Interdependent Scheduling Games

Andres Abeliuk
Data61/NICTA
andres.abeliuk@data61.csiro.au

Haris Aziz
Data61/NICTA and UNSW
haris.aziz@data61.csiro.au

Gerardo Berbeglia
University of Melbourne
g.berbeglia@mbs.edu

Serge Gaspers
UNSW and Data61/NICTA
sergeg@cse.unsw.edu.au

Petr Kalina
Czech Technical University
petr.kalina@fel.cvut.cz

Nicholas Mattei
Data61/NICTA and UNSW
nicholas.mattei@data61.csiro.au

Dominik Peters
University of Oxford
dominik.peters@cs.ox.ac.uk

Paul Stursberg
Technische Universität München
paul.stursberg@ma.tum.de

Pascal Van Hentenryck
University of Michigan
pvanhent@umich.edu

Toby Walsh
UNSW and Data61/NICTA
tw@cse.unsw.edu.au

Abstract

We propose a model of interdependent scheduling games in which each player controls a set of services that they schedule independently. A player is free to schedule his own services at any time; however, each of these services only begins to accrue reward for the player when all predecessor services, which may or may not be controlled by the same player, have been activated. This model, where players have *interdependent* services, is motivated by the problems faced in planning and coordinating large-scale infrastructures, e.g., restoring electricity and gas to residents after a natural disaster or providing medical care in a crisis when different agencies are responsible for the delivery of staff, equipment, and medicine. We undertake a game-theoretic analysis of this setting and in particular consider the issues of welfare maximization, computing best responses, Nash dynamics, and existence and computation of Nash equilibria.

1 Introduction

Restoring critical infrastructure in the aftermath of natural disasters or extreme weather events where water, power, and gas services may all be interrupted is one of the most important ways of limiting the impact of the disaster on society. Our motivation for this work is drawn from situations where companies and governments need to restore *interdependent infrastructure* after major disruptions due to disasters and other forces. For instance, the electric company may be able to restore power lines to individual homes, but no electricity will flow until the gas company can supply gas to the main generator. Once the power is flowing, the electric company receives its reward (income) from those customers receiving power. In order to pump water, power needs to have been restored and the water lines need to be repaired. Each of these objectives are typically broken down into smaller tasks that restore availability to a subset of customers. In these settings, multiple agents (also called players) are responsible for different services and may have conflicting interests: the power company may deploy its services in an order that maximizes reach to its subscriber base first, as opposed to undertaking repairs that allow another company to restart the water pumps. This paper formalizes a novel abstract model of this setting and studies the problem of finding a *joint deployment schedule* of services through a game theoretic lens, as players in this setting are independent decision makers. We consider classic questions such as welfare maximization, best responses, and the existence and computation of Nash equilibria.

From the community's perspective, the overall goal is to reduce the size and length of the blackout. Indeed, governments in the US plan for infrastructure restoration at a higher level than the individual company, e.g., the state government or regional emergency management planning. However, when disasters become too large or individual companies refuse to cooperate with regional disaster management plans then companies might be unable (or unwilling) to obey global welfare considerations in restoring their infrastructure. Cavdaroglu *et al.* [2013] and Coffrin *et al.* [2012] provide models that integrate the restoration planning and scheduling decisions to show that there is significant value in this integration as opposed to tackling both problems in a decentralized manner. Our model of *interdependent scheduling games* (ISGs) is a step towards understanding the im-

pact of decentralized decision making in settings with interdependencies. Other examples of ISGs include coordinating multiple providers for humanitarian logistics over multiple regions, where roads need to be repaired before supplies can be delivered and tents must be erected before supplies can be distributed, or the coordination of interdependent supply chains which may involve ports, terminals, railway, and truck operators [Van Hentenryck *et al.*, 2010; Simon *et al.*, 2012].

In our formalization, we consider a set of players, each of which has a set of services under their control that need to be deployed. The individual players' services may have dependencies among each other and, crucially, may also be dependent on the status of other players' services. In contrast to most traditional scheduling settings, where a task cannot be scheduled unless all of its dependencies have been fulfilled, services in our setting can be deployed at any time, even before its dependencies have been deployed. However, a player only starts *accruing reward* for a service v once all of its dependencies have been deployed as well. At this point, we say that v has been *activated* and the player continues to gather reward for every time step in which the service is active. A typical reward in our setting would be collecting fees from utility subscribers who have had their service restored.

Contributions. We present a scheduling model with dependencies among services that is suitable for scenarios in power restoration after natural disasters. We show that when there is only a single player, a welfare-maximizing schedule can be found in polynomial time. For more players, welfare maximization becomes NP-complete even with just two services per player. Regarding game-theoretic solution concepts, we prove that in general, pure Nash equilibria are not guaranteed to exist, and that it is NP-hard to decide their existence. On the positive side, we consider a restricted setting where all services have uniform (equal) reward and prove that a pure Nash equilibrium always exists and can be computed in polynomial time. Similarly, best responses can be computed efficiently but they need not converge to a Nash equilibrium, even if rewards are uniform. For the uniform rewards case, we also give bounds for the price of anarchy and the price of stability. Further, we provide an ILP formulation of the problem and demonstrate that, for generated data, we can find welfare maximizing schedules quickly.

2 Related Work

The problem of finding a schedule of tasks that maximizes the reward is an important question in scheduling, a classic area of computer science with many practical and important problems. Most classical scheduling problems focus on allocating scarce resources to multiple tasks in order to maximize an objective function or minimize total time [Brucker and Brucker, 2007; Lee *et al.*, 1997]. In contrast to most of the scheduling literature, the dependencies (or precedence constraints) between the services in our model do not prevent the player from scheduling a service before its prerequisites are fulfilled. Instead, they keep the player from receiving *reward* from the service until the prerequisites are fulfilled.

Encouraging distributed agents, each of which may be re-

sponsible for only a small piece of a larger task, to work together to solve complex problems has a rich history in artificial intelligence and multi-agent systems research. Scheduling distributed tasks in domains where agents are imbued with their own reward functions but are ultimately cooperative as they can jointly benefit from finding coordinated schedules, has been studied in a probabilistic setting by Zhang and Shah [2014]. Additionally, *task oriented domains* [Rosenschein and Zlotkin, 1994], which typically involve multiple agents working together cooperatively, are a popular framework for investigating mechanisms and properties of multi-agent domains where agents either need to work together or negotiate over work to be accomplished. Zlotkin and Rosenschein [1993] formalize the notion of strategic behavior when agents negotiate in task oriented domains. They provide a characterization of the type of lies (e.g. hiding jobs) and reward functions that admit incentive compatible mechanisms for a number of classic domains, though none of these classic domains involve scheduling with dependencies.

