A Parametric Study of Crossover Operators in Pareto-Based Multiobjective Evolutionary Algorithm

Shohei Maruyama and Tomoaki Tatsukawa(✉)

Tokyo University of Science, Tokyo 125-8585, Japan
4416631@ed.tus.ac.jp , tatsukawa@rs.tus.ac.jp

Abstract. The parameter setting in Multi-Objective Evolutionary Algorithms (MOEA) often affects the performance of optimization. Besides, the optimal values of parameters often depend on each optimization problem. However, it is difficult to decide the appropriate parameter setting for each problem in advance. In this study, the effect of parameter difference of the major crossover operators: Simulated Binary crossover (SBX), Differentioal Evolution operator (DE), Simplex crossover (SPX), Parent Centric crossover (PCX), and Unimodal Normal Distribution crossover (UNDX), is investigated by using 10 benchmark problems. DTLZ, WFG, and ZDT benchmark problems were considered. Non-dominated sorting genetic algorithm-II (NSGA-II) were used as a Pareto-based MOEA. The number of objectives was set to three. The experimental results on benchmark problems show that the effect of parameter variation is relatively small for SBX and SPX. On the other hand, there are optimum parameters in each benchmark problem as other crossover operators. This indicates that the choice of the crossover operator is significantly important in MOEA for achieving the good performance.

Keywords: Crossover operator · NSGA-II · SBX · DE · SPX · PCX · UNDX

1 Introduction

Multi-objective optimization problems (MOP) exist in various real-world applications such as engineering, financial, and scientific applications. In MOP, each objective function is often in a trade-off relationship, and in such a case, it is impossible to obtain one optimal solution. Therefore, the purpose of MOP is to get a set of solutions called Pareto-optimal (or non-dominated) solutions. Multi-objective optimization requires not only convergence but also the diversity of solutions. In recent years, many kinds of multi-objective evolutionary algorithms (MOEA) have been proposed and shown to demonstrate good performance in real-world optimization problems.

In MOEA, it is known that the performance is changed in different crossover operators. Each crossover operator has some parameters. For example, Simulated Binary crossover(SBX) [1] has a parameter η, and Parent Centric crossover

Y. Tan et al. (Eds.): ICSI 2017, Part II, LNCS 10386, pp. 3–14, 2017.
DOI: 10.1007/978-3-319-61833-3_1

(PCX) [10] has two parameters η and ζ. When we run MOEA, we have to properly determine the values of these parameters in advance. However, in most cases, we use the default values set in each MOEA as they are. Table 1 shows the default crossover operator and the default value of the crossover operator in some major MOEA. The default crossover operator as shown in Table 1 is almost SBX, and η is set to between 15 and 30. It is known that the optimal parameter value is changed by the optimization problem [7,15]. In [7], the effect of Differential Evolution(DE) operator [13] is investigated with 4 benchmark problems. However, the effect of various crossover operators has not yet been investigated.

In this study, we consider 5 crossover operators such as SBX, DE, simplex crossover (SPX) [14], PCX, and unimodal normal distribution crossover (UNDX) [8] as well as 10 benchmark problems to widely investigate the effects of crossover operators. For the number of parameters in each crossover operator, there is one in SBX and SPX, and two in DE, PCX, and UNDX. The objective of this study is to investigate the effect of changing parameters of each crossover operator. NSGA-II is used as a MOEA. DTLZ2,3,4 [3], WFG1,2,6,8,9 [9] and ZDT1,4 [17] were used as benchmark problems.

