

Routing of Airplanes to Two Runways: Monotonicity of Optimal Controls

Corrected Version

N. Bäuerle*, O. Engelhardt-Funke, M. Kolonko †

April 18, 2005

Abstract : We consider the problem of routing incoming airplanes to two runways of an airport. Due to air turbulence, the necessary separation time between two successive landing operations depends on the types of the airplanes. When viewed as a queuing problem, this means that we have dependent service times. The aim is to minimize waiting times of aircrafts.

We consider here a model where arrivals form a stochastic process and where the decision maker does not know anything about future arrivals. We formulate this as a problem of stochastic dynamic programming and investigate monotonicity of optimal routing strategies with respect e.g. to the workload of the runways.

We show that an optimal strategy is monotone (i.e. of switching type) only in a restricted case where decisions depend on the state of the runways only and not on the type of the arriving aircraft. Surprisingly, in the more realistic case where this type is also known to the decision maker, monotonicity need not hold.

*Inst. für Mathematische Stochastik, Universität Hannover, Germany

†Institut für Mathematik, Technical University Clausthal, Erzstr. 1, D-38678 Clausthal-Zellerfeld, Germany,
email: kolonko@math.tu-clausthal.de

1 Introduction

In modern air traffic the efficient use of the available runway capacity is of growing importance at least at major airports. Airplanes are often queued when waiting for a free runway to land on. While queuing and during the landing operations, the airplanes have to keep a large enough distance between each other to avoid air turbulence from aircrafts flying ahead. These separation times depend on the weight (more generally: types) of the airplanes involved. Typically, the necessary separation between a light aircraft trailing a heavy one will be larger than between the same types if the light one is flying in front. Hence, one can expect to increase the throughput of the runways by an efficient routing of arriving airplanes to runways.

The general problem is quite involved as a lot of side constraints and airport specific rules have to be considered. We restrict ourselves to a particular scenario, which may be viewed as a step towards more realistic models that we cannot analyse completely at present.

First, we assume that the arrival times of the aircrafts form a stochastic renewal process. This is in accordance with e.g. [4] and [8]. There it is assumed that the scheduled arrival times are highly disturbed (e.g. by varying flight conditions, delays on connecting flights or technical problems) such that in practice the arrival times may well be approximated by a Poisson process.

We further assume that the routing decision has to be made at the arrival instance of the airplane at the airport (or at a certain threshold). The decision maker may use information on the state of the runways and the type of the presently arriving aircraft but does not know anything about future arrivals. Once an aircraft has been assigned to a runway it must stay in the queue of that runway. The queues are served on a first come first served basis. The aim is to find a rule that assigns an incoming aircraft to a runway given the state of the system and the type of arriving aircraft such that the long term expected waiting times are minimized.

We formulate a stochastic dynamic programming model for this problem with the total expected discounted waiting time of the aircrafts as target function. We investigate monotonicity properties of optimal routing strategies (policies) in this model. Monotonicity here means e.g. that when the observed workload u on runway I increases while everything else is kept fixed then an assignment of an arriving plane to runway I will only be made for u up to a certain level l . For $u > l$ the aircraft will then be assigned to runway II . Such policies are also called 'switching policies' as they are completely determined by switching levels l . This will be made more precise by defining a partial order on the state space of the dynamic programming model and by proving monotonicity of optimal routing policies with respect to this ordering.

Here are our main findings: optimal policies are monotone only if we restrict ourselves to decisions that depend solely on the state of the runway and not on the type of the presently arriving aircraft. This result also yields conditions under which a simple Join-the-Least-Load strategy (JLL) is optimal. We give a rough bound on the error that is made when using JLL in the general case where it is not optimal. Surprisingly, monotonicity with respect to workloads need not hold in the more realistic model where decisions may depend on the state of the runways as well as on the type of the arriving airplane. We show by a counter example based on realistic data that in this case it may be optimal to route to runway II for a small workload u on runway I and to route to runway I for a larger workload u' . This somewhat unexpected result may be explained from the fact that in the first restricted case we use a cost function (see definition (3.5)) where the dependency of waiting times from the next arriving aircraft is 'averaged out' whereas in the second case we use as cost function the actual waiting time (see (6.2)) leading to a more sensitive criterion.

The proof of the main monotonicity result is based on the approach in [1]. Nevertheless, our model does not fit into the framework of [1]. We had to change the conditions at a minute

but decisive point and to prove that the main assertions of [1] still hold under the modified assumptions.

Optimal routing of airplanes has been treated e.g. in [3] where also a survey over different approaches is given. Deterministic models as in [3] often assume that a set of airplanes to be scheduled and routed is given. This allows to take into account more than one arrival and optimal schedules for a single set of aircrafts are obtained by mixed linear optimization. The stochastic queuing models in [4] and [8] do not take into account the dependency of the separation times from the two types involved. In [2] M/SM/1 queuing models are used to deal with this dependency. In a somewhat complementary way to the present paper, strategies that use only the type of the arriving aircrafts are considered. There, it is also shown that neglecting the dependencies may lead to a strongly biased estimation of waiting times under heavy traffic. The general literature on the routing of parallel queues (see e.g. [13]) seems to be restricted to the case of independent service times only. Queues with dependent service times are treated e.g. in [10], but without any reference to routing.

The paper is organized as follows. In Section 2, we collect a few results from stochastic dynamic programming, in particular about the optimality equations and value iteration. In Section 3 we use the dynamic programming framework to formalism the restricted model, where decisions depend on the state of the runways alone. Here, we also define the load of a runway and the ordering of the state space and give the main monotonicity results. We start with this restricted model as it takes most of the paper to prove these results. The rather technical proof is given in Section 7. A number of Corollaries about Join-the-Least-Load policies are collected in Section 5. The model of Section 3 is then enlarged in Section 6 to include the type of arriving airplane and a counter-example shows that the monotonicity property does no longer hold. In the Conclusion we address the potential benefit of our results for the practical solution of the problem and topics of our future research.

2 A Dynamic Programming Model

We shall first collect a few definitions and results from Markovian dynamic programming (see e.g. [5], [11] or [12] for a general discussion of dynamic programming). Here, we present only a simple type of dynamic programming model which is sufficient to cover the aircraft routing problem formulated in the next Section.

The model describes a system that is observed at discrete points of time (stages) $n = 1, 2, \dots$. At each stage the system is observed to be in a *state* $s \in S$. Then an *action* $a \in A$ is taken and the system moves to a new state $s' = g(s, a, z) \in S$ to be observed at the next stage. Here, $z \in \mathcal{Z}$ is an *external event* (disturbance) and $g : S \times A \times \mathcal{Z} \rightarrow S$ is the *state transition function*. The state-action pair (s, a) causes costs $c(s, a)$, where $c : S \times A \rightarrow \mathbb{R}_+$ is the *cost function*. Let $(Z_n)_{n \geq 1}$ be an i.i.d. sequence of external random variables with values in \mathcal{Z} and with a known distribution.

We shall leave aside questions of measurability and assume in particular that the action space A is finite.

If we observe the system for a finite number N of stages (*finite horizon case*), actions are chosen according to a policy $\delta = (f_N, \dots, f_1)$, where $f_n : S \rightarrow A$ is the decision rule for the $N - n + 1$ -th stage. The reverse numbering of the decision rules simplifies the description of the induction below. Given a starting state $X_1 = s$ from S and a policy δ , the sequence of states X_2, \dots, X_N is then defined recursively by

$$X_{n+1} := g(X_n, f_{N-n+1}(X_n), Z_n), \quad 1 \leq n \leq N - 1. \quad (2.1)$$

With $A_n := f_{N-n+1}(X_n)$, we obtain for a measurable set $B \subset S$ the simple Markovian dependency

$$\mathbf{P}[X_{n+1} \in B \mid X_1, \dots, X_n = s, A_1, \dots, A_n = a] = \mathbf{P}(g(s, a, Z_1) \in B). \quad (2.2)$$

The total expected discounted costs for policy $\delta = (f_N, \dots, f_1)$ and starting state $s \in S$ are given by

$$V_{N,\delta}(s) := \mathbf{E} \left[\sum_{k=1}^N \beta^{k-1} c(X_k, A_k) \mid X_1 = s \right] \quad (2.3)$$

where $\beta \in (0, 1]$ is a given *discount factor*. As our cost function is non-negative and the action space is finite, it is well known (see e.g. [12]) that an optimal policy exists, i.e. a policy δ^* , with

$$V_N(s) := V_{N,\delta^*}(s) = \min_{\delta} V_{N,\delta}(s) \quad \text{for all } s \in S. \quad (2.4)$$

In fact the min may be taken over a much larger class of policies than defined above. For (measurable) functions $w : S \rightarrow \mathbb{R}_+$ we define the one-stage cost operator L by

$$\begin{aligned} L_a w(s) &:= c(s, a) + \beta \mathbf{E}[w(X_2) \mid X_1 = s, A_1 = a] \\ &= c(s, a) + \beta \mathbf{E} w(g(s, a, Z_1)). \end{aligned} \quad (2.5)$$

Problem (2.4) can be solved recursively with the help of the so-called optimality equation for dynamic programs which in the finite horizon case reads

$$V_n(s) = \min_{a \in A} L_a V_{n-1}(s) \quad \text{for all } s \in S, 1 \leq n \leq N. \quad (2.6)$$

We assume throughout that $V_0 \equiv 0$. A policy $\delta^* = (f_N^*, \dots, f_1^*)$ is optimal iff $f_n^*(s)$ selects a minimizing action in (2.6) for $s \in S, 1 \leq n \leq N$ (see e.g. [12], Cor. 6.2).

In the *infinite horizon case* we only consider stationary policies $\delta = (f_n)_{n \geq 1}$ with $f_n \equiv f, n \geq 1$ and we write $\delta = f$ for short. With $A_k = f(X_k), k \geq 1$ we define the total expected discounted costs by

$$V_{\delta}(s) := \mathbf{E} \left[\sum_{k=1}^{\infty} \beta^{k-1} c(X_k, A_k) \mid X_1 = s \right]. \quad (2.7)$$

Again, δ^* is called optimal if

$$V(s) := V_{\delta^*}(s) = \min_{\delta} V_{\delta}(s), \quad s \in S. \quad (2.8)$$

It is shown in [12] that there exists an optimal stationary policy which can be obtained from the infinite horizon optimality equation

$$V(s) = \min_{a \in A} L_a V(s), \quad s \in S. \quad (2.9)$$

f^* forms an optimal stationary policy iff $f^*(\cdot)$ is a minimizer of (2.9). Moreover, value iteration holds, i.e.

$$\lim_{N \rightarrow \infty} V_N(s) = V(s), \quad s \in S. \quad (2.10)$$

Below, we use (2.6) to derive properties for the finite horizon case with arbitrary horizon N using inductive arguments. These properties then carry over to the infinite horizon case using (2.10).

