Wojciech BOŻEJKO
Politechnika Wrocławska
Paweł RAJBA
Uniwersytet Wrocławski
Mieczysław Wodecki
Politechnika Wrocławska

ANALIZA DOSTOSOWYWANIA TABU SEARCH DLA PROBLEMU SZEREGOWANIA NA JEDNEJ MASZYNIE W WARUNKACH NIEPEWNOŚCI

Streszczenie. W tej pracy kontynuujemy analizę jak dopasowanie algorytmu tabu search dla odpowiedniego poziomu zaburzenia danych danych daje lepsze rezultaty. W pracy rozpatrujemy szeregowanie zadań na jednej maszynie z danymi zaburzonymi modelowanymi przez rozkład normalny. Wyniki eksperymentów pokazują, że odpowiednie dopasowanie algorytmu daje lepsze rezultaty niż w przypadku braku uwzględnienia poziomu zaburzenia danych.

ANALYSIS OF CUSTOMIZING TABU SEARCH FOR UNCERTAIN SINGLE MACHINE SCHEDULING

Summary. This paper is a continuation of a similar one and we make an analysis to verify how much customizing tabu search algorithm for a specific data disturbance level make a difference. We consider a single machine scheduling problem and data modeled with the normal distribution on different data disturbance levels. The computational experiments results show that the investigated optimization method provides more robust results when customized for specific disturbed data distributions.

1. Introduction

Many optimization problems needs to be considered with the assumption that certain parameters are not known at the time when the solution is being calculated. In domains like production, manufacturing, delivering goods, supply chain and many others there are a lot of parameters which depends on external factors which can't be predicted in advance, for instance in the transportation domain delivery on time may be impacted by weather conditions, cars' breakdowns, traffic jams, driver's condition and others. In such a case where the problem input data is uncertain, applying the deterministic approach may be not appropriate. We may experience significant increase of execution cost (and by that losing the assumed delivery time) or even losing the acceptability (feasibility) of solutions.

Uncertainty in optimization problems has been investigated for many decades next to the research focused on the deterministic models. Basics of stochastic scheduling are described in Pinedo [10]. More detailed reviews dedicated to methods solving scheduling problems in stochastic models are presented in Cai et. al. [7], Dean [8], Soroush [14], Urgo and Vancza [15], Vondrák [16], and Zhang et al. [17]. The approach which is a baseline for the analysis in this paper was investigated in Bożejko et al. [1], [2], [3], [4], [6], Rajba et. al. [11], [12] and Rajba [13] where effective methods were proposed for single machine scheduling problem where parameters are modelled with random variables with the normal distribution. Moreover, in [1], [2] and [11] Erlang distribution was investigated and those papers cover $\sum w_i U_i$ and $\sum w_i T_i$ problem variants. On top of that in [3] and [4] techniques to decrease further the computational time (i.e. elimination criteria and random blocks) were introduced keeping the robustness of the determined solutions on the good level.

In this paper we investigate a tabu search method for a single machine scheduling problem tailored for uncertain data with the normal distribution as described in [1]. This is the continuation of the investigation presented in [5] where an another variant of the problem was analyzed. The test data is generated using the normal distribution with different deviation levels based on the OR Library data set. Having the test data, we execute algorithm configured for a specific disturbance level several times for each data set with data disturbed with the normal distribution with a given disturbance level. As an additional data set we included also data with the uniform distribution within a specific range. All the executions are to verify how much tailoring an algorithm for a given disturbance level gives better results when targeting data set of that specific disturbance level. The conducted computational experiments show that the considered robust optimization method provides better results when configured for given disturbed data distributions, however different variants in a slightly different way.

The rest of the paper is structured as follows: in Section 2 we describe a classic deterministic version of the problem, then in Section 3 we introduce a randomized variant of the one. In Section 4 we present disturbed data sets and the analysis approach, and in Section 5 the main results and a summary of computational experiments are described. Finally, in Section 6 conclusions and future directions close the paper.

