システムＬＳＩ設計技術　106－5
(2002.　5.　23)

シーケンスペアに基づく配置解空間の効率的なＳＡ探索のための隣接解選択

田湯　　智†　　小畑　貴之†　　金子　峰雄†

†北陸先端科学技術大学院大学 情報科学研究科　〒923–1292 石川県能美郡辰口町旭台 1–1

E-mail: †{tayu,t-obata,mkaneko}@jaist.ac.jp

あらまし　モジュール配置問題は，ＶＬＳＩレイアウト問題の中でも最も重要な問題であり，特に，面積最小化問題に関しては，多くの発見的手法の開発が行われてきた．しかし，近年のＶＬＳＩ技術の発展による高集積化により，発見的手法による最適化手法の開発は非常に困難となり，ＳＡなどのストキャスティックな手法が用いられるようになってきた．本研究では，シーケンスペアに基づくＳＡ探索のための解空間探索における遷移確率を与えることで解空間を効率的に探索する手法を提案し，提案手法をランダムに生成したモジュールとベンチマーク ami49 に適用することで実験的に提案手法のの有効性を示す．

キーワード　シーケンスペア，シミュレーテッドアニーリング，解空間

Efficient Search on Solution Space Based on Sequence-Pair for Simulated Annealing Approach

Satoshi TAYU†, Takayuki OBATA†, and Mineo KANEKO†

† School of Information Science, Japan Advanced Information Science and Technology
1–1 Asahidai, Tatsunokuchi, Ishikawa, 923–1292, Japan

E-mail: †{tayu,t-obata,mkaneko}@jaist.ac.jp

Abstract　The module placement is one of the most important problem in the VLSI design. In order to obtain an area minimizing placement, a numerical number of heuristic approaches are invented. As the number of modules in recent VLSI systems becomes larger, however, it becomes harder to compute area minimizing placement by heuristics effectively. Therefore, in the recent VLSI physical design, stochastic methods come to be employed. One of the most effective stochastic methods is the simulated annealing approach. In this paper, we propose a new neighborhood structure on the solution space and neighborhood selection scheme for simulated annealing approach of the placement problem with sequence-pair coding scheme. We show that our proposed method is effective by applying it to a randomly generated module set and MCNC benchmark ami49.

Key words　Sequence-Pair, Simulated Annealing, Solution Space.

1. Introduction

The module placement problem has been investigated in connection with the optimization problem of VLSI physical design and it is one of the most important subproblems. This problem is to place given modules on the plane without overlaps and its objective is to minimize the area of the minimum bounding box. There are several heuristic approaches for the problem and they have made a tremendous progress in VLSI technology. However, as the number of modules in recent VLSI systems becomes larger, it becomes harder to obtain area minimizing placement by heuristics. In order to solve this problem effectively, the stochastic methods such as simulated annealing approaches and genetic algorithms come to be employed. Simulated annealing approaches is applied to module placement together with some encoding schemes such as bounded slice-line grid (BSG) [1], sequence-pair [2], and ordered tree (O-tree) [3].

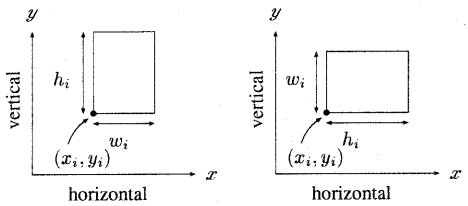
A sequence-pair is an ordered pair of permutations of modules and used as an encoding scheme in the simulated annealing approach to a placement problem. In a simulated annealing approach, we consider a solution space consisting of placements obtained from a sequence-pair by some decoding scheme and search the solution space by traversing neighbor solutions. In the conventional studies, simulated annealing approaches always search on the solution space whose neighborhood structures are given by neighborhoods defined in [2], [4].

In simulated annealing approach, if infinite computation time is allowed, we can reach an optimum with asymptotic probability one [5] though, in practical implementation with finite computation time, we often fall into some local optimum or other solutions. Hence, we need to find a way to escape from undesirable local optimums with high probability in limited computation time implementation.

