Mathematical Programming
https://doi.org/10.1007/s10107-020-01511-3

FULL LENGTH PAPER

Series B

Breaking symmetries to rescue sum of squares in the case of makespan scheduling

Victor Verdugo[1,2] · José Verschae[4] · Andreas Wiese[3]

Received: 31 May 2019 / Accepted: 22 April 2020

Abstract

The sum of squares (SoS) hierarchy gives an automatized technique to create a family of increasingly tight convex relaxations for binary programs. There are several problems for which a constant number of rounds of this hierarchy give integrality gaps matching the best known approximation algorithms. For many other problems, however, ad-hoc techniques give better approximation ratios than SoS in the worst case, as shown by corresponding lower bound instances. Notably, in many cases these instances are invariant under the action of a large permutation group. This yields the question how symmetries in a formulation degrade the performance of the relaxation obtained by the SoS hierarchy. In this paper, we study this for the case of the minimum makespan problem on identical machines. Our first result is to show that $\Omega(n)$ rounds of SoS applied over the *configuration linear program* yields an integrality gap of at least 1.0009, where n is the number of jobs. This improves on the recent work by Kurpisz et al. (Math Program 172(1–2):231–248, 2018) that shows an analogous result for the weaker LS_+ and SA hierarchies. Our result is based on tools from representation theory of symmetric groups. Then, we consider the weaker *assignment linear program* and add a well chosen set of symmetry breaking inequalities that removes a subset of the machine permutation symmetries. We show that applying $2^{\tilde{O}(1/\varepsilon^2)}$ rounds of the SA hierarchy to this stronger linear program reduces the integrality gap to $1+\varepsilon$, which yields a linear programming based polynomial time approximation scheme. Our results suggest that for this classical problem, symmetries were the main barrier preventing the SoS/SA hierarchies to give relaxations of polynomial complexity with an integrality gap of $1+\varepsilon$. We leave as an open question whether this phenomenon occurs for other symmetric problems.

Keywords Makespan scheduling · Polynomial optimization · Approximation algorithms · Symmetry breaking

This work has been partially funded by Fondecyt Projects (Nr. 1170223 and 1181527) and Conicyt (PCI PII 20150140). An extended abstract of this paper appeared in IPCO 2019 [53].

Extended author information available on the last page of the article

Published online: 14 May 2020

Mathematics Subject Classification 68W25 · 90C22 · 90B35

1 Introduction

The lift-and-project methods are powerful techniques for deriving convex relaxations of integer programs. The lift-and-project *hierarchies*, such as Sherali-Adams (SA), Lovász–Schrijver (LS), and sum of squares (SoS), are systematic methods for obtaining a family of increasingly tight relaxations, parameterized by the number of *rounds* of the hierarchy. For all of them, applying r rounds on a formulation with n variables yields a convex relaxation with $n^{O(r)}$ variables in the lifted space. Taking $r = n$ rounds gives an exact description of the integer hull [32], at the cost of having an exponential number of variables. On the other hand, taking $r = O(1)$ rounds yields a description with only a polynomial number of variables. Arguably, it is not well understood for which problems these hierarchies, with a constant number of rounds, yield relaxations that match the respective best possible approximation algorithm. Indeed, there are some positive results, but there are also many other strong negative results for algorithmically easy problems. These lower bounds show a natural limitation on the power of hierarchies as one-fits-all techniques. Quite remarkably, the instances used for obtaining lower bounds often have a very symmetric structure [17,31,34,42,44], which suggests a connection between the tightness of the relaxation given by these hierarchies and symmetries. The primary purpose of this article is to study this connection for a specific relevant problem, namely, the minimum *makespan scheduling on identical machines*.

Minimum makespan scheduling This is one of the first problems considered under the lens of approximation algorithms [16], and it has been studied extensively. The input of the problem consists of a set J of n jobs, each having an integral processing time $p_j > 0$, and a set $[m] = \{1, \ldots, m\}$ of m identical machines. Given an assignment $\sigma : J \to [m]$, the load of a machine i is the total processing time of jobs assigned to i, that is, $\sum_{j \in \sigma^{-1}(i)} p_j$. The objective is to find an assignment of jobs to machines that minimizes the *makespan*, that is, the maximum load. The problem is strongly NP-hard and admits several *polynomial-time approximation schemes* (PTAS) based on different techniques, such as dynamic programming, integer programming on fixed dimension, and integer programming under a constant number of constraints [1,2,11,19,20,22,23].

Integrality gaps The minimum makespan problem has two natural linear relaxations, which have been extensively studied in the literature. The *assignment linear program* uses binary variables x_{ij} which indicates whether a job j is assigned to a machine i, for each $i \in [m]$, $j \in J$. The stronger *configuration linear program* uses a variable y_{iC} for each machine i and multiset of processing times C, which indicates whether C is the multiset of processing times of jobs assigned to i. Kurpisz et al. [31] showed that the configuration linear program has an integrality gap of at least $1024/1023 \approx 1.0009$ even after $\Omega(n)$ of rounds of the LS_+ or SA hierarchies. On the other hand, Kurpisz et al. [31] leave open whether the SoS hierarchy applied to the configuration linear program has an integrality gap of $1+\varepsilon$ after applying a number of rounds that depends only on the constant $\varepsilon > 0$, i.e., $O_\varepsilon(1)$ rounds. Our first main contribution is a negative answer to this question.

Theorem 1 *Consider the minimum makespan problem on identical machines. For each $n \in \mathbb{N}$, there exists an instance with n jobs such that, after applying $\Omega(n)$ rounds of the SoS hierarchy over the configuration linear program, the obtained semidefinite relaxation has an integrality gap of at least* 1.0009.

Naturally, since the configuration linear program is stronger than the assignment linear program, our result holds if we apply $\Omega(n)$ rounds of SoS over the assignment linear program. The proof of the lower bound relies on tools from representation theory of symmetric groups over polynomials rings,and it is inspired on the recent work by Raymond et al. [45] for symmetric sums of squares in hypercubes. It is based on constructing high-degree *pseudoexpectations* on the one hand, and by obtaining symmetry-reduced decompositions of the polynomial ideal defined by the configuration linear program, on the other hand. The machinery from representation theory allows to restrict attention to invariant polynomials, and we combine this with a strong *pseudoindependence* result for a well chosen polynomial spanning set. Our analysis is also connected to the work of Razborov on flag algebras and graph densities, and we believe it can be of independent interest for analyzing lower bounds in the context of SoS in presence of symmetries [44,46,47].

Symmetries and Hierarchies Given the relation between hierarchies and symmetries above, it is natural to explore whether symmetry handling techniques might help to overcome the limitation given by Theorem 1. A natural source of symmetry of the problem comes from the fact that the machines are identical: Given a schedule, we obtain another schedule with the same makespan by permuting the machines. The same symmetries are encountered in the assignment and configuration linear programs, namely, if $\sigma : [m] \to [m]$ is a permutation and (x_{ij}) is a feasible solution to the assignment linear program then $(x_{\sigma(i)j})$ is also feasible. The same holds for solutions (y_{iC}) and $(y_{\sigma(i)C})$ for the configuration-LP. In other words, these linear programs are *invariant* under the action of the symmetric group on the set of machines. The question we study is the following: *Is it possible to obtain a polynomial size linear or semidefinite program with an integrality gap of at most $1+\varepsilon$ that is not invariant under the machine symmetries?* We aim to understand if these symmetries deteriorate the quality of the relaxations obtained from the SoS or SA hierarchies. This time, we provide a positive answer.

Theorem 2 *Consider the problem of scheduling identical machines to minimize the makespan. After adding linearly many inequalities to the assignment linear program (for breaking symmetries), $2^{\tilde{O}(1/\varepsilon^2)}$ rounds of the SA hierarchy yield a linear program with an integrality gap of at most $1+\varepsilon$, for any $\varepsilon > 0$.*

Notice that the same result is obtained by applying the SoS hierarchy instead of SA. The proof of Theorem 2 is based on introducing a formulation that *breaks* the symmetries in the assignment program by adding new constraints. The symmetry breaking constraints enforce that any feasible integral solution of the formulation respects a *lexicographic* order over the machine configurations. We show how to exploit this to obtain a polynomial time approximation scheme (PTAS) based on the SA hierarchy. Additionally, we show that by adding a polynomial number of new constraints, we can obtain a faster approximation scheme, such that poly$(1/\varepsilon)$ rounds of SA suffice. The extra constraints correspond to symmetry breaking inequalities for a modified instance with rounded job sizes. In particular, the added constraints are not necessarily valid for the original formulation (which considers the original job sizes). However, we can show that increasing the optimal makespan by a factor $1+\varepsilon$ maintains the feasibility of at least one integral solution. Thus, by breaking more symmetries, we make it easier for the hierarchies to produce good relaxations.

We remark that the framework we use for the minimum makespan problem can be, in principle, studied in other settings where symmetries are present in standard integer programming relaxations. This strategy opens the possibility of analyzing the effect of applying symmetry breaking techniques and hierarchies in order to generate strong linear or semidefinite relaxations.

1.1 Related work

Upper bounds The first application of semidefinite programming in the context of approximation algorithms is due to Goemans and Williamson for the Max-Cut problem [15]. Of particular interest to our work is the SoS based approximation scheme by Karlin et al. to the Max-Knapsack problem [24]. They use a structural *decomposition theorem* satisfied by the SoS hierarchy. For a constant number of machines, Levey and Rothvoss design an approximation scheme with a sub-exponential number of rounds in the weaker SA hierarchy [37], which is improved to a quasi-PTAS by Garg [12]. The SoS method has received a lot of attention for high-dimensional problems. Among them we find matrix and tensor completion [7,43], tensor decomposition [38] and clustering [27,44].

Lower bounds The first lower bound obtained in the context of *positivstellensatz certificates* is by Grigoriev [17], showing the necessity of a linear number of SoS rounds to refute an easy Knapsack instance. Similar results are obtained by Laurent [34] for Max-Cut and by Kurpisz et al. [28] for unconstrained polynomial optimization. The same authors show that for a certain polynomial-time single machine scheduling problem, the SoS hierarchy exhibits an unbounded integrality gap even in a high-degree regime [28,30]. Remarkable are the work of Grigoriev [18] and Schoenebeck [50] exhibiting the difficulty for SoS to certify the insatisfiability of random 3-SAT instances in subexponential time, and recently there have been efforts on unifying frameworks to show lower bounds on random CSPs [5,25,26]. For estimation and detection problems, lower bounds have been shown for the planted clique problem, k-densest subgraph and tensor PCA, among others [6,21].

Invariant sum of squares Gatermann and Parrilo study how to obtain reduced sums of squares certificates of non-negativity when the polynomial is invariant under the action of a group, using tools from representation theory [13]. Raymond et al. [45] develop on the Gatermann and Parrilo method to construct symmetry-reduced sum of squares certificates for polynomials over k-subset hypercubes. Furthermore, the authors make an interesting connection with the Razborov method and flag algebras [46,47]. Blekherman et al. [8] and Laurent [35] provide degree bounds on rational representations for certificates over the hypercube, recovering as a corollary known lower bounds for combinatorial optimization problems like Max-Cut. Kurpisz et al. [29] provide a method for proving SoS lower bounds when the formulations exhibits a high degree of symmetry.

2 Preliminaries: sum of squares (SoS) and Pseudoexpectations

In what follows we denote by $\mathbb{R}[x]$ the ring of polynomials with real coefficients. Binary integer programming belongs to a larger class of problems in *polynomial optimization*, where the constraints are defined by polynomials in the variables indeterminates. More specifically, consider the set $\mathcal{K}$ of feasible solutions to the polinomial optimization program defined by

$$g_i(x) \geq 0 \quad \text{for all } i \in M, \tag{1}$$

$$h_j(x) = 0 \quad \text{for all } j \in J, \tag{2}$$

$$x_e^2 - x_e = 0 \quad \text{for all } e \in E. \tag{3}$$

where $g_i, h_j \in \mathbb{R}[x]$ for all $i \in M$ and for all $j \in J$. In particular, for binary integer programming the equality and inequality constraints are affine functions.

Ideals, quotients and square-free polynomials In what follows we give a minimal introduction to the algebraic elements for polynomial optimization, for a comprehensive treatment see [10]. We denote by $\mathbf{I}_E$ the ideal of polynomials in $\mathbb{R}[x]$ generated by $\{x_e^2 - x_e : e \in E\}$, and let $\mathbb{R}[x]/\mathbf{I}_E$ be the quotient ring of polynomials with respect to the vanishing ideal $\mathbf{I}_E$. That is, $f, g \in \mathbb{R}[x]$ are in the same equivalence class of the quotient ring if $f - g \in \mathbf{I}_E$, that we denote $f \equiv g \mod \mathbf{I}_E$. Alternatively, $f \equiv g \mod \mathbf{I}_E$ if and only if the polynomials evaluate to the same values on the vertices of the hypercube, that is, $f(x) = g(x)$ for all $x \in \{0, 1\}^E$. Observe that the equivalence classes in the quotient ring are in bijection with the *square-free* polynomials in $\mathbb{R}[x]$, that is, polynomials where no variable appears squared. In what follows we identify elements of $\mathbb{R}[x]/\mathbf{I}_E$ in this way, that is, for $p \in \mathbb{R}[x]$ we denote by $\overline{p}$ the unique square-free representation of p, which can be obtained as the result of applying the polynomial division algorithm by the Gröbner basis $\{x_e^2 - x_e : e \in E\}$. Given $S \subseteq E$, we denote by x_S the square-free monomial that is obtained from the product of the variables indexed by the elements in S, that is, $x_S = \prod_{e \in S} x_e$. The degree of a polynomial $f \in \mathbb{R}[x]/\mathbf{I}_E$ is denoted by $\deg(f)$. We say that f is a *sum of squares* polynomial, for short SoS, if there exist polynomials $\{s_\alpha\}_{\alpha \in \mathcal{A}}$ for a finite family $\mathcal{A}$ in the quotient ring such that $f \equiv \sum_{\alpha \in \mathcal{A}} s_\alpha^2 \mod \mathbf{I}_E$.

Certificates and SoS method The question of certifying the emptiness of $\mathcal{K}$ is hard in general but sometimes it is possible to find *simple* certificates. We say that there exists a degree-ℓ SoS certificate of infeasibility for $\mathcal{K}$ if there exist SoS polynomials s_0 and $\{s_i\}_{i \in M}$, and polynomials $\{r_j\}_{j \in J}$ such that

$$-1 \equiv s_0 + \sum_{i \in M} s_i g_i + \sum_{j \in J} r_j h_j \mod \mathbf{I}_E, \tag{4}$$

and the degree of every polynomial in the right hand side is at most ℓ. Observe that if $\mathcal{K}$ is non-empty, then the right hand side is guaranteed to be non negative for at least one assignment of x in $\{0, 1\}^E$, which contradicts the equality above. In the case of binary integer programming, if $\mathcal{K}$ is empty there exists a degree-ℓ SoS certificate, for some $\ell \leq |E|$ [36,41]. The SoS algorithm iteratively checks the existence of a SoS certificate, parameterized in the degree, and each step of the algorithm is called a *round*. Since $|E|$ is an upper bound on the certificate degree, the method is guaranteed to terminate [36,41]. Furthermore, the existence of a degree-ℓ SoS certificate can be decided by solving a semidefinite program. This approach can be seen as the dual of the hierarchy proposed by Lasserre, which has been studied extensively in the optimization and algorithms community [9,32,33,48].

Pseudoexpectations To determine the existence of a SoS certificate one solves a semidefinite program, and the solutions of this program determine the coefficients of elements in the dual space of linear operators. We say that a linear functional $\widetilde{\mathbb{E}} : \mathbb{R}[x]/\mathbf{I}_E \to \mathbb{R}$ is a degree-ℓ SoS *pseudoexpectation* for the polynomial system (1)–(3), if it satisfies the following properties:

(SoS.1) $\widetilde{\mathbb{E}}(1) = 1$,
(SoS.2) $\widetilde{\mathbb{E}}(\overline{f^2}) \geq 0$ for all $f \in \mathbb{R}[x]/\mathbf{I}_E$ with $\deg(\overline{f^2}) \leq \ell$,
(SoS.3) $\widetilde{\mathbb{E}}(\overline{f^2 g_i}) \geq 0$ for all $i \in M$, for all $f \in \mathbb{R}[x]/\mathbf{I}_E$ with $\deg(\overline{f^2 g_i}) \leq \ell$,
(SoS.4) $\widetilde{\mathbb{E}}(\overline{f h_j}) = 0$ for all $j \in J$, for all $f \in \mathbb{R}[x]/\mathbf{I}_E$ with $\deg(\overline{f h_j}) \leq \ell$.

In what follows, every time we evaluate a polynomial in the pseudoexpectation we are doing it over the square-free representation. We omit the bar notation for simplicity. The next lemma shows that there is a duality relation between degree-ℓ SoS pseudoexpectation and SoS certificates of infeasibility of the same degree.

Lemma 1 *Suppose that $\mathcal{K}$ is empty. If there exists a degree-ℓ SoS pseudoexpectation then there is no degree-ℓ SoS certificate of infeasibility.*

The proof of this lemma is a simple check, see also [40]. The minimum value of ℓ for which there exists a SoS certificate of infeasibility tells how hard is determining the emptiness of $\mathcal{K}$ for the SoS method. Lemma 1 provides a way of finding lower bounds on the minimum degree of a certificate, which we use in Sect. 3 for the minimum makespan problem. There are many examples of problems that are extremely easy to certificate for humans, but not for the SoS method. For example, given a positive $k \in \mathbb{Q} \backslash \mathbb{Z}$, consider the program $\sum_{e \in E} x_e = k$ and $x_e^2 - x_e = 0$ for all $e \in E$. This problem is clearly infeasible, but there is no degree-ℓ SoS certificate of infeasibility for $\ell \leq \min\{2\lfloor k \rfloor + 3, 2\lfloor n - k \rfloor + 3, n\}$, as shown originally by Grigoriev and others recently using different approaches [17,42].

