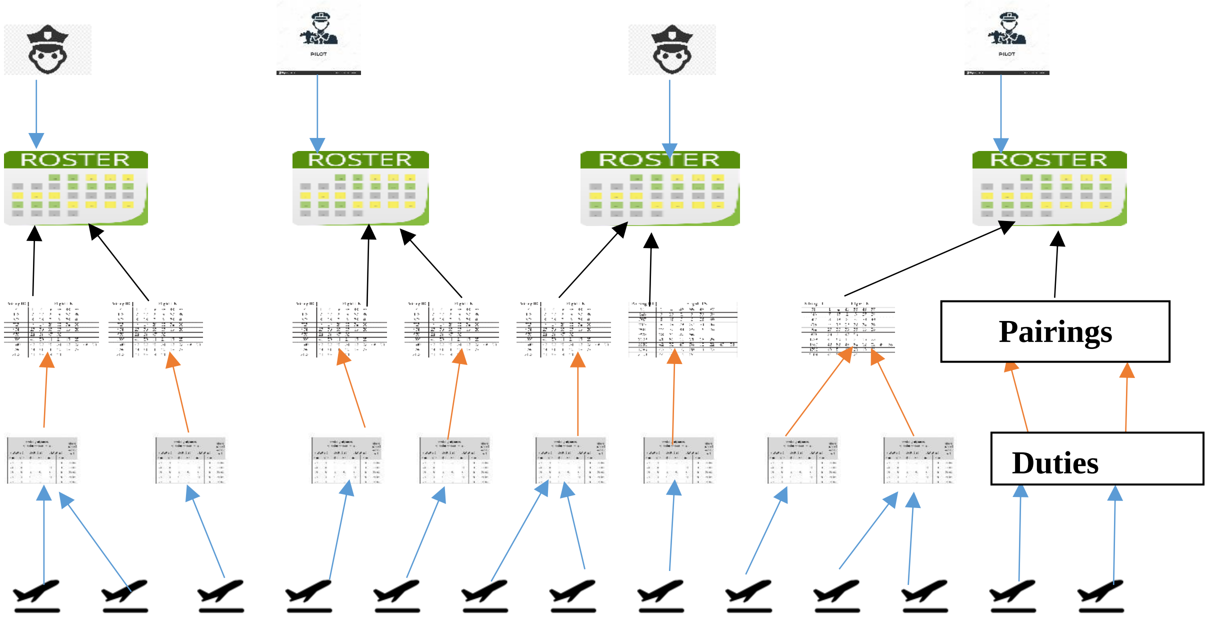


Figure 3: The Crew Scheduling Diagram

Generally, the four elements of crew-scheduling problem, as shown on Figure 3, above, are flight legs, duty periods, pairings, and monthly schedules. Building each of these components has to deal with distinct sets of regulations, which typically come from three sources. Firstly, governing agencies, such as the Civil Aviation Authority of Vietnam (CAAV) or the Federal Aviation Administrative (FAA) of United states, control the crew scheduling, primarily for safety purposes. Secondly, labour organizations on behalf of employees are often concerned about the crews' work conditions and, finally, the airlines pose other constraints in order to meet their objectives, such as, making the schedule more robust. In addition to these above constraints, the airlines' size, network structure, and cost structure also effect to the solution methods of crew scheduling problem (Barnhart et al., 2003; Kasirzadeh, Saddoune and Soumis, 2015).

II.1.1. Work rules and Pay structures

The most basic objectives of the crew-scheduling problem are how to assign individual crew-members to given flights and how to compute the cost of such an assignment. This is a very complex matter and, currently, there are two payment systems, namely, the fixed salary in European airlines and the credit hours payment, which is based on the time that crew spend flying, plus allowances, in North American airlines. Therefore, the ‘cost’ associated with an individual flight is simply the duration of that flight on air or the flight time of that flight. Accordingly, crew cost is usually expressed in terms of time rather than ‘real’ cost, due to the fact that the total flying time in the system is clearly fixed, and regarded to be a lower bound on the optimal crew cost. As a result, the goal of crew scheduling is to minimize “pay-and-credit, the payment made above and beyond the cost of the actual flying time” (p. 5) (Barnhart et al., 2003).

II.1.2. Duty Periods

When combining a sequence of flights to generate a duty period for a single crew over the course of a workday, there are a number of rules and regulations. The most obvious rule is that all flights in a duty period must be sequenced in space and time; this means that the destination of the preceding flight must be the origin of the following flight and the arrival time (ETA) of the preceding flight must be before the departure time (ETD) of the following flight, plus short break times.

The interval between the ETA of the preceding flight leg and the ETD of the following flight leg is called the connect time, and must be at least equal to or longer than the minimum connect time of the fleet type, which the airlines regulate. In addition, the restriction on the minimum connect time or sit time differs between international flights and domestic flights.

Finally, strict regulations control the maximum elapsed time of a duty period and the total flying hours, that a crew can conduct during the course of a single duty period (Barnhart et al., 2003). Under the CAAV, pilots can fly a maximum of 11 hours in a 24-hour period and they must rest at least the same time span as the duty period, or more (CAAV, 2015).

As regards the crew, the same crew complement typically flies together throughout the duration of a duty period. The crew cost associated with a duty period is the maximum of three quantities, the first of which is the flying time, which is also the block time. Block time is the total amount of time a flight takes from an aircraft is pushed back from the departure gate (“off-blocks”), to arriving at the destination gate (“on-blocks”). The second quantity is the product of the $Elapse\_Factor$ (a fraction associated with the duty elapsed time) and elapse (total elapsed time of the duty period), while the third quantity is a minimum duty guarantee (MDG), or a minimum guaranteed number of hours being paid for the crew on a duty. This payment method is primarily based on flying time, but also provides additional pay for the crew in cases of being assigned to very short duties or to duties with extensive idle time between the flights, such as delays caused by weather, aircraft maintenance, etc. (Barnhart et al., 2003). Formally, the cost of a duty is expressed as below:

$$DUTY_{COST} = \max\{flying\ time, Elapse\_Factor * elapse, MDG\}$$

Typically, *MDG* is 3 hours and the *Elapse_Factor* is $\frac{4}{7}$ (Barnhart et al., 2003; Gopalakrishnan and Johnson, 2005).

II.1.3. Pairings

A duty period may start and finish at different airports; thus, the crew cannot always return home at the end of the duty period but, instead, must stay over until the following day's duty period begins. A sequence of duty periods and the overnight rests in between forms a pairing, and the whole period of the pairing is called Time Away From Base (TAFB). A feasible pairing must fulfil a number of logical constraints. The first one is that the first duty period of a pairing must clearly begin at the crews' home base and the last duty period must end there as well. Moreover, each duty period of the pairing must start at the same airport where the preceding duty period ended. Other constraints are the compulsory minimum rest requirement between two consecutive duty periods, and the maximum elapsed time of a pairing.

In addition, flying time restrictions and the maximum number of duties in the pairing are included in the FOM of the airlines. A particularly complicated constraint is the 11-in -24 rule imposed by the CAAV in Vietnam or the 8-in-24 rule controlled by the FAA in the US. (Barnhart et al., 2003). The CAAV rules impose extra rest if a pairing spans more than 11 hours of flying in any 24-hour period.

As mentioned above, the salaries of cockpit crew dominate the overall personnel costs and there are two payment methods for pilots, the first of which is based on credit hours, as in North America, where the crew is paid per their flying hours over a month whereas European carriers apply a fixed salary for each crew (Barnhart et al., 2003).

Therefore, the crew pairing model concerns only the “pay-and-credit” costs, which are all the incremental crew-related costs, such as the cost of meals and lodgings and a per diems allowance, and positioning. Positioning is a flight commonly used to reposition a crew to an airport where they are needed to cover a flight, or to enable the crew to return to their home base at the end of a pairing. Typically, these excess costs of a pairing are added to the flying time and are caused by (a) long and frequent connect times within a duty period (b) long rest periods, and (c) deadheading.

The general formulation of the cost of a pairing is as below:

$$\max\{\ \#duties * PMDG, TAFB\_Factor * TAFB, Total\_Duty\_Cost\}$$

Pairing Minimum Duty Guarantee (PMDG) is the minimum number of hours that the crew is guaranteed to be paid for each duty in a pairing, irrespective of the length of the duty period, although it is typically 5 hours. The Time Away From Base Factor (TAFB_Factor) is a constant fraction associated with the time the crew are away from their home base, which normally is $\frac{2}{7}$, and Total_Duty_Cost is the summation of the costs of all duties in the pairing (Barnhart et al., 2003; Gopalakrishnan and Johnson, 2005).

In the scheduling process, the total flight time of all flights in the schedule is a lower bound on the cost of a given schedule, as the flying time of each flight is fixed. The TAFB is fluctuated depending on the algorithms; therefore, any pairing having a large TAFB relative to the total flying time of the pairing is an expensive pairing. However, a few such expensive pairings may be necessary to cover all the flight legs in the schedule with the least cost. Consequently, the main objective of the crew pairing optimization is to select a set of pairings that contains all the flight legs exactly once, and has a least cost close to the total flying time of the schedule (Gopalakrishnan and Johnson, 2005).

II.1.4. Schedules

Barnhart et al. (2003) state that, “Just as to a duty period is a sequence of flights with sit times in between, and a pairing is a sequence of duties with layovers in between, a schedule is simply a sequence of pairings with periods of time off in between” (p. 6). However, the main difference between schedules and the other components is that schedules are associated directly with individual crew members, rather than crews in total. In addition to flying duties, each crew member has other duties, such as training, office or ground duties, as well as individual needs for time-off throughout the schedule period. A typical month would include vacation time, training time, medical testing and so on. Thus, assigning flight pairings to crew must take into account the needs and preferences of individual crew members.

In addition to individual crew member preferences, other constraints similar to those when doing duties and pairings are limits on the maximum monthly flying time, the maximum duty period time in 7 days, 10 days and in a month, the minimum number of consecutive days off, the minimum total number of days off in a month and a quarter, the minimum rest between pairings and so on. As a result, the cost of a schedule is clearly different from the other elements, and the focuses within duties and pairings are on actual flight times or labour costs, whereas the cost of a schedule is considered more as a function of crew satisfaction and of workload balance (Barnhart et al., 2003).

Since the crew-scheduling problem involves many rules and regulations and flight data is complex in its structure and large in size, it is normally solved in two phases, the crew-pairing phase and the crew-rostering phase (crew assignment). In the first phase, the crew-pairing problem is solved to find a set of minimum-cost pairings, in which each scheduled flight over the time horizon is included in exactly one pairing. After solving the crew pairing problem, the optimal set of pairings is generated to cover all flights throughout a schedule period. The second phase is the process of allocating individual crew members to the optimal pairings (Bazargan, 2010) and is called the crew assignment/ rostering stage. Along with vacations, training time, rest periods and other breaks and extended individual work, schedules are created, typically spanning a period of one month.

II. 2. The Crew Pairing Problem

The crew pairing problem (CPP) focuses on minimization of costs by complying with all rules and regulations, and covering all flights on the flight schedule, as well as effectively using all resources to obtain high quality solutions (Deveci and Demirel, 2018b). For airliners, the crew pairing process minimizes the operational crew costs through the most cost effective pairing, comprising all the scheduled flights, which reduces the idle time (connect time over the limitation), increases the flight time of a duty period, reduces deadheads, and excess costs. Therefore, CPP is the cost-determining phase of the crew scheduling problem (Deveci and Demirel, 2018b).

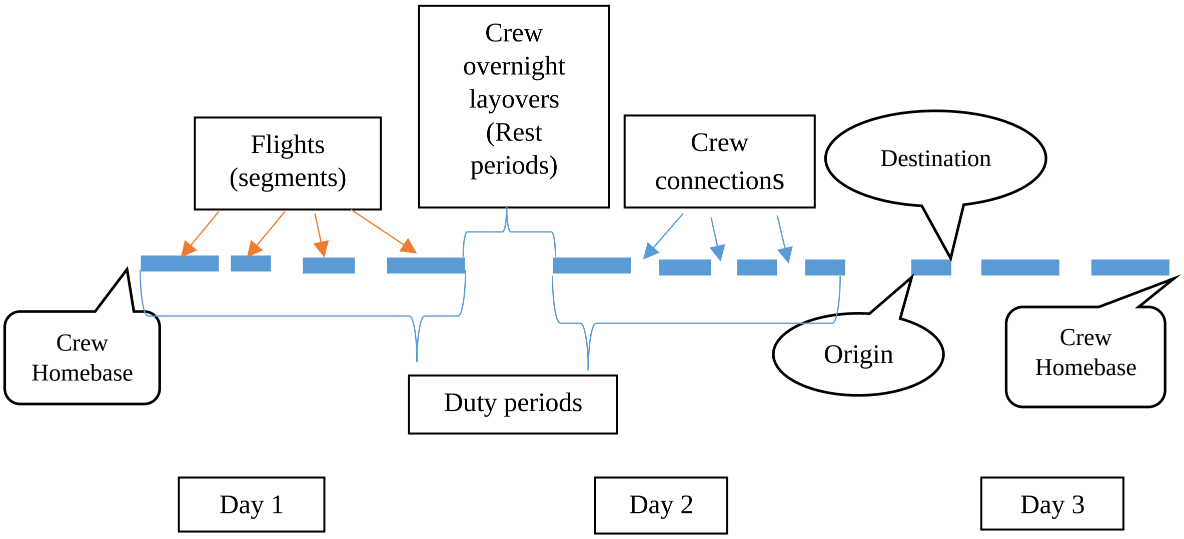


Figure 4: Example of a typical crew pairing (Abdelghany and Abdelghany, 2010)

II.2.1. Characteristics of the Crew Pairing Problem

The crew pairing problem, as shown in Figure 4, above, varies between different airlines; therefore, the solution methods also rely on the airlines' size or a set of data bases, the network structure, the rules and regulations, as well as the cost structure (Kasirzadeh, Saddoune and Soumis, 2015). There are some differences between the problem in North America, in Europe and elsewhere. The major differences between the situation in North America and in Europe are highlighted below (Andersson et al., 1998).

II.2.2. Crew Category

On any flight, there are two different groups of crew, those who are the cockpit crew (captains, first officers) and the cabin crew (pursers, hostess). Crew in one category usually cannot substitute for crew in the other category and different rules are applied to each category; hence, the pairing problem decomposes by crew category. Even within the same group, different rules may also apply to different crew members. For example, in the case of Vietnam Airlines (VNA) there are two groups of cockpit crew, namely, Vietnamese (VNese) and Non-Vietnamese (non-VNese) from different countries. These two groups of VNA cockpit crew are paid differently and their employment agreements are different as well.

II.2.3. Fleet

Since cockpit crew (pilots) are typically only qualified to operate one type of aircraft, the fleet also decomposes the crew pairing problem into many sub problems for each fleet (this thesis solves the subproblem of the Airbus 320 fleet only). However, cabin crew are often qualified to fly several types of aircraft.

II.2.4. Network Structure

Typically, there are three major network structures or route systems, the *hub-and-spoke* network, the *point-to-point* network, and hybrid network structure. Major North American airlines operate the hub and spoke route structure, in which a few main large airports are the hubs and are regularly connected to each other and to many other smaller airports (the spokes). The hub is also the home base of crew, whereby they start their duty periods at the hub, flying to the spoke and returning to the hub on the following flight; these two consecutive flights are called return flights or round-trip routes.

In the hub-and-spoke network, passengers travelling from a spoke to another spoke must transfer at the hub for a second flight to their destination, and the timetable of flights is constructed suitably in order that passengers arriving at a hub can have enough time to connect to many outgoing flights without much delay. In practice, a large number of flights arrive at the hub within a short time interval and, shortly after that, a large number of flights leave the hub. This enables the airline to offer transportation between most spokes via the hub with minimal waiting time and this kind of network structure creates very many possible pairings, as explained by Graves et al. (1993a).

However, in Europe, carriers do not operate the interactive hub-and -spoke networks as the US counterparts because of the geography of the route network, the hub and spoke route system is not popular; instead European carriers operate network focused on airports within the member state in which the carrier is based and linking that member state to a large number of other domestic, European and international destinations (Reynolds-Feighan, 2009) While the low cost carriers (LCCs) in the US as well as in Europe have tended to offer point-to-point flights rather than indirect, connecting air services (Reynolds-Feighan, 2007). . .

This structure avoids circuitous routings; therefore, the number of possible pairings is smaller in typical European problems than in typical North American problems. However, since there are many flights on an aircraft during an operating day, the total flight time of these flights exceeds the duty period limitation of a crew, a deadhead (a second crew travels as passengers on the flights which the first crew operates) often incurs to conduct the rest of flights after the first crew have finished their duty periods. Recently, most large airlines, such as VNA, operate some combination of the two route systems (Cook and Goodwin, 2008) called hybrid network structure which brings together several of the characteristics of the hub-and-spoke network and point-to-point networks. VNA adopting this network structure has two hubs named HAN and SGN, where passengers can connect through them. VNA also schedules nonstop service between spoke cities with nonstop flights. Following such structure, several flights carry only local traffic, and some other flights carry a mix of local and connecting traffic.

II.2.5. Rules and Regulations

Governmental rules and collective agreements impose conditions on the generation of pairings, such as the limitation regarding the length of the duty period and the rest requirements between duty periods in pairings. In the United States, the regulations of the Federal Aviation Administrative (FAA) are most important, but relatively simple and identical for all airlines. In addition, airline specific collective agreements also affect the legality of the pairing slightly; however, the structure of the problem is the same for all major North American airlines.

In contrast, in Europe very detailed collective agreements are usually much stronger than governmental regulations. Typically, the collective agreements change often and the rules are quite different from airline to airline.

II.2.6. Regularity of the Timetable

North American airlines operate the same flights Monday to Friday and over the weekend a subset of these flights is operated; therefore, the domestic US crew pairing problem is normally solved in three stages: daily, weekly exceptions and transition. However, European airlines have a lower frequency, so that it is very common for them to operate particular flights only once, twice or three times a week.

II.2.7. Cost Structure

The cost of a pairing in the U.S.A has two components, in which the first component, similar to the cost of a duty period, is the maximum of three quantities, and the second component represents the extra costs associated with the rest period between two duties, such as meals and lodging. The first quantity of the first component is $\sum_{d\in P} b_d$ the sum of the costs of the duties contained in the pairing. The second quantity $f_p * TAFB,$ is the product of the total elapsed time or Time Away From Base of the pairing and the constant f_p which is a fraction of the total elapsed time, ranging between 45 per cent and 65 per cent, depending on sector lengths for domestic pairings. In contrast, international flights tend to have longer TAFB, so the constant f_p is typically low, ranging between 15 per cent and 30 percent (Budd, L. et al., 2017). The third quantity is the minimum guaranteed number of minutes per pairing, which is the product of the Number of Duty Periods (NDP) in the pairing and a fixed Minimum Guaranteed (MG) number of minutes per duty period. Formally, the cost of a pairing p is calculated as below:

$$c_p = \max\left\{\sum_{d\in P} b_d\,, f_p * \ TAFB, NDP * \ mg,\right\} + \sum_{\substack{\hat{d}\ \in p, \bar{d}\ \in p \\ \hat{d}\to\bar{d}}} e\left(\hat{d},\bar{d}\right) \quad \text{(Barnhart \textit{et al.}, 2003)}$$

Where $\widehat{d}\,, \overline{d}$ represent the duty periods in the pairing p, $\hat{d} \ \to \ \overline{d}$ indicates that the duty period $\hat{d}$ occurs immediately before $\bar{d}$ in the pairing p, and $e\left(\hat{d},\bar{d}\right)$ is the extra cost associated with the rest between two duty periods $\hat{d}$ and $\bar{d}$.

However, European counterparts apply a fixed salary for each crew and in this case, the cost of a pairing is either *NDP* or 1 (Barnhart et al., 2003).

II.2.8. Problem Categories

Crew pairing problems are classified as daily, weekly, and monthly problems in traditional solving approaches. The daily schedules are usually assumed for domestic flights, which are identical for all of the days during the planning horizon, and the minimum-cost pairings are generated based on the scheduled flight legs for a single day. In the weekly problem, it is assumed that the flights, such as international flights, are repeated weekly, and the pairing problem is solved based on the scheduled flights for one week. The monthly problem has a time horizon of a full month. Due to vacation periods and variations in the flight schedules, the monthly time horizon is the most realistic one (Kasirzadeh, Saddoune and Soumis, 2015).

In the daily problem, the sets of flights with a frequency of at least four days per week are considered and treated as though they all operate daily. The purpose of this stage is to find a minimum cost set of feasible pairings, so that every flight in this set is covered exactly once. The pairings in this solution are then assumed to be repeated daily (Barnhart et al., 2003).

For medium or long-haul international flights, there are two or three flights per week and it normally takes more than 5 hours flying time each, such as the hypothetical international flights for pilots based at Pudong International Airport (PVG) in China and flying to Kansai (KIX) in Japan , which take 6 hours flying directly, and the crew cannot come back to their home base at the end of the duty period since the total flying time of 2 flight legs PVG – KIX – PVD exceeds the limit of 8 hour of a duty period . They have to stay over in KIX for 1 night; therefore, this pairing spans 2 days.

Finally, the monthly problem has a time horizon of a full calender month. Multi-day pairings can be problematic at the end of a monthly flight schedule. which constructs pairings to cover flights spanning the changeover from one monthly flight schedule to another. In addition, due to vacation periods of pilots and variations in the flight schedule, the monthly time horizon is the most realistic.

In these problems, the objective is always to minimize pay-and-credit and the labour costs beyond the minimum required flying time.

II.2.9. Solution of the Crew Pairing Problem

After solving the crew-pairing problem, an anonymous minimum cost set of pairings of all flight legs is obtained. Table 3, below, illustrates a partial feasible pairing solution of the crew pairing problem, in that each pairing is a sequence of combined flight legs. All pairings must satisfy all governmental rules and regulations, and union and company agreements.

Two pairings of flight legs in Table 3, below, represent two typical types of pairings, in which pairing 1 spans from the previous schedule (Dec 2013), with the first duty occurring in that period to this schedule Jan 2014 with the second duty (Duty 1) of the pairing 1 taking place in the current schedule, while pairing 5 occurs totally in the current schedule and also has two duties. In addition, at the end of the current schedule there are pairings where the first duties belong to this schedule but the second duties take place in the next schedule. These pairings, having one of two duties in different schedule periods, are called broken pairings. Pairings may also contain only one duty if the last arrival of the duty is the original home base of the crew, or two duties if the last arrival of the first duty is a place other than the crews' home bases.

In detail, the broken pairing 1 currently has one duty consisting of three flights, flight number 1, flight number 153 and flight number 154 in the flights list for Jan 1st 2014. Duty 1 starts at 1:30 GMT from KIX (Japan) and finishes at 12:20 GMT at HAN. This means that the crew had been assigned to a duty of the previous schedule which originated at HAN, flying to KIX, and that they stayed the previous night in KIX. Pairing 5 also includes two duties, spanning from Jan 1st to Jan 2nd 2014. The first duty is Duty 3, including three flight legs, legs 9, 10 and 188. Duty 3 begins with leg 9, departing at 6:30 GMT from SGN and ends at 14:10 GMT at KHH. The crew stays overnight at KHH and flies the flight leg 189 of Duty 79 on the Jan 2nd duty list back to SGN on Jan 2nd (23:30 GMT of Jan 1st but Jan 2nd local time). The crew is allocated to the other two flights and finish their duty at 10:10 GMT at their home base, SGN. Although each airline has different rules, they share the same common main characteristics in their pairings (Kasirzadeh, Saddoune and Soumis, 2015).

Table 3: A simple example of VNA crew pairing solution

KIX	**HAN**	**HAN**	**VII**	**VII**	**HAN**		**SGN**	**REP**	**REP**	**SGN**	**SGN**	**KHH**	**KHH**	**SGN**	**SGN**	**CAN**	**CAN**	**SGN**
1/1/2014		1/1/2014		1/1/2014			1/1/2014		1/1/2014		1/1/2014		1/1/2014		1/1/2014		1/1/2014	
1:30	7:15	10:30	11:00	11:50	12:20		6:30	7:30	8:20	9:20	11:15	14:10	23:30	2:30	3:20	6:10	7:20	10:10
Leg 1		Leg 153		Leg 154		…	Leg 9		Leg 10		Leg188		Leg 189		Leg 181		Leg 182	
Duty 1		Duty 1		Duty 1		…	Duty 3		Duty 3		Duty 3		Duty 79		Duty 79		Duty 79	
Pairing 1		Pairing 1		Pairing 1		…	Pairing 5		Pairing 5		Pairing 5		Pairing 5		Pairing 5		Pairing 5	

II. 3. The Crew Rostering Problem

II.3.1. Introduction

In the second phase, monthly schedules (rosters) for crew members are personally constructed. This called the crew rostering problem (in European airlines) or the crew assignment problem (in North American airlines). In the rostering problem, in addition to the pairings, vacations and pre-assigned activities, such as training periods, ground duties or medical appointments, are allocated properly into the timeline of each individual crew member (Gamache *et al.* (1999).

Similar to the pairings generation process, the rostering (assignment) problems must always fulfil the complex rules and regulations of the airlines and the FAA or the CAAV. While the most important objective of the pairing problem is cost minimization, the objectives of the crew rostering problem are broader, including the quality of life aspects for crew members together with expense reduction. Hall (2002) emphasized that the crew assignment problem focuses greater on satisfying crew requests and balancing the distribution of workload among the flight crew. Consequently, the combinatorial objectives are cost efficiency and social quality for the crew members (Kohl and Karisch, 2004).

Table 4, below, shows an example of the solution of assignment (rostering) problem, where the anonymous pairings are assigned to individual crew members after considering rest periods, vacations and other activities, such as training and ground duties or reserved duties (Maenhout and Vanhoucke, 2010). In this stage, all of the flights in the pairings must be assigned to all crew and every crew member is provided with a suitable roster, usually 4 weeks in advance of the first flight's departure. The crew assignment process typically respects both the airline perspective of operational cost minimization and the crew member requirement of social quality (Maenhout and Vanhoucke (2010). Moreover, the qualifications of individual crew members and the number of crew required for each function of the airplane must be strictly compliant with regulations (Gamache et al., 1999).

Table 4: A simple example of crew rostering solution

KIX	HAN	HAN	VII	VII	HAN		SGN	REP	REP	SGN	SGN	KHH	KHH	SGN	SGN	CAN	CAN	SGN
1/1/2014		1/1/2014		1/1/2014		...	1/1/2014		1/1/2014		1/1/2014		1/1/2014		2/1/2014		2/1/2014	
1:30	7:15	10:30	11:00	11:50	12:20		6:30	7:30	8:20	9:20	11:15	14:10	23:30	2:30	3:20	6:10	7:20	10:10
Leg 1		Leg 153		Leg 154		…	Leg 9		Leg 10		Leg188		Leg 189		Leg 181		Leg 182	
Duty 1		Duty 1		Duty 1		…	Duty 3		Duty 3		Duty 3		Duty 79		Duty 79		Duty 79	
Pairing 1		Pairing 1		Pairing 1		…	Pairing 5		Pairing 5		Pairing 5		Pairing 5		Pairing 5		Pairing 5	
Capt. Hung		Capt. Hung		Capt. Hung			Capt. Hunt		Capt. Hunt		Capt. Hunt		Capt. Hunt		Capt. Hunt		Capt. Hunt	
F.O. Son		F.O. Son		F.O. Son			F.O. Ben		F.O. Ben		F.O. Ben		F.O. Ben		F.O. Ben		F.O. Ben	

In contrast to the pairing problem, the rostering problem is resolved in various ways, following different approaches (Maenhout and Vanhoucke, 2010). Two general approaches are normally used:

- The Bidline approach, which is mostly applied by North American airlines, constructs anonymous schedules firstly and then publishes them to crew members. The crew bid on these schedules, after which the airline uses the bids to allocate the schedule to individual crew members. However, a drawback of this approach is that some bidlines cannot be assigned totally to crew members, due to conflicts with pre-assignments and vacations.
- Personalized schedules, commonly used by European airlines, often consider the preferences of the individual crew regarding tasks and special activities, such as

vacations and training periods, before constructing the roster for each crew (Hall, 2002). Two types of personalized schedules are rostering and seniority-based. Rostering is based on a fair sharing of duties to maximize the global satisfaction of all crew-members and is concerned with the objective of fairness, which is measured by the number of satisfied preferences. Seniority-based personalized schedules, in contrast, focus the priority on the satisfaction of the more senior crew members (Kohl and Karisch, 2004);(Kasirzadeh, Saddoune and Soumis, 2015).

Recently, the personalized schedule is increasingly being adopted by North American airlines, as it has many advantages for both the crew and the airlines. This approach concerns the requests of crew-members, as well as preassigned employee activities, vacations, training, and unfinished pairings from the previous month during the construction of the schedule. This decreases the number of schedule adjustments and increases productivity (Kasirzadeh, Saddoune and Soumis, 2015).

II.3.2. Problem Definition

The crew rostering problem focuses on generating accurate rosters for every crew-member, and is concerned with the requests of crew and the balance of tasks allocation. Therefore, the objectives of the crew rostering problem are a combination of the economic effectiveness and crew satisfaction (Kohl and Karisch, 2004).

The general solving method is based on the ‘generate – and – optimize principle’, in which as many as possible legal rosters are firstly created from the sub-problem by partial enumeration or a constrained shortest path problem. After that, a set partitioning problem in the master problem is solved to obtain the optimal roster for each crew member, such that all rules and regulations, as well as personal references, are fulfilled (Kohl and Karisch, 2004).

Figure 5, below, provides an overview of the crew rostering process, which includes input data, objectives, constraints, and output. Activities consisting of flight pairings and all information about individual crew members, relating to pilot qualification, certificates, ground duties, training, medical examination, as well as vacations, is provided in advance, as the input data for the rostering problem. Furthermore, the records of accumulated flight time and days off for each crew member are also retrieved, in order to check them before assigning any pairings to the crew. Because the flight time of the crew is not only the basis for his/her salary and experience, but also the object of the health and safety rules and regulations, it must be correctly recorded after any flight duty.

In addition to flight time, days off (rest time) before and after a flying duty are also allotted properly, as required by the rules and regulations. The general rules and regulations of each airline are clearly elaborated in the Flight Operation Manual (FOM) and the scheduling process must comply strictly with these constraints.

Each airline has its own objectives, which are classified into four groups, as detailed in the following section. The solution of rosters must fulfil the rules and regulations and satisfy the objectives of the crew assignment problem. The objectives can appear in the objective function or be implied as constraints.

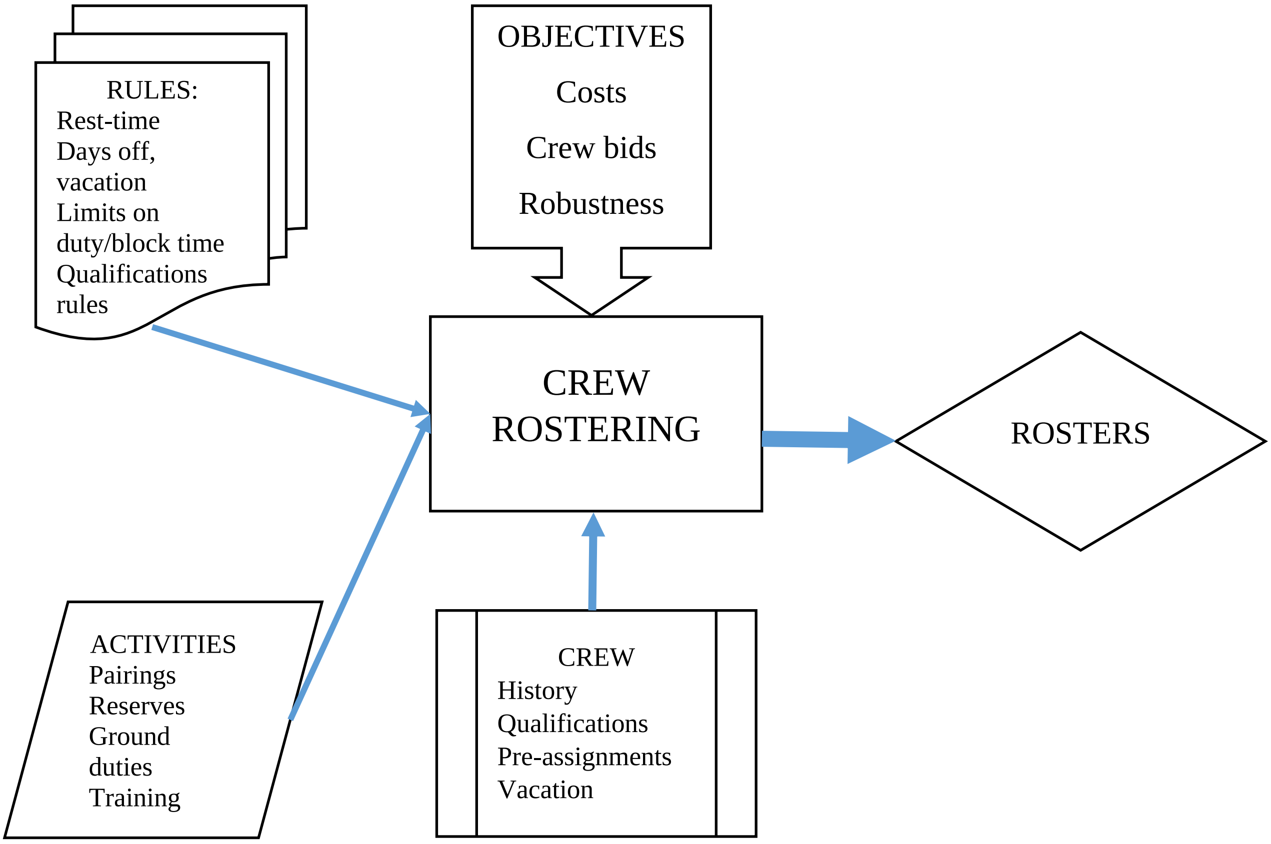


Figure 5: Representation of the airline crew rostering problem (Kohl and Karisch, 2004)

II.3.3. Typical Rules and Regulations

One of the complications of the crew rostering problem is that there are many rules and regulations relating to the crew and rosters. They can be classified into horizontal, vertical, and artificial rules, depending on how they impose on one roster or several rosters. Rules applied to one roster and one crew-member are called horizontal, whereas vertical rules are applied to several rosters and crew members. Artificial rules are additional constraints of both types, which are aimed at discarding feasible solutions which may be of poor quality and, thus, directing the formulation to the better solutions more efficiently.

Horizontal rules are constraints on a single roster, only relating to characteristics of a crew member and properties of the assigned activities for which the roster is generated. Thus, a roster is regarded as legal if it satisfies the horizontal rules and regulations. Most of rules of the crew assignment problem are of this type; some large European airlines have up to 100 horizontal rules and regulations. However, only a few of the important rules are presented below:

1. The compatibility between crew member, task, and time: these rules can be easily applied to check the legality of a person, task, and time combination before scheduling. Almost all incompatible cases may occur due to a lack of qualification or pre-assignments. However, there is a special but important situation when a crew member changes a role during the planning period, such as from F.O. to captain.
2. Break time or connection time between two consecutive tasks. The combination of consecutive tasks must be determined after calculating the time interval between them. The break time is to allow the crew proper rest between two jobs immediately following each other. This calculation is quite complicated since it involves many aspects.
3. Rest day patterns. The connection of pairings and other activities makes a working period. A working period may include a short haul or medium haul pairings and other tasks, whereas only a long-haul pairing spans a whole working period. After any working period finishes, days off are given. In addition, ‘fixed’ rest day patterns are applied by some airlines, such as VNA, whereby crew members sign contracts to work for a period of *n* days and then are off-duty for *m*-days.
4. Accumulated values. In airlines, rules and regulations about block time[1] and days off have strict limits regarding the maximum or minimum hours or days. For example, the VNA limitation on block hours is 100 hours for a period of 28 days or 1000 hours in any calendar year. Specifically, the cumulative duty period to which a crew is assigned will not exceed 60 hours in any seven consecutive days.

---

[1] Block time (block hours) is the time from the moment the aircraft “wheel chocks away” for taxiing out from the tarmac at the airport of departure to “wheel chock on” at the tarmac of the arrival airport. Whenever there is a push-back, block time is counted from the beginning of this procedure.

Vertical rules are those involving more than one roster, normally to a subset of rosters. The basic groups of these rules are crew complement, qualification related constraints and global constraints.

- Normally, the crew complement of a short or medium haul flight pairing is one captain and one first officer. However, long-haul pairings with flying times longer than the daily flying time limitation (11 hours at VNA) require two captains and one first officer (F.O.) or training tasks also need two captains and one F.O.
- Qualification constraints: qualification conditions are a flying license; sometimes certificates to operate to specific destinations or airports are required.
- Global constraints: global constraints are applied to the whole solution. These constraints include an upper bound on the cost of solutions, constraints on overall bid satisfaction and horizontal rules being defined on more than the planning period.

Artificial rules: additional conditions are imposed on airlines, depending on legislation or contractual agreements, to achieve a better quality of schedule. Artificial rules are dependent on the expertise of the schedulers and restrict the solution space in order to discard any poor-quality solutions. The factors are concerned with the robustness of the schedule and a gain in the efficiency of the solution methods.

