RESTLESS BANDIT MARGINAL PRODUCTIVITY INDICES II: MULTI-PROJECT CASE AND SCHEDULING A MULTICLASS MAKE-TO-ORDER/-STOCK $M/G/1$ QUEUE

José Niño-Mora*

Abstract

This paper develops a framework based on convex optimization and economic ideas to formulate and solve approximately a rich class of dynamic and stochastic resource allocation problems, fitting in a generic discrete-state multi-project restless bandit problem (RBP). It draws on the single-project framework in the author´s companion paper "Restless bandit marginal productivity indices I: Single-project case and optimal control of a make-to-stock $M/G/1$ queue", based on characterization of a project´s marginal productivity index (MPI). Our framework significantly expands the scope of Whittle (1988)´s seminal approach to the RBP. Contributions include: (i) Formulation of a generic multi-project RBP, and algorithmic solution via single-project MPIs of a relaxed problem, giving a lower bound on optimal cost performance; (ii) a heuristic MPI-based hedging point and index policy; (iii) application of the MPI policy and bound to the problem of dynamic scheduling for a multiclass combined MTO/MTS $M/G/1$ queue with convex backorder and stock holding cost rates, under the LRA criterion; and (iv) results of a computational study on the MPI bound and policy, showing the latter´s near-optimality across the cases investigated.

Keywords: Stochastic scheduling, restless bandit, index policy, hedging point, Markov decision process, resource allocation, diminishing returns, marginal productivity, shadow wage, efficient frontier, convex optimization, multiclass queue, make-to-order, make-to-stock, multi-product production-inventory control, relaxation, conservation laws, performance bound, achievable performance.

*Niño-Mora, Departamento de Estadística, Universidad Carlos III de Madrid, C/ Madrid 126, 28903 Getafe (Madrid), e-mail: jose.nino@uc3m.es. Supported in part a Ramón y Cajal Research Fellowship of the Spanish Ministry of Science and Technology (MCyT).

RESTLESS BANDIT MARGINAL PRODUCTIVITY INDICES II: MULTI-PROJECT CASE AND SCHEDULING A MULTICLASS MAKE-TO-ORDER/-STOCK $M/G/1$ QUEUE *

J. NIÑO-MORA **

Department of Statistics
Universidad Carlos III de Madrid
C/ Madrid, 126
28903 Getafe (Madrid), Spain
`jnimora@alum.mit.edu`
`http://alum.mit.edu/www/jnimora`

February 3, 2004

Abstract

This paper develops a framework based on convex optimization and economic ideas to formulate and solve approximately a rich class of dynamic and stochastic resource allocation problems, fitting in a generic discrete-state multi-project *restless bandit problem (RBP)*. It draws on the single-project framework in the author's companion paper "Restless bandit marginal productivity indices I: Single-project case and optimal control of a make-to-stock $M/G/1$ queue," based on characterization of a project's *marginal productivity index (MPI)*. Our framework significantly expands the scope of Whittle (1988)'s seminal approach to the RBP. Contributions include: (i) Formulation of a generic multi-project RBP, and algorithmic solution via single-project MPIs of a *relaxed problem*, giving a lower bound on optimal cost performance; (ii) a heuristic MPI-based *hedging point and index policy*; (iii) application of the MPI policy and bound to the problem of dynamic scheduling for a multiclass *combined MTO/MTS $M/G/1$* queue with convex backorder and stock holding cost rates, under the LRA criterion; and (iv) results of a computational study on the MPI bound and policy, showing the latter's near-optimality across the cases investigated.

1 Introduction

This paper develops a framework based on convex optimization and economic ideas to formulate and solve *approximately* a rich class of dynamic and stochastic re-

source allocation problems, fitting in a generic discrete-state multi-project *restless bandit problem (RBP)*. It draws on the single-project framework in the author's companion paper Niño-Mora (2004), based on a unifying definition and characterization of a project's *marginal productivity index (MPI)*.

The approach is deployed to address an important problem in manufacturing applications, concerning the dynamic scheduling of a multi-product *combined make-to-order (MTO)/make-to-stock (MTS)* production-inventory facility, modeled by a multiclass MTO/MTS $M/G/1$ queue with convex backorder/stock holding cost rates. Results include a *hedging point and index scheduling policy* coming close to minimizing the *long-run average (LRA)* cost rate per unit time, and a *lower bound* on optimal LRA cost performance.

Our framework significantly expands the scope of Whittle (1988)'s seminal approach to the multi-project RBP. Whittle considered a model concerning the optimal allocation of effort to a collection of discrete-state Markovian *restless bandit (RB)* projects, i.e. binary-action (work/rest) *Markov decision processes (MDPs)*, a *fixed* number of which must be active at each time. The special case where *one* project must be active, and rested projects do not change state, recovers the classic *multi-armed bandit problem (MBP)*, solved optimally by the *Gittins index* policy. See Gittins (1979). The increased modeling power of the RBP comes at the expense of tractability, as it is *P-space hard*. See Papadimitriou and Tsitsiklis (1999). Whittle (1988) introduced an index $\nu(i)$ attached to an RB project, depending only on its state i, and proposed as a heuristic the resulting index policy: Work at each time on the required number of projects having larger index values. The *Whittle index* emerges in the solution of a *relaxed problem*, which further gives a *performance bound*, in terms of the *Lagrange multiplier* associated to an average-activity constraint. Such index policy is optimal in the MBP case, and asymptotically optimal under certain conditions. See Weber and Weiss (1990).

Yet the Whittle index does *not* exist for all RB projects, only for a restricted class of so-called *indexable* projects. Whittle (1988) stated:

> "... one would very much like to have simple sufficient conditions for indexability; at the moment, none are known."

Such scope limitation is particularly severe in the multiclass queueing scheduling model considered in this paper, which is readily formulated as a multi-project RBP. The Whittle index does not exist for the constituent projects under the LRA criterion, as pointed out by Whittle (1996, Ch 14.7) himself, and by Veatch and Wein (1996). The latter authors state:

> "In contrast, the backorder problem is not indexable. $\nu(x)$ does not exist (i.e. equals $-\infty$) for all x. The difficulty is that ν is a Lagrange multiplier for the constraint on the time-average number of active arms. For the backorder problem, any stable policy must serve a time-average of ρ classes, so relaxing this constraint does not change the optimal value, and the Lagrange multiplier does not exist. In fact, no scheduling problem with a fixed utilization will be indexable."

In the companion paper Niño-Mora (2004) (cf. also Niño-Mora (2003)), we resolved both issues. Thus, we introduced a unifying definition of MPI for a generic

RB project, of which the Whittle index is a special case. We further introduced an MPI relative to a new, *mixed LRA-bias criterion*, which applies to the scheduling model of concern in this paper, where the Whittle index does not exist. We further furnished a *complete characterization* of indexability (existence of the MPI) for a generic RB project, showing its equivalence to satisfaction by the project of the economic *law of diminishing returns (LDR)* to effort. The paper further presented sufficient conditions for indexability, based on satisfaction by project performance measures of *partial conservation laws (PCLs)*, extending to the countable-state case the finite-state PCL framework introduced by the author in Niño-Mora (2001a, 2002); and *PCL-indexability* analyses of single-class service-controlled MTO and MTS $M/G/1$ queues with convex holding cost rates, which are the RB projects in the multiclass model considered in this paper.

Extensive research efforts have been devoted to the design of scheduling policies for multiclass queues, focusing either on the *pure* MTO or MTS cases. In the MTO case, most work has assumed linear holding cost rates, under which *static index rules* such as the $c\mu$ *rule* are often optimal. See, e.g. Niño-Mora (2001b) and the references therein. Haji and Newel (1971) argued the importance of incorporating instead *convex increasing costs of delay*, and proposed a corresponding *dynamic index rule*. Their results were extended by Van Mieghem (1995), who established a form of heavy-traffic optimality for such policy. Ansell et al. (2003) and Glazebrook et al. (2003) have addressed the pure MTO case of the model considered in this paper. They have sought to overcome the nonexistence of the Whittle index under the LRA criterion by showing its existence under the *discounted criterion*. Then, taking the limit of the discounted Whittle index scaled by the discount factor as this vanishes gives a convenient LRA index. They establish such results in the MTO $M/M/1$ and $M/G/1$ cases by an ad hoc DP analysis, under the assumption that holding cost rates are *convex increasing* in the queue's state. Their approach, however, fails to produce bounds on optimal LRA cost performance; and does not apply to the MTS case, where holding cost rates are V-shaped in the natural queues' state of *net backorder levels*. See Section 2.

