GA Based on the UV-Structure Hypothesis and Its Application to JSP

Kokolo Ikeda[1] and Shigenobu Kobayashi[1]

Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology { psyche@fe.dis.titech.ac.jp , kobayasi@dis.titech.ac.jp }

Abstract. Genetic Algorithms(GAs) are effective approximation algorithms which focus on "hopeful area" in searching process. However, in harder problems, it is often very difficult to maintain a favorable trade-off between exploitation and exploration. All individuals leave the big-valley including the global optimum, and concentrate on another big-valley including a local optimum often. In this paper, we define such a situation on conventional GAs as the "UV-phenomenon", and suggest UV-structures as hard landscape structures that will cause the UV-phenomenon. We propose Innately Split Model(ISM) as a new GA model which can avoid the UV-phenomenon. We apply ISM to Job-shop Scheduling Problem (JSP), which is considered as one of globally multimodal and UV-structural problems. It is shown that ISM surpasses all famous approximation algorithms applied to JSP.

1 Introduction

GA is an approximation algorithm which can find high quality solutions efficiently, by searching hopeful area intensively in the search space. However, if the population converges so early in searching process, GA may fail to find the global optimum ("the optimum" we call) and may converge to a local optimum. Big-valley structure is the well-known landscape structure proposed in [Boese 1994,1995]. When several big-valleys exist in the search space, and each valley has its own eminent local optimum, we call these landscape globally multimodal.

It is very difficult for conventional GAs to find the optimum in globally multimodal problems, especially when local optimums are scarcely inferior to the optimum and are located far from it in the search space. We call these local optimums "far-locals" briefly. As the global multimodality is a characteristic probably common to harder optimization problems, it is desirable to design a new GA which considers the global multimodality.

In section 2, we consider why and how GAs fail to find the optimum by the global multimodality, and propose the UV-structure hypothesis which explains why and how. In section 3 and 4, we verify our suggestion briefly on a function optimization problem and JSP. In section 5, we propose Innately Split Model(ISM), which splits individuals to several groups, initializes each group locally, and restricts the crossover within intra-group. In section 6, we apply a GA based on ISM to JSP, and confirm its superiority to conventional approximation algorithms.

2 The UV-structure hypothesis

In this section, we propose the hypothesis which qualitatively explains phenomena that GAs fail to find the optimum by the global multimodality. When the landscape has 2 big-valleys, a U-valley which is broad and shallow, and a V-valley which is narrow and deep, containing the optimum, we call this landscape the "typical UV-structure". Applying conventional GAs to such a landscape, the following phenomenon is expected. As individuals of the U-valley evolve rapidly compared with ones of the V-valley, individuals of the V-valley will be weeded out gradually through the alternation process, even if enough individuals exist on the both valleys in early generations. In consequence, GAs search among the U-valley intensively, and fail to find the optimum.

In globally multimodal problems, it is often observed that GAs search intensively among a valley which doesn't include the optimum and fail to find the optimum as above. We define the phenomenon the **UV-phenomenon**. When the quality of individuals searching among the valley which includes the optimum("opt-valley" we call) is inferior to the quality of individuals searching among valleys which have one of far-locals ("local-valleys" we call), the UV-phenomenon is caused by the alternation. Here we call the landscape which cause the UV-phenomenon the **UV-structure**, and propose 3 hypotheses of UV-structures.

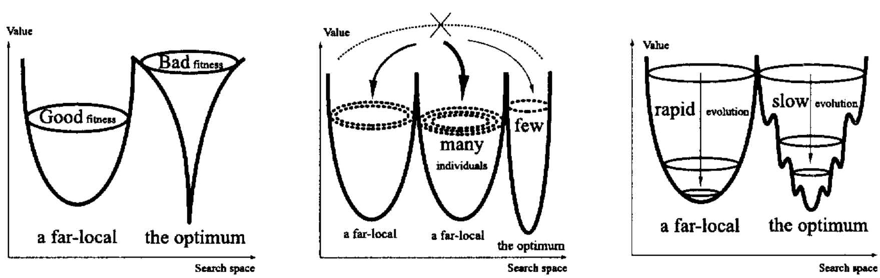


Fig. 1. UV Structures class1(left), class2(center), class3(right)