3 Optimal Aircraft Routing as Dynamic Programming Problem

We shall now specify the elements of the dynamic programming model such that the problem of optimal aircraft routing can be dealt with.

We start with the external events Z_n . Let $S_n, n \geq 1$, denote the arrival times of airplanes at the airport. We assume that the **interarrival times** $T_n := S_n - S_{n-1} \geq 0$ for $n \geq 1$, with $S_0 := 0$ are i.i.d. , i.e. the arrivals form a renewal process with some distribution F . Let J be the finite set of possible **types** of airplanes and denote by J_n the type of the n -th arriving plane. We assume that the $J_n, n \geq 1$, are i.i.d. and independent of the arrival process with

$$p_i := \mathbf{P}(J_n = i), \quad i \in J.$$

Then

$$Z_n := (J_n, T_{n+1}), \quad n \geq 1 \tag{3.1}$$

taking on values in $\mathcal{Z} := J \times \mathbb{R}_+$, describes the type of the n -th arriving aircraft and the following interarrival time. $(Z_n)_{n \geq 1}$ is an i.i.d. sequence, it is the external source of randomness in our model.

We assume that if an aircraft of type j is to land immediately behind an aircraft of type i on the same runway, then there must be a safety distance given as **separation time** $b(i, j) \geq 0, i, j \in J$. This means that the beginning of the two landing operations (touch-downs) must be $b(i, j)$ time units apart.

In our model, routing decisions have to be made at the arrival instances S_n of airplanes which define our decision epochs. Let $A := \{I, II\}$ where action $a = I$ ($a = II$) means to route the present airplane to runway I (II). Once an airplane has been routed to a queue it has to wait there for its service. The queues work on a first-come-first-served basis.

To define the 'state' of our system we first consider a single runway, say runway I . Of course, things are completely analogous on runway II .

Let ζ_n^I denote the type of airplane that is at the tail of the queue of runway I immediately before the n -th arrival takes place. If the queue I is empty at that time then ζ_n^I is the type of the last airplane that landed on I . With $\zeta_1^I := i_0$ (an arbitrary type), the formal definition of ζ_n^I is

$$\zeta_{n+1}^I := \begin{cases} J_n & \text{if } Z_n = (J_n, T_{n+1}) \text{ and } A_n = I \\ \zeta_n^I & \text{if } A_n = II \end{cases}, \quad n \geq 1. \tag{3.2}$$

For $n \geq 2$, ζ_n^I is the type of plane that the n -th plane will see as its predecessor if it is routed to runway I . Note that the index n counts arrivals to the airport, not to the particular runway.

Let U_n^I denote the **workload** on runway I immediately before the arrival of the n -th airplane. U_n^I is the time that the last plane at the tail of the queue has to wait until the beginning of its landing operation. If the queue is empty at that time then $U_n^I < 0$ denotes the time that has passed since the last plane began its landing on runway I .

Let $b^* := \max_{i,j \in J} b(i, j)$. With $U_1^I := -b^*$ the formal definition of the workload is given by

$$U_{n+1}^I := \begin{cases} [U_n^I + b(\zeta_n^I, J_n)]^+ - T_{n+1} & \text{if } Z_n = (J_n, T_{n+1}) \text{ and } A_n = I \\ U_n^I - T_{n+1} & \text{if } Z_n = (J_n, T_{n+1}) \text{ and } A_n = II \end{cases}, \quad n \geq 1. \tag{3.3}$$

Note that

$$[U_n^I + b(\zeta_n^I, J_n)]^+$$

is the waiting time of the n -th plane if it is routed to I . Here, $b(\zeta_n^I, J_n)$ can be regarded as the service time of this plane. The definition of ζ_1^I and U_1^I guarantees that the first aircrafts on each

runway have waiting time 0. In Fig. 3.1 the load on runway a is shown at the arrival instance of the n -th aircraft of type j_4 when two aircrafts of types j_2, j_3 are already waiting. The black triangles indicate the time at which the landing operation of the aircraft begins. Type j_1 has already landed, but j_2 still has to wait.

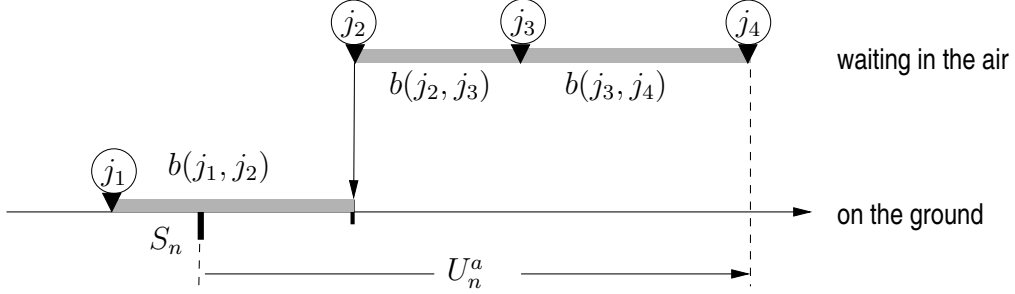


Figure 3.1: The load on runway a when aircrafts of types j_1, \dots, j_4 are present

The state of runway I at the arrival instance of the n -th plane is the pair (ζ_n^I, U_n^I) taking on values in $J \times \mathbb{R}$. Note that it is possible to bound the workload from below, as we need not keep track of negative loads that are larger than $b^* = \max_{i,j \in J} b(i, j)$.

The state of the second runway is defined in a completely analogous way and the **state of the system** at the n -th arrival instance is then defined as

$$X_n := (\zeta_n^I, U_n^I, \zeta_n^{II}, U_n^{II})$$

taking on values in the **state space** $S := J \times \mathbb{R} \times J \times \mathbb{R}$.

Note that in this model the presently arriving type J_n is not part of the state but part of the external event $Z_n = (J_n, T_{n+1})$ that drives the system. This follows from our particular restriction that the type J_n is not known to the decision maker. Below, in the case where J_n is known we shall use (J_{n+1}, T_{n+1}) as the external event driving the system (see Section 6).

From (3.2) and (3.3) we now see how the state transition function $g : S \times A \times \mathcal{Z} \rightarrow S$ must be defined. For $s = (i, u, j, v) \in S, a \in A$ and $z = (l, t) \in \mathcal{Z}$ let

$$g((i, u, j, v), a, (l, t)) := \begin{cases} (l, [u + b(i, l)]^+ - t, j, v - t) & \text{if } a = I \\ (i, u - t, l, [v + b(j, l)]^+ - t) & \text{if } a = II. \end{cases} \quad (3.4)$$

As **one-stage cost function** $c : S \times A \rightarrow \mathbb{R}_+$ we define for $s = (i, u, j, v) \in S$

$$c(s, a) := \begin{cases} \sum_{j' \in J} p_{j'} \cdot [u + b(i, j')]^+ & \text{if } a = I \\ \sum_{j' \in J} p_{j'} \cdot [v + b(j, j')]^+ & \text{if } a = II. \end{cases} \quad (3.5)$$

If we denote by

$$W_n := [U_n^{A_n} + b(\zeta_n^{A_n}, J_n)]^+$$

the waiting time of the n -th airplane, then we have

$$c(s, a) = \mathbf{E}[W_n \mid X_n = s, A_n = a].$$

Note that $c((i, u, j, v), a)$ depends only on the state of runway a , e.g. on (i, u) for $a = I$. The total discounted expected costs now read

$$V_{N, \delta}(s) = \mathbf{E} \left[\sum_{k=1}^N \beta^{k-1} c(X_k, A_k) \mid X_1 = s \right] = \mathbf{E} \left[\sum_{k=1}^N \beta^{k-1} [U_n^{A_n} + b(\zeta_n^{A_n}, J_n)]^+ \mid X_1 = s \right] \quad (3.6)$$

and similar for the infinite horizon. Starting in an empty system means to have $X_1 = s_0 := (i_0, -b^*, i_0, -b^*)$. As we have only two actions, the optimality equations (2.6) and (2.9) have a particularly simple structure:

$$V_n(s) = \min\{L_I V_{n-1}(s), L_{II} V_{n-1}(s)\} \quad \text{and} \quad V(s) = \min\{L_I V(s), L_{II} V(s)\}. \quad (3.7)$$

Also, (2.5) becomes

$$\begin{aligned} L_I w(s) &= c(s, I) + \beta \mathbf{E} w(g(s, I, Z_1)) \\ &= \sum_{l \in J} p_l \left([u + b(i, l)]^+ + \beta \int_0^\infty w((l, [u + b(i, l)]^+ - t, j, v - t)) F(dt) \right) \end{aligned}$$

where F is the distribution of the interarrival times.

4 Monotonicity Properties of Optimal Policies

In this Section we show that optimal routing policies are monotone with respect to a particular (partial) ordering of the state space. As usual, 'increasing' and 'decreasing' are used in the non-strict sense.

Let us first define a partial ordering on the set of types J . For $i, j \in J$ define

$$i \prec_J j \quad : \iff \quad b(i, k) \leq b(j, k) \quad \text{for all } k \in J. \quad (4.1)$$

In the aircraft setting, $i \prec_J j$ could indicate that j is a heavier plane that requires more separation than i . We do not make any assumptions on the ordering of the separation times, hence, in the extreme case, it may happen, that $i \prec_J j$ only holds for $i = j$.