The problem of scheduling a multiclass MTS queue to minimize discounted or LRA *linear* backorder and stock holding costs has attracted major research efforts since the 1990s. A variety of policies has been proposed, characterized by a *hedging point and index policy*. In the standard application to a multi-product production-inventory facility, the hedging point corresponds to a *base-stock level* for each product, and determines work vs. idling decisions: the facility works as long as there is a product whose stock level is below its base-stock level. The index policy dynamically determines which product is produced, among those whose stock is not full. See, e.g. Zheng and Zipkin (1990), Wein (1992), Zipkin (1995), Veatch and Wein (1996), and Peña-Pérez and Zipkin (1997). Further, Ha (1997), and de Véricourt et al. (2000) have shed light on the structure of optimal policies, justifying some of the proposed heuristics. In recent work, Dusonchet and Hongler (2003) have calculated the *discounted* Whittle index for an MTS $M/M/1$ queue with *linear* backorder and stock holding cost rates. However, they discard application of such approach under the LRA criterion, due to the nonexistence in such case of the Whittle index. We remark that, while the prevailing assumption of linear stock holding costs can be reasonable in practice, it appears more realistic to consider nonlinear convex increasing backorder cost rates, as we do in this paper.

3

Research on *combined* MTO/MTS multiclass queueing systems has received relative scarce attention, mostly addressing issues of performance analysis and "MTO vs. MTS" decisions. Such systems model flexible production facilities where standard products are MTS, while custom products are MTO. Such combined mode of operation is becoming increasingly pervasive in manufacturing, which underscores the importance of addressing the corresponding scheduling problem. To the best of our knowledge, this paper is the first to address the latter. We refer the reader to Soman et al. (2004) for a comprehensive review of work on combined MTO/MTS systems.

1.1 Contributions

Motivated by the issues discussed above, this paper presents the following contributions: (i) Formulation of a generic multi-project RBP, and algorithmic solution via single-project MPIs of a *relaxed problem*, giving a lower bound on optimal cost performance; (ii) a heuristic MPI-based *hedging point and index policy*; (iii) application of the MPI policy and bound to the problem of dynamic scheduling for a multiclass *combined MTO/MTS* $M/G/1$ queue with convex backorder and stock holding cost rates, under the LRA criterion; and (iv) results of a computational study on the MPI bound and policy, showing the latter's near-optimality across the cases investigated.

1.2 Structure of the paper

The rest of the paper is organized as follows. Section 2 introduces our motivating problem, concerning the dynamic scheduling of a multiclass MTO/MTS queue. Section 3 extends the single-project solution framework via MPIs in Niño-Mora (2004) to develop a heuristic hedging point and index policy for a generic multi-project RBP. Section 4 deploys the RBP policy and bound in the model of concern. Finally, Section 5 reports the results of a computational study.

In what follows, we refer the reader to the companion paper Niño-Mora (2004) for required background material on the single-project case.

2 Motivating problem

Consider a model for a multi-product production-inventory facility, where a product range labeled by $k \in \mathbb{K} = \{1, \ldots, K\}$ dynamically vies for access to shared production capacity. Products are partitioned as $\mathbb{K} = \mathbb{K}^{\text{MTO}} \cup \mathbb{K}^{\text{MTS}}$. Products $k \in \mathbb{K}^{\text{MTO}}$ *must* be MTO, whereas products $k \in \mathbb{K}^{\text{MTS}}$ *can* be MTS, allowing backorders. Note that such setting includes the *pure MTO case* ($\mathbb{K}^{\text{MTS}} = \emptyset$) and the *pure MTS case* ($\mathbb{K}^{\text{MTO}} = \emptyset$).

Customer orders of unit size for product $k \in \mathbb{K}$ arrive as a Poisson process with rate λ_k. A single flexible machine, which is used to process all orders, makes a unit of product k in a *production time* distributed as a random variable with Laplace-Stieltjes transform (LST) $\psi_k(\cdot)$, having finite mean $1/\mu_k$ and variance σ_k^2. Arrival streams and production times are mutually independent. Denoting by $\rho_k = \lambda_k/\mu_k$

product k's traffic intensity, we assume the *stability condition*

$$\rho \triangleq \sum_{k \in \mathbb{K}} \rho_k < 1.$$

For a product $k \in \mathbb{K}^{\text{MTO}}$, customer orders are placed upon arrival in a corresponding *backorder queue (BQ)*, whose *state* at time $t \geq 0$, given by its size, we denote by $X_k(t)$. The corresponding *state space* is $N_k = \{0, 1, \ldots\}$.

For a product $k \in \mathbb{K}^{\text{MTS}}$, items can be made in advance of demand, to be placed in a corresponding *finished goods stock (FGS)*, whose size at time t we denote by $X_k^-(t)$. The FGS has a *finite storage capacity* for up to $s_k \geq 1$ units. An arriving order finding an empty FGS is placed in the corresponding BQ, whose size at time t we denote by $X_k^+(t)$. We consider the product's state to be its *net backorder level* $X_k(t) = X_k^+(t) - X_k^-(t)$, so that its state space is $N_k = \{-s_k, \ldots, 0, 1, \ldots\}$.

A central controller governs system evolution by choice of a *scheduling policy* π, prescribing dynamically whether the machine is to be idle or working and, in the latter case, on which product. The policy is drawn from the class Π of *admissible policies*, which are: (i) *nonpreemptive*, i.e. production of an item cannot be interrupted; thus, the *decision epoch sequence* consists of order arrival epochs to an empty system, and product completion epochs; (ii) *nonanticipative*, i.e. decisions depend on the history of the system up to and including the present epoch; and (iii) *stable*, i.e. the policy must induce an equilibrium distribution on the joint state process, having finite moments of the required order. When service times are exponential, we can choose to expand Π to include *preemptive* policies.

The system incurs backorder and/or stock holding costs, separably across products. Product k accrues costs at rate $h_k(i_k)$ per unit time while its state is i_k. We will refer to the first and second-order differences $\Delta h_k(i_k) \triangleq h_k(i_k) - h_k(i_{k-1})$ and $\Delta^2 h_k(i_k) \triangleq \Delta h_k(i_k) - \Delta h_k(i_k - 1)$.

Assumption 2.1 Holding cost rates $h_k(i_k)$ satisfy the following:

(i) They are bounded below: $\inf\{h_k(i_k) : i_k \in N_k\} > -\infty$.

(ii) They are convex: $\Delta^2 h_k(i_k) \geq 0$, for $i_k \in N_k$ such that $i_k - 2 \in N_k$.

(iii) If $\psi_k(\cdot)$ has finite moments of up to order $m_k + 1$, then $h_k(i_k) = O(i_k^{m_k})$ as $i_k \to +\infty$.

Notice that we do not require holding cost rates $h_k(i_k)$ to be monotonic in $i_k \in N_k$, as such assumption is not appropriate in the MTS case with backorders. Instead, one will typically have that such rates are V-shaped: $h_k(i_k)$ is nondecreasing for $i_k \geq 0$ (i.e. backorder cost rates are nondecreasing in the backorder level); and $h_k(i_k)$ is nonincreasing for $i_k \leq 0$ (i.e. stock holding cost rates are nondecreasing in the stock level $-i_k$).

We will address the *LRA scheduling problem*, which is to find a policy $\pi^* \in \Pi$ attaining the minimum LRA value f^* of costs incurred.

$$f^* = \inf_{\pi \in \Pi} \lim_{T \to +\infty} \frac{1}{T} \mathbb{E}^\pi \left[\int_0^T \sum_{k \in \mathbb{K}} h_k \left(X_k(t) \right) \, dt \right]. \tag{1}$$

5

Given the intractability of problem (1) in such generality, our prime goals will be: (1) to design a well-grounded, tractable heuristic scheduling policy $\tilde{\pi}^*$, which comes close to attaining the optimal cost value f^*; and (2) to construct a tractable lower bound $\widehat{f} \leq f^*$.

3   Multi-project RBP

3.1   Problem description

This section extends the single-project framework and analysis in Section 3 of Niño-Mora (2004) to address a *multi-project RBP*, where a *central planner* wishes to optimally allocate effort to a collection of K RB projects, labeled by $k \in \mathbb{K} = \{1, \ldots, K\}$. The joint state process is $\mathbf{X}(t) = (X_k(t))_{k \in \mathbb{K}}$, for $t \geq 0$, where $X_k(t)$ is project k's state. Control is exercised by adoption of a policy π, drawn from a class Π of nonanticipative admissible policies. This prescribes how a *single operator*, able to work on at most one project at a time, is to be dynamically allocated. Our focus on the single-operator case is only due to ease of exposition, as the approach and results below readily extend to the *multi-operator* case. Each project has its own *manager*, in charge of policy implementation.

We refer the reader to Section 3 of Niño-Mora (2004) for a description of the individual RB projects considered here. Below we add their label k to the notation introduced there. Thus, project k has discrete state space

$$N_k = \left\{ j \in \mathbb{Z} : \ell_k^0 \leq j \leq \ell_k^1 \right\},$$

where $-\infty < \ell_k^0 < \ell_k^1 \leq +\infty$, with controllable (resp. uncontrollable) state space $N_k^{\{0,1\}}$ (resp. $N_k^{\{0\}} = \{\ell_k^0\}$). Its individual class of admissible policies is denoted by Π_k. We will also refer to the project's threshold, or $\mathscr{F}_k$-, policies, having active-state sets $S_k(i_k)$, for $i_k \in N_k$. Cost and work measures $f_k^{\pi_k}$ and $g_k^{\pi_k}$, for $\pi_k \in \Pi_k$, are extended to f_k^π and g_k^π, for $\pi \in \Pi$. The following conditions are required to hold.

