Optimal steady-state and transient trajectories of multi-queue switching servers with a fixed service order of queues

Optimal steady-state and transient trajectories of multi-queue switching servers with a fixed service order of queues

D.A.J. van Zwieten[a,*], E. Lefeber[a,], I.J.B.F. Adan[b,a]

[a] *Department of Mechanical Engineering, Eindhoven University of Technology, PO Box 513, 5600MB Eindhoven*
[b] *Department of Industrial Engineering and Innovation Sciences, Eindhoven University of Technology, PO Box 513, 5600MB Eindhoven*

Abstract

The optimal scheduling problem of a system with two fluid queues attended by a switching server is addressed from two angles, the optimal steady-state and the optimal transient problem. The considered system includes features, such as setup times, setup costs, backlog and constraints on queue contents, cycle times and service times. First, the steady-state problem is formulated as a quadratic problem (QP), given a fixed cycle time. Evaluation of the QP problem over a range of cycle times results in the optimal steady-state trajectory, minimizing the total cycle costs or time average costs. Second, given initial conditions, we derive the optimal transient trajectory that leads to the optimal steady-state trajectory in a finite amount of time at minimal costs. For systems with backlog, we introduce additional costs on the number of cycles required to reach the steady-state trajectory in order to simplify the transient trajectory. The transient switching behavior and optimal initial modes are also addressed. Furthermore, we show by means of an example that the method can be extended to multi-queue switching servers.

Keywords: Switching server, Optimization, Quadratic programming, Steady-state trajectory, Transient trajectory

1. Introduction

Optimal scheduling of systems with switching behavior is a problem of great importance. This problem, for even the most simple system, i.e., a server attending two queues, has been investigated by many researchers, see for example [1, 4, 5, 6, 7, 8, 10, 11, 14], and references therein. We follow the general

*Principal corresponding author
**Corresponding author
Email addresses: dirkvzwieten@gmail.com (D.A.J. van Zwieten), A.A.J.Lefeber@tue.nl (E. Lefeber), I.J.B.F.Adan@tue.nl (I.J.B.F. Adan)

framework introduced by [9] and model the production flow as continuous rather than discrete. The system considered in this paper is a server that can attend two fluid queues, where only a single queue can be attended at a time. Switching service to another queue might require a setup process, which might take time or involves switching costs. These systems arise in numerous contexts, such as manufacturing systems, signalized traffic intersections, computer communication networks and hospital rooms. Usage of optimal schedules can reduce time or costs. For instance, optimal schedules can reduce costs for manufacturing systems via lowering the required storage capacity and shortening lead times, or, for traffic signals at signalized intersections, optimal schedules can reduce congestion and thereby improve mobility and reduce the amount of environmentally harmful emissions.

In the aforementioned literature, the two queue switching servers are restricted in the sense that either setup times, setup costs, backlog or limited queue contents are required, omitted or not allowed. Also, most studies assume the simplifying condition that the system is symmetric, see [1, 4, 8, 11]. In the current work, a two queue switching server is considered without restrictions on any parameters and with the flexibility of allowing setup times, setup costs and backlog, as well as constraints on cycle time, service time and queue contents.

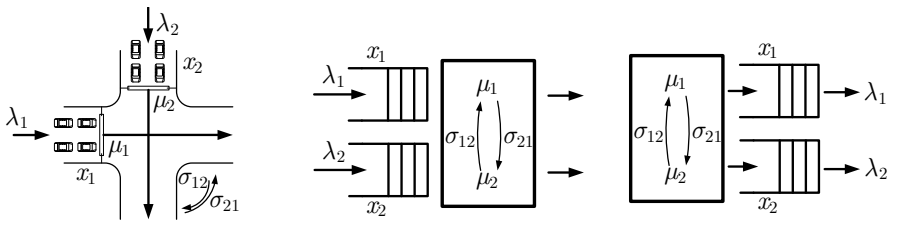
In this paper, we divide the optimal scheduling problem into two subproblems: the derivation of optimal steady-state trajectories and the derivation of optimal transient trajectories. The current study is an extension of the work in [17]. Similar to [15, 17], we formulate both subproblems as Quadratic Programming (QP) problems, with the addition of backlog and setup costs. Once the optimal steady-state trajectory is known, we study the best way of reaching it from any initial state, i.e., with minimal costs. This is a transient optimization problem, occurring for instance in case of a machine which is failure prone, or in case of a traffic intersection which gives priority to busses [16]. In these cases, we assume that deviations from the steady-state trajectory rarely occur, allowing the system to recover to the steady-state situation after each interruption. For systems without backlog and without capacity constraints, the policy for optimal transient behavior is presented. For systems with backlog, we introduce additional costs on the number of cycles required to reach the steady-state trajectory in order to simplify the transient trajectory.

Furthermore, we show by means of an illustration that the proposed methods can be extended to multi-queue switching servers, given the order of service of queues.

The remainder of this paper is organized as follows. Section 2 describes the system and presents the constraints. The optimal steady-state problem is addressed in Section 3 and examples of optimal trajectories are presented. In Section 4, the optimal transient problem is addressed. An illustration of optimal trajectories for a multi-queue switching server is presented in Section 5. Conclusions are provided in Section 6.

2. System description

We consider a system of two queues served by a single switching server. Fluid arrives at each queue $i = 1, 2$ with arrival rate λ_i. The content of queue i at time t is denoted by $x_i(t)$. The server is limited to serve only one queue at a time. If the server serves queue i, the service rate is given by $r_i \in [0, \mu_i]$. Three examples of the system under consideration are presented in Figure 1, a signalized traffic intersection with two flows in Figure 1a, a 2-queue switching server in Figure 1b and a 2-product manufacturing system in Figure 1c. The latter system has constant demands λ_i instead of constant arrivals.



(a) Signalized intersection with 2 flows.

(b) Switching server with 2 queues.

(c) Manufacturing system with 2 jobs.

Figure 1: Different two queue switching server layouts.

Typically, switching service between different queues implies a setup process, either a *setup time* $\sigma_{i,j} \geq 0$ for switching from queue i to queue j, *setup costs* $s_{i,j}$ or a combination of these. For instance, a setup time can be reserved for vehicles to leave the intersection after the queue has received a red light (end of service), thereby preventing collisions, or for a machine to adjust configurations or to perform cleaning. In the latter case, also switching costs might be involved. A *cycle* consists of the setup and service of both queues. The *total setup time* in a cycle is denoted by $\sigma = \sigma_{1,2} + \sigma_{2,1}$ and the *total setup costs* in a cycle by $s = s_{1,2} + s_{2,1}$.

Given the setup times and cyclic behavior, we assume that the system can operate in four modes, denoted by $m \in \{1, 2, 3, 4\}$. Without loss of generality, the first mode, $m = 1$, indicates a setup to serve queue 1 and possible idling of the server, $m = 2$ indicates serving queue 1, $m = 3$ indicates a setup to serve queue 2 and possible idling of the server and $m = 4$ indicates serving queue 2. Note that for a system without setup times, i.e., $\sigma = 0$, modes 1 and 3 can have a duration of zero time units. The state x of the system not only consist of queue levels x_1 and x_2, but also of the *remaining idle time* x_0 (including setup times) and mode m, i.e., $x(t) = \big[x_0(t), \quad x_1(t), \quad x_2(t), \quad m(t)\big]$.

A service time is defined as the uninterrupted interval during which the queue is served. The duration of a service time for queue n is nonnegative and is denoted by τ_n. Once the server is allocated to serve queue n, the server requires an *idle period* τ_n^0, which consists of the setup time and a possible idle time, i.e.,

$$\sigma_{j,n} \leq \tau_n^0, \qquad n, j = 1, 2, \quad n \neq j. \tag{1}$$

In this paper, we distinguish between a system with and a system without backlog. In case of backlog, denoting an accumulation over time of work waiting to be done or orders to be fulfilled, the contents of queue n can be negative and are therefore divided into an *inventory level* $x_n^+(t) = \max(x_n(t), 0)$ and a *backlog level* $x_n^-(t) = \min(x_n(t), 0)$. Hence, $x_n(t) = x_n^+(t) + x_n^-(t)$ and at each time instance either the backlog or inventory level is zero, i.e., $x_n^+(t)x_n^-(t) = 0$, $\forall t$. For a system without backlog, $x_n(t) = x_n^+(t)$, the indicator $^+$ is often omitted. Also, for this system it is shown in [13] that for optimal policies, a server, once serving a queue does not idle and serves at *maximal* rate. Hence, the service time allocated at queue n is divided into two parts,

$$\tau_n = \tau_n^\mu + \tau_n^\lambda.$$

where the duration of serving at maximal rate is indicated by τ_n^μ and the duration of serving at arrival rate by τ_n^λ. Note that serving at arrival rate, i.e., $\tau_n^\lambda > 0$, occurs only if the queue is empty. This duration is referred to as *slow-mode*, since capacity is wasted. However, as indicated in [14] and shown in Section 3, using a slow-mode might result in optimal trajectories, since it enlarges the cycle time and thereby reduces the fraction of time spent on setups, which also wastes capacity. For systems with backlog, we assume that the server, once serving a queue, can serve the queue at the maximal rate and subsequently can serve the queue at the arrival rate.

The *cycle time* T is the time it takes to serve both queues in a cycle. The cycle time consists of idle and service periods for each queue, i.e.,

$$T = \tau_1^0 + \tau_1^\mu + \tau_1^\lambda + \tau_2^0 + \tau_2^\mu + \tau_2^\lambda.$$

The workload of queue n is defined by $\rho_n = \frac{\lambda_n}{\mu_n}$. An important notion for steady-state trajectories is *stability*. A system is called *stable* if all queue contents remain bounded. For switching servers, cf. [2, 3, 12], this definition is commonly used. To achieve a stable system, the system capacity should be able to meet the inflow, i.e., is it possible to process all incoming fluid? For the considered two queue switching servers, all incoming fluid can be processed if $\rho_1 + \rho_2 \leq 1$. Note that the total workload for systems with setup times should be strictly less than 1, i.e., if the total workload equals 1 the server lacks capacity to serve the fluid that has arrived during setups. In a stable system the service periods must satisfy

$$\lambda_n T = \mu_n \tau_n^\mu + \lambda_n \tau_n^\lambda, \qquad n = 1, 2. \tag{2}$$

Condition (2) also ensures that the queue contents at the start of the cycle are identical to the queue contents at the end of the cycle, and therefore ensures *steady-state* behavior. If (2) is not satisfied, the system behavior is *transient*. Note that we can impose additional constraints regarding service periods and queue contents, depending on the system under consideration. These constraints can originate from, e.g., operational or safety issues. We remark that these constraints are not mandatory, but can be included if required. Some of them are discussed below.

Cycle time constraints can originate from, for instance, limiting the cycle time of a manufacturing system to the operator's available time or requiring a minimal cycle time for safety reasons in traffic intersections. Therefore, minimal and maximal cycle times, respectively $T^{\min}$ and $T^{\max}$, can be taken into account,

$$T^{\min} \leq T \leq T^{\max}. \tag{3a}$$

Furthermore, bounds on service times, denoted by $\tau_n^{\min}$ and $\tau_n^{\max}$, can be required, e.g., minimal and maximal service (green) times for traffic intersections. The service time constraints are imposed via

$$\tau_n^{\min} \leq \tau_n \leq \tau_n^{\max} \qquad n = 1, 2. \tag{3b}$$

In addition to the constraints on cycle and service periods, the queue lengths can be bounded, e.g., finite queue capacity, and also a minimal queue level (or maximal backlog level) can be desired, i.e.,

$$x_n^{\min} \leq x_n(t) \leq x_n^{\max}, \qquad n = 1, 2. \tag{3c}$$

Note that we can impose additional constraints, regarding service times and/or queue contents. Given the system description and the constraints, we present a method to derive the optimal steady-state trajectory in Section 3. This trajectory is used in Section 4 to derive the optimal transient trajectory.

3. Optimal steady-state trajectory

Multiple performance criteria exists for evaluating the trajectory. For the system under consideration, cycle time, flow-time, total costs or average costs are commonly used criteria. In this paper, we focus on minimizing the cycle time or the costs. However, other criteria can be easily incorporated.