We focus our analysis on game-theoretic issues such as best response dynamics and Nash equilibria that are keenly applicable in settings such as ours where agents, trying to maximize independent utility, may or may not have explicit incentives to cooperate towards maximizing global welfare. Scheduling domains in which players compete for common processing resources were introduced by Agnetis *et al.* [2000; 2004] and Baker and Smith [2003]. The most traditional approach in multi-agent scheduling is to consider a single centralized authority optimizing the whole domain. There have been a number of recent works focused on decentralized scheduling mechanisms. Agnetis *et al.* [2007] consider auction and bargaining models, which are useful when several players have to negotiate for processing resources on the basis of their scheduling performance. Scheduling auctions typically divide the schedule horizon into time slots, and these time slots are auctioned among the players. The bargaining approach considers two players that have to negotiate over possible schedules. Abeliuk *et al.* [2015] consider a two-player bargaining mechanism for any setting where the reward of one player does not depend on the actions taken by the other. Their results hence apply to special instances of ISGs with two players. For additional literature on mechanism design for non-cooperative scheduling games see, e.g, Heydenreich *et al.* [2007], Christodoulou *et al.* [2004], and Angel *et al.* [2006].

Another related line of research is multi-agent project-scheduling. Here, each project is composed of a set of activities, with precedence relations between the activities, and each activity belongs to an agent. Each activity is associated with a minimum and a maximum processing time and agents have to choose a duration for all their activities. Compressing the duration of an activity generates a cost to the agent, and an agents' payoff is a fixed proportion of the total project payment, which depends on the project completion time. A mechanism design approach for multi-agent project-scheduling by Confessore *et al.* [2007] proposes a decentralized mechanism using combinatorial auctions. Recently, Briand and Billaut [2011] took a first step in analyzing game theoretical properties such as the existence and computation

of Nash equilibria as well as studying the price of anarchy in this setting. However, their setting significantly differs from that considered in this paper in that activities of the same agent can be processed in parallel and that all agents receive some fraction of the reward of a common production process. In contrast, we focus on agents involved in independent projects with separate objective functions, only related by precedence constraints between each other.

3 Our Model

A *directed graph* G is a pair (V, E) of a finite set of *vertices* V and a set of directed edges $E \subseteq V \times V$ where $(u, v) \in E$ means that there is a directed edge from u to v in G. We will always assume that G is *acyclic*, i.e., there is no set of edges $\{(v_1, v_2), (v_2, v_3), \ldots, (v_n, v_1)\} \subseteq E$. We say that G is *transitive* if $(u, v), (v, w) \in E$ implies that $(u, w) \in E$. The *transitive closure* of a graph $G = (V, E)$ is a graph $G' = (V, E')$ such that $(u, v) \in E'$ if a directed path connects u and v in G. The *in-neighborhood* of a vertex v is the set of vertices with edges to v and is denoted by $N_G^-(v) = \{u \in V : (u, v) \in E\}$.

An *interdependent scheduling game* (ISG) with k players is given by a tuple $((T_1, \ldots, T_k), G, r)$. Each player i needs to schedule a set of *services* T_i, where the T_i are pairwise disjoint. We denote the set of all services by $T = \bigcup_{i=1}^k T_i$. We assume without loss of generality that $|T_1| = \cdots = |T_k| = q$. Within T there are dependencies: a service will not activate until it and all its prerequisites are deployed. We formalize this *dependency relation* as a transitive acyclic directed graph $G = (T, E)$. If $(u, v) \in E$, then service v will generate a reward only after service u has been deployed. To be precise, at each time step t, each player *deploys* exactly one service. In particular, we assume that every service takes exactly one unit of time to deploy. A service which takes longer to deploy can be represented as a series of services depending on each other where only the final service generates a reward. For each service $v \in T$, there is a *reward* $r(v) \geq 0$, representing payment received or subscribers served in each time period that the service is active. We will sometimes consider the more restrictive case of *uniform rewards* where for all $v \in T, r(v) = 1$.

A solution for an ISG is a schedule of all services in T. As rewards are non-negative, players do not have an incentive to leave a gap between the deployment of two services. We can hence represent a *schedule* by a tuple $\pi = (\pi_1, \ldots, \pi_k)$, where each $\pi_i : T_i \to \{1, \ldots, |T_i|\}$ is a permutation of the services T_i of player i. This permutation uniquely determines the schedule for player i and the position of a service in the permutation denotes the time when it is deployed.

A service v is *active* during a time step if itself and all services in $N_G^-[v]$ are deployed at or before that time step. We denote by $a(v)$ the time when v becomes active, i.e. $a(v) = \max\{\pi(w) : w \in \{v\} \cup N_G^-[v]\}$. At each time step, all active services v generate the reward $r(v)$. Thus, for a schedule $\pi = (\pi_1, \ldots, \pi_k)$, the *utility* of player i is $R_i(\pi) = \sum_{t=1}^q \sum_{v \in T_i, t \geq a(v)} r(v)$. The *utilitarian social welfare* (or just *welfare*) of a schedule π is $\sum_{i=1}^k R_i(\pi)$.

We graphically represent an ISG in Example 1. Player i's services T_i form the nodes shown in the ith row. The services in a row, from left to right, represent player i's schedule, while the label of a service indicates its reward. For ease of presentation, we omit arrows that are implied by transitivity of the dependency relation; the full dependency graph is the transitive closure of the depicted graph. This representation is not completely unambiguous as a service v is identified only by $r(v)$ and the edges in $N_G(v)$. However, while indistinguishable (subsets of) tasks may exist, these can be interchanged within any particular outcome without effect.

Example 1. *Consider the following example.*
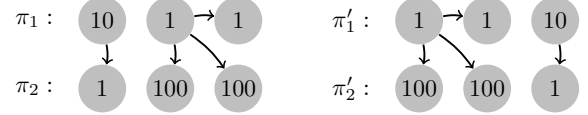
Both of the services with reward 100 *belong to Player 2 but depend upon a service belonging to Player 1. For schedule π, $R_1(\pi) = 3 \cdot 10 + 2 \cdot 1 + 1 = 33$ as the service with reward 10 is active for three time steps and the other services are active for two and one time step, respectively. Similarly, $R_2(\pi) = 3 \cdot 1 + 2 \cdot 100 + 100 = 303$. For π', $R_1(\pi') = 3 \cdot 1 + 2 \cdot 1 + 10 = 15$ while $R_2(\pi') = 3 \cdot 100 + 2 \cdot 100 + 1 = 501$. Hence, Player 1 can sacrifice some individual reward to increase welfare.*

4 Best Responses

If all other players' actions are fixed, the resulting problem for an individual player is that of finding a best response. Let $R_i(\pi_{-i}, \pi_i')$ be the reward for player i for the schedule $(\pi_1, \ldots, \pi_{i-1}, \pi_i', \pi_{i+1}, \ldots, \pi_k)$.

Problem: ISG BEST RESPONSE.
Instance: An ISG $((T_1, \ldots, T_k), G, r)$, a schedule π_{-i} for all players $\{1, \ldots, k\} \setminus \{i\}$, and an integer W.
Question: Is there a π_i' such that $R_i(\pi_{-i}, \pi_i') \geq W$?