Assumption 3.1 *For any project $k \in \mathbb{K}$ and policy $\pi \in \Pi$:*

  (i) $\sup \left\{ g_k^{\pi_k} : \pi_k \in \Pi_k \right\} \geq g_k^\pi$.

  (ii) $\inf \left\{ f_k^{\pi_k} : \pi_k \in \Pi_k \right\} \leq f_k^\pi$.

In words, the work (resp. cost) performance achieved on a project by a system-wide policy cannot exceed (resp. fall below) the corresponding supremum (resp. infimum) performance value under its individual policies.

Projects are assumed to be *indexable* relative to threshold policies. See Definition 3.6 in Niño-Mora (2004).

Assumption 3.2 *Project $k \in \mathbb{K}$ is $\mathscr{F}_k$-indexable, with MPI $\nu_k^*(j_k)$.*

Managers vie for access to the operator as this becomes available, at a decision epoch sequence $t_0 = 0 < t_1 < \cdots < t_n \to +\infty$ as $n \to +\infty$, consistent with the

individual projects'. Denote by $\mathbf{a}(t_n) = (a_k(t_n))_{k \in \mathbb{K}}$ the *joint action* at epoch t_n, where $a_k(t_n) \in \{0, 1\}$ is project k's action. The *sample-path activity constraint* is

$$a(t) \triangleq \sum_{k \in \mathbb{K}} a_k(t) \leq 1, \quad t \geq 0. \tag{2}$$

The cost performance under a policy π is evaluated by *holding cost measure*

$$f^\pi \triangleq \sum_{k \in \mathbb{K}} f_k^\pi.$$

The *multi-project RBP* of concern is to find a policy attaining the system's optimal cost performance:

$$\text{Find } \pi^* \in \Pi : f^{\pi^*} = f^* \triangleq \inf \left\{ f^\pi : \pi \in \Pi \right\}. \tag{3}$$

Our goals are: (i) design a well-grounded tractable policy $\tilde{\pi}^* \in \Pi$, coming close to minimizing cost performance; and (ii) produce a tractable lower bound $\widehat{f} \leq f^*$, which can be used to assess the policy's suboptimality gap.

3.2 Relaxed problem

We will develop an approach based on solution of a *relaxed problem*. This refers to *work measure* g^π, evaluating the effort under a policy π, given by

$$g^\pi \triangleq \sum_{k \in \mathbb{K}} g_k^\pi.$$

Note that Assumptions 3.1(ii) in Niño-Mora (2004) and and 3.1(i) ensure existence of an upper bound $\widehat{g}$:

$$g^\pi \leq \widehat{g}, \quad \pi \in \Pi. \tag{4}$$

Inequality (4) will furnish the key constraint to define the relaxed problem.

Consider a *modified system* where each project has its own operator, and hence the set of active projects ranges from $\emptyset$ to $\mathbb{K}$. Control is exercised through a *relaxed policy* $\widehat{\pi}$, drawn from a class $\widehat{\Pi}$ of *admissible relaxed policies*. Work and cost measures are extended to policies $\widehat{\pi} \in \widehat{\Pi}$, giving $f_k^{\widehat{\pi}}, g_k^{\widehat{\pi}}$,

$$f^{\widehat{\pi}} \triangleq \sum_{k \in \mathbb{K}} f_k^{\widehat{\pi}} \quad \text{and} \quad g^{\widehat{\pi}} \triangleq \sum_{k \in \mathbb{K}} g_k^{\widehat{\pi}}.$$

The following conditions are required to hold.

Assumption 3.3

(i) $\widehat{\Pi} \supset \Pi$.

(ii) $\widehat{\Pi} \supset \prod_{k \in \mathbb{K}} \Pi_k.$

(iii) $\left\{ (g_k^{\widehat{\pi}}, f_k^{\widehat{\pi}}) : \widehat{\pi} \in \widehat{\Pi} \right\} = \left\{ (g_k^{\pi_k}, f_k^{\pi_k}) : \pi_k \in \Pi_k \right\}, \quad k \in \mathbb{K}.$

Remark 3.4 In words, Assumption 3.3 says the following:

(i) Part (i) justifies the term "relaxed policies."

(ii) Part (ii) means that $\widehat{\Pi}$ includes the class $\prod_{k\in\mathbb{K}}\Pi_k$ of *admissible decentralized policies*. These are of the form $\widehat{\pi} = (\pi_k)_{k\in\mathbb{K}}$, i.e. each project k is autonomously controlled under its own individual policy $\pi_k \in \Pi_k$.

(iii) Part (iii) says that a project's work-cost performance achieved by relaxed policies $\widehat{\pi} \in \widehat{\Pi}$ is the same as that achieved by individual policies $\pi_k \in \Pi_k$.

The *relaxed problem* of concern is:

$$\text{Find } \widehat{\pi}^* \in \widehat{\Pi} : f^{\widehat{\pi}^*} = \widehat{f} \triangleq \inf\left\{ f^{\widehat{\pi}} : g^{\widehat{\pi}} \leq \widehat{g}, \widehat{\pi} \in \widehat{\Pi} \right\}. \tag{5}$$

3.3 Reformulation as convex resource allocation problem

We develop below a convex optimization approach to solve problem (5), drawing on Section 3.3 in Niño-Mora (2004). Define the *relaxed achievable work-cost (performance) region* by

$$\widehat{\mathbb{H}} \triangleq \left\{ (b, z) \in \mathbb{R}^2 : (b, z) = (g^{\widehat{\pi}}, f^{\widehat{\pi}}) \text{ for some } \widehat{\pi} \in \widehat{\Pi} \right\}, \tag{6}$$

and denote its projections over the work and cost spaces by $\widehat{\mathbb{B}}$ and $\widehat{\mathbb{V}}$, respectively.

We next show that such regions can be decomposed as *Minkowski sums* (denoted by operator $\oplus$) of their single-project counterparts $\mathbb{H}_k$, $\mathbb{B}_k$ and $\mathbb{V}_k$.

Lemma 3.5

(a) $\widehat{\mathbb{H}} = \oplus_{k\in\mathbb{K}}\mathbb{H}_k$.

(b) $\widehat{\mathbb{B}} = \oplus_{k\in\mathbb{K}}\mathbb{B}_k$.

(c) $\widehat{\mathbb{V}} = \oplus_{k\in\mathbb{K}}\mathbb{V}_k$.

Proof. Part (a) follows from Assumption 3.3(ii, iii), through

$$\widehat{\mathbb{H}} \triangleq \left\{ (b, z) \in \mathbb{R}^2 : (b, z) = (g^{\widehat{\pi}}, f^{\widehat{\pi}}) \text{ for some } \widehat{\pi} \in \widehat{\Pi} \right\}$$

$$= \left\{ (b, z) \in \mathbb{R}^2 : (b, z) = \left(\sum_{k\in\mathbb{K}} g^{\pi_k}, \sum_{k\in\mathbb{K}} f^{\pi_k}\right) : \pi_k \in \Pi_k, k \in \mathbb{K} \right\}$$

$$= \left\{ (b, z) \in \mathbb{R}^2 : (b, z) = \sum_{k\in\mathbb{K}}(b_k, z_k), \quad (b_k, z_k) \in \mathbb{H}_k, k \in \mathbb{K} \right\} \triangleq \oplus_{k\in\mathbb{K}}\mathbb{H}_k.$$

Parts (b) and (c) follow from part (a). □

Convexity of such regions follows from their single-project counterparts', extending to *closures* $\overline{\widehat{\mathbb{H}}}$, $\overline{\widehat{\mathbb{B}}}$ and $\overline{\widehat{\mathbb{V}}}$. Consider the *relaxed efficient work-cost frontier*

$$\underline{\partial}\widehat{\mathbb{H}} \triangleq \left\{ (b, z) \in \overline{\widehat{\mathbb{H}}} : b \in \widehat{\mathbb{B}} \text{ and } z \leq f^{\widehat{\pi}} \text{ for any } \widehat{\pi} \in \widehat{\Pi} \text{ with } g^{\widehat{\pi}} = b \right\}.$$

This is characterized as the graph of *relaxed cost function*

$$\widehat{C}(b) \triangleq \inf\left\{f^{\widehat{\pi}} : g^{\widehat{\pi}} = b, \widehat{\pi} \in \widehat{\Pi}\right\} = \inf\left\{z : (b, z) \in \widehat{\mathbb{H}}\right\}, \quad b \in \widehat{\mathbb{B}}, \quad (7)$$

whose convexity follows from that of region $\widehat{\mathbb{H}}$, so that

$$\underline{\partial}\widehat{\mathbb{H}} = \left\{(b, \widehat{C}(b)) : b \in \widehat{\mathbb{B}}\right\}. \quad (8)$$

We can now reformulate (5) as the *convex resource allocation problem*

$$\text{Find } b^* \in \widehat{\mathbb{B}} : \widehat{C}(b^*) = \widehat{f} \triangleq \inf\left\{\widehat{C}(b) : b \leq \widehat{g}, b \in \widehat{\mathbb{B}}\right\}. \quad (9)$$

To evaluate $\widehat{C}(b)$ we will further address the *relaxed b-work problem*:

$$\text{Find } \widehat{\pi}^* \in \widehat{\Pi} \text{ with } g^{\widehat{\pi}^*} = b : f^{\widehat{\pi}^*} = \widehat{C}(b) \triangleq \inf\left\{f^{\widehat{\pi}} : g^{\widehat{\pi}} = b, \widehat{\pi} \in \widehat{\Pi}\right\}. \quad (10)$$

A relaxed policy $\widehat{\pi} \in \widehat{\Pi}$ will be said to be *b-work feasible* if $g^{\widehat{\pi}} = b$.

