# A Test Function with Full Controllability over Overlapping and Confliction between Sub-Problems

**Jen-Hao Chang**
**Chung-Hsiang Hsueh**
**Hsuan Lee**
**Tian-Li Yu**
**Tsung-Yu Ho**

TEIL Technical Report No. 2009002
January, 2009

Taiwan Evolutionary Intelligence Laboratory (TEIL)
Department of Electrical Engineering
National Taiwan University
No.1, Sec. 4, Roosevelt Rd., Taipei, Taiwan

http://teil.ee.ntu.edu.tw

# A Test Function with Full Controllability over Overlapping and Confliction between Sub-Problems


| | |
|---|---|
| Jen-Hao Chang | backitup@teilab.ee.ntu.edu.tw |
| Chung-Hsiang Hsueh | r99921044@ntu.edu.tw |
| Hsuan Lee | r99921048@ntu.edu.tw |
| Tian-Li Yu | tianliyu@cc.ee.ntu.edu.tw |
| Tsung-Yu Ho | d98921029@ntu.edu.tw |


Jul, 2011


**Abstract**

In the real world, problems with overlapping sub-problems are common. In these problems, confliction between sub-problems are common. To study confliction between sub-problems, overlapping need to be studied first. To study overlapping, test functions with full controllability over overlapping are needed. With full controllability over overlapping, experiments can be implemented on desired extent of overlaps, and effects of overlaps can be precisely estimated and separated while confliction is being studied. However, existing test functions does not satisfy the requirement. In this paper, a test function with full controllability over overlapping is proposed. Four crossover methods capable of solving problems with overlaps are used and compared to study overlapping and confliction. Results show that the problem difficulties to these crossover methods exponentially increases with the extent of overlaps, and the effect of confliction decades while the extent of overlaps increases.


## 1 Introduction

In real world applications, overlapped sub-problems are common. Sometimes, no matter how hard we try decomposing a problem, interactions between sub-problems are unavoidable. To cope with real world applications efficiently, overlapping needs to be handled properly. In the area of genetic algorithm (GA), however, there are few researches related to the topic of overlapping (9; 10; 12; 6) , and even fewer test functions for overlapping(9; 12; 7). In problems with overlaps, confliction between sub-problems commonly happens, but there is no research on this phenomenon. This paper tries to pave the way for future researches on overlapping and confliction. We propose a test function with full controllability over overlapping. With full controllability, experiments on overlapping and confliction can be easily preformed.

In problems which highly correlated variables, or genes, are not tightly encoded, there exist several techniques to identify and to group those variables, like ecGA (2), DSMGA (1; 11), $D^5$ (8), etc. By using these techniques, related genes, or building blocks (BBs) can be identified. BB-wise uniform crossover (XO) can improve efficiencies of GA on problems without overlapping BBs. However, as shown in figure 1, on problems with overlapping building blocks, BB-wise uniform crossover performs poorly. In order to perform well on problems with overlapping BBs, modified crossover methods might be needed, and overlaps need to be studied further.

To study overlapping, test functions with known global optimum, controllability of overlaps and homogeneous overlapping structures are needed. Yu *et al.* (12) propose a test function with cyclical overlaps, shown in Figure 2(a). The test function is coarse and specially designed for theoretical development. Two-dimensional Ising spin-glass problem (7), shown in Figure 2(b), is used to test the performance of GAs (10; 6). Nevertheless, 2D Ising spin-glass problem lacks the ability to control the extent of overlaps, and the global optimum is hard to find. The test function proposed by Tsuji *et al.* was the first practical test function for overlaps, (9). It provides controllability of overlaps and known global optimum, but it does not provide the ability of construct homogeneous

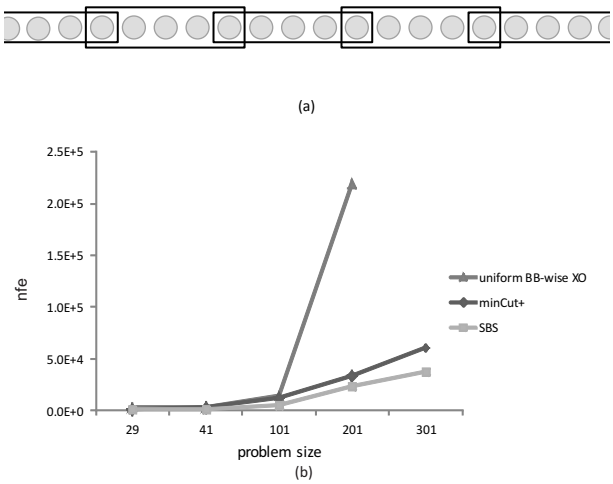


Figure 1: (a) is a problem with simple overlapping structure. In this problem, every BB (large black squares) is composed of five genes (circles) and overlapping with two other BBs, except the first and the last BB. All BBs are $trap_5$ functions. (b) is the number of function evaluations (nfe) used by : BB-wise uniform crossover, crossover method proposed by Tsuji *et al.* (minCut$^+$), and crossover method proposed by Yu *et al.* (SBS). The last two crossovers will be introduced in Section 2. Among these crossovers, uniform BB-wise crossover performs especially inefficiently on this problem.

overlapping structures. The purpose of this paper is to design a test function with full controllability over overlapping, which satisfies all the above requirements. With full controllability of overlapping, it is easier to study and analysis overlapping, confliction, or other phenomenons about overlaps.

