An Improved Parameter Free Genetic Algorithm for CSD-FIR Filter design

Ryota Kato* and Kenji Suyama*
* Department of Electrical and Electronic Engineering, School of Engineering, Tokyo Denki University
5 Senju Asahi–cho, Adachi–ku, Tokyo, 120-8551, Japan
E-mail: 20kmj12@ms.dendai.ac.jp Tel/Fax: +81 3-5284-5400/+81 3-5284-5690

***Abstract*—A Canonical Signed Digit FIR filter (CSD–FIR filter) design problem is difficult to solve in polynomial computation time. Therefore, metaheuristics are used to obtain a semi-optimal solution to the problem. Parameter free genetic algorithm (PfGA) does not require to set any parameters that we need to adjust in advance. On the other hand, diversity of the search is relatively inferior for the CSD–FIR filter design problem. In this paper, we propose the CSD–FIR filter design method with PfGA with adaptive size of offspring (PfGA–AO) for better diversity and keeping the parameter free characteristic of the algorithm. PfGA–AO is applied for several design examples to show it's effectiveness.**

I. Introduction

Digital filters are classified into FIR(Finite Impulse Response) and IIR(Infinite Impulse Response) filters. Compared to the IIR filter, the FIR filter has the guaranteed BIBO(Bounded–Input Bounded–Output) stability and the ability to achieve the perfect linear phase characteristics. However, when a steep cutoff characteristic is required, a very high order FIR filter is often necessary. Therefore, the circuit scale of FIR filters is often larger than that of IIR filters.

In embedded systems, the constraints on the available circuit scale may be strict by constraints such as cost and power consumption. Therefore, it is important to reduce the circuit scale of the FIR filter and to design them by the desired circuit size. It is well known that the shifters inside the multipliers are dominant to the circuit scale of the FIR filter. The number of the shifters corresponds to the number of non–zero digits involved in each filter coefficient. Thus, it is important to reduce the non–zero digits to reduce the overall circuit scale of the FIR filter.

To reduce the numbers of the shifters, canonical signed digit (CSD) representation of the filter coefficients are proposed [1][2]. In the CSD representation, each digits of the filter coefficients are represented by $0, 1, \text{and} -1$. Also, adjacent of the non–zero digits is prohibited. Lastly, the available non–zero digits are constrained to limit the overall circuit size of the filter.

CSD–FIR filter design problem can be defined as a mixed integer linear programming (MILP) problem [3]. This problem is known to be NP–hard problem and it is hard to obtain the optimal solution in polynomial computation time. To solve such a problem, approaches with metaheuristics are proposed [4][5][6].

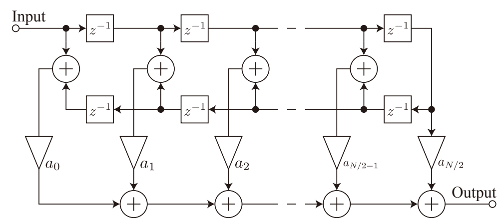

Fig. 1. Type–I FIR filter schematic

Many metaheuristics approaches was proposed for solving the CSD–FIR filter design problem, but for most approaches we need to adjust the parameters in advance. For example, the genetic algorithm (GA) have parameters such as the population size, the crossover rate, and the mutation rate. Although the behavior of these parameters can be predicted empirically, it is often determined by trial and error. To deal with this difficulty, parameter free GA (PfGA) was proposed [8]. For further improvement, distributed PfGA (DPfGA) is also proposed in the same literature. PfGA–AO consists of adaptively varying offspring number for more distributed search. In this paper, several design examples of CSD–FIR filters with PfGA and PfGA–AO are presented to show it's effectiveness.

II. Design problem

A. FIR filter design problem

The schematic of the even order – even symmetrical (Type–I) FIR filters are shown in Fig. 1. The magnitude response of the Type–1 FIR filters are formulated as follows,

$$H(\omega) = \left| \sum_{n=0}^{N/2} a_n \cos n\omega \right| \qquad \omega \in [0, \pi], \tag{1}$$

where N is the filter order, a_n is the filter coefficients, and ω is the normalized angular frequency. Because Type-I FIR filter is a perfect linear phased filter, it's group delay is always constant at $N/2$ samples.

