CrossMark

Decentralized subcontractor scheduling with divisible jobs

Behzad Hezarkhani[1,2] · **Wiesław Kubiak**[2]

Abstract Subcontracting allows manufacturer agents to reduce completion times of their jobs and thus obtain savings. This paper addresses the coordination of decentralized scheduling systems with a single subcontractor and several agents having divisible jobs. Assuming complete information, we design parametric pricing schemes that strongly coordinate this decentralized system, i.e., the agents' choices of subcontracting intervals always result in efficient schedules. The subcontractor's revenue under the pricing schemes depends on a single parameter which can be chosen to make the revenue as close to the total savings as required. Also, we give a lower bound on the subcontractor's revenue for any coordinating pricing scheme. Allowing private information about processing times, we prove that the pivotal mechanism is coordinating, i.e., agents are better off by reporting their true processing times, and by participating in the subcontracting. We show that the subcontractor's maximum revenue with any coordinating mechanism under private information equals the lower bound of that with coordinating pricing schemes under complete information. Finally, we address the asymmetric case where agents obtain savings at different rates per unit reduction in completion times. We show that coordinating pricing schemes do not always exist in this case.

Keywords Scheduling · Divisible jobs · Subcontracting · Coordination · Mechanism design

✉ Behzad Hezarkhani
  b.hezarkhani@tue.nl

  Wiesław Kubiak
  wkubiak@mun.ca

1 Department of Industrial Engineering, Eindhoven Technical University, Eindhoven, The Netherlands

2 Faculty of Business Administration, Memorial University, St. John's, NL, Canada

1 Introduction

As a common supply chain management practice, manufacturers take advantage of external resources to alleviate the burden of internal operations through subcontracting and outsourcing. While outsourcing externalizes some internal operations, subcontracting allows manufacturers to carry out jobs both internally and externally (Van Mieghem 1999). In this manner, subcontracting enables a manufacturer to speed up the completion times of his jobs. Instances of subcontracting practices can be found in quick-response industries characterized by volatile demand and inflexible capacities, e.g., metal fabrication industry (Parmigiani 2003), electronics assembly (Webster et al. 1997), high-tech manufacturing (Aydinliyim and Vairaktarakis 2011), textile production, and engineering services (Taymaz and Kiliçaslan 2005).

Although a considerable number of papers in the literature analyzes the subcontracting strategies in production planning and scheduling problems of manufacturers (Kamien and Li 1990; Tan and Gershwin 2004; Chen and Li 2008; Lee and Sung 2008), the subcontractors' scheduling problems have received less attention. In reality, a subcontractor by itself faces a limited capacity while providing service to several manufacturers. Given the time-sensitive nature of subcontracting operations, the subcontractor's schedule has critical impact on the performance of manufacturers as well as the supply chain. The lack of due attention to the subcontractors' operations can cause significant complications in the extended supply chain. A well-documented real-life example of this issue has been reported in Boeing's Dreamliner supply chain where the overloaded schedules of subcontractors, each working with multiple suppliers, resulted in long delays in the overall production due dates (see Vairaktarakis (2013) and the references therein).

An important feature of subcontractor's scheduling problem is that the jobs belong to different manufacturers who are concerned with the processing of their own jobs only.

Unlike *centralized systems* where a single decision maker controls all the relevant decision variables and possesses all the necessary information, the agents in a decentralized system have some control over their individual decisions and/or are privately informed about some aspects of the system. Therefore, in order for the system to achieve a particular objective a mechanism is needed that motivates the agents to make their decisions and reveal their private information in a way that the system's ultimate objective is attained indirectly. In this paper, we focus on the *efficiency* objective, that is, we are seeking mechanisms that result in subcontracting schedules that maximize the total savings obtained by all manufacturers. In other words, we address the problem of *coordination* in decentralized subcontracting systems.

The mechanisms considered in this paper are pricing and payment schemes that the subcontractor announces before the manufacturers choose their most desirable subcontracting intervals. Such mechanisms are in fact common in practice. An example is the online reservation system implemented by the Semiconductor Product Analysis and Design Enhancement (SPADE) center of the Hong Kong University of Science and Technology[1] wherein the semiconductor companies choose service time intervals in a first-come-first-book manner and in consideration of an announced price list for services in different time intervals.

Naturally, the design of mechanisms in the decentralized subcontracting systems must consider the utility of agents, i.e., savings due to subcontracting minus payments, as well as the subcontractor's revenue to ensure that all parties are sufficiently motivated to participate and operate in the system.

In this paper, we study a subcontractor scheduling problem with several manufacturer agents and a single subcontractor that carries out the agents' jobs on a single machine. Of particular interest to this paper is the scenario where a divisible job can be processed simultaneously on the manufacturer's private machine as well as on the subcontractor's machine. The reduction in completion time of an agent's job obtained by such parallel processing provides monetary saving for that agent.

The paper is organized as follows. Section 2 contains a review of related work. In Sects. 3 and 4, we assume complete information and give pricing schemes that coordinate the decentralized system. That is the pricing schemes enforce a choice of subcontracting time intervals that coincides with efficient allocations of the centralized solution

for every agent. In particular, Sect. 3 characterizes efficient allocations of the centralized solution. Section 4 provides sufficient conditions for the existence of coordinating pricing schemes and it introduces a family of pricing schemes that are strongly coordinating. With a strongly coordinating pricing scheme, the efficient allocations in the centralized solutions are *uniquely* optimal for all agents. Moreover, we show a lower bound on the total payments for any coordinating pricing scheme. Finally we show that with the appropriate choice of a single parameter, our proposed pricing schemes enable the subcontractor to obtain a total revenue anywhere between the lower bound and the maximum total savings.

In Sect. 5, we allow the true processing time of each job to be a private information of its agent and we address the intricacies resulting from agents possibly lying by reporting false processing times in order to obtain their preferred subcontracting intervals. In order to achieve efficiency, however, the subcontractor elicits the true processing times of jobs using certain mechanism. We draw upon the class of efficient and incentive compatible mechanisms and on the *pivotal mechanism* (Vickrey 1961; Clarke 1971; Groves 1973) in particular. The underlying reason for this choice is that the pivotal mechanism is the only efficient and incentive compatible mechanism that always results in nonnegative payments from agents to the subcontractor. This is a desirable property as agents must not be paid if their jobs are processed by the subcontractor in the subcontractor scheduling problem. We obtain a simple closed-form formula for the payments in the pivotal mechanism and prove that the pivotal mechanism results in truth-telling being the *unique* optimal choice of all agents except the one scheduled last on the subcontractor's machine who can possibly exaggerate his processing time without affecting anyone's utility. This result is particularly interesting as "uniqueness of equilibrium might be thought of as the exception rather than the rule" (Jackson 2000). Since we also show that the mechanism guarantees that all agents are better off by subcontracting, we actually prove that the pivotal mechanism coordinates the decentralized subcontracting problem under private information. Finally, we show that under any coordinating mechanism, the subcontractor's revenue is equivalent to the lower bound of that with the coordinating pricing schemes. Therefore, coordinated system under private information never generates higher total revenue for the subcontractor than that under complete information.

In Sect. 6, we address the asymmetric case where the agents obtain savings at different rates. The efficient centralized allocations in this case are more difficult to characterize. Moreover, we show that it is impossible to devise a coordinating pricing scheme in general for this case. Section 7 contains the concluding remarks.

---

2 Related work

The decentralized scheduling problems have been considered in cooperative and non-cooperative settings.

In the former, agents (jobs) are able to communicate and coordinate strategies. The early work of Curiel et al. (1989) studies the cost allocation problem in a cooperative game based on the perennial scheduling problem of Smith (1956). A survey of related research is given by Curiel et al. (2004).

In the non-cooperative settings agents choose their strategies individually and in competition with other agents. This paper falls in this category.

Heydenreich et al. (2007) provide an introduction to and a literature review of problems arising in the non-cooperative decentralized scheduling. One important problem pertains to determining the equilibria of individual decisions and the quality of the corresponding solution compared with that of the centralized optimal solution. The pioneering work of Koutsoupias and Papadimitriou (1999), which gave rise to the literature on *price of anarchy*, analyzes the worst-case performance of a decentralized scheduling system with parallel machines in comparison with that in the centralized system.

As the outcomes of a decentralized system depend on the "policies" employed to handle the agents, another important problem addresses a better design of such policies. The *coordination mechanisms* discussed by Christodoulou et al. (2004) seek policies which improve the performance of a decentralized parallel sequencing system—Immorlica et al. (2005) review and extend this line of research. Ideally, such policies could make the decentralized system as efficient as the centralized system. Wellman et al. (2001) study pricing schemes for a scheduling problem that could achieve this goal and show that such pricing schemes might not exist in general. Kutanoglu and Wu (1999) report similar non-existence results for another scheduling problem. Even if the existence of such pricing schemes could be proven, the problem of finding the pricing scheme might be NP-hard (Chen et al. 2004).

Private information further complicates the coordination problem. The well-known *incentive compatible mechanisms* draw upon payment schemes to truthfully elicit the private information of the agents. The Vickery–Clarke–Groves (VCG) mechanisms (Vickrey 1961; Clarke 1971; Groves 1973) characterize all efficient mechanisms that make truthtelling an undominated strategy of all agents. For single machine sequencing, Suijs (1996) shows that there exists no incentive compatible and individually rational mechanism that results in the total payment of zero. Nisan and Ronen (2007) propose payment schemes that guarantee a certain performance for a decentralized scheduling problem for which the payment schemes of VCG mechanisms are NP-hard to compute.

The decentralized subcontracting and outsourcing problems have also been addressed in a number of papers.

Aydinliyim and Vairaktarakis (2010) study a cooperative game where coalitions of agents reschedule their reserved time intervals to obtain savings. They prove non-emptiness of the core and the convexity of the corresponding game. Cai and Vairaktarakis (2012) investigate a problem with overtime and tardiness costs where the subcontractor announces the prices of time intervals before agents book their most preferred time intervals in a first-come-first-book manner. They show the balancedness of the corresponding cooperative game and implement the VCG mechanisms to elicit the private information of agents. Bukchin and Hanany (2007) compare the costs of a scheduling system comprised of multiple capacitated agents and an uncapacitated subcontractor in decentralized and centralized systems. Qi (2012) investigates a subcontractor's pricing problem with a single agent having multiple jobs which can be subcontracted (though not partially) to reduce the tardiness costs.