2. Deterministic scheduling problem

Let $\mathcal{J} = \{1, 2, \ldots, n\}$ be a set of jobs where for each $i \in \mathcal{J}$ we define p_i as a *processing time*, d_i as a *due date* and w_i as a cost for a delay. All jobs shall be executed on a single machine under the following main conditions: (1) at any given moment at most one job can be executed and (2) all jobs must be executed without preemption.

Let Π be the set of all permutations of the set $\mathcal{J}$. For each permutation $\pi \in \Pi$ we define

$$C_{\pi(i)} = \sum_{j=1}^{i} p_{\pi(j)}$$

as the completion time of a job $\pi(i)$.

Then we introduce the delay cost as

$$T_{\pi(i)} = \max\{0, C_{\pi(i)} - d_{\pi(i)}\}$$

and the cost function for the permutation π as

$$\sum_{i=1}^{n} w_{\pi(i)} T_{\pi(i)}. \tag{1}$$

Finally, the goal is to find a permutation $\pi^* \in \Pi$ which minimizes

$$W(\pi^*) = \min_{\pi \in \Pi} \left(\sum_{i=1}^{n} w_{\pi(i)} T_{\pi(i)} \right).$$

3. Probabilistic model

In this section we introduce a probabilistic version of the problem and we investigate two variants: (a) uncertain processing times and (b) uncertain due dates. In order to simplify the further considerations we assume w.l.o.g. that at any moment the considered solution is the natural permutation, i.e. $\pi = (1, 2, \ldots, n)$.

3.1. Random processing times

Random processing times are represented by random variables with the normal distribution $\tilde{p}_i \sim N(p_i, c \cdot p_i))$ ($i \in \mathcal{J}$, c determine the disturbance level and will be defined later) while due dates d_i and weights w_i are deterministic. Then, completion times $\tilde{C}_i$ are random variables

$$\tilde{C}_i \sim N\left(p_1 + p_2 \ldots + p_i, c \cdot \sqrt{p_1^2 + \ldots + p_i^2}\right). \tag{2}$$

and the costs are random variables

$$\tilde{T}_i = \max\{0, \tilde{C}_i - d_i\}$$

For each permutation $\pi \in \Pi$ the cost in the random model is defined as a random variable

$$\widetilde{W}(\pi) = \sum_{i=1}^{n} w_i \tilde{T}_i.$$

3.2. Random due dates

Random due dates are represented by random variables with the normal distribution $\tilde{d}_i \sim N(d_i, c \cdot d_i)$, $i \in \mathcal{J}$ while processing times p_i and weights w_i are deterministic. Completion times are the same as in the processing times variant and the costs are random variables

$$\tilde{T}_i = \max\{0, C_i - \tilde{d}_i\}$$

Again, cost functions are the same as for the variant with random processing times.

3.3. Comparison functions

As in the Tabu Search implementation we need a possibility to compare 2 candidate solution, we introduce the following comparison function for both variants

$$W(\pi) = \sum_{i=1}^{n} w_i E(\tilde{T}_i).$$

. Appropriate formulas for $E(\tilde{T}_i)$ has been derived in [1]:

Theorem 1. ([1]) *If the task completion times are independent random variables normally distributed* $\tilde{p}_i \sim N(p_i, c \cdot p_i)$ $(i = 1, 2, \ldots, n)$ *, then*

$$E(\tilde{T}_i) = (1 - F_{\tilde{C}_i}(d_i)) \left(\frac{\sigma}{\sqrt{2\pi}} e^{\frac{-(d_i-\mu)^2}{2\sigma^2}} + (\mu - d_i) \left(1 - F_{N(0,1)}(\frac{d_i - \mu}{\sigma}) \right) \right).$$

Theorem 2. ([1]) *If the expected due dates are independent random variables normally distributed* $\tilde{d}_i \sim N(d_i, c \cdot d_i)$*, then*

$$\begin{aligned} E(\tilde{T}_i) &= F_{N(0,1)} \left(\frac{C_i - \mu}{\sigma} \right) \left(C_i F_{N(0,1)} \left(\frac{C_i - \mu}{\sigma} \right) \right. \\ &\quad \left. + \frac{\sigma}{\sqrt{2\pi}} e^{-\frac{(C_i-\mu)^2}{2\sigma^2}} - \mu F_{N(0,1)} \left(\frac{C_i - \mu}{\sigma} \right) \right). \end{aligned}$$

4. Disturbed data and its analysis

The considered approach for robust optimization was considered in several papers ([1], [2], [3], [4], [6], [11], [13]). One of the key assumptions was the input data coming with a very specific probabilistic distribution. In majority of considered cases we investigated data generated from the normal distribution with a specific mean μ and standard deviation σ (as a fraction of μ). In all considered variants on average the proposed solution offered better results than the method based on deterministic approach.