The Sherali-Adams Hierarchy There is a weaker hierarchy obtained using linear programming due to Sherali and Adams (SA) [51]. Given disjoint subsets $S, R \subseteq E$, consider the polynomial $\varphi_{S,R} = \prod_{i \in S} x_i \prod_{j \in R}(1 - x_j)$, and for every

$\ell \in \{1, \dots, |E|\}$, let $\mathcal{E}_\ell = \{(S, R) : S, R \subseteq E$ with $|S \cup R| = \ell$ and $S \cap R = \emptyset\}$. We say that a linear functional $\widetilde{\mathbb{E}} : \mathbb{R}[x]/\mathbf{I}_E \to \mathbb{R}$ is a degree-ℓ SA pseudoexpectation for (1)–(3), if it satisfies the following properties:

(SA.1) $\widetilde{\mathbb{E}}(1) = 1$,
(SA.2) $\widetilde{\mathbb{E}}(\overline{\varphi_{S,R}}) \geq 0$ for all $(S, R) \in \mathcal{E}_\ell$,
(SA.3) $\widetilde{\mathbb{E}}(\overline{\varphi_{S,R} g_i}) \geq 0$ for all $i \in M$ and $(S, R) \in \mathcal{E}_\ell$ with $\deg(\overline{\varphi_{S,R} g_i}) \leq \ell$,
(SA.4) $\widetilde{\mathbb{E}}(\varphi_{S,R} h_j) = 0$ for all $j \in J$ and $(S, R) \in \mathcal{E}_\ell$ with $\deg(\varphi_{S,R} h_j) \leq \ell$.

Observe that by construction it holds that every degree-$(\ell + 1)$ SA pseudoexpectation is a degree-ℓ SA pseudoexpectation as well. Furthermore, it follows directly from the linearity of $\widetilde{\mathbb{E}}$ that deciding whether a degree-ℓ SA pseudoexpectation exists, and computing one if it exists, can be done by solving a linear program of size $|E|^{O(\ell)}$ over the variables $y_S = \widetilde{\mathbb{E}}(x_S)$, for every $S \subseteq E$ with $|S| \leq \ell$. This linear program is usually known as the *ℓ-round or ℓ-level* of the SA hierarchy. For a detailed exposition of this hierarchy we refer to [33]. In the following we refer to *low-degree* when the degree (SoS or SA) of a certificate or pseudoexpectation is $O(1)$.

3 Lower bound: symmetries are hard for SoS

In this section we show that the SoS method fails to provide a low-degree certificate of infeasibility for a certain family of scheduling instances. The program we analize in this section is the configuration linear program, that has proven to be powerful for different scheduling and packing problems [14,52]. Given a value $T > 0$, a *configuration* corresponds to a multiset of processing times such that its total sum does not exceed T. The multiplicity $m(p, C)$ indicates the number of times that the processing time p appears in the multiset C. The *load* of a configuration C is just the total processing time, $\sum_{p \in \{p_j : j \in J\}} m(p, C) \cdot p$ and let $\mathcal{C}$ denote the set of all configurations with load at most T. For each combination of a machine $i \in [m]$ and a configuration $C \in \mathcal{C}$, the program has a variable y_{iC} that models whether machine i is scheduled with jobs with processing times according to configuration C. Letting n_p denote the number of jobs in J with processing time p, we can write the following binary linear program, $\mathrm{clp}(T)$,

$$\sum_{C \in \mathcal{C}} y_{iC} = 1 \quad \text{for all } i \in [m], \tag{5}$$

$$\sum_{i \in [m]} \sum_{C \in \mathcal{C}} m(p, C) y_{iC} = n_p \quad \text{for all } p \in \{p_j : j \in J\}, \tag{6}$$

$$y_{iC} \in \{0, 1\} \quad \text{for all } i \in [m], \text{ for all } C \in \mathcal{C}. \tag{7}$$

Hard instances. We briefly describe the construction of a family of hard instances $\{I_k\}_{k \in \mathbb{N}}$ for the configuration linear program introduced in [31]. Let $T = 1023$, and for each odd $k \in \mathbb{N}$ we have $15k$ jobs and $3k$ machines. There are 15 different job-sizes with value $O(1)$, each one with multiplicity k. There exist a set of special configurations $\{C_1, \dots, C_6\}$, called *matching configurations*, such that the program above is feasible if and only if the program restricted to the matching configurations is feasible. The infeasibility of the latter program comes from the fact that there is no 1-factorization of a regular multigraph version of the Petersen graph [31, Lemma 2].

Theorem 3 ([31]) *For each odd $k \in \mathbb{N}$, there exists a degree-$\lfloor k/2 \rfloor$ SA pseudoexpectation for the configuration linear program. In particular, there is no low-degree SA certificate of infeasibility.*

3.1 A symmetry-reduced decomposition of the scheduling ideal

Given $T > 0$, the variables ground set for configuration linear program is $E = [m] \times \mathcal{C}$, and the symmetric group S_m acts over the monomials in $\mathbb{R}[y]$ according to $\sigma y_{iC} = y_{\sigma(i)C}$, for every $\sigma \in S_m$. The action extends linearly to $\mathbb{R}[y]/\mathbf{I}_E$, and the configuration linear program is invariant under this action, that is, for every $y \in \text{clp}(T)$ and every $\sigma \in S_m$ we have $\sigma y \in \text{clp}(T)$. We say that a polynomial $f \in \mathbb{R}[y]/\mathbf{I}_E$ is S_m-invariant if $\sigma f = f$ for every $\sigma \in S_m$. When it is clear from the context we drop the S_m in the notation. If f is invariant we have that $f = (1/|S_m|) \sum_{\sigma \in S_m} \sigma f := \text{sym}(f)$, which is the symmetrization of f. We say that a linear functional $\mathcal{L}$ over the quotient ring is S_m-symmetric if for every polynomial $f \in \mathbb{R}[y]/\mathbf{I}_E$ we have $\mathcal{L}(f) = \mathcal{L}(\text{sym}(f))$. The next lemma shows that when $\widetilde{\mathbb{E}}$ is symmetric it is enough to check symmetric polynomials in condition (SoS.2). Therefore, in this case we restrict our attention to those polynomials that are invariant and SoS.

Lemma 2 *Let $\widetilde{\mathbb{E}}$ be a symmetric linear operator over $\mathbb{R}[y]/\mathbf{I}_E$ such that for every invariant SoS polynomial g of degree at most ℓ we have $\widetilde{\mathbb{E}}(g) \geq 0$. Then, $\widetilde{\mathbb{E}}(f^2) \geq 0$ for every $f \in \mathbb{R}[y]/\mathbf{I}_E$ with $\deg(f^2) \leq \ell$.*

Proof Since the operator $\widetilde{\mathbb{E}}$ is symmetric, for every f in the quotient ring with $\deg(f^2) \leq \ell$ we have $\widetilde{\mathbb{E}}(f^2) = \widetilde{\mathbb{E}}(\text{sym}(f^2))$. The polynomial $\text{sym}(f^2)$ is symmetric, and it is SoS since $\text{sym}(f^2) = (1/|S_m|) \sum_{\sigma \in S_m} \sigma f^2$, which is a sum of squares. Since $\deg(\text{sym}(f^2)) \leq \ell$, we have $\widetilde{\mathbb{E}}(\text{sym}(f^2)) \geq 0$ and we conclude that $\widetilde{\mathbb{E}}(f^2) \geq 0$. □

In the following we focus on understanding polynomials that are invariant and SoS. To analize the action of the symmetric group over $\mathbb{R}[y]$ we introduce some tools from representation theory to characterize the invariant S_m-modules of the polynomial ring [49]. We maintain the exposition minimally enough for our purposes and we follow in part the notation used by Raymond et al. [45]. We say that V is an S_m-module if there exists a homomorphism $\rho : S_m \to \text{GL}(V)$, where $\text{GL}(V)$ is the linear group of V. A subspace W of V is *invariant* if it is closed under the action of S_m, that is, when $w \in W$ and $\sigma \in S_m$ we have that $\sigma w \in W$. We say that an S_m-module W is *irreducible* if the only invariant subspaces are $\{0\}$ and W. We refer to [49] for a deeper treatment of representation theory of symmetric groups.

Isotypic decompositions A *partition* of m is a vector $(\lambda_1, \ldots, \lambda_t)$ such that $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_t > 0$ and $\lambda_1 + \cdots + \lambda_t = m$. We denote by $\lambda \vdash m$ when λ is a partition of m. Any S_m-module has an *isotypic decomposition* $V = \bigoplus_{\lambda \vdash m} V_\lambda$, which decomposes V as a direct sum of S_m-modules, where each of the subspaces in the direct sum is called an *isotypic component*. In the following we introduce a combinatorial abstraction of the partitions and related subgroups that play a relevant role. A *tableau* of shape λ is a bijective filling between $[m]$ and the cells of a grid with t rows, and every row $r \in [t]$ has length λ_r. In this case, the *shape* or *Young diagram* of the tableau is λ. For a tableau τ_λ of shape λ, we denote by $\mathrm{row}_r(\tau_\lambda)$ the subset of $[m]$ that fills row r in the tableau.

Example 1 Let $m = 7$ and consider the partition $\lambda = (4, 2, 1)$. The following tableaux have shape λ,

$$\begin{array}{|c|c|c|c|}\hline 1 & 2 & 7 & 4 \\ \hline 5 & 6 \\ \cline{1-2} 3 \\ \cline{1-1}\end{array} \qquad\qquad \begin{array}{|c|c|c|c|}\hline 1 & 7 & 2 & 5 \\ \hline 3 & 6 \\ \cline{1-2} 4 \\ \cline{1-1}\end{array}$$

In the tableau τ_λ at the left, $\mathrm{row}_1(\tau_\lambda) = \{1, 2, 7, 4\}$. In the tableau τ'_λ at the right, $\mathrm{row}_3(\tau'_\lambda) = \{4\}$.

The row group $\mathcal{R}_{\tau_\lambda}$ is the subgroup of S_m that stabilizes the rows of the tableau τ_λ, that is,

$$\mathcal{R}_{\tau_\lambda} = \Big\{\sigma \in S_m : \sigma \cdot \mathrm{row}_r(\tau_\lambda) = \mathrm{row}_r(\tau_\lambda) \text{ for every } r \in [t]\Big\}. \tag{8}$$

Invariant SoS *polynomials.* We go back now to the case of the configuration linear program.

Definition 1 (*Scheduling Ideal*) We define **sched** to be the ideal of polynomials in $\mathbb{R}[y]$ generated by

$$\Bigg\{\sum_{C\in\mathcal{C}} y_{iC} - 1 : i \in [m]\Bigg\} \cup \Big\{y_{iC}^2 - y_{iC} : i \in [m], C \in \mathcal{C}\Big\}. \tag{9}$$

Recall that the set of polynomials above enforce the machines in the scheduling solutions to be assigned with exactly one configuration. Let $\mathbf{Q}^\ell$ be the quotient ring $\mathbb{R}[y]/\mathbf{sched}$ restricted to polynomials of degree at most ℓ and let $\bigoplus_{\lambda\vdash m} \mathbf{Q}^\ell_\lambda$ be its isotypic decomposition. Given a tableau τ_λ of shape λ, let $\mathbf{W}^\ell_{\tau_\lambda}$ be the subspace of $\mathbf{Q}^\ell_\lambda$ fixed by the action of the row group $\mathcal{R}_{\tau_\lambda}$, that is,

$$\mathbf{W}^\ell_{\tau_\lambda} = \Big\{q \in \mathbf{Q}^\ell_\lambda : \sigma q = q \text{ for all } \sigma \in \mathcal{R}_{\tau_\lambda}\Big\}. \tag{10}$$

In what follows we sometimes refer to these subspaces as *row subspaces*. The following result follows from the work of Gaterman and Parrilo [13] in the context of symmetry reduction for invariant semidefinite programs. In what follows, $\langle A, B\rangle$ is the inner product in the space of square matrices defined by the trace of AB. Given $\ell \in [m]$, we denote by Λ_ℓ the subset of partitions of m that are lexicographically larger than $(m - \ell, 1, \dots, 1)$.

Theorem 4 *Suppose that $g \in \mathbb{R}[y]/\boldsymbol{sched}$ is a degree-ℓ SoS and S_m-invariant polynomial. For each partition $\lambda \in \Lambda_\ell$, let τ_λ be a tableau of shape λ and let $\mathcal{P}^\lambda = \{p_1^{\tau_\lambda}, \dots, p_{n_\lambda}^{\tau_\lambda}\}$ be a set of polynomials such that span$(\mathcal{P}^\lambda) \supseteq \mathbf{W}^\ell_{\tau_\lambda}$. Then,* *for each partition $\lambda \in \Lambda_\ell$ there exists a positive semidefinite matrix M_λ such that $g = \sum_{\lambda \in \Lambda_\ell} \langle M_\lambda, Z^{\tau_\lambda} \rangle$, where $Z^{\tau_\lambda}_{ij} = sym(p_i^{\tau_\lambda} p_j^{\tau_\lambda})$.*

The theorem above is based on the recent work of Raymond et al. [45, p. 324, Theorem 3]. In our case the symmetric group is acting differently from Raymond et al., but the proof follows the same lines, and it can be found in "Appendix A". Together with Lemma 2, it is enough to study pseudoexpectations for each of the partitions in Λ_ℓ separately. We remark that for each partition in $\lambda \in \Lambda_\ell$ we can take any tableau τ_λ with that shape, and then consider a spanning set for its corresponding subspace $\mathbf{W}^\ell_{\tau_\lambda}$. In the following, for a matrix A with entries in $\mathbb{R}[y]$, we denote by $\widetilde{\mathbb{E}}(A)$ the matrix obtained by evaluating $\widetilde{\mathbb{E}}$ on each entry of A.

Lemma 3 *Let $\widetilde{\mathbb{E}}$ be a symmetric linear operator over $\mathbb{R}[y]/\mathbf{I}_E$. For each $\lambda \in \Lambda_\ell$, let τ_λ be a tableau of shape λ and let $\mathcal{P}^\lambda = \{p_1^{\tau_\lambda}, \dots, p_{n_\lambda}^{\tau_\lambda}\}$ be a set of polynomials such that $span(\mathcal{P}^\lambda) \supseteq \mathbf{W}^\ell_{\tau_\lambda}$. For each $\lambda \in \Lambda_\ell$, let Z^{τ_λ} such that $Z^{\tau_\lambda}_{ij} = sym(p_i^{\tau_\lambda} p_j^{\tau_\lambda})$ and suppose that $\widetilde{\mathbb{E}}(Z^{\tau_\lambda})$ is positive semidefinite. Then, $\widetilde{\mathbb{E}}(f^2) \geq 0$ for every $f \in \mathbb{R}[y]/\mathbf{I}_E$ with $\deg(f^2) \leq \ell$.*

Proof Let g be an invariant SoS polynomial of degree at most ℓ. By Theorem 4, for each $\lambda \in \Lambda_\ell$ there exist a positive semidefinite matrix M_λ such that $g = \sum_{\lambda \in \Lambda_\ell} \langle M_\lambda, Z^\lambda \rangle$. Therefore, we have that $\widetilde{\mathbb{E}}(g) = \sum_{\lambda \in \Lambda_\ell} \widetilde{\mathbb{E}} \langle M_\lambda, Z^\lambda \rangle = \sum_{\lambda \in \Lambda_\ell} \langle M_\lambda, \widetilde{\mathbb{E}}(Z^\lambda) \rangle \geq 0$, since both M_λ and $\widetilde{\mathbb{E}}(Z^\lambda)$ are positive semidefinite for each partition $\lambda \in \Lambda_\ell$. By Lemma 2 we conclude that $\widetilde{\mathbb{E}}(f^2) \geq 0$ for every $f \in \mathbb{R}[y]/\mathbf{I}_E$ with $\deg(f^2) \leq \ell$. □

3.2 Construction of the spanning sets

In this section we show how to construct the spanning sets of the row subspaces in order to apply Lemma 3, which together with a particular linear operator provides the existence of a high-degree SoS pseudoexpectation. The structure of the configuration linear program allows us to further restrict the canonical spanning set obtained from monomials, by one that is combinatorially interpretable and adapted to our purposes.

Definition 2 (*Partial Schedule*) Let G_S be the directed bipartite graph with vertex partition given by $[m]$ and $\mathcal{C}$ and edges $S \subseteq [m] \times \mathcal{C}$. We say that $S \subseteq [m] \times \mathcal{C}$ is a *partial schedule* if for every $i \in [m]$ we have $\delta_S(i) \leq 1$, where $\delta_S(i)$ is the degree of vertex i in G_S.

We say that S is a partial schedule over H if $\{i \in [m] : (i, C) \in S\} \subseteq H$. We denote by $\mathcal{M}(S)$ the set of machines in $\{i \in [m] : \delta_S(i) = 1\}$, and we call $\mathcal{M}(S)$ the set of machines *incident* to S. Sometimes it is convenient to see a partial schedule S as a function from $\mathcal{M}(S)$ to $\mathcal{C}$, so we also say that S is partial schedule with domain $\mathcal{M}(S)$.

Example 2 Let $m = 4$ and the set of configurations $\mathcal{C} = \{C_1, C_2, C_3\}$. Then, the set $T = \{(1, C_1), (2, C_1), (4, C_2)\}$ is a partial schedule. The machine $i = 3$ is not incident to T. In this case, $\delta_T(C_1) = 2$ since there are two machines, $\{1, 2\}$, incident to C_1. The domain of T is $\mathcal{M}(T) = \{1, 2, 4\}$. The set $S = \{(1, C_1), (1, C_2)\}$ is not a partial schedule since $\delta_S(1) = 2$.