3.4 Lagrangian multiplier analysis and decentralization

We address problem (10) via a Lagrangian approach, along the lines of Section 3.4 in Niño-Mora (2004). Dualizing constraint $g^{\widehat{\pi}} = b$ by multiplier $\nu \in \mathbb{R}$ gives the *Lagrangian function*

$$\mathscr{L}_b^{\widehat{\pi}}(\nu) \triangleq f^{\widehat{\pi}} + \nu\left[g^{\widehat{\pi}} - b\right] = \sum_{k \in \mathbb{K}} v_k^{\widehat{\pi}}(\nu) - \nu b,$$

where

$$v_k^{\widehat{\pi}}(\nu) \triangleq f_k^{\widehat{\pi}} + \nu g_k^{\widehat{\pi}}.$$

Again we interpret ν as the *wage* rate earned by operators. Hence, $v_k^{\widehat{\pi}}(\nu)$ gives project k's holding and labor costs, and $\mathscr{L}_b^{\widehat{\pi}}(\nu)$ is the system-wide cost where work expended above (resp. below) b units is paid (resp. sold) at wage ν.

The unconstrained *Lagrangian problem* is

$$\text{Find } \widehat{\pi}^* \in \widehat{\Pi} : \mathscr{L}_b^{\widehat{\pi}^*}(\nu) = \mathscr{L}_b^*(\nu) \triangleq \inf\left\{\mathscr{L}_b^{\widehat{\pi}}(\nu) : \widehat{\pi} \in \widehat{\Pi}\right\}. \quad (11)$$

We next show that use of decentralized policies $\widehat{\pi} = (\pi_k)_{k \in \mathbb{K}}$ suffices to solve (11). Let $v_k^*(\nu)$ be the optimal value of *project k's ν-wage subproblem*:

$$\text{Find } \pi_k^* \in \Pi_k : v_k^{\pi_k^*}(\nu) = v_k^*(\nu) \triangleq \inf\left\{v_k^{\pi_k}(\nu) : \pi_k \in \Pi_k\right\}. \quad (12)$$

Lemma 3.6

$$\mathscr{L}_b^*(\nu) = \sum_{k \in \mathbb{K}} v_k^*(\nu) - \nu b.$$

Proof. We can write

$$\mathscr{L}_b^*(\nu) \triangleq \inf\left\{\mathscr{L}_b^{\widehat{\pi}}(\nu) : \widehat{\pi} \in \widehat{\Pi}\right\} = \inf\left\{\sum_{k \in \mathbb{K}} v_k^{\widehat{\pi}}(\nu) : \widehat{\pi} \in \widehat{\Pi}\right\} - \nu b$$

$$= \inf\left\{\sum_{k \in \mathbb{K}} v_k^{\pi_k}(\nu) : \pi_k \in \Pi_k, k \in \mathbb{K}\right\} - \nu b = \sum_{k \in \mathbb{K}} v_k^*(\nu) - \nu b, \quad (13)$$

where the third identity follows from Lemma 3.5(a). $\qquad\square$

The central planner can thus solve the Lagrangian problem by quoting to managers wage ν, and letting them solve their ν-wage subproblems. This suggests decentralizing the relaxed b-work problem's solution through wage choice.

3.5 Duality-based optimality conditions and shadow wages

We next seek to find optimality conditions for a decentralized policy in *primal* relaxed b-work problem. To price the value of work we will use its *dual* (or *pricing*) problem, which is to find a wage $\nu^* \in \mathbb{R}$ maximizing (concave) objective $\mathscr{L}_b^*(\nu)$:

$$\text{Find } \nu^* \in \mathbb{R} : \mathscr{L}_b^*(\nu^*) = \widehat{Q}(b) \triangleq \sup\left\{\mathscr{L}_b^*(\nu) : \nu \in \mathbb{R}\right\}. \tag{14}$$

We will use the *duality gap* associated to a relaxed policy $\widehat{\pi}$ and a wage ν:

$$\Delta_b^{\widehat{\pi}}(\nu) \triangleq f^{\widehat{\pi}} - \mathscr{L}_b^*(\nu). \tag{15}$$

Notice that, for a decentralized policy $\widehat{\pi} = (\pi_k)_{k \in \mathbb{K}}$, the duality gap reduces to

$$\Delta_b^{\widehat{\pi}}(\nu) = \sum_{k \in \mathbb{K}} \left[v_k^{\pi_k}(\nu) - v_k^*(\nu)\right] + \nu \left[b - \sum_{k \in \mathbb{K}} g_k^{\pi_k}\right]. \tag{16}$$

The next result follows immediately.

Lemma 3.7 (Weak duality)

(a) *Let $\widehat{\pi} \in \widehat{\Pi}$ be b-work feasible and let $\nu \in \mathbb{R}$. Then, $\mathscr{L}_b^*(\nu) \leq f^{\widehat{\pi}}$.*

(b) $\widehat{Q}(b) \leq \widehat{C}(b)$.

Lemma 3.7 and identity (16) suggest the following sufficient optimality conditions for a decentralized policy $\widehat{\pi}^* = (\pi_k^*)_{k \in \mathbb{K}} \in \prod_{k \in \mathbb{K}} \Pi_k$ and a wage $\nu^* \in \mathbb{R}$:

(i) **Primal feasibility:** $\displaystyle\sum_{k \in \mathbb{K}} g_k^{\pi_k^*} = b$.

(ii) **Project-wise optimality:** Policy π_k^* is optimal for project k's ν^*-wage subproblem, i.e. $v_k^{\pi_k^*}(\nu^*) = v_k^*(\nu^*)$ for $k \in \mathbb{K}$.

Theorem 3.8 (Sufficient optimality conditions) *Under conditions* (i)–(ii) *above:*

(a) *Policy $\widehat{\pi}^*$ is optimal for primal relaxed b-work problem* (10).

(b) *Wage ν^* is optimal for its dual problem* (14).

(c) *Strong duality holds:* $\widehat{Q}(b) = \widehat{C}(b) = \displaystyle\sum_{k \in \mathbb{K}} f_k^{\pi_k^*}$.

Proof. The results follow via Lemma 3.7, using the fact that conditions (i)–(ii) and identity (16) ensure there is a zero duality gap. $\qquad\square$

We will refer to a wage ν^* satisfying the conditions in Theorem 3.8 as a *shadow wage* for the relaxed b-work problem. If $\widehat{C}(\cdot)$ is derivable at b, we have

$$\nu^* = -\frac{d}{db}\widehat{C}(b). \tag{17}$$

Namely, ν^* is the *marginal productivity of work* in the relaxed b-work problem.

As in Lemma 3.4 of Niño-Mora (2004), existence of a shadow wage is necessary for optimality.

Lemma 3.9 *Let $\widehat{\pi}^*$ be an optimal decentralized policy for the relaxed b-work problem. Then, there exists a corresponding shadow wage ν^*.*

3.6 Construction of relaxed cost function

We address next the *construction* of function $\widehat{C}(b)$. We will use the following notation. Given a joint state $\mathbf{j} = (j_k)_{k\in\mathbb{K}}$, denote by $\mathbf{S}(\mathbf{j}) = (S_k(j_k))_{k\in\mathbb{K}}$ the decentralized policy using the $S_k(j_k)$-active policy on project $k \in \mathbb{K}$. Write further $\boldsymbol{\ell}^0 = (\ell_k^0)_{k\in\mathbb{K}}$, $\boldsymbol{\ell}^1 = (\ell_k^1)_{k\in\mathbb{K}}$, and let $\mathbf{e}_k$ be the kth unit coordinate vector in $\mathbb{R}^{\mathbb{K}}$.

