

LSI의 Layout CAD에 있어서의 配線 混雜度를 고려한 配置 問題 (A Placement Problem with Wire Congestion in LSI Layout CAD)

林寅七*, 鄭正和*, 李丙鎬*

(In-Chil Lim, Jung-Hwa Jung and Byung-Ho Lee)

要 約

LSI, 프린트 基板 등의 配置 問題에 있어서의 목적 함수로써 總配線長의 最小化, 通過線數의 最小化가 지금까지 채용되어 왔었다. 이 목적 함수는 layout의 궁극적인 목표(100% 結線率)를 大局의으로는 反映하고 있다고 말할 수 있지만 局所의으로 혼잡한 配線 狀態를 나타내고 있지 않기 때문에 목적 함수로써 불충분한 경우가 많이 보여 지고 있다.

本 論文에서는 局所의인 配線 混雜度를 나타내기 위해 세그먼트(segment) 混雜度라고 하는 새로운 개념을 導入하여 이 세그먼트 混雜度를 最小로 하는 것을 목적 함수로 하는 것을 제안하였다. 그리고 이 목적 함수를 最適化 하기 위해 능률 좋은 휴리스틱 알고리즘을 제안하고 프로그램 실험에 의해 그 有用性을 확인하였다.

세그먼트 混雜度의 最大値는 總配線長이 짧게 되면 적게 되고 總配線長이 길게 되면 크게 되는 것이 실험의 결과로 얻어져 總配線長을 짧게 하고 동시에 세그먼트 混雜度의 最大値를 적게 하는 제어 파라미터(混雜度 parameter)의 값의 범위를 실험적으로 구하였다.

Abstract

Minimization of total routing and number of cuts has been adopted for the placement problem in LSI and printed wire board as the object function, recently. Although it is considered that in the general situation this object function reflects the final goal which is wiring of 100% of layout, it often seems to be insufficient because it does not exhibit partial wire congestion.

This paper introduces a new concept called the wire congestion of segment to get the partial wire congestion and proposes the object function to minimize the wire congestion of segments.

In order to optimize this object function, an effective heuristic algorithm is also suggested. Experimental results show that this algorithm sustains its efficiency. The experimental consequences point out that if the total routing length is short, maximum wire congestion of segment is small and vice versa. Therefore control parameter, that is, congestion parameter, which minimizes total length and concurrently reduces maximal wire congestion of segment, is obtained by experiment.

*正會員, 漢陽大學校 工科學科 電子工學科
(Dept. of Elec. Eng., Hanyang Univ.)

接受日字: 1982年 3月 6日

size)의 증대에 수반해서 1 칩 內에 搭載해야 할 素子는 최근 2~3年 사이에 대폭 증대하고 있다. 이 때문에 대규모의 복잡한 LSI를 高信賴度로써 값싸고 단기간에 설계 제조하는 것이 강력히 요구되고 있다.

LSI의 설계에 있어서 현재 최대의 과제중에 하나는 layout 설계이다.

Layout 설계는 素子를 칩에 配置하는 것과 素子間의 信號線을 配線하는 것으로 구성된다. 이들 두 가지의 문제는 본래 하나의 문제로서 동시에 풀어야 할 문제이나 CAD를 前提로 하는 경우 콤퓨터 처리 시간의 관점으로 부터 각각 나누어서 푸는 것이 불가피하다.

配置 問題에 있어서는 실제의 配線을 하지 않고 푸는 것이 되기 때문에 layout 문제의 궁극적인 목표(100% 結線率)를 정확하게 反映한 配置 問題의 목적 함수를 설정할 필요가 있다.

지금까지 配置 問題의 목적 함수로서 이하의 것이 채용되어 왔다.

