Polynomial Approximations for Continuous Linear Programs

Dimitra Bampou and Daniel Kuhn

Department of Computing

Imperial College of Science, Technology, and Medicine

180 Queen's Gate, London SW7 2BZ, UK.

April 5, 2012

Abstract. Continuous linear programs have attracted considerable interest due to their potential for modelling manufacturing, scheduling and routing problems. While efficient simplex-type algorithms have been developed for separated continuous linear programs, crude time discretization remains the method of choice for solving general (non-separated) problem instances. In this paper we propose a more generic approximation scheme for non-separated continuous linear programs, where we approximate the functional decision variables (policies) by polynomial and piecewise polynomial decision rules. This restriction results in an upper bound on the original problem, which can be computed efficiently by solving a tractable semidefinite program. To estimate the approximation error, we also compute a lower bound by solving a dual continuous linear program in (piecewise) polynomial decision rules. We establish the convergence of the primal and dual approximations under Slater-type constraint qualifications. We also highlight the potential of our method for optimizing large-scale multiclass queueing systems and dynamic Leontief models.

Key words. Continuous linear programming, polynomial decision rules, conic programming

1 Introduction

We consider continuous linear programming problems of the form

$$\begin{array}{lll} \text{CLP:} & \text{minimize} & \displaystyle\int_0^1 c(t)^\top x(t)\mathrm{d}t \\ & \text{subject to} & \left.\begin{array}{l} G(t)x(t) + \displaystyle\int_0^t H(t,r)x(r)\mathrm{d}r \geq b(t) \\ x(t) \geq 0 \end{array}\right\} \text{a.e.,} \end{array}$$

where the cost rate $c(t)$, the right hand side vector $b(t)$ as well as the matrices $G(t)$ and $H(t,r)$ are bounded measurable functions of time. The abbreviation 'a.e.' indicates that the constraints are required to hold for almost every $t \in [0,1]$. The optimization variables $x(t)$ represent vector-valued functions of time which are referred to as *policies* or *decision rules*. Problems of the type CLP were first studied by Bellman [5] to model economic processes, but they have also applications in various areas of engineering and management science such as manufacturing [4], communication and transportation [8], queueing theory [19], revenue management [6], etc.

CLP is perceived to be a very hard problem. Analytical methods based on continuous-time dynamic programming can only be used to solve small and well-structured instances. The most common numerical solution technique, which was pioneered by Buie and Abrham [7], is to convert CLP to a finite-dimensional linear program through time discretization. This approach was later refined by Pullan [23], Philpott and Craddock [22] and Luo and Bertsimas [16] for the subclass of *separated* continuous linear programs, which are representable as

$$\begin{array}{lll} \text{SCLP:} & \text{minimize} & \displaystyle\int_0^1 c(t)^\top x(t)\mathrm{d}t \\ & \text{subject to} & \left.\begin{array}{l} \displaystyle\int_0^t H(t,r)x(r)\mathrm{d}r \geq b(t) \\ G(t)x(t) \geq 0 \end{array}\right\} \text{a.e.,} \end{array}$$

see Anderson [2]. We remark that SCLP has a more benign structure than CLP as it separates integral and pointwise constraints. If the problem data is piecewise constant/linear, SCLP can be shown to admit a piecewise constant solution, see Anderson et al. [3]. This desirable property is heavily exploited in most existing time discretization schemes and has also motivated attempts to develop simplex-type algorithms for SCLP, see Lehman [14], Segers [24], Hartberger [10], Perold [21] and Weiss [28].

Despite the superior modeling power of CLP over SCLP, research on *non*-separated continuous linear programs has stagnated over the last decades. Due to a lack of structural results about their optimal solutions, time discretization remains the state-of-the-art method to solve generic instances of CLP. In this paper we propose a new solution technique that is *not* based on time discretization. Instead, we approximate the policies in CLP by polynomial and piecewise polynomial decision rules. This restriction results in a conservative approximation for CLP and an upper bound on its optimal value. The aproximation accuracy is controlled by a single parameter, that is, the degree of the polynomial decision rules. We remark that this approach is quite natural since arbitrary polynomials can emerge as solutions to CLP even if the problem data is time-independent.

Example 1.1. *Let $a_0, a_1, \ldots, a_d$ be a sequence of real numbers, and consider the optimization problem*

$$\begin{array}{ll} \text{minimize} & \displaystyle\int_0^1 x_1(t)\,\mathrm{d}t \\ \text{subject to} & \left.\begin{array}{ll} x_i(t) \geq \displaystyle\int_0^t x_{i+1}(r)\,\mathrm{d}r + a_{i-1}(i-1)! & i = 1, \ldots, d \\ x_{d+1}(t) \geq a_d\, d! & \end{array}\right\} \text{a.e.,} \end{array}$$

which can be recognized as an instance of CLP. It is easily seen that all inequalities are binding and that $x_1(t) = a_0 + a_1 t + \cdots + a_d t^d$ at optimality. Note that it is also possible to construct instances of CLP for which piecewise polynomial or piecewise exponential functions are optimal by suitably generalizing this example.

A key feature of the polynomial decision rule approach advocated in this paper is that the best polynomial policy of a given fixed degree can be found efficiently by solving a tractable conic optimization problem. Indeed, restricting the policies in CLP to polynomials yields a semi-infinite optimization problem with polynomial inequality constraints. By using sums-of-squares techniques due to Nesterov [20], we will show the equivalence of this problem to a semidefinite program that can be solved in polynomial time.

Another attractive property of our approach is that the approximation quality can be measured reliably and efficiently. By solving the dual of CLP in (piecewise) polynomial decision rules, we obtain a lower bound on the true optimal value. The gap between the upper and lower bounds associated with the primal and dual approximations, respectively, estimates the degree of suboptimality of the best polynomial policy. This optimality gap can be computed efficiently since the upper and lower bounds are equal to the optimal values of two tractable conic optimization problems.

We remark that the polynomial decision rule approach may also be useful to solve instances of SCLP with piecewise constant/linear data. Indeed, the complexity of SCLP is unknown, but there is evidence suggesting that the number of breakpoints of the optimal solutions can grow exponentially [8]. For large instances of SCLP it may therefore be unreasonable to search for an exact solution. In this case, the best (piecewise) polynomial policy (with a moderate number of breakpoints) may incur an acceptable loss of optimality at an affordable computational cost.

Finally, we emphasize that our approach is asymptotically consistent; we will demonstrate that the approximation error can be driven to zero by increasing the degree of the (piecewise) polynomial decision rules (and/or by partitioning the planning horizon).

In summary, the main contributions of this paper are:

- We propose an approximation scheme for generic continuous linear programs with (piecewise) polynomial data. Specifically, we restrict the functional form of the policies in CLP to (piecewise) polynomials of a fixed degree and demonstrate that the arising approximate problem is equivalent to a tractable semidefinite program, which can be solved efficiently.

- By applying our approximation not only to CLP but also to its dual, we obtain upper and lower bounds on the minimum of CLP, respectively. The gap between the bounds quantifies the degree of suboptimality of the best polynomial policy, and the trade-off between the precision and complexity of our approximation is controlled by the degree of the polynomial policies.

- We establish the convergence of the primal and dual approximations as the degree of the polynomial decision rules tends to infinity.

The rest of this paper is structured as follows. Section 2 discusses algebraic conditions ensuring the solvability of CLP, while Section 3 develops polynomial decision rule approximations for CLP and its dual, respectively, and derives tractable conic programming reformulations of the arising approximate problems. In Section 4 we demonstrate the asymptotic consistency of this approximation as the degree of the polynomials tends to infinity. A refined approximation based on piecewise polynomial decision rules is elaborated in Section 5, and Section 6 presents computational results that highlight the potential of our method for optimizing large-scale multiclass queueing systems and dynamic Leontief models.

Notation We denote by $|v|$ the Euclidean norm of a vector $v \in \mathbb{R}^n$. For any matrix $A \in \mathbb{R}^{n\times m}$, $\|A\| = \sup_{v\neq 0} |Av|/|v|$ is the operator norm of A and $\mathrm{pos}(A) = \{Av : v \geq 0\} \subseteq \mathbb{R}^n$ the cone generated by the columns of A. Moreover, I denotes the identity matrix and e stands for the vector of ones; their dimensions will always be clear from the context. We define $\mathbb{S}^n$ as the space of all symmetric $n \times n$ matrices. For $A, B \in \mathbb{S}^n$, the relation $A \succeq B$ means that $A - B$ is positive semidefinite, and $\mathrm{tr}(A)$ denotes the trace of A.

For any $p \in [1, \infty]$ we denote by $\mathcal{L}_n^p$ the space of all measurable functions $f : \mathbb{R} \to \mathbb{R}^n$ with $\|f\|_p < \infty$, where the $\mathcal{L}^p$-norm $\|\cdot\|_p$ is defined in terms of the uniform distribution on $[0, 1]$. We use the abbreviation 'a.e.' for 'almost everywhere with respect to the uniform distribution on $[0, 1]$.'

2 Formal Problem Statement

A continuous linear program is an optimization problem of the following type.

$$\begin{aligned} \text{CLP:} \quad & \text{minimize} && \int_0^1 c(t)^\top x(t) \mathrm{d}t \\ & \text{subject to} && x \in \mathcal{L}_n^\infty \\ & && \left. \begin{aligned} & Gx(t) + \int_0^t Hx(r) \mathrm{d}r \geq b(t) \\ & x(t) \geq 0 \end{aligned} \right\} \text{ a.e.} \end{aligned}$$

In the remainder, we often assume that CLP satisfies the following conditions.

(C1) $c \in \mathcal{L}_n^\infty$, $b \in \mathcal{L}_m^\infty$, $G \in \mathbb{R}^{m \times n}$ and $H \in \mathbb{R}^{m \times n}$.

(C2) $b(t) \in \text{pos}(G, -I)$ a.e. and $\text{pos}(-H) \subseteq \text{pos}(G, -I)$.

(C3) $c(t) \in \text{pos}(G^\top, I)$ a.e. and $\text{pos}(-H^\top) \subseteq \text{pos}(G^\top, I)$.

