



HAL
open science

Automatic Configuration of a Dynamic Hill Climbing Algorithm

Weerapan Sae-Dan, Marie-Eléonore Kessaci, Nadarajen Veerapen, Laetitia Jourdan

► **To cite this version:**

Weerapan Sae-Dan, Marie-Eléonore Kessaci, Nadarajen Veerapen, Laetitia Jourdan. Automatic Configuration of a Dynamic Hill Climbing Algorithm. SLS - International Workshop on Stochastic Local Search Algorithms, Sep 2019, Villeneuve d'Ascq, France. hal-02430770

HAL Id: hal-02430770

<https://hal.science/hal-02430770v1>

Submitted on 11 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Automatic Configuration of a Dynamic Hill Climbing Algorithm

Weerapan Sae-dan¹, Marie-Eléonore Kessaci¹, Nadarajen Veerapen¹, and Laetitia Jourdan¹

Université de Lille, CNRS, UMR 9189 - CRISTAL 59000 Lille, France
 {weerapan.saedan,mkessaci,nadarajen.veerapen,laetitia.jourdan}@univ-lille.fr

1 Introduction

In this paper, we propose an automatic algorithm configuration (AAC). It can manage many parameters and an online method produces a deterministic algorithm which parameters modified during execution. We explore the benefits of both methods by proposing a dynamic framework that switches between different configurations during execution and adapted AAC protocol. We define a control mechanism for parameters as a combination of three questions [1]: what is to be controlled, how the control is performed, and when the parameter is changed? Our method is tested on single-objective permutation scheduling flowshop problem with an automatic configuration of a dynamic hill-climbing.

2 Automatic Algorithm Configuration of Dynamic Framework

Our contribution is to propose a new method to parameterize a dynamic algorithm. To describe what is called a dynamic framework, C. Pageau et al. [2] propose using parameter configuration that changes at predetermined times, as show in figure 1. We will propose more flexibility on the execution times for each configuration. This concept enables the use of AAC instead of the parameter control mechanisms.

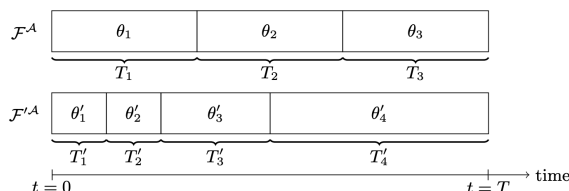


Fig. 1: Example of two configuration schedules [2]

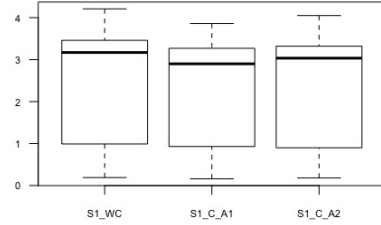
In this paper, the automatic configuration can fix the duration of each time split (s_i) in the list $\{10, 25, 50, 75, 90\}$ and each time split is calculated as follows. T_1 is equal to $(T \times s_1)/100$, T_2 is equal to $((T - T_1) \times s_2)/100$, T_i is equal to $((T - \sum(T_i - T_{i-1})) \times s_i)/100$, and T_k is the remaining budget where T is the time budget.

3 Experiments and Conclusion

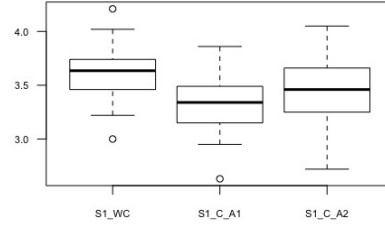
We investigated the proposed method to solve the single-objective permutation flowshop scheduling problem. We use Irace [3] to implement the AAC and to find the configuration of hill-climbing [4] and its components (exploration neighborhood, operation neighborhood, neighborhood order, and perturbation) best adapted to the instances of the problem to solve. We use the Taillard instances [5], we use different instance of N jobs, M machines: 20×5 , 20×10 , 20×20 , 50×5 , 50×10 , 50×20 , 100×5 , 100×10 , 100×20 , 200×10 , and 200×20 . For each size, 10 instances are used.

In this paper, two dynamic framework parameters have been implemented: K as the number of time splits and $\{T_1, T_2, \dots, T_K\}$ the associated time budget. 3 time budget configurations are

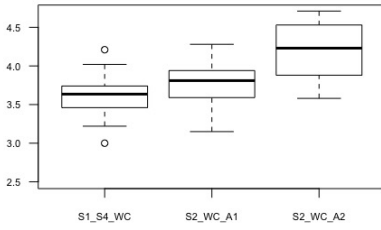
possible, corresponding to the different time splits $\{T\}$, $\{T_1, T_2\}$ and $\{T_1, T_2, T_3\}$. 24 configurations for our algorithm are present per time split. We test 3 scenarios in which K is equal to 1, 2 or 3. Since $|s| = |\{10,25,50,75,90\}| = 5$, the number of time configurations possible are 1, 5 and 25 for $K = 1, 2$ and 3 respectively. If $K = 1$, then there are 24 different configurations, $K = 2$ includes approximately 2.9×10^3 different configurations ($24 + 5 \times 24^2$) and, $K = 3$ a total of approximately 3.5×10^5 different configurations ($24 + 5 \times 24^2 + 25 \times 24^3$). We set to $N^2 \times M$ milliseconds, the stopping criterion of the algorithm.



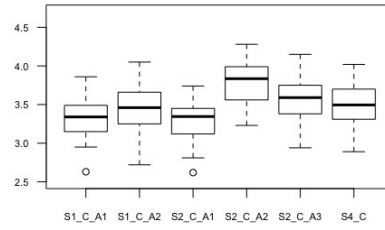
(a) all instances



(b) without and with control



(c) without parameter control



(d) with parameter control

Fig. 2: Instances 100 jobs and 20 machines. C-with control, WC-without control, S_1 = all instances, S_{2-4} -subset of instances, A_{1-4} -specific parameter configuration. Smaller values are better.

For the comparison between with control and without control, the experiments presented in figure 2 (a)(b) show that the value of control. Moreover, the experiments presented in (c)(d) show that it is more interesting without control to learn from all instances while with control performance is better with learning on instances of the same size. We hypothesize that parameter control makes the algorithm better adapted to the local properties of the instance.

References

1. A.E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms.3(2):124141,1999.
2. Camille Pageau, Aymeric Blot, Holger Hoos, Marie-Elonore Kessaci, and Laetitia Jourdan. Configuration of a Dynamic MOLS Algorithm for Bi-objective Flowshop Scheduling: 10th International Conference, EMO 2019, East Lansing, MI, USA, March 10-13, 2019, Proceedings, pages 565577. 01 2019.
3. López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M., and Stützle, T. The irace package: Iterated racing for automatic algorithm configuration. Operations Research Perspectives, 3:43–58, 2016.
4. Cohen, W., Greiner, R., Schuurmans, D. Probabilistic Hill-Climbing. Computational Learning Theory and Natural Learning Systems, Vol. II. Edited by Hanson, S., Petsche, T., Rivest R., Kearns M. MITCogNet, Boston:1994.
5. Taillard, E. Benchmarks for basic scheduling problems. European Journal of Operational Research, 64(2):278285,1993.