Consider the algorithm in Figure 1, which generates a sequence of joint state, project label, and wage triples $(\mathbf{j}^n, k_n, \nu^n)$, for $n \geq 0$. This is finite if all project state spaces are finite, and is infinite otherwise. The notation "$n \geq 0$" thus refers to relevant values of n. The algorithm constructively defines cost function $\widehat{C}^{\mathscr{F}}(b)$ for $b \in \widehat{\mathbb{B}}$, by *linear interpolation* on generated work-cost pairs $(g^{\mathbf{S}(\mathbf{j}^n)}, f^{\mathbf{S}(\mathbf{j}^n)})$.

The main result of this section, given in Theorem 3.11 below, is that $\widehat{C}^{\mathscr{F}}(b) = \widehat{C}(b)$. We will draw on the following preliminary result.

Lemma 3.10 *Sequence $(\mathbf{j}^n, k_n, \nu^n)$ satisfies the following:*

(a) $\nu^n \leq \nu^{n+1}$.

(b) $\max\limits_{j_k^n > \ell_k^0, k\in\mathbb{K}} \nu_k^*(j_k^n) \leq \nu^n = \nu_{k_n}^*(j_{k_n}^{n+1}) = \min\limits_{j_k^n < \ell_k^1, k\in\mathbb{K}} \nu_k^*(j_k^n + 1)$.

Proof. (a) It follows by construction, using nondecreasingness of each index $\nu_k^*(\cdot)$.

(b) The "min" equality follows by choice of k_n in the algorithm.

To show the "max" inequality use induction on n. We interpret the case $n = 0$ letting the maximum over $\emptyset$ be $-\infty$. The case $n = 1$ follows from $\nu_{k_0}^*(j_{k_0}^1) = \nu^0 \leq \nu^1$. Suppose it holds for $n - 1$. Then, using part (a) and the induction hypothesis gives that, for $k \neq k_{n-1}$ with $j_k^{n-1} > \ell_k^0$, we have

$$j_k^n = j_k^{n-1} \implies \nu_k^*(j_k^n) = \nu_k^*(j_k^{n-1}) \leq \nu_{k_{n-1}}^*(j_{k_{n-1}}^n) = \nu^{n-1} \leq \nu^n = \nu_{k_n}^*(j_{k_n}^{n+1}).$$

This completes the proof. $\qquad\qquad\square$

Theorem 3.11 *The relaxed b-work problem's cost function is given by*

$$\widehat{C}(b) = \widehat{C}^{\mathscr{F}}(b) = \max\left\{ f^{\mathbf{S}(\mathbf{j}^n)} + \nu^n\left[g^{\mathbf{S}(\mathbf{j}^n)} - b\right] : n \geq 0\right\}, \quad b \in \widehat{\mathbb{B}},$$

and is hence piecewise linear convex.

```
Initialization:
let  j⁰ := ℓ⁰; n := 0
Loop:

while  jⁿ ≠ ℓ¹  do
        choose kₙ ∈ arg min {νₖ*(jₖⁿ + 1) : jₖⁿ < ℓₖ¹, k ∈ 𝕂}
        let  jⁿ⁺¹ := jⁿ + eₖₙ;    let  νⁿ := νₖₙ*(jₖₙⁿ⁺¹)
        let  Ĉ𝓕(b) := f^S(jⁿ) + νⁿ [g^S(jⁿ) − b],    b ∈ [g^S(jⁿ⁺¹), g^S(jⁿ)]
        let  n := n + 1
end  { while }
```

Figure 1: Algorithmic construction of relaxed cost function.

Proof. For $b \in [g^{\mathbf{S}(\mathbf{j}^{n+1})}, g^{\mathbf{S}(\mathbf{j}^n)}]$, let

$$q_n \triangleq \frac{b - g^{\mathbf{S}(\mathbf{j}^{n+1})}}{g^{\mathbf{S}(\mathbf{j}^n)} - g^{\mathbf{S}(\mathbf{j}^{n+1})}} = \frac{b - g^{\mathbf{S}(\mathbf{j}^{n+1})}}{-\Delta g_{k_n}^{S_{k_n}(j_{k_n}^n + 1)}}.$$

Define decentralized policy $\widehat{\pi}^n = (\pi_k^n)_{k \in \mathbb{K}}$ by

$$\pi_k^n = \begin{cases} S_k(j_k^n) & \text{if } k \neq k_n \\ S_{k_n}^{q_n}(j_{k_n}^{n+1}) & \text{if } k = k_n, \end{cases}$$

where $S_{k_n}^{q_n}(j_{k_n}^{n+1})$ is the randomized policy defined as in Section 3.7 of Niño-Mora (2004). Then, it is easily seen, using Lemma 3.10(b), that policy $\widehat{\pi}^n$ and wage ν^n satisfy the sufficient optimality conditions for the relaxed b-work problem in Theorem 3.8. The "max" representation follows from the algorithm's construction and Lemma 3.10(a). This completes the proof. □

3.7 Algorithm and optimal decentralized policy for relaxed problem

We next draw on the above to solve relaxed problem (5) by a decentralized policy, thus producing the required lower bound $\widehat{f}$ on optimal cost. For such purpose, we introduce the algorithm RELAXED, described in Figure 3.

The algorithm takes as input an upper bound $\widehat{g}$ on work performance satisfying (4). Upon termination, it produces as output a 6-tuple $(\widehat{f}, \nu^*, n, \mathbf{j}^n, k_n, q_n)$. From this we construct decentralized policy $\widehat{\pi}^* = (\pi_k^*)_{k \in \mathbb{K}}$ by letting

$$\pi_k^* \triangleq \begin{cases} S_k(j_k^n) & \text{if } k \in \mathbb{K} \setminus \{k_n\} \\ S_k^{q_n}(j_{k_n}^{n+1}) & \text{if } k = k_n. \end{cases}$$

The main result of this section, given in Theorem 3.12 below, is that policy $\widehat{\pi}^*$ is optimal for the relaxed problem. This is graphically illustrated in Figure 2.

Theorem 3.12 *Decentralized policy $\widehat{\pi}^*$ solves optimally relaxed problem* (5). *Its optimal value is given by $\widehat{f}$, as computed by the algorithm.*
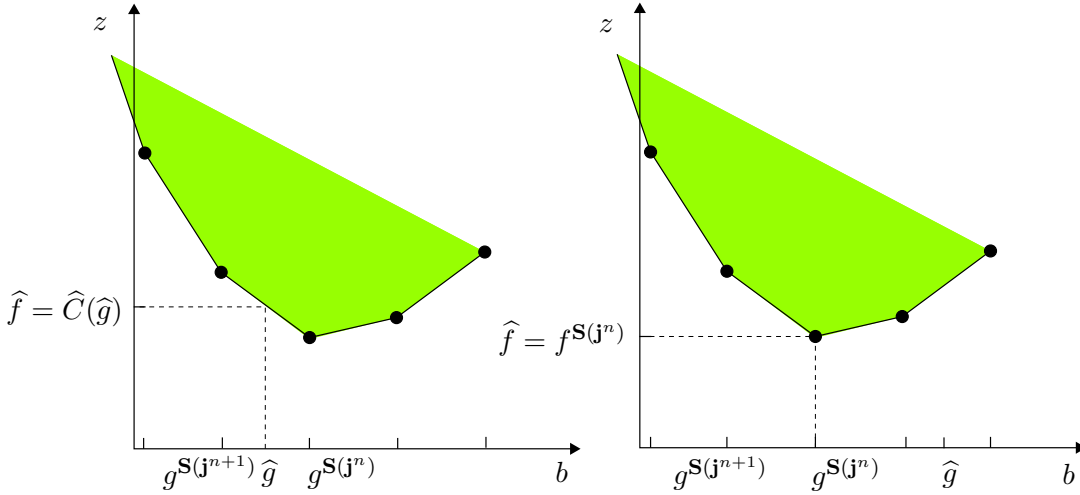
Figure 2: Solution of relaxed problem (depending on position of upper bound $\widehat{g}$).

Proof. From the discussion in Section 3.6, we see that algorithm RELAXED *traverses* from right to left the relaxed efficient work-cost frontier. The algorithm exploits the latter's piecewise linear structure by *pivoting* from a *corner* $(g^{\mathbf{S}(\mathbf{j}^n)}, f^{\mathbf{S}(\mathbf{j}^n)})$ to the left-adjacent corner $(g^{\mathbf{S}(\mathbf{j}^{n+1})}, f^{\mathbf{S}(\mathbf{j}^{n+1})})$ along such frontier. Termination occurs when a corner $(g^{\mathbf{S}(\mathbf{j}^{n+1})}, f^{\mathbf{S}(\mathbf{j}^{n+1})})$ is first reached having a feasible work performance $g^{\mathbf{S}(\mathbf{j}^{n+1})} \leq \widehat{g}$, *and* a nonnegative right slope $\nu^n \geq 0$. $\qquad\square$

3.8 Hedging point and index policy, and auction interpretation

We propose next a heuristic policy for RBP (3), based on the optimal index solution to the relaxed $\widehat{g}$-work problem. In contrast with the policy proposed by Whittle (1988) for a special type of RBP, prescribing to engage at each time a project with larger index, our proposed policy introduces a *hedging point* $\mathbf{i}^* = (i_k^*)_{k \in \mathbb{K}}$, where $i_k^* \in N_k$ for $k \in \mathbb{K}$, to determine *idling* decisions.

