

# 배선 밀집도를 완화하기 위한 개선된 광역배치 기법 (Improved Global Placement Technique to Relieve Routing Congestion)

오 은 경 <sup>†</sup> 허 성 우 <sup>\*\*</sup>  
(Eun Kyung Oh) (Sung Woo Hur)

**요 약** 기존의 CDP(Congestion Driven Placement) 기법[1]은 배선 밀집도에 크게 기여하는 넷에 속한 모든 셀들을 옮길 때 가능한 방향을 모두 고려하기 때문에 CPU 시간을 많이 소모한다. 본 논문에선 배선 밀집도를 완화하기 위한 빠른 배치 기법, 즉 ICDGP(Improved Congestion Driven Global Placement)라 불리는 개선된 기법을 제안한다. ICDGP는 밀집된 지역에 있는 넷에 속한 셀들의 이동 목표 지점을 정하기 위해 force-directed 기법을 사용하며, 각 셀은 이동 목표 지점으로만 옮기는 것을 고려한다. ICDGP는 또한 셀을 하나씩 옮기는 것보다 다수의 셀을 동시에 같이 옮기는 것이 낫다고 판단되면 같이 옮긴다. 실험 결과에 의하면 CDP보다 ICDGP가 배선 밀집도 면에서 좋은 배치를 생성한다. 특히 CPU 시간은 평균 36% 개선되었다.

**키워드** : 배선밀집도, 광역배치, FD 기법

**Abstract** Since previous work CDP (Congestion Driven Placement) [1] considers all possible directions to move every cell in nets which contribute highly to routing congestion, it consumes CPU time a lot. In this paper, we propose a faster global placement technique, so

· 이 논문은 2006년 정부(교육인적 자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(KRF-2006-311-D00823)

· 이 논문은 2007 학술심포지움에서 '개선된 배선밀집도 드러본 광역배치의 제목으로 발표된 논문을 확장한 것임

<sup>†</sup> 학생회원 : 동아대학교 컴퓨터공학과  
ekoh20@gmail.com

<sup>\*\*</sup> 정 회 원 : 동아대학교 컴퓨터공학과 교수  
swhur@dau.ac.kr  
(Corresponding author)

논문접수 : 2008년 1월 31일

심사완료 : 2008년 3월 24일

Copyright©2008 한국정보과학회 : 개인 목적이거나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 레터 제14권 제4호(2008.6)

called ICDGP (Improved Congestion Driven Global Placement) to relieve the routing congestion. ICDGP uses the force-directed method to determine the target locations of the cells in the nets in the congested spots, and considers only to move the target location for each cell. If moving multiple cells simultaneously is considered better than moving each cell one by one it moves multiple cells simultaneously. By experimental results, ICDGP produces less congested placement than CDP does. Particularly, the CPU time is reduced by 36% on average.

**Key words** : routing congestion, global placement, force-directed method

## 1. 서 론

VLSI 설계의 물리 설계 단계 중 주요한 위치를 차지하는 배치 과정은 배선길이(wire length), 성능(performance), 그리고 배선가능성(routability)과 같은 주요한 설계변수에 지대한 영향을 미친다.

배치의 전통적인 목표는 컷 사이즈를 줄이거나 배선 길이를 최소화 하는 것이다. 최소-컷을 찾는 기법을 사용하는 배치 툴로는 Capo[2], FengShui[3,4] Snap-on [5] 등이 있다. Terminal propagation 개념을 확장하여 배선길이를 줄이려는 시도도 최근에 있었다[6]. 배선 길이를 최소화하기 위해 simulated annealing 기법을 사용하는 툴로는 TimeberWolf[7]가 잘 알려져 있으며, 해석적 기법 또는 FD(Forced-Directed) 법을 사용하는 툴들이 있는데, FD 방법을 사용하는 기법은 그 변화가 매우 다양하다.

최소-컷을 찾는 기법이나 배선길이를 최소화하기 위한 기법은 거시적인 관점에서 볼 때 배선의 밀집도를 줄일 수 있으나 국부적으로 발생하는 높은 배선밀집도를 예방하지는 못하는 문제가 발생한다. 배치 단계에서 배선 밀집도를 고려하지 않으면 다음단계인 배선 단계가 불가능한 경우가 발생한다. 이런 문제를 예방하기 위해 최근에는 배치 단계에서 배선밀집도 및 배선가능성을 고려하는 연구가 많이 수행되고 있다. Meixner[8]는 다단계 분할을 이용하는 방법을 사용하였고, [9]에선 재귀적 분할과 annealing을 이용하여 배선가능성을 향상시키려고 했다. CDP[1]는 Mongrel[10]의 리플 이동(ripple movement) 기법과 증분 자료구조(incremental data structure)를 이용하였다.

