Efficiency of Equilibria in Restricted Uniform Machine Scheduling with Total Weighted Completion Time as Social Cost

José R. Correa,[1] Maurice Queyranne[2]

[1] *Departamento de Ingeniería Industrial, Universidad de Chile, Santiago, Chile*

[2] *Sauder School of Business at the University of British Columbia, Vancouver, British Columbia, Canada*

Received 6 December 2010; revised 16 February 2012; accepted 6 June 2012
DOI 10.1002/nav.21497
Published online 9 July 2012 in Wiley Online Library (wileyonlinelibrary.com).

Abstract: In the last decade, there has been much progress in understanding scheduling problems in which selfish jobs aim to minimize their individual completion time. Most of this work has focused on makespan minimization as social objective. In contrast, we consider as social cost the total weighted completion time, that is, the sum of the agent costs, a standard definition of welfare in economics. In our setting, jobs are processed on *restricted uniform parallel machines*, where each machine has a speed and is only capable of processing a subset of jobs; a job's cost is its weighted completion time; and each machine sequences its jobs in weighted shortest processing time (WSPT) order. Whereas for the makespan social cost the price of anarchy is not bounded by a constant in most environments, we show that for our minsum social objective the price of anarchy is bounded above by a small constant, independent of the instance. Specifically, we show that the price of anarchy is exactly 2 for the class of unit jobs, unit speed instances where the finite processing time values define the edge set of a forest with the machines as nodes. For the general case of mixed job strategies and restricted uniform machines, we prove that the price of anarchy equals 4. From a classical machine scheduling perspective, our results establish the same constant performance guarantees for WSPT list scheduling. © 2012 Wiley Periodicals, Inc. Naval Research Logistics 59: 384–395, 2012

Keywords: combinatorial optimization; game theory; scheduling

1. INTRODUCTION

Wine production, a major industry in Chile, gives rise to a number of operational and logistic issues, see for example, Sánchez [22]. One of the issues faced by a large wine producer, hereafter "the company," concerns the centralization of the bottling decisions. The company comprises a number of relatively independent brands, each of which encompasses a number of products. One of the bottlenecks, so to speak, in wine production is bottling. The company has access to several bottling facilities, each of which with a number of bottling lines that may differ in their speed and other characteristics. Currently, each product manager decides, relatively independently, which bottling line to use for her product. Consider the set of all products (or product batches) ready for bottling at date 0 and assume that the direct bottling costs are relatively constant across bottling lines. The main cost relevant to the bottling decision for product j is then the "deferral cost" $w_j C_j$, where the weight w_j depends on the volume of j to be bottled and the per unit revenue of j, and the completion time C_j is the date at which the batch is completed and the bottled product is delivered to market. The company is evaluating whether to centralize the bottling decisions. In this article, we consider a simplified representation of this situation as a scheduling model, where the jobs represent the products (batches of different wines) to be bottled, and the machines are the bottling lines.

Machine scheduling problems have their origin in the optimization of manufacturing systems and their formal mathematical treatment goes back to the 1950s. In general, they can be described as follows. Consider n tasks that have to be processed on m parallel machines. Task j requires a certain processing time $p_{i,j}$ to be completed, if processed on machine i, and has a weight w_j. In addition, tasks may have other characteristics such as time windows, processing delays when switching a task from one machine to another, or precedence constraints. On the other hand, there is a variety of possible parallel machine environments. For example, machines may be specialized to certain subsets of jobs; they may work at the same or different speeds, or they may incur totally unrelated job processing times. Finally, there are several possible objective functions to be minimized, the most prominent being the

Correspondence to: José R. Correa (correa@uchile.cl)

makespan (completion time of the whole schedule) and the sum of weighted completion times of all tasks. As the vast majority of these problems turn out to be NP-hard and therefore difficult to handle, a lot of work has been put in trying to design algorithms providing reasonably close to optimal solutions with limited computational effort.

Recently, distributed environments have emerged as the main architecture for parallel computing. There, as well as in the wine bottling case, a central question is to understand machine scheduling problems where each task is managed by a selfish agent, who is only interested in its own completion time. In particular, it is important to understand the existence, uniqueness, and characteristics of equilibrium when, given some processing rules, each agent seeks to maximize her own objective (say, jobs minimize their own completion time). A lot of attention has also been put in studying the "price of anarchy" [14], that relates, in the worst case scenario, the "social cost" of such a game with the efficient allocation of resources [19, Part III].

More precisely, in such a distributed environment, each job j selfishly minimizes its own completion time C_j, by choosing a machine to be processed on. To this end the job takes into account the scheduling policy used by machines, that is, the order in which a machine processes the jobs assigned to it, and the fact that the other jobs also seek to minimize their own completion times. In this article, we study the price of anarchy induced by a classic machine policy, namely the *weighted shortest processing time* or WSPT order (a.k.a., Smith ratio), which sorts jobs in decreasing order of their weight to processing time ratios, and the social objective is to minimize the sum of weighted job completion times. We consider the restricted uniform parallel machine case, in which each machine has a speed $\sigma_i > 0$, each job j has an amount $q_j > 0$ of work to be processed, and each $p_{i,j}$ is either q_j/σ_i or infinity. The resulting centralized optimization problem $RQ||\sum w_jC_j$ on restricted uniform parallel machine is NP-hard. However, we know from Smith [23] that, given any assignment of jobs to machines, it is optimal for each machine to process its jobs in *WSPT order*, that is, in order of nonincreasing weight-to-work ratios w_j/q_j. In this article, we assume that all ties in this WSPT ratio are broken in a consistent way across all machines, according to a common total order (consistent with WSPT). For such a policy, a pure strategy Nash equilibrium is guaranteed to exist, as it is also the case for many other machine policies, see for example, Immorlica et al. [12], Durr and Thang [8], Lu and Yu [15], Azar et al. [2].[1] In particular Heydenreich, Müller and Uetz [11] use a result by Immorlica et al. and show that the set of pure Nash equilibria is the set of schedules that can be obtained by list scheduling (LS) in WSPT order. Therefore, the (pure) price of anarchy equals the approximation ratio of the corresponding scheduling heuristic. LS heuristics, which are widely used both in theory and practice, consist in ordering the jobs according to some specific criteria and then scheduling them in that order on a machine on which they would complete first. When minimizing the sum of weighted completion times, a natural ordering is WSPT. The resulting LS heuristic, also known as Smith's rule, is frequently used in practice and has been widely studied in the literature for a variety of scheduling environments.

In the wine-bottling example above certain products have characteristics, for example, sparkling wines, or the use of a traditional cork or a screw cap, that may restrict their bottling to a subset of lines; this may be modeled with restricted uniform parallel machines. Recall that each product manager independently chooses a bottling line for his product. The price of anarchy bounds, for any set of products (with any processing requirements, i.e., batch volumes, and any weights) the relative benefit, in terms of total deferral costs, that can be gained by centralizing the bottling decisions. The price of anarchy may then be used in strategic decision making, for example, to rule out centralization in case its maximum possible benefit does not outweigh its disadvantages, some of which are organizational and hard to quantify.

Most literature dealing with selfish scheduling take makespan as social cost [5]. The resulting bounds on the price of anarchy are only constant for simple machine environments such as when machines are identical. However, in more complex situations most known machine policies do not achieve a constant price of anarchy. Indeed, Azar et al. [2], and Fleischer and Svitkina [10] show that, even for a restricted uniform machines environment and unit jobs, that is, $p_{i,j}$ is either $1/\sigma_i$ or infinity, no "reasonable" deterministic machine policy can achieve a constant price of anarchy for the makespan objective. The existence of a randomized machine policy with such a desirable property is unknown.

In contrast, we consider as social cost the sum of the agent costs, a standard definition of welfare in economics [16]. In our setting the agents are the selfish jobs, and job j's cost is its weighted completion time w_jC_j. As opposed to the makespan social cost case, we show that the price of anarchy is bounded above by a small constant, independent of the instance. Specifically, we show that the price of anarchy is exactly 2 for the class of unit jobs, unit speed instances where the finite $p_{i,j}$ values define the edge set of a forest with the machines as nodes. Furthermore, we exhibit instances showing that any deviation from these assumptions may lead to a price of anarchy larger than 2. In particular, when the instance is *job-matchable*, meaning that in the optimal assignment every machine gets a single job, we prove that the price of anarchy is between $2+\sqrt{2}$ and $2+\sqrt{3}$. Finally, for the general case of mixed (i.e., randomized) job strategies, as well as

[1] However Azar et al. exhibit simple machine policies that do not admit pure-strategy Nash equilibria on restricted uniform parallel machines.

for pure strategies, and restricted uniform machines we prove that the price of anarchy is exactly 4, which constitutes the first bound in this general setting with the sum of weighted completion times objective.

