Optimal Mechanisms for Scheduling

Birgit Heydenreich[1] * Debasis Mishra[2] Rudolf Müller[1] Marc Uetz[3]

November 11, 2009

Abstract

We study the design of optimal mechanisms in a setting where a service provider needs to schedule a set of non-preemptive jobs, one job at a time. Jobs need to be compensated for waiting, and waiting cost is private information. In this setting, an optimal mechanism is one that induces jobs to report truthfully their waiting cost, while minimizing the total expected compensation cost of the service provider. Here, truthful refers to Bayes-Nash implementability, and assumes that private information is independently drawn from known distributions. We derive closed formulae for the optimal mechanism, and show that it is a modification of Smith's ratio rule. We also show that it can be implemented in dominant strategies. Our analysis relies on a graph-theoretic interpretation of the incentive compatibility constraints. It parallels the analysis known for auctions with single parameter agents, yet it exhibits some subtle differences. We also consider the multi-dimensional case where also the service times of jobs are private information. We show that for this problem the optimal mechanism generally does not satisfy an independence condition known as IIA, and thus known approaches are doomed to fail.

Keywords. Auction/bidding, Scheduling, Economics, Combinatorial Optimization

1 Introduction

The design of optimal auctions is recognized as an intriguing issue in auction theory; first studied by (Myerson 1981) for the case of single object auctions. In that setting, the goal is to maximize the seller's revenue subject to Bayes-Nash incentive compatibility and individual rationality. We study the design of optimal auctions in a setting where job-agents compete for being processed by a service provider that can only handle one job at a time. No job can be interrupted once started, and each job is characterized by service time and weight, the latter representing his disutility for waiting per unit time. Jobs need to be compensated for waiting. It is well known that the total disutility of the jobs is minimized by a scheduling policy known as Smith's ratio rule: schedule jobs in order of non-increasing ratios of weight over service time (Smith 1956). We aim to find Bayes-Nash incentive compatible mechanisms that minimize the expected expenses of the service provider. Given jobs'

[1]Maastricht University, Quantitative Economics, P.O.Box 616, 6200 MD Maastricht, The Netherlands. Email: birgit.heydenreich@gmail.com, r.muller@maastrichtunversity.nl
*Supported by NWO grant 2004/03545/MaGW 'Local Decisions in Decentralised Planning Environments'.
[2]Indian Statistical Institute, Planning Unit, 7, S.J.S. Sansanwal Marg, New Delhi - 110 016, India. Email: dmishra@isid.ac.in
[3]University of Twente, Applied Mathematics, P.O. Box 217, 7500 AE Enschede, The Netherlands. Email: m.uetz@utwente.nl

Dagstuhl Seminar Proceedings 10071
Scheduling
http://drops.dagstuhl.de/opus/volltexte/2010/2540

reports about their private information, a mechanism determines both an order in which jobs are served and for each job a payment that the job receives.

1.1 Our Contribution

We consider two distinct cases. In the *one-dimensional* (1-d) case, service times of jobs are public information and a job's weight is only known to the job itself. In the *two-dimensional* (2-d) case, both weight and service time are private information of the jobs. In either case, we assume the type space of agents to be discrete. Though this is a departure from the traditional literature on auctions (Myerson 1981), it is not entirely uncommon to work with discrete type spaces. For example, some recent progress in deriving optimal auctions for multi-dimensional settings assume the type space to be discrete (Armstrong 2000; Malakhov and Vohra 2007; Pai and Vohra 2008). In particular, with discrete type spaces the computation of the maximum total revenue (or minimum total expenses, respectively) can be cast as a linear programming program. We find this useful in deriving impossibility results.

By a graph theoretic interpretation of the incentive compatibility constraints - as used e.g. by Rochet (1987), Malakhov and Vohra (2007), Müller, Perea, and Wolf (2007), Heydenreich, Müller, Uetz, and Vohra (2009), and many others - we derive optimal mechanisms in a very simple way. For the 1-d case, the result is that serving the jobs in the order of non-increasing ratios of 'virtual' job weights over service times is optimal for the service provider. We also show that this mechanism can be implemented in dominant strategies. It turns out that the optimal mechanism is not necessarily efficient, so in general it does not maximize total utility. But it does so if jobs are symmetric, i.e., have same distribution of weights and equal processing times. Yet, it is worth noting that the optimal mechanism differs from the generalized VCG mechanism, in contrast to auctions even in the symmetric case. For asymmetric jobs, we also show by example that the difference in expenses between an optimal mechanism and an efficient mechanism can be arbitrarily bad.

For the 2-d case, our main result is that the optimal mechanism does not satisfy an independence condition which is known as 'independence of irrelevant alternatives (IIA)' condition. From that we conclude that the optimal mechanism cannot be expressed in terms of modified weights along the lines of the 1-d case. In fact, any kind of priority based scheduling algorithm (e.g., scheduling using Smith's rule with modified weights), where the priorities of a job depend only on the characteristics of that job itself, cannot be an optimal mechanism in general. We conclude that optimal mechanism design for the two-dimensional case at hand is substantially more involved than two-dimensional mechanism design for auction settings, as studied by Malakhov and Vohra (2007) for example. We also show that the optimal mechanism for the 2-d case is not efficient even for symmetric jobs.

1.2 Related Work

Optimal mechanism design goes back to Myerson (1981). He studies optimal mechanism design for single item auctions and continuous, 1-dimensional type spaces. The optimal auction in his setup is to award the object to a bidder who has the highest *virtual valuation*, provided this virtual valuation is non-negative. In the symmetric case, this turns out to be the Vickrey auction with a reserve price. More generally, for single parameter agents the optimal auction is the one that maximizes the total virtual surplus (Hartline and Karlin 2007).

In fact, our work can be seen as analyzing how far the scheduling problem parallels the auction case. In that sense we exactly follow Myerson's approach, but with discrete type spaces, and using a graph-theoretic approach to obtain our results. We observe several close similarities to the auction case, but also some subtle differences. For example, the optimal mechanism for the scheduling problem turns out to be the one that maximizes total virtual surplus, just like for auctions. Yet, for the scheduling problem this result does not fall out of the analysis of the auction case, as the latter builds on revenue equivalence, which does not hold here.

Malakhov and Vohra (2007) derive optimal mechanisms for an auction setting with discrete 2-dimensional type spaces. They consider a multi-unit model where every bidder has a capacity constraint, and the marginal value per unit and capacity are the private type of the bidder. The derived optimal mechanisms also employs the efficient allocation rule with respect to modified (virtual) types. Contrasting their work, we show that for 2-d type spaces, the same graph-theoretic approach must fail to determine an optimal mechanism for the scheduling problem. This because an optimal mechanism is not the efficient mechanism with respect to modified types, no matter how these types are defined. This follows from the fact that an optimal mechanism is in general not IIA.

The fact that optimal mechanism design with multi-dimensional type spaces is harder than with 1-dimensional types is well-known. For example, Armstrong (2000) studies a multi-object auction model where valuations are additive and drawn from a binary distribution (i.e., high or low). He gives optimal auctions under specific conditions that allow to reduce the type graph. It becomes evident from his work that optimal mechanism design with multi-dimensional discrete types is difficult. For our model, we formalize this difficulty by showing that known approaches inevitably yield mechanisms that satisfy the IIA condition, and that in general none of these is optimal.

Scheduling models have been looked at from different perspectives, both in the Economic and Operations Research literature. There are some papers which are closely related to ours with respect to the model considered, but each with a different flavor when it comes to the game theoretic models. For example, Mitra (2001) analyzes efficient and budget balanced mechanism design in a 1-dimensional queueing model, and Kittsteiner and Moldovanu (2005) consider a model in which jobs arrive stochastically, and service time is private information. Moulin (2007) derives mechanisms that prevent merging and splitting of jobs. Suijs (1996) discusses the same sequencing model as ours, and derives results on the existence of payment schemes that are required to be budget balanced. The same problem is discussed from the perspective of cost sharing by Curiel, Pederzoli, and Tijs (1989), and later for m machines by Hamers, Klijn, and Suijs (1999). Yet, none of these papers addresses *optimal mechanism design* in the sense discussed here.

Organization. In Section 2, we study the 1-d discrete case and derive closed formulae for the optimal mechanism. We compare the optimal to the efficient mechanism in Section 3. In Section 4, we study the 2-d discrete case and show that the known approaches are doomed to fail here.

2 Optimal Mechanisms for the 1-Dimensional Setting

Consider a single machine which can handle one job at a time. Let $J = \{1, \dots, n\}$ denote the set of jobs. We regard jobs as selfish agents that act strategically. Each job j has a service time p_j and a weight w_j. While p_j is publicly known, the actual w_j is private information (type) of job j. Jobs share common beliefs about other jobs' types in terms of probability distributions. We assume discrete distribution of weights, that is, agent j's weight w_j follows a probability distribution over the discrete set $W_j = \{w_j^1, \dots, w_j^{m_j}\} \subset \mathbb{R}$, where $w_j^1 < \cdots < w_j^{m_j}$. Let φ_j be the probability distribution of w_j, that is, $\varphi_j(w_j^i)$ denotes the probability associated with w_j^i for $i = 1, \dots, m_j$. Let

$\Phi_j(w_j^i) = \sum_{k=1}^{i} \varphi_j(w_j^k)$ be the cumulative probability up to w_j^i. Probability distributions $\{\varphi_j\}_{j\in J}$ and type spaces $\{W_j\}_{j\in J}$ are public information. We assume that jobs' weights are independently distributed. Let us denote by $W = \Pi_{j\in J} W_j$ the set of all type profiles. For any job j, let $W_{-j} = \Pi_{k\neq j} W_k$. Let φ be the joint probability distribution of $w = (w_1, \dots, w_n)$. Then $\varphi(w) = \Pi_{j=1}^{n} \varphi_j(w_j^{i_j})$ for $w = (w_1^{i_1}, \dots, w_n^{i_n}) \in W$. Let w_{-j} and φ_{-j} be defined analogously. For $w_j^i \in W_j$ and $w_{-j} \in W_{-j}$, we denote by (w_j^i, w_{-j}) the type profile where job j has type w_j^i and types of other jobs are w_{-j}.

