Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling (ICAPS 2017)

An Investigation of Phase Transitions in Single-Machine Scheduling Problems

Zhihui Wang,[+,] Bryan O'Gorman,[+,†] Tony T. Tran,[*]**
Eleanor G. Rieffel,[+] Jeremy Frank,[‡] Minh Do[‡,†]
[+] Quantum Artificial Intelligence Lab., NASA Ames Research Center, Moffett Field, CA
‡Intelligent Systems Division, NASA Ames Research Center, Moffett Field, CA
** Universities Space Research Association, Mountain View, CA
†Stinger Ghaffarian Technologies, Inc., Greenbelt, MD
[*]Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON

Abstract

We investigate solvable-unsolvable phase transitions in the single-machine scheduling (SMS) problem. SMS is at the core of practical problems such as telescope and satellite scheduling and manufacturing. To study the solvability phase transition, we construct a variety of instance families parameterized by the set of the processing times, the window size (deadline minus release time), and the horizon. We empirically establish the phase transition and look for an easy-hard-easy pattern for this family using several common solvers. While in many combinatorial problems a phase transition coincides with typically hard instances, whether or not that is the case with SMS remains an open question, and merits further study.

1 Introduction

Single-machine scheduling (SMS), in which a set of jobs with release times, deadlines, and processing times are to be scheduled on a single machine, forms the backbone of many practical applications such as telescope scheduling, satellite scheduling, and manufacturing. A phase transition is a sudden change in a global property of a family with respect to an order parameter. For a number of NP-complete problems, the phase transition from solvable to unsolvable problems as a function of the number and tightness of constraints is of interest. On average, instances near this phase transition are typically exponentially harder than those that are not, and this clustering of hard instances near the phase transition becomes more concentrated as problem size increases.

In contrast to the many problems for which a phase transition has been found, to date the existence of phase transitions in SMS remains unexplored. In this paper, we explore solvable-unsolvable phase transitions in the decision version of the SMS problem. We construct a model for parameterized families of SMS instances and identify the order parameters. Our significant findings are: 1) empirical evidence of a rapid transition in solvability (phase transition) for SMS, where none was previously characterized. 2) Evidence of this transition in provably intractable SMS ensembles, where the transition is characterized by two parameters, which is novel for phase transitions.

Where phase transitions have been identified in hard problems, most published results have shown "easy-hard-easy" behavior accompanying the phase transition, with the notable exception of Hamiltonicity. Our empirical results shows that for the intractable SMS family we study, the hard instances are very rare and only loosely correlates with the phase transition. While our results don't rule out an easy-hard-easy pattern in the NP-complete SMS family, they invite further questions into the relationship between hardness and phase transitions, and demonstrate the challenge of generating hard families of instances.

2 Related Work

In their pioneering work (Erdős and Rényi 1960), Erdős and Rényi identified phase transitions of graph properties in their eponymous model. For combinatorial optimization problems, phase transition was first identified in SAT in (Hooker and Fedjki 1990). Later (Cheeseman, Kanefsky, and Taylor 1991) exhibited a connection between phase transitions and the location of hard instances in a handful of NP-complete problems (Hamiltonian cycle, graph coloring, k-SAT, and TSP). Since then, phase transitions have been found in a number of other combinatorial optimization problems such as independent set (Gent and Walsh 1994), number partitioning (Gent and Walsh 1996a; Mertens 1998; Borgs, Chayes, and Pittel 2001), and constraint satisfaction (Prosser 1996; Smith and Dyer 1996). The characterization of phase transitions in Cheeseman et al.'s original examples has later been refined (e.g. Hamiltonian cycle (Vandegriend and Culberson 1998), graph coloring (Achlioptas and Friedgut 1999), k-SAT (Mitchell, Selman, and Levesque 1992; Kirkpatrick and Selman 1994), and travelling salesman problem (Gent and Walsh 1996b)). Specifically, the typical solution time of instances at the phase transition grows exponentially with the problem size, and instances away from the transition are typically easy.

There is also evidence that in some problems the hardest instances suddenly emerge at some critical threshold below the solvability one (Hogg and Williams 1994). The concentration of hard instances near the phase transition is of practical interest because it enables the generation of hard instances for benchmarking algorithms and solvers (Rieffel et al. 2014; Hoos and Stützle 2000).