1. *Structures which affect the apparent hopefulness in early generations*
   The average quality of individuals searching among the opt-valley, so to say the apparent hopefulness of the opt-valley, is worse than of local-valleys in early generations. In this case, GAs may abandon the opt-valley before the real hopefulness of the opt-valley is known.
2. *Structures which affect the number of individuals in valleys*
   When the opt-valley is located on an outland in the search space or is very narrow, fewer offsprings will be generated by crossover or mutation. If individuals seraching among the opt-valley are relatively few, the probability of mating within the opt-valley will be small, so the speed of evolution in the opt-valley will be relatively slow. Consequently GAs may fail to find the optimum.
3. *Structures which affect the speed of evolution*
   The complexity of a valley will make much effect on the speed of evolution

of individuals searching among it. When the opt-valley is more complex to evolve than one of local-valleys, the quality of individuals searching among the local-valley will be superior to of the opt-valley. This may cause GAs to fail to find the optimum.

3 The UV-phenomenon on function optimizations

We introduce f_9 function minimization problems here. These functions have simple structures, and we can control the shape of these functions easily by operating parameters. The landscape of an example is shown in Fig2.

$$f_9(x_1, x_2) := \sum_{i=1}^{2} \{4(x_i - \lfloor x_i \rfloor - 0.5)^2 + bB(x_i)\} + A_{x_1 x_2}; x_1, x_2 \in (0, 3)$$

$$B(x_i) := 1 - \left\{ \frac{1 + cos(2\pi C_{x_1 x_2} x_i)}{2} \right\}^{D_{x_1 x_2}}$$

$$(A_{x_1 x_2}, C_{x_1 x_2}, D_{x_1 x_2}) := \begin{cases} (0, c_{opt}, d_{opt}), \text{ if } \lfloor x_1 \rfloor = r_1, \lfloor x_2 \rfloor = r_2 \\ (a, c_{loc}, d_{loc}), \text{ otherwise} \end{cases}$$

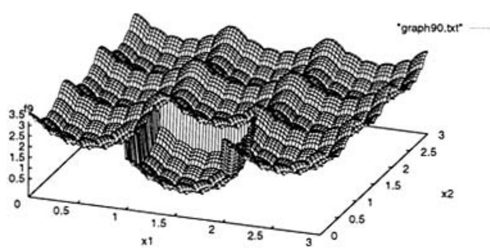


Fig. 2. an example of f9 : a=1.5, b=0.1, c=6, $d_{opt}=d_{loc}$=4 , r_1=1, r_2=0

These functions have 9 big-valleys. One of them lies down relatively to others, and has the optimum with the value of 0 ,and has 4 quasi optimums ,i.e. the closest local optimums to the optimum with the value of about $4/c^2$. And other 8 valleys have far-locals with a. In following experiments, we use as default parameters a=0.005, b=1, $c_{opt}=c_{loc}$=16, $d_{opt}=d_{loc}$=4. f_{9c} is the function of r_1=1 and r_2=1, this has its opt-valley in the center. f_{9s} is the function of r_1=1 and r_2=0, its opt-valley is in the side. f_{9e} is the function of r_1=0 and r_2=0, its opt-valley is in the edge. And, f_{9s+} is the function of r_1=1, r_2=0, and d_{opt}=8. In f_{9s+}, the opt-valley has narrower bays, so its apparent hopefulness in early generations is worse than others. f_{9s*} is with c_{loc}=6, far-locals are easily(rapidly) detected.

It is expected that we may confirm the hypothesis 2, the location of the opt-valley effects the difficulty, by comparing among f_{9c},f_{9s},and f_{9e}.Also we may confirm hypothesis 1, wrong apparent hopefulness of the opt-valley will cause

UV-phenomenon, by comparing f_{9s} with f_{9s+}. In following experiments, we employ UNDX[Ono 97](offsprings 100) and DDA[Takahashi 99] which is known as a very effective GA for function optimization problems.

Fig3 shows a typical behavior of population with size 500, about a successful case at f_{9c} (left) and an unsuccessful case at f_{9e} (right). If the evolution goes well, individuals closer to the optimum will increase. So, these graphs show the variation per generation of the number of individuals which is closer than 0.03, 0.06, ... , 0.9 to the optimum. In the successful case, closer individuals increase favorably. In the unsuccessful case, closer individuals increase instantaneously till about generation 5, and decrease gradually. Though a few individuals try to evolve at about generation 30, finally all individuals in the opt-valley disappear. It is important to see that population converges not on closest local optimums but on one of far-locals, i.e. a local-optimum located far from the optimum.