A related result was obtained by Farzad et al. [9], which applied to our context states that the price of anarchy for pure strategies in restricted uniform machines and when jobs' weights equal their processing work (i.e., $w_j = q_j$), is at most $3 + 2\sqrt{2} \approx 5.83$. Although their bound is weaker, it also applies to atomic selfish routing.

As mentioned above we consider mixed (i.e., randomized) strategies as well as pure strategies. Our motivation for doing so is twofold. First, mixed strategies are often considered in game theory (as is well known, Nash equilibria exist in mixed strategies under fairly general conditions) and they are considered explicitly in the seminal work of Koutsoupias and Papadimitriou [14] on the price of anarchy. Indeed, in competitive situations, for example, when having to share limited resources such as machines, players may want to use mixed strategies to avoid being taken advantage of by other players. Second, our results and methods apply to mixed strategies, with minor changes from the case of pure strategies. We then obtain that, for the situations considered herein, the addition of mixed strategies does not lead to any deterioration in the price of anarchy. This result, related to Roughgarden's work [21], however does not hold for other scheduling situations, as shown in section 2.2.

Our general bound of 4 is related to *greedy* load balancing as well. In this context, jobs arrive online according to some order σ and have to be scheduled on a machine so as to minimize a certain objective function. A result of Awerbuch et al. [1] states that if jobs with arbitrary processing times $p_{i,j}$ are scheduled online so as to locally minimize the square of the L_2 norm of the machine loads then the resulting schedule has squared norm within a factor of $(1 + \sqrt{2})^2 = 3 + 2\sqrt{2}$ of optimal. In restricted identical parallel machines, that is, $p_{i,j} \in \{p_j, +\infty\}$, this algorithm is equivalent to placing jobs on the least loaded machine and therefore the outcome is a pure strategy Nash equilibrium. Thus, the result in [1], together with the fact that all pure Nash equilibria can be obtained by LS in WSPT order [11], imply that the price of anarchy for pure strategies in restricted identical parallel machines when in addition $w_j = p_j$, is at most $3 + 2\sqrt{2}$, a bound which also follows from Farzad et al. [9]. Still in the case of restricted identical parallel machines, Suri et al. [24] improved the latter bound, to $2 + \sqrt{5} \approx 4.24$ and also provided a lower bound of 3.08, while Caragiannis et al. [4] provided a lower bound of 4 and an upper bound of $(2/3)\sqrt{21} + 1 \approx 4.06$. Interestingly as our result applies for arbitrary weights and the WSPT order, we may consider $w_j = p_j$ so that the sum of weighted completion times is half the square of the norm L_2 of the machine loads, plus the constant $\sum p_j^2/2$. As these weights amount to considering an arbitrary order as a WSPT order, our main result closes the gap left in [4]. On the other hand, Suri et al. proved that the competitiveness of greedy load balancing on restricted uniform machines is 17/3 while Caragiannis et al. proved that this bound is tight. Our general bound of 4 thus shows that (even mixed) Nash equilibria perform better than greedy assignments on restricted uniform machines.

In summary, the main contributions of this work are:

1. We provide an in-depth study of the efficiency of equilibria when the social objective is the sum of the agent costs, that is, the weighted sum of job completion times, in a restricted uniform parallel machines environment. In contrast with the case of the makespan social cost, for which the price of anarchy is not bounded by a constant in most environments, we obtain small constant bounds in our setting. This demonstrates the weighted sum of job completion times objective is, quite naturally, better aligned with the individual job objectives than the makespan.
2. Whereas most existing work, with a few notable exceptions such as the seminal work of Koutsoupias and Papadimitriou [14] and that of Czumaj and Vöcking [6], focus on pure strategy equilibria, our analysis and general results apply as well to mixed strategies.
3. From a classical machine scheduling perspective our results imply small constant bounds for the performance ratio of WSPT LS, an area where the most notable known result is Kawaguchi and Kyan's $(1 + \sqrt{2})/2$ bound for identical parallel machines [13]. We also close the gap in the analysis of greedy load balancing in Caragiannis et al. [4].

The paper is organized as follows. After formally defining the problem and establishing some preliminary results in section 2, we determine in section 3 the price of anarchy for the class of unit-job forest-restricted unit-speed instances. For the general setting, we derive in section 4 lower bounds on the cost of a mixed strategy profile, based on expected machine loads. In section 5, we compute the price of anarchy for a natural extension of forest-restricted instances, while in section 6 we derive our general upper bound of 4, together with a matching lower bound.

2. NOTATIONS AND PRELIMINARY RESULTS

An instance of the Restricted Uniform Machines scheduling environment includes a set N of n "selfish" jobs and a set M of m machines. Each job $j \in N$ has $q_j > 0$ units of work to perform nonpreemptively on a single machine to be determined; a nonempty set $M_j \subseteq M$ of machines that can perform this work; and a weight $w_j > 0$ which reflects its "social importance." Each machine $i \in M$ can process at most one job at a time, and has speed $\sigma_i > 0$, such that if job j is processed on machine $i \in M_j$ then its processing time is $p_{i,j} = q_j/\sigma_i$. By extension we let $p_{i,j} = +\infty$ if $i \notin M_j$. There is a total order $<$ on the job set N such that if jobs $j(1) < j(2) < \cdots < j(K)$ have chosen to be processed on machine i, then this machine will process them in this order, starting at time 0 and without any inserted idle time. As a result their completion times will be $C_{j(1)} = p_{i,j(1)}$ and $C_{j(k)} = C_{j(k-1)} + p_{i,j(k)}$ for $k = 2, \ldots, K$. All this information is known to all jobs (and machines).

In this article, we assume that the total order $<$ on the jobs is *WSPT-consistent*, that is, $j < k$ implies $w_j/q_j \geq w_k/q_k$. Thus, in the scheduling game, the jobs are the players, and M_j is the set of pure strategies available to job j. Each job $j \in N$ chooses a machine $I(j) \in M_j$ on which it will be processed in order to minimize its own completion time.

2.1. Existence of Pure Strategy Nash Equilibria

In a *pure strategy profile*, every job j is assigned to a machine $i \in M_j$. A pure strategy profile is a Nash equilibrium if no job has a unilateral incentive to move to another machine and reduce its completion time in the resulting schedule. If the total order is WSPT-consistent then a pure strategy Nash equilibrium is guaranteed to exist (see e.g., Immorlica et al. [12], Durr and Thang [8], Azar et al. [2], Heydenreich et al. [11]). The next result shows that a pure strategy Nash equilibrium always exists for unit-speed machines.

LEMMA 1: There exists a pure strategy Nash equilibrium for unit speed machines (all $\sigma_i = 1$), even with machine-dependent total orders.

PROOF: Consider the following algorithm. Initially all jobs are unassigned. Repeat the following process until all jobs are assigned. Choose a machine i with the least current load among those that can process a currently unassigned job. Let this machine i choose its preferred unassigned job j, and assign j to i.

We now argue that the resulting assignment is a Nash equilibrium. Consider any job j and the partial schedule just before j is assigned. At that date, every machine $h \in M_j$ has scheduled jobs that it strictly prefers to j. The choice of machine i, with the least current load L_i, to which j is assigned guarantees that j cannot improve its completion time $C_j = L_i + p_{i,j} = L_i + q_j$ by moving to another machine $h \in M_j$ since $L_h \geq L_i$. □

The following example, adapted from Azar et al. [2], shows that this lemma does not hold when the machines have different speeds, even if they use a WSPT order but break ties differently.

EXAMPLE 1: There are 4 jobs, $N = \{A, B, C, D\}$, and three machines, $M = \{1, 2, 3\}$, with $M_A = \{1\}$, $M_B = \{1, 2\}$, $M_C = \{1, 3\}$, and $M_D = \{1, 2\}$. The machine speeds are $\sigma_1 = 1$, and $\sigma_2 = 1/6$, $\sigma_3 = 1/4$, and the processing requirements are $q_A = 20$, $q_B = 2$, $q_C = 7$, and $q_D = 5$. The weights satisfy $w_j = q_j$, so any order is a WSPT order.