Our heuristic policy with hedging point $\mathbf{i}^*$, which we denote by $\tilde{\pi}(\mathbf{i}^*)$, operates as follows. At a decision epoch in state $\mathbf{j} = (j_k)_{k \in \mathbb{K}}$:

1. If $\mathbf{j} \leq \mathbf{i}^*$ (componentwise), let the operator rest (idle the system).

2. Otherwise, assign the operator to a project $k(\mathbf{j})$ satisfying

$$k(\mathbf{j}) \in \arg\max \{\nu_k^*(j_k) : j_k > i_k^*, k \in \mathbb{K}\}.$$

Such policy resolves the dynamic resource allocation problem by a *decentralized auction* mechanism. When the operator becomes free, managers of available projects vie for access to it during the next period by *bidding* an amount equal to their project's MPI. The central planner resolves the auction by allocating the operator to the highest bidder, among projects k whose state lies above their critical threshold i_k^*. In the multiple-operator model extension, an auction would be performed for each free operator.

It remains to determine an appropriate hedging point $\mathbf{i}^*$. A naive approach would use the optimal solution of the decentralized problem in the previous Section

```
ALGORITHM RELAXED
Input: ĝ
Output: (f̂, ν*, n, jⁿ, kₙ, qₙ)
Initialization:
let  jⁿ := ℓ⁰;    let  n := −1
Loop:

repeat
        let  n := n + 1
        choose kₙ ∈ arg min {ν*ₖ(jⁿₖ + 1) : jⁿₖ < ℓ¹ₖ, k ∈ 𝕂}
        let  jⁿ⁺¹ := jⁿ + eₖₙ;    let  νⁿ := ν*ₖₙ(jⁿ⁺¹ₖₙ)
until  gˢ⁽ʲⁿ⁺¹⁾ ≤ ĝ  and  νⁿ ≥ 0

Ending:

if  ĝ < gˢ⁽ʲⁿ⁾ then
   let  qₙ := (ĝ − gˢ⁽ʲⁿ⁺¹⁾) / (gₖₙ^(Sₖₙ(jⁿₖₙ)) − gₖₙ^(Sₖₙ(jⁿ⁺¹ₖₙ)));    let  ν* := νⁿ
   let  f̂ := fˢ⁽ʲⁿ⁾ + ν*[gˢ⁽ʲⁿ⁾ − ĝ]
else
   let  qₙ := 1;    let  ν* := 0;    let  f̂ := gˢ⁽ʲⁿ⁾
end  { if }
```

Figure 3: Algorithm for solving the relaxed problem.

to set $\mathbf{i}^* = \mathbf{j}^n$, where $\mathbf{j}^n$ is produced by algorithm RELAXED in Figure 3. We have found through computational experience, however, that such approach produces inadequate (too large) threshold levels i_k^*, yielding a policy whose performance is often far from optimal.

Instead, we propose to determine threshold point $\mathbf{i}^*$ using the *descent algorithm* described in Figure 4. The algorithm's validity relies on the following conjecture.

Conjecture 3.13 *The continuous extension by linear interpolation of discrete cost function $\mathbf{i} \mapsto f^{\tilde{\pi}(\mathbf{i})}$ is convex.*

Besides being intuitively appealing, we have experimentally verified Conjecture 3.13 across a wide range of problem instances, corresponding to the scheduling model of concern in this paper. Figure 5 graphically illustrates the conjecture's validity in one of the instances we have investigated.

Regarding implementation of algorithm DESCENT, in practice it will typically not be possible to evaluate function $\mathbf{i} \mapsto f^{\tilde{\pi}(\mathbf{i})}$. Instead, one can use *simulation* to obtain an estimate $\hat{f}^{\tilde{\pi}(\mathbf{i})}$ of $f^{\tilde{\pi}(\mathbf{i})}$, and perform the comparison step in the algorithm using corresponding estimates. We remark that in the algorithm's step for finding an *adjacent* hedging point improving upon the current one, it is meant a new hedging point obtained by a unit displacement (+/- 1) in *one* component of the current point.

```
ALGORITHM DESCENT
Output: i*
Initialization:
set i⁰ (arbitrarily);    let n := 0
let FOUND := false
Loop:

repeat
        try to find iⁿ⁺¹ adjacent to iⁿ such that  f^{π̃(iⁿ⁺¹)} < f^{π̃(iⁿ)}
        if there is no such iⁿ⁺¹ then let FOUND := true else let n := n + 1
until FOUND

Ending:

let i* := iⁿ
```

Figure 4: Algorithm DESCENT for hedging-point computation.

4 Application of the MPI policy and bound

In this section we return to our motivating scheduling problem in Section 2, to construct the policy and bound resulting from the above framework. We first note that the LRA scheduling problem (1) of concern is immediately formulated as a special case of multi-project RBP (3). The latter's individual projects are service-controlled MTO and MTS $M/G/1$ queues with convex holding cost rates. In Sections 6 and 7 of the companion paper Niño-Mora (2004) the reader will find *PCL-indexability analyses* of such RB projects, including closed-form expressions for the corresponding MPIs relative to the *LRA-bias criterion*, which is the appropriate one for our purposes. We obtain the following unifying formulation for the MPI $\nu_k^*(i_k)$ of queue k. Let L_k be a random variable having the equilibrium distribution of a standard $M/G/1$ queue with arrival rate λ_k and service-time LST $\psi_k(\cdot)$. Then,

$$\nu_k^*(i_k) = \mu_k \mathbb{E}\left[\Delta h_k(L_k + i_k)\right], \quad i_k \in N_k^{\{0,1\}} = \{-s_k + 1, \ldots, 0, 1, \ldots\}. \quad (18)$$

To obtain the lower bound $\widehat{f}$ on optimal LRA cost performance, the above framework requires us to produce an upper bound $\widehat{g}$ on overall work measure g^π. In our case, the latter is the *bias* or *excess work* performance achieved by policy π over the nominal allocation ρ, given by

$$g_{\mathbf{i}}^\pi \triangleq \lim_{\alpha \searrow 0}\left\{\mathbb{E}_{\mathbf{i}}^\pi\left[\int_0^\infty e^{-\alpha t}a(t)\,dt\right] - \frac{\rho}{\alpha}\right\} = \mathbb{E}_{\mathbf{i}}^\pi\left[\int_0^\infty (a(t) - \rho)\,dt\right], \quad (19)$$

where $a(t) \in \{0, 1\}$ is the facility's busy/idle indicator, and we have now made explicit the dependence on initial joint state $\mathbf{X}(0) = \mathbf{i}$. Given the latter, we will give a corresponding upper bound $\widehat{g}_{\mathbf{i}}$.

Define

$$\widehat{g}_{\mathbf{i}} \triangleq \sum_k \frac{i_k - s_k}{\mu_k} - \sum_k \frac{\lambda_k\left(\frac{1}{\mu_k^2} + \sigma_k^2\right)}{2(1 - \rho)}. \quad (20)$$

The following result establishes that $\widehat{g}_{\mathbf{i}}$ provides the required upper bound.
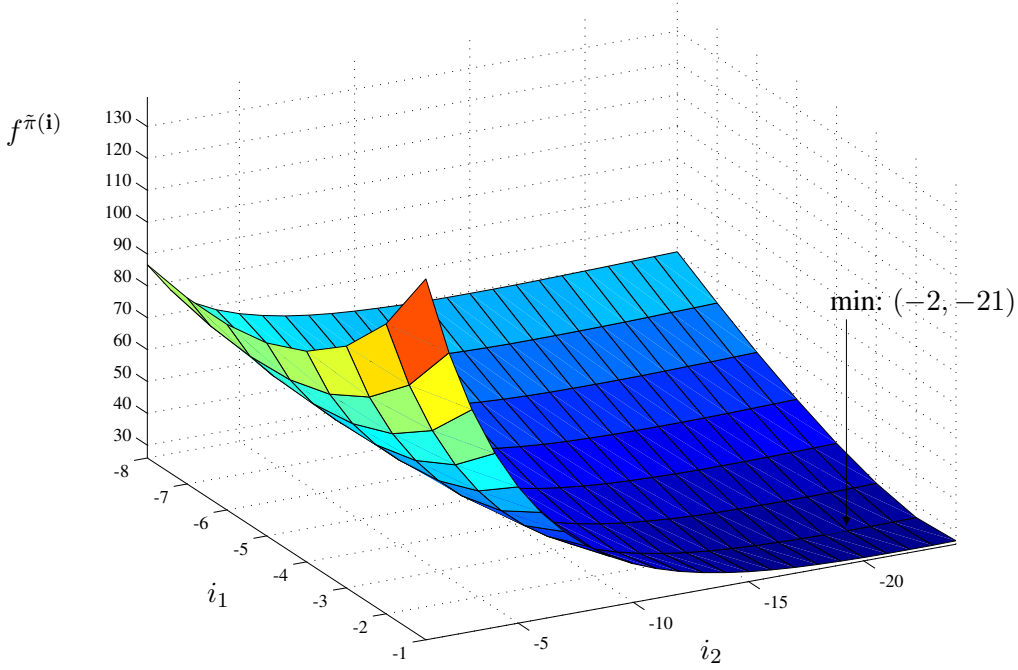
15

Figure 5: Convexity of $f^{\tilde{\pi}(\mathbf{i})}$ in case 5 of Table 1.