II.3.4. Objectives of the Crew Rostering Problem

As mentioned above, the objectives of the problem may vary from one carrier to another. Typically, there are four kinds of objectives in the crew rostering problem (Kohl and Karisch, 2004). Generally, carriers would combine some of them and the objective function may include all of the elements or account for some objectives in the global constraints, while optimizing the others (Gamache et al., 1999).

The first objective normally concentrates on real costs. The situation which often occurs in scheduling problems is unassigned activities; this is called ‘open time’. However, although there is open time in a solution, it does not mean that the solution is infeasible. If a pairing is unassigned, it can be resolved by certain plans, such as overtime scheduling, reducing the reserves duties, hiring temporary staff or even cancelling the flight. Therefore, minimization of open time is always a main component of the rostering objective function and overtime payment is also a real cost component. Because, in Europe, the permanent crew are usually paid a fixed salary for up to a certain number of block hours per year, when their total flying time exceeds the block hours, overtime payments are made. In order to avoid overtime payment for some crew members, while others fly considerably less than their block hours, an “equal assignment” approach is obtained (Kohl and Karisch, 2004). El Moudani et al.

(2001) minimize the operational cost of schedules, together with achieving crew satisfaction and Maenhout and Vanhoucke (2010) also aim to minimize the total operational cost and maximize schedule quality achievement. Guo et al. (2006) identify the most important variable costs, as overnight accommodation costs, proceeding costs (transfer) and compensation for time away from home.

Instead of overtime payment for permanent employees, some airlines hire temporary crew, called "freelancers", who are paid according to their number of flying hours. Even though open time is solved, it is not an optimal solution. Thus, the challenge is how to distribute tasks among crew members equally to reduce the cost of freelancers. In addition, training activities are also involved in positioning (deadheading) flights if simulator assignments are away from base, with hotel costs and expenses needing to be considered. Base constraints are involved in real cost objectives as well, since crew members moving from one base to the other require deadheads, extra compensation and accommodation expenditure. Some authors combine several objectives in their models including the minimization of operational cost and the deviation of working time (Souai and Teghem, 2009).

Secondly, the robustness of the solution is also one of the objectives for airlines. A general explanation of the robustness of the solution in the literature is that it is immune to small perturbations. It normally occurs within airlines that crew members with 'hard' rosters are more likely to report sick than crew assigned 'easy' rosters. "This kind of knowledge is often problematic to formalize, but should be taken into account in the formulation of the objective function by penalizing 'hard' rosters" (p.234) (Kohl and Karisch, 2004). Ehrgott and Ryan (2001) and Ehrgott and Ryan (2002) model the robustness by penalizing crew changing aircraft during a duty period for crews of Air New Zealand.

Thirdly, some special roster attributes are also modelled in the objective function of equal allocation. Fairness is the most important factor in the European airlines' assignment problem. The crew rosters should aim to be equal regarding flight time, departure time and arrival time, as well as destination. Iijima et al. (2013) focus on fair working conditions.

Finally, the objective is related to individual preferences, since crew-members can often suggest preferences with respect to their schedule before the planning process. (Kohl and Karisch, 2004).

III. LITERATURE REVIEW

"*At their core, the crew pairing and crew assignment models are set partitioning and set covering models with one constraint for each task to be performed (i.e. a flight or pairing to be covered) and one variable for each feasible combination of the tasks*" (p.14) (Barnhart et al., 2003).

III. 1. The Crew Pairing Problem

Three factors make the crew pairing problem difficult. Firstly, a lot of rules and regulations must be enforced and a mathematical formulation of a legal pairing is almost impossible (Gopalakrishnan and Johnson, 2005), which leads to difficulty in determining whether a combination of tasks is feasible. Secondly, these problems often contain a huge number of variables – often hundreds of millions or more. Finally, the requirement of all integer variables makes the solution process more complicated. The process is usually separated into two distinct phases: (1) Pairing Generation and (2) Optimization.

Crew pairing models are typically formulated as a Set Partitioning Problem (SPP) or a Set Covering Problem (SCP), in which each flight leg is a constraint (a row) and each feasible pairing is a variable (a column) (Kasirzadeh, Saddoune and Soumis, 2015). It means that when a set of feasible pairings is generated, a flight leg may belong to many feasible pairings, and then a minimum cost subset of the feasible pairings or an optimal pairings set is chosen, so that every flight is included in exactly one chosen pairing.

Let F be the set of all flights in the schedules and P is the set of all feasible pairings. Decision variable x_{p_j} is equal to 1 if pairing p_j is included in the solution and 0 otherwise. Also c_{p_j} is the cost of pairing p_j.

The crew-pairing problem is:

$$\min \sum_{p_j \in P} c_{p_j} x_{p_j} \tag{1}$$

$$\text{S.t} \qquad \textstyle\sum_{p_j : i \in p_j} x_{p_j} = 1 \qquad i \in F \tag{2}$$

$$x_{p_j} \in \{0,1\} \qquad j \in P \tag{3}$$

The objective function (1) is to find a least cost subset of feasible pairings covering all flights in the schedule. The constraint (2) ensures that each flight i in the set of flights F belongs to only one pairing in order to avoid duplicated flights in the solution.

The greatest challenge to solving the crew-pairing problem is the huge amount of feasible pairings being generated. Therefore, before the 1990's, only a subset of the pairings was firstly constructed, and then local improvement heuristics was applied to process the current solution and search for a better one. 'The small set partitioning problems can be solved quickly since the LP relaxation of set partitioning problems with a small number of rows frequently have integral or near-integral solutions' (p. 20)(Barnhart et al., 2003). This method was classified as the row approach by Gopalakrishnan and Johnson (2005).

Anbil et al. (1991) and Gershkoff (1989) introduced a local improvement heuristic for the crew pairing problem. Firstly, a feasible solution to the set partitioning problem is always constructed manually by modifying the solution used in the previous planning period. Then, the heuristic randomly selects a small number of pairings in the current solution and searches for a better solution for the flights covered by that subset of the pairings. The process of enumerating all possible pairings for the subset of flights and solving the small set partitioning IP to optimality by the branch-and-bound method is repeated until no further improvement is found, or until some present time limit is reached.

Housos and Elmroth (1997) report a very successful iterative scheme. Ball and Roberts (1985), Etschmaier and Mathaisel (1985), Gershkoff (1989), Graves et al. (1993b) also apply the row approach to solve the crew pairing problem.

There are two drawbacks to this method. Firstly, only a small subset of the flights is considered in each iteration. Therefore, these heuristics require a large number of iterations to find a good solution. Secondly, the local search heuristics do not provide a lower bound for the best possible solution value.

Hu and Johnson (1999) apply the column generation approach to generate all the candidate pairings for the subproblem and call in an optimizer to solve the subproblem to optimality. Thus, a new subproblem with a better objective value than the current one is produced in the next iteration. Once the optimal solution to the whole problem has been achieved, the process of subproblem generation and optimization can be stopped.

To overcome the two above obstacles, more global approaches have been developed to generate pairings that cover all of the flights (Barnhart et al., 2003; Kasirzadeh, Saddoune and Soumis, 2015). Two types of networks approaches have been introduced for generating legal pairings.

The first is a flight network whereby an arc presents each flight in the schedule and other arcs represent possible connections between flights. The second type of network, a duty period network, has an arc for each possible duty period and arcs representing possible overnight connections between the duties.

Minoux (1984); (Desrosiers *et al.*, 1991) propose a typical flight network in which nodes represent the departure and arrival of each flight, as well as a source *s*, and a sink *t*. The source node is connected to the departure node of every flight, starting at a home base (hub) and the arrival node of each flight finishing at that home base is connected to the sink. In addition, there are other arcs representing legal connections between those flights in the schedule. Connections between flights are legal if the destination airport of the first flight is the same as the departure airport of the second flight and the time interval between two flights is a break time within a duty period, or a rest time between two consecutive duty periods of a pairing. The break time and rest time must adhere to the break time and rest time limitations.

Figure 6, below, shows a partial flight network for the following flight schedule.

FLIGHT 1: AIRPORT A – AIRPORT B 08:00 – 09:00

FLIGHT 2: AIRPORT B – AIRPORT C 10:00 – 11:00

FLIGHT 3: AIRPORT C – AIRPORT D 13:00 – 14:00

FLIGHT 4: AIRPORT D – AIRPORT A 15:00 – 16:00

The network of a two-day schedule of each flight has solid arcs representing flights and dotted arcs representing possible connections between flights. The connection arcs are only connecting the two flights, which arrive and depart at the same airport, one after the other within the connection time limits. Each arrival node has two connections starting from it, one to the next departure and the other to the same departing flight one day later. If airport A is a crew base, a source node *s*, and a sink note *t,* are added to this network. Then, connecting *s* to the departure node of every flight arc, which begins at Airport A and connecting the arrival node of every flight that arrives at A to node *t*.

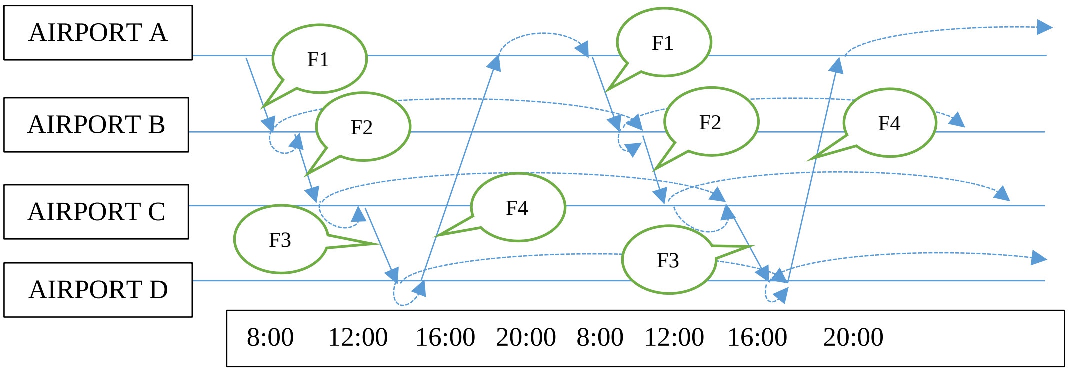


Figure 6: Flights network

Every legal pairing is represented by some *s-t* path in the flight network, but many *s-t* paths do not represent legal pairings. The network structure guarantees that two flights that do not have their respective arrival and departure at the same airport are not connected, but it does not catch the violation of other rules, such as the maximum number of flying hours allowed in a duty period, or the maximum TAFB in a pairing.

The duty period network is typically applied to international problems. Nodes represent the start or end of a duty period and an arc is for each possible duty period. Connection arcs between duties are included if two duties can be flown consecutively by the same crew.

Lavoie, Minoux and Odier (1988), Barnhart et al. (1994), Vance *et al.* (1997b) present this network. A pair of duties will have a connection arc between them if the arrival airport of the first is the departure airport of the second and the time between flights is a legal overnight rest. The required duration of an overnight rest might be a function of the attributes of the duty period that precedes it and possibly other attributes of the pairing. Unlike the flight network, it is possible to build explicitly into the duty period network, with the requirements involving the preceding duty period.

Figure 7, below, shows a two-day duty period network for the schedule shown in Figure 6, above. The solid arcs represent duty periods and the dotted ones are for connections between duties. The lighter solid arcs are the single-flight duty periods corresponding to each of the four flights in the schedule, while the darker solid lines correspond to two additional duties, one composed of flights 1 and 2, and the other composed of flights 3 and 4. It is possible to build more duty periods from this set of four flights. The single flight duty period arcs arrive much later than the corresponding flight arcs in the flight network, because the time of the overnight rest in the duration of the duty arc is included.

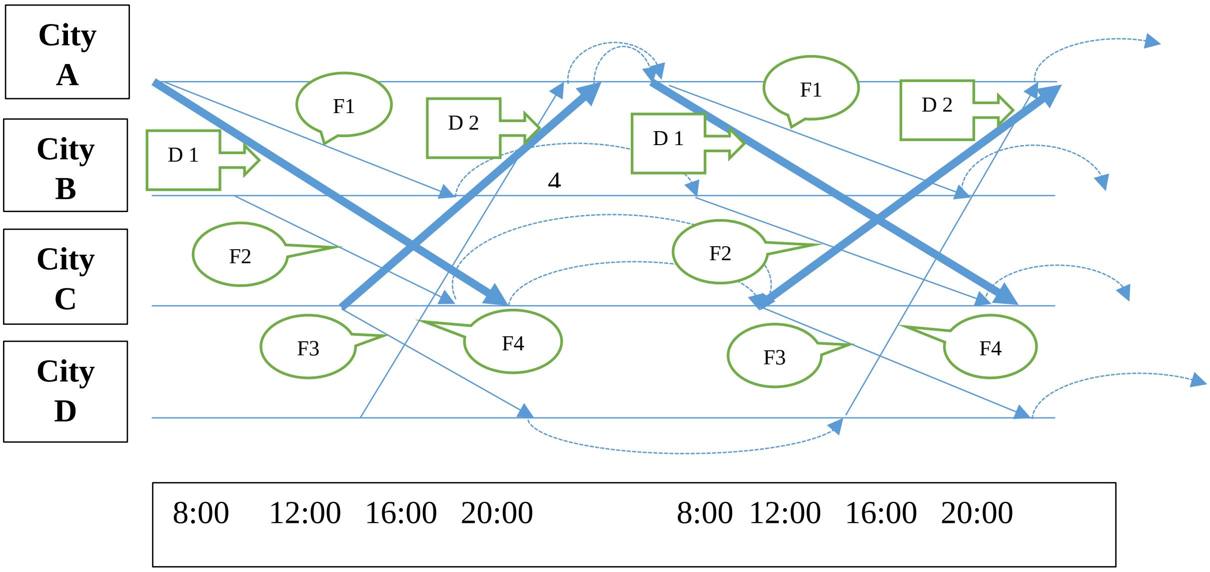


Figure 7: Duty Period Network

To generate a duty period, a depth-first search approach on the flight network could be applied for each flight arc, and all duty periods starting from this flight are constructed. When the duty period is extended by a flight in the flight network, all of the other duty feasibility rules are checked and satisfied to enumerate all duty periods.

Pairings can be enumerated from either the flight network or the duty period network and the generation is started from every flight or duty that begins at a crew base. In the duty network, where each flight starts or ends at a crew base, a source and a sink node are added. The source node is connected to the departure node of the flight of each duty period that originates at the specified crew base, and the arrival node of the flight of every duty that ends at that crew base is connected to the sink.

Anbil et al. (1991) and Gershkoff (1989) provide methodologies for the crew pairing problem that generate only a subset of pairings, since all of them cannot be handled in reality. An easy way to achieve this is by generating pairings only on a subset of flights, since it is substantially more difficult to generate a subset of pairings that cover all of the flights in the schedule. Andersson et al. (1998) give some details on how this operation was carried out at Carmen Crew Pairing.

Another method is the random generation of pairings proposed by Klabjan et al. (2001). To extend pairings, connections are randomly chosen using the connection times as greedy estimates; this means that the probability of selecting a connection depends on the connection time. This is because ‘short connections are more likely to yield pairings with low cost, the smaller the connection time, the larger the probability of selecting the connection' (p. 10)(Barnhart et al., 2003). Specifically, in hub-and-spoke flight networks, there are many connections; thus, the connection selection strategy has to be implemented carefully.

The pairing generation stage is followed by the stage of solving the crew pairing optimization problem. Several state-of-the-art solution methodologies are presented below.

III.1.1. The TPACS/TRIP Approach

Trip Pairing for Airline Crew Scheduling (TPACS) was developed by Rubin (1971); (Rubin, 1973), to solve the crew pairing optimization problem. TPACS was a creative improvement over previous scheduling methods and was used by several major airlines to save substantial operational costs on crew scheduling.

The trip re-evaluation and improvement program (TRIP) is based on the outstanding development of TPACS in crew pairing technology. TRIP applies the row approach to improve the initial set of pairings. About five pairings are chosen in each iteration to generate all legal pairings from the flights in the selected pairing, to form a subproblem set-partitioning problem. The set partitioning problem is then solved to find the optimal solution.

The TRIP methodology had been enhanced gradually to become one of the most successful technique for solving the crew pairing problem (Anbil et al., 1991). It successfully handles weekly pairing with crew base constraints. Subproblem selection, pairing generation, and subproblem optimization have been improved by the TRIP technology and this saves several millions of dollars for American Airlines. Hardware and software technologies have also contributed significantly to increasing TRIP iteration speed.

A hybrid algorithm is used to determine quickly "a good set of Lagrangian multiplier and a good warm start solution for the linear programming" (p. 319) (Gopalakrishnan and Johnson, 2005). The current TRIP solver can generate more than 500 legal pairings in a second and can solve a subproblem of 100000 columns quickly using column-screening techniques. The column screening approach considers pairings that have reduced cost less than some threshold value and heuristic, based on Lagrangian relaxation to compute the reduced costs efficiently.

However, the TRIP approach adopts the subproblem method; therefore, its major drawback is local optimal solutions, as it can prevent further enhancement of the solution. Some heuristics have been developed to reduce the impact of the local minima (Gopalakrishnan and Johnson, 2005).

III.1.2. Linear Programming Algorithms

Forrest (1989) presents the SPRINT method to solve the large-scale linear programming problem quickly. Some crew pairing systems use SPRINT to solve large-scale crew scheduling linear programming successfully. The problem of 850 flights and 5.5 million legal pairings have been solved by the SPRINT faster than by some of the traditional methods, due to fast performance in each iteration although the SPRINT iteration counts were quite high (Anbil, Tanga and Johnson, 1992).

The algorithm is based on the idea of solving a sequence of considerably smaller linear programs of a large-scale linear program. In the SPRINT methodology, a small subset of the columns of the problem is chosen and a linear program is solved by the dual simplex method on this subproblem to price out all columns of the original problem. If the current solution is not optimal for the original problem, a new problem is set up, based on the columns in the optimal basis and adding a small subset of the columns that have close to zero reduced cost. A bucket data structure is introduced to keep the columns based on reduced cost.

With the development of optimization solvers and computers, dynamic column generation techniques implicitly consider all possible pairings in solving the LP relaxation (Barnhart *et al.*, 1994; Desaulniers *et al.*, 1998). To generate columns, the set partitioning problem with all possible pairings is referred to as the master problem and a restricted master problem is one that considers only a subset of the possible pairing columns. The column generation algorithm to solve the crew pairing LP includes the following steps:

- Step 1: Solve the current Restricted Master Problem to find the optimal solution on a subset of all columns.
- Step 2: Solve the Pricing Subproblem by generating one or more columns that may improve the solution. If no columns are found, STOP: the LP relaxation is solved.
- Step 3: Construct a New Restricted Master Problem from the restricted master problem by adding the columns generated in solving the subproblem and return to Step 1 (Barnhart et al., 2003).

Steps 1 and 3 can be solved by using optimization software, whereas in step 2 the network structure of problems are different between airlines; therefore, it must be ‘tailored’ to produce columns.

There are many algorithms for solving the restricted master subproblem, but the volume algorithm has proven to be very successful in practice (Gopalakrishnan and Johnson, 2005).

Barahona and Anbil (2000) proposed the volume algorithm which is an enhancement of the sub-gradient algorithm to achieve primal and dual solutions (Gopalakrishnan and Johnson, 2005). The sub-gradient algorithm is well known for obtaining lower bounds for large-scale linear programs (Held and Karp, 1970;1971; Held, Wolfe and Crowder, 1974). However, convergence is a problem of the sub-gradient method, which does not have a properly defined stopping conditions, and the volume algorithm improves the stopping criterion and has a low computational cost per iteration. It is quick and does not require large computer memory, as well as having been successfully proved to solve large-scale set-partitioning and set-covering linear programming problems. US Airways and Southwest Airlines use the implementation of the volume algorithm as a submodule to solve the linear programs to solve the crew pairing problems in a crew pairing system developed by IBM (Anbil, Forrest and Pulleyblank, 1998).

The pricing subproblem in step 2 is the selection of the pairings or columns to be added to the restricted master problem to create a new restricted master problem in step 3. Two main factors in this step are “what the criteria to select the pairings are, and how to find pairings that fulfil these criteria” (p. 22) (Barnhart et al., 2003). Pairings used to be selected by the reduced cost criterion; however, Bixby et al. (1992) propose a new selection formula, which is that the pairing cost is divided by the sum of the dual values of the legs in the pairings to choose the pairing, as this method reduces considerably the number of iterations. Hu and Johnson (1999) suggest a primal-dual algorithm, relying on a dual feasible vector to choose the columns with the lowest reduced cost. The second question has two approaches to search for pairings, which satisfy the selection criteria. The first one applies the combination of a shortest path algorithm and the second is the brute force approach of enumerating the pairings (Barnhart et al., 2003).

Subramanian and Sherali (2008) developed a new deflected sub-gradient scheme to generate good quality dual solutions for a linear programming model of a large-scale airline crew planning problem. A phenomenon called *dual noise* is identified and is an explanation for the stalling behaviour of the software as still far from the optimality. The suggested method provides several features such as a self-correcting target value, a minimal number of algorithmic parameters and empirically observed accelerated convergence to good-quality dual solution.

1. Pricing with Shortest Path Algorithms

Shortest path approaches have been previously used only for the reduced cost criterion; nevertheless, multi-label or constrained shortest path methods are applied in a number of algorithms to find the best pairings in the pricing problem on especially structured networks (Desrochers and Soumis, 1988). The difference between multi-label and single label shortest path approaches that it is compulsory to keep many paths to each intermediate node in the network. There are two types of paths named nondominated and dominated paths. Two paths to the same node have all labels identical except one has more elapsed time than the other, by dominance the one with larger elapsed time can be eliminated and the nondominated path is 'better' with respect to all the costs and rules. In consequence, all non-dominated paths to each node between source and sink must be kept track of.

Similarly, Lavoie, Minoux and Odier (1988) also solved effectively the pricing problem over duty-based networks by generating a columns algorithm via multi-label shortest-path sub-problems. This approach is especially very good with a moderate number of duty periods. Likewise, Vance *et al.* (1997b) used a set of duty periods in the first stage of his two stage decision-making process model for the crew scheduling problem to cover the scheduled flights before constructing the pairings for these duty periods.

Barnhart et al. (1994) applied a duty-based network to solve the international crew problems which consist about two to three times as many duties as flights. Hjorring and Hansen (1999) created a black-box rule system that simply implemented the variety of rules and regulations and combined column generation with a pricing sub-problem, based on a duty network and a k^{th} shortest path algorithm to solve some realistic data sets.

2. Pricing by Enumeration

The drawback of the multi-label shortest path method is that the complication of the pairing feasibility rules and the cost structure prevent the dominated paths to occur clearly. Because if the dominated paths appear, they can be eliminated easily and the number of nondominated paths is decreased and the better one can be chosen quickly. Accordingly, another method is to generate all the pairings; however, it is impossible to enumerate all the pairings of medium and large crew pairing problems. Therefore, it is necessary to invent strategies to prevent this (Barnhart et al., 2003).

Marsten (1994) and Anbil, Forrest and Pulleyblank (1998) produced crew pairing optimizers that only enumerate partial pairing in pricing and used the reduced cost criterion (as mentioned above), while Makri and Klabjan (2001) applied the selection criterion, which was introduced by Bixby et al. (1992).

III.1.3. Integer Programming Methodologies

The problem that makes the crew pairing problem unable to be solved properly by traditional branch and bound methods is the size or the large number of flights of the instances. Therefore, only a rather small subset of columns having the small-reduced cost have been solved and it is also very hard to fathom the entire branch-and-bound tree on this restricted set. Some heuristic solution approaches have been developed, such as the branch-and-cut algorithm, to achieve optimal integer solutions. They are classified into three general types. The first class is algorithms that generate columns a priori for a subset of pairings and solve this subset by an integer program (Barnhart et al., 2003). Hoffman and Padberg (1993) also use this approach to solve an instance of 68 data sets from four major airlines; however, billions of pairings were enumerated and the integrality gaps were up to 5%, and it took much effort to achieve a good integer solution. Klabjan et al. (2001) implemented the approach of Hoffman and Padberg (1993) and this forms the second group of approaches.

In this class, firstly, dynamic column generation is used to solve the LP relaxation of the set partitioning problem to optimality or near optimality. Secondly, the subset of generated columns is solved by branch-and-bound to achieve the optimal IP solution. Barnhart et al. (1994) applied this to the international crew pairing problem and Ryan (1992a) solved the rostering problem. Other works, by Desaulniers et al. (1998) , Desrosiers et al. (1991) , Gamache et al. (1999) , Gamache and Soumis (1998) , Gamache et al. (1998) , Ryan (1992b), Vance *et al.* (1997a) and Anbil, Forrest and Pulleyblank (1998), have been developed using the branch-and bound framework with column generation. Klabjan et al. (2001) proposed an algorithm which enumerated millions of random pairings and then selected and solved these columns based on their reduced costs.

However, these approaches have not assured that a good solution, or even a feasible solution, will be found among a subset of columns that give a good LP solution. Consequently, branch-and-price approaches have developed as the third class of algorithms (Barnhart et al., 2003). This procedure enhances the enumeration strategy, similar to branch-and-bound at each node, where it is solved by LP relaxation, but using column generation for the huge constraint matrix. Branch and price methodology is broadly applied to transportation, scheduling, and combinatorial optimization (Gopalakrishnan and Johnson, 2005).

1. Branching Techniques

As generating pairings at each node in the branch-and-bound tree, an appropriate branching rule for the pairing generation procedure plays a crucial role in solving large-scale integer programming problems. Three branching rules for the crew pairing optimization problem have been proved successfully in several crew pairing systems (Gopalakrishnan and Johnson, 2005) and are discussed below.

i. Follow-on Branching

The *branch on follow-ons* branching heuristic was developed by Ryan and Foster (1981), who originally designed it for solving general set-partitioning problems. However, it has been applied usefully to crew pairing optimization. The rule starts with the optimal LP relaxation solution of a subset of pairings of the crew pairing optimization; normally, the solution of many millions of pairings is examined. Given *r* and *s* flights, the follow-on is the second flight *s* of the consecutive flights flown in a pairing. On the first branch, *r* and *s* are compulsory, appearing consecutively in a pairing in which the *s* flight departs after the flight *r*. On the second branch, these two flights do not include the same pairing. Vance *et al.* (1997a) proposed the follow-on branching rule as being a valid one.

Rasmussen *et al.* (2011) proposed a new integer programming model, based on the subsequence generation. In subsequence generation, the number of permitted subsequent flights is restricted leading to the number of pairings in the problem decreases and then new attractive subsequence is dynamically added to the problem. Consequently, the number of possible pairings is increasing and improving the solution quality. 19 real-life instances from Air New Zealand are tested and encouraging results are achieved to prove that the presented approach is a viable alternative to column generation.

ii. Timeline Branching

Timeline branching was presented by Klabjan *et al.* (2001), based on the SOS branching of Beale and Tomlin cited by Gopalakrishnan and Johnson (2005). In timeline branching, a set of all pairings which contain a flight f_i is denoted $\bar{P}_{f_i}$ and sorted in order of the connection time with the flight f_i. The connection time ct_{p_j} of a pairing $p_j \in \overline{P_{f_i}}$ is the difference between the departure time of a flight f_{i+1} and the arrival time of a flight f_i of every pair of consecutive flight f_i and f_{i+1}. For a given flight f_i, $\overline{P_{f_i}}$ is partitioned based on the connection time ct_{p_j} and a length of time τ; all pairings in which $ct_{p_j} \leq \tau$ is in one branch and the others to the other branch. All pairings in $\overline{P_{f_i}}$ and in the first branch are set to 0 and exactly one in the second branch is set to 1. Pairings with the last flight f_i are also set to 0. Klabjan *et al.* (2001) proved that the timeline branching is a valid branching rule under the assumption that no two flights depart from an airport at the same time.

iii. Strong Branching

In the strong branching rule, the choosing of the branching variable is obtained after the number of dual simplex iterations for each branching candidate in order to estimate the lower bound changing (Bixby et al., 1995; Linderoth and Savelsbergh, 1999). Klabjan et al. (2001) generalized the strong branching rule with the combination of follow-on and timeline branching rules.

2. Branch-and-Cut

This exact algorithm combines a cutting plane method with a branch-and-bound algorithm to solve efficiently a large number of integer programming problems (Gopalakrishnan and Johnson, 2005). The branch-and-cut method has proven to successfully obtain the optimal solution for a large set partitioning problem, such as the crew pairing optimization problem. Hoffman and Padberg (1993) developed a branch-and-cut solver to solve crew-pairing instances, with up to 1000 rows and 1.05 million variables to obtain optimality. The five main components of the branch-and-cut solver are a branch-and-cut optimizer in which the user-supplied formulation is processed firstly and then tightened, a linear programming solver, a heuristic to solve good integer feasible solutions quicker, a cut generation procedure to narrow the linear program relaxation, and a branching strategy to determine the search tree. When using a cut generation procedure, the LP solution is narrowed and often leads to an integer solution.

3. Branch-and-Price

The branch-and-price methodology is quite similar to the branch-and-cut approach; however it focuses on pricing or dynamically generating columns instead of row or constraint generation, as in branch-and-cut (Gopalakrishnan and Johnson, 2005). The procedure to generate variables dynamically regards as generating cutting planes for the dual of the current LP relaxation. Barnhart et al. (1998) intensively reviewed these methods, while Desrosiers et al. (1995) provided the branch-and-price framework to apply in routing and scheduling problems.

Sets of columns are set aside as solving the LP relaxation since the large numbers of columns, of which many have a value of zero in an optimal solution, make the problem more complicated and inefficient to solve. Instead, a sub-problem is solved separately for the dual LP to obtain the good columns entering the basis. When such columns are found, the LP is re-optimized, otherwise the current LP solution is optimal. The complicated column generation techniques for linear programming in integer programming solution methods are applied in the branch-and-price approach (Johnson, 1989).

Vance *et al.* (1997a) provided a heuristic framework combined with the branch-and-price approach to achieve near optimal integer solutions for the crew-pairing problem. Pairings are generated by a multi-label shortest paths algorithm on the duty period network. As combining dynamically variable and cutting planes generation in LP-based branch-and – bound, the implemented technique is known as branch-and-cut-and-price (BCP) (Ladányi, Ralphs and Trotter, 2001).

III.1.4. Parallel Approaches to Crew Pairing

As the crew pairing problems have the large number of flights legs and destinations, the computation time to solve the problem increases subsequently. Together with hardware development, the introduction of parallel computing for crew pairing has achieved significant implementation in the computational capacity of optimization solvers.

Since pairing generation is the most time-consuming section of crew pairing algorithms, the distribution of the starting legs at the crew home bases into the processors is the core idea of a parallel algorithm, in order to enumerate all the pairing beginnings with the assigned starting legs. However, to generate all the pairings starting with a given flight leg takes a long computational time, and so load balancing algorithms are required, for example, crew pairing problems with as few as 300 flight legs for hub-and-spoke networks or 2000 flight legs for point-to-point networks can take as much as 10 to 20 hours of CPU time to solve (Barnhart *et al.*, 2003). Goumopoulos, Housos and Liljenzin (1997) developed a pulling algorithm applying the master/workers paradigm, in which the master divides each leg amongst the workers and if a worker is idle, it sends the request to the master for a new starting leg. Klabjan, Johnson and Nemhauser (2000) implemented a randomized pricing strategy for an algorithm of parallel primal-dual. The computational result is very impressive on a variety of crew pairing optimization problems.

Dynamic domain decomposition methods have been applied successfully in a parallel pairing generation technique (Klabjan and Schwan, 2001). The combination of a parallel pairing generation algorithm with branch-and-price algorithms was developed by Klabjan (2001) and Barnhart et al. (2003).

Some other research, Alefragis et al. (1998), Sanders, Takkula and Wedelin (1999), Alefragis et al. (2000) concentrated on parallelizing the pairing enumeration and the Lagrangian decomposition algorithm of Andersson et al. (1998).

III.1.5. Other Approaches

There are some different approaches which efficiently solve the crew pairing optimization problem. Wedelin (1995) proposed an approximation algorithm for solving large-scale 0-1 integer programming problems. This algorithm solves the 0-1 set partitioning problem in a direct way compared to a sequence of LP's.

Desaulniers et al. (1997) modelled the crew-pairing problem as an integer nonlinear multi-commodity network flow model. The model, with additional resource variables and a large subset of constraints, is solved by a branch-and-bound algorithm being extended from the Dantzig-Wolfe decomposition principle.

Vance *et al.* (1997b) applied the duty network to model the crew pairing problem in which a set of duty periods separate the flight legs and a set of pairings separate the duty periods. The optimal pairing solution is achieved from good sets of duty periods.

The duty period formulation is described as below:

$$\min \sum_{d \in D} b_d \; x_d + \sum_{p \in P} \hat{c}_p \; z_p \qquad (1)$$

$$\text{Subject to: } \sum_{d_i \in d} x_d = 1 \qquad i \in F \qquad (2)$$

$$\sum_{p \; d \in p} z_{p_k} = x_d \qquad d \in D \qquad (3)$$

$$x_d \in \{0, 1\} \qquad d \in D \qquad (4)$$

$$z_p \in \{0, 1\} \qquad p \in P \qquad (5)$$

Let the binary $x_d = 1$ if duty d is chosen, and 0 otherwise, and the binary $z_p = 1$ if pairing p is chosen, and 0 otherwise. *F* is the set of flight legs in the schedule, *D* is the set of duty periods, and *P* is the set of all legal pairings. b_d is the cost of duty period d, and $\hat{c}_p$ is the excess cost of a pairing p which is the difference between the pairing cost and the sum of the costs of the duty periods in the pairing p. Objective (1) minimizes the summation of the total cost of duties and the pay-and-credit cost (excess cost) of pairings. Constraint (2) imposes on each flight being covered by exactly one duty period and the constraint (3) also enforces that each duty is covered by one pairing only.

The number of rows and columns in the duty period formulation above are more than the ones in the set partitioning problem model (SPP), but the duty network model has LP bound a little tighter than the SPP model (Vance *et al.*, 1997b). The algorithm to solve the crew-pairing problem obtains the Dantzig-Wolfe decomposition technique and imposes the set partitioning constraints on the duty period model. The computational results of Vance *et al.* (1997b) produce an integer optimal solution, whereas the LP relaxation of the set-partitioning model does not. It seems that the duty period model achieves a tighter bound on the optimal IP solution. Due to the relevant concept of this duty period formulation to our thesis, we have implemented the modal of Vance et al (1997b) to solve the VNA crew pairing problem.Saddoune *et al.* (2011a) introduced the aggregation method to solve the crew scheduling problem in one stage, based on the combination of column generation and dynamic constraint method. The computational results on the real-life data prove the significant saving on the total cost, but the processing time is much longer than the sequential approach. Saddoune *et al.* (2011b) implemented a model and the bi-dynamic constraint aggregation method of column generation with a neighbourhood structure to solve the crew scheduling in one stage. The computational time decreases by an average factor of 2.3 while improving the quality of the computed solutions.

The other trend of CPP methods is to combine heuristic and exact methods in solving crew pairing problems. Aydemir-Karadag, Dengiz and Bolat (2011) introduced three algorithms for the crew pairing problem, the first two of which are the knowledge based random algorithm (KBRA) and the hybrid algorithm (HA), both combining heuristics and exact methods. In KBRA, the solution search space is reduced by the knowledge received from the past, whereas HA applies some mechanisms in components of genetic algorithms to generate a high-quality legal pairings search space. A zero-one integer programming model of the set covering problem is then used to choose the minimal cost pairings from the reduced search space. The third technique is the integration of column generation (CG) with KBRA and HA respectively. The computational results indicate the effectiveness and efficiency of HA and CG-HA in solving CPP in terms of computational cost and solution quality.

Erdoğan et al. (2015) developed the optimization –driven heuristic or model-based metaheuristics that combine metaheuristic and exact optimization methods to solve a large scale crew pairing problem of up to 27,000 flight legs.

Zeren and Özkol (2016) proposed a model as the set – covering problem and the pricing sub-problem as a shortest-path problem. They combine heuristic and exact algorithms to effectively solve a duty-flight overnight connection graph. The approach reduces deadhead flight time and the number of international overnights.

Quesnel, Desaulniers and Soumis (2016) suggested four branch-and-price heuristics, of which three of them are developed from the branching scheme heuristics. A retrospective branching (RB) method is introduced to detect and revise poor branching decisions made in the search trees without backtracking. The result shows that the RB heuristics performance has smaller gaps between the value of the computed integer solution and the value of the computed linear relaxation solution than the other tested heuristics and is more reliable at finding good-quality solutions in reasonable times than the other branching method.

Agustín *et al.* (2016) introduced a meta - heuristic approach based on biased randomization to solve the CPP. The results of a real-life experiments shows the algorithm decrease overall crew flying times and the required number of accompanying crew compared to the pairings currently applied by the company. Agustin, Juan and Pardo (2017) developed several heuristics based on the Variable Neighbourhood Search approach.

Demirel and Deveci (2017) proposed heuristics which improve a dynamic-based genetic algorithm to solve medium scale scheduling problems. The partial solution approach along with a deadhead-minimizing pairing search, based on the development of genetic algorithm variants and a memetic algorithm, successfully handle medium sets of crew pairings and obtain better quality solutions than pervious methods.

Deveci and Demirel (2018a) applied the genetic algorithm variants and a memetic algorithm (MA) hybridising GA with hill-climbing to solve the CPP. The problem is solved in two stages and the empirical results on a set of benchmark real-world instances prove the MA is the best performing approach.