2.1 Preliminaries

A *direct revelation mechanism* consists of an *allocation rule* f and a *payment scheme* π. Jobs have to report their weights, and depending on those reports, the allocation rule selects a *schedule*, i.e. an order in which jobs are processed on the machine. The payment scheme assigns a payment that is made to jobs in order to reimburse thfooem for their waiting cost. By the revelation principle, we can restrict attention to such direct mechanisms.

Let $\mathfrak{S} = \{\sigma \,|\, \sigma \text{ is a permutation of } (1, \dots, n)\}$ denote the set of all feasible schedules. Then the allocation rule is a mapping $f\colon W \to \mathfrak{S}$ [4]. For any schedule $\sigma \in \mathfrak{S}$, let σ_j be the position of job j in the ordering of jobs in σ. Then, by $S_j(\sigma) = \sum_{\sigma_k<\sigma_j} p_k$, we denote the start time or waiting time of job j in σ. If job j has waiting time S_j and actual weight w_j^i, it encounters a valuation of $-w_j^i S_j$. If j additionally receives payment π_j, his total utility is $\pi_j - w_j^i S_j$, i.e., we assume quasi-linear utilities. Let us denote by $ES_j(f, w_j^i) := \sum_{w_{-j}\in W_{-j}} S_j(f(w_j^i, w_{-j}))\varphi_{-j}(w_{-j})$ the expected waiting time of job j if it reports weight w_j^i and allocation rule f is applied. Denote by $E\pi_j(w_j^i) := \sum_{w_{-j}\in W_{-j}} \pi_j(w_j^i, w_{-j})\varphi_{-j}(w_{-j})$ the expected payment to j.

Definition 1 *A mechanism (f, π) is* Bayes-Nash incentive compatible (BIC) *if for every agent j and every two types w_j^i,$w_j^k \in W_j$*

$$E\pi_j(w_j^i) - w_j^i ES_j(f, w_j^i) \geq E\pi_j(w_j^k) - w_j^i ES_j(f, w_j^k), \tag{1}$$

where the expectation is done under the assumption that all agents apart from j report truthfully. If for allocation rule f there exists a payment scheme π such that (f, π) is BIC, then f is called Bayes-Nash implementable. *The payment scheme π is referred to as an* incentive compatible *payment scheme.*

Definition 2 *A mechanism (f, π) is* individually rational (IR) *if for every agent j and every type $w_j^i \in W_j$*

$$E\pi_j(w_j^i) - w_j^i ES_j(f, w_j^i) \geq 0. \tag{2}$$

We say an allocation rule f is individually rational if there exists a payment scheme π such that (f, π) is IR.

IR guarantees non-negative expected utilities for all agents that report their true weight. It will be convenient to ensure IR by introducing a so-called dummy weight $w_j^{m_j+1}$, which we add to the type space W_j for every agent j and give it probability 0. We assume $ES_j(f, w_j^{m_j+1}) = 0$ and

[4] If a randomized allocation rule is used, expectations have to be taken according to the randomization of the allocation rule as well. Our results hold as well for such randomized rules.

$E\pi_j(w_j^{m_j+1}) = 0$ for all $j \in J$. Furthermore, we impose the BIC constraints (1) also for $k = m_{j+1}$ which implies (2). Therefore, the dummy weights together with the mentioned assumptions guarantee that IR is satisfied along with the BIC constraints. The dummy weight can be interpreted as an option for any job not to take part in the mechanism.

We next define the notion of monotonicity w.r.t. weights, which is easily shown to be a necessary condition for Bayes-Nash implementability. In our setting, it is even a sufficient condition.

Definition 3 *An allocation rule f satisfies* monotonicity with respect to weights *or short* monotonicity *if for every agent $j \in J$, $w_j^i < w_j^k$ implies that $ES_j(f, w_j^i) \geq ES_j(f, w_j^k)$.*

Theorem 1 *An allocation rule f is Bayes-Nash incentive compatible if and only if it satisfies monotonicity with respect to weights.*

The proof of the theorem is standard, and is given in the Appendix. However, we introduce some basic concepts underlying the proof that are needed later. In particular, we introduce the type graph for the Bayes-Nash setting. Type graph T_j^f has node set W_j and contains an arc (directed edge) from any node w_j^i to any other node w_j^k of length

$$\ell_{ik} = w_j^i[ES_j(f, w_j^k) - ES_j(f, w_j^i)].$$

Here, ℓ_{ik} represents the gain in expected valuation for agent j by truthfully reporting type w_j^i instead of lying type w_j^k, which could be both positive or negative. The incentive constraints for a BIC mechanism (f, π) and job j can be read as

$$E\pi_j(w_j^k) \leq E\pi_j(w_j^i) + w_j^i[ES_j(f, w_j^k) - ES_j(f, w_j^i)] = E\pi_j(w_j^i) + \ell_{ik}.$$

That is, the expected payments $E\pi_j(\cdot)$ constitute a node potential in T_j^f, and therefore Bayes-Nash implementability of an allocation rule f is equivalent to the non-negative cycle property of the type graph T_j^f for any agent j; see for example Müller, Perea, and Wolf (2007). Monotonicity is equivalent to the fact that there is no negative cycle consisting of only two arcs in T_j^f. We call this property the *non-negative two-cycle property.* It follows from

$$\begin{aligned} \ell_{ik} + \ell_{ki} &= w_j^i[ES_j(f, w_j^k) - ES_j(f, w_j^i)] + w_j^k[ES_j(f, w_j^i) - ES_j(f, w_j^k)] \\ &= (w_j^i - w_j^k)[ES_j(f, w_j^k) - ES_j(f, w_j^i)]. \end{aligned}$$

The last term is non-negative for all jobs j and any two types w_j^i and w_j^k if and only if monotonicity holds. The proof of Theorem 1 now boils down to proving that non-negative 2-cycles in the type graph implies non-negative cycles.

2.2 Optimal Mechanisms

Let us start by investigating the *efficient* allocation rule for the given setting, i.e., the allocation rule that minimizes the total waiting costs of agents. It is well known that scheduling in order of non-increasing weight over processing time ratios minimizes the sum of weighted start times $\sum_{j=1}^{n} w_j S_j(f(w))$ for any type profile $w \in W$, and therefore maximizes the total valuation of all agents. This allocation rule is known as Smith's ratio rule (Smith 1956).

Our goal is to set up a mechanism that is BIC and IR, and among all such mechanisms minimizes the expected total payment that has to be made to the jobs. Given any BIC mechanism (f, π), one can obviously substitute the payment scheme by its expected payment scheme yielding $(f, E\pi(\cdot))$ without loosing Bayes-Nash incentive compatibility. Moreover, the expected total payment to the agents remains unchanged under the substitution. Therefore, we may restrict focus to mechanisms in which agents always receive a payment which is independent of the specific report of the other agents and of the actual allocation. We discuss later how to turn this mechanism into a dominant strategy incentive compatible mechanism.

Note that, unlike e.g. in (Myerson 1981; Hartline and Karlin 2007), in the discrete setting considered here revenue equivalence does not hold. Therefore, there are possibly multiple payment schemes that make an allocation rule incentive compatible. Hence, the payment scheme is not uniquely determined by the allocation rule, and we first need to determine the minimum payment for any given allocation rule f.

Let f be any (implemententble) allocation rule and let $\pi^f(\cdot)$ be a payment scheme that minimizes expected expenses for the machine among all payment schemes that make f BIC. More specifically, $\pi_j^f(w_j^i)$ denotes the payment to agent j declaring weight w_j^i under this payment scheme. Then $P^{min}(f) = \sum_{j\in J}\sum_{w_j^i\in W_j}\varphi_j(w_j^i)\pi_j^f(w_j^i)$ is the minimum expected total expenses for allocation rule f. The following lemma specifies π^f.

Lemma 1 *For a Bayes-Nash implementable allocation rule f, the payment scheme defined by*

$$\pi_j^f(w_j^{m_j+1}) = 0, \quad \pi_j^f(w_j^i) = \sum_{k=i}^{m_j} w_j^k[ES_j(f, w_j^k) - ES_j(f, w_j^{k+1})] \text{ for } i = 1, \ldots, m_j$$

is incentive compatible, individually rational and minimizes the expected total payment made to agents. The corresponding expected total payment is given by

$$\begin{aligned} P^{min}(f) &= \sum_{j\in J}\sum_{i=1}^{m_j}\varphi_j(w_j^i)\sum_{k=i}^{m_j} w_k^k\left[ES_j(f, w_j^k) - ES_j(f, w_j^{k+1})\right] \\ &= \sum_{j\in J}\sum_{i=1}^{m_j}\varphi_j(w_j^i)\overline{w}_j^i ES_j(f, w_j^i), \end{aligned}$$

where the virtual weights $\overline{w}_j$ are defined as follows

$$\overline{w}_j^1 = w_j^1, \quad \overline{w}_j^i = w_j^i + (w_j^i - w_j^{i-1})\frac{\Phi_j(w_j^{i-1})}{\varphi_j(w_j^i)} \text{ for } i = 2, \ldots, m_j.$$

The proof of this lemma is in the appendix. The payment $\pi_j^f(w_j^i)$ for type w_j^i is found by taking the negative of the shortest path length from node w_j^i to dummy node $w_j^{m_j+1}$ in the type graph T_j^f. It crucially uses the fact that shortest paths in the type graph T_j^f are independent of the allocation rule f, which follows from the so-called decomposition monotonicity of the type graph; we refer to the appendix for details.