The subcontracting problem related to the one addressed in this paper was first studied by Vairaktarakis and Aydinliyim (2007) where they compare the performance in decentralized and centralized settings. Building upon the same model, Vairaktarakis (2013) analyzes the outcomes of a decentralized subcontracting system under different protocols announced by the subcontractor. However, both papers assume complete information and neither provides coordinating pricing schemes for the problem. The model considered in this paper generalizes the model of Vairaktarakis and Aydinliyim (2007) by allowing an agent to use more than one interval on the subcontractor's machine. This generalization is critical for a coordinating pricing schemes which need to give the agents freedom in choosing how many intervals on the subcontractor's machine to buy—the Vairaktarakis and Aydinliyim (2007) model would a priori limit this choice to at most a single interval which does not make their model adequate for the study of pricing schemes.

It is worth noticing that the models in Vairaktarakis and Aydinliyim (2007) and Vairaktarakis (2013) borrow from the concept of divisible jobs introduced in the context of job shops by Anderson (1981), and in the context of distributed computer systems scheduling by Bharadwaj et al. (1996), see also Drozdowski (2009) for a more recent review.

The distributed computer systems provide another important application area for the results obtained in this paper, where agents carry out their computations on their private machines as well as buy computational time on shared CPUs with available capacity.

3 The problem

Consider a set of agents $N = \{1, \ldots, n\}$. At time $t = 0$, agent $i \in N$ has a divisible job with processing time $p_i > 0$.[2] Let

---

[2] We refer to i as a job or an agent depending on the context.

$p = (p_1, \ldots, p_n)$ be the vector of processing times. We assume, except in Sect. 5, that p is given. An agent i has its own private machine to do its job in time p_i, besides private machines a subcontractor is available that can process any portion of job i on its machine that can be shared by all agents. Thus, by subcontracting, agent i can reduce the completion time of its job to less than p_i. Let $T_i = \cup_{k=1}^{m_i} T_i^k$ be the total time allocated to agent i consisting of m_i non-overlapping subintervals $T_i^k = [t_i^k, t_i^k + \bar{t}_i^k)$ where $t_i^k \geq 0$ and $\bar{t}_i^k > 0$ are the start time and the duration of the kth subinterval, respectively. The subintervals of agent i are indexed by the order of their start times. An *allocation* $T = \{T_i | i \in N\}$ is a set of allocations T_i on subcontractor's machine. Allocation T is *feasible* if and only if for any i, j, k, and k' the subintervals $[t_i^k, t_i^k + \bar{t}_i^k)$ and $[t_j^{k'}, t_j^{k'} + \bar{t}_j^{k'})$ do not overlap. Let $\mathfrak{T}$ denote the set of all feasible allocations T for the agents in N.

Our model allows *preemptions on subcontractor's machine*, that is the portion of a job allocated to the subcontractor's machine (or the subcontracted part of a job) is allowed to be executed in more than one disjoint time interval on that machine. This renders our model to be more general and arguably closer to real-life than the one in Vairaktarakis and Aydinliyim (2007) where only at most one time interval on subcontractor's machine is allowed for any subcontracted part. Consequently two main decisions need to be made for each job in the model: one is the size of the subcontracted part of a job (this part can be executed on the subcontractor's machine simultaneously with the remaining part of the job executed on private machine—thus the term divisible jobs), the other as to how to execute the subcontracted part (this part can possibly be executed in several disjoint time intervals—thus the term preemptions on the subcontractor's machine).

The saving obtained by agent i from an allocation T_i is calculated recursively as follows. Assume that initially, agent i uses its private machine only in the interval $[0, p_i], i \in N$. Take the earliest interval allocated to i, T_i^1. If the start time $t_i^1 < p_i$, then a portion of the remainder of job i done after t_i^1, i.e., $p_i - t_i^1$, on i's private machine can be transferred to the subcontractor. The most efficient way to do this is for agent i to split the remainder equally between its private and subcontractor's machines, unless the duration of the interval is too short (see Fig. 1). If the duration of the interval is too short, i.e., $\bar{t}_i^1 < (p_i - t_i^1)/2$, then T_i^1 is fully utilized by i on the subcontractor's machine. Therefore, by utilizing the allocated interval T_i^1, agent i can reduce the finish time of its job by an amount equal to

$$v_i(T_i^1) = \max \left\{ \min \left\{ \frac{p_i - t_i^1}{2}, \bar{t}_i^1 \right\}, 0 \right\}.$$

Clearly, the allocation T_i cannot be utilized at all by agent i if $t_i^1 \geq p_i$. The saving obtained by the next interval can be calculated in the same manner considering that the new
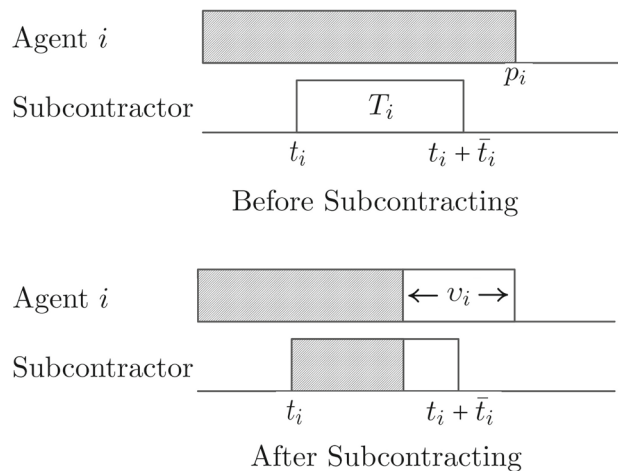

Fig. 1 Valuation in subcontracting

processing time of job i is $p_i - v_i(T_i^1)$ on i's private machine. Thus the saving due to the kth subinterval T_i^k to agent i is obtained recursively by

$$v_i(T_i^k) = \max \left\{ \min \left\{ \frac{p_i - \sum_{l=1}^{k-1} v_i(T_i^l) - t_i^k}{2}, \bar{t}_i^k \right\}, 0 \right\} \tag{1}$$

total saving due to the allocation T_i is calculated by summing up the savings obtained by all of its subintervals:

$$v_i(T_i) = \sum_{k=1}^{m_i} v_i(T_i^k). \tag{2}$$

If $T_i = \{[t_i, \bar{t}_i)\}$ is a single interval, the Eq. (2) simplifies to $v_i(T_i) = \max \{ \min \{ (p_i - t_i)/2, \bar{t}_i \}, 0 \}$. The *total saving* of a feasible allocation T is the sum of savings of all agents, that is $v(T) = \sum_{i \in N} v_i(T_i)$.

3.1 Characterization of efficient (centralized) allocations

The objective of the centralized problem is to find efficient allocations on the subcontractor machine for N. An *efficient allocation* $\mathcal{T}$ is a feasible allocation that maximizes the total saving, that is

$$\mathcal{T} \in \arg \max_{T \in \mathfrak{T}} v(T). \tag{3}$$

Let $\mathcal{T} = \{\mathcal{T}_i | i \in N\}$ be an efficient allocation. We have the following two simple observations that hold for any efficient allocation.

Observation 1 *For $i \in N$, job i does not finish on the subcontractor's machine later than on agent's i private machine.*

Observation 2 *The subcontractor's machine is never idle when some agent's private machine is busy.*

We now prove that *all* efficient allocations are non-preemptive.

Lemma 1 *No efficient allocation on subcontractor's machine is preemptive.*

Proof By contradiction. Let $\mathcal{T}$ be an efficient and preemptive allocation and let $n' \leq n$ be the number of agents with allocations on the subcontractor's machine. We show that then there exists another feasible allocation with higher total saving. Suppose that $m_i > 1$ for some $i \in N$. Without loss of generality we take the largest such i. Let $\mathcal{T}_j^{m_j} = [\mathfrak{t}_j^{m_j}, \mathfrak{t}_j^{m_j} + \bar{\mathfrak{t}}_j^{m_j})$ be the allocation immediately preceding $\mathcal{T}_i^{m_i}$. We have $i \neq j$, and $\mathfrak{t}_j^{m_j} + \bar{\mathfrak{t}}_j^{m_j} = \mathfrak{t}_i^{m_i}$ by Observation 2. Consider the following modification: [Step 1] delay $\mathcal{T}_i^{m_i-1}$ by $\mathfrak{t}_i^{m_i} - (\mathfrak{t}_i^{m_i-1} + \bar{\mathfrak{t}}_i^{m_i-1})$; speed up all the allocations between $\mathcal{T}_i^{m_i-1}$ and $\mathcal{T}_i^{m_i}$ by $\bar{\mathfrak{t}}_i^{m_i-1}$. Thus job i would be preempted one less time. By Observation 1 this modification does not change the completion time of any job. Therefore, total saving remains unchanged. [Step 2] Increase the duration of the last allocation of j by $\bar{\mathfrak{t}}_i^{m_i-1}/2$, so that by Observation 1 the private machine of agent j finishes earlier by $\bar{\mathfrak{t}}_i^{m_i-1}/2$. For $k = i, \ldots, n'$, i.e., jobs including and after i, start the last allocation of k by $\bar{\mathfrak{t}}_i^{m_i-1}/2^{k-i+1}$ later and finish it $\min\left\{\bar{\mathfrak{t}}_i^{m_i-1}/2^{k-i+2}, \bar{\mathfrak{t}}_k\right\}$ later on the subcontractor's machine (observe that if $\bar{\mathfrak{t}}_i^{m_i-1}/2^{k-i+2} \geq \bar{\mathfrak{t}}_k$ then k will no longer be executed on the subcontractor's machine) and $\min\left\{\bar{\mathfrak{t}}_i^{m_i-1}/2^{k-i+2}, \bar{\mathfrak{t}}_k\right\}$ later on agent k private machine.

The resulting allocation is feasible. Furthermore, jobs in $N \setminus \{j, i, \ldots, n'\}$ complete as before Step 2, and for the jobs in $\{j, i, \ldots, n'\}$ the total saving increases by

$$\frac{\bar{\mathfrak{t}}_i^{m_i-1}}{2} - \frac{\bar{\mathfrak{t}}_i^{m_i-1}}{4} - \cdots - \min\left\{\frac{\bar{\mathfrak{t}}_i^{m_i-1}}{2^{n'-i+2}}, \bar{\mathfrak{t}}_{n'}\right\}$$
$$\geq \frac{\bar{\mathfrak{t}}_i^{m_i-1}}{2} - \frac{\bar{\mathfrak{t}}_i^{m_i-1}}{4} - \cdots - \frac{\bar{\mathfrak{t}}_i^{m_i-1}}{2^{n'-i+2}} > 0.$$

Hence, the alternative allocation has a higher total saving which contradicts the efficiency of $\mathcal{T}$. □

The lemma *excludes* allocations with preemptions on subcontractor's machine from the set of efficient allocations. This result is key for our coordinating pricing scheme in Sect. 4 since the efficient allocations with preemptions on subcontractor's machine could make the existence of coordinating pricing scheme questionable or possibly more difficult to prove.