본 논문에서는 [1]를 수정, 보완한 새로운 배선밀집도 기반 배치기법을 제안한다. 제안 기법의 특징은 기존의 [1]에서 사용한 리플 이동 기법과 증분 자료구조를 그대로 유지하므로 주어진 상세배치의 배선길이를 크게 희생하지 않으면서 배선밀집도를 해결한다는 것이다. 또한

배선밀집도를 줄이기 위한 셀의 선택 및 이동시 Kraftwerk[11]에서 제안한 FD 기법을 통해 기존의 방법보다 빠른 실행속도로 배치결과를 얻는다.

## 2. 배선 밀집도 계산

### 2.1 용어 정의

회로는 하이퍼그래프  $G=(V,E)$  ( $V$ 는 셀들의 집합,  $E$ 는 넷들의 집합)로 나타낸다. 넷  $e \in E$ 는 2개 이상의 셀들을 연결하며,  $e \subseteq V$ 로 나타낸다. 배치는 크게 광역배치(global placement)와 상세배치(detailed placement)로 구분한다. 본 논문에서는 광역 배치를 위해 칩 영역을  $n \times m$  격자로 나눈다. 격자의 각 영역을 빈(bin)이라 부른다.

배선밀집도 추정은 빈의 경계선을 이용한다. 빈의 경계선을 통과하는 배선의 개수를 추정하여 이를 배선요구량(routing demand)이라고 하며, 배선을 위해 사용가능한 자원을 배선자원 용량이라 한다.

### 2.2 배선밀집도 추정과 배선요구량 계산

그림 1에서 배선자원요구량과 배선자원용량에 관한 개요를 보인다.  $b(i,j)$ 는  $i$ 행  $j$ 열에 있는 빈을 나타낸다.  $b(i,j)$ 와  $b(i,j+1)$ 의 경계선을 통과하는 배선은 수평선이며 이에 대응하는 배선요구량과 배선자원용량은 각각  $d_h(i,j)$ ,  $c_h(i,j)$ 로 나타낸다. 마찬가지로 빈  $b(i,j)$ 와  $b(i+1,j)$ 의 경계선을 통과하는 배선은 수직선이며 이에 대응하는 배선요구량과 배선자원용량은  $d_v(i,j)$ ,  $c_v(i,j)$ 가 된다.

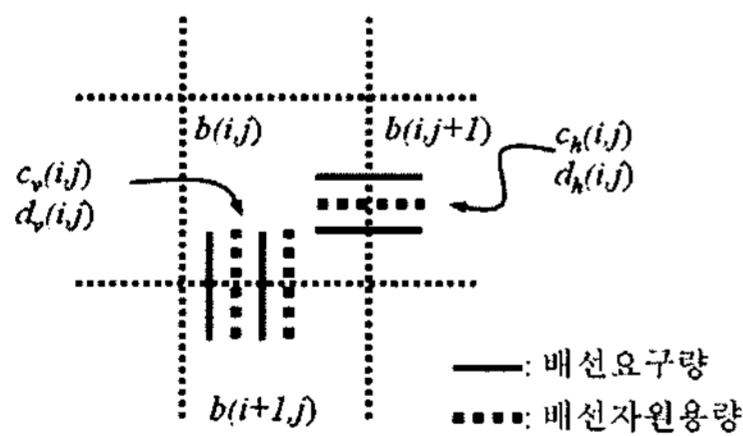


그림 1  $b(i,j)$ 의 배선요구량 및 배선자원용량

특정 넷  $e$ 에 의해 발생하는 배선요구량의 수평 및 수직성분은 각각  $d_{h_e}(i,j), c_{h_e}(i,j)$ 로 나타내고, 이는 아래 식 (1)과 (2)처럼 나타낼 수 있다.