Proposition 1 *If $S \subseteq [m] \times \mathcal{C}$ is not a partial schedule, we have $y_S \equiv 0 \mod$ **sched**.*

Proof Since S it is not a partial schedule, there exists a machine $i \in [m]$ such that $\delta_S(i) \geq 2$. Therefore, to prove the proposition it is enough to check that $y_{iC} y_{iR} \equiv 0 \mod$ **sched** for every pair of different configurations $R, C \in \mathcal{C}$. Given a configuration $C \in \mathcal{C}$, we have that $\sum_{R \in \mathcal{C}\setminus\{C\}} y_{iC} y_{iR} \equiv \sum_{S \in \mathcal{C}\setminus\{C\}} y_{iC} y_{iR} + y_{iC}^2 - y_{iC} \equiv y_{iC}(\sum_{R \in \mathcal{C}} y_{iS} - 1) \equiv 0 \mod$ **sched**. On the other hand, $y_{iC}^2 y_{iR}^2 \equiv y_{iC} y_{iR}$ for every $R \in \mathcal{C}\setminus\{C\}$. This yields the result. □

Proposition 2 *Let $S \subseteq [m] \times \mathcal{C}$ be a partial schedule of cardinality at most ℓ. Then, $y_S \in span(\{y_L : |L| = \ell$ and S is a partial schedule$\})$.*

Proof Assume that $|S| < \ell$ since otherwise we are done. Let $H \subseteq [m]$ such that $|H| = \ell - |S|$ and $\delta_S(h) = 0$ for every $h \in H$, that is, H is subset of machines that is not incident to the edges S in the bipartite graph G_S. Observe that since S is a partial schedule, it is incident to exactly $|S|$ machines. Let $\mathcal{C}^H$ be the set of partial schedules with domain H. Since $\sum_{C \in \mathcal{C}} y_{hC} \equiv 1 \mod$ **sched** for every $h \in H$, we have $y_S \equiv y_S \prod_{h \in H} \sum_{C \in \mathcal{C}} y_{hC} \equiv \sum_{R \in \mathcal{C}^H} y_{S \cup R} \mod$ **sched**. In particular, for every $R \in \mathcal{C}^H$ we have that $S \cup R$ is a partial schedule, and $\deg(y_{S \cup R}) = |S| + \ell - |S| = \ell$.

In the following we construct spanning sets for the row subspaces. Given a tableau τ_λ with shape λ, the hook(τ_λ) is the tableau with shape $(\lambda_1, 1, \ldots, 1) \in \mathbb{Z}^{m-\lambda_1+1}$, its first row it is equal to the first row of τ_λ and the remaining elements of τ_λ fill the rest of the cells in increasing order over the rows. That part is called the *tail* of the hook, and we denote by tail(τ_λ) the elements of $[m]$ in the tail of hook(τ_λ), and row$(\tau_\lambda) = [m]\setminus$tail$(\tau_\lambda)$, that is the elements in the first row of the tableau.

Example 3 Let $m = 7$ and consider the partition $\lambda = (4, 2, 1)$. The tableau τ_λ at the left has shape λ and the tableau at the right is hook(τ_λ), with shape $(4, 1, 1, 1)$; row$(\tau_\lambda) = \{1, 2, 7, 4\}$ and tail$(\tau_\lambda) = \{3, 5, 6\}$.

$$\begin{array}{|c|c|c|c|}\hline 1 & 2 & 7 & 4 \\ \hline 5 & 6 \\ \cline{1-2} 3 \\ \cline{1-1}\end{array} \qquad \begin{array}{|c|c|c|c|}\hline 1 & 2 & 7 & 4 \\ \hline 3 \\ \cline{1-1} 5 \\ \cline{1-1} 6 \\ \cline{1-1}\end{array}$$

The following lemma gives a spanning set for the row subspaces obtained from the hook tableau. We denote by $\mathrm{sym}_{\mathrm{hook}(\tau_\lambda)}$ the symmetrization respect to the row subgroup of hook(τ_λ),

$$\mathrm{sym}_{\mathrm{hook}(\tau_\lambda)}(f) = \frac{1}{|\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}|} \sum_{\sigma \in \mathbf{R}_{\mathrm{hook}(\tau_\lambda)}} \sigma f. \tag{11}$$

The following lemma provides a spanning set for the row subspace based on the above family polynomials. The proof follows the lines of [45, Lemma 2].

Lemma 4 *Given a tableau τ_λ, the row subspace $\mathbf{W}_{\tau_\lambda}^\ell$ of $\mathbf{Q}^\ell$ is spanned by*

$$\left\{ sym_{hook(\tau_\lambda)}(y_S) : |S| = \ell \text{ and } S \text{ is a partial schedule} \right\}. \tag{12}$$

Proof Let $\mathcal{A} = \{q \in \mathbf{Q}^\ell : \sigma q = q \text{ for all } \sigma \in \mathcal{R}_{\tau_\lambda}\}$ and $\mathcal{A}' = \{q \in \mathbf{Q}^\ell : \sigma q = q \text{ for all } \sigma \in \mathcal{R}_{\mathrm{hook}(\tau_\lambda)}\}$. By definition, we have that $\mathbf{W}^\ell_{\tau_\lambda} \subseteq \mathcal{A}$, and since $\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}$ is a subgroup of $\mathcal{R}_{\tau_\lambda}$ it follows that $\mathcal{A} \subseteq \mathcal{A}'$. By Propositions 1 and 2, and the linearity of the symmetrization operator, we have that $\mathcal{A}'$ is spanned by the set in (12). $\square$

In the row subgroup $\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}$, the elements of $[m]$ that are in the tail remain fixed. The rest of the elements on the first row are permuted arbitrarily. In particular, $\mathcal{R}_{\mathrm{hook}(\tau_\lambda)} \cong S_{\lambda_1}$. Therefore, any permutation σ in $\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}$ acts over a monomial y_S by separating the bipartite graph G_S into those vertices in $\mathrm{tail}(\tau_\lambda)$ that are fixed by σ and the rest in $\mathrm{row}(\tau_\lambda)$ that can be permuted.

Configuration profiles Observe that bipartite graphs corresponding to different partial schedules are isomorphic if and only if the degree of every configuration is the same in both graphs. We say that a partial schedule is in γ*-profile*, with $\gamma : \mathcal{C} \to \mathbb{Z}_+$, if for every $C \in \mathcal{C}$ we have $\delta_S(C) = \gamma(C)$. Observe that a partial schedule in γ-profile has size $\sum_{C\in\mathcal{C}} \gamma(C)$, quantity that we denote by $\|\gamma\|$. We denote by $\mathrm{supp}(\gamma)$ the support of the vector γ, namely, $\{C \in \mathcal{C} : \gamma(C) > 0\}$.

Definition 3 Given a partial schedule T, we say that a partial schedule A over $[m]\backslash\mathcal{M}(T)$ is a (T,γ)-extension if A is in γ-profile. We denote by $\mathcal{F}(T,\gamma)$ the set of (T,γ)-extensions. In particular, every (T,γ)-extension has size $\|\gamma\|$.

Example 4 Consider $m = 4$, $\mathcal{C} = \{C_1, C_2\}$ and the partial schedule $T = \{(2, C_1), (3, C_2)\}$. If γ is given by $\gamma(C_1) = \gamma(C_2) = 1$, we have $\mathcal{F}(T,\gamma) = \{\{(1, C_1), (4, C_2)\}, \{(4, C_1), (1, C_2)\}\}$. If μ is given by $\mu(C_1) = 1$ and $\mu(C_2) = 0$, we have $\mathcal{F}(T,\mu) = \{\{(1, C_1)\}, \{(4, C_1)\}\}$.

Given a partial schedule T and a γ-profile, let $\mathcal{B}_{T,\gamma}$ be the polynomial defined by

$$\mathcal{B}_{T,\gamma} = \sum_{A\in\mathcal{F}(T,\gamma)} y_A, \tag{13}$$

if $\gamma \neq 0$, and 1 otherwise. In words, the polynomial above corresponds to sum over all those partial schedules in γ-profile that are not incident to $\mathcal{M}(T)$. The following theorem is the main result of this section.

Theorem 5 *Let $\lambda \in \Lambda_\ell$ and a tableau τ_λ of shape λ. Then, the row subspace $\mathbf{W}^\ell_{\tau_\lambda}$ of $\mathbf{Q}^\ell$ is spanned by*

$$\mathcal{P}^\lambda = \bigcup_{\omega=0}^{\ell} \left\{ y_T \mathcal{B}_{T,\gamma} : T \textit{ is partial schedule with } \mathcal{M}(T) = \textit{tail}(\tau_\lambda) \textit{ and } \|\gamma\| = \omega \right\}. \tag{14}$$

Proof By Lemma 4 it is enough to check that the set of polynomials in (12) is spanned by those in (14). Let S be a partial schedule of size ℓ. Let $\mathrm{tail}(S,\tau_\lambda)$ be the subset of S that is incident to the tail of the tableau, that is, $\{(i, C) \in S : i \in \mathrm{tail}(\tau_\lambda)\}$, and let $\mathrm{row}(S,\tau_\lambda) = S\backslash\mathrm{tail}(S,\tau_\lambda)$ be the edges of the partial schedule S incident to the first row of the tableau.

Claim 1 $sym_{hook(\tau_\lambda)}(y_S) = y_{tail(S,\tau_\lambda)} \cdot sym_{hook(\tau_\lambda)}\left(y_{row(S,\tau_\lambda)}\right).$

Observe that $\mathrm{tail}(S, \tau_\lambda)$ is a partial schedule over $\mathrm{tail}(\tau_\lambda)$. Similarly as we did in Lemma 2, the partial schedule incident to the tail can be completed to be in the span of partial schedules with domain equal to $\mathrm{tail}(\tau_\lambda)$, that is,

$$\begin{aligned} y_{\mathrm{tail}(S,\tau_\lambda)} &\equiv y_{\mathrm{tail}(S,\tau_\lambda)} \prod_{h\in\mathrm{tail}(\tau_\lambda)\setminus\mathrm{tail}(S,\tau_\lambda)} \sum_{C\in\mathcal{C}} y_{hC} \\ &\equiv \sum_{L\in\mathcal{C}^{\mathrm{tail}(\tau_\lambda)\setminus\mathrm{tail}(S,\tau_\lambda)}} y_{\mathrm{tail}(S,\tau_\lambda)\cup L} \quad \bmod \mathbf{sched} \end{aligned}$$

where $\mathcal{C}^{\mathrm{tail}(\tau_\lambda)\setminus\mathrm{tail}(S,\tau_\lambda)}$ is the set of partial schedules with domain $\mathrm{tail}(\tau_\lambda)\setminus\mathrm{tail}(S, \tau_\lambda)$. Thus, every partial schedule in the summation above have domain $\mathrm{tail}(\tau_\lambda) \cup \mathrm{tail}(S, \tau_\lambda)\setminus\mathrm{tail}(S, \tau_\lambda) = \mathrm{tail}(\tau_\lambda)$. Therefore, it is enough to check that exists a constant κ such that $\mathrm{sym}_{\mathrm{row}(\tau_\lambda)}\left(y_{\mathrm{row}(S,\tau_\lambda)}\right) = \kappa \cdot \mathcal{B}_{\mathrm{tail}(\tau_\lambda),\gamma}$ for some profile γ with $\|\gamma\| = \ell - |\mathrm{tail}(S, \tau_\lambda)|$. Recall that $|\mathrm{tail}(S, \tau_\lambda)| \leq \ell$ since $\lambda \in \Lambda_\ell$. Let γ be the profile of the partial schedule $\mathrm{row}(S, \tau_\lambda)$. The equality follows since $\sigma \in \mathcal{R}_{\mathrm{hook}(\tau_\lambda)} \cong S_{\mathrm{row}(\tau_\lambda)}$, together with the fact that $\{(\sigma(i), C) : (i, C) \in \mathrm{row}(S, \tau_\lambda)\}$ is a $(\mathrm{tail}(\tau_\lambda), \gamma)$-extension for every permutation in $\sigma \in \mathcal{R}_{\mathrm{hook}(\tau_\lambda)}$. The constant κ is equal to $|\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}|$. □

Proof of Claim 1 Observe that for every permutation $\sigma \in \mathcal{R}_{\mathrm{hook}(\tau_\lambda)}$, we have

$$\sigma y_S = \prod_{(i,C)\in\mathrm{tail}(S,\tau_\lambda)} y_{\sigma(i)C} \prod_{(i,C)\in\mathrm{row}(S,\tau_\lambda)} y_{\sigma(i)C} = y_{\mathrm{tail}(S,\tau_\lambda)}\sigma y_{\mathrm{row}(S,\tau_\lambda)},$$

since the permutation fixes the edges in $\mathrm{tail}(S, \tau_\lambda)$. Therefore, symmetrizing yields that $\mathrm{sym}_{\mathrm{hook}(\tau_\lambda)}(y_S)$ is equal to

$$\begin{aligned} \frac{1}{|\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}|} \sum_{\sigma\in\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}} \sigma y_S &= y_{\mathrm{tail}(S,\tau_\lambda)} \cdot \frac{1}{|\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}|} \sum_{\sigma\in\mathcal{R}_{\mathrm{hook}(\tau_\lambda)}} \sigma y_{\mathrm{row}(S,\tau_\lambda)} \\ &= y_{\mathrm{tail}(S,\tau_\lambda)} \cdot \mathrm{sym}_{\mathrm{hook}(\tau_\lambda)}\left(y_{\mathrm{row}(S,\tau_\lambda)}\right). \end{aligned}$$

□

3.3 High-degree SoS pseudoexpectation: Proof of Theorem 1

We now have the ingredients to study the scheduling ideal and we describe the pseudoexpectations from Theorem 3, that are the base for our lower bound. Recall that for every odd $k \in \mathbb{N}$, the hard instance I_k has $m = 3k$ machines and the linear operators we consider are supported over partial schedules incident to a set of six so called *matching configurations*, $\{C_1, \ldots, C_6\}$. Consider the $\widetilde{\mathbb{E}} : \mathbb{R}[y]/\mathbf{I}_E \to \mathbb{R}$ such that for every partial schedule S of cardinality at most $k/2$,

$$\widetilde{\mathbb{E}}(y_S) = \frac{1}{(3k)_{|S|}} \prod_{j=1}^{6} (k/2)_{\delta_S(C_j)}, \tag{15}$$

where $(a)_b$ is the lower factorial function, that is, $(a)_b = a(a-1)\cdots(a-b+1)$, and $(a)_0 = 1$. The linear operator $\widetilde{\mathbb{E}}$ is zero elsewhere. We state formally the main result that implies Theorem 1.

Theorem 6 *For every odd $k \in \mathbb{N}$, the linear operator $\widetilde{\mathbb{E}}$ is a degree-$\lfloor k/6 \rfloor$ SoS pseudoexpectation for the configuration linear program in instance I_k and $T = 1023$.*

Proof of Theorem 1 For every odd k the instance I_k described in Sect. 3 is infeasible for $T = 1023$. By Theorem 6, the operator $\widetilde{\mathbb{E}}$ is a degree-$\lfloor k/6 \rfloor$ SoS pseudoexpectation, which in turns imply by Lemma 1 that there is no degree-$\lfloor k/6 \rfloor$ SoS certificate of infeasibility. For an instance with n jobs, let k be the greatest odd integer such that $n = 15k + \ell$, with $\ell < 30$. The theorem follows by considering the instance I_k above with ℓ dummy jobs of processing time equal to zero. □

Theorem 3 guarantees that for every k odd, $\widetilde{\mathbb{E}}$ is a degree-$\lfloor k/2 \rfloor$ pseudoexpectation, and therefore a degree-$\lfloor k/6 \rfloor$ pseudoexpectation as well. In particular, properties (SoS.1) and (SoS.4) are satisfied. Since the configuration linear program is constructed from equality constraints, it is enough to check property (SoS.2) for high enough degree, in this case $\ell = \lfloor k/6 \rfloor$. To check property (SoS.2) we require a notion of *conditional pseudoexpectations*. Given a partial schedule T, consider the operator $\widetilde{\mathbb{E}}_T : \mathbb{R}[y]/\mathbf{I}_E \to \mathbb{R}$ such that

$$\widetilde{\mathbb{E}}_T(y_S) = \frac{1}{(3k - |T|)!} \prod_{j=1}^{6} (k/2 - \delta_T(C_j))_{\delta_S(C_j)} \tag{16}$$

for every partial schedule S over the machines $[m] \setminus \mathcal{M}(T)$ and zero otherwise. Observe that if $T = \emptyset$ it corresponds to the linear operator $\widetilde{\mathbb{E}}$ in (15). The following lemmas about the conditional pseudoexpectation in (16) are key for proving that $\widetilde{\mathbb{E}}$ is a high-degree SoS pseudoexpectation. We state the lemmas and show how to conclude Theorem 6 using them. In particular, in Lemma 7 we prove a strong *pseudoindependence* property satisfied by the conditional pseudoexpectations and the polynomials (13) in the spanning set.