An equripple FIR filter is designed in a sense of min–max criterion defined as follows,

$$\min_{\boldsymbol{a}} \max_{\omega \in \Omega} |H(\omega) - D(\omega)|, \tag{2}$$

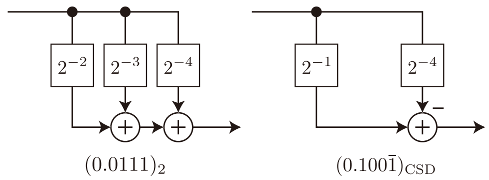

Fig. 2. Schematic of the multiplier
left: normal multiplier, right: CSD multiplier

$H(\omega)$ is the amplitude response of the FIR filter with continuous coefficients $\boldsymbol{a} = [a_0, a_1, \cdots, a_{N/2}]$, $D(\omega)$ is the desired filter's amplitude response, and Ω is the approximation band. Remez exchange algorithm is one of the well known algorithm that gives the optimal solution to this design problem [9].

B. CSD expression

A multiplier consists of shifters and adders as shown in Fig.2. It is clear that the number of shifters and the number of non–zero digits corresponds one on one from the schematic. Therefore, reducing the non–zero digits from the coefficient reduces the circuit scale. In literature [3], CSD expression are used to reduce the non–zero digits.

In the CSD expression, the filter coefficient is encoded by $x_{n,k} \in \{1, 0, -1 = \bar{1}\}$. Additionally, adjacent of the non–zero digits is forbidden. Where n is the filter coefficient index, and k is the digit index in the coefficient. The filter coefficient a_n is formulated as,

$$a_n = \sum_{k=0}^{p-1} x_{n,k} 2^{-k}, \tag{3}$$

where p is the word length. The schematic of the multiplier with CSD representation is also shown in Fig.2.

C. CSD–FIR filter design problem

Adjacency of the non–zero digit is forbidden as follows.

$$|x_{n,k}| + |x_{n,k+1}| \leq 1 \quad \forall\, n, k. \tag{4}$$

To limit the overall non–zero digits, to total number of available non–zero digits is defined as Λ an constrained as follows,

$$\sum_{n=0}^{N/2} \sum_{k=0}^{p-1} |x_{n,k}| \leq \Lambda. \tag{5}$$

With the min–max criterion, CSD–FIR filter design problem is defined as follows,

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \delta(\boldsymbol{x}) = \max_{\omega \in \Omega} |H(\omega_i) - D(\omega_i)| \\ \text{s.t.} \quad & |H(\omega_i) - D(\omega_i)| \leq \delta(\boldsymbol{x}) \\ & |x_{n,k}| + |x_{n,k+1}| \leq 1 \\ & \sum_{n=0}^{N/2} \sum_{k=0}^{p-1} |x_{n,k}| \leq \Lambda \\ & i \in \{0, 1, 2, \cdots, S\}, \\ & \boldsymbol{x} = [x_{0,0}, x_{0,1}, \cdots, x_{0,p-1}, \cdots, x_{N/2,p-1}], \end{aligned} \tag{6}$$

where S is the number of frequency divisions.

III. Genetic algorithm (GA)

Genetic algorithm (GA) is a metaheuristics which mimics the evolution of lifeforms [7]. In GA, $\boldsymbol{x}$ is called chromosome. Then, genetic manipulations such as selection, crossover, and mutation are applied.

A. Objective function

Objective function is defined as follows.

$$F(\boldsymbol{x}) = \delta(\boldsymbol{x}) + \phi_1(\boldsymbol{x}) + \phi_2(\boldsymbol{x}), \tag{7}$$

where $\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x})$ are the penalty functions. $\phi_1(\boldsymbol{x})$ is the penalty for the number of available non–zero digits defined as,

$$\phi_1(\boldsymbol{x}) = \begin{cases} 0 & \text{if } \lambda \leq \Lambda \\ \lambda - \Lambda & \text{if } \lambda > \Lambda, \end{cases} \tag{8}$$

where λ is the overall non–zero digits used in the CSD–FIR filter. $\phi_2(\boldsymbol{x})$ is the penalty to forbid the adjacency of the non–zero digits defined as,

$$\phi_2(\boldsymbol{x}) = \sum_{n=0}^{N/2} \sum_{m=0}^{p-2} |b_{n,m} b_{n,m+1}|. \tag{9}$$

B. Selection

In selection, the chromosomes is sorted depending on the objective function value. Then, the chromosomes with large objective function value is eliminated. This selection method is called as ranking selection. With selection, diversity of the population converges.