Table 1. Default values of crossover operator in MOEA

MOEA	Crossover method	Parameter value
NSGA-II [2]	SBX	$\eta = 20$
ϵ-MOEA [12]	SBX	$\eta = 15$
NSGA-III [11]	SBX	$\eta = 30$
MOEA/D	SBX	$\eta = 20$
MOEA/D-DE [16]	DE	$[cr, F] = [1.0, 0.5]$
IBEA [5]	SBX	$\eta = 20$
SPEA2 [4]	SBX	$\eta = 20$

2 Background

2.1 Related Work

Yuan et al. compared the performance when changing parameter values in the DE operator using several test problems. In Yuan's paper, cr is fixed to 0.1, and the performance is compared when F is varied. The performance of F is good at about 0.5 [15]. Besides, F is fixed to 0.5, and the performance is compared when cr is varied. The performance of cr is good at about 0.1. Hadka et al. examined the correlation of various parameters in the Borg MOEA, which has adaptive operator selection (AOS). However, in Hadka's paper, performance on each parameter value of crossover operator has not been reported [6].

Thus, a comparison of performance when fixing one of the parameter values in DE and a comparison of correlation with various parameters in AOS has already been done. However, the case of not fixing one of the parameter values and the case of changing to various parameter values in crossover operators excluding DE has not been compared yet.

In this study, we compare the performance when various parameter values are changed variously in SBX, DE, SPX, PCX, and UNDX. Although DE, PCX, and UNDX each have 2 parameters, in this study, we examine the performance without fixing one of these parameters. In addition, to gain insight into the parameters of each operator for each property of the problem, we used 10 kinds of benchmark problems with different features.

2.2 Crossover Operator

In MOEA, offspring solutions are generated from parent solutions by crossover and mutation in each generation. Crossover and mutation operators are different depending on whether the design variables are handled as binary or real numbers. In this research, we focus on cases where design variables are treated as real numbers. Five crossover operators are used in this research.

SBX imitates one point crossover of a binary number with a real number. The algorithm of SBX is shown below.

$$\beta(u) = \begin{cases} (2u)^{\frac{1}{\eta}}, & if \quad u(0,1) \leq 0.5 \\ \frac{1}{2(1-u)^{\frac{1}{\eta+1}}}, else & \end{cases} \tag{1}$$

$$y^{(1)} = \frac{1}{2}\{(1-\beta)x^{(1)} + (1+\beta)x^{(2)}\}. \tag{2}$$

$$y^{(2)} = \frac{1}{2}\{(1+\beta)x^{(1)} + (1-\beta)x^{(2)}\}. \tag{3}$$

where $x^{(1)}, x^{(2)}$ denote parent solutions, and $y^{(1)}, y^{(2)}$ denote offspring solutions. β denotes a distribution function, and η denotes a SBX parameter.

DE is a crossover operator featuring stochastic linear search. DE randomly selects 3 solutions from the parent population and calculates a vector from the first 2 solutions. It then generates an offspring solution from at last a parent solution to the vector direction. The algorithm of DE is shown below.

$$x'_i = x_{p1} + F(x_{p2} - x_{p3}). \tag{4}$$

$$y_{i,j} = \begin{cases} x'_{j,i}\ , & if \quad u(0,1) \leq cr \\ x_{j,i}\ , & else \end{cases} \tag{5}$$

Here, $\{x_{p1}, x_{p2}, x_{p3}\}$ denote parent solutions, y denotes offspring solution, and j denotes a design variable. Terms cr and F denote DE parameters.

SPX is a crossover operator that generates a child solution from the part that extends past the range surrounded by the centroid. These are parent individuals of multiple parent solutions. The SPX algorithm wich three parents is shown below.

$$O = \frac{1}{3}\sum_{i=1}^{3} x^{(i)}. \tag{6}$$

$$r_i = u^{\frac{1}{i}},\ i = 1, 2, 3. \tag{7}$$
$$v^{(i)} = O + \epsilon(x^{(i)} - O). \tag{8}$$
$$c^{(i)} = \begin{cases} 0,\ i = 1 \\ r_{i-1}(y^{(i-1)} - v^{(i)} + c^{(i-1)}),\ i = 2 \end{cases} \tag{9}$$
$$y = v^{(3)} + c^{(3)}. \tag{10}$$

Here, $\{x^i, i = 1, 2, 3\}$ denotes parent solutions, y denotes offspring solutions, and O denotes centroid parent solutions. Term ϵ denotes the SPX parameter.

