Proportional Scheduling with Dynamic Cycle Lengths

Gustav Nilsson, Michelle Chong and Giacomo Como

Abstract— **The problem of service allocation to fulfill demands emerges in many engineering applications. For example, vehicles queuing up at a signalized junction, the servicing of jobs by the central processing unit (CPU), service demands at a cloud computing facility and many more. In this paper, we propose a feedback-based solution for service allocation. Our controller keeps the queue lengths bounded without knowledge of the exogenous arrivals. Moreover, the proposed controller allows for dynamic cycle lengths, i.e., the length of each service cycle is decided by the controller. This to capture the fact that during high demands, it can be more useful to let the service cycles last longer to reduce the fraction of time wasted in shifting between different services during a cycle.**

I. Introduction

The allocation of service is not only a well-recognized problem in the field of queuing theory [1] and wireless communication [2], but also in many other application areas. To mention a few examples, a flexible manufacturing system [3] where one robot can switch between different tasks, or in cloud computing [4] where individual applications can request service from multiple virtual machines. All these examples can be modeled by a dynamical system with the queue lengths as states which evolve based on a mass conservation principle involving the exogenous arrival and service rates, with limitations on which queues that can receive service simultaneously. The service allocation problem is then to ensure that the queue length is bounded by designing the service rate or the cycle length of the system.

In the seminal paper by Tassiulas and Ephremides [5], a control strategy for service allocation was proposed and proved to be maximally stabilizing, in the sense that all the queue lengths are bounded for some demand and if the demand is higher, no other control strategy will manage to keep the queue lengths bounded either. Based on the ideas from [5], several service allocation policies have been developed, commonly referred to as MaxWeight policies (see e.g. [6]). The idea behind these policies is that the service mode chosen by the controller at each time instant is one that maximizes a cost function that depends on the current queue lengths. Using these kinds of policies, the same service mode is chosen during the whole service cycle. The controller in [5] has also been adapted to different applications, e.g., the MaxPressure controller for traffic networks [7].

In this paper, we present a control strategy which in place of choosing one service mode to activate at each decision instance, it determines the fraction of the service cycle in which each mode should be activated and also the length of the service cycle. Similar controllers have previously been proposed for computer networks in [8], [9] and traffic networks [7], [10] with a fixed cycle time, and in a continuous time setting for traffic light control in [11], [12]. For real-time applications, the idea of having a variable cycle length (i.e., different lengths in service time for each task) has previously been proposed as a static open-loop scheduling problem in [13]. Additionally, we take into account the overhead time in switching between service modes when designing the control policy. This is crucial because in applications such as traffic systems, the driver requires some time to react to the green light at a signalized traffic junction and to then clear the junction before other lanes can be served. In traffic theory, this is commonly referred to as start up and clearance lost time [14]. Another example where a overhead time is present is CPU scheduling, where there is an overhead time in switching between different jobs called a context switch [15].

The contributions of this paper are the following: We provide a version of the controller proposed in [11], [12] for discrete time systems. We also show that in contrast to the continuous counterpart, stability can only be guaranteed by introducing an upper bound on the cycle length. While the proofs in [11], [12] and [8], [9] are done in continuous time (in [8], [9] the authors preform a fluid approximation of a stochastic system), we provide a direct discrete time proof to show stability. Also, to the best of the authors' knowledge, this is the first time that the stability of a discrete time scheduler with dynamic cycle lengths is analyzed.

The outline of the paper is the following. The rest of this section is devoted to introducing some basic notation. In Section II, the model is presented. In Section III, we present a theorem about the maximum stability region for any controller, i.e., the maximum size of inflows that any controller will be able to handle together with the proposed proportional controller and also provide explicit examples of the control strategy. In Section IV, we show that the controller is stabilizing whenever the inflows are strictly inside the maximum stability region, and in Section V, we illustrate the control strategy on a couple of examples. The paper is concluded with some points of interests for future research.

A. Notation

Let $\mathbb{Z}_+$ denote the set of non-negative integers, $\mathbb{R}_{(+)}$ denote the set of (non-negative) reals and $\mathbb{R}^{\mathcal{A}}_{(+)}$ the vector of (non-negative) reals indexed by the elements of the set

G. Nilsson, M. Chong and G. Como are with the Department of Automatic Control, Lund University, SE-221 00 Lund, Sweden {gustav.nilsson,michelle.chong,giacomo.como} @control.lth.se.

$\mathcal{A}$. With $|\mathcal{A}|$, we mean the cardinality of the set $\mathcal{A}$. We let $\mathbf{1}$ denote the vector with all ones and $\mathbf{0}$ the vector with all zeros. Moreover, we let $\mathcal{S}_{\mathcal{A}} := \{x \in \mathbb{R}_+^{\mathcal{A}} \mid x^T\mathbf{1} = 1\}$ be the probability simplex over the set $\mathcal{A}$. The norm of a vector is denoted $\|\cdot\|$. Inequalities for vectors are applied component wise to all its components, i.e., if $a < b$ where $a, b \in \mathbb{R}_+^{\mathcal{A}}$, then $a_i < b_i$ for all $i \in \mathcal{A}$. For a vector $v \in \mathbb{R}_+^n$, with $(v)_i$ we mean the ith component, where $1 \leq i \leq n$, and for any $u \in \mathbb{R}_+^n, v \in \mathbb{R}_+^m$, let (u, v) denote the vector $[u^T v^T]^T \in \mathbb{R}_+^{n+m}$.