3.1. Cycle time

The *minimal cycle time* T^* required to serve all arrivals during a cycle can be easily derived. Idling of the server, or wasting capacity due to service at arrival rate both elongate the cycle time and are therefore not optimal, unless required to satisfy any constraints. For the unconstrained system, i.e., without constraints (3), the minimal cycle time follows from $T = \rho_1 T + \rho_2 T + \sigma$. Required lower bounds on service periods, given by (3b) or by (3c) via

$$\tau_n \geq T - \frac{x_n^{\max} - x_n^{\min}}{\lambda_n}, \qquad n = 1, 2,$$

also affect the minimal cycle time. In the remainder of this paper, we assume that $\tau_n^{\min}$ is such that

$$\tau_n^{\min} \geq T - \frac{x_n^{\max} - x_n^{\min}}{\lambda_n}, \qquad n = 1, 2.$$

From the relations $T = \tau_1+\tau_2+\sigma$, $\tau_1 = \max(\rho_1 T, \tau_1^{\min})$, and $\tau_2 = \max(\rho_2 T, \tau_2^{\min})$, we obtain (by considering all four cases):

$$T^* = \max\left(\tau_1^{\min} + \tau_2^{\min} + \sigma, \frac{\sigma}{1-\rho_1-\rho_2}, \frac{\sigma+\tau_1^{\min}}{1-\rho_2}, \frac{\sigma+\tau_2^{\min}}{1-\rho_1}\right).$$

If we also consider the lower bound on the cycle time (3a), where without loss of generality we can assume $T^{\min} \geq \tau_1^{\min} + \tau_2^{\min} + \sigma$, the minimal cycle time follows from

$$T^* = \max\left(T^{\min}, \frac{\sigma}{1-\rho_1-\rho_2}, \frac{\sigma+\tau_1^{\min}}{1-\rho_2}, \frac{\sigma+\tau_2^{\min}}{1-\rho_1}\right). \tag{4}$$

Then, $\tau_1 = \rho_1 T^*$ and $\tau_2 = T^* - \sigma - \tau_1$ are the, not necessarily unique (if $T^* = T^{\min}$), service periods from a steady-state trajectory with minimal cycle time.

3.2. Total costs

Total costs during a cycle is another performance criterion. This criterion is used, for instance, in [6]. Costs can arise from switching service between queues, i.e., setup costs s. Also, costs can be related to the queue contents. We consider inventory costs c_n^+, which are proportional with $x_n^+(t)$, and backlog costs c_n^-, proportional with $x_n^-(t)$, which for instance arise when production is behind on the demand for the system depicted in Figure 1c. This results in the following total costs J_c for the steady-state trajectory

$$J_c = \int_0^T \left[c_1^+ x_1^+(t) + c_1^- x_1^-(t) + c_2^+ x_2^+(t) + c_2^- x_2^-(t)\right] dt + s. \tag{5}$$

The trajectory minimizing J_c is the optimal steady-state trajectory. The optimal trajectory is a trade-off between loss of capacity due to setups, slow-modes and the average setup costs. Elongating the cycle time, by including a slow-mode or creating backlog, results in less switches over time where capacity is lost due to setups and thereby lowers the average setup costs. It can be seen in (5) that the setup costs s do not influence the optimal steady-state trajectory. However, these costs do play a role for the time average costs as performance indicator.

The total inventory and backlog of a queue during a cycle can be derived regarding the service periods, due to the fluid flows and cyclic behavior. Figure 2 presents the contents of queue n during a single cycle. All idle and service periods are indicated, together with the slope rates.

The minimal content of queue n in a steady-state trajectory is denoted by $\underline{x}_n$. For the optimal steady-state trajectory, this value is bounded, as presented in the following lemma.

Lemma 3.1. *For the optimal trajectory it holds that*

$$\max(x_n^{\min}, (\lambda_n - \mu_n)\tau_n^\mu) \leq \underline{x}_n \leq \min(x_n^{\max} + (\lambda_n - \mu_n)\tau_n^\mu, 0), \quad n = 1, 2,$$

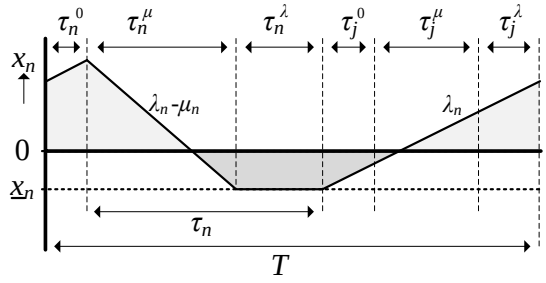
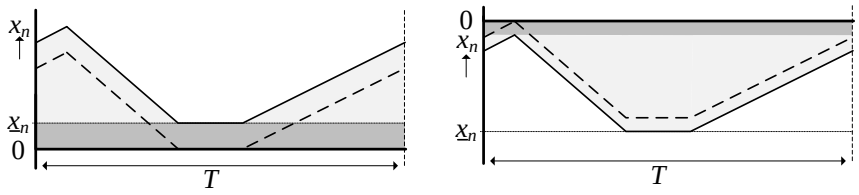


Figure 2: Evolution of x_n during a cycle, including setup and service periods and rates of increase/decrease.

Proof. The proof is twofold. First, consider a trajectory where $\underline{x}_n > 0$ and $x_n^{\min} < 0$, depicted by the solid line in Figure 3a. Then, an alternative trajectory exists with identical service periods and $\underline{x}_n = 0$, depicted by the dashed line. The alternative trajectory has lower costs, i.e., it has $\underline{x}_n T$ less total inventory, presented by the gray area. If $x_n^{\min} > 0$, for the same reasoning, $\underline{x}_n = x_n^{\min}$ is optimal. Second, consider a trajectory where $\underline{x}_n < (\lambda_n - \mu_n)\tau_n^\mu$, i.e., the maximal queue content is less than zero ($x_n(t) < 0$ for all $t \in [0, T)$). This trajectory is presented by the solid line in Figure 3b. Then, an alternative trajectory exists with identical service periods and $\underline{x}_n = (\lambda_n - \mu_n)\tau_n^\mu$, depicted by the dashed line. The alternative trajectory has lower costs, i.e., it has $(\underline{x}_n + (\mu_n - \lambda_n)\tau_n^\mu)T$ less total backlog, presented by the gray area. If $x_n^{\max} < 0$, for the same reasoning, $\underline{x}_n = x_n^{\max} + (\lambda_n - \mu_n)\tau_n^\mu$ is optimal. □



(a) Trajectory with $\underline{x}_n > 0$.

(b) $\underline{x}_n < -\lambda_n(\tau_j^0 + \tau_j^\mu + \tau_j^\lambda + \tau_n^0)$.

Figure 3: Graphical representation of Lemma 3.1.

Based on Lemma 3.1, the inventory and backlog can be easily derived for a steady-state trajectory. Total inventory of queue n during a cycle is denoted by w_n^+ and total backlog by w_n^-. These values depend on the queue content constraints $x_n^{\min}$ and $x_n^{\max}$, $n = 1, 2$.

If $x_n^{\min} \leq 0 \wedge x_n^{\max} \geq 0$, the total inventory and backlog are given by

$$w_n^+ = \int_0^T x_n^+(\tau)d\tau = \frac{1}{2}\frac{\lambda_n\mu_n}{\mu_n - \lambda_n}(T-\tau_n)^2 + \underline{x}_n T + w_n^-, \tag{6a}$$

$$w_n^- = \int_0^T x_n^-(\tau)d\tau = \frac{\underline{x}_n^2}{2}\left(\frac{1}{\mu_n - \lambda_n} + \frac{1}{\lambda_n}\right) + \tau_n^\lambda \underline{x}_n, \tag{6b}$$

where (6a) can be readily obtained from Figure 2 as the area above $\underline{x}_n$ from which we subtract $-\underline{x}_n T - w_n^-$.

If $x_n^{\min} > 0$, and therefore also $x_n^{\max} > 0$, it holds that

$$\begin{aligned} w_n^+ &= \int_0^T x_n^+(\tau)d\tau = \frac{1}{2}\frac{\lambda_n\mu_n}{\mu_n - \lambda_n}(T-\tau_n)^2 + \underline{x}_n T, \\ w_n^- &= 0, \end{aligned}$$

and for $x_n^{\max} < 0$, and therefore $x_n^{\min} < 0$, the contents are given by

$$\begin{aligned} w_n^+ &= 0, \\ w_n^- &= \int_0^T x_n^-(\tau)d\tau = \frac{1}{2}\frac{\lambda_n\mu_n}{\mu_n - \lambda_n}(T-\tau_n)^2 + \lambda_n\tau_n^\lambda(T-\tau_n) - (\underline{x}_n + (\mu_n - \lambda_n)\tau_n^\mu)T. \end{aligned}$$

It can be seen that the expressions for w_n^+ and w_n^- in (6) are quadratic in the optimization variables τ_n and $\underline{x}_n$. Hence, the optimization problem is formulated as a Quadratic Programming (QP) problem, given by

$$\begin{aligned} J_c^* = s + \min_{\tau_n^0, \tau_n^\mu, \tau_n^\lambda, \underline{x}_n} & \sum_{n=1}^{2} \left(c_n^+ w_n^+ + c_n^- w_n^-\right), \\ s.t. \quad \tau_n^{\min} \leq \tau_n \leq \tau_n^{\max}, & \qquad n = 1,2, \qquad (7a) \\ \underline{x}_n \leq x_n^{\max} - (\mu_n - \lambda_n)\tau_n^\mu, & \qquad n = 1,2, \qquad (7b) \\ \lambda_n T = \mu_n \tau_n^\mu + \lambda_n \tau_n^\lambda, & \qquad n = 1,2, \qquad (7c) \\ T = \tau_1^0 + \tau_1^\mu + \tau_1^\lambda + \tau_2^0 + \tau_2^\mu + \tau_2^\lambda, & \qquad (7d) \end{aligned}$$

where (7b) follows from (3c), i.e., the maximal queue content is given by $\underline{x}_n + (\mu_n - \lambda_n)\tau_n^\mu$, as can be seen in Figure 2. Note that the objective is quadratic, as the total inventory and backlog levels (6) are a quadratic combination of the optimization variables.

For the system without backlog, the minimal total costs J_c^* can be analytically derived, which is presented in Section 3.3. For the system with backlog and for systems with multiple queues or networks of switching servers, the analytical derivation is, if possible, much more complex.