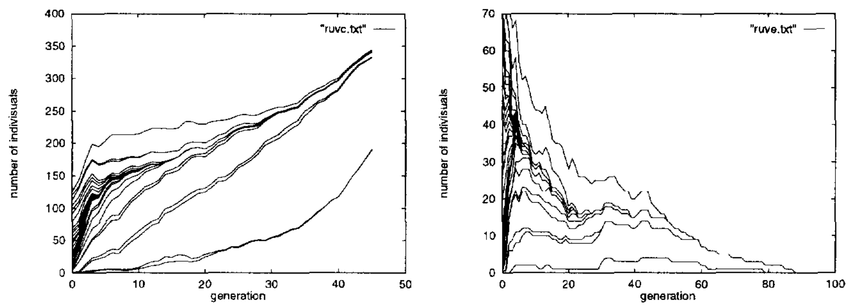


Fig. 3. variation of the number of individuals in some neighborhoods of the optimum: a successful case at f_{9c} (left), an unsuccessful case at f_{9e} (right)

Table1 shows the result of following experiments. To judge difficulties of introduced functions, we tried various population sizes for each function. We call a trial 'success' if a solution better than 10^{-7} is found within 1000 generations, and 60 trials were done in each condition. We can see by the result, f_{9c} is the easiest and f_{9e} is the hardest among f_{9c},f_{9s},and f_{9e}, this supports our hypothesis 2. And, f_{9s+} is much harder than f_{9s}, this supports hypothesis 1. f_{9s*} is harder than f_{9s}, this supports hypothesys 3. A characteristic of the crossover operator

Table 1. results of test functions by UNDX+DDA

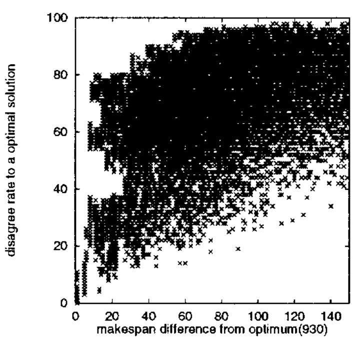
population size	f_{9c}	f_{9s}	f_{9e}	f_{9s+}	f_{9s*}
20	46/60	11/60	4/60	2/60	5/60
100	60/60	34/60	8/60	13/60	16/60
500	60/60	60/60	27/60	28/60	54/60

of GAs yields these results that the location of the opt-valley effects the difficulty. Although UNDX capably inherits statistics of parents such as the mean vector and the covariance matrix of the population[Kita 99], UNDX can't keep the higher distribution manner. Actually, though all 9 valleys contain 11% individuals when the initial population is uniformly generated, its offsprings fairly slant, center valley will have 20% individuals, one side valley 12%, one edge valley 8%. It is natural that, the more offsprings are generated in a valley, the more individuals can search among the valley. In these results, f_{9c} is harder than f_{9e}.

By these experiments, it is shown that UV-structures can cause the UV-phenomenon. 12(or 13,14...)-dimensional Fletcher-Powell function is extremely UV-structured we detected, they will be shown in following journal paper.

4 The UV-phenomenon on JSP

JSP is known as one of the hardest benchmarks, and would have the global multimodality. Fig4 shows the relationship between the distance to a solution from the optimum and its makespan difference from the optimum about famous instance ft10(from Fisher, Thompson). Here the distance is defined as the disagree rate of orders of operations(left) and I_2distance [Sakuma 2000] (right). Fig4 suggests that some eminent local-optimums exist and big-valleys are formed around them respectively . Actually, as the optimum makespan of ft10 is 930, ft10 also has an eminent local-optimum 938, which stays away from the optimum with 74% disagree rate, 170 I_2 distance. These are enough big distances compared with the average distance between randomly generated two individuals, 83% disagree rate and 231 I_2 distance. To conclude, JSP is considered as a globally multimodal problem.