II. Queue model and problem formulation

Let $\mathcal{Q}$ denote the set of queues. For each queue $i \in \mathcal{Q}$, its queue length $x_i \in \mathbb{R}_+$ changes at discrete time instants t_k, $k \in \mathbb{Z}_+$, according to

$$\begin{aligned} &t_0 = 0\,, \qquad x_i(0) = x_i^0\,, \qquad t_{k+1} = t_k + T(x(k))\,, \\ &x_i(k+1) = \max\{0, x_i(k) + T(x(k))(\lambda_i - u_i(x(k)))\}\,, \end{aligned} \tag{1}$$

where $x = (x_1, x_2, \ldots, x_q) \in \mathbb{R}_+^{\mathcal{Q}}$ is the vector of queue lengths, $\lambda_i > 0$ is the exogenous arrival rate, $T(x(k))$ and $u(x(k)) = (u_1(x(k)), u_2(x(k)), \ldots, u_q(x(k)))$ are the cycle length and, respectively, the averaged service allocation during cycle k, that are allowed to be dynamically adjusted as functions of the queue lengths through feedback policies $T : \mathbb{R}_+^{\mathcal{Q}} \to \mathbb{R}_+$ and $u : \mathbb{R}_+^{\mathcal{Q}} \to \mathbb{R}_+^{\mathcal{Q}}$ to be designed. In some applications, queues can receive services simultaneously. To model this, we introduce a finite set of modes $\mathcal{M}$ where each mode is a vector in $\mathbb{R}_+^{\mathcal{Q}}$, which determines how much service each queue will receive when the mode is activated. The feedback controller's task is then, apart from deciding the length of the cycle, also to decide the fraction of the cycle during which each mode should be activated. The fraction of the cycle when no queues can receive service is called the zero mode, which we represent by the $\mathbf{0}$ vector and is always in the set of modes. Moreover, we will assume that every queue can receive service according to at least one mode. Let m denote the number of modes and $M \in \mathbb{R}_+^{q \times m}$ be a matrix where each column is a mode. The first column of M contains the all zero mode. We formalize these assumptions below.

Assumption 1: The all zero mode is in the set of modes $\mathcal{M}$, i.e., $\mathbf{0} \in \mathcal{M}$. It is also assumed that $M\mathbf{1} > 0$, i.e., every queue can receive service according to at least one mode.

Example 1: For a simple system with two queues, where both can receive two units of service but not simultaneously, then the set of modes is

$$\mathcal{M} = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 2 \end{bmatrix} \right\}$$

and the corresponding M matrix is

$$M = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

The service allocation for each queue takes the form

$$u(x(k)) = M\theta(x(k))\,, \tag{2}$$

where $\theta = (\theta_0, \theta_1, \ldots, \theta_{m-1})$ and $\theta : \mathbb{R}_+^{\mathcal{Q}} \to \mathcal{S}_{\mathcal{M}}$ determines the fraction of the cycle length depending on the length of all the queues.

The fraction of the cycle allocated to the shifting between modes also depends on the length of all the queues, which corresponds to the all zero mode denoted by θ_0. Hence, the total cycle time for each cycle is

$$T(x(k)) = \frac{T_w}{\theta_0(x(k))}. \tag{3}$$

where $T_w \in (0, \infty)$ is the time needed to shift between different modes. We make the following assumptions about the overhead and the cycle times.

Assumption 2: We assume a strictly positive overhead time $T_w > 0$ and that there exists a maximum cycle time $T_{\max} > 0$ such that $T(x) \leq T_{\max}$.

This assumption is equivalent to assuming a minimum fraction of time is allocated to the zero mode, $w_0 > 0$, such that $\theta_0(x) \geq w_0$ for all $x \in \mathbb{R}_+^{\mathcal{Q}}$. In other words, we enforce a fraction of the cycle time to account for the overhead induced by switching between modes. Notice that by assuming a state-independent overhead time, we do also assume that each mode is activated only once during a cycle.

From this assumption, it follows that the cycle time is bounded both from below and above, i.e.,

$$T_w \leq T(x) \leq \frac{T_w}{w_0}, \quad \forall x \in \mathbb{R}_+^{\mathcal{Q}}.$$

Our objective is to guarantee bounded queue lengths (see Def. 1 below) via the feedback design of $\theta(x)$. To this end, we show in Proposition 1 that the arrival rate must lie within the region of stability (Def. 2) for any control strategy to ensure that the queue lengths are bounded.

Definition 1 (Bounded queue lengths): The dynamical system given by (1)-(3) is said to have *bounded queue lengths* if for all $i \in \mathcal{Q}$, there exists a constant vector $C \in \mathbb{R}_+^{\mathcal{Q}}$ such that $x(k) < C$ for all $k \in \mathbb{Z}_+$.

Definition 2 (Region of stability): For a given set of modes $\mathcal{M}$ and a given $w_0 > 0$, the arrival vector λ is said to be in the *region of stability* if there exists a $\bar{\theta} = (\bar{\theta}_0, \bar{\theta}_1, \ldots, \bar{\theta}_{m-1}) \in \mathcal{S}_{\mathcal{M}}$ such that

$$\lambda \leq M\bar{\theta}, \qquad \bar{\theta}_0 \geq w_0. \tag{4}$$

Moreover, if (4) is a strict inequality, i.e., $\lambda < M\bar{\theta}$, λ is said to be *strictly* inside the region of stability.