Lemma 5 *The linear operator $\widetilde{\mathbb{E}}$ is S_m-symmetric.*

Lemma 6 *Let T be a partial schedule. Then, the following holds:*

(*a*) *If S is a partial schedule and $T \cap S = \emptyset$, then $\widetilde{\mathbb{E}}(y_T y_S) = \widetilde{\mathbb{E}}_T(y_S)\widetilde{\mathbb{E}}(y_T)$.*
(*b*) *If S, R are two partial schedules such that $R \cap S = \emptyset$ and $T \cap (R \cup S) = \emptyset$, then $\widetilde{\mathbb{E}}_T(y_R y_S) = \widetilde{\mathbb{E}}_T(y_R)\widetilde{\mathbb{E}}_{T\cup R}(y_S)$.*
(*c*) *Let γ be a profile with $supp(\gamma) \subseteq \{C_1, \ldots, C_6\}$ and $|T| + \|\gamma\| \le k/2$. Then,*

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma}) = \prod_{j=1}^{6} \frac{1}{\gamma(C_j)!} (k/2 - \delta_T(C_j))_{\gamma(C_j)}.$$

Lemma 7 *Let T be a partial schedule and γ, μ a pair of configuration profiles with $|T| + \|\gamma\| + \|\mu\| \leq k/2$ and $supp(\gamma)$, $supp(\mu) \subseteq \{C_1, \ldots, C_6\}$. Then,*

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma}\mathcal{B}_{T,\mu}) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma})\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\mu}). \tag{17}$$

Proof of Theorem 6 Let $\ell = \lfloor k/6 \rfloor$. Given a partition $\lambda \in \lambda_\ell$, consider the tableau τ_λ such that $\mathrm{tail}(\tau_\lambda) = [3k - \lambda_1]$ and $\mathrm{row}(\tau_\lambda) = [3k] \backslash [3k - \lambda_1]$. The partial schedules with domain $[3k - \lambda_1]$ can be identified with $\mathcal{C}^{[3k-\lambda_1]}$, the set of functions from $[3k - \lambda_1]$ to $\mathcal{C}$. In particular the spanning set in (14) is described by $\mathcal{P}^\lambda = \bigcup_{\omega=0}^{\ell} \left\{ y_T \beta_{T,\gamma} : T \in \mathcal{C}^{[3k-\lambda_1]} \text{ and } \|\gamma\| = \omega \right\}$. To apply Lemma 3 we need to study the matrix $\widetilde{\mathbb{E}}(Z^\lambda)$. Recall that for $T, S \in \mathcal{C}^{[3k-\lambda_1]}$ and profiles γ, ν with $\|\gamma\|, \|\mu\| \leq \ell$, the corresponding entry of the matrix $\widetilde{\mathbb{E}}(Z^\lambda)$ is given by

$$\widetilde{\mathbb{E}}\left(\mathrm{sym}\Big(y_T y_S \beta_{T,\gamma}\beta_{S,\mu}\Big)\right) = \widetilde{\mathbb{E}}\left(\mathrm{sym}\Big(y_{T\cup S}\beta_{T,\gamma}\beta_{S,\mu}\Big)\right).$$

By Lemma 5 the operator $\widetilde{\mathbb{E}}$ is symmetric, and therefore,

$$\widetilde{\mathbb{E}}\left(\mathrm{sym}\Big(y_{T\cup S}\beta_{T,\gamma}\beta_{S,\mu}\Big)\right) = \widetilde{\mathbb{E}}\Big(y_{T\cup S}\beta_{T,\gamma}\beta_{S,\mu}\Big).$$

Since both T, S are partial schedules such that $\mathcal{M}(T) = \mathcal{M}(S)$, we have that $T \cup S$ is a partial schedule if and only if $T = S$. Thus, the matrix $\widetilde{\mathbb{E}}(Z^\lambda)$ is block diagonal, with a block for each partial schedule $T \in \mathcal{C}^{[3k-\lambda_1]}$. For every Θ indexed by the elements of the spanning set above, we have then

$$\left\langle \widetilde{\mathbb{E}}(Z^\lambda), \Theta\Theta^\top \right\rangle = \sum_{T\in\mathcal{C}^{[3k-\lambda_1]}} \sum_{\substack{\gamma:\|\gamma\|\leq\ell \\ \mu:\|\mu\|\leq\ell}} \widetilde{\mathbb{E}}\Big(y_T\beta_{T,\gamma}\beta_{T,\mu}\Big)\Theta_{T,\gamma}\Theta_{T,\mu}.$$

Since $|T| + \|\gamma\| + \|\mu\| \leq 3\ell \leq k/2$ for every partial schedule T and profiles γ, μ as above, by applying Lemma 6 (a) and Lemma 7 we obtain that

$$\begin{aligned}
&\sum_{T\in\mathcal{C}^{[3k-\lambda_1]}} \sum_{\substack{\gamma:\|\gamma\|\leq\ell \\ \mu:\|\mu\|\leq\ell}} \widetilde{\mathbb{E}}\Big(y_T\beta_{T,\gamma}\beta_{T,\mu}\Big)\Theta_{T,\gamma}\Theta_{T,\mu} \\
&\quad = \sum_{T\in\mathcal{C}^{[3k-\lambda_1]}} \widetilde{\mathbb{E}}(y_T) \sum_{\substack{\gamma:\|\gamma\|\leq\ell \\ \mu:\|\mu\|\leq\ell}} \widetilde{\mathbb{E}}_T(\beta_{T,\gamma})\widetilde{\mathbb{E}}_T(\beta_{T,\mu})\Theta_{T,\gamma}\Theta_{T,\mu},
\end{aligned}$$

and by rearranging terms we conclude that

$$\left\langle \widetilde{\mathbb{E}}(Z^\lambda), \Theta\Theta^\top \right\rangle = \sum_{T\in\mathcal{C}^{[3k-\lambda_1]}} \widetilde{\mathbb{E}}(y_T) \left(\sum_{\gamma:\|\gamma\|\leq\ell} \widetilde{\mathbb{E}}_T(\beta_{T,\gamma})\Theta_{T,\gamma} \right)^2 \geq 0.$$

□

Proof of Lemma 5 Given $\sigma \in S_m$ and a partial schedule S, $\widetilde{\mathbb{E}}(\sigma y_S) = \widetilde{\mathbb{E}}(y_{\sigma(S)})$, where $\sigma(S) = \{(\sigma(i), C) : (i, C) \in S\}$. In particular, since $|S| = |\sigma(S)|$ and profile of S is the same profile of $\sigma(S)$, it holds $\widetilde{\mathbb{E}}(y_S) = \widetilde{\mathbb{E}}(\sigma y_S)$. Therefore, $\widetilde{\mathbb{E}}(y_S) = \frac{1}{m!}\sum_{\sigma\in S_m}\widetilde{\mathbb{E}}(\sigma y_S) = \widetilde{\mathbb{E}}(\mathrm{sym}(y_S))$.

Proof of Lemma 6 Property b) implies a) by taking $T = \emptyset$. One can check from the definition of the lower factorial that $(x)_{a+b} = (x)_a(x-a)_b$. Since the partial schedules R, S and T are disjoint, it holds for every $C \in \mathcal{C}$ that $\delta_{R\cup S}(C) = \delta_R(C) + \delta_S(C)$ and $\delta_{T\cup R}(C) = \delta_T(C) + \delta_R(C)$. Therefore,

$$
\begin{aligned}
&(3k - |T|)_{|R\cup S|} \cdot \widetilde{\mathbb{E}}_T(y_R y_S) \\
&\quad = \prod_{j=1}^{6}(k/2 - \delta_T(C_j))_{\delta_R(C_j)} \cdot \prod_{j=1}^{6}(k/2 - \delta_T(C_j) - \delta_R(C_j))_{\delta_S(C_j)} \\
&\quad = (3k - |T|)_{|R|} \cdot \widetilde{\mathbb{E}}_T(y_R) \cdot (3k - |T| - |R|)_{|S|} \cdot \widetilde{\mathbb{E}}_{T\cup R}(y_S),
\end{aligned}
$$

and the lemma follows since $(3k - |T|)_{|R\cup S|} = (3k - |T|)_{|R|} \cdot (3k - |T| - |R|)_{|S|}$. We now prove property (c), that is more involved. First of all, observe that for every $H \in \mathcal{F}(T, \gamma)$ the value of $\widetilde{\mathbb{E}}_T(y_H)$ depends only on T and the configuration profile γ. More specifically,

$$
\widetilde{\mathbb{E}}_T(y_H) = \frac{1}{(3k - |T|)_{\|\gamma\|}} \prod_{j=1}^{6}(k/2 - \delta_T(C_j))_{\gamma(C_j)},
$$

since $|H| = \|\gamma\|$ and $\delta_H(C_j) = \gamma(C_j)$ for every $j \in \{1, \dots, 6\}$. Then, $\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma})$ equals $|\mathcal{F}(T, \gamma)|$ times the quantity above. The number of machines that can support a partial schedule H that extend T is $3k - |T|$, and since $|H| = \|\gamma\|$ the number of possible machine domains is $\binom{3k-|T|}{\|\gamma\|}$. Given a set of machines with cardinality $\|\gamma\|$, the number of partial schedules with domain equal to this set of machines and that are in configuration profile γ are $\|\gamma\|!\prod_{j=1}^{6}\frac{1}{\gamma(C_j)!}$. Then, overall, the value of $\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma})$ is equal to

$$
\begin{aligned}
&\binom{3k - |T|}{\|\gamma\|}\|\gamma\|!\frac{1}{(3k - |T|)_{\|\gamma\|}} \cdot \prod_{j=1}^{6}\frac{1}{\gamma(C_j)!}(k/2 - \delta_T(C_j))_{\gamma(C_j)} \\
&= \frac{(3k - |T|)!}{(3k - |T| - \|\gamma\|)!} \cdot \frac{1}{(3k - |T|)_{\|\gamma\|}} \cdot \prod_{j=1}^{6}\frac{1}{\gamma(C_j)!}(k/2 - \delta_T(C_j))_{\gamma(C_j)} \\
&= \prod_{j=1}^{6}\frac{1}{\gamma(C_j)!}(k/2 - \delta_T(C_j))_{\gamma(C_j)},
\end{aligned}
$$

in the last step we used that for every real x and non-negative integer b, it holds $(x - b)!(x)_b = x!$. ◻

To prove Lemma 7 we obtain first a weaker version, that together with a polynomial decomposition in the scheduling ideal yields to the pseudoindependence result.

Lemma 8 *Let $T \subseteq [m] \times \mathcal{C}$ be a partial schedule.*

(a) If ν and ξ are configuration profiles such that supp$(\nu) \cap$ supp$(\xi) = \emptyset$ and $|T| + \|\nu\| + \|\xi\| \leq k/2$, then $\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu}\mathcal{B}_{T,\xi}) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu})\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\xi})$.
(b) If ν and ξ are configuration profiles such that there exists $C \in \{C_1, \dots, C_6\}$ with supp(ν), supp$(\xi) \subseteq \{C\}$, and $|T| + \|\nu\| + \|\xi\| \leq k/2$, then we have that $\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu}\mathcal{B}_{T,\xi}) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu})\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\xi})$.

Proof In both case if one of the profiles is zero then the conclusion follows. Then, in what follows assume that ν and ξ are different from zero, and their support is contained in $\{C_1, \dots, C_6\}$. Consider ν and ξ satisfying the conditions in (a) and fix $A \in \mathcal{F}(T, \nu)$. Then,

$$\widetilde{\mathbb{E}}_T(y_A\mathcal{B}_{T,\xi}) = \sum_{B \in \mathcal{F}(T \cup A, \xi)} \widetilde{\mathbb{E}}(y_A y_B) + \sum_{B \in \mathcal{F}(T,\xi) \setminus \mathcal{F}(T \cup A, \xi)} \widetilde{\mathbb{E}}(y_A y_B),$$

where the equality holds since $\mathcal{F}(T \cup A, \xi) \subseteq \mathcal{F}(T, \xi)$. For every term $B \in \mathcal{F}(T, \xi) \setminus \mathcal{F}(T \cup A, \xi)$ we have that it is incident to at least one of the machines in G_A. Since every machine in G_A is connected to a machine in supp$(\nu) \subseteq \mathcal{C} \setminus$supp$(\xi)$, it follows that $A \cup B$ is not a partial schedule since at least one machine is connected to different configurations, and in consequence its pseudoexpectation is zero. Therefore, the second summation in the equality above is zero. Together with property (b) in Lemma 6 it implies that

$$\widetilde{\mathbb{E}}_T(y_A\mathcal{B}_{T,\xi}) = \sum_{B \in \mathcal{F}(T \cup A, \xi)} \widetilde{\mathbb{E}}(y_A y_B) = \widetilde{\mathbb{E}}_T(y_A) \cdot \widetilde{\mathbb{E}}_{T \cup A}(\mathcal{B}_{T \cup A, \xi}).$$

Since supp$(\nu) \cap$ supp$(\xi) = \emptyset$, we have that for every $C_j \in$ supp(ξ), $\delta_{T \cup A}(C_j) = \delta_T(C_j)$. On the other hand, if $C_j \notin$ supp(ξ) then $(x)_{\xi(C_j)} = (x)_0 = 1$ for every real x. Overall, and together with Lemma 6, it holds that

$$\begin{aligned}\widetilde{\mathbb{E}}_{T \cup A}(\mathcal{B}_{T \cup A, \xi}) &= \prod_{j=1}^{6} \frac{1}{\xi(C_j)!}(k/2 - \delta_{T \cup A}(C_j))_{\xi(C_j)} \\ &= \prod_{j \in \text{supp}(\xi)} \frac{1}{\xi(C_j)!}(k/2 - \delta_T(C_j))_{\xi(C_j)} = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\xi}).\end{aligned}$$

Together with the linearity of $\widetilde{\mathbb{E}}_T$ we conclude (a). Consider now ν, ξ satisfying the conditions in (b), and let $C \in \{C_1, \dots, C_6\}$ the configuration that supports both profiles. Without loss of generality suppose that $\nu(C) \geq \xi(C)$. For $A \in \mathcal{F}(T, \nu)$ and $B \in \mathcal{F}(T, \xi)$, we have that $A \cup B$ is always a perfect matching since the profiles are supported in the same configuration. If $B \subseteq A$, then the union has profile ν. Then, by Lemma 6 (c) we have

$$\begin{aligned}\widetilde{\mathbb{E}}_T(y_A\mathcal{B}_{T,\xi}) &= \sum_{B\in\mathcal{F}(T,\xi)} \widetilde{\mathbb{E}}_T(y_Ay_B)\\ &= \sum_{B\in\mathcal{F}(T,\xi):B\subseteq A} \widetilde{\mathbb{E}}_T(y_A) + \sum_{B\in\mathcal{F}(T,\xi):B\setminus A\neq\emptyset} \widetilde{\mathbb{E}}_T(y_Ay_{B\setminus A})\\ &= \widetilde{\mathbb{E}}_T(y_A)\left(\binom{\nu(C)}{\xi(C)} + \sum_{B\in\mathcal{F}(T,\xi):B\setminus A\neq\emptyset} \widetilde{\mathbb{E}}_{T\cup A}(y_{B\setminus A})\right).\end{aligned}$$

If $B \setminus A \neq \emptyset$, the union profile can be parameterized in $|B \setminus A| = \omega$, and let α_ω be the profile such that $\alpha_\omega(C) = \omega$ and zero otherwise. Thus,

$$\begin{aligned}&\sum_{B\in\mathcal{F}(T,\xi):B\setminus A\neq\emptyset} \widetilde{\mathbb{E}}_{T\cup A}(y_{B\setminus A})\\ &= \sum_{\omega=1}^{\xi(C)} \binom{\nu(C)}{\xi(C)-\omega}\binom{3k-|T|-\nu(C)}{\omega}\frac{(k/2-\delta_T(C)-\nu(C))_\omega}{(3k-|T|-\nu(C))_\omega}\\ &= \sum_{\omega=1}^{\xi(C)} \frac{1}{\omega!}\binom{\nu(C)}{\xi(C)-\omega}(k/2-\delta_T(C)-\nu(C))_\omega,\end{aligned}$$

and since $(k/2-\delta_T(C)-\nu(C))_0 = 1$, and running the summation over $A \in \mathcal{F}(T, \nu)$ we obtain over all that

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu}\mathcal{B}_{T,\xi}) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu}) \cdot \sum_{\omega=0}^{\xi(C)} \frac{1}{\omega!}\binom{\nu(C)}{\xi(C)-\omega}(k/2-\delta_T(C)-\nu(C))_\omega. \tag{18}$$

Claim 2 *Let a and b be two non-negative integers such that $a \leq b$. Then, for every real x,*

$$\sum_{\omega=0}^{a} \frac{1}{\omega!}\binom{b}{a-\omega}(x-b)_\omega = \frac{1}{a!}(x)_a.$$

The claim applied in (18) for $x = k/2 - \delta_T(C)$, $a = \xi(C)$ and $b = \nu(C)$ yields the result, since

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu}\mathcal{B}_{T,\xi}) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu}) \cdot \frac{1}{\xi(C)!}(k/2-\delta_T(C))_{\xi(C)} = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\nu})\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\xi}).$$

The claim follows by the Chu-Vandermonde identity [3, p. 59–60],

$$\begin{aligned}(x)_a &= \sum_{\omega=0}^{a}\binom{a}{\omega}(x-b)_\omega (b)_{a-\omega}\\ &= a!\sum_{\omega=0}^{a}(x-b)_\omega \frac{(b)_{a-\omega}}{(a-\omega)!} = a!\sum_{\omega=0}^{a}(x-b)_\omega \binom{b}{a-\omega}\end{aligned}$$

□

Proof of Lemma 7 Given a profile configuration γ and $C_j \in \{C_1,\ldots,C_6\}$, we denote by γ_j the profile that is zero for every $C \neq C_j$ and $\gamma_j(C_j) = \gamma(C_j)$. In the following, we prove that the following factorization holds:

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma}\mathcal{B}_{T,\mu}) = \widetilde{\mathbb{E}}_T\left(\prod_{j=1}^{6}\mathcal{B}_{T,\gamma_j}\mathcal{B}_{T,\mu_j}\right), \tag{19}$$

recalling that $\mathcal{B}_{T,\xi} = 1$ if $\xi = 0$. Before checking that the decomposition above is correct, we see how to conclude the lemma from that. Observe that by construction $\mathrm{supp}(\gamma_j) \cap \mathrm{supp}(\gamma_\ell) = \emptyset$ if $j \neq \ell$, and therefore by Lemma 8 (a), we have

$$\widetilde{\mathbb{E}}_T\left(\prod_{j=1}^{6}\mathcal{B}_{T,\gamma_j}\mathcal{B}_{T,\mu_j}\right) = \prod_{j=1}^{6}\widetilde{\mathbb{E}}_T\left(\mathcal{B}_{T,\gamma_j}\mathcal{B}_{T,\mu_j}\right). \tag{20}$$

Furthermore, since for every $j \in \{1,\ldots,6\}$ we have $\mathrm{supp}(\gamma_j), \mathrm{supp}(\mu_j) \subseteq \{C_j\}$, by Lemma 8 (b) we have

$$\prod_{j=1}^{6}\widetilde{\mathbb{E}}_T\left(\mathcal{B}_{T,\gamma_j}\mathcal{B}_{T,\mu_j}\right) = \prod_{j=1}^{6}\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma_j})\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\mu_j}) = \prod_{j=1}^{6}\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma_j})\cdot\prod_{j=1}^{6}\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\mu_j}).$$

By using Lemma 8 (a) the right hand side is equal to

$$\widetilde{\mathbb{E}}_T\left(\prod_{j=1}^{6}\mathcal{B}_{T,\gamma_j}\right)\cdot\widetilde{\mathbb{E}}_T\left(\prod_{j=1}^{6}\mathcal{B}_{T,\mu_j}\right) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma}\mathcal{B}_{T,\mu}),$$

where in the last equality we used the decomposition in (19) separately for γ and μ. We check now that the factorization in (19) is always valid. Let S be a partial schedule disjoint from T and with profile μ and let $C_j \in \mathrm{supp}(\gamma)$. It is enough to check that