III.1.6. Compare and contrast types of models

Deveci, M. and Demirel, N.Ç. (2018b) conduct a survey of airline crew scheduling problems and have a summary of types of models as detailed below:

The model approaches include mixed and integer programming models, zero-one integer linear programming model, non-linear integer programming, stochastic modelling, fuzzy sets and modelling, shortest path problem etc.

Mixed-integer programming (MIP) is one where some of decision variables are constrained to be integers in the optimal solution. Mixed-inter linear programming (MILP) problems are generally solved by using and LP-based branch-and-bound algorithm such as Quesada and Grossmann (1992), Barnhart et al. (1998), Vielma et al. (2008). Many other algorithms are also used to solve MIP. A number of studies use mixed- or integer-programming in specific parts of the solution of airline crew scheduling. The crew scheduling problem can be modelled as an LP problem but the solution must be an integer, therefore, the LP model is converted to an integer later. An LP model has one objective, a linear equation must be maximized or minimized.

Non-linear integer programming is a technique developed for the cases where at least one of the constraints of the decision model, or objective function, is nonlinear. These problems are real world problems and applying non-linear formulation increases the complexity of the problem. While increasing the problem solving time and allowing cost model to be relatively flexible, but does not add a hight added value. Therefore, studies mainly in the literature are solved with a linear model.

Stochastic crew scheduling: since airline companies often have to deal with irregularities, such as adverse weather condition, airline carrier delays, late arrival of aircraft, or diversion of aircraft, stochastic models take the influence of the random factors into account and the objective is to utilize that information to obtain robust solutions having a better ability to withstand disruptions.

Finally, fuzzy set theory approach to the crew scheduling, Teodorovic and Lucic (1998) proposed a fuzzy set theory approach to the aircrew monthly rostering problem. The basic algorithm is a modified version of the ‘day-by-day’heuristic method. An approximate logic algorithm is used by the decision maker to determine the power of a particular pilot assignment preference in a specific roation. The fuzziness found in some tables and use the fuzzy cluster theory to solve the problem of assigning roots pilots.

III. 2. The Crew Rostering Problem

In general, the crew assignment problem has received less attention than the crew-pairing problem. Moreover, since the number of constraints and objectives of the crew assignment are many more than the crew pairing, the crew assignment problem is more complex than the other one. Some crew assignment problems which have been researched (Kasirzadeh, Saddoune and Soumis, 2015) are discussed below.

Kohl and Karisch (2004) introduced the basic and simplified form of the problem, by which the crew rostering problem includes a set of tasks T, containing flight pairings, ground duties and reserved days off or vacations, and a set of all crew C, who are allocated proper tasks. The challenge is how to obtain a legal roster, $R_j \subset T$ for each crew member, $j \in C$ which partitions T so that:

$$T = R_1 \cup R_2 \cup \ldots \cup R_{|C|}$$

The objective function on the costs of each roster R_j requires minimization of the total cost of all rosters c_j, and a legal roster that must satisfy the horizontal rules and regulations. Solving the problem by a generate-and-optimized approach is typically one in which a set of feasible rosters $\mathcal{R}_j$ ($R_j \in \mathcal{R}_j \subset T$), has been firstly generated for each crew member j, and then the set partitioning problems with constraints on the subsets and scheduled tasks being fixed in time is solved for the best combination of rosters with respect to the linear objective function.

The mathematical model for generalized set partitioning provides a global view of the problem. Let $y \in \{0,1\}^n$ be the decision variable has value of 1 if a roster R_j is chosen and the cost of the roster R_j is c_j . The constraints matrix A, illustrated in Table 5, below, of the set partitioning problem is denoted by the $\{0,1\}$ with m rows and n columns, in which $m = \sum_{i \in T} i + \sum_{j \in C} j$ and $n = \sum_{j \in C} \mathcal{R}_j = \sum_{j \in C} \sum_{k \in \mathcal{R}_j} R_{jk}$. The k^{th} column of A has a value of 1 in row *i* if the task *i* is covered by the roster R_{jk}. This is called the *activity constraint* and has a value of 1 in row *j* if crew member *j* is allocated to the roster R_{jk}which is the *assignment constraint*.

Table 5: The constraints matrix A

$\boldsymbol{Crew_1}$			$\boldsymbol{Crew_2}$				$\boldsymbol{Crew}_{\|C\|}$					
$\boldsymbol{R_{11}}$	R_{12}	R_{13}	R_{21}	R_{22}	R_{23}	…	$R_{\|C\|1}$	$R_{\|C\|2}$	$R_{\|C\|3}$			
1	1	0	1	0	1	…	0	1	0	=	1	Activities
0	1	1	0	0	1	…	1	0	1	=	1	constraints
1	0	1	0	0	0		0	1	0	=	1	
						…						
1	1	1				…				=	1	Assignment
							1	1	1	=	1	constraints
						…				=	1	

The set partitioning problem is presented in the compact form as below:

$$(SPP)\ \ z^* \ := min\,\{c^t\, y: \quad Ay = e, \quad y \in \{0,1\}^n \} \tag{6}$$

Where c^t denotes the vector of costs of the rosters and e denotes the right-hand side vector of all 1, since only one roster is assigned to each crew member and each task must be covered by exactly one roster.

As mentioned above, the legal roster fulfils only the horizontal rules, whereas vertical rules involve a subset of rosters or the whole schedule. The basic model is extended to express the various vertical rules, which are generalizing the set partitioning constraints. For example, some tasks require more than one crew member, such as an instruction task, which requires two captains and one FO, as showed in Table 6, below. In this case, the right-hand side vectors have values of more than 1. Thus, the problem is seen as the generalized set partitioning problem:

$$(GSPP)\; z^* := \min\{c^t\, \mathrm{y}: \quad Ay = b,\; y \in \{0,1\}^n\} \qquad (7)$$

Where b denotes the vector of integer right-hand sides and while the right-hand side of the assignment constraints are all still 1, the right-hand sides of the activity constraints can take any positive integer value. Further details, are presented in Kohl and Karisch (2004).

Table 6: The constraint matrix A with the set partitioning constraints

$\boldsymbol{Crew_1}$			$\boldsymbol{Crew_2}$				$\boldsymbol{Crew_{\lvert C \rvert}}$					
Captain (CP)			Captain				First Officer (FO)					
$\boldsymbol{R_{11}}$	$\boldsymbol{R}_{12}$	$\boldsymbol{R}_{13}$	$\boldsymbol{R}_{21}$	$\boldsymbol{R}_{22}$	$\boldsymbol{R}_{23}$	…	$\boldsymbol{R}_{\lvert C \rvert 1}$	$\boldsymbol{R}_{\lvert C \rvert 2}$	$\boldsymbol{R}_{\lvert C \rvert 3}$			
1	1	0	1	0	1	…	0	0	0	=	1	CP Activities
0	1	1	0	0	1	…	0	0	0	=	1	constraints
1	0	1	0	1	0		0	0	0	=	2	
						…						
0	0	0	0	0	0	…	0	1	0	=	1	FO Activities
0	0	0	0	0	0		1	0	1	=	1	constraints
0	0	0	0	0	0	…	0	1	0	=	1	

III.2.1. Bidline approach

For the bidline approach, Beasley and Cao (1996) introduced an integer programming formulation and applied Lagrangian relaxation and a sub-gradient method to solve the problem. This method combines with a tree search algorithm to obtain the optimal solution. Campbell, Durfee and Hines (1997) proposed a meta-heuristic algorithm, based on simulated annealing to generate a bid line system for FedEx. The objective is to minimize the number of bidlines and the amount of unassigned flying time. Jarrah and Diamond (1997) proposed a priori column generation and a heuristic set partitioning problem (SPP) approach to maximize the covered credit time as minimizing the number of bidlines in the bidline assignment problem. The method is ‘semi-automatic’, as the subset of columns generated by the user idea and this is used by a large US airline for producing good solutions.

A two-phase method was presented by Christou et al. (1999) for bidline generation of Delta Airlines to obtain the maximization of the average total value and the quality of the bidlines. In the first phase of the algorithms, good bidlines were constructed and then the complement of valid bidlines construction was produced in the second phase.

In 2004, Weir and Johnson (2004) introduced a three-phase approach to generate the bidline. Firstly, a mixed integer problem produced tentative bid-line patterns and these bidlines were solved in the second phase to achieve the final schedules covering all the pairings. After two phases without success, the uncovered pairings were integrated into the schedules in phase three (Kasirzadeh, Saddoune and Soumis, 2015).

Some other authors also have applied two heuristic algorithms for solving the SPP based bidline schedule, such as Elhallaoui *et al.* (2005). Boubaker, Desaulniers and Elhallaoui (2010) used a standard branch-and-price algorithm, firstly to achieve integer solutions and then, in the second algorithm, dynamic constraint aggregation (Elhallaoui et al., 2005) was combined with the result of the first one. The result of the largest instances with 564 pilots and 2924 pairings proved that dynamic constraint aggregation heuristics provide a better solution than those of the standard branch-and-price heuristic.

Achour et al. (2007) proposed an exact solution method, using column generation for the preferential bidding system. A sequence of LPs was solved with senior priority, and the schedules of the employees were fixed as the algorithm was progressing. After a tentative maximum score for a crew had been established, all the feasible schedules with that score were enumerated for that crew-member. The solution of real data sets with up to 91 pilots showed a significant improvement in quality.

III.2.2. Rostering approach

In the context of the rostering problem, there are six general solution approaches that were discussed by Gamache *et al.* (1999). In the first method of roster construction, high-priority activities are assigned to high-priority employees first and then are continued to be allocated to other crew members (Marchettini, 1980; Glanert, 1984). The second approach to construct rosters is day by day assignment, in which, for each day of the month, pairings are assigned to individual crew members being chosen from a list of available crew members (Nicoletti, 1975; Buhr, 1978; Sarra, 1988; Gamache and Soumis, 1998). Rosters being constructed monthly for each crew member one after another is the third method. Moore, Evans and Noo (1978) and Byrne (1988) picked one crew at a time from the ordered list of crew with higher seniority first and build a fixed roster for a whole month. The next model is the combination of the third method of monthly roster construction and the second method to re-optimize rosters day by day (Giafferri, Hamon and Lengline, 1982).

Ryan (1992a) and Ryan and Falkner (1988) developed the fifth method for the rostering problem. In the new method, a set of feasible rosters is generated first for each crew member by a heuristic and then this set of rosters is modelled as a generalized set partitioning problem. The problem is solved by using specialized integer programming. The linear relaxation and branch-and-bound method are described in detail by Ryan and Falkner (1988). Gamache and Soumis (1998) applied column generation to solve the linear relaxation of the generalized set partitioning problem as the sixth approach. Columns are generated for each crew by solving a constrained shortest path problem on a network in which pairings are nodes and possible links between pairings are arcs with weights being free periods. An integer solution is obtained by branch-and-bound technique.

Ryan (1992a) modelled the crew rostering problem with 55 crew members and 120 pairings as the generalized set partitioning optimization. The problem was solved from 2 to 3 hours. The other work of Day and Ryan (1997) solved the cabin crew rostering problem for Air New Zealand’s short-haul operations. They applied integer programming and allocated the days off first, before assigning crew-members to the pairings and other activities. This approach brought an efficient constructive base for fine – quality schedules as almost all the pairings had a one-day period.

Gamache and Soumis (1998) introduced a prototype method of column generation embedded in a branch and bound algorithm to solve the rostering problem optimally. A generalized SPP and column generation-based heuristics were applied to obtain good integer solutions for Air France.

El Moudani et al. (2001) described a heuristic bi-criterion method to solve a new mathematical formulation of the crew rostering problem. A combination of a genetic algorithm and hard constraints obtained reduced cost solutions, which produced acceptable levels of crew satisfaction.

König and Strauss (2000a) and König and Strauss (2000b) proposed a propagation technique to enumerate schedules implicitly. This heuristic was enhanced in the SWIFT ROSTER algorithm and achieved good solutions for medium and large European airlines.

Constrained shortest path algorithm and pricing method are unable to satisfy all rules and regulations; therefore, a Constraint Programming (CP) method was developed to solve the rostering problem. Lustig and Puget (1999) and Brailsford, Potts and Smith (1999) discussed thoroughly the algorithms and applications of the constraint satisfaction problems. Fahle et al. (2002) and Junker et al. (1999) also solved the crew assignment problem of a large European airline by CP to generate columns of the pricing problem. Junker et al. (1999) provided a framework of column generation based on CP. A roster for each crew is represented by a variable and specific constraints cover all of the rules and regulations being related to crew-members. A shortest path algorithm is also applied to create rosters, as this method decreases the search space dramatically.

Another study also applying column generation and CP was the Parrot project (1997) that was presented by Kohl and Karisch (2000), Sellmann et al. (2002). The selection of schedules in the master problem was programmed as a linear program and then CP was used to prune the search. Kohl and Karisch (2004) conducted a thorough study of the Carmen crew rostering system of KLM. The system included three main components, the rule evaluator, the generator, and the optimizer.

Lucic and Teodorovic (2007) developed a personalized monthly schedule and applied Simulated Annealing, Genetic Algorithms, and Tabu Seach techniques to test on numerical examples.

Maenhout and Vanhoucke (2010) applied Dantzig-Wolfe decomposition and suggested a metaheuristic scatter search algorithm to allocate specific rosters to each crewmember. The approach’s objectives were to minimize the total operational cost and to obtain an expected quality schedule. Iijima et al. (2013) proposed cell-based and graph-based models. The solution method was based on labelling algorithms and used the Gurobi Optimizer to find a solution for the small-scale data. The cell-based model solved the problem quicker than graphic one, due to fewer decision variables.

Armas *et al.* (2016) introduced a multi-start randomized heuristic to solve a real crew rostering problem. The algorithm satisfied realistic constraints, regulations, and rules as well as distributing fair workload rosters for crew-members.

III.2.3. Compare and contrast the solution methods of the crew rostering problems

Two main approaches being used to solve the crew rostering problems are mathematical programming and (meta) heuristics algorithms. When tackling the crew rostering problem from a mathematical-programming point of view, column generation techniques are usually employed due to impossible to enumerate all the possible rosters. A specific sub-problem is required for each crew member to take into account the individual features and preferences.

These subproblems are solved by a couple of methods such as resource constrained shortest path problems, dynamic programming, labeling algorithms, Lagrangian relaxation, and constraint programming. Even though, the column generation approach is largely used in the literature, it has some important drawbacks in real-life problems

Because of the complexity and difficulty of the crew rostering problem, metaheuristic approaches are recommended by several other authors including of genetic algorithm, tabu search, ant colony, and simulated annealing methods. Meta-heuristics are applied to improve functions used in problem solving. Therefore, these algorithms are supportive in certain parts of the problem rather than the fundamental solution.

Most airline crew scheduling problem include mathematical based heuristic (matheuristic) evaluations that utilize both heuristic and exact methods. Colum generation and integer programming are the most frequently used approaches

III. 3. Summary

The crew pairing and crew rostering problems are difficult for three reasons. First, it is quite difficult to determine whether a combination of tasks is feasible because the wide array of rules and regulations must be enforced. Second, these problems often have an enormous number of variables – often in the hundreds of millions or more. And third, these variables are all integer, more complicating to process the solution.

Therefore, large scale crew scheduling problem is solved in two stages: crew pairing and crew rostering. A fundamental reason for this separation is the excessive growth of the search space by optimization model. More emphasis has been given to crew pairing work due to economicallt and financially important, and computationally cheaper. Crew rostering researches provide fairness in work distribtion and employee satisfaction. However, there has not been many studies addressing these aspects.

Recently, a topic of robustness is emphasized by airline companies because FAA statistics indicate that the percentages of flight delays are increasing by many reasons. Therefore, airline carriers should concern all of these factors in schedule planning.

IV. The Crew Scheduling of Vietnam Airlines (VNA)

VNA has many special characteristics due to the history and the development of VNA through different stages. VNA originally belonged to the Vietnamese army during the Vietnam war and then has become a state-owned enterprise. Therefore, the systems of VNA combines many factors from American airlines and European air carriers as well.

IV. 1. Special Characteristics of the Crew Scheduling of VNA

The crew-scheduling process of VNA is still being solved semi-manually by personnel in the crew-scheduling department, with the process as illustrated in Figure 8, below. On closer inspection, the reason why VNA continues to schedule crew semi-manually becomes clear. In terms of the problems evidenced in the patterns of crew pairing by Andersson et al. (1998), the crew scheduling at Vietnam Airlines is very complex which combines several characteristics of European airlines and North American airlines as shown in Table 7, below, and this leads to the application of the professional software is not efficient, as detailed below.

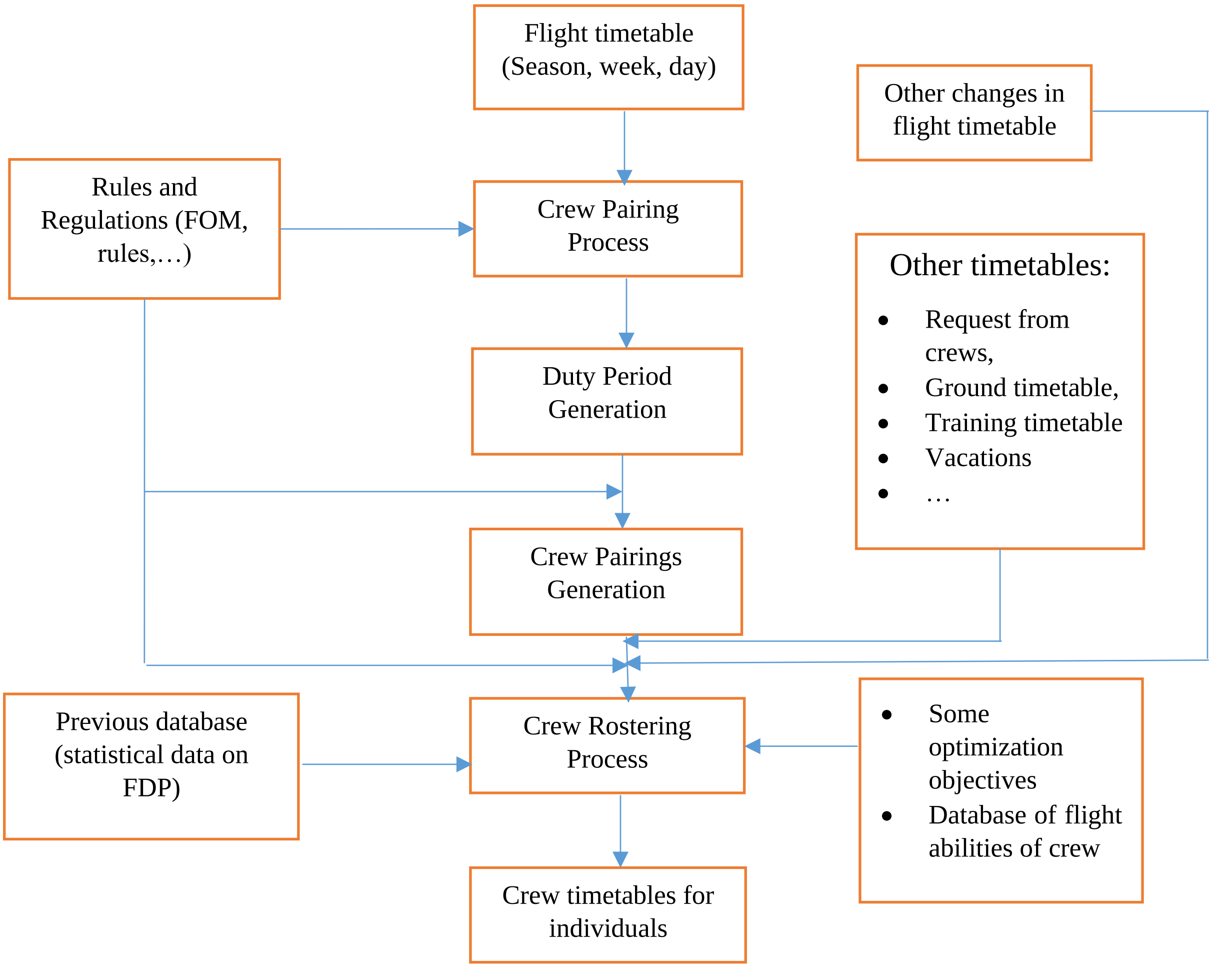


Figure 8: Cockpit crew scheduling steps at 919 Flight Division in Vietnam Airlines

IV. 2. Crew Category and Fleet (aircraft types)

Table 7: Comparison between North American, European airlines and Vietnam Airlines

	North America	Europe	Vietnam airlines
Crew category	Problem decomposes by crew category since crew of one category can generally not substitute another category.		
Fleet	For cockpit crew the problem decomposes by fleet type.		
Network structure.	Hub and spoke structure. Timetable constructed to make many connections possible.	Less structure, point to point route system.	Hub and spoke structure, but some routings have the point to point structure. Timetable constructed to make many connections possible.
Rules and regulations.	FAA regulations are the most important.	Complicated collective agreements which change often.	CAAV regulations are the most important. However, employment agreement also affects the scheduling process to allocate the pairings for foreign crew.
Regularity of the timetable.	Fixed timetable from Monday to Friday. Reduced timetable at weekends.	Less regularity from day to day.	Fixed timetable week by week. However, it is affected by charter flights.
Cost structure.	Payments based on credit hours.	Crews are paid a fixed salary.	Vietnamese crew are paid on credit hours Non-Vietnamese crew are paid a fixed salary.
The crew pairing problem types:	For North American Airlines, all	The flight schedules of European	To the VNA, domestics flights and short haul international flights are operated regularly

1. The daily problem. 2. The weekly problem. 3. The dated problem.	flights are flown almost every day of the week; thus, their flight operations resemble a daily problem.	airlines are more irregular. European airlines have a weekly problem.	every day, but there are only two to three medium or long haul international flights per week to each interntional destination.

Table 7, above, summarizes the differences and similarities between European airlines, North American airlines, and Vietnam Airlines (VNA). Details of the characteristics of VNA are represented as the following sections.

The crew scheduling problem of VNA is complicated, mostly due to the complexities of the crew categories. VNA has many fleets, as of September 2016, Vietnam airlines has a fleet of 82 aircrafts as detailed in Table 8, below, and each fleet requires pilots who qualify typically to fly only one specific aircraft in that fleet. Since scheduling process is conduced separately for each fleet, the more fleets an airline owns, the more the scheduling task increases.

Table 8: Vietnam airlines fleet

Aircraft	Total	Passengers/ aircraft	Routes

Airbus A321-200	56	266	Short-to-medium haul international and domestic
Airbus A330-200	8	280	Medium haul international and domestic
Airbus A350-900	4	305	Long haul international
ATR 72 -500	3	68	Short-haul international and domestic
Boeing 777-200ER	4	309	Long-haul international and domestic
Boeing 787-9	7	274	Long-haul international
Total	**82**		

In addition, the total number of VNA crew is large, as shown in Table 9, below, and the number of captains and first officers (FO) are not equal at both bases. In addition, there are two groups of crew, those who are Vietnamese (VNese) and those who are Non-Vietnamese (non-VNese). The VNese crew used to get scholarships by VNA to be trained as pilots and then typically to have signed permanent employment agreements with VNA for life, whereas the non-VNese crew are regularly hired on fixed-term contracts. Consequently, the two groups of crew members are paid by different payment methods and this makes the allocation of crew to flight pairings more difficult.

Table 9: Crew complement of Vietnam Airlines

	SGN	**HAN**	**DAD**	**TOTAL**
Captains	153	155	13	321
First Officers	144	106	7	257
TOTAL	297	261	20	578

The last matter is that the non-Vietnamese crew have different work patterns, depending on their labour agreement, as Table 10, below, illustrates. For instance, a crew member may require one week on and one week off, it means one week working and one week off, the second working pattern is the two on two off working pattern, which means 2 weeks working and 2 weeks off while another one may prefer the 3 on and 1 off pattern, which is 3 weeks working and 1 weeks off. Therefore, each roster for each individual crew member is unique and the increase in conditions leads to more constraints.

Non-Vietnamese crew are paid fix salaries and the working patterns are as below for individual crew members depending on the labour contract that he/she signs.

Table 10: Working pattern of non-Vietnamese crew

	Working weeks	**Off weeks**
Working pattern 1	1	1
Working pattern 2	2	2
Working pattern 3	3	1

IV. 3. Regularity of Timetable

VNA operates a constant schedule for every week within a session (a half year) and the flight schedule of the Airbus 321 family consists of approximately 6000 flights per month. These flight legs often fall into one of two categories, which are either short haul flights of less than three hours flying time each direct route or medium-haul flights, which have flying times from three to six hours each. About 90% of such flights are daily short-haul domestic routes and international routes to neighbouring countries in the Association of Southeast Asian Nations (ASEAN). The other category is medium-haul flight legs, normally flying to destinations in China, Taiwan, Japan, Korea, and some other Asian countries. The frequency of these flights is around two to three flights on each route per week and, generally, the carrier applies a weekly schedule.

IV. 4. Network Structure

The network structure of VNA is the incorporation of the Hub and Spoke route structure and the Point-to-Point system. Similar to North American airlines, VNA operates the Hub/Base-and-Spoke system with two hubs/ home-bases, which are Noi Bai International Airport (HAN) in the capital Hanoi and Tan Son Nhat International Airport (SGN) in Ho Chi Minh City, the biggest city in Vietnam. All routes originate from one of these two hubs, flying to other domestic or international airports, known as spokes and the hubs are also the home bases of crew.

Most of routes are operated as the hub-and-spoke architecture, whereby a flight starts from a hub and goes to a spoke airport and after having a minimum short break or sit for passengers getting off and boarding the plane, the consecutive flight departs the spoke airport to the original hub. These flights are mostly short-haul domestic and short / medium-haul international round-trip routes, which begin at the base and then finish at the original base on the same day or on the following days.

However, VNA also combines several routes having point-to-point system properties, which are a sequence of flights starting from a hub, flying to one or two other destinations before ending at the same hub or the other hub. Another characteristic is the same crew staying together through all of these flights until they come back to the home base. In addition, if the total actual flight time of all consecutive flights exceeds the flight time limit of a duty period, these flight legs are separated into two duties and the crew have an overnight rest at one of the destinations and continue to fly the other flights on the following day. The four flights round-trip routes and the six flights round-trip routes are the point-to-point route systems. Below are the details of these types of routes.

IV.4.1. The two short-haul flights round-trip routes

The two short-haul flights round-trip routes include only two consecutive short flying time flights (less than 3 hours of flying time of each direct flight) that originate from the hub/home-base. The first destination is a spoke and then from the spoke, the second destination is the original hub. All two short-haul flights round-trip routes are served by the same crew on the same aircraft and on the same day for domestic flights and some short-haul international flights; therefore, these two flights must be scheduled on the same duty period.

For instance, Figure 9, below, shows the two short-haul flights round-trip route as SGN – UIH – SGN with the two flights being VN1396 and VN1397. The flight VN1396 flies from Ho Chi Minh City (SGN) at 7:10 (GMT time) and arrives in Quy Nhon (UIH) at 8:20. Then, the return flight VN1397 departs at UIH at 9:10 and arrives to SGN at 10:15. These two flights are assigned to one crew on the same aircraft. The two short-haul round-trip flights always have consecutive flight numbers, such as VN1396 and VN1397.

However, the flight VN263 from HAN to SGN and the flight VN244 from SGN to HAN in Figure 10, below, are not the round-trip flights because the flight numbers are not consecutive and the time interval between the arrival time of the first flight and the departure time of the second flight may be quite large. It is even possible that these flights may be operated on different aircrafts; thus, these flights can be scheduled on different duty periods.

The time interval between the ETA of the first flight and the ETD of the second flight is called the connect time or break or sit. The minimum connect time of the Airbus aircraft between the domestic flights is 50 minutes and the minimum connect times are also different between international flights and domestic flights as the crew need enough time to transfer from the domestic terminal to the international terminal. In addition, different fleets of aircraft also have different minimum connect times as the bigger aircraft with more seats require more connect time to clean the passenger cabin and to receive delivery of meals from the catering services.

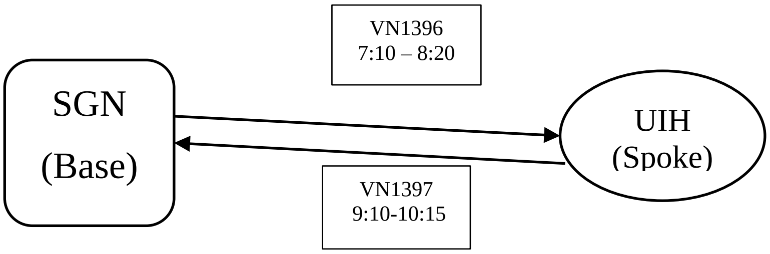


Figure 9: Two short-haul domestic flights round-trip route

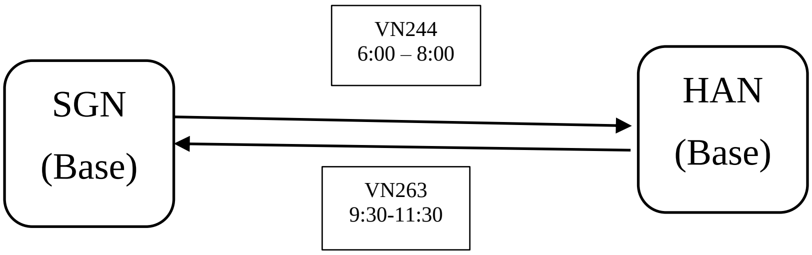


Figure 10: Two direct flights are not a round-trip route

IV.4.2. The two medium-haul international flights round-trip routes

The other type of two flights round-trip routes is the two medium-haul international flights round-trip routes. The first one departs from the home base and arrives at an international destination where, after an hour of break (connect) time, the return flight starts and flies back to the original base. The flying time of these flights is from 3 hours to 6 hours each flight.

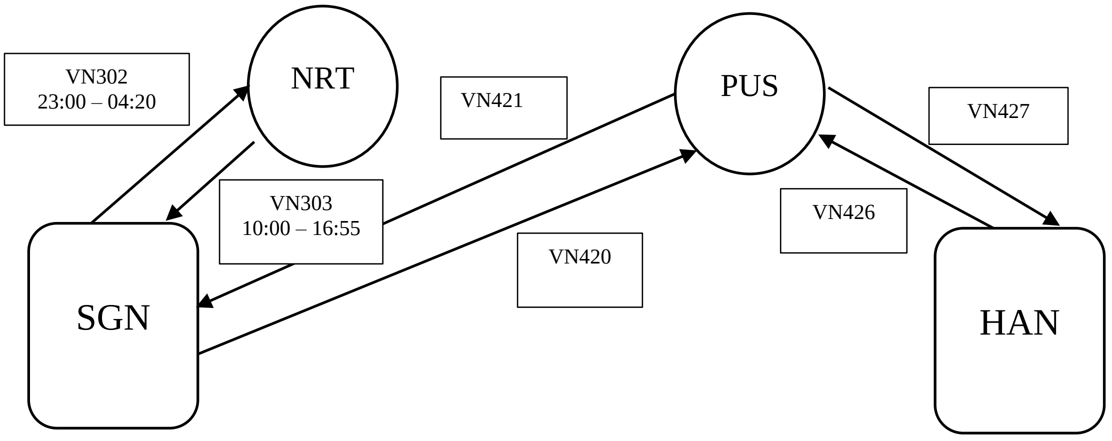
Since the total duty period time of two medium-haul international flights round-trip routes normally exceeds the duty period limitation, and the arrival time of the first (outbound or outgoing) flight is in the late afternoon or in the evening hours of local time, the crew have to stay overnight at the international destination. The following day, the crew operates the return flight back to the original home-base.

In the other case, the ETD of the return flight is about an hour immediately after the ETA of the first flight, so these two flights must be allocated a two crew complement. The first crew operates the first flight and then stays at the international destination waiting for the other outbound flight coming in the following days, whereas the second crew who had arrived one or two days ago from the previous outbound flight conducts the return (inbound or ingoing) flight back to the home base.

The time that the crew stays at places other than their home-base on duty is called time away from base (TAFB), and VNA must provide meals and accommodation for them, so this expenditure is a cost of a pairing. In addition, VNese crew also receive a per diem allowance when they are away from their home base. Therefore, the cost of a pairing in this research is basically the number of hours (minutes) in the pairing that the crew spends without flying.

Therefore, these two flights round-trip routes belong to two different duty periods (one flight on each duty period) and are assigned to either one crew on two consecutive duty periods of other days or two crew on two duty periods of the same day on the same aircraft.

For example, in Figure 11, below, the flight VN302 from SGN to the Japanese city of Narita (NRT) takes 5 hours and 20 minutes and the return flight VN303 from NRT to SGN takes 6 hours and 55 minutes. Thus, if only one crew flies both flights, the flight time limitation will be violated. In this case, two crews are assigned to these flights, whereby the first crew flies VN302 from SGN to NRT and stays at Narita, while the second crew, having flown VN302 on the previous day and stayed at NRT, now flies the return flight VN303 back to SGN. The same happens with flights VN420: SGN-PUSAN (PUS) and VN421: PUS – SGN.



IV.4.3. The four flights round-trip routes

The third round-trip routes have four flight legs with two outbound flights and two inbound flights. The first flight originates from the hub/base to the first spoke and then the second one continues from the first to the second spoke, while on the way back the third flight returns to the first spoke and the last one finishes at the original home base. These flights are a combination of the Hub-Spoke system with the point-to-point route structure of the European airlines system.

For instance, Figure 12, below, illustrates the first flight VN1324 SGN – DAD from 07:30 – 08:45, the second one VN546 DAD – Can Tho (CTU) from 10:30 – 13:25 and then the flight VN547 CTU – DAD departs at 14:25 and comes to DAD at 17:20 and the last one VN1303 DAD – SGN from 23:35 – 00:50. The second and third flights have consecutive flight numbers (VN546 & VN547); thus they are the round-trip flights but DAD is not the hub, as these round-trip flights are combined with other flights coming from the SGN hub and return to the SGN hub after that.

The total flight time and connect time of these four flights is 18 hours and 35 minutes, more than the duty period limitation. Therefore, they are separated into two duty periods and allocated to two crew complements. The first crew flies the first three flights and rests an overnight at DAD, due to the rest time rules of at least eleven hours. The last flight VN1303 is connected with other flights to form another duty by an algorithm being discussed in the following chapter, and this new duty period is assigned to the second crew to return to SGN. In this case, domestic accommodation costs at DAD are incurred.

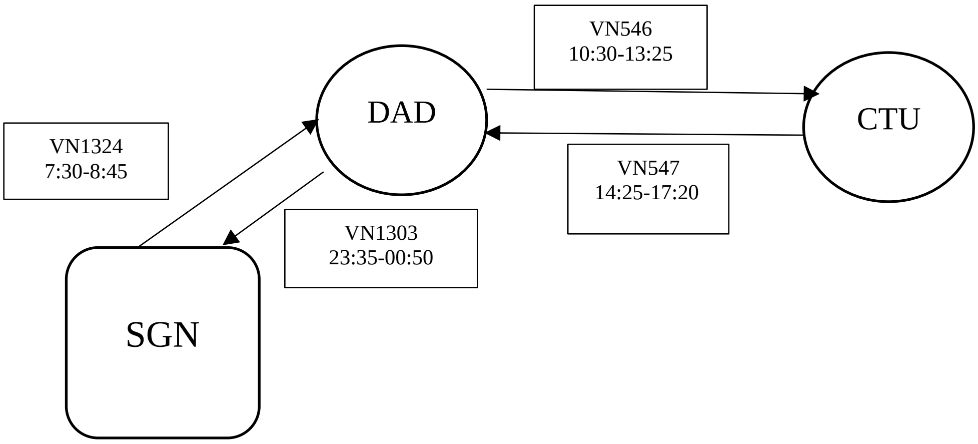


Figure 13, below, illustrates another type of the four flights round-trip routes, which are similar to the four domestic flights round-trip routes above. These flights are international four flights round-trip routes with only two flight numbers VN931 and VN930. The first flight begins from the HAN hub at 6:00 to Luang Prabang-Laos (LPQ) at 07:00 and the second segment from LPQ at 07:50 to Siem Reap-Angkor-Cambodia (REP) at 09:20. These two flights have only one flight number VN931 and are regarded as transit flights. On the way back, the second flight with the number VN930 departs from REP at 10:10 and returns to the first destination LPQ at 11:35 and then from LPQ at 12:25, flying back to HAN at 13:25. As the total flight time of these four flights is less than the 4 landings duty period limits of 11:30 hours, they are assigned to one crew only on the same duty period.