Proposition 1 (Necessary condition for stability): Consider the system (1)-(3) under Assumption 1 and 2 with any set of modes $\mathcal{M}$, $M \in \mathbb{R}_+^{q \times m}$, any arrival rate $\lambda > 0$, any possible mode allocation $\bar{\theta}(k)$ for all $k \geq 0$ and any $w_0 > 0$. If the queue lengths stay bounded, then λ must be in the region of stability.

Proof: Observe that

$$\begin{aligned} x(k) &= x(0) + \sum_{l=0}^{k-1} \max(T(x(l))(\lambda - u(l)), -x(l)) \\ &\geq x(0) + \sum_{l=0}^{k-1} T(x(l))(\lambda - u(l)). \end{aligned}$$

For $k \geq 1$, the inequality above can be rewritten as

$$\lambda \leq M\tilde{\theta}(k) + \frac{x(k) - x(0)}{t_k},$$

where

$$\tilde{\theta}(k) = \frac{1}{t_k}\sum_{l=0}^{k-1} T(x(l))\bar{\theta}(l),$$

belongs to $\mathcal{S}_{\mathcal{M}}$ such that $\theta_0 \geq w_0$, since $T(x) \geq T_w$ and $\frac{T(x(l))}{t_k} \in (0, 1]$.

Since $T(x) \geq T_w$, it follows that $t_k \to +\infty$ as $k \to +\infty$ and hence $\lambda \leq M\theta$. ■

Although the proof above is conservative, we will see that the bound is tight in Section IV. In particular, we will propose a control policy in the next section, for achieving the maximum possible arrival rate λ, without knowing the value of such λ.

III. Control strategy

Our control strategy is to design the fraction of the cycle length in which each mode is allocated, denoted as $\theta(x)$. We will refer to this strategy as the proportional controller given by

$$\theta(x) \in \underset{\theta \in \mathcal{S}_{\mathcal{M}}, \theta_0 \geq w_0}{\operatorname{argmax}} \sum_{i \in \mathcal{Q}} x_i \log\left((M\theta)_i\right) + \kappa \log(\theta_0), \quad (5)$$

where $\kappa > 0$ is a design parameter. The parameter κ determines how much the controller should deprioritize the overhead cost when splitting the cycle. A large κ will result in shorter cycles compared to a small κ.

Although it may not be possible to find explicit solutions to the convex optimization problem (5) for all possible set of modes $\mathcal{M}$, we provide an explicit solution for a common case below. In cases when an explicit solution is not possible to find, one can utilize the fact that the optimization problem (5) is convex, in order to numerically compute the service allocation and duration for the upcoming cycle.

Example 2: Let the modes be orthogonal, i.e., $m^T n = 0$ for all $m, n \in \mathcal{M}$ with arbitrarily large service rates, $\mu_i = (M\mathbf{1})_i$ for all $i \in \mathcal{Q}$, then

$$\theta_0(x(k)) = \max\left(\frac{\kappa}{\sum_{i \in \mathcal{Q}} x_i(k) + \kappa}, w_0\right),$$

and

$$u_i(x) = \begin{cases} \mu_i \frac{\sum_{j \in \mathcal{P}(i)} x_j(k)}{\sum_{j \in \mathcal{Q}} x_j(k) + \kappa} & \text{if } \theta_0(x) > w_0 \\ (1 - w_0)\mu_i \frac{\sum_{j \in \mathcal{P}(i)} x_j(k)}{\sum_{j \in \mathcal{Q}} x_j(k)} & \text{otherwise,} \end{cases}$$

for all $i \in \mathcal{Q}$, where $\mathcal{P}(i) \subset \mathcal{Q}$ is the set of queues in the same mode as queue $i \in \mathcal{Q}$.

IV. Main result and stability analysis

For the proportional controller presented in the previous section, the following theorem holds:

Theorem 1: Consider the system (1)-(3) under Assumptions 1 and 2 with any set of modes $\mathcal{M}$, $M \in \mathbb{R}_+^{q \times m}$, any arrival rate $\lambda > 0$ and any $w_0 > 0$. If λ is strictly inside the region of stability, then the proportional controller (5) ensures that the queue lengths are bounded for all initial conditions $x(0) \in \mathbb{R}_+^{\mathcal{Q}}$.

A. Proof of Theorem 1

To prove that the proportional controller given in (5) will keep the queue lengths bounded, we will use a Lyapunov function in the large [16].

Definition 3: (Lyapunov function in the large [16, Definition 3.2]) A *Lyapunov function in the large* for a discrete time dynamical system is a function $V : \mathbb{R}_+^n \to \mathbb{R}$ such that

i) V is radially unbounded, i.e., $\lim_{\|x\| \to +\infty} V(x) = +\infty$.
ii) there exists $\theta > 0$ with the following property: for all $r > \theta$ we can find $m > 0$ such that if $\theta \leq \|x\| \leq r$ then $V(x) \leq m$.
iii) there exists $\eta > 0$ such that if $\|x(k)\| \geq \eta$, then $V(x(k+1)) \leq V(x(k))$.

We start by showing that

$$V(x) = \sum_{i \in \mathcal{Q}} x_i \log\frac{u_i(x)}{\lambda_i} + \kappa \log(\theta_0(x)) \quad (6)$$

is a Lyapunov function in the large. The queue lengths are guaranteed to be bounded using Lyapunov based arguments [16, Theorem 3.1]. The following two results state some useful properties of the proportional controller and will be used to prove that $V(x)$ is a Lyapunov function in the large. The proofs of the following lemmas are in the appendix.