3.3. Time average costs

The time average costs are commonly used as performance indicator, also referred to as time average weighted work in process for manufacturing systems

or time average weighted queue lengths. In [1, 4, 8, 11, 14], this performance indicator is also considered. The time average costs, given a fixed cycle time T, are given by

$$J_w = \frac{1}{T} J_c. \tag{8}$$

For the system without backlog, the minimal time average costs, as well as the total costs, can be analytically derived. First, consider the unconstrained system. We assume, without loss of generality, that $c_1\lambda_1 \geq c_2\lambda_2$. Recall that, under optimal policies the server, once attending a queue, does not idle, i.e., $\tau_n^0 = \sigma_{j,n}$. The time average costs, see (6) for $x_n^{\min} \geq 0$, are given by

$$\begin{aligned} J_w(T) = \frac{1}{T}\big[a_1(T - \rho_1 T - \gamma([1-\rho_1-\rho_2]T - \sigma))^2 + \\ + a_2(T - \rho_2 T - (1-\gamma)([1-\rho_1-\rho_2]T - \sigma))^2 + s\big] + c_1 x_1^{\min} + c_2 x_2^{\min}, \end{aligned} \tag{9a}$$

$$a_n = \frac{c_n}{2}\frac{\lambda_n \mu_n}{\mu_n - \lambda_n}, \qquad n = 1, 2.$$

The minimal required service time is given by $(\rho_1 + \rho_2)T$. The remainder of the service time, i.e., $([1-\rho_1-\rho_2]T - \sigma)$, is divided in (9a) over both queues using $\gamma \in [0, 1]$, which denotes the fraction of remaining service allocated at queue 1. The optimum of (9a) for $T > T^*$ with respect to γ is given by

$$\gamma^* = \min\left(\frac{[c_1\lambda_1(1-\rho_2) - c_2\lambda_2\rho_1]T - c_2\lambda_2\sigma}{\left[c_1\lambda_1\frac{1-\rho_2}{1-\rho_1} + c_2\lambda_2\right](1-\rho_1-\rho_2)T - c_1\lambda_1\sigma\frac{1-\rho_2}{1-\rho_1} - c_2\lambda_2\sigma}, 1\right). \tag{9b}$$

Note that $\gamma^* > 0$ for $T > T^*$, as the denominator of (9b) is strictly positive if $T > T^*$ and the nominator is zero if both $T = T^*$ and $c_1\lambda = c_2\lambda$ and strictly positive otherwise. From (9b), we find that all additional service time is allocated at serving the first queue, i.e., $\gamma^* = 1$, if

$$T \leq \frac{c_1\lambda_1\sigma}{c_2\lambda_2(1-\rho_1) - c_1\lambda_1\rho_2} \qquad \vee \qquad c_1\lambda_1\rho_2 \geq c_2\lambda_2(1-\rho_1). \tag{9c}$$

Hence, for $c_1\lambda_1\rho_2 \geq c_2\lambda_2(1-\rho_1)$, a slow mode at queue 2 is never optimal. We first present the optimization problem if (9c) is satisfied. Given (9c), i.e., $\gamma^* = 1$ (all additional service allocated at queue 1), the time average costs are given by

$$J_w(T) = (a_1\rho_2^2 + a_2(1-\rho_2)^2)T + 2a_1\sigma\rho_2 + c_1 x_1^{\min} + c_2 x_2^{\min} + \frac{a_1\sigma^2 + s}{T}, \tag{9d}$$

and the optimal cycle time T^{opt} yields

$$T^{\text{opt}} = \frac{\sqrt{[a_1\rho_2^2 + a_2(1-\rho_2)^2]\,(s + a_1\sigma^2)}}{a_1\rho_2^2 + a_2(1-\rho_2)^2},$$

and $J_w^* = J_w(T^{\text{opt}})$. Next, the effect of the constraints on this system are regarded. By adding bounds on the cycle time (3a), the feasible area changes, i.e., $T \in [T^*, T^{\max}]$. Capacity constraints, i.e., bounds on service times and bounds on queue lengths, both limit the service time duration, as discussed in Section 3.1. For the system satisfying (9c) together with minimal service period constraints, the time average costs are given by (9d) if $\tau_2^{\min} \leq \rho_2 T^*$, i.e., $\tau_2 \geq \tau_2^{\min}$. Otherwise

$$J_w(T) = \frac{a_1(\sigma + \tau_2^{\min})^2 + a_2(T - \tau_2^{\min})^2 + s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}, \qquad \text{if } T < \frac{\tau_2^{\min}}{\rho_2}, \tag{9e}$$

which exceeds the time average costs given by (9d), as additional service time is allocated at queue 2. Note that the lower bound on the service period of queue 1 can only affect the minimal cycle time, as all extra service time is allocated at queue 1. Moreover, an upper bound on the service duration bounds the cycle time via

$$T \leq \frac{\tau_n^{\max}}{\rho_n}, \qquad n = 1, 2. \tag{9f}$$

If both (9c) and (9f) are satisfied, the maximal service time constraints only affect J_w if $\tau_1^{\max} \leq T - \sigma - \rho_2 T$, as for these cycle times the duration of the slow mode in queue 1 is limited. Then, the time averaged costs are given by

$$\frac{a_1(T - \tau_1^{\max})^2 + a_2(\sigma + \tau_1^{\max})^2 + s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}, \qquad \text{if } T > \frac{\tau_1^{\max} + \sigma}{1 - \rho_2}, \tag{9g}$$

which also exceeds the time averaged costs for the unconstrained system for similar reasoning. Then, the optimal time averaged costs J_w^* for the system satisfying (9c) is the minimum of the following optimization problems

$$\begin{aligned}
&\text{minimum of (9e) for} && \max\left(T^*, T^{\min}\right) \leq T < && \frac{\tau_2^{\min}}{\rho_2}, \\
&\text{minimum of (9d) for} && \max\left(T^*, T^{\min}, \frac{\tau_2^{\min}}{\rho_2}\right) \leq T \leq && \min\left(T^{\max}, \frac{\tau_1^{\max} - \sigma}{1 - \rho_2}\right), \\
&\text{minimum of (9g) for} && \frac{\tau_1^{\max} - \sigma}{1 - \rho_2} < T \leq && T^{\max},
\end{aligned}$$

which can be easily derived. As an example, consider a heterogenous (non-symmetric) system without backlog, with parameters

$$\begin{aligned}
&\lambda_1 = 2, \quad \mu_1 = 8, \quad \sigma_{2,1} = 3, \quad c_1^+ = 8, \\
&\lambda_2 = 1, \quad \mu_2 = 4, \quad \sigma_{1,2} = 7, \quad c_2^+ = 1.
\end{aligned} \tag{10}$$

A graphical representation of $J_w(T)$ for the system with parameters (10) (satisfying (9c)) is presented in Figure 4. The solid line presents the time average

costs for the unconstrained system. The costs for the system with $\tau_2^{\min} = 10$ is depicted by the dashed line and the costs for the system with $\tau_1^{\max} = 40$ is depicted by the dotted line. Note that the optimum for the unconstrained system and system with the maximal service period constraint on queue 1 is located at $T^{\text{opt}} = 32$. If the minimal service period for queue 2 is required, the optimum is located at $T = 40$.

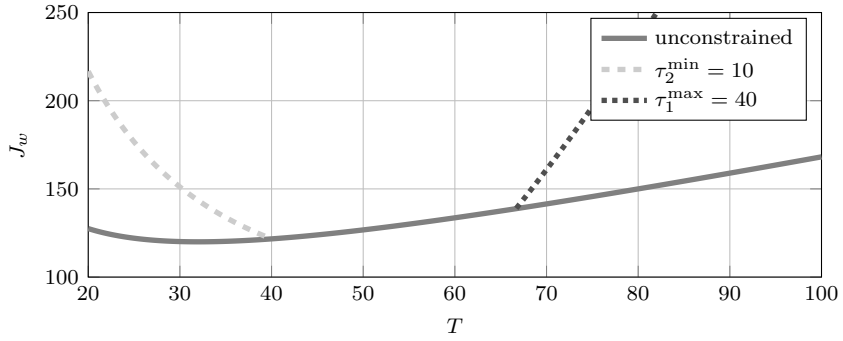


Figure 4: Time average costs for the system with parameters (10).

If the additional service time is allocated at service of both queues, i.e., (9c) does not hold, the time average costs are given by substituting (9b) into (9a), resulting in

$$J_w(T) = \frac{c_1\lambda_1 c_2\lambda_2(T+\sigma)^2}{2[c_1\lambda_1(1-\rho_2) + c_2\lambda_2(1-\rho_1)]T} + \frac{s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}.$$

Furthermore, for the system with minimal and maximal service time constraints, the costs $J_w(T)$ are given by

$$\begin{aligned}
&\frac{a_1(T-\tau_1^{\max})^2 + a_2(\sigma+\tau_1^{\max})^2 + s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}, \quad \text{if } T > \frac{\tau_1^{\max} + \gamma^*\sigma}{\rho_1 + \gamma^*(1-\rho_1-\rho_2)},\\
&\frac{a_1(\sigma+\tau_2^{\max})^2 + a_2(T-\tau_2^{\max})^2 + s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}, \quad \text{if } T > \frac{\tau_2^{\max} + (1-\gamma^*)\sigma}{\rho_2 + (1-\gamma^*)(1-\rho_1-\rho_2)},\\
&\frac{a_1(T-\tau_1^{\min})^2 + a_2(\sigma+\tau_1^{\min})^2 + s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}, \quad \text{if } T < \frac{\tau_1^{\min} + \gamma^*\sigma}{\rho_1 + \gamma^*(1-\rho_1-\rho_2)},\\
&\frac{a_1(\sigma+\tau_2^{\min})^2 + a_2(T-\tau_2^{\min})^2 + s}{T} + c_1 x_1^{\min} + c_2 x_2^{\min}, \quad \text{if } T < \frac{\tau_2^{\min} + (1-\gamma^*)\sigma}{\rho_2 + (1-\gamma^*)(1-\rho_1-\rho_2)},
\end{aligned}$$

Then, along the same lines as presented above, the optimum is given by the minimum of all optimization results within the feasible cycle time range. If the objective is to optimize the total costs in a cycle, a similar approach can be used where $J_c = TJ_w$.

To derive the optimal steady-state trajectory using QP, e.g., for the system with backlog or for systems with multiple queues, the cycle time T is required to be a constant value. Otherwise, as the cycle time depends on service and idle

periods, the objective function (8) is non-linear. Then, the solution $J_w(T)$ with minimal costs for cycle times within the range

$$T^* \leq T \leq \min\left(T^{\max}, \frac{\tau_n^{\max}}{\rho_n}\right), \quad n = 1, 2,$$

renders the optimal steady-state costs J_w^*, which can be easily found.

The optimal trajectory, within the constraints, is a trade-off between loss of capacity due to setups, slow-modes and the average setup costs. Elongating the cycle time by including a slow-mode or creating backlog, results in less switches over time where capacity is lost due to setups and lowers the average setup costs. For a system without backlog, a typical steady-state trajectory is depicted in Figure 5. The trajectory consists of six characteristic points, labeled in alphabetical order $A-F$. The optimal trajectory for systems *without constraints* contains at most one slow-mode, i.e., $F = A$ or $C = D$. Furthermore, if no setup periods are considered, $D = E$ and $A = B$. Optimal policies will serve queue i until the other queue j reaches a threshold. Therefore, the trajectory can include slow-modes at both queues. A special case of this model, with $\mu_1 = \mu_2$ and $c_1 = c_2$ has been studied in [1, 4, 7, 8, 11] and it is shown that the optimal policy is a *clearing* policy, i.e., the server empties a queue and then switches to serve the other queue.

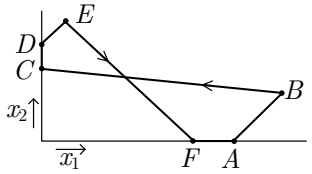
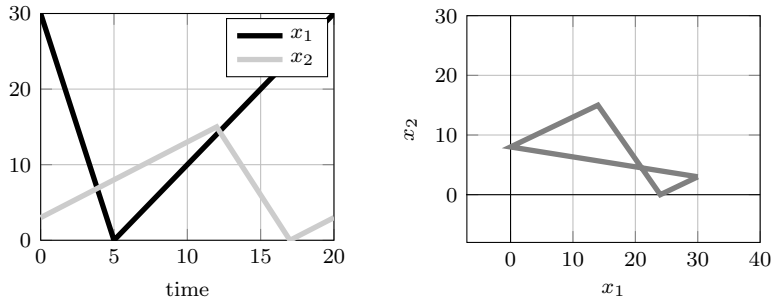


Figure 5: Optimal steady-state trajectory, with characteristic points $A - F$.

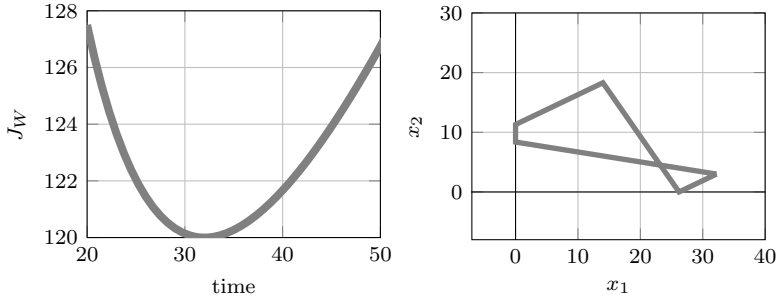
3.4. Illustrations

Using the method described above, we illustrate some optimal steady-state trajectories for the two queue switching server. We consider again the system with parameters (10). Note that, at first, backlog is not allowed for this system. From (4), we find the minimal cycle time $T^* = 20$. The corresponding steady-state trajectory with minimal costs is presented in Figure 6. Here, in the figure on the left, the evolution of the queue contents during a cycle are presented. In the figure on the right, the periodic trajectory, i.e., x_1 versus x_2, is presented. It can be seen that no slow-mode occurs in the trajectory, as expected by considering the minimal cycle time. This trajectory also yields the optimal total costs $J_c^* = 2550$ (and $J_w = 127.5$). For the time average costs as performance indicator, the trajectory with minimal cycle time does not result in the optimal trajectory. The steady-state costs are depicted in Figure 7a. The minimum is located at $T = 32$, and the corresponding trajectory, depicted in Figure 7b,



(a) Queue contents versus time. (b) Optimal trajectory.