In this paper, we propose a new neighborhood structure of the solution space based on sequence-pair encoding for a simulated annealing approach of the module placement problem. We also propose a weight function used to in the neighborhood selection. By the experiment, we show that our proposed neighborhood and weighted neighborhood selection scheme are effective on the exploration of the solution space.

2. Preliminaries

Each module m_i is a rectangle with its width w_i and height h_i. A module m_i must be located in a plane such that its bounding line segments have the vertical or horizontal direction on the xy-plane as shown in Fig. 1.



(a) $r_i = 0$ (b) $r_i = 1$

Figure 1 An example of a placement.

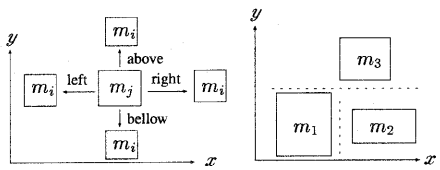
A variable r_i represents whether the rotation is applied ($r_i = 1$) or not ($r_i = 0$) to a module m_i (see Fig. 1 (a) and (b)). A module m_i is said to be *placed* if it has x- and y-coordinates and r_i is defined, where (x_i, y_i) denotes the coordinate of the left bottom corner of m_i as shown in Fig. 1. We call $R = (r_1, r_2, \ldots, r_n) \in \{0, 1\}^n$ *a rotation vector* of modules. In order to represent the lengths of horizontal and vertical segments of placed modules, we use two functions 'hor' and 'ver' defined as $\mathrm{hor}(i) = w_i$ and $\mathrm{ver}(i) = h_i$ if $r_i = 0$ (Fig. 1 (a)) and $\mathrm{hor}(i) = h_i$ and $\mathrm{ver}(i) = w_i$ if $r_i = 1$ (Fig. 1 (b)). Two placed modules m_i and m_j are said to *overlap* each other if the following four inequalities hold: $x_i < x_j + \mathrm{hor}(j)$, $x_j < x_i + \mathrm{hor}(i)$, $y_i < y_j + \mathrm{ver}(j)$, and $y_j < y_i + \mathrm{ver}(i)$. *A placement of a module set* $M = \{m_1, m_2, \ldots, m_n\}$ is the set of a tuple (x_i, y_i, r_i) of all modules m_i. A placement is said to be *feasible* if no two modules are overlapping. For a feasible placement P of modules $m_1, m_2, \ldots, m_n$, let $A(P)$ be the area of the minimum rectilinear bounding box including all modules, i.e., $A(P) = (\max_i(x_i + \mathrm{hor}(i)) - \min_i x_i)(\max_i(y_i + \mathrm{ver}(i)) - \min_i y_i)$. A feasible placement P is said to be *optimal* if it is feasible and no other feasible placement P' satisfies $A(P') < A(P)$. The objective of the placement problem is to compute a feasible placement P minimizing $A(P)$.

A sequence-pair $\langle \Gamma_+, \Gamma_- \rangle$ is an ordered pair of n module permutations Γ_+ and Γ_- and it is used as an encoding scheme for some feasible placements. Let $f_+(m_i)$ and $f_-(m_i)$ be the position of m_i appearing

in Γ_+ and Γ_-, respectively. For example, if $\Gamma_+ = (m_2, m_3, m_4, m_1)$ then $f_+(m_1) = 4$, $f_+(m_2) = 1$, $f_+(m_3) = 2$, and $f_+(m_4) = 3$. $\langle \Gamma_+, \Gamma_- \rangle$ is used to represent the following restrictions R1 and R2 of a feasible placement: For two modules m_i and m_j,

R1 if $f_+(m_i) < f_+(m_j)$ and $f_-(m_i) < f_-(m_j)$ (resp. $f_+(m_i) > f_+(m_j)$ and $f_-(m_i) > f_-(m_j)$), then $x_i + \mathrm{hor}(i) \leqq x_j$ (resp. $x_j + \mathrm{hor}(j) \leqq x_i$), i.e., m_i is *at the left* (resp. *right*) of m_j, and

R2 if $f_+(m_i) < f_+(m_j)$ and $f_-(m_i) > f_-(m_j)$ (resp. $f_+(m_i) > f_+(m_j)$ and $f_-(m_i) < f_-(m_j)$), then $y_i + \mathrm{ver}(i) \leqq y_j$ (resp. $y_j + \mathrm{ver}(j) \leqq y_i$), i.e, m_i is *below* (resp. *above*) m_j (see Fig. 2 (a)).