Lemma 1: Given a set of modes $\mathcal{M}$, a $w_0 > 0$ and an arrival vector λ which is strictly inside the region of stability, consider the system (1)-(3) under Assumption 1 and 2. For a constant $c \geq 0$, let $\mathcal{Q}_1(x) = \{i \in \mathcal{Q} \mid x_i < c\}$ and $Q_2(x) = \{i \in \mathcal{Q} \mid x_i \geq c\}$. Then, for service allocations given by the proportional controller (5), there exists some constants $C > 0$ and $\epsilon > 0$ such that, if $\|x\| > C$ then there exists at least one queue $i \in \mathcal{Q}_2$ such that $u_i(x) \geq \lambda_i + \epsilon$.

Lemma 2: Given a vector $x \in \mathbb{R}_+^{\mathcal{Q}}$ such that $x > 0$ and a vector $\epsilon \in \mathbb{R}^{\mathcal{Q}}$ such that $x + \epsilon > 0$. The proportional controller (5) satisfies the following: $\|\theta(x) - \theta(x + \epsilon)\| \to 0$ when $\|x\| \to +\infty$.

We now show that (6) is indeed a Lyapunov function in the large for system (1)-(3) with the proportional controller (5), by checking that all three conditions in Definition 3 are satisfied.

i) We observe from the definition of the region of stabilty that it is possible to find a $\bar{\theta}$ such that $\lambda_i + \epsilon_i \leq (M\bar{\theta})_i$ where $\epsilon_i > 0$ for all $i \in \mathcal{Q}$ and $\bar{\theta}_0 = w_0$. This choice of $\bar{\theta}$ is then a feasible but suboptimal solution to the maximization problem in (5). Hence

$$V(x) \geq \sum_{i \in \mathcal{Q}} x_i \log\frac{\lambda_i + \epsilon_i}{\lambda_i} + \kappa \log(w_0) \to +\infty,$$

when $\|x\| \to +\infty$.

ii) We have to ensure that for a bounded $\|x\|$, the Lyapunov function is bounded from above. Since $x \geq 0$ we have that

$$V(x) \leq \sum_{i \in \mathcal{Q}} x_i \log \frac{(M\mathbf{1})_i}{\lambda_i} + \kappa \log 1.$$

iii) As a last step, we need to show that the Lyapunov function is decreasing when $\|x\|$ is large enough. First observe that the dynamics (1) can be rewritten as

$$\begin{aligned} x_i(k+1) &= \max(x_i(k) + T(x(k))(\lambda_i - u_i(x(k))), 0) \\ &= x_i(k) + \max(T(x(k))(\lambda_i - u_i(x(k))), -x_i(k)). \end{aligned}$$

Introduce the set $\bar{\mathcal{Q}} := \{i \in \mathcal{Q} \mid x_i(k+1) > 0\}$. Then

$$\begin{aligned} & V(x(k+1)) - V(x(k)) = \\ & \sum_{i \in \mathcal{Q}} x_i(k+1) \log \frac{u_i(x(k+1))}{\lambda_i} + \kappa \log(\theta_0(x(k+1))) \\ & \quad - \sum_{i \in \mathcal{Q}} x_i(k) \log \frac{u_i(x(k))}{\lambda_i} - \kappa \log(\theta_0(x(k))) \\ & \leq \sum_{i \in \bar{\mathcal{Q}}} x_i(k+1) \log \frac{u_i(x(k+1))}{\lambda_i} + \kappa \log(\theta_0(x(k+1))) \\ & \quad - \sum_{i \in \bar{\mathcal{Q}}} x_i(k) \log \frac{u_i(x(k))}{\lambda_i} - \kappa \log(\theta_0(x(k))) \\ & \leq \sum_{i \in \bar{\mathcal{Q}}} x_i(k+1) \log \frac{u_i(x(k+1))}{\lambda_i} + \kappa \log(\theta_0(x(k+1))) \\ & \quad - \sum_{i \in \bar{\mathcal{Q}}} x_i(k) \log \frac{u_i(x(k+1))}{\lambda_i} - \kappa \log(\theta_0(x(k+1))) \\ & = \sum_{i \in \bar{\mathcal{Q}}} T(x(k))(\lambda_i - u_i(x(k))) \log \frac{u_i(x(k+1))}{\lambda_i} \\ & = \sum_{i \in \bar{\mathcal{Q}}} T(x(k))(\lambda_i - u_i(x(k))) \log \frac{u_i(x(k))}{\lambda_i} \\ & \quad + \sum_{i \in \bar{\mathcal{Q}}} T(x(k))(\lambda_i - u_i(x(k))) \log \frac{u_i(x(k+1))}{u_i(x(k))}, \end{aligned}$$

where the first inequality above follows from the fact that $x_i(k+1) = 0$ for $i \in \bar{\mathcal{Q}}$ while, for all $i \in \mathcal{Q} \setminus \bar{\mathcal{Q}}$ it must hold that $u_i(k) > \lambda_i$. The second inequality follows from suboptimality of the solution to (5) and the last equality from the fact that for all $i \in \bar{\mathcal{Q}}$, it holds that $\max(T(x(k))(\lambda_i - u_i(x(k))), -x_i(k)) = T(x(k))(\lambda_i - u_i(x(k)))$.

Obeserve that

$$T(x(k))(\lambda_i - u_i(x(k))) \log \frac{u_i(x(k))}{\lambda_i} \leq 0,$$

since $(a-b)\log \frac{b}{a} \leq 0$ for $a > 0$ and $b > 0$.