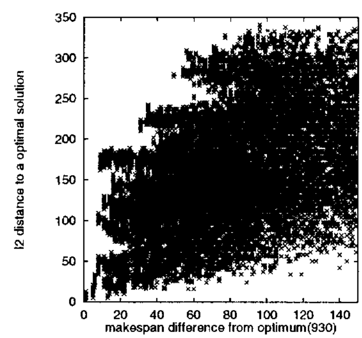


Fig. 4. relation between the makespan and the disagree rate(left) to the optimum ,I_2 distance(right) in ft10

Next we examine how the population converges in terms of the same expression as section 3. Fig5(left) shows the variation per generation of the number of individuals which is closer than 12,24,...,120 to the optimum by I_2 distance in a successful case of ft10. Closer individuals increased favorably as Fig3(left), and the optimum was found at the same time nearest individuals increased. In contrast, Fig5(right) shows the variation in an unsuccessful case of abz5, a harder instance than ft10. As closer individuals increased till generation 10, decreased gradually, and finally converged to far-locals. The search space and the landscape structure of JSP is very complicated to analyze. At the first, we try to know information about where the optimum and far-locals are located in the search space. We generated random solutions, and grouped them by I_2 distances to the optimum and to one of known far-locals. In the easier instance ft10, 52% of them are closer to the optimum 930 than to a far-local 938. This means, the optimum 930 and a far-local 938 are on an equal footing. On the other hand

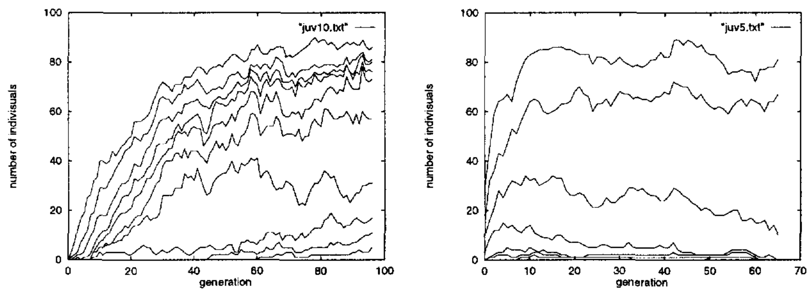


Fig. 5. variation of the number of individuals in some neighborhoods of the optimum: successful case at ft10(left), unsuccessful case at abz5(right)

in the harder instance abz5, only 26% of them are closer to the optimum 1234, than to a far-local 1238. Average I_2 distance of randomly generated solutions to the optimum 1234 is 145, and 129 to the 1238. We can say by these data, the optimum 1234 is located on an outland in the search space relatively to the 1238.

Next, we tried to investigate the behavior of evolution of the opt-valley and a local-valley in abz5. Although ideally we want to gather "individuals in the opt(or local)-valley" , it's very difficult to define or judge which individuals are in the opt(or local)-valley. Here we define "the opt(or local)-valley set" as "individuals which are closer to the optimum 1234 (or the far-local 1238) than a far-local(or the optimum) by d in I_2 distance". We sampled 30 randomly generated individuals of the opt(or local)-valley set in each trial, and tried to find the optimum 1234 (or 1238). It is expected that the opt-valley set find 1234, and the local-valley set find 1238. We employed CCM[Ono 98] as alternation model, 20 offsprings in every crossover, and tried till 200 generations. Table2

Table 2. Convergence experiment with specially initialized population

d	Generated rate: opt/local	Average makespan	on Opt	on Local	Merge
10(abz5)	19.1%/59.8%	1529/1502	4/20, 28s	18/20, 14s	0/20
30(abz5)	9.0%/37.7%	1535/1497	10/20,17s	17/20, 11s	2/20
50(abz5)	2.9%/17.4%	1531/1478	16/20, 6s	19/20, 7s	6/20
30(ft10)	32.0%/18.4%	1210/1204	11/20,20s	14/20, 13s	9/20

shows the probability randomly generated solution is grouped to the opt(local)-valley set, average makespans of the opt(local)-valley set t, the number of times the opt-valley set find the optimum (and average CPU time to find), the number of times the local-valley set find the far-locals 1238(and average time), and the number of times merged set of the both sets (population 60) finds the optimum.

The difference of generated rates shows that individuals closer to the optimum are rarer than to the local 1238 (this supports hypothesys 2). As opt(or local)-valley sets can find frequently the optimum(or 1238), our definition of the set of the opt(or local)-valley is enough adequate. The fact that the average quality of the opt-valley set is fairly worse than of the local-valley set ,supports hypothesis 1. We can observe the UV-phenomenon, because merged sets seldom find the optimum though opt-valley sets can find frequently. In other words, this instance is hard not because of the difficulty in finding the optimum in the opt-valley, but because of the UV-phenomenon that far-locals pull individuals in the opt-valley, i.e. because of the crossover between individuals in the opt-valley and individuals in a local-valley. Incidentally, we couldn't observe some significant differences of evolution speed between the both valleys. ft10 is not(or less) UV-structured, and experiments with merged sets show that UV-phenomenon is seldomely observed in ft10.