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma}y_S) = \widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma_j}\mathcal{B}_{T,\gamma-\gamma_j}y_S), \tag{21}$$

since the factorization follows by the linearity of $\widetilde{\mathbb{E}}_T$ and by applying iteratively for every $C_j \in \{C_1,\ldots,C_6\}$ the above factorization. We have that

$$\widetilde{\mathbb{E}}_T(\mathcal{B}_{T,\gamma} y_S) = \widetilde{\mathbb{E}}_T\left(\sum_{A\in\mathcal{F}(T,\gamma)} y_A y_S\right)$$
$$= \widetilde{\mathbb{E}}_T\left(\sum_{B\in\mathcal{F}(T,\gamma_j)} y_B \sum_{D\in\mathcal{F}(T\cup B,\gamma-\gamma_j)} y_D y_S\right).$$

Fix $B \in \mathcal{F}(T, \gamma_j)$ and consider a set $D \in \mathcal{F}(T, \gamma - \gamma_j) \setminus \mathcal{F}(T \cup B, \gamma - \gamma_j)$. In particular, D is in profile $\gamma - \gamma_j$ but is incident to at least one machine, say ℓ, that is also incident to B. Since B is in profile γ_j and it has disjoint support from $\gamma - \gamma_j$, the above implies that machine ℓ is incident to different configurations, and therefore its pseudoexpectation value is equal to zero. That is the contribution to the pseudoexpectation value of the terms in $\mathcal{F}(T, \gamma - \gamma_j) \setminus \mathcal{F}(T \cup B, \gamma - \gamma_j)$ is is zero. Furthermore, since $\mathcal{F}(T, \gamma - \gamma_j) \supseteq \mathcal{F}(T \cup B, \gamma - \gamma_j)$, we have that for every $B \in \mathcal{F}(T, \gamma_j)$,

$$\widetilde{\mathbb{E}}_T\left(y_B \sum_{D\in\mathcal{F}(T\cup B,\gamma-\gamma_j)} y_D y_S\right)$$
$$= \widetilde{\mathbb{E}}_T\left(y_B\left(\sum_{D\in\mathcal{F}(T\cup B,\gamma-\gamma_j)} y_D + \sum_{D\in\mathcal{F}(T,\gamma-\gamma_j)\setminus\mathcal{F}(T\cup B,\gamma-\gamma_j)} y_D\right) y_S\right)$$
$$= \widetilde{\mathbb{E}}_T\left(y_B \sum_{D\in\mathcal{F}(T,\gamma-\gamma_j)} y_D y_S\right) = \widetilde{\mathbb{E}}_T(y_B \mathcal{B}_{T,\gamma-\gamma_j} y_S).$$

We conclude by summing over $B \in \mathcal{F}(T, \gamma_j)$, $S \in \mathcal{F}(T, \mu)$ and using the linearity of $\widetilde{\mathbb{E}}_T$. □

Remark It is worth noticing that the lower bound of Theorem 1 translates to the weaker assignment linear program (see (22)–(23) in Sect. 4, which define the linear program assign(T)). More precisely, there exists an instance such that, after applying $\Omega(n)$ rounds of the SoS hierarchy to the assignment linear program, the semidefinite relaxation has an integrality gap of at least 1.0009. This follows by Theorem 1 and a general result by Au and Tunçel [4]. More details can be found in "Appendix B".

4 Upper bound: breaking symmetries to approximate the makespan

In the previous section we showed that the configuration linear program has an inherent difficulty for the SoS method with low (constant) degree to yield a $(1 + \varepsilon)$ integrality gap (and hence also the weaker assignment linear program below). It is natural to ask whether there is a way to avoid this lower bound. As suggested by our proof in Sect. 3 and several other lower bounds in the literature [17,31,34,42,44], symmetries seem to play an role in the quality of the relaxations obtained by the SoS and SA hierarchies.

A natural question is whether *breaking* the symmetries of a problem or instance might help avoiding the lower bounds. In what follows we show that this is the case for the makespan scheduling problem. We leave as an interesting open problem whether this is the case for other relevant problems.

Symmetry breaking Breaking symmetries is a common technique to avoid algorithmic problems of symmetric instances of non-convex programs, in particular integer programs [39]. Recall that given an optimization problem (P): $\min\{f(x) : x \in X\}$ for some set $X \subseteq \mathbb{R}^n$ and a group G acting on $\mathbb{R}^n$ by an action $(g, x) \mapsto gx$, we say that (P) is G-invariant if $f(x) = f(gx)$ and $gx \in X$ for all $x \in X$ and $g \in G$. Notice that if x^* is an optimal solution to (P), then gx^* is also optimal for every $g \in G$ in this case. Hence, if we add to the formulation any inequality $a^\top x \leq b$ that keeps at least one representative of any given orbit $\{gx : g \in G\}$ for any $x \in \mathbb{R}^n$, that is, for all $x \in \mathbb{R}^n$ there exists $g \in G$ such that $a^\top(gx) \leq b$, then we guarantee that (P'): $\min\{f(x) : x \in X, a^\top x \leq b\}$ contains at least one optimal solution. If such inequality is not valid for (P), we say that it is a *symmetry breaking inequality*.[1]

Application to scheduling We show that we can obtain almost optimal relaxations in terms of the integrality gap if we add a well chosen set of symmetry breaking inequalities to a ground formulation and then apply the SA hierarchy (which is even weaker than SoS). Furthermore, the ground formulation we use is the *assignment linear program*. In this LP there are variables x_{ij} indicating whether job j is assigned to machine i. For an estimate or guess T for the optimal makespan we denote by assign(T) the formulation given by

$$\sum_{i\in[m]} x_{ij} = 1 \quad \text{for all } j \in J, \tag{22}$$

$$\sum_{j\in J} x_{ij} p_j \leq T \quad \text{for all } i \in [m], \tag{23}$$

$$x_{ij} \geq 0 \quad \text{for all } i \in [m], \text{ for all } j \in J. \tag{24}$$

If we require that $T \geq \max_{j\in J} p_j$ then the assignment linear program has an integrality gap of 2 [54].

Roadmap In Sect. 4.1 we define the symmetry breaking inequalities that we will add to the assignment linear program. In Sect. 4.2 we will show how to round a feasible solution of the SA hierarchy with $2^{\tilde{O}(1/\varepsilon^2)}$ rounds over this program to obtain an integral solution with makespan $(1+\varepsilon)T$. In Sect. 4.3 we will show that breaking some new *approximate symmetries*, with just $O(1/\varepsilon^5)$ rounds of the SA hierarchy suffices to obtain a $(1+\varepsilon)$-approximate solution, yielding an exponential decrease in the number of necessary rounds. By approximate symmetries we mean that first we

[1] It is worth noticing that such an inequality might not break all symmetries, that is, we do not require that there is a unique representative of each orbit.

round similar processing times to the same value, and then add symmetry breaking inequalities for the new induced symmetries.

4.1 Symmetry breaking inequalities

In order to define our symmetry breaking inequalities we consider a partitioning obtained by grouping long jobs with a similar processing time. Let $\varepsilon \in (0, 1)$ such that $1/\varepsilon \in \mathbb{Z}$. We say that a job $j \in J$ is *long* if $p_j \geq \varepsilon \cdot T$, and it is *short* otherwise. The subset of long jobs is denoted by J_{long} and the short jobs are $J_{\text{short}} = J \setminus J_{\text{long}}$. For every $q \in \{1, \dots, (1-\varepsilon)/\varepsilon^2\}$ we define

$$J_q = \left\{ j \in J_{\text{long}} : \left(\frac{1}{\varepsilon} + q\right) \varepsilon^2 T > p_j \geq \left(\frac{1}{\varepsilon} + q - 1\right) \varepsilon^2 T \right\}.$$

Let $s := (1-\varepsilon)/\varepsilon^2$ denote the number of groups of long jobs. The reader may imagine that for each group J_q with $q \in [s]$ we round the size of each job $j \in J_q$ to $\left(\frac{1}{\varepsilon} + q\right) \varepsilon^2 T$. This increases the overall makespan at most by a factor $1 + \varepsilon$. Also note that if we can find a schedule for the long jobs with makespan at most $(1+\varepsilon)T$, then there is also a schedule for *all* jobs with makespan at most $(1+\varepsilon)T$ since we can add the short jobs in a greedy manner (see e.g.,[54]; we assume that $\text{assign}(T)$ is feasible and then $\sum_{j \in J} p_j \leq m \cdot T$ holds).

Configurations Based on the partition of the long jobs $\{J_q\}_{q \in [s]}$ we define configurations of the long jobs. We say that a *configuration* C is a multiset of elements in $\{1, \dots, s\}$. Let $\mathcal{C}$ denote the set of all configurations. Similarly as in Sect. 3, for a configuration C we define $m(q, C)$ to be the number of times that q appears (repeated) in C. Intuitively, this means that configuration C contains $m(q, C)$ slots for jobs in J_q. In what follows, we introduce a set of constraints that guarantees that *every* integer solution to $\text{assign}(T)$ obeys a specific order on the configurations over the machines, i.e., there is a total order of the configurations C such that for two machines $i, i' \in [m]$ with $i < i'$ the configuration on i is smaller according to this total ordering than the configuration on i'. This is a way of *breaking* the symmetries due to permuting machines. Formally, we say that a configuration C is *lexicographically* larger than a configuration C' if there exists $q \in [s]$ such that $m(\ell, C) = m(\ell, C')$ for all $\ell < q$ and $m(q, C) > m(q, C')$. We denote this by $C >_{\text{lex}} C'$. In particular, the relation $>_{\text{lex}}$ defines a total order over $\mathcal{C}$.

Integer linear program Let $B := 1 + 2s \max_{q \in [s]} |J_q| = O(|J|^2)$. We define an integer linear program $\text{assign}(B, T)$ below in which we enforce that the machines are ordered according to the relation $>_{\text{lex}}$.

$$\sum_{i \in [m]} x_{ij} = 1 \quad \text{for all } j \in J, \tag{25}$$

$$\sum_{j \in J} x_{ij} p_j \leq T \quad \text{for all } i \in [m], \tag{26}$$

$$\sum_{q=1}^{s} B^{s-q} \sum_{j \in J_q} \left(x_{ij} - x_{(i+1)j}\right) \geq 0 \quad \text{for all } i \in [m-1], \tag{27}$$

$$x_{ij} \geq 0 \quad \text{for all } i \in [m], \text{ for all } j \in J. \tag{28}$$

To avoid confusion we sometimes use the notation $\mathrm{assign}(J, B, T)$ to emphasize that we are considering the program for the job set J. Given a subset of jobs $K \subseteq J$ such that $\sum_{j \in K} p_j \leq T$, we denote by $\mathrm{conf}(K)$ the configuration such that for every $q \in \{1, \ldots, s\}$, $m(q, \mathrm{conf}(K)) = |K \cap J_q|$. We then say that $\mathrm{conf}(K)$ is the *configuration induced by* K. In the following we show that every integer solution to the program $\mathrm{assign}(B, T)$ obeys the lexicographic order $>_{\mathrm{lex}}$ on the configurations over the machines. More specifically, given a feasible integer solution $x \in \mathrm{assign}(T)$ and a machine $i \in [m]$, let $\mathrm{conf}_i(x) \in \mathcal{C}$ be the configuration defined by the job assignment of x to machine i, that is, for every $q \in \{1, \ldots, s\}$, $m(q, \mathrm{conf}_i(x)) = \sum_{j \in J_q} x_{ij}$.

Theorem 7 *In every integer solution $x \in assign(B, T)$, for every machine $i \in [m-1]$ we have that $conf_i(x) \geq_{lex} conf_{i+1}(x)$.*

To prove Theorem 7, we define $\mathcal{L}_B : \mathcal{C} \to \mathbb{R}$ to be the function such that for every configuration $C \in \mathcal{C}$, $\mathcal{L}_B(C) = \sum_{q=1}^{s} B^{s-q} m(q, C)$. The important point is that $\mathcal{L}_B$ is strictly increasing.

Lemma 9 *For two configurations $C, C' \in \mathcal{C}$ with $C <_{lex} C'$ we have that $\mathcal{L}_B(C) < \mathcal{L}_B(C')$.*

Proof Consider two configurations $C, C' \in \mathcal{C}$ such that $C <_{\mathrm{lex}} C'$. Let $\tilde{q}$ be the smallest integer in $\{1, \ldots, s\}$ such that the multiplicities of the configurations are different, that is, $m(\ell, C) = m(\ell, C')$ for every $\ell < \tilde{q}$. Hence it holds that $m(\tilde{q}, C) < m(\tilde{q}, C')$. In particular, every term up to $\max\{0, \tilde{q} - 1\}$ in the summation defining $\mathcal{L}_B(C) - \mathcal{L}_B(C')$ is equal to zero. By upper bounding the summation from $\min\{s, \tilde{q}+1\}$ we get that $\sum_{q=\min\{s,\tilde{q}+1\}}^{s} B^{s-q} \left(m(q, C) - m(q, C')\right)$ is at most

$$\begin{aligned} &\sum_{q=\min\{s,\tilde{q}+1\}}^{s} B^{s-q} \left(|m(q, C)| + |m(q, C')|\right) \\ &\quad \leq \sum_{q=\min\{s,\tilde{q}+1\}}^{s} B^{s-q} \cdot 2|J_q| < B^* \cdot B^{s-\tilde{q}-1} < B^{s-\tilde{q}}, \end{aligned}$$

and since $m(\tilde{q}, C') - m(\tilde{q}, C) \geq 1$ it follows that

$$\begin{aligned} \sum_{q=\tilde{q}}^{s} B^{s-q} \left(m(q, C) - m(q, C')\right) &< B^{s-\tilde{q}} \left(m(\tilde{q}, C) - m(\tilde{q}, C')\right) + B^{s-\tilde{q}} \\ &< B^{s-\tilde{q}} \left(m(\tilde{q}, C) - m(\tilde{q}, C') + 1\right) < 0 \end{aligned}$$

and hence $\mathcal{L}_B(C) < \mathcal{L}_B(C')$. ◻

Proof of Theorem 7 Fix a machine $i \in [m-1]$. Since x is an integral solution in $\mathrm{assign}(B, T)$, we have that $\mathrm{conf}_i(x), \mathrm{conf}_{i+1}(x) \in \mathcal{C}$. The symmetry breaking constraints implies that

$$\begin{aligned} 0 &\le \sum_{q=1}^{s} B^{s-q} \left(m(q, \mathrm{conf}_i(x)) - m(q, \mathrm{conf}_{i+1}(x))\right) \\ &= \mathcal{L}_B(\mathrm{conf}_i(x)) - \mathcal{L}_B(\mathrm{conf}_{i+1}(x)). \end{aligned}$$

Applying Lemma 9 it holds that $\mathcal{L}_B$ is strictly increasing and therefore we conclude that $\mathrm{conf}_i(x) \ge_{\mathrm{lex}} \mathrm{conf}_{i+1}(x)$. □

In general, $\mathrm{assign}(B, T)$ is *not* S_m-invariant, that is, given a solution to $\mathrm{assign}(B, T)$, if we permute the machines then we do not necessarily obtain another solution for it. However, it is a valid formulation, in the sense that if there exists a schedule with makespan at most T, then $\mathrm{assign}(B, T)$ has a feasible integral solution (more precisely, we retain a representative solution for each orbit). To show this, we can take an arbitrary schedule of makespan T and reorder the machines lexicographically according to their configurations.

Lemma 10 *If there exists an integral feasible solution to assign(T) then there exists also an integral feasible solution to assign(B, T).*

Proof Since there exists a schedule of makespan at most T, there exists an integral solution $x \in \mathrm{assign}(T)$. Since the lexicographic relation defines a total order over $\mathcal{C}$, there exists a permutation $\sigma \in S_m$ such that for every $i \in [m-1]$, $\mathrm{conf}_{\sigma(i)}(x) \ge_{\mathrm{lex}} \mathrm{conf}_{\sigma(i+1)}(x)$. Consider the integral solution $\tilde{x}$ obtained by permuting the solution according to σ, that is, $\tilde{x} = \sigma x$. Then, for every $i \in [m-1]$ it follows that $\sum_{q=1}^{s} B^{s-q} \sum_{j \in J_q} \left(\tilde{x}_{ij} - \tilde{x}_{(i+1)j}\right)$ is equal to

$$\begin{aligned} &\sum_{q=1}^{s} B^{s-q} \left(m(q, \mathrm{conf}_{\sigma(i)}(x)) - m(q, \mathrm{conf}_{\sigma(i+1)}(x))\right) \\ &= \mathcal{L}_B(\mathrm{conf}_{\sigma(i)}(x)) - \mathcal{L}_B(\mathrm{conf}_{\sigma(i+1)}(x)) \ge 0. \end{aligned}$$

The last step holds by Lemma 9. We conclude that $\tilde{x} \in \mathrm{assign}(B, T)$. □

4.2 LP based approximation scheme

In this section we prove Theorem 2, i.e., we show that if we apply $2^{\tilde{O}(1/\varepsilon^2)}$ rounds of the Sherali-Adams hierarchy to $\mathrm{assign}(B, T)$ then the integrality gap of the resulting LP is at most $1+\epsilon$, i.e., if it has a feasible solution then there exists an integral solution with makespan at most $(1+\varepsilon)T$. Recall the definition of a SA pseudoexpectation at the end of Sect. 2; in particular, recall that if a degree-r SA pseudoexpectation exists for a linear program, then it has a solution after applying r rounds of SA to it. The main result of this section is the following theorem.