The requirement in (C1) that G and H be time-independent seems restrictive, but it could easily be relaxed at the expense of additional notation. Condition (C2) ensures feasibility of CLP, while (C3) ensures dual feasibility; see [9, Corollary 7]. We emphasize that these assumptions are very mild. Note that (C2), for instance, is trivially satisfied if $\text{pos}(G^\top, I) = \{G^\top v + w : v \geq 0, w \geq 0\} = \mathbb{R}^n$, which, by cone duality, is equivalent to $\{x : Gx \geq 0, x \geq 0\} = \{0\}$. The latter condition can always be enforced when the feasible set of CLP is bounded (by appending redundant constraints if necessary). Moreover, it is a sufficient condition for boundedness of the feasible set [26, Lemma 7]. Note that we require the constraints in CLP to hold only for *almost* every $t \in [0, 1]$. This standard convention allows us later to use existing duality results in a straightforward way [9].

Theorem 2.1. *If (C1)–(C3) hold, then CLP has an optimal solution* $x_{\rm opt} \in \mathcal{L}_n^\infty$.

Proof. See e.g. [9, Corollary 2]. We remark that if CLP is feasible, the claim can be established even if only (C1) and (C3) hold. □

3 Polynomial Decision Rules

Let $\zeta(t) = (1, t, t^2, t^3, \ldots)$ be the sequence of monomials in t and denote by $\zeta_d(t)$ the finite subsequence of the first $d+1$ elements of $\zeta(t)$. Thus, any polynomial of degree d can be represented as $v^\top \zeta_d(t) = v^\top P_d \zeta(t)$ for some coefficient vector $v \in \mathbb{R}^{d+1}$, where P_d denotes the truncation operator that maps $\zeta(t)$ to $\zeta_d(t)$. In the remainder of this section we will assume that problem CLP has polynomial data. Thus, we tighten condition (C1) as follows.

(C1') b and c are polynomials of degree $d \in \mathbb{N}$, while $G \in \mathbb{R}^{m\times n}$ and $H \in \mathbb{R}^{m\times n}$.

Assumption (C1') implies that the right hand side vector and the cost rate of CLP are representable as $b(t) = B\zeta_d(t)$ and $c(t) = C\zeta_d(t)$ for some coefficient matrices $B \in \mathbb{R}^{m\times(d+1)}$ and $C \in \mathbb{R}^{n\times(d+1)}$, respectively. Note that (C1') implies the weaker condition (C1). As polynomials are dense under the uniform norm in the space of continuous functions on $[0, 1]$, condition (C1') is still very mild. We remark that all results developed below can be extended to the case where G and H are matrix-valued polynomials of time. To keep the exposition transparent, however, we sacrifice some generality by imposing time-independence of G and H. Unless otherwise stated, assumption (C1') is assumed to hold throughout Sections 3 and 4.

3.1 Upper Bound Approximation

Finding the optimal value and/or an optimal policy for CLP is hard, see e.g. [8]. Thus, we subsequently pursue the more modest goal of finding the best polynomial policy of a given degree. By restricting the decision rules in CLP to polynomials, we obtain an upper bound on its optimal value. In this section we demonstrate that finding the best polynomial policy (which is generically suboptimal) is a computationally tractable problem.

Select first the degree $\theta \in \mathbb{N}$ of the polynomial decision rule. Note that θ is a design parameter, which allows the modeler to control the approximation quality. In the following, we focus on polynomial decision rules of the form $x(t) = X\zeta_\theta(t)$ for some $X \in \mathbb{R}^{n\times(\theta+1)}$. Substituting this expression into CLP yields

$$
\begin{aligned}
\mathrm{CLP}_\theta\colon\quad & \text{minimize} && \operatorname{tr}\left(C^\top X P_\theta M P_d^\top\right) \\
& \text{subject to} && X \in \mathbb{R}^{n\times(\theta+1)} \\
& && \left.\begin{aligned} & GXP_\theta\zeta(t) + HXP_\theta \int_0^t \zeta(r)\,\mathrm{d}r \geq BP_d\zeta(t) \\ & XP_\theta\zeta(t) \geq 0 \end{aligned}\right\} \text{ a.e.,}
\end{aligned}
$$

where $M = \int_0^1 \zeta(t)\zeta(t)^\top \mathrm{d}t$ denotes the second-order moment matrix of $\zeta(t)$ under the uniform distribution on $[0,1]$. Problem CLP_θ is obtained by reducing the feasible set of the original problem, and thus we have $\min \mathrm{CLP}_\theta \geq \min \mathrm{CLP}$. Note also that CLP_θ involves finitely many decision variables (the coefficients X of the polynomial decision rules) but infinitely many constraints parameterized by $t \in [0,1]$. Since the constraint functions in CLP_θ are continuous in t, the requirement that a constraint holds almost everywhere is equivalent to the requirement that it holds for each $t \in [0,1]$. Using the linearity of integration we further obtain

$$
\int_0^t \zeta(r)\,\mathrm{d}r = J\zeta(t)\,,
$$

where the integration operator J is defined through $J_{ij} = 1/i$ if $j = i+1$; $= 0$ otherwise, $i,j \in \mathbb{N}$. Thus, the inequality constraints in CLP_θ are equivalent to

$$
(GXP_\theta + HXP_\theta J - BP_d)\zeta(t) \geq 0, \quad XP_\theta\zeta(t) \geq 0 \quad \forall t \in [0,1].
$$

By exploiting now standard sums-of-squares techniques, we can reformulate these inequalities in terms of manifestly tractable conic constraints. This reformulation relies in the following two theorems due to Nesterov [20].

Theorem 3.1 (Nonnegative Polynomials—Even Case)**.** *Assume that $p = 2q$ for some $q \in \mathbb{N}$. Then, for any $x \in \mathbb{R}^{p+1}$ the following statements are equivalent:*

(i) $x^\top \zeta_p(t) \geq 0$ *for all* $t \in [0,1]$*;*

(ii) $\exists\, Y_1 \in \mathbb{S}^{q+1}$, $Y_2 \in \mathbb{S}^q$ *such that* $Y_1 \succeq 0$, $Y_2 \succeq 0$ *and* $x = \Lambda_1^*(Y_1) + \Lambda_2^*(Y_2)$, *where* $\Lambda_1^* : \mathbb{S}^{q+1} \to \mathbb{R}^{p+1}$ *and* $\Lambda_2^* : \mathbb{S}^q \to \mathbb{R}^{p+1}$ *are defined through*

$$\left.\begin{array}{rcl} \left[\Lambda_1^*(Y_1)\right]_r & = & \displaystyle\sum_{i+j=r+1} (Y_1)_{ij} \\ \left[\Lambda_2^*(Y_2)\right]_r & = & \displaystyle\sum_{i+j=r} (Y_2)_{ij} - \sum_{i+j=r-1} (Y_2)_{ij} \end{array}\right\} \quad r = 1\ldots,p+1.$$

Proof. To keep this paper self-contained, we repeat the short proof of this result using our notation. Define $\Lambda_1 : \mathbb{R}^{p+1} \to \mathbb{S}^{q+1}$ and $\Lambda_2 : \mathbb{R}^{p+1} \to \mathbb{S}^q$ through

$$\Lambda_1(s) = \sum_{i=1}^{2q+1} s_i \Gamma^{(q,i)} \quad \text{and} \quad \Lambda_2(s) = \sum_{i=1}^{2q-1} (s_{i+1} - s_{i+2})\, \Gamma^{(q-1,i)}\,,$$

where $\Gamma^{(q,i)} \in \mathbb{R}^{(q+1)\times(q+1)}$ is the Hankel matrix with ones on the ith antidiagonal, that is, $\Gamma_{uv}^{(q,i)} = 1$ if $u + v = i + 1$; $= 0$ otherwise. By construction, we have

$$\Lambda_1(\zeta_{2q}(t)) = \zeta_q(t)\zeta_q(t)^\top \quad \text{and} \quad \Lambda_2(\zeta_{2q}(t)) = t(1-t)\zeta_{q-1}(t)\zeta_{q-1}(t)^\top \tag{3.1}$$

for all $t \in [0,1]$. It is easy to verify that the linear operators Λ_1 and Λ_1^* as well as Λ_2 and Λ_2^* are adjoint to each other in the sense that

$$\begin{array}{rcl} \mathrm{tr}(Y_1^\top \Lambda_1(x)) & = & \Lambda_1^*(Y_1)^\top x \quad \forall\, x \in \mathbb{R}^{p+1},\, Y_1 \in \mathbb{S}^{q+1}, \\ \mathrm{tr}(Y_2^\top \Lambda_2(x)) & = & \Lambda_2^*(Y_2)^\top x \quad \forall\, x \in \mathbb{R}^{p+1},\, Y_2 \in \mathbb{S}^{q}. \end{array}$$

Assume now that there exist $Y_1 \in \mathbb{S}^{q+1}$ and $Y_2 \in \mathbb{S}^q$ such that $Y_1 \succeq 0$, $Y_2 \succeq 0$ and $x = \Lambda_1^*(Y_1) + \Lambda_2^*(Y_2)$. Thus, we have

$$\begin{array}{rcl} x^\top \zeta_p(t) & = & \Lambda_1^*(Y_1)^\top \zeta_p(t) + \Lambda_2^*(Y_2)^\top \zeta_p(t) \\ & = & \mathrm{tr}[Y_1^\top \Lambda_1(\zeta_p(t))] + \mathrm{tr}[Y_2^\top \Lambda_2(\zeta_p(t))] \\ & = & \mathrm{tr}[Y_1^\top \zeta_q(t)\zeta_q(t)^\top] + t(1-t)\mathrm{tr}[Y_2^\top \zeta_{q-1}(t)\zeta_{q-1}(t)^\top] \geq 0 \end{array}$$

for all $t \in [0,1]$. Conversely, assume that $x^\top \zeta_p(t) \geq 0$ for all $t \in [0,1]$. By the Markov-Lukács theorem [12], there exist $x_1 \in \mathbb{R}^{q+1}$ and $x_2 \in \mathbb{R}^q$ such that

$$\begin{aligned} x^\top \zeta_p(t) &= [x_1^\top \zeta_q(t)]^2 + t(1-t)[x_2^\top \zeta_{q-1}(t)]^2 \\ &= \mathrm{tr}[x_1 x_1^\top \Lambda_1(\zeta_p(t))] + \mathrm{tr}[x_2 x_2^\top \Lambda_2(\zeta_p(t))] \\ &= \Lambda_1^*(Y_1)^\top \zeta_p(t) + \Lambda_2^*(Y_2)\zeta_p(t)\,, \end{aligned}$$

where $Y_1 = x_1 x_1^\top \succeq 0$ and $Y_2 = x_2 x_2^\top \succeq 0$. This implies $x = \Lambda_1^*(Y_1) + \Lambda_2^*(Y_2)$. □