The occurence of phase transitions is not limited to hard problems; they occur in provably easy problems as well, including graph properties such as connectivity (Erdős and Rényi 1960), 2-SAT (Chvatal and Reed 1992; Goerdt 1996), XOR-SAT (Creignou and Daude 1999) and Horn-SAT (Moore et al. 2005). For hard problems, the existence of a solvability threshold does not necessarily imply an easy-hard-easy pattern, though counterexamples are rare. One of such counterexample is the phase transition of Hamiltonicity in the Erdős-Rényi model (Vandegriend and Culberson 1998). Our work introduces the first analysis of the phase transition for SMS instances in the literature.

3 Preliminaries

3.1 Single-Machine Scheduling

In this paper we consider the decision version of (non-preemptive) single-machine scheduling with (integer-valued) release times, deadlines, and processing times (i.e. $1|r_j d_j|U_{\max}$), which we henceforth refer to simply as SMS. An instance of SMS consists of a set of n jobs; each job j has a release time, deadline, and processing time ($r_j, d_j, p_j \in \mathbb{Z}_0^+$, respectively). The problem is to decide whether or not there exists a *schedule* $\boldsymbol{\sigma} \in \left(\mathbb{Z}_0^+\right)$ on a single machine, where σ_j indicates the starting time of job j, such that

- every job j starts no sooner than its release time, $r_j \leq \sigma_j$;
- every job j finishes by its deadline, $\sigma_j + p_j \leq d_j$; and
- no two jobs i and j overlap, $\sigma_i + p_i \leq \sigma_j$ or $\sigma_j + p_j \leq \sigma_i$.

We refer to the difference between the release time and the deadline for each job as that job's *window* $w_j = d_j - r_j$. More concisely, we write an instance as a tuple (r_j, w_j, p_j). The *horizon* T is a time no earlier than the latest deadline, often exactly so.

In general, SMS is NP-complete, see (Pinedo 2002). It can also be shown by reduction from bin packing. However, there are numerous complexity results on refinements of the SMS problem. For example, with unit processing time, the greedy earliest-deadline algorithm suffices (Dürr and Hurand 2011; Sgall 2012). More generally, a valid schedule can be found in quasi-linear time when the processing times are identical (Simons 1978; Garey et al. 1981) and in polynomial time when the processing times are restricted to be either one or some arbitrary but fixed constant (Sgall 2012). When the processing times are restricted to two fixed constants greater than one, SMS remains NP-complete (Elffers and de Weerdt 2014).

3.2 Parameterized Ensembles

Unlike other problems in which phase transitions are studied, there is no generative model of SMS instances showing a phase transition in the literature. While not completely general, SMS with two processing times is NP-complete (Elffers and de Weerdt 2014) and constitutes the target of our study. We find phase transition in solvability characterized by *two* parameters, the scheduling horizon T and the window size W. We limit the sets of possible processing times to be $P = \{p_\mathrm{s}, p_\mathrm{l}\}$ and the window lengths in the range $W = [p_\mathrm{l}+1, w_{\max}]$. We fix $p_\mathrm{s}, p_\mathrm{l}$, and study the ensembles of instances parameterized by the tuple $(T, w_{\max})$. For each job, p_i and w_i are uniformly sampled from P and W, respectively. The release time r_i is then uniformly sampled from $[0,\ T - w_i]$.

We choose the problem family of parameters $P = \{3,\ 19\}$, $\{7,\ 11\}$, and $\{3,\ 11\}$, and problem size n ranging from 16 to 200. For each value of the tuple, 100 to 1000 instances are drawn.

We carefully chose processing times to reduce the chance of creating easy instances: the processing times are coprime; they are sufficiently different that for large window sizes they are not effectively the same, but not so different that the smaller one is effectively one.

3.3 Experimental Methods

We deployed a variety of solvers by mapping SMS to different canonical problems: Mixed Integer Linear Programming (MILP), Satisfiability (SAT), and Constraint Programming (CP). All the solvers we tried gave consistent answers with respect to solvability when run on the same instances. We found that IBM ILOG's CP solver, CP Optimizer, significantly outperforms the alternatives we tried, both mapping to MILP or SAT and using other CP solvers. This is to be expected, given that the CP more naturally captures the structure of SMS, whereas this structure is lost in the mappings to both MILP (because it requires ancillary variables to account for the disjunction in the overlap constraint) and to SAT (because integer variables must be encoded using Boolean variables). We allowed a maximum running time of 24 hours. The problem ensemble we study containes instances not solved within this time limit, and very few at that (fewer than one percent for every set of parameter values). These instances are negligible for computing the probability of solvability, but are counted when we examine the hardness. We present only the results of CP optimizer, and use the number of nodes explored in the constraint programming search as a measure of the computational cost, which is roughly proportional to the run time.