Now let $s = (i, u, j, v), \bar{s} = (\bar{i}, \bar{u}, \bar{j}, \bar{v}) \in S$, then we define

$$s \prec \bar{s} \quad : \iff \quad i \prec_J \bar{i}, u \leq \bar{u}, j \succ_J \bar{j}, v \geq \bar{v} \quad (4.2)$$

If $s \prec \bar{s}$ holds, then in state \bar{s} the load on runway I is at least as high as in state s and at its tail there is an aircraft that requires at least as much separation time as in s . For runway II the opposite relation holds. Hence, the balance of the two queues is more favorable for I in state s than it is in \bar{s} .

A function $f : S \rightarrow M$, where (M, \leq) is a (partially) ordered set, is called **s-increasing** if $s \prec \bar{s}$ implies $f(s) \leq f(\bar{s})$. For $f_n : S \rightarrow A$, we define the order ' \leq' ' on A by $I \leq II$. Note that $f : S \rightarrow \mathbb{R}$ is s-increasing if and only if

$$\begin{aligned} u, i &\mapsto f(i, u, j, v) \quad \text{are increasing for fixed } j, v \text{ and} \\ v, j &\mapsto f(i, u, j, v) \quad \text{are decreasing for fixed } i, u. \end{aligned} \quad (4.3)$$

Here, monotonicity in i and j is defined w.r.t. the ordering (4.1) whereas monotonicity in u and v w.r.t. the usual ordering on \mathbb{R} .

The following Theorem is the main result of this Section. It shows that optimal policies are monotone w.r.t. to the ordering (4.2) of the state space. Some implications are described below.

4.1 Theorem

a) (finite horizon case)

For any horizon N there is an optimal policy $\delta = (f_N, \dots, f_1)$ such that $s \mapsto f_n(s)$ is s-increasing for $n = 1, \dots, N$.

b) (infinite horizon case)

There is an optimal stationary policy $\delta = f$ such that $s \mapsto f(s)$ is s-increasing.

In fact any optimal policy is monotone in this sense if we agree to choose the smaller action I in cases where both actions are minimizing the optimality equations. Theorem 4.1 states that if it is optimal to route the next airplane to runway II in a state $s = (i, u, j, v)$, then we should do the same in all states \bar{s} with $s \prec \bar{s}$.

Note that f_n is s-increasing if and only if there exists a 'level' function $l_n : J^2 \times \mathbb{R} \rightarrow \mathbb{R}$ with $i \mapsto l_n(i, j, v)$ decreasing and $j, v \mapsto l_n(i, j, v)$ increasing such that for $s = (i, u, j, v) \in S$

$$f_n(s) = \begin{cases} I & \text{if } u < l_n(i, j, v) \\ II & \text{if } u > l_n(i, j, v) \end{cases} \quad (4.4)$$

(for the 'only-if'-part put $l_n(i, j, v) := \inf\{u \mid f_n(i, u, j, v) = II\}$). In this sense, an s-increasing policy is a 'switching policy'.

The proof of Theorem 4.1 is quite lengthy and only its main steps are given below, the remainder is split into several technical Lemmata given in Section 7 below.

Proof : (of Theorem 4.1)

a) For the finite horizon case, define

$$\Delta_n(s) := L_I V_n(s) - L_{II} V_n(s), \quad n \geq 1.$$

From the optimality equation (2.6) it follows that $(f_N, f_{N-1}, \dots, f_1)$ with

$$f_n(s) := \begin{cases} I, & \text{if } \Delta_{n-1}(s) \leq 0 \\ II, & \text{if } \Delta_{n-1}(s) > 0 \end{cases}, \quad 1 \leq n \leq N,$$

forms an optimal policy for the N -stage problem. In Lemma 7.3 below it is shown that $s \mapsto \Delta_n(s)$ is s-increasing, hence it follows that $s \mapsto f_n(s)$ is s-increasing as well.

b) For the infinite horizon, we conclude from the value iteration (2.10) that if $s \mapsto \Delta_n(s)$ is s-increasing for all $n \geq 1$ then

$$\Delta(s) := L_I V(s) - L_{II} V(s) = \lim_{n \rightarrow \infty} \Delta_n(s)$$

is an s-increasing function, too. Here, $\lim_{n \rightarrow \infty} L_I V_n = L_I V$ is shown in [12], 4.4.1. Now b) follows as in a). ■

5 When is Join-the-Least-Load (JLL) optimal?

A natural simple policy would be to route the next arriving airplane to the runway with the least load, i.e. for $s = (i, u, j, v)$ to decide according to $\bar{\delta} = (\bar{f}_N, \dots, \bar{f}_1)$ where

$$\bar{f}_n(s) := \bar{f}(s) := \begin{cases} I & \text{if } u \leq v \\ II & \text{if } u > v \end{cases}. \quad (5.1)$$

However, due to the structure of the service times $b(i, j)$ one cannot expect this policy to be optimal in general. We now examine some special cases where JLL is optimal. First, we state a simple consequence of the symmetry of the two runways.

5.1 Lemma

Let $s := (i, u, j, v) \in S$ and $\bar{s} := (j, v, i, u)$, then we have

$$V_n(s) = V_n(\bar{s}) \quad \text{for all } n \geq 1 \quad \text{and} \quad V(s) = V(\bar{s}).$$

For finite and infinite horizon it holds that action I is optimal in state s iff action II is optimal in state \bar{s}

Proof : From (3.5) and (3.4) we see that $c(s, I) = c(\bar{s}, II)$. Starting with $V_0 \equiv 0$ we obtain inductively

$$\begin{aligned} L_I V_{n-1}(s) &= c(s, I) + \beta \mathbf{E} V_{n-1}(g(s, I, Z_1)) \\ &= c(\bar{s}, II) + \beta \mathbf{E} V_{n-1}(g(\bar{s}, II, Z_1)) = L_{II} V_{n-1}(\bar{s}) \end{aligned}$$

hence

$$V_n(s) = \max\{L_I V_{n-1}(s), L_{II} V_{n-1}(s)\} = \max\{L_{II} V_{n-1}(\bar{s}), L_I V_{n-1}(\bar{s})\} = V_n(\bar{s})$$

and

$$L_I V_{n-1}(s) \leq L_{II} V_{n-1}(s) \iff L_{II} V_{n-1}(\bar{s}) \leq L_I V_{n-1}(\bar{s}).$$

The corresponding results for the infinite horizon follow from (2.10) and (2.9). ■

The following Theorem is a direct consequence of Theorem 4.1 and the symmetry of the runways. It shows in particular that it is optimal to use JLL in states where both runways have identical types waiting at their tails. Note that we use the ordering defined by (4.1) on the set J of types.

5.2 Theorem

- a) Let $s = (i, u, j, v) \in S$. If $u \leq v$ and $i \prec_J j$ then it is optimal to use action I , and if $u \geq v$ and $i \succ_J j$ then it is optimal to use action II .
- b) For any state $s = (i, u, i, v) \in S$ it is optimal to choose $a = I$ if and only if $u \leq v$.
- c) For any state $s = (i, u, j, u) \in S$ it is optimal to choose $a = I$ if $i \prec_J j$ and $a = II$ if $j \prec_J i$.

These statements hold for the finite as well as for the infinite horizon problem.

Proof : a) Let $\bar{s} := (j, v, i, u)$, then using Lemma 5.1 we have

$$\Delta_n(s) = L_I V_n(s) - L_{II} V_n(s) = L_{II} V_n(\bar{s}) - L_I V_n(\bar{s}) = -\Delta_n(\bar{s}). \quad (5.2)$$

The assumptions $u \leq v, i \prec_J j$ imply $s \prec \bar{s}$ and from Lemma 7.3 we obtain $\Delta_n(s) \leq \Delta_n(\bar{s}) = -\Delta_n(s)$. Hence $\Delta_n(s) \leq 0$, i.e. action I is optimal. An analogous argument holds for the infinite horizon case. The assertion concerning action II follows from symmetry.

b) and c) follow from a). ■

A degenerate special case is obtained if the separation times do not depend on the leading aircraft, i.e. $b(i, j) \equiv d(j)$ for all $i, j \in J$. In this situation it is unnecessary to keep track of the type of the last airplanes on the runways and the state space of the problem could be reduced to the load (u, v) on the two runways. Theorem 4.1 then implies that the JLL policy $\bar{\delta}$ is optimal.

5.3 Corollary

If $b(i, j) \equiv d(j)$ for all $i, j \in J$, then it is always optimal to route the next airplane to the runway with least load (for finite as well as infinite horizon).

Proof : We have for all $i, j \in J$

$$i \prec_J j \iff \forall k \in J \quad b(i, k) = d(k) \leq b(j, k) = d(k)$$

hence all types in J are equal with respect to the ordering given by (4.1). But then the assertion follows from Theorem 5.2 b). ■

Finally, we give a crude error bound on how much the waiting times under policy JLL can deviate from the optimum in the general model. With $b^* = \max_{i, j \in J} b(i, j)$ as before and $b_* := \min_{i, j \in J} b(i, j)$ let

$$\bar{B} := b^* - b_*$$

be the span of separation times.

5.4 Theorem

Let $\bar{\delta}$ be the JLL policy as defined in (5.1). For all $s \in S$ it holds that

$$0 \leq V_{N\bar{\delta}}(s) - V_N(s) \leq \bar{B} \sum_{k=1}^{N-1} \beta^k k \quad (5.3)$$

and for the infinite horizon and $\beta < 1$

$$0 \leq V_{\bar{\delta}}(s) - V(s) \leq \bar{B} \frac{\beta}{(1-\beta)^2}. \quad (5.4)$$

For the proof we have to consider auxiliary systems with the only difference being modified separation times $\hat{b}(i, j)$. Note that in these systems we not only have a different cost function $\hat{c}(s, a)$ (related to the separation times as given by (3.5)), but also a different transition function \hat{g} and a different state process

$$\hat{X}_n = (\hat{\zeta}_n^I, \hat{U}_n^I, \hat{\zeta}_n^{II}, \hat{U}_n^{II}).$$

The value functions for the modified system are denoted by $\hat{V}_n(s)$ etc.