PCX is a crossover operator that generates offspring solutions around the parent solution via the centroid of multiple parent solutions. The algorithm of PCX with three parents is shown below.

$$O = \frac{1}{3}\sum_{i=1}^{3} x^{(i)}. \tag{11}$$
$$d^{(p)} = x^{(p)} - O \quad p = 1, 2\ or\ 3. \tag{12}$$
$$\overline{D} = \frac{1}{2}\sum_{i=1, i\neq p}^{3} \frac{\sqrt{d(x^{(i)})}}{||d||}. \tag{13}$$
$$y = x^{(p)} + \omega_\zeta |d^{(p)}| + \sum_{i=1, i\neq p}^{3} \omega_\eta \overline{D} e^{(i)}. \tag{14}$$

Here, $\{x^i, i = 1, 2, 3\}$ denotes parent solutions, y denotes offspring solution, and O denotes the centroid of the parent solutions. Term $e^{(i)}$ is the orthogonal basis vector defined by $d^{(p)}$, and D denotes the distance of the perpendicular vectors of the remaining 2 individuals and the $d^{(p)}$. Terms η and ζ are parameters of PCX. Term ω_η denotes a random number generated from a Gaussian distribution with mean 0, and variance σ_η^2. Term ω_ζ denotes a random number generated from a Gaussian distribution with mean 0 and variance σ_ζ^2.

UNDX is a crossover operator creating a centroid from parent solutions. It generates offspring solutions using a normal distribution around the centroid. The algorithm of UNDX with three parents is shown below.

$$O = \sum_{i=1}^{2} x^{(i)}. \tag{15}$$
$$d = x^{(2)} - x^{(1)}. \tag{16}$$
$$D = \sqrt{|x^{(3)} - x^{(1)}|^2 - \frac{[d(|x^{(3)} - x^{(1)})]^2}{|d|^2}}. \tag{17}$$
$$y = O + \sigma_\zeta d + D\sum_{i=1}^{n-1} \sigma_{\eta_i} e_i. \tag{18}$$
$$\sigma_\zeta \sim N(0, \zeta^2), \qquad \sigma_\eta \sim N(0, \eta^2). \tag{19}$$

Here, $\{x^i, i = 1, 2, 3\}$ denotes parent solutions, y denotes offspring solutions, and O denotes centroid of 2 parent solutions. Term $e^{(i)}$ is the orthogonal basis vector defined by d. Terms η and ζ denote parameters of UNDX.

3 Computational Condition

In this study, we use NSGA-II as MOEA. The crossover operator in NSGA-II is calculated for each of the 5 operators described above. We also use polynomial mutation for mutation operator. Table 2 shows the parameter values of each crossover operator in this study.

Table 2. Parameter values of crossover operators

Crossover	Parameter	Range of parameter values
SBX	η	$\{5, 15, 25, 35, 45\}$
DE	cr	$\{0.1, 0.3, 0.5, 0.7, 0.9\}$
	F	$\{0.1, 0.3, 0.5, 0.7, 0.9\}$
SPX	ϵ	$\{1, 3, 5, 7, 9\}$
PCX	η	$\{0.1, 0.3, 0.5, 0.7, 0.9\}$
	ζ	$\{0.1, 0.3, 0.5, 0.7, 0.9\}$
UNDX	η	$\{0.1, 0.3, 0.5, 0.7, 0.9\}$
	ζ	$\{0.1, 0.3, 0.5, 0.7, 0.9\}$

Therefore, we adopt 10 benchmark problems with different properties (DTLZ2, 3, 4, WFG1, 2, 6, 8, 9, and ZDT1, 4). DTLZ and WFG problems can be scaled with any number of objectives and design variables. Here, we consider that the number of objectives in DTLZ and WFG is 3. The number of the objective functions in ZDT problems is set to 2. For DTLZ and WFG problems, the number of design variables is 12. In the ZDT1 problem, the number of design variables is 30, and in the ZDT4 problem, the number of design variables is 10. Therefore, none of the benchmark problems have a constraint function. Table 3 summarizes the properties of the benchmark problems from the view of separability of design variables, modality, shape of Pareto front, and whether each problem has bias or not. In all problems, both the population size and the number of generations are set to 100. Furthermore, the number of trials is 10.

