Routing jobs to servers with deterministic service times

by D.A van der Laan

July 5, 2000

Abstract

In this paper we consider the problem of routing deterministic arriving jobs to parallel servers with deterministic (distinct) service times, where we assume that the arrival rate of the jobs is equal to the total service capacity of the servers. Our goal is to find routing policies that minimize the long-run average waiting time of the arriving jobs. We give lower and upper bounds for the minimal long-run average waiting time and we present results on the structure of optimal policies. We derive mathematical programming problems which can be solved to obtain optimal routing policies and for a single queue we have results on routing according to a regular sequence to that queue. Finally we discuss several algorithms to obtain good deterministic routing policies.

Keywords : parallel deterministic servers, deterministic routing policies, regular sequences.

1 Introduction

We consider a queueing system with $n \geq 2$ parallel servers each having its own queue. Arriving jobs have to be routed to one of the servers at the moment of arrival. We assume that the arrival of jobs is deterministic with a constant rate. We also assume that the serving times are deterministic, but typically the servers have different rates. We may think of a computer system with several processors which has to perform the incoming jobs. Our goal is to minimize the long-run average waiting time. Similar queueing systems with parallel heterogeneous servers have been considered in literature, but in general Poisson arrivals are assumed and the serving times are exponentially distributed or general. Further in such stochastic models a distinction is made between dynamic and static routing policies. In the dynamic case the policy may depend on time dependent information, for example the number of jobs or the remaining workload in each queue. In the static case the policy should only depend on the base characteristics of the system, such as the arrival rate and service time distributions. However, it is clear that for our deterministic model there is no distinction between dynamic and static policies. The stochastic model that is closest to ours is the static case with allocation according to a fixed (periodic) pattern. This is also called semi-dynamic deterministic routing. Some papers dealing with such models are [5], [20], [6] and [9]. In these papers several algorithms and heuristics are given to obtain reasonable good policies for the models considered.

For this kind of models the optimization procedure actually consist of two steps:

1. Approximate for $i = 1, 2, \ldots, n$ the fraction p_i^* of jobs that should be routed to server i in the optimal pattern by fractions p_i such that $\sum_{i=1}^{n} p_i = 1$.

2. Construct an allocation pattern with the fractions p_i.

Usually most of the attention goes to step 1. In our model we concentrate entirely on step 2 where we assume that the arrival rate is equal to the combined service rate of the n servers, in other words, that the traffic intensity ρ satisfies $\rho = 1$. For a stochastic model the system overflows if $\rho = 1$ and waiting times will tend to ∞. However in the deterministic model the system can just be stabilized. The fact that the fractions p_i are fixed also implies that with minimizing the long-run average waiting time we also minimize the long-run average sojourn time (which is waiting time plus service time). We think that an optimal allocation pattern for given fractions in our model will at least in the heavy traffic region perform very well for more general service time distributions and arrival processes too.

Consider the following " most regular " zero-one valued splitting sequence of asymptotic mean p :

$$\{b_k^p(\phi)\}_{k=1}^{\infty} = \lfloor (k+1)p + \phi \rfloor - \lfloor kp + \phi \rfloor. \tag{1}$$

In (1) $\phi \in \mathbb{R}$ is an arbitrary phase. Such a sequence is called a regular, Beatty, Sturmian, or bracket sequence with density p. Such sequences are studied in several areas of mathematics and for more about this sequence see [12], [13], [2], [17], [18], [19], [10] and [11]. Remark that the sequence is periodic if $p \in \mathbb{Q}$. For a single server let $\{b_k\}_{k=1}^{\infty}$ be a zero-one splitting sequence, where the k-th arriving job is routed to the server if and only if $b_k = 1$. In [8] it is shown that if a fraction p of jobs has to be routed to a single exponential server, then the long-run average queue size is minimized, if sequence (1) is used. So according to Little's theorem the long-run average waiting time of jobs which are routed to that server is minimized by sequence (1). Hajek proved this for an exponential server, but it holds much more generally (see [1] and [3]) and we will see that it also holds in our model. The optimality of the regular sequence for a single queue can be used to prove that a static routing policy is optimal if the corresponding splitting sequences for every single server i are regular sequences with given (optimized) density p_i for $i = 1, 2, \dots, n$. This is done in [2] and [3] for several models. The integer sequence corresponding to such an optimal policy is called an exactly covering sequence or balanced sequence. However, if $n > 2$ then in general an exactly covering sequence does not exist for given densities p_i. Only for $n = 2$ there exists a balanced sequence for every pair of fractions $(p, 1-p)$, because the complement of a regular sequence with density p is a regular sequence with density $1 - p$. In [2], [18], [19] and [10] it is studied which densities are " balanceable " if $n > 2$. So the optimal routing policy in our model is in fact known if $n = 2$. In other cases the policy will be such that the corresponding splitting sequences are simultaneously as regular as possible in some sense.

Our paper is organised as follows. In section 2 we describe the queueing model and some notation is introduced. In section 3 we deduce a fundamental relation between the long-run average waiting time and the total unused work capacity of the system. In section 4 we will find a mathematical programming problem (MPP) which can be used to minimize the long-run average waiting time and we deduce an upper bound for the minimal long-run average waiting time. In section 5 we find some results on the structure of optimal policies. Further we show that the upper bound for the minimal long-run average waiting time is best possible if the fractions p_i are linearly independent over $\mathbb{Z}$. In section 6 we consider the case that all the fractions p_i are rational. We show that in that case we can restrict to proportional periodic policies to find an optimal routing policy and that an optimal periodic policy can be found by solving some linear programming (LP) problem with zero-one variables, which is an integer linear programming problem (ILP). We obtain a lower bound for the minimal long-run average waiting time by solving the LP-relaxation of this ILP. Further we give an upper bound for this case which is a fraction better than the earlier found general upper bound. In section 7 we show that the lower bound on the minimal long-run average waiting time that we obtained in section 6 is attained if the policy is such that for every single queue the corresponding splitting sequence is a regular sequence with appropriate density. Thus the lower bound can be attained if the given fractions are balanceable. Further we deduce an explicit formula for the long-run average waiting time of customers routed to some server i if the corresponding splitting sequence is regular. Finally in section 8 we consider some algorithms to obtain good policies.

2 The queueing model

We consider the following queueing system. Arriving jobs have to be routed at the moment of arrival to one of $n \geq 2$ parallel servers, each having its own queue. We assume that the arrival of jobs is deterministic with a constant rate. Starting at time $t = 0$ one job arrives every time unit. We also assume that the serving times of the n parallel servers are deterministic and that they have a total working capacity of 1. So,

$$a_1 + a_2 + \cdots + a_n = 1, \tag{2}$$

where a_i^{-1} is the service time per job of server i. Moreover without loss of generality we assume that $a_1 \geq a_2 \geq \cdots \geq a_n$ and we denote the system by $(a_1, a_2, \ldots, a_n)$.
If policy ψ is applied then for $t \in \mathbb{N}$ we define $W_t = W_t(\psi)$ as the waiting time of the t-th arriving job, that is the time between arrival and begin of the serving process of the t-th arriving job. We define the long-run average waiting time if policy ψ is applied as

$$W = W(\psi) = W(a_1, a_2, \ldots, a_n, \psi) = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=1}^{\tau} W_t.$$

Further if policy ψ is applied then for $t \in \mathbb{N}$ we define $V_t = V_t(\psi)$ as the sojourn time of the t-th arriving job, that is the time between arrival and end of the serving process of the t-th arriving job. We define the long-run average sojourn time if policy ψ is applied as

$$V = V(\psi) = V(a_1, a_2, \ldots, a_n, \psi) = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=1}^{\tau} V_t.$$

Our goal is to find routing policies which minimize the long-run average waiting time W. Such a policy also minimizes the long-run average sojourn time V. We introduce some more notation. Define for all $s \in \mathbb{Z}_{\geq 0}$ the variables

u_i^s = the total amount of time units that server i has not been used between $t = 0$ and $t = s$.

v_i^s = the total amount of time units from $t = s$ that server i needs to finish jobs that have been routed to server i before time $t = s$.

$k_s = k_s(\psi)$ is the server to which the job arriving at time $t = s$ is routed if policy ψ is applied.

We have the following lemma.

Lemma 2.1 *If policy ψ is applied we have*

$$W = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t(\psi)}^t \tag{3}$$

and

$$V = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} (v_{k_t(\psi)}^t + a_{k_t(\psi)}^{-1}). \tag{4}$$

Proof. Note for $t \in \mathbb{N}$ that W_t, V_t are resp. the waiting and sojourn time of the job arriving at moment $t-1$. Hence $W_t = v_{k_{t-1}(\psi)}^{t-1}$ and $V_t = W_t + a_{k_{t-1}(\psi)}^{-1}$. Thus

$$W = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=1}^{\tau} W_t = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=1}^{\tau} v_{k_{t-1}(\psi)}^{t-1} = \limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t(\psi)}^t$$

and

$$V = \limsup_{\tau\to\infty} \frac{1}{\tau}\sum_{t=1}^{\tau} V_t = \limsup_{\tau\to\infty} \frac{1}{\tau}\sum_{t=1}^{\tau}(v^{t-1}_{k_{t-1}(\psi)} + a^{-1}_{k_{t-1}(\psi)}) =$$

$$\limsup_{\tau\to\infty} \frac{1}{\tau}\sum_{t=0}^{\tau-1}(v^{t}_{k_{t}(\psi)} + a^{-1}_{k_{t}(\psi)}).$$

□

Further we define

$$\widetilde{W} = \widetilde{W}(a_1, a_2, \cdots, a_n) = \inf_{\psi} W(\psi)$$

as the minimal long-run average waiting time for the the given service rates and

$$\widetilde{V} = \widetilde{V}(a_1, a_2, \cdots, a_n) = \inf_{\psi} V(\psi)$$

as the minimal long-run average sojourn time for the the given service rates.

3 The relation between the waiting time and the unused capacity

In this section we will find a relation between W and the total amount of unused work capacity S of the system as $t \to \infty$. First we prove the following relation between the u_i and the v_i.

Lemma 3.1 *For all $t \in \mathbb{Z}_{\geq 0}$ we have*

$$\sum_{i=1}^{n} a_i \cdot u_i^t = \sum_{i=1}^{n} a_i \cdot v_i^t.$$

Proof. The proof is done by induction on t. If $t = 0$, then $u_1 = u_2 = \cdots = u_n = v_1 = v_2 = \cdots = v_n = 0$ and so the relation holds trivially. Assume that the relation holds for $t' := t - 1$. Further we assume that at time t' the incoming job is routed to server l. Then

$$u_l^t = u_l^{t'} \text{ and } v_l^t = v_l^{t'} + a_l^{-1} - 1. \tag{5}$$

Moreover for all $k \neq l$ we have

$$u_k^t = u_k^{t'} + \max(0, 1 - v_k^{t'}) \text{ and } v_k^t = \max(0, v_k^{t'} - 1) = v_k^{t'} + \max(-v_k^{t'}, -1). \tag{6}$$

So, by the induction hypothesis,

$$\sum_{i=1}^{n} a_i \cdot v_i^t - \sum_{i=1}^{n} a_i \cdot u_i^t = \sum_{i=1}^{n} a_i \cdot (v_i^t - v_i^{t'}) - \sum_{i=1}^{n} a_i \cdot (u_i^t - u_i^{t'}) =$$

$$a_l(v_l^t - v_l^{t'}) - a_l(u_l^t - u_l^{t'}) + \sum_{k \neq l} a_k \left\{ (v_k^t - v_k^{t'}) - (u_k^t - u_k^{t'}) \right\} =$$

$$1 - a_l - \sum_{k \neq l} a_k = 0$$

and the induction step is complete. $\square$

Remark that for $t \in \mathbb{Z}_{\geq 0}$

$$S^t := \sum_{i=1}^{n} a_i \cdot u_i^t$$

represents the total unused work capacity until time t, and $\sum_{i=1}^{n} a_i \cdot v_i^t$ the total remaining workload measured in amount of jobs at time t. So, we have proved that $S^t = \sum_{i=1}^{n} a_i \cdot v_i^t$ is equal to the total remaining workload in jobs at time t. Further S^t is monotonically non-decreasing in t, because the u_i^t are monotonically non-decreasing for $i \in \{1, 2, \ldots, n\}$. For a policy ψ we define the total unused work capacity S as follows:

$$S = S(\psi) = S(a_1, a_2, \ldots, a_n, \psi) = \lim_{t \to \infty} S^t.$$

Thus we have

$$S = \sum_{i=1}^{n} \lim_{t \to \infty} a_i u_i^t = \lim_{t \to \infty} \sum_{i=1}^{n} a_i u_i^t = \lim_{t \to \infty} \sum_{i=1}^{n} a_i v_i^t. \tag{7}$$

We define

$$\widetilde{S} = \widetilde{S}(a_1, a_2, \ldots, a_n) = \inf_{\psi} S(\psi)$$

as the minimal total unused work capacity for the given service rates.

The following fundamental relation exists between W and S.

Theorem 3.2 *For all $(a_1, a_2, \ldots, a_n)$ systems and policies ψ it holds that $W < \infty$ if and only if $S < \infty$. Further if $W < \infty$ then $\lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t$ exists and*

$$W = \lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t = S - \frac{n-1}{2}.$$

Before we prove Theorem 3.2 we derive some lemmata.

Lemma 3.3 *Let $f : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ be bounded. Suppose there exist $H \subseteq \mathbb{Z}_{\geq 0}$ and $a, b \in \mathbb{R}_{>0}$ such that*

$$\begin{array}{lll} f(n+1) - f(n) = a & \textit{for} & n \in H, \\ f(n+1) - f(n) = -b & \textit{for} & n \in \mathbb{Z}_{\geq 0} \setminus H. \end{array}$$

Let $H_N = \#(n \in \mathbb{Z}_{\geq 0} : n < N, n \in H)$,

$$A_N = \begin{cases} 0 & \textit{if } H_N = N \\ \frac{1}{N-H_N} \cdot \sum_{\{n \notin H, n < N\}} f(n) & \textit{if } H_N < N \end{cases} \quad \textit{and}$$

$$B_N = \begin{cases} 0 & \textit{if } H_N = 0 \\ \frac{1}{H_N} \cdot \sum_{\{n \in H, n < N\}} f(n) & \textit{if } H_N > 0 \end{cases}$$

for $N = 1, 2, \dots$. *Then*

$$\lim_{N \to \infty} (A_N - B_N) = \frac{a+b}{2}.$$

Proof. Because f is bounded, $\lim_{N\to\infty} H_N = \infty$ and $\lim_{N\to\infty}(N - H_N) = \infty$. So there exists an $N_0 \in \mathbb{N}$ such that $A_N = \frac{1}{N-H_N} \cdot \sum_{\{n \notin H, n<N\}} f(n)$ and $B_N = \frac{1}{H_N} \cdot \sum_{\{n \in H, n<N\}} f(n)$ for $N \geq N_0$. Let $\widetilde{f} : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be the continuous piecewise linear extension of f, $\widetilde{f}(x) = f(\lfloor x \rfloor) + (x - \lfloor x \rfloor) \cdot (f(\lfloor x \rfloor + 1) - f(\lfloor x \rfloor))$ for $x \in \mathbb{R}_{\geq 0}$. Let

$$C_N = \frac{1}{N - H_N} \cdot \sum_{\{n \notin H, n<N\}} \int_n^{n+1} \widetilde{f}(t)dt \text{ and}$$

$$D_N = \frac{1}{H_N} \cdot \sum_{\{n \in H, n<N\}} \int_n^{n+1} \widetilde{f}(t)dt$$

for $N \geq N_0$. If $n \in H$ then $\int_n^{n+1} \widetilde{f}(t)dt = \int_n^{n+1}(f(n) + a \cdot (t-n))dt = f(n) + \frac{a}{2}$, hence $D_N = B_N + \frac{a}{2}$ for $N \geq N_0$. If $n \in \mathbb{Z}_{\geq 0} \setminus H$ then

$$\int_n^{n+1} \widetilde{f}(t)dt = \int_n^{n+1} (f(n) - b \cdot (t-n))dt = f(n) - \frac{b}{2},$$

hence $C_N = A_N - \frac{b}{2}$ for $N \geq N_0$. So it suffices to prove that $\lim_{N\to\infty}(C_N - D_N) = 0$.

Because f is bounded, we can choose $m, M \in \mathbb{R}$ such that $f(n) \in [m, M]$ for all $n \in \mathbb{Z}_{\geq 0}$. Put $P_N = \{f(0), f(1), \dots, f(N-1)\}$ for $N = 1, 2, \dots$. Define step functions $g_N^- : [m, M] \to \mathbb{Z}_{\geq 0}$ and $g_N^+ : [m, M] \to \mathbb{Z}_{\geq 0}$ by

$$g_N^-(x) = \begin{cases} 0 & \text{if } x \in P_N \\ \#\{t \in [0, N) : \widetilde{f}(t) = x, \widetilde{f}'(t) < 0\} & \text{else} \end{cases}$$

and

$$g_N^+(x) = \begin{cases} 0 & \text{if } x \in P_N \\ \#\{t \in [0, N) : \widetilde{f}(t) = x, \widetilde{f}'(t) > 0\} & \text{else} \end{cases}.$$

If $n \in H$ then

$$\int_{f(n)}^{f(n+1)} x \cdot dx = \int_{f(n)}^{f(n)+a} x \cdot dx = a \cdot \int_n^{n+1} \widetilde{f}(t)dt,$$

whence $\int_m^M x \cdot g_N^+(x)dx = \sum_{n\in H, n<N} \int_{f(n)}^{f(n+1)} xdx = a \cdot \sum_{\{n\in H, n<N\}} \int_n^{n+1} \widetilde{f}(t)dt$ for $N = 1, 2, \ldots$. Similarly if $n \in H$ we have

$$\int_m^M g_N^+(x)dx = \sum_{n\in H, n<N} \int_{f(n)}^{f(n+1)} dx = a \cdot H_N.$$

So

$$D_N = \frac{\int_m^M x \cdot g_N^+(x)dx}{\int_m^M g_N^+(x)dx} \text{ if } N \geq N_0. \tag{8}$$

Analogously it follows that

$$C_N = \frac{\int_m^M x \cdot g_N^-(x)dx}{\int_m^M g_N^-(x)dx} \text{ if } N \geq N_0. \tag{9}$$

Since $\widetilde{f}$ is continuous, $|g_N^-(x) - g_N^+(x)| \leq 1$ for $N = 1, 2, \ldots$ and $x \in [m, M]$. Hence

$$|\int_m^M x \cdot g_N^-(x)dx - \int_m^M x \cdot g_N^+(x)dx| \leq$$

$$\int_m^M |x|dx \leq \max(m^2, M^2) \tag{10}$$

and

$$|\int_m^M g_N^-(x)dx - \int_m^M g_N^+(x)dx| \leq M - m \tag{11}$$

for $N = 1, 2, \ldots$. Further

$$\lim_{N\to\infty} \int_m^M g_N^-(x)dx = \lim_{N\to\infty} b \cdot (N - H_N) = \infty \tag{12}$$

and

$$\lim_{N\to\infty} \int_m^M g_N^+(x)dx = \lim_{N\to\infty} aH_N = \infty. \tag{13}$$

From (8) and (9) it follows that

$$C_N - D_N = \frac{\int_m^M x \cdot g_N^-(x)dx - \int_m^M x \cdot g_N^+(x)dx}{\int_m^M g_N^-(x)dx} +$$

$$\frac{\frac{\int_m^M x\cdot g_N^+(x)dx}{\int_m^M g_N^+(x)dx} \cdot (\int_m^M g_N^+(x)dx - \int_m^M g_N^-(x)dx)}{\int_m^M g_N^-(x)}. \tag{14}$$

From (10) and (12) we obtain

$$\lim_{N\to\infty} \frac{\int_m^M x \cdot g_N^-(x)dx - \int_m^M x \cdot g_N^+(x)dx}{\int_m^M g_N^-(x)dx} = 0.$$

So, by (8),(11) and (14) we have

$$|\lim_{N\to\infty}(C_N - D_N)| \leq (M-m) \cdot \lim_{N\to\infty} \frac{D_N}{\int_m^M g_N^-(x)dx}. \tag{15}$$

Because $m \leq D_N \leq M$ for $N \geq N_0$ it follows from (12) and (15) that

$$\lim_{N\to\infty}(C_N - D_N) = 0.$$

□

Corollary 3.4 *Let f, H, a, b be as in Lemma 3.3. Further for $N = 1, 2, \ldots$ let H_N, A_N, B_N as in Lemma 3.3 and let*

$$E_N = \frac{1}{N} \cdot \sum_{n=0}^{N-1} f(n).$$

Then

$$\lim_{N\to\infty}(E_N - B_N) = \frac{a}{2}.$$

Proof. Since f is bounded we have that $\lim_{N\to\infty} \frac{H_N}{N-H_N} = \frac{b}{a}$. Hence $\lim_{N\to\infty} \frac{N}{N-H_N} = \frac{a+b}{a}$ and thus

$$\lim_{N\to\infty} \frac{N-H_N}{N} = \frac{a}{a+b}. \tag{16}$$

So, by Lemma 3.3 and (16) we have that

$$\lim_{N\to\infty}(E_N - B_N) = \lim_{N\to\infty}(\frac{N-H_N}{N} \cdot A_N + \frac{H_N}{N} \cdot B_N - B_N) = \lim_{N\to\infty} \frac{N-H_N}{N}(A_N - B_N) =$$

$$\lim_{N\to\infty} \frac{N-H_N}{N} \cdot \lim_{N\to\infty}(A_N - B_N) = \frac{a}{a+b} \cdot \frac{a+b}{2} = \frac{a}{2}.$$

□

For $t \in \mathbb{Z}_{\geq 0}$ put $I_t = \{0, 1, \ldots, t-1\}$. Then if policy ψ is applied we define for $i \in \{1, 2, \ldots, n\}$ and $t \in \mathbb{Z}_{\geq 0}$ that

$$N_i^t = N_i^t(\psi) = \sum_{\{t' \in I_t : k_{t'}(\psi) = i\}} 1.$$

Hence N_i^t is the number of jobs among the first t incoming jobs that are routed to server i.