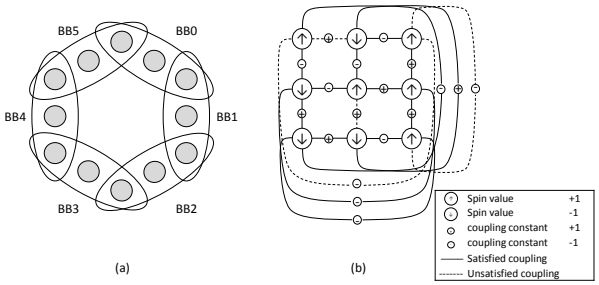


Figure 2: (a) A test function with cyclical overlaps proposed by Yu *et al.* (b) A $3 \times 3$ 2D Ising spin-glass problem.

In Section 2, four crossovers capable of solving problems with overlapping BBs are introduced. In Section 3, they are then compared with each other on the test function proposed by Tsuji *et al.* (9). In Section 4, the insufficiencies of the test function are then addressed. To overcome these insufficiencies, a test function with fully controllable overlapping structure is proposed and employed to study the effects of overlaps on the efficiencies of crossover methods. At the end of this paper, competitions of sub-problems, defined as conflicts, are employed to simulate real world applications. Conclusions are in Section 6.

## 2 Four existing crossover methods for overlapping problems

In this section, we introduce four existing crossover methods capable of solving problems with overlapping BBs: *minCut* proposed by Yu *et al.* (12), *minCut*$^+$ proposed by Tsuji *et al.* (9), *SBS* proposed by Yu *et al.* (10) and hBOA proposed by Pelikan *et al.* (4). Except hBOA, we name these methods for convenience.

### 2.1 minCut — minimal cut crossover

One early attempt to cope with overlaps is performed by Yu *et al.* (12). They construct the graph with nodes representing building blocks and edges representing overlap relationships. After two nodes are randomly selected, a minimal cut is determined to split the graph. The crossover exchanges the BBs separated by the cut. The algorithm is shown in Figure 3. Even though the idea of minimizing overlap disturbance is brilliant, minCut tends to cut the graph into unbalanced halves with one side containing only few BBs. Furthermore, because the crossover cuts the population similarly for any chosen pair of parents, randomness of crossover is reduced.

1. Constuct a graph $G = (V, E)$, where the nodes are BBs and the edges are the overlapping linkage between two nodes.
2. Randomly choose two nodes $n_1$ and $n_2$ and partition G into two subgraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ that $n_1 \in V_1$, $n_2 \in V_2$, and $|E| - |E_1| - |E_2|$ is minimal.
3. For each random pair in offspring, cross them on the cut in step 2.

Figure 3: The minCut algorithm

### 2.2 minCut$^+$ — an improved crossover of minCut

Tsuji *et al.* (9) improve minCut by eliminating nodes from the graph if a BB is identical in both parents and by eliminating edges if the site of the overlap contains identical genes in both parents. With this graph simplification procedure that differs with each pair of parents, compared to minCut, minCut$^+$ has higher rate of information exchanges. Figure 4 shows the algorithm of minCut$^+$.

1. Constuct a graph $G = (V, E)$, where the nodes are BBs and the edges are the overlapping linkage between two nodes.
2. Do the simplifications of nodes and edges on $G$.
3. Randomly choose two nodes $n1$ and $n2$ and partition $G$ into two subgraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ that $n_1 \in V_1$, $n_2 \in V_2$, and $|E| - |E_1| - |E_2|$ is minimal.
4. For each random pair in offspring, repeat steps 1 to 3.

Figure 4: The minCut algorithm

## 2.3 SBS — Strength-based Sampling

The idea of minCut is to disrupt as few BBs as possible while still achieving maximal information exchanges between any pair of BBs. Yu *et al.* (10) continue this idea and make some modifications. If a divergent BB was crossed by other BBs, the disruption might be minor; but if a BB was close to converge, crossing it might create a massive damage to it. They define strength of a BB as $Strength(\vec{x}) = \Sigma_i Entropy(x_i) - Entropy(\vec{x})$, which indicates how severe the disruption can be if the BB is crossed somewhere in the middle. $\vec{x}$ is the BB, $x_i$ is the $i^{th}$ gene of the BB, and $Entropy(\vec{x})$ is the joint entropy of the BB. The idea is: if a certain number of BBs have to be disrupted, the weaker ones, according to strength, are preferred. Suppose there is a method deciding alleles of a BB, and the method will do its best to prevent disruptions of BBs. Define decided BBs as BBs with at least one decided gene. Alleles are decided with the rule: pick the strongest BB which is overlapped with a decided BB; if there is no such case, pick the strongest one which has not yet been decided. We show an example in Figure 5. When deciding alleles, they choose conditional probability as

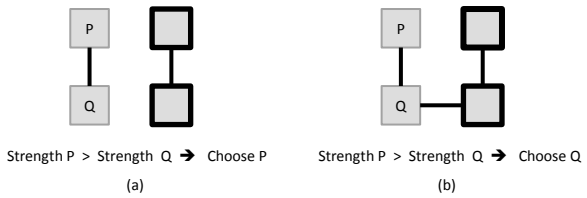


Figure 5: Squares in the figure represent BBs. Black square means a BB is decided, and grey square means the opposite. Edges means there are overlapping relationships between the two connected BBs. (a) shows that if there is no undecided BB overlapped with decided BB, the strongest undecided BB is chosen. (b) shows that if there are undecided BBs overlapped with decided BBs, the strongest undecided BB is chosen.