By Lemma 1, $m_i = 1$ for $i \in N$ in any efficient allocation $\mathcal{T}$ that is $\mathcal{T}_i = [\mathfrak{t}_i, \mathfrak{t}_i + \bar{\mathfrak{t}}_i)$ for $i \in N$. Vairaktarakis and

Aydinliyim (2007) observe that, for any efficient allocation $\mathcal{T}$ with $m_i = 1$, $i \in N$, each job finishes simultaneously on its own private and subcontractor's machines, that is

$$p_i - \bar{\mathfrak{t}}_i = \mathfrak{t}_i + \bar{\mathfrak{t}}_i \text{ for } i \in N. \tag{4}$$

They also show that for $m_i = 1$, $i \in N$, the agents' efficient allocations $\mathcal{T}_i$ are ordered in non-decreasing order of p_i. Thus, we assume hereafter in this section and in Sect. 4 that N is arranged in the non-decreasing order of processing times. In this manner, job i would be sequenced in ith position on subcontractor's machine. Finally, by Observation 2, there is no idle time on the subcontractor machine in efficient allocations, thus we have

$$\mathfrak{t}_1 = 0 \text{ and } \mathfrak{t}_i = \mathfrak{t}_{i-1} + \bar{\mathfrak{t}}_{i-1} \text{ for } i > 1. \tag{5}$$

The unique solution of the recursive equations (4) and (5) is given in the following theorem.

Theorem 1 (Efficient allocation) *(Vairaktarakis and Aydinliyim 2007) For efficient allocations with $m_i = 1$ for $i \in N$, we have*

$$\bar{\mathfrak{t}}_i = \frac{p_i}{2} - \sum_{k=1}^{i-1} \frac{\bar{\mathfrak{t}}_k}{2} = \frac{p_i}{2} - \sum_{k=1}^{i-1} \frac{p_k}{2^{i+1-k}}. \tag{6}$$

By this theorem

$$\upsilon_i(\mathcal{T}) = \upsilon_i(\mathcal{T}_i) = \bar{\mathfrak{t}}_i \tag{7}$$

and thus, $\upsilon(\mathcal{T}) = \sum_{i \in N} \bar{\mathfrak{t}}_i$. Equations (6) and (5) define vectors of efficient durations $\bar{\mathfrak{t}} = (\bar{\mathfrak{t}}_1, \ldots, \bar{\mathfrak{t}}_n)$ and start times $\mathfrak{t} = (\mathfrak{t}_1, \ldots, \mathfrak{t}_n)$ on subcontractor's machine. Finally, we let $\mathfrak{t}_{n+1} = \mathfrak{t}_n + \bar{\mathfrak{t}}_n$ denote the completion time of the efficient allocations, i.e., the makespan of efficient allocations.

Though multiple efficient allocations exist as long as there are jobs with equal processing times, in each one of them the jobs in the same position on subcontractor's machine start at the same time and have the same efficient durations. This observation is key to the coordinating pricing scheme developed in the next section.

4 Strongly coordinating pricing schemes

In contrast to a centralized system where allocations are chosen *for* the agents so as to maximize the total saving, in a decentralized system the decisions are made *by* the agents in a distributed fashion. Though the agents are self-interested they need, by definition of coordinating mechanism, to collectively converge to an efficient allocation. The convergence

depends on a mechanism used in the collective decision making. In this section we design such a mechanism based on a pricing scheme. A *pricing scheme* is a function q, defined on $t \geq 0$, which determines a price $q(t) \geq 0$ for acquiring the time t on the subcontractor's machine by any agent. The mechanism based on a pricing scheme q works as follows. The subcontractor announces its pricing scheme, and subsequently agent i buys T_i in a first-come-first-serve manner. The agent makes its decision so as to maximize its utility, which for a given q, and assuming quasilinear utilities, equals

$$u_i^q(T_i) = v_i(T_i) - \pi^q(T_i), \tag{8}$$

where

$$\pi^q(T_i) = \int_{t \in T_i} q(t)\mathrm{d}t \tag{9}$$

is the agent i's payment for T_i made to the subcontractor. We require q to be a locally integrable function so that (9) always exists.

Though each agent maximizes its utility with respect to q, the choice of q must ensure coordination in the decentralized system. That is q must guarantee that the agents choose their subcontracting intervals in the exact same manner as in some efficient allocation $\mathcal{T}$. We call such pricing schemes *coordinating*. Let $\mathcal{E}$ be the set of all efficient allocations. We formally define.

Definition 1 *(Coordinating pricing scheme)* The pricing scheme q is coordinating if for any $\mathcal{T} = (\mathcal{T}_1, \ldots, \mathcal{T}_n)$, $\mathcal{T} \in \mathcal{E}$, and any $T = (T_1, \ldots, T_n)$, $T \in \mathfrak{T} \setminus \mathcal{E}$,

$$\pi^q(\mathcal{T}_i) - \pi^q(T_i) \leq v_i(\mathcal{T}_i) - v_i(T_i) \tag{10}$$

for each $i \in N$. The pricing scheme q is *strongly* coordinating if (10) holds strictly for each $i \in N$.

A coordinating pricing scheme results in the situation where no agent would be worse off by choosing an efficient allocation. However, the implementation of a coordinating pricing scheme may not necessarily result in the efficiency of the system. This is due to the possibility that an agent chooses an interval which is not a part of any efficient allocation. Such a choice could hinder forthcoming agents to select their efficient allocations. Therefore, the strong coordination requires all agents to exclusively choose efficient allocations.

A natural class of pricing schemes to consider for scheduling consists of those with non-increasing q where agents have to pay a higher price for earlier intervals. When the choice of agent i consists of single interval only, i.e., $T_i = [t_i, t_i + \bar{t}_i)$, we let $\pi^q(T_i) = \pi^q(t_i, t_i + \bar{t}_i)$. We now show that the class of non-increasing pricing schemes contain coordinating pricing schemes.

Theorem 2 (Sufficient conditions) *A non-increasing pricing scheme q is coordinating if $q(0) < 1$ and the following two conditions are met:*

C1. For any $i = 2, \ldots, n$ and every ϵ, $0 < \epsilon \leq \bar{t}_{i-1}/2$:

$$\pi^q(t_i - 2\epsilon, t_i) - \pi^q(t_{i+1} - \epsilon, t_{i+1}) \geq \epsilon; \tag{11}$$

C2. For any $i = 1, \ldots, n-1$ and every ϵ, $0 < \epsilon \leq \bar{t}_i/2$, and for $i = n$ and every ϵ, $0 < \epsilon \leq \bar{t}_n$:

$$\pi^q(t_i, t_i + 2\epsilon) - \pi^q(t_{i+1}, t_{i+1} + \epsilon) \leq \epsilon. \tag{12}$$

Proof All $\mathcal{T} \in \mathcal{E}$ define the same start times $t = (t_1, \ldots, t_n)$ and durations $\bar{t} = (\bar{t}_1, \ldots, \bar{t}_n)$ as shown in the previous section.

The individual rationality requires that for $0 \leq t < t_{n+1}$, $q(t)$ must be less than 1, otherwise jobs would be either better off by not selecting the time with subcontracting price exceeding one or indifferent if the price equals one. Since q is non-increasing, the individual rationality holds if $q(0) < 1$.

For q to be a coordinating pricing scheme, it must be that no agent can deviate from an efficient allocation and improve its utility. The first-come-first-booked order breaks ties between jobs with equal processing times, if any, and picks a unique efficient allocation $\mathcal{T}$ from $\mathcal{E}$. In $\mathcal{T}$ agent i has the utility $v_i(\mathcal{T}_i) - q(\mathcal{T}_i)$. The fact that $t_i + 2\bar{t}_i = p_i$ and q is non-increasing, implies that agent i cannot choose another interval to increase its valuation and, at the same time, reduce its payment. In order to improve its utility, the agent i may be able to choose another interval to: (a) increase its valuation as well as payment such that the added valuation is greater than the additional payment, or (b) decrease its valuation as well as payment such that the saving in payment is greater than the reduction in valuation. We enforce conditions on q such that neither (a) nor (b) could possibly happen for any job.

(a) Suppose that the agent i chooses T_i in a way that $v_i(T_i) - v_i(\mathcal{T}_i) = \epsilon$, for some $\epsilon > 0$. Note that this is not an option for $i = 1$. As q is non-increasing, T_i has the cheapest payment if it starts as late as possible. Therefore, $T_i = [t_i - 2\epsilon, t_{i+1} - \epsilon)$ is the cheapest alternative interval for agent i which results in ϵ improvement in its valuation. Note that i cannot have negative start time, thus $\epsilon \leq t_i/2$. From Definition 1 it follows that for the pricing scheme q to be coordinating, it must hold for any $i = 2, \ldots, n$ and for every $0 < \epsilon \leq t_i/2$ that

$$\pi^q(t_i - 2\epsilon, t_{i+1} - \epsilon) - \pi^q(t_i, t_{i+1}) \geq \epsilon.$$

By the definition of π^q in (9), the last inequality can be rewritten as (11) (see Fig. 2). We use induction and show that the latter would be the case if for any $i = 2, \ldots, n$, (11)
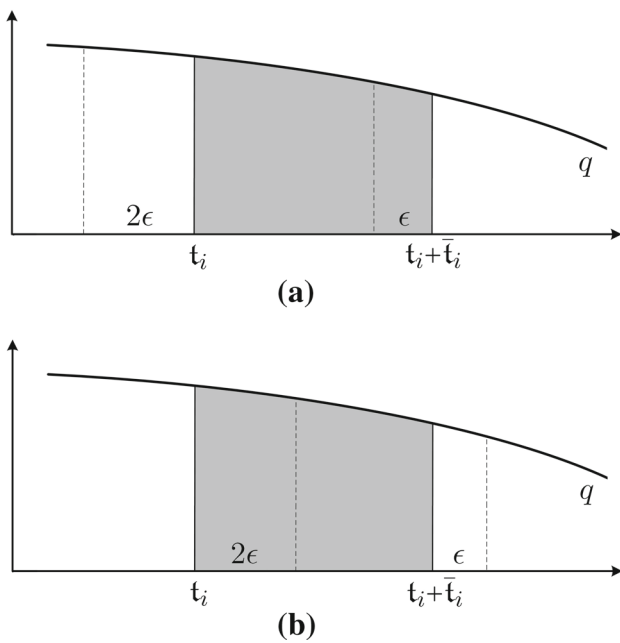
Fig. 2 Non-increasing pricing schemes

holds for every $0 < \epsilon \le \bar{t}_{i-1}/2$, i.e., the condition stated in C1. This holds trivially for $i = 2$ as $t_2 = \bar{t}_1$. Fix $i = 3, \ldots, n$ and suppose that for j, $1 \le j < i$, (11) holds for every ϵ such that

$$0 < \epsilon \le \sum_{k=1}^{j} \bar{t}_{i-k}/2.$$

We show that (11) holds as well for every ϵ such that,

$$0 < \epsilon \le \sum_{k=1}^{j+1} \bar{t}_{i-k}/2.$$

Observe that this would be the case if for every ϵ' such that $0 < \epsilon' \le \bar{t}_{i-j-1}/2$ we have,

$$\pi^q \left(t_{i-j} - 2\epsilon', t_{i-j} \right) - \pi^q \left(t_{i+1} - \sum_{k=1}^{j} \bar{t}_{i-k}/2 - \epsilon', t_{i+1} - \sum_{k=1}^{j} \bar{t}_{i-k}/2 \right) \ge \epsilon'. \tag{13}$$