Given the minimum payments for every allocation rule, we want to specify the allocation rule f which minimizes $P^{min}(f)$ among all Bayes-Nash implementable (and IR) allocation rules.

Definition 4 *If* $f \in \arg\min\{P^{min}(f) \,|\, f\colon W \to \mathfrak{S},\ f$ *Bayes-Nash implementable and IR*$\}$*, then we call the mechanism* (f, π^f) *an* optimal mechanism*.*

For the time being, let us impose the following regularity condition that ensures Bayes-Nash implementability of the allocation rule in our candidate mechanism. We will get rid of it afterwards, by using standard arguments.

Definition 5 *We say that* regularity *is satisfied if for every agent* j *and* $i = 1, \ldots, m_j - 1$ $\overline{w}_j^i < \overline{w}_j^k$ *whenever* $w_j^i < w_j^k$*.*

Note that regularity is satisfied e.g. if the differences $w_j^i - w_j^{i-1}$ are constant and the distribution has a non-increasing reverse hazard rate[5].

Theorem 2 *Let the virtual weights* $\overline{w}_j, j \in J$*, and payments* π^f *be defined as in Lemma 1. Let* f *be the allocation rule that schedules jobs in non-increasing order of ratios* $\overline{w}_j/p_j$*. If regularity holds, then* (f, π^f) *is an optimal mechanism.*

Proof. We show that f is Bayes-Nash implementable and minimizes $P^{min}(f)$ among all Bayes-Nash implementable allocation rules. For any allocation rule f, we can rewrite $P^{min}(f)$ as follows, using independence of weight distributions. Let $W'_j = W_j \setminus \{w_j^{m_j+1}\}$ and $W' = \Pi_{j\in J} W'_j$.

$$\begin{aligned}
P^{min}(f) &= \sum_{j\in J} \sum_{w_j^i \in W'_j} \varphi_j(w_j^i)\overline{w}_j^i ES_j(f, w_j^i) \\
&= \sum_{j\in J} \sum_{w_j^i \in W'_j} \varphi_j(w_j^i)\overline{w}_j^i \sum_{w_{-j}\in W_{-j}} S_j(f(w_j^i, w_{-j}))\varphi_{-j}(w_{-j}) \\
&= \sum_{j\in J} \sum_{(w_j^i, w_{-j})\in W'} \varphi(w_j^i,\, w_{-j})\overline{w}_j^i S_j(f(w_j^i, w_{-j})) \\
&= \sum_{w\in W'} \varphi(w) \sum_{j\in J} \overline{w}_j S_j(f(w)).
\end{aligned}$$

Thus, $P^{min}(f)$ can be minimized by minimizing $\sum_{j\in J} \overline{w}_j S_j(f(w))$ for every reported type profile w. This is achieved by using Smith's rule with respect to modified weights, i.e., scheduling in order of non-increasing ratios $\overline{w}_j/p_j$. Under Smith's rule, the expected start time $ES_j(w_j)$ is clearly non-increasing in the modified weight $\overline{w}_j$. The regularity condition ensures that it is non-increasing in the original weights w_j. Therefore, Smith's rule with respect to virtual weights satisfies monotonicity and is hence Bayes-Nash implementable by Theorem 1. This completes the proof. □

2.3 The Non-Regular Case

We need the regularity condition only because we require the mechanism to be monotone. In order to extend the optimal mechanism to the non-regular case, we introduce randomization. Indeed, we can apply a standard procedure known as 'ironing' which was already proposed by Myerson

[5]The reverse hazard rate of the distribution with pdf φ and cdf Φ is defined as $\varphi(x)/\Phi(x)$, see e.g. Krishna (2002).

(1981). Applied to the scheduling problem it works as follows. We virtually 'iron' the possibly non-monotone mapping $w_j \mapsto \overline{w}_j$ at any interval of non-monotonicity. This is achieved by 'flattening' the mapping by adapting some of the virtual weights $\overline{w}_j$. This is exemplified in the following.

Assume that there is a subsequence of virtual weights $I_j^{qr} := \{\overline{w}_j^q, \ldots, \overline{w}_j^r\}$ such that virtual weights $\overline{w}_j^i$ are monotone for $i < q$ and $i > r$. Let $\pi_j^i := \varphi_j(w_j^i)$ and let $\overline{w}_j^{qr} = (\sum_{i=q}^r \pi_j^i \overline{w}_j^i)/\sum_{i=q}^r \pi_j^i$ be the expected virtual weight, conditional on the virtual weight being from I_j^{qr}, and assume further that $\overline{w}_j^{q-1} \le \overline{w}_j^{qr} \le \overline{w}_j^{r+1}$. In this particular case, the randomized allocation rule will assign to job j with a report w_j^i, $q \le i \le r$, virtual weight $\overline{w}_j^l$, $q \le l \le r$, with probability $\pi_j^l/\sum_{i=q}^r \pi_j^i$. For $i < q$ or $i > r$, report w_j^i yields virtual weight $\overline{w}_j^i$, as before. Note that this does not change the expected start time for all other jobs, while the expected start time of job j is now monotone (in fact, constant in interval I_j^{qr}). Thus we obtain a monotone randomized allocation rule. It is a bit cumbersome, though not difficult to show that the payments from Lemma 1 are optimal incentive compatible payments for the randomized rule f, and that the formula for $P^{min}(f)$ in the proof of Theorem 2 remains valid after randomization. Thereby, scheduling along Smith's rule based on randomized virtual weights yields again the optimal mechanism.

The complete construction requires to construct a (finest) partition of agents' weights into subsequences, such that conditional expected virtual weights on the subsequences become monotone. Obviously, in the regular case such a sequence consists of subsequence of size 1. We do not go into further technical details, but rather state the corresponding result somewhat informally, noting that expectations have now to be taken over the possible types of jobs *and* over the random choices of the mechanism that follow from the ironing procedure.

Theorem 3 *Let the virtual weights $\overline{w}_j, j \in J$, and payments π^f be as defined in Lemma 1. Let f be the (randomized) allocation rule that first randomizes the mappings $w_j \mapsto \overline{w}_j$ as suggested by the ironing procedure described above, and then schedules jobs in non-increasing order of ratios $\overline{w}_j/p_j$. Then (f, π^f) is an optimal mechanism.*

Notice that the optimal mechanism in the non-regular case crucially uses randomization; we are not aware of how to get rid of this.

2.4 Implementation in Dominant Strategies

So far, we have discussed Bayes-Nash implementability according to Definition 1, and outcome as well as payment are expected values, the expectation taken over truthful reports of the other jobs. It is a standard question to ask if the optimal mechanism can also be implemented with respect to the much stronger, dominant strategy equilibrium. See for example Manelli and Vincent (2008).

Definition 6 *A mechanism (f, π) is* dominant strategy incentive compatible (DSIC) *if for every agent j and every two types w_j^i,$w_j^k \in W_j$, and any report w_{-j} of other jobs,*

$$\pi_j(w_j^i) - w_j^i S_j(f, (w_j^i, w_{-j})) \ge \pi_j(w_j^k) - w_j^i S_j(f, (w_j^k, w_{-j}))\,. \tag{3}$$

If for allocation rule f there exists a payment scheme π such that (f, π) is DSIC, then f is called dominant strategy implementable. *Mechanism (f, π) is* individual rational *if*

$$\pi_j(w_j^i) - w_j^i S_j(f, (w_j^i, w_{-j})) \ge 0$$

for any report w_{-j} of other jobs.

Clearly, dominant strategy implementability implies Bayes-Nash implementability. The definition of monotonicity, and the fact that implementability is equivalent with monotonicity, translate correspondingly, only replacing the expected waiting time $ES_j(f, w_j)$ by the point-wise waiting time $S_j(f, (w_j, w_{-j}))$, for all w_{-j}. We have the following theorem, which we can easily prove directly by giving the corresponding optimal payment.