Consider a pure Nash equilibrium for the case where machine 1 uses the order $A - B - C - D$ and machine 2 uses the order $D - B$. If job B is assigned to machine 1, then C is assigned to machine 3, and therefore, job D to machine 1, which implies that B would switch to machine 2. Therefore B is assigned to machine 2. Job C now goes to machine 1, and thus D goes to machine 2, which implies that B would switch back to machine 1. Thus a pure Nash equilibrium cannot exist.

REMARK: A pure Nash equilibrium need not exist on unrelated machines, where each machine i uses the "local WSPT order" of nonincreasing $w_j/p_{i,j}$ ratios, even when there are no ties in these ratios. To see this define the processing times $p_{i,j}$ by slightly perturbing those in Example 1, so that the machine orders remain the same but all $w_j/p_{i,j}$ ratios are different. The same argument as in Example 1 shows that no pure Nash equilibrium exists.

2.2. Mixed Strategies

Any job j may also use a *mixed strategy* defined by probabilities $x_{\cdot,j} = (x_{i,j})_{i \in M}$ where $x_{i,j} \geq 0$ is the probability that job j chooses machine i. Thus $x_{i,j} = 0$ if $i \notin M_j$ and $\sum_{i \in M} x_{i,j} = 1$. The matrix $x = (x_{i,j})_{i \in M, j \in N}$ is a (mixed) *strategy profile* if it satisfies these conditions for all $j \in N$. Job j's *rivals' profile* x_{-j} is the submatrix of x with columns indexed by $N \setminus \{j\}$. Slightly abusing notations we may write $x = (x_{\cdot,j}, x_{-j})$.

For every realization of the job choices, each machine then performs the jobs that chose it, according to the total order $<$ as described above, resulting in the job completion times C_j for all $j \in N$. Each job seeks to minimize its expected completion time $E_x C_j$. Since each machine processes the jobs that chose it in the order induced by $<$, we have

$$
\begin{aligned}
E_x C_j &= \sum_{i \in M} x_{i,j} E_{x_{-j}}[C_j | I(j) = i] \\
&= \sum_{i \in M} x_{i,j} E_{x_{-j}} \left[\sum_{k<j} p_{i,k} \mathbf{1}_{\{I(k)=i | x_{-j}\}} + p_{i,j} \right] \\
&= \sum_{i \in M} x_{i,j}/\sigma_i \left(q_j + \sum_{k<j} x_{i,k} q_k \right) \qquad (1)
\end{aligned}
$$

where the *indicator function* $\mathbf{1}_A$ equals 1 when event A occurs and 0 otherwise.

Given the rivals' profile x_{-j} a *best response* for j is any mixed strategy $x_{\cdot,j}$ which minimizes $E_{(x_{\cdot,j},x_{-j})}C_j$. A strategy profile x is a *Nash equilibrium*, abbreviated NE, if $x_{\cdot,j}$ is a best response to x_{-j} for every j. Thus x is a NE if no job j could strictly improve its expected completion time by unilaterally deviating from its current strategy $x_{\cdot,j}$.

The social objective we consider is to minimize the weighted sum $\sum_{j\in N} w_j C_j$ of job completion times. Let $\mathrm{cost}(x) = \sum_{j\in N} w_j E_x C_j$ denote the expected social objective value for strategy profile x, and let $\mathrm{OPT}(J)$ denote the optimum value of the social objective for instance J of this scheduling problem. The *price of anarchy* for instance J is

$$\begin{aligned}\mathrm{PoA}(J) &= \min\{\alpha : \mathrm{cost}(x) \le \alpha\,\mathrm{OPT}(J) \text{ for all NE } x \\ &\qquad \text{for instance } J\} \\ &= \max\{\mathrm{cost}(x) : x \text{ is a NE for instance } J\}/\mathrm{OPT}(J).\end{aligned}$$

The price of anarchy for an instance class $\mathcal{J}$ is $\mathrm{PoA}(\mathcal{J}) = \sup\{\mathrm{PoA}(J) : J \in \mathcal{J}\}$. We also define the pure price of anarchy $\mathrm{PPoA}(J)$ of an instance J as the price of anarchy defined by considering only *pure* strategy NE. Of course, the latter definitions depend of the total order considered, which we assume to be WSPT-consistent.

For the instance classes for which we determine the exact price of anarchy (Theorems 4, 8, and 9) we find that it coincides with the pure price of anarchy, an observation related to results of Roughgarden [21]. The following example however shows that this is not always the case, even for natural instance classes such as that of identical parallel machines. Indeed, for $P||\sum w_j C_j$ the pure price of anarchy equals $(1+\sqrt{2})/2$ (Kawaguchi and Kyan [13]). For the special case of m jobs with all $p_j = w_j = 1$, to be scheduled on m identical parallel machines, the mixed strategy in which every job chooses every machine with probability $1/m$ is an equilibrium with cost approaching $3m/2$ as m grows to infinity, while the optimum cost is m. Thus the price of anarchy is at least $3/2 > (1+\sqrt{2})/2$.

Let $\mathbf{1}_i$ denote the unit column vector in $\mathbb{R}^M$ with entry 1 in row i and 0 elsewhere, and let

$$B(j \mid x_{-j}) = \left\{ i \in M_j : \mathbf{1}_i \in \arg\min_{x_{\cdot,j}} E_{(x_{\cdot,j},x_{-j})}C_j \right\}$$

denote the set of pure strategies that are best responses to x_{-j}. It is well known (Nash [18]) that the set of all best responses to x_{-j} is the convex hull $\mathrm{conv}B(j \mid x_{-j})$. Thus if x is a NE then, for every j, every pure strategy $\mathbf{1}_i$ in the support $\mathrm{supp}(x_{\cdot,j}) = \{i \in M : x_{i,j} > 0\}$ is a best response to x_{-j}, that is,

$$x_{i,j} > 0 \quad \Longrightarrow \quad E_{(\mathbf{1}_i,x_{-j})}C_j = E_x C_j. \tag{2}$$

2.3. Two Basic Results

The Essential Machines Lemma below allows us, when determining the Price of Anarchy of a class $\mathcal{J}$ of instances, to restrict attention to a subclass of instances where each job may only be assigned to the machines corresponding to a worst NE or to a given optimum schedule.

LEMMA 2 (The Essential Machines Lemma): Consider an instance J of the restricted uniform parallel machines problem $RQ||\sum w_j C_j$ with a given total order $<$ on the jobs. Let x be a NE with worst social objective for instance J, i.e., such that $\mathrm{cost}(x) = \max\{\mathrm{cost}(x') : x' \text{ is a NE for instance } J\}$. Let $y \in \mathbb{B}^{M\times N} = \{0,1\}^{M\times N}$ denote the machine-job assignment matrix of a social optimum schedule. Then there exists an instance J' identical to J except that the machine set for each job j is $M'_j = \mathrm{supp}(x_{\cdot,j}) \cup \mathrm{supp}(y_{\cdot,j})$ (and thus the processing times are $p'_{ij} = p_{ij}$ if $i \in M'_j$, and $+\infty$ otherwise), and such that $\mathrm{PoA}(J') \ge \mathrm{PoA}(J)$.

PROOF: Given J, x and y, define J' as stated in the Lemma, that is, with machine set $M' = M$ and speeds $\sigma'_i = \sigma_i$; job set $J' = J$ and weights $w'_j = w_j$; machine sets $M'_j = \mathrm{supp}(x_{\cdot,j}) \cup \mathrm{supp}(y_{\cdot,j})$; and the same total order $<' = <$ on the jobs. On one hand, from a social objective point of view, note that every feasible schedule for J' is feasible for J and has same objective value. Since the optimum schedule used to define y is feasible for J' it follows that it is also optimum for J', and thus $\mathrm{OPT}(J') = \mathrm{OPT}(J)$. On the other hand, x is also a strategy profile for J' and, since the best response sets $B'(j|x_{-j})$ in J' satisfy $B'(j|x_{-j}) \subseteq B(j|x_{-j})$, x is also a NE for J'. Therefore $\mathrm{PoA}(J') \ge \mathrm{cost}'(x)/OPT(J') = \mathrm{cost}(x)/OPT(J) = \mathrm{PoA}(J)$. □

REMARK: Recall that the pure price of anarchy $\mathrm{PPoA}(J)$ of an instance J is the price of anarchy defined by considering only *pure* strategy NE. When considering the *pure* price of anarchy, we may, by Lemma 2, restrict attention to instance classes where each job may be processed on at most 2 machines, one corresponding to a given pure NE with worst social objective value, and the other (possibly the same machine) to a given social optimum schedule. Such instances induce a graph $G = (M, N)$ with the machines as nodes and the jobs as edges, where the edge induced by job j is M_j, i.e., it connects the two machines in M_j if $|M_j| = 2$, and is a loop if $|M_j| = 1$. A pure strategy, i.e., a choice of a machine in M_j by every job j, corresponds to an orientation of every edge in G toward the machine chosen by the corresponding job. A pure NE is then an orientation of the edge set in which no edge will decrease its completion time by swapping its orientation.

