

논문 2005-42SD-12-9

계층적 분할 기법과 완화된 국부 탐색 알고리즘을 이용한 효율적인 광역 배치

(Efficient Global Placement Using Hierarchical Partitioning Technique
and Relaxation Based Local Search)

성 영 태*, 허 성 우**

(Young-Tae Sung and Sung-Woo Hur)

요 약

본 논문에서는 “middle-down” 접근법에 기반한 기존의 표준 셀 배치기인 하이브리드 배치기^[25]의 단점을 보완한 효율적인 광역배치 알고리즘을 제안한다. hMETIS(클러스터링을 이용한 다단계 하이퍼그래프 분할기법)에 사용된 기법과 RBLs(Relaxation Based Local Search) 기법의 적절한 조합을 통해 기존 하이브리드 배치기의 광역배치 기능을 향상시킨다. hMETIS를 통한 분할기법을 “top-down” 방식으로 적용하고, 각 단계에서 RBLs를 사용하여 광역배치를 점진적으로 개선해 나가는 제안된 기법은 초기 배치에 크게 영향을 받는 기존 방법의 문제점을 해결하고, 실행 속도를 개선하면서도 배치의 질을 떨어뜨리지 않는 효과적인 기법이다. 제안한 알고리즘을 통해 구현된 개선된 배치기는 기존의 하이브리드 배치기나 FengShui와 같은 우수한 툴과 비교할 때 뒤지지 않는 성능을 보인다. 특별히 기존의 하이브리드 배치기에 비해 실행 속도 면에서 평균 5배 정도의 개선을 보였고, 큰 회로에 대해선 배선길어도 줄어드는 향상된 결과를 보였다.

Abstract

In this paper, we propose an efficient global placement algorithm which is an enhanced version of Hybrid Placer^[25], a standard cell placement tool, which uses a middle-down approach. Combining techniques used in the well-known partitioner hMETIS and the RBLs(Relaxation Based Local Search) in Hybrid Placer improves the quality of global placements. Partitioning techniques of hMETIS is applied in a top-down manner and RBLs is used in each level of the top-down hierarchy to improve the global placement. The proposed new approach resolves the problem that Hybrid Placer seriously depends on initial placements and it speeds up without deteriorating the placement quality. Experimental results prove that solutions generated by the proposed method on the MCNC benchmarks are comparable to those by FengShui which is a well known placement tool. Compared to the results of the original Hybrid Placer, new method is 5 times faster on average and shows improvement on bigger circuits.

Keywords: 표준 셀 배치, 광역 배치, hMETIS, RBLs

I. 서 론

표준 셀 배치는 VLSI(Very Large Scale Integration)

* 학생회원, ** 정회원, 동아대학교 컴퓨터공학과
(Dept. of Computer Engineering, Dong-A
University)

※ 본 논문은 '2001학년도 동아대학교에서 지원한 국외
파견 연구 지원사업'에 의해 연구되었음.

접수일자: 2005년6월8일, 수정완료일: 2005년11월25일

CAD(Computer Aided Design) 물리적 설계 단계에서 매우 중요한 과정으로 칩 면적, 라우팅 가능성, 칩 성능, 칩 생산수율(yield)에 큰 영향을 미친다. 또한 우수한 CAD 툴의 개발은 VLSI 칩 설계의 전체 공정을 자동화함으로써 칩 생산의 효율성을 가져온다. 특히 칩 성능을 결정짓는 타이밍 문제를 해결하는데 있어서 배치 과정은 매우 중요한 단계이다^[1].

배치 알고리즘의 목적은 크게 전체 칩 면적을 최소화

하는 것과 총 배선길이를 최소화하는 것이다. 칩 면적의 최소화는 주어진 칩 공간에 더 많은 모듈을 집적할 수 있는 여건을 제공하며 배선길이의 최소화는 신호지연을 유발하는 최장 경로를 가진 넷을 줄임으로써 칩의 성능을 높일 수 있기 때문이다. 이 두 가지 목적은 밀접한 관계를 가지고 있는데 이것은 전체 칩의 크기가 각 모듈의 크기와 배선이 차지하는 공간의 합으로 볼 수 있기 때문이다. 따라서 배선길이를 줄이는 것은 묵시적으로 칩 면적을 최소화시키는 효과가 있다.

셀 배치 문제는 *NP-hard* 문제로 알려져 있으며 이것을 해결하기 위해 수년간 다양한 휴리스틱이 발표되었다. 셀 배치의 공통적인 목표는 셀들이 중첩되지 않도록 하면서 배선길이를 최소화하는 것이다. 배치를 위한 최근의 기술은 크게 세 가지로 분류될 수 있다. 첫 번째 부류는 처음부터 중첩을 허용하지 않으면서 배치 과정을 처리하는 것이다. TimberWolf^[2]는 *annealing*을 기반으로 하는 잘 알려진 배치 툴이다. 이는 주어진 배치에서 셀들의 위치를 변형시킴으로써 새로운 배치를 얻어 가는데 이 기법을 사용하여 배치를 구한 결과들이 많이 발표되었다^[3-5]. 두 번째 부류는 계층구조에서 분할기법을 적용함으로써 광역배치(global placement)로부터 상세배치(detailed placement)를 점진적으로 구해가는 방법을 사용한다. 최소 컷을 재귀적으로 얻음으로써 가능한데, 이 기법을 사용하는 툴로는 Capo^[6], Quad^[7], FengShui^[8], Snap-on^[5] 등^[9,10]이 있다. Caldwell^[11]이 지적한대로 이 기법을 사용하는 툴들은 대부분 변형된 KL알고리즘^[12]이나 FM알고리즘^[13]을 이용하고 있다. 세 번째 부류는 해석적 기법 또는 FD(Force-Directed) 기법을 사용하는 것이다^[14-16]. FD 방법은 그 변화가 매우 다양하다. 셀의 위치를 구하기 위한 식을 어떻게 푸느냐에 따라 중복의 정도가 달라지고 결국 해의 질이 달라진다. 또한 해를 구하는 시간도 다르다. FD 방법은 근본적으로 중복을 허용하기 때문에 중복을 어떻게 해결하는가에 따라 해의 질이 크게 좌우된다. Goto는 iterative FD 방법을 이용한 독특한 알고리즘을 일찍이 제안하였다^[17]. Parakh^[18]와 Yang^[19]은 라우팅을 고려한 FD 배치 알고리즘을 제시하였으며, 셀 중복을 가능한 줄이기 위해 인장력뿐만 아니라 척력(repelling force)을 이용한 기법도 소개되었다^[14,15,20]. Snap-on^[5], Dragon^[21,22]은 광역배치를 최적화시키기 위해 분할기법을, 상세배치 최적화를 위해 *annealing* 기법을 사용하였다. 그 외에도 유전 알고리즘을 이용한 방법^[23]도 있다.