Theorem 4 *There exists a mechanism that is dominant strategy incentive compatible and individual rational, and achieves the same total expected payment as the optimal mechanism from Theorems 2 and 3.*

Proof. For any implementable f, a candidate for the optimal DSIC payment can be found in exactly the same way as in the Bayes-Nash case, namely as (negative of) shortest paths in the type graphs T_j^f, only that we now have $|W_{-j}|$ many type graphs for each job j, one for each possible report w_{-j} of the other jobs. The edge lengths in these type graphs are

$$\ell_{ik} = w_j^i \left[S_j(f, (w_j^k, w_{-j})) - S_j(f, (w_j^i, w_{-j})) \right].$$

The resulting payments, for any w_{-j}, are

$$\pi_j(w_j^i, w_{-j}) = \sum\nolimits_{k=i}^{m_j} w_j^k \left[S_j(f, (w_j^k, w_{-j})) - S_j(f, (w_j^{k+1}, w_{-j})) \right].$$

It is an easy exercise to verify incentive compatibility and individual rationality of these payments. If we now compute the total expected payment $P(f)$ of the resulting mechanism (f, π), we get

$$\begin{aligned} P(f) &= \sum_j \sum_{w_{-j}} \varphi(w_{-j}) \sum_i \varphi_j(w_j^i) \sum_{k=i}^{m_j} w_j^k \left[S_j(f, (w_j^k, w_{-j})) - S_j(f, (w_j^{k+1}, w_{-j})) \right] \\ &= \sum_j \sum_i \varphi_j(w_j^i) \sum_{k=i}^{m_j} w_k^k \left[\sum_{w_{-j}} \varphi(w_{-j}) S_j(f, (w_j^k, w_{-j})) - \sum_{w_{-j}} \varphi(w_{-j}) S_j(f, (w_j^{k+1}, w_{-j})) \right] \\ &= \sum_j \sum_i \varphi_j(w_j^i) \sum_{k=i}^{m_j} w_k^k \left[ES_j(f, w_j^k) - ES_j(f, w_j^{k+1}) \right] \\ &= P^{min}(f) \end{aligned}$$

That is, the minimal payments for dominant strategy incentive compatibility are, in expectation, identical to the optimal payments that we computed before for Bayes-Nash implementability. Finally, note that the allocation rule as defined in Theorems 2 and 3 is indeed monotone in w_j for any report w_{-j}.

□

3 Optimality versus Efficiency

In this section, we compare the efficient allocation rule with the optimal allocation rule. First, we generalize the symmetry condition in (Myerson 1981) to our setting, and show that under this condition, the efficient allocation rule is optimal.

Definition 7 *Agents are* symmetric *if* $W_1 = \ldots = W_n$*,* $\varphi_1 = \ldots = \varphi_n$ *and* $p_1 = \ldots = p_n = 1$ *(w.l.o.g.,* $p_j = 1$ *for all* $j \in J$*).*

So, when we say symmetric agents, we require symmetry with respect to private *and* public information.

Corollary 1 *If agents are symmetric and regularity holds, then the optimal mechanism is efficient.*

Proof. If $W_1 = \cdots = W_n = \{w^1, \ldots, w^m\}$ and $\varphi_1 = \cdots = \varphi_n$, then for any two agents j and k, and $i = 1, \ldots, m$, the modified weights are equal, i.e. $\overline{w}_j^i = \overline{w}_k^i$. In the case of regularity, modified weights are non-decreasing in the original weights, and as all $p_j = 1$, scheduling jobs in order of their non-increasing ratios w_j/p_j is equivalent to scheduling them in order of non-increasing ratios $\overline{w}_j/p_j$. That is, the efficient allocation rule and the allocation rule from the optimal mechanism in Theorem 2 coincide. □

Notice that this is no longer true in the non-regular case, as there is positive probability that the optimal mechanism does not schedule the jobs in order of non-increasing ratios w_j/p_j. If weight distributions may differ among agents or if agents have different processing times, then the optimal mechanism is in general not efficient either. In fact, when restricting to efficient mechanisms, the total expected payment can be arbitrarily bad in comparison to the optimal one. This is illustrated by the following two examples.

Example 1 *Let there be two jobs 1 and 2 with* $W_1 = \{M + 1\}$ *and* $W_2 = \{1, M\}$ *for some constant* $M > 2$*. Let* $\varphi_2(1) = 1 - 1/M$*,* $\varphi_2(M) = 1/M$ *and* $p_1 = p_2 = 1$*. Let Eff be the efficient and Opt be the optimal allocation rule. Then the ratio* $P^{min}(\mathit{Eff})/P^{min}(\mathit{Opt})$ *goes to infinity as* M *goes to infinity.*

Proof. The efficient allocation rule, Smith's rule, always allocates job 1 first. So the optimal payment for Smith's rule is to pay 0 to job 1 and to pay M to job 2, irrespective of its type. The minimum expected total payment is hence $P^{min}(Eff) = M$. For the optimal allocation, we compute modified weights after Lemma 1: $\overline{w}_1^1 = w_1^1 = M + 1$, $\overline{w}_2^1 = w_2^1 = 1$ and $\overline{w}_2^2 = M + (M-1)(1-1/M)/(1/M) = M^2 - M + 1$. The latter is larger than $M + 1$ if $M > 2$. Therefore, job 2 is scheduled in front of job 1 if he has weight M and behind if he has weight 1. The expected start times for job 2 are $ES_2(\mathit{Opt}, 1) = 1$ and $ES_2(\mathit{Opt}, M) = 0$, respectively. Optimal payments according to Lemma 1 are $\pi_2^{\mathit{Opt}}(1) = 1$ and $\pi_2^{\mathit{Opt}}(M) = 0$. For job 1, the expected start time is $ES_1(\mathit{Opt}, M + 1) = 1/M$ and the expected payment $\pi_1^{\mathit{Opt}}(M + 1) = 1 + 1/M$. Hence, $P^{min}(\mathit{Opt}) = 1 + 1/M + 1 \cdot (1 - 1/M) = 2$. Consequently, $P^{min}(Eff)/P^{min}(\mathit{Opt}) = M/2 \to \infty$ for $M \to \infty$. □

Example 2 *Let there be two jobs 1 and 2 with the same weight distribution* $W_1 = W_2 = \{1, M\}$*,* $\varphi_j(1) = 1 - 1/M$*,* $\varphi_j(M) = 1/M$ *for* $j = 1, 2$*. Let* $p_1 = 1/2$ *and* $p_2 = M/2 + 1$ *for some* $M > 2$*. Let Eff be the efficient and Opt be the optimal allocation rule. Then the ratio* $P^{min}(\mathit{Eff})/P^{min}(\mathit{Opt})$ *goes to infinity as* M *goes to infinity.*

Proof. The efficient allocation rule always schedules job 1 first, since $1/(1/2) = 2 > 2M/(M + 2) = M/(M/2 + 1)$. Therefore, the expected start time of job 1 is 0 and that of job 2 is 1/2. Optimal payments according to Lemma 1 are $\pi_1^{Eff}(1) = \pi_1^{Eff}(M) = 0$ and $\pi_2^{Eff}(1) = \pi_2^{Eff}(1) = M/2$. Hence, $P^{min}(Eff) = M/2$.

For the optimal mechanism, we compute modified weights as $\overline{w}_1^1 = \overline{w}_2^1 = 1$ and $\overline{w}_1^2 = \overline{w}_2^2 = M^2 - M + 1$. Job 1 is scheduled first, whenever both jobs have the same weight or job 1 has a larger weight than job 2. In the case where job 1 has (modified) weight 1 and job 2 has modified weight $M^2 - M + 1$, job 2 is scheduled first for $M > 2$, since $1/(1/2) < (M^2 - M + 1)/(M/2 + 1)$. The resulting expected start times and payments are given below:

$$\begin{array}{ll} ES_1(Opt, 1) = 1/2 + 1/M & \pi_1^{Opt}(1) = 1/2 + 1/M \\ ES_1(Opt, M) = 0 & \pi_1^{Opt}(M) = 0 \\ ES_2(Opt, 1) = 1/2 & \pi_2^{Opt}(1) = 1 - 1/(2M) \\ ES_2(Opt, M) = 1/(2M) & \pi_2^{Opt}(M) = 1/2. \end{array}$$

Hence,

$$\begin{aligned} P^{min}(Opt) &= (\frac{1}{2} + \frac{1}{M})(1 - \frac{1}{M}) + (1 - \frac{1}{2M})(1 - \frac{1}{M}) + \frac{1}{2} \cdot \frac{1}{M} \\ &= (1 - \frac{1}{M})(\frac{3}{2} + \frac{1}{2M}) + \frac{1}{2} \cdot \frac{1}{M}. \end{aligned}$$

Thus, the ratio $P^{min}(Eff)/P^{min}(Opt)$ tends to infinity if M tends to infinity. □

Comparison to Myerson's result. For the single item auction and continuous type spaces, Myerson (1981) has made similar observations: in his setting, the Vickrey auction is an efficient auction. The optimal auction can be seen as a Vickrey auction with a reserve price. In our setting also, the allocation in the optimal mechanism is equivalent to the efficient allocation rule with respect to modified data. Nevertheless, in Myerson (1981) the optimal and the efficient mechanism may differ. For the single item auction this can be due to the seller keeping the item (even in the symmetric case) or because a bidder that has not submitted the highest bid can get the item in the asymmetric case. In our setting, the optimal and the efficient mechanism can only differ if agents are asymmetric, see Corollary 1 and Examples 1 and 2.

On the generalized VCG Mechanism. The VCG mechanism is due to Vickrey (1961), Clarke (1971) and Groves (1973). The allocation rule is the efficient one. In our setting this means scheduling in order of non-increasing ratios w_j/p_j. The payment scheme pays to agent j an amount that is equal to an appropriate constant (possibly depending on other agents' types, but not on j's type) minus the total loss in valuation of the other agents due to j's presence. For agent j with processing time p_j, the total loss in valuation of the other agents is equal to the product of p_j and the total weight of all agents processed after j. In order to ensure individual rationality, we have to add p_j times the total weight of all agents except j. Therefore, the resulting payment to j for reported type profile w and efficient schedule σ is equal to

$$\pi_j^{VCG}(w) = p_j \sum_{\substack{k \in J \\ \sigma_k < \sigma_j}} w_k.$$

As illustrated by Examples 1 and 2, the allocation of the VCG mechanism can differ from the allocation of the optimal mechanism if agents are not symmetric. Moreover, if agents are symmetric, the VCG mechanism still can be non-optimal in terms of payments. This is illustrated by the following example.