- 1) 總配線長의 最小化^[1]
- 2) 通過線數의 最小化^[2]

이들 목적 함수는 계산이 容易함과 동시에 大局的으로는 100% 結線率을 反映하고 있는 것으로 알려져 있다. 그러나 이들은 局所的으로 혼잡한 配線 狀態를 나타내고 있지 않기 때문에 실제의 데이터에서는 목적 함수로서 불충분한 경우를 종종 볼 수 있다.

또한 混雜度가 증가할 경우 結線率의 低下는 불가피하며 특히 局所的인 混雜度의 증가가 대부분의 원인이 되고 있다.

本 論文에서는 局所的인 配線의 混雜을 충분히 反映한 목적 함수를 생각한다. 局所的인 混雜度를 나타내기 위해 세그먼트 混雜度를 導入하고 이 세그먼트 混雜度를 최소로 하는 것을 목적 함수로 하는 것을 제안한다. 여기서 대상으로 하는 문제는 어려운 組合 문제에 속하고 最適解를 얻는 것은 곤란하다. 따라서 능률 좋은 휴리스틱 알고리즘을 導入하고, 좋은 解를 빨리 구하는 알고리즘을 제안한다. 最後에, 제안한 알고리즘의 프로그램 실험을 하고 配置의 評價(figure of merit)에 대한 실험 결과에 관해서 記述한다.

II. 準 備

本 論文에서 취급하는 配置 問題는 미리 크기가 결정된 2次元 基板(chip)에 같은 크기의 要素(module)를 配置하는 것으로 한다. 구체적인 대상은 마스터 슬라이스 방식의 LSI를 말하고 있다.

以下の 용어를 정의한다.

모듈(module) : 配置하고자 하는 要素(보통 機能 블록이라 부름)

칩(chip) : 모듈을 배치해야 할 2次元 평면판

셀(cell) : 칩을 구성하는 要素로 모듈이 할당된다.

핀(pin) : 모듈 上에 있는 端子로 다른 모듈과의 電氣的인 접속은 이 端子를 통해서 행한다.

信號線 : 모듈 上의 몇 개의 핀을 전기적인 等電位로 연결한 線

여기서, 모듈→집적회로, 칩→보-드, 셀→스토포트라고 한다면 프린트 基板의 layout 문제에도 적용될 수 있다.

本 論文에서는 문제의 本質을 보다 명확히 하기 위해서 다음과 같이 문제를 간단히 하여 취급한다.

1) 모듈 上의 핀의 위치를 고려하지 않고 信號線은 가상적으로 모듈의 中心間을 연결하는 것으로 한다.

2) 셀은 칩 上에 규칙적이고 縱橫으로 배열되어 있는 것으로 한다. 信號線이 인접한 셀間을 연결할 때 그 길이는 1로 한다.

3) 하나의 셀에는 단 하나의 모듈이 할당되어 지는 것으로 한다.

칩이 m行 n列의 셀로 구성되고 그 칩 上에 (m×n)개의 모듈이 할당되어지는 것으로 한다. 일반적으로 모듈의 수는 셀의 數보다 적지만 여기에서는 信號線의 접속이 없는 모듈(dummy module)을 정의하고 兩者의 수는 같은 것으로 한다. 2개의 셀의 경계를 세그먼트라 부른다. 각 세그먼트에는 收容 可能한 配線數가 주어지고 있다. 信號線의 配線 經路는 세그먼트의 列로써 나타나고 모든 信號線의 配線 經路가 주어졌다면 각 세그먼트에 할당되는 配線數가 계산 가능하다.

다음의 기호를 導入한다. (그림 1 참조)

E_{ij}^x : 횡방향 세그먼트(X-directional segment)

단, $i=1,2,\dots,m$. $j=1,2,\dots,n-1$.

E_{ij}^y : 종방향 세그먼트(Y-directional segment)

단, $i=1,2,\dots,m-1$. $j=1,2,\dots,n$.