Assuming that players are individually rational they will favor schedules that maximize their own reward, i.e., their own subscriber base or service network. Hence an individual player will always favor a schedule such that every service v that he controls is deployed only after all other services under the player's control that v depends on have been deployed. Formally, the following Lemma holds:

Lemma 1. *Let $((T_1, \ldots, T_k), G, r)$ be an ISG with general rewards and π_{-i} a schedule for all players except player i. Let $G_i = (T_i, E_i)$ denote the subgraph of G induced by the vertices in T_i. Then, there exists a best response π_i for player i such that*

$$\pi_i(u) < \pi_i(v) \text{ for all } (u, v) \in E_i. \tag{1}$$

Proof. Let $\sigma(\pi_i) := |\{v \in T_i : \exists (u, v) \in E_i \text{ with } \pi_i(u) > \pi_i(v)\}|$ denote the number of services in T_i that depend on another service in T_i which is scheduled later. Let π_i' denote a best response of player i such that $\sigma(\pi_i')$ is minimal among all best responses. We suppose for contradiction that the statement is false, therefore $\sigma(\pi_i') \geq 1$. Choose $(u, v) \in E_i$ with $\pi_i'(u) > \pi_i'(v)$ in such a way that there is no u' with $(u', v) \in E_i$ and $\pi_i'(u') > \pi_i'(u)$. Consider the following

modified schedule π_i^* for player i:

$$\pi_i^*(w) := \begin{cases} \pi_i'(w) - 1 & \pi_i'(w) \in [\pi_i'(v)+1, \pi_i'(u)] \\ \pi_i'(u) & w = v \\ \pi_i'(w) & \text{else.} \end{cases}$$

The following two properties hold: **(i)**: The schedule π_i^* is also a best response. The only service that is scheduled to a later time in π_i^* (and hence could cause itself or services depending on it to generate a smaller reward) is v. However, v did not activate before time step $\pi_i'(u)$ under π' and as $\pi_i^*(v) = \pi_i'(u)$ the reward generated by v does not change. The same holds for all services that depend on v. **(ii)**: $\sigma(\pi_i^*) < \sigma(\pi_i')$. First, note that v does not contribute towards σ anymore as $\pi_i^*(v) > \pi_i^*(u)$ (the same holds for all other services that v depends upon by the maximality of u). Now, consider any service w that did not contribute to $\sigma(\pi_i')$. As the ordering among all services except v remains the same, such a w can only contribute to $\sigma(\pi_i^*)$ if it depends on v and $\pi_i'(v) < \pi_i'(w) < \pi_i^*(v) = \pi_i^*(u)$. But then, it must also depend on u by transitivity and hence it contributed to $\sigma(\pi_i')$ already. From (i) and (ii), we obtain a contradiction to minimality of $\sigma(\pi_i')$, concluding the proof. $\square$

Note that performing pairwise swaps in a player's scheduled services is not sufficient in the context of the above proof as this may introduce new forward edges. The above lemma holds for general rewards. If rewards are uniform, we can use Lemma 1 to derive a polynomial-time algorithm for an individual player's best response to all other players' schedules.

Theorem 1. *For an ISG with uniform rewards, there exists a polynomial-time algorithm to compute a best response.*

Proof. Consider the subgraph G_i of G induced by the set T_i of services belonging to player i. For every service u, denote by $\eta(u)$ the lower bound on its activation time imposed by π_{-i}. Formally, $\eta(u) := \max\{\pi(v) : v \in T \setminus T_i, (v, u) \in E\}$. Note that $(u, w) \in E_i$ implies $\eta(w) \geq \eta(u)$ by transitivity of E_i.

We give a greedy algorithm that solves the problem optimally. Starting from the first time step, the algorithm successively schedules a service which minimizes η among all services with no incoming edges in G_i. Such a service always exists, as G (and hence all subgraphs) is acyclic. The service with all its (outgoing) edges is then removed from G_i.

To prove that the algorithm yields an optimal solution, let π_i be the outcome of the algorithm. Suppose for contradiction that π_i is not optimal. Let π_i^* be an optimal schedule satisfying condition (1) (which exists by Lemma 1) maximizing the first time slot for which any such schedule differs from π_i. Formally, there exists $k \in N$ such that $(\pi_i^*)^{-1}(i) = (\pi_i)^{-1}(i)$ for all $i < k$ and there is no optimal schedule π_i' satisfying condition (1) with $(\pi_i')^{-1}(i) = (\pi_i)^{-1}(i)$ for all $i < k + 1$.

Let $a := (\pi_i^*)^{-1}(k)$ and $b := (\pi_i)^{-1}(k)$. Consider the subgraph of G_i from which the first $k - 1$ entries of π_i (and hence of π_i^*) have been removed. First, observe that a cannot have any incoming edges as π_i^* satisfies condition (1). Hence, it holds that $\eta(b) \leq \eta(a)$, otherwise the algorithm would have selected a rather than b. We distinguish three cases:

1. $\eta(b) \leq \pi_i^*(a)$. In this case, we set

$$\pi_i^{**}(w) := \begin{cases} \pi_i^*(w) + 1 & \pi_i^*(w) \in [\pi_i^*(a), \pi_i^*(b) - 1] \\ \pi_i^*(a) & w = b \\ \pi_i^*(w) & \text{else.} \end{cases}$$

The reward generated by b increases by $\pi_i^*(b) - \pi_i^*(a)$, at the same time the reward of at most $\pi_i^*(b) - \pi_i^*(a)$ services decreases by 1. Hence, π^{**} is still optimal and satisfies condition (1). Furthermore, $(\pi_i)^{-1}(k) = (\pi_i^{**})^{-1}(k)$, contradicting π^*'s maximality.

2. $\pi_i^*(a) < \eta(b) < \pi_i^*(b)$. We construct a new schedule $\tilde{\pi}_i^*$ with $\tilde{\pi}_i^*(b) = \eta(b)$ as above. Then, proceed as in 3.

3. $\pi_i^*(b) \leq \eta(b)$. Construct a schedule π_i^{**} as follows: Set $\pi_i^{**}(b) := \pi_i^{**}(a)$. Let a' be the earliest successor of a. If $\pi_i^*(a') > \pi_i^*(b)$, set $\pi_i^{**}(a) := \pi_i^*(b)$ and $\pi_i^{**}(w) := \pi_i^*(w)$ for all other services. Otherwise, set $\pi_i^{**}(a) := \pi_i^*(a')$ and let a'' be the earliest successor of a'. Proceed with a'' (and possibly its earliest successor) as above until an earliest successor a^* satisfies $\pi_i^*(a^*) > \pi_i^*(b)$. The resulting schedule π^{**} is still optimal and satisfies condition (1). Furthermore, $(\pi_i)^{-1}(k) = (\pi_i^{**})^{-1}(k)$, contradicting π^*'s maximality.

In all three cases, we reach a contradiction which proves that our assumption was wrong and π_i is indeed optimal. $\square$

In contrast, we can obtain the following statement about general rewards by reduction from single-player welfare maximization using Theorem 5.

Corollary 1. *For an ISG with general rewards, computing a best response is NP-complete.*

5 Welfare Maximization

A central planner would want to find a schedule that maximizes the *welfare*, i.e., the most profitable services in T activated for the longest amount of time.

Problem: ISG WELFARE.
Instance: An ISG $((T_1, \ldots, T_k), G, r)$ and an integer w.
Question: Is there a π such that $\sum_{i=1}^k R_i(\pi) \geq w$?