Lemma 3.5 *If in an $(a_1, a_2, \ldots, a_n)$ system a policy ψ is applied such that for some $i \in \{1, 2, \ldots, n\}$ the function $f : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ defined by $f(t) = v_i^t - u_i^t$ is bounded then*

$$\lim_{\tau\to\infty}\{\frac{\sum_{t=0}^{\tau-1} v_i^t}{\tau} - \frac{\sum_{\{t\in I_\tau : k_t = i\}} v_i^t}{N_i^\tau}\} = \frac{1}{2a_i} - \frac{1}{2}.$$

Proof. If $k_t = i$ then we have by (5) that

$$f(t+1) - f(t) = \frac{1}{a_i} - 1.$$

If $k_t \neq i$ then we have by (6) that

$$f(t+1) - f(t) = -1.$$

So f satisfies the conditions of Corollary 3.4 with $H = \{t : k_t = i\}$, $a = \frac{1}{a_i} - 1$ and $b = 1$. Hence

$$\lim_{\tau\to\infty}\{\frac{\sum_{t=0}^{\tau-1} (v_i^t - u_i^t)}{\tau} - \frac{\sum_{\{t\in I_\tau : k_t = i\}} (v_i^t - u_i^t)}{N_i^\tau}\} = \frac{1}{2a_i} - \frac{1}{2}. \quad (17)$$

Since $u_i^{t+1} > u_i^t$ implies that $v_i^{t+1} = 0$ it follows from the boundedness of f that $\lim_{t\to\infty} u_i^t =: L < \infty$ and thus

$$\lim_{\tau\to\infty} \frac{\sum_{t=0}^{\tau-1} u_i^t}{\tau} = \lim_{\tau\to\infty} \frac{\sum_{\{t\in I_\tau : k_t = i\}} u_i^t}{N_i^\tau} = L. \quad (18)$$

By (17) and (18) we have that

$$\lim_{\tau\to\infty}\{\frac{\sum_{t=0}^{\tau-1} v_i^t}{\tau} - \frac{\sum_{\{t\in I_\tau : k_t = i\}} v_i^t}{N_i^\tau}\} = \frac{1}{2a_i} - \frac{1}{2}.$$

□

Corollary 3.6 *If in an $(a_1, a_2, \ldots, a_n)$ system a policy ψ is applied with $S(\psi) < \infty$ then we have for every server $i \in \{1, 2, \ldots, n\}$ that*

$$\lim_{\tau\to\infty}\{\frac{\sum_{t=0}^{\tau-1} v_i^t}{\tau} - \frac{\sum_{\{t\in I_\tau : k_t = i\}} v_i^t}{N_i^\tau}\} = \frac{1}{2a_i} - \frac{1}{2}.$$

Proof. Since $S < \infty$ we have for every $i \in \{1, 2, \ldots, n\}$ that $\lim_{t\to\infty} a_i \cdot u_i^t := L_i \leq S < \infty$. It follows that $a_i \cdot v_i^t \leq S$ and $a_i \cdot u_i^t \leq L_i$ for every $t \in \mathbb{Z}_{\geq 0}$. Define $f_i : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ by $f_i(t) = v_i^t - u_i^t$ as in Lemma 3.5. Then $f_i(t) \in [-\frac{L_i}{a_i}, \frac{S}{a_i}]$ for every $t \in \mathbb{Z}_{\geq 0}$ and thus f_i is bounded. Now apply Lemma 3.5 . □

If $\lim_{t\to\infty} \frac{N_i^t}{t}$ exists for $i \in \{1, 2, \ldots, n\}$ then we define

$$p_i = \lim_{t\to\infty} \frac{N_i^t}{t}$$

as the fraction of jobs that is routed to server i.

Lemma 3.7 *For every $(a_1, a_2, \dots, a_n)$ system and policy ψ we have $S < \infty$ if and only if $W < \infty$. Further if $W < \infty$ then $\lim_{t\to\infty} \frac{N_i^t}{t}$ exists for all $i \in \{1, 2, \dots, n\}$ and*

$$p_i = \lim_{t\to\infty} \frac{N_i^t}{t} = a_i \tag{19}$$

for $i = 1, 2, \dots, n$.

Proof. Suppose $S < \infty$. Then there exists $M_0 \in \mathbb{R}$ such that $\sum_{i=1}^n a_i v_i^t < M_0$ for $t \in \mathbb{Z}_{\geq 0}$. It follows that $a_i v_i^t < M_0$ for $i = 1, 2, \dots, n$ and $t \in \mathbb{Z}_{\geq 0}$ and thus $v_i^t < M := \frac{M_0}{\min_{i=1}^n a_i}$. Hence $v_{k_t}^t < M$ for all $t \in \mathbb{Z}_{\geq 0}$ and thus $W \leq M$.
Suppose $S = \infty$. Let $L(t)$ be the total number of waiting jobs at time $t \in \mathbb{R}_{\geq 0}$. It is clear that $L(t) \geq S^{\lfloor t \rfloor} - n$ for every $t \in \mathbb{R}_{\geq 0}$ and thus $\lim_{t\to\infty} L(t) = \infty$. So the limiting time-average number of waiting jobs satisfies

$$L := \lim_{t\to\infty} \frac{1}{t} \cdot \int_0^t L(t)dt = \infty. \tag{20}$$

Let $J_k(t) = 1$ for $t \in \mathbb{R}_{\geq 0}$, if the k-th arriving job is waiting in one of the queues at time t and else $J_k(t) = 0$. Then,

$$W_k = \int_0^\infty J_k(t)dt, \tag{21}$$

$$L(t) = \sum_{k=1}^\infty J_k(t). \tag{22}$$

Let $U(t) := \sum_{k=1}^{t+1} W_k$ be the sum of the waiting times of jobs arriving in $[0, t]$ for $t \in \mathbb{R}_{\geq 0}$. Then for all $T \in \mathbb{R}_{\geq 0}$ we have by (21) and (22)

$$\int_0^T L(t)dt = \int_0^T \sum_{k=1}^\infty J_k(t)dt = \sum_{k=1}^\infty \int_0^T J_k(t)dt =$$

$$\sum_{k=1}^{T+1} \int_0^T J_k(t)dt \leq \sum_{k=1}^{T+1} \int_0^\infty J_k(t)dt = \sum_{k=1}^{T+1} W_k = U(T). \tag{23}$$

Hence by (20) and (23) we have

$$\lim_{t\to\infty} \frac{1}{t} \cdot U(t) = \infty.$$

Thus

$$W = \limsup_{t\to\infty} \frac{1}{t} \cdot \sum_{k=1}^t W_k = \limsup_{t\to\infty} \frac{1}{\lfloor t \rfloor + 1} \cdot U(t) = \lim_{t\to\infty} \frac{1}{t} \cdot U(t) = \infty.$$

The first part of the lemma is proved and for the second part we assume that $W < \infty$ and thus $S < \infty$. Let $\overline{p_i} = \limsup_{t\to\infty} \frac{N_i^t}{t}$ for $i \in \{1,2,\dots,n\}$. We will show that $\overline{p_i} = a_i$ for $i \in \{1,2,\dots,n\}$. We have that

$$\sum_{i=1}^{n} \overline{p_i} = \sum_{i=1}^{n} \limsup_{t\to\infty} \frac{N_i^t}{t} \geq \limsup_{t\to\infty} \sum_{i=1}^{n} \frac{N_i^t}{t} = 1.$$

Hence if there exists an $i \in \{1,2,\dots,n\}$ with $\overline{p_i} \neq a_i$ then it follows that there exists a $k \in \{1,2,\dots,n\}$ with $\overline{p_k} > a_k$. Let $a_k < \beta < \overline{p_k}$. Then for all $t_0 \in \mathbb{Z}_{\geq 0}$ there exist $t \geq t_0$ such that $\frac{N_k^t}{t} > \beta$ and thus $N_k^t > \beta \cdot t = a_k \cdot t + t \cdot (\beta - a_k)$. Hence

$$a_k \cdot v_k^t \geq N_k^t - a_k \cdot t > t \cdot (\beta - a_k) \geq t_0 \cdot (\beta - a_k).$$

Thus $v_k^t \geq \frac{\beta - a_k}{a_k} \cdot t_0$ and it follows that $\limsup_{t\to\infty} v_k^t = \infty$ and thus $S = \infty$, which is a contradiction. So $\overline{p_i} = a_i$ for $i \in \{1,2,\dots,n\}$. Let $\underline{p_i} = \liminf_{t\to\infty} \frac{N_i^t}{t}$ for $i \in \{1,2,\dots,n\}$. We have to show that $\underline{p_i} = a_i$ for $i \in \{1,2,\dots,n\}$. Analogously as above it follows that $\sum_{i=1}^{n} \underline{p_i} \leq 1$. Thus if $\underline{p_i} \neq a_i$ for an $i \in \{1,2,\dots,n\}$ then there exist a $k \in \{1,2,\dots,n\}$ such that $\underline{p_k} < a_k$. Let $\underline{p_k} < \beta < a_k$. Then for all $t_0 \in \mathbb{Z}_{\geq 0}$ there exist $t \geq t_0$ such that $\frac{N_k^t}{t} < \beta$ and thus $N_k^t < \beta \cdot t = a_k \cdot t - t \cdot (a_k - \beta)$. Hence

$$a_k \cdot u_k^t \geq a_k \cdot t - N_k^t \geq t_0 \cdot (a_k - \beta).$$

Thus $u_k^t \geq \frac{a_k - \beta}{a_k} \cdot t_0$ and it follows that $\limsup_{t\to\infty} u_k^t = \lim_{t\to\infty} u_k^t = \infty$ and thus $S = \infty$, which is a contradiction. So, if $S < \infty$ then $\underline{p_i} = \overline{p_i} = a_i$ for $i \in \{1,2,\dots,n\}$. Hence $\lim_{t\to\infty} \frac{N_i^t}{t}$ exists and $p_i = \lim_{t\to\infty} \frac{N_i^t}{t} = a_i$ for $i \in \{1,2,\dots,n\}$. $\square$

Remark. The argument that $W < \infty$ implies $S < \infty$ is a simplification of a proof of Little's theorem by Stidham in [16].

Proof of Theorem 3.2.

The first part of the theorem follows from Lemma 3.7. We now assume that $S < \infty$. From (19) and Corollary 3.6 it follows for every $i \in \{1,2,\dots,n\}$ that

$$\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \Big(\sum_{t=0}^{\tau-1} a_i \cdot v_i^t - \sum_{\{t \in I_\tau : k_t = i\}} v_i^t\Big) =$$

$$a_i \cdot \lim_{\tau\to\infty} \Big\{\frac{\sum_{t=0}^{\tau-1} v_i^t}{\tau} - \frac{\sum_{\{t \in I_\tau : k_t = i\}} v_i^t}{N_i^\tau}\Big\} = a_i \cdot \Big(\frac{1}{2a_i} - \frac{1}{2}\Big) = \frac{1}{2} - \frac{1}{2} a_i. \tag{24}$$

Because S^t is monotonically non-decreasing in t and bounded it follows that

$$S = \lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} S^t = \lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} \sum_{i=1}^{n} a_i \cdot v_i^t = \lim_{\tau\to\infty} \sum_{i=1}^{n} \sum_{t=0}^{\tau-1} \frac{a_i \cdot v_i^t}{\tau}.$$

Thus it follows from (24) that

$$S - \lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t = \lim_{\tau\to\infty} \sum_{i=1}^{n} \frac{1}{\tau} \cdot (\sum_{t=0}^{\tau-1} a_i \cdot v_i^t - \sum_{\{t\in I_\tau : k_t = i\}} v_i^t) =$$

$$\sum_{i=1}^{n} (\lim_{\tau\to\infty} \frac{1}{\tau} \cdot (\sum_{t=0}^{\tau-1} a_i \cdot v_i^t - \sum_{\{t\in I_\tau : k_t = i\}} v_i^t)) = \sum_{i=1}^{n} (\frac{1}{2} - \frac{1}{2} \cdot a_i) = \frac{n-1}{2}.$$

So, $\lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t$ exists and

$$W = \limsup_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t = \lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t = S - \frac{n-1}{2}.$$

□

The following proposition shows that minimizing the long-run average waiting time W also minimizes the long-run average sojourn time V and vice versa.

Proposition 3.8 *For all $(a_1, a_2, \ldots, a_n)$ systems and policies ψ it holds that $V < \infty$ if and only if $S < \infty$ and*

$$V = W + n \textit{ if } S < \infty.$$

Proof. It is clear that $V < \infty$ if and only if $W < \infty$ and so it follows from Theorem 3.2 that $V < \infty$ if and only if $S < \infty$. Let ψ be a policy such that $S < \infty$. Then by (19) it follows that

$$\lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} a_{k_t}^{-1} = \lim_{\tau\to\infty} \sum_{i=1}^{n} a_i^{-1} \frac{N_i^\tau}{\tau} = \sum_{i=1}^{n} a_i^{-1} a_i = n.$$

Hence

$$V = \limsup_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} (v_{k_t}^t + a_{k_t}^{-1}) = W + \lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} a_{k_t}^{-1} = W + n.$$

□

From now on we will only consider W and not V, but from Proposition 3.8 it can be deduced what the results imply for V.

4 An upper bound for the minimal long-run average waiting time

In this section we derive an upper bound for the minimal long-run average waiting time $\widetilde{W}$. Further we show that for every $(a_1, a_2, \ldots, a_n)$ system an optimal policy exists and we give a MPP such that from an optimal solution of the MPP an optimal policy can be obtained and vice versa.

Lemma 4.1 *For all $s \in \mathbb{Z}_{\geq 0}$ we have*

$$u_i^s = \max_{t \in I_{s+1}} (u_i^t - v_i^t) = \max_{t \in I_{s+1}} (t - a_i^{-1} \cdot N_i^t).$$

Proof. The proof of the first equality is done by induction on s. It holds for $s = 0$. Suppose that it holds for $s' = s$. If $k_s = i$ then by (5) we have $u_i^{s+1} = u_i^s$ and $v_i^{s+1} > v_i^s$. So, by the induction hypothesis,

$$u_i^{s+1} = u_i^s = \max_{t \in I_{s+1}} (u_i^t - v_i^t) = \max_{t \in I_{s+2}} (u_i^t - v_i^t).$$

If $k_s \neq i$ and $v_i^s \leq 1$, then by (6) we have $u_i^{s+1} \geq u_i^s$ and $v_i^{s+1} = 0$. Thus, by the induction hypothesis,

$$u_i^{s+1} - v_i^{s+1} \geq u_i^s = \max_{t \in I_{s+1}} (u_i^t - v_i^t).$$

Hence $u_i^{s+1} = u_i^{s+1} - v_i^{s+1} = \max_{t \in I_{s+2}} (u_i^t - v_i^t)$. Further if $k_s \neq i$ and $v_i^s > 1$ then by (6) we have $u_i^{s+1} = u_i^s$ and $v_i^{s+1} = v_i^s - 1 \geq 0$. So, by the induction hypothesis,

$$u_i^{s+1} - v_i^{s+1} \leq u_i^s = \max_{t \in I_{s+1}} (u_i^t - v_i^t).$$

Hence

$$u_i^{s+1} = u_i^s = \max_{t \in I_{s+1}} (u_i^t - v_i^t) = \max_{t \in I_{s+2}} (u_i^t - v_i^t)$$

and the induction step is complete.

For the second equality it suffices to show that $u_i^t - v_i^t = t - a_i^{-1} \cdot N_i^t$ for all $t \in \mathbb{Z}_{\geq 0}$. Because for all $t \in \mathbb{Z}_{\geq 0}$ we have

$$(u_i^{t+1} - v_i^{t+1}) - (u_i^t - v_i^t) = \begin{cases} 1 - a_i^{-1} & \text{if } k_t = i \\ 1 & \text{if } k_t \neq i \end{cases} \tag{25}$$

it follows by induction on t that

$$u_i^t - v_i^t = u_i^0 - v_i^0 + N_i^t \cdot (1 - a_i^{-1}) + (t - N_i^t) \cdot 1 = t - a_i^{-1} \cdot N_i^t \tag{26}$$

for all $t \in \mathbb{Z}_{\geq 0}$. $\square$

We define for $i \in \{1, 2, \ldots, n\}$ the variables

$$d_i(\psi) := \lim_{t \to \infty} a_i \cdot u_i^t,$$

the total amount of unused work capacity of server i if policy ψ is applied. It is clear that

$$S = \sum_{i=1}^{n} d_i(\psi). \tag{27}$$

Lemma 4.2 *For every $(a_1, a_2, \ldots, a_n)$ system and applied policy ψ we have*

$$d_i(\psi) = \sup_{t \in \mathbb{Z}_{\geq 0}} (a_i \cdot t - N_i^t)$$

for all $i \in \{1, 2, \ldots, n\}$.

Proof. From Lemma 4.1 it follows that

$$d_i(\psi) = \lim_{s \to \infty} a_i \cdot u_i^s = \lim_{s \to \infty} \max_{t \in I_{s+1}} (a_i \cdot t - N_i^t) = \sup_{t \in \mathbb{Z}_{\geq 0}} (a_i \cdot t - N_i^t).$$

□

Let $d_i \in \mathbb{R}_{\geq 0}$ be given. It follows from Lemma 4.2 for a policy ψ that $d_i(\psi) \leq d_i$ if and only if $t \cdot a_i - N_i^t(\psi) \leq d_i$ for all $t \in \mathbb{Z}_{\geq 0}$ and this is true if and only if $N_i^t(\psi) \geq t \cdot a_i - d_i$ for all $t \in \mathbb{Z}_{\geq 0}$. Hence $N_i^t(\psi) \geq \max(\lceil t \cdot a_i - d_i \rceil, 0)$ for all $t \in \mathbb{Z}_{\geq 0}$, because $N_i^t(\psi) \in \mathbb{Z}_{\geq 0}$. So suppose that for given $d_1, d_2, \ldots, d_n \in \mathbb{R}_{\geq 0}$ there exists a policy ψ with $d_i(\psi) = d_i$ for $i = 1, 2, \ldots, n$. Then

$$t = \sum_{i=1}^{n} N_i^t(\psi) \geq \sum_{i=1}^{n} \max(\lceil t \cdot a_i - d_i \rceil, 0) \text{ for } t = 0, 1, \ldots.$$

Thus

$$t \geq \sum_{i=1}^{n} \max(\lceil t \cdot a_i - d_i \rceil, 0) \text{ for } t = 0, 1, \ldots \tag{28}$$

is a necessary condition for the existence of a policy ψ such that $d_i(\psi) = d_i$ for $i = 1, 2, \ldots, n$ and we have proved the following lemma.

Lemma 4.3 *If a policy ψ is applied to an $(a_1, a_2, \ldots, a_n)$ system then*

$$t \geq \sum_{i=1}^{n} \max(\lceil t \cdot a_i - d_i(\psi) \rceil, 0) \textit{ for } t = 0, 1, \ldots.$$

On the other hand, (28) is not sufficient for the existence of a policy ψ with $d_i(\psi) = d_i$ for $i \in \{1, 2, \ldots, n\}$. For example, if $a_i = \frac{p}{q}$ with $\gcd(p, q) = 1$ then, by Lemma 4.2, $q \cdot d_i(\psi) = q \cdot \sup_{t \in \mathbb{Z}_{\geq 0}} (a_i \cdot t - N_i^t) \in \mathbb{Z}$. So, in that case $q \cdot d_i \in \mathbb{Z}$ is another necessary condition. However, we will show that there exists a policy ψ with $d_i(\psi) \leq d_i$ for $i = 1, 2, \ldots, n$ if $d_1, d_2, \ldots, d_n$ fulfill (28). To do so we use the following greedy algorithm to construct such a policy ψ. We denote this algorithm as the GG (General Greedy) algorithm.

GG Algorithm. Let $d_1, d_2, \ldots, d_n \in \mathbb{R}_{\geq 0}$ be given. Then we determine ψ inductively in the following way. Suppose $k_0(\psi), k_1(\psi), \ldots, k_{t-1}(\psi)$ have been determined. Then for $j = 1, 2, \ldots, n$ put $q_j(t) = \min\{t' \in \mathbb{Z}_{\geq 0} : t' \cdot a_j > d_j - t \cdot a_j + N_j^t(\psi)\}$. Choose $k_t(\psi)$ such that $k_t(\psi) = i \in \{1, 2, \ldots, n\}$ for which $q_i(t) = \min_{j \in \{1,2,\ldots,n\}} q_j(t)$.

Theorem 4.4 *Let ψ be a policy obtained by applying the GG algorithm. Then we have $d_i(\psi) \leq d_i$ for $i = 1, 2, \ldots, n$ if $d_1, d_2, \ldots, d_n$ fulfill (28).*

Proof. Suppose there exists an $j \in \{1, 2, \ldots, n\}$ with $d_j(\psi) > d_j$. Then there exist $t \in \mathbb{Z}_{\geq 0}$ for which $N_j^t(\psi) < t \cdot a_j - d_j$ and thus $t \cdot a_j > d_j$ and $q_j(t) = 0$. Let $t' = \min\{t \in \mathbb{Z}_{\geq 0} : q_j(t) = 0\}$. Then $N_j^{t'}(\psi) \leq \lceil t' \cdot a_j - d_j \rceil - 1$. Thus from $\sum_{i=1}^n N_i^{t'}(\psi) = t' \geq \sum_{i=1}^n \max(\lceil t' \cdot a_i - d_i \rceil, 0)$ it follows that there exists a $k \neq j$ with $N_k^{t'}(\psi) \geq \max(\lceil t' \cdot a_k - d_k \rceil, 0) + 1$. Hence $N_k^{t'}(\psi) \geq 1$ and $N_k^{t'}(\psi) \geq t' \cdot a_k - d_k + 1$. Let $t_0 = \max\{0 \leq t < t' : k_t(\psi) = k\}$. Then we have $N_j^{t_0}(\psi) \leq N_j^{t'}(\psi) < t' \cdot a_j - d_j$ and thus $q_j(t_0) \leq t' - t_0$. Moreover $N_k^{t_0}(\psi) = N_k^{t'}(\psi) - 1 \geq t' \cdot a_k - d_k$ and thus $q_k(t_0) \geq t' - t_0 + 1 > q_j(t_0)$. This contradicts the fact that the GG algorithm is applied. $\square$

Put

$$D = D(a_1, a_2, \ldots, a_n) = \{(d_1, d_2, \ldots, d_n) \in \mathbb{R}^n : d_1, d_2, \ldots, d_n \text{ fulfill condition (28)}\}.$$

Lemma 4.5 *Let $d_1, d_2, \ldots, d_n \in \mathbb{R}$ such that*

$$\sum_{i=1}^n d_i = n - 1 \text{ and } 0 \leq d_i < 1 + a_i \text{ for } i = 1, 2, \ldots, n. \tag{29}$$

Then $(d_1, d_2, \ldots, d_n) \in D$.

Proof. Because $d_i \geq 0$ for $i = 1, 2, \ldots, n$ we have $\max(\lceil -d_i \rceil, 0) = 0$ for $i = 1, 2, \ldots, n$. Hence $\sum_{i=1}^n \max(\lceil -d_i \rceil, 0) = 0$. Further by $d_i < 1 + a_i$ we have for $t = 1, 2, \ldots$ that

$$\max(0, \lceil t \cdot a_i - d_i \rceil) = \lceil t \cdot a_i - d_i \rceil < t \cdot a_i - d_i + 1$$

for $i = 1, 2, \ldots, n$. Hence

$$\sum_{i=1}^n \max(\lceil t \cdot a_i - d_i \rceil, 0) < \sum_{i=1}^n (t \cdot a_i - d_i + 1) =$$

$$t - (n-1) + n = t + 1$$

and $\sum_{i=1}^n \max(\lceil t \cdot a_i - d_i \rceil, 0) \in \mathbb{Z}_{\geq 0}$. So, $\sum_{i=1}^n \max(\lceil t \cdot a_i - d_i \rceil, 0) \leq t$ and condition (28) is fulfilled. $\square$

Particular $d_1, d_2, \ldots, d_n$ that fulfill (29) are for example that $d_i = 0$ for an arbitrary $i \in \{1, 2, \ldots, n\}$ and $d_j = 1$ for all $j \neq i$. Another example is $d_i = 1 - a_i$ for $i = 1, 2, \ldots, n$. Hence $D \neq \emptyset$ for every $(a_1, a_2, \ldots, a_n)$ system. Further it is obvious that if $(d_1, d_2, \ldots, d_n) \in D$ then $(d_1 + e_1, d_2 + e_2, \ldots, d_n + e_n) \in D$ for every $e_1, e_2, \ldots, e_n \in \mathbb{R}_{\geq 0}$. From the condition for $t = 0$ it follows from (28) that $d_i \geq 0$ for $i = 1, 2, \ldots, n$ if $(d_1, d_2, \ldots, d_n) \in D$. Thus $\inf_{(d_1, d_2, \ldots, d_n) \in D} \sum_{i=1}^n d_i \geq 0$.