C_{ij}^x : 세그먼트 E_{ij}^x 에 收容 可能한 配線數

C_{ij}^y : 세그먼트 E_{ij}^y 에 收容 可能한 配線數

X_{ij}^x : 세그먼트 E_{ij}^x 에 할당된 配線數

Y_{ij}^y : 세그먼트 E_{ij}^y 에 할당된 配線數

$f(X_{ij}, P)$: 세그먼트 E_{ij}^x 에 X_{ij} 개의 配線數가 할당되어진 때의 E_{ij}^x 의 코스트(cost) 단, p는 混雜度 파라미터

$f(Y_{ij}, P)$: 세그먼트 E_{ij}^y 에 Y_{ij} 개의 配線數가 할당되어진 경우의 E_{ij}^y 의 코스트 단, P는 混雜度 파라미터

X_{ij} / C_{ij}^x 및 Y_{ij} / C_{ij}^y 를 세그먼트 混雜度(wire congestion of segment)라고 한다. 코스트 함수 $f(X_{ij}, P)$ 는 다음 식을 만족하는 것으로 한다.

$$\lim_{P \rightarrow \infty} f(X_{ij}, P) = \begin{cases} 0 & : X_{ij} \leq C_{ij}^x \\ \infty & : X_{ij} > C_{ij}^x \end{cases} \quad (1)$$

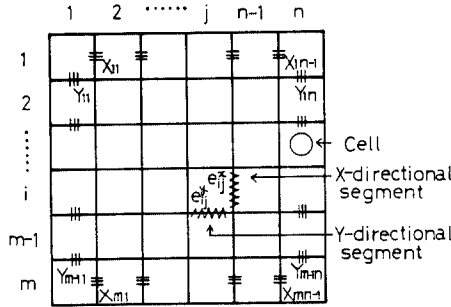


그림1. 칩의 構造
Fig. 1. Structure of a chip.

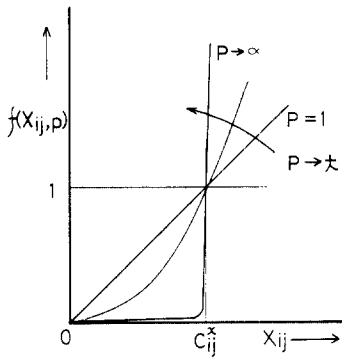


그림2. 코스트 함수 $f(X_{ij}, P)$
Fig. 2. Cost function $f(X_{ij}, P)$.

즉 세그먼트 混雜도가 1 이상(할당된 配線數가 收容 可能한 數를 초과한다.)의 경우 P를 크게 하면 코스트는 매우 높게 되며 세그먼트 混雜도가 1 이하(할당된 配線數가 收容 可能한 數 이하이다.)의 경우는 0에 수렴한다.

式(1)을 만족하는 함수로써 몇 개를 생각할 수 있지만 여기서는 계산을 간단히 할 수 있다는 관점에서 식(2)를 채용하는 것으로 한다(그림 2 참조).

$$f(X_{ij}, P) = \left(\frac{X_{ij}}{C_{ij}^x} \right)^P \quad (2)$$

코스트 함수 $f(Y_{ij}, P)$ 도 $f(X_{ij}, P)$ 와 같이 정의한다. 세그먼트 전체에 대해서 코스트 F_P 를 다음 식에 의해 정의하고 이 식을 最小化하는 것을 목적 함수로 설정한다.

$$F_P = \left(\sum_{i=1}^m \sum_{j=1}^{n-1} f(X_{ij}, P) + \sum_{i=1}^{m-1} \sum_{j=1}^n f(Y_{ij}, P) \right)^{\frac{1}{P}} = \left(\sum_{i=1}^m \sum_{j=1}^{n-1} \left(\frac{X_{ij}}{C_{ij}^x} \right)^P + \sum_{i=1}^{m-1} \sum_{j=1}^n \left(\frac{Y_{ij}}{C_{ij}^y} \right)^P \right)^{\frac{1}{P}} \quad (3)$$