4 Results: Phase Transitions

Figure. 1 (a) shows the probability of solvability in the two-dimensional parameter space $(T, w_{\max})$, for $P = \{3, 19\}$ and $n = 200$. With a normalization of the parameters as $T/(n\bar{p})$ and $w_{\max}/T$, where $\bar{p} = (p_\mathrm{s} + p_\mathrm{l})/2$, a clear transition is seen. On the bottom left of the scanning area, i.e., the time horizon is close to the sum of the processing times, and the jobs have very little flexibility, few solution exists. On the top right of the area, i.e., the horizon is ample to fit all jobs loosely, and the windows apply almost no constraint to the schedule and there are many solutions to the problem. The same plots for $P = \{7, 11\}$ are shown in Fig. 2 (a) and (d) for $n = 200$ and 40, respectively. This transition gets steeper as the problem size increases, as reflected by the reduction in width of the "dark (blue) band". The transition

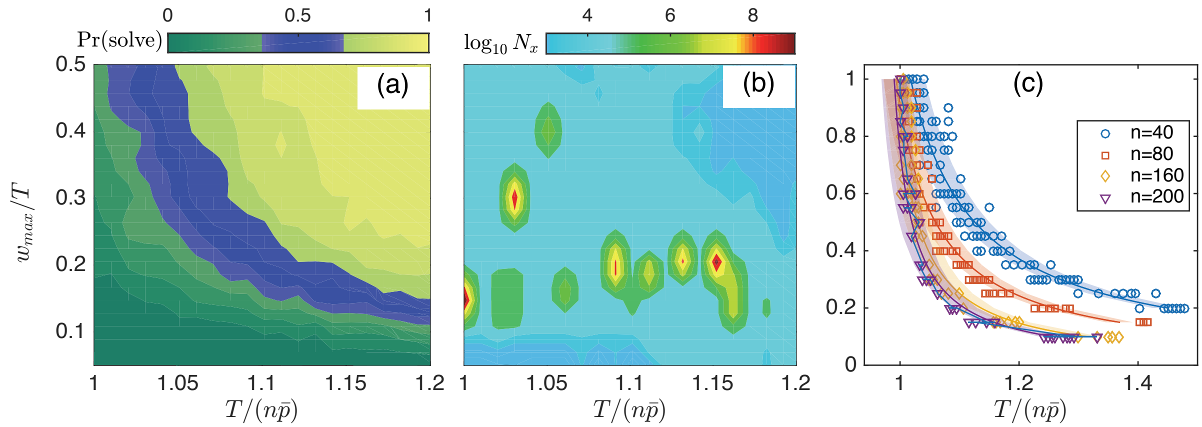

Figure 1: $P = \{3, 19\}$. For each parameter tuple $(T, w_{\max})$ 100 instances were drawn. (a) Contour plot of solvability probability $\Pr(\text{solve})$ for $n = 200$. The horizontal axis is the horizon, normalized by the average total processing time $n\bar{p}$, and the vertical axis is the ratio of the maximum window length to the horizon $w_{\max}/T$. The dark blue region marks the phase transition frontier. (b) Contour plot of hardness where color encodes $\log_{10}(N_x)$ for $x = 99$, where N_x is the x-th percentile of the number of nodes for each parameter tuple $(T, w_{\max})$. (c) Plot of $\Pr(\text{solve}) = 0.5 \pm 0.05$ for $n = 40, 80, 160, 200$. The transition frontier is fit as Eq. (1). The shaded area shows 95 percent confidence interval. The fitted coefficients $(a_0, c_0) = (0.11, 0.91)$, $(0.066, 0.93)$, $(0.035, 0.96)$, $(0.029, 0.96)$ for $n = 40, 80, 160, 200$, respectively.