To evaluate the performance in each pattern, the hypervolume (HV) [18] is used as the performance metric. The HV can measure both convergence and diversity of a optimal solution set and a higher value means better performance. The calculation formula of HV is shown below.

$$HV(P) = volume(\bigcup_{y \in S} [y_1, y_1^*] \times ..... \times [y_m, y_m^*]). \tag{20}$$

Table 3. Summary of the problems of the test problems

Problem	Separability	Modality	Bias	Shape
DTLZ2	S	U	×	concave
DTLZ3	S	M	×	concave
DTLZ4	S	U	○	concave
WFG1	S	U	○	convex
WFG2	N	M	×	convex
WFG6	N	U	×	concave
WFG8	N	U	○	concave
WFG9	N	M	○	concave
ZDT1	S	U	×	convex
ZDT4	S	M	×	convex

For separability, "S" denotes that design variables are separable, and "N" indicates that the design variables are non-separable. For each modality, "U" indicates that the problem is uni-modal, and "M" denotes that the problem is multi-modal. For bias, we indicate whether the problem has bias or not. Shape shows the shape of the Pareto front.

Here, m denotes the number of objectives, and S denotes the Pareto set. In this study, y^* is set to 1.5 times the maximum value of each objective function value in the true Pareto front except for DTLZ3. In DTLZ3, y^* is set to $\{150, 150, 150\}$. Furthermore, to avoid changing the value for each problem, divide by the volume surrounded by the origin point and y^* to convert it to the proportion of the HV value surrounded by the origin point and y^*. Pareto set in all search solutions is used for calculation of HV.

4 Results and Discussion

In all parameters mentioned above in Table 2, the trials are run using different 10 random seeds. The HV calculates the average of 10 trials and compares it to each parameter value. Figures 1, 2, 3, 4 and 5 show the average values of HV wich 10 trials in each crossover operator. A bluer color is preferred. In Figs. 1, 2, 3, 4 and 5, the HV value is converted from 0 to 1 for each problem, and in that range, each cell is colored in 50 hierarchies. The conversion formula is shown below.

$$HV_{new} = \frac{HV_{old} - min(HV_{old})}{max(HV_{old}) - min(HV_{old})}. \tag{21}$$

For each cell in Figs. 1, 2, 3, 4 and 5, each cell shows the proportion of the HV value surrounded by the origin point and y^*.

Figure 1 shows the result of SBX - the number of parameters is only one. From Fig. 1, it is clear when the value of η is large, each cell is bluer, but there