The next result reduces the uniform machines case with general-weight social objective $\sum_j w_j C_j$ and any

WSPT-consistent total order, to the special case with the work-weighted social objective $\sum_j q_j C_j$, that is, where $w_j = q_j$, and an arbitrary total order. Note that every total order is WSPT-consistent for the work-weighted objective $\sum_j q_j C_j$.

LEMMA 3 (The Work-As-Weight Lemma): Let $\mathcal{J}$ be a class of instances of the restricted uniform parallel machines problem, closed under taking job subsets (i.e., if $J' \subset J \in \mathcal{J}$ then $J' \in \mathcal{J}$) and with $\mathrm{PoA}(\mathcal{J}) \le \alpha$ for the work-weighted social objective $\sum_j q_j C_j$ and every total order on the jobs. Then $\mathrm{PoA}(\mathcal{J}) \le \alpha$ for the general-weigth social objective $\sum_j w_j C_j$ and every WSPT-consistent total order on the jobs.

PROOF: Fix an instance $J \in \mathcal{J}$ and job weights w_j for all $j \in N$. Let $v_1 > v_2 > \cdots > v_H > 0 =: v_{H+1}$ denote the distinct values of the WSPT ratios w_j/q_j for all $j \in N$. For $h = 1, \ldots, H$ let J_h denote the restriction of J to all jobs $j \in N_h := \{j \in N : w_j/q_j \ge v_h\}$. For every j let $H(j) := \min\{h \in \{1, \ldots, H\} : j \in N_h\}$ so we can write $w_j/q_j = v_{H(j)} = \sum_{h \ge H(j)}(v_h - v_{h+1})$ and, for any $C \in \mathbb{R}^N$,

$$\sum_{j\in N} w_j C_j = \sum_{j\in N}\left(\sum_{h\ge H(j)}(v_h - v_{h+1})\right) q_j C_j = \sum_{h=1}^{H}(v_h - v_{h+1}) \sum_{j\in N_h} q_j C_j.$$

Let x be a NE for instance J with social objective $\sum_j w_j C_j$. By linearity of the expectation,

$$\sum_{j\in N} w_j E_x C_j = \sum_{h=1}^{H}(v_h - v_{h+1}) \sum_{j\in N_h} q_j E_x C_j.$$

Consider any $h \in \{1, \ldots, H\}$ and $j \in N_h$. By Equation (1), $E_x C_j$ does not depend on $x_{\cdot,\ell}$ for any $\ell > j$ (in the total order) and therefore does not depend on $x_{\cdot,\ell}$ for any $\ell \notin N_h$. It follows that the restriction x^h of x to instance J_h is a NE for J_h.

By the assumptions on class $\mathcal{J}$, the price of anarchy for instance $J_h \in \mathcal{J}$ and work-weighted social objective $\sum_{j\in N} q_j C_j$ is at most α. Let C^* denote the completion time vector of a social optimum schedule for instance J, that is, with $\sum_{j\in N} w_j C_j^* = \mathrm{OPT}(J)$. Note that C^* induces a feasible schedule for every instance J_h and thus $\sum_{j\in N_h} q_j E_{x^h} C_j \le \alpha \sum_{j\in N_h} q_j C_j^*$. Therefore

$$\begin{aligned}\sum_{j\in N} w_j E_x C_j &= \sum_{h=1}^{H}(v_h - v_{h+1}) \sum_{j\in N_h} q_j E_{x^h} C_j \\ &\le \sum_{h=1}^{H}(v_h - v_{h+1})\alpha \sum_{j\in N_h} q_j E_{x^h} C_j^* \\ &= \alpha \sum_{j\in N} w_j E_x C_j^* = \alpha \mathrm{OPT}(J).\end{aligned}$$

Then the price of anarchy for instance class $\mathcal{J}$ and social objective $\sum_{j\in N} w_j C_j$ is at most α. □

With the previous result at hand, for the purpose of studying the price of anarchy, we can reduce to instances satisfying $w_j = q_j$, and for which machines process jobs in an arbitrary order (but the same order for every machine). This is exactly what we do in the remainder of the article, where the price of anarchy always refers to the weighted sum of completion times social objective and any WSPT-consistent order.

3. THE PRICE OF ANARCHY FOR FOREST-RESTRICTED UET INSTANCES

A *unit execution times* (UET) instance is one where each $p_{i,j} \in \{1, +\infty\}$. Thus we may assume, w.l.o.g., that all work requirements are $q_j = 1$ and all speeds are $\sigma_i = 1$. In this section, we consider UET instances where (i) each job may be processed on at most two machines and, (ii) an optimal schedule assigns at most one job to each machine. Restriction (i) is motivated by the remark following Lemma 2. Restriction (ii) is equivalent to considering *1-forest-restricted* UET instances, where an instance is *1-forest-restricted* if the resulting graph G is a 1-forest, that is, each connected component contains at most one cycle. (Similarly in a *forest-restricted* instance the resulting graph G is a forest.) The latter holds since under restriction (ii) the number of nodes in any connected component of G is at least the number of edges it contains, for otherwise more than one job would be assigned to some machine in any feasible schedule.

Note that the scheduling problem $RP|\text{1-forest}|\sum w_j C_j$ is trivial for a 1-forest-restricted instance on equal-speed parallel machines: In each connected component orient all edges in the cycle equally, say clockwise, and orient every other edge away from the cycle; as a result every job j is the unique job on its chosen machine and $C_j = p_j$. Thus $\mathrm{OPT}(J) = \sum_{j\in N} w_j p_j$ for every 1-forest-restricted instance J with unit-speed parallel machines. It follows that $\mathrm{OPT}(J) = \sum_{j\in N} w_j$ for every 1-forest-restricted UET instance J.

THEOREM 4 (PoA for 1-Forest-Restricted UET Instances): The price of anarchy for the class $\mathcal{F}$ of all 1-forest-restricted UET instances satisfies $\mathrm{PoA}(\mathcal{F}) = 2 = \mathrm{PPoA}(\mathcal{F})$.

PROOF: We first prove that $\mathrm{PoA}(\mathcal{F}) \le 2$. By the Work-As-Weight Lemma 3 it suffices to prove that $\mathrm{PoA}(J) \le 2$ for the subclass of all 1-forest-restricted UET instances J with unit weights, $w_j = 1$ for all $j \in N$ and an arbitrary total order $<$ on the jobs. Therefore $\mathrm{OPT}(J) = n$ for every such instance J. We prove that $\mathrm{cost}(x) < 2n$ for all NE x, by induction on the number n of edges in the instance $J \in \mathcal{F}$. This trivially holds for $n = 1$, where $\mathrm{cost}(x) = 1$. Thus assume that the inductive assumption $\mathrm{cost}(x') < 2\,(n-1)$ holds for all NE x' in all instances $J' \in \mathcal{F}$ with at most $n-1$ jobs, and consider an n-job instance $J \in \mathcal{F}$ and a NE x. If the graph G associated with J has $K \geq 2$ connected components, with edge sets $N_1, \ldots, N_K$, then let $J_1, \ldots, J_K$ be the corresponding (sub)instances. Note that every $J_k \in \mathcal{F}$. Since these instances are completely independent, the restriction x^k of x to each J_k is a NE for J_k and $\mathrm{cost}(x) = \sum_{k=1}^{K} \mathrm{cost}(x^k)$. If every $|N_k| \leq n-1$ then, by the inductive assumption, $\mathrm{cost}(x^k) < 2\,|N_k|$ and thus $\mathrm{cost}(x) < 2\sum_{k=1}^{K} |N_k| = 2n$, establishing the requisite induction. Thus we may assume that some $|N_k| = n$, that is, all edges in J are in the same connected component, the other components (if any) being isolated machines. We may eliminate these isolated machines, and the graph G associated with J then reduces to a single 1-tree with n edges (jobs) and n or $n+1$ nodes (machines).

