Improvement of Evolutionary Computation of Genetic Algorithm using SVM

Byungsun Cho, Sohee Han, Sung-Han Son, Jinsu Kim, and Kang-bak Park

Department of Control and Instrumentation Engineering, Korea University, Korea
(Tel : +82-2-922-0863; FAX : +82-2-922-0863; E-mail: kbpark@korea.ac.kr)

Abstract: Genetic algorithm is well known as a stochastic searching method. In this paper, a modified genetic algorithm using 'Suppor Vector Machines (SVM)' is proposed. SVM is used to reduce the number of calling the objective function which in turn accelerate the searching speed compared to the conventional GA.

Keywords: Genetic algorithm(GA), Support vector machines(SVM), Evolutionary computation speed, SVMGA.

1. Introduction

For past 30 years, people are interested in algorithms which are based on natural phenomena. Things which are widely known are Evolution Programming, Chaos theory, Neural network, Fuzzy system, etc. Among them, Genetic Algorithm (GA) is experimentally known as a useful tool which is able to search the optimal solution in a complex system. The beginning of GA was proposed by some biologists in the early 1950s, and organized by *Holland* in 1975.[2] Despite the fact the genetic algorithm, in general, lack a strong theoretical background, the experimental results were more than encouraging. Genetic algorithm acts well with only objective function to optimize without a prior knowledge, and does not need additional information such as continuity etc. Because of these special quality, GA has solved successfully for complicated problems which could not solve with existent algorithms. Nowadays, GA has been applied in various field according to the elevation of computation speed by development of supercomputer.

Support Vector Machine (SVM) is a recently developed learning method, and is is used for a powerful tool in intelligent system field.[13] This theory, which is now applied for a lot of fields connected with function regression and pattern classification, has much possibility to converge to the global optimal solution, and it is easy to apply for new input data. It has advantage that inner structure for necessary to learn is decided automatically, too.

In this paper, to obtain faster computation speed, a modified genetic algorithm using SVM is proposed. The proposed method will be called SVMGA. That is, GA is basically evaluated for fitness value, analyze genes that participate in evolution continuously according to this fitness value, and only good genes classified with SVM can participate in evolutionary process for next generation. Accordingly, evolutionary speed is improved.

2. Preliminary

2.1. Support Vector Machine (SVM)

2.1.1 Support vector learning method

Support Vector Machine is a new learning method that has been shown excellent performance in pattern classification and function regression problems. This method is used widely as a tool for speech recognition, image processing, statistical processing and many other fields rigorously.

Support vector learning method has Multi-layer perceptron (MLP) with one hidden layer and it has Radial basis function (RBF) network, so we get profits as following:

- The number of hidden nodes can be determined automatically.
- It is excellent in generalization capability, because its induction procedure is explained by statistical learning theory.
- Support vector means some data that decide optimal separating plane(or hyperplane) among training data.

2.1.2 Optimal separating hyperplane

The purpose of a pattern classification problem with two classes is to search a boundary that can divide into the two classes. Training data is given as following:

$$(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m), \tag{1}$$

where m is the number of data, $x \in R^n$ is input data, and $y \in \{-1, +1\}$ is output data. Training data are made up of subsets of $\{(x, y)|y = +1\}$ and $\{(x, y)|y = -1\}$. If these two subsets are linearly separable, they can be separable by the following hyperplane:

$$\langle w, x\rangle + b = 0, \tag{2}$$

where w is a weight vector and b is a bias such that

$$\begin{aligned} \langle w, x_i\rangle + b > 0 \quad &\text{for} \quad y_i = +1 \\ \langle w, x_i\rangle + b < 0 \quad &\text{for} \quad y_i = -1. \end{aligned} \tag{3}$$

These equations can be rewritten again as following:

$$y_i(\langle w, x_i\rangle + b) \geq 1, \quad i = 1, 2, \cdots, m. \tag{4}$$

There are many hyperplanes satisfying the equation (2). Thus, to find the optimal separating hyperplane, we define a margin as a distance between separating hyperplane and the nearest data from the hyperplane. The hyperplane which has a maximal margin becomes the optimal separating hyperplane. Support vector learning algorithm aims at finding this optimal separating hyperplane. Finding of this optimal separating hyperplane can be represented as a following problem:

$$\begin{aligned} &\min \frac{1}{2}||w||^2 \\ &\text{s.t.} \quad y_i(\langle w, x_i\rangle + b) \geq 1, \\ &\qquad i = 1, 2, \cdots, m. \end{aligned} \tag{5}$$

Equation (5) can be only applied to separate two classes perfectly. If they are not perfectly separable, input data in other class is penalized. So we select the optimal hyperplane that margin is maximized and penalties are minimized at the same time. That is, as we add to the new variable, the penalty term, the equation (5) is changed as following:

$$\begin{aligned} &\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i \\ &\text{s.t.} \quad y_i(\langle w, x_i\rangle + b) \geq 1 - \xi_i, \\ &\qquad \xi_i \geq 0, \quad i = 1, 2, \cdots, m, \end{aligned} \tag{6}$$

where C is regularization positive constant.