Theorem 8 *Consider a value $T > 0$ and suppose there exists a degree-$(1/\varepsilon)^{2/\varepsilon^2}$ SA pseudoexpectation for* $\mathrm{assign}(B, T)$*. Then, there exists an integral solution in* $\mathrm{assign}(B, (1+\varepsilon)T)$ *and it can be computed in polynomial time.*

In what follows we might omit SA when referring to pseudoexpectations since the context is clear. Given a degree-r pseudoexpectation $\widetilde{\mathbb{E}}$ and a subset $A \subseteq [m] \times J$ with $\widetilde{\mathbb{E}}(x_A) \neq 0$, we define the *A-conditioning* to be the linear operator over $\mathbb{R}[x]/\mathbf{I}_E$ defined by $\widetilde{\mathbb{E}}_A(x_I) = \widetilde{\mathbb{E}}(x_I x_A)/\widetilde{\mathbb{E}}(x_A)$, for every $I \subseteq [m] \times J$. We also say that *we condition on A*. The following lemma summarises some of the relevant properties of the conditionings. We refer to [33] for a proof of it as well as a detailed exposition of the SA hierarchy.

Lemma 11 *Let $\widetilde{\mathbb{E}}$ be a degree-r pseudoexpectation and let $\widetilde{\mathbb{E}}_A$ be the conditioning for some $A \subseteq [m] \times J$ of cardinality at most r. Then*

(*a*) *$\widetilde{\mathbb{E}}_A(x_A) = 1$ and $\widetilde{\mathbb{E}}_A(x_{ij}) = 1$ for every $(i, j) \in A$.*
(*b*) *$\widetilde{\mathbb{E}}_A$ is a degree-$(r - |A|)$ pseudoexpectation.*
(*c*) *For every $B \subseteq [m] \times J$ such that $\widetilde{\mathbb{E}}(x_B) \in \{0, 1\}$ we have $\widetilde{\mathbb{E}}(x_B) = \widetilde{\mathbb{E}}_A(x_B)$.*

Stability In the following consider a degree-r pseudoexpectation $\widetilde{\mathbb{E}}$ for the program $\mathrm{assign}(J_{\mathrm{long}}, B^*, T)$, and let $\{J_1, \ldots, J_s\}$ be the partitioning of J_{long} defined above. Recall that $s = (1-\varepsilon)/\varepsilon^2$. Our strategy is to find a set $A \subseteq [m] \times J$ with $|A| \leq 2^{\tilde{O}(1/\varepsilon^2)}$ such that in $\widetilde{\mathbb{E}}_A$ for each machine i and for each set $J_q \in \{J_1, \ldots, J_s\}$ an *integral* number of jobs from J_q are assigned to i. Since in each set J_q the jobs have essentially the same length, based on $\widetilde{\mathbb{E}}_A$ we can compute an assignment of the long jobs to the machines of makespan at most $(1+\epsilon)T$. In order to find the set A, we will apply Lemma 11 several times. In the process we will achieve that for some machines i the number of jobs from some set J_q is integral and does not change if we condition on further elements from $[m] \times J$. Formally, we define that for some $q \in \{1, \ldots, s\}$ and $a \in \mathbb{N}$ a machine i is *(q, a)-stable in $\widetilde{\mathbb{E}}$* if we have $\sum_{j \in J_q} \widetilde{\mathbb{E}}(x_{ij}) = a$ and $\sum_{j \in J_q} \widetilde{\mathbb{E}}_A(x_{ij}) = a$ for any A-conditioning with $A \subseteq [m] \times J$. We say that a machine $i \in [m]$ is *q-stable in $\widetilde{\mathbb{E}}$* if it is (q, a)-stable for some $a \in \mathbb{Z}_+$. We will apply the following lemma over $\widetilde{\mathbb{E}}$ several times, until each machine i is q-stable for each $q \in \{1, ..., s\}$.

Lemma 12 *Consider $q \in \{1, \ldots, s\}$, integers $a_1, ..., a_q$, a degree-r pseudoexpectation $\widetilde{\mathbb{E}}$ and a set of consecutive machines $\{i_L, ..., i_R\}$, with $r \geq 1/\varepsilon^2$. Suppose that every machine $i \in \{i_L, \ldots, i_R\}$ is $(\tilde{q}, a_{\tilde{q}})$-stable in $\widetilde{\mathbb{E}}$ for each $\tilde{q} \in \{1, ..., q\}$. Then, there is a degree-$(r - 2/\varepsilon^2)$ pseudoexpectation $\widetilde{\mathbb{E}}^{stab}$ such that each machine $i \in \{i_L, \ldots, i_R\}$ is $\hat{q}$-stable in $\widetilde{\mathbb{E}}^{stab}$ for each $\hat{q} \in \{1, ..., q+1\}$.*

Phase 1: Obtaining a good pseudoexpectation We first use Lemma 12 in order to prove Theorem 8. We prove Lemma 12 later in Sect. 4.2.1. In what follows let $\widetilde{\mathbb{E}}$ be a degree-$(1/\varepsilon)^{2/\varepsilon^2}$ pseudoexpectation for $\mathrm{assign}(B, T)$. Our algorithm works in *s stages*. After stage q we obtain a pseudoexpectation in which each machine is $\tilde{q}$-stable for each $\tilde{q} \in \{1, \ldots, q\}$. In the first stage we apply Lemma 12 on the solution $\widetilde{\mathbb{E}}$ with $i_L = 1$, $i_R = m$ and $q = 0$, and let $\widetilde{\mathbb{E}}^{(0)}$ be the pseudoexpectation obtained. Assume by induction that after stage q we have obtained a pseudoexpectation $\widetilde{\mathbb{E}}^{(q)}$ in which each machine is $\tilde{q}$-stable for $\tilde{q} \in \{1, \ldots, q\}$. Consider a partition of $[m]$ given by $\{M_1, \ldots, M_k\}$ of the machines such that in $\widetilde{\mathbb{E}}^{(q)}$, for each set M_ℓ with $\ell \in \{1, \ldots, k\}$, there are integers $a_{\ell,1}, \ldots, a_{\ell,q}$ such that each machine in M_ℓ is $(\tilde{q}, a_{\ell,\tilde{q}})$-stable for each $\tilde{q} \in \{1, \ldots, q\}$. Since $\widetilde{\mathbb{E}}^{(q)}$ is a pseudoexpectation for assign(B, T), which includes the symmetry breaking constraints, the machines in each set M_ℓ are consecutive. Since the possible number of combinations $a_{\ell,1}, \ldots, a_{\ell,q}$ is at most $(1/\varepsilon + 1)^q$ we can find such a partition with $k \leq (1/\varepsilon + 1)^q$. For each $\ell \in \{1, \ldots, k\}$ we apply Lemma 12. Hence, the total number of rounds in this stage is at most $(1/\varepsilon + 1)^q \cdot 2/\varepsilon^2$. Denote by $\widetilde{\mathbb{E}}^{(q+1)}$ the obtained solution. We continue for s stages. Let $\widetilde{\mathbb{E}}^{\mathrm{f}}$ be the pseudoexpectation returned by the algorithm. The degree of $\widetilde{\mathbb{E}}^{\mathrm{f}}$ is at least $(1/\varepsilon)^{2/\varepsilon^2} - \sum_{q=1}^{s}(1/\varepsilon + 1)^q \cdot 2/\varepsilon^2 \geq 0$. So in particular, during the process we can indeed apply Lemma 12 as needed. We showed that the following holds.

Proposition 3 *For every $\tilde{q} \in \{1, \ldots, s\}$, every machine is $\tilde{q}$-stable in $\widetilde{\mathbb{E}}^{\mathrm{f}}$.*

Phase 2: Integral assignment for J_{long} Based on $\widetilde{\mathbb{E}}^f$ we define an integral assignment of the long jobs. Note that for each machine i and each value $q \in [s]$, machine i is q-stable; we define $b_{iq} := \sum_{j \in J_q} \widetilde{\mathbb{E}}^f(x_{ij})$ which is the number of jobs of J_q that are assigned to i by $\widetilde{\mathbb{E}}^f$. Since $\widetilde{\mathbb{E}}^f$ yields a valid solution to assign(B, T), for each $q \in [s]$ we have that $\sum_{i \in [m]} b_{iq} = |J_q|$. For each $q \in [s]$ we assign now the jobs in J_q to the machines such that each machine i receives exactly b_{iq} jobs from J_q. Intuitively, all jobs in J_q have essentially the same length (up to a factor $1 + \epsilon$), and therefore it is not relevant which exact jobs from J_q we assign to i, as long as we assign b_{iq} jobs in total. Afterwards, we the short jobs in a standard greedy list scheduling procedure: We consider the jobs in an arbitrary order and assign each job on a machine that currently has the minimum load among all machines. Now we are ready to prove Theorem 8 by showing that the load of every machine is at most $(1 + \varepsilon)T$.

Theorem 8 Let $\{\bar{x}_{i,j}\}_{i \in [m], j \in J}$ denote the computed integral assignment of the jobs to the machines, i.e., $\bar{x}_{i,j} = 1$ if we assigned job j on machine i and $\bar{x}_{i,j} = 0$ otherwise. We first check that for each machine $i \in [m]$, we have that $\sum_{q=1}^{s} \sum_{j \in J_q} \bar{x}_{i,j} p_j \leq (1 + \varepsilon)T$. Since the solution given by $\widetilde{\mathbb{E}}^f$ feasible for assign(B, T), for each machine i we have that $\sum_{q=1}^{s} b_{iq} \left(\frac{1}{\varepsilon} + q - 1\right) \varepsilon^2 T \leq T$. This implies for each machine i that

$$
\begin{aligned}
\sum_{q=1}^{s} \sum_{j \in J_q} \bar{x}_{i,j} p_j &\leq (1 + \varepsilon) \sum_{q=1}^{s} \sum_{j \in J_q} \bar{x}_{i,j} \left(\frac{1}{\varepsilon} + q - 1\right) \varepsilon^2 T \\
&\leq (1 + \varepsilon) \sum_{q=1}^{s} \left(\frac{1}{\varepsilon} + q - 1\right) \varepsilon^2 T \sum_{j \in J_q} \bar{x}_{i,j} \\
&\leq (1 + \varepsilon) \sum_{q=1}^{s} \left(\frac{1}{\varepsilon} + q - 1\right) \varepsilon^2 T \cdot b_{iq} \leq (1 + \varepsilon)T.
\end{aligned}
$$

It remains to argue about the short jobs. If the global makespan does not increase while assigning them greedily, the overall makespan remains at most $(1 + \varepsilon)T$. Otherwise, the makespan of any two machines differ by at most εT. Since $\sum_j p_j \leq mT$ we conclude that the makespan is at most $(1+\varepsilon)T$. □

4.2.1 Stable conditionings: Proof of Lemma 12

Recall that $\widetilde{\mathbb{E}}$ is a degree-r pseudoexpectation with $r \geq 1/\varepsilon^2$. We use the following strategy to prove Lemma 12. First, we identify the rightmost machine i such that according to $\widetilde{\mathbb{E}}$ with non-zero probability there are $1/\epsilon$ jobs from J_{q+1} assigned to i. Let i_0 be this machine and let $A \subseteq [m] \times J$ denote the corresponding pairs (i, j) with $i = i_0$ and $j \in J_{q+1}$. We apply Lemma 11 on A. We argue that in the resulting pseudo-expectation $\widetilde{\mathbb{E}}_A$ with non-zero probability there are $1/\epsilon$ jobs from J_{q+1} assigned to i_L, let $A' \subseteq [m] \times J$ denote the corresponding pairs. We apply Lemma 11 on A' as well. In the resulting pseudoexpectation $\widetilde{\mathbb{E}}_{A\cup A'}$, the symmetry breaking constraints in assign(B, T) ensure that *each* machine i' between i_L and i_0 has *exactly* $1/\epsilon$ jobs from J_{q+1} assigned to i and therefore i' is $(q+1)$-stable. Also, no machine between i_0 and i_R will ever get $1/\epsilon$ jobs from J_{q+1} assigned to it with non-zero probability, no matter on which sets A'' we might condition later. We continue inductively: on the machines between i_0 and i_R we look for the rightmost machine i such that according to $\widetilde{\mathbb{E}}_{A\cup A'}$ with some non-zero probability there are $1/\epsilon - 1$ jobs from J_{q+1} assigned to i, etc. There are at most $1/\epsilon$ iterations in total and in each step the degree of the pseudo-expectation decreases by at most $2/\epsilon$. Therefore, at the end, we obtain a degree-$(r - 2/\varepsilon^2)$ pseudo-expectation in which all machines are $(q+1)$-stable.

Now we describe our argumentation in detail. First assume that there is no machine $i \in \{i_L, ..., i_R\}$ for which there exists a set $A \subseteq [m] \times J$ with $\widetilde{\mathbb{E}}(x_A) > 0$, $|A| = 1/\varepsilon$, and where each tuple $(h, \upsilon) \in A$ satisfies that $h = i$ and $\upsilon \in J_{q+1}$. In this case we define $\widetilde{\mathbb{E}}^{(0)} = \widetilde{\mathbb{E}}$ and $i_0 = i_L - 1$. Intuitively, in this case in our final assignment there will be no machine in $\{i_L, ..., i_R\}$ that has $1/\varepsilon$ jobs from J_{q+1} assigned to it. We will use later in our induction that $\widetilde{\mathbb{E}}^{(0)}$ is a degree-$(r - 2/\varepsilon)$ pseudoexpectation. Otherwise let i_0 be the rightmost machine in $\{i_L, ..., i_R\}$, i.e., the machine with largest index, satisfying the above for some set A. We condition on A and obtain the degree-$(r - 1/\varepsilon)$ SA conditioning $\widetilde{\mathbb{E}}_A$. Recall that by Lemma 11(a) each job $\upsilon \in J_{\text{long}}$ with $(i_0, \upsilon) \in A$ is scheduled integrally to i_0.

Lemma 13 *There exists a set $A' \subseteq [m] \times J$ with $\widetilde{\mathbb{E}}_A(x_{A'}) > 0$, $|A'| = 1/\varepsilon$, and for every $(h, \upsilon) \in A'$ we have that $h = i_L$ and $\upsilon \in J_{q+1}$.*

Proof Assume that this is not the case. Then let A' denote the set of maximum size such that $\widetilde{\mathbb{E}}_A(x_{A'}) > 0$ and such that each $(h, \upsilon) \in A'$ satisfies that $h = i_L$ and $\upsilon \in J_{q+1}$. Observe that $|A'| \leq 1/\varepsilon$ by Lemma 11(b), and let $\widetilde{\mathbb{E}}_{A\cup A'}$ be the degree-$(r - 2/\varepsilon)$ SA conditioning. Then, for each job $j \in J_q$ it holds that $\widetilde{\mathbb{E}}_{A\cup A'}(x_{i_L j}) = 0$, otherwise $0 < \widetilde{\mathbb{E}}_{A\cup A'}(x_{i_L j}) = \widetilde{\mathbb{E}}_A(x_{A'} x_{i_L j}) / \widetilde{\mathbb{E}}_A(x_{A'})$ and then $\widetilde{\mathbb{E}}_A(x_{A'} x_{i_L j}) = \widetilde{\mathbb{E}}_A(x_{A' \cup \{(i_L, j)\}}) > 0$, which contradicts the maximality of A'. But then the fractional schedule given by $\widetilde{\mathbb{E}}_{A'}(x_{ij})$ for every $(i, j) \in [m] \times J_{\text{long}}$ violates the symmetry breaking constraints of assign(J_{long}, B, T), which is a contradiction. □

Starting from $\widetilde{\mathbb{E}}_A$ we condition on A' (i.e., we apply Lemma 11) given by Lemma 13, obtaining $\widetilde{\mathbb{E}}^{(0)} = \widetilde{\mathbb{E}}_{A\cup A'}$. As a result, machine i_L and machine i_0 have both exactly $1/\varepsilon$

jobs from J_q assigned to it. Due to the symmetry breaking constraints of the program $\mathrm{assign}(J_{\mathrm{long}}, B, T)$ this implies that each machine in $\{i_L, ..., i_0\}$ has exactly $1/\varepsilon$ jobs from J_q (fractionally) assigned to it in $\widetilde{\mathbb{E}}^{(0)}$.

Lemma 14 *For each machine $i \in \{i_L, ..., i_0\}$ we have $\sum_{j \in J_q} \widetilde{\mathbb{E}}^{(0)}(x_{ij}) = 1/\varepsilon$.*

Proof Machine i_0 has $1/\varepsilon$ jobs from J_q assigned to it integrally. If there was yet another job $j \in J_q$ fractionally assigned to i_0 then we could condition on (i_0, j) and obtain a degree-$(r - 1/\varepsilon - 1)$ pseudoexpectation, with at least $1/\varepsilon + 1$ long jobs assigned to i_0, which is a contradiction. The same argument holds for machine i_L. The claim for the machines $\{i_L + 1, ..., i_0 - 1\}$ follows by the symmetry breaking constraints in $\mathrm{assign}(J_{\mathrm{long}}, B, T)$ that enforce the lexicographic ordering over the machines. □

Proof of Lemma 12 Assume by induction that for some $k \in \{0, ..., 1/\varepsilon - 1\}$ we obtained a degree-$(r - \sum_{\ell=1}^{k} 2(1/\varepsilon + 1 - \ell))$ pseudoexpectation $\widetilde{\mathbb{E}}^{(k)}$ such that there are machines $i_0, ..., i_k \in [m]$ such that for each $\ell \in \{0, ..., k\}$ we have that $i_\ell \le i_{\ell+1}$ and each machine $w \in \{i_{\ell-1} + 1, ..., i_\ell\}$ satisfies $\sum_{j \in J_q} \widetilde{\mathbb{E}}^{(k)}(x_{wj}) = 1/\varepsilon - \ell$, with $i_{-1} = i_L - 1$ for convenience. Moreover, assume by induction that there is no set $\Gamma \subseteq [m] \times J_{\mathrm{long}}$ with $\widetilde{\mathbb{E}}^{(k)}(x_\Gamma) > 0$ and $|\Gamma| = 1/\varepsilon - k$ such that each $(w, j) \in \Gamma$ satisfies that $w = i_k + 1$ and $j \in J_q$. The solution $\widetilde{\mathbb{E}}^{(0)}$ constructed above satisfies the base case of $k = 0$.