Proposition 4.1

$$g_{\mathbf{i}}^{\pi} \leq \widehat{g}_{\mathbf{i}}, \quad \pi \in \Pi.$$

Proof. We first transform the multiclass combined MTO/MTS $M/G/1$ model into a multiclass *pure* MTO model in the standard fashion, i.e. by redefining the state of queue k to be $L_k(t) = X_k(t) - s_k \geq 0$, for $k \in \mathbb{K}$. Now, it is clear that bias, or excess work, measure $g_{\mathbf{i}}^{\pi}$ in (19) is maximized by *any* work-conserving (nonidling) policy in the transformed system. Hence, we only need calculate the bias work corresponding to any such policy in the multiclass MTO $M/G/1$ queue, for which we draw on and extend standard results in Kleinrock (1976, Ch. 3).

Let $\phi_k = \phi_k(\alpha)$ be the LST of the busy period for the latter system starting with *one* class k customer, i.e. with initial state $\mathbf{L}(0) = \mathbf{e}_k$. The ϕ_k's are characterized as the unique solution of fixed-point equation system

$$\phi_k = \psi_k \left(\alpha + \sum_{l=1}^{K} \lambda_l (1 - \phi_l) \right) = \psi_k \left(\alpha + \lambda(1 - \phi) \right), \quad k \in \mathbb{K}, \qquad (21)$$

where we write

$$\lambda = \sum_k \lambda_k \quad \text{and} \quad \phi = \sum_k \frac{\lambda_k}{\lambda} \phi_k.$$

We can thus bound above the bias work $g_{\mathbf{i}}^{\pi}$ corresponding to a policy $\pi \in \Pi$ as

16

follows:

$$
\begin{aligned}
g_{\mathbf{i}}^{\pi} &\leq \lim_{\alpha \searrow 0} \mathbb{E}_{\mathbf{i}-\mathbf{s}}^{\text{FCFS}} \left[\int_0^\infty e^{-\alpha t} 1\{\mathbf{L}(t) \neq \mathbf{0}\} \, dt \right] - \frac{\rho}{\alpha} \\
&= \lim_{\alpha \searrow 0} \frac{1}{\alpha} - \frac{\prod_k \phi_k^{i_k - s_k}}{\alpha + \lambda - \lambda \phi} - \frac{\rho}{\alpha} \\
&= \lim_{\alpha \searrow 0} \frac{(1-\rho)(\alpha + \lambda - \lambda \phi) - \alpha \prod_k \phi_k^{i_k - s_k}}{\alpha(\alpha + \lambda - \lambda \phi)} \\
&= \lim_{\alpha \searrow 0} \frac{(1-\rho)(1 - \lambda \phi'(\alpha)) - \prod_k \phi_k^{i_k - s_k} - \alpha \sum_k (i_k - s_k) \phi_k^{i_k - s_k - 1} \phi_k' \prod_{l \neq k} \phi_l^{i_l - s_l}}{\alpha + \lambda - \lambda \phi + \alpha(1 - \lambda \phi'(\alpha))} \\
&= \lim_{\alpha \searrow 0} \frac{-(1-\rho)\lambda \phi''(\alpha) - 2 \sum_k (i_k - s_k) \phi_k^{i_k - s_k - 1} \phi_k' \prod_{l \neq k} \phi_l^{i_l - s_l} + O(\alpha)}{2 - 2\lambda \phi'(\alpha) + O(\alpha)} \\
&= -\frac{1}{2} \frac{(1-\rho)\lambda \phi''(0) + 2 \sum_k (i_k - s_k) \phi_k'(0)}{1 - \lambda \phi'(0)} \\
&= \sum_k \frac{i_k - s_k}{\mu_k} - \frac{\sum_k \lambda_k \left\{ \frac{1}{\mu_k^2} + \sigma_k^2 \right\}}{2(1-\rho)} \\
&= \widehat{g}_{\mathbf{i}}.
\end{aligned}
$$

(22)

Notice that, in (22), $\mathbb{E}^{\text{FCFS}}[\cdot]$ denotes expectation relative to the first-come firsr-serve (FCFS) policy, though any other nondling admissible policy would give the same evaluation for the right-hand side. Further, we have applied twice l'Hôpital's rule, and have used the identities

$$
-\phi_k'(0) = \frac{1/\mu_k}{1 - \rho},
$$
$$
-\lambda \phi'(0) = \frac{\rho}{1 - \rho},
$$

$$
\phi_k''(0) = \frac{\psi_k''(0)}{(1-\rho)^2} + \frac{\frac{1}{\mu_k} \sum_l \lambda_l \frac{\psi_l''(0)}{(1-\rho)^2}}{1 - \rho} = \frac{\left\{ \frac{1}{\mu_k^2} + \sigma_k^2 \right\}}{(1-\rho)^2} + \frac{\sum_l \lambda_l \left\{ \frac{1}{\mu_l^2} + \sigma_l^2 \right\}}{\mu_k(1-\rho)^3},
$$

and

$$
\begin{aligned}
\lambda \phi''(0) &= \sum_k \lambda_k \phi_k''(0) \\
&= \sum_k \lambda_k \frac{\frac{1}{\mu_k^2} + \sigma_k^2}{(1-\rho)^2} + \sum_k \rho_k \sum_l \lambda_l \frac{\frac{1}{\mu_l^2} + \sigma_l^2}{(1-\rho)^3} \\
&= \sum_k \lambda_k \frac{\frac{1}{\mu_k^2} + \sigma_k^2}{(1-\rho)^2} + \rho \sum_l \lambda_l \frac{\frac{1}{\mu_l^2} + \sigma_l^2}{(1-\rho)^3} \\
&= \left\{ \frac{1}{(1-\rho)^2} + \frac{\rho}{(1-\rho)^3} \right\} \sum_k \lambda_k \left\{ \frac{1}{\mu_k^2} + \sigma_k^2 \right\} \\
&= \frac{\sum_k \lambda_k \left\{ \frac{1}{\mu_k^2} + \sigma_k^2 \right\}}{(1-\rho)^3},
\end{aligned}
$$

17

Case	Type	λ_1	λ_2	μ_1	μ_2	c_{11}^{B}	c_{12}^{B}	c_{21}^{B}	c_{22}^{B}	c_1^{F}	c_2^{F}
1	MTS-MTS	0.4	0.4	1	1	50	25	0	0	1	1
2	MTS-MTS	0.4	0.4	1	1	50	5	0	0	1	1
3	MTS-MTS	0.6	0.1	1	1	200	5	0	0	10	0.25
4	MTS-MTS	1.2	0.1	2	1	50	20	0	0	1	1
5	MTS-MTS	0.4	0.4	1	1	50	25	3	4	10	0.25
6	MTS-MTS	0.6	0.1	1	1	200	5	3	4	10	0.25
7	MTS-MTS	0.4	0.4	1	1	50	25	1	1	1	1
8	MTS-MTO	0.4	0.4	1	1	50	25	1	1	1	1
9	MTS-MTO	0.4	0.4	1	1	50	5	1	1	1	1
10	MTS-MTO	0.6	0.1	1	1	200	5	1	1	10	0.25
11	MTS-MTO	1.2	0.1	2	1	50	20	1	1	1	1
12	MTO-MTO	1	5	3	12	5	1	2	0.1	0	0
13	MTO-MTO	1	5	3	12	5	1	2	0.2	0	0
14	MTO-MTO	1	5	3	12	5	1	2	0.5	0	0
15	MTO-MTO	1	5	3	12	5	1	2	1	0	0
16	MTO-MTO	1	5	3	12	5	1	2	2	0	0

Table 1: Cases investigated.

which are readily obtained from (21). This completes the proof. $\qquad\square$

5 Computational study

The author has implemented all the algorithms presented in this paper in C++, using the GNU gcc compiler. In this section we report the results of a computational study on the performance of our proposed MPI policy and bound across the range of 16 two-class instances shown in Table 1. All service times are exponential. The experiments have been run on a Pentium IV computer at 3.06 Ghz.