Example 3 *There are two symmetric agents with* $W_1 = W_2 = \{w^1, w^2\}$*,* $w^1 < w^2$*, and* $\varphi_j(w^1) = \varphi_j(w^2) = 1/2$ *for* $j = 1, 2$*. Processing times are equal and without loss of generality* $p_1 = p_2 = 1$*. Then the expected expenses of the VCG mechanism are strictly higher than those of the optimal mechanism.*

Proof. Regularity is trivially satisfied and therefore the allocation of the optimal mechanism from Section 2 is efficient. There are four possible type profiles, each occurring with probability 1/4: (w^1, w^1), (w^1, w^2), (w^2, w^1), (w^2, w^2). The resulting schedules are the same for the VCG and the optimal mechanism and schedule the job with the higher weight first and break ties arbitrarily in the case of equal weights, respectively. Let us first compute the expected total payment for the VCG mechanism. The VCG mechanism pays to the job that is scheduled last the weight of the job that is scheduled before him. Thus, the VCG mechanism has to spend w^1 in the first case, and w^2 in the second, third and fourth case, respectively. The total expected payment of the VCG mechanism is hence $(3w^2 + w^1)/4$. Let (f, π^f) denote the optimal mechanism from Section 2. In the optimal mechanism, the expected payment to a job with weight w^1 is equal to $E\pi_j^f(w^1) = w^1[ES_j(f, w^1) - ES_j(f, w^2)] + w^2 ES_j(f, w^2) = w^1[3/4 - 1/4] + w^2[1/4] = w^1/2 + w^2/4$. The expected payment to a job with weight w^2 is $E\pi_j^f(w^2) = w^2 ES_j(f, w^2) = w^2/4$. The total expected payment for the optimal mechanism is thus $2 \cdot 1/2 \cdot (w^1/2 + w^2/4 + w^2/4) = (w^1 + w^2)/2$. Since $w^2 > w^1$, the expected expenses of the VCG mechanism are strictly higher than those of the optimal mechanism. Therefore, the VCG mechanism is not optimal. □

4 The 2-Dimensional Setting

4.1 Setting and Notation

In contrast to the 1-dimensional setting, both weight and processing time of a job are now private information of the job. Hence j's type is the tuple (w_j, p_j). As before, we restrict attention to discrete type spaces, i.e., $(w_j, p_j) \in W_j \times P_j$, where $W_j = \{w_j^1, \ldots, w_j^{m_j}\}$ with $w_j^1 < \cdots < w_j^{m_j}$ and $P_j = \{p_j^1, \ldots, p_j^{q_j}\}$ with $p_j^1 < \cdots < p_j^{q_j}$. Let φ_j be the probability distribution of j's type, that is, $\varphi_j(w_j^i, p_j^k)$ denotes the probability associated with the type (w_j^i, p_j^k) for $i = 1, \ldots, m_j$ and $k = 1, \ldots, q_j$. Probability distributions $\{\varphi_j\}_{j \in J}$ and type space $\{W_j \times P_j\}_{j \in J}$ are publicly known. Distributions are independent between agents. Denote by $T = \Pi_{j \in J}(W_j \times P_j)$ the set of all type profiles. For any job j, let $T_{-j} = \Pi_{r \neq j}(W_r \times P_r)$ be the set of type profiles of all jobs except j. Let φ be the joint probability distribution of $(w_1, p_1, \ldots, w_n, p_n)$. Then for type profile $t = (w_1^{i_1}, p_1^{k_1}, \ldots, w_n^{i_n}, p_n^{k_n}) \in T$, $\varphi(t) = \Pi_{j=1}^n \varphi_j(w_j^{i_j}, p_j^{k_j})$. Let t_{-j} and φ_{-j} be defined analogously. For $(w_j^i, p_j^k) \in W_j \times P_j$ and $t_{-j} \in T_{-j}$, we denote by $((w_j^i, p_j^k), t_{-j})$ the type profile where job j has type (w_j^i, p_j^k) and the types of the other jobs are represented by t_{-j}. Denote by

$$ES_j(f, w_j^i, p_j^k) := \sum_{t_{-j} \in T_{-j}} S_j(f((w_j^i, p_j^k), t_{-j}))\varphi_{-j}(t_{-j})$$

the expected waiting time of job j if he reports type (w_j^i, p_j^k) and allocation rule f is applied. Denote by

$$E\pi_j(w_j^i, p_j^k) := \sum_{t_{-j} \in T_{-j}} \pi_j((w_j^i, p_j^k), t_{-j})\varphi_{-j}(t_{-j})$$

the expected payment to j.

We assume that an agent can only report a processing time that is not lower than his true processing time and that a job is processed for his reported processing time. We believe this is a natural assumption, as reporting a shorter processing time can be easily punished by preempting the job after the declared processing time, before it is actually finished.

4.2 Bayes-Nash Implementability and the Type Graph

Definition 8 *A mechanism (f, π) is called* Bayes-Nash incentive compatible *(BIC) if for every agent j and every two types $(w_j^{i_1}, p_j^{k_1})$ and $(w_j^{i_2}, p_j^{k_2})$ with $i_1, i_2 \in \{1, \dots, m_j\}$, $k_1, k_2 \in \{1, \dots, q_j\}$, $k_1 \leq k_2$,*

$$E\pi_j(w_j^{i_1}, p_j^{k_1}) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^{k_1}) \geq E\pi_j(w_j^{i_2}, p_j^{k_2}) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^{k_2}), \tag{4}$$

where the expectations are taken under the assumption that all agents apart from j report truthfully.

Note that by defining the incentive constraints only for $k_1 \leq k_2$, we account for the fact that agents can only overstate their processing time, but cannot understate it.

In order to ensure individual rationality, again add a dummy type t_j^d to the type space for every agent j, and let $ES_j(f, t_j^d) = 0$ and $E\pi_j(t_j^d) = 0$ for all $j \in J$. As in the 1-dimensional case, the dummy types together with the mentioned extra incentive constraints guarantee that individual rationality is satisfied along with the incentive constraints. Sometimes, it will be convenient to write $(w_j^{m_j+1}, p_j^k)$ for some $k \in \{1, \dots, q_j\}$ instead of t_j^d.

In the 2-dimensional setting, the type graph T_j^f of agent j has node set $W_j \times P_j$, where we add $w_j^{m_j+1} \in W_j$ for IR, and contains an arc from any node $(w_j^{i_1}, p_j^{k_1})$ to every other node $(w_j^{i_2}, p_j^{k_2})$ with $i \in \{1, \dots, m_j\}$, $i_2 \in \{1, \dots, m_j + 1\}$, $k \in \{1, \dots, q_j\}$, $k_1 \leq k_2$ of length

$$\ell_{(i_1 k_1)(i_2 k_2)} = w_j^{i_1} [ES_j(f, w_j^{i_2}, p_j^{k_2}) - ES_j(f, w_j^{i_1}, p_j^{k_1})].$$

Note that we have arcs only in direction of increasing processing times, since agents can only overstate their processing time. Furthermore, every node has an arc to the dummy type, but there are no outgoing arcs from the dummy type.

Similar as in Malakhov and Vohra (2007), one can show that for monotonic allocation rules some arcs in the type graph are not necessary, since the corresponding incentive constraints are implied by others. We first give the definition of monotonicity in the 2-dimensional setting and then formulate a lemma which reduces the set of necessary incentive constraints.

Definition 9 *An allocation rule f satisfies* monotonicity with respect to weights *if for every agent $j \in J$ and fixed $p_j^k \in P_j$, $w_j^{i_1} < w_j^{i_2}$ implies that $ES_j(f, w_j^{i_1}, p_j^k) \geq ES_j(f, w_j^{i_2}, p_j^k)$.*

Lemma 2 *Let f be an allocation rule satisfying monotonicity with respect to weights. For any agent j, the following constraints imply all other incentive constraints:*

$$\begin{aligned}
E\pi_j(w_j^i, p_j^k) - w_j^i ES_j(f, w_j^i, p_j^k) &\geq E\pi_j(w_j^{i+1}, p_j^k) - w_j^i ES_j(f, w_j^{i+1}, p_j^k) && (5)\\
&\quad \textit{for } i \in \{1, \dots, m_j\}, k \in \{1, \dots, q_j\} \\
E\pi_j(w_j^{i+1}, p_j^k) - w_j^{i+1} ES_j(f, w_j^{i+1}, p_j^k) &\geq E\pi_j(w_j^i, p_j^k) - w_j^{i+1} ES_j(f, w_j^i, p_j^k) && (6)\\
&\quad \textit{for } i \in \{1, \dots, m_j - 1\}, k \in \{1, \dots, q_j\} \\
E\pi_j(w_j^i, p_j^k) - w_j^i ES_j(f, w_j^i, p_j^k) &\geq E\pi_j(w_j^i, p_j^{k+1}) - w_j^i ES_j(f, w_j^i, p_j^{k+1}) && (7)\\
&\quad \textit{for } i \in \{1, \dots, m_j\}, k \in \{1, \dots, q_j - 1\}
\end{aligned}$$

The proof is given in the Appendix. Lemma 2 is in fact a generalization of *decomposition monotonicity* as discussed for the 1-dimensional case in the proof of Theorem 1. We can now define the reduced type graph of agent j, which contains only arcs that are necessary in the sense of Lemma 2. These arcs are:

- an arc from type (w_j^i, p_j^k) to (w_j^{i+1}, p_j^k) for all $i \in \{1, \dots, m_j\}$ and $k \in \{1, \dots, q_j\}$
- an arc from type (w_j^{i+1}, p_j^k) to (w_j^i, p_j^k) for all $i \in \{1, \dots, m_j - 1\}$ and $k \in \{1, \dots, q_j\}$
- an arc from type (w_j^i, p_j^k) to (w_j^i, p_j^{k+1}) for all $i \in \{1, \dots, m_j\}$ and $k \in \{1, \dots, q_j - 1\}$.