the way to achieve information exchange and reduce disruptions. Suppose a gene, $gene_a$, is to be decided, and genes related to it are known. The allele of $gene_a$ can be decided by sampling the conditional probability given its decided neighbours. The parameters of the conditional probability is generated from the mating pool. If all related genes have not been decided yet, $gene_a$ can be decided by its marginal probability in mating pool. Here is an example in Figure 6. Combine these

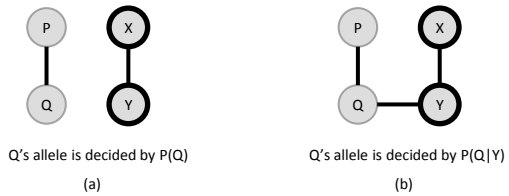


Figure 6: Nodes in the figure represent genes. Black circle means the allele is decided, and grey circle means the opposite. Edges means there are overlapping relationships between the two connected genes. (a) shows the situation that all related genes are not decided. The allele is decided by sampling marginal probability. (b) shows that when some related genes are decided, alleles are decided by sampling conditional probability given alleles of those genes and their statistics in mating pool.

two ideas, Yu *et al.* propose SBS, (10). Here is the algorithm of SBS in Figure 7.

1. Calculate strength of every BB.
2. Select a BB to be decided by using rules described in Figure 5, and randomly select an undecided gene in it.
3. If the gene has parents, use conditional probability given informations of its parents to sample the gene. Otherwise, use marginal probability of the gene to sample itself.

Figure 7: The algorithm of SBS

### 2.4 hBOA — Hierarchical BOA

Hierarchical Bayesian optimization algorithm, hBOA, was proposed by Pelikan and Goldberg. (4). Hierarchical BOA is a modified version of Bayesian optimization algorithm, BOA (5). Unlike ordinary genetic algorithms, BOA tries to learn the distribution behind the population and represent it with a Bayesian network (3). After the learning procedure, offspring is sampled from the Bayesian network by forward sampling (3). Selection is preformed to shape the distribution behind the population into the distribution of the problem. The procedure of forward sampling is shown in Figure 8. Hierarchical BOA is a modification of BOA for solving hierarchical problems. On the 2D Ising spin-glass problem described in Section 1, hBOA shows its powerful ability to solve problems with complex overlaps (6).

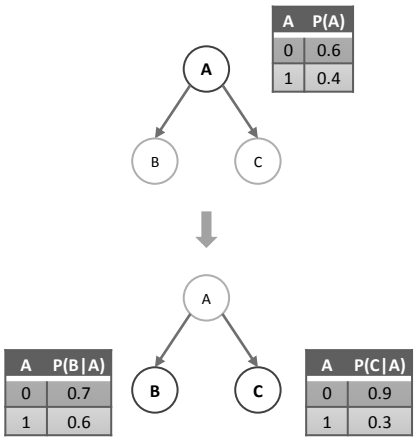


Figure 8: This figure shows the procedure of forward sampling used by BOA. Nodes in this figure represent genes. The graph is the Bayesian network learnt. First, the root of the graph is sampled by using its prior probability. After all parents are sampled, children, node B and node C in this example, are sampled by using conditional probabilities given alleles of their parents.

## 3 Comparisons on Tsuji et al.s' test function

In this section, we compare these four crossovers described in Section 2 with the test function proposed by Tsuji *et al.*, which is the first practical test function with controllable extent of overlaps. We will first define the notations used through this paper. In the next subsection, the test function is introduced. Results of the comparison are then presented.

A chromosome $C$ of length $l$ is represented as a series of genes, $C= g_1g_2...g_i...g_l$, where the subscripts are the index of gene. The fitness of $C$ is defined as $f(C) = \sum_{i=1}^{m} f_i(G_i)$, where $m$ is the total number of sub-problems, or building blocks, $f_i$ is the fitness function of $i^{th}$ sub-problem, and

$G_i$ is an ordered set of genes related with $f_i$. A problem is said to be with overlaps if a gene belongs to two or more sub-problems. For example, $G_1 = g_1g_3g_5g_9g_{10}$ and $G_2 = g_1g_4g_5g_7g_9$ overlap with gene $g_1$ and $g_5$.

### 3.1 Test function proposed by Tsuji et al.

In the test function, fitness function of each sub-problem is a 5-bit trap function defined as:

$$trap_5(G_i) = \begin{cases} \frac{4-u}{5} & \text{,u = 0,1,2,3,4} \\ 0 & \text{,u = 5} \end{cases} \quad (1)$$

, where $u$ is the number of 1's in $G_i$.

The ordered set of genes of a sub-problem, $G_j$, is defined as follows:

$$G_j = (N(3j, \sigma^2)\text{mod } l, N(3j, \sigma^2)\text{mod } l, ..., N(3j, \sigma^2)\text{mod } l),$$

where $N(\mu, \sigma^2)$ is the normal distribution with mean $\mu$ and variance $\sigma^2$. A gene cannot be in $G_j$ more than once. Therefore, the whole fitness function of a chromosome is

$$f(C) = trap5(G_1) + trap5(G_2) + ... + trap5(G_m). \quad (2)$$

Besides $\sigma$, $\mu$ also can control the test function. Define $\omega$ as the number of BBs a gene belongs to and $\bar{\omega}$ as the average of $\omega$ of all genes in a chromosome. We have $\bar{\omega} = \frac{k}{\mu}$. The comparison between $\omega$ and $\sigma$ on the problem difficulty is studied in the next subsection.