5.5 Lemma

a) Let $\hat{b}(i, j) \leq b(i, j)$ for all $i, j \in J$, then for any $s \in S$ we have

$$\hat{V}_N(s) \leq V_N(s), \quad N \geq 1 \quad \text{and} \quad \hat{V}(s) \leq V(s).$$

b) Let $\hat{b}(i, j) := b^*$, then we have for the JLL policy $\bar{\delta}$ with $\beta < 1$

$$0 \leq \hat{V}_{N\bar{\delta}}(s) - V_{N\bar{\delta}}(s) \leq \bar{B} \sum_{k=1}^{N-1} \beta^k k, \quad N \geq 1 \quad \text{and}$$

$$0 \leq \hat{V}_{\bar{\delta}}(s) - V_{\bar{\delta}}(s) \leq \bar{B} \frac{\beta}{(1-\beta)^2}.$$

Proof : a) The proof is done by induction on N . For $N = 0$ the statement is trivially true. Now suppose it holds for $n \in \mathbb{N}$, i.e. $\hat{V}_n(s) \leq V_n(s)$ is valid for all $s \in S$. We know that

$$V_{n+1}(s) = \min\{L_I V_n(s), L_{II} V_n(s)\}$$

and similar for \hat{V} . From our assumptions and the induction hypothesis we obtain (note that $s \mapsto V_n(s)$ is obviously increasing in u and v which can formally also be shown by induction)

$$\begin{aligned} L_I \hat{V}_n(s) &= \hat{c}(s, I) + \beta \mathbf{E}[\hat{V}_n(\hat{g}(s, I, Z_1))] \\ &\leq c(s, I) + \beta \mathbf{E}[\hat{V}_n(g(s, I, Z_1))] \\ &\leq c(s, I) + \beta \mathbf{E}[V_n(g(s, I, Z_1))] = L_I V_n(s) \end{aligned}$$

and similarly $L_{II} \hat{V}_n(s) \leq L_{II} V_n(s)$ which implies the result.

b) Let A_k and \hat{A}_k be the k -th action under the JLL policy in the original model and in the hat-model. Note that we may have $A_k \neq \hat{A}_k$ and that

$$U_k^{A_k} = \min\{U_k^I, U_k^{II}\} \quad \text{and} \quad \hat{U}_k^{\hat{A}_k} = \min\{\hat{U}_k^I, \hat{U}_k^{II}\}.$$

We first prove

$$\min\{U_k^I, U_k^{II}\} \leq \min\{\hat{U}_k^I, \hat{U}_k^{II}\} \quad \text{and} \quad \max\{U_k^I, U_k^{II}\} \leq \max\{\hat{U}_k^I, \hat{U}_k^{II}\}. \quad (5.5)$$

Again for $k = 1$ nothing has to be shown. Now assume that (5.5) holds for some k , we have

$$\begin{aligned} \min\{U_{k+1}^I, U_{k+1}^{II}\} & \\ &= \min\left\{[\min\{U_k^I, U_k^{II}\} + b(\zeta_k^{A_k}, J_k)]^+ - T_{k+1}, \max\{U_k^I, U_k^{II}\} - T_{k+1}\right\} \\ &\leq \min\left\{[\min\{\hat{U}_k^I, \hat{U}_k^{II}\} + b^*]^+ - T_{k+1}, \max\{\hat{U}_k^I, \hat{U}_k^{II}\} - T_{k+1}\right\} \\ &= \min\{\hat{U}_{k+1}^I, \hat{U}_{k+1}^{II}\}. \end{aligned} \quad (5.6)$$

We see that (5.6) also holds if the outer min is replaced by max, hence, (5.5) is proven. From this we obtain for all k

$$\begin{aligned} c(X_k, A_k) &= \sum_{j \in J} p_j [\min\{U_k^I, U_k^{II}\} + b(\zeta_k^{A_k}, J_k)]^+ \\ &\leq \sum_{j \in J} p_j [\min\{\hat{U}_k^I, \hat{U}_k^{II}\} + b^*]^+ = \hat{c}(\hat{X}_k, \hat{A}_k) \end{aligned} \quad (5.7)$$

which in turn implies

$$V_{N, \bar{\delta}}(s) \leq \hat{V}_{N, \bar{\delta}}(s). \quad (5.8)$$

To complete the proof of b) we have to show that

$$\hat{V}_{N, \bar{\delta}}(s) \leq V_{N, \bar{\delta}}(s) + \bar{B} \sum_{k=1}^{N-1} \beta^k k. \quad (5.9)$$

Similar to (5.6) we first show that

$$\begin{aligned} \min\{\hat{U}_k^I, \hat{U}_k^{II}\} &\leq \min\{U_k^I, U_k^{II}\} + (k-1)\bar{B} \quad \text{and} \\ \max\{\hat{U}_k^I, \hat{U}_k^{II}\} &\leq \max\{U_k^I, U_k^{II}\} + (k-1)\bar{B} \end{aligned} \quad (5.10)$$

which as in (5.7) implies

$$\hat{c}(\hat{X}_k, \hat{A}_k) \leq c(X_k, A_k) + (k-1)\bar{B} \quad (5.11)$$

and hence (5.9) follows. To prove (5.10) we first observe that for $k = 1$ nothing has to be shown. Now assume (5.10) holds for some k , then

$$\begin{aligned}
& \min\{\hat{U}_{k+1}^I, \hat{U}_{k+1}^{II}\} \\
&= \min\left\{[\min\{\hat{U}_k^I, \hat{U}_k^{II}\} + b^*]^+ - T_{k+1}, \max\{\hat{U}_k^I, \hat{U}_k^{II}\} - T_{k+1}\right\} \\
&\leq \min\left\{[\min\{U_k^I, U_k^{II}\} + (k-1)\bar{B} + b^*]^+ - T_{k+1}, \max\{U_k^I, U_k^{II}\} + (k-1)\bar{B} - T_{k+1}\right\} \\
&= \min\left\{[\min\{U_k^I, U_k^{II}\} + b(\zeta_k^{A_k}, J_k) + (k-1)\bar{B} + b^* - b(\zeta_k^{A_k}, J_k)]^+ - T_{k+1}, \right. \\
&\quad \left. \max\{U_k^I, U_k^{II}\} + (k-1)\bar{B} - T_{k+1}\right\} \\
&\leq \min\left\{[\min\{U_k^I, U_k^{II}\} + b(\zeta_k^{A_k}, J_k) + k\bar{B}]^+ - T_{k+1}, \max\{U_k^I, U_k^{II}\} + k\bar{B} - T_{k+1}\right\} \\
&\leq \min\left\{[\min\{U_k^I, U_k^{II}\} + b(\zeta_k^{A_k}, J_k)]^+ - T_{k+1}, \max\{U_k^I, U_k^{II}\} - T_{k+1}\right\} + k\bar{B} \\
&= \min\{U_{k+1}^I, U_{k+1}^{II}\} + k\bar{B}.
\end{aligned}$$

Again the same inequalities hold when the outer min is replaced by max. Hence, (5.10) holds and the proof of the Lemma is complete for finite horizons. The infinite horizon case again follows from (2.10). ■

For the **proof** of Theorem 5.4 we consider first an auxiliary system with $\hat{b}(i, j) := b_* = \min_{i, j \in J} b(i, j)$ whose value function will be denoted by $*V_N(s)$. Similarly, the system with $\hat{b}(i, j) \equiv b^*$ has value function $*V_{N\bar{\delta}}(s)$. From Corollary 5.3 we know that JLL is optimal in these systems, hence, from Lemma 5.5 a) and b) we see

$$*V_{N\bar{\delta}}(s) = *V_N(s) \leq V_N(s) \leq V_{N\bar{\delta}}(s) \leq *V_{N\bar{\delta}}(s). \quad (5.12)$$

Now we apply Lemma 5.5 b) to the two value functions $*V_{N\bar{\delta}}$ and $*V_{N\bar{\delta}}$ with fixed separation times. We obtain

$$*V_{N\bar{\delta}}(s) - *V_{N\bar{\delta}}(s) \leq \bar{B} \sum_{k=1}^{N-1} k\beta^k \quad (5.13)$$

which together with (5.12) implies the assertion of Theorem 5.4 for the finite horizon, the infinite horizon case again follows from (2.10).

6 The Case of Complete Information : A Counter Example

In this Section, we assume that the controller knows the type of the airplane that has to be routed in addition to the state of the runways. In this case, the control problem is much more complicated. A simple switching policy as in Section 4 need not be optimal any longer as we shall show by a counter example.

6.1 The Model

To model the new situation, the type of the newly arrived airplane is now included into the state space, i.e. we put

$$\tilde{X}_n := (J_n, \zeta_n^I, U_n^I, \zeta_n^{II}, U_n^{II}) \quad \text{and} \quad \tilde{S} := J \times (J \times \mathbb{R}) \times (J \times \mathbb{R})$$

where a state $\tilde{s} \in \tilde{S}$ is denoted by $\tilde{s} = (k, i, u, j, v)$. k gives the type of the newly arrived airplane and i, u, j, v are as before. As in Section 3 we have an external event $z = (l, t)$, but now l is the type of the airplane to arrive after the *next* interarrival time t , i.e. $Z_n = (J_{n+1}, T_{n+1})$. Using the notation of Section 3 we may write the transition function $\tilde{g} : \tilde{S} \times A \times Z \rightarrow \tilde{S}$ for $\tilde{s} = (k, i, u, j, v)$ and $z = (l, t)$ as

$$\tilde{g}(\tilde{s}, a, z) := \begin{cases} \left(l, k, [u + b(i, k)]^+ - t, j, v - t \right) & \text{if } a = I \\ \left(l, i, u - t, k, [v + b(j, k)]^+ - t \right) & \text{if } a = II. \end{cases} \quad (6.1)$$

As cost function $\tilde{c}(s, a)$ we now take the (deterministic) waiting time of the newly arrived airplane when routed to runway a , i.e.

$$\begin{aligned} \tilde{c}(\tilde{s}, I) &:= [u + b(i, k)]^+ \\ \tilde{c}(\tilde{s}, II) &:= [v + b(j, k)]^+. \end{aligned} \quad (6.2)$$

As in the model of Section 3, the optimality equation of the finite horizon dynamic program is given by

$$\tilde{V}_{n+1}(\tilde{s}) = \min_{a \in A} \tilde{L}_a \tilde{V}_n(\tilde{s}) = \min\{\tilde{L}_I \tilde{V}_n(\tilde{s}), \tilde{L}_{II} \tilde{V}_n(\tilde{s})\}$$

where for $\tilde{s} = (k, i, u, j, v)$ and $w : \tilde{S} \rightarrow \mathbb{R}_+$

$$\tilde{L}_I w(\tilde{s}) := \tilde{c}(\tilde{s}, I) + \beta \sum_{l \in J} p_l \int_0^\infty w(l, k, [u + b(i, k)]^+ - t, j, v - t) F(dt) \quad (6.3)$$

and

$$\tilde{L}_{II} w(\tilde{s}) := \tilde{c}(\tilde{s}, II) + \beta \sum_{l \in J} p_l \int_0^\infty w(l, i, u - t, k, [v + b(j, k)]^+ - t) F(dt). \quad (6.4)$$

6.2 A Counter Example

The following example shows that a monotonicity result similar to that of Theorem 4.1 cannot hold in the present scenario. More precisely, if routing the airplane to runway I in state $\tilde{s} = (k, i, u, j, v)$ is optimal then it need not be optimal to route the airplane to runway I in a state $\bar{s} = (k, i, \bar{u}, j, v)$, where $\bar{u} < u$. Hence, with respect to any (partial) ordering $' \prec'$ of \tilde{S} which has $\tilde{s} \prec \bar{s}$ for the above states, monotonicity as in Theorem 4.1 needs not hold. As a consequence, Theorem 5.2, Corollary 5.3 and Theorem 5.4 are no longer valid, though Lemmata 5.1 and 5.5 hold.