Theorem 4.6 *For every $(a_1, a_2, \ldots, a_n)$ system we have*

$$\widetilde{S} = \inf_{(d_1, d_2, \ldots, d_n) \in D} \sum_{i=1}^n d_i \leq n - 1.$$

Proof There exist $(d_1, d_2, \dots, d_n) \in D$ with $\sum_{i=1}^n d_i = n-1$ and thus we have $\inf_{(d_1,d_2,\dots,d_n)\in D} \sum_{i=1}^n d_i \le n-1$. So, it remains to prove that $\widetilde{S} = \inf_{(d_1,d_2,\dots,d_n)\in D} \sum_{i=1}^n d_i$. According to Lemma 4.3 we have for every policy ψ that $d_1(\psi), d_2(\psi), \dots, d_n(\psi)$ fulfill condition (28). Hence,

$$S(\psi) = \sum_{i=1}^n d_i(\psi) \ge \inf_{(d_1,d_2,\dots,d_n)\in D} \sum_{i=1}^n d_i$$

and thus

$$\widetilde{S} = \inf_{\psi} S(\psi) \ge \inf_{(d_1,d_2,\dots,d_n)\in D} \sum_{i=1}^n d_i.$$

On the other hand, if $(d_1, d_2, \dots, d_n) \in D$ then according to Theorem 4.4 there exists a policy ψ with $d_i(\psi) \le d_i$ for $i = 1, 2, \dots, n$ and thus $S(\psi) \le \sum_{i=1}^n d_i$. Hence,

$$\widetilde{S} = \inf_{\psi} S(\psi) \le \inf_{(d_1,d_2,\dots,d_n)\in D} \sum_{i=1}^n d_i.$$

□

From Theorem 3.2 and Theorem 4.6 we have the following corollary which gives an upper bound for the minimal long-run average waiting time $\widetilde{W}$ for every $(a_1, a_2, \dots, a_n)$ system.

Corollary 4.7 *For every $(a_1, a_2, \dots, a_n)$ system we have*

$$\widetilde{W} = \widetilde{S} - \frac{n-1}{2} \tag{30}$$

and

$$0 \le \widetilde{W} \le \frac{n-1}{2}.$$

Lemma 4.8 *$D \subseteq \mathbb{R}^n$ is closed for every $(a_1, a_2, \dots, a_n)$ system.*

Proof. For $t = 0, 1, \dots$ put

$$D^t = D^t(a_1, a_2, \dots, a_n) = \{(d_1, d_2, \dots, d_n) \in \mathbb{R}^n : t \ge \sum_{i=1}^n \max(0, \lceil t \cdot a_i - d_i \rceil)\}.$$

Then $D = \cap_{t=0}^{\infty} D_t$. So it suffices to show that $D_t \subseteq \mathbb{R}^n$ is closed for $t = 0, 1, \dots$. Let $t \in \mathbb{Z}_{\ge 0}$ be fixed and suppose $\underline{b} = (b_1, b_2, \dots, b_n) \in \mathbb{R}^n \backslash D^t$. Put $m_i = \lceil t \cdot a_i - b_i \rceil - 1$ for $i = 1, 2, \dots, n$. Then $m_i \in \mathbb{Z}$ and $t \cdot a_i - b_i = m_i + \delta_i$ with $\delta_i \in (0, 1]$ for $i = 1, 2, \dots, n$. Let $0 < \varepsilon < \min_{i=1}^n \delta_i$ and $\underline{c} = (c_1, c_2, \dots, c_n) \in \mathbb{R}^n$ with $\| \underline{c} - \underline{b} \|_\infty < \varepsilon$. Then

$$\lceil t \cdot a_i - c_i \rceil \ge \lceil t \cdot a_i - b_i - \varepsilon \rceil = \lceil m_i + \delta_i - \varepsilon \rceil =$$

$$m_i + 1 = \lceil t \cdot a_i - b_i \rceil \text{ for } i = 1, 2, \dots, n.$$

Hence,

$$\sum_{i=1}^{n} \max(\lceil t \cdot a_i - c_i \rceil, 0) \geq \sum_{i=1}^{n} \max(\lceil t \cdot a_i - b_i \rceil, 0) > t$$

and thus $\underline{c} \in \mathbb{R}^n \backslash D^t$. So, $\mathbb{R}^n \backslash D^t$ is open and $D^t \subseteq \mathbb{R}^n$ is closed for $t = 0, 1, \ldots$. □

Theorem 4.9 *For every $(a_1, a_2, \ldots, a_n)$ system we have*

$$\widetilde{S} = \min_{\psi} S(\psi). \tag{31}$$

Proof. Let $H = \{(d_1, d_2, \ldots, d_n) \in \mathbb{R}^n : \sum_{i=1}^{n} d_i \leq n-1\}$ and $D' = D \cap H$. Then D' is closed, because D and H are closed. Further $D' \neq \emptyset$, because $(1-a_1, 1-a_2, \ldots, 1-a_n) \in D'$. Let $(d_1, d_2, \ldots, d_n) \in D'$. As observed in the Proof of Theorem 4.6 we have $d_i \geq 0$ for $i = 1, 2, \ldots, n$. So, $0 \leq d_i \leq n-1$ for $i = 1, 2, \ldots, n$ and thus D' is bounded. Hence D' is compact. Because a continuous function assumes a minimum value on a compact set there exists $(d'_1, d'_2, \ldots, d'_n) \in D'$ for which

$$\sum_{i=1}^{n} d'_i = \min_{(d_1, d_2, \ldots, d_n) \in D'} \sum_{i=1}^{n} d_i = \min_{(d_1, d_2, \ldots, d_n) \in D} \sum_{i=1}^{n} d_i. \tag{32}$$

Because $(d'_1, d'_2, \ldots, d'_n) \in D$ it follows from Theorem 4.4 that there exists a policy ψ' with $d_i(\psi') \leq d'_i$ for $i = 1, 2, \ldots, n$. So, if $d_i(\psi') \neq d'_i$ for some $i \in \{1, 2, \ldots, n\}$ then $\sum_{i=1}^{n} d_i(\psi') < \sum_{i=1}^{n} d'_i$. This contradicts the minimality of $\sum_{i=1}^{n} d'_i$, because $(d_1(\psi'), d_2(\psi'), \ldots, d_n(\psi')) \in D'$. Hence $d_i(\psi') = d'_i$ for $i = 1, 2, \ldots, n$ and according to (32) and Theorem 4.6 we have

$$S(\psi') = \sum_{i=1}^{n} d_i(\psi') = \sum_{i=1}^{n} d'_i = \inf_{(d_1, d_2, \ldots, d_n) \in D} \sum_{i=1}^{n} d_i = \widetilde{S}.$$

□

We define a policy ψ to be optimal for a given $(a_1, a_2, \ldots, a_n)$ system if and only if $S(\psi) = \widetilde{S}$ and thus $W(\psi) = \widetilde{W}$. Consider the following MPP.

$$\begin{array}{lcl} \text{minimize } S & = & \sum_{i=1}^{n} d_i \\ \text{subject to:} & & \\ \sum_{i=1}^{n} \max(0, \lceil t \cdot a_i - d_i \rceil) & \leq & t \text{ for } t = 0, 1, \ldots \\ d_i & \in & \mathbb{R}_{\geq 0} \text{ for } i = 1, 2, \ldots, n \end{array} \tag{33}$$

It follows from Theorem 4.4,Theorem 4.6 and Theorem 4.9 that for every $(a_1, a_2, \ldots, a_n)$ system an optimal solution of (33) exists and that from an optimal solution of (33) an optimal policy can be obtained by applying the GG algorithm to that solution. Conversely, if ψ is an optimal policy then $d_1(\psi), d_2(\psi), \ldots, d_n(\psi)$ give an optimal solution of (33). Further, suppose that

$d_1, d_2, \ldots, d_n$ fulfill the constraints of (33). Then $(d_1, d_2, \ldots, d_n) \in D$ and a policy ψ can be obtained with $d_i(\psi) \leq d_i$ for $i = 1, 2, \ldots, n$ by applying the GG algorithm. Then we have

$$\widetilde{S} \leq S(\psi) = \sum_{i=1}^{n} d_i(\psi) \leq \sum_{i=1}^{n} d_i. \tag{34}$$

Moreover the $d_i(\psi)$ also fulfill the constraints of (33). So the same procedure can be applied to the $d_i(\psi)$. So, this may be a way to obtain iteratively better policies and in section 6 we will give an example of this.

5 The structure of optimal policies

Let ψ be a given policy. Then for all $s \in \mathbb{Z}_{\geq 0}$ we define the policy ψ^s by $k_t(\psi^s) = k_{t+s}(\psi)$ for $t = 0, 1, \ldots$. Thus ψ^s is a shift of policy ψ for $s = 0, 1, \ldots$.

Lemma 5.1 *Let ψ be a policy with $k_0(\psi) = j$. Then we have $d_j(\psi^1) \leq d_j(\psi) + 1 - a_j$ and $d_i(\psi^1) = d_i(\psi) - a_i$ for all $i \neq j$.*

Proof. By Lemma 4.2 we have that

$$d_j(\psi^1) = \sup_{t \in \mathbb{Z}_{\geq 0}} (t \cdot a_j - N_j^t(\psi^1)) = \sup_{t \in \mathbb{Z}_{\geq 0}} (t \cdot a_j - (N_j^{t+1}(\psi) - 1)) =$$

$$\sup_{t \in \mathbb{Z}_{\geq 1}} (t \cdot a_j - N_j^t(\psi)) + (1 - a_j) \leq \sup_{t \in \mathbb{Z}_{\geq 0}} (t \cdot a_j - N_j^t(\psi)) + (1 - a_j) = d_j(\psi) + 1 - a_j.$$

Further for every $i \neq j$ we have $\sup_{t \in \mathbb{Z}_{\geq 1}} (t \cdot a_i - N_i^t(\psi)) \geq a_i$ and thus by Lemma 4.2 we have

$$d_i(\psi^1) = \sup_{t \in \mathbb{Z}_{\geq 0}} (t \cdot a_i - N_i^t(\psi^1)) = \sup_{t \in \mathbb{Z}_{\geq 1}} ((t-1) \cdot a_i - N_i^t(\psi)) =$$

$$\sup_{t \in \mathbb{Z}_{\geq 0}} (t \cdot a_i - N_i^t(\psi)) - a_i = d_i(\psi) - a_i.$$

□

Corollary 5.2 *For every policy ψ we have $S(\psi^t) \leq S(\psi)$ for every $t \in \mathbb{Z}_{\geq 0}$.*

Proof. By Lemma 5.1 we have

$$S(\psi^1) = \sum_{i=1}^{n} d_i(\psi^1) \leq \sum_{i=1}^{n} (d_i(\psi) - a_i) + 1 = \sum_{i=1}^{n} d_i(\psi) = S(\psi).$$

By induction we have $S(\psi) \geq S(\psi^1) \geq S(\psi^2) \geq \ldots$. □

If policy ψ is applied in an $(a_1, a_2, \dots, a_n)$ system then for every $t \in \mathbb{Z}_{\geq 0}$ and $i \in \{1, 2, \dots, n\}$ we put

$$h_i^t = h_i^t(\psi) = d_i(\psi) - t \cdot a_i + N_i^t(\psi).$$

Observe that by Lemma 4.2 we have $h_i^t(\psi) \geq 0$ for every policy ψ, $t \in \mathbb{Z}_{\geq 0}$ and $i \in \{1, 2, \dots, n\}$. Further we have the following lemma.

Lemma 5.3 *For every policy ψ , $t \in \mathbb{Z}_{\geq 0}$ and $i \in \{1, 2, \dots, n\}$ we have that $h_i^t \geq a_i \cdot v_i^t$. Moreover $h_i^t - a_i \cdot v_i^t$ is monotonically non-increasing in t and $\lim_{t\to\infty}(h_i^t - a_i \cdot v_i^t) = 0$.*

Proof. From (26) we have that $a_i \cdot v_i^t = a_i \cdot u_i^t + N_i^t - t \cdot a_i$ and thus

$$h_i^t - a_i \cdot v_i^t = d_i - a_i \cdot u_i^t \text{ for every } t \in \mathbb{Z}_{\geq 0}. \tag{35}$$

Thus the lemma follows from the fact that u_i^t is monotonically non-decreasing in t and $d_i = \lim_{t\to\infty} a_i \cdot u_i^t$. □

We also have the following lemma.

Lemma 5.4 *If ψ is an optimal policy then for every $t \in \mathbb{Z}_{\geq 0}$ we have that ψ^t is an optimal policy and $d_i(\psi^t) = h_i^t(\psi)$ for $i = 1, 2, \dots, n$.*

Proof. By Lemma 5.1 and induction it follows for every $t \in \mathbb{Z}_{\geq 0}$ that $d_i(\psi^t) \leq h_i^t(\psi)$ for $i = 1, 2, \dots, n$. Suppose that there exists an $i \in \{1, 2, \dots, n\}$ for which $d_i(\psi^t) < h_i^t(\psi)$. Then

$$S(\psi^t) = \sum_{i=1}^{n} d_i(\psi^t) < \sum_{i=1}^{n} h_i^t(\psi) = S(\psi),$$

which contradicts the fact that ψ is an optimal policy. Hence $d_i(\psi^t) = h_i^t(\psi)$ for $i = 1, 2, \dots, n$ and $S(\psi^t) = S(\psi) = \widetilde{S}$. □

Corollary 5.5 *Let ψ be an optimal policy in an $(a_1, a_2, \dots, a_n)$ system. Then $d_1 := h_1^t(\psi)$, $d_2 := h_2^t(\psi)$, $\dots$, $d_n := h_n^t(\psi)$ is an optimal solution of (33) for every $t \in \mathbb{Z}_{\geq 0}$.*

Lemma 5.6 *Let ψ be an optimal policy in an $(a_1, a_2, \dots, a_n)$ system. Then for every $i \in \{1, 2, \dots, n\}$, $\varepsilon > 0$ and $t_0 \in \mathbb{Z}_{\geq 0}$ there exist $t \in \mathbb{Z}_{\geq t_0}$ such that $h_i^t(\psi) < \varepsilon$.*

Proof. By Lemma 5.4 and Lemma 4.2 we have that

$$\sup_{t\in\mathbb{Z}_{\geq t_0}} (d_i(\psi^{t_0}) - h_i^t(\psi)) = \sup_{t\in\mathbb{Z}_{\geq t_0}} (h_i^{t_0}(\psi) - h_i^t(\psi)) =$$

$$\sup_{t\in\mathbb{Z}_{\geq t_0}} ((t - t_0)a_i + N_i^t(\psi) - N_i^{t_0}(\psi)) = \sup_{t\in\mathbb{Z}_{\geq 0}} (t \cdot a_i - N_i^t(\psi^{t_0})) = d_i(\psi^{t_0}).$$

So, for every $\varepsilon > 0$ there exist $t \in \mathbb{Z}_{\geq t_0}$ such that $h_i^t(\psi) < \varepsilon$. □

Lemma 5.7 *Suppose that for some policy ψ we have for a $t_0 \in \mathbb{Z}_{\geq 0}$ that there exists a $j \in \{1, 2, \dots, n\}$ such that $h_j^{t_0}(\psi) \geq 1$, $N_j^{t_0}(\psi) \geq 1$ and $k_{t_0}(\psi) = l \neq j$. Let $\widetilde{t} = \max\{t < t_0 : k_t(\psi) = j\}$. Consider the policy φ defined by $k_t(\varphi) = k_t(\psi)$ for every $t \in \mathbb{Z}_{\geq 0}$, $t \neq \widetilde{t}, t_0$, $k_{\widetilde{t}}(\varphi) = l$ and $k_{t_0}(\varphi) = j$. Then we have that $d_l(\varphi) \leq d_l(\psi)$ and $d_i(\varphi) = d_i(\psi)$ for every $i \in \{1, 2, \dots, n\} \setminus \{l\}$.*

Proof. Since $N_l^t(\varphi) \geq N_l^t(\psi)$ for every $t \in \mathbb{Z}_{\geq 0}$ we have by Lemma 4.2 that $d_l(\varphi) \leq d_l(\psi)$. Further it is obvious for $i \in \{1, 2, \dots, n\} \setminus \{j, l\}$ that $d_i(\varphi) = d_i(\psi)$. So it remains to prove that $d_j(\varphi) = d_j(\psi)$. Because $N_j^t(\varphi) \leq N_j^t(\psi)$ for every $t \in \mathbb{Z}_{\geq 0}$ we have by Lemma 4.2 that $d_j(\varphi) \geq d_j(\psi)$. If $d_j(\varphi) > d_j(\psi)$ then

$$d_j(\varphi) = t_0 \cdot a_j - N_j^{t_0}(\varphi) = t_0 \cdot a_j - N_j^{t_0}(\psi) + 1 > d_j(\psi) \Rightarrow$$

$$h_j^{t_0}(\psi) = d_j(\psi) - t_0 \cdot a_j + N_j^{t_0}(\psi) < 1,$$

which is a contradiction. Thus $d_j(\varphi) = d_j(\psi)$. □

Theorem 5.8 *For every $\varepsilon > 0$ and $i \in \{1, 2, \dots, n\}$ there exists an optimal solution $d_1, d_2, \dots, d_n$ of (33) with $d_i < \varepsilon$ and $d_j < 1$ for all $j \neq i$.*

Proof. Without loss of generality we can assume $i = 1$. Let $d_1^*, d_2^*, \dots, d_n^*$ be an optimal solution of (33) and ψ an optimal policy obtained by applying the GG algorithm to this optimal solution. Then according to Corollary 5.5 it is sufficient to show that for every $\varepsilon > 0$ there exists a $t \in \mathbb{Z}_{\geq 0}$ such that $h_1^t(\psi) < \varepsilon$ and $h_j^t(\psi) < 1$ for $j = 2, 3, \dots, n$.
Suppose this does not hold. Then there exists an $0 < \varepsilon < a_1$ such that for all $t \in \mathbb{Z}_{\geq 0}$ for which $h_1^t < \varepsilon$ there exists an $i \in \{2, 3, \dots, n\}$ with $h_i^t \geq 1$. Let $t' = \min\{t \in \mathbb{Z}_{\geq 0} : N_i^t(\psi) \geq 1\}$ for all $i \in \{2, 3, \dots, n\}$ and $J = \{t \in \mathbb{Z}_{\geq 0}, t \geq t' : h_1^t < \varepsilon\}$. Then by Lemma 5.6 we have that $J \neq \emptyset$ and it follows for every $t \in J$ that there exists an $i \in \{2, 3, \dots, n\}$ such that $h_i^t \geq 1$ and $N_i^t \geq 1$. By Lemma 5.1 we have for $t \in J$ that $k_t(\psi) = 1$, because $h_1^t < \varepsilon < a_1$ and $h_1^{t+1} \geq 0$. We construct a new policy ϕ from policy ψ by changing some of the $k_t(\psi)$ in the following way. Let $J = \{c_0, c_1, c_2, \dots\}$ with $c_0 < c_1 < c_2 < \dots$. We start with $k_t(\phi) = k_t(\psi)$ for every $t \in \mathbb{Z}_{\geq 0}$ and then we do consecutively for $s = c_0$, $s = c_1$, … the following. Let $i \in \{2, 3, \dots, n\} : h_i^s \geq 1$ and $\widetilde{t} = \max\{t < s : k_t(\psi) = i\}$. For the new policy ϕ we take $k_{\widetilde{t}}(\phi) = 1 = k_s(\psi)$ and $k_s(\phi) = i = k_{\widetilde{t}}(\psi)$.
Then by Lemma 5.7 and induction we have that $d_1(\varphi) \leq d_1(\psi)$ and $d_i(\varphi) = d_i(\psi)$ for $i = 2, 3, \dots, n$. If $d_1(\varphi) < d_1(\psi)$ then $S(\varphi) = \sum_{i=1}^n d_i(\varphi) < \sum_{i=1}^n d_i(\psi) = S(\psi)$, which contradicts the fact that ψ is an optimal policy. Hence, $d_i(\varphi) = d_i(\psi)$ for $i = 1, 2, \dots, n$ and φ is also an optimal policy. Thus for every $t \in \mathbb{Z}_{\geq 0}$ we have that

$$h_1^t(\varphi) = d_1(\psi) - t \cdot a_1 + N_1^t(\varphi) \geq d_1(\psi) - t \cdot a_1 + N_1^t(\psi) = h_1^t(\psi)$$

and for every $t \in J$ we have that $h_1^t(\varphi) = h_1^t(\psi) + 1 \geq 1$. So for every $t \geq t'$ we have that $h_1^t(\varphi) \geq \varepsilon$. However, by Lemma 5.6 we have that φ is not an optimal policy, which is a contradiction. □

From Lemma 5.3 , Lemma 5.6 and Theorem 5.8 we have the following property of optimal policies.

Corollary 5.9 *If in an* $(a_1, a_2, \ldots, a_n)$ *system an optimal policy* ψ *is applied then for all* $t \in \mathbb{Z}_{\geq 0}$, $i \in \{1, 2, \ldots, n\}$ *and* $\varepsilon > 0$ *there exist* $t' \in \mathbb{Z}_{\geq 0}, t' > t$ *such that* $a_i \cdot v_i^{t'} < \varepsilon$ *and* $a_j \cdot v_j^{t'} < 1$ *for all* $j \neq i$.

If $d_i < 1$ then $\max(0, \lceil t \cdot a_i - d_i \rceil) = \lceil t \cdot a_i - d_i \rceil$ for every $t \in \mathbb{Z}_{\geq 0}$. So, to obtain an optimal policy for an $(a_1, a_2, \ldots, a_n)$ system we can, according to Theorem 5.8, solve the following MPP instead of (33).

$$\begin{array}{lcl} \text{minimize } S & = & \sum_{i=1}^{n} d_i \\ \text{subject to:} & & \\ \sum_{i=1}^{n} \lceil t \cdot a_i - d_i \rceil & \leq & t \text{ for } t = 0, 1, \ldots \\ 0 \leq d_i < 1 & & \quad \text{for } i = 1, 2, \ldots, n \end{array} \tag{36}$$

From an optimal solution $d_1, d_2, \ldots, d_n$ of (36) an optimal policy ψ can be obtained by applying the GG algorithm. Further the GG algorithms can be used to search iteratively for better solutions of (36). We now introduce a simple greedy algorithm which in fact is a (special) GG algorithm and thus has the above properties. We denote this algorithm as the SG (Special Greedy) algorithm and in the SG algorithm we completely fix the k_t. This in contrary to the GG algorithm, where there is sometimes a freedom in the choice of k_t.