Intuitively, it might seem desirable to design schedules where no service has to wait for its activation after it has been deployed. We call such schedule *conflict-free*. For uniform rewards, if a conflict-free schedule exists then every welfare-maximizing schedule obviously has to be conflict-free. A similar statement holds for single-player games by the construction of π_i^* in Lemma 1 (proof omitted).

Theorem 2. *For one player and general rewards, every welfare-maximizing schedule is a conflict-free schedule.*

However, this property does not hold in the case of more than one player and general rewards. This can be seen by considering Example 1 and making all other services dependent on π_1's service with reward 10. Then, any conflict-free schedule will yield welfare 319 while the welfare-maximizing schedule is 417, yielding the following theorem:

Theorem 3. *For multiple players and general rewards, even if a conflict-free schedule exists, the welfare-maximizing schedule(s) might not be conflict-free.*

Turning to computational complexity, we observe that for one player welfare maximization is equivalent to finding a best response, hence with Theorem 1 we get the following.

Corollary 2. *For uniform rewards,* ISG WELFARE *can be solved in polynomial time for a single player.*

However, when we either increase the number of players (Thm. 4) or relax the restriction of uniform rewards (Thm. 5), the problem is NP-hard for surprisingly restricted cases.

Theorem 4. ISG WELFARE *is NP-complete, even when the rewards are uniform and each player has two services.*

Proof. The problem is in NP since we can efficiently compute the welfare of a given schedule. For NP-hardness, we reduce from MIN 2SAT [Kohli *et al.*, 1994] which asks: Given a 2CNF formula F where each clause contains exactly two literals, and an integer k, is there an assignment to the variables of F such that at most k clauses are satisfied?

For each variable x in F, create a player P_x with services $T_x = \{x, \neg x\}$. For each clause c in F, create a player P_c with services $T_c = \{c_1, c_2\}$. For each clause $c = (\ell_1 \vee \ell_2)$, the precedence graph contains (c_1, c_2), (ℓ_1, c_1), and (ℓ_2, c_1). Rewards are uniform, and we set $w = 3n + 3m - k$, where n and m are the number of variables and clauses of F.

It remains to prove that F has an assignment satisfying at most k clauses if and only if the ISG has a schedule generating a reward of at least w. For the forward direction, suppose F has an assignment $\alpha : \text{var}(F) \to \{0, 1\}$ satisfying at most k clauses. Consider the schedule where, for each variable x, the player P_x schedules first the literal of x that is set to false by α, i.e., x is scheduled before $\neg x$ iff $\alpha(x) = 0$. Additionally, for each clause c, the service c_1 is scheduled before c_2. This schedule generates a reward of 3 for each variable: a reward of 1 at the first time step and a reward of 2 at the second time step. For a satisfied clause c, the schedule generates a reward of 2: at the first time step no reward is generated since the literal satisfying the clause is scheduled at the second time step and there is an arc from that literal to c_1, and a reward of 2 is generated at the second time step. For an unsatisfied clause c, the schedule generates a reward of 3: since neither literal satisfies the clause, both literals are scheduled at the first time step. Thus, the utility generated for this schedule is at least $3n + 3m - k$.

For the reverse direction, let π be a schedule generating a reward of at least w. Consider the assignment $\alpha : \text{var}(F) \to \{0, 1\}$ with $\alpha(x) = 0$ iff player P_x schedules x at the first time step. Note that at the second time step, each player generates a reward of 2. Also, each player corresponding to a variable generates an additional reward of 1 at the first time step since his services have in-degree 0. So, at least $3n + 3m - k - (3n + 2m) = m - k$ additional clause players generate a reward of 1 at the first time step. But, for each such clause c, c_1 is scheduled before c_2 and both literals occurring in c are scheduled at the first time step, which means that the assignment α sets these literals to false. Therefore, α does not satisfy c. We conclude that α satisfies at most k clauses. $\square$

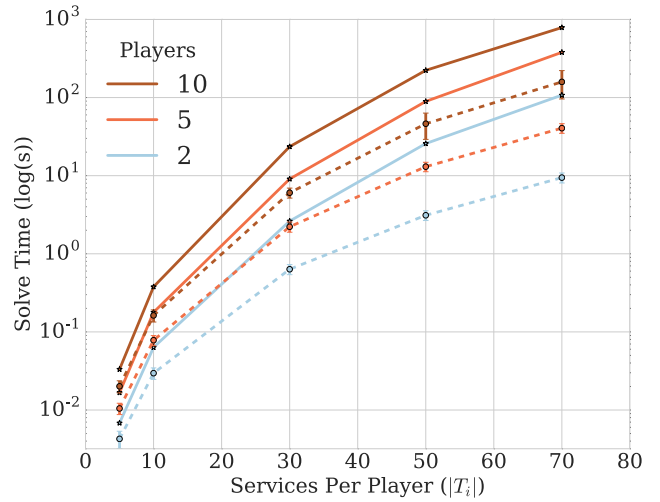**Theorem 5.** *For general rewards,* ISG WELFARE *is NP-complete even for a single player.*

Figure 1: Mean runtime of the ILP over 1,000 random ISG instances varying the number of players, services, and reward type; error bars represent one standard deviation (σ). The solid lines are instances with general rewards, the dashed lines are instances with uniform rewards. The plot is semi-logarithmic, so a straight line represents an exponential increase in time. For the general rewards case, error bars are not included for clarity; for $|T_i| \in \{10, 30, 50\}$ the numbers are small, $\sigma = 30$ seconds in the worst case, however, for 70 services and 10 players this balloons to 200 seconds.

The proof, omitted for space, is a reduction from the NP-hard problem SINGLE MACHINE WEIGHTED COMPLETION TIME [Lenstra and Rinnooy Kan, 1978]. It relies on Theorem 2 and an adjustment of rewards.

5.1 Integer Programming Formulation

While the general problem of finding a welfare maximizing schedule for an ISG instance is computationally hard, it may still be solvable for instances of moderate size. The ISG WELFARE problem admits a natural integer linear programming (ILP) formulation. For each service $v \in T$ and time step $t \in [q]$, we introduce two binary decision variables $a_{v,t}$ and $s_{v,t}$. Let $s_{v,t} = 1$ if and only if service v is scheduled at time t, and $a_{v,t} = 1$ if and only if service v is active at time t.

$$
\begin{aligned}
\max \quad & \sum_{v \in T} \sum_{t=1}^{q} a_{v,t} \cdot r(v) \\
\text{s.t.} \quad & \sum_{t=1}^{q} s_{v,t} = 1 && \forall v \in T \\
& \sum_{v \in T_i} s_{v,t} = 1 && \forall i \in [k], \forall t \in [q] \\
& a_{v,t} \leq \sum_{t'=1}^{t} s_{v,t'} && \forall v \in T, \forall t \in [q] \\
& a_{v,t} \leq a_{w,t} && \forall (w, v) \in E, \forall t \in [q]
\end{aligned}
$$