A sketch of the reduced type graph is given in Figure 1. Expected payments correspond to negative of shortest paths in the reduced type graph. Whenever we refer to the type graph for a monotonic allocation rule in the following, we mean the reduced type graph. The reduced type graph comes handy particularly when considering our (counter) examples in the next subsection.

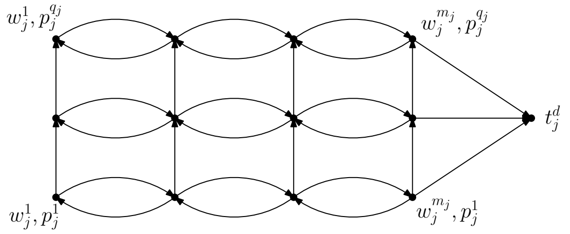

Figure 1: Reduced type graph

We finally give the characterization of BIC allocation rules for the 2-dimensional setting, which is a consequence of our restriction of the strategy space for each job (i.e., the assumption that no job can understate its required service time).

Theorem 5 *An allocation rule f is BIC in the 2-dimensional setting if and only if it satisfies monotonicity with respect to weights.*

Proof. Implementability implies monotonicity as before. The claim reduces to showing that in the type graph of any agent j the non-negative cycle property is equivalent to the non-negative two-cycle property. After the reduction, every cycle in T_j^f consists of a finite number of two-cycles. Hence the non-negative cycle property is equivalent to the non-negative two-cycle property. □

4.3 On Optimal Mechanisms

Given the successful and elegant approach by Malakhov and Vohra (2007) for an auction setting with 2-dimensional type spaces, it is tempting to try and use the network approach also in the 2-dimensional setting for the scheduling problem. In this section, we show that this won't work.

A first problem is that, in contrast to the 1-dimensional case, the shortest paths in the type graph may now depend on the allocation rule f, which was not the case before. Hence, we cannot express minimum payments in a closed formula. Exactly this problem is ruled out by Malakhov and Vohra (2007) in their 2-dimensional auction problem. But it turns out that the situation here is worse.

Definition 10 *We say that an allocation rule f satisfies* independence of irrelevant alternatives *(IIA) if the relative order of any two jobs j_1 and j_2 is the same in the schedules $f(s)$ and $f(t)$ for any two type profiles $s, t \in T$ that differ only in the types of agents from $J \setminus \{j_1, j_2\}$.*

Note that the IIA axiom has no bite if number of jobs is two. Another way to state the IIA axiom is that the relative order of two jobs is independent of all other jobs. For the 2-d setting, this is not necessarily the case for optimal mechanisms.

Theorem 6 *The optimal allocation rule for the 2-dimensional setting does not satisfy IIA.*

Proof. Consider the following instance with three jobs. Job 1 has type space $\{(1,1)\}$, job 2 has type space $\{(2,2)\}$ and job 3 has type space $\{1.9, 2\} \times \{1, 2\}$. The probabilities for job 3's types are $\varphi_3(1.9,1) = 0.8$, $\varphi_3(2,2) = 0.2$ and $\varphi_3(1.9,2) = \varphi_3(2,1) = 0$ respectively. The following argumentation would still work if we assumed small positive probabilities for types $(1.9,2)$ and $(2,1)$ as well, but everything would become much more technical. We will show that the best allocation rule that satisfies IIA achieves a minimum expected total payment of at least 5.6, whereas there exists an allocation rule – violating IIA – with an expected total payment of 4.88.

There are six possible schedules for three jobs, where we denote e.g. by 312 the schedule where job 3 comes first and job 2 last. There are only two cases that occur with positive probability: job 3 has type $(1.9,1)$, which we refer to as case a, and job 3 has type $(2,2)$, which we refer to as case b. An allocation rule that satisfies IIA must schedule job 1 and 2 in the same relative order in case a and b. Therefore, any such rule must either choose a schedule from $\{123, 132, 312\}$ or from $\{213, 231, 321\}$ in both cases.

As an example, we compute a lower bound on the optimal payment $P^{min}(f)$ for the case where f chooses schedule 123 in case a and schedule 132 in case b. Since there is only one possible type for job 1 and 2, only individual rationality matters for the optimal payments to those jobs and hence $\pi_1^f(1,1) = 0$ and $\pi_2^f(2,2) = 2(0.8 \cdot 1 + 0.2 \cdot (1+2)) = 2.8$. For job 3, we take individual rationality into account as well as the incentive constraint $\pi_3^f(1.9,1) - 1.9 \cdot ES_3(1.9,1) \geq \pi_3^f(2,2) - 1.9 \cdot ES_3(2,2)$. While individual rationality requires $\pi_3^f(1.9,1) \geq 1.9 \cdot 3 = 5.7$ and $\pi_3^f(2,2) \geq 2$, the latter is equivalent to $\pi_3^f(1.9,1) \geq \pi_3^f(2,2) + 3.8$. Therefore, $\pi_3^f(2,2) \geq 2$ and $\pi_3^f(1.9,1) \geq 5.8$. Hence $P^{min}(f) \geq 2.8 + 0.8 \cdot 5.8 + 0.2 \cdot 2 = 7.84$. Note that this is only a lower bound, since for the exact value of $P^{min}(f)$, we must additionally consider the incentive constraints that result from the two types $(1.9,2)$ and $(2,1)$, which have zero probability, but are in the type space of job 3.

In total, there are 18 allocation rules that satisfy IIA. We list the corresponding lower bounds (LB) on $P^{min}(f)$ in Table 1.

Hence, 5.6 is a lower bound for the expected total payment made by any IIA mechanism. On the other hand, consider the allocation rule that chooses schedule 132 in case a and schedule 231 in case b. We extend the allocation rule to the zero probability type such that it chooses schedule 132 for type $(2,1)$ and schedule 231 for type $(1.9,2)$. Clearly, this allocation rule violates IIA. The optimal payments to job 1 and 2 are $\pi_1^f(1,1) = 0.8$ and $\pi_2^f(2,2) = 1.6$ respectively. For the optimal payment to job 3, we depict the type graph with associated arc lengths in Figure 2. The shortest path lengths from $(1.9,1)$ and $(2,2)$ to the dummy node are -2.1 and -4, respectively. Hence, $\pi_3^f(1.9,1) = 2.1$ and $\pi_3^f(2,2) = 4$. Consequently, $P^{min}(f) = 0.8 + 1.6 + 0.8 \cdot 2.1 + 0.2 \cdot 4 = 4.88$. This proves the claim. □

Theorem 6 shows that any priority based algorithm where the priority of a job is computed from the characteristics of the job itself cannot be optimal in general. We can conclude that the

$f(a)$	$f(b)$	π_1^f	π_2^f	LB $\pi_3^f(1.9,1)$	LB $\pi_3^f(2,2)$	LB $P^{min}(f)$
123	123	0	2	6	6	8
123	132	0	2.8	5.8	2	7.84
123	312	0.4	2.8	5.7	0	7.76
132	123	0	3.6	2.2	6	6.56
132	132	0	4.4	2	2	6.4
132	312	0.4	4.4	1.9	0	6.32
312	123	0.8	3.6	0.3	6	5.84
312	132	0.8	4.4	0.1	2	5.68
312	312	1.2	4.4	0	0	5.6
213	213	2	0	6	6	8
213	231	2.4	0	5.9	4	7.92
213	321	2.4	0.8	5.7	0	7.76
231	213	2.8	0	4.1	6	7.28
231	231	3.2	0	4	4	7.2
231	321	3.2	0.8	3.8	0	7.04
321	213	2.8	1.6	0.3	6	5.84
321	231	3.2	1.6	0.2	4	5.76
321	321	3.2	2.4	0	0	5.6

Table 1: Lower bounds for payments of different schedules

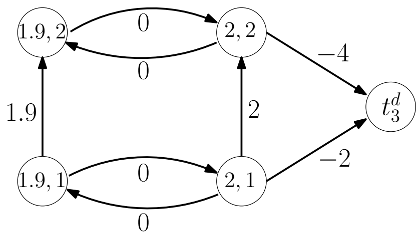

Figure 2: type graph job 3

network approach which we used for the 1-dimensional case, and which is used also by Malakhov and Vohra (2007), is doomed to fail in the 2-dimensional case we consider here. This because the network approach will inevitably lead to an allocation rule f that is IIA[6]. But in general f is not of that form; see Theorem 6.

One explanation for this complication may lie in the fact that the 2-d setting considered here in fact entails informational externalities, as opposed to the auction setting of Malakhov and Vohra (2007). On the other hand, the informational externalities introduced by private processing times are not the only cause for complications in the 2-dimensional setting: Consider the 1-dimensional setting, where only the processing times are private, but the weights are public information. It turns out that all allocation rules are implementable, even when we allow that jobs understate

[6]The type graph T_j^f for job j is defined on the basis of the data of this job only. For any given f, minimum payments correspond to (the negative of) shortest path lengths in T_j^f. Now the total expected payment $P^{min}(f)$ is a linear function in the values $ES_j(f, w_j^i)$, and the coefficients of any $ES_j(f, w_j^i)$ will depend on the data of job j only, by construction. Hence, the term $P^{min}(f)$ is minimized only if f equals Smith's ratio rule, for some virtual weights $\overline{w}_j$, $j \in J$ (even if we fail to express these $\overline{w}_j$ in closed form).