Theorem 3.2 (Nonnegative Polynomials—Odd Case)**.** *Assume that* $p = 2q+1$ *for some* $q \in \mathbb{N}$*. Then, for any* $x \in \mathbb{R}^{p+1}$ *the following statements are equivalent:*

(i) $x^\top \zeta_p(t) \geq 0$ *for all* $t \in [0,1]$*;*

(ii) $\exists\, Y_1 \in \mathbb{S}^{q+1}$, $Y_2 \in \mathbb{S}^{q+1}$ *such that* $Y_1 \succeq 0$, $Y_2 \succeq 0$ *and* $x = \Lambda_1^*(Y_1) + \Lambda_2^*(Y_2)$, *where* $\Lambda_1^* : \mathbb{S}^{q+1} \to \mathbb{R}^{p+1}$ *and* $\Lambda_2^* : \mathbb{S}^{q+1} \to \mathbb{R}^{p+1}$ *are defined through*

$$\left.\begin{aligned} [\Lambda_1^*(Y_1)]_r &= \sum_{i+j=r} (Y_1)_{ij} \\ [\Lambda_2^*(Y_2)]_r &= \sum_{i+j=r+1} (Y_2)_{ij} - \sum_{i+j=r} (Y_2)_{ij} \end{aligned}\right\} \quad r = 1\ldots,p+1.$$

Proof. The proof parallels that of Theorem 3.1 and is therefore omitted. □

For the further argumentation we define

$$\mathcal{K}_\alpha^\beta = \left\{ X \in \mathbb{R}^{\alpha\times(\beta+1)} : X\zeta_\beta(t) \geq 0\; \forall\, t \in [0,1] \right\}$$

as the cone of nonnegative univariate α-dimensional polynomials of degree β, where $\alpha, \beta \in \mathbb{N}$. By Theorems 3.1 and 3.2, the cone $\mathcal{K}_\alpha^\beta$ is computationally tractable as it is representable as the image of a 2α-fold product of semidefinite cones under a linear mapping.

Note that the infinite column matrix $GXP_\theta + HXP_\theta J - BP_d$ contains only zeros in all columns $i > \eta = \max\{d, \theta+1\}$. These zero columns can be removed by postmultiplying by $P_\eta^\top$. Thus, the polynomial inequality constraints in CLP_θ can be reexpressed as

$$(GXP_\theta + HXP_\theta J - BP_d)P_\eta^\top \in \mathcal{K}_m^\eta, \quad \text{and} \quad X \in \mathcal{K}_n^\theta.$$

In summary, the semi-infinite program CLP_θ is therefore equivalent to the following tractable conic program.

$$\begin{array}{ll} \text{minimize} & \mathrm{tr}\left(C^\top X P_\theta M P_d^\top\right) \\ \text{subject to} & X \in \mathbb{R}^{n\times(\theta+1)} \\ & (GXP_\theta + HXP_\theta J - BP_d)P_\eta^\top \in \mathcal{K}_m^\eta \\ & X \in \mathcal{K}_n^\theta \end{array} \tag{3.2}$$

Proposition 3.3. *The conic program* (3.2) *can be solved to any accuracy* ϵ *in polynomial time* $\mathcal{O}(\eta^{\frac{13}{2}}(n+m)^{\frac{7}{2}}\log\frac{1}{\epsilon})$.

Proof. Under mild assumptions, interior point methods can solve semidefinite programs of the type

$$\min_{z\in\mathbb{R}^p}\left\{d^\top z : A_0 + \sum_{i=1}^{p} z_i A_i \succeq 0\right\},$$

where $A_i \in \mathbb{S}^q$ for $i = 1,\ldots,p$, to accuracy ϵ in time $\mathcal{O}(p^2 q^{\frac{5}{2}}\log\frac{1}{\epsilon})$, see [27]. Moreover, if all matrices A_i have a block-diagonal structure with blocks $A_{ij} \in \mathbb{S}^{q_j}$, $j = 1,\ldots,J$, with $\sum_j q_j = q$, then the computational effort can be reduced to $\mathcal{O}(p^2 q^{\frac{1}{2}}\sum_j q_j^2 \log\frac{1}{\epsilon})$. By using Theorems 3.1 and 3.2, problem (3.2) can be reformulated as a standard semidefinite program involving $\mathcal{O}(\eta^2(m+n))$ variables. The underlying matrix inequality has a block-diagonal structure with $\mathcal{O}(n+m)$ blocks of dimension $\mathcal{O}(\eta)$ each and $\mathcal{O}(\eta n)$ one-dimensional blocks. Thus the problem can be solved to accuracy $\mathcal{O}(\eta^{\frac{13}{2}}(n+m)^{\frac{7}{2}}\log\frac{1}{\epsilon})$. Note that the objective function and the constraints of the semidefinite program can be constructed in time $\mathcal{O}(\eta^2 mn)$, which is dominated by the solution time. □

The main insights of this section are summarized in the following theorem.

Theorem 3.4 (Upper Bounds)**.** *If (C1') holds, then* $\inf CLP_\theta \geq \inf CLP$*, and* $\inf CLP_\theta$ *that can be computed efficiently for any fixed* $\theta \in \mathbb{N}$ *by solving the conic program* (3.2)*.*

3.2 Lower Bound Approximation

The approximation scheme proposed in Section 3.1 has the desirable property that the best polynomial policy of a given fixed degree is implementable in reality (as it is feasible in CLP). Even more importantly, this policy can be computed efficiently. A weakness of the method is that it provides no information about the degree of suboptimality of the best polynomial policy. In order to measure the loss of optimality incurred by the approximation, we now investigate the *dual* of CLP. Solving this dual problem in polynomial policies will enable us to estimate the degree of suboptimality of the best *primal* polynomial policy.

For the further argumentation, we consider the following dual problem of CLP.

$$
\begin{array}{lll}
\text{CLP}^*: & \text{maximize} & \displaystyle\int_0^1 b(t)^\top y(t)\mathrm{d}t \\
& \text{subject to} & y \in \mathcal{L}_m^\infty \\
& & \left.\begin{array}{l} G^\top y(t) + \displaystyle\int_t^1 H^\top y(r)\mathrm{d}r \leq c(t) \\ y(t) \geq 0 \end{array}\right\} \text{ a.e.}
\end{array}
$$

Theorem 3.5. *If (C1)–(C3) hold, then CLP* has an optimal solution* $y_{\mathrm{opt}} \in \mathcal{L}_m^\infty$*.*

Proof. See e.g. [9, Theorem 1]. We remark that if CLP* is feasible, the claim can be established even if only (C1) and (C2) hold. □

Theorem 3.6.

(i) If (C1) holds, then $\inf CLP \geq \sup CLP^*$ *(Weak Duality).*

(ii) If (C1)–(C3) hold, then $\min CLP = \max CLP^*$ *(Strong Duality).*

Proof. The assertions (i) and (ii) follow from Proposition 1.6 and Theorem 3.8 in [9], respectively. Note that (C1)–(C3) imply via Theorems 2.1 and 3.5 that both CLP and CLP* are indeed solvable. If CLP is feasible, the weaker relation $\min \text{CLP} = \sup \text{CLP}^*$ can be established even if only (C1) and (C3) hold. Conversely, if CLP* is feasible, $\inf \text{CLP} = \max \text{CLP}^*$ can be established even if only (C1) and (C2) hold; see Theorem 5 and Corollary 6 in [9]. □

Solving CLP* in polynomial decision rules of degree $\theta \in \mathbb{N}$ results in the following approximate problem.

$$
\begin{aligned}
\text{CLP}^*_\theta\text{:} \quad & \text{maximize} && \operatorname{tr}\left(B^\top Y P_\theta M P_d^\top\right) \\
& \text{subject to} && Y \in \mathbb{R}^{m\times(\theta+1)} \\
& && \left.\begin{aligned} & G^\top Y P_\theta \zeta(t) + H^\top Y P_\theta \int_t^1 \zeta(r)\,\mathrm{d}r \le C P_d \zeta(t) \\ & Y P_\theta \zeta(t) \ge 0 \end{aligned}\right\} \text{ a.e.}
\end{aligned}
$$

Note that CLP$^*_\theta$ is obtained by reducing the feasible set of the original dual problem, and thus we have $\sup \text{CLP}^*_\theta \le \sup \text{CLP}^*$. If the conditions for weak duality are satisfied (see Theorem 3.6), then we further have $\sup \text{CLP}^*_\theta \le \inf \text{CLP}$. It can be shown that CLP$^*_\theta$ is equivalent to the tractable conic program

$$
\begin{aligned}
& \text{maximize} && \operatorname{tr}\left(B^\top Y P_\theta M P_d^\top\right) \\
& \text{subject to} && Y \in \mathbb{R}^{m\times(\theta+1)} \\
& && (C P_d - G^\top Y P_\theta - H^\top Y P_\theta J^*) P_\eta^\top \in \mathcal{K}_n^\eta \\
& && Y \in \mathcal{K}_m^\theta,
\end{aligned}
\tag{3.3}
$$

where the adjoint integration operator J^* is defined through $\int_t^1 \zeta(r)\mathrm{d}r = J^*\zeta(t)$, that is, $J^*_{ij} = 1/i$ if $j = 1$ or $j = i + 1$; $= 0$ otherwise, $i, j \in \mathbb{N}$. Problem (3.3) is equivalent to a semidefinite program that can be solved to any accuracy ϵ in polynomial time $\mathcal{O}(\eta^{\frac{13}{2}}(n + m)^{\frac{7}{2}} \log \frac{1}{\epsilon})$.

The insights of this section culminate in the following theorem.

Theorem 3.7 (Lower Bounds)**.** *If (C1') holds, then* $\sup CLP^*_\theta \leq \sup CLP^*$*, and* $\sup CLP^*_\theta$ *can be computed efficiently for any fixed* $\theta \in \mathbb{N}$ *by solving the conic program* (3.3)*. Since weak duality holds, we further have* $\sup CLP^*_\theta \leq \inf CLP$*.*

4 Convergence

If the degree θ of the primal and dual polynomial decision rules grows, we expect the optimal values of the tractable approximate problems CLP_θ and CLP^*_θ to converge from above and below, respectively, to the optimal value of the original problem CLP. We prove this convergence result under two additional mild assumptions, which are obtained by tightening (C2) and (C3).