Inductive step. Given the solution $\widetilde{\mathbb{E}}^{(k)}$ we construct a solution $\widetilde{\mathbb{E}}^{(k+1)}$ as follows. Let i_{k+1} denote the rightmost machine larger than i_k such that there is a set $A_k \subseteq [m] \times J_{\mathrm{long}}$ with $\widetilde{\mathbb{E}}^{(k)}(x_{A_k}) > 0$, $|A_k| = 1/\varepsilon - k - 1$, and each tuple $(w, \upsilon) \in A_k$ satisfies that $w = i_{k+1}$ and $\upsilon \in J_q$. If there is no such machine then we define $i_{k+1} = i_k$ and set $\widetilde{\mathbb{E}}^{(k+1)} = \widetilde{\mathbb{E}}^{(k)}$ which is a pseudoexpectation of degree $r - \sum_{\ell=1}^{k+1} 2(1/\varepsilon + 1 - \ell)$. Otherwise we condition on A_k and obtain $\widetilde{\mathbb{E}}^{(k)}_{A_k}$, of degree $r - \sum_{\ell=1}^{k} 2(1/\varepsilon + 1 - \ell)) - (1/\varepsilon - k - 1)$. Following the same lines of Lemma 13 we have that there exists a set $B_k \subseteq [m] \times J_{\mathrm{long}}$ with $\widetilde{\mathbb{E}}^{(k)}_{A_k}(x_{B_k}) > 0$, $|B_k| = 1/\varepsilon - k - 1$, and each tuple $(h, \upsilon) \in B_k$ satisfies that $h = i_k + 1$ and $\upsilon \in J_q$. We then define $\widetilde{\mathbb{E}}^{(k+1)} = \widetilde{\mathbb{E}}^{(k)}_{A_k \cup B_k}$, which is a pseudoexpectation of degree $r - \sum_{\ell=1}^{k+1} 2(1/\varepsilon + 1 - \ell)$.

Claim 3 *For each $w \in \{i_k + 1, ..., i_{k+1}\}$ we have that $\sum_{j \in J_q} \widetilde{\mathbb{E}}^{(k+1)}(x_{wj}) = 1/\varepsilon - k - 1$.*

We see how to conclude the lemma and then we show check the claim. Since we chose machine i_{k+1} to be the rightmost machine with the claimed properties, $\widetilde{\mathbb{E}}^{(k+1)}$ satisfies the induction hypothesis for $k + 1$. Finally, we define $\widetilde{\mathbb{E}}^{\mathsf{stab}} = \widetilde{\mathbb{E}}^{(1/\varepsilon)}$, which yields that $\widetilde{\mathbb{E}}^{\mathsf{stab}}$ is a pseudoexpectation of degree $r - 2/\varepsilon^2$, since $\sum_{\ell=1}^{1/\varepsilon} 2(1/\varepsilon + 1 - \ell) \le 2\,/\varepsilon^2$. That concludes the lemma.

Proof of Claim 3 On machine $w = i_k + 1$ we conditioned on the set B_k due to the previous claim with $|B_k| = 1/\varepsilon - k - 1$. Therefore $\sum_{j \in J_q} \widetilde{\mathbb{E}}^{(k+1)}(x_{wj}) \ge 1/\varepsilon - k - 1$. On the other hand, if $\sum_{j \in J_q} \widetilde{\mathbb{E}}^{(k+1)}(x_{wj}) > 1/\varepsilon - k - 1$ then there must be a pair $(w, j) \notin B_k$ with $\widetilde{\mathbb{E}}^{(k+1)}(x_{wj}) > 0$ and $j \in J_q$. But this implies that $\widetilde{\mathbb{E}}^{(k)}(x_{B_k \cup \{(w,j)\}}) > 0$ which contradicts the induction hypothesis. With the same reasoning we argue that $\sum_{j\in J_q} \widetilde{\mathbb{E}}^{(k+1)}(x_{i_{k+1}j}) = 1/\varepsilon - k - 1$. The claim for the machines in $\{i_k+2, ..., i_{k+1}-1\}$ then follows from the symmetry breaking constraints in assign(J_{long}, B, T) that enforce the lexicographic over the machines. ◻

4.3 A faster LP based approximation scheme

In Sect. 4.2 we proved that after applying $(1/\varepsilon)^{2/\varepsilon^2}$ rounds of Sherali-Adams to assign(B, T) we obtain a linear relaxation with an integrality gap of at most $1+\varepsilon$. In this section, we add to assign(B, T) a set of constraints that we refer to as the *ordering constraints*, obtaining a linear program that we refer to as order(B, T). Intutitively, we prove that if we apply only poly(1/") rounds of Sherali-Adams to this new program then its integrality gap drops to $1+\varepsilon$. On the other hand, it might be that there is no optimal solution (i.e., a solution with makespan OPT) that satisfies the ordering constraints and in particular it might be that order(B, OPT) does not have a feasible solution (in contrast to assign(B, OPT) which is always feasible). However, we can guarantee that order$(B, (1+\varepsilon)\text{OPT})$ is always feasible.

Ordering constraints Roughly speaking, we use a new set of constraints that allow us to break symmetries due to permutations of jobs in the same class J_q (and not only the symmetries corresponding to permutations of machines), which is a key difference to the approach used for the approximation scheme of Sect. 4.2. For each $q \in \{1, \dots, s\}$ assume that $J_q = \{j_{q,1}, j_{q,2}, ..., j_{q,|J_q|}\}$ and we impose that the jobs in J_q are scheduled in this order, i.e., if jobs $j_{q,\ell}$ and $j_{q,\ell+1}$ are scheduled on machines $i \in [m]$ and $h \in [m]$ for some $\ell \in \{1, \dots, |J_q|-1\}$, then $i \le h$. To enforce this, for each $q \in \{1, \dots, s\}$, each $\ell \in \{1, ..., |J_q|-1\}$, and for each $h \in [m]$ we add to assign(B, T) the constraint

$$\sum_{i=1}^{h} x_{ij_{q,\ell}} \ge \sum_{i=1}^{h} x_{ij_{q,\ell+1}}. \tag{29}$$

Denote by order(B, T) the LP obtained by adding the above set of constraints to assign(B, T). It might be that there is no feasible solution to order(B, OPT). However, in the following lemma we show that there exists always a solution to order$(B, (1+\varepsilon)\text{OPT})$.

Lemma 15 *There exists a feasible integral solution to* order$(B, (1+\varepsilon)\text{OPT})$.

Proof Consider the integral vector x to assign(B, OPT) that stems from the optimal solution to the given instance. For each machine $i \in [m]$ denote by conf$_i(x)$ its vector $(a_{i,1}, \dots, a_{i,s})$ as defined in Sect. 4.2. For each $q \in \{1, \dots, s\}$ we rearrange the jobs in J_q on the machines such that the resulting schedule satisfies the order constraints and on each machine the number of jobs from each set J_q stays the same. Given $q \in \{1, \dots, s\}$, for each machine $i \in [m]$ let b_{iq} denote the number of jobs from J_q on $i \in [m]$ in a schedule with optimal makespan OPT. In our new schedule, for each machine $i \in [m]$ we define $c_{iq} = \sum_{h=1}^{i} b_{hq}$ and we set $c_{0q} = 0$. Then we assign to each machine $i \in [m]$ the jobs $\{j_{q,c_{i-1,q}+1}, ..., j_{q,c_{iq}}\}$. We repeat this operation for every $q \in \{1, \ldots, s\}$. Within each set J_q the processing times of two jobs can differ by at most ε^2OPT. Each machine has at most $1/\varepsilon$ long jobs assigned to it. Therefore, due to our reassignment of jobs the makespan of the schedule can increase by at most $\frac{1}{\varepsilon} \cdot \varepsilon^2\text{OPT} = \varepsilon\text{OPT}$ on each machine. This integral schedule yields a solution to order$(B, (1+\varepsilon)\text{OPT})$. □

In the remainder of this section we prove the following theorem.

Theorem 9 *Consider a value $T > 0$ and suppose there exists a degree $4/\varepsilon^5$ SA pseudoexpectation for* order(B, T). *Then, there exists an integral solution for* order$(B, (1+\varepsilon)T)$ *and it can be computed in polynomial time.*

As before we first construct a solution for the long jobs only and afterwards argue that we can add the short jobs with only marginal increase of the makespan. For a degree-r pseudoexpectation $\widetilde{\mathbb{E}}$ and a set of machines M^*, we say that M^* is *focused* if each job $j \in J_{\text{long}}$ is either completely assigned to machines in M^*, i.e., $\sum_{i \in M^*} \widetilde{\mathbb{E}}(x_{ij}) = 1$, or to no machine in M^*, i.e., $\sum_{i \in M^*} \widetilde{\mathbb{E}}(x_{ij}) = 0$ and the same holds for any conditioning obtained from $\widetilde{\mathbb{E}}$.

Overview We first apply Lemma 12 to make sure that each machine is 1-stable. This partitions the machines into sets $\{M_0, \ldots, M_{1/\epsilon}\}$ such that the machines in each set M_ℓ have exactly ℓ jobs from J_1. The machines in each set M_ℓ are consecutive. Then, intuitively, for each set M_ℓ we take the rightmost machine i and condition on every single long job on i (so not just on the jobs in J_1). As a result, due to the ordering constraints each set M_ℓ is focused. This operation is formalized in Lemma 16. Then, we observe that for each set of machines M_ℓ we obtain a degree-$(r - 4/\varepsilon^3)$ pseudoexpectation for order$(J_{\text{long},\ell}, B, T, M_\ell)$ for some set $J_{\text{long},\ell} \subseteq J_{\text{long}}$ that is completely independent of all other sets $M_{\ell'}$ with $\ell' \neq \ell$. Therefore, we can recurse on each set of machines M_ℓ *independently* such that the degree of our pseudoexpectation drops by at most $4/\varepsilon^3$ in each level. Since there are only $1/\epsilon^2$ levels, it suffices to start with a pseudoexpectation of degree at most $4/\epsilon^5$.

Lemma 16 *Consider $q \in \{0, 1, \ldots, s\}$, integers $a_1, \ldots, a_q$ and a degree-r pseudoexpectation $\widetilde{\mathbb{E}}$ such that each machine $i \in [m]$ is $(\tilde{q}, a_{\tilde{q}})$-stable for each $\tilde{q} \in \{1, \ldots, q\}$. Then there is a degree $r - 4/\varepsilon^3$ pseudoexpectation $\widetilde{\mathbb{E}}^{\text{focus}}$ obtained from $\widetilde{\mathbb{E}}$ via conditioning on at most $4/\varepsilon^3$ variables such that each machine in $[m]$ is $\hat{q}$-stable for each $\hat{q} \in \{1, \ldots, q+1\}$ and there is a partioning of $[m]$ given by $\{M_1, \ldots, M_k\}$ of consecutive machines such that for each $\ell \in \{1, \ldots, k\}$ we have that M_ℓ is focused and each machine $i \in M_\ell$ is $(\tilde{q}+1, \ell)$-stable.*

Proof First, we apply Lemma 12 with the same value q, the integers $a_1, \ldots, a_q$, and with $M^* = [m]$. Let $\widetilde{\mathbb{E}}^{\text{stab}}$ the degree $(r - 2/\varepsilon^2)$ pseudoexpectation obtained. Since $\widetilde{\mathbb{E}}^{\text{stab}}$ is $(q+1)$-stable, we obtain a partition of $[m]$ given by $\{M_0, \ldots, M_k\}$ with $k \leq 1/\varepsilon$ such that each machine $i \in M_\ell$ is $(q+1, \ell)$-stable for each $\ell \in \{0, \ldots, k\}$. For $\ell \in \{0, \ldots, k\}$ we proceed as follows: For each $\tilde{q} \in \{1, \ldots, s\}$ let $\eta(\tilde{q}, \ell)$ be the largest index such that there is a job $j_{\tilde{q},\eta(\tilde{q},\ell)}$ with $\widetilde{\mathbb{E}}^{\text{stab}}(x_{ij_{\tilde{q},\eta(\tilde{q},\ell)}}) > 0$ for some machine $i \in M_\ell$. We condition on $(i, j_{\tilde{q},\ell(\tilde{q},\ell)})$. More precisely, we iterate over the values $\tilde{q} \in \{1, \ldots, s\}$

and condition on the respective jobs one by one, obtaining a pseudoexpectation $\widetilde{\mathbb{E}}^{\text{focus}}$. Since $2/\varepsilon^2 - s(1/\varepsilon + 1) \leq 4/\varepsilon^3$ this pseudoexpectation is of degree $r - 4/\varepsilon^3$.

We claim that in $\widetilde{\mathbb{E}}^{\text{focus}}$ each subset M_ℓ with $\ell \in \{0, \dots, k\}$ is focused. At the beginning the set $[m]$ is focused. At the first step $\ell = 1$, for each $\tilde{q} \in \{1, \dots, s\}$ either there is no job $j \in J_{\tilde{q}}$ fractionally assigned on a machine in M_1 or we conditioned on the job $j_{\tilde{q},\eta(\tilde{q},1)}$ with largest index $\eta(\tilde{q}, 1)$. Hence, due to the order constraints, no job $j_{\tilde{q},\eta'}$ with $\eta' > \eta(\tilde{q}, 1)$ can be fractionally assigned to a machine in M_1. Hence, for each $\tilde{q} \in \{1, \dots, s\}$ there is a set $\tilde{J}_{\tilde{q}} \subseteq J_{\tilde{q}}$ such that all jobs in $\tilde{J}_{\tilde{q}}$ are assigned on M_1 and no job in $J_{\tilde{q}} \setminus \tilde{J}_{\tilde{q}}$ is fractionally assigned. Hence, M_1 is focused and $[m] \setminus M_1$ is also focused. The remainder follows by induction with the same argument. □

Algorithm In the remaining fix $r = 4/\varepsilon^5$. We take a degree-r-pseudoexpectation for $\text{order}(B, T)$. We first apply Lemma 16 with $q = 0$ and obtain a solution $\widetilde{\mathbb{E}}^{\text{focus}}$. For each group M_ℓ with $\ell \in \{1, \dots, k\}$ denote by $J_{\text{long},\ell}$ the jobs from J_{long} assigned on M_ℓ in according to $\widetilde{\mathbb{E}}^{\text{focus}}$. Then, for each $\ell \in \{1, \dots, k\}$ this yields a pseudoexpectation $\widetilde{\mathbb{E}}_\ell^{\text{focus}}$ for the program $\text{order}(J_{\text{long},\ell}, B, T, M_\ell)$ of degree $r - 4/\varepsilon^3$, in which each machine is 1-stable. Intuitively, we continue recursively on each part. The depth of this recursion is s. In each level, we condition on at most $4/\varepsilon^3$ variables. Hence, if we obtain a pseudoexpectation of degree $s \cdot 4/\varepsilon^3 \leq 4/\varepsilon^5$ in $\text{order}(B, T)$ then we obtain that there exists a solution for $\text{order}(B, T)$ that is q-stable for each $q \in \{1, \dots, s\}$. Formally, we prove the following lemma by induction.

Lemma 17 *Consider $q \in \{0, 1, \dots, s\}$, integers $a_1, \dots, a_q$ and a degree $4(s - q)/\varepsilon^3$ pseudoexpectation such that each machine $i \in [m]$ is $(\tilde{q}, a_{\tilde{q}})$-stable for each $\tilde{q} \in \{1, \dots, q\}$. Then, there is a solution in* $\text{order}(B, T)$ *such that each machine $i \in [m]$ is $\hat{q}$-stable for each $\hat{q} \in \{1, \dots, s\}$.*

Proof We prove the lemma by induction. If $q = s$ then the lemma is trivially true. Now suppose that the lemma is true for some value $q + 1$. Given a pseudoexpectation corresponding to solution to $\text{order}(B, T)$ we apply Lemma 16 and obtain a solution $\widetilde{\mathbb{E}}^{\text{focus}}$ and the partition $\{M_1, \dots, M_k\}$. For each $\ell \in \{1, \dots, k\}$ this yields a pseudoexpectation $\widetilde{\mathbb{E}}_\ell^{\text{focus}}$ with degree $4(s - q - 1)/\varepsilon^3$ such that in $\widetilde{\mathbb{E}}_\ell^{\text{focus}}$ each machine $i \in M_\ell$ is $(q + 1, \ell)$-stable and also $\tilde{q}$-stable for each $\tilde{q} \in \{1, \dots, q\}$. On each pseudoexpectation $\widetilde{\mathbb{E}}_\ell^{\text{focus}}$ with $\ell \in \{1, \dots, k\}$ we apply the induction hypothesis and obtain a solution $x^\ell \in \text{order}(J_{\text{long},\ell}, B, T, M_\ell)$ such that in x^ℓ each machine $i \in [m]$ is $\hat{q}$-stable for each $\hat{q} \in \{1, \dots, s\}$. We define the solution to be the direct sum of the solutions x^ℓ over $\ell \in \{1, \dots, k\}$. □

The lemma above yields that if there exists degree $4/\varepsilon^5$ pseudoexpectation to $\text{order}(B, T)$ then there exists a solution $x \in \text{order}(B, T)$ in which each machine is $\hat{q}$-stable for each $\hat{q} \in \{1, \dots, s\}$. The assignment of the long jobs to the machines is identical to the proof of Lemma 12. Finally, we add the short jobs greedily like in Sect. 4.2. This completes the proof of Theorem 9.

Appendix A: Proof of Theorem 4

We show how to prove Theorem 4 following the lines in the work of Raymond et al. [45]. We need a few intermediate results, and the symmetry reduction theorem from Gaterman and Parrilo [13], stated in our setting.

Theorem 10 ([13]) *Suppose that $g \in \mathbb{R}[y]/\textbf{sched}$ is a degree-ℓ SoS and S_m-invariant polynomial. For each partition $\lambda \vdash m$, let τ_λ be a tableau of shape λ and let $\{b_1^\lambda, \ldots, b_{m_\lambda}^\lambda\}$ be a basis $\mathbf{W}_{\tau_\lambda}$. Then, for each partition $\lambda \vdash m$ there exists a $m_\lambda \times m_\lambda$ positive semidefinite matrix Q_λ such that $g = \sum_{\lambda \vdash m} \langle Q_\lambda, Y^\lambda \rangle$, where $Y_{ij}^\lambda = sym(b_i^\lambda b_j^\lambda)$.*

Given two partitions λ, μ, we say that $\lambda \trianglerighteq \mu$ if $\lambda \geq_{\text{lex}} \mu$ and the number of parts of μ is at least the number of parts of λ. The following lemma is a variant of [45, Theorem 2] for the action of the symmetric group in our setting. Together with the theorem of Gatermann and Parrilo this yields Theorem 4.