이들 알고리즘은 수행 형태에 따라 구성적 방법

(constructive method)과 반복 개선 방법(iterative improvement method)으로 분류된다. 구성적 방법을 위한 알고리즘은 주로 연결도(connectivity) 정보만을 이용하여 셀의 위치를 결정한다. 이것은 수행 속도 면에서 탁월한 성능을 보였으나 좋은 결과를 생성하지는 못했다. 최근의 배치기법들은 빠른 수행 속도의 이점 때문에 구성적 방법을 이용하여 초기 배치를 얻고 이어서 반복 개선법을 이용하여 이를 개선시켜 나간다. 반복 개선법을 이용하면 일반적으로 좋은 배치를 얻을 수 있지만 전체적인 수행 시간이 많이 드는 단점이 있다.

셀 배치 과정은 다양한 제약 조건을 만족하도록 수행되어야 하며 동시에 목적 함수를 최적화 해야 하는 복잡한 과정이다. 이러한 문제를 효과적으로 제어하기 위해 배치 과정은 다단계(multi-level)^[24] 혹은 광역배치(global placement)와 상세배치(detailed placement)의 2 단계 과정을 통해 수행된다.

다양한 셀 배치 툴 중 주목할 만한 몇 가지는 KraftWerk^[14], Dragon^[21,22], Capo^[6], FengShui^[8] 그리고 하이브리드기법^[25]이 있다. KraftWerk는 FD 기법을 사용하는 배치기로서 광역배치 단계에서 셀 겹침을 허용하며 적법화(legalization) 과정을 통해 셀 겹침을 해결하는 해석적 알고리즘이다. 재귀적 최소컷 분할기법(recursive min-cut partitioning technique)을 사용하는 Dragon은 “*simulated annealing*” 기법을 통해 광역 배치를 얻는다. 확장된 Dragon^[22]은 라우팅을 고려한 배치 기능을 지원했으며 최근에는 타이밍을 고려할 수 있는 기능까지 추가되었다. Dragon과 마찬가지로 광역 배치 단계에서 재귀적 최소컷 분할기법을 사용하는 FengShui 툴은 다단계 분할기법을 통해 배선길이를 개선한다. Dragon과 FengShui 툴은 모두 광역배치의 분할과정에 hMETIS^[26]를 사용한다는 공통점이 있다. Capo 툴은 전통적인 FM 분할기법을 사용하여 재귀적으로 2분할을 수행하는 구성적 배치 툴이다. Capo 툴은 라우팅 공간의 확보에 좋은 결과를 보여준다.

하이브리드 기법을 사용한 하이브리드 배치기^[25]는 “*middle-down*” 접근법을 사용하는 배치기로서 광역 배치 단계에서 반복 개선을 기반으로 하는 전통적인 국부탐색 알고리즘 RBLs를 사용한다.

본 논문에서는 “*top-down*” 접근법에 기반한 RBLs 알고리즘을 제안한다. 또한 기존 하이브리드 배치기의 수행 성능에 관한 문제점을 고찰하고 배치의 성능을 효율적으로 향상시키는 새로운 광역배치 프레임워크를 제안한다.

II. 본 론

1. 관련 연구

하이브리드 배치기는 표준 셀 배치를 위해 다양한 기법을 복합적으로 사용하는 하이브리드형 배치기로서 광역배치와 상세배치의 2단계 과정을 거친다. 광역배치 단계에서 RBLS(Relaxation Based Local Search)를 통해 각 셀의 최적 빈(bin) 위치를 찾고, 상세배치 단계에서 최적 인터리빙(Optimal Interleaving) 기법을 사용하여 셀 겹침을 제거한 최종 배치를 얻는다.

“Middle-down” 접근법으로 기술되는 하이브리드 배치기의 광역배치 기법은 임의의 격자 상에서 RBLS를 적용시킬 수 있는 독특한 배치기법이다. 또한 각 단계에서 적용되는 최적화 기법은 단계에 수반되는 제약조건과 배치목적에 효율적으로 반영하도록 설계되었다.

가. RBLS(Relaxation Based Local Search)

RBLS는 기본적으로 전형적인 국부탐색 기법과 동일한 방법을 이용하되 독특한 해석적 기법을 사용하여 탐색공간을 이동한다. 전형적인 국부탐색 기법은 그림 1 과 같은 구조를 가진다.

하이브리드 배치기에선 이웃 해를 어떻게 효율적으로 찾을 것인가에 초점을 맞추었다. “Simulated annealing” 기법에서 주로 사용한 방법으로 이웃 해를 찾았다면 너무 국부적인 탐색이 되어 거시적인 관점을 놓칠 가능성이 높아 이런 문제를 해결하면서 효율적으로 이웃 해를 찾을 수 있도록 RBLS 기법을 제시하였다. RBLS 기법에서 이웃 해를 찾기 위한 방법은 다음

```

Input : given solution
         ε // threshold value to determine improvement
         k // max number of trials to find a better solution
Output : improved new solution

S ← input solution
Qs ← Quality(S) // Evaluate solution quality
count ← k
while (count > 0) {
    Snew ← A neighbor solution of S
    Qnew ← Quality(Snew) // Evaluate solution quality
    if ( (Qnew - Qs) > ε ) then // new solution is improved
        S ← Snew
        count ← k
    else
        count ← count - 1
}
return S
    
```

그림 1. 전형적인 국부탐색 프레임워크
Fig. 1. Typical local search framework.

과 같이 크게 네 단계로 나누어 실행된다.

(1) 부분회로 추출 (Subcircuit extraction): 주어진 인자 m 에 대하여 이동 가능한 모바일 셀(mobile cell)의 집합 $M(m = |M|)$ 을 추출한다. 추출된 모바일 셀로부터 고정 셀의 집합 F 를 결정한다.

(2) 빈 용량의 조건을 무시한 최적의 위치 (Optimal relaxed placement): 모바일 셀의 최적 위치를 찾는 과정으로 이때 빈(bin) 용량에 대한 제약조건은 무시한 상태에서 최적의 위치를 찾는다.

(3) 배치 적법화 (Placement legalization): 조건이 완화된 배치로부터 조건을 만족시키는 적법한 배치를 얻기 위해 셀들을 이동한다. 이 과정에서 이동되는 셀의 선택 시 목적함수를 최적화 시킬 수 있도록 선택한다. 셀들의 이동을 통해 제약조건이 만족된 새로운 배치를 얻는다.