(C2') $\exists \varepsilon > 0$ with $B_\varepsilon(b(t)) \subseteq \text{pos}(G, -I)$ a.e. and $\text{pos}(-H) \subseteq \text{pos}(G, -I)$.

(C3') $\exists \varepsilon > 0$ with $B_\varepsilon(c(t)) \subseteq \text{pos}(G^\top, I)$ a.e. and $\text{pos}(-H^\top) \subseteq \text{pos}(G^\top, I)$.

Condition (C2') requires the closed ball of radius ε around $b(t)$ to be contained in $\text{pos}(G, -I)$ for almost all t. Thus, (C2') implies (C2). Conversely, if (C2) holds, we can enforce (C2') by slightly perturbing the function b if necessary. This is always possible since $\text{pos}(G, -I)$ is a fully-dimensional convex cone containing the non-positive orthant. Similar comments apply to condition (C3').

To prove the postulated convergence result, we first demonstrate that (C2') and (C3') imply strict feasibility of CLP and CLP^*, respectively.

Definition 4.1 (Strict Feasibility)**.** *A policy* $x \in \mathcal{L}^\infty_n$*,* $x(t) \geq 0$ *a.e., is strictly feasible in CLP if there exists* $\delta > 0$ *with*

$$Gx(t) + \int_0^t Hx(r)\mathrm{d}r \geq b(t) + \delta e \quad \textit{a.e.} \tag{4.1a}$$

Similarly, a dual policy $y \in \mathcal{L}^\infty_m$*,* $y(t) \geq 0$ *a.e., is strictly feasible in* CLP^* *if*

$$G^\top y(t) + \int_t^1 H^\top y(r)\mathrm{d}r \leq c(t) - \delta e \quad \text{a.e.} \tag{4.1b}$$

for some $\delta > 0$. Problems CLP and CLP are called strictly feasible if they admit strictly feasible policies, respectively.*

Proposition 4.2. *If (C1) holds, then (C2') implies strict feasibility of CLP, while (C3') implies strict feasibility of CLP*.*

Proof. Assume that (C1) and (C2') hold. Then, CLP is strictly feasible iff

$$\infty > \inf_{x \in \mathcal{L}_n^\infty} \{0 : Gx(t) + \int_0^t Hx(r)\mathrm{d}r \geq b(t) + \delta e \text{ a.e., } x(t) \geq 0 \text{ a.e.}\} \tag{4.2}$$

for some $\delta > 0$. Next, set $\delta = \varepsilon/\sqrt{m}$ and notice that $b(t) + \delta e \in \mathrm{pos}(G, -I)$ due to condition (C2'). Thus, the feasibility problem on the right hand side of (4.2) satisfies the conditions (C1) and (C2), which guarantee gap-free duality, see the proof of Theorem 3.6. The inequality (4.2) is therefore equivalent to

$$\begin{aligned} \infty > \sup_{y \in \mathcal{L}_m^\infty} \quad & \int_0^1 (b(t) + \delta e)^\top y(t)\mathrm{d}t \\ \text{s.t.} \quad & G^\top y(t) + \int_t^1 H^\top y(r)\mathrm{d}r \leq 0, \text{ a.e., } y(t) \geq 0 \text{ a.e.} \end{aligned} \tag{4.3}$$

By construction, the trivial policy $y(t) \equiv 0$ is feasible in (4.3). Since the feasibility problem in (4.2) satisfies (C1) and (C2), its dual in (4.3) is solvable and has a finite optimal value, see the proof of Theorem 3.5. Thus, the inequality in (4.3) is true, which implies that CLP is strictly feasible. Strict feasibility of CLP* is proved in a similar way. □

Strict feasibility will enable us to approximate the optimal (generically non-smooth and/or discontinuous) policies of CLP and CLP* by feasible polynomial policies. In order to construct these polynomial approximations, we will need the concept of a *mollifier*. We thus introduce a sequence $\{\phi_v\}_{v \in \mathbb{N}}$ of mollifier functions with the following properties [1].

(i) $\phi_1 : \mathbb{R} \to \mathbb{R}$ is continuous.

(ii) $\phi_1(t) > 0$ for $|t| < 1$ and $\phi_1(t) = 0$ for $|t| \geq 1$.

(iii) $\displaystyle\int_{\mathbb{R}} \phi_1(t)\mathrm{d}t = 1$.

(iv) $\phi_v(t) = v\phi_1(vt)$ for all $t \in \mathbb{R}$ and $v \in \mathbb{N}$.

Moreover, for any $v, w \in \mathbb{N}$ we introduce a linear mollification operator defined as

$$\mathcal{M}_v : \mathcal{L}_w^1 \to \mathcal{L}_w^1, \quad \mathcal{M}_v f(t) = \frac{\int_0^1 f(r)\phi_v(t-r)\mathrm{d}r}{\int_0^1 \phi_v(t-r)\mathrm{d}r},$$

where the dependence of $\mathcal{M}_v$ on w is notationally suppressed.

Lemma 4.3. *The mollification operators $\mathcal{M}_v$, $v \in \mathbb{N}$, satisfy:*

(i) if $f \in \mathcal{L}_w^1$, then $\mathcal{M}_v f$ is continuous on $[0,1]$;

(ii) if $f \in \mathcal{L}_w^p$ for any $1 \le p < \infty$, then $\mathcal{M}_v f$ converges to f in $\mathcal{L}_w^p$;

(iii) if f is continuous, then $\mathcal{M}_v f$ converges uniformly to f on $[0,1]$;

(iv) for any $f, g \in \mathcal{L}_w^1$ the following implication holds:

$$f(t) \ge g(t) \quad a.e. \quad \Longrightarrow \quad \mathcal{M}_v f(t) \ge \mathcal{M}_v g(t) \quad a.e.$$

Proof. Basic properties of mollifiers are established in [1]. □

Theorem 4.4 (Primal Convergence)**.** *If (C1')–(C3') hold, then for any $\epsilon > 0$ there exists a polynomial policy $x^{(0)} \in \mathcal{L}_n^\infty$ feasible in CLP such that*

$$|\varphi_{\mathrm{opt}} - \varphi(x^{(0)})| \le \epsilon,$$

where $\varphi(x) = \int_0^1 c(t)^\top x(t)\mathrm{d}t$ and $\varphi_{\mathrm{opt}} = \inf$ CLP.

Proof. The conditions (C1') and (C3') imply that CLP is solvable, while (C1') and (C2') imply that it is strictly feasible, see Theorem 2.1 and Proposition 4.2. Thus, there exists $x^{(1)} \in \mathcal{L}_n^\infty$ feasible in CLP with $\varphi(x^{(1)}) = \varphi_{\mathrm{opt}}$, and there exists $x^{(2)} \in \mathcal{L}_n^\infty$ that satisfies the strict feasibility condition (4.1a) with $\delta^{(2)} > 0$. For $\lambda \in [0,1]$ consider the convex combination

$$x_\lambda = (1-\lambda)x^{(1)} + \lambda x^{(2)}.$$

For every $\lambda \in (0,1]$, x_λ is strictly feasible in CLP. Since the function $\lambda \mapsto \varphi(x_\lambda)$ is affine and thus continuous, there exists $\lambda_0 \in (0,1]$ such that $x^{(3)} = x_{\lambda_0}$ is strictly feasible in CLP with $\delta^{(3)} = \lambda_0 \delta^{(2)}$ and satisfies

$$|\varphi(x^{(1)}) - \varphi(x^{(3)})| \leq \frac{\epsilon}{3}. \tag{4.4}$$

By using the mollification operators, we can define policies $x_v = \mathcal{M}_v x^{(3)}$. Note that x_v is continuous for each $v \in \mathbb{N}$, see Lemma 4.3(i). For the further argumentation, we define $z^{(3)}(t) = \int_0^t x^{(3)}(r)\mathrm{d}r$ and $z_v(t) = \int_0^t x_v(r)\mathrm{d}r$ for all $v \in \mathbb{N}$. By construction, we have

$$\begin{aligned} |z_v(t) - z^{(3)}(t)| &= |\int_0^t x_v(r) - x^{(3)}(r)\,\mathrm{d}r| \\ &\leq \int_0^1 |\mathcal{M}_v x^{(3)}(r) - x^{(3)}(r)|\,\mathrm{d}r \\ &= \|\mathcal{M}_v x^{(3)} - x^{(3)}\|_1 \end{aligned}$$

for all $t \in [0,1]$. Note that the last term in the above expression converges to zero as v tends to infinity, see Lemma 4.3(ii). This implies that z_v converges to $z^{(3)}$ uniformly on $[0,1]$. Continuity of $z^{(3)}$ and Lemma 4.3(iii) further imply that $\mathcal{M}_v z^{(3)}$ converges uniformly to $z^{(3)}$ on $[0,1]$. The estimate

$$\|z_v - \mathcal{M}_v z^{(3)}\|_\infty \leq \|z_v - z^{(3)}\|_\infty + \|z^{(3)} - \mathcal{M}_v z^{(3)}\|_\infty$$

thus implies that $z_v - \mathcal{M}_v z^{(3)}$ converges uniformly to zero on $[0,1]$. Select now $v_1 \in \mathbb{N}$ such that

$$\|z_v - \mathcal{M}_v z^{(3)}\|_\infty \leq \frac{\delta^{(3)}}{3\|H\|} \quad \text{and} \quad \|b - \mathcal{M}_v b\|_\infty \leq \frac{\delta^{(3)}}{3} \quad \forall\, v \geq v_1\,. \tag{4.5}$$

Note that v_1 exists by Lemma 4.3(iii) since the right hand side vector b of the constraints in CLP is a continuous polynomial. Moreover, select $v_2 \in \mathbb{N}$ such that

$$\|x_v - x^{(3)}\|_1 \leq \frac{\epsilon}{3\|c\|_\infty} \quad \forall\, v \geq v_2\,. \tag{4.6}$$

The existence of v_2 is guaranteed by Lemma 4.3(ii). Next, define $\bar{v} = \max\{v_1, v_2\}$ and set $x^{(4)} = x_{\bar{v}}$. We argue now that $x^{(4)}$ is strictly feasible in CLP with