As q is non-increasing, we have

$$\pi^q \left(t_{i+1} - \sum_{k=1}^{j} \bar{t}_{i-k}/2 - \epsilon', t_{i+1} - \sum_{k=1}^{j} \bar{t}_{i-k}/2 \right) \le \pi^q \left(t_{i-j+1} - \epsilon', t_{i-j+1} \right)$$

Thus, (13) would be the case if for every ϵ' such that $0 < \epsilon' \le \bar{t}_{i-j-1}/2$ we have,

$$\pi^q \left(t_{i-j} - 2\epsilon', t_{i-j} \right) - \pi^q \left(t_{i-j+1} - \epsilon', t_{i-j+1} \right) \ge \epsilon'.$$

Condition C1 for the job $i - j$ ensures that the latter holds. Therefore, if for any $i = 2, \ldots, n$, (11) holds (strictly) for every $0 < \epsilon \le \bar{t}_{i-1}/2$, then (11) holds (strictly) for every $0 < \epsilon \le \bar{t}_i/2$.

(b) It is straightforward to see that the interval which reduces the valuation of the agent i by $\epsilon > 0$, while having the largest decrease in its payment, is $T_i = (t_i + 2\epsilon, t_{i+1} + \epsilon)$, as long as $\epsilon \le \bar{t}_i$. Therefore, for q to be coordinating it must hold for every $i = 1, \ldots, n$ and for every ϵ such that $0 < \epsilon \le \bar{t}_i$ that,

$$\pi^q \left(t_i, t_{i+1} \right) - \pi^q \left(t_i + 2\epsilon, t_{i+1} + \epsilon \right) \le \epsilon.$$

By the definition of π^q in (9), the last inequality can be rewritten as (12). We use induction to show that for $i = 1, \ldots, n-1$ the latter would be the case if (11) holds for every $0 < \epsilon \le \bar{t}_i/2$, i.e., the condition stated in C2. Fix $i = 1, \ldots, n-1$ and suppose that for j, $1 \le j < i$, (12) holds for every ϵ such that,

$$0 < \epsilon \le \min \left\{ \sum_{k=1}^{j} \bar{t}_{i+k-1}/2, \bar{t}_i \right\}.$$

We show that (12) holds as well for every ϵ such that,

$$0 < \epsilon \le \min \left\{ \sum_{k=1}^{j+1} \bar{t}_{i+k-1}/2, \bar{t}_i \right\}.$$

In case $\sum_{k=1}^{j} \bar{t}_{i+k-1}/2 < \bar{t}_i$, observe that the latter would hold if for every $0 < \epsilon' \le \min \{\bar{t}_{i+j}/2, \bar{t}_i\}$ we have,

$$\pi^q \left(t_{i+j}, t_{i+j} + 2\epsilon' \right) - \pi^q \left(t_{i+1} + \sum_{k=1}^{j} \bar{t}_{i+k-1}/2, t_{i+1} + \sum_{k=1}^{j} \bar{t}_{i+k-1}/2 + \epsilon' \right) \le \epsilon'. \tag{14}$$

As q is non-increasing, we have

$$\pi^q \left(t_{i+1} + \sum_{k=1}^{j} \bar{t}_{i+k-1}/2, t_{i+1} + \sum_{k=1}^{j} \bar{t}_{i+k-1}/2 + \epsilon' \right) \ge \pi^q \left(t_{i+j+1}, t_{i+j+1} + \epsilon' \right)$$

Thus, (14) would be the case if for every ϵ' such that $0 < \epsilon' \leq \min\{\bar{t}_{i+j}/2, \bar{t}_i\}$ we have

$$\pi^q\left(t_{i+j}, t_{i+j} + 2\epsilon'\right) - \pi^q\left(t_{i+j+1}, t_{i+j+1} + \epsilon'\right) \leq \epsilon'.$$

Condition C2 for the job $i + j$ ensures that the latter holds. Therefore, if for $i = 1, \ldots, n-1$, (12) holds (strictly) for every $0 < \epsilon \leq \bar{t}_i/2$, then (12) holds (strictly) for every $0 < \epsilon \leq \bar{t}_i$. □

The conditions in Theorem 2 are designed to make the deviations from $t = (t_1, \ldots, t_n)$ undesirable for all agents in the sense that the agent's utilities would be reduced if they choose any intervals other than those defined by t. Clearly with quasilinear utilities any agent can increase its utility either by choosing an interval that either increases its valuation (saving due to reduction in completion time) or reduces its payment (see Fig. 2). With a non-increasing pricing scheme attaining both of these at the same time is impossible. Therefore, a deviation which increases (decreases) an agent's valuation, simultaneously increases (decreases) its payment.

Only strongly coordinating pricing schemes can guarantee efficient schedules in the decentralized system. In order to introduce a family of strongly coordinating pricing schemes, we need the following technical lemma which states that the efficient duration of any job on subcontractor's machine is not longer than twice the efficient duration of the job proceeding it.

Lemma 2 *For $i = 1, \ldots, n-1$, $\bar{t}_i \leq 2\bar{t}_{i+1}$.*

Proof From (6) we get

$$\bar{t}_i - 2\bar{t}_{i+1} = \frac{p_i}{2} - \sum_{k=1}^{i-1}\frac{\bar{t}_k}{2} - p_{i+1}$$
$$+ \sum_{k=1}^{i-1}\bar{t}_k + \left(\frac{p_i}{2} - \sum_{k=1}^{i-1}\frac{\bar{t}_k}{2}\right)$$
$$= p_i - p_{i+1}.$$

The agents in N are sequence according to non-decreasing order of processing times, thus we have $p_i \leq p_{i+1}$ which obtains $\bar{t}_i \leq 2\bar{t}_{i+1}$. □

Note that in Lemma 2, the equality holds for jobs with equal processing times. More precisely, $p_i = p_{i+1}$ if and only if $\bar{t}_i = 2\bar{t}_{i+1}$.

We are now ready to give a family of non-increasing pricing schemes that are strongly coordinating, that is, any deviation from efficient allocation by any agent results in utility loss for that agent.

Theorem 3 *Consider the coefficient set $\kappa = \{\kappa_1, \ldots, \kappa_n\}$ such that for $i = 1, \ldots, n-1$, $0 < \kappa_i < \kappa_{i+1}/2$ and $0 <$*

$\kappa_n \leq 2^{n-1}\delta/3$ *where $0 < \delta \leq 3/2^{n+1}$. The following pricing scheme is strongly coordinating:*

$$q_O(t) = \begin{cases} 1 - 2^{i-1}\delta \\ +\kappa_i\left[1 - 2(t - t_i)/\bar{t}_i\right] & \text{if } t_i \leq t < t_i + \bar{t}_i, i \in N \\ 1 - 2^{n-1}\delta - \kappa_n & \text{if } t \geq t_n + \bar{t}_n \end{cases} \tag{15}$$

Proof The assumptions on κ ensure that: (a) q_O is non-increasing, (b) $q_O(0) < 1$, and (c) $q_O(t) \geq 0$ for $t \geq 0$.

To verify (a), it suffices to consider the points of (possible) discontinuity, i.e., $t = t_{i+1}$ for $i = 1, \ldots, n-1$. There, it must hold that $1 - 2^{i-1}\delta - \kappa_i \geq 1 - 2^i\delta + \kappa_{i+1}$ or

$$\kappa_i + \kappa_{i+1} \leq 2^{i-1}\delta. \tag{16}$$

By extending the assumption $\kappa_i < \kappa_{i+1}/2$ we get

$$\kappa_i < \kappa_n/2^{n-i} \tag{17}$$

for $i = 1, \ldots, n-1$. Thus, $\kappa_i + \kappa_{i+1} < 3\kappa_n/2^{n-i}$. It follows that for (16) to hold, we need

$$\kappa_n \leq 2^{n-1}\delta/3. \tag{18}$$

In order for (b) to hold we must have $1 - \delta + \kappa_1 < 1$ or $\kappa_1 < \delta$. Note that (17) and (18) yield $\kappa_i < 2^{i-1}\delta/3$ for $i = 1, \ldots, n-1$ which implies that $\kappa_1 < \delta/3$. Thus (b) holds.

Given that q_O is non-increasing, to verify (c) we must check the non-negativity of $q_O(t)$ at $t \geq t_n + \bar{t}_n$. Note that from (17) we get

$$1 - 2^{n-1}\delta - \kappa_n \geq 1 - 2^{n-1}\delta - 2^{n-1}\delta/3.$$

Hence, for (c) to hold we need $\delta \leq 3/2^{n+1}$.

We now check the conditions in Theorem 2. Observe that q_O is piecewise linear and has a constant slope in between any consecutive points of possible discontinuity.