So we restrict our attention to a connected graph $G = (V, E)$ containing at most one cycle. If G contains a cycle, denoted by $\mathcal{C} \subseteq E$, then for every edge $e \in E \setminus \mathcal{C}$ let $h(e)$ and $\ell(e)$ be the vertices which are closer and farther from $\mathcal{C}$, respectively; otherwise, fix an arbitrary root $r \in V$ and for every edge $e \in E$ let $h(e)$ and $\ell(e)$ be the vertices which are closer and farther from r, respectively. Suppose that for some $e \in E \setminus \mathcal{C}$ we have that $x_{\ell(e),e} = 1$. Then e is assigned to vertex $\ell(e)$ both in the NE and in an optimal solution. We can thus remove e from the graph and replace it by a loop at vertex $\ell(e)$, disconnecting the graph, but not affecting the total cost of either the NE or the optimal solution, and the result follows by the inductive hypothesis. Thus, we assume that $x_{h(e),e} > 0$ for all $e \in E \setminus \mathcal{C}$. Consider now some $e \in \mathcal{C}$, and an adjacent vertex v. If $x_{v,e} = 1$, then e is assigned to vertex v both in the NE and in an optimal solution, so we can replace e by a loop at vertex v without changing the cost of either solution. Thus, we assume also that $\mathcal{C}$ is simply a loop or in a NE all edges in $\mathcal{C}$ choose both their adjacent vertices with positive probability.

For an edge $e \in E$ and a vertex $v \in V$ let $L(v, e) = \sum_{f<e} x_{v,f}$ be the expected load on vertex v due to edges with higher priority than e. The NE condition says that for every edge $e = (u, v) \in \mathcal{C}$ we have $L(v, e) = L(u, e)$ and for $e \in E \setminus \mathcal{C}$ we have $L(h(e), e) \leq L(\ell(e), e)$. Now,

$$
\begin{aligned}
\mathrm{cost}(x) &= \sum_{e \in E\setminus\mathcal{C}} (L(h(e), e) + 1) + \sum_{e=(u,v)\in\mathcal{C}} (L(v, e) + 1) \\
&\leq n + \sum_{e \in E\setminus\mathcal{C}} L(\ell(e), e) + \sum_{e=(u,v)\in\mathcal{C}} L(v, e) \\
&\leq n + \sum_{v \in V} (\deg(v) - 1) \leq 3n - m \leq 2n.
\end{aligned}
$$

The second inequality follows since in the definition of $L(v, e)$, e is not counted in the summation.

To show the converse inequality, we construct a sequence $J^K \in \mathcal{F}$ of work-weighted, forest-restricted UET

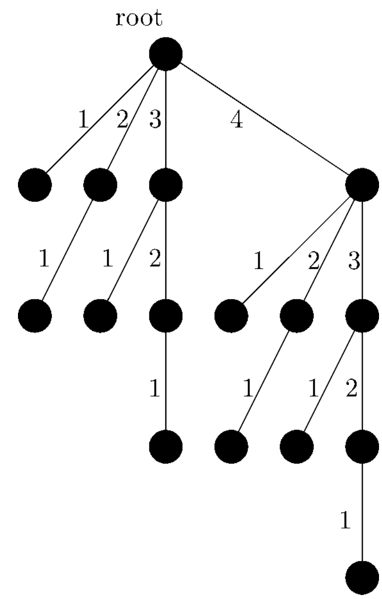

Figure 1. Rooted tree T^4 with job levels shown next to the corresponding edges.

instances, adapted from Davis and Jaffe [7], such that $\lim_{K\to+\infty} \mathrm{PPoA}(J^K) = 2$. The graph T^K associated with J^K is a rooted tree with node set $\{1, \ldots, 2^K\}$, node 1 as *root*, and K is the maximum number of edges in a path connecting the root to a leaf. Each edge j in T^K has a *level* $\ell(j) \in \{1, \ldots, K\}$ defined below. Rooted tree T^1 consists of the single edge $\{1, 2\}$, with root 1 and edge level $\ell(\{1, 2\}) = 1$. Inductively, let $\tilde{T}^{K-1}$ be a copy of T^{K-1} with the same edge levels but node labels shifted by 2^{K-1}, that is, with node set $\{2^{K-1}+1, \ldots, 2^K\}$. Then T^K is constructed by joining T^{K-1} and $\tilde{T}^{K-1}$ using the new edge $\{1, 2^{K-1}+1\}$ with edge level $\ell(\{1, 2^{K-1}+1\}) = K$; see Fig. 1 for the case $K = 4$. Finally the total order $<$ on the edges (jobs) is any order such that if edge $k \neq j$ is on the path between j and the root then $j < k$.

Note that J^K contains 2^K machines, $2^K - 1$ jobs, and $\mathrm{OPT}(J^K) = 2^K - 1$. There are 2^{K-k} edges (jobs) with level k for every $k = 1, \ldots, K$. Also note that, by construction, if a node i has degree k in T^K then the k edges incident to i have distinct labels $1, 2, \ldots, k$. Consider the pure strategy profile x^K defined by orienting every edge toward the root. Then, in the schedule induced by x^K, every job (edge) j has completion time $C_j = \ell(j)$ and cannot improve its completion time by switching from its current upward machine to its other, downward machine (where it would also get completion time $\ell(j)$). Therefore x^K is a NE. Its social objective value is $\mathrm{cost}(x^K) = \sum_{k=1}^{K} 2^{K-k}k = 2^{K+1} - K - 2$. Thus, $2 \geq \lim_{K\to+\infty} \mathrm{PPoA}(J^K) \geq \lim_{K\to+\infty} \mathrm{cost}(x^K)/\mathrm{OPT}(J^K) = 2$, and equality must hold. □

The following examples show that each of the properties defining instance class $\mathcal{F}$, that is, unit-work jobs; 1-forest structure; and equal-speed machines, is necessary to obtain a price of anarchy of 2.

EXAMPLE 2: Modify instance J^{10} defined in the proof of Theorem 4 into instance J' by adding a loop job at the root with $p_1 = 4$, and keeping $p_j = 1$ for all other jobs, still with work-weights $w_j = p_j$ for all jobs j. Then $\mathrm{OPT}(J') = p_1 + (2^{10} - 2) = 1038$. The strategy profile x^{10} defined in the same proof is still a NE, as we only modified the processing time of the very last job in the corresponding schedule. Its social objective value is now: $\mathrm{cost}'(x^{10}) = (\mathrm{cost}(x^{10}) - 10) + p_1 * (9 + p_1) = 2078 > 2\,\mathrm{OPT}(J')$, hence $\mathrm{PPoA}(J') > 2$.

EXAMPLE 3: Modify instance J^4 defined in the proof of Theorem 4 into instance J'' by adding 3 loop jobs j with $M_j = \{1\}$ to the root, and putting these 3 loop jobs at the end of the total order. A feasible (and optimum) schedule is obtained by orienting all 2-node edges downward, and processing the loop jobs on the root machine, so $\mathrm{OPT}(J') \le (2^4 - 1) + (1 + 2 + 3) = 21$. The strategy profile x^4 defined earlier, augmented with assigning the 3 loop jobs to the root machine, still defines a NE $\tilde{x}^4$, as we only modified the end of the schedule. Its social objective value is: $\mathrm{cost}''(\tilde{x}^4) = \mathrm{cost}(x^4) + (5 + 6 + 7) = 44 > 2\,\mathrm{OPT}(J'')$, hence $\mathrm{PPoA}(J'') > 2$.

EXAMPLE 4: Modify the instance J'' of Example 3 by adding 3 very slow machines, such that each of the formerly loop jobs is now an ordinary 2-machine job that can be processed on the root machine 1 and on its own very slow machine. The resulting instance is now a forest-restricted (actually, tree-restricted) instance with unit-work jobs and work-weights, and different speeds. When the newly added machines are so slow that they are never used in a NE, the augmented strategy profile $\tilde{x}^4$ has the same objective value 44, more than twice the social optimum value 21.

REMARK: In sharp contrast with the result of Theorem 4, the price of anarchy with respect to the makespan social objective can grow logarithmically with the size of the instance, even for the simple class $\mathcal{F}$ of all 1-forest-restricted UET instances. Indeed, for the instances J^K defined in the proof above, the root machine has load $K = \log_2(n)$ in the strategy profile x^K, whereas the minimum makespan is 1.