We implemented the ILP and solved 1000 randomly generated instances where (a) general rewards are drawn from [50,100] and (b) rewards are uniform. The dependency graphs are generated by first randomly permuting the list of all services; then for each service i, drawing a random number of child services $c \in \{0, 1, 2\}$ and adding edge $(i, i + c)$ with probability 0.5. Increasing the number/likelihood of dependencies by increasing the potential number of children or increasing the connection probability significantly increases
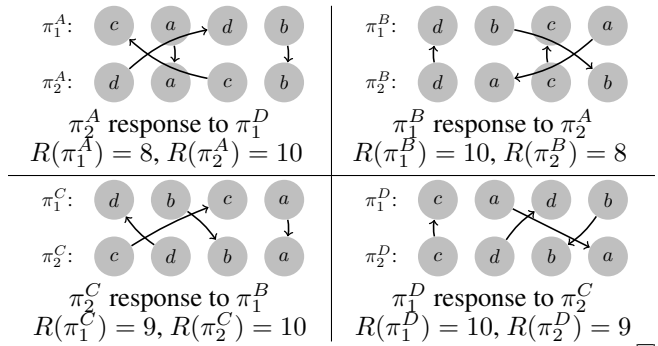
runtime. Figure 1 shows the results for different parameters using Gurobi 6.5 on a computer equipped with an 2.0 GHz Intel Xeon E5405 CPU with 4 GB of RAM. The results suggest that, despite worst case hardness, the running times remain feasible, at worst $\approx 600s$, for practically relevant problem sizes: up to 10 players with 70 services each.

6 Nash Dynamics and Equilibria

We now turn to the situation where players may respond to each other's schedule changes. This is an important question for game theoretic analyses as it allows us to see which states leave no incentives for self-interested players to deviate; and what can happen when players are continually responding to the moves of one another. An important first question is whether a sequence of best responses terminates.

Theorem 6. *For ISGs with uniform rewards, best responses can cycle.*

Proof. Consider the following example depicting a sequence of best responses. Starting with the lower right schedule π_D we move to the upper left schedule π_A where Player 2 has changed his schedule in a best response to π_D. We then read left to right, top to bottom, to end up back at π_D.

$$\pi_1^A \text{ response to } \pi_1^D$$
$$R(\pi_1^A) = 8, R(\pi_2^A) = 10$$

$$\pi_1^B \text{ response to } \pi_2^A$$
$$R(\pi_1^B) = 10, R(\pi_2^B) = 8$$

$$\pi_2^C \text{ response to } \pi_1^B$$
$$R(\pi_1^C) = 9, R(\pi_2^C) = 10$$

$$\pi_1^D \text{ response to } \pi_2^C$$
$$R(\pi_1^D) = 10, R(\pi_2^D) = 9$$
□

6.1 ISGs with Uniform Rewards

A schedule π is in *pure Nash equilibrium* (PNE) if no player can obtain strictly more utility by unilaterally changing his own schedule; formally, $R_i(\pi_{-i}, \pi) \geq R_i(\pi_{-i}, \pi_i')$ for all players i and all schedules π_i' of player i. For instance, note that the above example, despite having a sequence of best responses that cycle, does admit the PNE depicted below:
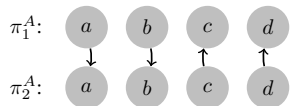
Questions of existence and computation of PNEs are fundamental to a game theoretic analysis as a PNE schedule is stable with respect to selfish players who may try to unilaterally increase their utility by playing a different schedule.

Theorem 7. *Any ISG with uniform rewards admits a pure Nash equilibrium which can be computed in polynomial time.*

Proof (some details omitted). We iteratively construct a schedule such that every player's schedule is a best response.

Let $N_i^-(v) := (N_G^-(v) \cup \{v\}) \cap T_i$ denote the closed in-neighborhood of service v under player i's control, $T_i^{(t)}$ the set of services of player i already scheduled before iteration t and $\alpha_i^{(t)} := |T_i^{(t)}|$. In every iteration, we will choose a service and schedule it together with all remaining services that it depends on. This means that for a service $v \in T_i^{(t)}$, $a(v)$ is well-defined during iteration t. We can therefore define

$$\bar{\eta}_i^{(t)}(v) := \begin{cases} \max_{w \in N_i^-(v)} a(w), & N_i^-(v) \setminus T_i^{(t)} = \emptyset \\ \alpha_i^{(t)} + |N_i^-(v) \setminus T_i^{(t)}|, & \text{else} \end{cases}$$

Now, $\bar{\eta}^{(t)}(v) := \max_{i \in N} \bar{\eta}_i^{(t)}(v)$ represents a tight lower bound for $a(v)$ in any schedule which is a "completion" of the partial schedule from iteration t (achieved if v and all prerequisites are scheduled immediately).

Similar to Theorem 1, it can be verified that player i's schedule π_i is a best response if for every iteration t and player i, the condition (1) from Lemma 1 holds for all services $v, w \in T_i^{(t)}$ and if $v \in T_i^{(t)} \setminus T_i^{(t-1)}$, then $\bar{\eta}^{(t-1)}(v)$ is minimal among all services from the set $T_i \setminus T_i^{(t-1)}$. Furthermore, for every iteration t and services $v \in T_i \setminus T_i^{(t)}$ and $w \in T_i^{(t)}$, we show that $(v, w) \notin E_i$ and $\bar{\eta}^{(t)}(w) \leq \bar{\eta}^{(t)}(v)$.

In iteration t, we proceed in the following way: Choose a service v^* that minimizes $\bar{\eta}^{(t)}$ over all services not yet scheduled and that has no incoming edges from services belonging to the same player. Such a service must exist, as if $(w, v^*) \in E$ for some service w, then $\bar{\eta}^{(t)}(w) \leq \bar{\eta}^{(t)}(v^*)$. Let i be the player such that $v^* \in T_i$.

Assuming that the above conditions are satisfied for iteration t, we can now show that they also hold for iteration $t+1$. The described procedure hence constructs a pure Nash equilibrium for the given game in time polynomial in $|T|$. □

As every player strives to activate his services as early as possible, which is also in the interest of other players whose services depend on them, one may think that the schedule that maximizes welfare is always a PNE. However, this is not the case. The ratio of the maximum welfare to the maximum welfare in a PNE is called the *price of stability*. The following theorem shows that this ratio may be strictly greater than 1.

Theorem 8. *Even for uniform rewards, a welfare-maximizing schedule is not necessarily a pure Nash equilibrium.*

*Proof.* Consider the following example.
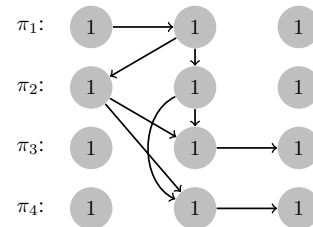
The schedule shown is not a Nash equilibrium: if Player 2 shifts the last service to the first slot, he increases his reward by 1. In fact, any schedule that is a PNE must have Player 2's last service (in π_2 as shown) in the first slot as both other

services, depending (by transitivity) on the two services of Player 1 cannot activate before the second time step. Hence, one of the remaining two services of Player 2 (the two with dependencies), that other services depend on, will only be deployed in the last time step. This implies that in any schedule that is a Nash equilibrium, the two services of both Players 3 and 4 that depend on Player 2's services will not activate before the last time step, either. Hence, Players 3 and 4 cannot achieve a reward higher than $3 \cdot 1 + 2 \cdot 0 + 1 \cdot (1+1) = 5$ each. Even if both other players receive the maximal reward of 6, then the welfare in any Nash equilibrium schedule cannot exceed 22. On the other hand, the schedule shown achieves a total welfare of 23. Hence, no welfare maximizing schedule can be a Nash equilibrium. □

Since there may be more than one PNE profile in ISGs with uniform rewards, it is natural to ask how bad the price of anarchy, the ratio of the maximum welfare schedule to the maximum welfare in a PNE, can become.