The counter example has a very small $\bar{u} < 0$, i.e. the last touch-down on runway I was a long time ago. Then, if the current airplane needs only a small safety distance to the preceding one, it might be better to save runway I for a future airplane which needs a larger separation time.

Example:

We assume that there are three types of aircrafts, $J = \{1, 2, 3\}$ and that the matrix of separation times is as given in [4]:

$$\left(b(i, j) \right)_{i, j=1, \dots, 3} = \begin{pmatrix} 96 & 120 & 144 \\ 72 & 72 & 96 \\ 72 & 72 & 72 \end{pmatrix}.$$

Let $F(t) := 1 - e^{-t}$, i.e. we assume that the arrivals form a Poisson stream with rate $\lambda = 1$.

For a planning horizon of 2, i.e. $N = 2$, we obtain from (6.3) and (6.4) for the difference of the expected cost between routing to I and routing to II in state $\tilde{s} = (k, i, u, j, v)$

$$\begin{aligned} \tilde{\Delta}_2(\tilde{s}) &= \tilde{L}_I \tilde{V}_1(\tilde{s}) - \tilde{L}_{II} \tilde{V}_1(\tilde{s}) = [u + b(i, k)]^+ - [v + b(j, k)]^+ \\ &+ \beta \sum_{l \in J} p_l \int_0^\infty \lambda e^{-\lambda t} \left(\min \left\{ [[u + b(i, k)]^+ - t + b(k, l)]^+, [v - t + b(j, l)]^+ \right\} \right. \\ &\quad \left. - \min \left\{ [u - t + b(i, l)]^+, [[v + b(j, k)]^+ - t + b(k, l)]^+ \right\} \right) dt. \end{aligned} \quad (6.5)$$

Now assume that we are in state $\tilde{s} = (k, i, u, j, v) := (1, 1, -96, 2, -72)$. Then we have $\tilde{c}(\tilde{s}, I) = \tilde{c}(\tilde{s}, II) = 0$ and from (6.5), singling out type 3,

$$\begin{aligned} \Delta_2(\tilde{s}) &= \beta p_3 \left(\int_0^{24} (-24) e^{-t} dt - \int_{24}^{48} (48 - t) e^{-t} dt \right) + \beta(1 - p_3) C_1 \\ &\leq \beta p_3 (-24)(1 - e^{-24}) + \beta(1 - p_3) C_1 \end{aligned}$$

for a constant C_1 not depending on p_3 . For p_3 large enough, we therefore have $\tilde{\Delta}_2(\tilde{s}) < 0$, i.e it is optimal to route to runway I .

Now let $\bar{s} := (1, 1, -144, 2, -72)$ then $\bar{u} = -144 < -96 = u$ but

$$\begin{aligned} \tilde{\Delta}_2(\bar{s}) &= \beta p_3 \int_0^{24} (24 - t) e^{-t} dt + \beta(1 - p_3) C_2 \\ &= \beta p_3 (23 + e^{-24}) + \beta(1 - p_3) C_2 \end{aligned}$$

again for a constant C_2 not depending on p_3 . This time we have $\tilde{\Delta}_2(\bar{s}) > 0$ for p_3 large enough, i.e. in state \bar{s} it is optimal to route to II . Hence, for any ordering \prec on \tilde{S} where $u < \bar{u}$ implies $(k, i, u, j, v) \prec (k, i, \bar{u}, j, v)$, $\tilde{\Delta}_n(\tilde{s})$ need not be s-increasing.

7 The Monotonicity of $s \mapsto \Delta_n(s)$

In this Section we complete the proof of Theorem 4.1. We are using the model of Section 3, i.e. $s = (i, u, j, k)$ and $Z_n = (J_n, T_{n+1})$.

Let us first introduce some notation, that allows to describe the behavior of the system under fixed sequences of actions and external events. For $k \geq 0$ let

$$g^k : S \times A^k \times \mathcal{Z}^k \rightarrow S$$

be defined by

$$\begin{aligned} g^0(s) &:= s \\ g^1(s, a, z) &:= g(s, a, z) \quad \text{as defined in (3.4)} \\ g^{k+1}(s, (a_1, \dots, a_{k+1}), (z_1, \dots, z_{k+1})) &= g\left(g^k(s, (a_1, \dots, a_k), (z_1, \dots, z_k)), a_{k+1}, z_{k+1}\right). \end{aligned} \quad (7.1)$$

Let $\mathbf{a} = (a_1, \dots, a_k) \in A^k$ and $\mathbf{z} = (z_1, \dots, z_k) \in \mathcal{Z}^k$. We denote the components of the state $g^k(s, \mathbf{a}, \mathbf{z})$ in the following way

$$g^k(s, \mathbf{a}, \mathbf{z}) =: \left(\tau_I^k, h_I^k, \tau_{II}^k, h_{II}^k \right) (s, \mathbf{a}, \mathbf{z}) \quad (7.2)$$

$g^k(s, \mathbf{a}, \mathbf{z})$ is the state after k stages, starting in state $s \in S$, applying actions $\mathbf{a} \in A^n$ and given that the external events $\mathbf{z} \in \mathcal{Z}^n$ were observed. Then, e.g. $h_I^k = h_I^k(s, \mathbf{a}, \mathbf{z})$ denotes the load on runway I and $\tau_{II}^k = \tau_{II}^k(s, \mathbf{a}, \mathbf{z})$ is the type of the airplane at the tail of queue II at that time.

We make extensive use of ideas from [1]. In [1], a more general state transition mechanism is considered. There, the transition function g depends on the action a only via an additional random event y whose distribution q_a is controlled by a . Our approach is the special case where q_I, q_{II} are distinct one-point measures (cp. Lemma 2.3(i) in [1]). In [1] it is shown inductively that $s \mapsto \Delta_n(s)$ is increasing with respect to some partial order on S under a number of conditions. Translated into our context the following conditions are used

C.1 $s \mapsto c(s, I) - c(s, II)$ is s-increasing

C.2 $s \mapsto g(s, a, z)$ is s-increasing

C.3 $g^2(s, I, II, z, z') \prec g^2(s, II, I, z, z')$ for all $z, z' \in \mathcal{Z}$

C.4 $s \mapsto [c(s, I) + \beta c(g(s, I, z), II) - c(s, II) - \beta c(g(s, II, z), I)]$ is s-increasing for all $z \in \mathcal{Z}$

C.5 $s \mapsto [c(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), a) - c(g^k(g^2(s, II, I, z, z'), \mathbf{a}, \mathbf{z}), a)]$ is s-increasing for all $\mathbf{a} \in A^k, \mathbf{z} \in \mathcal{Z}^k, a \in A, z, z' \in \mathcal{Z}, k \geq 0$.

Note that conditions C.3 and C.5 examine the permutation of actions I, II in the first two stages with the rest of the actions and all external events fixed.

It turns out that these assumptions do not hold in our context. Instead, we need slightly modified conditions C.3 - C.5 which differ from the above only in that we permute the two first actions as well as the two first external events. We therefore use C.1, C.2 and

C.3* $g^2(s, I, II, z, z') \prec g^2(s, II, I, z', z)$ for all $z, z' \in \mathcal{Z}$

C.4* $s \mapsto [c(s, I) + \beta c(g(s, I, z), II) - c(s, II) - \beta c(g(s, II, z'), I)]$ is s-increasing for all $z, z' \in \mathcal{Z}$

C.5* $s \mapsto [c(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), a) - c(g^k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}), a)]$ is s-increasing for all $\mathbf{a} \in A^k, \mathbf{z} \in \mathcal{Z}^k, a \in A, z, z' \in \mathcal{Z}, k \geq 0$.

We shall refer to the set $\{C.1, C.2, C.3^*, C.4^*, C.5^*\}$ of modified conditions as (C*). We now have to show (a) that the conditions (C*) hold in our model and (b) that the conclusions of [1], namely that $s \mapsto \Delta_n(s)$ is s-increasing, hold under (C*).

7.1 Verifying the Conditions (C*)

We start with a Lemma.

7.1 Lemma

With the preceding notation we have for $s = (i, u, j, v) \in S$ and for any $\mathbf{a} \in A^k, \mathbf{z} \in \mathcal{Z}^k, k \geq 0$

- a) $i \mapsto \tau_I^k(s, \mathbf{a}, \mathbf{z})$ is increasing (with respect to the ordering defined in (4.1)), τ_I^k does not depend on u, j, v .
- b) $i, u \mapsto h_I^k(s, \mathbf{a}, \mathbf{z})$ are increasing, h_I^k does not depend on j, v .

- c) $j \mapsto \tau_{II}^k(s, \mathbf{a}, \mathbf{z})$ is increasing (with respect to the ordering defined in (4.1)), τ_{II}^k does not depend on i, u, v .
- d) $j, v \mapsto h_{II}^k(s, \mathbf{a}, \mathbf{z})$ are increasing, h_{II}^k does not depend on i, u .