For all $i \in \mathcal{Q} \setminus \bar{\mathcal{Q}}$, there exists a constant $c > 0$, such that $x_i(k) < c$. Then, using Lemma 1, when $\|x\| > C$ where C is constant, there exists at least one $i \in \bar{\mathcal{Q}}$, with $u_i(x) > \lambda_i + \epsilon'$ with $\epsilon' > 0$. Hence, there exists a $\epsilon > 0$ such that

$$\sum_{i \in \bar{\mathcal{Q}}} T(x(k))(\lambda_i - u_i(x(k))) \log \frac{u_i(x(k))}{\lambda_i} < -\epsilon.$$

And since $|x_i(k+1) - x_i(k)|$ is bounded, due to the fact that $T(x(k))$ is bounded, it follows from Lemma 2 that for large enough $\|x\|$ that there exists $\epsilon'' > 0$ such that $\|u_i(x(k+1)) - u_i(x(k))\| < \epsilon''$. This implies that

$$\sum_{i \in \bar{\mathcal{Q}}} T(x(k))(\lambda_i - u_i(x(k))) \log \frac{u_i(x(k+1))}{u_i(x(k))} \leq \xi(x),$$

where $\xi(x) \to 0$ as $\|x\| \to \infty$. Hence

$$V(x(k+1)) - V(x(k)) \leq -\epsilon + \xi(x),$$

when $\|x\| > C$ and since there exists a $C' > C$ such that $-\epsilon + \xi(x) < 0$, negative drift is ensured.

Hence $V(x)$ is a Lyapunov function in the large (Def. 3), and the theorem is proved.

V. Examples

In this section, we illustrate the stabilization for the queue model (1)-(3) using the controller (5) on numerical examples. First, we consider a system of two queues and then illustrate the importance of having a non-zero w_0.

Example 3: Consider a system of two queues with the set of modes

$$M = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The optimization in (5) then becomes

$$\theta(x) \in \operatorname*{argmax}_{\theta \in \mathcal{S}_{\mathcal{M}},\, \theta_0 \geq w_0} x_1 \log(\theta_1) + x_2 \log(\theta_2) + \kappa \log(\theta_0),$$

with the solution

$$\theta_0(x) = \max\left\{w_0, \frac{\kappa}{x_1 + x_2 + \kappa}\right\}$$

and

$$u_i(x) = \begin{cases} \frac{x_i}{x_1+x_2+\kappa} & \text{if } \theta_0(x) > w_0 \\ (1-w_0)\frac{x_i}{x_1+x_2} & \text{otherwise,} \end{cases} \quad i \in \{1, 2\}.$$

Moreover, let $\lambda_1 = 0.4, \lambda_2 = 0.3, x_1(0) = 6, x_2(0) = 4, \kappa = 0.1, w_0 = 0.1$ and $T_w = 1$. In Fig. 1, it is shown how the queues evolve with time, together with the cycle time for each iteration. With $\kappa = 0.1$, we see in the figure that the controller becomes 2-periodic after some time, i.e., $T(x(k+2)) = T(x(k))$. However, if we let $\kappa = 1$, the cycle time $T(x(k))$ reaches a steady state instead. From Fig. 1, it can also be seen that during large queue lengths, the cycles will be longer. When the queue length has decreased, the cycle lengths are related to the choice of κ in (5).

The need for having a lower bound on the zero mode allocation can be illustrated with the following example.

Example 4: Let the set of modes be the same as in the previous example and let $\lambda_1 = \lambda_2 = \lambda$, $T_w = 1$, $w_0 = 0$. and $x_1(0) = C, x_2(0) = 0$. The service allocations and the cycle time for the first iteration is then given by

$$\begin{aligned} u_1(x(0)) &= \frac{C}{C+\kappa}, \\ u_2(x(0)) &= 0, \\ T(x(0)) &= \frac{C+\kappa}{\kappa}. \end{aligned}$$

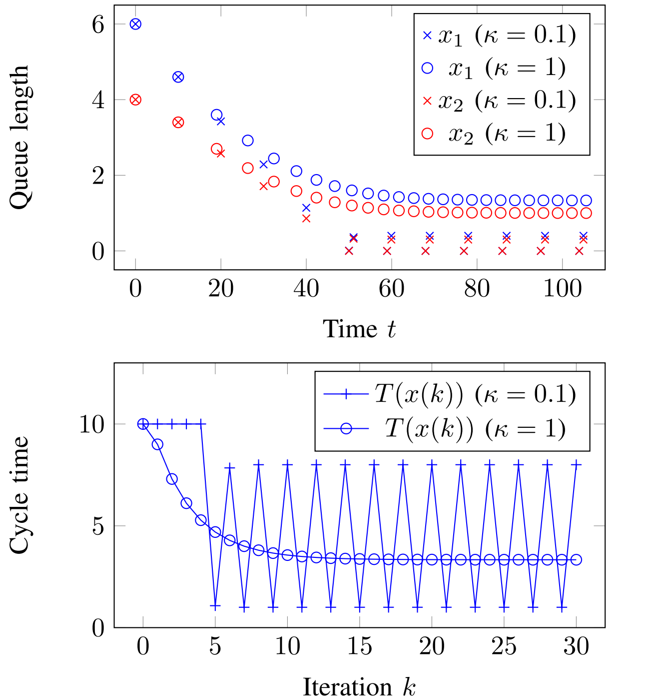

Fig. 1: How the queue lengths evolves in time together with the cycle times for the system in Example 3. The cross markers and the solid lines are with $\kappa = 0.1$ while the circle markers and the dashed lines are with $\kappa = 1$. Observe that the queue lengths are plotted with the actual time t, so the points are not equidistant due to the variable cycle length decided by the proportional controller (5).