Theorem 9. *The price of anarchy of ISGs with uniform rewards is* $\geq {}^{k(q+1)}/_{(q+2k-1)}$ *with k players, q services each.*

Proof. Consider the following example.
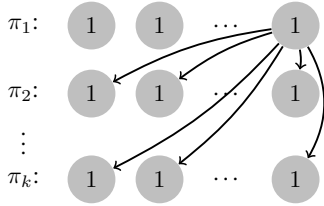
The worst PNE is obtained (as shown) by scheduling player 1's service, on which all others depend, at the end; as opposed to the PNE achieved when this service is at the beginning, which is welfare-maximizing. The ratio between the welfares is $\frac{k \cdot q(q+1)/2}{q(q+1)/2+(k-1)q} = \frac{k(q+1)}{q+2k-1}$. □

If we fix the number of players k, the ratio is bounded by $\lim_{q \to \infty} k(q+1)/(q+2k-1) = k$. Similarly, when fixing the number of services q, then $\lim_{k \to \infty} k(q+1)/(q+2k-1) = (q+1)/2$. This motivates the following theorem.

Theorem 10. *The price of anarchy of ISG with uniform rewards is at most $(q+1)/2$.*

Proof. The worst PNE profile cannot be worse than the schedule in which all services activate at the last time step q, which obtains welfare $k \cdot q$. The maximum-welfare schedule cannot be better than a schedule in a game without any precedence constraints, which obtains welfare $k \cdot q(q+1)/2$. Together, we have: $PoA \leq \frac{kq(q+1)/2}{kq} = \frac{q+1}{2}$. □

6.2 ISGs with General Rewards

Our results for the general setting are not as positive as our results for the uniform rewards setting. We show that for the general rewards setting, an ISG with two players does not always admit a pure Nash equilibrium.

Theorem 11. *An ISG with two players and general rewards does not always admit a pure Nash equilibrium.*

Proof. Consider the the following instance.
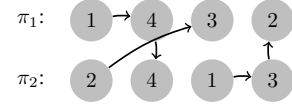
Assume this game admits a PNE, any best response of Player 1 must satisfy that service 4, being the highest reward service, is scheduled immediately after service 1. Therefore, any possible best response of Player 1 has to adopt one of the following schedule configurations: (*i*) $\pi_1 \in (1, 4, *, *)$, (*ii*) $\pi_1 \in (*, 1, 4, *)$ or (*iii*) $\pi_1 \in (*, *, 1, 4)$.

In a similar way, service 4 of Player 2, for any best response of Player 2, must be scheduled as soon as possible. These observations narrow the number of possible PNE configurations to three cases: **Case** (*i*) Player's 2 best response, given any schedule of the form $\pi_1 \in (1, 4, *, *)$ is $\pi_2 = (2, 4, 1, 3)$. However, such a schedule triggers a best response for Player 1 of $\pi_1 = (3, 1, 4, 2)$, which take us to case (ii). **Case** (*ii*) Player's 2 best response, given any schedule of the form $\pi_1 \in (*, 1, 4, *)$ is $\pi_2 = (1, 3, 4, 2)$. However, such a schedule triggers a best response for Player 1 of $\pi_1 = (1, 4, 2, 3)$, which is an instance of case (i). This leads to a cycle of best responses. **Case** (*iii*) Player's 2 best response, given any schedule of the form $\pi_1 \in (*, *, 1, 4)$ is $\pi_2 \in \{(2, 1, 3, 4), (1, 3, 2, 4)\}$. However, such schedules trigger a best response for Player 1 of $\pi_1 = (3, 1, 4, 2)$ if $\pi_2 = (2, 1, 3, 4)$, or $\pi_1 = (1, 4, 3, 2)$ in the other case. Both schedules being an instance of case (ii) or (i), respectively. Therefore, for any schedule π_1, there is no schedule π_2, such that (π_1, π_2) is a PNE. □

We conjecture that the example in Theorem 11 is minimal with respect to the number of services and dependencies. We can embed this example into a 3SAT reduction to show that checking the existence of a PNE is NP-hard.

Theorem 12. *Deciding whether an ISG with general rewards admits a PNE schedule is NP-hard, even when each player has at most 4 services.*

7 Conclusions

We have introduced a class of interdependent scheduling games that are motivated by large-scale infrastructure restoration and humanitarian logistics; answering many important questions that arise when the players are independent decision makers, including questions of welfare maximization and existence of PNEs. An interesting technical open problem is to determine the complexity of welfare maximization when the number of players is bounded. More broadly, there are a number of promising directions for future work including the extension of the model to include cyclic interdependencies [Coffrin *et al.*, 2012] or considering other types of manipulation such as adding services or misreporting utilities [Zlotkin and Rosenschein, 1993]. Also note that approximation algorithms for traditional scheduling settings (with hard dependencies and non-accruing rewards) cannot be directly applied to our model. Hence, another possible avenue of research would be a study of fixed parameter tractability and approximation algorithms for ISGs.

Acknowledgments

References

[Abeliuk *et al.*, 2015] Andres Abeliuk, Gerardo Berbeglia, and Pascal Van Hentenryck. Bargaining mechanisms for one-way games. *Games*, 6(3):347–367, 2015.

[Agnetis *et al.*, 2000] Alessandro Agnetis, Pitu B Mirchandani, Dario Pacciarelli, and Andrea Pacifici. Nondominated schedules for a job-shop with two competing users. *Computational & Mathematical Organization Theory*, 6(2):191–217, 2000.

[Agnetis *et al.*, 2004] Allesandro Agnetis, Pitu B Mirchandani, Dario Pacciarelli, and Andrea Pacifici. Scheduling problems with two competing agents. *Operations Research*, 52(2):229–242, 2004.

[Agnetis *et al.*, 2007] Alessandro Agnetis, Dario Pacciarelli, and Andrea Pacifici. Combinatorial models for multi-agent scheduling problems. *Multiprocessor Scheduling*, page 21, 2007.

[Angel *et al.*, 2006] Eric Angel, Evripidis Bampis, and Fanny Pascual. Truthful algorithms for scheduling selfish tasks on parallel machines. *Theoretical Computer Science*, 369(1):157–168, 2006.

[Baker and Smith, 2003] Kenneth R Baker and J Cole Smith. A multiple-criterion model for machine scheduling. *Journal of Scheduling*, 6(1):7–16, 2003.

[Briand and Billaut, 2011] Cyril Briand and J Billaut. Cooperative project scheduling with controllable processing times: a game theory framework. In *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference on*, pages 1–7. IEEE, 2011.

[Brucker and Brucker, 2007] Peter Brucker and P Brucker. *Scheduling Algorithms*, volume 3. Springer, 2007.