(4) 배치 최적화 (Further Optimization): 인접한 빈에 대한 분할 기법인 FM 기법을 사용해 추가적인 최적화를 수행한다.

이웃 해를 찾는 과정을 포함하여 RBLS 알고리즘이 어떻게 동작하는지를 그림 2에서 보인다.

부분회로의 추출 전략과 모바일 셀의 개수는 RBLS의 수행 성능과 광역배치의 질에 큰 영향을 미친다. 하이브리드 배치기에선 넷 정보에 기반하여 “wavefront” 알고리즘을 통해 효과적으로 부분회로를 추출한다. 부분회로의 추출 매커니즘은 광역배치 단계에서 거시적인 최적화를 가능하게 하는 이유가 된다. 제약조건이 완화

```

Input : m, k and current placement P
Output : new placement P

counter ← k
while (counter > 0) {
    Extract a mobile node set M (|M|=m)
    Determine a fixed node set F using M
    For  $\forall v \in M$ , determine optimal relaxed location
    P' ← new placement after legalization
    P' ← optimize P' using FM algorithm
    if (WL(P') < WL(P)) then
        P ← P'
        counter ← k
    else
        counter ← counter - 1
}
return P
    
```

그림 2. RBLS(Relaxation Based Local Search) 알고리즘
Fig. 2. RBLS(Relaxation Based Local Search) algorithm.

된 최적배치를 찾는 과정은 국부탐색 구조에서 우수한 이웃 해(neighborhood solution)를 구하기 위한 과정으로 단순한 셀 교환 방식을 가지지 않는다. 이것은 완화된 최적 배치를 위해 각각의 하이퍼에지를 선형식으로 모델링하고 이것을 해결하는 효율적인 네트워크 플로우(network flow) 기법을 통해 최적의 완화된 셀 위치가 결정된다. 이 과정에서는 셀 겹침 현상을 허용한다. 적법화 과정은 과밀된 빈(excess bin)과 결손된 빈(deficit bin) 사이의 조율과정으로 빈 제약조건이 만족할 수 있도록 한다. 적법화는 “ripple-movement” 알고리즘을 통해 수행된다. 마지막 최적화 과정은 현재 배치를 초기 분할로 보고 각각의 인접한 빈 쌍이 최소 컷(cut)을 가지도록 FM(Fidduccia-Mattheyesis)^[13] 분할 기법을 적용한다.

하이브리드 배치기에서의 RBLs에서는 주어진 입력 배치와 현재의 입력배치에 대해 이웃 해를 어떻게 구할 것인가에 중점을 두었으며, 시작배치 혹은 초기배치의 중요성은 크게 다루지 않았다.

나. Middle-Down 접근법

RBLs를 통해 좋은 광역 배치를 얻는 하이브리드 배치는 임의의 $n \times m$ 격자 상에서 RBLs를 적용할 수 있는 “middle-down” 접근법을 가진다. 즉, $n \times m$ 격자 상에서 단순한 방법으로 생성한 초기배치를 시작으로 RBLs를 적용시키는 구조이다. 초기 배치 생성하기 위해 선형 순서화 함수^[27]을 이용한 방법 또는 분할 기법을 이용하여 셀들을 임의로 배치하는 방법이 사용되었는데 좋은 광역배치를 얻기 위한 최선의 방법이 아님을 우리는 실험적 경험을 통해 알았다.

다. hMETIS : 클러스터링을 이용한 다단계 하이퍼그래프 분할 기법

Dragon, FengShui와 같은 우수한 배치 툴은 분할 기법을 이용해 배치 선처리 과정을 거친다. 분할기법은 배치 문제의 복잡도를 줄이고 성능을 향상시키기 위한 일반적인 기법으로서, 좋은 분할기법을 사용한다면 좋은 배치를 얻을 수 있음을 말해 준다. 수년간 연구되어

<p>hMETIS : Multilevel Hypergraph Partitioning</p> <p>(1) 압축 그래프 생성(Coarsening phase)</p> <p>(2) 초기 분할(Initial partitioning)</p> <p>(3) 압축 해제와 정제(Uncoarsening and refinement)</p>

그림 3. 다단계 하이퍼 그래프 분할 프레임워크
Fig. 3. Multilevel hypergraph partitioning framework.

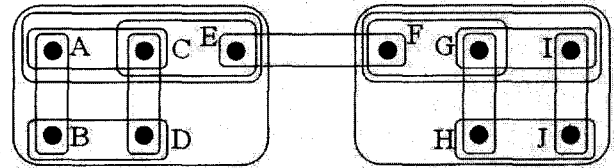


그림 4. 초기 하이퍼 그래프
Fig. 4. Initial hypergraph.

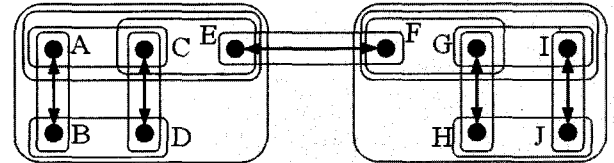


그림 5. 압축될 에지의 그룹
Fig. 5. Edge groups to be coarsened.

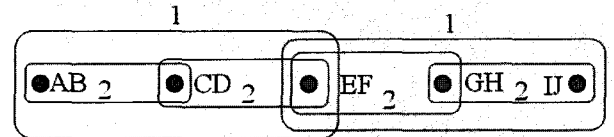


그림 6. 압축된 하이퍼 그래프
Fig. 6. Coarsened hypergraph.

온 분할기법 중 hMETIS는 가장 우수한 성능을 보여주는 다중 패러다임(multi-paradigm) 다단계 분할기법으로서 KL-FM류의 분할기법과 더불어 광범위하게 사용되고 있다. hMETIS를 이용한 하이퍼 그래프 분할 과정은 크게 세 단계로 이루어진다.

압축 그래프의 생성 과정은 초기 하이퍼그래프로 부터 연속적으로 압축된 그래프를 얻는 과정이다. 이것은 압축 그래프로부터 얻은 분할의 질(quality)이 처음 그래프로부터 얻은 분할의 질(quality)보다 우수하다는 점과 그래프의 크기를 줄임으로써 수행 속도의 향상을 얻을 수 있는 이점 때문이다. 그래프의 압축 매커니즘은 단순한 에지압축(edge coarsening)과 하이퍼에지 압축(hyperedge coarsening)의 선택에 따라 달라진다. 그림 4, 5, 6에서 간단한 에지압축의 예를 보였다.