With regard to condition C1, note that for $0 < \epsilon \leq \bar{t}_{i-1}/2$ both $\pi^{q_O}(t_i - 2\epsilon, t_i)$ and $\pi^{q_O}(t_{i+1} - \epsilon, t_{i+1})$ can be obtained by calculating the areas of the corresponding trapezoids. Let $t^- = \lim_{\epsilon \to 0^+}(t - \epsilon)$. For any $i = 2, \ldots, n$

we have

$$\pi^{qo}\left(t_i - 2\epsilon, t_i\right) - \pi^{qo}\left(t_{i+1} - \epsilon, t_{i+1}\right)$$
$$= \frac{1}{2} 2\epsilon \left[q_O\left(t_i - 2\epsilon\right) + q_O\left(t_i^-\right) \right]$$
$$\quad - \frac{1}{2}\epsilon \left[q_O\left(t_{i+1} - \epsilon\right) + q_O\left(t_{i+1}^-\right) \right]$$
$$= \frac{1}{2} 2\epsilon \left[1 - 2^{i-2}\delta - \kappa_{i-1}\left(1 - \frac{4\epsilon}{\bar{t}_{i-1}}\right) + 1 - 2^{i-2}\delta - \kappa_{i-1} \right]$$
$$\quad - \frac{1}{2}\epsilon \left[1 - 2^{i-1}\delta - \kappa_i\left(1 - \frac{2\epsilon}{\bar{t}_i}\right) + 1 - 2^{i-1}\delta - \kappa_i \right]$$
$$= \epsilon \left[1 - 2\kappa_{i-1}\left(1 - \frac{2\epsilon}{\bar{t}_{i-1}}\right) + \kappa_i\left(1 - \frac{\epsilon}{\bar{t}_i}\right) \right].$$

From Lemma 2 we know that $\bar{t}_{i-1} \leq 2\bar{t}_i$ and consequently $1 - 2\epsilon/\bar{t}_{i-1} \leq 1 - \epsilon/\bar{t}_i$. By assumption we have $2\kappa_{i-1} < \kappa_i$ which leads to the observation that for every $0 < \epsilon < \bar{t}_{i-1}/2$,

$$\pi^{qo}\left(t_i - 2\epsilon, t_i\right) - \pi^{qo}\left(t_{i+1} - \epsilon, t_{i+1}\right) > \epsilon \quad (19)$$

with regard to $\epsilon = \bar{t}_{i-1}/2$ we consider two cases. If $\bar{t}_{i-1}/2 \neq \bar{t}_i$ then at $\epsilon = \bar{t}_{i-1}/2$, (19) holds strictly as well which implies that (19) holds strictly for every $0 < \epsilon \leq t_i$. Otherwise, if $\bar{t}_{i-1}/2 = \bar{t}_i$, then at $\epsilon = \bar{t}_{i-1}/2$, (19) holds as equality. However, this requires that $p_{i-1} = p_i$ which corresponds to an alternative efficient allocation wherein job i is allocated with the interval allocated to job $i - 1$. Hence, the choice of any earlier intervals for a given agent i which is not part of an efficient allocation results in loss of utility for that agent. Also, for jobs with equal processing times the choice of an earlier alternative efficient intervals does not alter the utility to the corresponding agents.

With regard to condition C2, note that for $0 < \epsilon \leq \bar{t}_i/2$ both $\pi^{qo}\left(t_i, t_i + 2\epsilon\right)$ and $\pi^{qo}\left(t_{i+1}, t_{i+1} + \epsilon\right)$ can also be obtained by calculating the areas of the corresponding trapezoids. for any $i = 1, \ldots, n - 1$ we have

$$\pi^{qo}\left(t_i, t_i + 2\epsilon\right) - \pi^{qo}\left(t_{i+1}, t_{i+1} + \epsilon\right)$$
$$= \frac{1}{2} 2\epsilon \left[q_O\left(t_i\right) + q_O\left(t_i + 2\epsilon\right) \right]$$
$$\quad - \frac{1}{2}\epsilon \left[q_O\left(t_{i+1}\right) + q_O\left(t_{i+1} + \epsilon\right) \right]$$
$$= \frac{1}{2} 2\epsilon \left[1 - 2^{i-1}\delta + \kappa_i + 1 - 2^{i-1}\delta + \kappa_i\left(1 - \frac{4\epsilon}{\bar{t}_i}\right) \right]$$
$$\quad - \frac{1}{2}\epsilon \left[1 - 2^i\delta + \kappa_{i+1} + 1 - 2^i\delta + \kappa_{i+1}\left(1 - \frac{2\epsilon}{\bar{t}_{i+1}}\right) \right]$$
$$= \epsilon \left[1 + 2\kappa_i\left(1 - \frac{2\epsilon}{\bar{t}_i}\right) - \kappa_{i+1}\left(1 - \frac{\epsilon}{\bar{t}_{i+1}}\right) \right].$$

From Lemma 2 we know that $\bar{t}_i \leq 2\bar{t}_{i+1}$ and consequently $1 - 2\epsilon/\bar{t}_i \leq 1 - \epsilon/\bar{t}_{i+1}$. By assumption we have $\kappa_i < \kappa_{i+1}/2$. Therefore, for $0 < \epsilon < \bar{t}_{i-1}/2$ we would have

$$\pi^{qo}\left(t_i, t_i + 2\epsilon\right) - \pi^{qo}\left(t_{i+1}, t_{i+1} + \epsilon\right) < \epsilon \quad (20)$$

If $\bar{t}_i/2 \neq \bar{t}_{i+1}$ then the latter also holds for $\epsilon = \bar{t}_i/2$. In case of $\bar{t}_i/2 = \bar{t}_{i+1}$, (20) would hold as equality for $\epsilon = \bar{t}_i/2$. However, this requires $p_i = p_{i+1}$ which implies the existence of an alternative efficient allocation. Hence, the choice of any later intervals for a given agent i which is not part of an efficient allocation results in the loss of utility for that agent. Also, for jobs with equal processing times the choice of a later alternative efficient intervals does not alter the utility to the corresponding agents.

Considering the above, we conclude that q_O is a coordinating pricing scheme where for any $\mathcal{T} \in \mathcal{E}$ and every $T \in \mathfrak{T}\setminus\mathcal{E}$, it holds for all $i \in N$ that $\pi^q(\mathcal{T}_i) - \pi^q(T_i) < \upsilon_i(\mathcal{T}_i) - \upsilon_i(T_i)$. □

To see how the pricing scheme introduced in Theorem 3 can coordinate the subcontractor scheduling problem first note that for all agents with unequal processing times, corresponding efficient allocations are exclusively the best choices of subcontracting intervals. For agents with equal processing time jobs, however, the best choices are multiple. Nevertheless, q_O would result in the situation where the number of best choices for the jobs with equal processing times are exactly the same as the number of those jobs. Therefore, a first-come-first-serve rule would result in strict coordination of individual choices.

4.1 Subcontractor's revenue with coordinating pricing schemes

Let $\Pi^q = \sum_{i \in N} \pi^q(T_i)$ be the total payment from agents to the subcontractor under the pricing schemes q. We call Π^q the subcontractor's revenue under q. In this section we characterize the range of Π^q with q being a coordinating pricing scheme.

The coordinating pricing schemes introduced in Theorem 3 can extract (almost) all of the total saving $\upsilon(\mathcal{T})$ from the agents by selecting δ small enough which is shown in the following theorem.

Theorem 4 $\Pi^{qo} = \upsilon(\mathcal{T}) - \delta \sum_{i \in N} 2^{i-1}\bar{t}_i$.

Proof For every $i \in N$ we have,

$$\pi^{qo}(\mathcal{T}_i) = \bar{t}_i\left(1 - 2^{i-1}\delta + \kappa_i + 1 - 2^{i-1}\delta + \kappa_i - 2\kappa_i\right)/2$$
$$= \bar{t}_i\left(1 - 2^{i-1}\delta\right).$$

Therefore $\Pi^{qo} = \sum_{i \in N} \bar{t}_i - \delta \sum_{i \in N} 2^{i-1}\bar{t}_i$. □

Hence, the total payments can be made arbitrary close to the maximum possible which is the total saving $\upsilon(\mathcal{T})$.

We now provide a *lower* bound for the total subcontractor's revenue attainable under any coordinating pricing

scheme. We show in Corollary 1 that this bound is an *upper* bound on the subcontractor's revenue attainable by any coordinating mechanism with private information.

Theorem 5 (Lower bound) *For any coordinating pricing scheme* q, *we have* $\Pi^q \geq \sum_{i \in N} \bar{t}_i \left(1 - 1/2^{n-i}\right)$.

Proof Let $\mathcal{T} \in \mathcal{E}$ be an efficient allocation. If q is coordinating, then for all $i \in N$ there is no T_i which results in higher utility than $\mathcal{T}_i$. In particular, for any agent $i = 2, \ldots, n$, the following deviation from efficient allocation should not be profitable: construct T_i by removing the last $\epsilon > 0$ interval at the end of $\mathcal{T}_i$ and add instead another interval with duration 2ϵ starting at time $t_i - 2\epsilon$ (to have feasible allocations it must be that $\epsilon \leq t_i/2$). With the new allocation, i finishes ϵ units of time earlier, i.e., $v_i(\mathcal{T}_i) - v_i(T_i) = -\epsilon$. To make this deviation unprofitable, the condition in (10) requires that

$$\pi^q (t_i - 2\epsilon, t_i) - \pi^q (t_{i+1} - \epsilon, t_{i+1}) \geq \epsilon. \tag{21}$$

We construct a pricing scheme q_L which satisfies the above condition for every $i = 2, \ldots, n$ and every $0 < \epsilon \leq t_i/2$, and has the lowest possible total revenue.

Let $q_L(t) = 0$ for $t \geq t_n$. Observe that in the latter range q_L has the lowest possible prices, so under q_L the agent n pays 0 for its efficient allocation. The condition in (21) for $i = n$ and any $0 < \epsilon \leq t_n/2$ requires that

$$\pi^{q_L} (t_n - 2\epsilon, t_n) \geq \epsilon. \tag{22}$$

To obtain the cheapest pricing scheme we require q_L to yield $\pi^{q_L} (t_n - 2\epsilon, t_n) = \epsilon$ for every $0 < \epsilon \leq \bar{t}_{n-1}/2$. In particular for $\epsilon = \bar{t}_{n-1}/2$ we get $\pi^{q_L} (t_{n-1}, t_n) = \bar{t}_{n-1}/2$. This means that under q_L the agent $i = n - 1$ pays $\bar{t}_{n-1}/2$.

Next, the condition (21) for agent $i = n - 1$ and every $0 < \epsilon \leq t_{n-1}/2$ requires that

$$\pi^{q_L} (t_{n-1} - 2\epsilon, t_{n-1}) - \pi^{q_L} (t_n - \epsilon, t_n) \geq \epsilon. \tag{23}$$

To obtain the cheapest pricing scheme we can consider q_L to be such that (23) holds as equality for every $0 < \epsilon \leq \bar{t}_{n-2}/2$. In this case, for $\epsilon = \bar{t}_{n-2}/2$ we get

$$\pi^{q_L} (t_{n-2}, t_{n-1}) - \pi^{q_L} \left(t_n - \bar{t}_{n-2}/2, t_n\right) = \bar{t}_{n-2}/2. \tag{24}$$

By Lemma 2, we know that $\bar{t}_{n-2}/2 \leq \bar{t}_{n-1}$ so $t_n - \bar{t}_{n-2}/2 \geq t_{n-1}$. Hence we have $\pi^{q_L} \left(t_n - \bar{t}_{n-2}/2, t_n\right) = \bar{t}_{n-2}/4$ and eventually

$$\pi^{q_L} (t_{n-2}, t_{n-1}) = 3\bar{t}_{n-2}/4. \tag{25}$$

The latter implies that agent $i = n - 2$ under q_L pays $\bar{t}_{n-2}(3/4)$ to the subcontractor.