$\delta^{(4)} = \delta^{(3)}/3 > 0$. Indeed, we have

$$\begin{aligned}
Gx^{(4)}(t) + \int_0^t Hx^{(4)}(r)\mathrm{d}r &= Gx_{v^*}(t) + Hz_{v^*}(t) \\
&\geq G\mathcal{M}_{v^*}(x^{(3)})(t) + H\mathcal{M}_{v^*}(z^{(3)})(t) - \frac{\delta^{(3)}}{3}\, e \\
&= \mathcal{M}_{v^*}(Gx^{(3)} + Hz^{(3)})(t) - \frac{\delta^{(3)}}{3}\, e \\
&\geq \mathcal{M}_{v^*}b(t) + \frac{2\delta^{(3)}}{3}\, e \\
&\geq b(t) + \delta^{(4)}\, e
\end{aligned}$$

for each $t \in [0,1]$, where the first inequality follows from the first estimate in (4.5), while the second inequality in the fourth line holds because of Lemma 4.3(iv) and the strict feasibility of $x^{(3)}$ in CLP. The last inequality follows from the second estimate in (4.5) and the definition of $\delta^{(4)}$. Moreover, we have

$$x^{(4)}(t) = \mathcal{M}_{v*}(x^{(3)})(t) \geq 0 \quad \forall t \in [0,1],$$

where the inequality follows from Proposition 4.3(iv) and the strict feasibility of $x^{(3)}$ in CLP. This establishes strict feasibility of $x^{(4)}$ in CLP. Finally, we find

$$|\varphi(x^{(3)}) - \varphi(x^{(4)})| \leq \|c\|_\infty \|x^{(3)} - x^{(4)}\|_1 \leq \frac{\epsilon}{3}\,, \tag{4.7}$$

where the first inequality holds due to Hölder's inequality, and the second inequality follows from (4.6). We have thus shown that $x^{(4)}$ represents a near-optimal and strictly feasible policy which is continuous.

Next, set

$$\eta = \min\left\{ \frac{\delta^{(4)}/2}{1 + \|H\| + \|G\|}, \frac{\epsilon/3}{\|c\|_1} \right\} > 0\,.$$

By the Stone-Weierstrass theorem [11] there exists a polynomial $x^{(0)} : [0,1] \to \mathbb{R}^n$ of unspecified degree with the property that

$$\eta \geq x^{(0)}(t) - x^{(4)}(t) \geq 0 \quad \forall\, t \in [0,1]. \tag{4.8}$$

We next show that $x^{(0)}$ is strictly feasible in CLP with $\delta^{(0)} = \delta^{(4)}/2 > 0$. To this end, we define $z^{(4)}(t) = \int_0^t x^{(4)}(r)\mathrm{d}r$ and $z^{(0)}(t) = \int_0^t x^{(0)}(r)\mathrm{d}r$ and observe that

$$|z^{(4)}(t) - z^{(0)}(t)| \leq \int_0^t |x^{(0)}(r) - x^{(4)}(r)|\,\mathrm{d}r \leq \eta \quad \forall t \in [0,1]. \tag{4.9}$$

Next, we obtain

$$\begin{aligned} Gx^{(0)}(t) + \int_0^t Hx^{(0)}(r)\mathrm{d}r &= Gx^{(0)}(t) + Hz^{(0)}(t) \\ &\geq Gx^{(4)}(t) + Hz^{(4)}(t) - \eta(\|H\| + \|G\|)e \\ &\geq b(t) + \delta^{(0)}e\,, \end{aligned}$$

where the first inequality holds due to (4.8) and (4.9), while the second inequality follows from the definition of η and the strict feasibility of $x^{(4)}$. From (4.8) it is also clear that $x^{(0)}(t) \geq x^{(4)}(t) \geq 0$, while the definition of η implies

$$|\varphi(x^{(4)}) - \varphi(x^{(0)})| \leq \|c\|_1 \|x^{(4)} - x^{(0)}\|_\infty \leq \|c\|_1 \eta \leq \frac{\epsilon}{3}\,. \tag{4.10}$$

We have thus shown that $x^{(0)}$ represents a strictly feasible and polynomial policy which is near-optimal. Indeed, the estimates (4.4), (4.7) and (4.10) imply that $|\varphi_{\mathrm{opt}} - \varphi(x^{(0)})| \leq \epsilon$. This observation completes the proof.

We remark that the theorem remains valid if only (C1') and (C2') hold while CLP has a finite optimal value (but may not be solvable). □

Theorem 4.5 (Dual Convergence)**.** *If (C1')–(C3') hold, then for any $\epsilon > 0$ there exists a polynomial policy $y^{(0)} \in \mathcal{L}_m^\infty$ feasible in CLP* such that*

$$|\varphi^*_{\mathrm{opt}} - \varphi^*(x^{(0)})| \leq \epsilon,$$

where $\varphi^*(x) = \int_0^1 b(t)^\top y(t)\mathrm{d}t$, and φ^*_{opt} denotes the optimal value of CLP.*

Proof. The proof widely parallels that of Theorem 4.4 and is therefore omitted. The theorem remains valid if only (C1') and (C3') hold while CLP* has a finite optimal value. □

Corollary 4.6. *If (C1')–(C3') hold, then $\{\inf CLP_\theta\}_{\theta \in \mathbb{N}}$ and $\{\sup CLP^*_\theta\}_{\theta\in\mathbb{N}}$ converge from above and below, respectively, to $\sup CLP^* = \inf CLP$.*

5 Piecewise Polynomial Decision Rules

It is well-known that the optimal solutions of CLP and CLP* can have kinks or even discontinuities, which are difficult to approximate with polynomials. In order to improve the approximation quality and to tighten the bounds of Section 3, one could therefore allow for more flexible decision rules with kinks and jumps. In view of our previous results, it is natural to investigate the class of piecewise polynomial policies with a finite set of preassigned breakpoints at $0 = t_0 < t_1 < \cdots < t_k = 1$. Having introduced the notation for discontinuous policies, we can relax condition (C1') to allow for piecewise polynomial right hand side vectors and cost rate functions without further complicating our exposition.

(C1") b and c are polynomials of degree $d \in \mathbb{N}$ on the interval $t \in [t_{l-1}, t_l)$ for each $l = 1, \ldots, k$, while $G \in \mathbb{R}^{m\times n}$ and $H \in \mathbb{R}^{m\times n}$.

Condition (C1") implies that there exist $B_l \in \mathbb{R}^{m\times(d+1)}$ and $C_l \in \mathbb{R}^{n\times(d+1)}$ such that $b(t) = B_l\zeta_d(t)$ and $c(t) = C_l\zeta_d(t)$ for $t \in [t_{l-1}, t_l)$, $l = 1, \ldots, k$. The more restrictive condition (C1') is recovered by setting $B_l = B_1$ and $C_l = C_1$ for all l.

Denote by CLP$_{k,\theta}$ a restriction of CLP in which we optimize only over piecewise polynomial policies of degree $\theta \in \mathbb{N}$, that is, policies of the form $x(t) = X_l\zeta_\theta(t)$ for $t \in [t_{l-1}, t_l)$, $l = 1, \ldots, k$. The matrices $X_l \in \mathbb{R}^{n\times(\theta+1)}$, $l = 1, \ldots, k$, provide a finite parameterization of the policy space. Using arguments familiar from Section 3.1 it can be shown that CLP$_{k,\theta}$ is equivalent to

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{l=1}^{k} \operatorname{tr}(C_l^\top X_l P_\theta M_l P_d^\top) \\ \text{subject to} & \left.\begin{array}{l} X_l \in \mathbb{R}^{n\times(\theta+1)} \\ (GX_lP_\theta + H[\displaystyle\sum_{q=1}^{l-1} X_qP_\theta K_q + X_lP_\theta J_l] - B_lP_d)P_\eta^\top \in \mathcal{K}_{l,m}^{\eta} \\ X_l \in \mathcal{K}_{l,n}^{\theta} \end{array}\right\} \forall l, \end{array} \tag{5.11}$$

where $\eta = \max\{\theta+1, d\}$ and $M_l = \int_{t_{l-1}}^{t_l} \zeta(t)\zeta(t)^\top \mathrm{d}t$. Here, we also use the integration operators K_l and J_l defined through $(K_l)_{ij} = (t_l^i - t_{l-1}^i)/i$ if $j=1$; $= 0$ otherwise and $(J_l)_{ij} = 1/i$ if $j = i+1$; $= -t_{l-1}^i/i$ if $j=1$; $= 0$ otherwise, $i,j \in \mathbb{N}$. They are designed to satisfy the relations

$$K_l\zeta(t) = \int_{t_{l-1}}^{t_l} \zeta(s)\mathrm{d}s \quad \text{and} \quad J_l\zeta(t) = \int_{t_{l-1}}^{t} \zeta(s)\mathrm{d}s\,.$$

The conic program (5.11) also involves generalized cones of the type

$$\mathcal{K}_{l,\alpha}^{\beta} = \left\{X \in \mathbb{R}^{\alpha\times(\beta+1)} : X\zeta_\beta(t) \geq 0 \;\forall\, t \in [t_{l-1}, t_l]\right\}$$

for $\alpha, \beta \in \mathbb{N}$ and $l = 1,\ldots,k$, which are representable as linear images of 2α-fold products of semidefinite cones; see Theorems 3.1 and 3.2. This implies that $\mathrm{CLP}_{k,\theta}$ is computationally tractable for fixed k and θ.

In analogy to the discussion above, we can introduce a restriction $\mathrm{CLP}_{l,\theta}^*$ of the dual problem CLP^* in which we optimize only over polynomial policies of the form $y(t) = Y_l\zeta_\theta(t)$ for $t \in [t_{l-1}, t_l)$, $l = 1,\ldots,k$. Here, the matrices $Y_l \in \mathbb{R}^{n\times(\theta+1)}$, $l = 1,\ldots,k$, provide a finite parameterization of the policy space. Using arguments familiar from Section 3.2 it can be shown that $\mathrm{CLP}_{k,\theta}^*$ is equivalent to

$$\begin{array}{ll} \text{maximize} & \displaystyle\sum_{l=1}^{k} \mathrm{tr}(B_l^\top Y_l P_\theta M_l P_d^\top) \\ \text{subject to} & \left.\begin{array}{l} Y_l \in \mathbb{R}^{m\times(\theta+1)} \\ (C_l P_d - G^\top Y_l P_\theta - H^\top[Y_l P_\theta J_l^* + \displaystyle\sum_{q=l+1}^{k} Y_q P_\theta K_q])P_\eta^\top \in \mathcal{K}_{l,n}^{\eta} \\ Y_l \in \mathcal{K}_{l,m}^{\theta} \end{array}\right\} \forall l. \end{array} \tag{5.12}$$

Here, the operators K_l are defined as in (5.11), while the adjoint integration operators J_l^* are defined through $(J_l^*)_{ij} = -1/i$ if $j = i+1$; $= t_l^i/i$ if $j=1$; $= 0$ otherwise, $i,j \in \mathbb{N}$. They are designed to satisfy the relations

$$J_l^*\zeta(t) = \int_{t}^{t_l} \zeta(s)\mathrm{d}s\,.$$

The above findings are summarized in the following main theorem.