SG Algorithm. The servers are ordered such that $a_1 \geq a_2 \geq \ldots \geq a_n$ and let $d_1, d_2, \ldots, d_n \in \mathbb{R}_{\geq 0}$ be given. Then the SG algorithm determines ψ inductively in the following way. Suppose $k_0(\psi), k_1(\psi), \ldots, k_{t-1}(\psi)$ have been determined. Then $k_t(\psi) := i \in \{1, 2, \ldots, n\}$ for which $\frac{h_i^t}{a_i}$ is minimal, where $h_i^t = d_i - t \cdot a_i + N_i^t(\psi)$ for $i = 1, 2, \ldots, n$. In case of a tie the server with the lowest index is chosen.

Proposition 5.10 *The SG algorithm is a special GG algorithm.*

Proof. For the GG algorithm we have that

$$q_i(t) = \min\{t' \in \mathbb{Z}_{\geq 0} : t' \cdot a_i > d_i - t \cdot a_i + N_i^t(\psi)\} =$$

$$\lfloor \frac{d_i - t \cdot a_i + N_i^t(\psi)}{a_i} \rfloor + 1 = \lfloor \frac{h_i^t}{a_i} \rfloor + 1 \text{ for } i = 1, 2, \ldots, n.$$

So, for an $i \in \{1, 2, \ldots, n\}$ for which $\frac{h_i^t}{a_i}$ is minimal also $q_i(t)$ is minimal. Thus the SG algorithm is a special GG algorithm. □

Now we have the following on the structure of optimal policies. For every $(a_1, a_2, \ldots, a_n)$ system there exists an optimal policy ψ which is obtained by applying the SG algorithm to an optimal solution $d_1, d_2, \ldots, d_n$ of (36). This optimal policy ψ can be represented in the following way. For $i \in \{1, 2, \ldots, n\}$ we represent server i as a point P_i on a circle that moves from $t = 0$ clockwise over the circle with an angular momentum of $a_i \cdot 2\pi$ radians per time unit. So, the

angular momentum is directly proportional to the work capacity a_i of the server. At $t = 0$ the point P_i starts $d_i \cdot 2\pi$ radians before the top of the circle. Then the order $k_0(\psi), k_1(\psi), \ldots$ in which the arriving jobs are routed to the servers if policy ψ is applied is the order in which the points representing the servers pass the top of the circle. If several points pass the top of the circle at exactly the same moment then the convention for the SG algorithm is used, namely that the server with the lowest index of the tied servers comes first in order.
Such sequences $k_0(\psi), k_1(\psi), \ldots$ obtained by applying the SG algorithm on $d_1, d_2, \ldots d_n$ where $d_i < 1$ for $i = 1, 2, \ldots, n$ are in the literature known as billiard sequences (see for example [4]). Namely, if a billiard ball bounces in an n - dimensional cube and a sequence of the integers $\{1, 2, \ldots, n\}$ (the integers correspond to the sides of the cube and opposite sides correspond to the same integer) is constructed by writing down a integer if the ball bounces against the side of the cube corresponding to that integer then such a sequence is a billiard sequence. It is clear that the sequence obtained from the SG algorithm is a billiard sequence for some well chosen initial position and direction of the billiard ball.
Observe that in the circle representation the constraints $\sum_{i=1}^{n} \lceil t \cdot a_i - d_i \rceil \leq t$ of (36) imply that the starting points on the circle corresponding with the d_i must be such that the total number of passages through the top of the circle in the time-interval $[0, t)$ is smaller or equal than t for $t = 0, 1, \ldots$. Further remark that for GG algorithms the same representation can be used. The only difference for a GG algorithm is that if for some $t_0 \in \mathbb{Z}_{\geq 0}$ several points $P_{i_1}, P_{i_2}, \ldots, P_{i_k}$ pass the top of the circle in the time-interval $[t_0, t_0 + 1)$ then the choice of order of the routing to the corresponding servers $i_1, i_2, \ldots, i_k$ is free.

According to the following theorem the upper bound of Corollary 4.6 is best possible if the work capacities $a_1, a_2, \ldots, a_n$ are linearly independent over $\mathbb{Z}$.

Theorem 5.11 *If the work capacities $a_1, a_2, \ldots, a_n$ are linearly indepent over $\mathbb{Z}$ and ψ is a policy such that $d_i = d_i(\psi) \leq z_i$ for $i = 1, 2, \ldots, n$ with $\sum_{i=1}^{n} z_i = n - 1$ then $d_i = z_i$ for $i = 1, 2, \ldots, n$*

Proof. Let $\varepsilon > 0$ be an arbitrary small positive number. We have that $1, a_2, a_3, \ldots, a_n$ are linearly independent over $\mathbb{Z}$, because $\sum_{i=1}^{n} a_i = 1$ and $a_1, a_2, \ldots, a_n$ are linearly independent over $\mathbb{Z}$. So, according to Kronecker's theorem on inhomogeneous simultaneous Diophantine approximation, there exist $m_i \in \mathbb{Z}, t \in \mathbb{N}$ such that $|t \cdot a_i - m_i - (z_i + \frac{\varepsilon}{2})| < \frac{\varepsilon}{2}$ for $i = 2, 3 \ldots, n$ where ε is some positive number less than $\frac{1}{n}$. Then

$$t \cdot a_i = m_i + z_i + \theta_i \tag{37}$$

with $\theta_i \in (0, \varepsilon)$ for $i = 2, 3, \ldots, n$. Further

$$t \cdot a_1 = t - \sum_{i=2}^{n} t \cdot a_i = t - \sum_{i=2}^{n} m_i - \sum_{i=2}^{n} z_i - \sum_{i=2}^{n} \theta_i =$$

$$t - M_1 - (n - 1 - z_1) + \theta_1 = m_1 + z_1 + \theta_1 \tag{38}$$

where $M_1 := \sum_{i=2}^{n} m_i \in \mathbb{Z}$, $m_1 := t - M_1 - n + 1 \in \mathbb{Z}$ and $\theta_1 := -\sum_{i=2}^{n} \theta_i \in (-\delta, 0)$ with $\delta = (n - 1) \cdot \varepsilon$. By Lemma 4.2 we have that $t \cdot a_1 - N_1^t \leq d_1 \leq z_1$. Hence by (38) we have $m_1 + z_1 + \theta_1 - N_1^t \leq z_1$ and thus

$$N_1^t \geq m_1. \tag{39}$$

For $i = 2, 3, \ldots, n$ we have $t \cdot a_i - N_i^t \leq d_i \leq z_i$ by Lemma 4.2. Thus by (37) $m_i + z_i + \theta_i - N_i^t \leq z_i$. So, $N_i^t \geq m_i + \theta_i$ whence

$$N_i^t \geq m_i + 1 \text{ for } i = 2, 3, \ldots, n. \tag{40}$$

Because $\sum_{i=1}^n m_i = \sum_{i=1}^n (t \cdot a_i - z_i - \theta_i) = t - n + 1$, we have by (39) and (40) that

$$t = \sum_{i=1}^n N_i^t \geq m_1 + \sum_{i=2}^n (m_i + 1) = \sum_{i=1}^n m_i + n - 1 = t.$$

Hence $N_1^t = m_1$ and $N_i^t = m_i + 1$ for $i = 2, 3, \ldots, n$. From $N_1^t = m_1$ and (38) it follows that $t \cdot a_1 - N_1^t = z_1 + \theta_1$ with $\theta_1 \in (-\delta, 0)$ where δ can be made arbitrarily small. So, by Lemma 4.2 $d_1 = z_1$. Analogously it follows that $d_i = z_i$ for $i = 2, 3, \ldots, n$. □

Corollary 5.12 *If $a_1, a_2, \ldots, a_n$ are linearly independent over $\mathbb{Z}$ then $\widetilde{S} = n - 1$ and $\widetilde{W} = \frac{n-1}{2}$. In that case all $d_1, d_2, \ldots, d_n$ that fulfill (29) are optimal solutions of (33). If furthermore $d_i < 1$ for $i = 1, 2, \ldots, n$ then those d_i are also an optimal solution for (36).*

Conjecture 5.13 *Let $q := \dim_{\mathbb{Q}}(a_1\mathbb{Q} + a_2\mathbb{Q} + \ldots + a_n\mathbb{Q})$ be the irrationality degree of $a_1, a_2, \ldots, a_n$. Then $\widetilde{S} \geq \frac{n+q}{2} - 1$.*

From Theorem 3.2, $\widetilde{W} \geq 0$ and Corollary 5.12 it follows that the conjecture is true in the extremal cases $q = 1$ and $q = n$.

6 The optimal policy in case of rational a_i

In this section we consider the case that all the work capacities a_i are rational, i.e $a_i \in \mathbb{Q}$ for $i = 1, 2, \ldots, n$. Then there exist $T \in \mathbb{N}$, $x_i \in \mathbb{N}$ such that $a_i = \frac{x_i}{T}$ for $i = 1, 2, \ldots, n$ and $\gcd(x_1, x_2, \ldots, x_n) = 1$. By (2) we have that $T = \sum_{i=1}^n x_i$ and we denote T as the period of the system, because T has the following property.

Theorem 6.1 $T = \min\{t \in \mathbb{N} :$ *there exist periodic policies with period t which are optimal* $\}$.

Proof. In the previous section we have shown that a policy ψ obtained from applying the SG algorithm to an optimal solution of (36) is optimal and we have given a circle representation for such a policy ψ. From the circle representation it follows that if $a_i = \frac{x_i}{T}$ then in T time-units the point P_i representing server i makes exactly $a_i \cdot T = x_i$ turns around the circle. So, at $t = T$ the point P_i has passed the top of the circle exactly x_i times and is at the same place on the circle as where it started at $t = 0$. Hence, if $a_i = \frac{x_i}{T}$ for $i = 1, 2, \ldots, n$ then the optimal policy ψ is periodic with period T and from T consecutively arriving jobs exactly x_i are routed to server i for $i = 1, 2, \ldots, n$. In other words for policy ψ we have for all $t \in \mathbb{Z}_{\geq 0}$ that

$$k_{t+T}(\psi) = k_t(\psi) \tag{41}$$

and

$$N_i^{t+T}(\psi) = N_i^t(\psi) + x_i \text{ for } i = 1, 2, \ldots, n. \tag{42}$$

So, indeed there exist optimal policies with period T. We still have to show that there do not exist $t' \in \mathbb{N}$, $t' < T$ such that optimal policies with period t' exist.

Suppose there exists an optimal policy ψ' with period $t' < T$. Then there exist $m_i \in \mathbb{N}$ such that for all $t \in \mathbb{Z}_{\geq 0}$ we have $N_i^{t+t'}(\psi') = N_i^t(\psi') + m_i$ for $i = 1, 2, \ldots, n$. Then it follows that

$$p_i = \frac{m_i}{t'} \text{ for } i = 1, 2, \ldots, n. \tag{43}$$

Because ψ' is optimal it follows from Corollary 4.6 that $S(\psi') \leq n - 1$. So, $S(\psi') < \infty$ and by Lemma 3.7 we have $p_i = a_i = \frac{x_i}{T}$ for $i = 1, 2, \ldots, n$. So, by (43) it follows that $\frac{m_i}{t'} = \frac{x_i}{T}$ and thus $x_i = \frac{m_i \cdot T}{t'}$ for $i = 1, 2, \ldots, n$. Because $T > t'$ there exists a prime number p such that $\mathrm{ord}_p(T) > \mathrm{ord}_p(t')$. Hence we have that

$$\mathrm{ord}_p(x_i) = \mathrm{ord}_p(m_i) + \mathrm{ord}_p(T) - \mathrm{ord}_p(t') \geq \mathrm{ord}_p(T) - \mathrm{ord}_p(t') \geq 1$$

and thus $p | x_i$ for $i = 1, 2, \ldots, n$. Hence $\gcd(x_1, x_2, \ldots, x_n) \geq p > 1$, which is a contradiction. □

So, if all the work capacities a_i are rational, $a_i = \frac{x_i}{T}$ with $x_i, T \in \mathbb{N}$ and $\gcd(x_1, x_2, \ldots, x_n) = 1$, then we can restrict to policies ψ for which (41) and (42) hold to obtain optimal policies. We will denote policies for which (41) and (42) hold as proportional periodic (**p.p**) policies. Observe that a p.p policy ψ is characterized by $k_0(\psi)$, $k_1(\psi)$, $\ldots$, $k_{T-1}(\psi)$ and therefore we denote ψ with $\psi = (k_0(\psi), k_1(\psi), \ldots, k_{T-1}(\psi))^\infty$.

Further we will render such a system with rational work capacities as $(x_1, x_2, \ldots, x_n)$ instead of $(a_1, a_2, \ldots, a_n)$. We have the following lemma which states that the performance of p.p policies is invariant under shifts.

Lemma 6.2 *For an p.p policy ψ in an $(x_1, x_2, \ldots, x_n)$ system we have that*

$$S(\psi^t) = S(\psi) \textit{ for every } t \in \mathbb{Z}_{\geq 0}.$$

Proof. For a p.p policy ψ we have that ψ is the same policy as ψ^{mT} for $m \in \mathbb{Z}_{\geq 0}$. Hence, $S(\psi) = S(\psi^T) = S(\psi^{2T}) = \ldots$ and the lemma follows from Lemma 5.2. □

Moreover, it is convenient to multiply some of the earlier introduced variables with the period T, because we then get integers. If policy ψ is applied to an $(x_1, x_2, \ldots, x_n)$ system then we define for $i = 1, 2, \ldots, n$ the variables

$$s_i = s_i(\psi) := T \cdot d_i(\psi) = \lim_{t \to \infty} x_i \cdot u_i^t.$$

We have the following lemma.

Lemma 6.3 *If $s_i < \infty$ for $i \in \{1, 2, \ldots, n\}$ then $s_i = \max_{t \in \mathbb{Z}_{\geq 0}}(t \cdot x_i - T \cdot N_i^t)$ and $s_i \in \mathbb{Z}_{\geq 0}$.*

Proof. It is obvious that $s_i \geq 0$ and $(t \cdot x_i - T \cdot N_i^t) \in \mathbb{Z}$ for all $t \in \mathbb{Z}_{\geq 0}$. So, if $s_i < \infty$ then we have by Lemma 4.2 that

$$s_i = d_i \cdot T = T \cdot \sup_{t \in \mathbb{Z}_{\geq 0}} (t \cdot a_i - N_i^t) = \max_{t \in \mathbb{Z}_{\geq 0}} (t \cdot x_i - T \cdot N_i^t) \in \mathbb{Z}_{\geq 0}.$$

□

Because

$$S \cdot T = T \cdot \sum_{i=1}^{n} d_i = \sum_{i=1}^{n} s_i \tag{44}$$

we have the following corollary.

Corollary 6.4 *For an $(x_1, x_2, \ldots, x_n)$ system we have*

$$S \cdot T \in \mathbb{Z}_{\geq 0} \text{ if } S < \infty. \tag{45}$$

Further by Corollary 4.7 and Theorem 3.2 we have the following corollary.

Corollary 6.5 *For an $(x_1, x_2, \ldots, x_n)$ system we have*

$$2T \cdot W \in \mathbb{Z}_{\geq 0} \text{ if } W < \infty. \tag{46}$$

Further we have

$$\widetilde{S} \cdot T \in \mathbb{Z}_{\geq 0} \tag{47}$$

and

$$2T \cdot \widetilde{W} \in \mathbb{Z}_{\geq 0}. \tag{48}$$

Moreover the factors 2 in (46) and (48) can be deleted if n is odd.

We have for $t \in \mathbb{Z}_{\geq 0}$ that $S^t \cdot T = \sum_{i=1}^n (a_i \cdot T) \cdot v_i^t = \sum_{i=1}^n x_i \cdot v_i^t$. It follows that it is convenient to define for $t \in \mathbb{Z}_{\geq 0}$ and $i \in \{1, 2, \ldots, n\}$ the variables

$w_i^t = x_i \cdot v_i^t$ which is T times the remaining workload in server i at time t.

We note the following properties of w_i^t:

- w1 $S^t \cdot T = \sum_{i=1}^n w_i^t$.
- w2 $w_1^0 = w_2^0 = \cdots = w_n^0 = 0$.
- w3 If $t' = t - 1$ and $k_{t'} = l$, then $w_l^t = w_l^{t'} + T - x_l$ and $w_k^t = \max(w_k^{t'} - x_k, 0) = w_k^{t'} + \max(-x_k, -w_k^{t'})$ for $k \neq l$.
- w4 $w_i^t \in \mathbb{Z}_{\geq 0}$ for $i \in \{1, 2, \ldots, n\}$ and $t \in \mathbb{Z}_{\geq 0}$.

w5 $S^t \cdot T - S^{t'} \cdot T = \sum_{k \neq l} \max(x_k - w_k^{t'}, 0)$.

Proof. The properties w1,w2 and w3 are obvious and property w4 follows from w3. So it only remains to prove property w5. We have $S^t \cdot T - S^{t'} \cdot T \overset{1}{=} \sum_{i=1}^{n}(w_i^t - w_i^{t'}) \overset{3}{=} T - x_l + \sum_{k \neq l} \max(-x_k, -w_k^{t'}) = \sum_{k \neq l}\{x_k + \max(-x_k, -w_k^{t'})\} = \sum_{k \neq l} \max(x_k - w_k^{t'}, 0)$. $\square$

From properties w1 and w4 it follows that $S^t \cdot T \in \mathbb{Z}_{\geq 0}$ for all $t \in \mathbb{Z}_{\geq 0}$. If a p.p policy ψ is applied then the w_i^t have some more interesting properties. To illustrate this we first give an example. We calculate the w_i^t for the situation that $n = 3$, $(x_1, x_2, x_3) = (4, 3, 2)$, so $T = 9$, and we apply the policy $\psi = (1, 2, 1, 3, 2, 1, 2, 1, 3)^\infty$. See Table 1 for the results.

Table 1

t	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
k_t	1	2	1	3	2	1	2	1	3	1	2	1	3	2	1	2	1	3
w_1^t	0	5	1	6	2	0	5	1	6	2	7	3	8	4	0	5	1	6
w_2^t	0	0	6	3	0	6	3	9	6	3	0	6	3	0	6	3	9	6
w_3^t	0	0	0	0	7	5	3	1	0	7	5	3	1	8	6	4	2	0
$S^t \cdot T$	0	5	7	9	9	11	11	11	12	12	12	12	12	12	12	12	12	12

Table 1 contains all the calculated w_i^t until the end of the second period. Note that the w_i^t for $t = 8$ and $t = 17$ are the same. It follows that it is not necessary to calculate the w_i^t for $t > 17$, because the k_t have also period 9. So for all $t \geq 8$ we have $w_i^t = w_i^{T+t}$, or in other words the w_i^t are periodical with period $T = 9$ from moment $t = 8$ on. Further it follows that $S^t \cdot T = \sum_{i=1}^{3} w_i^t$ is constant from $t = 8$. In this example $S^t \cdot T = 12$ for all $t \geq 8$ and thus $S^t = \frac{4}{3}$ for $t \geq 8$. So $S = \frac{4}{3}$. In the next theorem we show that these properties hold generally.

Theorem 6.6 *If a p.p policy ψ is applied to an $(x_1, x_2, \ldots, x_n)$ system then for all $t \in \mathbb{Z}_{\geq T-1}$ we have $S = S^t$, $d_i(\psi) = a_i \cdot u_i^t$, $s_i(\psi) = x_i \cdot u_i^t$,*

$$v_i^t = v_i^{t+T} \text{ and } w_i^t = w_i^{t+T} \tag{49}$$

for $i = 1, 2, \ldots, n$. Further for all $t \in \{T, 2T, 3T, \ldots\}$ we have

$$d_i(\psi) = a_i \cdot v_i^t \text{ and } s_i(\psi) = w_i^t. \tag{50}$$

Proof. We consider an arbitrary server i. Because a p.p policy is applied we know that in the first period exactly x_i jobs are routed to server i. Assume in the first period an arriving job is routed to server i at moments $t_i^1, t_i^2, \ldots, t_i^{x_i}$ with $0 \leq t_i^1 < t_i^2 < \ldots < t_i^{x_i} < T$. After the first period this is periodically repeated, so the arriving jobs are routed to server i at moments $t_i^1 + m \cdot T,\ t_i^2 + m \cdot T, \ldots, t_i^{x_i} + m \cdot T$ with $m \in \mathbb{Z}_{\geq 0}$. We first show that from time $t = t_i^{x_i}$ server i is constantly in use for processing jobs. Let $t > t_i^{x_i}$ be an arbitrary moment and $t_0 = \max(0, t - T)$. Then exactly x_i jobs have been routed to server i in the interval $[t_0, t)$, because a p.p policy is applied and $t > t_i^{x_i}$. Since server i could not start earlier than at time t_0 with the processing of those jobs, server i is busy with those jobs at least until time $t_0 + x_i \cdot a_i^{-1} = t_0 + x_i \cdot \frac{T}{x_i} = t_0 + T \geq t$. So server i is busy at any arbitrary moment $t > t_i^{x_i}$, whence $u_i^{t_i^{x_i}} = u_i^{t_i^{x_i}+1} = \cdots$. Because $t_i^{x_i} \leq T-1$ for $i = 1, 2, \ldots, n$, we have for $i = 1, 2, \ldots, n$ that $u_i^{T-1} = u_i^T = \cdots$. Hence $S^{T-1} = S^T = \cdots = S$ and it follows for all $t \in \mathbb{Z}_{\geq 0}$, $t \geq T-1$ that $S = S^t$ and $d_i(\psi) = a_i \cdot u_i^t$, $s_i(\psi) = x_i \cdot u_i^t$ for $i = 1, 2, \ldots, n$. Further if $t \geq t_i^{x_i}$, $t \in \mathbb{N}$ and $k_t \neq i$, then $v_i^t \geq 1$ and thus $w_i^t \geq x_i$. Hence we have by property w3 for $t \in \mathbb{N}$, $t \geq t_i^{x_i}$ that

$$\begin{aligned} w_i^{t+1} &= w_i^t + T - x_i \quad && \text{if } k_t = i, \\ w_i^{t+1} &= w_i^t - x_i && \text{if } k_t \neq i \end{aligned}$$

and $w_i^{t+T} = w_i^t + x_i(T - x_i) - (T - x_i)x_i = w_i^t$. This implies that $w_i^{T+t} = w_i^t$, $v_i^{T+t} = v_i^t$ for $i \in \{1, 2, \ldots, n\}$ and $t \geq T-1$.

Thus $v_i^T = v_i^{2T} = v_i^{3T} = \ldots$ and we only have to show that $a_i \cdot v_i^T = d_i(\psi)$ for $i = 1, 2, \ldots, n$ to complete the proof. From the proof of Lemma 4.1 we have for $t \in \mathbb{Z}_{\geq 0}$ and $i \in \{1, 2, \ldots, n\}$ that $a_i \cdot u_i^t - a_i \cdot v_i^t = a_i \cdot t - N_i^t(\psi)$. Hence $a_i \cdot u_i^T - a_i \cdot v_i^T = a_i \cdot T - N_i^T(\psi) = 0$, because ψ is a p.p policy. So $a_i \cdot v_i^T = a_i \cdot u_i^T = d_i(\psi)$ and $w_i^T = T \cdot a_i \cdot v_i^T = T \cdot d_i(\psi) = s_i(\psi)$ for $i = 1, 2, \ldots, n$. □

By Theorem 3.2 we have the following corollary.

