The optimal filtering set problem with application to surrogate evaluation in Genetic Programming

Francisco J. Gil-Gala
University of Oviedo. Department of Computing
Gijón, Spain
giljavier@uniovi.es

María R. Sierra
University of Oviedo. Department of Computing
Gijón, Spain
sierramaria@uniovi.es

Carlos Mencía
University of Oviedo. Department of Computing
Gijón, Spain
menciacarlos@uniovi.es

Ramiro Varela*
University of Oviedo. Department of Computing
Gijón, Spain
ramiro@uniovi.es

ABSTRACT

Surrogate evaluation is common in population-based evolutionary algorithms where exact fitness calculation may be extremely time consuming. We consider a Genetic Program (GP) that evolves scheduling rules, which have to be evaluated on a training set of instances of a scheduling problem, and propose exploiting a small set of low size instances, called filter, so that the evaluation of a rule in a filter estimates the actual evaluation of the rule on the training set. The calculation of filters is modelled as an optimal subset problem and solved by a genetic algorithm. As case study, we consider the problem of scheduling jobs in a machine with time-varying capacity and show that the combination of the surrogate model with the GP termed SM-GP, outperforms the original GP.

CCS CONCEPTS

• **Applied computing → Transportation**; • **Computing methodologies → Planning for deterministic actions**; • **Theory of computation → Design and analysis of algorithms**.

KEYWORDS

Evolutionary computation, Surrogate models, Scheduling, Hyper-heuristics

ACM Reference Format:
Francisco J. Gil-Gala, María R. Sierra, Carlos Mencía, and Ramiro Varela. 2021. The optimal filtering set problem with application to surrogate evaluation in Genetic Programming. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3449726.3459484

1 INTRODUCTION

In many population-based evolutionary algorithms, fitness evaluation is the most time consuming part. For this reason, surrogate evaluation models are often used to improve their performance [1, 2, 5, 6].

In this work, we consider the Genetic Programming (GP) approach developed in [3] to evolve priority rules for the problem of scheduling jobs in a machine with variable capacity over time, denoted $(1, Cap(t)||\sum T_i)$. In this GP, the evaluation of a candidate rule requires solving a set of instances of the $(1, Cap(t)||\sum T_i)$ problem, the *training set*. For the evaluation to be accurate, the number of instances in the training set must be sufficiently large; therefore, the fitness evaluation may take very long time. In order to reduce this time, we devised a surrogate model. Given a training set, our proposal relies on the hypothesis that it is possible to calculate another set, called *filter*, containing just a few instances, even of lower size than those in the training set, so that the performance of a priority rule on this filter may predict the performance of the rule on the training set. More specifically, given two candidate rules a and b, if a is better than b in the reduced set, then it is likely that a will be better than b in the training set as well.

2 CALCULATION AND EXPLOITATION OF FILTERS

The problem of calculating filters may be formulated as a variant of the optimal subset problem as follows:

Given are

- A problem $\mathcal{P}$ and an ordered set of heuristics $H = \{h_1, \ldots, h_n\}$ to solve $\mathcal{P}$. Two ordered sets $R = \{R_1, \ldots, R_r\}$ and $S = \{S_1, \ldots, S_s\}$ of instances of $\mathcal{P}$. X_{ij}, Y_{ij}, $1 \leq i \leq n$, $1 \leq j \leq r$, the performance measures of heuristic h_i on the sets R_j and S_j respectively. A parameter $k > 0$.

The goal is to find a subset $F \subset S$, $F = \{S_{[1]}, \ldots, S_{[k']}\}$, $k' \leq k$, where $[i]$, $1 \leq [i] \leq s$, $1 \leq i \leq k'$, is the index of the ith instance of F in S, such that

- The correlation between the paired observations $X = \{X_1, \ldots, X_n\}$ and $Y = \{Y_1, \ldots, Y_n\}$, where

$$X_i = \sum_{j=1,\ldots,r} X_{ij}, Y_i = \sum_{[j]=1,\ldots,k'} Y_{i[j]}, \quad (1)$$

  is maximized and k' is minimized.

*Corresponding author

GECCO '21 Companion, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8351-6/21/07.
https://doi.org/10.1145/3449726.3459484

These two objective functions are considered hierarchically in the order they are declared, i.e., we will try to maximize the correlation and only in the case of ties we will prefer the set F with lowest cardinality.

In this work, we opted to use the Kendall Tau-b (τ_b) coefficient to measure the correlation, given its robustness and that it has a direct interpretation in terms of probability of observing concordant or discordant pairs of observations.

The above problem was solved by means of a Genetic Algorithm (GA). The filters calculated by the GA are then exploited in combination with the GP to speed up the evaluation process. To do that, we explored different ways of combining surrogate and exact evaluation. The best of these combinations consists in generating a number of N offspring from each pair of chromosomes selected for mating; the offspring are evaluated with the surrogate method and only the best one, in accordance with the simplified evaluation, is evaluated on the training set. The method is termed SM-GP.