### 3.2 Experiments with known BBs

To compare the effect of different crossovers separately, we provide each crossover method with perfect information of BBs. We perform bisections on 10 different problems to find the number of function evaluations (nfe) required to successively find the global optimum 10 times by bisection method. The maximal number of generation is limited to 200. No mutation is applied in the experiments. RTR is applied to all methods. Each crossover is tested with $l = 20$ and $k = 5$. We set $\sigma$ = 1, 5, 10, 100 and $\omega$ = 1, 1.5, 2, 2.5 to compare their impact on the problem difficulty, or nfe. Figure 9 shows the comparison between $\omega$ and $\sigma$ on nfe, or problem difficulty. The left side of Figure 9 shows the nfe of each crossover scaled with $\omega$, and the right side shows the nfe scaled with $\sigma$. As shown in the left-side figures, generally, nfe grows beyond exponentially with $\omega$ when $\sigma$ is large. In right-side figures, nfe does not increase with $\sigma$ when $\sigma$ is large. The result is reasonable because when $\sigma$ is large enough, every gene has almost the same probability to be selected. This phenomenon also occurs in large problems, as shown in Figure 10.

### 3.3 Experiments with unknown BBs

In general, relationships between genes, or BBs, are not known and must be identified with some methods. Inorder to compare with hBOA, we use DSMGA with these crossovers to identify BBs. We find nfe required to successively find the global optimum 10 times on 5 different problems with bisection. No mutation is applied. RTR is applied. The results are shown in Figure 11. It shows that hBOA performs best.

## 4 Proposed Test Function on overlapping

A good test function for overlapping researches requires not only controllability of overlaps but also the ability to construct a homogeneous structure. Heterogeneity in structure makes a problem harder without expectation. Figure 12 shows how heterogeneity can cause unexpected difficulties of a problem. If those unexpected overlaps are not noticed and handled well, comparison on those heterogeneous problems is not fair. At the beginning of this section, insufficiencies of the test function proposed by Tsuji *et al.* are discussed. We then propose a test function satisfying both requirements above. In the end, crossover methods described in Section 2 are compared by using the proposed test function.