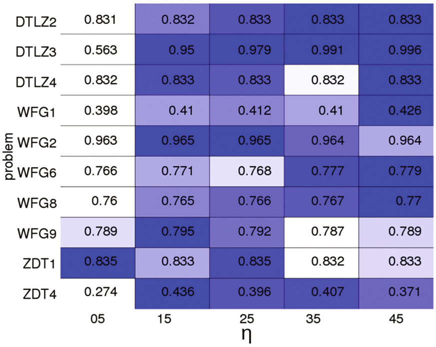


Fig. 1. Result of SBX

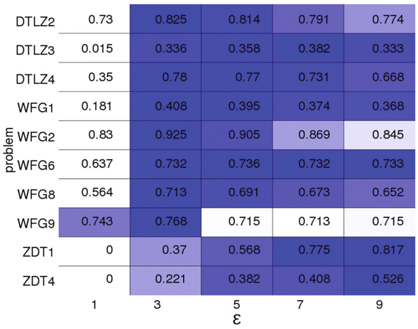


Fig. 2. Result of SPX

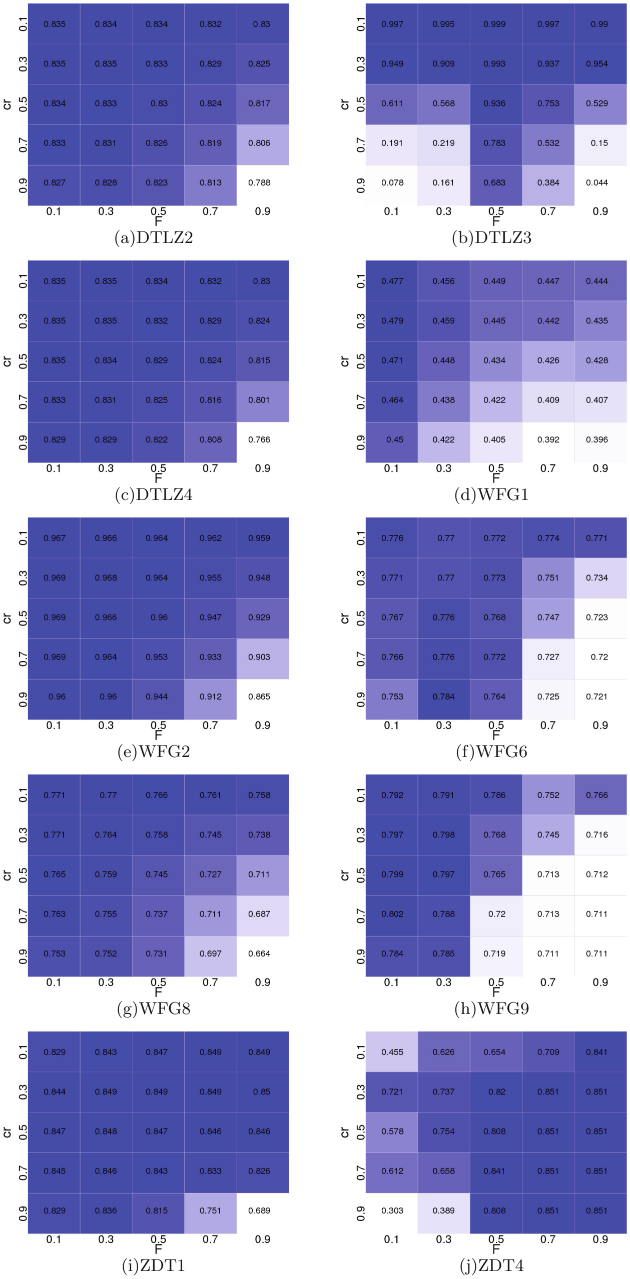
is little difference in the HV value for each parameter except DTLZ3 and ZDT4. We think that η does not change the performance in most test problems. In the case of DTLZ3 and ZDT4, there is a large difference in the $\eta = 5$ case and other cases. Performance is not good with $\eta = 5$.

Figure 2 shows the result of SPX. The number of parameters is only one. Figure 2 shows that the value of $\epsilon = 3, 5, 7$, each cell is bluer. Thus, the performance is considered good with $\epsilon = 3, 5, 7$ except WFG9. With WFG9, there is almost no difference in the HV value for each parameter, and the value of ϵ does not change the performance in WFG9.

Figure 3 shows the result of DE - the number of parameters is two. In DE, PCX, and UNDX, there are two parameters: the horizontal axis and the vertical axis, and these are set as the respective parameters. The performance is compared with each problem. Therefore, in DE, PCX, and UNDX, the performance is compared to 10 figures. Thus, Fig. 3(a)–(j) show the result of DE in each test problem. From Fig. 3, it is understood when the value of cr and F are small, each cell is bluer in most test problems. However, for ZDT4, it is understood that each cell is bluer when the value of cr and F are small. Thus, in most test problems, the performance is good when the value of cr and F are small. For ZDT4, the performance is good when the value of cr and F are large.