Note that $u \mapsto h_I^k(s, \mathbf{a}, \mathbf{z})$ and $v \mapsto h_{II}^k(s, \mathbf{a}, \mathbf{z})$ are also convex.

Proof : a) With $s = (i, u, j, v)$ we have $\tau_I^0(s, \mathbf{a}, \mathbf{z}) = i$ and for $k \geq 1$

$$\tau_I^k(s, \mathbf{a}, \mathbf{z}) = \begin{cases} i & \text{if } a_1 = \dots = a_k = II \\ l_m & \text{if } m = \max\{\nu | 1 \leq \nu \leq k, a_\nu = I\} \text{ and } z_m = (l_m, t_m) \end{cases} .$$

Hence $i \mapsto \tau_I^k$ is increasing and independent of u, j, v for fixed \mathbf{a}, \mathbf{z} .

To prove b) we proceed by induction on $k \geq 0$. Let $k = 0$. We put

$$H(u, b, t, a) := \begin{cases} [u + b]^+ - t & \text{if } a = I \\ u - t & \text{if } a = II \end{cases}$$

Then $u, b \mapsto H(u, b, t, a)$ are increasing and convex functions and with $z_{k+1} = (l_{k+1}, t_{k+1})$ we see from (7.1) and (3.4)

$$\begin{aligned} h_I^{k+1}(s, \mathbf{a}, a_{k+1}, \mathbf{z}, z_{k+1}) &= H(h_I^k(s, \mathbf{a}, \mathbf{z}), b(\tau_I^k(s, \mathbf{a}, \mathbf{z}), l_{k+1}), t_{k+1}, a_{k+1}) \\ &= \begin{cases} [h_I^k(s, \mathbf{a}, \mathbf{z}) + b(\tau_I^k(s, \mathbf{a}, \mathbf{z}), l_{k+1})]^+ - t_{k+1} & \text{if } a_{k+1} = I \\ h_I^k(s, \mathbf{a}, \mathbf{z}) - t_{k+1} & \text{if } a_{k+1} = II. \end{cases} \end{aligned} \quad (7.3)$$

Assume that a) holds for k . Then $u, i \mapsto h_i^k$ and $u, i \mapsto b(\tau_I^k(s, \mathbf{a}, \mathbf{z}), l_{k+1})$ are increasing mappings that do not depend on j, v . From (7.3) it is then obvious that b) holds for $k + 1$.

c) and d) follow in the same way. ■

Now we can show that the above conditions (C*) hold in our model.

7.2 Lemma

- a) $s \mapsto c(s, I) - c(s, II)$ is s-increasing
- b) $s \mapsto g(s, a, z)$ is s-increasing
- c) $g^2(s, I, II, z, z') \prec g^2(s, II, I, z', z)$ for all $z, z' \in \mathcal{Z}$
- d) $s \mapsto \left[c(s, I) + \beta c(g(s, I, z), II) - c(s, II) - \beta c(g(s, II, z'), I) \right]$ is s-increasing for all $z, z' \in \mathcal{Z}$
- e) $s \mapsto \left[c(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), a) - c(g^k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}), a) \right]$ is s-increasing for all $\mathbf{a} \in A^k, \mathbf{z} \in \mathcal{Z}^k, a \in A, z, z' \in \mathcal{Z}, k \geq 0$.

Proof : a) We have $u, i \mapsto [u + b(i, j')]^+$ are increasing for all $j' \in J$, hence from (4.3)

$$\begin{aligned} s &\mapsto c(s, I) - c(s, II) \\ &= \sum_{j' \in J} p_{j'} \cdot \left([u + b(i, j')]^+ - [v + b(i, j')]^+ \right) \end{aligned}$$

is s-increasing.

b) follows from Lemma 7.1.

c) To prove c), we look at the components of $g^2(s, I, II, z, z')$ and $g^2(s, II, I, z', z)$. With $z = (l, t), z' = (l', t')$ we have $\tau_I^2 = l$ and $\tau_{II}^2 = l'$ in both cases and as $[x]^+ - t \leq [x - t]^+$, we obtain

$$\begin{aligned} h_I^2(s, I, II, z, z') &= [u + b(i, l)]^+ - t - t' \\ &\leq [u - t' + b(i, l)]^+ - t \\ &= h_I^2(s, II, I, z', z). \end{aligned}$$

In the same way one shows $h_{II}^2(s, I, II, z, z') \geq h_{II}^2(s, II, I, z', z)$. Hence c), i.e. C.3* follows. Note that $h_I^2(s, II, I, z, z') = [u - t + b(i, l')]^+ - t'$ which we cannot relate to $h_I^2(s, I, II, z, z')$ without serious restrictions on the separation times $b(i, \cdot)$. Hence the original condition C.3 as used in [1] need not hold in our model.

d) From the definition of the one-stage cost function in (3.5) we have for $s = (i, u, j, v), z = (l, t), z' = (l', t')$

$$\begin{aligned} &c(s, I) - c(s, II) + \beta \left(c(g(s, I, z), II) - c(g(s, II, z'), I) \right) \\ &= \sum_{j' \in J} p_{j'} \left[[u + b(i, j')]^+ - [v + b(j, j')]^+ + \beta [v - t + b(j, j')]^+ - \beta [u - t' + b(i, j')]^+ \right] \\ &= \sum_{j' \in J} p_{j'} \left[\left([u + b(i, j')]^+ - \beta [u - t' + b(i, j')]^+ \right) - \left([v + b(j, j')]^+ - \beta [v - t + b(j, j')]^+ \right) \right]. \end{aligned} \tag{7.4}$$

It is not difficult to see that an expression of the form $x \mapsto r(x) - \beta r(x - t)$ with r increasing and convex and $0 < \beta \leq 1$ is increasing in x . As $u, i \mapsto u + b(i, j')$ and $v, j \mapsto v + b(j, j')$ are increasing we see from (4.3) that d) holds.

e) Fix $s = (i, u, j, v) \in S$ and $z = (l, t), z' = (l', t') \in \mathcal{Z}$. Define

$$\begin{aligned} \sigma_1 &:= g^2(s, I, II, z, z') = (l, [u + b(i, l)]^+ - t - t', l', [v - t + b(j, l')]^+ - t') \\ \sigma_2 &:= g^2(s, II, I, z', z) = (l, [u - t' + b(i, l)]^+ - t, l', [v + b(j, l')]^+ - t' - t). \end{aligned}$$

Let us assume that $a = I$. From Lemma 7.1 b) we see that

$$h_I^k(\sigma_1, \mathbf{a}, \mathbf{z}) = h_I^k((l, [u + b(i, l)]^+ - t - t', l', [v - t + b(j, l')]^+ - t'), \mathbf{a}, \mathbf{z})$$

is increasing in the first two coordinates of σ_1 and does not depend on the last two, hence it is increasing in i, u and does not depend on j, v . Similarly, $h_I^k(\sigma_2, \mathbf{a}, \mathbf{z})$ is increasing in i, u and is independent of j, v . Using Lemma 7.1 a) we therefore have

$$\begin{aligned} &c(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), I) - c(g^k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}), I) \\ &= \sum_{j' \in J} p_{j'} \left([h_I^k(\sigma_1, \mathbf{a}, \mathbf{z}) + b(\tau_I^k(\sigma_1, \mathbf{a}, \mathbf{z}), j')]^+ - [h_I^k(\sigma_2, \mathbf{a}, \mathbf{z}) + b(\tau_I^k(\sigma_2, \mathbf{a}, \mathbf{z}), j')]^+ \right) \\ &= \sum_{j' \in J} p_{j'} \left([h_I^k(l, [u + b(i, l)]^+ - t - t', \cdot, \cdot) + b(\tau_I^k(l, \cdot, \cdot, \cdot), j')]^+ \right. \\ &\quad \left. - [h_I^k(l, [u - t' + b(i, l)]^+ - t, l', \cdot, \cdot) + b(\tau_I^k(l, \cdot, \cdot, \cdot), j')]^+ \right) \end{aligned} \tag{7.5}$$

It is not difficult to show that if $r : \mathbb{R} \rightarrow \mathbb{R}$ is an increasing function and $d \in \mathbb{R}, t > 0$, then

$$x \mapsto r([x]^+ + d - t) - r([x - t]^+ + d)$$

is increasing. With $x = u + b(i, l)$ we obtain, that (7.5) is increasing in i, u and independent from j, v , hence it is s-increasing. Similarly, for $a = II$ we see that

$$\begin{aligned}
& c(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), II) - c(g^k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}), II) \\
&= \sum_{j' \in J} p_{j'} \left([h_{II}^k(\sigma_1, \mathbf{a}, \mathbf{z}) + b(\tau_{II}^k(\sigma_1, \mathbf{a}, \mathbf{z}), j')]^+ - [h_{II}^k(\sigma_2, \mathbf{a}, \mathbf{z}) + b(\tau_{II}^k(\sigma_2, \mathbf{a}, \mathbf{z}), j')]^+ \right) \\
&= \sum_{j' \in J} p_{j'} \left([h_{II}^k(\cdot, \cdot, l', [v - t + b(j, l')]^+ - t')] + b(\tau_{II}^k(\cdot, \cdot, l', \cdot), j')]^+ \right. \\
&\quad \left. - [h_{II}^k(\cdot, \cdot, l', [v + b(j, l')]^+ - t - t)] + b(\tau_{II}^k(\cdot, \cdot, l', \cdot), j')]^+ \right)
\end{aligned} \tag{7.6}$$

is independent of i, u and decreasing in j, v , hence s-increasing. ■

7.2 Monotonicity of $s \mapsto \Delta_n(s)$ under Conditions (C*)

We now show that $\Delta_n(s)$ is s-increasing under our modified conditions. For the sake of completeness, we give a streamlined version of the proofs of [1] here .