3 EXPERIMENTAL STUDY

In this study, our aim is to analyze the performance of the GA in solving the OFSP and then to study how the calculated filters may improve the performance of SM-GP w.r.t. the GP. All the algorithms were implemented in Java 8 language and the target machine was a Linux cluster (Intel Xeon 2.26 GHz. 128 GB RAM. 28 nodes). We consider the training and test sets of instances of the $(1, Cap(t)||\sum T_i)$ problem proposed in [4], which include 50 and 1000 instances respectively. These are large instances with 60 jobs each and maximum capacity of the machine of 10 jobs. We also consider another set of 1000 small instances (10 jobs and maximum capacity of the machine of 3) generated by the same procedure. The training set plays the role of the set R in the GA, while the set S is given by the small instances. The test set is only used to evaluate the rules evolved by the GP and SM-GP on unseen instances. The maximum size of the filters is $k = 5$; in this way, the ratio between the time taken to evaluate a candidate rule on the set R and on one filter is about 60/1. The set H includes 600 rules with different characteristics obtained previously by the GP.

Table 1 shows results from the GA compared to random filters; as we can observe, the filters calculated by the GA are much more stable and they are able to reduce the gap between random filters and *perfect filters* (with ideal coefficient of 1.0) in more than 50%.

The results of the best filter combined with the GP may be appreciated in Figure 1. In these results, the GP and SM-GP were given 300 minutes and were run 28 times each. In SM-GP, $N = 50$. We can see that SM-GP outperforms the GP in best and average tardiness and standard deviation, in both training and test sets. These differences were confirmed by Mann-Whitney U tests, showing $p-$values 1.03E-05 (two sided) and 2.05E-05 (less) in the training set, and 1.93E-09 (two sided) and 9.58E-10 (less) in the test set. Besides,

Table 1: Filters generated by the GA vs. random ones ($k = 5$).

	Best	Avg.	Worst	SD
GA	0.8839	0.8834	0.8828	0.0004
Random	0.7862	0.7037	0.5884	0.0502

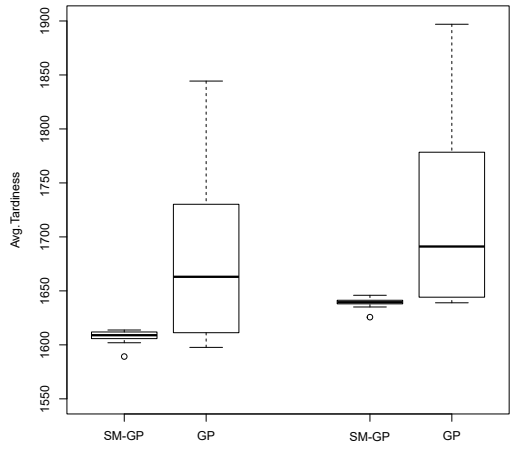

Figure 1: Box plots from the results from SM-GP and GP on the training (left) and test (right) sets.

the rules evolved by SM-GP were slower than those evolved by the GP.

4 CONCLUSIONS AND FUTURE WORK

We have seen that the proposed surrogate model (SM) based on small subsets of simple instances of a scheduling problem, called filters, may be effective to evolve priority rules via Genetic Programming (GP) to solve a scheduling problem. Besides, effective filters may be calculated by a Genetic Algorithm (GA) from a large set of simple problem instances. This work leaves open some interesting lines for further research as for example how to obtain simplified instances bearing resemblance with the training set, how to devise new strategies to exploit the surrogate model in combination with GP or even with other methods as local search, or how to apply the proposed method to other scheduling problems.

ACKNOWLEDGMENTS

Research supported by the Spanish Government (TIN2016-79190-R, PID2019-106263RB-I00, FPI17/BES-2017-08203).

REFERENCES

[1] Maumita Bhattacharya. 2008. Reduced computation for evolutionary optimization in noisy environment. In *GECCO'08: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation 2008.* 2117–2122. https://doi.org/10.1145/1388969.1389033

[2] J. Branke and C. Schmidt. 2005. Faster Convergence by Means of Fitness Estimation. *Soft Comput.* 9, 1 (Jan. 2005), 13âĂŞ20. https://doi.org/10.1007/s00500-003-0329-4

[3] Francisco J. Gil-Gala, Carlos Mencía, María R. Sierra, and Ramiro Varela. 2019. Evolving priority rules for on-line scheduling of jobs on a single machine with variable capacity over time. *Applied Soft Computing* 85 (2019), 105782.

[4] Francisco J. Gil-Gala, María R. Sierra, Carlos Mencía, and Ramiro Varela. 2020. Combining hyper-heuristics to evolve ensembles of priority rules for on-line scheduling. *Natural Computing* (2020).

[5] Rayan Hussein and Kalyanmoy Deb. 2016. A Generative Kriging Surrogate Model for Constrained and Unconstrained Multi-Objective Optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016* (Denver, Colorado, USA) *(GECCO '16)*. Association for Computing Machinery, New York, NY, USA, 573âĂŞ580. https://doi.org/10.1145/2908812.2908866

[6] Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane, and K. Y. Lum. 2007. Combining Global and Local Surrogate Models to Accelerate Evolutionary Optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37, 1 (2007), 66–76. https://doi.org/10.1109/TSMCC.2005.855506