Figure 3: Type graphs for the w-rule for jobs 1 and 2

their processing times. The optimal payment to a job j that reports processing time p_j^k is equal to $w_j ES_j(f, p_j^k)$, and therefore the total payment to jobs for allocation rule f is equal to $P^{min}(f) = \sum_{j\in J}\sum_{k=1}^{q_j}\varphi_j(p_j^k)w_j ES_j(f, p_j^k)$. This is minimized by Smith's ratio rule.

When there are only two agents present, then IIA is trivially satisfied. Recall that in the 1-dimensional case the optimal mechanism is efficient for symmetric agents and regular distributions and that the uniform distribution is regular. This is contrasted by the following theorem for the 2-dimensional case.

Theorem 7 *Even for two symmetric agents,* 2×2*-type spaces and uniform probability distributions, the optimal mechanism is not efficient.*

Proof. Consider the following example with two jobs, $W_1 = W_2 = \{1, 2\}$ and $P_1 = P_2 = \{1, 2\}$. We assume that $\varphi_1(i, k) = \varphi_2(i, k) = \frac{1}{4}$ for $i, k \in \{1, 2\}$. On one hand, consider the efficient allocation rule f_e, which schedules the job with higher weight over processing time ratio first. On the other hand, regard the so-called w-rule, f_w, that schedules the job with the higher weight first. In case of ties, both rules schedule job 1 first. The expected start times are listed below.

$$ES_1(f_w, 1, 1) = ES_1(f_w, 1, 2) = 3/4 \qquad ES_2(f_w, 1, 1) = ES_2(f_w, 1, 2) = 3/2$$
$$ES_1(f_w, 2, 1) = ES_1(f_w, 2, 2) = 0 \qquad ES_2(f_w, 2, 1) = ES_2(f_w, 2, 2) = 3/4$$

$$ES_1(f_e, 1, 1) = ES_1(f_e, 2, 2) = 1/4, \qquad ES_2(f_e, 1, 1) = ES_2(f_e, 2, 2) = 1,$$
$$ES_1(f_e, 1, 2) = 1, \qquad ES_2(f_e, 1, 2) = 3/2,$$
$$ES_1(f_e, 2, 1) = 0, \qquad ES_2(f_e, 2, 1) = 1/4.$$

The type graphs corresponding to f_w for job 1 and 2 respectively are shown in Figure 3. From this, the optimal payments can be computed as:

$$\pi_1^{f_w}(2, 1) = \pi_1^{f_w}(2, 2) = 0, \qquad \pi_2^{f_w}(2, 1) = \pi_2^{f_w}(2, 2) = 3/2,$$
$$\pi_1^{f_w}(1, 1) = \pi_1^{f_w}(1, 2) = 3/4, \qquad \pi_2^{f_w}(1, 1) = \pi_2^{f_w}(1, 2) = 9/4.$$

Hence the (minimum) total expected payment for the w-rule is:

$$P^{min}(f_w) = \frac{1}{4}\sum_j \sum_{(i,k)} \pi_j^{f_w}(i, k) = 9/4.$$

Figure 4: Type graphs for the efficient rule for job 1 and 2

The type graphs corresponding to f_e for agent 1 and 2 respectively are shown in Figure 4. From this, the optimal payments can be computed as:

$$\begin{aligned} \pi_1^{f_e}(1,1) &= \pi_1^{f_e}(2,2) = 1/2, \\ \pi_1^{f_e}(2,1) &= 0 \\ \pi_1^{f_e}(1,2) &= 5/4, \end{aligned}$$

$$\begin{aligned} \pi_2^{f_e}(1,1) &= \pi_2^{f_e}(2,2) = 2, \\ \pi_2^{f_e}(1,2) &= 5/2, \\ \pi_2^{f_e}(2,1) &= 1/2. \end{aligned}$$

Hence the (minimum) total expected payment in the efficient rule is:

$$P^{min}(f_e) = \frac{1}{4} \sum_j \sum_{(i,k)} \pi^j(i,k) = 37/16.$$

Hence, $P^{min}(f_e) > P^{min}(f_w)$. Thus, the efficient allocation is dominated by the w-rule, and consequently does not correspond to the optimal mechanism. □

5 Discussion

We have seen that the graph theoretic approach is an intuitive tool for optimal mechanism design and yields a closed formula for the optimal mechanism in the 1-dimensional case. The results parallel Myerson's results for single item auctions; although there are some subtle differences.

Moreover, we have seen that in the two-dimensional case the canonical (network) approach does not work and that optimal mechanism design seems to be considerably more complicated than in some auction models. We leave it as an open problem to identify (closed formulae for) optimal mechanisms for the 2-d case. It is conceivable, however, that closed formulae do not exist, and the problem may be provably harder than the one-dimensional problem.

Acknowledgements

Thanks to Rakesh Vohra and Jason Hartline for some very helpful remarks on the paper.

References

Armstrong, M. (2000). Optimal multi-object auctions. *Review of Economic Studies 67*, 455–481.

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

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

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

Hamers, H., F. Klijn, and J. Suijs (1999). On the balancedness of multiple machine sequencing games. *European Journal of Operational Research 119*, 678–691.

Hartline, J. and A. Karlin (2007). Profit maximization in mechanism design. In N. Nisan, T. Roughgarden, É. Tardos, and V. Vazirani (Eds.), *Algorithmic Game Theory*. Cambridge University Press.

Heydenreich, B., R. Müller, M. Uetz, and R. Vohra (2009). Characterization of revenue equivalence. *Econometrica 77*(1), 307–316.

Kittsteiner, T. and B. Moldovanu (2005). Priority auctions and queue disciplines that depend on processing time. *Management Science 51*(2), 236–248.

Krishna, V. (2002). *Auction Theory*. Academic Press.

Malakhov, A. and R. Vohra (2007). An optimal auction for capacity constrained bidders: a network perspective. *Economic Theory*.

Manelli, A. M. and D. Vincent (2008). Bayesian and dominant strategy implementation in the independent private values model. Working paper, http://terpconnect.umd.edu/~dvincent/.

Mitra, M. (2001). Mechanism design in queueing problems. *Economic Theory 17*(2), 277–305.

Moulin, H. (2007). On scheduling fees to prevent merging, splitting, and transferring of jobs. *Mathematics of Operations Research 32*, 266–283.

Müller, R., A. Perea, and S. Wolf (2007). Weak monotonicity and Bayes-Nash incentive compatibility. *Games and Economic Behavior 61*(2), 344–358.

Myerson, R. (1981). Optimal auction design. *Mathematics of Operations Research 6*(1), 58–73.

Pai, M. and R. Vohra (2008). Optimal auctions with financially constrained bidders. Discussion Paper 1471, The Center for Mathematical Studies in Economics & Management Sciences, Northwestern University, Evanston, IL.

Rochet, J.-C. (1987). A necessary and sufficient condition for rationalizability in a quasi-linear context. *Journal of Mathematical Economics 16*(2), 191–200.

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

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

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

Appendix

Proof of Theorem 1. All that remains to show is that the non-negative two-cycle property implies the non-negative cycle property. We first show that the arc lengths satisfy a property called *decomposition monotonicity*, i.e., whenever $i < k < l$ then $\ell_{ik} + \ell_{kl} \leq \ell_{il}$ and $\ell_{lk} + \ell_{ki} \leq \ell_{li}$.

Decomposition monotonicity follows from

$$\begin{aligned}
\ell_{ik} + \ell_{kl} &= w_j^i[ES_j(f, w_j^k) - ES_j(f, w_j^i)] + w_j^k[ES_j(f, w_j^l) - ES_j(f, w_j^k)] \\
&\leq w_j^i[ES_j(f, w_j^k) - ES_j(f, w_j^i)] + w_j^i[ES_j(f, w_j^l) - ES_j(f, w_j^k)] \\
&= w_j^i[ES_j(f, w_j^l) - ES_j(f, w_j^i)] = \ell_{il},
\end{aligned}$$

where the inequality follows from monotonicity. Note that everything remains true if the dummy type is involved, i.e., if $l = m_j + 1$. The inequality $\ell_{lk} + \ell_{ki} \leq \ell_{li}$ follows similarly.