Corollary 6.7 *If a p.p policy is applied in an $(x_1, x_2, \ldots, x_n)$ system then for all $t \in \mathbb{Z}_{\geq T-1}$ we have*

$$W = S^t - \frac{n-1}{2}.$$

By Theorem 6.6 and Corollary 6.7 we have according to Table 1 in the $(4, 3, 2)$ example that $s_1 = 2$, $s_2 = 3$, $s_3 = 7$ and the long-run average waiting time W equals $S^8 - 1 = \frac{4}{3} - 1 = \frac{1}{3}$. We can compute W also in a more direct way. Namely, note that the total waiting time for jobs that arrive consecutively at the moments $t_0, t_0+1, \ldots, t_0+t_1$ with $t_0, t_1 \in \mathbb{Z}_{\geq 0}$ equals $\sum_{t=t_0}^{t_0+t_1} v_{k_t}^t = \sum_{t=t_0}^{t_0+t_1} \frac{w_{k_t}^t}{x_{k_t}}$. If a p.p policy is applied, then it follows from Theorem 6.6 that $\sum_{t=t_0}^{t_0+T-1} \frac{w_{k_t}^t}{x_{k_t}}$ is independent of t_0 if $t_0 \geq T-1$. Thus from $t = T-1$ on the total waiting time of T consecutively arriving jobs is constant if a p.p policy is applied and W can be computed by the formula

$$T \cdot W = \sum_{t=t_0}^{t_0+T-1} \frac{w_{k_t}^t}{x_{k_t}} \text{ for } t_0 \geq T-1. \tag{51}$$

In (51) it is convenient to take $t_0 = T$, so that the total waiting time of the T jobs that arrive in the second period is computed. If we do so for the $(4, 3, 2)$ example we find that $T \cdot W = \sum_{t=9}^{17} \frac{w_{k_t}^t}{x_{k_t}} = \frac{2}{4} + \frac{0}{3} + \frac{3}{4} + \frac{1}{2} + \frac{0}{3} + \frac{0}{4} + \frac{3}{3} + \frac{1}{4} + \frac{0}{2} = 3$ (see Table 1). So $W = \frac{3}{T} = \frac{1}{3}$, as expected.

If a p.p policy ψ is applied in an $(x_1, x_2, \ldots, x_n)$ system then for $i \in \{1, 2, \ldots, n\}$ we put

$$t_i = \min\{t \in \mathbb{Z}_{\geq 0} : u_i^t = \lim_{t \to \infty} u_i^t\}. \tag{52}$$

So t_i is the moment from which server i has no idle periods anymore and from the proof of Theorem 6.6 follows that $t_i \in \{0, 1, \ldots, T-1\}$ for $i = 1, 2, \ldots, n$.

Proposition 6.8 *If a p.p policy is applied in an* $(x_1, x_2, \ldots, x_n)$ *system, then for each* $i \in \{1, 2, \ldots, n\}$ *there exists at least one moment* t' *in each period for which* $w_i^{t'} = 0$.

Proof. Let $i \in \{1, 2, \ldots, n\}$ be arbitrary. It is clear that $w_i^{t_i} = 0$. Further from the proof of Theorem 6.6 it follows that $w_i^{t+T} = w_i^t$ for all $t \geq t_i$. So, for all $m \in \mathbb{Z}_{\geq 0}$ we have $w_i^{t_i + m \cdot T} = w_i^{t_i} = 0$. □

For an optimal p.p policy we have the following stronger result.

Theorem 6.9 *If an optimal p.p policy is applied in an* $(x_1, x_2, \ldots, x_n)$ *system, then for each* $i \in \{1, 2, \ldots, n\}$ *there exists at least one moment* t' *in each period such that* $w_i^{t'} = 0$ *and* $w_j^{t'} \leq T - 1$ *for all* $j \neq i$.

Proof. By property w2 the statement holds for the first period. So, by (49) we only have to proof that for each $i \in \{1, 2, \ldots, n\}$ there exist $t' \in \{T-1, T, T+1, \ldots\}$ such that $w_i^{t'} = 0$ and $w_j^{t'} \leq T-1$ for all $j \neq i$. Let $0 < \varepsilon < \frac{1}{T}$. Then by Corollary 5.9 there exist $t' \in \{T-1, T, T+1, \ldots\}$ such that $a_i \cdot v_i^{t'} < \varepsilon$ and $a_j \cdot v_j^{t'} < 1$ for all $j \neq i$. Hence $w_i^{t'} < T \cdot \varepsilon < 1$ and $w_j^{t'} < T$ for all $j \neq i$ and from property w4 it follows that $w_i^{t'} = 0$ and $w_j^{t'} \leq T - 1$ for all $j \neq i$. □

At the end of section 4 we said that a GG algorithm may be used to obtain iteratively better policies. In fact it is mainly practical in case of rational a_i, because if an obtained policy ψ is a p.p policy we can calculate the $d_i(\psi)$ quickly. Namely, in that case we have by Theorem 6.6 that $d_i(\psi) = a_i \cdot v_i^T$ for $i = 1, 2, \ldots, n$ and v_i^T can quickly be determined. So, we want to get p.p policies and we will use the SG algorithm. If we start with a feasible solution $d_1, d_2, \ldots, d_n$ of (33) for which $d_i < 1$ for $i = 1, 2, \ldots, n$ we have no problem since according to (41) and (42) the SG algorithm always yields a p.p policy in that case. On the other hand if the SG algorithm is applied to a feasible solution $d_1, d_2, \ldots, d_n$ of (33) for which $d_i \geq 1$ for some $i = 1, 2, \ldots, n$ then it is possible that the obtained policy ψ is not a p.p policy. However, it is easily seen that the obtained policy ψ is always ultimately proportional periodic, i.e there exists a $t \in \mathbb{Z}_{\geq 0}$ such that ψ^t is a p.p policy. In this case we think it is best to take $d_1(\psi^t), d_2(\psi^t), \ldots, d_n(\psi^t)$ as new feasible solution of 33 instead of $d_1(\psi), d_2(\psi), \ldots, d_n(\psi)$. Namely, ψ^t is a p.p policy and according to Corollary 5.2 we have that policy ψ^t is at least as good (and probably better) than policy ψ. We now give an example of obtaining iteratively better policies by using the SG algorithm.

We apply the SG algorithm to an $(7, 6, 3, 2, 1)$ system and we start with the feasible solution $d_1 = d_2 = \ldots = d_5 = \frac{18}{19}$ of (33). Then applying the SG algorithm to this solution yields the policy $\psi_1 := (1, 2, 1, 3, 2, 1, 4, 2, 1, 3, 2, 1, 2, 1, 5, 4, 3, 2, 1)^\infty$ and we have $d_1(\psi_1) = \frac{12}{19}$, $d_2(\psi_1) = \frac{7}{19}$, $d_3(\psi_1) = \frac{10}{19}$, $d_4(\psi_1) = \frac{12}{19}$ and $d_5(\psi_1) = \frac{14}{19}$. Next we find the policy $\psi_2 = (2, 1, 3, 2, 1, 4, 1, 2, 3, 1, 2, 1, 2, 5, 1, 4, 3, 2, 1)^\infty$ with $d_1(\psi_2) = \frac{12}{19}$, $d_2(\psi_2) = \frac{7}{19}$, $d_3(\psi_2) = \frac{10}{19}$, $d_4(\psi_2) = \frac{11}{19}$ and $d_5(\psi_2) = \frac{13}{19}$. Next we find the policy $\psi_3 = (2, 1, 3, 2, 1, 4, 1, 2, 3, 1, 2, 1, 5, 2, 4, 1, 3, 2, 1)^\infty$ with $d_1(\psi_3) = \frac{12}{19}$, $d_2(\psi_3) = \frac{7}{19}$, $d_3(\psi_3) = \frac{10}{19}$, $d_4(\psi_3) = \frac{10}{19}$ and $d_5(\psi_3) = \frac{12}{19}$. Next we find the policy $\psi_4 = (2, 1, 3, 2, 1, 4, 1, 2, 3, 1, 2, 5, 1, 2, 4, 1, 3, 2, 1)^\infty$ with $d_1(\psi_4) = \frac{12}{19}$, $d_2(\psi_4) = \frac{7}{19}$, $d_3(\psi_4) = \frac{10}{19}$, $d_4(\psi_4) = \frac{10}{19}$ and $d_5(\psi_4) = \frac{11}{19}$. Finally applying the SG algorithm to this feasible solution of (33) yields again policy ψ_4 and the iterative process stops. We say that policy ψ_4 is stable under applying the SG algorithm. However, policy ψ_4 is not an optimal policy for this system.

Namely, we have found $s_1 = 0$, $s_2 = 8$, $s_3 = 18$, $s_4 = 6$, $s_5 = 15$ as an optimal solution of (59). Hence $\widetilde{S} = \frac{47}{19}$ for this system, while $S(\psi_4) = \frac{50}{19}$.

If at the beginning of the iterative process a better start solution is used then in general it will take less iterations before a stable policy is obtained and the obtained stable policy will often be better. Therefore it is a good idea to obtain a reasonable good policy and corresponding feasible solution of (33) by some algorithm and then applying the iterative process to this solution. Another idea is to apply the iterative process on several feasible solutions of (33) (which can be obtained by different algorithms) and then see which one yields the best stable policy. In section 8 we will give several algorithms that can be applied to this problem.

For given $s_i \in \mathbb{Z}_{\geq 0}$ with $i \in \{1, 2, \ldots, n\}$ we have that

$$\max(0, \lceil t \cdot a_i - d_i \rceil) = \max(0, \lceil t \cdot a_i - \frac{s_i}{T} \rceil) =$$

$$\max(0, \lceil \frac{t \cdot x_i - s_i}{T} \rceil) \text{ for all } t \in \mathbb{Z}_{\geq 0}. \tag{53}$$

So, by Lemma 6.3 we can solve the following MPP instead of (33) to obtain an optimal policy for an $(x_1, x_2, \ldots, x_n)$ system.

$$\begin{array}{lcl} \text{minimize } S \cdot T & = & \sum_{i=1}^{n} s_i \\ \text{subject to:} & & \\ \sum_{i=1}^{n} \max(0, \lceil \frac{t \cdot x_i - s_i}{T} \rceil) & \leq & t \text{ for } t = 0, 1, \ldots \\ s_i & \in & \mathbb{Z}_{\geq 0} \text{ for } i = 1, 2, \ldots, n \end{array} \tag{54}$$

If an optimal p.p policy ψ is applied then $s_1 := s_1(\psi) = w_1^T$, $s_2 := s_2(\psi) = w_2^T$, ..., $s_n := s_n(\psi) = w_n^T$ is an optimal solution of (54). In fact, by (35), Corollary 5.5 and Theorem 6.6 we have the following lemma.

Lemma 6.10 *If an optimal p.p policy is applied then for all $t \in \{T-1, T, T+1, \ldots\}$ we have that $s_1 := w_1^t$, $s_2 := w_2^t$, ..., $s_n := w_n^t$ is an optimal solution of (54).*

By Theorem 6.9 and Lemma 6.10 we have analogously to Theorem 5.8 the following theorem.

Theorem 6.11 *For every $i \in \{1, 2, \ldots, n\}$ there exists an optimal solution $s_1, s_2, \ldots, s_n$ of (54) with $s_i = 0$ and $s_j \leq T - 1$ for all $j \neq i$.*

Corollary 6.12 *For every $(x_1, x_2, \ldots, x_n)$ system we have that*

$$\widetilde{S} \leq n - 1 - \frac{n-1}{T}.$$

Proof. From Theorem 6.11 follows that
$\widetilde{S} \cdot T \leq (n-1)(T-1)$ and thus $\widetilde{S} \leq n - 1 - \frac{n-1}{T}$. □

The upper bound of Corollary 6.12 is a fraction better than the general upper bound of Theorem 4.6. In Theorem 6.14 we will improve the upper bound a little more in case there exist $i \in \{1, 2, \ldots, n\}$ with $\gcd(x_i, T) > 1$.

If $s_i \leq T - 1$ then for all $t \in \mathbb{Z}_{\geq 0}$ we have

$$\max(0, \lceil \frac{t \cdot x_i - s_i}{T} \rceil) = \lceil \frac{t \cdot x_i - s_i}{T} \rceil. \tag{55}$$

Further we have for all $t \in \mathbb{Z}_{\geq 0}$ and $m \in \mathbb{N}$ that

$$\lceil \frac{(t + m \cdot T) \cdot x_i - s_i}{T} \rceil = m \cdot x_i + \lceil \frac{t \cdot x_i - s_i}{T} \rceil. \tag{56}$$

If $s_i \leq T - 1$ for $i = 1, 2, \ldots, n$ then by (55) and (56) we have for all $t \in \mathbb{Z}_{\geq 0}$ and $m \in \mathbb{N}$ that $\sum_{i=1}^{n} \max(0, \lceil \frac{t \cdot x_i - s_i}{T} \rceil) \leq t$ if and only if $\sum_{i=1}^{n} \lceil \frac{(t+m \cdot T) \cdot x_i - s_i}{T} \rceil \leq t + m \cdot T$. So, by Theorem 6.11 and (55) we can to obtain an optimal policy for an $(x_1, x_2, \ldots, x_n)$ system solve the following MPP instead of (54).

$$\begin{array}{lcl} \text{minimize } ST & = & \sum_{i=1}^{n} s_i \\ \text{subject to:} & & \\ \sum_{i=1}^{n} \lceil \frac{t \cdot x_i - s_i}{T} \rceil & \leq & t \text{ for } t = 0, 1, \ldots, T-1 \\ s_i & \in & \mathbb{Z}_{\geq 0} \text{ for } i = 1, 2, \ldots, n \\ s_i & \leq & T - 1 \text{ for } i = 1, 2, \ldots, n \end{array} \tag{57}$$

From an optimal solution $s_1, s_2, \ldots, s_n$ of (57) an optimal p.p policy ψ such that $w_i^T = s_i(\psi) = s_i$ for $i = 1, 2, \ldots, n$ can be obtained by applying the SG algorithm to $d_i = \frac{s_i}{T}$ for $i = 1, 2, \ldots, n$ to determine $k_0(\psi)$, $k_1(\psi)$, $\ldots$, $k_{T-1}(\psi)$.

Lemma 6.13 *Let $s_1, s_2, \ldots, s_n$ be an optimal solution of (57). Then*

$$\gcd(x_i, T) \mid s_i \textit{ for } i = 1, 2, \ldots, n. \tag{58}$$

Proof. There exists an optimal p.p policy ψ such that $s_i(\psi) = s_i$ for $i = 1, 2, \ldots, n$. For every policy it follows from properties w2 and w3 that $\gcd(x_i, T) \mid w_i^t$ for all $t \in \mathbb{Z}_{\geq 0}$ and $i \in \{1, 2, \ldots, n\}$. Hence by (50) we have that $\gcd(x_i, T) \mid w_i^T = s_i$ for $i = 1, 2, \ldots, n$. □

Theorem 6.14 *For every $(x_1, x_2, \ldots, x_n)$ system we have that*

$$\widetilde{S} \leq n - 1 - \frac{D(x_1, x_2, \ldots, x_n)}{T}$$

with $D(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \gcd(x_i, T) - \min_{i \in \{1, 2, \ldots, n\}} \gcd(x_i, T).$

Proof. Let $j \in \{1, 2, \dots, n\}$ such that $\gcd(x_j, T) = \min_{i\in\{1,2,\dots,n\}} \gcd(x_i, T)$. Then by Theorem 6.11 there exists an optimal solution $s_1, s_2, \dots, s_n$ of (57) for which $s_j = 0$. Further for all $i \neq j$ we have by Lemma 6.13 that $\gcd(x_i, T)|s_i$ and thus $s_i \leq T - \gcd(x_i, T)$. Hence

$$\widetilde{S} \cdot T = \sum_{i=1}^{n} s_i \leq \sum_{i\neq j} (T - \gcd(x_i, T) = T \cdot (n-1) - D(x_1, x_2, \dots, x_n).$$

So, $\widetilde{S} \leq n - 1 - \frac{D(x_1,x_2,\dots,x_n)}{T}$. □

We will show that (57) is equivalent to a linear problem with zero-one variables and thus an integer linear problem (ILP). In the sequel we denote for $z \in \mathbb{Z}$ with $z \pmod T$ the integer $a \in \{0, 1, \dots, T-1\}$ for which there exists a $k \in \mathbb{Z}$ such that $z = a + k \cdot T$. Then for all $t \in \mathbb{Z}_{\geq 0}$, $s_i \in \{0, 1, \dots, T-1\}$ we have

$$\sum_{i=1}^{n} \lceil \frac{t \cdot x_i - s_i}{T} \rceil \leq t \Leftrightarrow \sum_{i=1}^{n} \frac{t \cdot x_i - s_i + (s_i - t \cdot x_i) \pmod T}{T} \leq t \Leftrightarrow$$

$$\sum_{i=1}^{n} s_i \geq \sum_{i=1}^{n} \{(s_i - t \cdot x_i) \pmod T\}.$$

So, the following MPP is equivalent to (57).

$$\begin{array}{lcl} \text{minimize } ST & = & \sum_{i=1}^{n} s_i \\ \text{subject to:} & & \\ \sum_{i=1}^{n} s_i & \geq & \sum_{i=1}^{n} \{(s_i - t \cdot x_i) \pmod T\} \text{ for } t = 0, 1, \dots, T-1 \\ s_i & \in & \mathbb{Z}_{\geq 0} \text{ for } i = 1, 2, \dots, n \\ s_i & \leq & T - 1 \text{ for } i = 1, 2, \dots, n \end{array} \tag{59}$$

For $i = 1, 2, \dots, n$, $t = 0, 1, \dots, T-1$ and $j = 0, 1, \dots, T-1$ we put $e_{it}(j) = (j - t \cdot x_i) \pmod T$. Further we define for $i = 1, 2, \dots, n$ and $j = 0, 1, \dots, T-1$ zero-one variables λ_{ij} by

$$\lambda_{ij} = \begin{cases} 1 & \text{if } s_i = j \\ 0 & \text{if } s_i \neq j \end{cases}.$$

The λ_{ij} correspond to the variables s_i of (59). If $s_i \in \{0, 1, \dots, T-1\}$ for $i = 1, 2, \dots, n$ then we have $\sum_{i=1}^{n} s_i = \sum_{i=1}^{n} \sum_{j=0}^{T-1} j \cdot \lambda_{ij}$ and

$$\sum_{i=1}^{n} \{(s_i - t \cdot x_i) \pmod T\} = \sum_{i=1}^{n} e_{it}(s_i) = \sum_{i=1}^{n} \sum_{j=0}^{T-1} e_{it}(j) \cdot \lambda_{ij} \text{ for } t = 0, 1, \dots, T-1.$$

Hence, (57) and (59) are equivalent to the following ILP.

$$\begin{array}{lcll} \text{minimize } ST & = & \sum_{i=1}^{n} \sum_{j=0}^{T-1} j \cdot \lambda_{ij} & \\ \text{subject to:} & & & \\ \sum_{i=1}^{n} \sum_{j=0}^{T-1} (e_{it}(j) - j) \cdot \lambda_{ij} & \leq & 0 & \text{for } t = 0, 1, \dots, T-1 \\ \sum_{j=0}^{T-1} \lambda_{ij} & = & 1 & \text{for } i = 1, 2, \dots, n \\ \lambda_{ij} & \in & \{0, 1\} & \text{for } i = 1, 2, \dots, n \\ & & & \text{and } j = 0, 1, \dots, T-1 \end{array} \tag{60}$$

Thus solving this ILP yields an optimal policy in case the a_i are rational. By solving the LP - relaxation of (60) we can get a lower bound for $\widetilde{S}$ and thus for $\widetilde{W}$. The LP - relaxation of (60) is the following LP -problem.

$$\begin{array}{lcll} \text{minimize} & \sum_{i=1}^{n}\sum_{j=0}^{T-1} j\cdot\lambda_{ij} & & \\ \text{subject to:} & & & \\ \sum_{i=1}^{n}\sum_{j=0}^{T-1}(e_{it}(j)-j)\cdot\lambda_{ij} & \leq & 0 & \text{for } t=0,1,\ldots,T-1 \\ \sum_{j=0}^{T-1}\lambda_{ij} & = & 1 & \text{for } i=1,2,\ldots,n \\ \lambda_{ij} & \geq & 0 & \text{for } i=1,2,\ldots,n \\ & & & \text{and } j=0,1,\ldots,T-1 \end{array} \tag{61}$$

Put $F_i=\{0,\gcd(x_i,T),2\gcd(x_i,T),\ldots,(T-\gcd(x_i,T))\}$ for $i=1,2,\ldots,n$. Then we have the following theorem.