4. MACHINE LOADS AND LOWER BOUNDS ON EXPECTED COST

We now go back to considering a general instance with restricted uniform machines and all $w_j = q_j$ in order to obtain a lower bound on the cost of any strategy profile. Recall that a strategy profile x is a matrix such that $x_{i,j} \ge 0$ for all $i \in M, j \in N$, $x_{i,j} = 0$ if $i \notin M_j$, and $\sum_{i \in M} x_{i,j} = 1$ for all $j \in N$. For a strategy profile x of an instance J, let $L_i(x) = \sum_{j \in N} x_{i,j} q_j$ denote the *expected load* of machine $i \in M$.

LEMMA 5 (Lower Bound Lemma): The social objective value of any strategy profile x satisfies

$$\mathrm{cost}(x) \ge \frac{1}{2} \sum_{i \in M} 1/\sigma_i \sum_{j \in N} x_{i,j}\, q_j^2 + \frac{1}{2} \sum_{i \in M} (L_i(x))^2/\sigma_i. \quad (3)$$

PROOF: By Eq. (1) we have that:

$$\begin{aligned}\mathrm{cost}(x) &= \sum_{j \in N} q_j \sum_{i \in M} x_{i,j}/\sigma_i \left(q_j + \sum_{k<j} x_{i,k} q_k \right) \\ &= \sum_{i \in M} 1/\sigma_i \left(\sum_{j \in N} x_{i,j} q_j^2 + \sum_{j \in N} \sum_{k<j} x_{i,j} q_j x_{i,k} q_k \right).\end{aligned}$$

Using that $x_{i,j} \ge \frac{1}{2} x_{i,j} + \frac{1}{2} x_{i,j}^2$ holds for all $0 \le x_{i,j} \le 1$, we obtain

$$\begin{aligned}\mathrm{cost}(x) &\ge \sum_{i \in M} 1/\sigma_i \left(\frac{1}{2} \sum_{j \in N} x_{i,j} q_j^2 + \frac{1}{2} \left[\sum_{j \in N} x_{i,j} q_j \right]^2 \right) \\ &= \frac{1}{2} \sum_{i \in M} 1/\sigma_i \sum_{j \in N} x_{i,j}\, q_j^2 + \frac{1}{2} \sum_{i \in M} (L_i(x))^2/\sigma_i\end{aligned}$$

as claimed. □

REMARK: When x is a pure strategy profile, i.e., $x_{i,j} \in \{0, 1\}$ for all i, j, then inequality (3) simplifies to the machine capacity constraint:

$$\sum_{j \in N} q_j C_j \ge \sum_{i \in M} 1/\sigma_i \left(\frac{1}{2} \sum_{j \in J_i(x)} q_j^2 + \frac{1}{2} \left[\sum_{j \in J_i(x)} q_j \right]^2 \right),$$

where $J_i(x) = \{j \in J : x_{i,j} > 0\}$ is the set of jobs that choose machine i. This inequality also follows from the single machine capacity constraints of Wolsey [25] and Queyranne [20] applied to uniform machine scheduling. Note that inequality (3) actually holds with equality when x is a pure strategy profile; indeed, the resulting schedule incurs no unnecessary idle time.

If, on the contrary, $x_{i,j}$ is fractional, then the term $x_{i,j} q_j$ can be interpreted as the expected processing time of job j in machine i, where $p_{i,j} = q_j$ with probability $x_{i,j}$ and $p_{i,j} = 0$ with probability $1 - x_{i,j}$. In this setting and for a single machine with unit speed ($m = 1 = \sigma_1$), Lemma 5 presents some analogies with the stochastic single machine conservation law of Bertsimas and Niño-Mora [3]

$$\sum_{j \in J_i(x)} E[p_{i,j}]E[C_j] \geq \frac{1}{2} \sum_{j \in J_i(x)} E[p_{i,j}]^2 + \frac{1}{2} \left(\sum_{j \in J_i(x)} E[p_{i,j}] \right)^2.$$

It differs, however, in the left hand side as the completion times $C_{i,j}$ of job j need not be identical on the different machines. Also it has similarities with the stochastic parallel machine conservation law of Möhring, Schulz and Uetz [17]

$$\sum_{j \in N)} E[p_j]E[C_j] \geq \frac{1}{2m} \sum_{j \in N} E[p_j]^2 + \frac{1}{2} \left(\sum_{j \in N} E[p_j] \right)^2 - \frac{m-1}{2m} \sum_{i \in N} \mathrm{Var}[p_j],$$

although in their setting it is the processing time, rather than the schedule, which is random.

Defining the vector $\lambda(x) \in \mathbb{R}^M$ as $\lambda_i(x) = L_i(x)/\sqrt{\sigma_i}$, Lemma 5 implies:

COROLLARY 6: For any strategy profile x,

$$||\lambda(x)||^2 \leq 2 \operatorname{cost}(x). \tag{4}$$

Furthermore, if all speeds $\sigma_i = 1$ then

$$||\lambda(x)||^2 \leq 2 \operatorname{cost}(x) - \sum_{j \in N} q_j^2. \tag{5}$$

For the case of pure strategy profiles (i.e., integral assignments) the first summand in inequality (3) can be evaluated explicitly to obtain a slightly stronger inequality:

COROLLARY 7: For any pure strategy profile x,

$$||\lambda(x)||^2 \leq 2 \operatorname{cost}(x) - \sum_{j \in N} \frac{q_j^2}{\sigma_{i(j,x)}}, \tag{6}$$

where $i(j, x)$ is the machine in which j is processed under the assignment x.

5. JOB-MATCHABLE INSTANCES

In this section, we analyze a natural extension of the forest-restricted UET instances, namely *job-matchable*, in which the optimal solution assigns at most one job to each machine. We prove that the price of anarchy for mixed strategies for job-matchable unit-speed instances with the $\sum w_j C_j$ social objective and a WSPT-consistent total order is exactly $2 + \sqrt{2} \approx 3.414$, and at most $2 + \sqrt{3}$ for general speed job-matchable instances.

THEOREM 8: The price of anarchy for the class of job-matchable instances is at most $2 + \sqrt{3}$. In addition, the price of anarchy for the class of job-matchable instances on unit-speed machines is exactly $2 + \sqrt{2}$, and is equal to the pure price of anarchy.

PROOF: By the Work-As-Weight Lemma 3 it suffices to prove that $\mathrm{PoA}(J) \leq 2 + \sqrt{3}$ for the subclass of all work-weighted instances J, that is, with weights, $w_j = q_j$ for all $j \in N$, and an arbitrary total order $<$ on the jobs. Let $y \in \mathbb{B}^{M \times N}$ denote the machine-job assignment matrix of a social optimum schedule and let $x \in \mathbb{R}^{M \times N}$ denote a NE.

Since x is NE, for every job $j \in N$, we have that

$$E_x C_j \leq \frac{L_i(x) + q_j}{\sigma_i} = \frac{L_i(x) + L_i(y)}{\sigma_i}$$

where $i = i(j, y)$ is the machine processing j in the optimum schedule y. Multiplying the previous inequality by $q_j = L_i(y)$, and summing over all j we obtain

$$\operatorname{cost}(x) \leq \sum_{j \in N} \frac{L_{i(j,y)}(x)L_{i(j,y)}(y) + [L_{i(j,y)}(y)]^2}{\sigma_{i(j,y)}}$$
$$= \operatorname{cost}(y) + \sum_{i \in M} \frac{L_i(x)L_i(y)}{\sigma_i},$$

since the instance is job-matchable. By the Cauchy-Schwartz inequality it follows that $\operatorname{cost}(x) - \operatorname{cost}(y) \leq \sum_{i \in M} \lambda_i(x)\lambda_i(y) \leq ||\lambda(x)|| \cdot ||\lambda(y)||$. Corollary 6 states that $||\lambda(x)|| \leq \sqrt{2\operatorname{cost}(x)}$, while, since the instance is job-matchable and work-weighted, Corollary 7 says that $||\lambda(y)|| \leq \sqrt{2\operatorname{cost}(y) - \sum_{j \in N} w_j(q_j/\sigma_{i(j,y)})} = \sqrt{\operatorname{cost}(y)}$. Thus, $\operatorname{cost}(x) - \operatorname{cost}(y) \leq \sqrt{2\operatorname{cost}(x)\operatorname{cost}(y)}$, which is a quadratic equation leading to $\operatorname{cost}(x) \leq (2 + \sqrt{3})\operatorname{cost}(y)$. This concludes the proof of the first part of the result.