Figure 6: Optimal queue contents over time (left) and periodic trajectory (right) during a cycle for the system with parameters (10) and $T = T^* = 20$.



(a) Time average costs. (b) Optimal trajectory.

Figure 7: Time average costs versus cycle time (left) and optimal trajectory for $T = 32$ (right) for the system with parameters (10).

is the optimal steady-state trajectory. The optimal costs are $J_w^* = 120$ with service periods $\tau_1^\mu = 6$, $\tau_1^\lambda = 8$, $\tau_2^\mu = 8$ and $\tau_2^\lambda = 0$.

Next, the effects of setup costs, backlog and the constraints are presented in a stepwise manner. Starting from the system with parameters (10), without setup costs, without constraints on service periods and no backlog, we add parameters and restrictions step by step and analyze the steady-state trajectory optimizing the time averaged costs. Note that, for each trajectory, the previous constraints are preserved. Adding setup costs $s_{2,1} = 300$ and $s_{1,2} = 200$ to the system results in the optimal trajectory depicted in Figure 8a. The optimal cycle time, costs, service periods and minimal queue contents for each trajectory are presented in Table 1. Compared to the trajectory depicted in 7b, it can be seen that the addition of setup costs elongates the cycle time and increases the duration of the slow-mode. This is also expected, since it is beneficial to enlarge the cycle time as the costs of switching become larger.

Allowing backlog, with backlog costs $c_1^- = 50$ and $c_2^- = 3$, shifts the optimal

trajectory downwards and enlarges the cycle time, see Figure 8b. For queue 2, the inventory and backlog costs are equal. Note that no backlog occurs at queue 1, which is optimal due to the long slow-mode and the high costs of backlog. Next, the service period of queue 1 is restricted ($\tau_1^{\max} = 15$), resulting in the optimal trajectory depicted in Figure 8c. The service period of queue 1 for this trajectory is the maximal service period, see Table 1. Adding upper bounds on the queue contents, $x_1^{\max} = 35$ and $x_2^{\max} = 16$, also reduces the cycle time, see Figure 8d. Also, both upper bounds are reached in the trajectory. In Figure 8e, the optimal trajectory of the system with a maximal cycle time $T^{\max} = 25$ is depicted. This trajectory has a cycle time of 20 time units, the minimal required cycle time (no slow-modes), and also queue 1 has backlog.

For systems that require a minimal amount of products in the queue, or a maximal allowed amount of backlog, we add minimal queue constraints to the model. In Figure 8f, the optimal trajectory is depicted where the content of queue 1 is at least 2 and the backlog of queue 2 can not exceed 2, i.e., $x_1^{\min} = 2$ and $x_2^{\min} = -2$. Unlike the previous trajectory, the optimal trajectory is not the trajectory with the minimal cycle time and queue 2 reaches both boundaries. Finally, in Figure 8g the optimal steady-state trajectory for this system without setup periods is depicted. Due to the setup costs, this trajectory is not the fixed point $(2, 0)$. The optimal cycle time is 20 time units, which is not the minimal cycle time for this system.

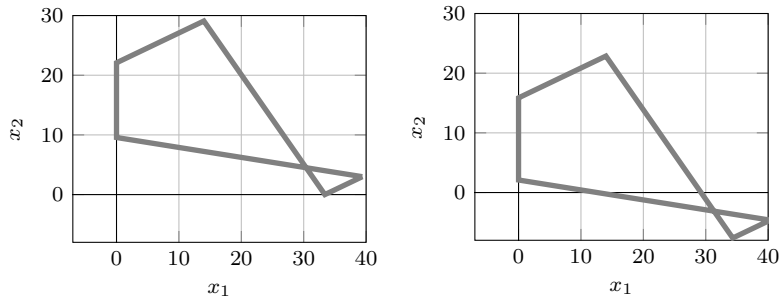
Fig.	T	J_w^*	τ_1^μ	τ_1^λ	τ_2^μ	τ_2^λ	$\underline{x}_1$	$\underline{x}_2$
7b	32	120	6	8	8	0	0	0
8a	38,78	134,13	6,57	12,52	9,70	0	0	0
8b	40,65	130,41	6,72	13,77	10,16	0	0	-7,62
8c	33,33	131,93	6,11	8,89	8,33	0	0	-6,25
8d	30	134,06	5,83	6,67	7,5	0	0	-6,5
8e	20	134,07	5	0	5	0	-4,14	-3,75
8f	24	158,06	5,33	2,67	6	0	2	-2
8g	20	60,37	1,67	13,33	5	0	2	-2

Table 1: Optimal cycle times, costs, service periods and minimal queue contents for the trajectories depicted in Figures 7-8.

4. Optimal transient trajectory

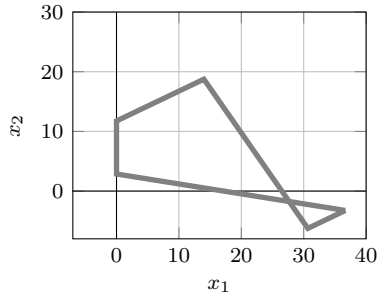
The transient optimization problem is that of steering the system towards the optimal steady-state trajectory at minimal costs. Machine failure in a manufacturing application or bus priorities in a signalized traffic intersection are two examples that can remove the system from the steady-state trajectory. We assume that deviations from the steady-state trajectory rarely occur, allowing the system to recover to the steady-state situation after each interruption,

which is reasonable for systems such as traffic intersections or manufacturing applications.

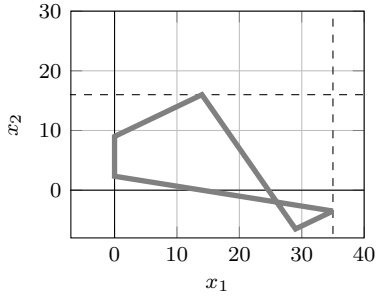


(a) $s_{2,1} = 300$ and $s_{1,2} = 200$.

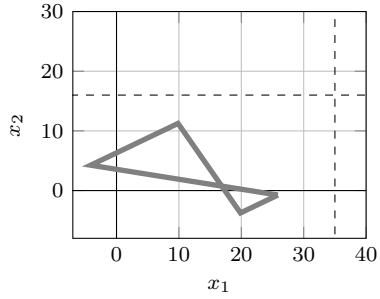
(b) $c_1^- = 50$ and $c_2^- = 3$.



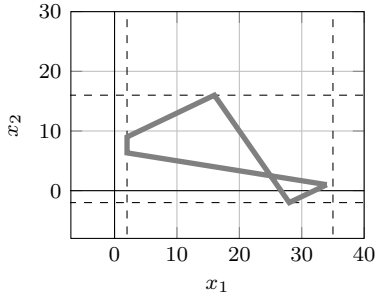
(c) $\tau_1^{\max} = 15$.



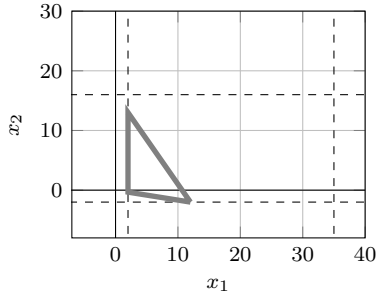
(d) $x^{\max} = [35, 16]^\top$.



(e) $T^{\max} = 25$.



(f) $x^{\min} = [2, -2]^\top$.



(g) No setup periods.

Figure 8: Optimal steady-state trajectories for the system with parameters (10), minimizing the time average costs. The system is subsequently extended with setup costs (a), backlog (b), maximal service period (c), maximal queue contents (d), maximal cycle time (e) and minimal queue contents (f). In (g), no setup periods are considered.

A transient solution is defined as a trajectory in the $x_1 - x_2$ space that leads to the optimal steady-state trajectory in a *finite* amount of time. An *optimal* transient solution is a transient solution which minimizes the costs of reaching the optimal steady-state trajectory. For the remainder of this paper, we define the initial state $x(0)$ as the state immediately after removal from the periodic solution, e.g., after the machine failure or bus priority. In order to reach the steady-state trajectory from every possible initial state in finite time, the steady-state trajectory requires a slow-mode, since serving at a lower rate, i.e., not at full capacity, provides the transient trajectory to 'catch up' with the steady-state trajectory.

For a *fixed* number of cycles C, we present the transient optimization problem as a QP problem. A cycle, starting at mode m, is defined as the series of operations until the end of the previous mode (which is 4 for mode 1, 1 for mode 2, etc.). Denote by $\tau_{i,c}$ the service time of queue i for the c-th cycle ($c \leq C$), consisting of the service time at maximal rate $\tau^{\mu}_{i,c}$ and the service time at arrival rate $\tau^{\lambda}_{i,c}$. Also, $x_{n,c}$ denotes the content of queue n at the end of the c-th cycle:

$$x_{n,c+1} = x_{n,c} + \lambda_n(T_c - \tau^{\lambda}_{n,c}) - \mu_n \tau^{\mu}_{n,c}, \quad n = 1, 2, \quad c = 1, 2, ..., C,$$

where $x_{n,0} = x_n(0)$. In the remainder of this paper we assume that the initial mode is 1, i.e., start setting up to serve queue 1, and derive the QP problem for this particular case. The QP problems for the other initial modes can be derived similarly. Constraints for the transient problem (c.f. (1) and (3)) are listed below. Minimal and maximal cycle time constraints are:

$$T^{\min} \leq T_c \leq T^{\max}, \qquad c = 1, 2, ..., C, \tag{11a}$$

with $T_c = \tau^0_{1,c} + \tau^{\mu}_{1,c} + \tau^{\lambda}_{1,c} + \tau^0_{2,c} + \tau^{\mu}_{2,c} + \tau^{\lambda}_{2,c}$. Minimal and maximal service periods:

$$\tau^{\min}_n \leq \tau_{n,c} \leq \tau^{\max}_n, \qquad \text{for } n = 1, 2, \quad c = 1, 2, ..., C, \tag{11b}$$

and minimal idle time

$$\sigma_{j,n} \leq \tau^0_{n,c}, \qquad n, j = 1, 2, \quad j \neq n, \quad c = 1, 2, ..., C. \tag{11c}$$

If the transient optimization problem is considered for an infinite number of cycles, the transient trajectory would remain on the steady-state trajectory once it is reached. However, due to the finite number of cycles considered in the QP problem, a termination effect occurs. Clearly, the direct costs vary over the optimal periodic orbit, and typically, at the end of a service mode the direct costs are less than the average. Therefore, assuming the transient solution is on the periodic orbit at the final cycle, prolonging this final cycle by enlarging $\tau^{\lambda}_{2,C}$ (instead of switching) is beneficial. To negate this termination effect, we enforce the final state of the final cycle C of the transient solution to be identical to the final state of the steady-state solution, i.e.,

$$x_{n,C} = x^*_n, \qquad n = 1, 2, \tag{11d}$$

with x_n^* the content of queue n at the start/end of the optimal steady-state trajectory, i.e., the content at the start of the setup to serve queue 1. If (11d) holds, the trajectory is defined as a *feasible* transient trajectory, otherwise the trajectory is *infeasible*. Deriving the optimal transient trajectory is discussed for two classes of two queue switching servers: servers without backlog and servers with backlog.