本 論文에서의 문제는 P값이 주어진 상태에서 F_P 를 최소로 하는 모듈의 配置를 구하는 것이다. 이와 같이 목적함수 F_P 를 설정하면 配置의 評價를 통일적으로 해석하는 것이 可能하게 된다. 예를 들면 式(3)에 있어서 $P=1$ 및 $C_{ij}^x=C_{ij}^y=C$ 라고 하면

$$F_1 = \frac{1}{C} \left(\sum_{i=1}^m \sum_{j=1}^{n-1} X_{ij} + \sum_{i=1}^{m-1} \sum_{j=1}^n Y_{ij} \right) \quad (4)$$

으로 되고 式(4)는 總配線長을 표시하고 있다. 또, $P=\infty$ 라고 하면

$$F_\infty = \max_{ij} \left(\left(\frac{X_{ij}}{C_{ij}^x} \right), \left(\frac{Y_{ij}}{C_{ij}^y} \right) \right) \quad (5)$$

으로 되고 式(5)는 最大 세그먼트 混雜도를 표시하고 있다.

III. 配置 評價의 計算法

엄밀한 의미에서의 配置의 評價는 이상적인 配線 經路를 선택했을 때에 100% 結線 可能性 여부에 의해 결정되어야 할 것이나 문제가 너무 복잡해지기 때문에 근사적인 配置의 評價 基準을 導入해서 그것에 의하여 配置 問題를 配線 問題로부터 분리해서 論하는 것이 현실적이다.*

最小化 해야 할 配置의 評價 함수로써 지금까지 總 配線長, 通過線數가 이용되고 있었지만 이들은 매우 大局의인 評價이며 配線의 局所的인 混雜도는 評價할 수 없다.

*한 개의 配線 프로그램을 실행시켜 配線率의 大小에 따라 配置의 좋고 나쁨을 評價해도 결코 公正한 評價로 된다고 말할 수는 없다.

本論文에서는 局所的인 混雜度를 評價 可能하며 더욱 실제의 配線 처리에 가까운 하나의 評價 方法을 제안하고 그 계산법에 관하여 記述한다.

1. 多點間 配線의 취급

일반적으로 信號線은 한개당 S개(S ≥ 2)의 모듈을 연결하는 것으로 된다. S ≥ 3개의 信號線(多點間 配線)에 대해서는 문헌^[3]에 제안되어 있는 完全重量 그래프(weighted complete graph)의 개념을 사용하여 근사적으로 2점간의 配線으로 分解하기로 한다. 즉 2/S의 重量을 갖는 2점간 配線이 S개 모듈의 모든 페어(pair) 間에 존재하는 것으로 가정한다.

2. 配線 徑路의 決定法

각 세그먼트당 몇 개의 配線이 통과할까를 계산한다. 실제의 配線 處理 結果에 가까운 結果가 용이하게 계산할 수 있다는 이유에 의해 이하의 가정을 설정해서 配線 徑路를 결정한다.

[가정 1] 信號線의 配線 徑路는 다른 信號線의 配線 徑路에는 영향을 받지 않고 獨立的으로 결정한다.

[가정 2] 信號線의 配線 徑路는 접속된 모듈을 포함하는 最小 사각형의 내부에 있다.

[가정 3] 配線 徑路의 방향은 x축 또는 y축에 平行하고 配線 徑路의 굵은수는 2번 이하로 하고 3번 이상의 굵은수는 고려하지 않는다.

[가정 4] 한 개의 信號線에 複數개의 配線 徑路가 존재하는 경우, 이들의 配線 徑路는 같은 확률로 일어난다.

信號 相互間의 영향을 고려해서 配線 徑路를 구하는 것은 복잡한 처리를 필요로 하고 계산 시간이 걸리는 이유 때문에 가정 1을 설정했다. 또 실제의 配線을 행한 결과로는 가정 2 및 3을 만족하는 패턴이 거의 대부분을 차지하고 있다. 가정 4는 가정 2, 3을 만족하는 配線 徑路가 複數개 있어도 그間에서는 특별히 우선 순위를 부여하지 않는다는 配線 戰略上的의 가정이다. 만약 特定 配線 徑路를 우선 시키고 싶다면 그 徑路를 통과하는 配線의 확률을 높임으로써 실현 가능하다.