Theorem 5.1. *If (C1") holds, then* $\inf CLP_{k,\theta} \geq \inf CLP$ *and* $\sup CLP^*_{k,\theta} \leq \sup CLP^*$*, where* $\inf CLP_{k,\theta}$ *and* $\sup CLP^*_{k,\theta}$ *can be computed efficiently for any fixed* $k, \theta \in \mathbb{N}$ *by solving the conic programs* (5.11) *and* (5.12)*, respectively. If strong duality holds, then we further have* $\sup CLP^*_{k,\theta} \leq \inf CLP$*.*

Remark 5.2. *Instead of using preassigned breakpoints and a uniform fixed degree for the polynomial policies on all subintervals, one could devise an adaptive algorithm that sequentially adds or removes breakpoints and increases or decreases the polynomial degrees on the subintervals with the goal to minimize the optimality gap subject to size constraints on the arising conic programs. Adaptive algorithms for the placement of breakpoints in SCLP have been suggested by Pullan [23]. Luo and Bertsimas [16] went even further by treating the length of each discretization interval as a decision variable.*

6 Numerical Examples

We illustrate the performance of the proposed solution methods on example problems from queueing theory and economics. Problems of this kind have motivated much of the research on continuous linear programming. All computations are performed within Matlab 2008b and by using the YALMIP interface [15] of the SDPT3 optimization toolkit [25].

6.1 Multiclass Queueing Networks

Consider the multiclass queueing network depicted in Figure 1, which could represent a multi-tier web system [17]. The system consists of four *front servers* (A–D) and two *back-end servers* (E and F), and it accepts eight different classes of requests that arrive in a single burst. We assume that class i requests arrive at rate $\dot{b}_i(t) = \kappa_i \max\{t - \frac{10}{3}t^2, 0\}$, where κ_i is specified in Table 1. This is a stress case for a real system since handling burstiness is a difficult unsolved problem in system management [18]. Requests processed by the front servers are either routed to the back-end servers with probability $\frac{4}{5}$ or leave the system with probability $\frac{1}{5}$ (EXIT). Upon processing, the back-end servers feed the requests back to the front servers.

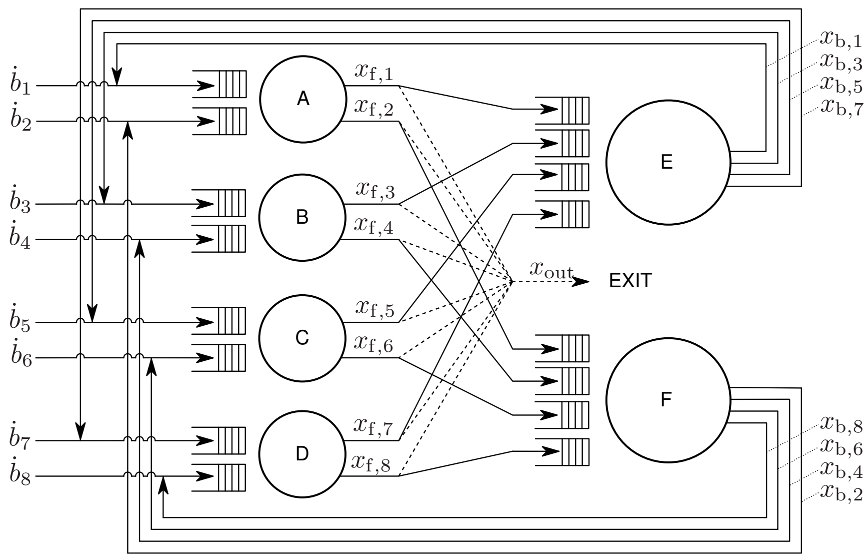

Figure 1: Multiclass queueing network

Class i requests are processed at rate $x_{\text{f},i}(t)$ by the front servers and at rate $x_{\text{b},i}(t)$ by the back-end servers. Moreover, they arrive at rate $\dot{b}_i(t) + x_{\text{b},i}(t)$ at the front servers and at rate $\frac{4}{5}\,x_{\text{f},i}(t)$ at the back-end servers. Thus, at time t there is a queue of $n_{\text{f},i}(t) = b_i(t) + \int_0^t x_{\text{b},i}(r) - x_{\text{f},i}(r)\text{d}r$ class i requests waiting to be processed by the front servers, where $b_i(t) = \int_0^t \dot{b}_i(r)\text{d}r$. Similarly, $n_{\text{b},i}(t) = \int_0^t \frac{4}{5}\,x_{\text{f},i}(r) - x_{\text{b},i}(r)\text{d}r$ class i requests await processing by the back-end servers. All queues are assumed to have infinite buffer capacities.

Denote by τ_i the time required to process a class i request on a single processor, see Table 1. If a front server is not fully loaded (e.g., it has idle processors), its output rate for class i requests is therefore bounded above by $n_{\text{f},i}(t)/\tau_i$. This implies $\tau_i x_{\text{f},i}(t) \leq n_{\text{f},i}(t)$. Similarly, we have $\tau_i x_{\text{b},i}(t) \leq n_{\text{b},i}(t)$ for partially loaded back-end servers. Assume now that each front (back-end) server accommodates $N_\text{f} = 25$ ($N_\text{b} = 41$) processors. Thus, the total number of requests processed

i	1	2	3	4	5	6	7	8
κ_i	546	378	504	247	561	522	503	571
τ_i	0.0800	0.0512	0.1120	0.0704	0.0320	0.0960	0.0720	0.0640

Table 1: Parameters of request classes

in parallel by front server A, $\tau_1 x_{\mathrm{f},1}(t) + \tau_2 x_{\mathrm{f},2}(t)$, may never exceed N_{f}. Similar constraints hold for all other servers in the system. Completed requests flow out of the system at rate $x_{\mathrm{out}}(t) = \frac{1}{5}\sum_{i=1}^{8} x_{\mathrm{f},i}(t)$. We choose the servers' processing rates in order to minimize the average backlog of the system over a planning horizon $T = 2.2$, that is, we seek to minimize $\frac{1}{T}\int_0^T \sum_{i=1}^{8} n_{\mathrm{f},i}(t) + n_{\mathrm{b},i}(t)\mathrm{d}t$.

The control problem outlined above can be viewed as an instance of CLP with piecewise polynomial data. Note that the problem cannot be reformulated as an instance of SCLP because the output rate constraints for partially loaded servers couple the output rates with the queue lengths. We obtain upper and lower bounds on its optimal value by solving the conic programs $\mathrm{CLP}_{k,\theta}$ and $\mathrm{CLP}^*_{k,\theta}$, respectively, for different values of k and θ. One breakpoint is always placed at $t = \frac{3}{10}$, where the $\dot{b}_i(t)$ have a kink. All other $k-1$ breakpoints are equally spaced on the interval $[0, T]$. Figure 2 reports the optimality gaps

$$\Delta_{k,\theta} = 2 \times \frac{\inf \mathrm{CLP}_{k,\theta} - \sup \mathrm{CLP}^*_{k,\theta}}{\inf \mathrm{CLP}_{k,\theta} + \sup \mathrm{CLP}^*_{k,\theta}}$$

and the CPU times $\tau_{k,\theta}$ for solving both $\mathrm{CLP}_{k,\theta}$ and $\mathrm{CLP}^*_{k,\theta}$. First we notice that polynomial policies of degree 4 and 5 *without* breakpoints achieve a smaller optimality gap than piecewise constant policies with 20 breakpoints. Moreover, the polynomial policies can be computed about 3 times faster. To showcase the merits of using piecewise polynomial policies, we determine for each time budget τ the smallest optimality gap $\Delta(\tau) = \min\{\Delta_{k,\theta} : \tau_{k,\theta} \le \tau\}$ that can be computed in time less than τ. Similarly, we determine the smallest optimality gap $\Delta_0(\tau) = \min\{\Delta_{k,0} : \tau_{k,0} \le \tau\}$ achievable with *piecewise constant* policies only. Figure 2 shows that $\Delta(\tau)$ (solid line) is significantly smaller than $\Delta_0(\tau)$ (dashed line) for $\tau \gtrsim 10\,\text{s}$, while smaller time budgets lead to unacceptably high optimality gaps. Thus, for a given time budget (target optimality gap), the polynomial decision rule approach achieves superior accuracy (shorter computation time) than a naive time-discretization approach.

We emphasize that even if the arrival rates $\dot{b}_i(t)$ were time independent, most algorithms for *separated* continuous linear programs could not be applied to the example problem at hand since the capacity constraints couple the output rates with the queue lengths.