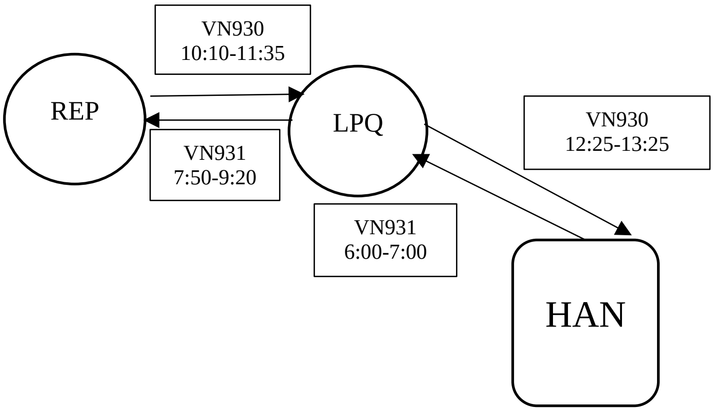
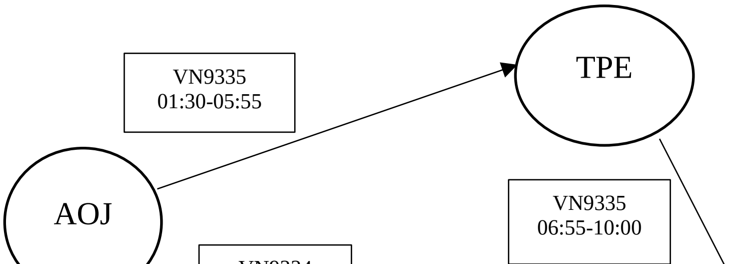


Figure 13: Four flights round-trip route through two international destinations

Figure 14, below, shows the last type of the four flights round-trip routes with the point-to-point characteristics originating from the base to three international destinations and finishing at the base. The first two flights have the same flight number and the last two flights also have the consecutive same flight number, VN9334 HAN – Niigata_ Japan (KIJ) 17:15 – 22:00, VN9334 KIJ – Aomori _ Japan (AOJ) 23:00 – 23:55, VN9335 AOJ – Taipei (TPE) 01:30 – 05:55 next day and VN9335 TPE – HAN 06:55 – 10:00.



These flights span two days and are allocated to two crews, with the first transit flights, VN9334, assigned to the first crew and this crew stays at AOJ waiting for the other flight coming later. Meanwhile, the second transit flights, VN9335, coming back to HAN are allocated to the other crew, who came and stayed at AOJ on previous days.

IV.4.4. The point-to-point routes.

Finally, the point-to-point routes are the routes transiting through several destinations before reaching the final destination. All flights on these routes have the same flight numbers and at the transit points, no new passengers are boarded, as shown in Figure 15, below. The transit flight number, VN920 starts from SGN to Phnom Penh - Cambodia (PNH), and then continues from PNH to Vientiane – Laos (VTE) and, finally, from VTE arrives to HAN. The individual flights must be on the same duty period and are assigned to one crew on the same aircraft.

The other transit flight is VN921, which originates at HAN to VTE, after that taking off from VTE to PNH and then from PNH, arriving in SGN. The flight is also flown on the same aircraft by another crew based in HAN. These routes are similar to the European airlines' structure.

There are many point-to-point flights between two hubs HAN – SGN during the day for example VN 240 and VN 274 in Figure 15. HAN and SGN become transit points between international flights and domestic flights where passengers from international flights can be offloaded and then take domestic flights to other spokes.

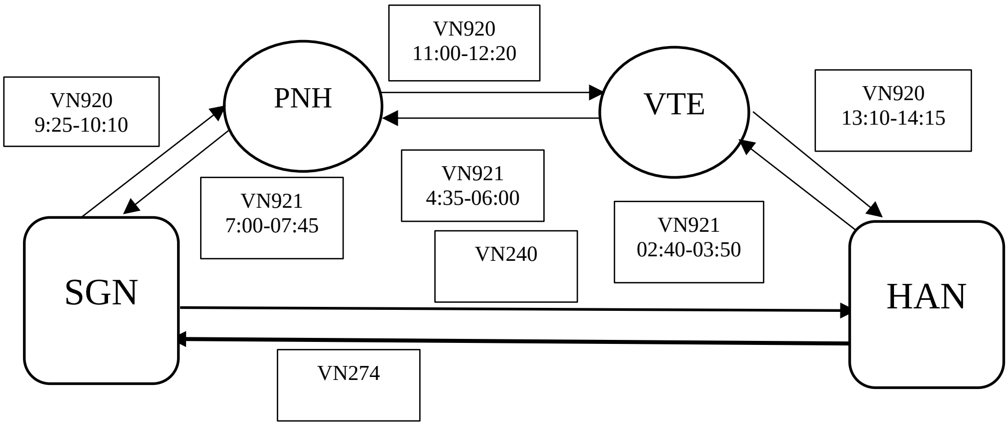


Figure 15: Point-to-point routes

The combination of flight network structures, as illustrated in Figure 16 and Table 11, below, makes the scheduling task more complicated. The proposed algorithm must consider these factors and integrate these two systems to provide an appropriate approach.

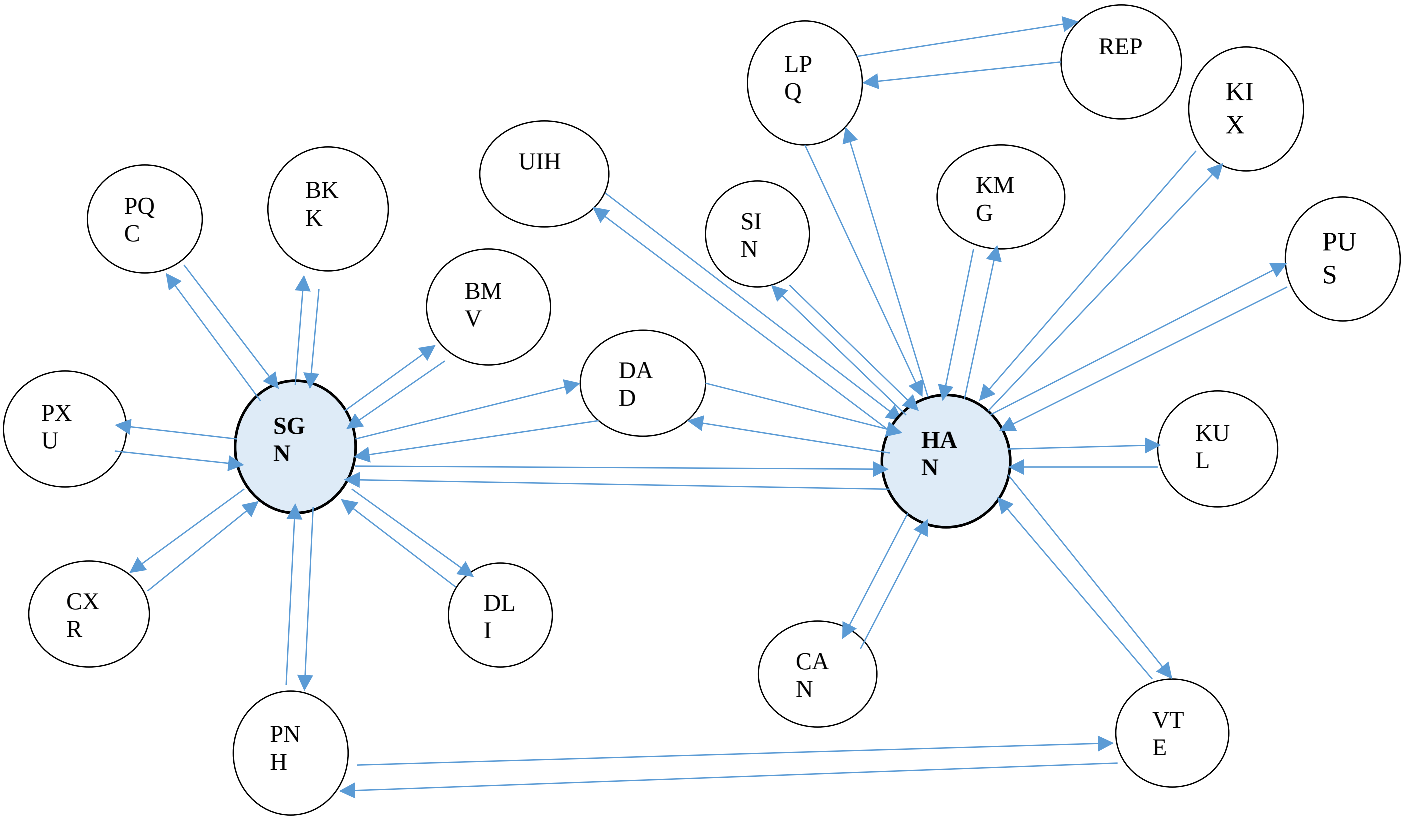


Figure 16: Hubs and spokes network of Vietnam airlines

Table *Hubs/Home bases are denoted by shaded ellipses*

Types of routes	**Hubs**	**Spokes**	**International Destination**
Two short haul flights round-trip routes	HAN	BMV; CXR; DAD; DLI; PXU; PQC;…	

	HAN		BKK; SIN; PNH; KUL;...
	SGN	BMV; CXR; DAD; DLI; PXU; PQC;…	
	SGN		BKK; SIN; PNH; KUL;…
Two medium haul international flights round-trip routes	HAN		PUS; NRT; HKG; KHH;…
	SGN		PUS; NRT; HKG; KHH;…
Four flights round-trip routes	HAN		LPQ – REP;…
	HAN		KIJ – AOJ – TPE;…
	SGN	DAD – CTU;…	
	SGN		REP – LPQ;…
Point-to-point routes	HAN		VTE – PNH; …
	SGN		PHN – VTE; …
	HAN - SGN		

IV. 5. Rules and Regulations

The scheduling process must strictly follow the current flight operations manual (FOM) (CAAV, 2015), which has been passed by the CAAV. Several rules which impact the crew scheduling process have been collected and are presented as follows:

1. The flight duty period must not exceed the times in Table 12, below, according to the current number of flight legs that have been taken.
2. The minimum rest time period, which must be provided before undertaking a flight duty period, shall be at least as long as the preceding duty period, and not less than 11 hours. In the case of a time zone difference, rest periods would be longer (more details in FOM).
3. Some other control variables:
    - Pre-flight and post-flight-time,

- Minimum ground time (minimum time between two legs that can be connected together in a pairing),
- Maximum number of flight legs in a pairing.

All pairings must begin and finish at one of the two bases and there is no strict minimum and maximum number of duties in a pairing being written, but in this study, it is fixed at a maximum of two duties in a pairing. As a result, there are one duty pairings and two duty pairings to reduce the meals and accommodation expenditure and per diem allowances for crew when they are away from the home base on duty. The pairing period has a maximum of four days, since the frequency of international flights are two or three flights per week and crew have to stay at the international destination up to two nights waiting for the following flight.

The minimum connect time of the Airbus 320 fleet is 50 minutes for cleaning and catering services as well as a short break for crew if they are on the same aircraft during the duty period and both consecutive flights are either domestic or international. Nevertheless, the connect time must be longer if the crew have to change aircrafts between two consecutive flights in a duty or two consecutive flights are a domestic and an international ones because the crew have to travel from domestic terminal to international one and go through immigration process.

Table 12: Flight Duty Period limitation

Reporting time (LT)	**Number of landings**				
	1,2	3	4	5	6 or over

07:00 – 17:59	13.00	12.00	11.30	11.00	10.00
18:00 – 21:59	12.30	12.00	11.30	11.00	10.00
22:00 – 04:59	12.00	11.30	11.00	10.00	09.00
05:00 – 06:59	12.30	12.00	11.30	10.30	09.30

Some of horizontal rules of VNA are related to the flight time of pilots, such as that the total block time of the flight on which an individual crew member is assigned as an operating crew member does not exceed:

1. 100 hours in any 28 consecutive days.
2. 270 hours in any calendar quarter.
3. 1000 hours in any calendar year.

The maximum uninterrupted block times to which a flight crew may be assigned in one flight duty period shall not exceed the following block times shown in Table 13, below.

Table 13: Maximum uninterrupted block times

Reporting time (local time)	Maximum uninterrupted block times
07:00 – 13:59	11 hours
14:00 – 17:59	10 hours
18:00 – 04:59	09 hours
05:00 – 06:59	10 hours

The cumulative duty period to which a flight crew is assigned will not exceed:

1. 60 hours on any 7 consecutive days. This figure can be increased to 63 hours when a rostered duty, consisting of a series of duty periods, has commenced and is subject to unforeseen delays.
2. 190 hours over 28 consecutive days.
3. 1800 hours over a 12 calendar month period.

A flight crew may request days off in a block of up to a maximum of 3 days during a roster period. A maximum of six requests for the day(s) off per year shall be accepted. A flight crew member has to inform the Crew Rostering Office of medical status by check in time: 4 hours before ETD and 12 hours at stopovers. Rules relating to rest time are illustrated below:

1. The minimum rest periods are increased to at least:
   - One 36 hours period within 7 consecutive days.
   - One 60 hours period within 10 consecutive days.
2. The minimum time free of all duty and standby – may include annual leave stipulated by the Labour Law of Vietnam, day-off away from main base, medical leave and other day-offs – totals 118 days per calendar year. These 118 days must be scheduled to ensure:
   - At least 7 days per calendar month (which may include required rest periods).
   - At least 24 days per calendar quarter.
3. If the flight duty time exceeds 18 hours including positioning, crew members shall have one night’s rest.

A roster is considered legal if it satisfies the horizontal rules and regulations. Vertical rules relate to crew complement, which is normally one crew, including one captain and one FO, but some tasks require two captains and one FO or one captain and two FOs with one FO on training. The qualification type constraints, which apply to crew, are special airport landing or taking off certificates, or simulation certificates. Global constraints that are applied to the entire solution are the upper bound on the costs of the solution, constraints on overall bid satisfaction and horizontal rules defined on more than the planning period (Kohl and Karisch, 2004).

IV. 6. Cost Structures

There are two groups of cockpit crews at VNA and they are paid by different methods. The Vietnamese crews are paid according to the below formulation:

$$Job\ title\ benefits + credit\ hour\ payment\ (flying\ time) + perdiem\ allowances$$

This payment method is similar to the North American airlines, whereas the second group of foreign crews are paid fixed packages, which are the same as the European payment method.

IV. 7. Summary

This chapter analyses all characteristics of crew scheduling of VNA. Understanding the special factors of the airline will help to design appropriate algorithms. First of all, VNA has many fleets of aircrafts and each fleet is scheduled separately therefore the scheduling task is quite time consumed. Second, regarding to complement of crew, two groups of Vietnamese and Non-Vietnamese crew have different labour agreements and payment methods leading to complicated scheduling. In addition, Non-Vietnamese crew has individual working pattern, therefore each crew has individual reference of scheduling. Third, numbers of captains and first officers at two hubs are not equal, therefore during high seasons crew has to be positioning between hubs. Fourth, network structure of VNA is hub and spokes, however there are several point-to-point domestic and international routes, this must be concerned in algorithms. Four types of routes are the two short-haul flights round-trip, the two medium-haul international flights round-trip, the four flight round-trip and the point-to-point routes. Fifth, most routes are short and medium haul, in order to reduce meals and accommodation cost as crew are away from home base, two duty period are maximum in a pairing. Finally, there are some fixed international destination where crew spend overnight there. In addition, there are rules and regulations regarding to minimum connect time, maximum uninterrupted block time, and others.

V. The Crew Pairing Problem of VNA

After analysing the structure of VNA flight routes and the data of maintenance routing, we observe that it is hard to effectively apply the existing methods of the crew scheduling problems to meet the objectives of VNA. In order to model the CPP of VNA accurately and design suitable algorithm, the objectives of CPP are declared firstly. Next, all constraints of the problem have to be mentioned carefully. And then a formulation is developed based on the duty period model of Vance et al (1997b). Two algorithms are implemented one after the other to solve the problem. Pseudo codes are presented along with computational experiment. Result assessment states advantage and disadvantage of the algorithm.

V. 1. Objectives of the Crew Pairing Problem

Since the crew pairing model is concerned only with ‘pay-and-credit’ costs, such as the crew-related cost and deadheading (see section II.1.3), the objectives of the VNA crew pairing problem are the removal of deadhead, reduction of meal and accommodation expenses when crew are away from their home base on duty, and minimizing the number of required crew members. The issue of deadhead or positioning is the flight that crew members travel as passengers. Deadheads are basically used to reposition a crew to an airport where they are needed to man a flight or to enable the crew to return to their home base at the end of their pairing (Vance et.al, 1997). For instance, the two medium-haul international flights round trip routes in Table 11 would require two crew (one deadhead crew at a time) to cover both outbound and inbound flights of the routes since total flight time of the routes exceeds the flight duty period limitation. As FOM of VNA states that positioning shall be included as part of the flight duty period (FDP) provided that the flight crew member is assigned to flight duty no later than 10 hours after positioning. If positioning time is less than 4 hours, 100% positioning time will be included in the calculation of FDP. If positioning time is more than 4 hours, only one half of that time shall be included. The case is of two crews being needed on two consecutive flight segments and this, in turn, becomes a serious expense for the airline, as all crews will have to be paid for both flight segments, even only one of them really operates each flight. Investigation is therefore required to identify why the issue of deadhead occurs. One major problem with deadhead happens in the case of long flight time duties of more than eleven hours, especially with a long-haul route of a 13-15 hour direct flight, where a deadhead is unavoidable, that is the second crew (had been a deadhead) will operate the aircraft when the first crew (being a deadhead) finish their duty period of 11 hours. This case study is about short and medium haul routes of the fleet of Airbus 321; therefore, the above situation will not occur. However, deadheads still happen if the duties of crews finish at any places other than their home bases (where they reside). This is because after the duties of the current crew, there are still several flights on the aircraft, and as the duty of the current crew ends at a place other than the crews' home base, there are no available crew to continue the following flights, as well as the current crew being unable to return to their home base.

To overcome this situation, a deadhead crew, including a captain and a FO, has to accompany the current crew as passengers on the flights and then operate the aircraft when the current crew finish their duties. Now, the current crew becomes a deadhead on the following flights or they have to stay over at the last destination where they finish their duties until their next day's duties, when the other aircraft comes and they swap with that crew. In this case, VNA has to pay for both the main crew and the deadhead for all the time that they started their duties as main crew, as well as their time being deadhead until all of them come back their home bases, together with their meals and accommodation and even per diem allowances for VNese crew. This becomes a hugely excessive crew-related cost.

Consequently, the first objective of deadhead removal is obtained by imposing rules or more constraints on the duty generation process. To be a legal duty, in addition to the rules and regulations in FOM[2], there are some constraints on this research. Firstly, duties must at least either start or finish at the crews' home base, as when the crew begin and finish their duties at their home base, VNA does not pay for their accommodation. However, there are international medium-haul routes, where the total flight time of both inbound and outbound flights on the routes exceed the duty period limitation and the crew have to stay over at an international destination until the following day, when they conduct the next day's duties back to their home base. Therefore, the crew have two duties conjoined, in that the first one starts at their home base and finishes at an other place and the second one starts at the same place as the first one's destination and ends at their home base. These two duties make a pairing. To sum up, there are three types of duties which were examined by this research: the first type starts and finishes at the home base, the second type begins at the home base only and the last one ends at the home base.

The second constraint, to eliminate deadhead, imposes on the second and the third types of duties. The constraint allows only a few destinations where crew are allowed to stay over, this means that duties cannot finish at any random destinations. The second type of duties

---

[2] Flight Operation Manual

finishes at an allowable destination and the third type of duties begins at the same airport of the second duty. With this constraint, the duty generation process is much more complicated and includes many assessments and adjustments.

The second objective, the reduction of meals and accommodation expenses, is also achieved by imposing the second constraint mentioned above. For the first type of duty, starting and finishing at the same home base, no meals and accommodation expenses occur. For the second type of duty, beginning at the home base and ending at a different destination, and the last type, starting at the same destination as the second type duty and finishing at the home base, there are meals and accommodation expenses, as they form a pairing. The first type of duty does not have to pair with another duty and becomes a one-duty pairing; the second type of duty is paired with the third type duty, to create a pairing. Therefore, the pairings always start and end at the home base of the crew and the maximum number of duties in one pairing is two duties, spanning from one day for the one-duty pairing to four days for the two-duty pairings. The layover destinations, other than the home base, are specific areas only. Currently, international destinations where accommodation are rented, we cannot change, others airports where VNA has its own hotel are preferred to be the layover destinations to reduce meals and accommodation cost. For instance, the international medium-haul route of HAN – KIX – HAN starts from 17:30 to 21:40 GMT the previous day for the outbound flight and from 01:30 – 07:15 GMT the following day for the inbound flight, thus the total flight time and the connect time of this route of two flights are 13^h 45’, which is over the limitation. This route is therefore separated into two duties for two crews; the first crew flies the HAN – KIX flight and stays over at KIX for a maximum of 2 days, while the other crew has stayed over at KIX from the previous similar flight and are going to operate the returning KIX – HAN flight. The first crew stays for the following two days and come back to their home base on the next incoming flight. The crew’s total of four days comprises day 1, flying the HAN – KIX flight, days 2 & 3, rest, and day 4, flying the KIX – HAN flight back home.

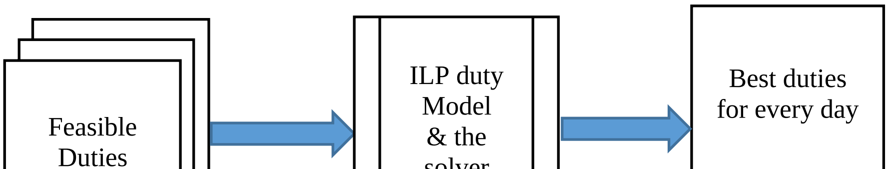
The last objective is to minimize the number of required crew by minimising the number of pairings or duties, which would then lead to fewer crew members being required. This would remove the need to hire new pilots and the expenses of overtime remuneration. An important point to consider is that a duty requires one hour pre-flight and 15 minutes post flight report time; these times are regarded as idle or waste time in term of cost effectiveness, and in this study, as the aim is to minimise the number of duties, then subsequent report times can be reduced also. In order to reduce the number of duties, small duties must be combined to become a longer flight time duty period, and the connect time between two consecutive flights in a duty should be limited.

We propose a new heuristic and exact combinational algorithm to solve the VNA crew pairing problem, specific to the Airbus 321 fleet at VNA, and then to implement the first one to obtain a better result with the second algorithm. The proposed methods apply a generate-optimize structure (see chapter III), which in the generation stage, uses heuristics to create duties and pairings on a day by day basis, and then, in the optimization stage, all duties or pairings for that day are input into the Set Partitioning model, using the solver to select the best subset of duties or pairings, respectively, for each day. Details of each method are presented below.

V. 2. The daily duty-period based approach

V.2.1. Proposed Algorithm

The daily optimal duty-period based method applies the generate-optimize structure, as shown in Figure 17, below. In the first stage, a heuristic is used to generate duty periods for each day, and then these duty periods are input into the duty-period formulation for the solver to select the subsets of daily best duties. The purpose of this step is to reduce the number of not-good duty period. Because any good quality duty periods which start and finish at the same home base with short connect time do not need to pair with any other duty. They will be chosen in the final set of best duty periods, and become one-duty pairings or one day pairings. Therefore, using the solver to discard poor quality duty periods from each daily set of duties will reduce processing time later while the result will not change. After that, the remainder of these daily sets of duty periods, starting and ending at the different places, are combined with one of the other best duties of the following days which have started at the same place with the duties of the previous day into pairings. . Finally, all the generated pairings of the whole schedule are input into the set partitioning model, using the solver to select the optimal pairings for the schedule.



The main contribution of the proposed method is to decompose the main problem with a large data set into many phases and solve in daily sequence, which enables the original problem to be solved quicker, with the result meeting all the desired objectives. Focusing on high efficiency, in terms of computational time and the reduction of excess costs regarding deadheads or positioning and crew related costs, such as meals and accommodation costs and per diem allowance when they are away from their home base, the heuristic fulfils all conditions of the CAAV rules and regulations and applies more constraints to discard poor quality duty periods with long connection times, or short total flight times as well as random finishing destinations. Only good quality feasible duties and then pairings are generated and input into the models; therefore, the final solution chosen by the solver is the best subset of legal feasible pairings, with large total flight times and short connection times.

The other advantage of this approach is that it can accommodate four types of flight routes (see chapter IV) in an integrated model, with the final objective of this research being to create an integrated approach, able to solve these together, so that no type of duty needs to be treated separately. Since the route structure of VNA is a combination of the hub and spoke system and the point-to-point structure, each type of route is carefully investigated and flight legs in each route are connected if possible to become a node of flights in the flight network. This process not only reduces the number of nodes in the flight network, but decreases the number of randomly finished duties.

V.2.2. Problem Formulation

The aim of CPP is to find a minimum-cost set of pairings, so that each scheduled flight is assigned to only one feasible pairing, and the method has two stages, which occur in sequence. In the first stage, the proposed heuristic generates a set of good quality feasible duties for each day. The feasible duties must satisfy all conditions, which the CAAV defines in the current Flight Operation Manual with regard to total flight time limitations of a duty, connection times between flights and the number of flights in a duty. In addition, we have imposed some more constraints on the duty generating process, in order to obtain a good quality solution.

The first constraint enforces that any duty either starts or finishes at one of the two Bases (HAN and SGN) or, much better, the duty should begin and end at the home base of the crew; thus, the airline can reduce excessive costs for crew meals and lodging after their duty periods. This is because, if a duty finishes at any airport randomly, VNA has to pay for the crew's accommodation away from their home base, or cover their travel expenses back to their home base, and it is necessary of other crew to take over the aircraft for the following duty. If a duty finishes at a spoke, there is no crew residing there to complete the following duty, since crew only reside at one of the two bases. As a result, in order to assign the next crew to the subsequent duty on the aircraft landing at the spoke, a deadhead must travel on the first duty or flight, as passengers to carry on the second duty. The other soft constraint, to eliminate short-time duties, which have a total flight time of less than the 3 hour[3] minimum duty guarantee is applied to connect these duties together.

After generating good quality feasible duties, in the second stage all of these duties are input into the modified duty-period ILP formulation to choose the best duties for each day in terms of total connection time and the total number of duty periods. There are a couple of reasons for applying the modified duty-period formulation after the duty generation process: firstly, the number of one-duty pairings is more than the number of two-duty pairings, and the one-duty pairings occurring each day are independent from the ones on the following day. Hence, the selection of these one-duty (one day) pairings occurring before or after the pairing generation process does not change the final solution, with only the second and the third type of duty periods being affected by this method (this is implemented in the second method). Secondly, the process of choosing the best duties day-by-day reduces the computational time,

---

[3] A Minimum Duty Guarantee (MDG), or a minimum guaranteed number of hours being paid for the crew on a duty is 3 hours.

as decomposing the big data set into the subset of daily duties allows the solver to run quicker and the solution is not much different.

The notation that will be used here for describing the CPP is as follows:

F: Set of flights to be covered in a month

F_i: Subset of flights to be covered on day i , $F_i \subset F$

f_{ij}: a flight to be occurred on day i , indexed by j, flight $f_{ij} \in F_i$

D: Set of all duties

D_i: Subset of all duties in day $i, D_i \subset D$

d_{ik}: a duty in day i, indexed by k, duty $d_{ik} \in D_i$

$c_{d_{ik}}$: Total connect time of a duty d_{ik}

$w = 1\ 000\ 000$ a large coefficient for the minimization problem

$x_{d_{ik}}$: Binary decision variable that takes a value of 1 if a duty d_{ik} is selected,

and 0 otherwise, $d_{ik} \in D_i$

The modified duty-period-based formulation for day i is :

$$\min \sum_{d_{ik} \in D_i} (c_{d_{ik}} + w) * x_{d_{ik}} \qquad (1)$$

Subject to

$$\sum_{f_{ij} \in d_{ik} \in D_i} x_{d_{ik}} = 1 \qquad \forall f_{ij} \in F_i \qquad (2)$$

$$x_{d_{ik}} \in \{0,1\}, \qquad \forall d_{ik} \in D_i \qquad (3)$$

The objective (1) is to find the minimum number of best duties with short connection times every day. The number of daily duties requires a similar number of pilots to conduct them every day; thus, a smaller number of duties needs a smaller number of crews and *w* (a large coefficient) effects the IPL model and the solver in choosing smaller amounts of large duties (large connection times as well), rather than many small duties with short connection times.

Another reason for choosing a fewer number of duties has been mentioned above: any duty has 1 hour of pre-flight reporting (briefing) and 15 minutes of post-flight debriefing (a period of time at the end of each duty that gives the crew-member an understanding of the events that occurred and their implications), which is regarded as idle time (or waste time) in terms of cost effectiveness so, as the number of duties decreases, the briefing (a period of time before the start of each duty that is spent on instructions and reports for any duty) and debriefing time also decreases. In addition, the connect time included in the duty period is also paid for as the flight time and is wasted time for VNA; therefore, it should be reduced in order to increase the number of flight legs in each duty.

The constraint (2) requires that each flight must belong to only 1 duty to avoid duplicated flights being allocated to two crews. The constraint (3) is the binary values of variables.

After choosing the set of best duties for every day, any duties that start and finish at two different places are combined with one of the other duties of the following day to create two-duty pairings, while duties beginning and finishing at the same home base do not need to pair with any other duties and become one-duty pairings. The main purpose of duties being paired is to allow crew to get back to their home base, and the rest time between two paired duties is time off duty for the crew, which adheres strictly to the rest time requirement of the FOM. Any expenses for meals and accommodation, as well as per diem allowance for the crew, during the rest time are excess costs of the pairing (Vance *et al.*, 1997b).

The pairing generation step starts with a rest time calculation method. As the frequency of international flights is approximately two or three flights per week for each route, the rest time is set to a maximum of two days and the total duration of a pairing is limited to four days. The minimum rest time is 11 hours, or at least equal to the first duty period[4]. Similar to the duty generation step, the number of generated feasible pairings must be large, and then the SPP formulation is applied to choose the set of optimal pairings.

The formulation of the CPP as a SPP :

$\bar{D}_{\imath}$: A subset of the chose duty periods of day i from the set of all duties on day i $\bar{D}_{\imath} \subset D_i$

$\overline{d_{\imath k}}$: The chose duty period of day $i,$ indexed $k,$ a duty $\overline{d_{\imath k}} \in \bar{D}_{\imath}$

P: Set of all pairings are generated from subsets of the chose duty periods

$p_{\bar{d}_{ik}}$: A pairing is created by duties $\overline{d_{\imath k}}$, $p_{\bar{d}_{ik}} \in P$

$y_{p_{\overline{d_{\imath k}}}}$: Binary decision variable having the value of 1 if a pair $p_{\overline{d_{\imath k}}}$ containing a duty $\overline{d_{\imath k}}$

is selected, and 0 otherwise, $p_{\overline{d_{\imath k}}} \in P$

---

[4] FOM: 7.3.9 (a) The minimum rest period which must be provided before undertaking a flight duty period shall be at least as long as the preceding duty period and not less than 11 hours.

$r_{p_{\overline{d_{\imath k}}}} =$ RestTime if the pairing has the night rest away from original base

The objective of CPP is:

$$\min \sum_{p_{\overline{d_{\imath k}}} \in P} r_{p_{\overline{d_{\imath k}}}}\, y_{p_{\overline{d_{\imath k}}}} \tag{4}$$

Subject to

$$\sum_{p_{\overline{d_{\imath k}}} \in P} y_{p_{\overline{d_{\imath k}}}} = 1 \qquad \forall\, \overline{d_{\imath k}} \in \overline{D_{\imath}} \tag{5}$$

$$y_{p_{\overline{d_{\imath k}}}} = \{0, 1\} \tag{6}$$

The objective (4) is to choose the minimum rest-time pairings, with a rest time being the period from the 16th minutes after ETA of the last arrival of the first duty to the 60th minute before the ETD of the first departure of the second duty. The one-duty pairing is set a rest time value of 1, as it has exactly no rest time. The reason we choose rest time as the cost parameter of the formulation is that expenses for crew meals and accommodation were not available and these expenses are different at different destinations. In addition, the per diem allowance for VNese crew is also based on the rest time.

In addition, Barnhart *et al.*,(2003) also indicate that the second component of the pairing cost is the extra cost associated with the rest time of the pairing, when the crew is away from base and the carrier has to pay for the crew's meals and lodging as well as their time away from base. Consequently, the longer the rest time, the more expenses are incurred for VNA. For this reason, when generating duties, we attempted to combine flights that bring the crew back to the home base as soon as possible, in order to save on such expenses. The other reason for not putting a real value on the pairings in the formulation is that VNA has two types of crew, so these anonymous pairings (pairing without crew) cannot carry any cost at this stage.

The constraint (5) requires that every duty on each day must be covered by one pairing only, either a one-duty pairing or a two-duty pairing.

V.2.3. Solution method

The method is based on LEAN thinking and dynamic programming, furthermore, Vance (1997b) stated that "Rather than considering crew scheduling as choosing pairings to partition the scheduled flights… we break the decision process into two stages. First, we select a set of duty periods that partitions the flight segments and then we select a set of pairings that partitions theses duty periods. By looking at the problem in this manner, we motivate a new decomposition scheme that chooses good pairing solutions by first identifying goods sets of duty period to use as the building blocks of the pairings." (p190-191) so that only good quality legal feasible duties and pairings are generated; thus, the question is how good a quality are the feasible duty and pairings? There are several rules and conditions imposed on the processing of the duty generation to obtain good quality feasible duties and pairings based on the analyse of VNA flights data, which are detailed below:

1. First of all, a two short-haul flights round-trip route is a node in the flight network, whereby all flights of the route cannot be separated, as the connect time between two flights is already set to a minimum of 50 minutes and a crew carries out these flights together, since crew are based at the HAN and SGN bases only. Therefore, crews cannot finish their duties in any random places, because if the crew end their duties or a crew changes at the spoke, there are no available crew members residing at the spoke to take over the subsequent flight. This is conducted in step 1 of Figure 18.
2. Also in step 1, all point-to-point flights on the same aircraft is also a node, which must be connected together. Only one crew is necessary to operate all these flights, since there is also no crew residing at international destinations to take over the following flights if the first crew end their duties at any of those destinations.
3. There are a few fixed destinations where accommodation contracts are available for crew to stay over during the pairing period; therefore duties cannot finish at any random destinations. The destinations are the home bases, international destinations of the international medium haul routes, and Danang (DAD), the secondary hub.
4. In order to reduce connect time, in Step 2, only consecutive flights with connect times from 50 minutes (minimum connect time of the aircraft A320 family fleets) to less than 3 hours, are combined together.
5. A heuristic in Step 2 is applied in order to increase the working time of each duty period to the limitation of the flight time per day of 11 hours, the number of flights in a duty has to rise up to 5 or 6 flights per duty, and the number of small duties has to be reduced by joining small duties together to make bigger duties.
6. After using a solver to choose the best duties for each day in Step 3, in order to decrease the expense of accommodations and meals for crew when they are away from base on duty, the last destination of a duty should preferably be the home base, as the expenses are not incurred if the crew finish their duties at their home base. As a result, a good quality duty has its origin and destination in the same home base.

However, not all duties can finish at the original home base, such as some international flights with long flying times, so crew have to rest at the international destination until the following days. A heuristic in Step 4 combines duties of different days into pairings.

7. Finally, as the routes of VNA are almost direct and use the hub and spokes structure as their main one, the pairings have maximum of two duties and span up to 4 days.

Based on these conditions, the proposed method fulfils all conditions to obtain the objectives mentioned above. The steps to generate duties and pairings are shown in Figure 18, below.

The solution presented here is specific to the Airbus 321 fleet, which is the main fleet of VNA, which has 56 aircraft in operation. The given data (see Appendix) is the group of flights scheduled for each aircraft, which, in general, are from 6 to 7 flights per aircraft. Departure times, flight times, arrival times of all flights and connect times between the flights of each aircraft are fixed. With this information, the minimum number of duties to cover all flights every day are calculated by the following formulation:

$$Number\ of\ duties = \frac{\sum flight\ time\ of\ all\ flights\ \ + \sum ConnectTime}{Maximum\ blocktime\ per\ day - report\ time} \tag{7}$$

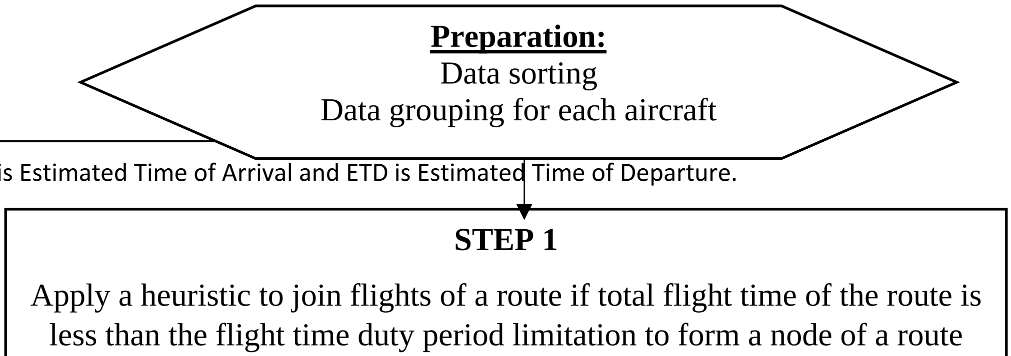
Flight Time of all flights is fixed and totalled from each flight in the flight schedule

Connect Time is the interval between the ETD[5] and the ETA of two consecutive flights of the same aircraft in the flight schedule.

Maximum block time per day is also fluctuated depending on the number of flights in a duty period and the time window of the reporting time (see Table 11).

Report time is 1 hour before the ETD of the first flight and 15’ after the ETA of the last flight of a duty.

The number of duties from the formulation (7) is regarded as the lower bound. However, this number fluctuates as the connect time is changes, depending on the algorithm. The combination of flights is enumerated with connect time as an arc or an edge and a route is a node with weight being the total flight time of a node.



---

[5] ETA is Estimated Time of Arrival and ETD is Estimated Time of Departure.