In this paper we investigate how much the model and based on it Tabu Search implementation is vulnerable on deviations from the assumed distribution parameters.

Baseline test instances come from OR-Library ([9]) where there are 125 examples for $n = 40$, 50 and 100 (in total 375 examples). All the disturbed data has been generated targeting a specific problem variant, i.e. for each OR Library instance 100 disturbed data instances have been generated assuming: **(a)** problem parameter which is uncertain, i.e. either processing times p_i or due dates d_i, **(b)** disturbance level c introduced in Sections 3.1. and 3.2. which takes the following values $c \in \{0.02, 0.04, 0.06, 0.08, 0.1, 0.15, 0.2, 0.25, 0.3\}$. On top of that we also generated data with the uniform distribution where each disturbed value comes from the range $[0.8x, 1.2x]$ where x is the considered uncertain variable (i.e. either p_i or d_i).

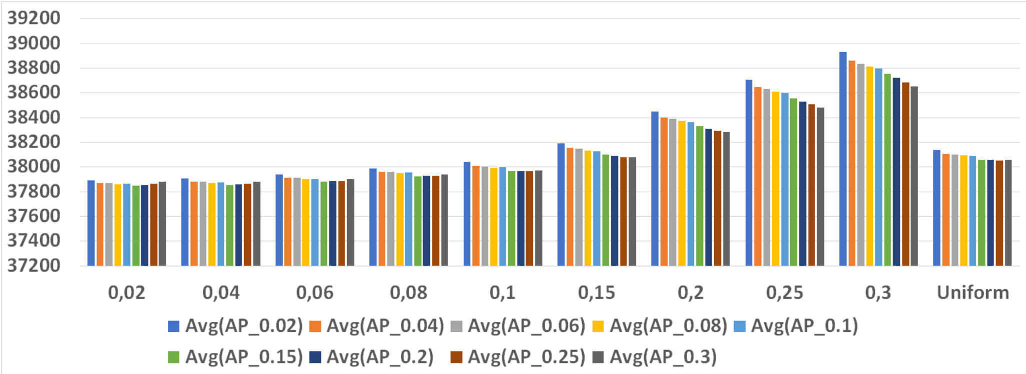
In our analysis we take a data set for a specific disturbance context, i.e. problem variant (uncertain p_i or d_i) and disturbance level (normal distribution with c value or uniform distribution) and then execute tabu search algorithm 9 times each time configured for a different disturbance value c. As a result we get a Table 1 with raw data of the following structure: N refers to the number of tasks, *Factor* refers to the data disturbance level (parameter c), *Task No.* refers to the OR Library instance number (1-125), *Dist. Item* refers to the disturbed instance number for a given OR library instance (1-100), and the rest of columns refer to outcome of execution of the algorithm configured for a certain disturbance level.

The goal is to determine for a considered method if and how much configuring algorithm for a specific method make a difference in obtained results. For instance, having a data disturbed a little like for $c = 0.02$, if there is difference between executing algorithm configured for $c = 0.02$ and algorithm configured for $c = 0.3$. In our analysis we check all those combinations and in this paper we do that on an aggregated level, i.e. we consider all values for given n and disturbance level as a comparison baseline.

Tabela 1

Raw result data for each n, distribution, instance number and disturbed item number

N	Factor	Task No.	Dist. Item	$AP_{.02}$	$AP_{.04}$	$AP_{.06}$	$AP_{.08}$	$AP_{.1}$	$AP_{.15}$	$AP_{.2}$	$AP_{.25}$	$AP_{.3}$
40	0.02	0	0	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}	v_{16}	v_{17}	v_{18}	v_{19}
40	0.02	0	1	v_{21}	v_{22}	v_{23}	v_{24}	v_{25}	v_{26}	v_{27}	v_{28}	v_{29}
40	...	...	...	...	...	...	...	...	...	...	...	...