7.3 Lemma

Let (C*) hold. For any $s \in S, n \geq 0$

$$s \mapsto \Delta_n(s) \text{ is s-increasing.}$$

Proof : We proceed by induction on $n \geq 0$. For $n = 0$ we have with $V_0 \equiv 0$ and $s = (i, u, j, v)$

$$\Delta_0(s) = L_I V_0(s) - L_{II} V_0(s) = c(s, I) - c(s, II) \tag{7.7}$$

which is s-increasing by condition C.1. Now assume that $s \mapsto \Delta_k(s)$ is s-increasing for all $k \leq n - 1$. We shall show that the same holds for $\Delta_n(s)$. Note that for any $a, b \in \mathbb{R}$ we have

$$\min\{a, b\} = b - [b - a]^+ = a + [b - a]_- \tag{7.8}$$

where we denote

$$[a]_- := \min\{0, a\}. \tag{7.9}$$

We now obtain for any $s \in S, n \geq 1$

$$\begin{aligned}
\Delta_n(s) &= L_I V_n(s) - L_{II} V_n(s) \\
&= c(s, I) + \beta \mathbf{E} [V_n(X_2) \mid X_1 = s, A_1 = I] - c(s, II) - \beta \mathbf{E} [V_n(X_2) \mid X_1 = s, A_1 = II] \\
&= c(s, I) + \beta \mathbf{E} \left[\min_{a=I, II} L_a V_{n-1}(X_2) \mid X_1 = s, A_1 = I \right] \\
&\quad - c(s, II) - \beta \mathbf{E} \left[\min_{a=I, II} L_a V_{n-1}(X_2) \mid X_1 = s, A_1 = II \right] \\
&= c(s, I) + \beta \mathbf{E} \left[L_{II} V_{n-1}(X_2) + [L_I V_{n-1}(X_2) - L_{II} V_{n-1}(X_2)]_- \mid X_1 = s, A_1 = I \right] \\
&\quad - c(s, II) - \beta \mathbf{E} \left[L_I V_{n-1}(X_2) - [L_I V_{n-1}(X_2) - L_{II} V_{n-1}(X_2)]^+ \mid X_1 = s, A_1 = II \right] \\
&= c(s, I) + \beta \mathbf{E} [L_{II} V_{n-1}(X_2) \mid X_1 = s, A_1 = I] + \beta \mathbf{E} \left[[\Delta_{n-1}(X_2)]_- \mid X_1 = s, A_1 = I \right]
\end{aligned} \tag{7.10}$$

$$\begin{aligned}
& -c(s, II) - \beta \mathbf{E} [L_I V_{n-1}(X_2) \mid X_1 = s, A_1 = II] + \beta \mathbf{E} \left[[\Delta_{n-1}(X_2)]^+ \mid X_1 = s, A_1 = II \right] \\
= & c(s, I) + \beta \mathbf{E} L_{II} V_{n-1}(g(s, I, Z_1)) + \beta \mathbf{E} [\Delta_{n-1}(g(s, I, Z_1))]_- \\
& -c(s, II) - \beta \mathbf{E} L_I V_{n-1}(g(s, II, Z_1)) + \beta \mathbf{E} [\Delta_{n-1}(g(s, II, Z_1))]^+.
\end{aligned}$$

From the induction hypotheses and condition C.2 we now infer that $s \mapsto \Delta_{n-1}(g(s, a, z))$ is s-increasing. As $[\cdot]^+$ and $[\cdot]_-$ are monotone functions we obtain that

$$s \mapsto \mathbf{E} [\Delta_{n-1}(g(s, I, Z_1))]_- + \mathbf{E} [\Delta_{n-1}(g(s, II, Z_1))]^+$$

is s-increasing. For the proof of the Lemma it is therefore sufficient to show that the remainder of (7.10) is s-increasing, i.e.

$$\begin{aligned}
& s \mapsto c(s, I) + \beta \mathbf{E} L_{II} V_{n-1}(g(s, I, Z_1)) - c(s, II) - \beta \mathbf{E} L_I V_{n-1}(g(s, II, Z_1)) \\
& = \mathbf{E} \left[c(s, I) + \beta c(g(s, I, Z_1), II) + \beta^2 V_{n-1}(g^2(s, I, II, Z_1, Z_2)) \right. \\
& \quad \left. - c(s, II) - \beta c(g(s, II, Z_2), I) - \beta^2 V_{n-1}(g^2(s, II, I, Z_2, Z_1)) \right] \\
& = \mathbf{E} \xi(s, Z_1, Z_2)
\end{aligned} \tag{7.11}$$

where ξ is defined in (7.12) below.

Note that in the second equation of (7.11) we have exchanged Z_1, Z_2 , which is possible as they are i.i.d.. This is a minor change from the derivation in [1] and allows to use conditions C.3* - C.5*. As Z_1, Z_2 are also independent of the rest, the monotonicity of (7.11) follows if $s \mapsto \xi(s, z_1, z_2)$ is s-increasing for all $z_1, z_2 \in \mathcal{Z}$ which is shown in the next Lemma. \blacksquare

7.4 Lemma

If $s \mapsto \Delta_k(s)$ is s-increasing for all $k \leq n$ then the following expression is s-increasing for all $z, z' \in \mathcal{Z}$:

$$\begin{aligned}
\xi(s, z, z') := & c(s, I) + \beta c(g(s, I, z), II) + \beta^2 V_n(g^2(s, I, II, z, z')) \\
& - c(s, II) - \beta c(g(s, II, z'), I) - \beta^2 V_n(g^2(s, II, I, z', z)).
\end{aligned} \tag{7.12}$$

For the proof of this Lemma we need the following definition let $R_0(s) := 0$ and for $\mathbf{a} = (a_1, \dots, a_k) \in A^k$ and $\mathbf{z} = (z_1, \dots, z_k) \in \mathcal{Z}^k$ let

$$R_k(s, \mathbf{a}, \mathbf{z}) = \sum_{m=0}^{k-1} \beta^m c(g^m(s, \mathbf{a}, \mathbf{z}), a_{m+1}).$$

Then $R_k(s, \mathbf{a}, \mathbf{z})$ is the discounted cost over k stages when starting in state s and following a fixed routing policy \mathbf{a} , with fixed external events \mathbf{z} . Note that g^m depends only on part of the sequences \mathbf{a} and \mathbf{z} .

Proof : (of Lemma 7.4)

1. We follow the lines of the proof of Lemma 2.2 in [1]. For $0 \leq k \leq n$ and $z, z' \in \mathcal{Z}$ let

$$\begin{aligned}
\Phi_k(s, \mathbf{a}, \mathbf{z}) := & c(s, I) + \beta c(g(s, I, z), II) + \beta^2 R_k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) \\
& + \beta^{k+2} V_{n-k}(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z})) \\
& - c(s, II) - \beta c(g(s, II, z'), I) - \beta^2 R_k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}) \\
& - \beta^{k+2} V_{n-k}(g^k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}))
\end{aligned} \tag{7.13}$$

where $\mathbf{a} = (a_1, \dots, a_k) \in A^k$, $\mathbf{z} = (z_1, \dots, z_k) \in \mathcal{Z}^k$ are arbitrary fixed sequences. The first half of $\Phi_k(\cdot)$ describes the cost from $n+2$ stages starting in state s , when in the first $k+2$ stages the fixed policy (I, II, a_1, \dots, a_k) is used, the external events (z, z', z_1, \dots, z_k) occur and an optimal policy is used for the remaining $n-k$ stages. The second half of $\Phi_k(\cdot)$ interchanges the actions I, II and the first two external events z, z' .

2. The main step of the proof is to show that Φ_k is s -increasing for all $k = 0, \dots, n$ by downward induction on $k = n, \dots, 0$. The Lemma then follows as $\Phi_0(s) = \xi(s)$.

2.1. For $k = n$, Φ_n reduces to the expression (7.21) in Lemma 7.5 below. It is proven there that this expression is s -increasing for any $n \geq 1$.

2.2 Now assume that $\Phi_{k+1}(\cdot)$ is increasing in s for any sequences $\mathbf{a} \in A^{k+1}$ and $\mathbf{z} \in \mathcal{Z}^{k+1}$. We want to show that the analogous result holds for $\Phi_k(\cdot)$. Let $s \prec s'$ and put

$$\begin{aligned}\sigma_1 &:= g^k(g^2(s', I, II, z, z'), \mathbf{a}, \mathbf{z}) & \sigma_2 &:= g^k(g^2(s', II, I, z', z), \mathbf{a}, \mathbf{z}) \\ \sigma_3 &:= g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) & \sigma_4 &:= g^k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}).\end{aligned}$$

From C.2 and C.3* we obtain that $\sigma_1 \prec \sigma_2$ and $\sigma_3 \prec \sigma_4$. Again from C.2 follows $\sigma_3 \prec \sigma_1$ and $\sigma_4 \prec \sigma_2$, hence we have

$$\sigma_3 \prec \sigma_1, \sigma_4 \quad \text{and} \quad \sigma_1, \sigma_4 \prec \sigma_2. \quad (7.14)$$

Then

$$\begin{aligned}\Phi_k(s', a_1, \dots, a_k, z_1, \dots, z_k) - \Phi_k(s, a_1, \dots, a_k, z_1, \dots, z_k) \\ = c(s', I) + \beta c(g(s', I, z), II) + \beta^2 R_k(g^2(s', I, II, z, z'), \mathbf{a}, \mathbf{z}) \\ - c(s', II) - \beta c(g(s', II, z'), I) - \beta^2 R_k(g^2(s', II, I, z', z), \mathbf{a}, \mathbf{z}) \\ - c(s, I) - \beta c(g(s, I, z), II) - \beta^2 R_k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) \\ + c(s, II) + \beta c(g(s, II, z'), I) + \beta^2 R_k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}) \\ + \beta^{k+2} (V_{n-k}(\sigma_1) - V_{n-k}(\sigma_2) - V_{n-k}(\sigma_3) + V_{n-k}(\sigma_4))\end{aligned} \quad (7.15)$$

2.2.1 We shall now show that there is $a' \in A$ with

$$\begin{aligned}\Gamma &:= V_{n-k}(\sigma_1) - V_{n-k}(\sigma_2) - V_{n-k}(\sigma_3) + V_{n-k}(\sigma_4) \\ &\geq L_{a'} V_{n-k-1}(\sigma_1) - L_{a'} V_{n-k-1}(\sigma_2) - L_{a'} V_{n-k-1}(\sigma_3) + L_{a'} V_{n-k-1}(\sigma_4).\end{aligned} \quad (7.16)$$

