# Parameter Space Analysis of Genetic Algorithm Using Support Vector Regression

Hwi-Yeon Cho Dept. Comp. Sci., Kwangwoon Univ. Seoul, Republic of Korea hwyn.cho@gmail.com Hye-Jin Kim Dept. Comp. Sci., Kwangwoon Univ. Seoul, Republic of Korea ovoa22@gmail.com Yong-Hyuk Kim Dept. Comp. Sci., Kwangwoon Univ. Seoul, Republic of Korea yhdfly@kw.ac.kr

# ABSTRACT

Tuning various parameters coexisting in genetic algorithms (GAs) has a direct impact on the performance of GA. Because of this, finding a proper parameter value is challenging. In this study, we use support vector regression to show the appropriate parameter space of GA. Moreover, this was applied and analyzed to solving *NK*-landscape problems. As a result, we show the complexities and difficulties of GA parameter space through this paper.

# **CCS CONCEPTS**

• Computing methodologies  $\rightarrow$  Genetic algorithms; *Machine learning*;

# **KEYWORDS**

genetic algorithm, machine learning, parameter space

#### **ACM Reference Format:**

Hwi-Yeon Cho, Hye-Jin Kim, and Yong-Hyuk Kim. 2018. Parameter Space Analysis of Genetic Algorithm Using Support Vector Regression. In *GECCO* '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3205651.3208770

## **1** INTRODUCTION

Genetic algorithms (GAs) are a typical approach in evolutionary computation and are a representative meta-heuristic approach. The key operators of the GA are selection, crossover, mutation, and replacement. By properly organizing the operators, the performance of GA can be moderated. Therefore, proper tuning of GA parameters can be considered as the most important part in building a GA. There has been much research on parameter optimization. Research that attempted to automate GA tuning using meta-genetic algorithms has been very popular following the first attempt by Grefenstette [1]. There was also an early attempt similar to Grefenstette's using meta-GA. Lee and Takagi [2] used the GA and penalty strategies and proposed a method to design an automated fuzzy system. Until recently, research on parameter optimization using meta-GA was actively conducted.

As in this paper, there has been research using machine learning in GA parameter optimization. Cooray and Rupasinghe [3] used

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan © 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5764-7/18/07.

https://doi.org/10.1145/3205651.3208770

machine learning to solve the energy minimizing vehicle routing problem. They bundled the data using k-means clustering method, and they categorized data based on basic characteristics. The GA parameters were chosen based on the various tuning methods, which resulted in outstanding performance.

We intend to take a new approach in GA parameter optimization using machine learning. We would like to apply our approach to the *NK*-landscape problem. The objective of this study is to train with data consisting of a GA parameter set using support vector regression (SVR) and an approximate GA parameter space. In addition, based on the approximation results, we intend to analyze the GA parameter space.

# 2 PARAMETER SPACE OF GA & SVR APPROXIMATION

#### 2.1 Parameters of GA

The GA parameters are presented in this section. The relevant GA parameters are selection, crossover, mutation, and replacement. Moreover, tuning parameters of GA can be seen as moderating these operators. The parameters to be covered in this research are as follows.

• **Population size:** the population size was a randomly generated number between 10 and 5, 000.

• Selection type: when moving from the existing generation to the next generation, one has to choose a population. Four selection methods were used—roulette wheel selection, scaled roulette wheel selection, tournament selection, and rank-based selection.

• **Crossover type:** we used *n*-point crossover and uniform crossover. Moreover, when using *n*-point crossover, the number of points ranged from 1 to 5.

• **Mutation rate:** the mutation rate ranged from 0 to 0.25, and random numbers were generated and used per 0.001 unit.

• **Replacement type:** we used 4 methods, which were replacement of the worst individuals in the population, replacement of similar individuals in the population, replacement of worst individuals among the parents, and the replacement of similar individuals among the parents.

### 2.2 SVR Approximation

SVR is a supervised learning model used to analyze regression. By using SVR, the GA parameter space was approximated as shown in Figure 1. For SVR modeling, 100 data points generated f(x), and 10 arbitrarily chosen points were extracted to train f(x). Using the SVR model built upon this process, we generated 100 points and approximated the original f(x). Figure 1(a) demonstrates a rather simple problem, and the SVR model could make up the

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).



(a) Well approximated f(x) (b) Badly approximated f(x)

Figure 1: Examples of function approximation by SVR

trends of changes highly similar to the original data with just 10 data points. On the other hand, Figure 1(b) illustrates rather more complex functions. The trained SVR model did not yield the best approximations of the original function.