Theorem 6.15 *For every $(x_1,x_2,\ldots,x_n)$ system an optimal solution of (61) is given by*

$$\lambda_{ij}^{*}=\begin{cases} 0 & \text{if } j\notin F_i \\ \frac{gcd(x_i,T)}{T} & \text{if } j\in F_i \end{cases} \tag{62}$$

and the value of (61) is

$$\frac{nT}{2}-\frac{C(x_1,x_2,\ldots,x_n)}{2},\ \text{where } C(x_1,x_2,\ldots,x_n)=\sum_{i=1}^{n}\gcd(x_i,T).$$

Proof. Let $t\in 0,1,\ldots,T-1$ be fixed. Then for every $i\in\{1,2,\ldots,n\}$ we have that

$$\{e_{it}(j):j\in F_i\}=\{0,\gcd(x_i,T),2\gcd(x_i,T),\ldots,(T-\gcd(x_i,T))\}=F_i$$

and thus $\sum_{j\in F_i}e_{it}(j)=\sum_{j\in F_i}j$. Hence

$$\sum_{i=1}^{n}\sum_{j=0}^{T-1}(e_{it}(j)-j)\cdot\lambda_{ij}^{*}=\sum_{i=1}^{n}\sum_{j\in F_i}(e_{it}(j)-j)\cdot\frac{\gcd(x_i,T)}{T}=\sum_{i=1}^{n}0=0$$

for $t=0,1,\ldots$. Further for every $i\in\{1,2,\ldots,n\}$ we have that

$$\sum_{j=0}^{T-1}\lambda_{ij}^{*}=\sum_{j\in F_i}\frac{\gcd(x_i,T)}{T}=\frac{T}{\gcd(x_i,T)}\cdot\frac{\gcd(x_i,T)}{T}=1.$$

Thus (62) is a feasible solution of (61) and the value of this solution is

$$\sum_{i=1}^{n}\sum_{j=0}^{T-1}j\cdot\lambda_{ij}^{*}=\sum_{i=1}^{n}\sum_{j\in F_i}j\cdot\frac{\gcd(x_i,T)}{T}=\sum_{i=1}^{n}\{\frac{\gcd(x_i,T)}{T}\cdot(\sum_{k=0}^{\frac{T}{\gcd(x_i,T)}-1}k\cdot\gcd(x_i,T))=$$

$$\sum_{i=1}^{n}\frac{\gcd(x_i,T)}{2}\cdot(\frac{T}{\gcd(x_i,T)}-1)=\frac{nT}{2}-\frac{1}{2}\cdot C(x_1,x_2,\ldots,x_n).$$

We consider the following dual problem of (61).

$$\begin{array}{lccl}
\text{maximize} & \sum_{i=1}^{n} z_i & & \\
\text{subject to:} & & & \\
\sum_{t=0}^{T-1}\{(j - e_{it}(j)) \cdot y_t\} + z_i & \leq & j & \text{for } i = 1, 2, \ldots, n \\
 & & & \text{and } j = 0, 1, \ldots, T-1 \\
y_t & \geq & 0 & \text{for } t = 0, 1, \ldots, T-1
\end{array} \tag{63}$$

We claim that an optimal solution of (63) is given by

$$\begin{array}{lcll}
y_t^* & = & \frac{1}{T} & \text{for } t = 0, 1, \ldots, T-1 \text{ and} \\
z_i^* & = & \frac{T}{2} - \frac{\gcd(x_i,T)}{2} & \text{for } i = 1, 2, \ldots, n.
\end{array} \tag{64}$$

Namely, let $i \in \{1, 2, \ldots, n\}$ be fixed. Then for every $j \in \{0, 1, \ldots, T-1\}$ we have that

$$\sum_{t=0}^{T-1}(e_{it}(j) \cdot y_t^*) \geq \frac{1}{T} \cdot \sum_{t=0}^{T-1}\{-t \cdot x_i \pmod{T}\} = \frac{\gcd(x_i,T)}{T} \cdot \sum_{k=0}^{\frac{T}{\gcd(x_i,T)}-1} k \cdot \gcd(x_i,T) =$$

$$\frac{\gcd(x_i,T)}{2} \cdot (\frac{T}{\gcd(x_i,T)} - 1) = \frac{T}{2} - \frac{\gcd(x_i,T)}{2}.$$

Hence

$$\sum_{t=0}^{T-1}(j - e_{it}(j)) \cdot y_t^* + z_i^* = j - \sum_{t=0}^{T-1}(e_{it}(j) \cdot y_t^*) + \frac{T}{2} - \frac{\gcd(x_i,T)}{2} \leq j$$

for $i = 1, 2, \ldots, n$ and $j = 0, 1, \ldots, T-1$. So, (64) is a feasible solution of (63) and the value of this solution is

$$\sum_{i=1}^{n} z_i^* = \sum_{i=1}^{n}(\frac{T}{2} - \frac{\gcd(x_i,T)}{2}) = \frac{nT}{2} - \frac{1}{2} \cdot C(x_1, x_2, \ldots, x_n).$$

This lower bound for the value of (63) is equal to the earlier obtained upper bound for the value of (61). So, by the duality theorem of linear programming, due to von Neumann, we have that (62) is an optimal solution of (61), (64) is an optimal solution of (63) and the optimal value of both (61) and (63) is $\frac{nT}{2} - \frac{1}{2} \cdot C(x_1, x_2, \ldots, x_n)$. □

By Theorem 6.15 and Theorem 3.2 we have the following lower bound on $\widetilde{S}$ and $\widetilde{W}$ in case of rational a_i.

Corollary 6.16 *For every $(x_1, x_2, \ldots, x_n)$ system we have that*

$$\widetilde{S} \geq \frac{n}{2} - \frac{C(x_1, x_2, \ldots, x_n)}{2T} \tag{65}$$

and

$$\widetilde{W} \geq \frac{1}{2} - \frac{C(x_1, x_2, \ldots, x_n)}{2T}, \text{ where } C(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \gcd(x_i, T). \tag{66}$$

The lower bounds of Corollary 6.16 can be attained in some cases and in the next section we give a criterion for this. In case of $n = 2$ the lower bounds of Corollary 6.16 can always be attained.

Corollary 6.17 *For every (x_1, x_2) system the lower bounds of Corollary 6.16 can be attained and we have that $\widetilde{S} = 1 - \frac{1}{T}$ and $\widetilde{W} = \frac{1}{2} - \frac{1}{T}$.*

Proof. By definition we have that $\gcd(x_1, x_2) = 1$ and $T = x_1 + x_2$. Hence $\gcd(x_1, T) = \gcd(x_2, T) = 1$. Hence Corollary 6.16 yields that $\widetilde{S} \geq 1 - \frac{1}{T}$. By Corollary 6.12 we have that $\widetilde{S} \leq 1 - \frac{1}{T}$. So, $\widetilde{S} = 1 - \frac{1}{T}$, $\widetilde{W} = 1 - \frac{1}{T}$ and every optimal policy attains the lower bounds of Corollary 6.16. □

Remark. By Theorem 6.11 and Corollary 6.17 we have for every (x_1, x_2) system that both $s_1 = 0, s_2 = T - 1$ and $s_1 = T - 1, s_2 = 0$ are optimal solutions of (57) and in fact it is easily seen that all the optimal solutions of (57) are the pairs of $s_1, s_2 \in \mathbb{Z}_{\geq 0}$ such that $s_1 + s_2 = T - 1$. Thus there are exactly T optimal solutions of (57) and it is easily seen that every optimal solution corresponds to exactly one optimal p.p policy. Namely, there is never freedom of choice when the GG algorithm is applied to such a solution. So, if ψ is such an optimal p.p policy for an (x_1, x_2) system then all the optimal policies for that system are given by ψ^0, ψ^1, ..., ψ^{T-1}. Namely, these T shifts of policy ψ are mutually different and by Lemma 6.2 they are all optimal. Thus in an (x_1, x_2) system all the optimal policies are cyclic shifts of each other.

7 The optimality of regular policies

In this section we will show that to minimize W the applied policy ψ should be as regular (balanced) as possible in some way. We first give some definitions. Let ψ be the applied policy. Then for $i \in \{1, 2, \ldots, n\}$ we define the splitting sequence of the routing to server i

$$\underline{\delta}(i) = (\delta_t(i))_{t=0}^{\infty}, \text{ where } \delta_t(i) = \begin{cases} 1 & \text{if } k_t(\psi) = i \\ 0 & \text{if } k_t(\psi) \neq i \end{cases}.$$

Further we define the support of this sequence by the set

$$A(i) := \{t \in \mathbb{Z}_{\geq 0} : \delta_t(i) = 1\}.$$

Lemma 7.1 *Let $t^* \in \mathbb{Z}_{\geq 0}$ be such that for some $i \in \{1, 2, \ldots, n\}$ and some $t \in \mathbb{Z}_{\geq 0}$ we have that $t^* \leq t$, $v_i^{t^*} = 0$ and $u_i^{t^*} = u_i^t$. Then*

$$v_i^t = \frac{1}{a_i} \cdot (N_i^t - N_i^{t^*}) - (t - t^*) = \frac{1}{a_i} \cdot \sum_{t'=t^*}^{t-1} \delta_{t'}(i) - (t - t^*).$$

Proof. If $u_i^t = u_i^{t+1}$ for $t \in \mathbb{Z}_{\geq 0}$ then by (25) we have

$$v_i^{t+1} - v_i^t = \begin{cases} \frac{1}{a_i} - 1 & \text{if } k_t = i \\ -1 & \text{if } k_t \neq i \end{cases} \tag{67}$$

Since $u_i^{t^*} = u_i^{t^*+1} = \ldots = u_i^t$ the lemma follows from (67) by induction. $\square$.

For any $N \in \mathbb{N}$ we mean by an N -block a set of N consecutive non-negative integers. We say that the splitting sequence $\underline{\delta}(i)$ is a regular (the term Beatty is also used in the literature) sequence with density p if for every N -block ω we have that $\sum_{t\in\omega} \delta_t(i)$, the number of ones in the N -block , equals $\lfloor Np \rfloor$ or $\lceil Np \rceil$. In that case we also say that the corresponding set $A(i)$ is a regular set. We have the following lemma (see e.g [18]).

Lemma 7.2 *If $\underline{\delta}(i)$ is a regular sequence with density d then $A(i)$ is of the form $\{\lfloor k \cdot \frac{1}{d} + \beta \rfloor\}_{k=1}^{\infty}$ for some $-1/d \leq \beta < 0$ or of the form $\{\lceil k \cdot \frac{1}{d} + \beta \rceil\}_{k=1}^{\infty}$ for some $-1-1/d < \beta \leq -1$. Conversely, if the support is of this form then the corresponding zero-one sequence is a regular sequence.*

Corollary 7.3 *$\underline{\delta}(i)$ is a regular sequence with density d if and only if there exists a $\theta \in \mathbb{R}$ such that $\delta_t(i) = \lfloor (t+1)d + \theta \rfloor - \lfloor td + \theta \rfloor$ or $\delta_t(i) = \lceil (t+1)d + \theta \rceil - \lceil td + \theta \rceil$ for $t = 0, 1, \ldots$*

Further we say the splitting sequence is balanced if the number of ones in any two N-blocks differ by at most one. The corresponding set $A(i)$ is then a balanced set. It is obvious that every regular sequence is balanced. The converse is not always true. For example the sequence $1(10)^{\infty}$ is balanced, but not regular. However, in [18] it is shown that the converse is true for periodic sequences and for sequences with irrational density. Further it is shown that every balanced sequence is eventually regular, i.e if the sequence is balanced then there exists a $t' \in \mathbb{Z}_{\geq 0}$ such that the sequence coincides with a regular sequence for every integer $t \geq t_0$.

In general the density of the splitting sequence $\underline{\delta}(i)$ is defined as $\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} \delta_t(i)$ if this limit exists. It is obvious that this is just p_i, the fraction of jobs that are routed to server i. Hence, Lemma 3.7 has the following corollary.

Corollary 7.4 *Let ψ be a policy applied to an $(a_1, a_2, \ldots, a_n)$ system with $W(\psi) < \infty$. Suppose that the splitting sequence $\underline{\delta}(i)$ is a regular sequence for some $i \in \{1, 2, \ldots, n\}$. Then $\underline{\delta}(i)$ is a regular sequence with density $d = a_i$.*

In the sequel when we say that the splitting sequence $\underline{\delta}(i)$ is regular or balanced for the applied policy ψ in an $(a_1, a_2, \ldots, a_n)$ system then we mean that the sequence is regular or balanced respectively with the appropriate density $d = a_i$.

If for $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is regular then the following theorem yields a nice bound for the maximal waiting time of jobs routed to server i.

Theorem 7.5 *If for $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is regular then $v_i^t < 1$ for every $t \in A(i)$.*

Proof. Suppose that $v_i^t > 0$ for some $t \in A(i)$ and let $t^* = \max\{t' < t : v_i^{t'} = 0\}$. Then $v_i^{t^*} = 0$, $u_i^{t^*} = u_i^{t^*+1} = \ldots = u_i^t$ and thus $v_i^t = \frac{1}{a_i} \cdot \sum_{t'=t^*}^{t-1} \delta_{t'}(i) - (t - t^*)$ by Lemma 7.1. Because $\underline{\delta}(i)$ is regular we have that $\sum_{t'=t^*}^{t} \delta_{t'}(i) \leq \lceil a_i \cdot (t - t^* + 1) \rceil$ and thus

$$\sum_{t'=t^*}^{t-1} \delta_{t'}(i) \leq \lceil a_i \cdot (t - t^* + 1) \rceil - 1 < a_i \cdot (t - t^* + 1).$$

So, $v_i^t < t - t^* + 1 - (t - t^*) = 1$. □

Corollary 7.6 *If for $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is regular then $v_i^t < \frac{1}{a_i}$ for every $t \in \mathbb{Z}_{\geq 0}$.*

The following lemma implies a bound on the total time a server is idle when the splitting sequence is regular.

Lemma 7.7 *If a policy ψ is applied such that for some $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is regular then $d_i(\psi) \leq 1$.*

Proof. Since $\underline{\delta}(i)$ is regular we have for every $t \in \mathbb{Z}_{\geq 0}$ that $N_i^t \geq \lfloor t \cdot a_i \rfloor$. Hence by Lemma 4.2 we have that $d_i \leq 1$. □

If for an $(a_1, a_n, \ldots, a_n)$ system there exists a policy ψ such that for every $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is balanced then we say that the $(a_1, a_2, \ldots, a_n)$ system is balanceable. The policy ψ is then called a balanced policy. Because every balanced sequence is eventually a regular sequence we have for a balanceable system also that there exists a policy ϕ such that for every $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is a regular sequence. Namely, one can take $\phi = \psi^s$ for some $s \in \mathbb{Z}_{\geq 0}$, s large enough. We define such a policy ϕ to be a regular policy. Observe that for a regular policy the set of regular sets $\{A(i)\}_{i=1}^n$ is an exact cover of $\mathbb{Z}_{\geq 0}$. Therefore it is also said that there exists an exact cover set for balanceable densities $(a_1, a_2, \ldots, a_n)$.

Further if Q is a finite set, a so-called alphabet and $I \subseteq \mathbb{Z}$ is a subinterval of $\mathbb{Z}$ then a mapping $W : I \to Q$ is called a I - word or just word. Thus a word is a sequence of letters from the alphabet Q. A subword of W is the restriction of W to some interval $J \subseteq I$. For a finite (sub)word W we denote the length of W by $|W|$. Further for a letter $a \in Q$ the number of a's in W is denoted by $|W|_a$ and the support set of letter a is denoted by $S_a := \{t \in I : W(t) = a\}$. For an I -word W we have that letter a has density d if for any non-decreasing sequence of finite subintervals I_i with union I we have that $\frac{|I_i \cap S_a|}{|I_i|}$ tends to d. A policy ψ for an $(a_1, a_2, \ldots, a_n)$ system can be identified with a $\mathbb{Z}_{\geq 0}$ - word or $\mathbb{N}$ - word on the alphabet $\{1, 2, \ldots, n\}$ by taking $W(t) = k_t(\psi)$ for $t = 0, 1, \ldots$ or $W(t) = k_{t-1}(\psi)$ for $t = 1, 2, \ldots$ respectively. In the literature also the terms balanced word, regular word and Beatty word are used. Each of these terms has the obvious meaning. For example a balanced word is a word such that for every letter $a \in Q$ it holds for any two finite subwords W and W' with $|W| = |W'|$ that $||W|_a - |W'|_a| \leq 1$. Thus a balanced policy corresponds to a balanced word (with the appropriate densities of the letters) and in the same way a regular policy corresponds to a regular (Beatty) word.

Remember that $I_t = \{0, 1, \ldots, t-1\}$. For $i = 1, 2, \ldots, n$ and $t \in \mathbb{Z}_{\geq 0}$ we put $A^t(i) := A(i) \cap I_t$. Note that $|A^t(i)| = N_i^t$. If $\lim_{t\to\infty} \frac{\sum_{\tau \in A^t(i)} v_i^\tau}{N_i^t}$ exists for $i \in \{1, 2, \ldots, n\}$ then we define

$$z_i := \lim_{t\to\infty} \frac{\sum_{\tau \in A^t(i)} v_i^\tau}{N_i^t}$$

as the long-run average waiting time of jobs routed to server i.

Lemma 7.8 *If for an $(a_1, a_2, \ldots, a_n)$ system the applied policy ψ is such that*

$$\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$$

exists and is finite for $i = 1, 2, \ldots, n$ then $W(\psi) = \sum_{i=1}^n a_i \cdot z_i$.

Proof.

$$W(\psi) = \limsup_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} v_{k_t}^t = \limsup_{\tau\to\infty} \frac{1}{\tau} \sum_{i=1}^n \sum_{t\in A^\tau(i)} v_i^t \leq$$

$$\sum_{i=1}^n \limsup_{\tau\to\infty} \frac{1}{\tau} \sum_{t\in A^\tau(i)} v_i^t \leq \sum_{i=1}^n \limsup_{\tau\to\infty} \frac{\sum_{t\in A^\tau(i)} v_i^t}{N_i^\tau} = \sum_{i=1}^n z_i < \infty.$$

Hence, (19) holds and

$$W(\psi) = \lim_{\tau\to\infty} \frac{1}{\tau} \sum_{i=1}^n \sum_{t\in A^\tau(i)} v_i^t = \lim_{\tau\to\infty} \sum_{i=1}^n a_i \sum_{t\in A^\tau(i)} \frac{v_i^t}{N_i^\tau} =$$

$$\sum_{i=1}^n a_i \cdot \lim_{\tau\to\infty} \sum_{t\in A^\tau(i)} \frac{v_i^t}{N_i^\tau} = \sum_{i=1}^n a_i \cdot z_i.$$

□

If $a_i = \frac{p}{q}$ with $p, q \in \mathbb{N}$ then we say that the routing to server i is proportional periodic with period q if from every q consecutively arriving jobs exactly p jobs are routed to server i.

If the routing to server i is proportional periodic with period q then we have by the argument used in the proof of Theorem 6.6 that server i has no idle periods from some moment $t \in \mathbb{Z}_{\geq 0}$, $t < q$ and thus we can define t_i as in (52). Then we have the following lemma.

Lemma 7.9 *If for some $i \in \{1, 2, \ldots, n\}$ the routing to server i is proportional periodic with period q then we have that $t_i \in \mathbb{Z}_{\geq 0}$, $t_i < q$, $v_i^{t_i} = 0$ and for every $t \geq t_i$ we have that*

$$v_i^t = \frac{q}{p} \cdot \sum_{t'=t_i}^{t_i+t-1} \delta_{t'}(i) - (t - t_i) \tag{68}$$

and $v_i^{t+q} = v_i^t$.

Proof. By the arguments used in the proof of Theorem 6.6 it follows that $t_i \in \mathbb{Z}_{\geq 0}$, $t_i < q$, $v_i^{t_i} = 0$. Further $u_i^{t_i} = u_i^{t_i+1} = \ldots$. So, by Lemma 7.1 we have that $v_i^t = \frac{q}{p} \cdot \sum_{t'=t_i}^{t_i+t-1} \delta_{t'}(i) - (t - t_i)$ for every $t \geq t_i$. Hence for every $t \geq t_i$ we have that

$$v_i^{t+q} - v_i^t = \frac{q}{p} \cdot \sum_{\tau=t_i+t}^{t_i+t+q-1} \delta_\tau(i) - q = \frac{q}{p} \cdot p - q = 0.$$

□

Corollary 7.10 *If $a_i = \frac{p}{q}$ and the routing to server i is proportional periodic with period q then $\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$ exists and*

$$z_i = \frac{1}{p} \cdot \sum_{t\in B^{t'}} v_i^t \tag{69}$$

for any $t' \geq t_i$, where $B^{t'} = \{t', t'+1, \ldots, t'+q-1\} \cap A(i)$.

By Lemma 7.8 and Corollary 7.10 we have the following theorem.

Theorem 7.11 *If a p.p policy ψ is applied to an $(x_1, x_2, \ldots, x_n)$ system then*

$$\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$$

exists and is finite for $i = 1, 2, \ldots, n$ and $W(\psi) = \sum_{i=1}^n a_i \cdot z_i$.

We have the following theorem which gives an explicit formula for the long-run average waiting time of customers routed to some server i if the routing to that server i is regular.

Theorem 7.12 *If policy ψ is applied to an $(a_1, a_2, \ldots, a_n)$ system such that for some $i \in \{1, 2, \ldots, n\}$ the splitting sequence $\underline{\delta}(i)$ is a regular sequence then*

$$\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$$

exists. Moreover, if $a_i = \frac{p}{q}$ with $\gcd(p,q) = 1$ then $z_i = \frac{1}{2} - \frac{1}{2p}$ and if a_i is irrational then $z_i = \frac{1}{2}$.

Proof. We first treat the case that a_i is rational, $a_i = \frac{p}{q}$ with $\gcd(p,q) = 1$. Since $\lceil q \cdot a_i \rceil = \lfloor q \cdot a_i \rfloor = p$ we have by definition for the splitting sequence $\underline{\delta}(i)$ that every q -block contains exactly p ones. Thus the routing to server i is proportional periodic with period q. So, by Corollary 7.10 we have that $\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$ exists and (69) holds. Let $f : \mathbb{Z}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be the function given by $f(t) = v_i^t$ for $t \in \mathbb{Z}_{\geq 0}$. We have for $t' \geq t_i$ that

$$z_i = \frac{1}{p} \cdot \sum_{t\in B^{t'}} v_i^t = \frac{1}{p} \cdot \sum_{t\in B^{t'}} f(t). \tag{70}$$

Let $C = \{0, \frac{1}{p}, \ldots, \frac{p-1}{p}\}$. We claim that f maps $B^{t'}$ bijectively to C for $t' \geq t_i$. By this claim and (70) we have that

$$z_i = \sum_{k=0}^{p-1} \frac{1}{p} \cdot \frac{k}{p} = \frac{1}{p^2} \cdot \frac{1}{2} p \cdot (p-1) = \frac{1}{2} - \frac{1}{2p}.$$

Hence, for the rational case it suffices to prove the claim. We first prove that $f(t) \in C$ for every $t \in A(i)$. By (5), (6) and induction we have that $p \cdot v_i^t \in \mathbb{Z}_{\geq 0}$ for every $t \in \mathbb{Z}_{\geq 0}$. Further by

Theorem 7.5 we have that $v_i^t < 1$ for every $t \in A(i)$. Hence $f(t) = v_i^t \in C$ for every $t \in A(i)$. Since $|B^{t'}| = |C| = p$ it suffices to show that f is injective on $B^{t'}$. Suppose $t_i \le t_a < t_b$ and $v_i^{t_a} = v_i^{t_b}$. Let $P = N_i^{t_b} - N_i^{t_a}$. By (68) we have that $\frac{q}{p} \cdot P = t_b - t_a$ and thus $q \cdot P = p \cdot (t_b - t_a)$. Since $\gcd(p, q) = 1$ it follows that there exists a positive integer k such that $P = k \cdot p$ and $t_b - t_a = k \cdot q \ge q$. Because $|t_1 - t_0| < q$ if $t_0, t_1 \in B^{t'}$ it follows that f is injective on $B^{t'}$ and the claim is proved.

Now we consider the case that a_i is irrational. If the routing to server i is according to some arbitrary splitting sequence $b = (b_0, b_1, b_2, \dots)$ of zeros and ones then it is clear that v_i^t depends on $b_0, b_1, \dots, b_{t-1}$, but not on $b_t, b_{t+1}, \dots$. We define for $t = 1, 2, \dots$ the functions $f_t : \{0,1\}^t \to \mathbb{R}_{\ge 0}$ by $f_t = f_t(b_0, b_1, \dots, b_{t-1}) := v_i^t$ if the routing to server i is according to the splitting sequence $(b_0, b_1, \dots)$. Further we put $f_0 = 0$. It is clear that these functions f_t have the following properties:

$$f_t(b_0, b_1, \dots, b_{t-1}) \ge f_{t-1}(b_1, b_2, \dots, b_{t-1}) \tag{71}$$

for every $t \in \mathbb{N}$ and

$$f_{t+k}(0, 0, \dots, 0, b_0, b_1, \dots, b_{t-1}) = f_t(b_0, b_1, \dots, b_{t-1}) \tag{72}$$

for every $k, t \in \mathbb{N}$.