그림 4의 A, B, C, D, E, F, G, H, I, J는 셀을 표현하며 사각형 박스는 하이퍼 넷을 각각 표현한다. 그림 4에 의하면 {A, B}, {C, D}, {E, F}, {G, H}, {I, J} 등이 하이퍼 넷을 이룬다.

에지압축 매커니즘은 그룹핑 될 최대 셀 쌍을 결정하고 각 쌍을 하나의 슈퍼 셀로 묶는다. 그림 5에서는 {A, B}, {C, D}, {E, F}, {G, H}, {I, J}가 압축될 셀 쌍으로 결정된 모습이다.

그림 6은 슈퍼 셀을 가지는 압축된 하이퍼그래프를 나타내며 박스의 각 숫자는 기존 하이퍼그래프의 하이

퍼넷 수를 말한다. 즉, 두 슈퍼 셀 AB와 CD를 가지는 하이퍼 넷은 압축을 해제할 경우 두 개의 하이퍼 넷으로 구성됨을 알 수 있다.

가장 작은 압축 그래프가 생성된 후 초기 분할이 수행된다. 초기 분할은 압축된 그래프의 크기가 매우 작기 때문에 수행에 드는 비용은 크지 않으며 크게 두 가지 방법으로 수행한다. 하나는 분할의 크기가 거의 같아지도록 랜덤하게 노드를 나누는 것이며 또 다른 하나는 처음 노드를 랜덤하게 선택하고 BFS 방식으로 분할의 한 영역의 크기를 조금씩 키워가는 방법이다. 두 가지 방법 모두 랜덤 알고리즘(randomized algorithm)에 기반하기 때문에 최종 분할의 질을 보장하기 위해서는 몇 개의 후보 초기 분할을 생성한다. 마지막 단계인 압축 해제와 정제과정은 압축된 그래프로부터 압축을 원래의 그래프로 복원하는 동시에 해제된 압축 그래프의 컷을 다시 줄이는 과정이다. 정제과정에서 사용되는 기법은 효율적인 이득버킷 자료구조에 기반을 둔 $O(E^2)$ 의 복잡도를 가지는 FM 분할 알고리즘이 사용된다.

hMETIS는 FM 알고리즘의 최적화 패스를 최초의 2 단계로 제한하는데(early-exit FM) 이것은 최적의 분할이 처음 혹은 두 번째 패스에 모두 결정되는 FM 알고리즘의 특징 때문이다. 하지만 최대 이득을 가지는 셀을 우선적으로 이동시키는 FM 알고리즘의 성질로 인해 개선의 여지가 남아있는 하이퍼 넷 조차 FM 알고리즘만으로는 개선시키지 못하는 경우가 있다. hMETIS는 HER(HyperEdge Refinement) 기법으로 이러한 상황을 해결한다. 그림 7에 균형비 40/60을 가지는 간단한 분할에서 HER 기법으로 넷 컷을 개선할 수 있는 예를 보였다.

그림 7에서 각 셀의 이득값은 FM 알고리즘에 의해

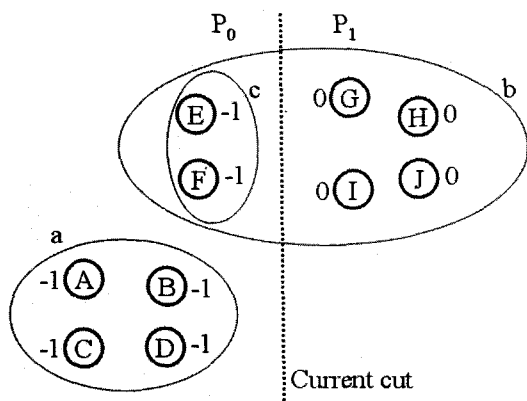


그림 7. 분할된 하이퍼 그래프와 각 셀의 이득 값
Fig. 7. Partitioned hypergraph and the gain of each vertex.

계산한 것으로써 0 혹은 -1이다. 최대 이득 셀을 옮기는 FM 알고리즘으로는 컷 크기를 줄일 수 있는 셀을 선택하지 못한다. 반면, 만약 하이퍼 넷 b 의 모든 셀을 분할 P_0 에서 P_1 으로 옮긴다면 균형비 40/60을 위반하지 않으면서 하나의 넷 컷을 줄일 수 있다. HER은 두 분할에 걸친 하이퍼 넷을 중심으로 하이퍼 넷 전체를 이동시키며 넷 컷을 정제하는 기법이다.

2. 제안 기법

하이브리드 배치기에서는 매우 단순한 방법을 사용하여 초기배치를 생성하며, 이 배치는 RBLS의 입력으로 주어지는데 실험에 의하면 최종 배치의 질은 초기 배치의 질에 크게 영향을 받는 것을 알 수 있었다. 이런 영향을 적게 받기 위해선 상당히 많은 CPU 시간을 사용하여 RBLS 알고리즘을 적용하여야 한다. 실험에 의하면 최종 배치까지 소요되는 전체 시간의 약 85%~90% 이상이 광역배치 최적화에 소모되는 것을 알았다. 즉, RBLS 알고리즘 수행시간의 비중이 그만큼 크다는 뜻이다.

이런 문제점을 해결하기 위해 RBLS의 장점인 거시적인 광역배치 관점을 유지하면서 효율적으로 광역배치를 최적화할 수 있는 새로운 매커니즘을 찾게 되었고, 우리는 분할 기법을 통한 “top-down” 접근법을 통해 이 문제를 해결한다.

본 논문에서는 hMETIS에서 제안한 분할 매커니즘을 이용하여 “middle-down” 접근법이 아닌 “top-down” 접근법을 통한 광역배치 알고리즘을 제안한다. 또한 2-pin 넷들을 압축하여 얻은 수정된 회로를 사용한다. 제안하는 기법은 hMETIS와 RBLS의 결합을 통한 수정된 광역배치 알고리즘으로 볼 수 있는데, 이는 하이브리드 배치기의 광역배치 속도를 향상시키는데 결정적인 역할을 한다. 제안한 알고리즘의 프로토타입 툴을 HGP (Hierarchical Global Placer)라 부른다.

가. 사전 지식

주어진 회로는 시그널 넷들의 집합으로 간주하며 넷리스트는 하이퍼그래프(hypergraph) $G = (V, E)$ 로 모델링한다. 집합 V 는 각 셀의 집합이며 E 는 넷의 집합이다. 하이퍼에지(hyperedge) $e \in E$ 는 둘 혹은 그 이상의 셀을 가지는 V 의 부분집합, 즉 $e \subset V$ 이다. $|e|$ 는 넷 e 에 연결된 셀들의 개수를 말하며 $s(v)$ 는 셀 v 의 크기를 말한다.