Because of decomposition monotonicity, length of any edge (w_j^i, w_j^k) with $i - k > 1$ can be lower bounded by edges $(w_j^i, w_j^{i+1}), (w_j^{i+1}, w_j^{i+2}), \ldots, (w_j^{k-1}, w_j^k)$ (call such edges *neighboring edges*). Similarly, length of any edge (w_j^i, w_j^k) with $k - 1 > 1$ can be lower bounded by neighboring edges. Hence, any cycle can be lower bounded by the lengths of a finite number of two cycles, which proves the theorem. □

Proof of Lemma 1. Let $\mathfrak{p} = (w_j^i = a_0, a_1, \ldots, a_m = w_j^{m_j+1})$ denote a path from w_j^i to $w_j^{m_j+1}$ in the type graph T_j^f for agent j. Denote by $length(\mathfrak{p})$ the sum of its arc lengths. Let (f, π) be a Bayes-Nash incentive compatible mechanism. Adding up the incentive constraints

$$E\pi_j(a_i) \leq E\pi_j(a_{i-1}) + a_{i-1}[ES_j(f, a_i) - ES_j(f, a_{i-1})] = E\pi_j(a_{i-1}) + \ell_{a_{i-1}a_i}$$

for $i = 1, \ldots, m$ yields

$$E\pi_j(w_j^{m_j+1}) \leq E\pi_j(w_j^i) + length(\mathfrak{p}).$$

Assuming $E\pi_j(w_j^{m_j+1}) = 0$, this is equivalent to $-length(\mathfrak{p}) \leq E\pi_j(w_j^i)$. As f is Bayes-Nash implementable, T_j^f satisfies the non-negative cycle property. Consequently, we can compute shortest paths in T_j^f. With $dist(w_j^i, w_j^{m_j+1})$ being the length of a shortest path from w_j^i to $w_j^{m_j+1}$, the above yields $-dist(w_j^i, w_j^{m_j+1}) \leq E\pi_j(w_j^i)$. Therefore, $-dist(w_j^i, w_j^{m_j+1})$ is a lower bound on the expected payment for reporting w_j^i. On the other hand, since we have

$$dist(w_j^i, w_j^{m_j+1}) \leq \ell_{ik} + dist(w_j^k, w_j^{m_j+1})$$

for any two types w_j^i and w_j^k, it follows that

$$-dist(w_j^k, w_j^{m_j+1}) \leq -dist(w_j^i, w_j^{m_j+1}) + \ell_{ik}.$$

Consequently, setting $\pi_j^f(w_j^i) = -dist(w_j^i, w_j^{m_j+1})$ yields an incentive compatible payment scheme that minimizes the expected payment to every agent for any reported type of the agent. Recall that individual rationality is satisfied along with the incentive constraints.

Since arc lengths in T_j^f satisfy decomposition monotonicity, a shortest path from w_j^i to $w_j^{m_j+1}$ is the path that includes all intermediate nodes $w_j^{i+1}, \ldots, w^{m_j}$. Observing that $-dist(w_j^{m_j+1}, w_j^{m_j+1}) =$

0 and $-dist(w_j^i, w_j^{m_j+1}) = \sum_{k=i}^{m_j} w_j^k[ES_j(f, w_j^k) - ES_j(f, w_j^{k+1})]\ \forall w_j^i \in W_j \setminus \{w_j^{m_j+1}\}$ proves the first claim.

Next, we compute the minimum expected total payment for allocation rule f.

$$\begin{aligned}
P^{min}(f) &= \sum_{j\in J}\sum_{i=1}^{m_j} \varphi_j(w_j^i)\pi_j^f(w_j^i) \\
&= \sum_{j\in J}\sum_{i=1}^{m_j} \varphi_j(w_j^i)\sum_{k=i}^{m_j} w_j^k[ES_j(f, w_j^k) - ES_j(f, w_j^{k+1})] \\
&= \sum_{j\in J}\sum_{i=1}^{m_j} \varphi_j(w_j^i)\left(\sum_{k=i}^{m_j} w_j^k ES_j(f, w_j^k) - \sum_{k=i+1}^{m_j} w_j^{k-1} ES_j(f, w_j^k)\right) \\
&= \sum_{j\in J}\sum_{i=1}^{m_j} \varphi_j(w_j^i)\left(w_j^i ES_j(f, w_j^i) + \sum_{k=i+1}^{m_j} ES_j(f, w_j^k)(w_j^k - w_j^{k-1})\right) \\
&= \sum_{j\in J} ES_j(f, w_j^1) w_j^1 \varphi_j(w_j^1) \\
&\quad + \sum_{j\in J}\sum_{i=2}^{m_j} ES_j(f, w_j^i)\left(\varphi_j(w_j^i) w_j^i + (w_j^i - w_j^{i-1})\sum_{k=1}^{i-1}\varphi_j(w_j^k)\right) \\
&= \sum_{j\in J} ES_j(f, w_j^1) w_j^1 \varphi_j(w_j^1) \\
&\quad + \sum_{j\in J}\sum_{i=2}^{m_j} ES_j(f, w_j^i)\left(\Phi_j(w_j^i) w_j^i - \Phi_j(w_j^{i-1}) w_j^{i-1}\right)
\end{aligned}$$

Let us define modified weights $\overline{w}_j$ by setting $\overline{w}_j^1 = w_j^1$ and for $i = 2, \ldots, m_j$

$$\begin{aligned}
\overline{w}_j^i &= \frac{w_j^i \Phi_j(w_j^i) - w_j^{i-1}\Phi_j(w_j^{i-1})}{\varphi_j(w_j^i)} \\
&= \frac{w_j^i \varphi_j(w_j^i) + w_j^i \Phi_j(w_j^{i-1}) - w_j^{i-1}\Phi_j(w_j^{i-1})}{\varphi_j(w_j^i)} \\
&= w_j^i + (w_j^i - w_j^{i-1})\frac{\Phi_j(w_j^{i-1})}{\varphi_j(w_j^i)}.
\end{aligned}$$

This yields

$$P^{min}(f) = \sum_{j\in J}\sum_{i=1}^{m_j} \varphi_j(w_j^i)\overline{w}_j^i ES_j(f, w_j^i).$$

□

Proof of Lemma 2. For any $i_1, i_2, i_3 \in \{1, \ldots, m_j + 1\}$, $i_1 < i_2 < i_3$, and any $k \in \{1, \ldots, q_j\}$ the constraint

$$E\pi_j(w_j^{i_1}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^k) \geq E\pi_j(w_j^{i_3}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_3}, p_j^k)$$

is implied by

$$E\pi_j(w_j^{i_1}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^k) \geq E\pi_j(w_j^{i_2}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^k)$$

and

$$E\pi_j(w_j^{i_2}, p_j^k) - w_j^{i_2} ES_j(f, w_j^{i_2}, p_j^k) \geq E\pi_j(w_j^{i_3}, p_j^k) - w_j^{i_2} ES_j(f, w_j^{i_3}, p_j^k).$$

In fact, adding up the latter two constraints yields

$$\begin{aligned} & E\pi_j(w_j^{i_1}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^k) \\ \geq\ & E\pi_j(w_j^{i_3}, p_j^k) + w_j^{i_2}(ES_j(f, w_j^{i_2}, p_j^k) - ES_j(f, w_j^{i_3}, p_j^k)) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^k) \\ \geq\ & E\pi_j(w_j^{i_3}, p_j^k) + w_j^{i_1}(ES_j(f, w_j^{i_2}, p_j^k) - ES_j(f, w_j^{i_3}, p_j^k)) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^k) \\ =\ & E\pi_j(w_j^{i_3}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_3}, p_j^k), \end{aligned}$$

where the second inequality follows from monotonicity and $w_j^{i_1} < w_j^{i_2}$. Note that everything remains true if the dummy type is involved, i.e., if $(w_j^{i_3}, p_j^k) = (w_j^{m_j+1}, p_j^k) = t_j^d$. These arguments imply that all constraints of the type

$$E\pi_j(w_j^{i_1}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^k) \geq E\pi_j(w_j^{i_2}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^k) \tag{8}$$

are implied by the subset of constraints where $i_2 = i_1 + 1$.

A similar effect can be shown for the "reverse" incentive constraints, i.e., the above constraints for $i_3 < i_2 < i_1$, where $i_1, i_2, i_3 \in \{1, \ldots, m_j\}$. Again, out of all constraints of the type

$$E\pi_j(w_j^{i_1}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^k) \geq E\pi_j(w_j^{i_2}, p_j^k) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^k), \tag{9}$$

only those with $i_2 = i_1 - 1$ are necessary.

Similarly, out of all constraints of the type

$$E\pi_j(w_j^i, p_j^{k_1}) - w_j^i ES_j(f, w_j^i, p_j^{k_1}) \geq E\pi_j(w_j^i, p_j^{k_2}) - w_j^i ES_j(f, w_j^i, p_j^{k_2}), \tag{10}$$

for $i \in \{1, \ldots, m_j\}$, $k_1, k_2 \in \{1, \ldots, q_j\}$, $k_1 < k_2$ only those with $k_2 = k_1 + 1$ are necessary.

For any types $(w_j^{i_1}, p_j^{k^1})$,$(w_j^{i_2}, p_j^{k^2})$ with $i_1 < i_2$ and $k_1 < k_2$ the corresponding "diagonal" constraint

$$E\pi_j(w_j^{i_1}, p_j^{k_1}) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^{k_1}) \geq E\pi_j(w_j^{i_2}, p_j^{k_2}) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^{k_2})$$

follows by adding up the corresponding constraints of type (10) and (8)

$$E\pi_j(w_j^{i_1}, p_j^{k_1}) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^{k_1}) \geq E\pi_j(w_j^{i_1}, p_j^{k_2}) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^{k_2})$$

$$E\pi_j(w_j^{i_1}, p_j^{k_2}) - w_j^{i_1} ES_j(f, w_j^{i_1}, p_j^{k_2}) \geq E\pi_j(w_j^{i_2}, p_j^{k_2}) - w_j^{i_1} ES_j(f, w_j^{i_2}, p_j^{k_2}).$$

For any $(w_j^{i_1}, p_j^{k^1})$,$(w_j^{i_2}, p_j^{k^2})$ with $i_2 < i_1$ and $k_1 < k_2$, the corresponding "diagonal" constraint follows by adding up the appropriate constraints of type (10) and (9). □