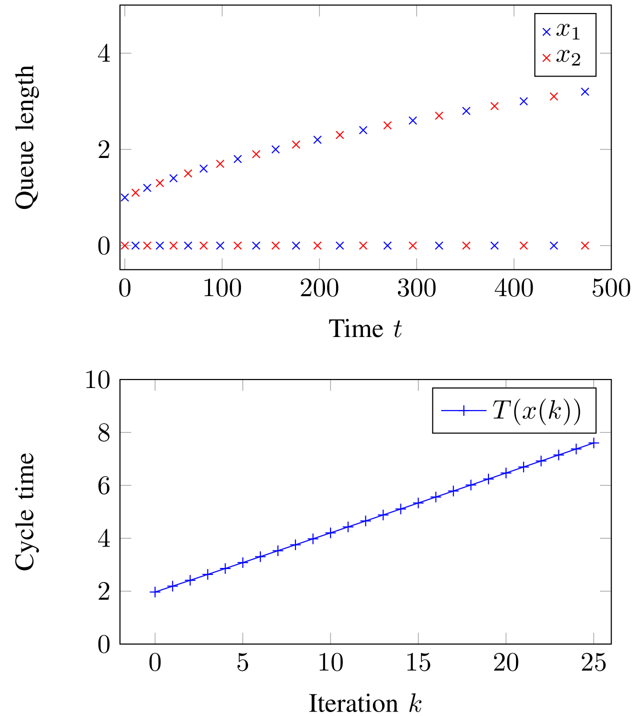

Fig. 2: How the queue lengths evolves in time together with the cycle times for the system in Example 4.

Observe that the cycle time $T(x(0))$ is strictly increasing with C. After one iteration the queue lengths are

$$x_1(1) = C + T(x(0))\left(\lambda - \frac{C}{C+\kappa}\right) = C + \overbrace{\lambda \frac{C+\kappa}{\kappa} - \frac{C}{\kappa}}^{f(C)},$$

$$x_2(1) = T(x(0))\lambda = \lambda\left(\frac{C+\kappa}{\kappa}\right).$$

If $x_1(1) = 0$, then due to symmetry, the analysis of the system can be repeated in the same way with a new initial condition. Observe that the queues will grow unbounded when $f(C) \leq 0, f'(C) \leq 0$ and $x_2(1) > C$, which can be equivalently expressed as

$$\kappa + \lambda - 1 \leq 0\,,$$
$$C\kappa - \lambda(C+\kappa) < 0\,,$$
$$C\kappa + \lambda(C+\kappa) - C \leq 0.$$

The choice of $\lambda = \kappa = 0.1$ and $C = 1$ is one set of parameters satisfying the constraints, and will hence make the queue lengths and cycle times grow unbounded. How the queue lengths and cycle times evolve is shown in Fig. 2.

VI. Conclusions and Future Work

In this paper, we have presented a controller for service allocation. Apart from deciding which queues should receive service, the proposed controller also decides the cycle length dynamically. This is to allow for longer cycle lengths during high demands, which is useful in practice. We have showed that the proposed controller is able to stabilize the network whenever it is possible for any controller to do so, without knowing the exogenous arrival rates.

Future research will involve studying how the choice of the design parameter κ affects the dynamics, and attempt to find an optimal choice of κ. Also, extensions to queueing networks will be studied, i.e., when the jobs has been served at one queue, the jobs proceed to another queue. Although the control strategies proposed in [5], [7], [9] have been shown to be stabilizing in networks, all of them are only designed for a fixed cycle length, i.e., the cycle time is not dynamically allocated as in this paper. Since the cycle times then may be different at different nodes, this will require a continuous time model with sampled control actions.

Appendix

A. Proof of Lemma 1

Proof: First we show that there exists a constant C' such that if $\|x\|_\infty > C'$ then $\theta_0(x) = w_0$.

Take $i \in \mathcal{Q}$ such that $x_i \geq x_j$ for all $j \in \mathcal{Q}$, then there must exists a j such that $M_{ij} > 0$. For that j and any choice of $\theta \in \mathcal{S}_\mathcal{M}$ with $\theta_0 > w_0$ define the function $\hat{\theta} : \mathbb{R}_+ \to \mathbb{R}_+^\mathcal{M}$ as $\hat{\theta}_j(\beta) = \theta_j + \beta$, $\hat{\theta}_0(\beta) = \theta_0 - \beta$ and $\hat{\theta}_k = \theta_k$ for all $k \notin \{0, j\}$, where $0 \leq \beta \leq \theta_0 - w_0$. Moreover, let $f(\theta, x) = \sum_{i\in\mathcal{Q}} x_i \log((M\theta)_i) + \kappa \log(\theta_0)$ denote the objective function in (5). Then

$$\frac{\partial}{\partial \beta} f(\hat{\theta}(\beta), x) = \frac{\partial f(\hat{\theta}(\beta), x)}{\partial \theta_j} - \frac{\partial f(\hat{\theta}(\beta), x)}{\partial \theta_0}$$
$$= \sum_{i \in \mathcal{Q}} \frac{M_{ij} x_i}{\sum_{k \in \mathcal{M}} M_{ik} \hat{\theta}_k(\beta)} - \frac{\kappa}{\hat{\theta}_0(\beta)}.$$