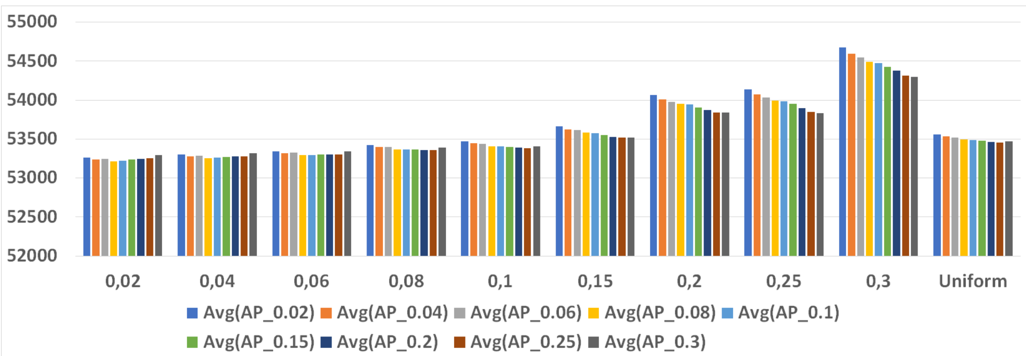
Rys. 1. Average values for each disturbance data group, random p_i, $n = 40$

5. Computational Experiments

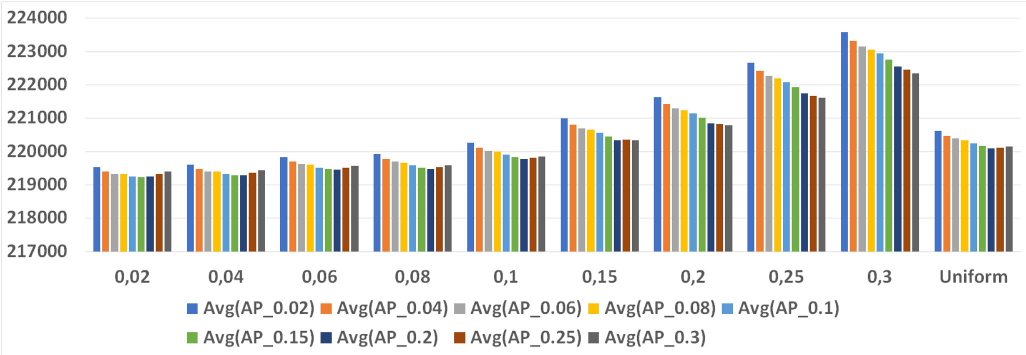
All the tests are executed on the data described in Section 4. and using standard tabu search method implementation with small adjustments related to the way how the two candidate solutions are compared. Due to the limited space in this paper we skip the algorithm description including configuration parameters and refer to [1] for more details.

5.1. Results

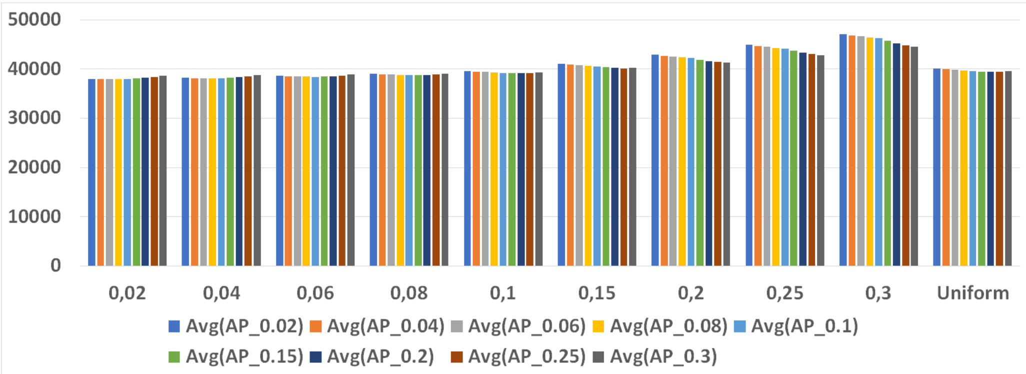
On Figures 1, 2, 3, 4, 5, 6 we can observe visual representation of the obtain results. On each figure we can see 10 groups for each disturbance data set (i.e. for each disturbance level c and uniform distribution). Within each group we can see 9 values representing execution of the algorithm configured for all 9 different disturbance levels, i.e. all c values.