4.1. System without backlog

For the system without backlog, the transient costs are defined by

$$J_p = \liminf_{t\to\infty} \int_0^t [c_1^+ x_1(\tau) + c_2^+ x_2(\tau) + s_{21} v_1(\tau) + s_{12} v_2(\tau) - J_w^*] d\tau, \tag{12}$$

with $v_n(t) = \frac{1}{\sigma_{j,n}}$, $j \neq n$ during a setup to queue n and $v_n(t) = 0$ otherwise, and recall that J_w^* is the optimal time average steady-state cost of the system. Furthermore, service is at maximal rate if the queue is nonempty, otherwise at arrival rate. The total queue contents $w_{n,c}$ of queue n for cycle c satisfies:

$$\begin{aligned} w_{1,c} =& (x_{1,c-1} + \tfrac{1}{2}\lambda_1 \tau_{1,c}^0)\tau_{1,c}^0 + (x_{1,c-1} + \lambda_1 \tau_{1,c}^0 - \tfrac{1}{2}\mu_1 \tau_{1,c}^\mu)\tau_{1,c}^\mu + \\ &+ (x_{1,c-1} + \lambda_1 \tau_{1,c}^0 - \mu_1 \tau_{1,c}^\mu)\tau_{1,c}^\lambda + (x_{1,c-1} + \lambda_1 \tau_{1,c}^0 + \\ &+ \tfrac{1}{2}\lambda_1(\tau_{2,c}^0 + \tau_{2,c}^\mu + \tau_{2,c}^\lambda) - \mu_1 \tau_{1,c}^\mu)(\tau_{2,c}^0 + \tau_{2,c}^\mu + \tau_{2,c}^\lambda), \quad c = 1, 2, ..., C, \end{aligned} \tag{13a}$$

$$\begin{aligned} w_{2,c} =& (x_{2,c-1} + \tfrac{1}{2}\lambda_2(\tau_{1,c}^0 + \tau_{1,c}^\mu + \tau_{1,c}^\lambda + \tau_{2,c}^0))(\tau_{1,c}^0 + \tau_{1,c}^\mu + \tau_{1,c}^\lambda + \tau_{2,c}^0) + \\ &+ (x_{2,c-1} + \lambda_2(\tau_{1,c}^0 + \tau_{1,c}^\mu + \tau_{1,c}^\lambda + \tau_{2,c}^0) - \tfrac{1}{2}\mu_2 \tau_{2,c}^\mu)\tau_{2,c}^\mu + \\ &+ (x_{2,c-1} + \lambda_2(\tau_{1,c}^0 + \tau_{1,c}^\mu + \tau_{1,c}^\lambda + \tau_{2,c}^0) - \mu_2 \tau_{2,c}^\mu)\tau_{2,c}^\lambda, \quad c = 1, 2, ..., C. \end{aligned} \tag{13b}$$

Using (13), the transient costs (12), considering C cycles, can be written as

$$J_p(C) = C(s_{1,2} + s_{2,1}) + Q_p(C),$$

where $Q_p(C)$ is the solution to the quadratic programming problem, for C cycles, given by

$$Q_p(C) = \min_{\tau_{n,c}^0, \tau_{n,c}^\mu, \tau_{n,c}^\lambda} \sum_{n=1}^{2} \sum_{c=1}^{C} \left[c_i^+ w_{n,c} - J_w^*(\tau_{n,c}^0 + \tau_{n,c}^\mu + \tau_{n,c}^\lambda) \right], \tag{14}$$

subject to constraints (3c), (11), (13), and

$$x_{1,c} \geq \lambda_1(\tau_{2,c}^0 + \tau_{2,c}^\mu + \tau_{2,c}^\lambda), \qquad c = 1, 2, ..., C, \tag{15a}$$

$$x_{2,c} \geq 0, \qquad c = 1, 2, ..., C, \tag{15b}$$

$$x_{1,c-1} \leq x_1^{\max} - \lambda_1 \tau_{1,c}^0, \qquad c = 1, 2, ..., C, \tag{15c}$$

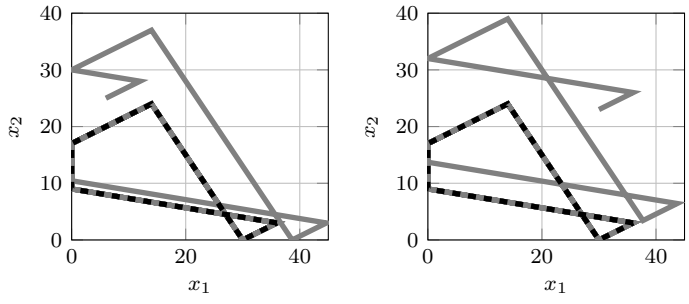
$$x_{2,c} \leq x_2^{\max} - (\mu_2 - \lambda_1)\tau_{2,c}^\mu, \qquad c = 1, 2, ..., C, \tag{15d}$$

where constraints (15a) and (15b) follow from $x_i(t) \geq 0$, and constraints (15c) and (15d) follow from (3c).

Given a system with the initial state outside the steady-state trajectory, the number of cycles required to derive the optimal transient trajectory is not easily determined. However, a lower bound on the number of cycles required for a feasible transient trajectory $C^{\min}$ can be determined by using a clearing policy, regarding the initial state and considering a system without capacity or service period constraints. For a system with capacity or service period constraints, this number of cycles is usually not enough to reach the steady-state trajectory. Starting from this lower bound, and by adding extra cycles, we solve the QP problem until a feasible transient trajectory is derived. Note that this transient trajectory is not necessarily the optimal trajectory, i.e., adding more cycles may lower the costs. Therefore, the number of cycles considered in the QP problem (14) is increased until the total costs required for the transient trajectory to reach the steady-state trajectory does no longer change, i.e., $J_p^*(C) = J_p^*(C+i)$, $\forall i > 0$. Then, adding more cycles does not result in a different transient trajectory, in the sense that it only adds steady-state cycles to the solution. Hence, this suggests that the transient solution is the optimal one.

4.1.1. Illustrations

For the system with parameters (10) and without constraints on queue length, cycle time and service periods, the optimal transient trajectory for initial state $x(0) = \begin{bmatrix} 3, & 6, & 25, & 1 \end{bmatrix}$ is presented in Figure 9a by the solid line. The optimal steady-state trajectory is depicted by the dashed line. Note, that the initial mode is 1, and that the steady-state trajectory is reached during the second cycle. It can be seen that for this initial state a clearing policy (until the steady-state trajectory is reached) yields the optimal performance. However, the optimal trajectory for the system with initial state $x(0) = \begin{bmatrix} 3, & 30, & 23, & 1 \end{bmatrix}$, presented in Figure 9b, gives a different result. First, after the setup, queue 1 is emptied. Second, after the setup, queue 2 is served until a content of 3.43 is reached, then the system switches to serve queue 1. Note that queue 2 is not emptied. Next, queues 1 and 2 are both cleared before reaching the steady-state trajectory.



(a) $x(0) = \begin{bmatrix} 3, & 6, & 25, & 1 \end{bmatrix}$. (b) $x(0) = \begin{bmatrix} 3, & 30, & 23, & 1 \end{bmatrix}$.

Figure 9: Optimal transient trajectories with different initial states. In (a) the clearing policy is optimal, in (b) it is not.

For the trajectory depicted in Figure 9b, it is clearly shown that a trade-off exists between a build-up of the much more expensive queue 1 and switching before emptying queue 2. This behavior is not present in symmetric systems, as a clearing policy is optimal for symmetric systems, see for instance [1, 11].

Each optimal transient trajectory contains *switching points*. A switching point is the state $x = \begin{bmatrix} x_0, & x_1, & x_2, & m \end{bmatrix}$ at which the system switches to serve the other queue, i.e., switching between modes $m = 2$ and $m = 3$ and between modes $m = 4$ and $m = 1$. Experimentally combining the switching points of optimal trajectories, i.e., solving the transient problem for a set of initial states and collecting the switching points, results in a *switching curve*. A switching curve characterizes the optimal transient structure for any given initial state, provided that the server works at maximal rate. Note that for a system with constraints on the service period, switching curves may not exist in general, as the switching points are affected by the constraints and will depend on the initial state. For the system without service time constraints, switching curves can possibly be derived analytically as follows. Starting from the steady-state trajectory, an area can be characterized from which the transient trajectory converges with a single operation to the steady-state trajectory. Next, an area can be characterized for which the system converges to the steady-state trajectory in two steps, and the optimal service times can be derived. Continuing these steps might result in the switching curves.

The (experimentally determined) switching curves for the system with parameters (10) are presented in Figure 10, along with a trajectory for initial state $x(0) = \begin{bmatrix} 3, & 45, & 80, & 1 \end{bmatrix}$. The switching curve for a transition between modes $m = 2$ and $m = 3$ is given by the line starting from $x_1 = 0$ and $x_2 \geq 17$, where $(0, 17)$ is the switching point of the optimal steady-state trajectory. The switching curve for a transition between modes $m = 4$ and $m = 1$ is *discontinuous* with linear segments. These segments do not overlap, i.e., each initial state has a single optimal trajectory.

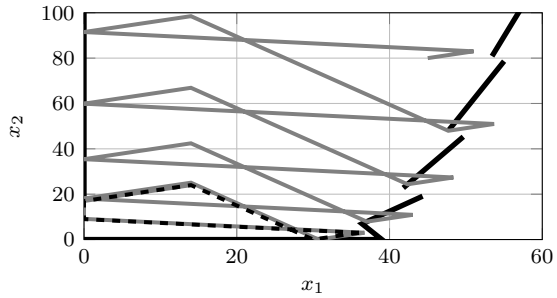
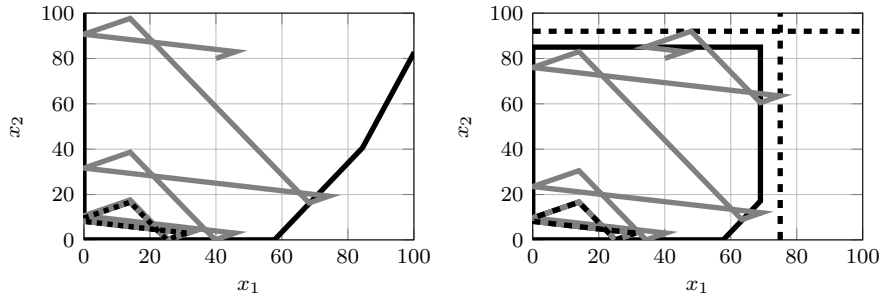


Figure 10: Discontinuous switching curves (black), for the system with parameters (10), and transient trajectory (gray) for $x(0) = [3 \;\; 45 \;\; 80 \;\; 1]$.

For the system with parameters (10) and $c_1^+ = 2$, the switching curve is continuous, see Figure 11a. Here, the switching curve for a transition between

modes $m = 4$ and $m = 1$ is piecewise linear.



(a) Unconstrained system

(b) Bounds $x_1^{\max} = 75$ and $x_2^{\max} = 92$.

Figure 11: Switching curves (black), for the system with parameters (10) and $c_1^+ = 2$, and transient trajectory (gray) for $x(0) = [3 \;\; 40 \;\; 80 \;\; 1]$.

Adding maximal queue length constraints $x_1^{\max} = 75$ and $x_2^{\max} = 92$ to this model results in the switching curves depicted in Figure 11b. The figure also displays the optimal transient trajectory for $x(0) = \begin{bmatrix} 3, & 40, & 80, & 1 \end{bmatrix}$. It can be seen that the switching curves, originating from the queue level constraints, are located $\lambda_n \sigma_{j,n}$ below $x_n^{\max}$, as the queue length increases during the setup. The initial queue contents, for starting in mode 1, are limited to $x_1(0) \leq x_1^{\max} - \lambda_1 \sigma_{2,1}$ and $x_2(0) \leq x_2^{\max} - \lambda_2 \sigma$.

For an optimal transient policy, the switching curves can be used to indicate the switching moments. From our experiments we find that for $c_n^+ \mu_n \geq c_j^+ \mu_j$, queue n is always emptied and the optimal policy for $c_n^+ \mu_n = c_j^+ \mu_j$ is, as expected, a clearing policy (unless prohibited by restrictions (11)).