2.1.3 Kernel method

In general, lots of pattern classification problems have a nonseparable hyperplane. In such cases, kernel method can be used to find a separable hyperplane by mapping the input space to the feature space.

$$K(x_i, y_i) = \langle \Phi(x_i), \Phi(y_i)\rangle, \tag{7}$$

where K is a kernel function. RBF network, MLP network, polynomial function are used by representative kernel functions for mapping to feature space.

2.2. Genetic algorithm

A fundamental structure of genetic algorithm is that among individuals which is made up of one generation, some individuals which are adapt in environment and which are had a high fitness value can be alive with high probability. So, these individuals make offsprings for next generation using genetic operator - reproduction, crossover, and mutation etc. As this procedure do repeatedly, individuals converge a optimal solution gradually. In 1975, genetic algorithm was arranged and systemized by *Holland.*[2] After that, people have been vigorous to research about genetic algorithm and evolutionary computation. As a result, to find more suitable algorithm than conventional algorithm, various topics were occurred. The important topics of genetic algorithm are diversity problem, selection pressure, representation of solution parameter, genetic operator, search strategy, etc.

3. Proposal of Modified Genetic Algorithm

3.1. Modified simple genetic algorithm using SVM

In this paper, we would get exacter results than common method and improve computation speed applying SVM's method in conventional genetic algorithm. In other words, in case of simple genetic algorithm, basically, it has procedures - crossover and mutation - to produce offsprings. We adopted SVM in this procedures. Among reproducing individuals, they are reordered by fitness value. Weak individuals among them cannot be participated to operation. Also, criteria learning by SVM makes separating hyperplane. With this separating hyperplane, each individuals which is ending operation - crossover and mutation - is classified to some individuals that is expected GOOD and other individuals that is expected BAD. Only individuals that is expected GOOD are participated in next generation, so can diminish in load of whole plant and speed up evolutionary computation and search a exact solution.

Modified Simple Genetic Algorithm using SVM
Set k=0;
Create an initial population $P(k)$;
Evaluate $P(k)$;
While $\langle$ the termination conditions are not met $\rangle$
  Set k=k+1;
  Generate training data and do SVM;
  Reproduce mating pool $\bar{P}(k)$ from $P(k-1)$
    using roulette wheel selection;
  Crossover $\bar{P}(k)$ to form a tentative population $\tilde{P}(k)$;
  Mutate $\tilde{P}(k)$ to form the new population $P(k)$;
  Adopt the SVM criterion
    (Classify GOOD and BAD individuals);
  Evaluate P(k);
End while
Output the solution;

The first data evaluating populations become SVM's training data. That is, x_is are gene value in equation (1) and y_is are output data which are determined by fitness value . Classifier created by training data is applied for individuals about finishing genetic operation and it decides whether each individual takes part in next generation or not. Finally, the evolutionary speed can be improved because BAD genes are weeding out and only selected GOOD genes are able to participate in next generation.

3.2. Modified parallel genetic algorithm using SVM

In case of serial genetic algorithm, if population size is large, it needs much time according to increase of the computing amount. So, parallel genetic algorithm is invented.[7][9][10][11] Parallel genetic algorithm is applied to two methods. One is that decrease computation responsibility through computation amount is divided by using multiprocessor in supercomputer. The other method is that initial population divide into various subpopulation as serial genetic algorithm is modified. The isolation problems of solution in parallel genetic algorithm can solve such methods as global model, migration model. diffusion model etc.[10][11][12] In general, it was known as parallel genetic algorithm is more effective than serial genetic algorithm in optimizing problem of multi-objective function and can find solution rapidly though evaluation of objective function is less. We would get to improve speed and performance by applying SVM in parallel genetic algorithm ,too.

Modified Parallel Genetic Algorithm using SVM
Set k=0;
Create an initial subpopulation $P_1(k), P_2(k), \ldots, P_m(k)$
Evaluate $P_N(k)$; $(N \in 1, 2, \ldots, m)$
While $\langle$ the termination conditions are not met $\rangle$
  Set k=k+1;
  Generate training data and do SVM
  Reproduce mating pool $\bar{P}_N(k)$ from $P_N(k-1)$
    using roulette wheel selection;
  Crossover $\bar{P}_N(k)$ to form a tentative population
    $\tilde{P}_N(k)$;
  Mutate $\tilde{P}_N(k)$ to form the new population $P_N(k)$;
  Adopt the SVM criterion
    (Classify GOOD and BAD individuals)
  Migrate some individuals among each subpopulation.
  Evaluate $P_N(k)$; $(N \in 1, 2, \ldots, m)$
End while
Output the solution;

As we know in above algorithm, we divide initial population into some subpopulation and we would get to better results by applying SVM in each computation process. Some individuals that classified in subpopulation are migrated, so various properties are improved.