$$d_h(i,j) = \sum_{e \in E} d_{h_e}(i,j) \quad (1)$$

$$d_v(i,j) = \sum_{e \in E} d_{v_e}(i,j) \quad (2)$$

$b(i,j)$ 에서 수직, 수평성분의 배선밀집도는 각각 다음과 같이 정의한다.

$$rc_v(i,j) = d_v(i,j)/c_v(i,j) \quad (3)$$

$$rc_h(i,j) = d_h(i,j)/c_h(i,j) \quad (4)$$

$b(i,j)$ 의 배선밀집도  $rc(i,j)$ 는 식 (5)처럼  $rc_v(i,j)$ 와  $rc_h(i,j)$ 중 큰 값으로 정의한다.

$$rc(i,j) = \begin{cases} rc_v(i,j), & \text{if } rc_v(i,j) > rc_h(i,j) \\ rc_h(i,j), & \text{otherwise} \end{cases} \quad (5)$$

넷  $e = \{A, B\}$ 라고 가정하자. 즉,  $e$ 는 두 셀  $A$ 와  $B$ 를 연결하는 넷이다. 셀  $A$ 는  $b(p,q)$ 에 있고, 셀  $B$ 는  $b(r,s)$ 에 있다고 가정하자. 두 셀을 연결하는 모든 단조경로(monotone path) 가운데 그림 2는 넷  $e$ 가  $b(i,j)$  ( $p \leq i \leq r, q \leq j \leq s$ )를 통과하는 단조 경로를 보여준다.

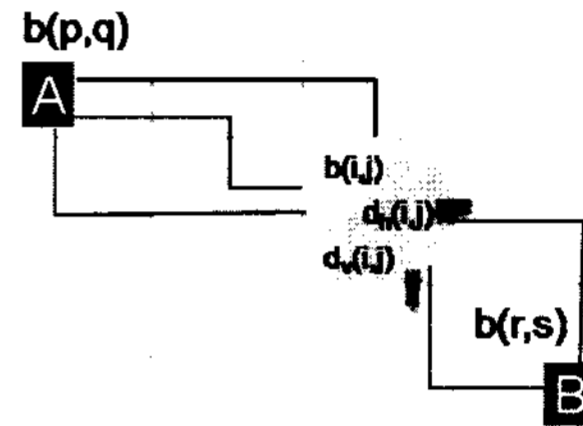


그림 2 배선요구량 계산 방법을 보이는 예

$A$ 와  $B$ 를 연결하는 모든 단조경로의 수는  $\binom{r-p+s-q}{r-p}$ 개이며, 두 빈  $b(p,q)$ 와  $b(r,s)$ 사이에는 모든 빈  $b(i,j)$  ( $p \leq i \leq r, q \leq j \leq s$ )에서 넷  $e$ 가  $A, B$ 를 연결하기 위해 필요한 수직, 수평 배선요구량  $d_v(i,j), d_h(i,j)$ 는 확률에 근거하여 식 (6), (7)과 같이 계산할 수 있다.

$$d_{h_e}(i,j) = \frac{\binom{i-p+j-q}{i-p} \binom{r-i+s-j-1}{r-i}}{\binom{r-p+s-q}{r-p}} \quad (6)$$

$$d_{v_e}(i,j) = \frac{\binom{i-p+j-q}{i-p} \binom{r-i-1+s-j}{r-i-1}}{\binom{r-p+s-q}{r-p}} \quad (7)$$

### 2.3 배선 요구량 추정을 위한 넷 모델

본 논문에서는 [1]에서처럼 배선을 위해 RMST(Rectilinear Minimum Spanning Tree) 모델을 사용하여 셀들을 연결한다. RMST는 넷에 연결된 셀들의 맨해튼 거리의 합이 최소가 되도록 셀들을 연결한 신장 트리(spanning tree)이다.