By these results, we can say that the UV-phenomenon surely occurs in JSP, and hypothesis 2 and/or 1 is influential at abz5.

5 Innately Split Model

All results suggest that UV-structures will cause UV-phenomenon, and it is UV-phenomenon that makes problems difficult. In this paper, to avoid the UV-phenomenon, we focus on countermeasures the phenomenon that individuals in local-valleys pull individuals in the opt-valley by the crossover operator. Here, we alter the framework of conventional GAs, "all individuals search among the whole search space together, and crossover wherever they are", and introduce **Innately Split Model** in which several groups search among small areas independently.

// Innately Split Model //

1. Individuals are separated into G groups. Each group consists of N individuals.
2. **Each group is initialized within a small area respectively**.
3. In the crossover phase, parents are selected from the same group. The alternation model in each group is not restricted to be fixed.
4. When two groups seem to search among the same area, kill one group and generate a new group.
5. When a group stand still long with bad quality, kill it and generate a new group.

(N.B. In 2, for example, a group is formed with solutions once mutated from one randomly generated solution. Of course, desirable extent depends on problems. In 3, we employ CCM. 4 and 5 are not essential but optional items to accelerate the evolution.)

We are convinced that, it is imperfect to search only among **the most** hopeful area in harder problems, it is necessary to search among **several** hopeful areas simultaneously. By ISM, it is expected that each group can search among close region to the initialized area, without being deceived by the apparent hopefulness of local-valleys located far from the area now they are searching. As another countermeasure against the UV-phenomenon, clustering of population seems to be a natural idea to employ. However, we consider it is difficult to avoid the UV-phenomenon especially caused by hypothesis 2 in early generations only with clustering. Moreover, clustering technique is very awkward in general.

6 Apply ISM to JSP / Experiments

In this section, we apply ISM to JSP which is a very hard benchmark. There enumerated all configurations of our experiments. Previous experiments on JSP were done with the same condition except for the alternation model.

- GTb method (see Appendix) is also used as the enforce method in addition to GT method [Giffler 60]. GTb is a stronger enforcement for the quality of individuals.
- LR-method [Yamada 96,97] is used. By this method, we can expect for better enforcement, and the absorption of bad biases caused by the enforcement.
- Couple Weeding Mutation (CWM) is introduced. This is a new mutation system that, if and only when parents which have the same quality are selected, the mutation is used instead of the crossover. The reason why we introduce CWM bases on the idea and experiences on JSP, if parents which have the same quality are selected, they are probably ordinary, maybe are local-optimums, so go for nothing for the benefit of the search.
- Job-order based mutation [Ono 98] is used as the mutation operator.
- The crossover operator is JOX[Ono 96]. Though more effective crossovers MSXF [Yamada 96,97] and EDX[Sakuma 2000] were proposed, they contain Simulated Annealing(SA) algorithm with CB neighborhood unique to JSP. As it is not the crossover operator but the alternation model that this paper deals with, we employed JOX, a very simple crossover.

We applied our proposal method to the well-known large-size instances of JSP, ten tough problems(abz is from Adams,Balas,Zawack; la is from Lawrence). We set G=40, N=30, and stopped the algorithm if the GA found the optimum or 50 hours past. As ISM contains many inner trials structurally, it is better to continue one trial with long time. YN96(SA) [Yamada 95,96], the best algorithm proposed before now, set the limit time 3 hours, and did 20 trials. So from the viewpoint of total experiment time, our limit time 50 hours is not too long. All experiments are done on Windows98(Celeron 400MHz) , coded by Delphi4 (Object Pascal).

Table3 shows, name of instances, the optimum confirmed by Branch And Bound method or Upper Bound(UB), best solutions of ISM, time at ISM detected them, and best solutions of CCM with even configuration. ISM found the optimum/UB in 6 instances. And, 4 of them were found comparatively in a matter of hours. We can insist that ISM is perfectly superior to CCM.