Figure 4 shows the result of PCX, and Fig. 4(a)–(j) show the results in each problem. From Fig. 4, the color tone differs for each problem, and the optimal parameter value may be different depending on the problem in PCX. Especially for WFG2, 6, 8, 9, each cell is bluer when the value of η is large. A common characteristic of WFG2, 6, 8, 9 is that design variables are non-separable. Thus in PCX, if the design variables are non-separable, then the performance is good when the value of η in PCX is large.

Figure 5 shows the result of UNDX, and Fig. 5(a)–(j) show the results in each problem. Figure 5 shaws that when the value of η is large, each cell is bluer in most test problems. Thus, the performance is good when the value of η is large. However, in the case of DTLZ3 and WFG1, there is little difference in the HV value for each parameter. Thus, the value of η likely does not change



(a)DTLZ2 (b)DTLZ3

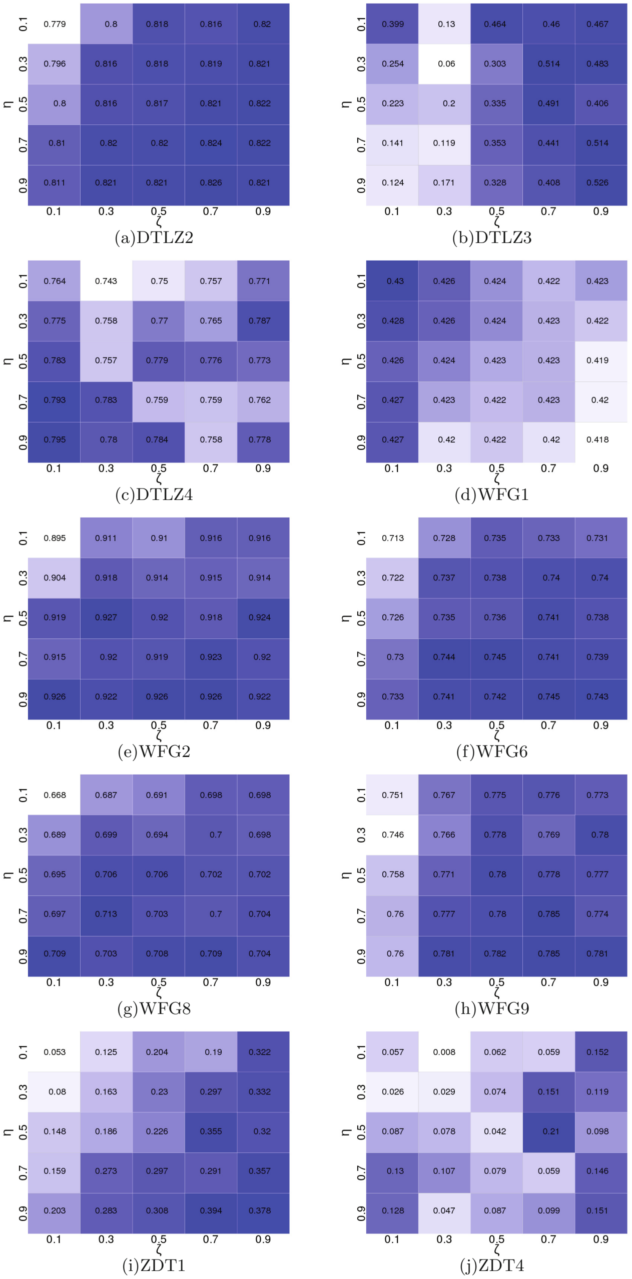
(c)DTLZ4 (d)WFG1

(e)WFG2 (f)WFG6

(g)WFG8 (h)WFG9

(i)ZDT1 (j)ZDT4

Fig. 3. Result of DE