그림 3에서  $A, B, C$  세 개의 셀과 이를 연결하는 넷  $e = \{A, B, C\}$ 를 가진 배치에서 RMST와 그에 따라 계산된 배선요구량의 예를 보인다. 그림 3(a)의  $b(3,4)$ 에서 수직성분의 배선요구량은 두 간선  $(A, B), (A, C)$ 에 의해

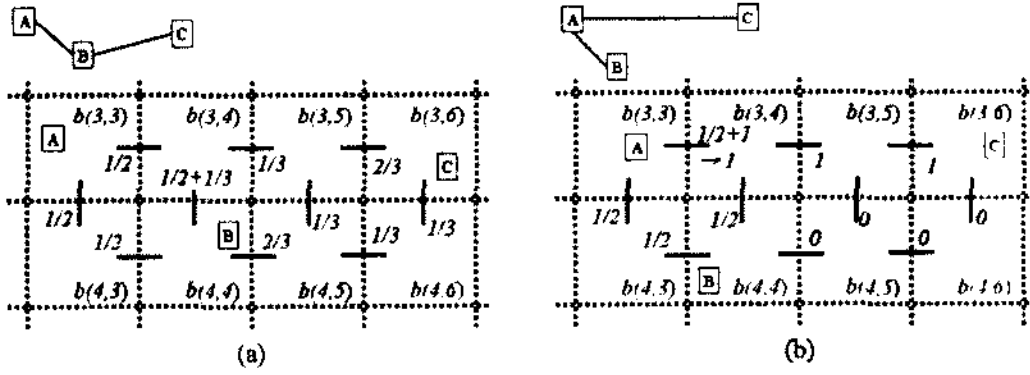


그림 3 세 개의 셀로 구성된 넷에서 MST를 구하고, 배선요구량을 계산한 예

파생되는 배선요구량의 합을 나타낸다. 실제 배선에서 한 넷에 의해 파생되는 배선요구량은 1을 넘지 않으므로 1을 초과하는 배선요구량은 그 값을 1로 보정한다.

### 3. 제안 기법

본 논문에서는 배선밀집도에 영향을 미치는 셀을 모두 이동 가능한 셀로 선택한다. 이동 가능한 셀은 Kraftwerk[11]에서 제안한 FD기법을 이용하여 셀의 최적 위치가 결정되고, 이동 여부는 배선 밀집도 개선의 이득에 따라 결정된다. 이동하게 될 때에는 그 셀에 연결된 다른 넷들을 고려하여 다른 넷에 있는 셀들을 이동하는 셀과 동일한 방향으로 같이 이동하는 것이 도움이 되는지를 판단하여, 도움이 되는 경우 같이 이동한다.

그림 4는 제안된 기법의 예를 나타낸다. 넷  $e_1 = \{A, B, C\}$ ,  $e_2 = \{A, D, E\}$ 이 있다. 셀 A가 속한 빈이 수평 성분의 배선으로 인해 밀집도가 한계값을 초과하였다고 가정하자. 이러면,  $e_1$ 에 속한 셀 A, B, C가 이동 가능한 셀이 되고, FD 기법을 이용하여 계산된 A의 목적 빈이 T1이라고 하자. A를 T1으로 이동할 때, 이 때 다른 넷인  $e_2$ 에 속한 셀 중에 같은 행에 있는 셀 D를 같은 방향인 T2빈으로 이동시키는 것이 전체 배선 밀집도에 이득이 되는지를 검사하여, 이득이 된다면 D를 T2 빈으로 이동한다.

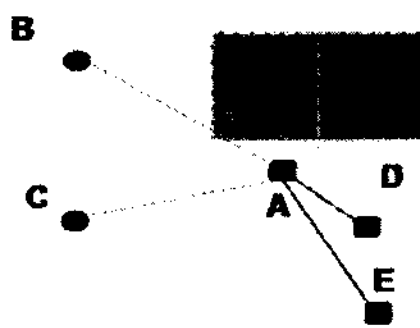


그림 4 ICDGP에서의 셀 이동

#### 3.1 Force-directed 기법

Kraftwerk는 셀을 정점(vertex)으로, 넷은 간선(edge)들의 집합으로 표현하여 회로를 그래프로 모델링 한다.