Some GA/GLS[Ulder 90] methods were applied to JSP. We compare our results to other GA/GLS approaches (EDX00[Sakuma 2000], YN97[Yamada 96,97], Ono96[Ono 96], Matt[Mattfeld 94]). Comparison results are also summarized on Table3, we can see the proposal method is superior or equal to all methods in all instances. We also compare our results with famous approximation algorithms. Nowi is the taboo search algorithm proposed in [Nowicki 93], Aarts is the SA method proposed in [Aarts 94], Appl is a shuffle algorithm method proposed in [Applegate 91]. And YN96 is the improved SA method proposed in [Yamada 95,96], which used to be the best approximation approach applied to JSP. Comparison results are also summarized on Table3. We can see the proposal method is almost superior to all methods in all instances.

What is more, ISM found UB solutions in abz9(679),la29(1153) at additional experiments. Furthermore, we picked up swv1, swv6, yn4 as more difficult or large-size instances, and applied ISM with 50 hours. ISM found 1424

Table 3. Results of experiments in 10 tough problem using ISM, comparison with CCM and other approaches

instance	Result	time	CCM	EDX	YN97	Ono	Matt	Nowi	Aarts	Appl	YN96
*abz7(*656)*	664	19h27m	671	670	678	680	672	-	668	668	665
abz8(669)	669	43h52m	671	683	686	685	683	-	670	687	675
abz9(679)	683	37h8m	688	686	697	702	703	-	691	707	686
*la21(*1046)*	*1046	3h19m	1052	*1046	*1046	1050	1053	1047	1053	1053	*1046
*la24(*935)*	*935	1h31m	938	*935	*935	944	938	939	*935	*935	*935
*la25(*977)*	*977	4h46m	984	*977	*977	984	*977	*977	983	*977	*977
*la27(*1235)*	*1235	29h18m	1249	1236	*1235	1258	1236	1236	1249	1269	*1235
la29(1153)	1157	36h10m	1167	1167	1166	1189	1184	1160	1185	1195	**1154**
*la38(*1196)*	*1196	47m	1208	*1196	*1196	1202	1201	*1196	1208	1209	1198
*la40(*1222)*	1224	11h25m	1228	1224	1224	1235	1228	1229	1225	***1222**	1228

in swv1(UB1418, from Storer, Wu, Vaccari), 1702 in swv6(UB1696), 971 in yn4(UB972, from Yamada). These results suggest the robustness of ISM for harder instances. We can expect stronger robustness by the force of non-GA methods, e.g. shuffle algorithm, or EDX+MSXF.

7 Conclusion and Future Work

Conventional GAs can be well applied to problems that are not globally multimodal. However, it is often observed that they fail to find the optimum and converge to a local optimum located far from it. In this paper, we defined this situation the "UV-phenomenon", proposed the UV-structure hypothesis that would cause the UV-phenomenon, and verified them by function optimization problems and JSP. Next, we proposed ISM that could avoid the UV-phenomenon, and applied ISM to JSP. Excellent performance of ISM on JSP suggests that ISM is superior to conventional GAs (e.g. to CCM , or obviously to simple GAs) in herder problems. Consequently, we are convinced that hypothesized UV-structures surely cause the UV-phenomenon.

We showed the performance of ISM on JSP, then we would like to apply ISM to other hard benchmarks, e.g. QAP (new instances are on http://www.fe.dis. titech.ac.jp/ ~ psyche/qap/top.htm), or function optimization problems. As a future work, formulations of UV-structures and further analysis of the UV-phenomenon are desirable. They would show a new orientation of GAs.

References

[Boese 1994,1995] K.Boese, "Cost versus Distance in the Traveling Salesman Problem," Tech.Rep. TR-950018, UCLA CS Department, 1995
K.D. Boese and S.Muddu, "A New Adaptive Multi-Start Technique for Combinatorial Global Optimizations," Operations Research Letters 16, pp. 101-113, 1994

[Takahashi 99] Takahashi, O. "A Distance Dependent Alternation Model on Real-coded Genetic Algorithms" IEEE International Conference on Systems, Man, and Cybernetics, pp. 619-624, 1999

[Ono 97] Ono, I. "A Real-coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distribution Crossover", Proc. of 7th Intr. Conf. on Genetic Algorithms , pp 246-253, 1997

[Kita 99] H. Kita : A Functional Specialization Hypothesis for Designing Genetic Algorithm , IEEE International Conference on Systems, Man, and Cybernetics (1999)