C. Crossover

In crossover, two chromosomes after the selection are selected randomly. Genes with the same index are swapped randomly. This crossover method is called as uniform crossover. With crossover, diversity of the population increases.

D. Mutation

In mutation, a randomly selected chromosome based on the mutation rate is mutated. A random gene in the selected chromosome mutates to allele. There are two alleles for CSD expression. *e.g.* if the gene is 0, it's alleles will be 1 and $\bar{1}$. In this paper, one of the allele will be selected with same probability.

IV. Parameter free GA (PfGA)

GA have parameters such as the population size, the cross over rate, and the mutation rate. Additionally, it is not mentioned in this paper, but GA have numerous amount of crossover and selection methods. Adjusting these parameters usually requires trial and error. To deal with this difficulty, parameter free GA (PfGA) was proposed[8]. Abstract diagram of the PfGA is shown in Fig.3. For the initial local group, two chromosome are selected from the full search section.

In literature [8], new chromosome is randomly generated. For CSD–FIR filter design problem, it initializes the population by Remez exchange algorithm and rounds the coefficients

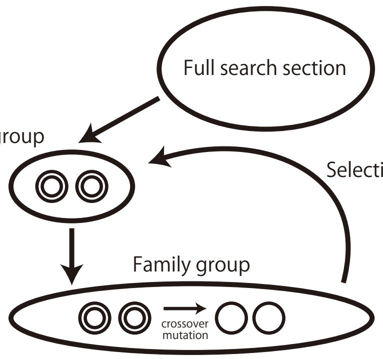

Fig. 3. Diagram of the PfGA

to the CSD expression. Also, manipulation generating new chromosome is performed by this method.

Crossover is performed by uniform crossover. Two chromosomes are selected randomly from the local group for crossover as parent. Since the probability of swapping or not swapping genes is equal, uniform crossover is a method that does not require any particular parameters. After crossover, there will be two parent chromosomes and two offspring chromosomes. These four chromosomes are called the family group.

Mutation occurs to one of the offspring randomly by the same probability. A gene is selected randomly and swapped to allele same as the normal GA.

For selection, there are four different cases.

Case–1
: In case–1, both offspring have better objective function value than the parents. Both offspring and the better parent are returned to the local group. Since three chromosomes are returned, the size of the local group increases.

Case–2
: In case–2, both parents have better objective function value than the offspring. The best parent is returned to the local group. Since only single chromosome is returned, the size of the local group decreases. Because we need at least two chromosomes in the local group, if the local group size is one, we generate new chromosome as same as the initialization.

Case–3
: In case–3, at least single parent have better objective function value than the offspring. The best parent and the best offspring is returned to the local group. Since two chromosomes are returned, the size of the local group is unchanged.

Case–4
: In case–4, at least single offspring have better objective function than the parents. The best offspring and newly generated chromosome are returned to the local group. The size of the local group is unchanged, as same as case–3.

With PfGA, there is no need to tune the parameter such as the population size, the mutation rate, and the position of crossover in advance.

A. PfGA–AO

For further improvement of CSD-FIR filter design, we applied PfGA–AO which is also proposed in literature [8] to the problem. PfGA–AO adapts the number of offspring by changing the number of crossovers defined as follows,

$$
\begin{aligned}
&P_{\mathrm{size}}(t+1) = P_{\mathrm{size}} + 2 - good2 - \sqrt{good1} \\
&\text{if } \ good2 \geq 1, bad \geq 1 \\
&P_{\mathrm{size}}(t+1) = P_{\mathrm{size}} + 1 - good2 - \sqrt{good1} \\
&\text{if } \ good2 \geq 1, bad = 0 \\
&P_{\mathrm{size}}(t+1) = P_{\mathrm{size}} + 2 - \left(good1 + \frac{good1}{2} + \sqrt{bad}\right) \\
&\text{if } \ good2 = 0, bad \geq 1 \\
&P_{\mathrm{size}}(t+1) = P_{\mathrm{size}} + 1 - \left(good1 + \frac{good1}{2}\right) \\
&\text{if } \ good2 = 0, bad = 0
\end{aligned}
\tag{10}
$$

where P_{size} is the number of crossovers, $good2$ is the number of times when the two offspring are better than the parents, $good1$ is the number of times when one among the two offspring is better than the parents, and bad is the number of times when the two offspring are the worse. $P_{\mathrm{size}}(0) = 1$ and the minimum value of the P_{size} is 1. By adapting the crossover number, it's search diversity increases. Additionally, selection rule is defined as follows.