나. 제약 조건

광역배치 단계에서는 표준 셀이 위치할 코어영역 (core region)을 $n \times m$ 격자로 분할하며 n 과 m 은 각각 행(row)과 열(column)을 나타낸다. 광역배치 문제는 주어진 R (row), C (column), 그리고 허용 가능한 행의 최소, 최대 크기에 대하여 아래의 두 제약 조건을 지키는 가정 하에 최적의 배선 길이를 가지도록 각각의 셀이 위치할 빈 좌표를 찾는 문제이다.

$$(1 - \epsilon) \cdot \frac{A}{R \times C} \leq \sum_{v \in b_{i,j}} s(v) \leq (1 + \epsilon) \cdot \frac{A}{R \times C} \quad (1)$$

$$lowerbound \leq \sum_{1 \leq j \leq m} \sum_{v \in b_{i,j}} s(v) \leq upperbound \quad (2)$$

식(1)은 각 빈(bin)의 용량에 대한 제약조건이다. 모든 셀의 크기의 합을 A 라 할 때 각각의 빈은 파라미터 ϵ 에 따라 각각의 용량 제한을 가지게 된다. 그리고 식(2)를 통해 각 행의 용량이 제어된다. 이것은 모든 행이 균등하게 셀들을 가지도록 하기 위함이다.

다. HGP (Hierarchical Global Placer)

제안기법은 초기에 2×2 격자를 고려하고, hMETIS의 “2-way” 분할 매커니즘을 이용하여 이 격자 상에서 셀들의 위치를 결정한다. 셀들이 어느 빈에 속하는지 결정되면 같은 빈에 속한 셀들 중 동일한 2-pin 넷에 있는 것끼리 그룹핑하여 압축회로를 만든다. 그 다음 압축된 회로에 RBLS 알고리즘을 적용하여 현재의 배치를 개선한다. 이때 셀들이 서로 다른 빈으로 이동하는데 이 셀들 중 어떤 것들은 슈퍼 셀이고 어떤 것들은 원래의 셀이다. 더 이상 개선되지 않으면 압축을 해제하여 원래의 회로를 복구한다. 이때 각 빈에 속한 셀 (또는 슈퍼 셀)의 위치는 그대로 유지된다.

2×2 격자에서 구한 현재의 배치에서 격자상의 각 빈을 동일한 방법으로 4등분 하며, 각 빈에 속한 셀들도 같이 분할하여 전체적으로 4×4 격자 상에서 새로운 배치를 얻고, 이 격자 상에서 압축과정과 RBLS 알고리즘을 다시 적용한다. 이런 과정은 격자가 $2^k \times 2^k$ 이 될 때까지 반복한다.

우리는 배치에 사용될 행의 수 n 은 설계자에 의해 미리 정의된다고 가정한다. 주어진 n 에 대해 $2^k < \frac{n}{2}$ 을 만족하는 최대의 k 를 구한 후 $m = 2^k$ 로 둔다. 예를 들어 회로의 행수가 100이면 $32 = 2^5 < \frac{100}{2} = 50$ 이 되

어 k 는 5가 되고 따라서 $m = 32$ 가 된다. 이는 코어 영역을 $2^k \times 2^k$ 의 격자로 나눈 경우 한 빈에 2 이상의 행이 포함되는 것을 의미한다. 격자를 2×2 에서 시작하여 $2^k \times 2^k$ 되기까지 분할한 후, 각 빈을 다시 수평 분할하여 최종적으로 $n \times m$ 격자를 얻게 되고, 여기서 다시 RBLS 알고리즘을 적용한다. 이 마지막 과정은 원래의 하이브리드 배치기에서 한 것과 동일하다.

각 빈에 속한 부분 회로를 4 등분하는 과정은 그림 8과 같다. 우선 각 빈의 좌표를 기준으로 수평선을 설정한다. 이 수평선이 행의 경계와 일치하면 그대로 두고, 만약 행의 경계선과 일치하지 않으면 위에 있는 행의 경계선에 맞춘다. 예를 들어 현재의 빈에 있는 행수가 11인 경우 가로분할을 위한 경계선은 다섯째 행과 여섯째 행의 경계선이 된다. 그런 다음 위에 있는 분할에 대해 수직 성분의 직선에 대해 다시 이등분 한다. 그리고 방금 분할될 정보를 이용하여 아래 부분을 이등분함으로써 각 빈에 속한 모든 셀을 사등분하게 된다. 4분할 과정과 RBLS의 적용에 관한 개략적인 과정을 그림 8에 보였다.

그림 8(a)는 표준 셀이 위치할 코어 영역 내에서 입력 그래프를 수평 분할한 경우이다. H-분할, V-분할은

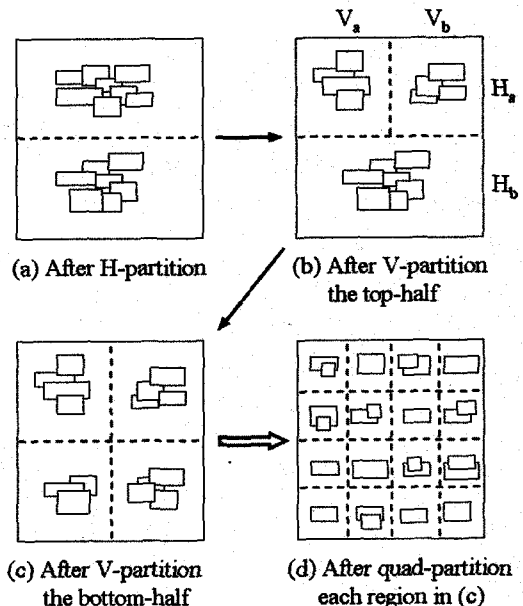


그림 8. 계층적 분할기법의 적용 예. 그림 (c)에서 RBLS 알고리즘을 적용하여 광역 배치를 최적화시킨다. 그림 (c)의 각 빈에 대해 같은 방법으로 4-분할하여 그림 (d)를 얻음

Fig. 8. An example to show the hierarchical partitioning technique. In fig (c), the global placement is optimized using RBLS algorithm. Then each bin in fig (c) is quad-partitioned to get fig (d).

```

Input : n // # rows
         k // parameter for RBLs
Output : new global placement P
Determine m
i ← 0
s ← 0.1 • number of total cells
P ← Place all cells in 2i × 2i grid //initial placement
while(i < k) {
    P ← Partition current P into 2i × 2i grid
    P ← Make a clustered circuit on P'
    P ← Call RBLs(s, k, P)
    P ← Flatten P
    s ← 0.5 • s
    i++
}
P ← Partition each bin in P horizontally to get an n × 2k grid
while(∃ improvement) {
    P ← Call RBLs(s, t, P)
    s ← 0.5 • s
    t ← 2 • t
}
return P
    
```

그림 9. 계층적 분할기법과 RBLs를 이용한 HGP기법
 Fig. 9. HGP Algorithm using the hierarchical partitioning technique and RBLs technique.