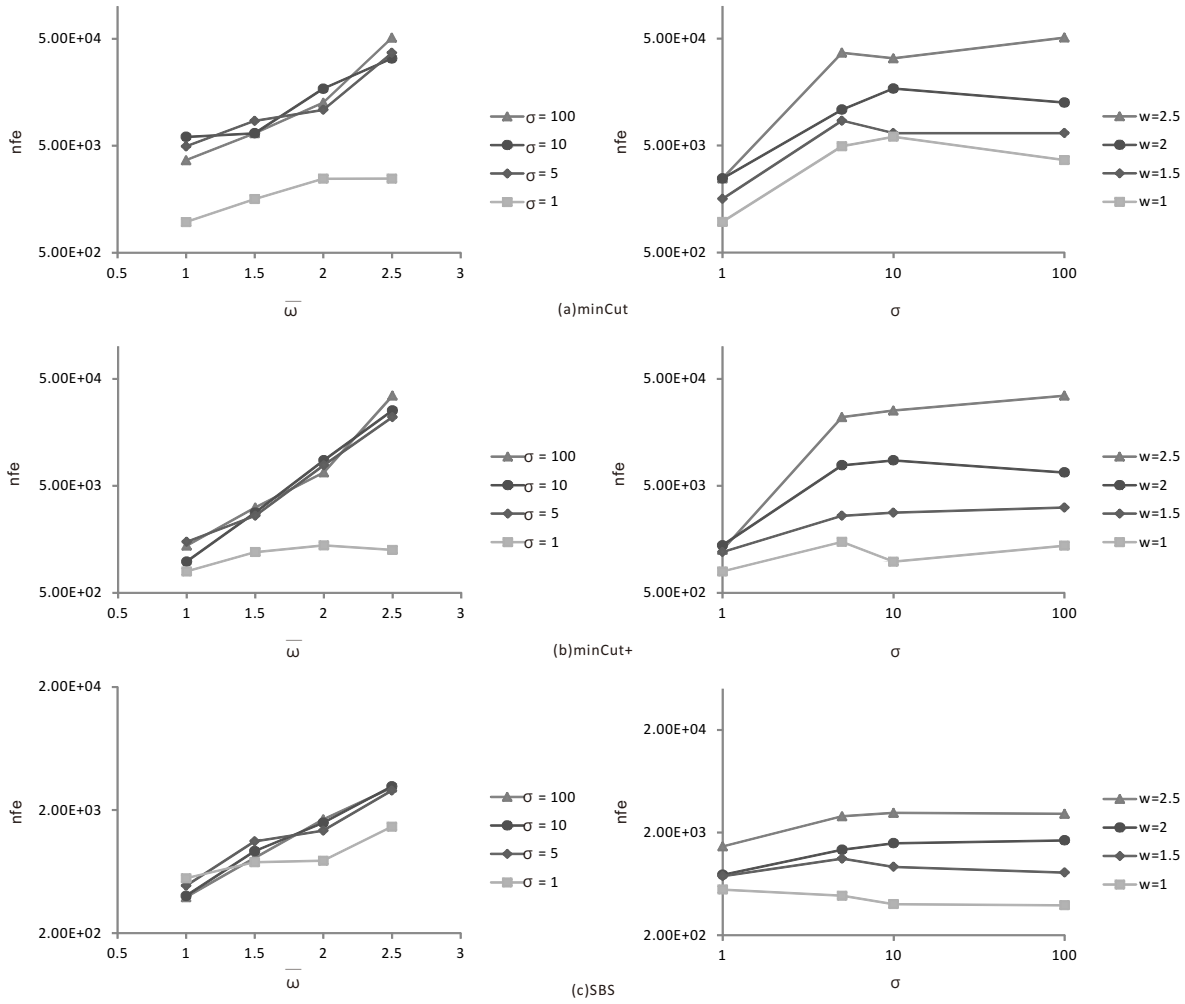


Figure 9: The comparison between $\omega$ and $\sigma$ on the problem difficulty of test function proposed by Tsuji *et al.* The left side is the results scale with $\omega$ while the right side scales with $\sigma$. The chromosome length is 20, and the size of BBs is 5.

Although the test function proposed by Tsuji *et al.* provides adjust-ability of $\omega$, it lacks the ability to construct a homogeneous overlapping structure. The standard deviation, $\sigma$, indicates the localities of the relationships among genes. Figure 13 and Figure 14 show adjustments of $\sigma$ and adjustments of $\omega$ both affect the homogeneity of structures. It shows even when $\omega$ is equal to 1, homogeneity is not guaranteed. The test function proposed by Tsuji *et al.* is good but not good enough.

## 4.1 Full controllability over overlapping

A test function with fully controllable overlapping structure is proposed. Full controllability means we can directly assign $\omega$ to each gene. It provides not only intuitive control of overlaps but also the ability to construct a homogeneous overlapping structure. To achieve full controllability, the building block assigning problem is reduced to a bipartite matching problem. By setting parameters, expected overlapping structures are constructed. Bipartite matching problem can be easily solved by finding maximal flow. The procedure to create a heterogeneous structure is trivial so is omitted. We introduce the procedure to construct a homogeneous structure. Suppose every BB contains $k \in \mathbb{N} - \{0\}$ different genes, and the chromosome length is $l \in \mathbb{N} - \{0\}$. The desired $\omega$ of each gene is $\omega_{Desired} \in \mathbb{R}$, $\omega_{Desired} \geq 1$. The size of a BB, $m$, is set to be floor of $\frac{l\omega_{Desired}}{k}$. When

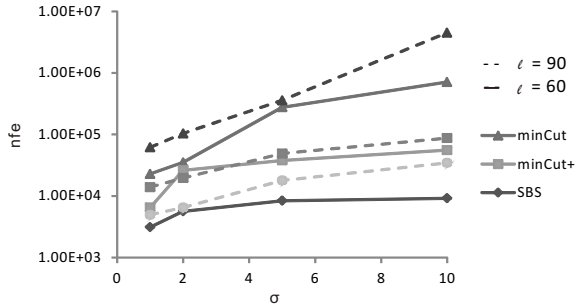


Figure 10: Results of the test function proposed by Tsuji *et al.* with $\mu = 3$, $k = 5$. Information of BBs is provided.

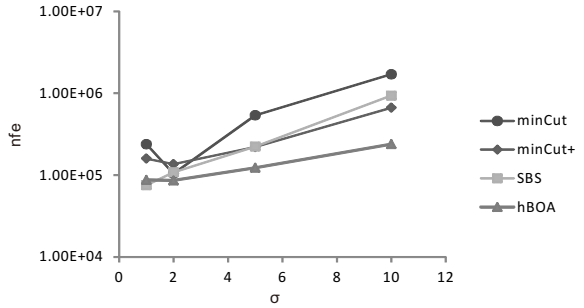


Figure 11: Results of the test function proposed by Tsuji *et al.* with $l = 60$, $\mu = 3$, $k = 5$. Information of BBs is not provided.

difference between any two flow from genes to the target is at most 1, homogeneity is achieved. To ensure homogeneity, difference between any two capacities from genes to the target is at most 1, and the total capacities from genes to the target should be equal to maximal flow; that is $mk = l\omega$. The capacities of edges from $mk \bmod l$ genes to the target are set at ceiling of $\frac{mk}{l}$, and others are set at floor of $\frac{mk}{l}$. The capacity of minimal cut, $ml$, should be larger than $mk$ to ensure every BB is fully assigned. An example of $l = 6, k = 3, \omega_{desired} = 2.5$ is shown in Figure 15. When maximal flow is achieved, $\omega$ of all genes are either $\omega_{Desired}$ or $\omega_{Desired} - 1$. Homogeneity is achieved. Because of multiple solutions of maximal flow, randomness is kept. Figure 16 shows statistics of 1000 different constructions. It shows that the average $\omega$ of all genes is always close to $\omega_{Desired}$. The number of BBs a BB overlaps is also drawn, which shows there still exist randomness in the structures. Therefore, the proposed test function can construct a homogeneous structure without loss of randomness. By using this test function, experiments on overlaps are robust.