Rys. 2. Average values for each disturbance data group, random p_i, $n = 50$

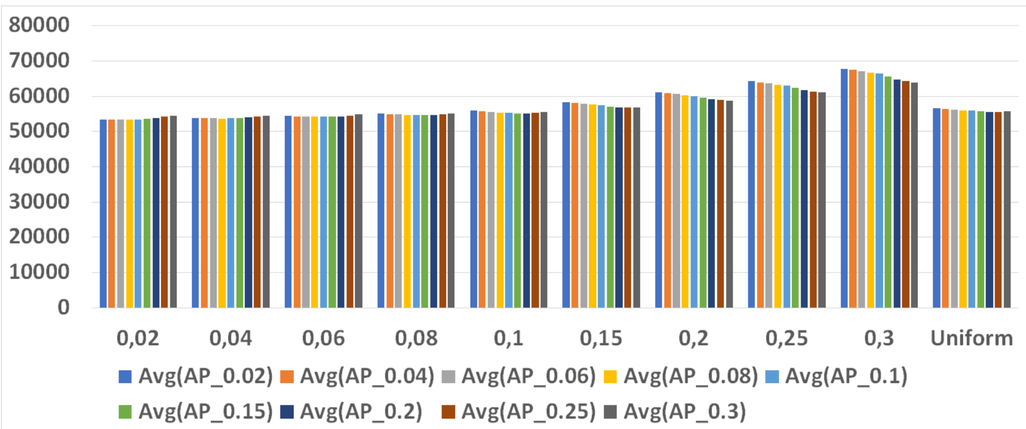


Rys. 3. Average values for each disturbance data group, random p_i, $n = 100$

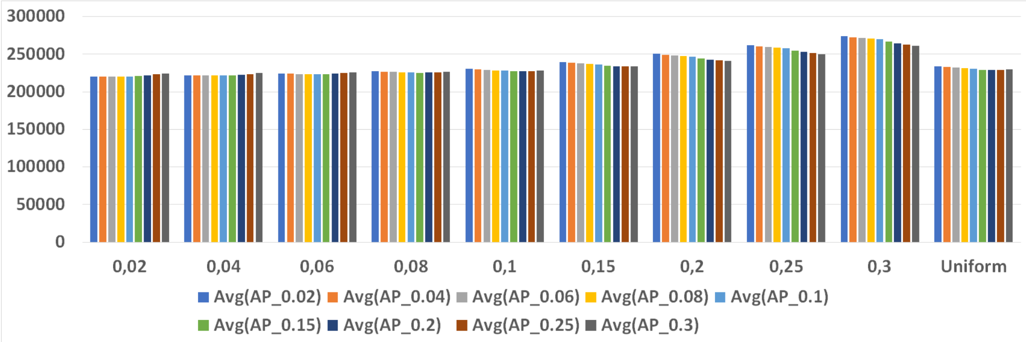


Rys. 4. Average values for each disturbance data group, random d_i, $n = 40$

We can immediately observe that the algorithm configured for a specific disturbance level (or close to) on average gives better results than the algorithm configured for other disturbance level, however more meaningful difference we can observe for disturbance level from 0.15. We can also see that the effect is much stronger for random p_i than random d_i what is somehow interesting as this is the opposite than for the problem variant described in [5]. On the other hand for smaller disturbances (0.02–1.0) values are



Rys. 5. Average values for each disturbance data group, random d_i, $n = 50$



Rys. 6. Average values for each disturbance data group, random d_i, $n = 100$

rather similar, however, there is no case where for the small disturbances (0.02–0.06) the tailored algorithm is the best choice, in most of cases it is algorithm designed for a little bigger disturbance levels like 0.08 or 0.1.