In the preparation stage, after analysing the groups of flights on each aircraft, we do not generate duties and pairings directly from a single flight, but firstly combine flights of the round-trip routes together to form a node for the route. In the first step, each flight of the two short-haul flights round-trip routes are connected to form a route node. A similar procedure is applied to the four short-haul flights round-trip routes and three point-to-point routes. This procedure prevents the separation of round-trip routes, since if occurred it would lead to a requirement for crew to change at other places than the homebase, which is unnecessary.

In addition, the combination of short-haul round-trip routes, as well as point-to-point routes, avoids deadheading and accommodation costs, because a crew of these routes always starts and finishes at their home base, as explained above.

Step 1 is conducted by procedure 1, which combines the single flights of a two-flight round-trip route; other point-to-point routes are also combined in this procedure. In addition, if all flights on one aircraft have a total flight time of less than 11 hours of a duty period (it is rarely incurred), these flights are kept as one duty.

Notation of crew pairings algorithm

i : a day in the whole schedule period.

A_i: the set of operating aircraft on day i, indexed $a_{i\alpha} \in A_i$

F_i: the set of all flights on day i

$f_{ij\alpha}$: a flight j is operated on an aircraft α, on day i, $f_{ij\alpha} \in F_i$

D : all feasible duties being generated

D_i: the set of duty being generated for day i, indexed $d_{ik} \in D_i \subset D$

N_i: the set of nodes on day i, indexed $n_{i\beta} \in N_i$

C_i: the set of connection time of nodes on day i.

$\boldsymbol{Procedure}$ 1: Create nodes of routes by combining flights of each round-trip route or point-to-point route together.

```
1. Input: a set of all flights $F_i$ being grouped in the each aircraft $a_{i\alpha}$
2. $\boldsymbol{D_i} = \emptyset$
3. $N_i = \emptyset$
4. $\boldsymbol{For}\ aircraft\ a_{i\alpha} = 1\ to\ m$
5. $\boldsymbol{Condition1}$:    $\boldsymbol{If}\ all\ flights\ of\ an\ aircraft\ a_{i\alpha}\ fulfill\ the\ constraints\ of$
          $the\ total\ flight\ time\ in\ a\ duty\ and\ connect\ time\ condition$
              $\boldsymbol{add}\ the\ all\ flights\ into\ a\ new\ duty\ in\ D_i$
       $\boldsymbol{Else}$
         $\boldsymbol{For}\ (flight\ f_{ij\alpha} = 1\ to\ o\ of\ the\ aircraft\ a_{i\alpha})$
6.
$\boldsymbol{Condition2}$:   $\boldsymbol{If}$ the flight $f_{ij\alpha}$ and the next flight $f_{i(j+1)\alpha}$ are the two return flights
```

```
                    $\textbf{combine}\ these\ two\ flights\ into\ a\ node\ of\ round\ trip\ route$
                        $and\ add\ in\ N_i$
                  $\textbf{Else if}\ \ these\ fights\ f_{ij\alpha} are\ in\ point - to - point\ route$
                      $\textbf{combine}\ these\ flights\ into\ a\ node\ of\ point - to - point\ route\ and\ add\ in\ N_i$
                  $\textbf{Else}\ \ flight\ f_{ij\alpha}\ is\ a\ single\ node\ and\ add\ in\ N_i$
                $End\ If$
            $End\ For$
        $End\ If$
    $End\ For$
$\textbf{Output}: a\ set\ of\ nodes\ of\ routes\ N_i$
```

As a result, the number of nodes of routes is much smaller than the number of all flights, since a route node now may contain one flight of a route or many flights of one route. The next step (Step 2) is to calculate the connect time between any two consecutive route nodes. There are several conditions to combine two nodes together, whereby the first node must occur before the second node, the destination of the first node must be the origin of the second node, and the last ETA of the first node must be smaller the first ETD of the second node, plus connection time. When calculating the connection time, the minimum and maximum connection time condition must be satisfied before combining nodes into a new duty.

Procedure 2 in Step 2 connects two nodes together based on the result of the connect time calculation. It also examines the total flight time of a new duty, with the maximum block time rules and regulations, before adding the new duty into the duty list D_i. If some small flight time nodes cannot be grouped with any other nodes, they are saved as independent duties in D_i. During this step, the flight time limitation and the connection time constraints are always addressed before generating any new duties, as show in Procedure 2, below.

```
$\textbf{Procedure 2}$:  Combine nodes of routes into Duties

$\textbf{1. Input}: a\ set\ of\ nodes\ N_i\ from\ procedure\ 1\ and\ a\ set\ of\ connection\ time\ C_i$
```

```
2. For node $n_{i\beta} = 1$ to $q$ in the set of nodes $N_i$
3.         Check all conditions of the combination of nodes
               Total flight time, connection time, destination and origin
4.         Combine nodes of routes into a duty and add in $D_i$
End For
Output: a set of feasible duties $D_i$ for a day
```

After conducting Procedure 2, the number of duties is larger than the number of nodes, even when we combine small nodes of routes into a duty. Because with each node we generate a couple of different combinations with several other nodes, this method provides more options at the duty selection stage and the solution is better overall. For example, node 1 can combine with one of nodes 3, 4, and 5, respectively. The combination of nodes 1 & 4 has the smallest connection time of 1 hour, but the combination of nodes 3 & 5 creates a large connection time of 4 hours, leading to overall connection time of 5 hours. Whereas, if node 1 connects with node 3, it has a 2 hour connection time and if node 4 combines with node 5, it has a 2 hour connection time, so the overall connection time is 4 hours, which is 1 hour less than the first selection.

In addition, some combinations are good in terms of the connection time aspect, but the total flight time of the duties is not always effective, and the crew has to change aircraft more than twice on the same duty, which is inconvenient for the crew. Thus, we allow different combinations of each node in the set of duties D_i to enable more options to be compared. After this step, each day *i* has a set of feasible duties D_i.

The Step 3 of the duty generation process is to search for a best duties set $\overline{D_i}$ for each day from the set of feasible dutiesD_i, which have been generated above. The formulation (1) is to allow the solver to choose the minimum number of duties to cover all flights every day.

After selecting the minimum number of the best duties for every day over the whole scheduled period, any duties which start and finish at the same home base are ready for assignment to crew, while other duties starting at the home base and finishing at any other place must be combined with one of the following days' duties which begin at the same place as the former duties and end at the home base; this step is called pairing generation. The purpose of pairing generation is to bring crew back to their original base, since a crew starts the first duty at their home base but may finish at an other place, and the process of pairing the first duty with the second one allows the crew to fly the second duty back to the home base.

Another advantage of duties pairing is to reduce the number of crew assigned every day, because the crew have been allocated to pairings of two duties, which span a few days. In other words, the number of unassigned duties for the following days is reduced.

Step 4 begins with Procedure 3 to calculate the rest time between any two duties on different days; in this step we limit the combination to two duties only (as explained earlier). Since a duty can combine with many duties of the following days and the number of constraints that must be checked before pairing duties is quite a lot, procedure 3 is completed first and separates with the pairing procedure. In calculation of the rest time, the maximum rest time is set to two days or 48 hours, in order to eliminate poor quality pairings and to limit the number of duties combinations as well. As only the best duties of every day are combined, the number of pairings is not too large for the solver to handle.

While calculating the rest time, all conditions regarding time, space, and maximum and minimum rest time must be satisfied. Rest time between two duties plays a vital role in the pairing. The cost c_p of a pairing *p* is formally presented below:

$$c_p = \max\left\{\sum_{d\in P} b_d \, , f_p \, .TAFB, ndp.mg,\right\} + \sum_{\substack{\hat{d} \in p, \bar{d} \in p \\ \hat{d} \to \bar{d}}} e\left(\hat{d}, \bar{d}\right) \qquad (8)\text{(Barnhart \textit{et al.}, 2003)}$$

The first quantity of the first component is $\sum_{d\in P} b_d$, the sum of the costs of the duties contained in the pairing. The second quantity $f_p * TAFB$ is the product of the total elapsed time or Time Away From Base of the pairing and the constant f_p. Finally, the third quantity is a minimum guaranteed number of minutes per pairing, which is the product of the number of duty periods (*ndp*) in the pairing and a fixed minimum guaranteed (*mg*) number of minutes per duty period.

The second component $e\left(\hat{d}, \bar{d}\right)$ is the extra cost associated with the rest between two duty periods, $\hat{d}$ and $\bar{d}$ representing the duty periods in the pairing *p;* $\hat{d} \to \bar{d}$ indicates that the duty period $\hat{d}$ occurs immediately before $\bar{d}$ in the pairing *p*.

As the cost formulation shows, both the second quantity of the first component and the second component are related to the rest time of a crew during the period of a pairing and the most expensive elements are those of meals and accommodation for crew and per diem allowances when they are away from the base. For this reason, the rest time between two duties of a pairing is the main factor for selection and in order to reduce the number of poor quality pairings, the maximum rest time between two duties is set as two days or 48 hours.

$\boldsymbol{Procedure}$ **3: Rest time calculation**

```
$\boldsymbol{Input}$: $all\ sets\ of\ the\ best\ Duties\ \overline{D_l}\ of\ every\ day\ selected\ from\ Step\ 3$
        $are\ added\ together\ into\ a\ single\ large\ set\ \overline{D}$
$RestTime\ set = \emptyset$
$\boldsymbol{Loop\_1}\ (duty\ \ d_k = 1\ to\ z\ \ in\ \overline{D})$
  $\boldsymbol{If\_1}\ \ (first\ departure\ of\ d_k\ is\ different\ last\ arrival\ of\ d_k)$
      $\boldsymbol{Loop\_2}\ (duty\ d_l = 1\ to\ z\ in\ \overline{D})$
        [6] $\boldsymbol{If\_2}\ \begin{pmatrix} \big((last\ ETA\ of\ d_k + 11\ hours) \leq\ first\ ETD\ of d_l\big) \\ and\ (first\ ETD\ of d_l \leq (last\ ETA\ of\ d_k + 48\ hours) \end{pmatrix}$
          $\boldsymbol{If\_3}\ ((first\ departure\ of\ d_k == last\ arrival\ of\ d_l)$
          $\&\&\ (\ last\ arrival\ of\ d_k == first\ departure\ of\ d_l))$
            $\boldsymbol{calculate}\ the\ rest\ time =$
                $first\ ETD\ of\ duty\ d_l - last\ ETA\ of\ duty\ d_k$
            $\boldsymbol{add}\ this\ pairing\ with\ rest\ time\ into\ RestTime\ set$
          $End\ If\_3$
        $End\ If\_2$
      $End\ loop\_2$
  $End\ If\_1$
$End\ Loop\_1$
$\boldsymbol{Output}$: $the\ RestTime\ set$
```

Procedure 3 begins by adding all of the best duties sets of each day into a big best duties set $\overline{D}$. *Loop_1* of Procedure 3 picks a duty from the set $\overline{D}$ to pair with another duty, the condition

---

[6] ETA is Estimated Time of Arrival, ETD is Estimated Time of Departure, the minimum rest time is 11 hours

If_1 checks whether the duty is needed to pair with another one or not. If the origin (first departure) of the duty is different from its last arrival, it needs to pair with another duty to allow a crew of two pilots to come back to their home-base.

The second loop (*Loop_2*) is to choose the second duty for combining with the first duty (chosen from *Loop_1*). There are several logical constraints to be satisfied, of which the first one (*If_2*) is with regards to logical time and the second one (*If_3*) ensures the logical places. The condition *If_2* states that the last arrival time of the first duty must be smaller (earlier) than the first departure time of the second duty plus 11 hours of compulsory rest time but not over 48 hours rest time (maximum rest time of two days). The condition *If_3* checks that the origin of the first duty must be the last destination of the second duty, and the last destination of the first duty must be the first departure of the second duty.

Procedure 4 generates pairings based on the result of the RestTime calculation. There are numbers of different duties combinations for increased selection at the later stage.

***Procedure* 4**: **Pairing Generation**

```
$\textbf{\textit{Input}}: the\ set\ of\ all\ Duties\ \overline{D}\ selected\ from\ the\ step\ 3\ and$
          $the\ set\ of\ RestTime\ from\ procedure\ 3$
$PairingMap\ =\ \emptyset$
$\textbf{\textit{Loop\_1}}\ duty\ \overline{d_k} = 1\ to\ z\ \ in\ \overline{D}$
    $\textbf{\textit{If\_1}}\ (the\ first\ departure\ of\ \overline{d_k}\ is\ same\ the\ last\ arrival\ of\ \overline{d_k})$
        $\textbf{\textit{Add}}\ duty\ \overline{d_k}\ as\ a\ new\ one-duty\ pairing\ in\ the\ PairingMap$
    $\textbf{\textit{else}}$
        $\textbf{\textit{Find}}\ a\ set\ of\ Neighbour\_Duties\ of\ Duty\ \overline{d_k}\ in\ the\ RestTime\ set$
        $\textbf{\textit{If\_2}}\ \big(Neighbor\_\ Duty\ set\ of\ Duty\ \overline{d_k}\ is\ not\ empty\big)$ ;
        // $Neighbour\_Duty\ set\ of\ Duty\ \overline{d_k}\ has\ \ l\ \ duties$
            $\textbf{\textit{Loop\_2}}\ Neighbour\_Duty\ \overline{d_j} = 1\ to\ l\ in\ Neighbour\_Duty\ set\ of\ Duty\ \overline{d_k}$
                $\textbf{\textit{If\_3}}\ (constraint\ of\ RestTime\ of\ a\ proposed\ pairing\ is\ true)$
                 && $(constraint\ of\ total\ flight\ time\ of\ a\ proposed\ pairing\ is\ true)$
                    $\textbf{\textit{create}}\ a\ pairing\ for\ these\ two\ duties\ in\ PairingMap$
```

```
              $End\ If\_3$
          $End\ Loop\_2$
        $End\ If\_2$
      $End\ If\_1$
  $End\ Loop\_1$
$\boldsymbol{Output}: the\ feasible\ PairingMap$
```

The pairing generation process in Procedure 4 starts by picking a duty $\overline{d_k}$ from the best Duty set $\overline{D}$ in *Loop_1* and checking whether the duty is needed to pair or not. If it is not, it is added into the *PairingMap*, otherwise, its neighbouring duties are found in the *RestTime* set and a *Neighbour_Duty* set is created for the duty. *Loop_2* runs through the *Neighbour_Duty* set and selects one duty at a time to combine with $\overline{d_k}$. The condition *If_3* checks all rules and regulations regarding the total flight time and the pairing time of a pairing before adding a pairing of two duties into the *PairingMap*.

The final step (Step 5) is for the solver to solve the SPP formulation by choosing the minimum number of the best pairings from all the feasible pairings generated in Procedure 4.

The formulation of the CPP as a SPP :

$$\min \sum_{p_{\overline{d_{ik}}} \in P} r_{p_{\overline{d_{ik}}}}\, y_{p_{\overline{d_{ik}}}} \tag{4}$$

Subject to

$$\sum_{p_{\overline{d_{ik}}} \in P} y_{p_{\overline{d_{ik}}}} = 1 \qquad \forall\, \overline{d_{ik}} \in \overline{D_i} \subset \overline{D} \tag{5}$$

$$y_{p_{\overline{d_{ik}}}} = \{0, 1\} \tag{6}$$

When running the model (4), overlapping duties (where one duty is in many pairings) may happen sometimes and the solver cannot find the optimal solution. To overcome this matter, as many as possible feasible pairings must be created.

V.2.4. Computational Experiments

The program was coded by Java in NetBeans IDE 8.0.2, with Gurobi 7.0.2 solver. The hardware was a Processor Intel(R) Core(TM) i5-2400 CPU @ 3.10GHz, 3101 Mhz, 4 Core(s), 4 Logical Processor(s). The test data was taken from Vietnam Airlines and it covered timetables for the first six months of 2014, running the algorithms in Java using Gurobi 7.0.2 solver.

The number of flights in January 2014 was 6949 flights, with around 45 aircraft operating. The average number of flights each day was about 200 flights, but as the last week of January 2014 was the Tet Lunar New Year period, the number of flights rose to more than 300 flights each day.

The number of flights fluctuates daily, depending on whether it is high season or low season, a weekday or the weekend, while the number of aircraft operating every day relies on the maintenance schedule of each aircraft. Normally, January is one of the busiest months of the year, due to the New Year and Tet Lunar New Year month as well; therefore, January's schedule was taken to test the algorithm, with the aim of examining all the scenarios that may occur. As mentioned before, some duties of the first few days of the month and the last few days of the month were paired with duties of the previous month or the following month, respectively, so in the current month's schedule, they are broken pairings, which have only one duty in two duties pairings.

The minimum number of duties is calculated by the below formulation:

$ft : fixed\ flight\ time\ in\ the\ flight\ schedule$

$Mt_d : Maximum\ time\ in\ a\ duty\ period$

$ct: connect\ time\ between\ two\ consecutive\ flights$

$bf: preflight\ breafing\ of\ 1\ hour$

$df: postflight\ debreafing\ of\ 15\ minutes$

$$Minimum\ number\ of\ duties = \frac{\sum(ft+ct)}{(Mt_d{}^{7} - bf - df) * 60'} \qquad (9)$$

It can be seen in formulation (9) that the minimum number of duties depends on the connect times, since other factors are fixed. However, in reality, the total hours of each duty also

---

[7] details in Flight Operation Management (FOM)

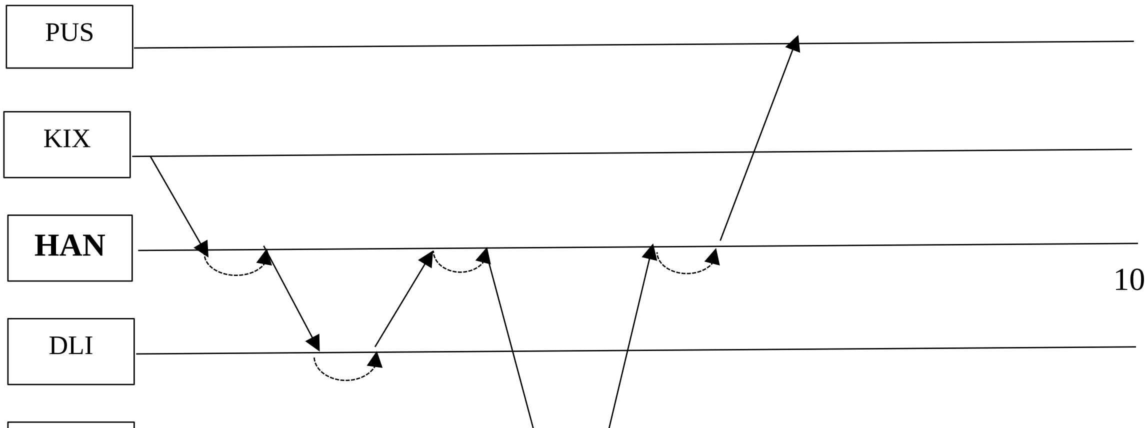
effects the number of duties; hence, when generating duties, the connect time should be reduced to the minimum time and the duty period should be increased to the maximum limitation.

After applying a heuristic of Step 1, we present the typical routes regarding the nodes in diagrams, as shown below. The two short-haul flights round-trips routes are quite popular in the hub and spokes structure and in VNA’s route system, as they take nearly 80% of scheduled flights, as illustrated in Figure 19, below.

The six flights shown in Table 14, below, are operated by the aircraft VNA322 from 1:30 am GMT to 22:15 pm GMT and the total flight time of these flights is 985 minutes (17.25 hours). These flights include four routes, namely, KIX – HAN, HAN – DLI – HAN, HAN – HUI - HAN, and HAN – PUS. HAN is the hub, while DLI and HUI are spokes (domestic airports), and KIX is an abbreviation for Kansai International airport in Japan, PUS stands for Gimhae International airport in South Korea. In the first step, two single flights of the two short-haul round-trip HAN-DLI-HAN route are combined together into one route node, HAN-DLI-HAN. The two single flights of the route HAN – HUI – HAN are united into one node. This step has some advantages, as it reduces the number of nodes in the flight network and does not allow the separation round-trip routes. As shown in the example, 6 flights reduce to 4 nodes of routes and the route HAN – DLI – HAN, as well as HAN – HUI – HAN cannot be separated.

Table 14: Partial data of Jan 1st 2014

Leg No	AC No	Flight No	From	To	Departure	Arrival	Flight Time
1	VNA322	VN331	KIX	HAN	1:30	7:15	5.45
2	VNA322	VN7565	HAN	DLI	8:05	9:55	1.50
3	VNA322	VN7564	DLI	HAN	10:45	12:30	1.45
4	VNA322	VN1547	HAN	HUI	13:30	14:40	1.10
5	VNA322	VN1379	HUI	SGN	15:30	16:50	1.20
6	VNA322	VN420	SGN	PUS	17:40	22:15	4.35



After completing step 1, the flight network of the aircraft VNA322 becomes neat and clearer, as shown in Figure 20, below.

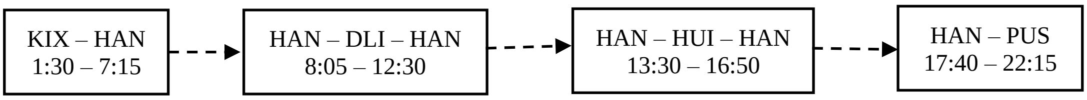


Figure 20: The implemented flight network of VNA322

Figure 21, below, illustrates the second type of VNA route, which is the two international medium-haul flights round-trip route, HAN – KIX – HAN and, clearly, the total flight time of these round-trip flights exceeds the duty period limitation of 11 hours. Therefore, one crew cannot operate the two international medium-haul flights round-trip. For this type of route, it is not possible to combine two flights into one node, as with the domestic round-trip routes, so it leaves each flight as a node with a single flight.

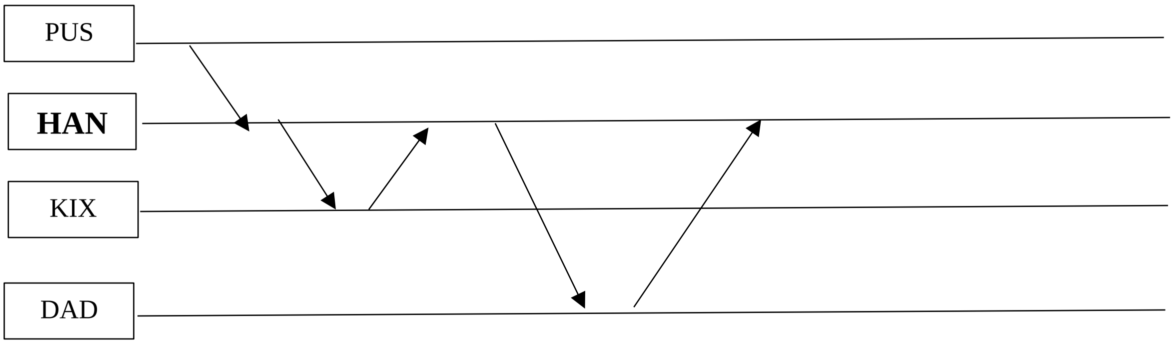


Figure 21: The international medium-haul round-trip route

Figure 22 and Table 15, below, provide the information for the international indirect short-haul round-trip route, the third type of route. These four flights are actually only two indirect international round-trip flights, which have two consecutive flight numbers and each one transits at a middle airport, LPQ, to drop off some passengers but not take on any new passengers. These flights must be connected together and are assigned to only one crew, who can operate these flights in one duty period, since the total flight time of these flights is 445 minutes (7.25 hours), well within the duty period limitation. These flights are regarded as one node in the flight network.

Table 15: The international indirect short-haul flight round-trip route

```
D_ 18***  Leg_ 1_99     Date 1   ACNo_VNA353   FlCNo_VN931   Depart :HAN   Arrival :LPQ
D_ 18***  Leg_ 1_100    Date 1   ACNo_VNA353   FlCNo_VN931   Depart :LPQ   Arrival :REP
D_ 18***  Leg_ 1_101    Date 1   ACNo_VNA353   FlCNo_VN930   Depart :REP   Arrival :LPQ
D_ 18***  Leg_ 1_102    Date 1   ACNo_VNA353   FlCNo_VN930   Depart :LPQ   Arrival :HAN
Departure time of the first leg  is 360
Arrival time of the last leg  is 805
Total connect time  is 150
Total flight time period is 445
```

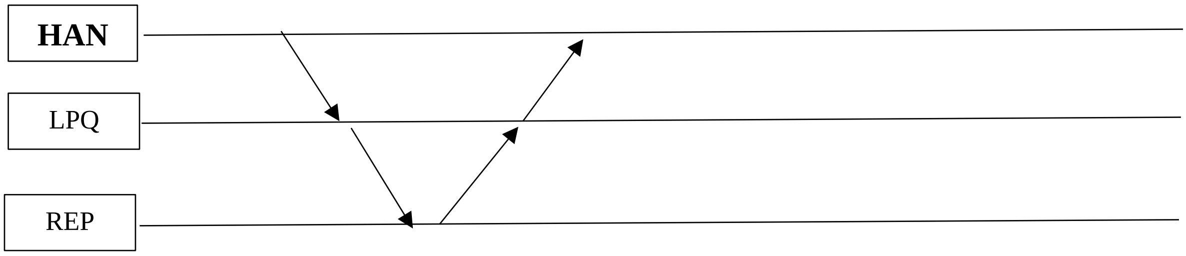


Figure 22: The international indirect short-haul flights round-trip route

Table 16 and Figure 23, below, show data about the last type of VNA route system, which applies the point-to-point architecture. The first flight, with the number VN921, starts from HAN, stops at Wattay International Airport (VTE), Laos and continues flying to Phnom Penh International Airport (PNH), Cambodia and then, finally, ends at SGN. These three flights cannot be separated as it is only one indirect flight with one flight number, VN921. These flights become one route node and are assigned to one crew, since their total flight time is 305 minutes (5.08 hours). Similarly, the flight VN920 starts at SGN, and goes to PNH, and VTE before finishing at HAN. The flight VN920 is also one node and is assigned to one crew, only as one node will always be assigned to one crew only.

Table 16: Point-to-point routes

```
D_ 70***  Leg_ 1_145    Date 1   ACNo_VNA362   FlCNo_VN920   Depart :SGN   Arrival :PNH
D_ 70***  Leg_ 1_146    Date 1   ACNo_VNA362   FlCNo_VN920   Depart :PNH   Arrival :VTE
D_ 70***  Leg_ 1_147    Date 1   ACNo_VNA362   FlCNo_VN920   Depart :VTE   Arrival :HAN
Departure time of the first leg  is 565
Arrival time of the last leg  is 855
Total connect time  is 100
Total flight time period is 290
```

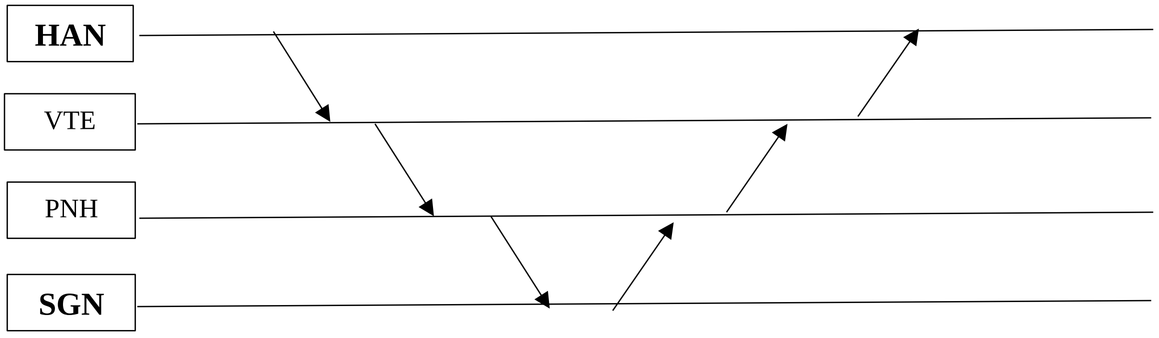


Figure 23: Point-to-Point routes

To sum up, step 1 creates route nodes in the flights network, as shown in Table 17, below. The number of nodes is always smaller than the number of all flights in the timetable (from 190 flights on day 1 down to 103 route nodes). The other advantage of this step is that the connect time between two round-trip flights is set to a minimum of 50 minutes; thus, we just combine them without comparison to other flights.

These nodes are combined together in the second step to form a duty period. Procedure 2 of this step recurs a couple of times to combine nodes together until the duty periods reach the flight time limitation. In order to avoid a crew changing aircraft regularly in their duty period, route nodes for the same airplane are connected first, and only under the circumstance that the connect time between 2 consecutive flights is too long will the crew have to change to another aircraft to continue their duty.

For instance, an aircraft carries eight flights in a day with a total flight time up to 14 hours, as illustrated in Figure 24, below. These flights are too many for one crew's duty and they must be broken down into two smaller duties. The first two short-haul round-trip routes become one duty, while the following two short-haul round-trip routes create a second duty.

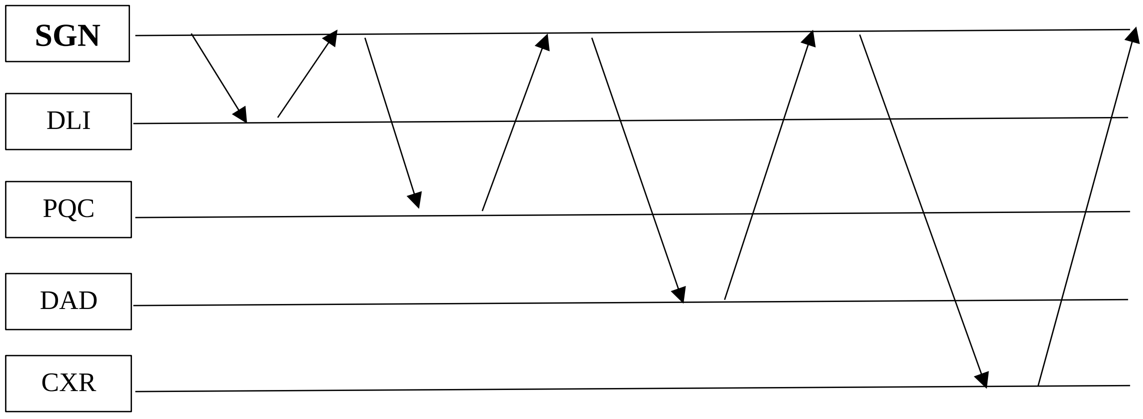


Figure 24: The eight flights on the same aircraft are separated into two small duties.

Table 17: The partial result of Procedure 1 in step 1

```
Number of Nodes 103
NodeNo              LegNo                 Aircraft No           Flight No            Depart     Arrival
*******
N_ 1***  Leg_ 1_1  Date 1     ACNo_VNA322    FlCNo_VN331      Depart :KIX          Arrival :HAN
Departure time of the first leg  is 90
Arrival time of the last leg  is 435
Total connect time  is 0
Total flight time period is 345
----------------------
N_ 2***  Leg_ 1_2  Date 1     ACNo_VNA322    FlCNo_VN7565    Depart :HAN          Arrival :DLI
N_ 2***  Leg_ 1_3  Date 1     ACNo_VNA322    FlCNo_VN7564    Depart :DLI          Arrival :HAN
Departure time of the first leg  is 485
Arrival time of the last leg  is 750
Total connect time  is 50
Total flight time period is 265
----------------------
N_ 3***  Leg_ 1_4  Date 1     ACNo_VNA322    FlCNo_VN1547    Depart :HAN          Arrival :HUI
N_ 3***  Leg_ 1_5  Date 1     ACNo_VNA322    FlCNo_VN1379    Depart :HUI          Arrival :SGN
Departure time of the first leg  is 810
Arrival time of the last leg  is 1010
Total connect time  is 50
Total flight time period is 200
```

In step 2, 103 route nodes on the 1st Jan 2014 were combined to create as many feasible duties as possible. A total of 582 feasible duties was generated as shown in Table 18, below.

The results in Table 18 show that one route node can be combined with several nodes, even when the constraint of the shortest path algorithm is applied. The reason for this has been explained earlier; however, further explanation is necessary here. Duties 3, 4, and 5 in Table 17, below, have two flight legs 1_2 and 1_3 being connected to one each of other flights. The total flight times and connect times of each duty are all different, and if the constraint shortest path algorithm takes connect time as a variable to choose the best option, Duty 3 is chosen. But the flight time is a node weight, which is a constraint to choose, in this case, Duty 4. One more constraint is accommodation cost; both Duties 4 and 5 increase accommodation costs as the last duty destinations are not the home base. So, finally, Duty 3 may become the reasonable option, as no accommodation costs are incurred, and the total connect time and the total duty period are acceptable.

Table 18: A partial result of Procedure 2 for 1st Jan 2014

```
Number of Duty 582
DutyNo   LegNo    Aircraft No            Flight No            Depart     Arrival
*******
D_ 1***  Leg_ 1_1  Date 1     ACNo_VNA322    FlCNo_VN331      Depart :KIX          Arrival :HAN
Departure time of the first leg  is 90
Arrival time of the last leg  is 435
Total connect time  is 0
Total flight time period is 345
----------------------
D_ 2***  Leg_ 1_1  Date 1     ACNo_VNA322    FlCNo_VN331      Depart :KIX          Arrival :HAN
```

As a result, the number of duties is more than the number of nodes, as well as the number of flights, since flights may overlap with several duties. However, this brings more options for the solver to choose the optimal duties.

Table 19: Best result of day 1 after Step 3

```
Number of Duty 70
DutyNo   LegNo    Aircraft No           Flight No            Depart    Arrival
*******
D_ 1***  Leg_ 1_1  Date 1     ACNo_VNA322     FlCNo_VN331       Depart :KIX          Arrival :HAN
Departure time of the first leg  is 90
Arrival time of the last leg  is 435
Total connect time  is 0
Total flight time period is 345
--------------------------------------------
D_ 2***  Leg_ 1_2  Date 1     ACNo_VNA322     FlCNo_VN7565     Depart :HAN         Arrival :DLI
D_ 2***  Leg_ 1_3  Date 1     ACNo_VNA322     FlCNo_VN7564     Depart :DLI          Arrival :HAN
D_ 2***  Leg_ 1_107           Date 1     ACNo_VNA354     FlCNo_VN1527     Depart :HAN         Arrival :DAD
Departure time of the first leg  is 485
Arrival time of the last leg  is 895
Total connect time  is 120
Total flight time period is 410
--------------------------------------------
D_ 3***  Leg_ 1_7  Date 1     ACNo_VNA323     FlCNo_VN1372     Depart :SGN         Arrival :HUI
D_ 3***  Leg_ 1_8  Date 1     ACNo_VNA323     FlCNo_VN1373     Depart :HUI          Arrival :SGN
D_ 3***  Leg_ 1_27            Date 1     ACNo_VNA326     FlCNo_VN655       Depart :SGN         Arrival :SIN
```

Step 3 is to solve the modified duty-period formulation by a solver, 70 optimal duties were chosen by the solver, as shown in Table 19, above. Looking at the optimal duties, it can be seen that there are still some small duties with flight times less than the MDG[8]; these occur due to the connect time constraint preventing the combination of two flights with connect times larger than 180 minutes.

Table 20: Total feasible pairings of Jan 2014 are conducted in Step 4

```
Date:     Pairing No          First Duty No     Next date           Next Duty No     Total Flight time
*******
Date: 1   Pairing_ 1***First Duty_ empty      Next Date_1        Next Duty_1        Total Flight time : 345
Date: 1   Pairing_ 2***First Duty_ 2          Next Date_2        Next Duty_47       Total Flight time : 880
Date: 1   Pairing_ 3***First Duty_ 2          Next Date_3        Next Duty_15       Total Flight time : 1110
…….
Date: 31  Pairing_ 12367***First Duty_ 86     Next Date_ empty   Next Duty_0        Total Flight time : 585
Date: 31  Pairing_ 12368***First Duty_ 87     Next Date_ empty   Next Duty_0        Total Flight time : 400
Date: 31  Pairing_ 12369***First Duty_ 88     Next Date_ empty   Next Duty_0        Total Flight time : 440
---------------------
 Number of Pairing 12369
```

Procedures 3 and 4 of Step 4 paired optimum duties to create feasible pairings, as illustrated in Table 20, above. The result was 12369 feasible pairings being generated to be input into the SPP formulation later. Finally, the solver in Step 5 selected 2082 optimal pairings to be the result of the January schedule, as Table 21, below, illustrates.

---

[8] Minimum Duty Guarantee

Table 21: Total best pairings of Jan 2014 after Step 5

```
Date:      Pairing No          First Duty No      Next date          Next Duty No     Total Flight time  Rest time
*******
Date: 1    Pairing_ 1***First Duty_ 0              Next Date_1        Next Duty_1       Total Flight time : 345
Date: 1    Pairing_ 2***First Duty_ 3              Next Date_         Next Duty_0       Total Flight time : 615
Date: 1    Pairing_ 3***First Duty_ 4              Next Date_         Next Duty_0       Total Flight time : 285
Date: 1    Pairing_ 4***First Duty_ 0              Next Date_1        Next Duty_5       Total Flight time : 105
Date: 1    Pairing_ 5***First Duty_ 2              Next Date_2        Next Duty_47      Total Flight time : 880
Date: 1    Pairing_ 6***First Duty_ 0              Next Date_1        Next Duty_7       Total Flight time : 390
……..
Date: 31   Pairing_ 2081***First Duty_ 87          Next Date_         Next Duty_0       Total Flight time : 400
Date: 31   Pairing_ 2082***First Duty_ 88          Next Date_         Next Duty_0       Total Flight time : 440
----------------------
 Number of Optimal Pairings 2082
```

V.2.5. Result Assessment

Table 22, below, details the information about the duties and pairings being scheduled within 7 days of the first week of Jan 2014. The number of flights every day fluctuated with the different demands of weekdays and the weekend. The numbers of aircraft operating daily varied from 40 – 43.