모두 hMETIS의 “2-way” 분할 매커니즘을 사용한다. 그림 (b)는 그림 (a)의 상단부에 있는 셀들에 대해 V-분할 수행한 결과를 보여주는데, 이때 기존의 다른 셀들의 위치를 고려하여 컷 사이즈가 최소가 되도록 한다. 그림 (c)에선 그림 (a)의 하단부에 있는 셀들에 대해 V-분할 수행한 결과를 보여주는데, 이때엔 그림 (b)에서 구한 V_a와 V_b에 속한 셀들의 위치도 같이 고려하여 컷 사이즈가 최소가 되도록 한다. 그림 (c)에 보인 것과 같이 hMetis에 의해 4-분할된 셀들에 대해 RBLs를 적용하여 중간 광역배치를 개선한다. 그림 (d)는 그림 (c)로부터 4-분할 과정을 적용하여 얻은 결과를 보여준다.

그림 9에서는 계층적 분할 기법과 RBLs 알고리즘을 이용한 계층적 광역 배치기 HGP의 개요를 보여준다.

라. 배선 길이의 측정

최적화의 목적 함수로는 추정된 배선길이를 사용한다. 일반적으로 배선 길이의 측정은 정확한 라우팅 모델이 수반되어야 한다. 본 논문에서는 정확성과 측정의 용이함 그리고 비교 대상 틀과의 형평성을 고려해 HP(Half-Perimeter) 측정법을 통해 배선 길이를 측정한다. HP측정법은 2차원 평면상에 맨하탄 기하로 구성된 2-pin 또는 3-pin 넷에 대해 정확한 배선길이를 산출하지만 4-pin 이상의 핀 수를 가지 넷의 경우 실제

배선길이보다 일반적으로 작은 값을 산출한다^[28]. 식 (3)에서 x와 y는 셀 u와 v가 위치한 격자의 중심좌표이다.

$$length(e_i) = \max_{u,v \in e_i} |x_u - x_v| + \max_{u,v \in e_i} |y_u - y_v| \tag{3}$$

$$WL(P) = \sum_{i=1}^{|E|} length(e_i) \tag{4}$$

즉, HP 측정법은 넷 e_i의 모든 핀을 둘러싸는 사각박스 둘레의 반을 배선길이라고 한다. 따라서 주어진 배치의 질을 결정하는 배선길이는 식 (4)와 같이 기술된다.

III. 실험 및 고찰

hMETIS를 이용한 분할기법과 RBLs 알고리즘을 이용한 “top-down” 방식으로 구현한 HGP의 성능을 측정하기 위해 우리는 MCNC 표준 벤치마크 회로^[29]를 사용하였으며, 기존 하이브리드 배치기^[25]의 결과와 FengShui 툴의 결과를 상호 비교하였다. 실험을 위해

표 1. 회로 사양

Table 1. Circuit characteristics.

Circuit	#Nets	#Cells
Prim1	904	833
Struct	1,920	1,888
Prim2	3,019	3,014
Biomed	5,742	6,417
Ind2	13,419	12,142
Ind3	21,940	15,059
Avqs	22,124	21,854
Avql	25,384	25,114

표 2. 배선길이 비교(with row spacing), 단위 : micron

Table 2. Comparison of wire-length (with row spacing) in micro-meter.

Circuit	하이브리드배치기	Feng Shui	HGP	Improvement over	
				하이브리드배치기	Feng Shui
Prim1	965,615	1,044,291	983,195	-1.82%	5.85%
Struct	706,980	755,176	723,433	-2.33%	4.20%
Prim2	3,592,968	3,781,567	3,668,276	-2.10%	3.00%
Biomed	3,485,918	3,403,408	3,264,574	6.35%	4.08%
Ind2	16,160,831	15,627,343	15,200,894	5.94%	2.73%
Ind3	44,687,804	45,960,343	44,561,743	0.28%	3.04%
Avqs	5,722,467	5,653,141	5,392,366	5.77%	4.61%
Avql	6,161,967	6,210,026	5,834,924	5.30%	6.04%

표 3. 배선길이 비교(without row spacing), 단위 : micron

Table 3. Comparison of wire-length (without row spacing) in micro-meter.

Circuit	하이브리드배치기	Feng Shui	HGP	Improvement over	
				하이브리드배치기	Feng Shui
Prim1	849,603	846,822	866,426	-1.98%	-2.31%
Struct	569,646	516,236	559,056	1.86%	-8.29%
Prim2	2,916,407	3,007,493	2,780,673	4.65%	7.54%
Biomed	2,725,123	2,735,042	2,481,783	8.93%	9.26%
Ind2	12,364,077	10,454,680	11,112,050	10.13%	-6.29%
Ind3	34,726,862	33,832,792	32,375,585	6.77%	4.30%
Avqs	4,381,825	4,347,417	4,178,180	4.65%	3.89%
Avql	4,802,575	4,921,154	4,497,068	6.36%	8.62%

표 4. 수행시간 비교 (with row spacing), 단위 : sec

Table 4. Comparison of CPU time (with row spacing) in second.

Circuit	하이브리드배치기	HGP	Speedup
Prim1	34	45	5
Struct	63	9.4	5
Prim2	200	26.3	5
Biomed	423	67.2	2
Ind2	1,140	467.2	2
Ind3	2,328	248.6	6
Avqs	4,869	363.3	9
Avql	5,626	555.3	6

표 5. 수행시간 비교(without row spacing), 단위 : sec

Table 5. Comparison of execution time (without row spacing) in second.

Circuit	하이브리드배치기	HGP	Speedup
Prim1	60	5.5	6
Struct	72	8.8	5
Prim2	214	31.6	5
Biomed	444	58.7	5
Ind2	771	328.3	2
Ind3	2952	240.0	8
Avqs	7070	507.6	9
Avql	8464	540.7	10

펜티엄IV, 2GHz/Linux 컴퓨터를 사용하였다. 실험에 사용된 회로의 사양은 표 1과 같다.