## 4.2 Experiments on overlapping

Without lose of generality, all BBs are $trap_k^{one}$ with $k = 5$, which is defined as

$$trap_k^{one}(G) = \begin{cases} c\frac{k-1-u}{k-1} & u < k \\ 1 & u = k \end{cases}$$

, where $c = 0.8$ is a constant, $G$ is the ordered set of genes related to the sub-problem, and $u$ is number of 1's in $G$. We compare minCut, minCut$^+$, and SBS with full information of BBs. When information of BBs is not provided, we use DSMGA with these crossovers to compare with hBOA. Each point is a result of 5 independent bisections with 10 successive runs to find the global optimum. The results are shown in Figure 17.

When the information of BBs is given, SBS outperforms minCut and minCut$^+$. When the information of BBs is unknown, hBOA performs best. SBS outperforms minCut and minCut$^+$

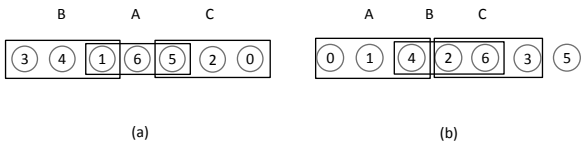


Figure 12: This figure shows how heterogeneity can cause unexpected difficulties of a problem. Black square with label A,B,and C are BBs. Circles are genes. (a) and (b) are both problems with 7 genes, and 3 BBs. Each BB contains 3 genes. In average, genes of both problems belong to $\frac{3\times3}{7} \doteqdot 1.4$ BBs, but heterogeneity makes problem (b) harder. (a) is homogeneous and (b) is heterogeneous. In (a), two genes are overlapped. In (b), because a gene is unused, three genes are overlapped causing the overlaps more complex in effective part of chromosome.

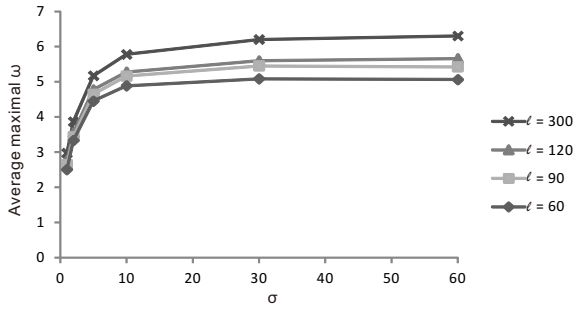


Figure 13: This figure shows statistics of the test function proposed by Tsuji *et al.* with $\mu = 3$ and $k = 5$. The $\bar{\omega}$ is 1.67. It shows adjustments of $\sigma$ affect the homogeneity of structures.

only when $\sigma$ gets larger. These results are consistent with results in Section 3.2. They both show that the difficulties of problems with overlaps are highly related to $\omega$. The nfe required to solve this problem approaches an exponential function of $\omega$ for most of the existing methods, not a beyond exponential function as in Section 3.2. It is reasonable because problems with homogeneous overlapping structures should be easier than problems with heterogeneous overlapping structures.

# 5 Proposed test function on confliction

In real world problems, one particular about problems with overlaps is that overlapped sub-problems may compete with each other. This phenomenon especially happens when local optima conflict with each other. As a result, local optima might not compose the global optimum. We call this phenomenon confliction. Studying confliction on heterogeneous overlapping structure is difficult. It is hard to discriminate whether effects result from extra-overlaps or conflicts. Studying confliction on homogeneous overlapping structure is relatively easy. Influence of overlaps can be removed, and conflicts can be studied separately.

## 5.1 Study confliction

To study confliction, the global optimum should be known in advance. However, the global optimum of a conflictive problem is usually hard to find. As a start, we use enumeration to find the global optimum and focus on homogeneous structure and sub-problems with same maximal fitness. Suppose there exist two kinds of sub-problems, defenders and competitors, and their optima conflict with each other. Suppose there are fewer competitors. Define the value of conflict as the number of competitors, so conflict is at most $\frac{m}{2}$, where $m$ is the number of sub-problems, or BBs. We choose defenders as $trap_k^{one}$ functions. To study the effect of conflict separately, the fitness functions of competitors should similar to those of defenders. As a consequence, the fitness

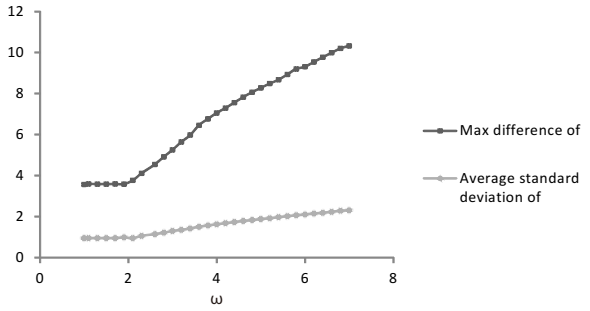


Figure 14: This figure shows statistics of the test function proposed by Tsuji *et al.* with $\mu = 3$, $l = 50$, and $\sigma = 100$. It shows adjustments of $\omega$ affect the homogeneity of structures.

