Mongrel : 계층적 분할 기법을 이용한 광역 배치

성영태°, 허성우
동아대학교 컴퓨터 공학과
saint°@donga.ac.kr, swhur@daunet.donga.ac.kr

Mongrel : Global Placement with Hierarchical Partitioning

YoungTae Sung° SungWoo Hur
Dept. of Computer Engineering, Dong-A University

요 약

본 논문에서는 표준 셀 배치기 Mongrel의 성능을 개선하기 위해 사용된 다양한 기법에 관해 살펴보고 top-down 방식의 계층적 분할 기법을 이용한 광역 배치(Hierarchical Global Placement)를 제안한다. 계층적 분할 기법을 이용한 광역 배치는 RBLS(Relaxation Based Local Search) 기법과 더불어 Mongrel의 성능 개선에 결정적인 역할을 하고 있으며 분할 기법으로 hMETIS(클러스터링을 이용한 다단계 분할 기법)를 사용한다. 우리는 표준 벤치마크 회로를 이용한 실험을 통해 계층적 분할 기법을 이용한 광역 배치 기법이 안정적이면서 효율적인 배치 결과를 가져옴을 보인다.

1. 서 론

표준 셀 배치기 Mongrel은 [1]에 처음 소개된 이후 계속적인 수정 보완 과정을 통해 개선되어 왔다. Mongrel은 광역 배치와 상세 배치의 2단계 배치 과정을 거치는 복합형 배치기로서 핵심 배치 기법으로 RBLS와 최적 인터리빙을 사용한다. 초기 Mongrel은 광역 배치 단계에서 RBLS를 수행함에 있어 middle-down 방법론을 채택했다. Middle-down 방법론은 n X m 격자 상에 이미 셀들이 배치되어 있다는 가정하에 RBLS를 적용하는 구조로써 기본적인 목적 함수인 배선 길이의 최적화에는 상당히 효과적임을 보였다. 그러나 배치 과정 중 RBLS가 차지하는 배치 소요 시간이 전체 80-90%이 될 정도로 속도상의 문제점을 보였다.

본 논문에서는 먼저 광역 배치의 핵심 기법인 RBLS에 관해 살펴보고 수행 시간 개선을 위해 사용되었던 몇 가지 기법들을 고찰한다. 그리고 수행 시간상의 효과적인 개선을 보인 계층적 분할 기법을 이용한 광역 배치를 제안하고 실험을 통해 배치 기법의 안정성과 효율성을 보인다.

2. Relaxation Based Local Search 기법의 개요

RBLS(Relaxation Based Local Search)[2]는 해석적 기법을 이용하여 거시적인 관점에서 전역 배치를 수행할 수 있도록 한다. 이의 동작은 다음과 같이 요약될 수 있다. 우선 넷 정보에 근거하여 mobile 셀의 집합인 부분 회로를 추출한다. 그리고 네트웍 플로우 기법을 통해 n X m 격자 상에 부분 회로의 최적 위치를 찾는다. 이때 격자에 있는 각 빈의 제약 조건은 무시한다. 그런 후 빈의 제약조건을 만족시키기 위해 ripple-move 기법을 사용한 legalization(적법화) 과정을 거친다. 새롭게 얻은 광역 배치의 각 인접 빈에 대해 FM(Fiduccia-Mattheyses) 알고리즘을 적용시켜 이를 더욱 개선한다. 이 과정을 그림 1에 요약하였다.

```
Input: s, k and current placement P
Output: new placement
counter ← k
while(counter > 0) {
  Select a mobile cell set M(|M|=s)
  For each cell in M, Determine optimal relaxed location
  P' ← new placement after ripple-movement
  P' ← optimize P' with FM-partitioning
  if(WL(P') < WL(P)) then
    counter ← k
    P ← P'
  else
    counter ← counter-1
}
return P
```

그림 1. RBLS 알고리즘

다음으로 RBLS와 더불어 Mongrel의 성능 개선을 위해 사용되었던 몇 가지 기법을 개괄적으로 살펴 본다.

3. 순서화 알고리즘을 이용한 초기 배치

클러스터링 기법은 다양한 논문을 통해 배치에 소요되는 수행 시간을 개선 할 수 있는 한 가지 방법으로 알려져 왔다. Mongrel은 이러한 클러스터링 기법을 이용하여 초기 배치를 얻은 후 광역 배치를 수행하는 2단계 배치 기법을 보였다. 클러스터링 기법을 사용한 대부분의 논문에서는 회로의 특성을 분석하여 연관도가 높은 셀들을 클러스터링 하는 기법을 사용하였지만 연관도가 높은 셀들의 집합이 미리 회로 특성에 의해 결정되기 때문에 다양한 압축 회로를 얻을 수 없는 단점이 있다.