Assume that in addition machines have unit speed, that is, $\sigma_i = 1$ for all $i \in M$. This immediately implies that the cost of an optimal assignment is exactly $\sum_{j \in N} q_j^2$, and from (5) we get

$$||\lambda(x)||^2 \leq 2\operatorname{cost}(x) - \sum_{j \in N} q_j^2 = 2\operatorname{cost}(x) - \operatorname{cost}(y).$$

With this, following the previous analysis we obtain $\operatorname{cost}(x) - \operatorname{cost}(y) \leq \sqrt{(2\operatorname{cost}(x) - \operatorname{cost}(y))\operatorname{cost}(y)}$, which occurs if and only if $\operatorname{cost}(x) \leq (2 + \sqrt{2})\operatorname{cost}(y)$.

To conclude the proof of the theorem, we construct a sequence of unit-speed job-matchable instances asymptotically achieving a price of anarchy of $2+\sqrt{2}$. Modify each instance J^K defined in the proof of Theorem 4 into a work-weighted forest-restricted instance $\widetilde{J}^K$ with the same tree structure T^K and total order $<$, but with processing times $q_j = 2^{\frac{\ell(j)-K}{2}}$ for all j. Because of the structure of the tree T^K,

$$\mathrm{OPT}(\widetilde{J}^K) = \sum_{j=1}^{2^K-1} q_j^2 = \sum_{k=1}^{K} 2^{K-k}2^{k-K} = K.$$

The strategy profile x^K defined in the same proof is still a NE, as the completion time of every level-k job is $\Gamma_k := \sum_{h=1}^{k} 2^{\frac{h-K}{2}} = (2+\sqrt{2})2^{-\frac{K}{2}}(2^{\frac{k}{2}}-1)$ on each of its two machines. Its social objective value is

$$\begin{aligned}\widetilde{\mathrm{cost}}(x^K) &= \sum_{k=1}^{K} 2^{K-k}2^{\frac{k-K}{2}}\Gamma_k \\ &= (2+\sqrt{2})\left(K-\left(1+\sqrt{2}\right)\left(1+2^{-\frac{K}{2}}\right)\right).\end{aligned}$$

Therefore, $\lim_{K\to\infty} PoA(\widetilde{J}^K) \geq 2+\sqrt{2}$. □

6. GENERAL INSTANCES

In this section, we prove the main result of this article, namely, that the price of anarchy for mixed strategies on restricted uniform machines with the minsum $\sum w_j C_j$ social objective and a WSPT-consistent total order, is exactly 4.

THEOREM 9 (Upper Bound Theorem): The price of anarchy for the class $\mathcal{J}$ (resp., $\mathcal{J}'$) of all restricted uniform machines (resp., UET) instances is $\mathrm{PoA}(\mathcal{J}) = \mathrm{PoA}(\mathcal{J}') = 4 = \mathrm{PPoA}(\mathcal{J}) = \mathrm{PPoA}(\mathcal{J}')$.

PROOF: By the Work-As-Weight Lemma 3 it suffices to prove that $\mathrm{PoA}(J) \leq 4$ for the subclass of all work-weighted instances J, that is, with weights, $w_j = q_j$ for all $j \in N$, and an arbitrary total order $<$ on the jobs. Let $y \in \mathbb{B}^{M\times N}$ denote the machine-job assignment matrix of a social optimum schedule and $x \in \mathbb{R}^{M\times N}$ denote a NE.

As in the proof of Theorem 8, since x is NE, for every job $j \in N$, we have that $E_x C_j \leq (L_i(x)+q_j)/\sigma_i$, where $i = i(j,y)$ is the machine processing j in the optimum schedule y (i.e., $j \in J_i(y)$). Multiplying the previous inequality by q_j and summing over all j we obtain

$$\begin{aligned}\mathrm{cost}(x) &\leq \sum_{j\in N} \frac{L_{i(j,y)}(x)q_j + q_j^2}{\sigma_{i(j,y)}} \\ &= \sum_{i\in M}\sum_{j\in J_i(y)} \frac{L_i(x)q_j}{\sigma_i} + \sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}}.\end{aligned}$$

Using that $\sum_{j\in J_i(y)} q_j = L_i(y)$ for all $i \in M$ we get

$$\begin{aligned}\mathrm{cost}(x) &\leq \sum_{i\in M}\frac{L_i(y)\,L_i(x)}{\sigma_i} + \sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}} \\ &\leq \sum_{i\in M}\lambda_i(x)\lambda_i(y) + \sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}} \\ &\leq ||\lambda(x)||\cdot||\lambda(y)|| + \sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}},\end{aligned}$$

by the Cauchy-Schwartz inequality. Applying Corollaries 6 and 7 to $\lambda(x)$ and $\lambda(y)$, respectively

$$\mathrm{cost}(x) - \sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}} \leq \sqrt{2\mathrm{cost}(x)\left(2\mathrm{cost}(y) - \sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}}\right)}.$$

Squaring and simplifying the inequality we get

$$\begin{aligned}\mathrm{cost}(x)^2 &\leq 4\mathrm{cost}(x)\mathrm{cost}(y) - \left(\sum_{j\in N}\frac{q_j^2}{\sigma_{i(j,y)}}\right)^2 \\ &\leq 4\mathrm{cost}(x)\mathrm{cost}(y)\end{aligned}$$

implying $\mathrm{cost}(x) \leq 4\mathrm{cost}(y)$, which concludes the proof of the first part of the result.

We now exhibit a family J^k (for $k = 1,2,\ldots$) of instances for which the price of anarchy approaches 4, adapted from [4, Theorem 7]. Let m be large enough so that m/i^2 is integer for all $i = 1,\ldots,k$. Consider a UET instance (i.e., restricted unit-speed with unit jobs) with m machines, $\{1,\ldots,m\}$ and jobs $j_{i,\ell}$ for $i = 1,\ldots,k$ and $\ell = 1,\ldots,m/i^2$, such that job $j_{i,\ell}$ can only be processed in machines $\{1,\ldots,\ell\}$. The global priority on the jobs is such that jobs with larger second index have higher priority (and ties are broken arbitrarily).

Note that if for all $i = 1,\ldots,k$ and $\ell = 1,\ldots,m/i^2$, job $j_{i,\ell}$ is assigned to machine ℓ the resulting assignment has m/i^2 jobs finishing at time i, as depicted at the left in Fig. 2. Therefore,

$$\mathrm{OPT}(J^k) \leq m\sum_{i=1}^{k} i/i^2 = m\sum_{i=1}^{k} 1/i.$$

On the other hand, the schedule x^k, resulting from LS jobs in priority order and in which every job selects the machine with smallest index among those that minimize her completion time, is a pure strategy Nash equilibrium. In x^k, jobs $j_{1,m}, j_{1,m-1},\ldots,j_{1,m/2+1}$ complete at time 1 on machines $1,2,\ldots,m/2$, respectively; jobs $j_{1,m/2},\ldots,j_{1,m/4+1}$ complete at time 2 on machines $1,\ldots,m/4$, respectively. Then