Since $\underline{\delta}(i)$ is a regular sequence we have by Corollary 7.3 that there exists a $\theta \in \mathbb{R}$ such that $\delta_t(i) = \lfloor (t+1)a_i + \theta \rfloor - \lfloor t \cdot a_i + \theta \rfloor$ or $\delta_t(i) = \lceil (t+1)a_i + \theta \rceil - \lceil t \cdot a_i + \theta \rceil$ for $t = 0, 1, \dots$. We assume that for some $\theta \in \mathbb{R}$ we have that

$$\delta_t(i) = \lfloor (t+1)a_i + \theta \rfloor - \lfloor t \cdot a_i + \theta \rfloor \text{ for } t = 0, 1, \dots. \tag{73}$$

Let $b_t(\theta) = \lfloor (t+1)a_i + \theta \rfloor - \lfloor t \cdot a_i + \theta \rfloor$ for $t \in \mathbb{Z}$. Then we have that

$$b_t(\theta) = b_t(\theta + 1) \text{ for every } t \in \mathbb{Z}, \theta \in \mathbb{R} \tag{74}$$

and

$$b_{t-m}(\theta + m \cdot a_i) = b_t(\theta) \text{ for every } t, m \in \mathbb{Z}, \theta \in \mathbb{R}. \tag{75}$$

If the routing to server i is according to (73) then $v_i^t = f_t(b_0(\theta), b_1(\theta), \dots, b_{t-1}(\theta))$ for every $t \in \mathbb{Z}_{\ge 0}$. For $t \in \mathbb{Z}_{\ge 0}, \theta \in \mathbb{R}$ we define $g(\theta, t) := f_t(b_0(\theta), b_1(\theta), \dots, b_{t-1}(\theta))$ and $g'(\theta, t) := f_t(b_{-t}(\theta), b_{-t+1}(\theta), \dots, b_{-1}(\theta))$. Then we have by (75) that $g(\theta, t) = g'(\theta + t \cdot a_i, t)$. Further by (74) we have that $g(\theta, t)$ and $g'(\theta, t)$ are both periodic in θ with period 1. Moreover by (71) we have that $g'(\theta, t)$ is non-decreasing in t. Hence $g'_\infty(\theta) := \lim_{t\to\infty} g'(\theta, t)$ exists and by Corollary 7.6 we have that $0 \le g'_\infty(\theta) \le \frac{1}{a_i}$ for every $\theta \in \mathbb{R}$. Further $g'_\infty(\theta)$ is also periodic in θ with period 1. If the routing to server i is according to (73) for some fixed $\theta = \theta_0$ then we have that

$$\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t = \lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} g(\theta_0, t) =$$

$$\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} g'(\theta_0 + t \cdot a_i, t) = \lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} g'_\infty(\theta_0 + t \cdot a_i).$$

Since $g'_\infty(\theta)$ is periodic in θ with period 1 and a_i is irrational we have by the ergodic theorem of Weyl and Von Neumann (see for example [15]) that

$$\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} g'_\infty(\theta_0 + t \cdot a_i) = \int_0^1 g'_\infty(\theta) d\theta.$$

So, if the routing to server i is according to (73) for some $\theta \in \mathbb{R}$ then $\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t = \int_0^1 g'_\infty(\theta) d\theta$ and in particular $\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t$ exists and is independent of θ. By Lemma 7.7 we have that $\lim_{t\to\infty} u_i^t = \frac{d_i}{a_i} \leq \frac{1}{a_i}$. Further by Corollary 7.6 we have that $v_i^t < \frac{1}{a_i}$ for every $t \in \mathbb{Z}_{\geq 0}$. So, $v_i^t - u_i^t \in [-\frac{1}{a_i}, \frac{1}{a_i}]$ for every $t \in \mathbb{Z}_{\geq 0}$ and thus the condition of Lemma 3.5 is satisfied for the considered server i. So, because $\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t$ exists and by Lemma 3.5, we have that $z_i = \lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$ exists and

$$z_i = \lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t - \frac{1}{2a_i} + \frac{1}{2} = \int_0^1 g'_\infty(\theta) d\theta - \frac{1}{2a_i} + \frac{1}{2}. \tag{76}$$

Now we consider the case that for some $\theta \in \mathbb{R}$ we have that

$$\delta_t(i) = \lceil (t+1)a_i + \theta \rceil - \lceil t \cdot a_i + \theta \rceil \text{ for } t = 0, 1, \ldots. \tag{77}$$

Then it follows analogously to the case that the routing to server i is according to (73) that $\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$ exists and that z_i is independent of θ.

Choose a $\theta \in \mathbb{R}$ such that $1, a_i, \theta$ are linearly independent over $\mathbb{Z}$. Then $\lceil t \cdot a_i + \theta \rceil = \lfloor t \cdot a_i + \theta \rfloor + 1$ for every $t \in \mathbb{Z}_{\geq 0}$ and thus

$$\lceil (t+1)a_i + \theta \rceil - \lceil t \cdot a_i + \theta \rceil = \lfloor (t+1)a_i + \theta \rfloor - \lfloor t \cdot a_i + \theta \rfloor \text{ for } t = 0, 1, \ldots .$$

Hence the routing to server i according to (73) is exactly the same as the routing to server i according to (77) for such θ. So, splitting sequences according to (77) have the same z_i as splitting sequences according to (73) and thus (76) also holds for splitting sequences according to (77). Thus $\lim_{t\to\infty} \frac{\sum_{\tau\in A^t(i)} v_i^\tau}{N_i^t}$ exists if the splitting sequence $\underline{\delta}(i)$ is regular and z_i is the same for every regular splitting sequence.

To complete the proof it now suffices to show that $z_i = \frac{1}{2}$ for a particular regular splitting sequence with density a_i. Let the routing to server i be such that $N_i^t = \lceil t \cdot a_i \rceil$ for every $t \in \mathbb{Z}_{\geq 0}$. Then we have for the corresponding splitting sequence $\underline{\delta}(i)$ that

$$\delta_t(i) = N_i^{t+1} - N_i^t = \lceil (t+1) \cdot a_i \rceil - \lceil t \cdot a_i \rceil \text{ for } t = 0, 1, \ldots .$$

Hence $\underline{\delta}(i)$ is according to (77) with $\theta = 0$ and thus $\underline{\delta}(i)$ is indeed a regular sequence. By Lemma 4.2 we have for the applied policy ψ that $d_i(\psi) = 0$ and thus $u_i^t = 0$ for every $t \in \mathbb{Z}_{\geq 0}$. So, by (26) we have for every $t \in \mathbb{Z}_{\geq 0}$ that

$$v_i^t = \frac{N_i^t}{a_i} - t = \frac{\lceil t \cdot a_i \rceil - t \cdot a_i}{a_i}. \tag{78}$$

Since a_i is irrational we can apply the ergodic theorem of Weyl and von Neumann again and by (78) we have that

$$\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t = \lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} \frac{\lceil t \cdot a_i \rceil - t \cdot a_i}{a_i} = \int_0^1 \frac{x}{a_i} dx = \frac{1}{2a_i}. \tag{79}$$

Hence by (76) we have that $z_i = \frac{1}{2}$. □

Remark. For the irrational case we have used the same technique as in the proof of Theorem 3.2 in [3] .

The following Theorem gives a criterion for the attainability of the lower bounds of Corollary 6.16.

Theorem 7.13 *If for an $(x_1, x_2, \ldots, x_n)$ system the fractions $(\frac{x_1}{T}, \frac{x_2}{T}, \ldots, \frac{x_n}{T})$ are balanceable then the lower bounds of Corollary 6.16 can be attained, i.e*

$$\widetilde{S} = \frac{n}{2} - \frac{C(x_1, x_2, \ldots, x_n)}{2T} \tag{80}$$

and

$$\widetilde{W} = \frac{1}{2} - \frac{C(x_1, x_2, \ldots, x_n)}{2T}, \text{ where } C(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \gcd(x_i, T). \tag{81}$$

Moreover, every regular policy attains the lower bounds of Corollary 6.16 and is optimal.

Proof. Let ψ be a regular policy for the system. If for some $i \in \{1, 2, \ldots, n\}$ we write $a_i = \frac{x_i}{T}$ as $\frac{p}{q}$ with $\gcd(p, q) = 1$ then $p = \frac{x_i}{\gcd(x_i,T)}$. Hence by Theorem 7.12 we have that $z_i = \frac{1}{2} - \frac{\gcd(x_i,T)}{2x_i}$ for $i = 1, 2, \ldots, n$. So, by Lemma 7.8 we have that

$$W(\psi) = \sum_{i=1}^{n} a_i \cdot z_i = \frac{1}{2} \cdot \sum_{i=1}^{n} a_i - \frac{\sum_{i=1}^{n} \gcd(x_i, T)}{2T} = \frac{1}{2} - \frac{C(x_1, x_2, \ldots, x_n)}{2T}.$$

Thus ψ attains the lower bounds of Corollary 6.16 and ψ is an optimal policy. □

In fact it holds much more general that a regular policy is optimal if the given (optimized) fractions are balanceable. Namely, in [1] and [3] the following is shown for the more general case of stochastically independent stationary interarrival and service times. Let p be fixed. Then, if you want to route at least a fraction p of the arriving jobs (customers) to a certain server, then the best way to route to that server is according to a regular splitting sequence of density p. More precisely, the authors have shown the following: Let $f_k = f_k(b_1, b_2, \ldots, b_k) : \mathbb{Z}^k_{\geq 0} \to \mathbb{R}$ be a sequence of functions with the following properties:

$$f_k \text{ is multimodular on } \mathbb{Z}^k_{\geq 0}, \tag{82}$$

for every sequence $\{b_k\}$

$$f_k(b_1, b_2, \ldots, b_k) \geq f_{k-1}(b_2, b_3, \ldots, b_k) \text{ for every } k \in \mathbb{N}, \tag{83}$$

For every sequence $\{b_k\}$ there exists a sequence $\{c_k\}$ such that

$$\text{for every } k, m \in \mathbb{N}, k > m, f_k(c_1, c_2, \dots, c_{k-m}, b_1, b_2, \dots, b_m) = f_m(b_1, b_2, \dots, b_m) \tag{84}$$

and

$$f_k(b_1, b_2, \dots, b_k) \text{ is non-decreasing in } b_i \text{ for } i = 1, 2, \dots, k. \tag{85}$$

Then a regular sequence of density p minimizes

$$\limsup_{N\to\infty} \frac{1}{N} \cdot \sum_{k=1}^{N} f_k(b_1, b_2, \dots, b_k)$$

among all splitting sequences $(b_1, b_2, \dots)$ of density at least p. Moreover, let $\{T_k\}_{k\in\mathbb{N}}$ be the sequence of arrival times, with the convention that $T_1 = 0$ and the queue is empty at time 0 and let $(b_1, b_2, \dots)$ be a splitting sequence of integers, where b_i gives the number of customers admitted to the queue at time T_i. Further let $\{\tau_k\}$ be the sequence of interarrival times and $\{\sigma_k\}$ the sequence of service times in the server and let $\mathbf{E}_{\sigma,\tau}$ denote the expectation w.r.t the service times and interarrival times. Then it is proved in [2] that the functions $\mathbf{E}_{\sigma,\tau}W_k(b_1, b_2, \dots, b_k)$, where $W_k(b_1, b_2, \dots, b_k)$ is the workload in the queue at time T_k under control sequence $(b_1, b_2, \dots)$ have the above properties if $\{\tau_k\}$ and $\{\sigma_k\}$ are stochastically independent stationary sequences. Note that (84) is fulfilled because $\mathbf{E}_{\sigma,\tau}W_k(0, 0, \dots, 0, b_1, b_2, \dots, b_m) = \mathbf{E}_{\sigma,\tau}W_m(b_1, b_2, \dots, b_m)$ if $k > m$. The multimodularity of the workload in case of fixed deterministic interarrival times and service times is explicitly proved in [3] and it follows in particular that the sequence of functions f_t that we defined in the proof of Theorem 7.12 have the above properties. So, for our case we have for every server i that routing to server i according to a regular splitting sequence of density a_i minimizes $\limsup_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t$ (and also $\limsup_{\tau\to\infty} \frac{\sum_{t\in A^\tau(i)} v_i^t}{N_i^\tau}$), given that at least a fraction a_i of the arriving jobs has to be routed to server i. The fact that multimodular functions are minimized by regular sequences originates from [8], in which the optimality of the regular splitting sequence is proved in case of an exponential server queue.

Further we want to note that in [2] it is concluded that not only the regular splitting sequences are optimal, but also the slightly more general balanced sequences if the traffic intensity is less than one. Then namely with probability one the queue becomes empty after some time. In that case only the " right tail " of the splitting sequence matters and thus a balanced sequence is just as good as a regular sequence since every balanced sequence is eventually regular. However, in our deterministic case it can happen that a queue never empties after some moment, although the queue is stable. Therefore in our case it can happen that a balanced (but not regular) splitting sequence of a certain density behaves worse than a regular sequence of the same density. For example in case of $a_i = \frac{1}{2}$ and thus service time 2, we compare for server i the regular splitting sequence $(10)^\infty$ with the balanced (but not regular) splitting sequence $1(10)^\infty$. Then for the regular sequence we have that $v_i^t = 0$ if $t \in \mathbb{Z}_{\geq 0}$ is even and $v_i^t = 1$ if $t \in \mathbb{Z}_{\geq 0}$ is odd. Hence $\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t = \frac{1}{2}$ and $z_i = 0$. For the balanced sequence we have that $v_i^0 = 0$, $v_i^t = 2$ for $t \in \mathbb{N}$ even and $v_i^t = 1$ for $t \in \mathbb{N}$ odd. Hence $\lim_{\tau\to\infty} \frac{1}{\tau} \cdot \sum_{t=0}^{\tau-1} v_i^t = \frac{3}{2}$ and $z_i = 1$. So, for this example we see for the balanced sequence that indeed the queue never becomes empty after $t = 0$ and that the balanced sequence is worse than the regular sequence of the same density.

Now we know that if the fractions $(a_1, a_2, \dots, a_n)$ are balanceable then a policy corresponding to a regular word with these fractions is optimal, the natural question is which sets of fractions

$(a_1, a_2, \ldots, a_n)$ with $\sum_{i=1}^{n} a_i = 1$ are balanceable. This problem in general is studied in [2] and [10] and in [7], [14], [18] and [19] and several other papers it is studied for several special cases. See [19] for other references. We mention here some of the results. In case $n = 2$ every pair of fractions $(p, 1-p)$ is balanceable. This is easily seen since if a sequence of zeros and ones is balanced with density p then the complementary sequence is " automatically " balanced with density $1-p$. For the rational case if $n = 2$ we have seen in Corollary 6.17 that the lower bounds of Corollary 6.16 can be attained and now it follows that the regular policies attain those lower bounds. In the irrational case of $n = 2$ we have by Corollary 5.12 that $\widetilde{W} = \frac{1}{2}$ and it follows by Theorem 7.12 that the regular policies are indeed optimal. So, for $n = 2$ the problem is completely solved and we know the optimal policies for our queueing system.
For $n > 2$ it is much more complicated. In fact it is then rather rare that a set of fractions is balanceable, especially if most of the fractions are mutually different. If most of the fractions are equal to each other then it is more likely that they are balanceable. For example if they are all equal to $\frac{1}{n}$ or there are only two distinct fractions then they are always balanceable. If they are all equal to $\frac{1}{n}$ then the corresponding regular optimal policy is just the round robin policy which has average waiting time $W = 0$ for the corresponding $(1, 1, \ldots, 1)$ system. Further if the fractions are balanceable and one of them is irrational then they have to be all irrational and all the regular words in this case are classified in [7]. Note that Conjecture 5.13, Theorem 3.2, Theorem 7.12 and Lemma 7.8 imply that if irrational fractions $(a_1, a_2, \ldots, a_n)$ are balanceable then $\dim_{\mathbb{Q}}(a_1\mathbb{Q} + a_2\mathbb{Q} + \ldots + a_n\mathbb{Q}) = 2$. This follows indeed from the classification in [7]. Further for $n > 2$ it is known that if the fractions are distinct then they must be all rational. In fact for every $n > 2$ there is exactly one set of distinct balanceable fractions known. Namely, the fractions $\{\frac{2^{n-1}}{2^n-1}, \frac{2^{n-2}}{2^n-1}, \ldots, \frac{1}{2^n-1}\}$ corresponding to a $(2^{n-1}, 2^{n-2}, \ldots, 1)$ system. For example for $n = 4$ the fractions $(\frac{8}{15}, \frac{4}{15}, \frac{2}{15}, \frac{1}{15})$ are balanceable and $(abacabadabacaba)^\infty$ is a regular word on the alphabet $\{a, b, c, d\}$ with those fractions. Hence the corresponding policy $(1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1)^\infty$ is optimal for an $(8, 4, 2, 1,)$ system. It is natural to make the following conjecture which is known as Fraenkel's conjecture.

Conjecture 7.14 *A set of distinct fractions $\{a_1, a_2, \ldots, a_n\}$ with $n \geq 3$ and $\sum_{i=1}^{n} a_i = 1$ is balanceable if and only if*

$$\{a_1, a_2, \ldots, a_n\} = \{\frac{2^{n-1}}{2^n - 1}, \frac{2^{n-2}}{2^n - 1}, \ldots, \frac{1}{2^n - 1}\}.$$

For $n = 3$ the conjecture is proved in [10] and [17] and for $n = 4$ in [2]. Further in [18] and [19] the conjecture is proved for $n = 5$ and $n = 6$. Further in [10] and [17] all the regular words corresponding to the balanceable fractions for $n = 3$ are classified. The same is done in [10] for $n = 4$ except for a special case. Namely, not for rational fractions such that for the corresponding (x_1, x_2, x_3, x_4) system $\gcd(x_i, T) > 1$ for some $i \in \{1, 2, 3, 4\}$.
Finally we remark that a set of fractions $(a_1, a_2, \ldots, a_n)$ is balanceable if and only if there exists a solution $d_1, d_2, \ldots, d_n$ of (36) such that all the inequalities $\sum_{i=1}^{n} \lceil t \cdot a_i - d_i \rceil \leq t$ for $t = 0, 1, \ldots$ hold with equality. From such a solution if it exists a regular word with the given fractions can be obtained by applying the SG algorithm to the solution.

8 Algorithms to find good policies

For large systems it is hard to find an optimal policy by solving the mathematical programming problem. In this section we will consider some greedy algorithms that can be used to obtain rather good (but not necessarily optimal) policies. We apply these algorithms in the rational case, but they can also be applied to the general case. However, in the rational case the obtained policy ψ will be (ultimately) periodic and $W(\psi)$ can be determined, while in case of irrational a_i the performance $W(\psi)$ can only be estimated. Further in the rational case we can try to improve the obtained policy by iteratively using the SG algorithm, as we showed in section 6.

The first greedy algorithm that we consider is based on the fact that to minimize W we have to minimize S. So, when at some time $t_0 \in \mathbb{Z}_{\geq 0}$ we have to route the incoming job to a server then it may be a good idea to determine k_{t_0} in such a way that S^{t_0+1} is minimized. We call an algorithm that follows this rule an OSSM (One Step S Minimalization) algorithm. We have (see property w5 of the w_i^t for the rational case and the irrational case is similar) that

$$S^{t_0+1} - S^{t_0} = \sum_{k \neq k_{t_0}} \max(a_k - a_k \cdot v_k^{t_0}, 0). \tag{86}$$

So, defining

$$r_i^{t_0} = \max(a_i - a_i \cdot v_i^{t_0}, 0), \tag{87}$$

for $i \in \{1, 2 \ldots, n\}$ it follows from (86) that at time t_0 an OSSM algorithm routes the job to a server l for which

$$r_l^{t_0} = \max_{i \in \{1,2,\ldots,n\}} r_i^{t_0}. \tag{88}$$

Applying this rule we still have to choose to which server the job is routed if $r_i^{t_0}$ is maximal for several servers i. First we consider the case that $r_1^{t_0} = r_2^{t_0} = \cdots = r_n^{t_0} = 0$. Then $S^{t_0+1} = S^{t_0}$ no matter to which server the job is routed. Therefore in this case we let the algorithm already minimize S^{t_0+2} instead of S^{t_0+1}. This can be done by redefining the variables $r_i^{t_0}$ as $r_i^{t_0} = \max(2a_i - a_i \cdot v_i^{t_0}, 0)$ and then choose the server for which the new $r_i^{t_0}$ is maximal. If for the new $r_i^{t_0}$ it again holds that $r_1^{t_0} = r_2^{t_0} = \cdots = r_n^{t_0} = 0$, then the $r_i^{t_0}$ can be redefined as $r_i^{t_0} = \max(3a_i - a_i \cdot v_i^{t_0})$, etc. Now we still have to consider the case that $r_i^{t_0}$ is maximal for the servers $i_1, i_2, \ldots, i_m$ with $m > 1$ and that $r_{i_1}^{t_0} = r_{i_2}^{t_0} = \cdots = r_{i_m}^{t_0} > 0$. An OSSM algorithm has to route the job to one of these m servers $i_1, i_2, \cdots, i_m$ and we introduce two variants of the OSSM algorithm, denoted by OSSM1 and OSSM2. If we assume without loss of generality that $a_{i_1} \geq a_{i_2} \geq \cdots \geq a_{i_m}$ then OSSM 1 routes the job to server i_1 , the fastest server among them, while OSSM2 routes the job to server i_m, the slowest server among them. We only consider these two variants, but other variants are possible. We have the following theorem for every OSSM algorithm.

Theorem 8.1 *In case $n = 2$ a policy ψ obtained by applying an OSSM algorithm is optimal.*

Proof. It suffices for both the rational and irrational case to show that $S^t(\psi) < 1$ for every $t \in \mathbb{Z}_{\geq 0}$. Namely, then in the rational case we have that $S(\psi) \cdot T \leq T - 1$ and thus $S(\psi) \leq 1 - \frac{1}{T}$.

Hence ψ is optimal by Corollary 6.17. For the irrational case it follows that $S(\psi) \leq 1$ and thus ψ is optimal by Corollary 5.12. So if ψ is not an optimal strategy then there exists a $t_0 \in \mathbb{N}$ such that $S^{t_0} = a_1 \cdot v_1^{t_0} + a_2 \cdot v_2^{t_0} \geq 1$, but $S^{t_0-1} = a_1 \cdot v_1^{t_0-1} + a_2 \cdot v_2^{t_0-1} < 1$. We may assume that $k_{t_0-1} = 1$ without loss of generality. Then $S^{t_0} = S^{t_0-1} + r_2^{t_0-1}$ and $r_1^{t_0-1} \geq r_2^{t_0-1} > 0$. Thus $v_1^{t_0-1} < 1$ and $v_2^{t_0-1} < 1$. Hence

$$1 \leq a_1 \cdot v_1^{t_0} + a_2 \cdot v_2^{t_0} = a_1 \cdot v_1^{t_0-1} + 1 - a_1 < 1,$$

which is a contradiction. □

Remarks.

1. In the rational case we can multiply with T as usual and thus the integer numbers $(r_i^{t_0})^* := T \cdot r_i^{t_0} = \max(x_i - w_i^{t_0}, 0)$ for $i = 1, 2, \ldots, n$ instead of the $r_i^{t_0}$ could be used when k_{t_0} is determined by an OSSM algorithm.
2. Note that the OSSM2 algorithm yields the policy $(1, 2, 1, 3, 2, 1, 2, 1, 3)^\infty$ for the $(4, 3, 2)$ system that we used in section 6 as example (see Table 1), while the OSSM1 algorithm yields the policy $(1, 2, 1, 3, 2, 1, 1, 2, 3)^\infty$ for this system.