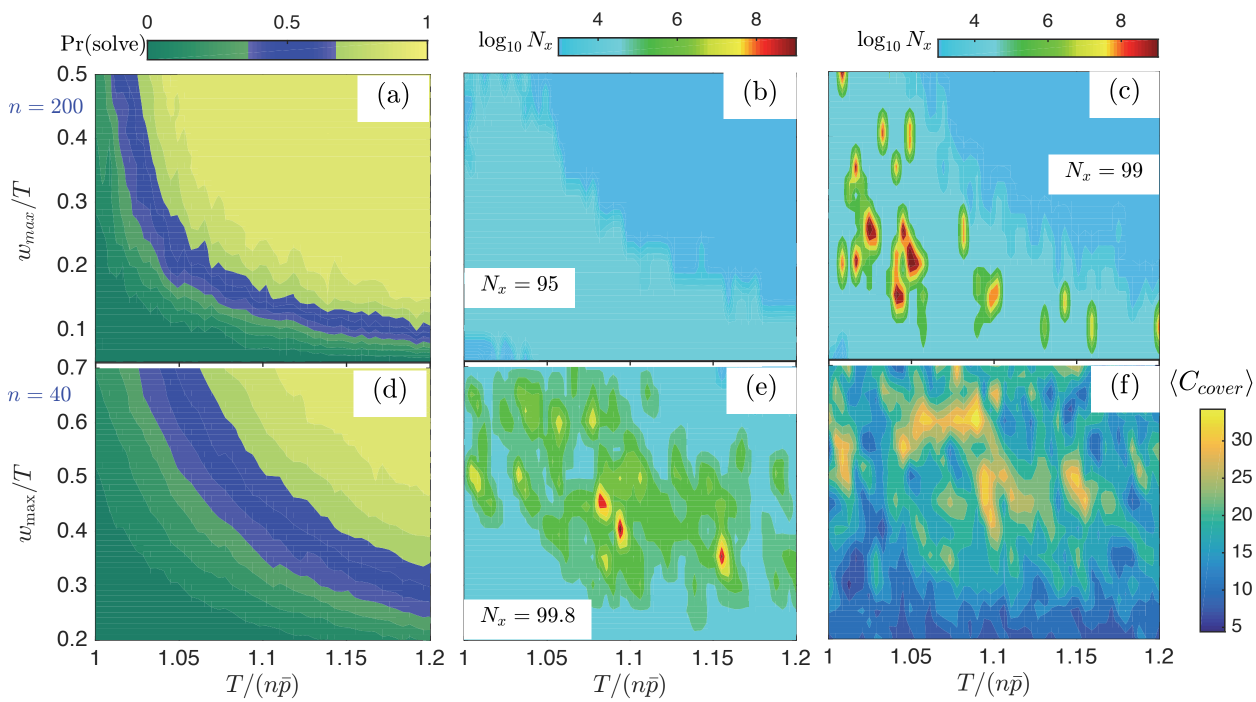

Figure 2: $P = \{7, 11\}$. Results on the top and bottom panels are for $n = 200$, and $n = 40$, with 100 and 1000 instances drawn for each parameter tuple $(T, w_{\max})$, respectively. The horizontal axis is the horizon, normalized by the average total processing time $n\bar{p}$, and the vertical axis is the ratio of the maximum window length to the horizon $w_{\max}/T$. (a) and (d) Contour plot of solvability probability $\Pr(\text{solve})$. The dark blue region is the phase transition frontier. (b), (c), and (e) Contour plot of hardness where color encodes $\log_{10}(N_x)$, where N_x is the x-th percentile of the number of nodes for each parameter tuple $(T, w_{\max})$. (f) Number of covers C_{cover} averaged over the top 5 hardest instances for each parameter.

between these two regimes is a simple curve that is a function of T and $w_{\max}$. The shape of the curves also suggests that the transition region is converging as n increases. We take a closer look in Fig. 1 (c) where for $P = \{3,\ 19\}$, numerical data with $\Pr(\text{solve}) = 0.5 \pm 0.05$ is singled out as the transition frontier, and fit as

$$\frac{T}{n\bar{p}} = \frac{a_0}{w_{\max}/T} + c_0 \tag{1}$$

As n increases, the fit parameters a_0 decreases, and c_0 approaches 1.

A similar pattern in the phase transition is observed for the other set of processing times we studied, $P = \{3,\ 11\}$.

5 Results: Complexity and Empirical Hardness

With two non-unit processing times and arbitrary window lengths (that may grow with n), SMS is NP-complete. The ensembles we investigate should therefore contain some hard problems. We found that such instances are typically easy, while rare hard instances exhibit a loose correlation with the solvability phase transition.

To study the hardness of the problem family, for each parameter tuple $(T, w_{\max})$, out of all instances generated, 50-th to 99.8-th percentiles of the number of nodes explored in the attempt to solve the problem instances are taken as a measure.