(a) (b)

Figure 2 Restrictions.

Figure 2 (b) shows a placement which satisfies R1 and R2 for a sequence-pair $\langle (m_1, m_2, m_3), (m_3, m_1, m_2) \rangle$.

$S = \langle \Gamma_+, \Gamma_-, R \rangle$, a sequence-pair together with a rotation vector, is useful to represent some restrictions of a placement and we use it as an encoding. We now define a one-to-one mapping $\mathcal{P}$ from such codes S to placements. Consider $S = \langle \Gamma_+, \Gamma_-, R \rangle$ of n modules. Under the decoding scheme we use, the coordinates (x_j, y_j) of the left bottom corner of module m_j can be computed by the following equations:

$$x_j = \begin{cases} \max\{x_i + \mathrm{hor}(i) | f_+(m_i) < f_+(m_j), \\ \quad f_-(m_i) < f_-(m_j)\} \\ \quad \text{if at least one such } i \text{ exists and} \\ 0 \quad \text{if otherwise, and} \end{cases} \tag{1}$$

$$y_j = \begin{cases} \max\{y_i + \mathrm{ver}(i) | f_+(m_i) < f_+(m_j), \\ \quad f_-(m_i) > f_-(m_j)\} \\ \quad \text{if at least one such } i \text{ exists and} \\ 0 \quad \text{if otherwise,} \end{cases} \tag{2}$$

where the minimum values of x_i and y_i are both 0. Let us denote the resultant placement of S by $\mathcal{P}(S)$. Such a placement can be obtained in $O(n \log n)$ time [6]. It is known that $\mathcal{P}(S)$ is feasible and there exists at least one code S whose placement $\mathcal{P}(S)$ is optimal [2]. Let $\mathbb{S}$ be the (sub-)solution space on which we search by the simulated annealing approach. A non-optimal solution S is said to be *locally optimal* on a neighborhood structure $N_{\mathbb{S}}$ of $\mathbb{S}$ if we cannot reach better solution by traversing on $N_{\mathbb{S}}$ without visiting worse solution S' than S, i.e., S' with $A(S') > A(S)$. For a locally optimal solution S, the minimum of $A'(S') - A(S)$ is called *the depth of* S if we cannot reach better solution from S without traversing at least one solution S'' with $A(S'') \geqq A(S')$. The maximum of the depth over all local optimum solution S, denoted by $d(\mathbb{S})$, is called *the depth of* $\mathbb{S}$. A simulated annealing search described in Fig. 4 in Section 3. 1 with a temperature schedule $T_0, T_1, \ldots$ on a solution space $\mathbb{S}$ guarantees to reach an optimal solution with asymptotic probability one [5] if (a) it takes infinite time under the temperature schedule, (b) the solution space $\mathbb{S}$ is finite and irreducible, (c) there exists an equilibrium distribution for the transition probability matrix, (d) $T_i \geqq T_{i+1}$ and $T_i > 0$ for all i, (e) $\lim_{i \to \infty} T_i = 0$, and

$$\text{(f)} \qquad \sum_{k=0}^{\infty} \exp(-d(\mathbb{S})/T_k) = \infty.$$

3. Neighborhood structure

3.1 Conventional works

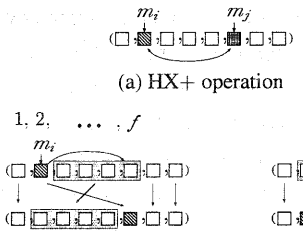
We often regard S as a solution instead of $\mathcal{P}(S)$. For some codes S and S', $\mathcal{P}(S) = \mathcal{P}(S')$ but, in this case, we regard that S and S' are different solutions though they represent the same placement. A neighborhood structure of the solution space can be represented by a set of neighborhoods. Main types of neighborhoods for this approach in the conventional studies are described as follows: Two solutions S and S' are neighborhoods of the other if and only if S' is obtained from S by one of following operations:

HX+ (−): (half-exchange) choose two modules m_i and m_j and then exchange m_i and m_j in Γ_+ (Γ_-) (see Fig. 3 (a)).