To simplify the notation we put $w(s) := V_{n-k-1}(s)$. Assume that $a \in A$ is optimal in σ_1 and $b \in A$ is optimal in σ_4 . If $a = b$ then (7.16) holds for $a = a'$. If $a \neq b$ then

$$\begin{aligned}\Gamma &\geq L_a w(\sigma_1) - L_{II} w(\sigma_2) - L_I w(\sigma_3) + L_b w(\sigma_4) \\ &= L_I w(\sigma_1) - L_I w(\sigma_2) - L_I w(\sigma_3) + L_I w(\sigma_4) \\ &\quad - (L_I w(\sigma_1) - L_a w(\sigma_1)) + L_I w(\sigma_2) - L_{II} w(\sigma_2) - (L_I w(\sigma_4) - L_b w(\sigma_4)) \\ &= L_I w(\sigma_1) - L_I w(\sigma_2) - L_I w(\sigma_3) + L_I w(\sigma_4) \\ &\quad - \mathbf{1}_{[a=II]} \Delta_{n-k-1}(\sigma_1) + \Delta_{n-k-1}(\sigma_2) - \mathbf{1}_{[b=II]} \Delta_{n-k-1}(\sigma_4) \\ &\geq L_I w(\sigma_1) - L_I w(\sigma_2) - L_I w(\sigma_3) + L_I w(\sigma_4).\end{aligned} \quad (7.17)$$

Here, the last inequality follows as it is assumed in this Lemma that $s \mapsto \Delta_l(s)$ is increasing for all $l \leq n$, hence we see from (7.14) that

$$\Delta_{n-k-1}(\sigma_2) - \Delta_{n-k-1}(\sigma_1) \geq 0 \quad \text{and} \quad \Delta_{n-k-1}(\sigma_2) - \Delta_{n-k-1}(\sigma_4) \geq 0.$$

In the same way we obtain

$$\begin{aligned}\Gamma &\geq L_{II}w(\sigma_1) - L_{II}w(\sigma_2) - L_{II}w(\sigma_3) + L_{II}w(\sigma_4) \\ &\quad + \mathbf{1}_{[a=I]}\Delta_{n-k-1}(\sigma_1) - \Delta_{n-k-1}(\sigma_3) + \mathbf{1}_{[b=I]}\Delta_{n-k-1}(\sigma_4) \\ &\geq L_{II}w(\sigma_1) - L_{II}w(\sigma_2) - L_{II}w(\sigma_3) + L_{II}w(\sigma_4).\end{aligned}$$

2.2.2 Inserting the definition of L_a into (7.16) we obtain

$$\begin{aligned}\Gamma &\geq c(\sigma_1, a') - c(\sigma_2, a') - c(\sigma_3, a') + c(\sigma_4, a') \\ &\quad + \beta \mathbf{E} \left(V_{n-k-1}(g(\sigma_1, a', Z_{k+1})) - V_{n-k-1}(g(\sigma_2, a', Z_{k+1})) \right. \\ &\quad \left. - V_{n-k-1}(g(\sigma_3, a', Z_{k+1})) + V_{n-k-1}(g(\sigma_4, a', Z_{k+1})) \right).\end{aligned}\tag{7.18}$$

As Φ_k depends only on the first k components of $\mathbf{a} = (a_1, \dots, a_{k+1})$ we may choose $a_{k+1} = a'$ to obtain

$$\begin{aligned}R_{k+1}(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) &= R_k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) + \beta^k c(g^k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), a_{k+1}) \\ &= R_k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) + \beta^k c(\sigma_3, a_{k+1})\end{aligned}\tag{7.19}$$

and similarly for σ_1, σ_2 and σ_4 .

2.2.3 Returning to (7.15) we obtain from (7.18) and (7.19)

$$\begin{aligned}\Phi_k(s', \mathbf{a}, \mathbf{z}) - \Phi_k(s, \mathbf{a}, \mathbf{z}) &\geq c(s', I) + \beta c(g(s', I, z), II) + \beta^2 R_{k+1}(g^2(s', I, II, z, z'), \mathbf{a}, \mathbf{z}) \\ &\quad - c(s', II) - \beta c(g(s', II, z'), I) - \beta^2 R_{k+1}(g^2(s', II, I, z', z), \mathbf{a}, \mathbf{z}) \\ &\quad - c(s, I) - \beta c(g(s, I, z), II) - \beta^2 R_{k+1}(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) \\ &\quad + c(s, II) + \beta c(g(s, II, z'), I) + \beta^2 R_{k+1}(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}) \\ &\quad + \beta^{k+3} \mathbf{E} \left(V_{n-k-1}(g(\sigma_1, a_{k+1}, Z_{k+1})) - V_{n-k-1}(g(\sigma_2, a_{k+1}, Z_{k+1})) \right. \\ &\quad \left. - V_{n-k-1}(g(\sigma_3, a_{k+1}, Z_{k+1})) + V_{n-k-1}(g(\sigma_4, a_{k+1}, Z_{k+1})) \right) \\ &= \mathbf{E} \left(\Phi_{k+1}(s', a_1, \dots, a_{k+1}, z_1, \dots, z_k, Z_{k+1}) \right. \\ &\quad \left. - \Phi_{k+1}(s, a_1, \dots, a_{k+1}, z_1, \dots, z_k, Z_{k+1}) \right) \\ &\geq 0\end{aligned}\tag{7.20}$$

where the last step follows from the induction hypotheses. ■

7.5 Lemma

Let C.4* and C.5* hold. Then for all $k \geq 0, \mathbf{a} \in A^k, \mathbf{z} \in \mathcal{Z}^k$ the following expression is s-increasing:

$$\begin{aligned}\zeta(s) &= c(s, I) + \beta c(g(s, I, z), II) + \beta^2 R_k(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}) \\ &\quad - c(s, II) - \beta c(g(s, II, z'), I) - \beta^2 R_k(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}).\end{aligned}\tag{7.21}$$

Proof : We have from the definition of R_k

$$\begin{aligned} \zeta(s) = & \left(c(s, I) - c(s, II) \right) + \beta \left(c(g(s, I, z), II) - c(g(s, II, z'), I) \right) \\ & + \beta^2 \sum_{m=0}^{k-1} \beta^m \left(c(g^m(g^2(s, I, II, z, z'), \mathbf{a}, \mathbf{z}), a_{m+1}) - c(g^m(g^2(s, II, I, z', z), \mathbf{a}, \mathbf{z}), a_{m+1}) \right). \end{aligned} \tag{7.22}$$

The first part of this expression is s-increasing by condition C.4*, the second is a sum of terms which are s-increasing by C.5*. ■

8 Conclusion

In this paper we have investigated optimal assignment rules in a particular model of aircraft arrivals. We have shown that optimal policies are of switching type only if we restrict the information on which decisions are based to the state of the two runways, i.e. to the workload and the types of aircrafts waiting at the end of the queues.

Determining the optimal assignment policy explicitly is a most difficult task. Classical approaches as policy iteration or value iteration (see [11]) are of limited use here due to the complex search space of possible decision rules. Recent approaches to incorporate numerical approximation techniques are presented e.g. in [6].

Our results narrow the space of possible decision rules. If we restrict the search to monotone rules, i.e. to switching levels, we are sure to cover policies that are optimal among those that depend only on the state of the runways.

The authors of the present paper have some experience with the optimization of assignment policies using heuristic search methods as genetic algorithms where the expected waiting times are estimated by discrete event simulation. Monotone policies or rather the switching levels are easily stored and manipulated on a computer. Though our results above show that these rules need not be optimal in general, they perform quite well in practice.

Our future research will focus on two topics. First, we shall investigate models that take into account more than just one arrival. Even in the random environment assumed here, airport controllers usually know about the next few arrivals and may base their decision on that information. Secondly, we shall work on approximation techniques as e.g. in [6] making use of structural properties as proven in the present paper.

Acknowledgment: The authors would like to thank the referee of a former version of this paper for useful suggestions concerning the monotonicity with respect to the types.

References

- [1] ALTMAN, E. AND S. STIDHAM, JR. (1995) Optimality of monotonic policies for two-action Markovian decision processes, with applications to control of queues with delayed information. *Queueing Systems* **21**, p. 267 - 291.
- [2] BÄUERLE, N., O. ENGELHARDT-FUNKE AND M. KOLONKO (2002) Routing of Airplanes to Two Runways: Stability and Bounds for the Waiting Times. Submitted for publication.

- [3] BEASLEY, J.E., M. KRISHNAMOORTY, Y.M. SHARAIHA AND D. ABRAMSON (2000) Scheduling Aircraft Landings - The Static Case. *Transportation Science* **34**, p 180-197.
- [4] BOLENDER, M.A. AND G.L. SLATER (2000) Evaluation of Scheduling Methods for Multiple Runways. *Journal of Aircrafts* **37**, 410 - 416.
- [5] BERTSEKAS, D.(1987) Dynamic Programming: Deterministic and Stochastic Models. *Prentice Hall, Englewood Cliffs, N.J.*
- [6] BERTSEKAS, D. AND JOHN N. TSITSIKLIS.(1996) Neuro-Dynamic Programming *Prentice Hall, Englewood Cliffs, N.J.*
- [7] HORDIJK, A. , G. M. KOOLE AND J. A. LOEVE (1994) Analysis of a customer assignement model with no state information. *Prob. in the Engineering and Information Sciences* **8**, p. 419 - 429.
- [8] HORONJEFF, R. AND F.X. MCKELVEY (1994) *Planning and Design of Airports*. McGraw-Hill, 4. ed., Boston.
- [9] KOOLE, G. (1996) On the pathwise optimal Bernoulli routing policy for homogeneous parallel servers. *Mathematics of Operations Research* **21**, p. 469-478.
- [10] NEUTS, M.F. (1977) Some explicit formulas for the steady-state behaviour of the queue with semi-markovian service times . *Adv. Appl. Prob.* **9**, 141-157.
- [11] PUTERMAN, M.L. (1994) *Markov Decision Processes. Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Mathematical Statistics, New York.
- [12] SCHÄL, M. (1975) Conditions for the Optimality in Dynamic Programming and for the Limit of n-Stage Optimal Policies to Be Optimal. *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, **32**, 179-196
- [13] STIDHAM, S. JR. AND R. WEBER (1993) A Survey of Markov Decision Models for Control of Networks of Queues. *Queueing Systems*, **13**, 291 - 314. New York.