The lower bound (LB) on the number of duty periods was calculated by the formulation[9] and the upper bound (UB) number of duty periods was the total notes of routes. This means that in the worse case scenario, each crew flies a route, and these routes become the total number of duty periods. The number of feasible duties was created from the combination of the small route nodes in the UB row and the number of duty periods generated each day was nearly 4 times the number of the UB. This shows that the spaces for feasible solutions was quite large, and the subsets of the best duties of each day were chosen from these solutions.

To assess the numbers of the duties and pairings in the best solutions with the LBs and the UBs every day, it can be seen that the numbers of duties are always lower than the means of LB & UB and near the LBs, with the percentages of deviation with lower bounds mostly lower than 30 % and, specifically, some days the percentage is only 19%, which shows the result from using this method to be trustworthy.

The numbers of pairings every day are mostly lower than the numbers of the duties and reduce significantly day by day, since several duties of the first day combine with a number of duties of the following days and only following days' duties without pairing with the previous duties are left to continue pairing with the following days' duties. This means that pairing duties decreases the required number of crew to be scheduled, as each crew may be assigned to two paired duties spanning several days; therefore only day-by-day unassigned duties are reduced.

In general, this proposed method meets all objectives, and the solutions are close to the LBs, as seen in Tables 22 and 23, below. The numbers of duties every day are different because the frequency of the routes of flights during the week are not the same, as some routes only operate two or three times a week, depending on passenger demand. Some other routes have

---

[9] The minimum number of duties to cover all flights every day are calculated by the below formulation: $Number\ of\ duties = \frac{Total\ Flight\ Time\ of\ all\ flights\ \ +\ totalConnectTime}{Maximum\ blocktime\ per\ day - report\ time}$

different flight times each day, such as evening flights for some routes during weekdays but afternoon flights at weekends.

Table 22: Summary of 7 days

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
Number of scheduled flights	215	219	218	204	225	189	185
Number of flights after sorting	190	218	218	207	223	197	182
LB	52	68	67	67	69	67	60
UB	103	118	123	116	123	111	103
Number of feasible duties	582	552	687	552	638	509	470
Number of optimal duties	70	88	87	84	86	80	74
Mean of LB & UB	77.5	93	95	91.5	96	89	81.5
% Deviation with LB	34%	29.4%	29.8%	25.4%	24.6%	19.4%	23.3%
Number of feasible pairings	315	329	359	350	373	349	326
Number of the best pairings	70	72	74	56	72	56	60

Table 23, below, summarizes the 4 weeks' data, with the number of flights increasing gradually from week 3 and significantly in week 4, as it was the Tet Luner new year, and the demand for transportation rose dramatically. The number of duties of the whole month are always smaller the means of LB and UB for that period.

The specific characteristics of VNA make the task of scheduling more complicated when doing it manually, and it was still hard to apply the commercial scheduling software without modification to the VNA system. The results of this algorithm cannot be compared with the schedule which the VNA staff did manually, as they did not provide us with the individual pairing files, but only with the whole schedule as it had been done in Excel. However, comparing the time of 3-5 months in advance taken by five staff to compile the schedule manually with the time taken by the proposed algorithm shows the great advantage of the program.

Table 23: Summary of 4 weeks

	Week 1	Week 2	Week 3	Week 4
Number of flights in GMT	1455	1437	1545	1893
Number of flights after sorting	1435	1431	1546	1862
LB	450	458	488	585
UB	797	810	874	1017
Number of feasible duties	3990	4124	4926	6425
Number of optimal duties	569	577	605	669
Mean (LB & UB)	624	634	681	801
% Deviation with LB	26.4%	26%	24%	14.4%
Number of feasible pairings	2401	2462	3175	3833
Number of optimal pairings	460	440	453	517

The time taken to schedule duties for a whole month was only 17 minutes per fleet. As VNA pay Non-Vietnamese crew a fixed salary, the number of crew needed each day is more important than the total flight time, leading to a reduced number of duties per day and removing the need for deadhead are focused on in this paper, and the proposed method reaches these objectives.

The solution of pairings shows that some duties having an the origin different from the destination cannot be paired with the other following days' duties, due to the shortage of the next days' duties; these are called broken pairings. To overcome this problem, the daily-rolling pairings approach was implemented, as discussed in the next section.

V. 3. Daily rolling pairings approach

V.3.1. Proposed Algorithm

The daily duty period based approach incurs several broken pairings and manual adjustment is required after to amend this. To overcome the disadvantage, a daily-rolling pairings approach was developed in order to provide better results and also to reduce the computational time of the solver. Instead of solving the problem with a whole month's data at a time, which normally takes a few hours for the large data set, based on the idea of column generation, after generating all possible pairings for one day, this set of pairings is input into the daily-rolling pairings formulation, using the solver to select the optimal pairings for a particular day, as shown in Figure 25, below.

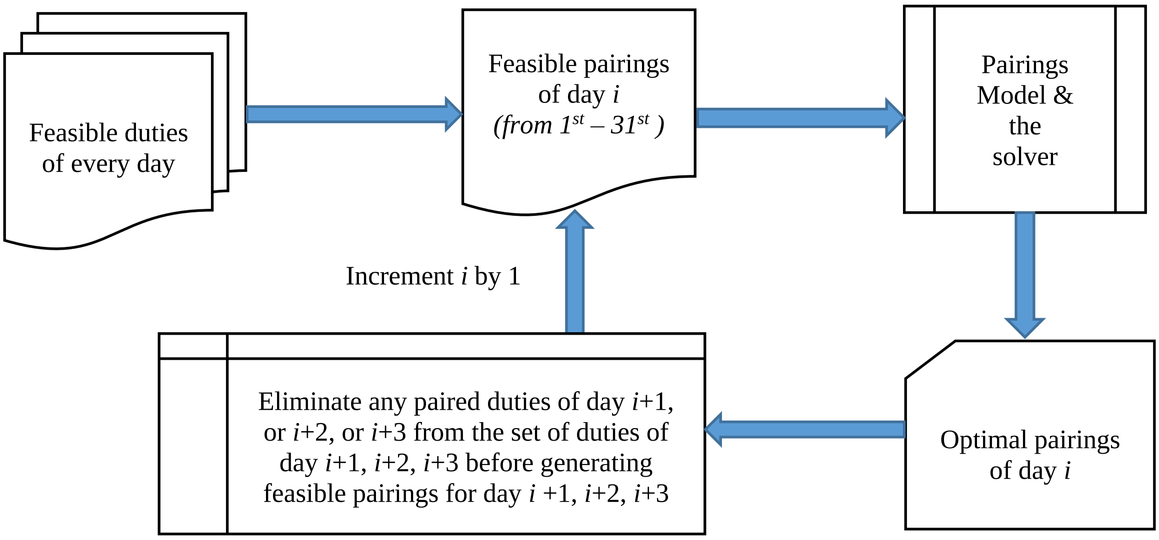


Figure 25: The daily-rolling pairings algorithm

The first two steps are the same as the previous method, only step 3 is discarded. The new step 3 in this approach begins from day 1, whereby any feasible duties of day 1 that start and finish at different places are paired with suitable duties of the following day. Following the duty and pairing generating process of day 1, all pairings are input into the daily-rolling pairings formulation, based on the method devised by Vance *et al.* (1997b) to determine the least number of optimal pairings for day 1. Then, any duties of the other days in the optimal pairings solution of day 1 are eliminated from the duty set of the following days and the process of pairings generation for the next day continues and the solver keep choosing the optimal pairings day by day for the following days until the end of the schedule.

This approach is a hybrid heuristic algorithm, as it combines a heuristic with mathematical formulation. With this algorithm, the computational time is shorter and the quality of the pairings is significantly improved since in each day only all feasible duties and pairings are created and the optimal pairings chosen before moving on to the following days.

The main contribution of the new method is that this approach analyses the problem and solves it in a rolling manner, which combines the enumeration method to generate all possible good quality feasible pairings with the daily-rolling pairing model to select the optimal pairings solution. As the duty-period-based formulation of Vance *et al.* (1997b) has a tighter LP bound than the set partitioning model (Gopalakrishnan and Johnson, 2005), thus our daily-rolling pairing model also 'inherits' this property. Furthermore, the computational time is much shorter, while it still obtains the best solution, which satisfies all the objectives of the problem.

The second contribution is that the suggested approach allows all duties of every day that need to pair with one of the other duties of the following days to have many more opportunities to combine, which as not the case with the first algorithm. Therefore, the optimal solution does not have to be manually modified. In addition, all the problem objectives, as stated in the first approach, are fulfilled.

V.3.2. Problem Formulation

As mentioned above, the method has two stages, the proposed heuristic generates all feasible duties for every day first, and any duties from one day starting and finishing at different places are combined with one of other duties from the following days to form two-duty pairings, the remainder of duties, which do not need to pair with any other duties, becoming one-duty pairings.

After generating all possible feasible pairings for one day, the set of these pairings is input into the ILP formulation, below, to select the optimal solution for that day.

The notation used here for describing the CPP is as follows:

F: Set of flights in a month to be covered

F_i: A $s$ubset of flights on day i to be covered, indexed by j, a flight $f_{ij} \in F_i$, $F_i \subset F$

D: Set of all duties

D_i: A $s$ubset of all duties on day i, indexed k, a duty $d_{ik} \in D_i \subset D$

P_i: A set of all pairings in which the first duties of these pairings start on day i,

indexed l, a pairing $p_{il} \in P_i$

w : A big constant to choose the minimum number of pairings

$c_{d_{ik}}$: Connect time of duty d_{ik}

$r_{p_{il}}$: Rest time of a pairing p_{il}

$x_{d_{ik}}$: Binary decision variable that takes value of 1 if a duty d_{ik} is selected, and 0 otherwise, $d_{ik} \in D_i$

$y_{p_{il}}$: Binary decision variable having the value of 1 if a pair p_{il} is selected, and 0 otherwise, $p_{il} \in P_i$

The daily-rolling pairing formulation for day i is :

$$\min \sum_{d_{ik} \in D_i} c_{d_{ik}} * x_{d_{ik}} + \sum_{p_{il} \in P_i} (r_{p_{il}} + w) y_{p_{il}} \qquad (10)$$

Subject to

$$\sum_{f_{ij} \in D_i} x_{d_{ik}} = 1 \qquad \forall f_{ij} \in F_i \qquad (11)$$

$$\sum_{d_{ik} \in P_i} y_{p_{il}} = x_{d_{ik}} \qquad \forall d_{ik} \in D_i \qquad (12)$$

$$x_{d_{ik}} \in \{0,1\}, \qquad \forall d_{ik} \in D_i \qquad (13)$$

$$y_{p_{il}} \in \{0,1\}, \qquad \forall p_{il} \in P_i \qquad (14)$$

The objective of (10) is to find the minimum total number of good quality pairings for each day with the smallest connect time between flights of duties of the pairing and the smallest rest time between the two duties of the pairing. The cost of a duty is $c_{d_{ik}}$ is the connect time between two consecutive flights in a duty. The connect time in a duty is counted as working time of crew-members and it fluctuates depending on the combination of flights in a duty; therefore, the connect time is a cost of a duty. The excess costs of a pairing is the crew-related cost of meals and accommodation, as well as per diem allowances for VNese crew $r_{p_{il}}$, which has three values. Firstly if a one-duty pairing, $r_{p_{il}} = 0$, secondly, if a two-duty pairing has real rest time, $r_{p_{il}}$= rest time, and, finally, when the pairings of the last few days of the schedule period have second duties belonging to the next scheduling period. In this case, $r_{p_{il}}$ is set with a maximum rest time of 48^{h}. The parameter w is used to achieve the least number of pairings, which covers all flights. As a result, fewer pairings leads to less crew being required and ensures a more efficient use of crew working time.

It seems that there is a conflict between two objectives including the minimum number of duties and the minimum connect time. Since the reduction of the duty numbers is going to increase the connect time, because the more flights are combined in a duty, the more connect time between these flights increases. However, from the VNA point of view, the rise in the connect time is less expensive than the increased duty numbers, as it is quite a small expense compared to the daily salary of a pilot. However, an increase of one duty requires one more pilot and in the case of no more available crew at a base, positioning is needed to transfer a pilot from the other base and one passenger seat is taken for the pilot to travel, together with accommodation expenses.

Another reason of being less concerned about connect time is that with small duty periods of less than 3 hours, the carrier still has to pay at lease 3 hours per duty for the crew[10].In addition, any duty must have 1 hour of pre-flight reporting and 15 minutes of post-flight debriefing; thus, when the number of duties increases the briefing and debriefing time increases as well, and both connect time and report time are idle time in terms of cost effectiveness. The minimum total number of duties is also the minimum total number of crew VNA needed. Constraint (11) requires that each flight must be covered by only one duty.

Constraint (12) requires that for each duty d_{ik} in the set of daily duties D_i , the total value of the pairing variables $y_{p_{il}}$ containing the duty d_{ik} must equal the value of that duty variable $x_{d_{ik}}$, which means that both sides must be 1 or 0.

V.3.3. Solution Method

The steps of the proposed method are listed in Figure 26, below. Steps 1 and 2 are similar to the ones for the daily duty period based optimal pairing algorithm.

---

[10] $DUTY_{COST} = \max\{\, flying\ time\ , Elapse\_Factor * elapse, MDG\}$

Typical values for the cost parameters MDG is 3 hour and Elapse_Factor is 4/7 Barnhart, C. *et al.* (2003) 'Airline Crew Scheduling', in Hall, R.W. (ed.) *Handbook of Transportation Science* Secaucus, US: Kluwer Academic Publishers, pp. 517-560. , Gopalakrishnan, B. and Johnson, E.L. (2005) 'Airline Crew Scheduling: State-of-the-Art', *Annals of Operations Research*, 140(1-4), pp. 305-337..

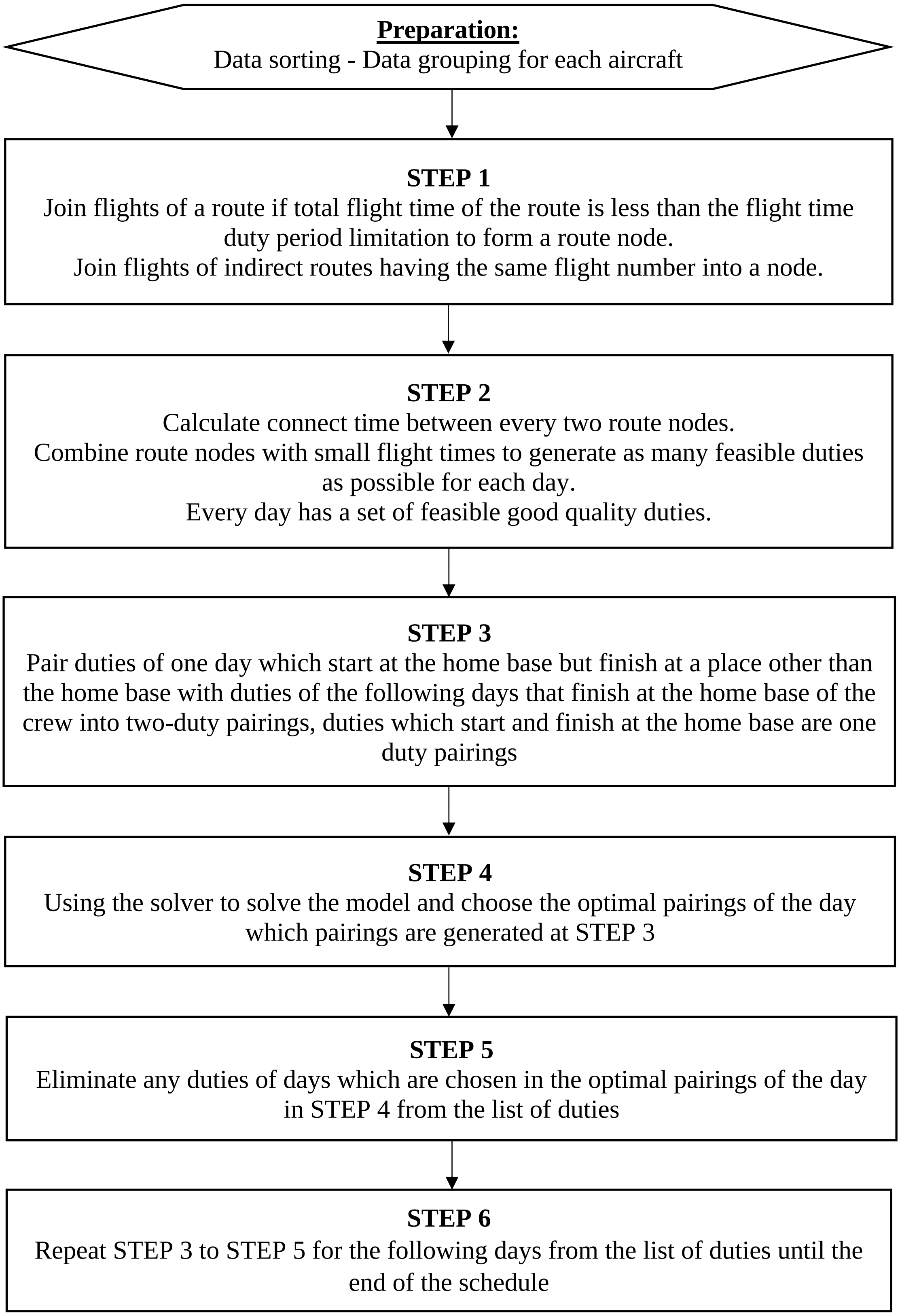


Figure 26: The process of daily-rolling pairing algorithm

The third step is to generate pairings from duties from different days, which have been generated in Step 1 and Step 2. Any duties that start and finish at the same home base do not need to pair with other duties and become one duty pairings. Only duties which start at the home base and finish at other places must be paired with one of the other duties that begins at the same place as the first duty and ends at the same home base of the first duty. For instance, duty 1 of day 1: SGN – PUS has to pair with duty 5 of day 2 PUS - SGN or duty 10 of day 3: PUS – SGN. The minimum rest time is at least equal to the maximum time of the first duty period (11 hours) of any pairings and the maximum rest time is set at 48 hours.

Procedure 4 of Step 3 generates feasible pairings for day 1, the starting day of the schedule, as illustrated below:

***Procedure* 4**: **Pairing Generation for the first day**

```
1. Input: $the\ sets\ of\ all\ Duties\ D_i\ of\ each\ day\ created\ from\ the\ procedures$ 2
2. $P_1 = \emptyset$   // Pairing set of day 1
   $add\ all\ sets\ of\ Duties\ D_i\ into\ a\ big\ set\ D$
3. Loop_1 $duty\ d_{1k} = 1\ to\ n\ in\ the\ set\ of\ the\ first\ day\ Duty\ D_1$
4.    If_1 $(first\ departure\ of\ d_{1k}\ equal\ last\ arrival\ of\ d_{1k})$ ;
              $one -\ duty\ pairing\ is\ added\ in\ P_1$
      Else
5.       If_2 $(first\ departure\ of\ d_{1k}\ equal\ bases$ HAN $or\ SGN$
6.          Loop_2 $duty\ d_{i\gamma} = 1\ to\ m\ in\ the\ Duty\ set\ D$
7.             If_3 $(the\ constraint\ of\ RestTime\ of\ a\ proposed\ pairing\ true)$
                  && $(the\ constraint\ of\ total\ flight\ time\ of\ a\ proposed$
                  $pairing\ true))$
8.                create $the\ pairing\ and\ add\ in\ P_1$
               $End\ If\_3$
            $End\ Loop\_2$
         $End\ If\_2$
      $End\ If\_1$
   $End\ Loop\_1$
```

```
$\boldsymbol{Output}: the\ set\ of\ feasible\ pairings\ P_1$
```

After applying Procedure 4, a set P_1 of feasible pairings is generated including one-duty and two-duty pairings. The next step is to apply the formulation (10) and use the solver to choose the set of best pairings $\overline{P_1}$ from all feasible pairings of the current day (being generated from Procedure 4).

The solver used was Gurobi Java interface which creates an environment object in an integrated development environment, Eclipse®. The environment acts as the container for all data associated with a set of optimization runs. The model (10) is coded within this environment and has an Integer Linear objective function and two sets of decision variables consisting of a set of duty variables and a set of pairing variables.

In addition, there are two sets of constraints, of which the first one (11) ensures that each flight in the flight schedule must be covered by exactly one duty, and the second one (12) confirms that for any duty in a set of all feasible duties D_i is chosen in the solution, and that it appears in only one pairing, otherwise a duty is not in any pairings in the solution.

After obtaining the best pairings solution for the previous day (say day 1), any duties of the following days that are already paired with duties of the previous day must be eliminated from the current day's duty list to avoid duplicate duties in the current day's pairings. Procedure 5 of Step 5 is developed from Procedure 4 but firstly discards any paired duties of the current day from the duty list of current day and updates the current duty list before processing pairing generation for the current day, as below:

$\boldsymbol{Procedure}\ \mathbf{5}$: **Pairing Generation for any day after the first day**

```
1. $\boldsymbol{Input}: the\ set\ of\ accummulated\ bestPairings\ of\ previous\ days\ \overline{P_{\iota-1}},$
   $the\ set\ of\ all\ duties\ in\ the\ month\ D\ and\ the\ set\ of\ duties\ of\ the\ current\ day\ D_i$
2.   $\widehat{D_\iota} = \emptyset$     // updated set of duties of day $i$
3.   $P_i = \emptyset$     // set of feasible Pairings of day $i$
4.   $\boldsymbol{Loop\_1}\ duty\ d_{ik} = 1\ to\ n\ of\ the\ set\ of\ D_i$
5. $\boldsymbol{If\_1}\ (d_{ik}\ is\ not\ in\ the\ set\ \overline{P_{\iota-1}})$ ;
             $add\ duty\ d_{ik}\ in\ the\ \widehat{D_\iota}$
       $\boldsymbol{End\ if\_1}$
```

```
    End Loop_1
6. Loop_2 $duty\ d_{ik} = 1\ to\ m\ in\ \widehat{D_i}$
 7. If_2 $(first\ departure\ of\ d_{ik}\ equal\ last\ arrival\ of\ d_{ik})$ ;
                  $one - duty\ pairing\ is\ added\ in\ P_i$
     Else 8. If_3 $(first\ departure\ of\ d_{ik}\ equal\ bases$ HAN $or\ SGN$
9. Loop_3 $duty\ d_{(i+1)k} = 1\ to\ q\ in\ D$
10. If_4 $(the\ constraint\ of\ RestTime\ of\ a\ proposed\ pairing\ true)$
                    && $(the\ constraint\ of\ total\ flight\ time\ of\ a\ proposed\ pairing\ tr$
11. create $a\ pairing\ and$ add $in\ P_i$
                     End If_4
                   End Loop_3
                End If_3
             End If_2
        End Loop_2
Output: $the\ set\ of\ feasible\ pairings\ P_i$
```

After pairing generation for a day, repeat step 4 with the solver again (as after day 1 presented above) to select the best pairings solution for the current day. Applying the proposed algorithm, no deadheading occurs as it only allows crew changes at bases where the first crew finishes his/her duty and the next crew takes over the aircraft for the next flight. The processing time is fast and the outcome is also of high quality.

V.3.4. Computational Experiments

The program is coded by Java in Eclipse Java 2018-09, with Gurobi 8.0.1 solver. Hardware is a Processor Intel(R) Core(TM) i7-8550U CPU @ 1.80 GHz, 1.99 GHz, RAM 16.0 GB, 64-bit operating system, x64-based Processor. The test data was taken from Vietnam Airlines and it covered timetables for the first month of 2014, running the algorithms in Java using Gurobi 8.0.1 solver.

The same data set of flights in January 2014 was again tested in this algorithm. The first two steps ran the data to generate duties for every day. Step 3 is the pairing generation process by Procedure 3 to generate all feasible pairings and Table 24, below, is the result of 4600 pairings, which include both one duty pairings and two duty pairings. The duties for day 1 can be paired with duties for up to day 3 in order to prevent an infeasible situation, which normally incurs when the constraint (12) is conflicted. This means that duplicated duties exist in several pairings, for instance, duty 5 of day 1 is paired with duty 7 and duty 9 of day 2 respectively, duty 6 of day 1 is also paired with duty 7 and duty 9 of day 2 respectively, and then duty 7 of day 1 is again connected with duties 7 and 9 of day 2 respectively. The solver cannot find a solution satisfying the constraint (12) because there is at least duty 7 or duty 9 of day 2 being duplicated in two pairings and to avoid it, one duty must be combined with many other duties and the number of pairings must be large.

Table 24: The total number of pairings of day 1 is generated

Start date	Pairing No.	First duty	Origin	Dest.	ETD	ETA	Return Date	Second duty	Origin	Dest.	ETD	ETA	Total FT
0	1	0	-	-	-	-	1	1	KIX	HAN	90	435	345
1	2	0	-	-	-	-	1	2	KIX	HAN	90	2305	555
1	3	3	HAN	HAN	485	2370	-	-	-	-	-	-	-
1	4	4	SGN	KHH	230	850	3	795	KHH	SGN	2850	3470	945
….													
1	4598	303	SGN	SGN	55	225	-	-	-	-	-	-	-
1	4599	309	SGN	SGN	745	855	-	-	-	-	-	-	-
1	4600	310	HAN	KHH	350	695	2	-	-	-	-	-	-

For example, duty 4 of day 1 can combine with each one of 52 other duties from day 2 to day 3 or duty 5 is paired with 11 other duties, respectively. Therefore, the number of pairings being generated for day 1 in Table 24 is 4600. The set of pairings on day 1 was input in the ILP formulation and use the solver to obtain the best solution of pairings. Then, 70 optimal pairings on day 1 are chosen as shown in Table 25, below.

Table 25: The best pairings for day 1

Date	Pairing No./ Duty No./ Flight No.	Dept.	ETD	Arr.	ETA	Total Pairing time	Total Flight time	Rest time
1	P No_1							
	Duty 2							
	FlNo. 1-1	KIX	90	HAN	345	-	345	2880
1	P No_2					890	555	995
	Duty 1							
	FlNo. 1-2	HAN	485	DLI	595		110	
	FlNo. 1-3	DLI	645	HAN	750		105	
	FlNo.1-107	HAN	820	DAD	895		75	
	Duty 2							
	FlNo. 2-54	DAD	1890	SGN	1965		75	
	FlNo. 2-35	SGN	2080	PNH	2125		45	
	FlNo. 2-36	PNH	2175	VTE	2255		80	
	FlNo. 2-37	VTE	2305	HAN	2370		65	
1	P No_3					615	400	0
	Duty 1							
	FlNo. 1-7	SGN	130	HUI	210		80	
	FlNo. 1-8	HUI	260	SGN	340		80	
	FlNo. 1-27	SGN	445	SIN	565		120	
	FlNo. 1-28	SIN	625	SGN	745		120	
…	…							
1	P No_69					120	120	0
	Duty 1							
	FlNo.1-214	SGN	745	REP	805		60	
	FlNo.1-215	REP	855	SGN	915		60	
1	P No_70					495	370	6000
	Duty 1							
	FlNo.1-198	HAN	350	CAN	445		95	
	FlNo.1-199	CAN	505	HAN	630		125	
	FlNo. 1-73	HAN	695	KHH	845		150	
Number of the best Pairings 70 Number of flights in the list 221								

Procedure 5 eliminates all paired duties existing in the previous optimal pairings solution from the set of current day duties before continuing to pair the current day’s duties with the subsequent days’ duties day-by-day. Steps 3, 4 and 5 are repeated in sequence until the end of the month. The total number of optimal pairings for Jan, 2014 are 2334 pairings.

V.3.5. Result Assessment

Table 26 below, details the result of the duties and pairings generation within the first 7 days of Jan 2014. The feasible duties for every day are created from the combination of the route nodes of every day in the flight network and the number of duties is around three times the number of nodes, which indicates that all possible combinations of nodes are covered. The number of generated feasible pairings is much larger than the result of the first algorithm to prove that they contain all good quality feasible pairings and the best pairings solution may be globally optimal.

In general, the numbers of daily best pairings in the solution increase differently by up to a maximum of 15 pairings a day, but the deviation percentages with LBs are really small, even in one day, the number of pairing reaches the lower bound, besides, no broken pairings and no manual correction after the scheduing process. This explains the reason why the first algorithm has broken pairings. Because the first one choose the numbers of best duties for each day before pairing them with duties of the following days, and this leads to reducing the number of duties combinations. Consequently, there are duties of the previous days which do not have suitable duties of the following days to pair with. The number of pairings in day 1 of January is quite high comparing with LBs, due to there are several broken pairings with the previous days of December being counted in January.

Table 26: Pairings Statistic of the 1st week of Jan 2014

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
No. of flights in GMT	215	219	218	204	225	189	185
No. of flights after sorting in local time	190	218	218	207	223	197	182
LBs	52	68	67	**67**	69	67	60
No. of nodes of routes UBs	103	118	123	116	123	111	103
Mean of LBs & UBs	77.5	93	95	91.5	96	89	81.5
No. of feasible duties	312	330	382	321	375	296	284
No. of feasible pairings	4600	3889	4977	4751	5151	5705	4619
% Deviation with LBs	34.6%	16.1%	14.9%	0	10.1%	5.9%	3.3%
No. of optimal pairings	70	79	77	**67**	76	71	62

In contrast, this method generates all possible feasible duties for every day and as many as possible of these duties of each day in sequence are paired with duties of the following days. Then, the solver chooses the subset of the best pairings from the pool of the newly generated feasible pairings of each day right away before moving on to create pairings for the next day.

Table 27, below, summarizes the result of 4 weeks; the number of flights increase gradually from week 3 and significantly in week 4 as it is the Tet Luner new year period, and the demand of transportation rises dramatically. Consequently, the result of this algorithm covers all scheduled flights and provides the best quality pairings, which start and finish at the same home base with the minimum connect time and the least rest time from the pools of feasible pairings being generated every day. Moreover, the deviation percentage with LBs is really small, down to 0.8% on the last week of January .

Table 27: Statistics of 4 weeks in Jan 2014

	Week 1	Week 2	Week 3	Week 4
No. of flights in GMT	1455	1437	1545	1893
No. of flights after sorting	1435	1431	1546	1862
No. of feasible duties	2300	2319	2761	3533
LBs	450	458	488	585
UBs	797	810	874	1017
% Deviation with LBs	11.5%	8.7%	5.5%	0.8%
No. of feasible pairings	33692	40400	60162	99509
No. of the best pairings	502	498	515	580

V.3.6. Conclusion

The CPP is a complex optimization problem, which has been researched thoroughly in the operation research field (Anbil *et al.*, 1991; Barnhart *et al.*, 2003; Gopalakrishnan and Johnson, 2005; Kasirzadeh, Saddoune and Soumis, 2015; Deveci and Demirel, 2018b). In this thesis, we have studied and solved the CPP of VNA with unique characteristics and more real constraints than the previous similar research. We focused our objectives on minimizing crew-based costs, including dead-headings, international layovers and domestics layovers. We proposed two optimization driven heuristic algorithms that combine heuristics to generate duties and pairings and exact methods to obtain the best solution. Duties and pairings are enumerated step by step and the feasible pairings are input into the ILP model and use the solver to choose the solution for the pairings. The solution methods are solved in a rolling manner; hence, the computational time is short.

The suggested approaches are based on analysing the VNA network structure of flights and as well as its cost structure. The identification of the best method to reduce the number of daily duties, deadheading, and layover costs is the most important objective. The way to decrease crew costs is to reduce the numbers of duties, which, in turn, decreases the number of required crew every day.

Furthermore, the suggested algorithm eliminates deadheading because the duty period of the pairing is always kept within the flight time limitation, and the control of origin and destination of a duty period also contributes to reducing the number of deadhead flights. This means that the deadheading of the Airbus 320 fleet is controlled.

In addition, layover costs are also reduced by fixing the changes of crew and aircraft at home bases or appropriate destinations. The final destinations of duties are referred to bases or domestic airports. However, some international layovers are fixed to medium-haul routes, such as SGN – NRT – SGN, DAD – ICN – DAD, SGN – KHH – SGN, due to the flight time limitation and rest regulations. The proposed algorithm cannot change the medium-haul routes, but domestic layovers are reduced significantly through this method.

The proposed method satisfies all of the objectives and provides a good result to apply in reality. This method not only shows clearly the objectives in the formulation, but through constraints and domains as well; therefore, it is partly similar to constraint programming (CP). The solution is globally optimal and it is very suitable for the carrier, which has a few fleets of aircraft.

VI. The Crew Rostering Problem of VNA

VI. 1. Objectives of the Crew Rostering Problem

Typically, one of the main objectives of the Crew Rostering Problem (CRP) focuses on real costs. Therefore, the proposed method must cover all flight pairings, even if, deadhead or overtime scheduling is required to minimize or avoid open time[11] or unassigned flight pairings. Regarding crew members' preferences, fairness or equal assignment is of primary concern. In addition, due to some specific characteristics of VNA mentioned above, the difficulties of the VNA rostering problem are mainly due to many levels of the rules and regulations, the working-pattern of the non-VNese crew-members and the balance of work load and income between two groups of crew. The goals of the CRP are to satisfy the preferences of three partners, as shown in Figure 27, below. Consequently, the proposed algorithms concentrate on solving these objectives.

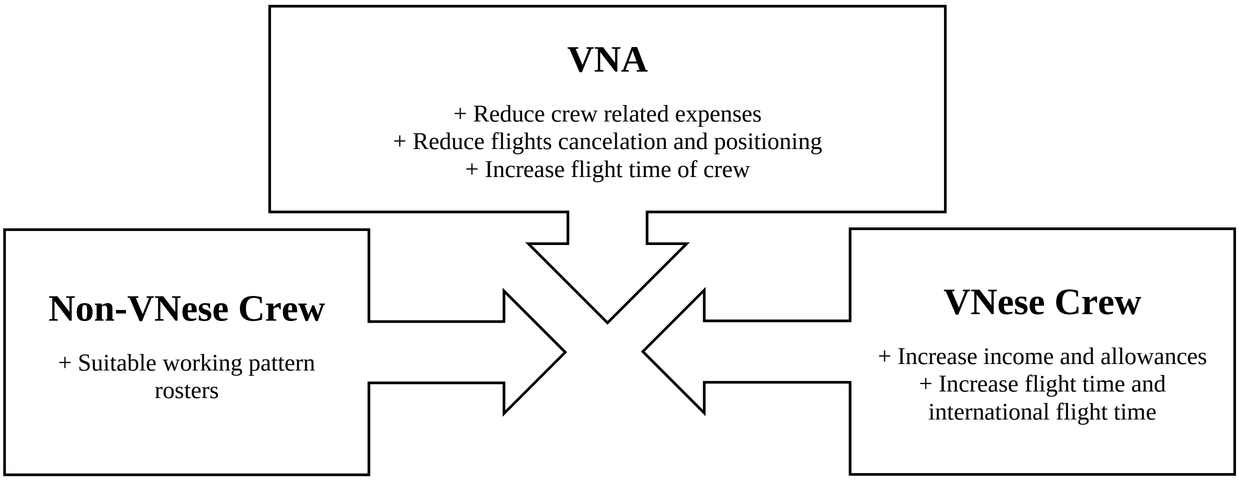


Figure 27: Three partners in the VNA crew scheduling problem

To VNA, the carrier and employer, the main goals are to decrease employee related expenditure, minimizing overtime payment and to reduce the flights' cancelation and positioning. In order to accommodate these goals, VNA must use their manpower efficiently. Currently, VNA pays non-VNese crew a fixed package, and expects crew to work up to the maximum flight time of 100h/ month (see chapter IV.6). However, manual scheduling may not be able to assign flights appropriately and accurately to each crew's timelines to meet that requirement. The heuristic algorithm can generate many rosters for each crew in a short

---

[11] Open time is the situation often occurring in the scheduling problem which flight pairings are unassigned

time; therefore, the selection of appropriate rosters for crew is easier and total flight times of crew members increase significantly, leading to the required amount of crew decrease.

The demands of non-VNese crew are simpler, in that their rosters fulfil their working pattern and vacation periods as well as their preferences. Some crew prefer to fly on their favourite routes or to start their duties at particular times (morning, afternoon or evening).

The requirements of the VNese crew are more complicated than their counterparts, since their incomes are not fixed but based on their flying time and per diems during the month. VNA pays VNese crew a credit-hour salary and a per diem allowance for the total amount of time while they are away from their home base on duty. However, their salaries are much lower than non-VNese crews', so they expect to be allocated more flights and prefer the international flight pairings with an international per diem allowance.

To balance the requirements of three parties, the main objectives of the problem are to improve the productivity of crew members (currently flight time of crew members is only 72 hours average) and to meet the VNese crew references which are allocated long hours pairing and international flights. This procedure ensures a productivity close to 95% for each crew leading to reduce the required number of crew but still cover all flights, this save much more for VNA as salaries of crew is the second highest expenditure of the carrier.