우선, 配線 徑路의 線數를 가정 2, 3下에서 2개의 모듈간을 연결할 경우에 대해서 구한다. 모듈을 A, B 라 하고 할당된 셀의 장소들 (x_A, y_A), (x_B, y_B) 로

한다. 장소의 組合은 4개의 경우가 있지만 x_A ≤ x_B, y_A ≤ y_B의 경우만 고려한다. 다른 3개의 경우도 똑같이 취급 가능하다.

A와 B를 연결한 配線 徑路의 線數를 l_{AB}로 한다면

$$l_{AB} = \begin{cases} 1 & : x_A = x_B \text{ 또는 } y_A = y_B \\ (x_B - x_A) + (y_B - y_A) & : x_A \neq x_B \text{ 동시에 } y_A \neq y_B \end{cases} \quad (6)$$

로써 주어진다.

세그먼트 E_{i,j}를 통과하는 配線의 線數를 l(E_{i,j})라 한다면 이하의 式으로 구해진다.

1) x_A = x_B일 때

$$l(E_{i,j}) = 0 \quad (7)$$

2) x_A ≠ x_B 동시에 y_A = y_B일 때

$$l(E_{i,j}) = \begin{cases} 1 & : x_A \leq i \leq x_B - 1 \text{ 동시에 } j = y_A \\ 0 & : \text{上記 以外의 경우} \end{cases} \quad (8)$$

3) x_A ≠ x_B 동시에 y_A ≠ y_B일 때

$$l(E_{i,j}) = \begin{cases} x_B - i & : x_A \leq i \leq x_B - 1 \text{ 동시에 } j = y_A \\ i - x_A + 1 & : x_A \leq i \leq x_B - 1 \text{ 동시에 } j = y_B \\ 1 & : x_A \leq i \leq x_B - 1 \text{ 동시에 } y_A < j < y_B \\ 0 & : \text{上記 以外의 경우} \end{cases} \quad (9)$$

E_{i,j}를 통과하는 配線의 線數 l(E_{i,j})도 같은 방법으로 구한다. 따라서 모듈 A와 B를 연결하는 한개의 信號線이 세그먼트 E_{i,j}, E_{i',j'}에 할당 되어진 線數를 X_{i,j}, Y_{i',j'}라 한다면 가정 4에 의해

$$\left. \begin{aligned} X_{i,j} &= l(E_{i,j}) / l_{AB} \\ Y_{i',j'} &= l(E_{i',j'}) / l_{AB} \end{aligned} \right\} \quad (10)$$

으로 구해질 수 있다.

그림 3은 모듈 A와 B를 연결하는 信號線 徑路를 열거하고 있고 l_{AB} = 5로 된다.

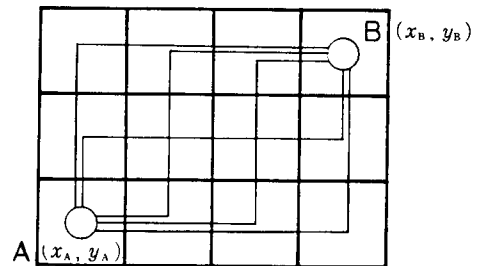


그림3. 모듈 A와 B를 연결하는 配線 徑路
Fig.3. Routing paths between module A and B.

	$\frac{1}{5}$	$\frac{2}{5}$	$\frac{3}{5}$ ○ B
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{2}{5}$
$\frac{2}{5}$ ○ A	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$

그림4. 모듈 A와 B를 연결하는 신호선이 각 세그먼트에 할당되는 배선 수

Fig. 4. Number of signal nets assigned to each segment.