4. Illustrative Example

The objective function used in the example is the harvest problem[5] which is a one dimensional equation of growth:

$$x_{k+1} = a \cdot x_k - u_k$$

with one equality constraint,

$$x_0 = x_N,$$

where x_0 is the initial condition of the state, a is a scalar constant, and $x_k \in R$ and $u_k \in R^+$ are the state and nonnegative control respectively. The objective function is defined as:

$$f = \max \sum_{k=0}^{N-1} \sqrt{u_k} \tag{8}$$

where N is the number of control steps over which the problem is to be solved. Note that we change to minimize problem in this section as multiplies f by -1. The exact optimal solution for this problem can be determined analytically as:

$$f^* = \sqrt{\frac{x_0(a^N - 1)^2}{a^{N-1}(a-1)}} \tag{9}$$

The number of control steps for this problem $N = 20$. The decision variables are bounded in the range from 0 to 300. The number of individuals is 30. And the generation is 200.

Figures 1$\sim$4 show the distribution, best, mean, and standard derivation of object value with respect to generation. It can be known that the proposed method implements much faster and better performance than the conventional one.

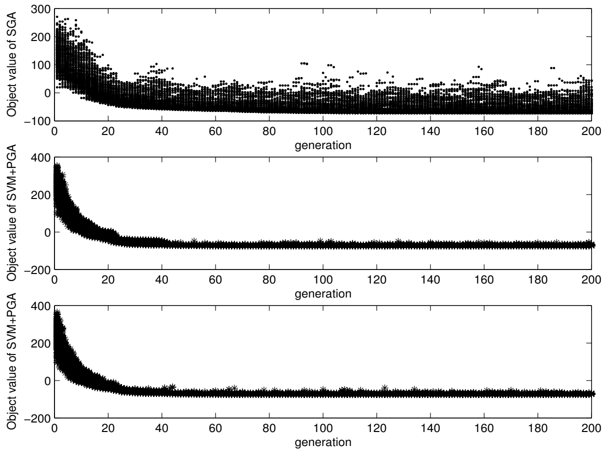

Fig. 1. Distribution of object value vs. generation

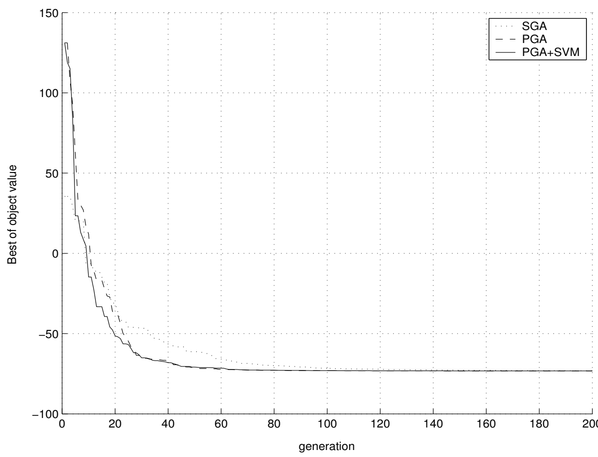

Fig. 2. Best of object value vs. generation

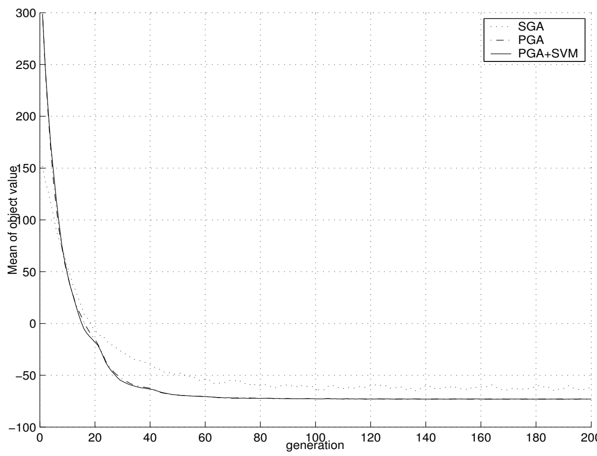

Fig. 3. Mean of object value vs. generation

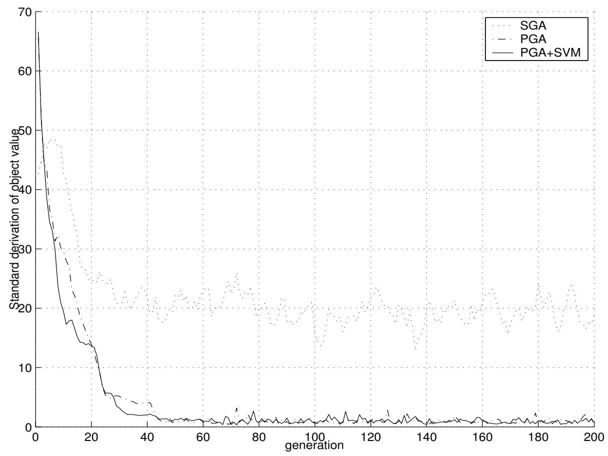

Fig. 4. Standard derivation of object value vs. generation

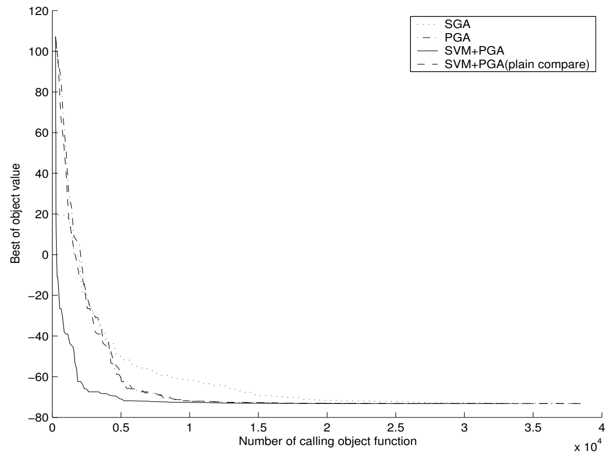

Fig. 5. Best of object value vs. the number of calling object function

Figure 5 shows that the proposed method converges to the optimal object value faster than the conventional approach.

5. Conclusion

In this paper, the modified genetic algorithm using SVM is proposed. The classifier which is made using SVM algorithm can expect only gene having good quality and only good gene can be taken part in evolutionary computation. So, we can show tendencies that speed and performance of proposed algorithm in this paper are more improved than conventional genetic algorithm because of the fewer data processing.