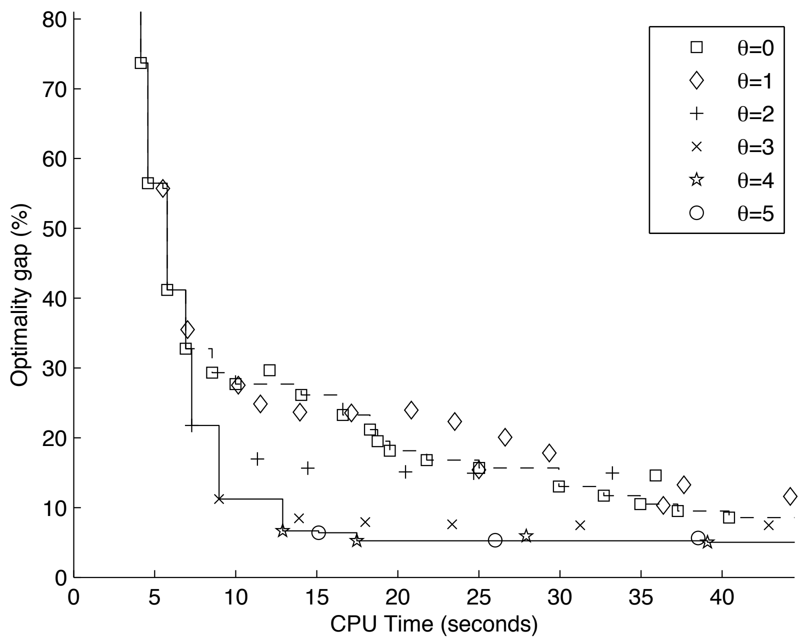

Figure 2: Queueing model: Trade-off between CPU time and optimality gap

6.2 Continuous-Time Leontief Models

Consider a closed economy consisting of n industries, each producing a different homogeneous good $i = 1, \ldots, n$. We denote by $A \in \mathbb{R}^{n\times n}$ the economy's consumption matrix, where A_{ij} represents the input amount of good i consumed by the jth industry in order to produce a unit output of good j. Thus, in order to sustain the output rate $x(t) \in \mathbb{R}^n$ the economy requires an input rate $Ax(t)$, implying that the net production rate amounts to $(I - A)x(t)$. We require that there should be no disinvestment of stocks of goods in the production process at any time, that is, $(I - A)x(t) \geq 0$ a.e. If $\alpha \in \mathbb{R}^n$ denotes the initial stock then the stock of goods accumulated by time t is $\int_0^t (I - A)x(s)\mathrm{d}s + \alpha$. Let $B \in \mathbb{R}^{n\times n}$ be the matrix of capital coefficients, where B_{ij} defines the stock of good i required per unit of capacity of industry j. Thus, we impose the production capacity constraint $Bx(t) \leq \int_0^t (I - A)x(s)\mathrm{d}s + \alpha$ a.e. By definition, A and B have nonnegative entries. Additionally, the consumption matrix A must be productive, that is, in order to produce a unit of good i, no more than one unit of good i should be required in the corresponding production process. It can be shown that this requirement is satisfied iff $I - A$ has a nonnegative inverse. Therefore, the constraint $(I - A)x(t) \geq 0$ a.e. implies that $x(t) \geq 0$ a.e.

The objective is to maximize the total value of all goods produced in the economy within a period $[0, T]$, where $c \in \mathbb{R}^n$ represents the vector of values of the different goods. Thus, we aim at solving the following continuous linear program with constant data, which was originally discussed in [26].

$$
\begin{array}{ll}
\text{maximize} & \displaystyle\int_0^T c^\top (I - A)x(t)\,\mathrm{d}t \\
\text{subject to} & x \in \mathcal{L}_n^\infty \\
& \left.\begin{array}{l} Bx(t) \leq \displaystyle\int_0^t (I - A)x(s)\mathrm{d}s + \alpha \\ (I - A)x(t) \geq 0 \end{array}\right\} \text{a.e.}
\end{array}
\tag{6.1}
$$

We solve an instance of (6.1) with $n = 25$ industries and a planning horizon of $T = 30$ years. The input data of the problem is provided in Tables 2 and 3. The consumption matrix A and the initial stock α are based on aggregate data for the US economy [13, § 2.6]. We obtain upper and lower bounds on the optimal value of problem (6.1) by solving the associated conic programs $\mathrm{CLP}_{k,\theta}$ and $\mathrm{CLP}^*_{k,\theta}$, respectively, for different values of k and θ. As usual, the breakpoints are equally spaced within the interval $[0, T]$, and the optimality gaps $\Delta_{k,\theta}$ are computed as in

Section 6.1. We remark that the approximations corresponding to $k = 0$ coincide with the approximations based on time discretization described in [26]; to the best of our knowledge these are the only existing approximations applicable to (non-separated) problems of the type (6.1).

The best piecewise constant policy found within 200 s has $k = 39$ uniform breakpoints and achieves a disappointing optimality gap of 46%. In contrast, the best piecewise polynomial policy found within the same time frame corresponds to $k = 10$ and $\theta = 3$ and achieves an excellent optimality gap of 1%. Figure 3 illustrates the trade-off between accuracy and complexity of the various approximations. As usual, the solid line traces out the best gap for a given time budget over all policies, while the dashed line traces out the best gap achieved with piecewise constant policies only. Even though the optimality gaps for the piecewise constant approximations ($\theta = 0$) are known to converge to zero as k tends to infinity, the convergence is much slower than for the piecewise polynomial approximations ($\theta > 0$). The polynomial decision rules outperform the piecewise constant decision rules more clearly than in the example of Section 6.1 because the optimal policies of the Leontief model are highly nonlinear while those for the queueing model tend to be constant on a sizeable portion of the planning period.

Acknowledgements. We are indebted to Giuliano Casale and Steve Zymler for valuable discussions, and we thank EPSRC for financial support under grant EP/H0204554/1.

References

[1] Adams, A. *Sobolev Spaces.* Academic Press, New York, 1975.

[2] Anderson, E. A new continuous model for job-shop scheduling. *International Journal of Systems Science 12*, 12 (1981), 1469–1475.

A																									α
.245	.390	.094	.021	.001	.040	.013	.000	.001	.000	.000	.000	.000	.000	.000	.000	.001	.000	.000	.000	.000	.004	.057	.003	.008	12.800
.054	.143	.003	.000	.002	.026	.116	.000	.000	.000	.000	.000	.000	.000	.004	.000	.005	.002	.003	.000	.000	.008	.227	.000	.019	26.440
.002	.005	.308	.034	.006	.016	.024	.006	.000	.001	.001	.006	.018	.005	.019	.008	.001	.001	.000	.000	.000	.002	.003	.002	.031	13.260
.003	.003	.001	.168	.020	.002	.005	.004	.003	.006	.007	.018	.006	.013	.009	.007	.001	.001	.000	.006	.000	.000	.008	.089	.026	2.650
.000	.014	.004	.002	.310	.016	.013	.037	.000	.010	.003	.010	.002	.005	.022	.000	.003	.016	.010	.005	.480	.010	.051	.006	.046	2.400
.032	.039	.043	.020	.024	.281	.047	.037	.058	.015	.016	.030	.045	.010	.039	.058	.045	.008	.000	.019	.002	.020	.006	.046	.096	7.620
.000	.000	.002	.001	.000	.000	.273	.000	.000	.000	.001	.000	.001	.000	.003	.000	.000	.000	.000	.000	.000	.001	.001	.000	.013	2.270
.001	.006	.000	.005	.002	.011	.003	.089	.011	.007	.005	.014	.013	.003	.013	.002	.001	.001	.000	.000	.000	.001	.004	.061	.017	0.680
.000	.000	.000	.014	.001	.007	.000	.008	.369	.243	.133	.125	.090	.108	.039	.005	.010	.000	.000	.000	.000	.000	.010	.042	.057	0.970
.002	.015	.000	.020	.001	.007	.005	.000	.003	.041	.041	.041	.068	.025	.016	.000	.002	.001	.000	.000	.000	.001	.005	.109	.067	1.880
.001	.000	.003	.002	.004	.000	.000	.002	.004	.027	.076	.020	.044	.055	.004	.003	.003	.000	.000	.001	.000	.005	.005	.018	.104	9.350
.000	.000	.000	.000	.000	.000	.000	.002	.003	.023	.038	.103	.043	.030	.007	.002	.002	.000	.016	.000	.002	.003	.003	.027	.059	3.540
.003	.000	.000	.000	.000	.000	.000	.000	.000	.003	.002	.001	.308	.000	.000	.001	.006	.001	.000	.000	.000	.034	.005	.001	.031	7.680
.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.001	.075	.000	.000	.011	.000	.000	.000	.000	.000	.001	.000	.021	2.960
.000	.000	.011	.002	.004	.001	.005	.002	.000	.006	.006	.014	.006	.005	.057	.000	.000	.000	.000	.000	.031	.016	.009	.002	.043	3.880
.001	.005	.007	.005	.011	.026	.005	.041	.019	.008	.007	.006	.004	.008	.006	.138	.024	.012	.003	.077	.000	.017	.014	.001	.001	0.580
.024	.027	.012	.043	.032	.043	.016	.048	.041	.015	.013	.012	.021	.013	.010	.020	.053	.009	.000	.013	.012	.004	.026	.045	.022	1.950
.031	.011	.026	.014	.015	.008	.016	.010	.019	.019	.017	.017	.004	.018	.013	.005	.021	.005	.003	.019	.028	.022	.075	.089	.047	3.410
.000	.001	.001	.002	.004	.001	.000	.002	.001	.002	.002	.002	.001	.003	.003	.002	.003	.008	.019	.004	.084	.007	.001	.001	.004	1.460
.059	.006	.007	.016	.009	.007	.008	.015	.006	.007	.006	.008	.003	.010	.010	.011	.028	.071	.016	.068	.016	.050	.030	.017	.000	28.470
.000	.016	.007	.009	.006	.016	.013	.002	.002	.005	.006	.007	.006	.003	.016	.001	.002	.041	.028	.004	.012	.008	.004	.005	.019	0.220
.008	.003	.000	.005	.001	.001	.000	.006	.001	.001	.001	.000	.000	.000	.000	.002	.017	.034	.006	.004	.014	.038	.017	.029	.062	24.350
.000	.000	.001	.000	.017	.001	.000	.002	.059	.002	.003	.000	.000	.000	.000	.000	.002	.009	.003	.003	.004	.005	.000	.000	.001	13.140
.005	.003	.003	.002	.004	.003	.005	.006	.005	.003	.003	.002	.003	.005	.004	.029	.057	.004	.057	.099	.000	.014	.005	.000	.000	21.110
.000	.046	.060	.113	.047	.079	.071	.097	.017	.110	.112	.106	.029	.085	.154	.027	.008	.062	.003	.026	.061	.074	.038	.015	.000	0.000