The proposed approach is a two – phase algorithm to achieve the best quality set of rosters for the VNA crew members. It is similar to the generate-and-optimize approach, where, in the first phase, a heuristic generates as many legal rosters as possible, and in the second phase, the IPL is applied to select the best roster solution by the solver.

VI. 2. Proposed Approach

Taking the above factors into consideration, we combined the rostering and seniority based method[12] to assign pairings to crew. Seniority-based personalized schedules focus the priority on the satisfaction of the more senior crew-members, since they also have other ground duties or managerial tasks. In addition, non-VNese crew-members have specific working patterns as defined in the labour agreement. For instance, two week on two week off means two working weeks and two free weeks. The proposed method has several advantages of computational time and a significant reduction in the required number of crew

---

[12] Two types of personalized schedules are rostering and seniority-based. Maenhout, B. and Vanhoucke, M. (2010) 'A hybrid scatter search heuristic for personalized crew rostering in the airline industry', *European Journal of Operational Research*, 206(1), pp. 155-167.

members to cover all scheduled flight legs, as well as the capability of choosing crew members on the basis of nationality or ranking priority as table 28 below:

Table 28: Ranking table of crew

Crew	**Ranking**
Vietnamese – management/ instructors	1
NonVietnamese - instructors	2
Vietnamese	3
NonVietnamese	4

.

The contributions of this method are explained in detail as follows: firstly, the combination of rostering and the seniority-based personalized schedules method meets the requirements of the management crew-members with ground tasks and flight duties. The ranking system attached to each crew member's record allows the heuristic to identify the important crew members and priotizes them in the pairings allocation. Secondly, we apply several techniques of the constraint programming algorithm, such as domain, local consistency and constraint propagation, in the heuristics to generate a domain for each roster variable and to reduce the search space of each variable. Therefore, the processing of roster generation is quicker and fulfills the preferences of the working-pattern of non-VNese crew-members. Finally, we also achieve fairness for Vietnamese crew members by having the cost parameters for VNese crew members different from that of the non-VNese crew. These cost parameters effect the selection by the solver of the VNese rosters with lower cost than the non-VNese roster.

We have developed several heuristic algorithms to generate as many feasible legal rosters as possible for all crew members in a reasonable computational time (about 10 to a maximum of 20 rosters for each crew). The heuristics are developed based on randomization with constraint programming techniques and a local search and presented in the sequence of the objectives. One of the first heurisitics to generate rosters is the multi start randomization method (Juan, A., Faulin, J., Ferrer, A., Lourenço, H.,&Barrios, B. 2013a), (Armas, J. *et al.* 2016), in which each crew member in the ordered list of crew members is assigned to randomly picked pairings that suit his/her timeline. However, before flight pairings can be assigned, many rules and regulations must be checked and fulfilled.

During the scheduling period, crew conduct many tasks, including flights duties, reserved ground duties and periodic training. In addition, they also have days off or vacations booked in advance. As a result, each roster is a subset of all of these activities, which have been arranged properly in order to fit the timeline of each crew member. The pre-assigned ground tasks, periodic training or reserved vacations must be considered prior to the allocation of flight duties in rosters and, to make the process easier, these activities are assigned firstly in the timeline of each crew and any available days left are for flight pairings.

As mentioned above, each crew member has several rosters created for them so, let $r_j^k \quad be$ a roster k of crew j, elements in a r_j^k are several flight pairings i. A pairing is represented by the following fact:

$$pairing(ID, Date, FirstDuty, SecondDate, SecondDuty, TotalFlightTime, RestTime, ConnectTime)$$

Where ID is a number that identifies uniquely a pairing, $FirstDuty$ is the duty of the first day ($Date$) of the pairing, each duty includes one or several flights originating at the home base of the crew. $SecondDuty$ is the second duty of the pairing, starting on $SecondDate$, which also contains one or several flights finally ending at the home base as well. $SecondDate$ may be the consecutive day, or two, or three day later from the first day ($Date$) if a crew member lays over two days at a place before conducting the $SecondDuty$ back to the homebase.

A pairing must have $ID, Date$ and $FirstDuty$ but may not have $SecondDate$ and$SecondDuty$. For example, a one-duty pairing has the first flight starting at the home base and the last flight of the duty finishing at the home base. Some pairings of the first few days of the schedule have the first duties belonging to the previous schedule and only the second duties in the current schedule, whereas several pairings of the last few days of the schedule have only the first duties, as the second duties belonging to the next schedule.

$TotalFlightTime$ is the total flight time in the air of all flights in the pairing, $ConnectTime$ is the break time between two consecutive flights of the same duty and is also added to the duty period time. $RestTime$ is the time between two duties in the pairing, which is at least 11 hours (660 minutes) or equal the first duty time. Rest time includes the whole time of the pairing that the crew are away from their home base (TAFB) on duty. Therefore, the accommodation and meals expenses for the crew are paid by VNA. In addition, a per diem allowance is also paid to VNese crew only, whereas Non-VNese counterparts are paid fixed packages, so they do not have per diem allowances. The 100 hours flight time limitation consists of total flight time (block time) and connect time only, not rest time or reporting time.

Some of the constraints are applied to roster generation are as follows:

Notation of the algorithm

P : the set of all best pairings of the whole schedule period, indexed i, a pairing $p_i \in P$

d_i^1 : the start date (the first day or date of duty 1) of a pairing p_i

d_i^2 : the second date (the return date or start date of duty 2) of a pairing p_i

r_j^k : a roster k of crew j

tft_j^k : total flight time of roster k of crew j

ft_i : total flight time of pairing p_i

$cmdp_j^k$: the cumulative duty period of roster k of crew j

$c7dp_j^k$: the cumulative duty period of roster k of crew j in 7 days

ct_i : connection time of pairing p_i

$mrt7_j^k$: minimum rest time in 7 days of a roster k of crew j

$mrt10_j^k$: minimum rest time in 10 days of a roster k of crew j

mft_j^k : minimum free time of all duties

1. Pairings in a roster cannot be overlapped and after finishing any pairings, the crew is allowed to have 1 day off (free day), which means that *SecondDate* of $SecondDuty$ of pairing *i* must be smaller than *Date* of the pairing *i*+1 at least 1 day (1 free day).

   $$d_i^2 + 1 < d_{i+1}^1 d_{1p_{i+1}}, \quad \forall\, p_i, p_{i+1} \in r_j^k r_{kj}$$

2. The total flight time of each roster k being assigned to crew j must be less than the flight time limitation of 100 hours or 6000 minutes in any 28 consecutive days. $tft_j^k = \sum_{p_i:\, d_n^2 - d_i^1 = 28 \in r_j^k} ft_{p_i} <= 6000 \;, \quad \forall\, p_i \in r_j^k$

3. The cumulative duty period cannot exceed 190 hours over 28 consecutive days or 60 hours in any 7 consecutive days. A duty period includes the flight time of all flights and the connect time ct_i between them.

   $$cmdp_j^k = \sum_{p_i:\, d_n^2 - d_i^1 = 28 \in r_j^k} \left(ft_{p_i} + ct_{p_i}\right) <= 11{,}400, \quad \forall p_i \in r_j^k$$

$$c7dp_j^k \;=\; \sum_{p_i:\, d_n^2 \;-\, d_i^1 = 7 \,\in\, r_j^k} \quad (ft_{p_i} + ct_{p_i}) \;<=\; 3{,}600 \;, \qquad \forall\, p_i \in r_j^k$$

4. The minimum rest period between any two duties is 11 hours and is increased to at least one 36 hours period within 7 consecutive days or one 60 hours period within 10 consecutive days.

$$mrt7_j^k \;=\; d_{i+1}^1 \;-\; d_i^2 \qquad \geq\; 2160 \qquad \forall\, p_i, p_{i+1} \,\in\, r_j^k \;\; |d_{i+n}^2 \;-\; d_i^1 \;\; = 7$$

$$mrt10_j^k \;=\; d_{i+1}^1 \;-\; d_i^2 \;\;\geq\; 3600 \qquad \forall\, p_i, p_{i+1} \,\in\, r_j^k \mid d_{i+n}^2 \;-\; d_i^1 \qquad = 10$$

5. The minimum time free of all duty and standby is at least 7 days per calendar month.

$$mft_j^k \;=\; \sum_{p_i \in r_j^k} \quad d_{i+1}^1 \;-\; d_i^2 \qquad > 7days * 1440 \qquad , \qquad \forall\, p_i \,\in r_j^k$$

After generating a set of rosters $R = \left\{ r_1^1 \;,\; r_1^2 \;, \ldots r_j^k \;, \ldots, r_n^m \right\}$ with $(m * n)$ rosters for all n crew-members, in which each crew member has a subset of m (about 20 in our experiments) rosters. The second phase is to select the most suitable roster for each crew member from the set R, so each roster is a variable of the ILP model and the product of m rosters for each crew member multiplied by n crew members is the number of variables of the model. The optimal solution contains only n rosters that satisfy all conditions of crew-members C as well as the objectives of the model.

VI. 3. Problem Formulation

In the first phase, the random heuristic was applied to generate the set of legal feasible rosters for all crew members and the rules and regulations, together with reasonable preferences of crew as constraints. The constraints are extremely important as they affect both memory requirements and execution time. Some constraints were added to limit $RestTime$ and the number of duties in a pairing in order to eliminate poor quality rosters; this technique reduces the processing time of the solver in the next stage.

In the second phase, all generated rosters were input into the ILP model and the solver was used to select the optimal solution. Let M be the number of all pairings. In a feasible solution, each pairing must be assigned to only one crew and all crew members must have at most one roster, or none, as the number of crew currently exceeds the number of required crew. The minimum number of crew is roughly calculated, as per the following formulation and is regarded as the lower bound of the model:

$$Minimum\ No.of\ required\ crew = \frac{Total\ flight\ time\ of\ schedule\ in\ a\ month}{The\ maximum\ flight\ time\ limitation\ per\ crew\ per\ month}$$

Since the main objective of the formulation is to minimize the number of rosters (minimize the number of the required crew or increase the productivity of crew), big value w is added to reduce the number of rosters being chosen. The second objective is to satisfy the reference of VNese crew for increasing the income. As costs of flight time and connect time of duties are fixed, only a value of $RestTime$ allowance in a pairing is effected to VNese crew's income therefore, we took $RestTime$ as the cost of a pairing. In addition, in order to allocate long time pairings to VNese crew, we apply a cost parameter and set values of a cost parameter being 1 for VNese crew and 2 for Non-VNese crew, thus favouring the VNese crew when choosing long pairings.

The formulation is detailed below:

Notation:

C : set of all pilots, indexed $j \in C$

C_H : set of pilots based at HAN $\subset C$

C_S : set of pilots based at SGN $\subset C$

D : set of all pairings P and pre − assigned activities.

P : set of all pairings, indexed i, a pairing $p_i \in P \subset D$

P_H : set of all pairings starting from HAN, $p_i \in P_H \in P \subset D$

P_S : set of all pairings starting from SGN, $p_i \in P_S \in P \subset D$

OA : set of all pre − assigned activities $\subset$ D

rp_i : a rest time of a pairing i

R : a set of all feasible rosters

R_j : a subset of all feasible rosters of crew $j \in C$, indexed $k \in R_j \subset R$

r_j^k : a feasible roster $k \in R_j$ of crew $j \in C$, containing several pairings p_i that are allocated to crew j $p_i \in r_j^k$

Parameters

c_j : a cost parameter of a crew j, if a crew j is VNese, $c_j = 1$,

if a crew j is Non − VNese, $c_j = 2$

w : a big value of 1 000 000

The total cost of a roster k of crew j : $tc_j^k = \sum_{p_i \in r_j^k} (c_j * rp_i)$

Binary variable x_j^k has a value of 1 if a roster k of crew j r_j^k is chosen, otherwise 0.

Objective function:

$$\min \sum_{j\in C_r}\sum_{k\in R} w * \ x_j^k \quad + \sum_{j\in C_r}\sum_{k\in R} tc_j^k \quad * x_j^k$$

in a short form : $\min \sum_{j\in C_r} \sum_{k\in R}(w + \ tc_j^k \ ) * x_j^k$

in detail : $\min \sum_{j\in C} \ \sum_{k\in R} \sum_{p_i\in x_j^k} \left(w + \left(c_j * \ rp_i\right)\right) * \ x_j^k$ (1)

St. $\sum_{k\in R_j} x_j^k \leq 1, \qquad \forall\, j \ \in C$ (2)

$\sum_{j\in C} \sum_{p_i\in r_j^k \in R_j} x_j^k = 1, \qquad \forall\, p_i \ \in P$ (3)

$x_j^k \ \in \{0,1\}$ (4)

The objectives of this problem are the reduction of the number of rosters and the fairness of the schedule. The objective (1) has two elements, the first of which is the big value *w* to decrease the number of rosters chosen , while the second element is the *Rest time* in minutes. .

The fairness of the schedule is in the form of a cost parameter which has two values: a value of 1 if the roster r_j^k is allocated to VNese crew and a value of 2 for the roster of Non_VNese crew. The total cost of a roster tc_j^k is the sum of the cost of all pairings in the roster $k \in \ R_j$ being allocated to crew *j*. The value of rp_i is the rest time of each pairing p_i if it has TAFB more than 1 day or 1 if it is one day pairing, which starts and ends on the same day at the same home base. When a roster having many long rest time pairings is assigned to VNese crew with the cost parameter $c_j = 1$, the tc_j^k of the roster is the true value of the sum of all rest time in the roster, but if the same roster is allocated to Non-VNese crew, the tc_j^k is now double the true value of all rest time because c_j = 2.

$tc_j^k = \sum_{p_i\in r_j^k}(c_j * rp_i) \ = \ \sum_{p_i\in r_j^k} 1 * \ rp_i$ (for VNese crew)

$tc_j^k \quad = \sum_{p_i\in r_j^k}(c_j * rp_i) \ = \ \sum_{p_i\in r_j^k} 2 * \ rp_i$ (for Non_VNese crew)

The solver is going to choose any rosters with long rest times being allocated to VNese crew, since this reduces the total cost of the solution. The parameter *w* is a large number to reduce the number of rosters.

As explained above, there were specific reasons why rest time was chosen as the cost factor of a pairing and given the role of cost parameter. First of all, Non-VNese crew are paid a fixed package, and as long as their flight time is within the limitation, their salary is not changed. However, VNese crew are paid on the basis of their fly time credit hours [13] and per diem allowances; therefore, the more they fly the more their salary increases and the long rest time pairings also affect their income, as the TAFB allowances are added to their salary.

To VNA, the minimization of required crew to cover all flights is vital since a procedure being proposed in the NBAA Management Guide (2014) roughly estimates that the required number of crew of VNA is 160, much less than the current number of 578. Furthermore, the salary payment of Non-Vnese crew is much more than per diem payments to VNese crew; Therefore, a reduction in the number of Non-VNese crew members saves a lot of expenditure for the carrier. As a result, we used the cost parameter c_j to favour VNese crew when allocating long hour pairings to VNese crew, as well as choosing more VNese crew's rosters than non-VNese's rosters. The result of this model will also minimize the number of required crew members.

Constraint (2) ensures that every crew member is assigned a maximum of one roster from a subset of rosters R_j, or none if all pairings for the schedule are already assigned. The reason for this is that manual scheduling cannot effectively allocate crew to flights, leading to the requirement for more crew to cover all flight pairings or having to cancel flights or position pilots from another base during the high season. Whereas, currently the real average flying hours of crew is just only 72 hours a month, which is much lower than the limitation of 100 hours flight time per month. The proposed method creates rosters with a total flight time up to 100 hours and the required amount of crew decreases significantly.

Constraint (3) makes sure that each pairing is allocated to only one roster of one crew, with no unassigned pairings, this constraint is to avoid duplicated pairings in the whole schedule. Duplicate pairings occur when two crew members are allocated to one pairing. The constraint (4) ensures the binary variable of a roster, which has a value of 1 if it is chosen or 0 if it is not.

VI. 4. Solution Method

Since the enumeration of all possible rosters is usually impossible, when tackling the crew rostering problem from a mathematical – programming point of view, column generation techniques are often employed, how ever it has some drawback in real-life scenarios (Armas,

---

[13] Vietnamese crew payment formulation:

$$Job\ title\ benefits + credit\ hour\ payment\ (flying\ time) + layover\ expenses$$

J. *et al.* 2016). Because of the complexity and difficulty of the roster generation, in the first stage, there are several methods to generate rosters for crew members. The main objective of this stage is that all rosters are legal and feasible, which means that each roster must fulfil all horizontal rules and fit properly with the timeline of the individual crew member. We applied the concept of constraint programming in designing the heuristic for roster generation. The set of variables of the first stage was the set of rosters for the list of crew in which each crew member has a subset of rosters and the set of values is all pairings which are generated by the crew pairing problem and preassigned tasks. Each roster or variable has a subset of values that are a subset of suitable pairings with the crew member's time line. In addition, there are several constraints which available pairings must satisfy before being added into the roster variable, and a solution of each roster variable is only a partial of a subset of those available pairings.

The roster generation is constructed by several algorithms based on the reasearches of Lucic, P. and Teodorovic, D. ,(2007), Juan, A., Faulin, J., Ferrer, A., Lourenço, H.,&Barrios, B. (2013a) , and Armas, J. *et al.* (2016), presented below :

VI.4.1. Algorithm 1: Crew-by-Crew random algorithm

```
1. Loop 1 : ( crew $c_j = 1\ to\ n \in C$)
2.     Allocate all pre-assigned tasks of crew $c_j$ into individual crew timeline
3.     Create a subset of available pairings for crew $c_j$ which match with available
       working days of the crew.
       End loop 1
4. Loop 2: ( a roster $r_j^k = 1\ to\ 10$ are generated for a crew-member)
5.     Loop 3: (a crew $c_j = 1\ to\ n$  in the crew list).
6.         Loop4 (accummulated flight time < limitation or domain is empty)
                   Pick randomly pairing by pairing from his/her subset of
                   available pairings (domain) to generate his/her roster
                   Check whether a new pairing is suitable with the previous
                   pairings in the processing roster before adding it in the roster.
```

```
                Eliminate any pairings which occur on the same period with the assigned pairings and during the compulsary free time of the crew from the available pairings domain.
                Until the accummulated flight time or the accumulated pairing time of the roster reach to the maximum standards.
            End loop4
        End loop3
    End loop 2
```

The roster generation is separately conducted for each base; therefore, the crew list contains only crew members residing at a particular home base and the available pairings are also devided into each home base, in which the first flight of a duty or a pairing decides the homebase of the duty/ the pairing.

Before the scheduling process, all crew members must provide the pre-assigned tasks and booked vacation, as well as preferences in advance. The first three steps define a domain for each crew; firstly, all of these pre-assigned tasks are allocated into each crew member’s timeline in line 2. After finishing step 2, the rest of the days in each crew member’s timeline is available for flight pairings allocation. Step 3 in line 3 is to create a domain for each crew member, which contains all suitable pairings to be allocated to the crew member. Each pairing is checked with available time windows in the particular crew member’s time line, before being added to the domain of the crew member.

Other conditions are checked in advance, such as visas for international flights, certification for specific airports etc. As a result, one pairing can fit in many crew members’ timelines, therefore, it belongs to many domains. In contrast, many pairings are suitable for a crew member’s time line. As step 3 is done, each crew member has a domain of all available pairings, which are ready for the rostering process.

Line 4 starts the loop of roster generation; the size of the loop is dependent on the computational time and computer memory. In this problem, we set the size of $r = 10$ as the computational time as reasonable (around 15 minutes to finish). Line 5 is to pick each crew member and begin the procedure of roster generation for them. The list of crew members could be sorted into many orders, depending on the objective of the problem and the algorithm.

Line 6 is a roster generation procedure, as shown in Figure 28, below, and in this algorithm, we used a randomization heuristic to pick a pairing and propagation technique to reduce the search space of the crew member's domain. The randomly picked first pairing was added directly into a newly created roster without applying any constraints to it. After adding the pairing into the roster, the propagation process starts to eliminate the first pairing together with all other pairings in the domain that occur at the same period of the first pairing, because they do not fit into the new timeline of the crew member anymore. The search space reduces significantly after propagation.

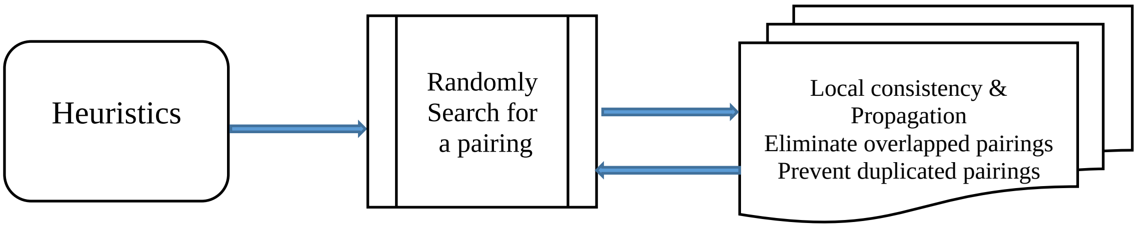


Figure 28: The process of randomized roster generation procedure

In order to reduce further the search space of the domain, any pairings which occur in the compulsary free time before and after the first pairing are also eliminated. From the second pairing picking onward, after choosing a pairing, the pairing must fulfill several constraints regarding the accummulated flight time limitation over 7 days, free time between two pairings and free time over 7 days. These constraints are applied on a rolling basis from day to day and week to week. The roster processing finishes when the accummulated flight time of the roster reaches the limitation or when none of available pairings is left in the domain.

Lines 4,5, and 6 continue with the other crew members until they all have rosters or all of pairings are assigned.

The "Crew-by-Crew" heuristic algorithm is fast (Lucic, P. and Teodorovic, D. ,2007), and can generate the maximum flight time rosters as VNA requested since the stopping criteria of the loop is the maximum limitations of the total flight time and the total pairing time, which means that each loop of roster generation only exists when the total flight time of a roster or the total pairing time of a roster reach the limitation or when all pairings have been assigned. However, this algorithm does not assure the quality of rosters regarding the balance of rosters among the whole schedule, the preferences of management or senior crew members as well as the working-pattern of Non-Vietnamese crew. Therefore, several algorithms were developed later to improve the quality of the rosters.

The approrpriate sorting of crew members supports the process of pairing allocation and feasible solutions were obtained. The crew list is sorted after Step 3 and before Step 4 in ascending order and then decending order, based on their size of available pairings domains. In ascending order, the crew member with the smallest available pairings domain would be assigned to the pairings at the very beginning. In contrast, in decending order, the crew members with the largest size of pairings domains are generated rosters first.

With this method, a pairing can be assigned to more than 1 crew member if it is suitable for any particular crew member's time line in each loop of roster generation. Therefore, the numbers of duplicated pairings are huge, leading to infeasible solution in the ILP model later.

To implement the drawback of the first algorithm, the second method avoids duplicated pairings in each loop of roster generation by imposing the condition that any pairing is allocated to only one crew member. In order to do this, an all-different constraint is enforced in the form of an assigned-pairings set being created to keep pairings which are already allocated to previous crew members. With this technique, in each loop of roster generation, each pairing is allocated to one roster of a crew member only and a loop will stop when the whole list of pairings is allocated to crew members or the size of the assigned-pairing set is equal to the size of available pairings set.

Nevertheless, crew members who are at the top of the crew list have more chances of obtaining the maximum number of flight time rosters, whereas the ones near the bottom of the crew list may not have many choices, as most of the pairings in the list have been assigned already. It means that if the number of crew members exceeds the necessary crew number for all flight pairings in the whole schedule (as one constraint to escape the roster generation loop 4 above is the maximum flight time hours of a roster – 100 hours), there are several crew members who do not have any rosters as all of pairings are allocated.

Consequently, to avoid bias, the assigned-pairings set is emptied before starting a new loop of roster generation. This opens more opportunities for all crew members being allocated to appropriate pairings. This method has the advantage of reducing the number of required crew to the minimum. However, some crew at the bottom of the crew list may not have opportunities to have rosters and the inequality in each roster is quite large.

The third method implements the second one and sorts the crew list in ascending order of available days (ones with the least available days first). The crew having least available days have their rosters generated first and more options to pick their favourite pairings. However, some crew members who are available for the whole month but at the end of the list, may not have opportunities to allocate any pairings if the flights schedule is already assigned.

The fourth method is the opposite of the third one regarding the order of the crew list, which is sorted in decreasing order of available days (ones with the largest available days first). To reduce the case of available crew being unassigned, this method sorts the crew list into the order of most available crew members at the top of the list; therefore, they have more opportunities to obtain full flight time rosters.

Each method has advantages and disadvantages. Therefore, a combination of all methods to generate as many rosters as possible will obtain a set of quality rosters in terms of roster balancing. The set of all rosters being generated at this stage was input into the ILP model for the solver to choose the best solution.

After selecting the best solution, the rosters are not balanced regarding the number of pairings and the total flight time of each roster. In order to achieve the fainess of rosters among all crew members, a roster balancing algorithm was developed to balance the rosters, by using an exchanging scheme of a local search to subtract or move one or several pairings from the largest rosters to the smallest rosters.

Roster Balancing Algorithm

```
1. Loop 1 - find out the roster with maximum accummulated flight time (MaxFT) and
            the roster with minimum accummulated flight time (MinFT).
2.     Loop 2 - choose each pairing in the MaxFT roster and check whether it
       fits into the timeline of the crew who has the MinFT roster.
3.         If_1 it is suitable with the timeline of the crew of MinFT roster
4.             Loop 3 – check the combination of the new pairing with each
                   current pairing in the MinxFT roster to see if the
                   combination satisfies all constraints or not
5.                 If_2 All combinations are fulfilled, add the new pairing
                       into the MinFT roster.
                   End If_2
               End Loop 3
           End If_1
```

```
          End Loop 2
6.        Check the condition between the MaxFT roster and MinFT roster being
          satisfied or not
     End Loop1
```

The balancing algorithm moves one pairing in each loop from the maximum flight time roster (max FT roster) to the minimum flight time roster (min FT roster) in a loop. After each move, the calculation of max FT roster and min FT roster is conducted again to decide which are the new max FT roster and min FT roster. The adjustment keeps doing this until the termination criterion is fulfilled.

VI.4.2. Algorithm 2: Working-pattern Roster Generation Algorithm (for Non-Vietnamese crew)

The crew-by-crew method has some advantages, as presented above. However, the above methods do not meet the working patterns of non-VNese crew, such as 2 week working and 2 week off. We proposed another new algoirthm, based on the combination of the ‘day-by-day’ method and the ‘crew-by-crew’ method. The day-by-day loop is in the form of *bound consistency*, whereby the domain of a crew member is updated from the total number of pairings in the period of the crew member’s timeline D (pairings) to only pairings in the period which starts from the day of the loop to the end of the schedule, e.g. domain D contains all pairings in the range :

$$[\text{min of a range } (start - day\ of\ pairing = day\ of\ the\ loop$$

$$\ldots\ldots\ldots\ldots\ldots \ \text{max of a range } (start - day\ of\ pairing = end\ of\ the\ schedule]$$

in each loop, which means that the start day of a roster is changed day by day. With this technique, the roster is not only fulfilling the crew member’s preference for a specific working-pattern, but also the search space of domain is reduced significantly.

Working-pattern Roster Generation Algorithm

1. Create a list of crew members

2. Allocate all pre-assigned tasks of each crew into individual crew timeline

3. Create a subset of available pairings for each crew which match with the available working days of each crew.

4. The crew list is sorted in the acending order first

```
5.  Loop 1: day $d$ = 1 until 15 ( each day starts a new roster for every crew)
        Assigned-pairings set = $\emptyset$
6.      Loop 2: pick a crew $c_j$ from the sorted crew list
7.          Loop 3: roster generation process for crew $j$
8.      Loop 4: pick randomly a pairing $p_i$ occurring on of after the
        day $d$ of the loop 1 (the start day of a roster) in the
        available pairing list   of the crew $j$ (the crew of Loop2)
9.                  If (Check the pairing with other pairings in the
                    processing roster is suitable, add it in the roster)
                        Add the pairing Id into a roster
                        Add the pairing Id into the assigned-pairings set
                        Take the $d_{p_i}^2$ ( $SecondDate$) of the new
                        added pairing of the day and add 1 free day
                    End If
                    Jump to the ($d_{p_i}^2$ +1 )$^{th}$ day
                    Pick randomly pairings of the (Date2 +1)$^{th}$ day
                End Loop 4
            End loop 3 ends when a pairing reaches to the last day or
            Accummulated FT and AccPT reach to the limitation
        End loop 2
    End loop 1
```

10. Check whether the size of an assigned-pairing set is equal to the all pairings set's size, or not. If the assigned-pairing set's size is smaller than the latter one, allocate any suitable unassigned pairing to the small rosters.

The first four steps are the same as the previous algorithms, while line 5 starts the day loop. In order to provide equal opportunities for all crew members, in each loop we used both a decending sorted crew list and an acending sorted crew list to generate rosters for all crew members. Lines 7-9 are the roster generation procedure which develops from the previous roster generation procedure and adds the *start day* to the roster. Inside the procedure, we updated the domain of the pairings with only pairings starting from the *start day* onward before picking pairings. The search space of the updated domain is smaller day-by-day and only the few crew members who require these preferences are allocated to the pairings occuring in those periods of time.

The All-different constraint is still adhered to in order to obtain the optimal solution, as the assigned-pairings set keeps all assigned pairings in each loop of roster generation and is emptied before starting each new loop.

There are also some small rosters in the best rosters solution and in order to reduce the number of rosters, an implemented algorithm based on local search was developed to move pairings from the smallest roster to others, as follows:

Implemented algorithm

```
Search for the smallest flight time roster
Loop 1( pairing $p_i$ =0 to n in the pairings set of the smallest roster
        Loop 2: ( roster $r_j^k$ = 0 to m in RosterMap)
                If ( check all conditions before input a pairing $p_i$ into a new roster)
                        the new pairing is fitted in the current roster
                         update tempRoster after adding new minPairngInfo
                        break;
                EndIf
        End loop 2
    End loop 1.
```

VI.4.3. Algorithm 3: Seniority-based Priority Algorithm (for Management crew)

In VNA, almost all management staff are pilots as well, especially the head positions of crew division, such as the heads of each fleet, the head of aviation safety division, the head of training division, the head of human resource etc. In order to keep their licences active, management crew still fly, but the flight time requirements are less than that for other crew members and rosters must be fitted in with their ground tasks. As a result of having these ground tasks, they have privileges to choose the specific flights they prefer which suit their timetables. In addition, other senior crew may also have the right to choose flights. For that purpose, we designed an algorithm that combines a depth first search algorithm for management crew with a randomization algorithm for other crew in order to schedule this real scenario.

Seniority-based Priority algorithm

```
1. Loop 1: day $d =1 to 15$
2.     Loop 2: crew $c_j =1 to q$ in the Ascending sorted crew list
3.             If (rank of crew j ==1) && (rank of crew j ==2)
4.                 Loop 3 (pair $p_i = 0 ... m$ in the available pairings of the
                       particular crew
                       Search for the unassigned pairing with largest rest
                       time
                       Create a roster $r_j^k$ for a crew j and
                       Add pairing $p_i$ in the roster of the crew and
                           continue with the following days
                 End Loop 3
             End if
         End loop 2
5.     Loop 4: crew $c_j = 1$ to $q$ in the Descending sorted crew list
6              If (rank of crew j who is not management or senior crew)
7                  Loop 5 (pair $p_i = 0 ... m$ in the available pairing of the
                       particular crew
                       Randomly pick any unassigned pairings for the crew
                       Create a roster for a crew and Add $p_i$ in the roster of
                       the crew and continue with the following days
                 End loop 5
             End if
         End loop 4
8.           Loop 6: sort the roster map
9.               If ((rosters size >= 5 pairings && total flight time >3500')
                 Or (rosters size <= 15 pairings && total flight time <=5500')
                 Or (rank of crew j ==1) or (rank of crew j ==2))
```

```
                    add to Big roster map
                End if
            End loop 6
10. Loop 7: For $c_j$ = 1 to n in the Ascending sorted crew list
11.     If (crew $j$ is not in the Big roster map)
                Randomly pick any unassigned pairings for the crew which do
                not have rosters
                Create a roster for a crew and
                Add $p_i$ in the roster of the crew and
                continue with the following days
            End if
        End loop 7
 End loop 1
```

Line 1 starts the loop for beginning the roster day as the second algorithm. Lines 2 to 4 are the roster generation procedure for management and senior crew (rankings of 1 and 2, respectively) based on the greedy algorithm. The search criteria is the longest rest time of each pairing, as the flight time requirement of these crew is lower than normal crew and most of them are middle- to late-middle age. Therefore, long rest time pairings or pairings with one or two landings are their preferences. All pairings assigned to the management and senior crew are removed from the available pairings set, and the rest of the available pairings sets are allocated to other crew members using the randomization method.

Lines 5 to 8 are the roster generation procedure for normal crew members using the descending crew list, as in the previous algorithm. Line 9 sorts the first map of all generated rosters in order to keep the management and senior crew rosters unchanged. Any rosters that are too big or too small are deleted and regenerated in the Line 10. The purpose of Line 10 and 11 is to balance the workload of the rosters between crew members in the current situation. Line 11 assigns the remainder of the pairings to the crew members.

In the set of rosters, which is generated for all crew, there are sometimes several duplicated pairings and some crew have more than one roster. In order to obtain a feasible solution, first the constraints of the formulation must be relaxed as below:

$$\text{St.} \quad \sum_{k\in R_j} x_j^k \geq 1, \qquad \forall j \in C \qquad (5)$$

$$\sum_{j\in C} \sum_{i\ \in r_j^k \in R_j} x_j^k \geq 1, \qquad \forall i \in P \qquad (6)$$

$$x_j^k \in \{0,1\} \qquad (4)$$

Constraint (5) become equal to or greater than 1 inequation to force the solver to choose rosters for all crew and constraint (6) become equal to or greater than 1 inequation to allow duplicate pairings. Since there are duplicate pairings, an adjustment algorithm is used to delete duplicate pairings in several rosters.

Adjustment algorithm

```
1: Loop 1 (roster $r_j^k$ : the roster map)
                Count pairings in each roster and record them in a CountNoPairing map.
                Count and record no of crewId and roster of crewId into a CountNoCrew
                map.
        End loop 1
2: Loop 2 ( dupcrewId ∈ duplicatedCrew)
                Compare rosters of a crewId
                Keep a big size roster
 3:             If_1 (pairings in small rosters are fitted in the big size roster
                        add them in the big size roster
                else
                        create a NotDupPairings set and add them into for other rosters}
                End if_1
        End loop 2
 4:     Delete small duplicated rosters
 5: Loop 3 (duppairingId ∈ duplicatedPairing)
                Compare rosters of a pairingId
 6:             If_2 (pairings in big rosters are deleted from the big size roster
                        keep them in the small size roster
                End if_2
        End loop 3
```

Line 1 sorts the roster list to find out which crew members have more than one roster and which pairings are duplicated. Lines 2 to 4 are to eliminate a crew member's duplicate rosters, and to keep the big roster and delete the small rosters of crew having more than one roster. Before deleting the small rosters, a pairing examination is conducted to move unduplicated pairings from small rosters to other suitable rosters. Lines 5 to 6 eliminate duplicate pairings, which are kept in the small rosters and deleted from the big size rosters.

This algorithm satisfies the management crew members' requirements and reduces the number of required crew. Further constraints in the loop of randomly picking rosters can be set to reduce small rosters.

VI. 5. Computational Experiments

VI.5.1. Algorithm 1: Crew-by-Crew random algorithm

The results of each algorithm and the combination of them are presented sequentially in Tables 29 below:

Table 29: The result of each methods running separately with the data of Hanoi base

	Method 1	**Method 2**	**Method 3**	**Method 4**
Number of rosters generation loops for each crew	20	20	20	20
Number of crew assigned	95	80	76	77
Total number of rosters generated	1900	773	737	738
Total number of best rosters selected	0	74	72	71
Number of VNese crew	0	39	42	45
Number of non-VNese crew	0	35	30	26

Table 29, above, shows the results of the four methods, three of which were implemented from the crew-by-crew random algorithm (Method 1) presented the previous section. The first method generates 20 rosters for each crew member, but when we input these rosters into the model and used the solver to select the optimal solution, it came out as unfeasible. The reason for this was that the number of duplicate pairings was large, since this method allows any crew member to be allocated to any suitable pairings for his/her timeline, and some pairings can be assigned to 40 crew members.

Even when we relaxed the model to allow one pairing to be assigned to more than 1 crew member and that a crew member can have more than 1 roster, the number of rosters in the solution was still larger than the best result of the other methods (125 rosters with a gap of 7% compared to about 75 rosters of the other methods) and a running time of nearly 3 hours. However, we still kept this method in order to combine it with the other methods, as the combination of the first one with the others brought a better solution than each of the other methods individually.

The second method only generated a total of 773 rosters for 80 crew members out of the total of 95 crew. This means that, in each loop of roster generation, a maximum 80 crew are needed to cover all pairings, starting from Hanoi base of the schedule. Since this method has an assigned-pairings set to keep all pairings once they are already assigned to any crew member, subsequent crew members do not have any chance of being allocated to those pairings. This technique prevents the duplicated pairings in each loop of roster generation, but it may have the disadvantage that crew members at the top of the crew list have more privileges for choosing the suitable pairings than those at the bottom of the crew list. To avoid this bias, each crew is picked randomly in order to allow all crew to have a roster generated, and the assigned-pairing set is emptied when it starts the new loop of roster generation, which means that with the new loop of roster generation, all crew have the same rights to be assigned any pairings.