[Cavdaroglu *et al.*, 2013] Burak Cavdaroglu, Erik Hammel, John E Mitchell, Thomas C Sharkey, and William A Wallace. Integrating restoration and scheduling decisions for disrupted interdependent infrastructure systems. *Annals of Operations Research*, 203(1):279–294, 2013.

[Christodoulou *et al.*, 2004] George Christodoulou, Elias Koutsoupias, and Akash Nanavati. Coordination mechanisms. In *Automata, Languages and Programming*, pages 345–357. Springer, 2004.

[Coffrin *et al.*, 2012] Carleton Coffrin, Pascal Van Hentenryck, and Russell Bent. Last-mile restoration for multiple interdependent infrastructures. In *Proc. of the 26th AAAI Conference on Artificial Intelligence*, pages 455–463, 2012.

[Confessore *et al.*, 2007] Giuseppe Confessore, Stefano Giordani, and Silvia Rismondo. A market-based multi-agent system model for decentralized multi-project scheduling. *Annals of Operations Research*, 150(1):115–135, 2007.

[Heydenreich *et al.*, 2007] Birgit Heydenreich, Rudolf Müller, and Marc Uetz. Games and mechanism design in machine scheduling—an introduction. *Production and Operations Management*, 16(4):437–454, 2007.

[Karp, 1972] Richard M Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, page 85, 1972.

[Kohli *et al.*, 1994] Rajeev Kohli, Ramesh Krishnamurti, and Prakash Mirchandani. The minimum satisfiability problem. *SIAM J. Discrete Math.*, 7(2):275–283, 1994.

[Lee *et al.*, 1997] Chung-Yee Lee, Lei Lei, and Michael Pinedo. Current trends in deterministic scheduling. *Annals of Operations Research*, 70:1–41, 1997.

[Lenstra and Rinnooy Kan, 1978] Jan Karel Lenstra and AHG Rinnooy Kan. Complexity of scheduling under precedence constraints. *Operations Research*, 26(1):22–35, 1978.

[Rosenschein and Zlotkin, 1994] Jeffrey S Rosenschein and Gilad Zlotkin. *Rules of encounter: Designing conventions for automated negotiation among computers*. MIT press, 1994.

[Simon *et al.*, 2012] Ben Simon, Carleton Coffrin, and Pascal Van Hentenryck. Randomized adaptive vehicle decomposition for large-scale power restoration. In *Proc. of the 9th Conference on the Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR)*, pages 379–374, 2012.

[Van Hentenryck *et al.*, 2010] Pascal Van Hentenryck, Russell Bent, and Carleton Coffrin. Strategic planning for disaster recovery with stochastic last mile distribution. In *Proc. of the 7th Conference on the Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR)*, pages 318–333, 2010.

[Zhang and Shah, 2014] Chongjie Zhang and Julie A. Shah. Fairness in multi-agent sequential decision-making. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 2636–2644, 2014.

[Zlotkin and Rosenschein, 1993] Gilad Zlotkin and Jeffrey S. Rosenschein. A domain theory for task oriented negotiation. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 416–422, 1993.

A Full Version of Theorem 2

Theorem. *For one player and general rewards, every welfare-maximizing schedule is a conflict-free schedule.*

Proof. This follows by an observation about the proof of Lemma 1: In the one-player case, service u activates immediately under schedule π'_i by its maximality among dependencies for which v has to wait. Hence, it also activates immediately under schedule π^*_i, which is one time step earlier than under schedule π'_i. Schedule π^*_i hence generates strictly more reward than schedule π'_i. □

B Full Version of Theorem 3

Theorem. *Even if a conflict-free schedule exists, the welfare-maximizing schedule might not be conflict-free.*

Proof. Consider the following example.

$$R(\pi^A) = 309 \qquad R(\pi^B) = 407$$
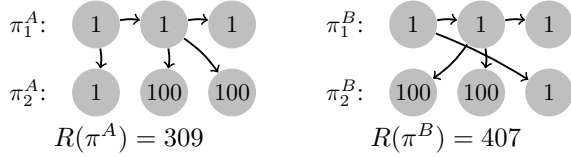
The schedule on the left is conflict-free while the one on the right has a conflict. Despite the conflict, the right schedule has higher welfare; the two services with reward 100 become active simultaneously in step two, providing more utility to Player 2 and more welfare. □

C Full Version of Theorem 5

Theorem. *For general rewards,* ISG WELFARE *is NP-complete even for a single player.*

Proof. We give a reduction from the NP-hard problem SINGLE MACHINE WEIGHTED COMPLETION TIME [Lenstra and Rinnooy Kan, 1978]: given a set of jobs $J_i \in J$ with each having weight w_i, processing time $p_i = 1$, and precedence constraints where $i \prec j$ means J_j cannot be scheduled before J_i, and integer k, is there an ordering of the jobs such that $\sum_{i \in J} w_i C_i \leq k$ where C_i is the completion time of J_i?

For each job $J_i \in J$, create service t_i with reward $r_i = w_i$ and consider the same precedence graph as the one given for jobs. We set $w = (|J| + 1) \sum_{i \in J} w_i - k$.

By Theorem 2, without loss of generality, we can assume that any schedule for ISGs with one player are conflict-free schedules. It remains to prove that there is an ordering π of jobs with a weighted completion time of at most k if and only if the ISG has a conflict-free schedule π' with $R(\pi') \geq w$.

Let $\pi = \pi'$, then C_i is the completion time of both, job J_i and service t_i given ordering π. Given that π is a conflict-free schedule, the contribution of t_i to the objective function is $(|T| + 1 - C_i) r_i$. Thus, $R(\pi) = \sum_{i \in T} (|T| + 1 - C_i) r_i = (|T| + 1) \sum_{i \in T} r_i - \sum_{i \in T} r_i C_i$. But, $\sum_{i \in T} r_i C_i = \sum_{i \in J} w_i C_i$, which corresponds to the weighted completion time of ordering π. Therefore, $R(\pi) \geq w \Leftrightarrow \sum_{i \in J} w_i C_i \leq k$, which concludes the proof. □

D Full Version of Theorem 7

Theorem. *Any ISG with uniform rewards admits a pure Nash equilibrium which can be computed in polynomial time.*

Proof. We iteratively construct a schedule in a way which guarantees that every player's schedule is a best response.

Let $N_i^-(v) := (N_G^-(v) \cup \{v\}) \cap T_i$ denote those services controlled by player i that v depends on. Denote by $T_i^{(t)}$ the set of services of player i already scheduled before iteration t. Let $\alpha_i^{(t)} := |T_i^{(t)}|$ denote the number of such services. In every iteration, we will choose a service and schedule it together with all other (remaining) services that it depends on. This means that for a service $v \in T_i^{(t)}$, $a(v)$ is well-defined during iteration t. We can therefore define

$$\bar{\eta}_i^{(t)}(v) := \begin{cases} \max_{w \in N_i^-(v)} a(w), & N_i^-(v) \setminus T_i^{(t)} = \emptyset \\ \alpha_i^{(t)} + |N_i^-(v) \setminus T_i^{(t)}|, & \text{else} \end{cases}$$

In particular, observe that if v is controlled by player i and $v \notin T_i^{(t)}$, the second case always applies (as $v \in N_i^-(v)$).