References

[1] K. A. De Jong, "Analysis of the Behaviour of a Class of Genetic Adaptive systems," Ph.D thesis, University of Michigan, Ann Arbor, 1975.

[2] J. H. Holland, *Adaptation in Natural and Aritificial Systems*, The University of Michigan Press, Michigan, 1975.

[3] T. Back, H.P. Schwefel, "An overview of evolutionary algorithms for parameter optimization," Evolutionary Computation, Vol. 1, No. 1, pp. 1-23, 1993.

[4] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Co. Inc., N.Y., 1989.

[5] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, Berlin Heidelberg, 1996.

[6] M. G. Bulmer, *The mathematical Theory of Quantitative Genetics*, Clarendon Press, Oxford, 1980.

[7] A. J. Chipperfield, P. J. Fleming and A. Y. Zomaya, *Parallel Genetic Algorithm, Parallel & Distributed Computing Handbook*, McGraw-Hill, 1996.

[8] A. J. Chipperfield and P. J. Fleming, "The Matlab Genetic Algorithm Toolbox"

[9] M. Schwehm, "Massively Parallel Genetic Algorithms," *Proc. Int. Conf. MPP'94*, Delft, Nethelands, pp. 505-512, 1994.

[10] P. Spiessens and B. Manderick, "A Massively Parallel Genetic Algorithm: Implementation and First Analysis," *Proc. 4th Int. Conf. on Genetic Algorithms*, Morgan Kaufman, San Mateo, CA, pp.279-286, 1991.

[11] H. Tamaki and Y. NIshikawa, "A Parallel Genetic Algorithm based on a Neighborhood Model and Its Application to the Jobshop Scheduling," *Parallel problem Solving From Nature 2, R. Manner and B. Maderick(ed)*, North-Holland, pp. 573-582, 1992.

[12] R. Tanese, "Distributed Genetic Algorithms," *Proc. 3rd Int. Conf. on Genetic Algorithms, J.D. Schaffer(ed)*, Morgan Kaufmann Publishers, San Mateo, pp. 434-439, 1989.

[13] V. N. Vapnik, *The Nature of statistical Learning Theory (Statstics for Engineering and Information Science)*, Springer-Verlag, 1999.

[14] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and other kernel-based learning methods*, Cambridge UK: Cambridge Univ. Press, 2000.

[15] B. Schölkopf and A. J. Smola, *Learning with Kernels.* The MIT Press, 2001.

[16] S. Gunn, "Support Vector Machines for Classification and Regression," ISIS Technical Report, University of Southampton, 1998.