Case–P1
: When more than two offspring chromosomes have better objective function value than there parents, two chromosomes are randomly chosen to return to the local group from the best three chromosomes in the family group.

Case–P2
: When all offspring chromosomes have worse objective function value then there parents, better parent chromosome and newly generated chromosome are returned to the local group.

Case–P3
: When at least single parent have better objective function value than all other offspring, both best parent and offspring chromosomes are returned to the local group.

Case–P4
: When at least single offspring have better objective function value than all other parents and offspring, the best offspring and newly generated chromosome are returned to the local group.

V. Numerical Experiments

To show the effectiveness of the PfGA and PfGA–AO in CSD–FIR filter desinge problem, we conducted several numerical experiments. Each CSD–FIR filter's parameters are shown in TableI. The results are shown in Table.II and Fig. 4.

The initial coefficients for the filters are obtained by Remez exchange algorithm, then converted to CSD expression. To fit the non–zero digit number within the Λ, the coefficients are rounded down. This simply reduced CSD–FIR filter's results are also shown in the results.

TABLE I
DESIGN EXAMPLES

	Ex.1	Ex.2	Ex.3	Ex.4	Ex.5
N	50	100	150	200	250
p	16	16	16	16	16
ω_p	0.2π	0.2π	0.2π	0.2π	0.2π
ω_s	0.3285π	0.2654π	0.2436π	0.2327π	0.2262π
Λ	50	100	150	200	250

TABLE II
RESULTS

		Ex.1	Ex.2	Ex.3	Ex.4	Ex.5
Average δ $(\times 10^{-3})$	PfGA	3.4742	3.2830	2.5165	2.7206	2.6634
	PfGA–AO	3.4244	3.1519	2.3251	2.4122	2.5255
Min. δ $(\times 10^{-3})$	PfGA	3.4050	3.0920	2.3440	2.4180	2.3850
	PfGA–AO	3.4050	2.8440	2.2490	2.1480	2.3420
Max. δ $(\times 10^{-3})$	PfGA	3.7540	3.5000	2.8350	2.9450	2.8810
	PfGA–AO	3.7800	3.3770	2.5740	2.7450	2.7720
Simple reduction δ $(\times 10^{-3})$		4.1975	3.9063	3.2514	2.8332	2.9761

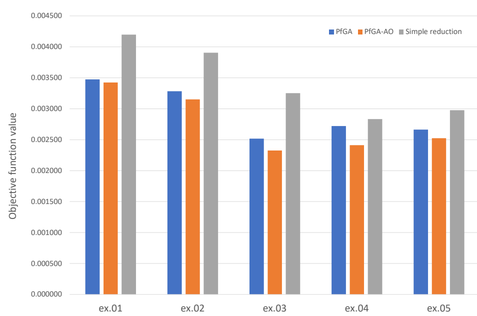

Fig. 4. Objective function value (Average)

From TableII and Fig.4, it was shown that PfGA and PfGA–AO was capable of obtaining better design result than simple reduction. Additionally, PfGA–AO was able to obtain better average, minimum, and maximum objective function value against PfGA. Amplitude response of each design examples are shown in Fig.5 – Fig.9. Since PfGA and PfGA–AO got the same result for Ex.1, Fig.5 only shows single plot. Updating curves of the best iteration and crossover number of PfGA–AO are shown in Fig.10 – Fig.14. Since PfGA-AO has a larger numbers of chromosomes to be evaluated in each generation, it can be expected that better results will be obtained. However, since the number of crosses is adaptive, the user can obtain better results without setting the parameters of the algorithm.