6. Conclusions

In this paper we solved a single machine scheduling problem with a tabu search method configured for defined several disturbance levels of the normal distribution and uniform distribution. The investigation is a continuation of the previous work where different problem variant was considered. The obtained results show that for bigger disturbance levels the closer the considered robust optimization method is configured to the disturbance level of the data set, the better results it provides. Even though the overall trend has been observed in all considered problem variants, the actual results depend on the disturbance levels and the considered uncertain parameter what has been presented on the respective diagrams.

The results of the analysis are promising and interesting and we plan to continue further the investigation by performing more deep analysis and continue investigating other problems and target functions.

REFERENCES

1. Bożejko, W., Rajba, P., & Wodecki, M. (2014). Stable Scheduling with Random Processing Times. In Advanced Methods and Applications in Computational Intelligence (pp. 61–77). Springer, Heidelberg.

2. Bożejko, W., Rajba, P., & Wodecki, M. (2017). Stable scheduling of single machine with probabilistic parameters. Bulletin of the Polish Academy of Sciences Technical Sciences, 65(2), 219–231.

3. Bożejko, W., Rajba, P., & Wodecki, M. (2018, July). Robustness of the Uncertain Single Machine Total Weighted Tardiness Problem with Elimination Criteria Applied. In International Conference on Dependability and Complex Systems (pp. 94–103). Springer, Cham.

4. Bożejko W., Rajba P., Wodecki M. (2020) Robust Single Machine Scheduling

with Random Blocks in an Uncertain Environment. In: Krzhizhanovskaya V. et al. (eds) Computational Science – ICCS 2020. ICCS 2020. Lecture Notes in Computer Science, vol 12143. Springer, Cham

5. Bożejko, W., Rajba, P., & Wodecki, M. (2022). Robustness on Diverse Data Disturbance Levels of Tabu Search for a Single Machine Scheduling. In International Conference on Dependability and Complex Systems (pp. 1-10). Springer, Cham.
6. Bożejko, W., Rajba, P., Uchroński, M., & Wodecki, M. (2021, June). A Job Shop Scheduling Problem with Due Dates Under Conditions of Uncertainty. In International Conference on Computational Science (pp. 198-205). Springer, Cham.
7. Cai, X., Wu, X., & Zhou, X. (2014). Optimal stochastic scheduling (Vol. 4). New York: Springer.
8. Dean, B. C. (2005). Approximation algorithms for stochastic scheduling problems (Doctoral dissertation, Massachusetts Institute of Technology).
9. OR-Library `http://www.brunel.ac.uk/~mastjjb/jeb/info.html`
10. Pinedo, M. L. (2016). Scheduling: Theory, Algorithms, and Systems, Springer International Publishing.
11. Rajba, P., & Wodecki, M. (2012). Stability of scheduling with random processing times on one machine. Applicationes Mathematicae, 2(39), 169-183.
12. Rajba, P., & Wodecki, M. (2017, June). Sampling Method for the Flow Shop with Uncertain Parameters. In IFIP International Conference on Computer Information Systems and Industrial Management (pp. 580-591). Springer, Cham.
13. Rajba, P. (2021, June). Sampling Method for the Robust Single Machine Scheduling with Uncertain Parameters. In International Conference on Computational Science (pp. 594-607). Springer, Cham.
14. Soroush, H. M. (2013). Scheduling stochastic jobs on a single machine to minimize weighted number of tardy jobs. Kuwait Journal of Science, 40(1), 123–147.
15. Urgo, M., & Váncza, J (2019). A branch-and-bound approach for the single machine maximum lateness stochastic scheduling problem to minimize the value-at-risk. Flexible Services and Manufacturing Journal 31, 472-–496.
16. Vondrák, J. (2005). Probabilistic methods in combinatorial and stochastic optimization (Doctoral dissertation, Massachusetts Institute of Technology).
17. Zhang, L., Lin, Y., Xiao, Y., & Zhang, X. (2018). Stochastic single-machine scheduling with random resource arrival times. International Journal of Machine Learning and Cybernetics, 9(7), 1101–1107.