In the method 3, before selecting a crew member to generate a roster, the crew list is sorted in the accending order of available days of each crew member. As a result, both the number of rosters and the number of required crew decrease (737 rosters and 76 crew). This indicates that the sorted crew list helps the pairings' allocation effectively and efficiently.

The fourth method shows a much better result, as the number of required crew in the best roster solution is reduced to only 71, to cover all pairings from the Hanoi base. The sorted crew list is in the decending order, which is any crew having most available days at the top of the list, and they are chosen to allocate to as many pairings as they can manage; this leads to less crew being required. The last three methods have the same disadvantage of inequality and unbalanced roster generation for all crew members. This is overcome by the combination of these methods, as presented in Tables 30 and 31 below.

Each method is combined with one of the others and the number of crew members being allocated rosters and the number of rosters being generated increases significantly as seen in Table 30, below. Almost the total of crew members are generated rosters, except the combination of Method 2 and Method 3 only creates rosters for 88 crew members. This brings more options for the solver to select the best solution of rosters and all crew members have equality to be allocated. Method 4 brings the better result, when joining with the other methods, and only 70 crew are required to cover all pairings at the Hanoi base.

The lower bound of crew at the Hanoi base is calculated by the formulation[14] as below:

---

[14] $Minimum\ No\ of\ required\ crew = \frac{Total\ flight\ time\ of\ schedule\ in\ a\ month}{The\ maximum\ flight\ time\ limitation\ per\ crew\ per\ month}$

$$Lower\ bound\ of\ crew\ number = \frac{378595\ flight\ time\ in\ minutes}{6000\ minutes = 100\ hours\ flight\ time\ limits} = 63\ crew$$

Table 30: The result of the combination of each two methods running with the data of Hanoi Base

	Method 1 and Method 2	Method 1 and Method 3	Method 1 and Method 4	Method 2 and Method 3	Method 2 and Method 4	Method 3 and Method 4
10 Loops for each method	20	20	20	20	20	20
Number of Rosters	1723	1908	1674	1504	1494	1464
Crews being assigned	95	95	95	88	94	95
Pairings being allocated	933	933	933	933	933	933
Optimal Roster List size	75	73	70	72	73	70
Vietnamese crew being assigned	40	43	45	42	45	45
Non-Vietnamese crew being assigned	35	30	25	30	28	25

The combination of all 4 methods to generate 1916 rosters does not bring a better result, as shown in Table 31, below, and computational time is longer as the number of duplicate pairings has increased.

Table 31: The result of the combination of 4 methods together with the data of Hanoi Base

Method 1, Method 2, Method 3 and Method 4 together
Number of loops for each method = 5 * 4 = 20
Number of crew assigned = 95
Total number of rosters generated = 1916
Total number of optimal rosters selected = 72

Number of VNese crew = 45 Vnese Crew

Number of non-VNese crew = 27 non-VNese Crew

The result of the scheduled pairings from the Saigon Base also shows that Method 4 is the best method of all to generate rosters, as it requires fewer crew member to cover all flight pairings, as illustrated in Tables 32,33, and 34. The lower bound of crew at the Saigon base is calculated by the formulation as below:

$$Lower\ bound\ of\ crew\ number = \frac{424700\ flight\ time\ in\ minutes}{6000\ minutes = 100\ hours\ flight\ time\ limits} = 71\ crew$$

Table 32: The result of each method running separately with the data of Saigon Base

	Method 1	**Method 2**	**Method 3**	**Method 4**
Number of rosters generation loops for each crew	20	20	20	20
Number of crew assigned	114	89	89	84
Total number of rosters generated	2280	867	874	811
Total number of optimal rosters selected	0	85	86	78
Number of VNese crew	0	47	51	53
Number of non-VNese crew	0	38	35	25

From the results of all 4 algorithms with all flight pairings, it can be seen that the algorithm to generate rosters is the most important factor in finding the optimal solution. All rosters of all crew members are not equal and balanced; therefore, the adjusted method enhances the objective of a balanced workload between the crew rosters.

Table 33: The result of the combination of each two methods running with the data of SGN Base

	Method 1 and Method 2	Method 1 and Method 3	Method 1 and Method 4	Method 2 and Method 3	Method 2 and Method 4	Method 3 and Method 4
10 Loops for each method	20	20	20	20	20	20
Number of Rosters	1993	2010	1962	1727	1672	1696

Crew being assigned	114	114	114	105	112	114
Optimal Roster List	83	86	80	84	79	80
Vietnamese crew being assigned	47	51	55	50	54	55
Non-Vietnamese crew being assigned	36	35	25	34	25	25

The combination of all methods

Table 34: The result of the combination of 4 methods together with the data of the SGN Base

Method 1, Method 2, Method 3 and Method 4 together
Number of loops for each method = 5 * 4 = 20
Number of crew assigned = 114
Total number of rosters generated = 2220
Total number of optimal rosters selected = 80
Number of VNese crew = 54 Number of non-VNese crew = 26

As the results show in Tables 29-34, when the crew list is sorted in descending order (method 4 and the combination of method 1 and method 4), the number of rosters to cover all flight pairings is smallest. The objectives are not only to achieve the smallest amount of rest time, but also the smallest number of required crew and that the number of VNese crew is more than double the number of non-VNese crew being chosen. This solution meets the main objectives of the crew rostering problem.

VI.5.2. Algorithm 2: Working-pattern Roster Generation Algorithm (for Non-Vietnamese crew)

The working-pattern roster generation algorithm was tested on both the SGN base data and the HAN base data, with the day loop from day 1 to day 15. We ran both ascending and descending crew lists to reduce bias and the number of rosters being generated was 2255 rosters for all 114 crew members at the SGN base. The total number of best rosters in the solution was 80, see Table 35 below. With the HAN base, the result came out much better, with 67 rosters, while the lower bound was 63 rosters. In the solution, there is a roster starting from day 21 and finishing at day 31; another roster starts at day 14 and finishes at day 29, as seen in the appendix.

With the new algorithm, the number of feasible rosters being generated increased, but the computational time was the same and the best pairings solution reduced to only 67 rosters (HAN base), still covering all flight pairings. However, due to pairings being chosen randomly, the result is different each time of running. The implemented algorithm was developed to move pairings from small rosters into bigger ones and to reduce the total number of rosters overall.

The number of rosters is reduced partly due to that the bound consistency limits domains (the range of available pairings) of each crew member and, as the range is smaller and the number of crew in those ranges is also reduced, only a few crew members in these ranges have more opportunities to be assigned all suitable pairings. It also forces the the randomization function to pick many pairings in these small ranges.

For example, if a domain (a range of available pairings) is large, the random function of computer chooses any pairings that fit in the time line of a crew member and fulfil all constraints, and the free time between any two consecutive pairings is also large, this leads to the crew has to work for the whole period of schedule (for the whole month). While as the domain is reduced, the number of available pairings is also smaller and the random function has to pick pairings in this small subset of available pairings to satisfy the roster requirements; the free time between two consecutive pairings just meets the minimum free time only. This advantage satisfies VNA and Non-VNese crew as well, because the number of required crew reduces and the foreign crew works intensively for a shorter period of time for VNA and then has a large amount free time staying in their countries.

Table 35: The result of Working-pattern Roster Generation Algorithm for SGN Base

```
Crew Id  Crew First Name          Date      From                Total Flight Time
RosterID_83 of crewID_16
Pairing_1           Aras            1         SGN                 FlightTime_125
Pairing_2          Aras             5         SGN                 FlightTime_500
Pairing_3          Aras             6         SGN                 FlightTime_990
Pairing_4          Aras             9         SGN                 FlightTime_610
Pairing_5          Aras             13        SGN                 FlightTime_430
Pairing_6          Aras             15        SGN                 FlightTime_565
Pairing_7          Aras             17        SGN                 FlightTime_320
Pairing_8          Aras             18        SGN                 FlightTime_300
Pairing_9           Aras            19        SGN                  FlightTime_370
Pairing_10         Aras             20        SGN                 FlightTime_240
              ACC FT_4450     ACC RT_3512    ACC PT_6260
 **************************RosterID_73 of crewID_76
Pairing_1          Tuan16           20        SGN                  FlightTime_745
Pairing_2           Tuan16 25       SGN                 FlightTime_500
Pairing_3           Tuan16 26       SGN                  FlightTime_535
Pairing_4           Tuan16 30       SGN                 FlightTime_350
              ACC FT_2130     ACC RT_3257    ACC PT_2795
 **************************


There are 46 Vietnamese crew being assigned and  34 Non-Vietnamese crew being assigned

Number of Rosters 80
```

VI.5.3. Algorithm 3: Seniority-based Priority Algorithm (for Management crew)

With the Seniority-based Priority algorithm, the management crew have the rank of 1 and senior crew have the rank of 2. These crew are allocated the best suitable pairings at the time of scheduling by a greedy method before the rest of the crew list, as shown in Table 36, below. All pairings that are allocated to management and senior crew are eliminated from the available pairngs list; therefore, at the second stage, the solver places all management and senior crew rosters into the best roster solution.

Table 36: The roster of a management crew is generated by the greedy algorithm

```
RosterID_1 of crewID_2   FirstName_Ha     LastName_Tran   Ranking: 1
-------------------------
Pairing_1       PairId_54       Date1_1       From_HAN      To_HAN   FlightTime_560
Pairing_2       PairId_210      Date1_3       From_HAN      To_HAN   FlightTime_555
Pairing_3       PairId_438      Date1_6       From_HAN      To_KHH   FlightTime_150
Pairing_4       PairId_639      Date1_10      From_HAN      To_HAN   FlightTime_660
Pairing_5       PairId_1143     Date1_17      From_HAN      To_HAN   FlightTime_655
Pairing_6       PairId_1351     Date1_19      From_HAN      To_HAN   FlightTime_555
Pairing_7       PairId_1585     Date1_22      From_HAN      To_PUS    FlightTime_220
Pairing_8       PairId_1855     Date1_26      From_HAN      To_HAN   FlightTime_805
Pairing_9       PairId_2206     Date1_30      From_HAN      To_HAN   FlightTime_140

 ACC FT_4300     ACC RT_22806   ACC PT_5845
 Size of RosterPair is 9
 Management crew
```

The number of best rosters in the solution shown in Table 37, below, is larger than the solution of the crew-by-crew randomization algorithm. However, this algorithm satisfies the crew preferences, which are more important than the number of required crew.

Table 37: The number of rosters is 80

```
RosterID_1 of crewID_2          FirstName_Ha           LastName_Tran           ranking: 1
-------------------------
RosterID_2 of crewID_82         FirstName_Tung1        LastName_Tung1          ranking: 2
-------------------------
………….

RosterID_78 of crewID_61        FirstName_Ninh         LastName_Ninh           ranking: 3
-------------------------
RosterID_79 of crewID_73        FirstName_Soroka       LastName_Soroka         ranking: 3
-------------------------
RosterID_80 of crewID_114       FirstName_Trang        LastName_Trang          ranking: 3
-------------------------
***************************
There are 44 Vietnamese crew being assigned and  36 Non-Vietnamese crew being assigned
There are 932 pairings 932 pairs being allocated on the total of Pairs 932
There are 353781 minutes rest time of 44 VNese crew
There are 198828 minutes rest time of 36 NonVietnamese crew
Number of Rosters 80
```

There are a few crew member who have more than one roster and one pairing is duplicated. Thus, we applied the adjust algorithm to move and delete some rosters and the duplicate pairing. The outcome satisfies all the objectives.

VI. 6. Result Assessment

The proposed algorithms meet all the objectives and the results are very good. The formulation with cost parameters allows the management to adjust the ratio of crew in each group. Specifically, we can decide the number of VNese and non-VNese crew in the schedule and calculate the minimum number of required crew with a simple formulation. With the combination of different heuristics, we can allocate particular crew to specific pairings or routes, as well as provide suitable rosters for crew with different work-patterns.

The domain architechture and propagation algorithm help to reduce the search space of domains, leading to a shorter computational time. Furthermore, local consistency prevents duplicate rosters and makes the solver select the best solution quicker.

As we use the randomization method to pick a pairing, the result of each run is different but not too much. The best solution of rosters achieved is 67 rosters and the Lower bound of 63 rosters for the HAN base and 79 rosters over the LB of 70 rosters for the SGN base.

When we apply the Seniority-based Priority algorithm, the number of rosters increases because the total flight time of management and senior crew reduce, leading to the number of required crew rises to cover all flight pairings of the whole schedule.

VI. 7. Summary

This chapter applies a multi-start randomized heuristic of Juan, A., Faulin, J., Ferrer, A., Lourenço, H.,&Barrios, B. (2013a) along with multi-objective optimization model for solving the real-life crew rostering problem of VNA. The section describes realistic constraints, regulations, and rules that have not been considered in the literature so far. Our algorithm is designed to provide quality solutions satisfying these real-life specifications while, at the same time, it aims at partly balancing the income among the VNese crew's group. Thus, our approach promotes corporate social responsibility by distributing the workload in a fair way, these aspects have selfdom been considered in the crew scheduling literature. The experimental tests show that our algorithm is capable of generating feasible quality solutions to the real-life crew rostering problems in just a short computational time.

In our case, the carrier used to try a commercial software to generate an optimal solution. However, the program was not able to consider all the real-life constraints, regulations, and rules of the airline. For that reason, the solution provided by the software required further manual adjustments before it could be implemented in real – life. This manual setting process might take several hours in the best of the cases and, of course, after the modifications the resulting solution is not optimal anymore. At the same time, our algorithm aims at optimizing the workload distribution among the two different crew groups – Vietnamese crew and NonVietnamese crew. In addition, our algorithm also satisfies the working patterns for NonVietnamese crew. This way, our approach promotes corporate social responsibility by distributing the workload in a fair way. As discussed in Juan et al. (2013a), and Armas, J. *et al.* (2016), multi-start randomized heuristics are relatively simple-to-implement algorithm that offer several benefit for solving real-life combinatorial optimization problems, including their flexibility, their efficiency, and their capacity to be run in parallel.

VII. CONCLUSION

The main goal of the current study is to design an appropriate method to solve the crew scheduling problem of Vietnam airlines. Typically, the crew pairing problems are solved separatedly in three stages: daily, weekly exceptions and transition as the U.S domestic problems (Barnhart *et al.*, 2003; Kasirzadeh, Saddoune and Soumis, 2015). The VNA flights schedule also contains two types of flights, which is daily flights, whereby the flight legs are identical for all days of the planning horizon, and weekly flights, in which the flights legs are repeated every week. However, after investigating the attributes of Vietnam Airlines and data of the flights schedule, together with the crew complement information, we developed and tested several integrated algorithms. Although, it was complicated to code, the research has shown very impressive results. With the crew pairing problem, we generated larger duties, leading to minimization of the number of duties and pairings, reduction of deadheads, and a decrease in the crew-related cost of meals and accommodation when crew stay away from their home base on duty. With the crew rostering problem, we not only covered all flight pairings, but also reduced the number of required crews. Furthermore, we met most of management and senior crews' preferences, as well as the work-pattern of the Non-Vietnamese crew members. We also provided flexible formulation to adjust the ratio of Vietnamese and non-Vietnamese crews.

We proposed two algorithms for the crew pairing problem, which have been developed from the duty period model by Vance *et al.* (1997b), based on the day-by-day rolling manner and combining heuristics and a mathematic model. The main advantage of the Vance *et al.* (1997b) formulation is that its LP relaxation provides a tighter bound on the optimal IP solution value than the traditional set partitioning formulation and we enhanced our model based on their formulation to take in this strength and the solutions reach near lower bounds. Consequently, the results are impressive both in their quality as well as the computational time.

In real-life practice, rosters are created and assigned to individual crew members of an airline in compliance with legal and contractual requirements. Crew rosters usually span during a month and are composed of work activities such as flights, training periods, rest periods, and imaginary shifts. When assigning crew rosters, the airline usually tries to meet the personal preferences of each crew while minimizing crew costs. Thus, a series of factors need to be considered during the rostering process.

Number of crew members: despite each airline has its own peculiarities when it comes to deciding how many crew members are required, most airline use a procedure proposed in the NBAA Management Guide (2014) to estimate this number. Since each schedule is unique (different types of aircrafts, airports, etc.), in order to decide about the right number of required crew members airlines need to take into account a series of additional factors. These include: (a) type of aircraft used; (b) block time; (c) number of hours flown per year; (d) number of simultaneous flights; (e) number of flights that require the crew to fly several consecutive days; (f) number of flights which require spending nights outside the base; (g) number of night flights; (h) number of flights that require more crew members than the minimum (e.g., due to crew re-allocation); and (i) the holidays and training policies of the company.

Airline's objectives: While designing crew schedule, an airline's manager might have different goals in mind, including: (a) minimizing the cost of assigning crews to their respective activities; (b) minimizing the cost of unassigned activities; (c) minimizing overtime payment; and (d) generating balanced rosters.With the crew rostering problem, we decomposed the problem into a two-phase method and designed a new mathematic formulation and three heuristic algorithms to achieve the specific objectives of VNA. To explore a different approach that avoids the necessity for manual tuning and that significantly reduces the time needed to get a feasible solution, we developed a multi-start approach - similar to the one presented in Juan et al. (2013a), Armas, J. et al. (2016) - based on a constraint programming heuristic in order to generate feasible solutions. The heuristics applied the techniques of randomization and a constraint programming method, which are domain, local consistency, bound consistency and constraint propagation, to generate many feasible rosters in a short computational time in the first phase. Afterwards, these feasible solutions are chosen by a solver to obtain the best solution, and then this feasible solution is enhanced via a local search process.

The results of this research provide suitable approaches to the crew scheduling problem of VNA and contribute to existing knowledge of the crew scheduling problem through several new algorithms, as well as an integrated method to approach the crew scheduling problem of airlines which have the special routing structure combination of the hub-and-spoke structrure with the point-to-point route system, while also running the two payment methods.

A main contribution with respect the existing literature is that our approach considers many real life constraints, rules and regulations that have never been analyzed in previous work. Our algorithm is able to obtain several different and feasible practical solutions within short computational time, thus saving the company hours of manual adjustments that are necessary to obtain a feasible roster. Furthermore, our approach can be used to generate balanced solutions in terms of workload distribution across crew members inside a month as well as for each crewmember throughout the year.

The main difference of our approach with respect to other previous approaches is the volume of realistic constraints we are considering. Notice that including all these constraints and decision rules in a formal optimization model can be a quite tedious error prone, and time-consuming task, which explains why some airlines use a two-stage approach: first they use a commercial software to solve a simplified version of the real-life model; then, they manually adjust the solution to meet all the required specifications.

The study suggests further research regarding the scheduling problem and constraint programming, to implement the algorithm and create global constraints suitable for solving the problem, in order to design scheduling software that will meet many more specific requirements of VNA.

REFERENCES

http://www.Vietnamairlines.com (Accessed: 5/10/2016).

Abdelghany, A. and Abdelghany, A.D. (2010) 'Crew Planning', in *Modeling Applications in the Airline Industry*. Farnham, GB: Routledge, p. 89.

Achour, H. *et al.* (2007) 'An exact solution approach for the preferential bidding system problem in the airline industry', *Transportation Science*, 41(3), pp. 354-365.

Agustín, A. *et al.* (2016) *Advances in Artificial Intelligence: 17th Conference of the Spanish Association for Artificial Intelligence, CAEPIA 2016, Salamanca, Spain, September 14-16, 2016. Proceedings*. Cham. Springer International Publishing. Available at: http://dx.doi.org/10.1007/978-3-319-44636-3_31.

Agustin, A., Juan, A. and Pardo, E.G. (2017) 'A variable neighborhood search approach for the crew pairing problem', *Electronic Notes in Discrete Mathematics*, 58, pp. 87-94.

Alefragis, P. *et al.* (1998) *European Conference on Parallel Processing*. Springer.

Alefragis, P. *et al.* (2000) 'Parallel integer optimization for crew scheduling', *Annals of Operations Research*, 99(1-4), pp. 141-166.

Anbil, R., Forrest, J.J. and Pulleyblank, W.R. (1998) 'Column generation and the airline crew pairing problem', *Documenta Mathematica*, 3(1), pp. 677-686.

Anbil, R. *et al.* (1991) 'Recent Advances in Crew-Pairing Optimization at American Airlines', *Interfaces*, 21(1), pp. 62-74.

Anbil, R., Tanga, R. and Johnson, E.L. (1992) 'A Global Approach to Crew-Pairing Optimization', *IBM Systems Journal*, 31(1), pp. 71-78.

Andersson, E. *et al.* (1998) 'Crew Pairing Optimization', in Yu, G. (ed.) *Operations research in the airline industry*. Boston: Kluwer Academic Publishers, pp. xvii, 473 p : ill., maps ; 24 cm.

Armas, J. *et al.* (2016) 'A multi-start randomized heuristic for real-life crew rostering problems in airlines with work-balancing goals', *Annals of Operations Research*, 258(2), pp. 825-848.

Aydemir-Karadag, A., Dengiz, B. and Bolat, A. (2011) 'Crew pairing optimization based on hybrid approaches', *Computers & Industrial Engineering*, 65(1), pp. 87-96.

Ball, M. and Roberts, A. (1985) 'A Graph Partitioning Approach to Airline Crew Scheduling', *Transportation Science*, 19(2), pp. 107-126.

Barahona, F. and Anbil, R. (2000) 'The volume algorithm: producing primal solutions with a subgradient method', *Mathematical Programming*, 87(3), pp. 385-400.

Barnhart, C. *et al.* (2003) 'Airline Crew Scheduling', in Hall, R.W. (ed.) *Handbook of Transportation Science* Secaucus, US: Kluwer Academic Publishers, pp. 517-560.

Barnhart, C. *et al.* (1994) 'A column-generation technique for the long-haul crew-assignment problem', in *Optimization in industry 2*. John Wiley & Sons, Inc., pp. 7-24.

Barnhart, C. *et al.* (1998) 'Branch-and-Price: Column Generation for Solving Huge Integer Programs', *Operations Research*, 46(3), pp. 316-329.

Bazargan, M. (2010) *Airline Operations and Scheduling*. Farnham, GB: Routledge.

Beasley, J.E. and Cao, B. (1996) 'A tree search algorithm for the crew scheduling problem', *European Journal of Operational Research*, 94(3), pp. 517-526.

Bixby, R.E. *et al.* (1995) 'Parallel mixed integer programming', *Rice University Center for Research on Parallel Computation Research Monograph CRPC-TR95554*.

Bixby, R.E. *et al.* (1992) 'Very Large-Scale Linear Programming: A Case Study in Combining Interior Point and Simplex Methods', *Operations Research*, 40(5), pp. 885-897.

Boubaker, K., Desaulniers, G. and Elhallaoui, I. (2010) 'Bidline scheduling with equity by heuristic dynamic constraint aggregation', *Transportation Research Part B: Methodological*, 44(1), pp. 50-61.

Brailsford, S.C., Potts, C.N. and Smith, B.M. (1999) 'Constraint satisfaction problems: Algorithms and applications', *European Journal of Operational Research*, 119(3), pp. 557-581.

Buhr, J. (1978) *Proceedings of the AGIFORS Symposium*.

Budd, L. and Ison, S. (2017) Air Transport Management - An international persppective. Routledge

Byrne, J. (1988) *AGIFORS PROCEEDINGS*.

CAAV (2015) *Viet nam Airlines Flight Operations Manual*

Campbell, K.W., Durfee, R.B. and Hines, G.S. (1997) 'FedEx Generates Bid Lines Using Simulated Annealing', *INFORMS Journal on Applied Analytics*, (2), pp. 1-16.

Christou, I.T. *et al.* (1999) 'A two-phase genetic algorithm for large-scale bidline-generation problems at Delta Air Lines', *Interfaces*, 29(5), pp. 51-65.

Cook, G.N. and Goodwin, J. (2008) 'Airline Networks: A comparison of Hub-and-Spoke and Point-to-Point Systems', *Journal of Aviation/ Aerospace Education & Research*, 17(2), pp. 51-60.

Day, P.R. and Ryan, D.M. (1997) 'Flight attendant rostering for short-haul airline operations', *Operations research*, 45(5), pp. 649-661.

Demirel, N.C. and Deveci, M. (2017) 'Novel search space updating heuristics-based genetic algorithm for optimizing medium-scale airline crew pairing problems', *International Journal of Computational Intelligence Systems*, 10(1), pp. 1082-1101.

Desaulniers, G. *et al.* (1997) 'Crew pairing at Air France', *European Journal of Operational Research*, 97(2), pp. 245-259.

Desaulniers, G. *et al.* (1998) 'A Unified Framework for Deterministic Time Constrained Vehicle Routing and Crew Scheduling Problems', *Fleet Management & Logistics*, pp. 57-93.

Desrochers, M. and Soumis, F. (1988) 'A Generalized Permanent Labelling Algorithm for the Shortest Path Problem with time windows', *Information Systems and Operational Research*, 26(3), pp. 191-212.

Desrosiers, J. *et al.* (1991) 'A breakthrough in airline crew scheduling', *The Canadian Transportation Research Forum*. p. 17.

Desrosiers, J. *et al.* (1995) *Time Constrained Routing and Scheduling*. Handbooks in Operations Research and Management Science.

Deveci, M. and Demirel, N.Ç. (2018a) 'Evolutionary algorithms for solving the airline crew pairing problem', *Computers & Industrial Engineering*, 115, pp. 389-406.

Deveci, M. and Demirel, N.Ç. (2018b) 'A survey of the literature on airline crew scheduling', *Engineering Applications of Artificial Intelligence*, 74, pp. 54-69.

Ehrgott, M. and Ryan, D. (2001) *Proceedings of the 35th annual conference of the operational research society of New Zealand*.

Ehrgott, M. and Ryan, D.M. (2002) 'Constructing robust crew schedules with bicriteria optimization', *Journal of Multi-Criteria Decision Analysis*, 11(3), pp. 139-150.

El Moudani, W. *et al.* (2001) *International Conference on Evolutionary Multi-Criterion Optimization*. Springer.

Elhallaoui, I. *et al.* (2005) 'Dynamic aggregation of set-partitioning constraints in column generation', *Operations Research*, 53(4), pp. 632-645.

Erdoğan, G. *et al.* (2015) 'Solving a large-scale crew pairing problem', *Journal of the Operational Research Society*, 66(10), pp. 1742-1754.

Etschmaier, M.M. and Mathaisel, D.F.X. (1985) 'Airline Scheduling: An Overview', *Transportation Science*, 19(2), pp. 127-139.

Fahle, T. *et al.* (2002) 'Constraint programming based column generation for crew assignment', *Journal of Heuristics*, 8(1), pp. 59-81.

Forrest, J.J. (1989) 'Mathematical programming with a library of optimization subroutines', *presented at the ORSA/TIMS Joint National Meeting, New York*.

Gamache, M. and Soumis, F. (1998) 'A method for optimally solving the rostering problem', in *Operations Research in the Airline Industry*. Springer, pp. 124-157.

Gamache, M. *et al.* (1999) 'A Column Generation Approach for Large-Scale Aircrew Rostering Problems', *Operations Research*, 47(2), pp. 247-263.

Gamache, M. *et al.* (1998) 'The preferential bidding system at Air Canada', *Transportation Science*, 32(3), pp. 246-255.

Gershkoff, I. (1989) 'Optimizing Flight Crew Schedules', *Interfaces*, 19(4), pp. 29-43.

Giafferri, C., Hamon, J. and Lengline, J. (1982) *AGIFORS PROCEEDINGS*.

Glanert, W. (1984) *PROCEEDINGS OF THE AGIFORS SYMPOSIUM*. International Federation OF Operational Research.

Gopalakrishnan, B. and Johnson, E.L. (2005) 'Airline Crew Scheduling: State-of-the-Art', *Annals of Operations Research*, 140(1-4), pp. 305-337.

Goumopoulos, C., Housos, E. and Liljenzin, O. (1997) *European Parallel Virtual Machine/Message Passing Interface Users' Group Meeting*. Springer.

Graves, G.W. *et al.* (1993a) 'Flight crew scheduling', *Management Science*, 39(6), pp. 736-745.

Graves, G.W. *et al.* (1993b) 'Flight Crew Scheduling', *Management Science*, (6), p. 736.

Hall, R.W. (2002) *Handbook of Transportation Science*. Secaucus, UNITED STATES: Kluwer Academic Publishers.

Held, M. and Karp, R.M. (1970) 'The Traveling-Salesman Problem and Minimum Spanning Trees', *Operations Research*, 18 (6), pp. 1138-1162.

Held, M. and Karp, R.M. (1971) 'The traveling-salesman problem and minimum spanning trees: Part II', *Mathematical Programming*, 1(1), pp. 6-25.

Held, M., Wolfe, P. and Crowder, H.P. (1974) '*Validation of subgradient optimization*', *Mathematical Programming*, 6(1), pp. 62-88.

Hjorring, C.A. and Hansen, J. (1999) *Proceedings of the 34th Annual Conference of the Operational Research Society of New Zealand*.

Hoffman, K.L. and Padberg, M. (1993) 'Solving Airline Crew Scheduling Problems by Branch-and-Cut', *Management Science*, 39(6), pp. 657-682.

Housos, E. and Elmroth, T. (1997) 'Automatic Optimization of Subproblems in Scheduling Airline Crews', *Interfaces*, (5), pp. 68-77.

Hu, J. and Johnson, E.L. (1999) 'Computational results with a primal-dual subproblem simplex method', *Operations Research Letters*, 25(4), pp. 149-157.

Iijima, Y. *et al.* (2013) *2013 IEEE International Conference on Industrial Engineering and Engineering Management* Bangkok, Thailand. IEEE. Available at: http://search.ebscohost.com/login.aspx?direct=true&db=edseee&AN=edseee.6962386&site=eds-live.

Jarrah, A.I.Z. and Diamond, J.T. (1997) 'The Problem of Generating Crew Bidlines', *INFORMS Journal on Applied Analytics*, 27(4), pp. 49-64.

Juan, A., Faulin, J., Ferrer, A., Lourenço, H.,&Barrios, B. (2013a). MIRHA: multi-start biased randomization of heuristics with adaptive local search for solving non-smooth routing problems. TOP, 21, 109–132.

Johnson, E.L. (1989), 19890101. Springer, Berlin. Available at: http://search.ebscohost.com/login.aspx?direct=true&db=msn&AN=MR996643&site=eds-live

http://www.ams.org/mathscinet/MRAuthorID/95070

http://www.ams.org/mathscinet-getitem?mr=996643.

Junker, U. *et al.* (1999) *International Conference on Principles and Practice of Constraint Programming*. Springer.

Kasirzadeh, A., Saddoune, M. and Soumis, F. (2015) 'Airline crew scheduling: models, algorithms, and data sets', *EURO Journal on Transportation and Logistics*, pp. 1-27.

Klabjan, D. (2001) *Next generation airline crew scheduling*.

Klabjan, D., Johnson, E.L. and Nemhauser, G.L. (2000) 'A parallel primal–dual simplex algorithm', *Operations Research Letters*, 27(2), pp. 47-55.

Klabjan, D. *et al.* (2001) 'Solving large airline crew scheduling problems: Random pairing generation and strong branching', *Computational Optimization and Applications*, 20(1), pp. 73-91.

Klabjan, D. and Schwan, K. (2001) *PPSC*.

Kohl, N. and Karisch, S.E. (2000) *Proceedings of the 40th Annual AGIFORS Symposium*.

Kohl, N. and Karisch, S.E. (2004) 'Airline Crew Rostering: Problem Types, Modeling, and Optimization', *Annals of Operations Research*, 127(1), pp. 223-257.

König, J. and Strauss, C. (2000a) 'Rostering-integrated services and crew efficiency', *Information Technology & Tourism*, 3(1), pp. 27-39.

König, J. and Strauss, C. (2000b) 'Supplements in airline cabin service', in *Information and Communication Technologies in Tourism 2000*. Springer, pp. 365-374.

Ladányi, L., Ralphs, T.K. and Trotter, L.E. (2001) *Branch, cut, and price: Sequential and parallel*. Springer Verlag. Lecture Notes in Computer Science.

Lavoie, S., Minoux, M. and Odier, E. (1988) 'A new approach for crew pairing problems by column generation with an application to air transportation', *European Journal of Operational Research*, 35(1), pp. 45-58.

Linderoth, J.T. and Savelsbergh, M.W. (1999) 'A computational study of search strategies for mixed integer programming', *INFORMS Journal on Computing*, 11(2), pp. 173-187.

Lucic, P. and Teodorovic, D. (2007) 'Metaheuristics approach to the aircrew rostering problem', Ann Oper Res, 155, pp. 311 - 338

Lustig, I.J. and Puget, J.-F. (1999) 'Program!= program: Constraint programming and its relationship to mathematical programming', *Interfaces*, 31(6), pp. 29-53.

Maenhout, B. and Vanhoucke, M. (2010) 'A hybrid scatter search heuristic for personalized crew rostering in the airline industry', *European Journal of Operational Research*, 206(1), pp. 155-167.

Makri, A. and Klabjan, D. (2001) *Efficient column generation techniques for airline crew scheduling*.

Marchettini, F. (1980) *1980 AGIFORS symposium proceedings*.

Marsten, R. (1994) *XV Mathematical Programming Symposium. Presentation*.

Maximilian , E. and Dennis , M. (1985) 'Airline Scheduling: An Overview', *Transportation Science*, 19(2), p. 127.

Minoux, M. (1984) *International Federation of Operational Reseach*.

Moore, R., Evans, J. and Noo, H. (1978) *AGIFORS symposium proceedings*.

Nicoletti, B. (1975) 'Automatic crew rostering', *Transportation Science*, 9(1), pp. 33-42.

Quesnel, F., Desaulniers, G. and Soumis, F. (2016) 'A new heuristic branching scheme for the crew pairing problem with base constraints', *Computers & Operations Research*, 80, pp. 159-172.

Rasmussen, M.S. *et al.* (2011) 'A subsequence generation approach for the airline crew pairing problem', *51st AGIFORS Annual Proceedings - Annual Symposium and Study Group Meeting, AGIFORS 2011*. 2011 / 01 / 01 /. pp. 584-605. Available at: http://search.ebscohost.com/login.aspx?direct=true&db=edselc&AN=edselc.2-52.0-84879475085&site=eds-live.

Reynolds-Feighan, A. (2007) 'Carrier network structures and the Spatial distribution of Air traffic in the European Air Transport market, 1996 - 2006', *Spatial Economic Analysis*, Vol 2, No 3.

Reynolds-Feighan, A. (2009) 'Characterisation of airline networks: A North American and European comparison', *Journal of Air Transport Management*, 16, pp.109-120

Rubin, J. (1971) 'Airline Crew Scheduling - The Non-Mathematical Problem', *IBM Technical Report: 320.3006*.

Rubin, J. (1973) 'A Technique for the Solution of Massive Set Covering Problems, with Application to Airline Crew Scheduling', *Transportation Science,* (1), pp. 34-48.

Ryan, D. and Falkner, J. (1988) 'On the integer properties of scheduling set partitioning models', *European journal of operational research*, 35(3), pp. 442-456.

Ryan, D.M. (1992a) 'The solution of massive generalized set partitioning problems in aircrew rostering', *Journal of the operational research society*, 43(5), pp. 459-467.

Ryan, D.M. (1992b) 'The Solution of Massive Generalized Set Partitioning Problems in Aircrew Rostering', (5), p. 459.

Ryan, D.M. and Foster, B.A. (1981) 'An integer programming approach to scheduling', *Computer scheduling of public transport urban passenger vehicle and crew scheduling*, pp. 269-280.

Saddoune, M. *et al.* (2011a) 'Integrated Airline Crew Pairing and Crew Assignment by Dynamic Constraint Aggregation', *Transportation Science*, 46(1), pp. 39-55.

Saddoune, M. *et al.* (2011b) 'Integrated airline crew scheduling: A bi-dynamic constraint aggregation method using neighborhoods', *European Journal of Operational Research*, 212(3), pp. 445-454.

Sanders, P., Takkula, T. and Wedelin, D. (1999) *International Conference on High-Performance Computing and Networking*. Springer.

Sarra, D. (1988) *1988 AGIFORS symposium proceedings*.

Sellmann, M. *et al.* (2002) 'Crew assignment via constraint programming: integrating column generation and heuristic tree search', *Annals of Operations Research*, 115(1-4), pp. 207-225.

Subramanian, S. and Sherali, H.D. (2008) 'An effective deflected subgradient optimization scheme for implementing column generation for large-scale airline crew scheduling problems', *INFORMS Journal on Computing*, 20(4), pp. 565-578.

Vance, P.H. *et al.* (1997a) 'A heuristic branch-and-price approach for the airline crew pairing problem'.

Vance, P.H. *et al.* (1997b) 'Airline Crew Scheduling: A New Formulation and Decomposition Algorithm', (2), pp. 188-200.

Wedelin, D. (1995) 'An algorithm for large scale 0-1 integer programming with application to airline crew scheduling', *Annals of Operations Research*, 57(1-4), pp. 283-301.

Weir, J. and Johnson, E. (2004) 'A Three-Phase Approach to Solving the Bidline Problem', *Annals of Operations Research*, (1), pp. 283-308.

Zeren, B. and Özkol, I. (2016) 'A novel column generation strategy for large scale airline crew pairing problems', *Expert Systems with Applications*, 55, pp. 133-144.