By induction on i it is easily verifiable that under the pricing scheme that satisfies the condition (21) and has the lowest possible price, every agent $i \in N$ pays an amount equal to $\bar{t}_i (1 - 1/2^{n-i})$ to the subcontractor. Thus for every coordinating pricing scheme q, it must be the case that $\Pi^q \geq \sum_{i \in N} \bar{t}_i \left(1 - 1/2^{n-i}\right)$.   □

We close with the observation that the lower bound in Theorem 5 is attainable by the following pricing scheme

$$q_L(t) = \begin{cases} 1 - 1/2^{n-i} & \text{if } t_i \leq t < t_{i+1}, i \in N; \\ 0 & \text{if } t \geq t_n. \end{cases} \tag{26}$$

Clearly q_L is non-increasing, and coordinating. To see the latter we check the conditions in Theorem 2: C1 holds since we have $\left(1 - 1/2^{n-i+1}\right) 2\epsilon - \left(1 - 1/2^{n-i}\right) \epsilon = \epsilon$. Also, we have $\left(1 - 1/2^{n-i}\right) 2\epsilon - \left(1 - 1/2^{n-i-1}\right) \epsilon = \epsilon$. Thus C2 holds as $(0) 2\epsilon - (0) \epsilon \leq \epsilon$. Clearly, $q_L(0) < 1$.

Finally we get $\Pi^{q_L} (\mathcal{T}) = \sum_{i \in N} \bar{t}_i \left(1 - 1/2^{n-i}\right)$.

5 Private processing times

Section 4 shows how to coordinate a set of self-interested agents through a coordinating pricing scheme under the assumption that the vector of true processing times p is given and known to the subcontractor. In this section, we relax this assumption by allowing true processing times to be private information of agents. Thus reporting an untruthful processing time by an agent can deceive the subcontractor who consequently moves the agent's job in the sequence of jobs in efficient allocations which can potentially increase the agent's time saving or reduce his payment.

This section presents a mechanism that guarantees that the agents are always *better off* by reporting their true processing times, and that they are also *better off* by participating in the subcontracting—such mechanism is refereed to as the *coordinating* mechanism for the subcontractor scheduling problem under private information. Finally, the section calculates the amount the subcontractor forfeits to the agents to extract the true processing times from them.

We draw upon mechanism design theory (Nisan 2007; Jackson 2000) and introduce a *payment scheme* that motivates agents to report true processing times knowing the subcontractor's intention to maximize the total savings.

A pricing scheme in Sect. 4 determines payments made by agents to the subcontractor for time *intervals* that they choose to utilize on the subcontractor's machine. In this section the payments are made for reporting particular processing *times* by the agents to the subcontractor. Formally, each agent reports a processing time r_i from its individual processing time space $P_i \subset \mathbb{R}_+$. We use p_i to denote the true processing time of agent i. Let $P = P_1 \times \cdots \times P_n$ be the *processing time space* of all agents.

Definition 2 A mechanism M is defined by a $(n + 1)$-tuple $(f^M, \pi_1^M, \ldots, \pi_n^M)$ consisting of an allocation function $f^M : P \to \mathfrak{T}$, and a payment scheme $\pi^M : P \to \mathbb{R}^n$.

Given a vector of reported processing times r and the sequence in which the agents report them (to break ties in the allocation), the payment scheme of a mechanism determines the monetary amount $\pi_i^M(r)$ that $i \in N$ pays in return for receiving the allocation prescribed for i by f^M.

With quasilinear utilities, a mechanism M would result in the utility

$$u_i^M \left(f^M(r) \right) = \upsilon_i \left(f^M(r) \right) - \pi_i^M(r) \tag{27}$$

for $i \in N$.

The subcontractor is seeking to allocate the subcontracting intervals *efficiently*, thus we focus on mechanisms whose allocation functions yield efficient allocations, i.e., $f^M \equiv \mathcal{T}$. We call such mechanisms efficient.

We concentrate on incentive compatible mechanism since they have the potential to motivate the agents to report their true processing times.

Definition 3 A mechanism M is *incentive compatible* if for every reported processing time vector $r = (r_1, \ldots, r_n)$ and for every agent $i \in N$

$$u_i^M \left(f^M \left(p_i, r^{-i} \right) \right) \geq u_i^M \left(f^M(r) \right), \tag{28}$$

where r^{-i} is the vector r without its ith coordinate.

Although incentive compatibility implies that agents are not better off by lying about their true processing times, it does not generally imply that the agents are better off by reporting their true processing times—there could generally exist untruthful agent reports that result in the same utility as the truthful ones. Therefore, unfortunately, agents may generally choose to be untruthful after all even with the incentive compatible mechanisms. This is generally the case for VCG mechanisms, in particular the pivotal mechanism, which characterize the class of efficient and incentive compatible mechanisms for quasilinear utilities (Green and Laffont 1977). However, we show later in Theorem 6 that for the subcontractor scheduling problem studied in this paper the pivotal mechanism makes the agents always better off by reporting their true processing times.

5.1 Payment scheme

We draw upon the VCG mechanisms to obtain a desirable payment scheme for the subcontractor scheduling problem. The payment scheme for VCG mechanisms is defined as follows

$$\pi_i^{\text{VCG}}(r) = h_i \left(r^{-i} \right) - \sum_{j \in N \setminus \{i\}} \upsilon_j \left(\mathcal{T}(r) \right),$$

where h_i is a function independent of agent i's reported processing time r_i. Among the class of VCG mechanisms, the pivotal mechanism with

$$h_i \left(r^{-i} \right) = \sum_{j \in N \setminus \{i\}} \upsilon_j \left(\mathcal{T} \left(r^{-i} \right) \right) \tag{29}$$

guarantees that all payments from the agents to the mechanism are non-negative. Here $\mathcal{T}(r^{-i})$ is an efficient allocation for the set of jobs excluding job i and for processing time vector r. Thus, the payment by pivotal mechanism for $i \in N$ equals

$$\pi_i^{PM}(r) = \sum_{j \in N \setminus \{i\}} \left[\bar{t}_j \left(r^{-i} \right) - \bar{t}_j(r) \right], \tag{30}$$

where $\bar{t}_j(r)$ is the duration of subcontracting time allocated to j by the efficient (optimal) allocation for the reported r and $\bar{t}_j(r^{-i})$ is the duration of subcontracting time allocated to j by the efficient allocation for the reported r^{-i} that excludes i.

By (30) the payment $\pi_i^{PM}(r)$ charged by the pivotal mechanism to agent i is the total subcontracting time that could have been allocated to all other agents had not i asked for subcontracting and reported r_i. It follows from the characterization of efficient solutions that the exclusion of an agent i reporting r_i would only affect the agents that follow i in $\mathcal{T}(r)$. Let σ be the permutation of jobs in $\mathcal{T}(r)$ and let $[i]$, for $i \in N$, be the position of i in $\mathcal{T}(r)$.

Lemma 3 *For all $i \in N$, we have*

$$\pi_i^{PM}(r) = \bar{t}_i(r) \left(1 - 1/2^{n-[i]} \right). \tag{31}$$

Proof We start by showing that for any $i, j \in N$, we have $\bar{t}_j(r^{-i}) - \bar{t}_j(r) = 0$ if $[j] < [i]$ and $\bar{t}_j(r^{-i}) - \bar{t}_j(r) = \bar{t}_i(r)/2^{[j]-[i]}$ if $[j] > [i]$.

By (6) r_i does not appear in $\bar{t}_j(r)$ for $[j] < [i]$. Therefore, $\bar{t}_j(r^{-i}) - \bar{t}_j(r) = 0$ for $[j] < [i]$.

If $[j] > [i]$, then the exclusion of i affects the duration of allocated time to jobs j without altering the relative position of other jobs. By (4) and (5), we have

$$\bar{t}_j(r) = \frac{1}{2} \left(r_j - \left[t_i(r) + \bar{t}_i(r) + \sum_{[i]<[k]<[j]} \bar{t}_k(r) \right] \right)$$

and

$$\bar{t}_j \left(r^{-i} \right) = \frac{1}{2} \left(r_j - \left[t_i(r) + \sum_{[i]<[k]<[j]} \bar{t}_k \left(r^{-i} \right) \right] \right)$$

Thus

$$\bar{t}_j \left(r^{-i} \right) - \bar{t}_j (r)$$

$$= \frac{1}{2} \left(\bar{t}_i (r) - \sum_{[i]<[k]<[j]} \left[\bar{t}_k \left(r^{-i} \right) - \bar{t}_k (r) \right] \right)$$

By induction on $[j] - [i]$ we obtain $\bar{t}_j(r^{-i}) - \bar{t}_j(r) = \bar{t}_i(r)/2^{[j]-[i]}$ if $[j] > [i]$.

Based on the above observation, the total payment in (30) becomes

$$\pi_i^{PM}(r) = \sum_{[k]>[i]} \bar{t}_i(r)/2^{[k]-[i]}$$

$$= \bar{t}_i(r) \sum_{[k]>[i]} 1/2^{[k]-[i]}$$

$$= \bar{t}_i(r) \left(1 - 1/2^{n-[i]} \right).$$

$\square$

5.2 Coordinating meachanism

Lemma 3 provides a closed-form formula for the pivotal mechanism payments. With these payments, each agent pays an amount directly proportional to the duration of its allocation on subcontractor's machine and inversely proportional to its order in the sequence of jobs on subcontractor's machine. Therefore if an agent announces a shorter than true processing time, which could possibly precipitate its position in the sequence and thus result in the agent's higher time saving, then the agent pays more. We show that as a result of this misrepresentation, the agent's utility decreases. At the same time, we show that reporting longer than true processing time reduces agent's utility as well, unless the agent comes in the last position in the sequence. This position is special since it neither incurs any payment by (31) nor it affects any other agent. Therefore, whether the agent in this position lies or not is irrelevant. Alternatively, one can assume that the longest job always belongs to the subcontractor—this dummy job could represent the continuation of subcontractor's operations. Consequently, the mechanism enforces agents to report their actual processing time which we now formally prove.

Theorem 6 *Let $r = (r_1, \ldots, r_n)$ be the reported processing times and let p_i be true processing time of agent i. Then $u_i^{PM}(\mathcal{T}(p_i, r^{-i})) > u_i^{PM}(\mathcal{T}(r))$, provided that $p_i \neq r_i$ and i is not in the last position of $\mathcal{T}(r)$.*

Proof Assume $p_i \neq r_i \in P_i$ for some agent i. Let $p = (p_i, r_{-i})$. Suppose job i is in position h and l in efficient solutions $\mathcal{T}(p)$ and $\mathcal{T}(r)$, respectively, and assume $l \neq n$.