이 때 비용(cost)은 넷에 관계된 모든 간선들의 비용 합으로 표현되고, 한 간선의 비용은 그 간선에 있는 두 정점 사이 거리제곱으로 모델링되며, 목적함수는 모든 넷의 비용 합으로 표시된다. 고정된 패드(pad)를 제외한 셀 즉, 움직일 수 있는 셀이  $n$ 개 일 때, 셀  $i(1 \leq i \leq n)$ 의 좌표는 셀의 중심의 좌표를 의미하며,  $(x_i, y_i)$ 로 나타낸다. 두 셀  $i$ 와  $j$ 사이에서 있는 간선의 x-성분에 대한 비용은  $(x_i - x_j)^2 = x_i^2 + 2x_i x_j + x_j^2$  이고, y-성분에 대한 비용도 유사하게 표현된다. 목적함수는 [11]에서처럼 다음과 같이 식 (8)로 나타낼 수 있다.

$$\Phi = \sum_{i,j} a_{ij} (x_i - x_j)^2 + \sum_{i,j} a_{ij} (y_i - y_j)^2 \quad (8)$$

여기서  $a_{ij} \in \{0,1\}$ 로서 셀  $i$ 와 셀  $j$ 사이에서 간선이 존재하면 1이고 없으면 0이 된다. 식 (8)은 행렬과 벡터의 곱으로 표현된 선형방정식으로 나타낼 수 있고, 이는 식 (9)에서 보인다.

$$\Phi = \frac{1}{2} \vec{p}^T C \vec{p} + \vec{d}^T \vec{p} + const \quad (9)$$

이때,  $C$ 는  $2n \times 2n$ 인 Laplacian 행렬이고,  $\vec{p} = (x_1, \dots, x_n, y_1, \dots, y_n)^T$ ,  $\vec{d} = (d_1, \dots, d_{2n})^T$ 이다. 벡터  $\vec{d}$ 의 각 원소  $d_i$ 는  $1 \leq i \leq n$ 인 경우 셀  $i$ 에 직접 또는 간접적으로 연결된 패드의 x-좌표에 의해 결정된 상수이고,  $n+1 \leq i \leq 2n$ 인 경우 패드의 y-좌표에 의해 결정된 상수이다.  $const$ 는 패드의 위치로부터 결정된 상수이다.

목적함수  $\Phi$ 를 최소로 만들기 위해  $\nabla \Phi = 0$ 를 풀고 이는 곧 식 (10)으로 표현된다.

$$C \vec{p} + \vec{d} = 0 \quad (10)$$

식 (10)을 풀어 벡터  $\vec{p}$ 의 각 원소의 값을 결정한다는 것은  $\Phi$ 값을 최소로 하는 셀의 위치를 구하는 것을 의미한다. 식 (10)을 이용했을 때 발생하는 중첩의 문제를 해결하기 위해 [11]에서는 중첩의 정도 즉, 배치 영역에서 셀 밀집도 분포로부터 유도된 벡터  $\vec{e}$ 를 추가한 식 (11)를 이용한다.

$$C \vec{p} + \vec{d} + \vec{e} = 0 \quad (11)$$

벡터  $\vec{e}$ 를 분산력 벡터라 하며, 이는 높은 셀 밀집도를 가진 영역에서 낮은 셀 밀집도를 가진 영역으로 셀을 옮기기 위해 사용한다.

분산력 벡터  $\vec{e}$ 는 다음과 같다.

$$\vec{f}(x, y) = \frac{k}{2\pi} \cdot \frac{H}{n} \cdot \frac{W}{m} \cdot \sum_{y'=0}^{m-1} \sum_{x'=0}^{n-1} D(x', y') \frac{\vec{r} - \vec{r}'}{|\vec{r} - \vec{r}'|^2} \quad (12)$$

이때,  $n \times m$ 은 배치영역의 격자 수이고,  $W$ 와  $H$ 는 배치영역의 너비와 높이이다. 이때  $D(x', y')$ 은 셀 밀집도와 배선 밀집도를 같이 고려한 값이다.

#### 3.2 비용 평가함수

셀의 이동에 대한 배선밀집도 이득을 다음과 같은 식으로 평가한다. 여기서  $cost_{con}$  는 배선 밀집도만을 고려한 비용을 나타낸다.  $length$  는 배선길이를 나타내고,  $w$  는 외부변수이다.

$$\delta = rc(i,j) - th \quad (th: \text{임계값}) \quad (13)$$

$$cost_{con} = (1 + \delta)^a \quad (a > 1, \delta > 0 \text{ 인 경우}) \quad (14)$$

$$cost = w \times cost_{con} + (1 - w) \times length \quad (15)$$

식 (15)는 배선밀집도와 배선길이를 모두 고려한 평가함수이다.

### 3.3 개선된 기법

제안한 기법 ICDGP는 다음과 같이 기술할 수 있다.