5.1 Typical instances are easy

We first examine the median effort to solve problems. We do not observe a clear correlation between the locations of relative hard instances and the regions of the phase transition. Also, no evidence of an exponential increase in the median hardness with problem size is observed. This indicates that the median case for such generated problems are probably easy. The same phenomenon appears in higher percentiles, we observed no hard instances up to 95-th percentiles, see the results for 95-th percentiles in Fig. 2 (b) for $n = 200$.

5.2 High percentiles: rare hard instances, weak correlation with the phase transition

We then look at even higher percentiles of the number of nodes explored. As shown in Fig. 1 (b) and Fig. 2 (c), (e), relatively hard instances show up at these very high percentiles (e.g., 99-th percentiles for $n = 200$ and 99.8-th percentiles for $n = 40$). In Fig. 2 (c), (e), high percentiles of the nodes explored is plotted in logarithmic scale for parameter $P = \{7,\ 11\}$, and the phase transitions are shown in parallel in (a), (d) for comparison. Hard instances are scattered around the phase transition region, suggesting a weak correlation. Due to the rareness of the hard instances and large statistical fluctuations in the high percentiles, we cannot quantify the correlation between the phase transition and the hardness in high percentiles. Note that also because the hard instances are rare, one would expect them to be sparsely distributed, resulting in no clear boundary. The hard instances do not correspond exactly with the solvability transition, but this is often the case with phase transitions.

To compare hardness for different problem sizes, due to the diversion of hard instances in the parameter space, a reliable fitting for extracting the number of nodes of certain percentiles is missing and the results are thus tempered by large statistical fluctuations. Furthermore, for larger problem sizes, results of higher percentile is also limited by the 24-hour timeout of the solver. Therefore, we do not draw conclusions on the empirical time complexity for such high percentiles.

5.3 Structure of hard instances

What makes these instances hard? There are both solvable and unsolvable instances that are hard. Define job j' as a *cover* of job j if 1) $r_j > r_{j'}$ and $d_j < d_{j'}$ and 2) the window of job j cannot accomodate both jobs, i.e., $d_j - r_j < p_j + p_{j'}$. The underlying intuition is that when j' covers j, then the scheduler must decide if j' is scheduled on the left or right of the smaller window, forcing some amount of disjunctive search. These disjunctions are more constrained than those in which, for instance, the two windows overlap enough to admit either ordering. We hypothesize that the more covers, the higher the search cost. We found that the number of such covers in an instance exhibits a weak correlation with the hardness, as shown in Fig. 2 (f) for $n = 40$, $P = \{7, 11\}$, where, to capture the rare hard instances, for each parameter tuple $(T, w_{\max})$, the number of covers C_{cover} is averaged over the hardest five instances instead of all instances (most of them are easy).

6 Conclusions

Our work is the first study of phase transitions for the SMS problem. We empirically identified solvable-unsolvable phase transitions in an intractable parametrized family of SMS instances. The main ensemble we studied is known to contain hard problems, but typical problems are easy. The rare hard instances correlated loosely with the phase transition region. Note that tractable SMS families can also show solvable-unsolvable phase transition. One example is when all jobs have the same processing time and the same window size; the window size, after proper normalization, serves as an order parameter for the phase transition. Other families include when the window lengths are either identical or restricted to two constants that differ by one, for which a greedy earliest-deadline algorithm solves the problem in polynomial time. Analyzing the location and behavior of the phase transition, starting with these simple tractable cases, is part of our future work. Our results also indicate that it is likely that instances with high number of job covers tend to be harder. Hence constructing problem families where the covers have a higher likelihood to exist could be helpful in finding hard problems. While this work gives insight into what makes certain SMS instances tractable and others not, it also leaves many mysteries and tantalizing open questions for further exploration: For SMS, are there effective ways of locating hard problem instances? Are there ways of identifying and efficiently generating one or more parametrized families of hard SMS instances? For combinatorial optimization problems in general, under what circumstances should one expect there to be a close relation between a phase transition and hard problem instances? Are there accessible features of problem classes or properties of phase transitions that suggest when one would expect to hard instances at the phase transition?

7 Acknowledgements

We thank Chris Beck and Davide Venturelli for useful discusions. The authors would like to acknowledge support from the NASA Advanced Exploration Systems program and NASA Ames Research Center. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright annotation thereon.

References

Achlioptas, D., and Friedgut, E. 1999. A sharp threshold for k-colorability. *Random Structures and Algorithms* 14(1):63–70.