이것을 해결하기 위해 Mongrel은 초기 배치 단계에서 Alpert와 Kahng[3]이 제안한 순서화 함수를 사용하여 각 셀들을 선형 배치한 후 선형 배치된 결과로부터 인접한 셀들을 클러스터링 하여 다양한 압축된 회로를 얻었다. 압축된 회로는 FM 알고리즘을 통해 단계적으로 분할 과정을 거쳐 초기 배치를 얻게 된다.[4]

4. 분할 기법을 이용한 초기 배치

초기 배치를 얻는 또 다른 시도로 분할 기법을 사용하였다. 분할에 사용된 기법은 크게 FM 알고리즘과 클러스터링을 이용한 다단계 분할기법(hMETIS)이 있다. 분할 기법을 이용한 초기 배치는 다음과 같은 단계를 거친다. 먼저 주어진 배치를 1 X 1격자 상태로 보고 이것을 단계적으로 2분할 한다. 이때 분할 정보는 슬라이싱 트리에 저장되고 반복적으로 2분할과 슬라이싱 트리 재구성 과정을 거쳐 초기 배치를 얻는다. 최종 분할 후 각 셀의 위치는 해당 격자의 중심 좌표로 할당되고 RBLS를 이용한 광역 배치를 수행한다.[5]

5. 계층적 분할 기법을 이용한 광역 배치

광역 배치에 소요되는 CPU시간은 전체 배치 시간에 거의 대부분을 차지한다. 광역 배치가 최적화에 많은 시간을 소비하는 주된 원인은 다음과 같다. Mongrel은 처음부터 n X m 격자를 이용하고, 초기 배치를 이 격자 상에서 구한 후 이를 RBLS 알고리즘을 이용하여 개선하게 되는데 이때, 회로에 대한 거시적인 관점을 가지기 위해 부분 회로를 추출함에 있어 전체 셀의 약 80%에 해당하는 셀을 선택하게 된다. 이렇게 많은 셀들을 선택하여 최적 위치를 찾다 보니 특정 빈에 많은 셀들이 놓이게 되어 중첩의 정도가 심해지고, 이를 해결하기 위해 셀들을 이동하는 시간이 많이 필요하기 때문이다.

```
Input : circuit information, #rows
Output : optimized global placement
Determine m(m=2^k)
i ← 0
P ← place all cells in 2^i X 2^i grid
while(i < k) {
  i++
  P ← partition current P into 2^i X 2^i grid
  P' ← call RBLS(s, t, P)
  while(WL(P' ) < WL(P)) {
    P ← P'
    P' ← call RBLS(s, t, P)
    change s and t
  }
}
P ← Partition current P into n X 2^k grid
while(WL(P' ) < WL(P)) {
  P ← P'
  P' ← call RBLS(s, t, P)
  change s and t
}
return P
```

그림 2. 계층적 분할 기법을 이용한 광역배치 알고리즘

본 논문에서 제안한 기법은 이러한 단점을 해결하기 위해 초기부터 n X m 격자를 사용하는 middle-down 접근 대신 top-down 방식과 hMETIS에서 제안한 분할 기법을 사용하였다. 즉, 초기에 2 X 2 격자를 고려하고, 분할기법을 이용하여 이 격자 상에서 셀들의 위치를 결정한다. 그런 다음 RBLS 알고리즘을 적용하여 초기의 배치를 개선한다. 더 이상 개선되지 않으면 격자 상의 각 빈을 4등분 하며, 각 빈에 속한 셀들도 같이 분할하여 전체적으로 4 X 4 격자 상에서 새로운 배치를 얻고, 이 격자 상에서 RBLS 알고리즘을 적용한다. 이런 과정을 반복하여 격자가 2^k X 2^k가 될 까지 한다. 그런 다음 각 빈을 수평적으로 다시 분할하여 전체적으로 격자가 n X m이 되도록 조정한다. 이 과정을 그림 2에 요약하였다.

6. 실험 및 고찰

RBLS 알고리즘과 hMETIS을 이용한 top-down 방식의 계층적 분할 기법을 이용한 광역 배치기의 성능을 비교하기 위해 우리는 MCNC 벤치마크 회로를 사용하였으며, 실험을 위해 펜티엄IV, 2GHz/Linux 컴퓨터를 사용하였다. 실험에 사용된 회로의 사양은 표 1과 같다.

표 1. 회로 사양

Circuit	#Nets	#Cells
Prim1	904	833
Struct	1,920	1,888
Prim2	3,019	3,014
Biomed	5,742	6,417
Industry2	13,419	12,142
Industry3	21,940	15,059
Avq small	22,124	21,854
Avq large	25,384	25,114

행간에 라우팅을 위한 스페이스가 있는 경우와 없는 경우에 대해 각각 실험하였고 표2와 3 그리고 표 4와 5에 각각 배선 길이와 수행 시간을 나누어 정리하였다. 그리고 결과 비교는 기존의 Mongrel과 Feng-Shui[6] 툴을 대상으로 수행하였다. 표 2~7에서 보인 결과는 각 회로에 대해 10개의 결과를 구하여 그 중 배선길이가 가장 짧은 것에 대한 결과이다. 몇몇 경우를 제외하고 배선길이가 개선되었으며 속도도 industry2 회로를 제외하면 최소 5배에서 10배까지 빨라진 것을 볼 수 있다.