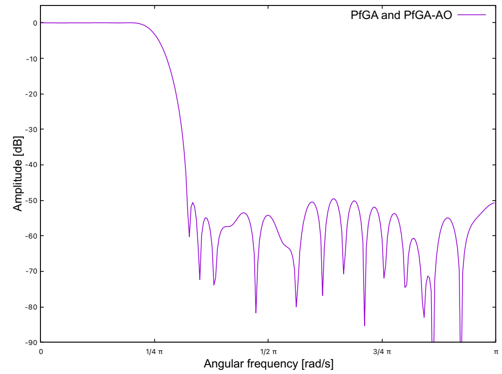

Fig. 5. Amplitude response (Ex.1)

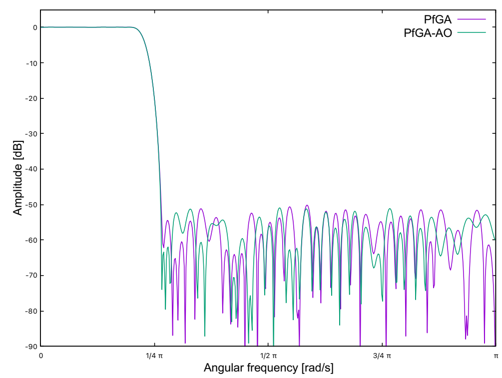

Fig. 6. Amplitude response (Ex.2)

VI. CONCLUSIONS

In this paper, we proposed a CSD–FIR filter design method with PfGA and PfGA–AO. Both algorithms do not require to adjust parameters such as population size, crossover rate, and mutation rate. We showed that both algorithms are capable of optimizing CSD–FIR filters from comparing it's objective function value against CSD–FIR filter designed by simply reducing it non–zero digits by rounding down. Additionally, we showed that PfGA–AO further more minimize the objective function from PfGA.

REFERENCES

[1] A. Peled, "On the hardware implementation of digital signal processors," in IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 24, no. 1, pp. 76-86, February 1976, doi: 10.1109/TASSP.1976.1162774.

[2] R. Jia, F. Wang, R. Chen, X. Wang, D. Shang and H. Yang, "High-order reconfigurable FIR filter design based on statistical analysis of CSD coefficients," 2013 International Conference on Field-Programmable Technology (FPT), Kyoto, 2013, pp. 402-405, doi: 10.1109/FPT.2013.6718399.

[3] H. Samueli, "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," in IEEE Transactions on Circuits and Systems, vol. 36, no. 7, pp. 1044-1047, July 1989, doi: 10.1109/31.31347.

[4] T. Imaizumi and K. Suyama, "An effective allocation of non-zero digits for CSD coefficient FIR filters using 0–1PSO," 2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, Kaohsiung, 2013, pp. 1-6, doi: 10.1109/APSIPA.2013.6694193.

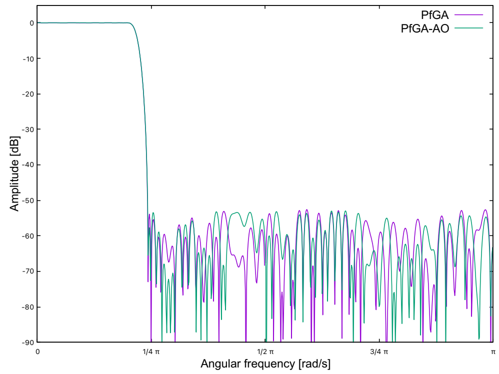

Fig. 7. Amplitude response (Ex.3)

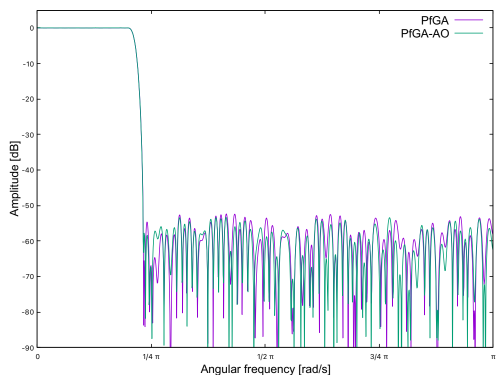

Fig. 8. Amplitude response (Ex.4)

[5] T. Sasahara and K. Suyama, "An ACO approach for design of CSD coefficient FIR filters," 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, 2015, pp. 463-468, doi: 10.1109/APSIPA.2015.7415314.
[6] N. Nakajima and K. Suyama, "A Diversification Strategy for CSD Coefficient FIR Filter Design using GA," 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Honolulu, HI, USA, 2018, pp. 291-295, doi: 10.23919/APSIPA.2018.8659683.
[7] HOLLAND, John Henry, et al. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.
[8] F. Kondo and T. Watanabe, "A study on distributed parameter free genetic algorithm for TSP problem," 2011 IEEE International Conference on Systems, Man, and Cybernetics, 2011, pp. 675-680, doi: 10.1109/ICSMC.2011.6083718.
[9] J. H. McClellan and T. W. Parks, "A personal history of the Parks-McClellan algorithm," in IEEE Signal Processing Magazine, vol. 22, no. 2, pp. 82-86, March 2005, doi: 10.1109/MSP.2005.1406492.