그림 4는 모듈 A와 B를 연결하는 한개의 신호선이 각 세그먼트에 提供하는 配線數이다. 따라서 몇개의 신호선이 주어 졌을 때 만약 그 中에서 多點間 配線이 있다면 III. 1.의 방법에 의해 2點間 配線으로 분할한다.

다음에 각각의 신호선에 대해서 각 세그먼트에 할당되어진 配線數를 式(6)~(10)으로 계산하고 세그먼트당 할당되어진 配線數의 總合을 구한다. 이값이 III에서 記述한 X_{ij}, Y_{ij} 에 해당하고 式(3)이 계산 가능하게 된다.

IV. 配置 알고리즘

本 論文에서 취급하고 있는 문제는 규모가 크게 되면 最適解를 구하는 것은 실용상 불가능하다는 의미에서 어려운 문제의 크래스에 속한다고 생각할 수 있기 때문에 능률좋은 휴리스틱 알고리즘을 導入하지 않을 수 없다. *

여기서는 $p=1$ (總配線長의 最小化)의 경우에 有效하였던 重心을 이용한 알고리즘^[6]을 확장해서 임의의 P의 경우에 적용한다. 알고리즘을 記述하기 전에 모듈M의 重心 및 그의 ϵ -근처에 대해 정의한다. 모듈M만을 이동하여 他셀상에 M을 겹치게 한 가상적인 상태에 있어 總配線長을 最小로 하는 셀의 위치를 M의 重心이라 부른다. 또 總配線長을 最小로 하는 것으로 부터 작은 順으로 ϵ 개를 取出한 셀의 위치를 M의

重心의 ϵ -근처라 부른다.

명백히 重心의 x 성분과 y 성분과는 독립적으로 구할 수 있으므로 여기서는 x 성분만을 생각한다. 모듈 M의 x 좌표값을 M_x 로 하고 좌표값 $i_x (i=1,2,\dots, n)$ 에 놓인 모듈과 모듈 M과의 사이에 연결되었던 신호선의 線數를 L_{ix} 로 한다. 이때 多點間 配線에 의한 신호선의 重量을 고려하고 있다.

문제는

$$\sum_{i=1}^n L_{ix} |M_x - i_x| \rightarrow \min \quad (11)$$

이 되는값 M_x 를 구하는 것이다. 이 문제는 이하의 정리를 이용해서 풀 수 있다.

[정리 1] q_x 가 重心이기 위한 필요충분조건은

$$\sum_{i=1}^{q_x-1} L_{ix} \leq \frac{N}{2} \leq \sum_{i=x+1}^{q_x} L_{ix}$$

가 성립하는 것이다.

단, $N = \sum_{i=1}^n L_{ix}$

[증 명] 부록에서 記述한다.

配置 알고리즘은 直接 構成法과 逐次 改良法으로 이루어진다. 直接 構成法에서는 빠르고 좋은 初期解를 구하는 것이다.

여기서는 신호線數에 착안하여 모듈을 順次的으로 선택해서 局所的으로 제일 좋은 셀의 위치에 놓는 것으로 하였다. 逐次 改良法에서는 종래의 FDP法^[3]과 FDR法^[3]을 확장해서 새로운 방법을 제안한다. 여기서 제안한 방법은 重心에 있는 모듈에 交換해야 할 모듈의 數를 限定함으로써 계산의 고속화를 도모하고 있다.

1. 直接 構成法

未配置 모듈 각각에 대해서 (既配置 모듈에 연결되었던 신호선의 線數) - (未配置 모듈에 연결되었던 신호선의 線數)를 구한다. 이값이 最大로 된 모듈을 선택한다. 선택된 모듈을 칩상의 비어 있는 셀내에서 목적 함수 F_p 를 最小로 하는 셀상에 모듈을 두고 그 장소에 모듈의 위치를 고정한다.