IN+ (−): (insert) choose a module m_i and a position f in Γ_+, put m_i to f and the other modules between $f_+(m_i)$ and f are shifted by 1 toward $f_+(m_i)$ (see Fig. 3 (b) and (c)).

FX: (full-exchange) choose two modules m_i and m_j and then exchange m_i and m_j in both Γ_+ and Γ_-.

RT: (rotate) choose one modules m_i and rotate it, i.e., $r_i := 1 - r_i$.



(a) HX+ operation

(b) IN+ operation for $f_+(m_i) < f$ (c) $f_+(m_i) > f$

Figure 3 Operations HX+ and IN+.

The area minimization by the simulated annealing approach can be described by using temperature schedule $\mathcal{T} = (T_0, T_1, \ldots)$ with $T_i > T_{i+1}$ and $\lim_{i\to\infty} T_i = 0$ in Fig. 4. Note that, if $A(S') \leq A(S)$, Inequality (†) in

INPUT: n modules m_i, temperature schedule $\mathcal{T}$, finish temperature $T_e > 0$, and initial solution S_0.
OUTPUT: A solution S.
begin
construct an initial solution S randomly.
set $i := 0$, $S := S_0$, and $T := T_0$.
while $T > T_e$ **do**
repeat t times **do**
choose a neighborhood S' of S
choose p with $0 < p < 1$ randomly.
if $p \leqq \exp\left(-\dfrac{A(S') - A(S)}{T}\right)$ (†)
$S := S'$.
end if
end repeate
$i := i + 1, T := T_i$.
end while
end

Figure 4 Simulated Annealing Algorithm.

Fig. 4 is always true. Let $X(T_i)$ be the random variable representing the area $A(S)$ of solution S at temperature T_i in the algorithm with $T_e \to 0$ and A_{opt} be the minimum area over all solutions. If the transition probability matrix on the solution space has an equilibrium distribution, $\lim_{i\to\infty} T_i = 0$, and $\lim_{i\to\infty} \sum_{k=0}^{i} \exp(-\frac{d}{T_k}) = \infty$ for the depth d of $\mathbb{S}$, then $\lim_{i\to\infty} X(T_i) = A_{\text{opt}}$ [5]. However, in a practical implementation, we cannot spend infinite computation time and we have to simulate it in finite computation time by some stochastic strategy. In this paper we set $T_i = r^{\lfloor i/t \rfloor} T_0$ for some integer t and real number r suth that $0 < r \leqq 1$ and $T_e > 0$ as many other researchers do.

3.2 Priority of the neighborhood selection

In this section, we give the priority for each operation based on Conjecture 1. In our neighborhood selection scheme, we choose operation O with the probability in proportion to its priority.

[**Conjecture** 1] If the difference of evaluations (the values of objective function) between two adjacent solutions becomes smaller, simulated annealing may accomplish to reach a better solution. □

Of cause, there may exist many other factors which effect the search in a sense of efficiency and the assertion of the conjecture does not always hold.

In the placement problem, however, it is hard to evaluate such a difference $|A(S_1) - A(S_2)|$ *for all pairs* of adjacent solutions S_1 and S_2 since there are an exponential number of solutions. Therefore, instead of $A(S) = W(S)H(S)$, we give other evaluations for each operation using the upper bound of $|W(S_1) - W(S_2)| + |H(S_1) - H(S_2)|$, where $W(S)$ and $H(S)$ are the vertical and horizontal width of $\mathcal{P}(S)$.

For each operation O, we define two types of weights denoted by $D(O)$ and $F(O)$ which are used in the definition of priority as follows:

$$P(O) = F(O)\left(1 - \exp(-\sqrt{A_M}D(O)/T)\right), \quad (3)$$

where A_M is the total area of modules.