Alongside the switching curves, for an optimal transient policy, also the optimal initial mode (given contents $x_0(0)$, $x_1(0)$ and $x_2(0)$), if it is not predefined, can be derived. Together with the switching curves, this gives the policy for optimal transient behavior given initial queue contents. Once the optimal initial state is known, the queues are served until a switching point is reached, switch to the successive mode, until converging to the optimal steady-state trajectory. If all initial modes are allowed, the states with setup modes $m(0) = 1$ and $m(0) = 3$ are of course not optimal. Therefore, a comparison of the transient costs starting with both modes $m(0) = 2$ and $m(0) = 4$ results in the optimal initial mode. For the system with parameters (10) and $c_1^+ = 2$, the optimal initial modes are presented in Figure 12, along with the switching curves. For initial queue contents in the gray area the optimal initial mode is $m(0) = 2$, $m(0) = 4$ otherwise.

Note that, for the system without backlog, the optimal transient trajectory does not include idling of the server. Also, a slow-mode only occurs while on the steady-state trajectory or to converge to this trajectory. For the system with backlog, presented next, idling of the server and slow-modes can occur in the optimal transient trajectory.

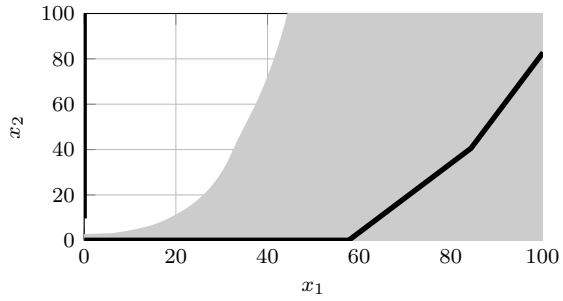


Figure 12: Switching curves (black) and optimal initial mode for the system with parameters (10) and $c_1^+ = 2$, $m(0) = 2$ in the gray area, $m(0) = 4$ otherwise.

4.2. System with backlog

For the system with backlog, the transient costs are defined by

$$J_p = \liminf_{t\to\infty} \int_0^t [c_1^+ x_1^+(\tau) + c_1^- x_1^-(\tau) + c_2^+ x_2^+(\tau) + c_2^- x_2^-(\tau) + \\ + s_{2,1} v_1(\tau) + s_{1,2} v_2(\tau) - J_w^*] d\tau,$$

Since Lemma 3.1 does not hold for each cycle in a transient trajectory, the inventory and backlog can not be calculated by (6). Therefore, extra variables β and δ are introduced, representing the duration of which the queue content is either positive or negative in a mode, to calculate the inventory and backlog. Consider queue 1 that starts cycle c in mode 1. The queue contents during this cycle are depicted in Figure 13.

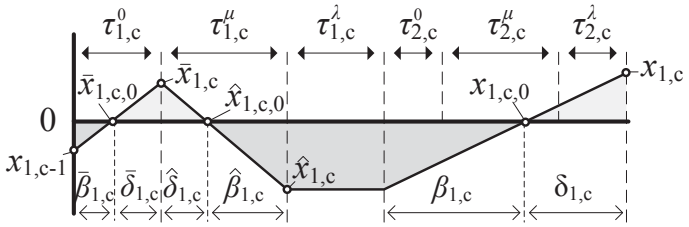


Figure 13: Evolution of x_1 during cycle c.

The periods in which the queue contents change, i.e., all periods except those serving at arrival rate, are divided into a part in which the content of queue n is positive, with a duration of $\delta_{n,c} \geq 0$, and a part in which the content is negative, with a duration of $\beta_{n,c} \geq 0$. For queue 1 it holds that

$$\bar{\delta}_{1,c} + \bar{\beta}_{1,c} = \tau^0_{1,c}, \quad \hat{\delta}_{1,c} + \hat{\beta}_{1,c} = \tau^\mu_{1,c}, \quad \delta_{1,c} + \beta_{1,c} = \tau^0_{2,c} + \tau^\mu_{2,c} + \tau^\lambda_{2,c}. \quad (16)$$

Furthermore, we denote by $\bar{x}_{1,c}$ the content of queue 1 in cycle c after the idle period $\tau^0_{1,c}$ and by $\hat{x}_{1,c}$ the content of queue 1 in cycle c after the period of

maximal service $\tau^{\mu}_{1,c}$, see Figure 13. We also denote by $\bar{x}_{1,c,0}$, the contents of queue 1 in cycle c when the idle period has taken $\bar{\beta}_{1,c}$, by $\hat{x}_{1,c,0}$ the content of queue 1 in cycle c after the period of maximal service has taken $\hat{\delta}_{1,c}$, and by $x_{1,c,0}$ the content of queue 1 in cycle c when the period of serving queue 2 has taken $\beta_{1,c}$, see Figure 13.

The queue levels at these time instances are divided into a positive and negative part, i.e.,

$$x_{n,c} = x^+_{n,c} - x^-_{n,c}, \qquad x_{n,c,0} = x^+_{n,c} - x^-_{n,c,0}, \qquad n = 1,2, \tag{17a}$$
$$\bar{x}_{n,c} = \bar{x}^+_{n,c} - \bar{x}^-_{n,c}, \qquad \bar{x}_{n,c,0} = \bar{x}^+_{n,c,0} - \bar{x}^-_{n,c,0}, \qquad n = 1,2, \tag{17b}$$
$$\hat{x}_{n,c} = \hat{x}^+_{n,c} - \hat{x}^-_{n,c}, \qquad \hat{x}_{n,c,0} = \hat{x}^+_{n,c,0} - \hat{x}^-_{n,c,0}, \qquad n = 1,2, \tag{17c}$$

where

$$x^+_{n,c} \geq 0 \qquad x^-_{n,c} \geq 0 \qquad x^+_{n,c,0} \geq 0 \qquad x^-_{n,c,0} \geq 0 \qquad n = 1,2, \tag{17d}$$
$$\bar{x}^+_{n,c} \geq 0 \qquad \bar{x}^-_{n,c} \geq 0 \qquad \bar{x}^+_{n,c,0} \geq 0 \qquad \bar{x}^-_{n,c,0} \geq 0 \qquad n = 1,2, \tag{17e}$$
$$\hat{x}^+_{n,c} \geq 0 \qquad \hat{x}^-_{n,c} \geq 0 \qquad \hat{x}^+_{n,c,0} \geq 0 \qquad \hat{x}^-_{n,c,0} \geq 0 \qquad n = 1,2. \tag{17f}$$

We also add the following equality constraints describing the evolution of the contents of queue 1 in cycle c, as well as the constraints that for changing queue levels only the positive respectively negative part is changing.

$$\bar{x}_{1,c,0} = x_{1,c-1} + \lambda_1 \bar{\beta}_{1,c} \qquad \bar{x}^-_{1,c,0} = x^-_{1,c-1} - \lambda_1 \bar{\beta}_{1,c} \tag{18a}$$
$$\bar{x}_{1,c} = \bar{x}_{1,c,0} + \lambda_1 \bar{\delta}_{1,c} \qquad \bar{x}^+_{1,c} = \bar{x}^+_{1,c,0} + \lambda_1 \bar{\delta}_{1,c} \tag{18b}$$
$$\hat{x}_{1,c,0} = \bar{x}_{1,c} - \mu_1 \hat{\delta}_{1,c} \qquad \hat{x}^+_{1,c,0} = \bar{x}^+_{1,c} - \mu_1 \hat{\delta}_{1,c} \tag{18c}$$
$$\hat{x}_{1,c} = \hat{x}_{1,c,0} - \mu_1 \hat{\beta}_{1,c} \qquad \hat{x}^-_{1,c} = \hat{x}^-_{1,c,0} + \mu_1 \hat{\beta}_{1,c} \tag{18d}$$
$$x_{1,c,0} = \hat{x}_{1,c} + \lambda_1 \beta_{1,c} \qquad x^-_{1,c,0} = \hat{x}^-_{1,c} - \lambda_1 \beta_{1,c} \tag{18e}$$
$$x_{1,c} = x_{1,c,0} + \lambda_1 \delta_{1,c} \qquad x^+_{1,c} = x^+_{1,c,0} + \lambda_1 \delta_{1,c} \tag{18f}$$

In terms of these new variables, the total inventory and backlog of queue 1 in cycle c are defined by respectively

$$\begin{aligned} w^+_{1,c} = {} & \tfrac{1}{2}\bar{\beta}_{1,c} x^+_{1,c-1} + \tfrac{1}{2}\tau^0_{1,c} x^+_{1,c,0} + \tfrac{1}{2}(\bar{\delta}_{1,c} + \hat{\delta}_{1,c})\bar{x}^+_{1,c} + \tfrac{1}{2}\tau^{\mu}_{1,c}\hat{x}^+_{1,c,0} + \\ & + \tfrac{1}{2}\hat{\beta}_{1,c}\hat{x}^+_{1,c} + \tau^{\lambda}_{1,c}\hat{x}^+_{1,c} + \tfrac{1}{2}\beta_{1,c}\hat{x}^+_{1,c} + \tfrac{1}{2}(\beta_{1,c} + \delta_{1,c})x^+_{1,c,0} + \tfrac{1}{2}\delta_{1,c}x^+_{1,c} \end{aligned} \tag{19a}$$

$$\begin{aligned} w^-_{1,c} = {} & \tfrac{1}{2}\bar{\beta}_{1,c} x^-_{1,c-1} + \tfrac{1}{2}\tau^0_{1,c} x^-_{1,c,0} + \tfrac{1}{2}(\bar{\delta}_{1,c} + \hat{\delta}_{1,c})\bar{x}^-_{1,c} + \tfrac{1}{2}\tau^{\mu}_{1,c}\hat{x}^-_{1,c,0} + \\ & + \tfrac{1}{2}\hat{\beta}_{1,c}\hat{x}^-_{1,c} + \tau^{\lambda}_{1,c}\hat{x}^-_{1,c} + \tfrac{1}{2}\beta_{1,c}\hat{x}^-_{1,c} + \tfrac{1}{2}(\beta_{1,c} + \delta_{1,c})x^-_{1,c,0} + \tfrac{1}{2}\delta_{1,c}x^-_{1,c} \end{aligned} \tag{19b}$$

For queue 2, we use a similar approach to derive the backlog and inventory during cycle c. The queue contents during this cycle are depicted in Figure 14. Denote by $\bar{x}_{2,c}$ the content of queue 2 in cycle c after the idle period $\tau^0_{2,c}$.