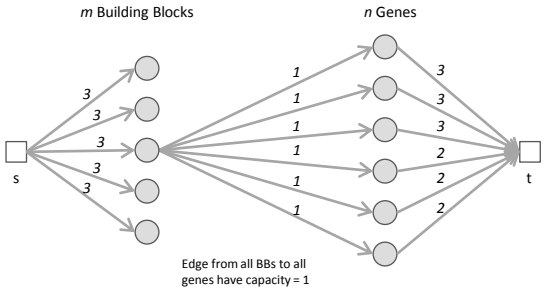


Figure 15: The flow graph of $l = 6, k = 3, \omega_{Desired} = 2.5$. Six nodes represent genes, and $floor(\frac{l\omega_{Desired}}{k}) = 5$ nodes represent BBs. All edges from source to BBs have capacity equal to $k = 3$. Edge from all BBs to all genes have capacity = 1. Three edges from gene to the target have capacity equal to 3. The others have capacity equal to 2.

functions of competitors are defined as

$$ufo_k^{zero}(G) = \begin{cases} c & u = k \\ \frac{k-1-u}{k-1} & u < k \end{cases} \tag{3}$$

, where $c$ is a constant, $G$ is the ordered set of genes related to the sub-problem, and $u$ is number of 1's in $G$. Figure 18 shows an example of $ufo_5^{zero}(G)$. There are only slight differences between $trap_k^{one}$ and $ufo_k^{zero}$, and the optimum of $trap_k^{one}$ conflict with the optimum of $ufo_k^{zero}$.

### 5.2 Experiments on confliction

Experiments on $l = 20$, $k = 5$ are preformed. No information about BBs is given. RTR is adopted. Each point is the average nfe of 5 different problems by using bisections with 10 successive runs to find the global optimum. Figure 19 shows the results. The slopes of the curves of SBS and hBOA are gentler than those of minCut and minCut$^+$. It means that SBS and hBOA handle conflicts better than minCut and minCut$^+$. When overlaps are moderate, or $\omega$ is small, the effects of conflicts are obvious; when overlaps are severe, or $\omega$ is large, the effects of conflicts are relatively minor. These results suggest that when dealing with conflictive problems with slight overlaps, we might want to hBOA.

## 6 Conclusion

This paper tries to pave the way for future researches on overlapping, confliction, or other phenomenons about overlaps. A test function with full controllability over overlapping is proposed. By using the proposed test function, the number of BBs a gene belongs to, or $\omega$, of each gene

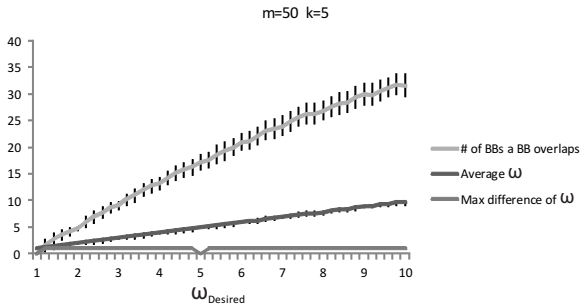


Figure 16: This figure shows the relationship between $\omega_{desired}$ and $\bar{\omega}$ of structures constructed by the proposed test function. Standard deviation is used to draw error bars. Each point is a result of 1000 different assignments. It shows that $\bar{\omega}$ is always close to $\omega_{Desired}$, and the maximal difference of $\omega$ between any two genes is 1.

can be directly assigned, and homogeneous structures can be easily achieved without loss of randomness. With the two characteristics, experiments on any desired extent of overlapping can be implemented, and overlapping can be focused or separated from the phenomenons to be studied.

Using this test function, four crossover methods are used and compared to study overlapping and confliction. Results of experiments on overlapping show that the difficulties of problems with overlapping to these crossover methods increase exponentially with the extent of overlapping. When studying confliction, we define confliction as the competitions between sub-problems, and the value of confliction, or conflict, as the number of competitive sub-problems, or competitors. Results show that the effect of confliction decrease as the extent of overlapping increases. Overall, SBS (given inflormation of BBs) and hBOA (unknown BBs) outperform minCut and minCut$^+$ on problems with overlaps or conflicts.

In order to efficiently handle problems with overlapping sub-problems in real world applications, both overlapping and confliction need to be studied further. Experiments on problems with large size need to be implemented, and an algorithm to find the global optima of a conflictive problem is needed. Results in this paper also show that some crossover methods have greater ability against conflicts. It might be an interesting and worthy topic to find out the source of this ability.

# References

[1] K.-C. Fan, J.-T. Lee, T.-L. Yu, and T.-Y. Ho. Interaction-detection metric with differential mutual complement for dependency structure matrix genetic algorithm. In *IEEE Congress on Evolutionary Computation*, pages 1–8, 2010.

[2] G. Harik and G. Harik. Linkage learning via probabilistic modeling in the ecga. Technical report, 1999.

[3] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[4] M. Pelikan and D. E. Goldberg. Hierarchical bayesian optimization algorithm = bayesian optimization algorithm + niching + local structures. pages 525–532. Morgan Kaufmann, 2001.

[5] M. Pelikan, D. E. Goldberg, and E. Cantu-Paz. BOA: The bayesian optimization algorithm. In W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1, pages 525–532, Orlando, Florida, USA, 13-17 July 1999. Morgan Kaufmann.

[6] M. Pelikan, M. Pelikan, D. E. Goldberg, and D. E. Goldberg. Hierarchical boa solves ising spin glasses and maxsat. In *In Proc. of the Genetic and Evolutionary Computation Conference (GECCO 2003), number 2724 in LNCS*, pages 1271–1282. Springer, 2003.

[7] D. Sherrington and S. Kirkpatrick. Solvable Model of a Spin-Glass. *Phys. Rev. Lett.*, 35:1792, 1975.