Table 2: Consumption Matrix and Initial Stock

B																									c
.818	.310	.320	.853	.594	.924	.454	.786	.908	.086	.374	.601	.864	.869	.586	.412	.232	.287	.072	.698	.990	.802	.008	.095	.218	.540
.877	.787	.391	.846	.964	.494	.499	.295	.177	.793	.333	.190	.127	.980	.160	.354	.586	.985	.331	.064	.359	.893	.348	.737	.848	.755
.794	.719	.992	.002	.176	.152	.856	.855	.058	.518	.941	.244	.641	.263	.354	.750	.989	.649	.323	.550	.797	.808	.787	.659	.537	.797
.304	.344	.324	.503	.820	.256	.249	.578	.695	.352	.167	.826	.393	.758	.168	.563	.346	.139	.617	.478	.599	.738	.177	.615	.744	.402
.841	.262	.695	.236	.093	.091	.442	.680	.370	.760	.799	.195	.445	.993	.889	.568	.246	.143	.886	.223	.583	.925	.807	.130	.018	.929
.140	.439	.491	.693	.140	.956	.707	.962	.719	.619	.489	.020	.481	.637	.072	.808	.861	.861	.304	.263	.291	.217	.512	.701	.644	.602
.821	.712	.800	.937	.156	.210	.494	.235	.060	.161	.412	.723	.571	.707	.907	.981	.916	.252	.921	.583	.537	.235	.595	.742	.480	.791
.629	.466	.201	.722	.294	.782	.189	.982	.791	.048	.531	.634	.655	.244	.103	.033	.410	.695	.688	.600	.106	.022	.375	.708	.541	.134
.173	.131	.656	.028	.447	.408	.570	.440	.660	.364	.964	.033	.419	.018	.100	.181	.357	.063	.254	.984	.170	.491	.670	.661	.448	.981
.268	.162	.493	.314	.012	.364	.486	.443	.210	.541	.852	.176	.932	.358	.276	.404	.676	.554	.846	.022	.134	.297	.723	.951	.139	.105
.845	.274	.121	.865	.930	.485	.708	.664	.241	.690	.326	.724	.618	.141	.426	.677	.163	.487	.680	.462	.860	.148	.523	.666	.032	.285
.842	.264	.491	.732	.733	.721	.918	.410	.159	.324	.961	.887	.325	.274	.109	.136	.016	.692	.838	.118	.372	.218	.107	.343	.969	.057
.846	.423	.030	.623	.218	.090	.019	.393	.527	.823	.761	.263	.975	.908	.256	.105	.187	.240	.940	.539	.038	.919	.782	.594	.435	.313
.122	.006	.058	.621	.891	.865	.592	.290	.974	.981	.772	.477	.840	.065	.997	.195	.592	.903	.781	.439	.259	.386	.164	.895	.392	.962
.259	.481	.507	.568	.892	.651	.306	.898	.710	.826	.778	.478	.215	.265	.660	.340	.455	.867	.389	.231	.997	.767	.032	.710	.006	.407
.722	.142	.366	.767	.229	.975	.275	.019	.331	.526	.731	.939	.643	.101	.059	.006	.087	.573	.944	.774	.381	.866	.738	.711	.726	.555
.011	.773	.486	.739	.074	.600	.446	.154	.424	.146	.801	.744	.025	.566	.109	.869	.950	.819	.835	.975	.316	.852	.171	.129	.088	.719
.150	.672	.156	.106	.299	.940	.126	.339	.631	.434	.873	.336	.978	.955	.560	.989	.883	.303	.556	.317	.972	.751	.249	.929	.032	.811
.811	.972	.010	.216	.466	.094	.902	.938	.937	.974	.154	.823	.156	.669	.736	.732	.575	.727	.129	.798	.754	.886	.850	.436	.300	.763
.345	.404	.328	.287	.375	.527	.661	.724	.642	.133	.297	.042	.410	.617	.439	.077	.711	.207	.163	.793	.940	.416	.869	.822	.636	.148
.585	.366	.107	.005	.067	.664	.994	.879	.953	.910	.179	.602	.949	.821	.233	.216	.618	.210	.762	.481	.482	.513	.940	.175	.562	.591
.240	.572	.015	.001	.907	.073	.979	.973	.507	.052	.490	.800	.648	.046	.930	.150	.819	.657	.525	.929	.236	.845	.277	.027	.263	.095
.738	.862	.876	.535	.688	.229	.533	.127	.884	.748	.697	.893	.224	.566	.816	.647	.928	.669	.945	.828	.284	.717	.042	.107	.672	.876
.593	.429	.665	.918	.660	.682	.666	.643	.471	.676	.869	.061	.852	.270	.096	.414	.919	.984	.605	.564	.448	.052	.417	.581	.590	.868
.942	.658	.186	.048	.504	.193	.525	.804	.890	.359	.207	.562	.157	.394	.860	.960	.439	.146	.534	.389	.407	.391	.112	.329	.143	.563

Table 3: Capital Coefficients and Values of Goods

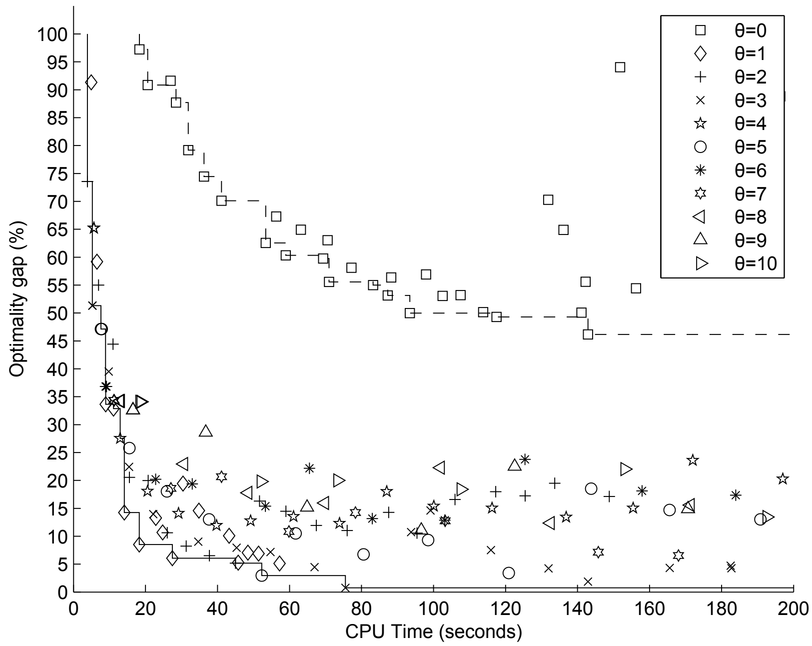

Figure 3: Leontief model: Trade-off between CPU time and optimality gap

[3] Anderson, E., Nash, P., and Perold, A. Some properties of a class of continuous linear programs. *SIAM Journal on Control and Optimization 21*, 5 (1983), 758–765.

[4] Bai, S., and Gershwin, S. Scheduling manufacturing systems with work-in-process inventory. In *Proceedings of the 29th IEEE Conference on Decision and Control* (Honolulu, HI, 1991), pp. 557–564.

[5] Bellman, R. Bottleneck problems and dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America 39*, 9 (1953), 947–951.

[6] Boyd, A., and Karaesmen, I. An abstraction of revenue management for the air cargo industry. Working paper, Carnegie Mellon University, Pittsburgh, 2001.

[7] Buie, R., and Abrham, J. Numerical solutions to continuous linear programming problems. *Mathematical Methods of Operations Research 17*, 3 (1973), 107–117.

[8] Fleischer, L., and Sethuraman, J. Efficient algorithms for seperated continuous linear programs: The multicommodity flow problem with holding costs and extensions. *Mathematics of Operations Research 30*, 4 (2005), 916–938.

[9] Grinold, R. Continuous programming part one: Linear objectives. *Journal of Mathematical Analysis and Applications 28* (1969), 32–51.

[10] Hartberger, R. Representation extended to continuous time. In *Optimization methods for resource allocation (Proc. NATO Conf., Elsinore, 1971)*. English Univ. Press, London, 1974, pp. 297–307.

[11] Kantorovich, L., and Akilov, G. P. *Functional Analysis*. Pergamon Press, 1982.

[12] Krein, M., and Nudelman, A. *The Markov Moment Problem and Extremal Problems*, vol. 50 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, Rhode Island, 1977.

[13] Lay, D. *Linear Algebra and its Applications*, 4th ed. Addison-Wesley, Reading, MA, 2011.

[14] Lehman, R. On the continuous simplex method. *Technical Report RP-1386, Rand Corporations* (1954).

[15] Löfberg, J. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference* (Taipei, Taiwan, 2004).

[16] Luo, X., and Bertsimas, D. A new algorithm of state-constrained separated continuous linear programs. *SIAM Journal on Control and Optimization 37*, 1 (1998), 177–210.

[17] Mi, N., Casale, G., Cherkasova, L., and Smirni, E. Burstiness in multi-tier applications: Symptoms, causes, and new models. In *Proceedings of the ACM/IFIP/USENIX 9th International Middleware Conference* (Berlin, Heidelberg, 2008), Springer-Verlag, pp. 265–286.

[18] Mi, N., Casale, G., Cherkasova, L., and Smirni, E. Injecting realistic burstiness to a traditional client-server benchmark. In *Proceedings of the 6th International Conference on Autonomic Computing* (New York, NY, USA, 2009), ICAC '09, ACM, pp. 149–158.

[19] Nazarathy, Y., and Weiss, G. Near optimal control of queueing networks over a finite time horizon. *Annals of Operations Research 170*, 1 (2009), 233–249.

[20] Nesterov, Y. Squared functional systems and optimization problems. In *High Performance Optimization*, J. Frenk, C. Roos, T. Terlaky, and S. Zhang, Eds. Kluwer Academic Publishers, 2000, pp. 405–440.

[21] Perold, A. Extreme points and basic feasible solutions in continuous time linear programming. *SIAM Journal on Control and Optimization 19*, 1 (1981), 52–63.

[22] Philpott, A., and Craddock, M. An adaptive discretization algorithm for a class of continuous network programs. *Networks 26*, 1 (1995), 1–11.

[23] Pullan, M. An algorithm for a class of continuous linear programs. *SIAM Journal on Control and Optimization 31*, 6 (1993), 1558–1577.

[24] Segers, R. A generalised function setting for dynamic optimal control problems. In *Optimization methods for resource allocation (Proc. NATO Conf., Elsinore, 1971)*. English Univs. Press, London, 1974, pp. 279–296.

[25] Tütüncü, R., Toh, K., and Todd, M. Solving semidefinite-quadratic-linear programs using SDPT3. *Mathematical Programming 95*, 2 (2003), 189–217.

[26] Tyndall, W. A duality theorem for a class of continuous linear programming problems. *Journal of the Society for Industrial and Applied Mathematics 13*, 3 (1965), 644–666.

[27] Vandenberghe, L., and Boyd, S. Semidefinite programming. *SIAM Review 38*, 1 (1996), 49–95.

[28] Weiss, G. A simplex based algorithm to solve sepearated continuous linear programs. *Mathematical Programming 115*, 1 (2002), 151–198.