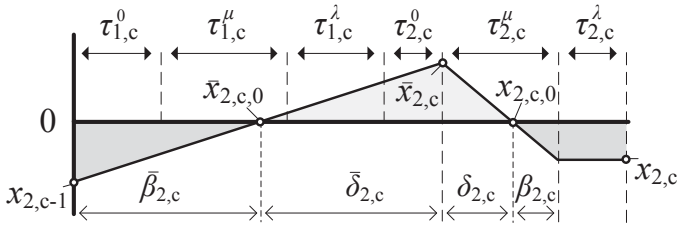


Figure 14: Evolution of x_2 during cycle c

The periods during which queue 2 is not served at arrival rate are divided according to

$$\bar{\delta}_{2,c} + \bar{\beta}_{2,c} = \tau^0_{1,c} + \tau^\mu_{1,c} + \tau^\lambda_{1,c} + \tau^0_{2,c}, \qquad \delta_{2,c} + \beta_{2,c} = \tau^\mu_{2,c}. \tag{20}$$

Then, the total inventory and backlog in cycle c is given by respectively

$$\begin{aligned} w^+_{2,c} = {} & \tfrac{1}{2}\bar{\beta}_{2,c}x^+_{2,c-1} + \tfrac{1}{2}(\bar{\beta}_{2,c} + \bar{\delta}_{2,c})\bar{x}^+_{2,c,0} + \tfrac{1}{2}(\bar{\delta}_{2,c} + \delta_{2,c})\bar{x}^+_{2,c} + \\ & + \tfrac{1}{2}\tau^\mu_{2,c}x^+_{2,c,0} + (\tfrac{1}{2}\beta_{2,c} + \tau^\lambda_{2,c})x^+_{2,c}, \end{aligned} \tag{21a}$$

$$\begin{aligned} w^+_{2,c} = {} & \tfrac{1}{2}\bar{\beta}_{2,c}x^-_{2,c-1} + \tfrac{1}{2}(\bar{\beta}_{2,c} + \bar{\delta}_{2,c})\bar{x}^-_{2,c,0} + \tfrac{1}{2}(\bar{\delta}_{2,c} + \delta_{2,c})\bar{x}^-_{2,c} + \\ & + \tfrac{1}{2}\tau^\mu_{2,c}x^-_{2,c,0} + (\tfrac{1}{2}\beta_{2,c} + \tau^\lambda_{2,c})x^-_{2,c}. \end{aligned} \tag{21b}$$

We also have the following equality constraints:

$$\bar{x}_{2,c,0} = x_{2,c-1} + \lambda_2\bar{\beta}_{2,c} \qquad \bar{x}^-_{2,c,0} = x^-_{2,c-1} - \lambda_2\bar{\beta}_{2,c} \tag{22a}$$

$$\bar{x}_{2,c} = \bar{x}_{2,c,0} + \lambda_2\bar{\delta}_{2,c} \qquad \bar{x}^+_{2,c} = \bar{x}^+_{2,c,0} + \lambda_2\bar{\delta}_{2,c} \tag{22b}$$

$$x_{2,c,0} = \bar{x}_{2,c} - \mu_2\delta_{2,c} \qquad x^+_{2,c,0} = \bar{x}^+_{2,c} + \mu_2\delta_{2,c} \tag{22c}$$

$$x_{2,c} = x_{2,c,0} - \mu_2\beta_{2,c} \qquad x^-_{2,c} = x^-_{2,c,0} - \mu_2\beta_{2,c} \tag{22d}$$

Then, the transient costs (16) for the system with backlog, considering C cycles, can be written as

$$J_p(C) = C(s_{1,2} + s_{2,1}) + Q_b(C).$$

Here, $Q_b(C)$ is the solution to the quadratic programming problem, for C cycles, given by

$$Q_b(C) = \min_{\tau^\mu_{n,c},\tau^\lambda_{n,c}} \sum_{n=1}^{2}\sum_{c=1}^{C} \left[c^+_n w^+_{n,c} + c^-_n w^-_{n,c} - J^*_w(\sigma_{n,j} + \tau^\mu_{n,c} + \tau^\lambda_{n,c})\right], \qquad n \neq j, \tag{23}$$

Objective function (23) is subject to constraints (16)–(22), and, from (3c),

$$x_{1,c} \leq x^{\max}_1, \qquad \bar{x}_{1,c} \leq x^{\max}_1, \qquad \bar{x}_{2,c} \leq x^{\max}_2.$$

Note that the constraints (17) only guarantee that $x^+_{n,c} = \max(0, x_{n,c}) + k$ and $x^-_{n,c} = \max(0, -x_{n,c}) + k$ for some constant $k \geq 0$. However, minimizing the objective function (23) with $c^+_n > 0$ and $c^+_n > 0$ guarantees that $k = 0$.

4.2.1. Illustrations

Consider the system with the following parameters

$$\begin{aligned} &\lambda_1 = 3, \quad \mu_1 = 8, \quad \sigma_{2,1} = 1, \quad c_1^+ = 2, \quad c_1^- = 20, \\ &\lambda_2 = 1, \quad \mu_2 = 9, \quad \sigma_{1,2} = 3, \quad c_2^+ = 1, \quad c_2^- = 10, \end{aligned} \tag{25}$$

and without setup costs or constraints on capacity or service periods. In Figure 15, the optimal steady-state trajectory (black solid line) and three optimal transient trajectories are depicted. The solid (gray) line represents the transient trajectory with $x(0) = \begin{bmatrix} 3, & 10, & -10, & 1 \end{bmatrix}$, where the server sets up to serve queue 2 directly after the setup to queue 1. The dashed line depicts the trajectory starting from $x(0) = \begin{bmatrix} 3, & -10, & -10, & 1 \end{bmatrix}$. Here, after the setup, the server idles until $x_1 = 0$ and then converges to the steady-state cycle using a slow-mode. This example shows that optimal transient trajectories can include idling of the server, unlike the trajectories for the system without backlog. This is also intuitive, as idling of the server is the quickest way to remove backlog. A third trajectory, depicted by the dotted line in Figure 15, starts from $x(0) = \begin{bmatrix} 3, & -10, & 10, & 1 \end{bmatrix}$ and also converges directly after emptying queue 1 and continuing service at arrival rate.

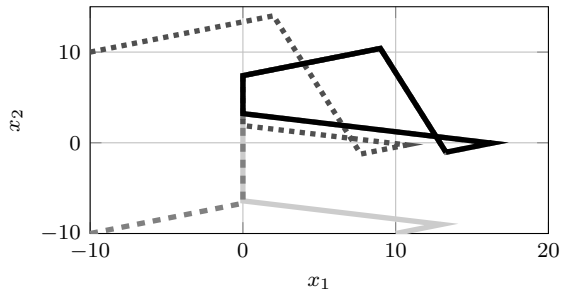


Figure 15: Optimal transient trajectories with different initial states, for the system with parameters (25).

In Figure 16, three optimal periodic trajectories are presented starting with a large backlog in queue 1 ($x_1(0) = -30$). The initial state for the trajectory depicted by the gray solid line is $x(0) = \begin{bmatrix} 3, & -30, & -5, & 1 \end{bmatrix}$, the trajectory depicted by dashed line $x(0) = \begin{bmatrix} 3, & -30, & 0, & 1 \end{bmatrix}$ and the trajectory depicted by dotted line $x(0) = \begin{bmatrix} 3, & -30, & 15, & 1 \end{bmatrix}$. It can be seen that the backlog is minimized as fast as possible, by switching to serve queue 2 and serving this queue or idling until $x_2 = 0$. Note that, for convergence to the steady-state trajectory, the server can switch earlier to serve queue 1. However, this reduction in time does not outweigh the extra backlog costs. Moreover, when the trajectory reaches the origin, the server can immediately switch to serve queue 1, resulting in a lower total inventory. For this system, this reduction in costs does not outweigh the cost reduction by elongating the cycle time. Therefore, queue 2 is served at arrival rate a little longer, i.e., until $x_2 = 2.06$.

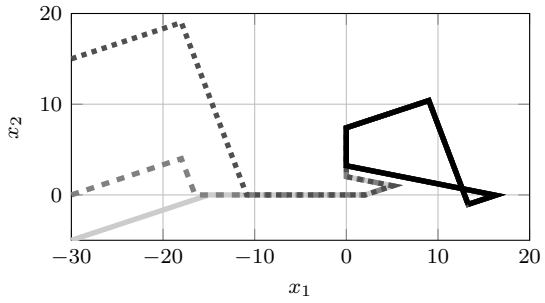
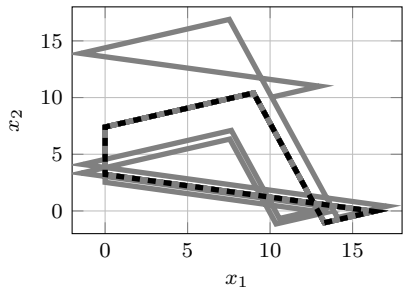
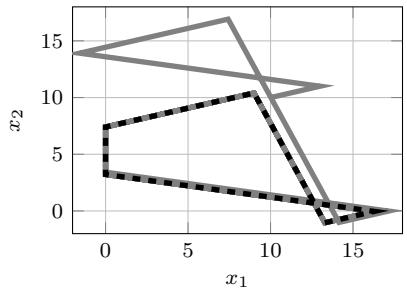


Figure 16: Optimal transient trajectories with different initial states, for the system with parameters (25).

In some cases, the optimal transient trajectory requires a few additional cycles to converge to the steady-state trajectory after the queue levels are directed to the vicinity of the steady-state trajectory. This trajectory has optimal costs, but another transient trajectory, which converges with less cycles to the steady-state trajectory and almost identical costs, might be desired by the operator, e.g., for simplicity of the trajectory. As an illustrative example, Figure 17 presents two different transient trajectories, both with initial state $\begin{bmatrix}3, & 10, & 0, & 1\end{bmatrix}$. Figure 17a depicts the optimal transient trajectory, which requires four cycles to converge to the steady-state trajectory. In Figure 17b, a transient trajectory is depicted which converges after two cycles, with almost similar costs, i.e., a difference of 1,7%. This trajectory might be desired over the optimal trajectory, as the costs are almost similar and the trajectory converges faster to the steady-state trajectory.



(a) Convergence after 4 cycles, J_P^*=105.85.



(b) Convergence after 2 cycles, J_P=107.64.

Figure 17: Two trajectories for $x(0) = [3\ \ 10\ \ 10\ \ 1]$.

Also, switching curves for the system with backlog, as presented for the system without backlog, provide no insight in the optimal transient behavior, due to this complex convergence for some of the optimal transient trajectories. To negate the occurrence of extra cycles just before converging, we add costs to

the number of cycles required to converge. Denote by c_c the additional transient costs of requiring an extra cycle to converge to the steady-state trajectory. Then, the transient costs $\bar{J}_p$ are denoted by

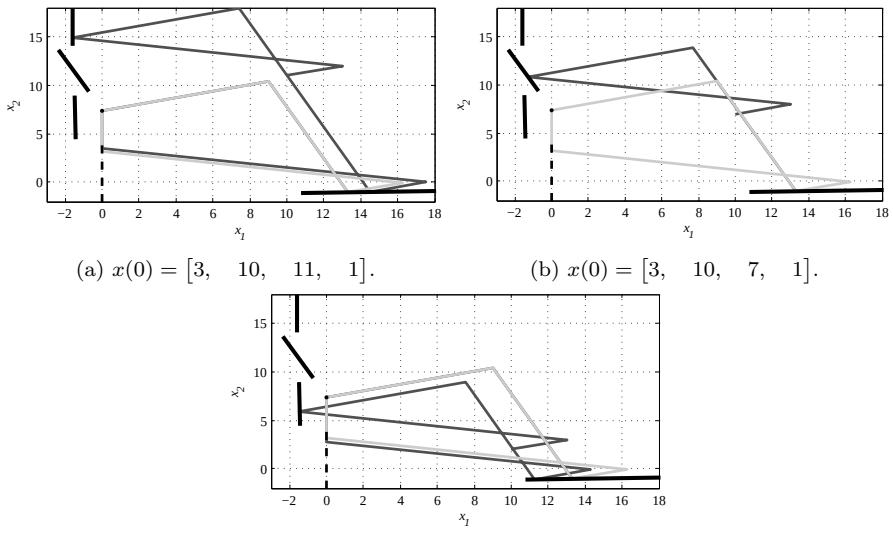
$$\bar{J}_p(C) = C(s_{1,2} + s_{2,1} + c_c) + Q_b(C). \tag{26}$$

Using (26), for the system with parameters (25) and $c_c = 5$, which is only 31% of the optimal steady-state costs J_w^*, the switching curves and three transient trajectories are depicted in Figure 18. These switching curves are depicted with the black solid lines, and are generated by optimizing trajectories for the system without backlog in the initial states. Also the optimal steady-state trajectory is depicted (gray) with corresponding switching points. It can be seen that the switching curves on the left side, which represent a transition between modes $m = 2$ and $m = 3$, consists of three line segments and single switching point on the steady-state trajectory. In Figure 18a a transient trajectory is depicted, with $x(0) = \begin{bmatrix} 3, & 10, & 11, & 1 \end{bmatrix}$ and converges to the steady-state trajectory during the slow-mode in the second cycle. The transient trajectory presented in Figure 18b, with initial state $x(0) = \begin{bmatrix} 3, & 10, & 7, & 1 \end{bmatrix}$, converges to the steady-state trajectory during service of queue 2 in the first cycle. Due to allowed backlog, this is possible for a whole range of trajectories, as they switch at the second line segment of the switching curve. Finally, in Figure 18c a transient trajectory is depicted for the system with initial state $x(0) = \begin{bmatrix} 3, & 10, & 0, & 1 \end{bmatrix}$. Note that the trajectory converges to the steady-state trajectory during service of queue 1 in the second cycle, while convergence was already possible during service of queue 1 in the first cycle. However, using two cycles results in lower costs (even with the added costs c_c). For a larger cost c_c, the use of this second cycle can be removed. Trajectories that reach a point on the dashed line, while serving queue 1, converge to the steady-state trajectory by using a slow-mode from that point.