If this derivative is strictly positive in β, then the maximum of (5) is achieved when $\beta = \theta_0 - w_0$ and $\hat{\theta}_0(\theta_0 - w_0) = w_0$. With the observation that

$$\sum_{i \in \mathcal{Q}} \frac{M_{ij} x_i}{\sum_{k \in \mathcal{M}} M_{ik} \hat{\theta}_k(\beta)} - \frac{\kappa}{\hat{\theta}_0(\beta)} \geq \frac{m_*}{m^*} \|x\|_\infty - \frac{\kappa}{w_0},$$

where $m^* > 0$ is the maximal element in M and $m_* > 0$ is the non-zero minimal element in M, the derivative is strictly positive when

$$\|x\|_\infty > \frac{\kappa}{w_0} \frac{m^*}{m_*}.$$

Now, recall that $u(x) = M\theta(x)$, where

$$\theta(x) \in \operatorname*{argmax}_{\theta \in \mathcal{S}_\mathcal{M}, \theta_0 \geq w_0} \sum_{i \in \mathcal{Q}} x_i \log((M\theta)_i) + \kappa \log(\theta_0).$$

Suppose for the rest of the proof that $\|x\|_\infty > C'$, then the maximization problem can equivalently be written as

$$\theta(x) \in \operatorname*{argmax}_{\theta \in \mathcal{S}_\mathcal{M}, \theta_0 = w_0} \sum_{i \in \mathcal{Q}} x_i \log((M\theta)_i). \tag{7}$$

Moreover, since λ is supposed to be strictly inside the region of stability, there exists a $\bar{\theta} \in \mathcal{S}_\mathcal{M}$ with $\bar{\theta}_0 = w_0$ such that $\lambda + \epsilon \leq M\bar{\theta}$ for some $\epsilon > 0$. For all $x \in \mathbb{R}_+^\mathcal{Q}$ we have that

$$\sum_{i \in \mathcal{Q}} x_i \log(\lambda_i + \epsilon_i) \leq \sum_{i \in \mathcal{Q}} x_i \log((M\bar{\theta})_i),$$

but also

$$\sum_{i \in \mathcal{Q}} x_i \log((M\bar{\theta})_i) \leq \sum_{i \in \mathcal{Q}} x_i \log(u_i(x)),$$

since $\bar{\theta}$ is a suboptimal solution to (7). Combining the inequalities above gives that

$$\sum_{i \in \mathcal{Q}} x_i \log(\lambda_i + \epsilon_i) \leq \sum_{i \in \mathcal{Q}} x_i \log(u_i(x)) \tag{8}$$

and hence there must exist at least one $i \in \mathcal{Q}$ such that $\lambda_i + \epsilon_i \leq u_i(x)$.

Rearranging the inequality in (8), we obtain

$$\sum_{i \in \mathcal{Q}} x_i \log \frac{u_i(x)}{\lambda_i + \epsilon_i} \geq 0. \tag{9}$$

Now, let $i^* \in \operatorname{argmax}_{i \in \mathcal{Q}} x_i$. If $u_{i^*}(x) \geq \lambda_i + \epsilon$, for some $\epsilon > 0$, then the proof is done. If not, then rewrite inequality (9) as

$$\sum_{i \in \mathcal{Q} \setminus \{i^*\}} x_i \log \frac{u_i(x)}{\lambda_i + \epsilon} + x_{i^*} \log \frac{u_{i^*}(x)}{\lambda_{i^*} + \epsilon/2} +$$
$$x_{i^*} \log \frac{\lambda_{i^*} + \epsilon/2}{\lambda_{i^*} + \epsilon} \geq 0.$$

Since

$$x_{i^*} \log \frac{u_{i^*}(x)}{\lambda_{i^*} + \epsilon} \leq 0,$$

it holds that

$$\sum_{i \in \mathcal{Q} \setminus \{i^*\}} x_i \log \frac{u_i(x)}{\lambda_i + \epsilon} \geq x_{i^*} \log \frac{\lambda_{i^*} + \epsilon}{\lambda_{i*} + \epsilon/2} \geq x_{i^*}.$$

On the other hand,

$$\sum_{i \in \mathcal{Q} \setminus \{i^*\}} x_i \log \frac{u_i(x)}{\lambda_i + \epsilon} \leq$$
$$\sum_{i \in \mathcal{Q}_1} c \log \frac{(M\mathbf{1})_i}{\lambda_i + \epsilon} + \sum_{i \in \mathcal{Q}_2 \setminus \{i^*\}} x_i \log \frac{u_i(x)}{\lambda_i + \epsilon},$$

where the inequality follows from the definition of $\mathcal{Q}_1$, and hence

$$\sum_{i \in \mathcal{Q}_2 \setminus \{i^*\}} x_i \log \frac{u_i(x)}{\lambda_i + \epsilon} \geq x_{i^*} - \sum_{i \in \mathcal{Q}_1} c \log \frac{(M\mathbf{1})_i}{\lambda_i + \epsilon}.$$