Note that instances 1–7 correspond to pure MTS problems, 8–11 are combined MTS-MTO problems, and 12–16 are pure MTO problems. Cost parameters are to be read as follows. The backorder cost rate for queue $k \in \{1, 2\}$ in state $j_k \geq 0$ is

$$h_k(j_k) = c_{1k}^{\text{B}} j_k + c_{2k}^{\text{B}} j_k^2, \quad j_k \geq 1.$$

The finished goods stock holding cost rate for queue $k \in \{1, 2\}$ (when $k \in \mathbb{K}^{\text{MTS}}$) in state $j_k \leq -1$ is

$$h_k(j_k) = c_k^{\text{F}} j_k, \quad j_k \leq -1.$$

Instances 1–4 are taken from Table 1 in de Véricourt et al. (2000). Instances 5–7 introduce quadratic backorder cost rates into some of the above pure MTS problems. Instaces 8–11 have queue 1 operated in MTS mode and queue 2 in MTO mode. Instances 12–16 are taken from Table 1 in Ansell et al. (2003).

Table 2 reports the result of the computational study. For each instance, we have computed an approximation f^* to the optimal LRA cost rate per unit time using value iteration on a truncated state space of size 200×200. We have observed that increasing the state space size beyond such limits does not significantly change the value of f^*. The hedging points have been computed by the algorithm

Case	Hedging point	f^*	f^{MPI}	f^{myopic}	$\widehat{f}$	f^{SDP}
1	(-8, -7)	15.467	15.467	27.832	7.646 (50.6 %)	
2	(-6, -4)	10.467	10.467	13.847	6.137 (41.4 %)	
3	(-5, -3)	60.703	60.752	60.752	59.775 (1.5 %)	
4	(-8, -2)	10.273	10.273	11.322	8.764 (14.7 %)	
5	(-2, -21)	25.956	25.957	26.065	21.586 (16.8 %)	
6	(-6, -6)	62.222	62.228	62.439	60.556 (2.7 %)	
7	(-10, -6)	16.416	16.562	38.860	7.784 (52.6 %)	
8	(-15, 0)	34.056	34.056	116.115	22.021 (35.3 %)	
9	(-12, 0)	18.001	18.001	55.489	8.688 (51.7 %)	
10	(-6, 0)	64.892	64.912	66.991	60.535 (6.7 %)	
11	(-10, 0)	13.273	13.273	22.086	10.142 (23.6 %)	
12	(0, 0)	15.404	15.427	15.412	11.592 (24.7 %)	15.089 (2 %)
13	(0, 0)	17.985	17.990	17.985	13.474 (25.1 %)	17.778 (1.2 %)
14	(0, 0)	20.992	21.096	21.096	14.934 (28.9 %)	20.660 (1.6 %)
15	(0, 0)	22.917	22.917	22.999	15.867 (30.8 %)	22.418 (2.1 %)
16	(0, 0)	25.146	25.146	25.353	17.092 (32 %)	24.703 (1.8 %)

Table 2: Results of computational experiments.

DESCENT in Figure 4. After computing a hedging point (i_1^*, i_2^*), we have set $(s_1, s_2) = (i_1^*, i_2^*)$, and then used the initial state $(i_1, i_2) = (s_1, s_2)$ to compute $\widehat{g}_\mathbf{i}$ by (20), and hence the lower bound $\widehat{f}$ on optimal cost. The column for f^{MPI} shows the LRA cost performance for our proposed MPI-based hedging point and index policy, calculated by value iteration on the stated truncated state space. The column for f^{myopic} shows the corresponding LRA cost performance for the policy that uses the stated hedging point, along with the myopic index ν_i^{myopic} discussed in Sections 6 and 7 of Niño-Mora (2004). Further, the column for f^{SDP} borrows results from Ansell et al. (2003) on a lower bound on LRA cost based on a *semi-definite programming* relaxation.

We see in Table 2 that our proposed MPI policy is nearly optimal across the 16 instances considered, exhibiting a negligible suboptimality gap. The myopic policy is close to optimal in some instances, but in others its performance is poor. As we argued in Sections 6 and 7 of Niño-Mora (2004), the myopic index does not account for long-term effects, which explains its poor results in more congested systems. Regarding our proposed lower bound $\widehat{f}$, we see that in most instances it is not as close to the optimal cost f^* as would be desirable. However, notice that, to the best of our knowledge, no other lower bound on optimal cost of a comparable scope to ours has been proposed in the literature. The column on the semi-definite programming lower bound f^{SDP} shows that the latter, in the cases where it is available, is relatively close to optimal.

References

Ansell, P. S., Glazebrook, K. D., Niño-Mora, J., and O'Keeffe, M. (2003). Whittle's index policy for a multi-class queueing system with convex holding costs. *Math. Methods Oper. Res.*, 57(1):21–39.

de Véricourt, F., Karaesmen, F., and Dallery, Y. (2000). Dynamic scheduling in a make-to-stock system: A partial characterization of optimal policies. *Oper. Res.*, 48:811–819.

Dusonchet, F. and Hongler, M. O. (2003). Continuous-time restless bandit and dynamic scheduling for make-to-stock production. *IEEE Trans. Robotics and Automat.*, 19(6):977–990.

Gittins, J. C. (1979). Bandit processes and dynamic allocation indices. *J. Roy. Statist. Soc. Ser. B*, 41(2):148–177. With discussion.

Glazebrook, K. D., Lumley, R. R., and Ansell, P. S. (2003). Index heuristics for multiclass $M/G/1$ systems with nonpreemptive service and convex holding costs. *Queueing Systems Theory Appl.*, 45:81–111.

Ha, A. Y. (1997). Optimal dynamic scheduling policy for a make-to-stock production system. *Oper. Res.*, 45:42–53.

Haji, R. and Newel, G. F. (1971). Optimal strategies for priority queues with non-linear costs of delay. *SIAM J. Appl. Math.*, 20:224–240.

Kleinrock, L. (1976). *Queueing Systems. Vol. II: Computer Applications*. Wiley, New York.

Niño-Mora, J. (2001a). Restless bandits, partial conservation laws and indexability. *Adv. in Appl. Probab.*, 33(1):76–98.

Niño-Mora, J. (2001b). Stochastic scheduling. In Floudas, C. A. and Pardalos, P. M., editors, *Encyclopedia of Optimization. Vol. 5*, pages 367–372. Kluwer, Dordrecht.

Niño-Mora, J. (2002). Dynamic allocation indices for restless projects and queueing admission control: a polyhedral approach. *Math. Program.*, 93(3, Ser. A):361–413.

Niño-Mora, J. (2003). Restless bandit marginal productivity indices, diminishing returns, and scheduling a multiclass make-to-order/-stock queue. In *Proceedings of the 41st Annual Allerton Conference on Communications, Control and Computing*, pages 100–109.

Niño-Mora, J. (2004). Restless bandit dynamic allocation indices I: Single-project case and optimal control of a make-to-stock $M/G/1$ queue. Technical report, Department of Statistics, Universidad Carlos III de Madrid. Submitted 2001.

Papadimitriou, C. H. and Tsitsiklis, J. N. (1999). The complexity of optimal queuing network control. *Math. Oper. Res.*, 24(2):293–305.

Peña-Pérez, A. and Zipkin, P. (1997). Dynamic scheduling rules for a multiproduct make-to-stock queue. *Oper. Res.*, 45:919–930.

Soman, C. A., Van Donk, D. P., and Gaalman, G. (2004). Combined make-to-order and make-to-stock in a food production system. *Int. J. Production Economics*. Forthcoming.

Van Mieghem, J. A. (1995). Dynamic scheduling with convex delay costs: The generalized $c\mu$ rule. *Ann. Appl. Probab.*, 5:809–833.

Veatch, M. H. and Wein, L. M. (1996). Scheduling a multiclass make-to-stock queue: Index policies and hedging points. *Oper. Res.*, 44:634–647.

Weber, R. R. and Weiss, G. (1990). On an index policy for restless bandits. *J. Appl. Probab.*, 27:637–648.

Wein, L. M. (1992). Dynamic scheduling of a multiclass make-to-stock queue. *Oper. Res.*, 40:724–735.

Whittle, P. (1988). Restless bandits: Activity allocation in a changing world. In Gani, J., editor, *A Celebration of Applied Probability*, *J. Appl. Probab. Special Vol. 25A*, pages 287–298. Applied Probability Trust.

Whittle, P. (1996). *Optimal Control: Basics and Beyond*. Wiley, Chichester.

Zheng, Y. and Zipkin, P. (1990). A queueing model to analyze the value of centralized inventory information. *Oper. Res.*, 38:296–307.

Zipkin, P. (1995). Performance analysis of a multi-item production-inventory system under alternative policies. *Management Sci.*, 41:690–703.