For obtaining the results in this section, we used Matlab R2014b on a 1.9GHz Intel Core i5-4300U CPU. We used `quadprog` for solving the quadratic programs. Solving the QP's for determining the optimal steady-state trajectory took either 0.001s or 0.002s. Solving the QP's for the transient problem took in between 0.26s and 1.21s with an average of 0.34s.

5. Multi-queue switching servers

Extending this work to multi-queue switching servers imposes several problems. The first problem is the order of serving the queues. The proposed method can be extended to derive optimal steady-state and transient trajectories for multi-queue switching servers if the order of service of queues is predefined, for instance due to safety issues at traffic intersections or a fixed order of assembly at manufacturing systems. If the order is not predefined, the optimal trajectory can be derived by first determining the optimal trajectory for all possible orders of serving queues, and then selecting the one yielding the best performance. The second problem is the interpretation of the switching curves and optimal



(a) $x(0) = \begin{bmatrix} 3, & 10, & 11, & 1 \end{bmatrix}$.

(b) $x(0) = \begin{bmatrix} 3, & 10, & 7, & 1 \end{bmatrix}$.

(c) $x(0) = \begin{bmatrix} 3, & 10, & 2, & 1 \end{bmatrix}$.

Figure 18: Switching curves, optimal steady-state trajectory and three transient trajectories.

transient trajectories. If, for any initial state, the transient trajectories can be derived, a common policy (as specified by the switching curves) is not easily found or does not exist.

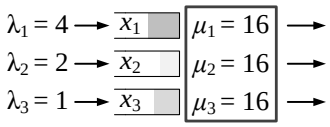
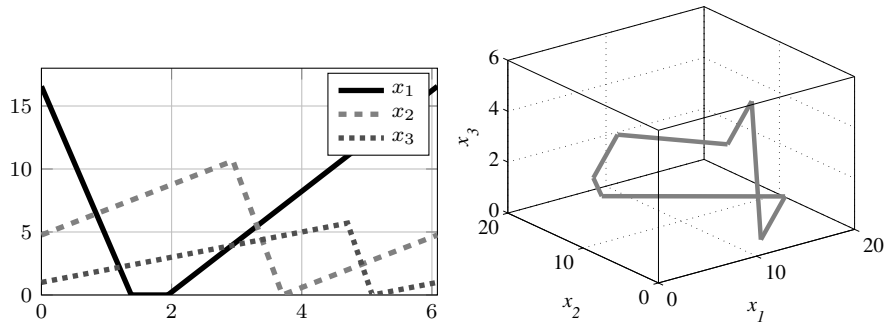


Figure 19: Three-queue switching server.

For example, consider a three-queue switching server, presented in Figure 19. The state of the 3-queue server is given by $x = \begin{bmatrix} x_0, & x_1, & x_2, & x_3, & m \end{bmatrix}$. The arrival and service rates are indicated in Figure 19, all setup periods have a duration of 1 and the costs are given by $c_1 = 4$, $c_2 = 2$, $c_3 = 1$. Furthermore, the server is restricted to serve only one queue at a time. Due to, for example, operator requirements, the service order of queues is fixed, i.e., assume without loss of generality that queues are served in the order 1, 2, 3 and then queue 1 again and so on. Also, backlog is not allowed, i.e., the queue lengths are nonnegative. Furthermore, no restrictions are imposed on the service periods or queue lengths. For this system, the optimal steady-state trajectory is presented in Figure 20. This trajectory has a single slow mode, after emptying queue 1.

Then, the optimal transient trajectory is derived similar as presented in Section 4.1, with the addition of service of queue 3. The combined switching points,



(a) Queues vs. time. (b) Trajectory.

Figure 20: Optimal steady-state trajectory of the 3-queue system.

derived by experiments, result in *switching areas*. A switch to serve queue 2 or queue 3 occurs when queues 1 or 2 are emptied, i.e., $x = \begin{bmatrix} 0, & 0, & \mathbb{R}, & \mathbb{R}, & 1 \end{bmatrix}$ or $x = \begin{bmatrix} 0, & \mathbb{R}, & 0, & \mathbb{R}, & 2 \end{bmatrix}$, respectively. The switching area indicating a switch between serving queue 3 and setting up to serve queue 1 is presented in Figure 21. It can be seen that, queue 3 is, once served, not always emptied, as the switching area is not represented by $x = \begin{bmatrix} 0, & \mathbb{R}, & \mathbb{R}, & 0, & 3 \end{bmatrix}$. This behavior is similar to the transient behavior of a two queue system.

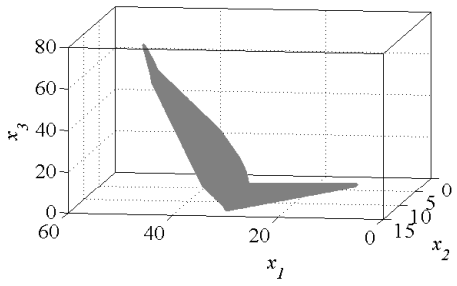


Figure 21: Switching area, switch from serving queue 3 to serve queue 1.

For this illustrative example, no constraints are imposed on the service or cycle times. Therefore, the server is allowed to switch to serve the next queue right after finishing a setup. This occurs for instance if $x(0) = \begin{bmatrix} 1, & 200, & 200, & 200, & 1 \end{bmatrix}$. The corresponding optimal transient trajectory is presented in Figure 22. It can be clearly seen that queue 3 is not served in the first cycle, i.e., after emptying queue 2, the system starts serving queue 1 again. To do this, first a setup to queue 3 is performed, followed by a setup to serve queue 1. This indicates that, if the order of service is not predefined, another sequence can result in better performance.

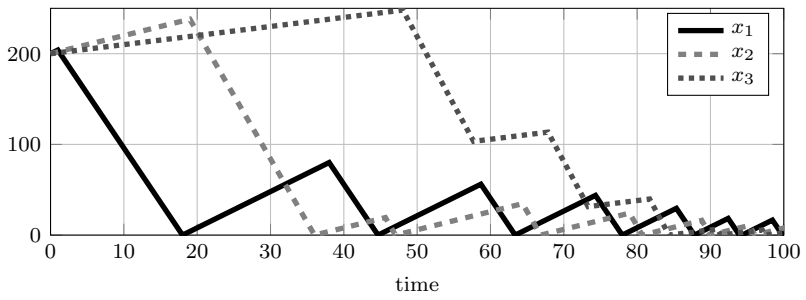


Figure 22: Optimal transient trajectory for $x(0) = [1,\ \ 200,\ \ 200,\ \ 200,\ \ 1]$.

Acknowledgments

This work is supported by the Netherlands Organization for Scientific Research (NWO-VIDI grant 639.072.072).

6. Conclusions

In this paper we studied the optimal steady-state and transient trajectories for a two queue switching server. The optimal steady-state trajectory is derived analytically for the system without backlog. For the system with backlog, and to allow multi-queue switching servers, the problem is formulated as a QP problem given a fixed cycle time. By solving the QP over a range of cycle times, the optimal steady-state trajectory is derived. An advantage of this method is that it is flexible in adding objectives and constraints, e.g., setup times and/or setup costs, including backlog and constraints on cycle times, service times and queue lengths.

Second, we formulated the transient problem, i.e., a transient trajectory which minimizes the costs of reaching the optimal steady-state trajectory. This problem is also formulated as a QP problem, depending on the number of cycles C imposed to reach the steady-state trajectory. Evaluating a range of values C, results in the optimal transient trajectory, given initial queue contents and initial mode. For the system without capacity constraints and no backlogs, switching curves can be derived by combining switching points of optimal trajectories, i.e., points at which the system switches to serve other queues. These switching curves are the blueprint of a policy for optimal transient behavior. Furthermore, the optimal initial mode, if not predescribed, can be derived. Together with the switching curves, this yields the control policy for optimal transient behavior. For the system with backlog, we introduced additional costs on the number of cycles required to converge to the steady-state trajectory, to derive a more simple schedule and to be able to derive switching curves for the unconstrained system.

Finally, we have shown that the presented approaches can be extended to multi-queue switching servers. For a fixed queue routing, i.e., fixed service order of queues, the approach can be easily extended by adding the extra queues.

[1] M. Boccadoro and P. Valigi. A switched system model for the optimal control of two symmetric competing queues with finite capacity. In L. Menini, L. Zaccarian, and C. Abdallah, editors, *Current Trends in Nonlinear Systems and Control*, Systems and Control: Foundations and Applications, pages 455–474. Birkhauser Boston, 2006.

[2] M. Bramson. Stability of queueing networks. *Probability Surveys*, 5:169–345, 2008.

[3] C. Chase and P. Ramadge. On real-time scheduling policies for flexible manufacturing systems. *IEEE Transactions on Automatic Control*, 37(4):491–496, 1992.

[4] M. Del Gaudio, F. Martinelli, and P. Valigi. A scheduling problem for two competing queues with finite capacity and non-negligible setup times. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 3, pages 2355–2360, 2001.

[5] J. Haddad, B. De Schutter, D. Mahalel, I. Ioslovich, and P.-O. Gutman. Optimal Steady-State Control for Isolated Traffic Intersections. *IEEE Transactions on Automatic Control*, 55(11):2612–2617, 2010.

[6] J. Haddad, P.-O. Gutman, I. Ioslovich, and D. Mahalel. Discrete dynamic optimization of N-stages control for isolated signalized intersections. *Control Engineering Practice*, 21(11):1553 – 1563, 2013.

[7] M. Hofri and K. Ross. On the optimal control of two queues with server setup times and its analysis. *SIAM Journal on Computing*, 16(2):399–420, 1987.

[8] V. Imbastari, F. Martinelli, and P. Valigi. An optimal scheduling problem for a system with finite buffers and non-negligible setup times and costs. In *Proceedings of the 41st IEEE Conference on Decision and Control*, volume 1, pages 1156– 1161, 2002.

[9] J. Kimemia and S. B. Gershwin. An algorithm for the computer control of a flexible manufacturing system. *IIE Transactions*, 15(4):353–362, 1983.

[10] Z. Liu, P. Nain, and D. Towsley. On optimal polling policies. *Queueing Systems*, 11:59–83, 1992.

[11] F. Martinelli and P. Valigi. Dynamic scheduling for a single machine system under different setup and buffer capacity scenarios. *Asian Journal of Control*, 6(2):229–241, 2004.

[12] A. Savkin. Optimal distributed real-time scheduling of flexible manufacturing networks modeled as hybrid dynamical systems. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, volume 5, pages 5468–5471, 2003.

[13] J. van Eekelen. *Modelling and control of discrete event manufacturing flow lines*. PhD thesis, Eindhoven, University of Technology, 2008.

[14] J. van Eekelen, E. Lefeber, and J. Rooda. Feedback control of 2-product server with setups and bounded buffers. In *Proceedings of the 2006 American Control Conference*, pages 544–549, 2006.

[15] D. van Zwieten, E. Lefeber, and I. Adan. Optimal periodical behavior of a multiclass fluid flow network. In *Proceedings of the 6th IFAC Conference on Management and Control of Production and Logistics*, pages 95–100, Fortaleza, Brazil, 2013.

[16] Y. Yang, B. Mao, S. Chen, S. Liu, and M. Liu. Effect of bus rapid transit signal priority effect on traffic flow. *Shenzhen University Science and Engineering*, 30(1):91–97, 2013.

[17] Zwieten, D.A.J. van, E. Lefeber, and I. Adan. Optimal steady-state and transient trajectories of a two queue switching server. In *Proceedings of the 7th International Conference on Performance Evaluation Methodologies and Tools*, pages 79–87, Torino, Italy, 2013.