We first consider HX± operation for given modules m_i and m_j. Let $d^{\text{HX}\pm}(m_i, m_j)$ be the upper bound of $|W(S_1) - W(S_2)| + |H(S_1) - H(S_2)|$ such that S_2 is obtained from S_1 by half-exchange operation for m_i and m_j. Clearly, the difference is at most $\text{hor}(i) + \text{hor}(j) + \text{ver}(i) + \text{ver}(j) = h_i + w_i + h_j + w_j$. With little more detailed analysis, we can obtain upper bound $d^{\text{HX}\pm}(m_i, m_j) = h_i + w_i + h_j + w_j - \min\{h_i, w_i, h_j, w_j\}$. For example, if $r_i = r_j = 1$ and $H(S_2) - H(S_1) = w_i + w + j$, m_i and m_j are included in a horizontal critical path of S_2, and then every vertical critical path includes at most one of m_i and m_j (see Fig. 5). $D(\text{HX}\pm)$ is defined as the average of

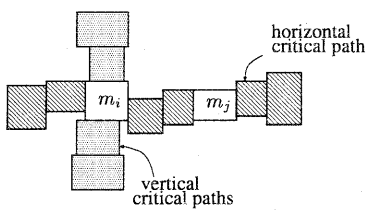


Figure 5 Horizontal and vertical critical paths.

$d^{\text{HX}\pm}(m_i, m_j)$ for all pair of two modules m_i and m_j.

It should be noted that, for operation O, $\sqrt{A_M}D(O)$ in (3) is closely rerated with $A(S') - A(S)$ of (†) in Fig. 4 and, if $\exp\left(-\frac{\sqrt{A_M}D(O)}{T}\right)$ is close to 1, i.e., $D(O)$ is very small, operation O is accepted with high probability. In this case, operation O may work as if every randomly chosen neighborhoods are accepted and, if there is some operation O' which is accepted with lower probability, O has only a small influence. On the other hand, from Conjecture 1, even though $d^{\mathrm{HX}\pm}(m_i, m_j)$ is very large for some pairs of m_i and m_j, if there are many pairs having small $d^{\mathrm{HX}\pm}(m_i, m_j)$, then we would like to give high priority to the operation. Therefore, since the operation with $d^{\mathrm{HX}\pm}(m_i, m_j) = 0$ makes no effect on the area optimization, we give the other type of priority $F(\mathrm{HX}\pm)$ as the average of $1/d^{\mathrm{HX}\pm}(m_i, m_j)$ over non-zero $d^{\mathrm{HX}\pm}(m_i, m_j)$ i.e., $F(\mathrm{HX}\pm)$ is defined by

$$F(\mathrm{HX}\pm) = \frac{1}{\binom{|M|}{2}} \sum_{m_i, m_j \in M} f(m_i, m_j), \text{ where}$$

$$f(m_i, m_j) = \begin{cases} 0 & \text{if } d^{\mathrm{HX}\pm}(m_i, m_j) = 0 \\ \dfrac{1}{d^{\mathrm{HX}\pm}(m_i, m_j)} & \text{if otherwise.} \end{cases}$$

By the similar ways, once $d^O(*)$ is defined for operation O, $D(O)$ and $F(O)$ can be defined automatically.

For IN$\pm$ operation, upper bound for given m_i is denoted by $d^{\mathrm{IN}\pm}(m_i) = w_i + h_i$ and

$$F(\mathrm{IN}\pm) = \frac{1}{|M|} \sum_{m_i \in M} f^{\mathrm{IN}\pm}(m_i),$$

where $f^{\mathrm{IN}\pm}$ is defined similarly to $f^{\mathrm{HX}\pm}$.

For FX operation, upper bound for given modules m_i and m_j is given by $|\mathrm{hor}(i) - \mathrm{hor}(j)| + |\mathrm{ver}(i) - \mathrm{ver}(j)| \leqq \max\{|w_i - w_j| + |h_i - h_j|, |w_i - h_j| + |h_i - w_j|\}$. We can easily accomplish more detail analysis for this operation. For each pair m_i and m_j with $w_i \neq h_i$ and $w_j \neq h_j$, there are following two cases: i) $\mathrm{hor}(i) < \mathrm{ver}(i)$ and $\mathrm{hor}(j) < \mathrm{ver}(j)$ or $\mathrm{hor}(i) > \mathrm{ver}(i)$ and $\mathrm{hor}(j) > \mathrm{ver}(j)$, ii) $\mathrm{hor}(i) < \mathrm{ver}(i)$ and $\mathrm{hor}(j) > \mathrm{ver}(j)$ or $\mathrm{hor}(i) > \mathrm{ver}(i)$ and $\mathrm{hor}(j) < \mathrm{ver}(j)$. There are same number of solutions for both cases. In order to consider both cases independently, we introduce two functions $d_1^{\mathrm{FX}}(m_i, m_j) = |w_i - w_j| + |h_i - h_j|$ and $d_2^{\mathrm{FX}}(m_i, m_j) = |w_i - h_j| + |w_i - h_j|$. So, the priority function is defined by