Lemma 18 *The dimension m_λ of $\mathbf{Q}_\lambda^\ell$ in the isotypic decomposition of $\mathbf{Q}^\ell$ is zero unless $\lambda \geq_{lex} (m-\ell, 1, \ldots, 1)$.*

Proof Let y_S be a monomial of degree at most ℓ with $S = \{(i_k, C_k) : k \in [\ell]\}$. In particular, $|\{i_k : k \in [\ell]\}| \leq \ell$. Let τ be any tableau with shape $(m-\ell, 1, \ldots, 1)$, where the tail of τ contains every elements of $\{i_k : k \in [\ell]\}$. The subgroup $\mathcal{R}_\tau$ fixes S, therefore $y_S \in \mathbf{W}_\tau^\ell$, and we have then

$$\mathbf{Q}^\ell \subseteq \bigoplus_{\tau:\text{shape}(\tau)=(m-\ell,1,\ldots,1)} \mathbf{W}_\tau^\ell \subseteq \bigoplus_{\lambda \trianglerighteq (m-\ell,1,\ldots,1)} \mathbf{Q}_\lambda^\ell,$$

where the second containment holds by [45, Lemma 1]. To conclude, observe that if $\lambda \trianglerighteq (m-\ell, 1, \ldots, 1)$ then $\lambda_1 \geq m - \ell$. Since $\lambda \vdash m$, the maximum number of parts for λ is $m - \lambda_1 \leq \ell$, that is, λ has at most $\ell + 1$ parts. Therefore, $\lambda \trianglerighteq (m-\ell, 1, \ldots, 1)$ if and only if $\lambda \geq_{\text{lex}} (m-\ell, 1, \ldots, 1)$. $\square$

Proof of Theorem 4 Let $g \in \mathbb{R}[y]/\textbf{sched}$ be a degree-ℓ SoS and S_m-invariant polynomial. By Theorem 10 and Lemma 18, for each $\lambda \in \Lambda_\ell$ there exists a positive semidefinite matrix Y^λ such that $g = \sum_{\lambda \in \Lambda_\ell} \langle Q_\lambda, Y^\lambda \rangle$. Since $\{b_1^\lambda, \ldots, b_{m_\lambda}^\lambda\} \subseteq \text{span}(\mathcal{P}^\lambda)$, there exists a real matrix $\mathcal{T}_\lambda$ such that $\mathcal{T}_\lambda(p_1^\lambda, \ldots, p_{\ell_\lambda}^\lambda) = (b_1^\lambda, \ldots, b_{m_\lambda}^\lambda)$. Consider the congruent transformation $M_\lambda = \mathcal{T}_\lambda^\top Q_\lambda \mathcal{T}_\lambda$. In particular, M_λ is also positive semidefinite. Furthermore, $\mathbf{b}^\top Q_\lambda \mathbf{b} = (\mathcal{T}_\lambda \mathbf{p})^\top Q_\lambda (\mathcal{T}_\lambda \mathbf{p}) = \mathbf{p}^\top M_\lambda \mathbf{p}$, where $\mathbf{b} = (b_1^\lambda, \ldots, b_{m_\lambda}^\lambda)$ and $\mathbf{p} = (p_1^\lambda, \ldots, p_{\ell_\lambda}^\lambda)$. That is, $g = \sum_{\lambda \in \Lambda_\ell} \langle Q_\lambda, Y^\lambda \rangle = \sum_{\lambda \in \Lambda_\ell} \langle M_\lambda, Z^\lambda \rangle$. $\square$

Appendix B: SoS Lower Bound for the Assignment Linear Program

We now show that the lower bound of Theorem 1 translates to the assignment linear program. Recall that the r-th level of the SoS hierarchy corresponds to a semidefinite program with variables y_S for any subset $S \subseteq E$ with $|S| \leq r$. The inequalities defining this program can be obtained by considering properties (SoS.1)–(SoS.4) in the definition of degree-r SoS pseudoexpectations and identifying $\widetilde{\mathbb{E}}(x_S) = y_S$; see for example [40] for details. For any polytope $P \subseteq [0,1]^E$, we denote by $\mathrm{SoS}_r(P)$ the projection of the r-th level of the SoS hierarchy over $y_i = y_{\{i\}}$ for each $i \in E$. Au and Tunçel [4, Proposition 1] showed that for any polytope $P \subseteq [0,1]^E$, if $L : \mathbb{R}^E \to \mathbb{R}^E$ is an affine transformation such that $L(x) \in [0,1]^E$ for all elements in the unit hypercube $x \in [0,1]^E$, then $\mathrm{SoS}_r(L(P)) = L(\mathrm{SoS}_r(P))$. In our case, we consider the configuration linear program and the assignment linear program within the same space. Let T be a target makespan and consider

$$P = \mathbb{R}^{[m]\times J} \times \mathrm{clp}(T) = \{(x,y) \in \mathbb{R}^{[m]\times J} \times \mathbb{R}^{[m]\times \mathcal{C}} : y \in \mathrm{clp}(T)\}.$$

We define the projection $L(x,y) = (x',0)$ where x' is defined as

$$x'_{ij} = \frac{1}{n_{p_j}} \sum_{C\in\mathcal{C}} m(C,p_j)\cdot y_{iC} \quad \text{for all } i \in [m] \text{ and for all } j \in J.$$

Notice that x' belongs to the assignment linear program, and hence $L(P) \subseteq \mathrm{assign}(T) \times [0,1]^{[m]\times\mathcal{C}}$ is within the unit hypercube and the result by Au and Tunçel can be applied. Therefore,

$$L(\mathrm{SoS}_{r+1}(P)) = \mathrm{SoS}_{r+1}(L(P)) \subseteq \mathrm{SoS}_r(\mathrm{assign}(T) \times [0,1]^{[m]\times\mathcal{C}}),$$

where the last inclusion follows since $L(P) \subseteq \mathrm{assign}(T) \times [0,1]^{[m]\times\mathcal{C}}$ and the general property of the next lemma. We remark that this is enough to get an integrality gap of 1.0009 for $\Omega(n)$ rounds of the SoS hierarchy applied to the assignment linear program.

Lemma 19 *If P and Q are two polytopes with $P \subseteq Q$, then $SoS_{r+1}(P) \subseteq SoS_r(Q)$.*

Proof Let us assume that $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ and $Q = \{x \in \mathbb{R}^n : Cx \leq d\}$ for some $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $C \in \mathbb{R}^{p\times n}$ and $d \in \mathbb{R}^p$. Let $a_i^\top$ be the i-th row of A and $c_i^\top$ the i-th row of C. We will show that a degree-$(r+1)$ SoS pseudoexpectation for P is also a degree-r SoS pseudoexpectation for Q. Indeed, recall that if $P \subseteq Q$, then every inequality $c_i^\top x \leq d_i$, where c_i is the i-th row of C, is a valid inequality for P. Hence, by Farkas lemma, for each row $i \in [p]$ there exists a non-negative vector $\gamma \in \mathbb{R}^m$ such that $c_i = \gamma^\top A$ and $\gamma^\top b \leq d_i$. Let $\widetilde{\mathbb{E}}$ be a degree-$(r+1)$ SoS pseudoexpectation for P. We need to show that property (SoS.3) is satisfied for every inequality $(d_i - c_i^\top x) \geq 0$, with $i \in [p]$. Let $f \in \mathbb{R}[x]/\mathbf{I}_n$ with $\deg\left(\overline{f^2(d_i - c_i^\top x)}\right) \leq r$. By basic algebraic manipulation it holds that

$$\widetilde{\mathbb{E}}(f^2(d_i - c_i^\top x)) = (d_i - \gamma^\top b)\widetilde{\mathbb{E}}(f^2) + \sum_{j=1}^m \gamma_j \widetilde{\mathbb{E}}(f^2(b_j - a_j^\top x)) \geq 0,$$

where the last inequality follows from the construction of γ, the fact that for each $j \in [m]$ we have $\deg\left(\overline{f^2(b_j - a_j^\top x)}\right) \leq r+1$, and hence $\widetilde{\mathbb{E}}(f^2(b_j - a_j^\top x)) \geq 0$ and $\widetilde{\mathbb{E}}(f^2) \geq 0$. $\square$

References

1. Alon, N., Azar, Y., Woeginger, G.J., Yadid, T.: Approximation schemes for scheduling. In: Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 493–500 (1997)
2. Alon, N., Azar, Y., Woeginger, G.J., Yadid, T.: Approximation schemes for scheduling on parallel machines. J. Sched. **1**(1), 55–66 (1998)
3. Askey, R.: Orthogonal polynomials and special functions. In: CBMS-NSF Regional Conference Series in Applied Mathematics, vol 21. SIAM (1975)
4. Au, Y.H.G., Tunçel, L.: Elementary polytopes with high lift-and-project ranks for strong positive semidefinite operators. Discrete Optim. **27**, 103–129 (2018)
5. Barak, B., Chan, S.O., Kothari, P.K.: Sum of squares lower bounds from pairwise independence. In: Proceedings of the 47th Annual ACM Symposium on Theory of Computing (STOC), pp. 97–106 (2015)
6. Barak, B., Hopkins, S.B., Kelner, J., Kothari, P., Moitra, A., Potechin, A.: A nearly tight sum-of-squares lower bound for the planted clique problem. In: Proceedings of the 57th Annual Symposium on Foundations of Computer Science (FOCS), pp. 428–437 (2016)
7. Barak, B., Moitra, A.: Noisy tensor completion via the sum-of-squares hierarchy. In: Proceedings of the 29th Conference on Learning Theory (COLT), pp. 417–445 (2016)
8. Blekherman, G., Gouveia, J., Pfeiffer, J.: Sums of squares on the hypercube. Math. Z. **284**(1–2), 41–54 (2016)
9. Chlamtac, E., Tulsiani, M.: Convex relaxations and integrality gaps. In: Handbook on Semidefinite, Conic and Polynomial Optimization, pp 139–169. Springer, Belin (2012)
10. Cox, D.A., Little, J.B., O'Shea, D.: Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra. Undergraduate Texts in Mathematics. Springer, New York (2007)
11. Eisenbrand, F., Weismantel, R.: Proximity results and faster algorithms for Integer Programming using the Steinitz Lemma. In: Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 808–816 (2018)
12. Garg, S.: Quasi-PTAS for scheduling with precedences using LP hierarchies. In: Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP), pp. 59:1–59:13 (2018)
13. Gatermann, K., Parrilo, P.A.: Symmetry groups, semidefinite programs, and sums of squares. Journal of Pure and Applied Algebra **192**(1–3), 95–128 (2004)
14. Goemans, M., Rothvoß, T.: Polynomiality for bin packing with a constant number of item types. In: Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 830–839 (2014)
15. Goemans, M., Williamson, D.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. JACM **42**, 1115–1145 (1995)
16. Graham, R.L.: Bounds for certain multiprocessing anomalies. Bell Syst. Tech. J. **45**(9), 1563–1581 (1966)
17. Grigoriev, D.: Complexity of positivstellensatz proofs for the knapsack. Comput. Complex. **10**(2), 139–154 (2001)
18. Grigoriev, D.: Linear lower bound on degrees of positivstellensatz calculus proofs for the parity. Theor. Comput. Sci. **259**(1–2), 613–622 (2001)
19. Hochbaum, D.: Approximation Algorithms for NP-Hard Problems. PWS Publishing Co., Boston (1996)
20. Hochbaum, D., Shmoys, D.: Using dual approximation algorithms for scheduling problems theoretical and practical results. JACM **34**, 144–162 (1987)
21. Hopkins, S.B., Kothari, P.K., Potechin, A., Raghavendra, P., Schramm, T., Steurer, D.: The power of sum-of-squares for detecting hidden structures. In: Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science (FOCS), pp. 720–731 (2017)

22. Jansen, K.: An EPTAS for scheduling jobs on uniform processors: using an milp relaxation with a constant number of integral variables. SIAM J. Discrete Math. **24**(2), 457–485 (2010)
23. Jansen, K., Klein, K., Verschae, J.: Closing the gap for makespan scheduling via sparsification techniques. In: Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming (ICALP), pp. 72:1–72:13 (2016)
24. Karlin, A., Mathieu, C., Nguyen, C.: Integrality gaps of linear and semi-definite programming relaxations for knapsack. In: Proceedings of the 15th International Conference on Integer Programming and Combinatoral Optimization (IPCO), pp. 301–314. (2011)
25. Kothari, P., O'Donnell, R., Schramm, T.: SOS lower bounds with hard constraints: think global, act local. arXiv:1809.01207 (2018)
26. Kothari, P.K., Mori, R., O'Donnell, R., Witmer, D.: Sum of squares lower bounds for refuting any CSP. In: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC), pp. 132–145 (2017)
27. Kothari, P.K., Steinhardt, J., Steurer, D.: Robust moment estimation and improved clustering via sum of squares. In: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, pp. 1035–1046 (2018)
28. Kurpisz, A., Leppänen, S., Mastrolilli, M.: On the hardest problem formulations for the 0/1 lasserre hierarchy. Math. Oper. Res. **42**(1), 135–143 (2016)
29. Kurpisz, A., Leppänen, S., Mastrolilli, M.: Sum of squares hierarchy lower bounds for symmetric formulations. In: Proceedings of the 18th International Conference on Integer Programming and Combinatorial Optimization (IPCO), pp. 362–374 (2016)
30. Kurpisz, A., Leppänen, S., Mastrolilli, M.: An unbounded sum-of-squares hierarchy integrality gap for a polynomially solvable problem. Math. Program. **166**(1–2), 1–17 (2017)
31. Kurpisz, A., Mastrolilli, M., Mathieu, C., Mömke, T., Verdugo, V., Wiese, A.: Semidefinite and linear programming integrality gaps for scheduling identical machines. Math. Program. **172**(1–2), 231–248 (2018)
32. Lasserre, J.: Global optimization with polynomials and the problem of moments. SIAM J. Optim. **11**, 796–817 (2001)
33. Laurent, M.: A comparison of the Sherali-Adams, Lovász-Schrijver, and Lasserre relaxations for 0–1 programming. Math. Oper. Res. **28**, 470–496 (2003)
34. Laurent, M.: Lower bound for the number of iterations in semidefinite hierarchies for the cut polytope. Math. Oper. Res. **28**(4), 871–883 (2003)
35. Laurent, M.: Semidefinite representations for finite varieties. Math. Program. **109**(1), 1–26 (2007)
36. Laurent, M.: Sums of squares, moment matrices and optimization over polynomials. In: Emerging Applications of Algebraic Geometry, pp. 157–270. Springer, New York (2009)
37. Levey, E., Rothvoss, T.: A (1+epsilon)-approximation for makespan scheduling with precedence constraints using LP hierarchies. In: Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing (STOC), pp. 168–177 (2016)
38. Ma, T., Shi, J., Steurer, D.: Polynomial-time tensor decompositions with sum-of-squares. In: Foundations of Computer Science (FOCS), pp. 438–446 (2016)
39. Margot, F.: Symmetry in integer linear programming. In: 50 Years of Integer Programming 1958–2008. Springer, Berlin (2010)
40. Mastrolilli, M.: High degree sum of squares proofs, Bienstock–Zuckerberg hierarchy and cg cuts. In: International Conference on Integer Programming and Combinatorial Optimization, pp. 405–416. Springer, New York (2017)
41. Parrilo, P.: Semidefinite programming relaxations for semialgebraic problems. Math. Program. **96**(2), 293–320 (2003)
42. Potechin, A.: Sum of squares lower bounds from symmetry and a good story. arXiv:1711.11469 (2017)
43. Potechin, A., Steurer, D.: Exact tensor completion with sum-of-squares. arXiv:1702.06237 (2017)
44. Raghavendra, P., Schramm, T., Steurer, D.: High-dimensional estimation via sum-of-squares proofs. arXiv:1807.11419 (2018)
45. Raymond, A., Saunderson, J., Singh, M., Thomas, R.R.: Symmetric sums of squares over k-subset hypercubes. Math. Program. **167**(2), 315–354 (2018)
46. Razborov, A.A.: Flag algebras. J. Symb. Log. **72**(4), 1239–1282 (2007)
47. Razborov, A.A.: On 3-hypergraphs with forbidden 4-vertex configurations. SIAM J. Discrete Math. **24**(3), 946–963 (2010)
48. Rothvoß, T.: The Lasserre hierarchy in approximation algorithms. Lecture notes for MAPSP (2013)

49. Sagan, B.: The Symmetric Group. Graduate Texts in Mathematics. Springer, New York (2001)
50. Schoenebeck, G.: Linear level Lasserre lower bounds for certain k-CSPS. In: Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 593–602 (2008)
51. Sherali, H., Adams, W.: A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. SIAM J. Discrete Math. **3**(3), 411–430 (1990)
52. Svensson, O.: Santa claus schedules jobs on unrelated machines. SIAM J. Comput. **41**(5), 1318–1341 (2012)
53. Verdugo, V., Verschae, J.: Breaking symmetries to rescue sum of squares: the case of makespan scheduling. In: 20th International Conference on Integer Programming and Combinatorial Optimization (IPCO 2019), pp. 427–441 (2019)
54. Williamson, D., Shmoys, D.: The Design of Approximation Algorithms. Cambridge University Press, Cambridge (2011)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Victor Verdugo[1,2] · **José Verschae**[4] · **Andreas Wiese**[3]

✉ Victor Verdugo
v.verdugo@lse.ac.uk; victor.verdugo@uoh.cl

José Verschae
jose.verschae@uoh.cl

Andreas Wiese
awiese@dii.uchile.cl

1 Department of Mathematics, London School of Economics and Political Science, London, UK

2 Institute of Engineering Sciences, Universidad de O'Higgins, Rancagua, Chile

3 Department of Industrial Engineering, Universidad de Chile, Santiago, Chile

4 Institute for Mathematical and Computational Engineering, Faculty of Mathematics and School of Engineering, Pontificia Universidad Católica de Chile, Santiago, Chile