단계 5에서 배선 밀집도를 평가하고, 밀집도에 따른 배선 요구량 리스트를 구한 뒤 내림차순으로 정렬한다. 배선밀집도가 높은 빈들을 순차적으로 찾아 빈에 속한 넷  $e$ 에 대해  $eval\_FD(e)$ 를 수행한다.  $eval\_FD(e)$ 은 3.1, 3.2에서 설명한 계산식을 이용하여 밀집도가 낮은 목적지 빈을 동시에 찾아준다. 목적지 빈으로 셀들을 이동 시켜 새로운 배치  $P'$ 를 구한다.

```

procedure ICDGP (Improved Congestion
Driven Global Placer)
Input:
P: a detailed placement
k: // max num of iteration
w:// weight for congestion gain
th:// threshold value of congestion
Output: new placement P
1. counter = k
2. S =  $\Phi$  // S: a set of congested bins
//Consider input placement as the best
4. while (counter > 0) {
5. EvaluateCongestion(P, S, th)
//S contains a set of congested
bins
6. Sort S in non-increasing order
according to congestion value
7. for each bin  $b \in S$  {
// Assume  $b(i,j)$  is horizontally
// congested
8. for every  $e \in b(i,j)$ {
9. eval_FD(e)
//eval_FD(e) determines the
//target bin of each cell  $e$ 
//simultaneously
10. Determine cells to move
and direction for each
11. Move cells to target bins
} // for every  $e \in b(i,j)$ 
12. } // for each bin  $b \in S$ 
13. for each bin  $b(i,j)$  {
14. Move cells in  $b(i,j)$  within
the bin to resolve overlap }
15.  $P' \leftarrow$  New placement after
moving cells
// Now compare the new placement
// with so-far best placement
16. if (IsImproved(P, P', th, w) > 0) {
17.  $P = P'$ 
18. counter = k
19. } else {
20. counter = counter - 1; }
21. } // while (counter > 0)
22. return P
    
```

### 4. 실험 및 고찰

ICDGP는 C언어로 구현되었으며 펜티엄4 2G/리눅스 상에서 실험을 수행하였다. 실험은 A사에서 개발한 5개의 회로에 대하여 수행하였으며, 표 1은 회로의 사양이다. 사용된 회로는 최근에 상용화된 제품에 사용되고 있는 것들이다.

표 1 회로 정보

회로 이름	셀의 개수	넷의 개수
ckt1	5,592	6,016
ckt2	10,117	12,015
ckt3	22,406	23,557
ckt4	23,471	21,346
ckt5	24,373	20,392

표 2는 [1]에서 제안한 CDP와 본 논문에서 제안한 ICDGP를 각 회로에 대해 비교 실험한 결과이다.  $th$ 는 임계값으로써 예를 들어  $th=0.7$ 인 경우, 특정 빈의 밀집도가 임계값  $th=0.7$  이상일 때 그 빈의 밀집도를 낮추기 위해 셀을 이동시킴을 의미한다.

빈의 개수는 배선밀집도가 특정 값 사이에 있는 빈의 개수가 몇 개인지 보여준다.

표 2의 각 회로에 대하여 CDP에서 해결하지 못한 높은 배선밀집도의 빈을 ICDGP에서 해결하고 있음을 보여준다. 예를 들어 ckt1의 경우, CDP에서  $th=0.9$ 인 빈이 2개였으나 ICDGP에서  $th=0.9$ 인 빈을 모두 제거하였고  $th=0.8$ 인 빈은 CDP가 13인데 비해 ICDGP는 3개로 85% 개선율을 보였다.

수행속도 측면에서는 모든 회로에 대하여 기존의 CDP보다 ICDGP가 평균 36%, 최고 48% 빠른 시간에 배선밀집문제를 해결함을 알 수 있다.

### 5. 결론

본 논문에서 제시한 기법은 배선밀집도를 매우 효과적으로 개선하며, 또한 기존의 방법에 비해 실행시간이 상당히 개선되는 효과가 있다. 본 논문에서는 배선밀집도 추정을 위해 [1]에서처럼 RMST 모델을 사용하여 넷을 연결하였으나, 넷 연결을 위해 사용된 모델과 상관없이 배선밀집도를 개선할 수 있다는 장점이 있다. 또한 배선요구량 추정을 위해 다른 방법을 사용할지라도 본 논문에서 제시한 알고리즘의 기본 틀은 그대로 유지하면서 배선밀집도를 개선할 수 있다는 장점이 있다. 즉, 넷 연결을 위한 모델이나, 배선밀집도 추정을 위한 방법과 상관없이 제시한 기법으로 배선밀집도를 개선할 수 있는 장점이 있다.