(a)DTLZ2 (b)DTLZ3

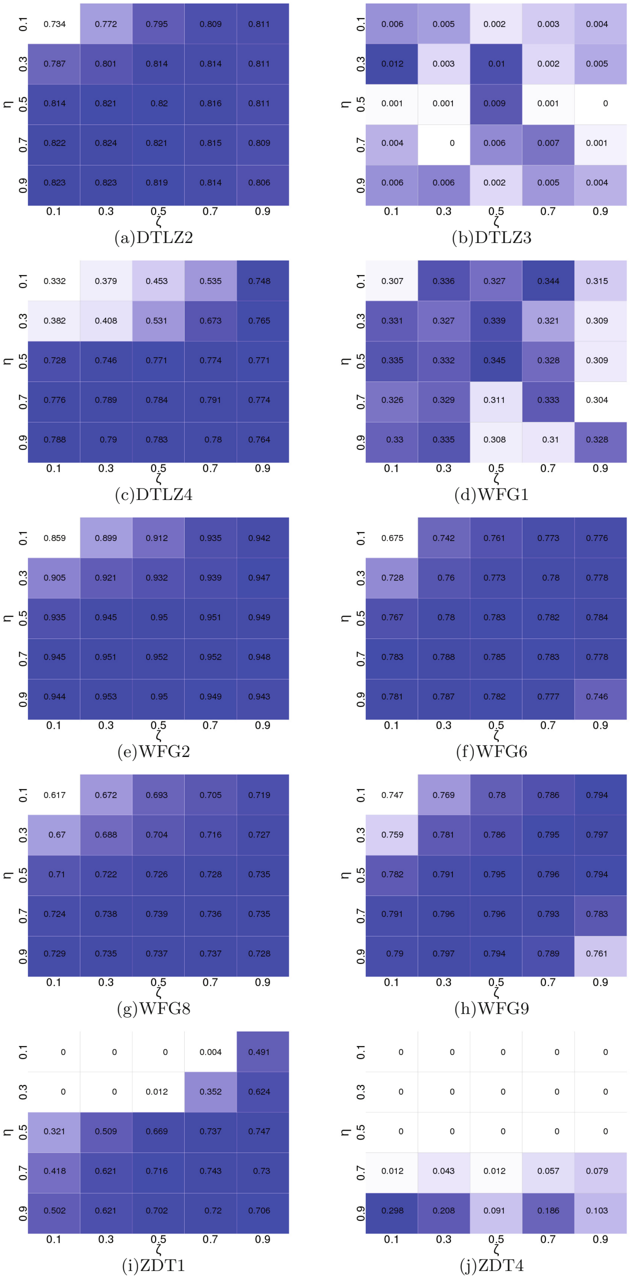
(c)DTLZ4 (d)WFG1

(e)WFG2 (f)WFG6

(g)WFG8 (h)WFG9

(i)ZDT1 (j)ZDT4

Fig. 4. Result of PCX



(a)DTLZ2 (b)DTLZ3

(c)DTLZ4 (d)WFG1

(e)WFG2 (f)WFG6

(g)WFG8 (h)WFG9

(i)ZDT1 (j)ZDT4

Fig. 5. Result of UNDX

the performance in DTLZ3 and WFG1. On the other hand, in the case of ζ, it becomes bluer from 0.3 to 0.5 in each problem. Thus, the performance is good when $\zeta = 0.3, 0.5$.roblem. Thus, the performance is good when $\zeta = 0.3, 0.5$.

5 Conclusion

In this study, we consider five crossover operators such as SBX, DE, SPX, PCX, and UNDX as well as 10 benchmark problems to widely investigate the effects of crossover operators. For the number of parameters in each crossover operator, there is one in SBX and SPX, and there are two in DE, PCX, and UNDX. The objective of this study is to investigate the effect of changing the parameters in each crossover operator. NSGA-II is used as a MOEA. DTLZ2,3,4 [3], WFG1,2,6,8,9 [9] and ZDT1,4 [17] is used as benchmark problems.