## **3 RESULTS & ANALYSIS**

The SVR model presented in Section 2 was applied to the NKlandscape problem. The value of N in the NK-landscape problem was 50, and K was 4. The fitness contribution was arbitrarily generated from real numbers ranging between 0 and 1. Moreover, we use steady-state GA with a generation size of 10,000. Table 1 describes the performance of 10,000 best fitness data and SVR model according to the parameters of GA. Table 2 shows the results obtained by finding 100 optimal points of SVR model using GA\*1 and performing GA with the optimal points. The parameters of the 100 optimal points in SVR model identified by GA\* can be delineated as follows. The scaled roulette wheel selection method one-point crossover method, and similar individual replacement method were used. The best fitness results showed differences according to the size of population and mutation rate. The performance of the parameter set found was on average top 11.7%. In addition, one point was equal to 37.452, which was the best result in 10,000 best fitness data. Accordingly, although we could not find the parameter better than the one from 10,000 datasets, we could arrive at a relatively proper parameter set. The mean absolute error (MAE) of the dataset from 10-fold cross-validation was approximately 0.478. The difference between the optimal points of SVR model and actual results of GA was approximately 2.101, which was a considerable difference.

Figure 2 shows the best fitness results for 100 optimal points in the SVR model according to population size and mutation rate of GA. Figure 2(a) shows a 3D view of the best fitness over the two parameters, and Figure 2(b) presents a cross-sectional view of Figure 2(a). As can be seen in Figure 2(a), the best fitness values are extremely vulnerable to minor changes in the parameters. From these, one can realize that the GA parameter space is structured in a complex way. As a result, due to the complexities of the GA parameter space, it is not simple to arrive at optimal GA parameters with machine learning.

Table 1: SVR model results from the 10,000 dataset

	Best	Average	$SD^2$
Best fitness	37.452	33.316	1.707
SVR model	37.873	33.280	1.581

Table 2: Actual GA results and 100 optimal points of model

	Best	Average	SD
SVR model	37.965	37.955	0.008
GA	37.452	35.836	0.726



Figure 2: GA results according to parameters (using Plotly)

## 4 CONCLUSION

Machine learning has applications in various fields. Parameter optimization in GA is an important factor to determine the GA performance. In this study, we proposed an approach that applies machine learning to the GA parameter space. By using SVR, we trained the data consisting of the GA parameter set and built models. In addition, the SVR model was applied to the NK-landscape problem, and tried to find the optimal parameter. The identified parameter set showed performance of the top 11.7% on average, which can be considered as an outstanding set, but this was still not as good as the one from the dataset. As well, based on the findings from SVR approximation in the GA parameter space, GA performance is vulnerable to minor changes in the parameters. From these findings, we confirmed the challenges facing the optimization of GA parameters using machine learning. In conclusion, we could understand the complexities of the GA parameter space. The research findings will contribute to future research on GA parameter optimization.

#### ACKNOWLEDGMENTS

This research was supported by a grant [KCG-01-2017-05] through the Disaster and Safety Management Institute funded by Korea Coast Guard of Korean government. This research was also supported by the MIST (Ministry of Science and ICT), Korea, under the National Program for Excellence in SW supervised by the IITP (Institute for Information & communications Technology Promotion)(2017-0-00096).

#### REFERENCES

- J. J. Grefenstette, "Optimization of control parameters for genetic algorithms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 122–128, 1986.
- M. A. Lee and H. Takagi, "Integrating design stage of fuzzy systems using genetic algorithms," in *IEEE International Conference on Fuzzy Systems*, pp. 612–617, 1993.
  P. Cooray and T. D. Rupasinghe. "Machine learning-based parameter tuned genetic
- [3] P. Cooray and T. D. Rupasinghe, "Machine learning-based parameter tuned genetic algorithm for energy minimizing vehicle routing problem," *Journal of Industrial Engineering*, vol. 2017, 2017.

<sup>&</sup>lt;sup>1</sup>GA\* is a GA used to find the optimal point of the SVR model, so it is a steady-state GA with a generation size of 20, 000. The population size is 1, 000, the chromosomes of each individual are the 5 parameter values used in Section 2.1, which are integer encoding. This GA\* uses rank-based selection, uniform crossover, one point mutation, and worst individual replacement, and searches for the optimal point in an SVR model. <sup>2</sup>Standard deviation