2. 逐次 改良法

어떤 可能解를 γ 로 한다. λ 개 이하의 要素의 交換에 의해 γ 로 부터 도달 가능한 解의 집합중 어떤 요

* $P=1$ 의 경우는 2차 計劃 問題라고 일컬으며 NP-完全임이 증명되어 있다.^[4]

소 보다도 γ 의 편이 좋다면 γ 를 λ -最適이라 부른다.

λ -最適 解는 λ 의 값이 크면 클수록 좋은 解가 얻어 지지만 계산 시간은 증대되어 간다. λ -最適을 保障하기 위해서는 λ 개 이하의 모든 要素의 組合을 생각할 필요가 있지만 실제로는 限定된 特定の 要素를 조사함으로써 保證되는 경우가 많다. 여기서는 特定 要素의 선택 기준으로써 4.에서 記述한 重心의 근처를 선택하여, 調査하는 要素數를 비약적으로 감소 시킨다. 이렇게 하므로써 큰값의 λ 에 대해 알고리즘이 적용 가능하다. 단, 조사하는 要素를 限定하고 있기 때문에 解는 반드시 λ -最適이 된다는 保證은 없다.

그림 5에 따라서 알고리즘의 설명을 한다. 交換 操作의 概念을 나타내기 위해 임의의 탐색트리를 導入했으며 그림의 탐색트리에서 노드는 모듈을, 에지(edge)는 交換 操作을 나타낸다.

탐색트리의 루트A는 交換 操作을 시작하는 모듈을 나타내며 交換 開始 모듈이라 부른다. A와 다른 노드를 연결하는 徑路에 交換 操作列을 나타내고 있다.

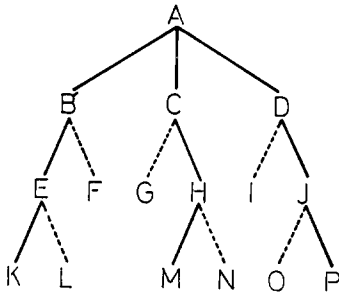


그림5. 탐색트리
Fig. 5. Search tree.

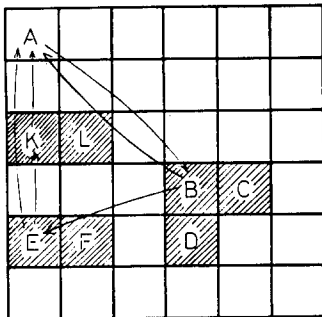


그림6. 모듈의 交換
Fig. 6. Interchange of modules.

예를 들면 $A \rightarrow B \rightarrow E$ 로 이어지는 徑路는 A, B, E를 각각 B, E, A가 있었던 위치에 삽입한다고 하는 交換 操作을 나타낸다. λ 의 값은 이와 같은 交換에 의해 이동되는 모듈의 數를 나타낸다.

탐색트리는 以下와 같이 탐색된다. (그림 6 참조)

1) $\lambda = 2$

모듈 A와 A의 重心의 ϵ - 근처에 있는 모듈(B, C, D)와 1對1 交換을 시도한다. 여기서는 $\epsilon = 3$ 으로 하고 있다.

목적 함수 F_p 가 감소하면 이의 1對1 交換을 실시한다. 또 F_p 를 감소시키는 複數개의 1對1 交換이 있으면 最大로 감소를 하는 1對1 交換을 채택한다. 어떤 1對1 交換도 감소에 기여하지 않으면 다음을 행한다.

2) $\lambda = 3$

모듈 A를 B의 셀의 장소에 두고 B의 重心의 ϵ - 근처에 있는 모듈(E, F)를 구한다. 여기서는 $\epsilon = 2$ 로 하고 있다.

B를 E 또는 F에 놓고 E 또는 F를 A가 最小 값에 놓아 두었던 셀의 장소에 놓는다. 즉 3개의 모듈의 交換을 시도한다. 이 交換 操作으로 목적 함수 F_p 가 감소하면 交換을 실행한다.