In SBX, a large value of η is preferred. In SPX, values of 3 to 7 are preferred for ϵ. However, there is little difference in the performance for η and ϵ. These results indicate that the effect of parameter variation is relatively small.

In DE, small values are preferred for both cr and F. In PCX, appropriate values of η and ζ depend on the problem. When the design variable is non-separable, a large value is preferred for η. In UNDX, a large value of η is preferred. For ζ, values about 0.3 to 0.5 are preferred. These results indicate that as for DE, PCX, and UNDX, there are optimum parameters in each benchmark problem. This also indicates that the choice of crossover operator is significantly important in MOEA for achieving good performance.

In future work, we will investigate not only the effect when the number of objectives is more than four but also the effect when the different mutation is considered. In addition, it is also necessary to consider other types of MOEA such as MOEA/D and IBEA.

Acknowledgment. This research was partially supported by MEXT as "Priority Issue 8 (Sub-issue A) on Post-K computer" (Development of innovative design and production processes that lead the way for the manufacturing industry in the near future, sub-issue A: Research and development of multi-objective design exploration and high-performance computing technologies for design innovation).

References

1. Deb, K., Agrawal, R.B.: Simulated binary crossover for continuous search space. Complex Syst. **50**(9), 115–148 (1994)
2. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
3. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable multi-objective optimization test problems. In: Proceedings of the Congress Evolutionary Computation CEC 2002, vol. 1, pp. 825–830, May 2002
4. Zitzler, E., Laumanns, M., Thiele, L.: Spea 2: improving the strength pareto evolutionary algorithm for multiobjective optimization. International Center for Numerical Methods in Engineering (CIMNE) (2002)

5. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Yao, X., et al. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 832–842. Springer, Heidelberg (2004). doi:10.1007/978-3-540-30217-9_84
6. Hadka, D., Reed, P.M., Simpson, T.W.: Diagnostic assessment of the borg moea for many-objective product family design problems. In: Proceedings of IEEE Congress on Evolutionary Computation, pp. 1–10, June 2012
7. Hitomi, N., Selva, D.: A classification and comparison of credit assignment strategies in multiobjective adaptive operator selection. IEEE Trans. Evol. Comput. (2016) (in press)
8. Kita, H., Ono, I., Kobayashi, S.: Multi-parental extension of the unimodal normal distri- bution crossover for real-coded genetic algorithms. In: Proceedings of Congress on Evolutionary Computation, pp. 581–1588 (1999)
9. Huband, S., Hingston, P., Barone, L., While, R.L.: A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans. Evol. Comput. **10**(5), 477–506 (2006). http://dx.doi.org/10.1109/TEVC.2005.861417
10. Deb, K., Joshi, D., Anand, A.: Real-coded evolutionary algorithms with parent-centric recombination. In: Proceedings of the World Congress on Computational Intelligence, pp. 61–66 (2002)
11. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. IEEE Trans. Evol. Comput. **18**(4), 577–601 (2014)
12. Deb, K., Mohan, M., Mishra, S.: A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions. KanGAL report 2003002, pp. 1–18 (2003)
13. Storn, R., Price, K.: Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. **11**(4), 341–359 (1997)
14. Tsutsui, S., Yamamura, M., Higuchi, T.: Multi-parent recombination with simplex crossover in real coded genetic algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 1999), pp. 657–664 (1999)
15. Yuan, Y., Xu, H., Wang, B.: An experimental investigation of variation operators in reference-point based many-objective optimization. In: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, pp. 775–782. ACM (2015)
16. Zhang, Q., Liu, W., Li, H.: The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances. In: IEEE CEC, pp. 203–208 (2009). http://dx.doi.org/10.1109/CEC.2009.4982949
17. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. Evol. Comput. **8**(2), 173–195 (2000). http://dx.doi.org/10.1162/106365600568202
18. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Trans. Evol. Comput. **3**(4), 257–271 (1999). http://dx.doi.org/10.1109/4235.797969