By (27), (31), (7), and (4), for p_i, the utility of agent i is

$$u_i^{PM}(\mathcal{T}(p)) = \frac{p_i - t_i(p)}{2} - \frac{p_i - t_i(p)}{2} \left(1 - \frac{1}{2^{n-h}} \right)$$

$$= \frac{p_i - t_i(p)}{2} \frac{1}{2^{n-h}}. \tag{32}$$

There are two cases for r_i: (a) $r_i > p_i$, and (b) $r_i < p_i$.

(a) With $r_i > p_i$ we have

$$u_i^{PM}(\mathcal{T}(r)) = \max \left\{ \frac{p_i - t_i(r)}{2}, 0 \right\}$$

$$- \frac{r_i - t_i(r)}{2} \left(1 - \frac{1}{2^{n-l}} \right), \tag{33}$$

where $n > l \geq h$ and $t_i(r)$ is the start time of job i in $\mathcal{T}(r)$.

Suppose $l = h$. In this case we have $t_i(r) = t_i(p)$. Since $h < n$, it directly follows from (32) and (33) that $u_i^{PM}(\mathcal{T}(r)) < u_i^{PM}(\mathcal{T}(p))$.

Next, suppose $l > h$. Let $\sigma(k)$ be the job in position k in $\mathcal{T}(p)$. By (4) and (5) and induction on $l - h$ we have,

$$t_i(r) = t_i(p) + \sum_{k=h+1}^{l} \frac{r_{\sigma(k)} - t_i(p)}{2^{l+1-k}}$$

$$= \frac{r_{\sigma(l)}}{2} + \cdots + \frac{r_{\sigma(h+1)}}{2^{l-h}} + \frac{t_i(p)}{2^{l-h}}$$

Using the fact that $r_{\sigma(k)} \geq p_i$ for $k \geq h$ we get

$$t_i(r) \geq \sum_{k=h+1}^{l} \frac{p_i}{2^{l+1-k}} + \frac{t_i(p)}{2^{l-h}} = p_i - \frac{p_i}{2^{l-h}} + \frac{t_i(p)}{2^{l-h}} \tag{34}$$

For a contradiction assume $u_i^{PM}(\mathcal{T}(r)) \geq u_i^{PM}(\mathcal{T}(p))$, i.e., by (32) and (33) assume that

$$\max \left\{ \frac{p_i - t_i(r)}{2}, 0 \right\} - \frac{r_i - t_i(r)}{2} \left(1 - \frac{1}{2^{n-l}} \right)$$

$$\geq \frac{p_i - t_i(p)}{2} \frac{1}{2^{n-h}}.$$

If $p_i - t_i(r) \leq 0$, then the contradiction readily follows. Otherwise, suppose $p_i - t_i(r) > 0$. Then, we have

$$\frac{p_i - t_i(r)}{2} - \frac{r_i - t_i(r)}{2} \left(1 - \frac{1}{2^{n-l}} \right) \geq \frac{p_i - t_i(p)}{2} \frac{1}{2^{n-h}}.$$

The last inequality simplifies to

$$2^{l-h} [r_i - t_i(r)] - [p_i - t_i(p)] \geq 2^{n-h} [r_i - p_i].$$

Using (34), note that the last inequality remains valid if $t_i(r)$ is replaced with $p_i - p_i/2^{l-h} + t_i(p)/2^{l-h}$. The inequality then simplifies to $2^{l-h} [r_i - p_i] \geq 2^{n-h} [r_i - p_i]$.

This is a contradiction since we have $2^{l-h} < 2^{n-h}$. Thus $u_i^{PM}(\mathcal{T}(r)) < u_i^{PM}(\mathcal{T}(p))$ for $r_i > p_i$.

(b) With $r_i < p_i$ we have

$$
\begin{aligned}
u_i^{PM}(\mathcal{T}(r)) &= \frac{r_i - \mathsf{t}_i(r)}{2} - \frac{r_i - \mathsf{t}_i(r)}{2}\left(1 - \frac{1}{2^{n-l}}\right) \\
&= \frac{r_i - \mathsf{t}_i(r)}{2}\frac{1}{2^{n-l}},
\end{aligned}
\tag{35}
$$

where $1 \leq l \leq h$.

Suppose $h = l$. Then we have $\mathsf{t}_i(r) = \mathsf{t}_i(p)$, and it follows from (32) and (35) that $u_i^{PM}(\mathcal{T}(r)) < u_i^{PM}(\mathcal{T}(p))$.

Next, suppose $1 \leq l < h$. By (4) and (5) and induction on $h - l$

$$
\begin{aligned}
\mathsf{t}_i(p) &= \mathsf{t}_i(r) + \sum_{k=l}^{h-1} \frac{r_{\sigma(k)} - \mathsf{t}_i(r)}{2^{h-k}} \\
&= \frac{r_{\sigma(h-1)}}{2} + \cdots + \frac{r_{\sigma(l)}}{2^{h-l}} + \frac{\mathsf{t}_i(r)}{2^{h-l}}.
\end{aligned}
$$

Using the fact that $r_{\sigma(k)} \leq p_i$ for $k \leq h$ we get

$$
\mathsf{t}_i(p) \leq \sum_{k=1}^{h-1} \frac{p_i}{2^{h-k}} + \frac{\mathsf{t}_i(r)}{2^{h-l}} = p_i - \frac{p_i}{2^{h-l}} + \frac{\mathsf{t}_i(r)}{2^{h-l}}
\tag{36}
$$

For a contradiction assume $u_i^{PM}(\mathcal{T}(r)) \geq u_i^{PM}(\mathcal{T}(p))$ or, by (32) and (35),

$$
\frac{r_i - \mathsf{t}_i(r)}{2}\frac{1}{2^{n-l}} \geq \frac{p_i - \mathsf{t}_i(p)}{2}\frac{1}{2^{n-h}}.
$$

The last simplifies to

$$
r_i - \mathsf{t}_i(r) \geq 2^{h-l}[p_i - \mathsf{t}_i(p)].
$$

Using (36), the last inequality remains valid if $\mathsf{t}_i(p)$ is replaced with $p_i - p_i/2^{h-l} + \mathsf{t}_i(r)/2^{h-l}$. The inequality then simplifies to $r_i \geq p_i$ which is a contradiction. Thus $u_i^{PM}(\mathcal{T}(r)) < u_i^{PM}(\mathcal{T}(p))$ for $r_i < p_i$. □

Finally, to ensure agents' participation in subcontracting, the efficient mechanism M must satisfy the *strict individual rationality* condition requiring that for all agents, truth-telling results in positive utility, irrespective of the reported processing times of other agents. That is, for every $r = (r_1, \ldots, r_i = p_i, \ldots, r_n)$ and all $i \in N, u_i^M(\mathcal{T}(r)) > 0$. The efficient mechanism $(\mathcal{T}, \pi^{PM})$ satisfies the strict individual rationality condition since, by (27), (31) and (7), we have $u_i^{PM}(\mathcal{T}(r)) = \bar{\mathsf{t}}_i(r)/2^{n-[i]} > 0$, for $r_i = p_i$.

Thus we showed the mechanism $(\mathcal{T}, \pi^{PM})$ guarantees, by Theorem 6, that the agents are always better off by reporting their true processing times, and that, by strict individual rationality, they are better off by participating in the subcontracting. Therefore the mechanism coordinates the subcontractor scheduling problem under private information.

One possible disadvantage of coordination based on mechanism design, the VCG mechanisms in particular, is that this coordination is implicitly centralized—the efficient schedule is computed by the subcontractor and then reported to the agents. Although distributed implementations of the pivotal mechanism where the schedule is determined by the self-interested agents themselves are beyond the scope of this paper, we refer the reader to Parkes and Shneidman (2004) where guidelines for the distributed implementations of the VCG mechanisms are proposed.

5.3 Subcontractor's revenue with coordinating mechanisms

The payment scheme π^{PM} provides the subcontractor with a positive revenue for any given r. Let us define $\Pi^{PM}(r) = \sum_{i \in N} \pi_i^{PM}(r)$ as the total payment from agents to the subcontractor in pivotal mechanism. By Theorem 6 all agents except the last in $\mathcal{T}(r)$ report their true processing times while the last one pays nothing to the subcontractor regardless of his report. Therefore, we focus on the subcontractor's revenue for the true processing times p.

Although the subcontractor's services result in the total savings of $\upsilon(\mathcal{T}(p))$ for all agents, it follows from Lemma 3 that the mechanism $(\mathcal{T}, \pi^{PM})$ can only achieve revenue $\Pi^{PM}(p)$ for the subcontractor which is always less than $\upsilon(\mathcal{T}(p))$. In fact, instances can be found for which the ratio of subcontractor's revenue to total savings obtained via subcontracting, i.e., $\Pi^{PM}(p)/\upsilon(\mathcal{T}(p))$, is arbitrarily close to zero. For example, consider a set of two jobs with $p_1 = 2$ and $p_2 = 101$. Despite having a total saving of 51, we have $\pi_1^{PM} = 0.5$ and $\pi_2^{PM} = 0$ resulting in the ratio $\upsilon(\mathcal{T}(p))/\Pi^{PM}(p) < 0.01$.

Nevertheless, Moulin (1986) shows that if for all agents it holds that $\min_T \upsilon_i(T) = 0$—which is the case in subcontractor's subcontracting problem with divisible jobs—then among all efficient, incentive compatible, and individually rational mechanisms, the pivotal mechanism generates the highest total payment. Therefore, the subcontractor cannot gain a higher revenue from any other efficient, incentive compatible, and individually rational mechanisms than it does from $(\mathcal{T}, \pi^{PM})$. Thus, by the juxtaposition of Lemma 3 and Theorem 5 we get the following observation.

Corollary 1 *Subcontractor's revenue with any efficient, incentive compatible, and individually rational mechanisms under private information never exceeds that with a coordinating pricing scheme under complete information.*

However, the construction of a coordinating pricing scheme q requires the knowledge of true processing times p, thus the difference $\Pi^q(p) - \Pi^{PM}(p)$ may be considered

as the amount the subcontractor forfeit to the agents to extract the true processing times from them.