$$F(\mathrm{FX}) = \frac{1}{2}\frac{1}{\binom{|M|}{2}} \sum_{m_i, m_j \in M} \sum_{k=1}^{2} f_k^{\mathrm{FX}}(m_i, m_j),$$

where f_k^{FX} is defined similarly to $f^{\mathrm{HX}\pm}$.

For RT operation, upper bound for m_i is denoted by $d^{\mathrm{RT}}(m_i) = |w_i - h_i|$ and so

$$F(\mathrm{RT}) = \frac{1}{|M|} \sum_{m_i \in M} f^{\mathrm{RT}}(m_i),$$

where f^{RT} is defined similarly to $f^{\mathrm{HX}\pm}$.

4. Proposed Methods

4.1 New Neighborhood

We now define new types of neighborhoods by giving operations constructing them:

RF: choose two modules m_i and m_j and exchange m_i and m_j in both sequence Γ_+ and Γ_- and, if $(\mathrm{ver}(i) - \mathrm{hor}(i))(\mathrm{ver}(j) - \mathrm{hor}(j)) < 0$, then rotate both m_i and m_j (i.e., $r_i := 1 - r_i$ and $r_j := 1 - r_j$).

The weights of this operation is given as follows:

$$d^{\mathrm{RF}}(m_i, m_j) = |\max\{w_i, h_i\} - \max\{w_j, h_j\}| + |\min\{w_i, h_i\} - \min\{w_j, h_j\}| \text{ and}$$

$$F(\mathrm{RF}) = \frac{1}{\binom{|M|}{2}} \sum_{m_i, m_j \in M} f^{\mathrm{RF}}(m_i, m_j).$$

Since $d^{\mathrm{RF}}(m_i, m_j) \leqq d_k^{\mathrm{FX}}(m_i, m_j)$, for each pair of m_i and m_j, $F(\mathrm{RF}) \leqq F(\mathrm{FX})$ if $d^{\mathrm{RF}}(m_i, m_j) > 0$.

In conventional studies, rotation and reconfiguration of $\Gamma\pm$ are treated separately, and such kinds of neighborhoods are not considered. We can make many other arranges, e.g., by combining RT with some of HX$\pm$, IN$\pm$, and FX operations, but we omit the details of them for the lack of space.

4.2 Weighted Neighborhood Structure

In conventional approach, each operation is basically chosen with same probability. In this paper, we propose to assign different probabilities to each operation O according to their priority $F(O)$ and $D(O)$ which we define in this subsection. Let

$$\Omega \subseteq \{\mathrm{HX}\pm, \mathrm{IN}\pm, \mathrm{FX}, \mathrm{RF}, \mathrm{RT}\} \quad (4)$$

be a set of some operations we considered. In order to guarantee irreducible, RT $\in \Omega$, at least one of HX$\pm$ and IN$\pm$ is included in Ω, and $|\Omega| \geqq 3$ under (4).

At a high temperature T, if an operation O is accepted with probability about 1/2 and another operation O' with $1 - p$ with $p \ll 1$, then the operation O works as if every randomly chosen neighborhood is accepted though O'

is meaningful. Therefore, we set such that the expected value of an operation O to be chosen is concerning to $1-\exp\left(-\sqrt{A_M}D(O)/T\right)$ which we employ instead of $1-p$. As mentioned, the priority is given by (3). In our proposed search of simulated annealing approach, an operation $O \in \Omega$ is chosen with probability

$$R_\Omega(O) = P(O)/\sum_{O'\in\Omega} P(O'). \qquad (5)$$

The expected value of the number of iterations for operation O at temperature $T' = r^{\lfloor i/t \rfloor}T_0$ is $tR_\Omega(O)$.