표 2. 배선 길이 비교1 (With Row Spacing, #runs=10), 단위 : micron

Circuit	기존 Mongrel	FengShui	HG Mongrel	Improvement Over	
				기존 Mongrel	FengShui
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%
Industry2	16,160,831	15,627,343	15,200,894	5.94%	2.73%
Industry3	44,687,804	45,960,343	44,561,743	0.28%	3.04%
Avq small	5,722,467	5,653,141	5,392,366	5.77%	4.61%
Avq large	6,161,967	6,210,026	5,834,924	5.30%	6.04%

표 3. 배선 길이 비교2 (Without Row Spacing, #runs=10), 단위 : micron

Circuit	기존 Mongrel	FengShui	HG Mongrel	Improvement Over	
				기존 Mongrel	FengShui
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%
Industry2	12,364,077	10,454,680	11,112,050	10.13%	-6.29%
Industry3	34,726,862	33,832,792	32,375,585	6.77%	4.30%
Avq small	4,381,825	4,347,417	4,178,180	4.65%	3.89%
Avq large	4,802,575	4,921,154	4,497,068	6.36%	8.62%

표 4. 수행 시간 비교1 (With Row Spacing, #runs=10), 단위 : sec

Circuit	기존Mongrel	HG Mongrel	Speed-up
Prim1	34	4.5	5
Struct	63	9.4	5
Prim2	200	26.3	5
Biomed	423	67.2	2
Industry2	1,140	467.2	2
Industry3	2,328	248.6	6
Avq small	4,869	363.3	9
Avq large	5,626	555.3	6

표 5. 수행 시간 비교2 (Without Row Spacing, #runs=10), 단위 : sec

Circuit	기존Mongrel	HG Mongrel	Speed-up
Prim1	60	5.5	6
Struct	72	8.8	5
Prim2	214	31.6	5
Biomed	444	58.7	5
Industry2	771	328.3	2
Industry3	2,952	240.0	8
Avq small	7,070	507.6	9
Avq large	8,464	540.7	10

일부 경우에 결과가 다소 나빠진 것을 볼 수 있는데 이는 top-down 방식으로 접근해 가면서 격자의 dimension, 특히 열의 수에 대한 조정이 정확하지 못해서 그런 것으로 추정하고 있다. 그러나 큰 회로에 대해 모두 좋은 결과를 보이는 점에서 새로운 배치기의 가능성을 볼 수 있다. 또한 새로운 배치기의 결과가 얼마나 안정적인지를 보이기 위해 10개의 결과 모두에 대한 평균값과 최악의 경우에 대해 표 6과 표 7에서 각각 보였다.

표 6. 결과 분석1 (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
Industry2	15,200,894	15,492,207	15,383,167	240.49
Industry3	44,561,743	45,544,652	45,068,204	297.20
Avq small	5,392,366	5,671,353	5,504,492	494.79
Avq large	5,834,924	6,073,246	5,943,218	510.46

표 7. 결과 분석2 (Without Row Spacing, #runs=10)

Circuit	Best	Worst	Average	Average CPU Time
Prim1	866,426	894,802	882,081	5.99
Struct	559,056	571,387	565,936	10.59
Prim2	2,780,673	2,908,603	2,841,269	33.53
Biomed	2,481,783	2,566,588	2,523,849	65.29
Industry2	11,112,050	11,379,511	11,232,828	205.65
Industry3	32,375,585	32,968,142	32,692,002	342.81
Avq small	4,178,180	4,297,542	4,230,068	434.88
Avq large	4,497,068	4,686,960	4,580,994	486.83

7. 결 론

본 논문에서는 계층적 분할 기법을 이용한 광역 배치 기법을 제안하였다. MCNC 벤치마크 회로를 이용한 실험을 통해 제안한 Mongrel의 수행 속도가 최대 10배 까지 향상 되었음을 알 수 있었다. 또한 각각의 회로에 대한 수행 결과를 분석해 볼 때 가장 좋은 결과와 가장 나쁜 결과의 차이가 크지 않아 제안한 배치기가 매우 안정적으로 좋은 결과를 생성하는 것을 알 수 있다.

참고문헌

[1] Sung-Woo Hur and John Lillis, "Mongrel: Hybrid Techniques for Standard Cell Placement," *Proc. of ICCAD*, pp.165-170, 2000.

[2] Sung-Woo Hur and John Lillis, "Relaxation and Clustering in a Local Search Framework: Application to Linear Placement," *Proc. of the 36th ACM/IEEE conference on Design Automation*, 1999.

[3] Charles J. Alpert and Andrew B. Kahng, : "A general framework for vertex orderings with applications to circuit clustering," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, v.4 n.2, p.240-246, June 1996.

[4] 배종국, 오은경, 허성우, " 초기 분할이 VLSI 칩 설계에 미치는 영향," *한국 정보 과학회*, 2001.

[5] 성영태, 허성우, " 효율적인 초기 배치를 이용한 개선된 Mongrel," *한국 정보 과학회*, 2004.

[6] Patrick H. Madden, "Reporting of Standard Cell Placement Results," *IEEE Transaction of CAD*, pp. 240-247, 2002.