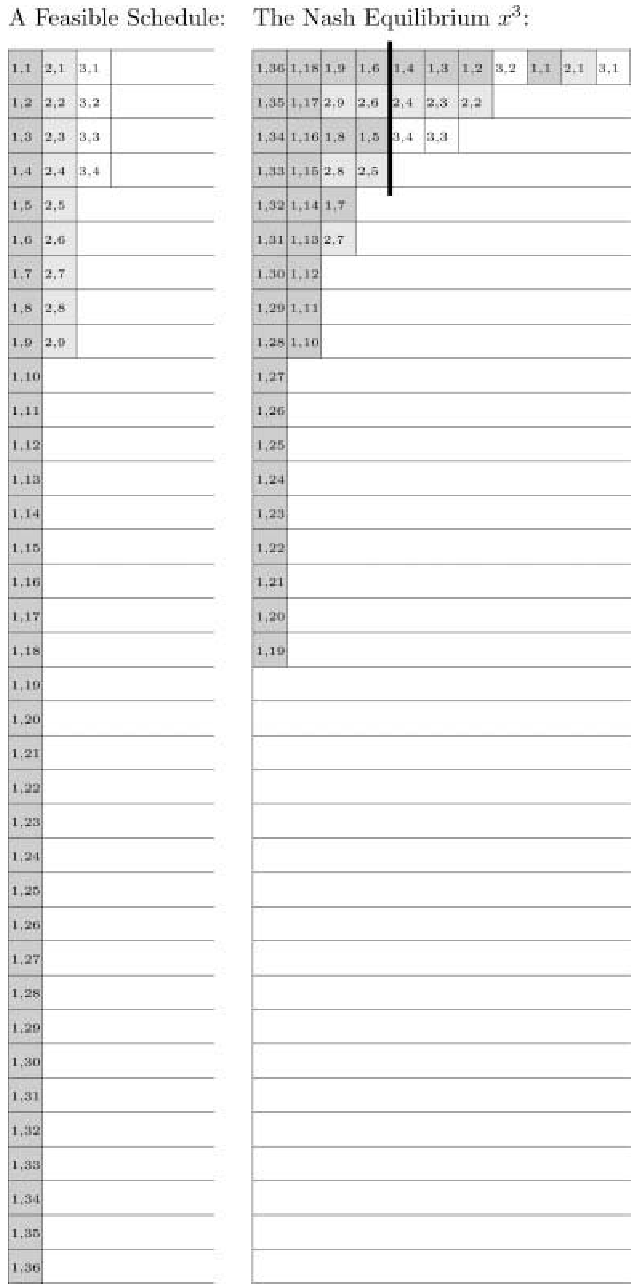

Figure 2. Gantt chart for instance J^3 with $m = 36$ in the proof of Theorem 9. Rows represent machines. The lower bound on $\text{cost}(x^3)$ in Eq. (7) is the total completion time of all jobs to the left of the thick vertical bar.

we can verify by induction that for $i = 2, \ldots, k-1$, the $m/(i(i+1))$ jobs $j_{h,\ell}$ with $\ell = m/i^2, m/i^2 - 1, \ldots, m/(i(i+1))+1$ and $1 \le h \le i$ (in tie-breaking order), complete at time $2i-1$ on machines $1, \ldots, m/(i(i+1))$; and the $m/(i+1)^2$ jobs $j_{h,\ell}$ with $\ell = m/(i(i+1), \ldots, m/(i+1)^2 + 1$ and $1 \le h \le i$, complete at time $2i$ on machines $1, \ldots, m/(i+1)^2$ (see the vertical bar to the right of Figure 2). Therefore,

$$\begin{aligned} \text{cost}(x^k) &\ge \sum_{i=1}^{k-1} m \left(\frac{2i-1}{i(i+1)} + \frac{2i}{(i+1)^2} \right) \\ &= 4m \sum_{i=1}^{k} 1/i - O(m). \end{aligned} \qquad (7)$$

It follows that the ratio between the costs approaches 4 as $k \to \infty$. □

ACKNOWLEDGMENTS

The authors thank Christoph Dürr for ideas that led to the formulation of this problem. They also thank Christoph and Martín Matamala for stimulating discussions on the material in section 3, and Yossi Azar for pointing out reference [10]. We finally thank the Associate Editor and the referees for valuable comments and suggestions. The research of the first author was partially supported by FONDECYT grant 1090050 and by the Millenium Nucleus Information and Coordination in Networks ICM/FIC P10-024F. This research was done while the second author was at Centro de Modelamiento Matemático, Unité Mixte Internationale du CNRS (France), Universidad de Chile.

REFERENCES

[1] A. Awerbuch, A. Azar, E.F. Grove, P. Krishnan, M.Y. Kao, and J.S. Vitter, "Load balancing in the L_p norm," in: Proceedings of 36th Annual IEEE Symposium on Foundations of Computer Science (FOCS 1995), IEEE Comp Soc Press, Milwaukee, Wisconsin, USA, 1995, pp. 383–391.

[2] Y. Azar, K. Jain, and V. Mirrokni, "(Almost) Optimal coordination mechanisms for unrelated machine scheduling," in: Proceedings of 19th ACM-SIAM Symposium on Discrete Algorithms (SODA 2008), ACM-SIAM, San Francisco, 2008, pp. 323–332.

[3] D. Bertsimas and J. Niño-Mora, Conservation laws, extended polymatroids and multi-armed bandit problems: A unified polyhedral approach, Math Oper Res 21 (1996), 257–306.

[4] I. Caragiannis, M. Flammini, C. Kaklamanis, P. Kanellopoulos, and L. Moscardelli, Tight bounds for selfish and greedy load balancing, Algorithmica 61 (2011), 606–637.

[5] G. Christodoulou, E. Koutsoupias, and A. Nanavati, "Coordination mechanisms," in: J. Diaz, J. Karhumäki, A. Lepistö, and D. Sannella (Editors), ICALP 2004, LNCS, vol. 3142, Springer, Heidelberg, 2004, pp. 345–357.

[6] A. Czumaj and B. Vöcking, Tight bounds for worst-case equilibria, ACM Trans Algorithms 3 (2007), 000–00.

[7] E. Davis and J.M. Jaffe, Algorithms for scheduling tasks on unrelated processors, J ACM 28 (1981), 721–736.

[8] C. Dürr and K.T. Nguyen, Non-clairvoyant scheduling games, in: Proceedings of the 2nd International Symposium on Algorithmic Game Theory (SAGT 2009), Paphos, Cyprus. LNCS 5814, Springer, pp. 135–146.

[9] B. Farzad, N. Olver, and A. Vetta, A priority-based model of routing, Chicago J Theor Comput Sci (2008), Article 1.

[10] L. Fleischer and Z. Svitkina, Preference-constrained oriented matching, in: Proceedings of the 7th Workshop on Analytic Algorithmics and Combinatorics (ANALCO 2010), SIAM, Austin, 2010, pp. 56–65.
[11] B. Heydenreich, R. Müller, and M. Uetz, Games and mechanism design in machine scheduling—An introduction, Prod Oper Manage 16 (2007), 437–454.
[12] N. Immorlica, L.(E.) Li, V.S. Mirrokni, and A.S. Schulz, Coordination mechanisms for selfish scheduling, Theor Comp Sci 410 (2009), 1589–1598.
[13] T. Kawaguchi and S. Kyan, Worst case bound of an LRF schedule for the mean weighted flow-time problem, SIAM J Comput 15 (1986), 1119–1129.
[14] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," in: C. Meinel and S. Tison (Editors), STACS 1999, LNCS, vol. 1563, Springer, Heidelberg, 1999, pp. 404–413.
[15] P. Lu and C. Yu, "Worst-case nash equilibria in restricted routing," in: C. Papadimitriou and S. Zhang (Editors), WINE 2008, LNCS, vol. 5385, Springer, Heidelberg, 2008, pp. 231–238.
[16] A. Mas-Colell, M.D. Whinston, and J.R. Green, Microeconomic theory, Oxford University Press, New York, USA, 1995.
[17] R.H. Möhring, A.S. Schulz, and M. Uetz, Approximation in stochastic scheduling: The power of LP-based priority policies, J ACM 46 (1999), 924–942.
[18] J. Nash, Equilibrium points in N-person games, Proc Natl Acad Sci 36 (1950), 48–49.
[19] N. Nissan, T. Roughgarden, E. Tardos, and V. Vazirani, Algorithmic game theory, Cambridge University Press, New York, USA, 2007.
[20] M. Queyranne, Structure of a simple scheduling polyhedron, Math Prog 58 (1993), 263–285.
[21] T. Roughgarden, "Intrinsic robustness of the price of anarchy", in: Proceedings of the 41st annual ACM symposium on Theory of computing (STOC '09), ACM, New York, NY, 2009, pp. 513–522.
[22] G. Sánchez, Programación de Trabajos en Líneas de Envasado y Etiquetado para Viña San Pedro, Mathematical Engineering Thesis, Universidad de Chile, 2009. Available at http://www.cybertesis.cl/tesis/uchile/2009/sanchez_g/sources/sanchez_g.pdf.
[23] W. Smith, Various optimizers for single stage production, Naval Res Log Q 3 (1956), 59–66.
[24] S. Suri, C. Tóth, and Y. Zhou, "Selsh load balancing and atomic congestion games", in: Proceedings of 16th Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA 2004), Barcelona, Spain, 2004, pp. 188–195.
[25] L.A. Wolsey, "Mixed integer programming formulations for production planning and scheduling problems", Invited talk, in: Proceedings of 12th International Symposium on Mathematical Programming, MIT, Cambridge, MA, 1985.