[8] M. Tsuji, M. Munetomo, and K. Akama. Modeling dependencies of loci with string classification according to fitness differences. In *GECCO (2)*, pages 246–257, 2004.

[9] M. Tsuji, M. Munetomo, and K. Akama. A crossover for complex building blocks overlapping. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, GECCO '06,

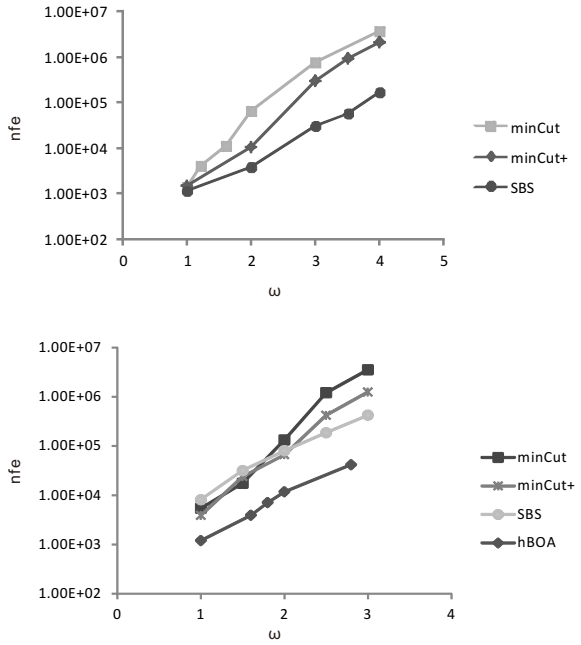


Figure 17: Comparison on the proposed test function. The chromosome length is 30. (a) shows the results with perfect BB information. (b) shows the results without perfect BB information.

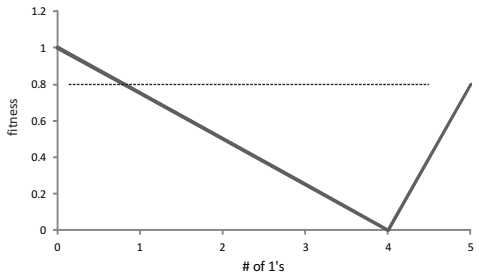


Figure 18: This figure shows the $ufo_5^{zero}$ function. In order to study conflict separately, the $ufo_5^{zero}$ function is similar to the $trap_5^{one}$ function.

pages 1337–1344, New York, NY, USA, 2006. ACM.

[10] T.-L. Yu, D. E. Goldberg, K. Sastry, C. F. Lima, and M. Pelikan. Dependency structure matrix, genetic algorithms, and effective recombination. volume 17, pages 595–626, Cambridge, MA, USA, December 2009. MIT Press.

[11] T.-L. Yu, D. E. Goldberg, A. Yassine, and Y.-P. Chen. Genetic algorithm design inspired by organizational theory: Pilot study of a dependency structure matrix driven genetic algorithm. In *GECCO*, pages 1620–1621, 2003.

[12] T.-L. Yu, K. Sastry, and D. E. Goldberg. Linkage learning, overlapping building blocks, and systematic strategy for scalable recombination. In *Proceedings of the 2005 conference on Genetic and evolutionary computation*, GECCO '05, pages 1217–1224, New York, NY, USA, 2005. ACM.

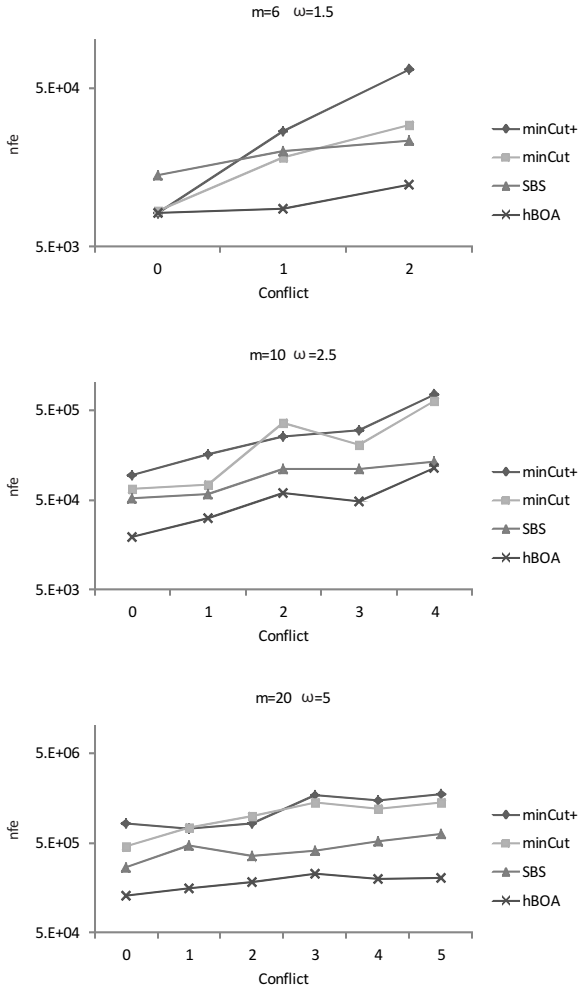


Figure 19: This figure shows the phenomenon that the effect of conflicts decreases as the extent of overlaps increase. Each point is a result of 5 different bisections with 10 succeeding runs to find the global optimum.