6 Asymmetric valuations

We now relax the assumption of symmetric valuations and assume that the value of a unit time saving for agent i is w_i. Let $w = (w_i | i \in N)$. The saving to agent i resulting from the allocation T thus equals

$$\upsilon_i^w(T) = w_i \upsilon_i(T). \tag{37}$$

The total saving for all agents is defined as the sum of individual savings, that is $\upsilon^w(T) = \sum_{i \in N} \upsilon_i^w(T)$. The efficient allocation is $\mathcal{T} = \arg\max_T \upsilon^w(T)$. Unlike the symmetric case where each job in N is allocated a non-empty interval on subcontractor's machine, in asymmetric case efficient allocations may exclude some jobs from the subcontractor's schedule. Let $\mathcal{I} = \{i \in N | \bar{t}_i > 0\}$ be the set of jobs that receive non-empty allocations, and $\mathcal{O} = \{i \in N | \bar{t}_i = 0\}$ the set of jobs that receive no such allocations. We denote the cardinality of $\mathcal{I}$ with l.

In the remainder of this section, we focus on coordinating pricing schemes in situations with asymmetric valuations. In these situations, any coordinating pricing scheme q meets the following conditions: for $t_i \leq t < t_{i+1}$ it holds that $q(t) < w_i$, that is, it is required that subcontracting price across the allocation of job i do not exceed its unit saving from subcontracting.

We show by a counterexample that it is generally impossible to construct coordinating pricing schemes in the asymmetric case. Without loss of generality we assume $\mathcal{I} = \{1, \ldots, l\}$ and that the jobs are sequenced in efficient solutions by their index order.

Theorem 7 *There exists no coordinating pricing scheme in the asymmetric valuation case.*

Proof To construct coordinating pricing schemes, we must ensure that agents would not benefit by deviating from the efficient allocations. In particular, deviations of the similar type as in Theorem 5 must be unprofitable for any agent who receives non-zero allocations on the subcontractor's machine. That is for every $i = 2, \ldots, l$ and every $0 < \epsilon < t_i$ it must hold that,

$$q(t_i - 2\epsilon, t_i) - q(t_{i+1} - \epsilon, t_{i+1}) \geq w_i \epsilon. \tag{38}$$

Consider the following example. There are four jobs $p = (p_1 = 6, p_2 = 18, p_3 = 19, p_4 = 25)$ with $w = (w_1 = 9, w_2 = 2, w_3 = 12, w_4 = 11)$. The unique efficient allocation is $\mathcal{I} = \{1, 3, 4\}$ with $\mathcal{T}_1 = [0, 3)$, $\mathcal{T}_3 = [3, 11)$, $\mathcal{T}_4 = [11, 18)$, and $\mathcal{O} = \{2\}$. For any coordinating pricing

scheme, we must have $q(t) > 2$ for $0 \leq t < 18$ so that job 2 would not be able to choose any interval on subcontractor's machine.

The condition in (38) for job $i = 4$ and $\epsilon = 0.25$ requires that,

$$q(10.5, 11) - q(17.75, 18) \geq 2.75. \tag{39}$$

Considering that we need $q(t) > 2$ for $0 \leq t < 18$, it must be that $q(17.75, 18) > 0.5$ and consequently $q(10.5, 11) > 3.25$.

The condition in (38) for job $i = 3$ and $\epsilon = 0.5$ requires that,

$$q(2, 3) - q(10.5, 11) \geq 6. \tag{40}$$

We already know that $q(10.5, 11) > 3.25$ thus it must be that $q(2, 3) > 9.25$. This implies that in the cheapest coordinating pricing scheme, agent 1 must pay 9.25 for acquiring the time interval $[2, 3)$ on subcontractor's machine. However, since the valuation of agent 1 is only 9, this means that he would be better off by not accepting that interval. This contradicts the fact that q is coordinating. Therefore, in situations with asymmetric valuations, coordinating pricing schemes may be impossible to find. $\qquad\square$

7 Final remarks and conclusions

We studied the decentralized subcontractor scheduling problem under complete and private information. We designed the coordinating pricing schemes in case of complete information and studied the range of subcontractor's revenue for these schemes. Our coordinating pricing schemes are piecewise linear and non-increasing.

We also devised a mechanism that is incentive compatible, individually rational, and efficient in case of private information. The mechanism is based on the pivotal mechanism and it is moreover coordinating. Another desirable property of the mechanism, that immediately follows from Theorem 6, is *Envy-Freeness* (Foley 1967) which is defined as the condition where no agent prefers another agent's allocation to its own, that is $u_i(\mathcal{T}_i(p)) \geq u_i(\mathcal{T}_j(p))$ for all j and all i. The mechanism produces the highest subcontractor revenue among efficient, incentive compatible, and individually rational mechanisms.

For the asymmetric case, we showed that the coordination through pricing schemes is not always possible.

An open problem is the complexity of finding the efficient allocations for the asymmetric case.

Several other extensions are possible for this problem: job's release dates and due dates, setup times required for subcontractor to prepare for processing different jobs, and

transportation costs for transportation from and to manufacturers. Finally, the divisibility of jobs can be altered by requesting that only certain parts of jobs with fixed durations can be subcontracted. We leave these issues for future research.

References

Anderson, E. J. (1981). A new continuous model for job–shop scheduling. *International Journal of System Science*, *12*(12), 1469–1475.

Aydinliyim, T., & Vairaktarakis, G. L. (2010). Coordination of outsourced operations to minimize weighted flow time and capacity booking costs. *Manufacturing & Service Operations Management*, *12*(2), 236–255.

Aydinliyim, T., & Vairaktarakis, G. L. (2011). Planning production and inventories in the extended enterprise, chap Sequencing Strategies and Coordination Issues in Outsourcing and Subcontracting Operations, Springer, pp 269–319.

Bharadwaj, V., Ghose, D., Mani, V., & Robertazzi, T. (1996). *Scheduling divisible loads in parallel and distributed systems*. Los Alamitos: IEEE Computer Society Press.

Bukchin, Y., & Hanany, E. (2007). Decentralization cost in scheduling: A game-theoretic approach. *Manufacturing & Service Operations Management*, *9*(3), 263–275.

Cai, X., & Vairaktarakis, G. L. (2012). Coordination of outsourced operations at a third-party facility subject to booking, overtime, and tardiness costs. *Operations Research*, *60*(6), 1436–1450.

Chen, N., Deng, X., & Sun, X. (2004). On complexity of single-minded auction. *Journal of Computer and System Sciences*, *69*(4), 675–687.

Chen, Z., & Li, C. (2008). Scheduling with subcontracting options. *IIE Transactions*, *40*(12), 1171–1184.

Christodoulou, G., Koutsoupias, E., & Nanavati, A. (2004). Coordination mechanisms. In Automata, languages and programming, Springer (pp. 345–357).

Clarke, E. (1971). Multipart pricing of public goods. *Public Choice*, *11*(1), 17–33.

Curiel, I., Pederzoli, G., & Tijs, S. (1989). Sequencing games. *European Journal of Operational Research*, *40*(3), 344–351.

Curiel, I., Hamers, H., & Klijn, F. (2004). Sequencing games: a survey. In Chapters in game theory (pp. 27–50). US: Springer.

Drozdowski, M. (2009). *Scheduling for parallel processing*. London: Springer.

Foley, D. (1967). Resource allocation and the public sector. *Yale Economic Essays*, *7*(1), 45–98.

Green, J., & Laffont, J. (1977). Characterization of satisfactory mechanisms for the revelation of preferences for public goods. *Econometrica*, *45*(2), 427–438.

Groves, T. (1973). Incentives in teams. *Econometrica*, *41*, 617–631.

Heydenreich, B., Muller, R., & Uetz, M. (2007). Games and mechanism design in machine schedulingan introduction. *Production and Operations Management*, *16*(4), 437–454.

Immorlica, N., Li, L., Mirrokni, V.S., & Schulz, A. (2005). Coordination mechanisms for selfish scheduling. In Internet and network economics, Springer, pp 55–69.

Jackson, M. (2000). Encyclopedia of life support systems, chap Mechanism theory, London: EOLSS Publishers.

Kamien, M. I., & Li, L. (1990). Subcontracting, coordination, flexibility, and production smoothing in aggregate planning. *Management Science*, *36*(11), 1352–1363.

Koutsoupias, E., & Papadimitriou, C. (1999). Worst-case equilibria. In STACS 99, Springer, pp 404–413.

Kutanoglu, E., & Wu, S. (1999). On combinatorial auction and lagrangean relaxation for distributed resource scheduling. *IIE Transactions*, *31*(9), 813–826.

Lee, I., & Sung, C. (2008). Single machine scheduling with outsourcing allowed. *International Journal of Production Economics*, *111*(2), 623–634.

Moulin, H. (1986). Characterizations of the pivotal mechanism. *Journal of Public Economics*, *31*(1), 53–78.

Nisan, N. (2007). Algorithmic game theory, chap Introduction to mechanism design (for computer scientists), Cambridge University Press, pp 209–242.

Nisan, N., & Ronen, A. (2007). Computationally feasible VCG mechanisms. *Journal of Artificial Intelligence Research*, *29*, 19–47.

Parkes, D.C., & Shneidman, J. (2004). Distributed implementations of Vickery–Clarke–Groves mechanisms. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, 2004*. AAMAS 2004, pp. 261–268.

Parmigiani, A.E. (2003). Concurrent sourcing: When do firms both make and buy? PhD thesis, University of Michigan.

Qi, X. (2012). Production scheduling with subcontracting: The subcontractors pricing game. *Journal of Scheduling*, *15*(6), 773–781.

Smith, W. (1956). Various optimizers for single-stage production. *Naval Research Logistics Quarterly*, *3*(1–2), 59–66.

Suijs, J. (1996). On incentive compatibility and budget balancedness in public decision making. *Review of Economic Design*, *2*(1), 193–209.

Tan, B., & Gershwin, S. (2004). Production and subcontracting strategies for manufacturers with limited capacity and volatile demand. *Annals of Operations Research*, *125*(1), 205–232.

Taymaz, E., & Kiliçaslan, Y. (2005). Determinants of subcontracting and regional development: An empirical study on turkish textile and engineering industries. *Regional Studies*, *39*(5), 633–645.

Vairaktarakis, G. L. (2013). Noncooperative games for subcontracting operations. *Manufacturing & Service Operations Management*, *15*(1), 148–158.

Vairaktarakis, G.L., & Aydinliyim, T. (2007). Centralization versus competition in subcontracting operations, technical Memorandum Number 819, Case Western Reserve University.

Van Mieghem, J. (1999). Coordinating investment, production, and subcontracting. *Management Science*, *45*(7), 954–971.

Vickrey, W. (1961). Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, *16*(1), 8–37.

Webster, M., Alder, C., & Muhlemann, A. (1997). Subcontracting within the supply chain for electronics assembly manufacture. *International Journal of Operations & Production Management*, *17*(9), 827–841.

Wellman, M., Walsh, W., Wurman, P., & MacKie-Mason, J. (2001). Auction protocols for decentralized scheduling. *Games and Economic Behavior*, *35*(1–2), 271–303.