[Sakuma 2000] J.Sakuma, S.Kobayashi : Extrapolation-Directed Crossover for Job-shop Scheduling Problems ; complementary combination with Job-based Order Crossover, GECCO 2000 , to appear

[Aarts 94] Aarts, E. , van Laarhoven, P. , Lenstra, J and Ulder, N. : A Computational Study of Local Search Algorithms for Job-shop Scheduling, ORSA J. on Comput, Vol. 6, No. 2, pp. 118-125 (1994)

[Applegate 91] Appleegate, D. and Cook, W. : A Computational Study of the Job-shop Scheduling Problem, ORSA J. on Comput. , Vol. 3, No. 2, pp. 149-156 (1991)

[Giffler 60] Giffler, B. and Thompson, G. : Algorithms for Solving Production Schedulng Problems, Oper. Res. , Vol. 8, pp. 487-503 (1960)

[Mattfeld 94] Mattfeld, D. C. , Kopfer, H. and Bierwirth, C. : Control of Parallel Population Dynamics by Social-like Behavior of Ga-individuals, 3rd PPSN (1994)

[Nowicki 93] Nowicki, E. and Smutnicki, C. : A Fast Taboo Search Algorithm for the Job Shop Problem, Institute of Engineering Cybernetics, Technical University of Wroclaw, Poland. , Vol. Preprinty nr 8/93 (1993)

[Ono 96] Ono, I. , Yamamura, M. and Kobayashi, S. : A Genetic Algorithm for Job-shop Scheduling Problems Using Job-based Order Crossover, Proceedings of 1996 IEEE International Conference on Evolutionary Computation, pp. 547-552 (1996)

[Ono 98] Ono, I. and Kobayashi, S. : A Genetic Algorithm Taking Account of Characteristics Preservation for Job-shop Scheduling Problems, Proceedings of the 5th conference on Intelligent Autonomous Systems, pp. 711-718 (1998)

[Ulder 90] Ulder, N., Pesch, E., van Laarhoven, P., Bandelt, H.J. and Aarts, E.: Genetic Local Search Algorithm for the Traveling Salesman Problem, 1st PPSN, pp.109-116

[Yamada 95,96] Yamada, T. and Ryohei, N. : Job-shop Scheduling by Simulated Annealing Combined with Deterministic Local Search, Kluwer Academic Publishers, MA, USA (1995); Transactions of Information Processing Society of Japan Vol.37 No.4 , pp.597-604(1996)

[Yamada 96,97] Yamada, T. and Ryohei, N. : Scheduling by Generic Local Search with Multi-Step Crossover, Proceedings of the 4th conference on 4th PPSN , pp. 960-969 (1996); Transactions of Information Processing Society of Japan Vol 38 No.6, pp.1126-1138(1997)

Appendix

GTb method successively remakes a schedule better. In each phase of remaking,
Let C be the set of operations that can be executed (C$\neq \emptyset$).
Let m_c be the machine at which a operation c$\in$C is executed.
Let g_m be the operation which is scheduled to be executed at first in machine m (not necessarily $g_m \in$C).
Let St(c) be the fastest time to start the operation c$\in$C.
Let Et(c) be the fastest time to finish the operation c$\in$C.
Let t_m be the time machine m finish last operation.
In these notations,
Phase1. execute any(if exist) g_m $\in$C s.t. t_m=St(g_m)
$\hookleftarrow$Phase 1.
Phase2. C_2:={c$\in$C|Et(c)$\leq$St(g_{m_c})}, if $C_2 = \emptyset$ $\hookrightarrow$Phase 3.
else execute any c_0 s.t. St(c_0)=$\min_{c \in C_2}$St(c)
$\hookleftarrow$Phase 1.
Phase3. C_3:={$g_m \in C$ }, if $C_3 = \emptyset$ $\hookrightarrow$Phase 4.
else execute any c_0 s.t. Et(c_0)=$\min_{c \in C_3}$Et(c)
$\hookleftarrow$Phase 1.
Phase4. execute any c_0 s.t. Et(c_0)=$\min_{c \in C}$Et(c), $\hookleftarrow$Phase 1.

Calculated amount of GTb is about 1.3 times bigger than GT method experientially. And diversity of offsprings are tunable by loosening the attribute of selection of executing operation in phase 2,3,4.