5. Experimental Results

In this section, we give the experimental results of simulated annealing searches on $\mathbb{S}$ under conventional and our proposed neighborhood structures. We implement the simulated annealing approach to BenchMark ami49 and a randomly generated module sets M_{100}, where a module m_i in M_{100} is generated satisfying that $100 \leqq w_i, h_i \leqq 1000$, $\max\{h_i/w_i, w_i/h_i\} \leqq 3$, and $\min\{h_i/w_i, w_i/h_i\} \leqq 3/2$. The neighborhood structure in conventional study is given by $\Omega = \{\mathrm{RT}, \mathrm{FX}, \mathrm{IN}\pm\}$, and each operation is chosen with same probability, where we regard IN+ and IN− as one operation here. In our method, $\Omega' = \{\mathrm{RT}, \mathrm{RF}, \mathrm{IN}+, \mathrm{IN}-\}$, and each operation is chosen with the probability denoted by (5). We apply simulated annealing approach 100 times with parameters $r = 0.98$ and $t = 10n$, where $n = 49$ is the number of modules. We compute the ratio $A(S)/A_M$ of $A(S)$ to the total area A_M of modules and show the results in Table 1. Our improved method generates better

Table 1 Experimental results for ami49.

	$T_s = 10^5$, $T_e = 10^2$		$T_s = 10^6$, $T_e = 10^1$	
	Ω	Ω'	Ω	Ω'
ave.	1.0369	1.0269	1.0320	1.0236
best	1.0227	1.0174	1.0202	1.0153
worst	1.0573	1.0407	1.0430	1.0310

solutions than the conventional method.

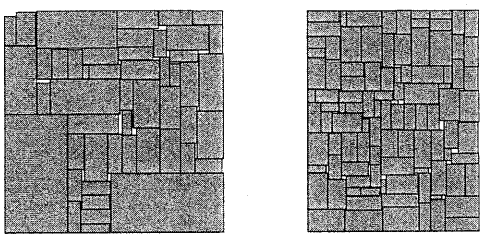
We apply our method 100 times to M_{100} and we show the results in Table 2, where we set $r = 0.98$ and $t = 10n$. For M_{100}, our method also generates better solutions. We also give the best placements for ami49 and M_{100} in Fig. 6.

6. Conclusion

In this paper, we propose a new neighborhood and a weight function of transition probability of the neighborhood selection for placement problems with the sequence-pair encoding scheme. By experimental results, we show that our proposed scheme is more effective than that of conventional studies.

Table 2 Experimental results for ami49.

	$T_s = 10^5$, $T_e = 10^2$		$T_s = 10^6$, $T_e = 10^1$	
	Ω	Ω'	Ω	Ω'
ave.	1.0344	1.0257	1.0318	1.0239
best	1.0273	1.0207	1.0251	1.0178
worst	1.0447	1.0338	1.0396	1.0309



(a) ami49 (1.015273) (b) M_{100} (1.017834)

Figure 6 Best Placements.

References

[1] S. Nakatake, H. Murata, K. Fujiyoshi, and Y. Kajitani, "Module packing based on the BSG-structure and IC layout application," IEEE Trans. Computer-Aided Design, vol. 17, no. 6, pp. 519–530, Jun. 1998.

[2] H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, "VLSI module placement based on rectangle-packing by the sequence-pair," IEEE Trans. Computer-Aided Design, vol. 15, no. 12, pp. 1518–1524, Dec. 1996.

[3] T. Takahashi, "A new encoding scheme for rectangle packing problem," Proc. Asia and South Pacific Design Automation Conference, pp. 175–178, Dec. 2000.

[4] K. Fujiyoshi, T. Ohomura, and K. Ijiri, "Solution space by sequence-pair suitable for simulated annealing search," Tech. Rep. IEICE, vol. VLD99–118, no. 659, Mar. 2000 (in Japanese).

[5] B. Hajek, "Cooling schedules for optimal annealing," Mathematics of Operations Research, vol. 13, no. 2, pp. 311–329, May 1988.

[6] T. Takahashi, "An algorithm for finding a maximum-weight decreasing sequence in a permutation, motivated by rectangle packing problem," Tech. Rep. IEICE, vol. VLD96–30, no. 201, Jul. 1996 (in Japanese).