The next greedy algorithm we consider follows the rule that the incoming job is routed to the server that is the first that can start with the processing of that job. So at time t the algorithm routes the job to a server i for which v_i^t (or $\frac{w_i^t}{x_i}$ in the rational case) is minimal. If for several servers $i_1, i_2, \ldots, i_m$ it holds that v_i^t is minimal then the job will be routed to a server which is the fastest of those m servers. So, if those m servers are ordered such that $a_{i_1} \geq a_{i_2} \geq \ldots \geq a_{i_m}$ then the job is routed to server i_1. We will denote this as the SWT (Shortest Waiting Time) algorithm. Just as the OSSM algorithms the SWT algorithm gives an optimal policy in case $n = 2$.

Theorem 8.2 *In case $n = 2$ a policy ψ obtained by applying the SWT algorithm is optimal.*

Proof. It suffices again to show that $S^t(\psi) < 1$ for every $t \in \mathbb{Z}_{\geq 0}$. We prove this by induction on t. For $t = 0$ it is obvious. So suppose it holds for $t = t'$. We may assume that $k_{t'} = 1$ without loss of generality. Suppose that the induction hypothesis does not hold for $t = t' + 1$ and thus $S^{t'+1} \geq 1$. Then analogously to the proof of Theorem 8.1 it follows that $\max(a_2 - a_2 \cdot v_2^{t'}, 0) = r_2^{t'} > 0$ and thus $v_2^{t'} < 1$. Because the SWT algorithm is applied we have that $v_1^{t'} \leq v_2^{t'} < 1$. Hence

$$1 \leq a_1 \cdot v_1^{t'+1} + a_2 \cdot v_2^{t'+1} = a_1 \cdot v_1^{t'} + 1 - a_1 < 1,$$

which is a contradiction. □

Remark. The SWT and the SG algorithm are closely related. Namely, if $h_i^t = a_i \cdot v_i^t$ for $i = 1, 2, \ldots, n$ then it is clear that the SWT algorithm and the SG algorithm route the job arriving at time t to the same server. In Lemma 5.3 we saw that $h_i^t - a_i \cdot v_i^t$ is monotonically non-increasing in t and $\lim_{t\to\infty}(h_i^t - a_i \cdot v_i^t) = 0$ for $i = 1, 2, \ldots, n$. So, if ψ is a policy obtained by using the SWT algorithm and φ is the policy obtained by applying the SG algorithm to $d_1(\psi), d_2(\psi), \dots, d_n(\psi)$ then ψ and φ practically coincide after some time. In the rational case we can say more. Then we have for $i = 1, 2, \dots, n$ that $h_i^t = a_i \cdot v_i^t$ for every $t \geq t_i$, where $t_i \leq T - 1$. Hence $k_t(\varphi) = k_t(\psi)$ for every $t \in \mathbb{Z}_{\geq t'}$ for some $0 \leq t' \leq T - 1$. In fact in most cases we have that $\varphi = \psi$ or $\varphi = \psi^T$. However, if φ' is the policy obtained by applying the SG algorithm to $d_1(\varphi)$, $d_2(\varphi)$, $\dots$, $d_n(\varphi)$ then there might not exist a $t \in \mathbb{Z}_{\geq 0}$ such that $\varphi' = \psi^t$.

The last algorithm we consider is based on the fact that for some server i the splitting sequence $\underline{\delta}_i$ should be as regular as possible. We denote this algorithm as the GR (Greedy Regular) algorithm. We assume without loss of generality that the servers are ordered such that $a_1 \geq a_2 \geq \dots \geq a_n$. The GR algorithm is such that $\underline{\delta}(1)$, the splitting sequence of the routing to server 1, is a regular sequence. Namely, the GR algorithm constructs a policy ψ such that $N_1^t(\psi) = \lceil t \cdot a_1 \rceil$ for every $t \in \mathbb{Z}_{\geq 0}$. Now $\underline{\delta}(1)$ is a regular sequence and we would like the algorithm to construct a policy such that $\underline{\delta}(2), \underline{\delta}(3), \dots$ are also regular sequences. However, from the previous section we know that this is in general not possible. Therefore we let the GR algorithm construct a policy ψ such that for every $i \in \{2, 3, \dots, n\}$ we have the following. Let W be the word on the alphabet $\{1, 2, \dots, n\}$ corresponding to policy ψ and W' be the word on the alphabet $\{i, i+1, \dots, n\}$ obtained from W by omitting the letters $1, 2, \dots, i-1$. Then the support of letter i in the word W' is a regular set.

GR Algorithm. Policy ψ is inductively determined in the following way. Suppose $k_0(\psi)$, $k_1(\psi)$, $\dots$, $k_{t-1}(\psi)$ have been determined. Then

$$k_t(\psi) := \min\{i \in \{1, 2, \dots, n\} : \frac{a_i}{\sum_{j=i}^n a_j} \cdot (1 + \sum_{j=i}^n N_j^t(\psi)) > N_i^t(\psi)\}.$$

Remarks.

1. Note that for $i = n$ it always holds that $\frac{a_i}{\sum_{j=i}^n a_j} \cdot (1 + \sum_{j=i}^n N_j^t(\psi)) > N_i^t(\psi)$.

2. The GR algorithm is very similar to Algorithm 2 in [5] which is denoted as the CGRR (Conditional Generalized Round Robin) policy and which is such that

   $$k_t(\psi) := \min\{i \in \{1, 2, \dots, n\} : \frac{a_i}{\sum_{j=i}^n a_j} \cdot \sum_{j=i}^n N_j^t(\psi) \geq N_i^t(\psi)\}.$$

   Both algorithms share the property that the routing to server 1 is regular and that for every other server i the routing is regular with respect to the servers with index number $\geq i$. These properties are proved in [5]. Further if $a_1, a_2, \dots, a_n$ are rational then both algorithms construct a p.p policy. It is likely that both algorithms have on average the same performance. The main difference is that for the GR algorithm we have that $N_1^t = \lceil t \cdot a_1 \rceil$ for every $t \in \mathbb{Z}_{\geq 0}$, while for the CGRR policy $N_1^t = \lfloor t \cdot a_1 \rfloor$ or $N_1^t = \lceil t \cdot a_1 \rceil$.

In case $n = 2$ it is obvious that a policy obtained by the GR algorithm is regular. So, from the previous section we have the following theorem.

Theorem 8.3 *In case $n = 2$ a policy ψ obtained by applying the GR algorithm is optimal.*

Example. We apply the OSSM1,OSSM2,SWT and GR algorithm to an $(7, 5, 3, 2)$ system. For these algorithms we give the obtained policy ψ , $W(\psi)$, the policy ψ' we get after applying the SG algorithm iteratively to $d_1(\psi)$, $d_2(\psi)$, $d_3(\psi)$, $d_4(\psi)$ and $W(\psi')$.
For OSSM1 we get that $\psi = 1, 2, 1, 3, 2, 1, 4, 1,(2, 3, 1, 2, 1, 1, 2, 3, 4, 1, 2, 1, 3, 2, 1, 1, 4)^\infty$ and $W(\psi) = \frac{1}{2}$. Further $\psi' = (1, 2, 1, 3, 2, 1, 4, 1, 2, 3, 1, 2, 1, 2, 1, 3, 4)^\infty$ and $W(\psi') = \frac{1}{2}$.
For OSSM2 we get that $\psi = 1, 2, 1, 3, 2, 1, 4, 1, 2, 3, 1, 2, 1,(4, 3, 1, 2, 1, 2, 1, 3, 1, 2, 4, 1, 2, 3, 1, 1, 2)^\infty$ and $W(\psi) = \frac{23}{34}$. Further $\psi' = (1, 2, 1, 3, 1, 2, 4, 1, 3, 2, 1, 1, 2, 4, 3, 1, 2)^\infty$ and $W(\psi') = \frac{21}{34}$.
For the SWT algorithm we get that $\psi = 1, 2, 3, 1, 4,(2, 1, 3, 2, 1, 1, 2, 4, 3, 1, 2, 1, 2, 1, 3, 4, 1)^\infty$ and $W(\psi) = \frac{19}{34}$. Further $\psi' = (2, 1, 3, 4, 1, 2, 1, 3, 2, 1, 1, 2, 4, 3, 1, 2, 1)^\infty$ (note that $\psi' = \psi^T$) and $W(\psi') = \frac{1}{2}$.
Finally for the GR algorithm we have that $\psi = (1, 2, 1, 3, 1, 2, 3, 1, 2, 1, 4, 2, 1, 3, 1, 2, 4)^\infty$ with $W(\psi) = \frac{23}{34}$. Further $\psi' = (1, 2, 1, 3, 1, 2, 1, 2, 3, 1, 4, 2, 1, 3, 1, 2, 4)^\infty$ with $W(\psi') = \frac{23}{34}$.

From this example it follows for the rational case that a policy ψ obtained by applying one of the OSSM algorithms or the SWT algorithm is in general not a p.p policy. However, ψ is eventually a p.p policy since the v_i^t get in a loop. In fact ψ^T is always a p.p policy and therefore it suffices to determine $k_t(\psi)$ for $t = 0, 1, \ldots , 2T - 1$ to obtain ψ and $W(\psi)$. It is also possible in the rational case to modify the algorithms such that always a p.p policy is obtained. This can be done by imposing that at moment $t \leq T - 1$ the incoming job may not be routed to server i if $N_i^t = x_i$. In this way we get for the obtained policy ψ that $N_i^T(\psi) = x_i$ for every $i \in \{1, 2, \ldots , n\}$. Then we determine ψ by $\psi := (k_0(\psi), k_1(\psi), \ldots , k_{T-1}(\psi))^\infty$. Then ψ is a p.p policy and the algorithm can be stopped after $k_0(\psi), k_1(\psi), \ldots , k_{T-1}(\psi)$ have been determined. Another advantage is that the $d_i(\psi)$ and $s_i(\psi)$ can be determined by Theorem 6.6 if ψ is a p.p policy. Further the performance of the obtained policy does not change by this modification since the $d_i(\psi)$ will not change.

For several systems with rational a_i we have calculated the long-run average waiting time W for policies obtained by applying the several algorithms. We denote with $W(OS1)$, $W(OS2)$, $W(SWT)$, $W(GR)$ the long-run average waiting time for the policy obtained by applying the $OSSM1$, $OSSM2$, SWT, GR algorithm respectively. In fact we give $T \cdot W(OS1)$ where T is the period of the system and next to $T \cdot W(OS1)$ we give between parentheses T times the average waiting time of the policy we get after applying the SG algorithm iteratively on the solution of (59) obtained by applying the OSSM1 algorithm. For the other algorithms we do the same. Further for some of those systems we have also determined the optimal long-run average waiting time $\widetilde{W}$ in some way. For example by solving (59) or (60) or using the fact that the fractions a_i are balanceable, etc. However, for the "larger" systems we could not do this. Then we denote with $\geq \ldots$ the lower bound following from Corollary 6.16 for the corresponding system. The results are in Table 2.

Table 2

$(x_1, x_2, \dots, x_n)$	T	$T \cdot \widetilde{W}$	$T \cdot W(OS1)$	$T \cdot W(OS2)$	$T \cdot W(SWT)$	$T \cdot W(GR)$
(105,84,54)	243	189	201 (189)	189 (189)	189 (189)	207 (195)
(1000,17,5)	1022	954	963 (954)	963 (954)	968 (954)	966 (966)
(10000,66,1)	10067	9850	9851 (9850)	9851 (9850)	9885 (9850)	9899 (9899)
(10000,2,1)	10003	6666	6667 (6666)	6666 (6666)	9996 (9994)	6667 (6667)
(41,41,41,33)	156	75	80 (75)	78 (75)	78 (75)	108 (108)
(59,59,30,25)	173	$143\frac{1}{2}$	$154\frac{1}{2}$ ($143\frac{1}{2}$)	$154\frac{1}{2}$ ($143\frac{1}{2}$)	$143\frac{1}{2}$ ($143\frac{1}{2}$)	$179\frac{1}{2}$ ($179\frac{1}{2}$)
(118,95,69,35)	317	$351\frac{1}{2}$	$351\frac{1}{2}$ ($351\frac{1}{2}$)	$351\frac{1}{2}$ ($351\frac{1}{2}$)	$361\frac{1}{2}$ ($361\frac{1}{2}$)	$391\frac{1}{2}$ ($384\frac{1}{2}$)
(1000,13,1,1)	1015	$\geq 503\frac{1}{2}$	$1238\frac{1}{2}$ ($1170\frac{1}{2}$)	$1238\frac{1}{2}$ ($1170\frac{1}{2}$)	$1453\frac{1}{2}$ ($1438\frac{1}{2}$)	$903\frac{1}{2}$ ($903\frac{1}{2}$)
(553,447,301,259)	1560	≥ 777	2022 (1983)	2022 (1983)	2043 (1989)	2094 (2018)
(1000,3,2,1)	1006	≥ 500	667 (667)	834 (834)	1487 (1480)	668 (667)
(5,4,3,2,1)	15	5	7 (7)	6 (6)	6 (6)	7 (7)
(13,12,9,5,2)	41	31	36 (36)	36 (36)	37 (35)	48 (42)
(1000,513,102,14,3)	1632	≥ 757	2480 (2431)	2480 (2431)	2585 (2440)	2848 (2677)
(10000,3,3,1,1)	10008	≥ 4996	7498 (7498)	7498 (7498)	19976 (19964)	7498 (7498)
(7,6,5,4,3,2,1)	28	12	21 (18)	15 (15)	15 (15)	18 (17)
(4,4,4,3,3,3,3)	24	0	12 (12)	12(12)	12(12)	12(12)
(33,33,33,33,33, 26,26,26,26)	269	130	244 (162)	244 (154)	262 (130)	330 (330)
(129,101,98,87,82, 61,52,38,29,17)	694	≥ 340	1340 (1274)	1406 (1302)	1332 (1263)	1789 (1402)

We make a few remarks about these results. First of all we see that none of these algorithm always yields an optimal policy and further it is also clear that for all these algorithms there exist systems for which this particular algorithm yields a better policy than the other algorithms. Further it seems that the various algorithms always yield a policy ψ with $W(\psi) \leq \frac{n-1}{2}$ and thus the obtained policies always satisfy the upper bound of Corollary 4.7. By Corollary 5.12 this would imply that these algorithms yield optimal policies in case that the service capacities $a_1, a_2, \dots, a_n$ are linearly independent over $\mathbb{Z}$. Often the GR algorithm yields the worse policy of these algorithms. An explanation is that if the GR algorithm is applied then $k_t(\psi)$ only depends on $N_1^t(\psi)$, $N_2^t(\psi)$, $\dots$, $N_n^t(\psi)$, but not on the order of the routing of the foregoing jobs. So, the GR algorithm uses less information in choosing the server to which the arriving job is routed than the other algorithms. However, the GR algorithm performances rather good if a_1 is very large compared to the other fractions $a_2, a_3, \dots, a_n$. The reason for this is that the routing to server 1 is regular and thus optimal and most of the arriving jobs are routed to server 1 in this case. So, the routing to the other servers has not much influence on W and thus the GR algorithm yields a rather good policy in such cases. We note that on the contrary the SWT algorithm yields inferior policies if a_1 is very large compared to the other fractions (see for example the $(10000, 3, 3, 1, 1)$ system), but rather good results if the a_i's have similar size (sometimes even the best result). This can be explained by the fact that for systems with large a_1 in the beginning the SWT algorithm routes too many arriving jobs to other servers than the fast server 1. Therefore the splitting sequence of server 1, $\underline{\delta}(1)$ becomes rather irregular if the SWT algorithm is applied to such systems. It seems even that in general it holds that $W(SWT) \uparrow \frac{n-1}{2}$ if $a_1 \uparrow 1$. We think that the OSSM algorithms on average have the best performance of the considered algorithms. There is in most cases not much difference in the performance of the

OSSM1 and OSSM2 algorithm. However, for some systems (see the $(1000,3,2,1)$ system and the same difference occurs for all the $(x,3,2,1)$ systems with x large) a remarkable difference in performance occurs.
We notice that applying the SG algorithm iteratively to a solution obtained by one of these algorithm often gives a small improvement. Sometimes the improvement is rather large like in the $(33,33,33,33,33,26,26,26,26)$ system for the OSSM algorithms and especially for the SWT algorithm. Moreover it occurs regularly that from a nearly optimal policy an optimal policy can be obtained by using the SG algorithm in this way.

We now discuss an argument for the fact that in general these greedy algorithms and similar greedy algorithms do not yield optimal policies. For this we consider the $(4,4,4,3,3,3,3)$ system with the corresponding balanceable set of fractions $(\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8})$. In tables 2 and 3 we see that for this system none of the considered algorithms yields an optimal policy. The reason for this is that a policy obtained by applying such an algorithm always starts with $1,2,3$, i.e the first three arriving jobs are routed to the three fast servers. In fact it is clear that policies obtained by applying an OSSM algorithm or SWT algorithm have the SFQ (Shorter Faster Queue) property, i.e an arriving job is never routed to some server i if there exist a faster server j (thus $a_i < a_j$) for which the remaining workload at that moment is smaller or equal than for server i. However, optimal policies do not always have this property. For example for this system a policy is optimal if and only if it corresponds to a regular word on the alphabet $\{1,2,3,4,5,6,7\}$ with the above densities for the letters. In [10] the regular words with these densities are classified and it follows that there exist no regular words with these densities which have $1,2,3$ as subword. Hence policies starting with $1,2,3$ and policies that have the SFQ property are not optimal for this system. The most obvious regular word and corresponding optimal policy for these fractions is $(1,4,2,5,3,6,1,7,2,4,3,5,1,6,2,7,3,4,1,5,2,6,3,7)^\infty$, i.e the policy that alternately routes the arriving jobs to the next fast and the next slow server. So, if we put an one for a fast server and a zero for a slow server then this policy corresponds to the regular sequence $(10)^\infty$ which has density $\frac{1}{2} = 3 \cdot \frac{1}{6}$ which is the fraction of jobs that has to be routed to one of the fast servers. This construction to obtain regular words can be used in general for systems with only two distinct fractions. It is clear that corresponding optimal policies have in general not the SFQ property. Also for systems with fractions that are not balanceable the optimal policies do not always have the SFQ property.
For systems in which some of the fractions are equal to each other it seems a good idea to first apply the several algorithms to the altered system in which these fractions are taken together. From the best obtained policy for the altered system a good policy for the original system can be obtained in the same way we just obtained a regular policy for the $(4,4,4,3,3,3,3)$ system. For example for the $(10000,3,3,1,1)$ system this means that first a good (optimal) policy ψ for the $(5000,3,1)$ system is obtained and then from policy ψ follows a policy ψ' for the original $(10000,3,3,1,1)$ system.

We conclude that the considered algorithms and similar greedy algorithms yield in general policies that are not optimal. However, by applying the several algorithms and possibly using the SG algorithms iteratively we can obtain good policies quickly by taking the best one obtained. It seems that from the algorithms we considered the OSSM algorithms give in most cases the best result.

Acknowledgement. The author thanks A. Hordijk and R. Tijdeman for their helpful com-

ments.

References

[1] E. Altman, B. Gaujal and A. Hordijk. Admission control in stochastic event graphs, Technical Report 3179, INRIA, 1997. To appear in *IEEE Trans. Automat. Control.*

[2] E. Altman, B. Gaujal and A. Hordijk. Balanced sequences and Optimal Routing, Technical Report 3180, INRIA, 1997. To appear in *J. ACM.*

[3] E. Altman, B. Gaujal and A. Hordijk. Multimodularity, convexity and optimization properties, *Math. Oper. Res.* **25** (2000), 324-347.

[4] P. Arnoux, C. Mauduit, I.Shiokawa and J. Tamura. Complexity of sequences defined by billiards in the cube, *Bull. Soc. Math. France* **122** (1994), 1-12.

[5] Y. Arian and Y. Levy, Algorithms for generalized round Robin routing, *Oper. Res. Lett.* **12** (1992), 313-319.

[6] M.B. Combé and O.J. Boxma, Optimization of static traffic allocation policies, *Theoret. Comput. Sci.* **125** (1994), 17-43.

[7] R. Graham, Covering the positive integers by disjoint sets of the form $\{[n\alpha + \beta] : n = 1, 2, \ldots\}$, *J. Combin. Theory Ser. A* **15** (1973), 354-358.

[8] B. Hajek, Extremal splittings of point processes, *Math. Oper. Res.* **10** (1985), 543-556.

[9] A. Hordijk, G.M. Koole, J.A. Loeve, Analysis of a customer assignment model with no state information, *Probab. Engrg. Inform. Sci.* **8** (1994), 419-429.

[10] R. Morikawa, On eventually covering families generated by the bracket function i-v, *Bull. Fac. Liberal Arts Nagasaki Univ.* **23** no. 1 (1982), 17-22, **24** no. 1 (1983), 1-9, **25** no. 1 (1984), 1-11, **25** no. 2 (1985), 1-8, **26** no. 1 (1985), 15-17, **36** no. 1 (1995), 1-17.

[11] R. Morikawa, Disjoint sequences generated by the bracket function i-vi, Number theory and combinatorics. Japan, 1984. World Sci. Publishing, Singapore (1985), 305-321 and *Bull. Fac. Liberal Arts Nagasaki Univ.* **26** no. 1 (1985), 1-13, **28** no. 2 (1988), 1-24, **30** no. 1 (1989), 1-10, **32** no. 2 (1992), 181-185, **34** no. 1 (1993), 1-23.

[12] M. Morse and G.A. Hedlund, Symbolic dynamics, *Amer. J. Math.* **60** (1938), 815-866.

[13] M. Morse and G.A. Hedlund, Symbolic dynamics II - Sturmian trajectories, *Amer. J. Math.* **62** (1940), 1-42.

[14] J. Simpson, Disjoint covering systems of rational Beatty sequences, *Discrete Math.* **92** (1991), 361-369.

[15] Y.G. Sinai, *Introduction to Ergodic Theory*, Princeton University Press, Princeton, N.J. (1976).

[16] S. Stidham, Jr. , A last Word on $L = \lambda W$, *Oper. Res.* **22** (1974), 417-421.

[17] R. Tijdeman, On comlementary triples of Sturmian bisequences, *Indag. Math. (N.S)* **7** (1996), 419-424.

[18] R. Tijdeman, Fraenkel's conjecture for six sequences, Report W98-24, Leiden University, The Netherlands, 1998. To appear in *Discrete Math.*

[19] R. Tijdeman, Exact covers of balanced sequences and Fraenkel's conjecture, *Algebraic Number Theory and Diophantine Analysis*, Edited by F. Halter-Koch and R.F. Tichy, Walter de Gruyter, Berlin (2000), 467-483.

[20] T.P. Yum, The design and analysis of a semi-dynamic deterministic routing rule, *IEEE Trans. Comm.* **29** (1981), 498-504.

D.A van der Laan: Mathematical Institute, Leiden University, P.O. Box 9512, 2300 RA Leiden, The Netherlands; email: dvdlaan@math.leidenuniv.nl