Borgs, C.; Chayes, J.; and Pittel, B. 2001. Phase transition and finite-size scaling for the integer partitioning problem. *Random Structures and Algorithms* 19(3-4):247–288.

Cheeseman, P.; Kanefsky, B.; and Taylor, W. M. 1991. Where the really hard problems are. In *IJCAI*, volume 91, 331–337.

Chvatal, V., and Reed, B. 1992. Mick gets some (the odds are on his side) (satisfiability). In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, SFCS '92, 620–627. Washington, DC, USA: IEEE Computer Society.

Creignou, N., and Daude, H. 1999. Satisfiability threshold for random XOR-CNF formulas. *Discrete Applied Mathematics* 96–97:41 – 53.

Dürr, C., and Hurand, M. 2011. Finding total unimodularity in optimization problems solved by linear programs. *Algorithmica* 59(2):256–268.

Elffers, J., and de Weerdt, M. 2014. Scheduling with two non-unit task lengths is NP-complete. *arXiv preprint arXiv:1412.3095*.

Erdős, P., and Rényi, A. 1960. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci* 5:17–61.

Garey, M. R.; Johnson, D. S.; Simons, B. B.; and Tarjan, R. E. 1981. Scheduling unit–time tasks with arbitrary release times and deadlines. *SIAM Journal on Computing* 10(2):256–269.

Gent, I. P., and Walsh, T. 1994. The hardest random SAT problems. In *Annual Conference on Artificial Intelligence*, 355–366. Springer.

Gent, I., and Walsh, T. 1996a. Phase transitions and annealed theories: Number partitioning as a case study. In *In Proceedings of ECAI-96*, 170–174. John Wiley & Sons.

Gent, I. P., and Walsh, T. 1996b. The TSP phase transition. *Artificial Intelligence* 88(1):349 – 358.

Goerdt, A. 1996. A threshold for unsatisfiability. *Journal of Computer and System Sciences* 53(3):469 – 486.

Hogg, T., and Williams, C. P. 1994. The hardest constraint problems: A double phase transition. *Artif. Intell.* 69(1-2):359–377.

Hooker, J. N., and Fedjki, C. 1990. Branch-and-cut solution of inference problems in propositional logic. *Annals of Mathematics and Artificial Intelligence* 1(1):123–139.

Hoos, H. H., and Stützle, T. 2000. SATLIB: An online resource for resesarch on SAT. In Gent, I.; van Maaren, H.; and Walsh, T., eds., *SAT 2000*. IOS Press. 283–292.

Kirkpatrick, S., and Selman, B. 1994. Critical behavior in the satisfiability of random boolean expressions. *Science* 264(5163):1297–1301.

Mertens, S. 1998. Phase transition in the number partitioning problem. *Phys. Rev. Lett.* 81:4281–4284.

Mitchell, D.; Selman, B.; and Levesque, H. 1992. Hard and easy distributions of SAT problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, AAAI'92, 459–465. AAAI Press.

Moore, C.; Istrate, G.; Demopoulos, D.; and Vardi, M. Y. 2005. *A Continuous-Discontinuous Second-Order Transition in the Satisfiability of Random Horn-SAT Formulas*. Berlin, Heidelberg: Springer Berlin Heidelberg. 414–425.

Pinedo, M. 2002. *Scheduling: Theory, Algorithms and Systems*. Prentice Hall, 2nd edition.

Prosser, P. 1996. An empirical study of phase transitions in binary constraint satisfaction problems. *Artificial Intelligence* 81(1):81–109.

Rieffel, E. G.; Venturelli, D.; Hen, I.; Do, M.; and Frank, J. 2014. Parametrized families of hard planning problems from phase transitions. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI-14)*, 2337 – 2343.

Sgall, J. 2012. *Open Problems in Throughput Scheduling*. Berlin, Heidelberg: Springer Berlin Heidelberg. 2–11.

Simons, B. 1978. A fast algorithm for single processor scheduling. In *Foundations of Computer Science, 1978., 19th Annual Symposium on*, 246–252.

Smith, B. M., and Dyer, M. E. 1996. Locating the phase transition in binary constraint satisfaction problems. *Artificial Intelligence* 81(1):155–181.

Vandegriend, B., and Culberson, J. 1998. The $G(n, m)$ phase transition is not hard for the Hamiltonian cycle problem. *J. Artif. Int. Res.* 9(1):219–245.