비록 연구결과는 [1]의 결과와만 비교하였지만 앞에서

표 2 실험 결과

회로 이름	적용 기법	빈(bin)의 개수							배선길이	CPU 시간	CPU 시간 개선율(%)
		th=0.7	th=0.8	th=0.9	th=1.0	th=1.1	th=1.2	th=1.3			
ckt1	실험전	20	20	15	0	1	0	0	12,978,178		
	CDP	55	13	2	0	0	0	0	14,000,749	8m 44s	
	ICDGP	59	2	0	0	0	0	0	14,716,639	4m 44s	46%
ckt2	실험전	451	342	288	213	106	88	93	40,344,321		
	CDP	743	476	256	105	34	8	12	39,913,256	1h21m20s	
	ICDGP	718	501	232	89	24	10	6	39,960,957	49m54s	39%
ckt3	실험전	410	309	243	163	106	46	35	118,710,121		
	CDP	708	399	141	21	0	0	1	115,621,690	3h59m32s	
	ICDGP	719	393	121	8	1	0	1	116,052,538	2h20m28s	41%
ckt4	실험전	261	166	100	39	22	10	14	66,760,034		
	CDP	499	98	17	1	0	1	4	66,111,991	27m 45s	
	ICDGP	494	85	11	2	2	0	4	66,263,063	14m 27s	48%
ckt5	실험전	180	72	33	17	7	1	5	67,599,010		
	CDP	570	15	6	1	1	0	3	67,301,895	5m 13s	
	ICDGP	238	9	4	2	0	1	3	67,410,449	4m 49s	8%

제시한 장점으로 인해 제시한 기법의 성능은 객관적으로 우수하다고 말할 수 있다.

또한 제시한 기법은 배선 밀집도 뿐만 아니라 배선길이를 같이 고려할 수 있도록 설계되어 잠재적으로 타이밍 문제까지 고려 대상에 포함시킬 수 있는 장점이 있다.

참 고 문 헌

[1] 오은경, 허성우, "배선밀집도 드리븐 배치", 한국정보처리학회 논문지, 제13-A권, 제1호, pp. 57-70, 2006.

[2] A. E. Caldwell, A. B. Kahng, and I.L. Markov, "Can Recursive Bisection Alone Produce Routable Placements?," Proc. of DAC, pp. 477-482, 2000.

[3] M.C.Yildiz and P.H.Madden "Improved Cut Sequences for Partitioning Based Placement," Proc. of DAC, pp. 776-729, 2001.

[4] A. Agnihotri, et al. "Fractional Cut: Improved Recursive Bisection Placement," Proc. of ICCAD, pp. 307-310, 2003.

[5] X. Yang, M. Wang, K. Egur and M. Sarrafzadeh, "A Snap-on Placement Tool," Proc. of ISPD, pp. 153-158, 2000.

[6] A.B.Kahng and S. Reda, "Wirelength Minimization for Min-Cut Placements via Placement Feedback," IEEE Trans. on CAD of Integrated Circuits and Systems, Vol.25, No.7, pp. 1301-1312, 2006.

[7] Wern-Jieh and Carl Sechen, "Efficient and Effective Placement for Very Large Circuits," IEEE Trans. on CAD of Integrated Circuits and Systems, pp. 349-359, 1995.

[8] G. Meixner and U. Lauther, "Congestion-Driven Placement Using a New Multi-Partitioning Heuristic," Proc. of ICCAD, pp. 332-335, 1990.

[9] X. Yang, B-K. Choi, and M. Sarrafzadeh, "Routability Driven White Space Allocation for Fixed-

die Standard Cell Placement," Proc. of ISPD, pp. 42-50, 2002.

[10] S. Hur and J. Lillis, "Mongrel: Hybrid Techniques for Standard Cell Placement," Proc. of ICCAD, pp. 165-170, 2000.

[11] H. Eisenmann and F. M. Johannes, "Generic Global Placement and Floorplanning," Proc. of DAC, pp. 269-274, 1998.