So when

$$\|x\|_\infty \geq \sum_{i \in \mathcal{Q}_1} c \log \frac{(M\mathbf{1})_i}{\lambda_i + \epsilon},$$

it must hold for at least one $i \in \mathcal{Q}_2 \setminus \{i^*\}$ that $u_i(x) > \lambda_i + \epsilon$. Hence it is possible to find a constant $C \geq C'$ such that if $\|x\|_\infty > C$, then $u_i(x) \geq \lambda_i + \epsilon$ for at least one $i \in \mathcal{Q}_2$. ■

B. Proof of Lemma 2

Proof: Introduce the function $\tilde{\theta} : \mathbb{R}_+^\mathcal{Q} \times \mathbb{R}_+ \to \mathcal{S}_\mathcal{M}$ as

$$\tilde{\theta}(x, \kappa) = \operatorname*{argmax}_{\theta \in \mathcal{S}_\mathcal{M}, \theta_0 \geq w_0} \sum_{i \in \mathcal{Q}} x_i \log((M\theta)_i) + \kappa \log(\theta_0).$$

Observe that $\tilde{\theta}(x, \kappa) = \theta(x)$ where $\theta(x)$ is given by (5). Moreover, if $\|x\| > 0$, it holds that $\tilde{\theta}(\frac{x}{\|x\|}, \frac{\kappa}{\|x\|}) = \tilde{\theta}(x, \kappa)$. Then, from the maximum theorem [17, Theorem 9.14] it follows that $\bar{\theta}(x, \kappa)$ is continuous in x and κ, i.e., for a given $\epsilon' > 0$ there exists a $\delta > 0$ such that if

$$\left\| \frac{1}{\|x\|}(x_1, x_2, \ldots, x_q, \kappa) - \frac{1}{\|x\|}(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_q, \tilde{\kappa}) \right\| < \delta,$$

then

$$\left\| \tilde{\theta}\left(\frac{x}{\|x\|}, \frac{\kappa}{\|x\|}\right) - \tilde{\theta}\left(\frac{\tilde{x}}{\|x\|}, \frac{\tilde{\kappa}}{\|x\|}\right) \right\| < \epsilon'.$$

Let $\tilde{x}_i = x_i + \epsilon_i$ and $\tilde{\kappa} = \kappa$. Hence, for every choice of ϵ', we can choose $x \in \mathbb{R}_+^\mathcal{Q}$ such that

$$\|x\| > \frac{\|(\epsilon_1, \epsilon_2, \ldots, \epsilon_q, 0)\|}{\delta},$$

which proves the lemma. ■

REFERENCES

[1] L. Kleinrock, *Queueing systems, volume 2: Computer applications*. wiley New York, 1976, vol. 66.

[2] L. Georgiadis, M. J. Neely, L. Tassiulas *et al.*, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1–144, 2006.

[3] N. Gans and G. Van Ryzin, "Optimal control of a multiclass, flexible queueing system," *Operations Research*, vol. 45, no. 5, pp. 677–693, 1997.

[4] S. T. Maguluri, R. Srikant, and L. Ying, "Stochastic models of load balancing and scheduling in cloud computing clusters," in *The Proceedings of IEEE INFOCOM 2012*. IEEE, 2012, pp. 702–710.

[5] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, 1992.

[6] M. G. Markakis, E. Modiano, and J. N. Tsitsiklis, "Max-weight scheduling in queueing networks with heavy-tailed traffic," *IEEE/ACM Transactions on Networking (TON)*, vol. 22, no. 1, pp. 257–270, 2014.

[7] P. Varaiya, "The max-pressure controller for arbitrary networks of signalized intersections," in *Advances in Dynamic Network Modeling in Complex Transportation Systems*. Springer, 2013, pp. 27–66.

[8] L. Massoulié, "Structural properties of proportional fairness: stability and insensitivity," *The Annals of Applied Probability*, pp. 809–839, 2007.

[9] N. S. Walton, "Concave switching in single and multihop networks," *SIGMETRICS Perform. Eval. Rev.*, vol. 42, no. 1, pp. 139–151, 2014.

[10] T. Le, P. Kovács, N. Walton, H. L. Vu, L. L. Andrew, and S. S. Hoogendoorn, "Decentralized signal control for urban road networks," *Transportation Research Part C: Emerging Technologies*, vol. 58, pp. 431–450, 2015.

[11] G. Nilsson, P. Hosseini, G. Como, and K. Savla, "Entropy-like lyapunov functions for the stability analysis of adaptive traffic signal controls," in *2015 IEEE 54th Annual Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 2193–2198.

[12] G. Nilsson and G. Como, "On generalized proportional allocation policies for traffic signal control," in *Preprints of the 20th World Congress, The International Federation of Automatic Control, Toulouse, France*, 2017, pp. 10 053–10 058.

[13] G. C. Buttazzo, G. Lipari, M. Caccamo, and L. Abeni, "Elastic scheduling for flexible workload management," *IEEE Transactions on Computers*, vol. 51, no. 3, pp. 289–302, 2002.

[14] R. P. Roess, E. S. Prassas, and W. R. McShane, *Traffic engineering*. Prentice Hall, 2011.

[15] A. Bastoni, B. B. Brandenburg, and J. H. Anderson, "Is semi-partitioned scheduling practical?" in *23rd Euromicro Conference on Real-Time Systems (ECRTS)*. IEEE, 2011, pp. 125–135.

[16] A. Bacciotti and A. Biglio, "Some remarks about stability of non-linear discrete-time control systems," *NoDEA: Nonlinear Differential Equations and Applications*, vol. 8, no. 4, pp. 425–438, 2001.

[17] R. K. Sundaram, *A first course in optimization theory*. Cambridge University Press, 1996.