Furthermore, we define $\bar{\eta}^{(t)}(v) := \max_{i \in N} \bar{\eta}_i^{(t)}(v)$ which represents a tight lower bound for the activation time of v in any schedule which is a "completion" of the partial schedule from iteration t (achieved if all prerequisites are scheduled immediately as the next services). Note that $\bar{\eta}^{(t)}(v)$ can hence only increase from one iteration to the next and that it reaches the value $a(v)$ as soon as service v and all its predecessors are scheduled and is constant after that.

By Theorem 1, player i's schedule π_i is a best response if it satisfies condition (1) from Lemma 1 and for all $v \in T_i$, $\eta(v)$, as defined in Theorem 1, is minimal among all services from the set $\{w \in T_i | \pi_i(w) \geq \pi_i(v)\}$.

We will show instead that for every iteration t and player i, the condition (1) from Lemma 1 holds for all services $v, w \in T_i^{(t)}$ and if $v \in T_i^{(t)} \setminus T_i^{(t-1)}$, then $\bar{\eta}^{(t-1)}(v)$ is minimal among all services from the set $T_i \setminus T_i^{(t-1)}$. To see that this condition is also sufficient for π_i being a best response, observe the following: While it may happen for a player i^* and $v, v' \in T_{i^*}$ that $\bar{\eta}^{(t)}(v)$ is minimal among all services from the set $\{w \in T_i^{(t)} | \pi_i(w) \geq \pi_i(v)\}$ but $\eta(v) > \eta(v')$, this can only occur if for both services the maximum in the definition of $\bar{\eta}$ is assumed for $i = i^*$ as well as $N_i^-(v) \setminus T_i^{(t)}0\{v\}$ and $N_i^-(v') \setminus T_i^{(t)}0\{v'\}$. This however means that both v and w are equivalent at this point in that they both activate immediately after being deployed.

Furthermore, for every iteration t and services $v \in T_i \setminus T_i^{(t)}$ and $w \in T_i^{(t)}$, we show that $(v, w) \notin E_i$ and $\bar{\eta}^{(t)}(w) \leq \bar{\eta}^{(t)}(v)$. This yields that every player's schedule is a best response to the other players' schedules and hence the schedule is in a pure Nash equilibrium.

Assume that the above conditions are satisfied for iteration t and proceed in the following way: Choose a service v^* that minimizes $\bar{\eta}^{(t)}$ over all services not yet scheduled and that has no incoming edges from services belonging to the same player. Such a service must exist, as if $(w, v^*) \in E$ for some service w, then $\bar{\eta}^{(t)}(w) \leq \bar{\eta}^{(t)}(v^*)$. Let i be the player such

that $v^* \in T_i$. If v^* has no incoming edges from any of the services not yet scheduled, then scheduling v^* as the next service of player i satisfies the best-response criterion, no matter the ordering of the unscheduled services.

Hence, suppose that v^* depends on some other services not yet scheduled. Denote this set of services by S. By induction, scheduling all services in S (respecting the ordering required by edges in E_i if necessary) satisfies condition (1) for all players i and $v, w \in T_i^{(t+1)}$. Furthermore, note that for every $w \in S$, $\bar{\eta}^{(t)}(w) = \bar{\eta}^{(t)}(v^*)$ by minimality of v^* and the dependency of v^* on w, thus $\bar{\eta}^{(t)}(w) = \eta^{(t)}(v^*)$. Hence for every player i, if $v \in T_i^{(t+1)} \setminus T_i^{(t)} (\subseteq S \cup \{v^*\})$, then $\bar{\eta}^{(t)}(v)$ is minimal among all services from the set $T_i \setminus T_i^{(t)}$.

Finally, the criteria for every $v \in T_i \setminus T_i^{(t+1)}$ and $w \in T_i^{(t+1)}$ are satisfied as well: For every $v \notin S$ and $w \in S$, $(v, w) \notin E_i$ as otherwise $w \in S$. Furthermore, $\bar{\eta}^{(t+1)}(w) = \bar{\eta}^{(t)}(w) = \bar{\eta}^{(t)}(v^*) \leq \bar{\eta}^{(t)}(v) \leq \bar{\eta}^{(t+1)}(v)$ where the first equality holds because w and all its dependencies are scheduled in iteration t, the second equality was shown above and the inequalities follows by minimality of v^* and monotonicity of $\bar{\eta}^{(t)}(v)$ in t.

The described procedure hence constructs a pure Nash equilibrium for the given game in time polynomial in $|T|$. $\square$

E Full Version of Theorem 12

Theorem. *Deciding whether there exists a PNE schedule is NP-hard, even when each player has at most 4 services.*

Proof. We give a reduction from the NP-hard problem 3SAT [Karp, 1972]: given a CNF formula F where each clause contains exactly 3 literals, is there an assignment to the variables of F such that all clauses are satisfied?

For each variable x in F, create a player P_x with services $T_x = \{x, \neg x\}$. Both services have the same reward $r(x) = r(\neg x) = 1$. For each clause c in F, create a player P_c with services $T_c = \{c_1, c_2, c_3, d_c\}$ and set rewards to be $r(d_c) = 3$ and $r(c_1) = r(c_2) = r(c_3) = 4$. For each clause c, we create a gadget G_c corresponding to a copy of the ISG from Theorem 11 which admits no PNE and consists of 2 players with 4 services each. For each clause $c = (\ell_1 \vee \ell_2 \vee \ell_3)$ in F, the precedence graph contains arcs $(\ell_1, c_1), (\ell_2, c_2), (\ell_3, c_3)$ and arcs from service d_c to the 8 services of gadget G_c.

It remains to prove that F has an assignment satisfying all clauses if and only if the ISG admits a pure Nash equilibrium. For the forward direction, suppose F has an assignment $\alpha : \text{var}(F) \to \{0, 1\}$ satisfying all clauses. Consider the schedule where, for each variable x, the player P_x schedules first the literal of x that is set to true by α, i.e., x is scheduled before $\neg x$ iff $\alpha(x) = 1$. For each clause c, the player P_c schedules its true literals, then its false literals given α, and then service d_c. Services in gadget G_c can be scheduled arbitrarily. This schedule is in a pure Nash equilibrium: for each variable player this is the best that player can do. For each clause player this is the best that player can do given that all dependencies from the services of the variable players are met. Finally, the players in gadget G_c are indifferent between all schedules because their services all become active in the last time step, given that service d_c was scheduled at the end.

For the reverse direction, suppose conversely that the game has a pure Nash equilibrium. Consider the assignment $\alpha : \text{var}(F) \to \{0, 1\}$ with $\alpha(x) = 1$ iff player P_x schedules x at the first time step. We show that the assignment α satisfies F. Suppose some clause c is not satisfied. Then, none of its literal services will be activated before the second time step, only service d_c is activated in the first time step. Hence, all best responses for the clause player P_c put service d_c into the first time slot, giving the player a reward of 36. This means that services in gadget G_c have no restrictions imposed. But G_c for itself does not admit a Nash equilibrium, and hence the entire game does not either, a contradiction. So all clauses are satisfied. $\square$