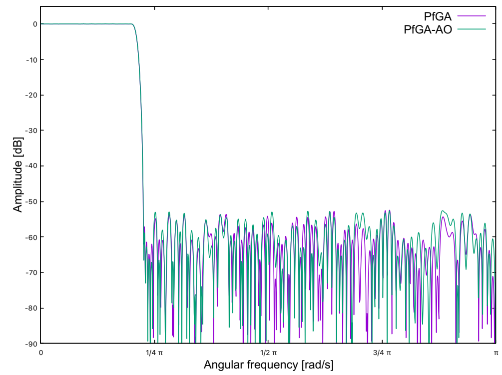

Fig. 9. Amplitude response (Ex.5)

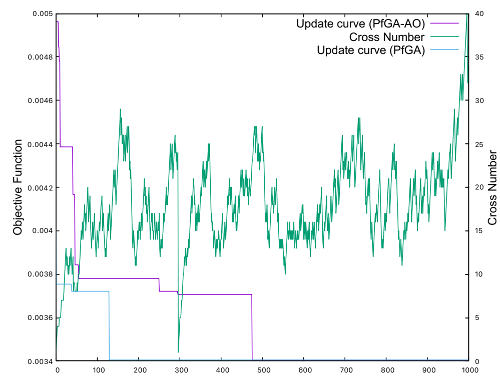

Fig. 10. Update curve (Ex.1)

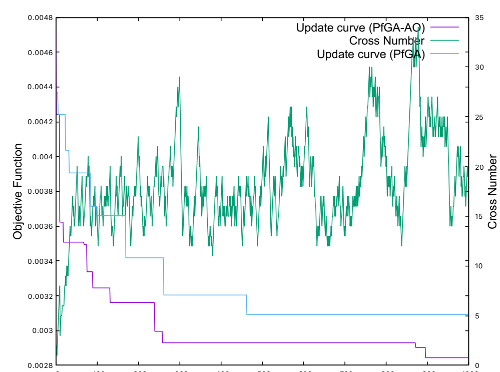

Fig. 11. Update curve (Ex.2)

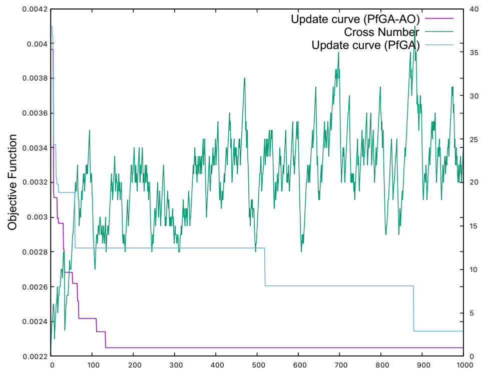

Fig. 12. Update curve (Ex.3)

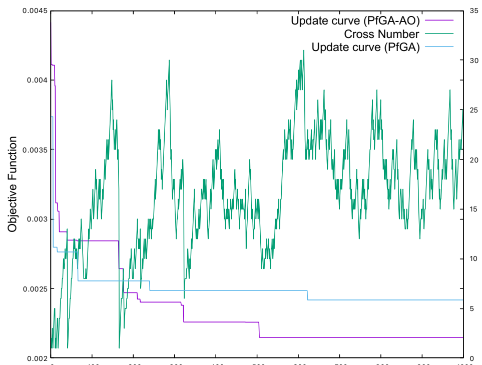

Fig. 13. Update curve (Ex.4)

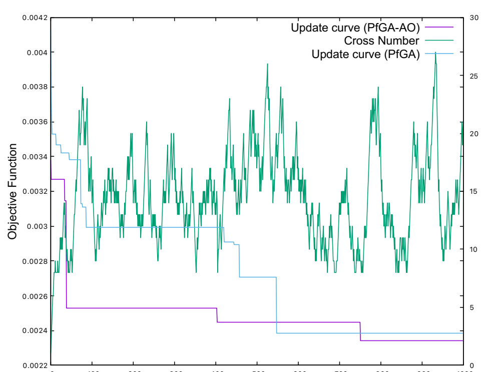

Fig. 14. Update curve (Ex.5)