FengShui의 결과와 동일한 조건에서 비교하였으며, 행간에 라우팅을 위한 스페이스가 있는 경우와 없는 경우 각각에 대해 비교하였다. 하이브리드 배치가 발표된 CPU 시간은 Sun Ultra 10/Solaris 상에서 수행한 것이어서 CPU 시간 자체를 직접 비교하는 대신, 이를 정규화시켜 HGP 상대적으로 얼마나 빠른지를 나타내었다. 상이한 두 머신의 속도를 보정하기 위해 우리는 펜

표 6. 수행시간 분석(with row spacing, #runs=10)

Table 6. Analysis of CPU time (with row spacing, #runs=10).

Circuit	Best	Worst	Average	Average CPU Time
Prim1	983,195	1,000,956	9,992,412	5.98
Struct	723,433	758,375	734,774	11.18
Prim2	3,668,276	3,836,315	3,734,385	28.44
Biomed	3,264,574	3,368,890	3,318,839	74.98
Ind2	15,200,894	15,492,207	15,383,167	240.49
Ind3	44,561,743	45,544,652	45,068,204	297.20
Avqs	5,392,366	5,671,353	5,504,492	494.79
Avql	5,834,924	6,073,246	5,943,218	510.46

표 7. 수행시간 분석(without row spacing, #runs=10)

Table 7. Analysis of CPU time (without row spacing, #runs=10).

Circuit	Best	Worst	Average	Average CPU Time
Prim 1	866,426	894,802	882,081.8	5.99
Struct	559,056	571,387	565,936.4	10.59
Prim 2	2,780,673	2,908,603	2,841,269	33.53
Biomed	2,481,783	2,566,588	2,523,849	65.29
Ind 2	11,112,050	11,379,511	11,232,828	205.65
Ind 3	32,375,585	32,968,142	32,692,002	342.81
Avqs	4,178,180	4,297,542	4,230,068	434.88
Avql	4,497,068	4,686,960	4,580,994	486.83

티엄 IV, 2GHz/Linux 머신이 Sun Ultra 10/Solaris보다 대략 1.5배 빠름을 알았고 이를 고려하여 속도가 얼마나 개선되었는지를 나타내었다.

광역배치를 최적화한 다음 상세배치로 변환하는 방법과, 상세배치를 최적화하는 기법은 하이브리드 배치기에서 사용하는 것과 동일한 방법을 사용하였다.

표 2와 3에서 보인 결과는 각 회로에 대해 10번 (#runs=10)의 수행을 거친 결과를 구하여 그 중 배선길이 가장 짧은 것에 대한 요약이다. "middle-down" 접근법을 사용한 기존 하이브리드 배치기에 대하여 비교적 회로의 크기가 작은 Prim1, Prim2, Struct를 제외하고 배선 길이 면에서 약 4.7%의 개선율을 보였으며 FengShui 툴과의 비교에서는 약 4.2%의 개선율을 보였다.

"top-down" 접근법을 사용한 제안기법은 수행 속도 면에서 월등히 개선됨을 표 4와 5을 통해 알 수 있다. 기존 하이브리드 배치기와 비교해 볼 때 평균 5배의 속도향상을 가져왔다.

또한 새로운 배치기의 결과가 얼마나 안정적인지를 보이기 위해 10개의 결과 모두에 대한 평균/최고/최악의 수행시간을 표 6과 7에 비교하였다. 표에서 보듯이

가장 좋은 결과와 가장 나쁜 결과의 차이가 크지 않아 제안된 기법을 사용하는 배치기 HGP는 매우 안정적으로 좋은 결과를 생성하는 것을 알 수 있다.

IV. 결 론

본 논문에서는 “middle-down” 접근법의 단점을 해결하고 RBLs의 독특한 광역배치 매커니즘을 향상시킬 수 있는 “top-down” 방식의 개선된 광역배치 알고리즘을 제안하였다. 실험 결과에 의하면 제안된 기법이 다음과 같은 측면에서 기존의 하이브리드 배치기보다 우수함을 보여준다.

- ① 매우 안정적이다. 즉, 배치결과의 편차가 크지 않다.
- ② 속도가 빠르다: 속도가 개선된 이유는 크게 세 가지가 있다. 첫째, “top-down” 방법을 사용함으로써 셀의 위치를 거시적인 관점에서 빨리 구한다. 둘째, hMetis를 사용함으로써 각 셀이 속할 빈을 효과적으로 구한다. 셋째, RBLs 알고리즘이 적용될 회로 자체가 군집회로이어서 회로 내에 있는 셀 수가 줄어든다. 즉, RBLs 알고리즘의 입력 사이즈가 줄어드는 효과가 있다.
- ③ 크기가 큰 회로에 대해선 결과도 개선된다.

배선길이의 측정에서 일부 경우에 결과가 다소 나빠진 것을 볼 수 있는데 그 이유는 “top-down” 방식으로 접근해 가면서 격자의 차원(dimension), 특히 열의 수에 대한 조정이 정확하지 않음(열의 수를 2^k 로 고정시키는 것)에 기인하는 것으로 추정하고 있다. 그러나 큰 회로에 대해서 모두 좋은 결과를 보이는 점에서 새로운 배치기의 가능성을 볼 수 있다. 앞으로는 IBM 회로와 같이 규모가 큰 회로를 효과적으로 다루기 위한 클러스터링 기법의 연구가 필요할 것이다.

또한 타이밍을 고려한 광역배치도 앞으로 연구할 제목이다. 하이브리드 배치기에서는 “ripple-movement”를 통해 셀들이 격자 사이를 이동하는데 지금은 배선길이란만 고려하지만 타이밍 정보, 예를 들어 어떤 넷이 크리티컬 넷인지를 알 수 있다면 쉽게 타이밍을 고려할 수 있도록 수정할 수 있을 것으로 사료된다. “Ripple-movement”를 위해 셀과 목적지를 선택하는 알고리즘이 타이밍을 고려하기에 적합하기 때문에 이에 대한 문제는 쉽게 해결 될 수 있을 것으로 보인다.

또한 배선 밀집을 고려한 연구도 앞으로의 과제로 보

인다. 배치 후 배선이 불가능하다면 상당 수, 어쩌면 모든 셀의 위치를 원점에서 다시 고려해야 할 상황이 발생할 수 있기 때문이다. 배선이 설사 가능하다 하더라도 특정 지역에 배선 밀집도가 너무 높으면 돌아가는 배선(detour)이 불가피하게 되고, 이는 타이밍에 또 영향을 미칠 수 있다. 그러므로 배치 과정에서 배선 밀집도가 높지 않으면서 좋은 배치를 얻는 것은 매우 중요한데 이는 앞으로 계속 연구되어야 할 과제로 생각된다.

참 고 문 헌

- [1] P. Villarrubia, "Important Placement Considerations for Modern VLSI Chips," *Proc. of ISPD*, pp.6, 2003.
- [2] C. Sechen and A. Sangiovanni-Vincentelli, "TimberWolf3.2: A New Standard Cell Placement and Global Routing Package," *Proc. of the DAC*, pp.432-439, 1986.
- [3] M. Sarrafzadeh and M. Wang, "NRG: global and detailed placement," *Proc. of ICCAD*, pp.532-537, 1997.
- [4] C. Sechen and K. W. Lee, "An Improved Simulated Annealing Algorithm for Row-Based Placement," *Proc. of ICCAD*, pp. 478-481, 1987.
- [5] X. Yang, M. Wang, K. Eguro, and M. Sarrafzadeh, "A snap-on placement tool," *Proc. of ISPC*, pp.153-158, 2000.
- [6] A. E. Caldwell, A. B. Kahng, and Igor L. Markov, "Can Recursive Bisection Alone Produce Routable Placements?," *Proc. of DAC*, pp.477-482, 2000.
- [7] D. J.-H. Huang, and A. B. Kahng, "Partitioning-Based Standard-Cell Global Placement with an Exact Objective," *Proc. of ISPD*, pp.18-25, 1997.
- [8] M. C. Yildiz and P. H. Madden, "Improved Cut Sequences for Partitioning Based Placement," *Proc. of DAC*, pp.776-779, 2001.
- [9] Ke Zhong, and Shantanu Dutt, "Effective Partition-Driven Placement with Simultaneous Level Processing and Global Net Views," *Proc. of ICCAD*, pp. 254-259, 2000.
- [10] Jason Cong, Michail Romesis, and Min Xie, "Optimality, Scalability, and Stability Study of Partitioning and Placement Algorithms," *Proc. of ISPD*, pp.88-94, 2003.
- [11] A. E. Caldwell, A. B. Kahng, and I. L. Markov, "Optimal Partitioners and End-Case Placers for Standard-Cell Layout," *Proc. of ISPD*, pp.

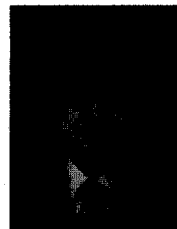
- 90-96, 1999.
- [12] B. W. Kernighan, and S. Lin, "An Efficient Heuristic Procedure for Partitioning Graphs," *Bell Syst. Tech.*, vol.49, no.2, pp.291-308, 1970.
- [13] C. M. Fiduccia and R. M. Mattheyses, "A Linear-Time Heuristic for Improving Network Partitions," *Proc. of DAC*, pp. 175-181, 1982.
- [14] H. Eisenmann and F. M. Johannes, "Generic Global Placement and Floorplanning," *Proc. of DAC*, pp.269-274, 1998.
- [15] Karthik Rajagopal, Tal Shaked, Yegna Parasuram, Tung Cao, Amit Chowdhary, Bill Halpin, "Timing Driven Force Directed Placement with Physical Net Constraints," *Proc. of ISPD*, pp. 60-66, 2003.
- [16] N. Viswanathan and Chris C. Chu, "FastPlace: Efficient Analytical Placement Using Cell Shifting, Iterative Local Refinement and a Hybrid Net Model," *IEEE Trans. CAD of Integrated Circuits and Systems*, vol.24, no.5, pp.722-733, 2005.
- [17] S. Goto, "An Efficient Algorithm for the Two-Dimensional Placement Problem in Electrical Circuit Layout," *IEEE Trans. on Circuits and Systems*, vol.28-1, pp.12-18, 1981.
- [18] P. N. Parakh, R. B. Brown, and K. A. Sakallah, "Congestion Driven Quadratic Placement," *Proc. of the 35th DAC*, pp. 275-278, 1998.
- [19] X. Yang, B.-K. Choi, and M. Sarrafzadeh, "Routability Driven White Space Allocation for Fixed-Die Standard-Cell Placement," *Proc. of ISPD*, pp.42-47, 2002.
- [20] H. Etawil, S. Areibi, and A. Vannelli, "Attractor-Repeller Approach for Global Placement," *Proc. of ICCAD*, pp.20-24, 1999.
- [21] M. Wang, X. Yang, and M. Sarrafzadeh, "Dragon2000: Standard-Cell Placement Tool for Large Industry Circuits," *Proc. of ICCAD*, pp.260-263, 2000.
- [22] X. Yang, B.-K. Choi and M. Sarrafzadeh, "A Standard-Cell Placement Tool for Designs with High Row Utilization," *Proc. of the 2002 IEEE International Conference on Computer Design*, pp.45-49, 2002.
- [23] R.-M. Kling and P. Banerjee, "ESP: A New Standard Cell Placement Package Using Simulated Evolution," *Proc. of DAC*, pp. 60-66, 1987.
- [24] T. Chan, J. Cong, T. Kong, and J. Shinnerl, "Multilevel Optimization for Large-Scale Circuit Placement," *Proc. of ICCAD*, pp.171-176, 1999.
- [25] 허성우, 오은경, "표준 셀 배치를 위한 하이브리드 기법," *정보과학회 논문지: 시스템 이론*, Vol.30, No.9·10, pp.595-602, 2003.
- [26] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar, "Multilevel Hypergraph Partitioning: Application in VLSI Domain," *Proc. of DAC*, pp.526-529, 1997.
- [27] C. J. Alpert and A. B. Kahng, "A General Framework for Vertex Orderings, with Applications to Netlist Clustering," *Proc. of ICCAD*, pp.63-67, 1994.
- [28] S. N. Adya, M. C. Yildiz, I. L. Markov, P. G. Villarrubia, P. N. Parakh, and P. H. Madden, "Benchmarking for Large-Scale Placement and Beyond," *Proc. of ISPD*, pp. 95-103, 2003.
- [29] <http://vlsicad.eecs.umich.edu/BK>

 저 자 소 개



성 영 태 (학생회원)
 2002년 동아대학교 컴퓨터공학과
 학사 졸업.
 2004년 동아대학교 컴퓨터공학과
 석사 졸업.
 2004년~현재 동아대학교 컴퓨터
 공학과 대학원 박사과정.

<주관심분야 : 알고리즘, VLSI CAD>



허 성 우 (정회원)
 1981년 경북대학교 전자공학과
 학사 졸업.
 1983년 한국과학기술원(KAIST)
 전산학과 석사 졸업.
 2000년 UIC Dept. of EECS 박사
 졸업.

1986년~현재 동아대학교 전자컴퓨터공학부
 교수.

2001년~현재 미국 Intel사 Physical Design분야
 기술자문위원.

<주관심분야 : CAD, 알고리즘, 계산 기하학,
 Combinatorial optimization>