F_p 가 감소하지 않으면 이들의 交換 操作으로 F_p 가 가장 작은 쪽을 후에 交換 操作을 위해 남기고 다른 것은 버린다. 여기서는 $A \rightarrow B \rightarrow E$ 를 남기고 $A \rightarrow B \rightarrow F$ 는 버리고 있다. 탐색트리의 에지의 실선 부분은 탐색이 그후에도 계속됨을 사선 부분은 탐색이 종료하는 것을 의미한다. 루트 A를 제외하고는 각 노드의 아래에는 단지 한 개의 실선 에지가 있다. A를 다른 重心의 ϵ - 근처 모듈 C, D에 대해서도 똑같은 것을 실행한다. 트리의 탐색은 H또는 J 以下에 대해서도 수행한다.

3) $\lambda \leq \lambda^*$ 의 체크

$\lambda \leq 4$ 의 경우 즉 E, H, J 以下에 대해서도 같은 操作이 행해진다. λ 의 上限(λ^* 로 한다)을 미리 설정해 두어 노드 A로 부터의 徑路의 길이 λ 가 λ^* 가 될 때까지 계속하여 $\lambda > \lambda^*$ 가 되면 트리의 탐색은 종료된다. 또한 각 모듈은 점차로 交換 開始 모듈이 되며 어떤 모듈을 선택해서도 목적 함수 F_p 가 감소하지 않으면 操作은 종료한다. ϵ 의 값도 외부로부터 설정한다.

ϵ 및 λ^* 의 값은 크면 클수록 좋은 解를 얻을 수 있지만 계산 시간은 그만큼 많이 걸린다. 계산 시간과

解의 良好性 두 가지 관점으로 부터 最適의 ϵ 값은 4, λ^* 의 값도 4인 것이 실험적으로 확인되고 있다.^[6]

上記 방법으로 $\lambda^*=2$ 로 한 것은 FDPR法이며 $\epsilon=1$, $\lambda^*=\infty$ 로 한 것은 FDR法으로 되어 있다. FDPR法은 操作을 단지 2개의 모듈의 交換으로 限定하고 있기 때문에 도달되는 解가 나쁘다. 또 FDR法은 操作 도중에서 비어 있는 셀이 발견되지 않는다고 하는 缺點이 각각 있었으나 本 方法은 이들의 缺點을 극복하고 있다.^[6]

V. 프로그램 實驗의 結果

IV章에서 記述한 알고리즘을 프로그램化하여 실험을 행했다. 프로그램은 포트란으로 작성하고 사용한 컴퓨터는 유니백90/30이다. 사용한 데이터는 표 1에 보는 바와 같이 모듈의 總數는 151개, 信號線數는 419, 보드 上의 스토트數는 11×15개로서 $C_{11}^x=C_{15}^y=10$ 으로 하였다.

표 1. 실험용 데이터

Table 1. Test data.

내부 모듈 數	136
외부 모듈 數	15
전체 모듈 數	151
신호선 數	419
1 모듈당 평균 신호선 數	5.98
1 신호선에 포함된 최대 모듈 數	9
1 신호선당 평균 모듈 數	2.35
칩의 크기(셀의數)	11×15

1. 總配線長과 最大 세그먼트 混雜度

混雜度 파라미터 p 값을 하나 주어 $\lambda^*=4$, $\epsilon=4$ 로 해서 IV章에서 記述한 알고리즘을 적용했다. 그 結果로써 모듈의 配置가 결정되고 그 配置에 대해서 總配線長과 最大세그먼트 混雜度를 계산하였다. P=1,2,4,8의 각각에 대해서 얻었다. 總配線長과 중 및 횡 방향의 最大 세그먼트 混雜度の 값이 그림 7과 같다.

2. 세그먼트 混雜度の 分布

混雜度 파라미터 P의 값을 변화시켰을때 세그먼트 混雜度 값의 범위의 분포(Occurence frequencies)를 그림 8에 나타냈다.

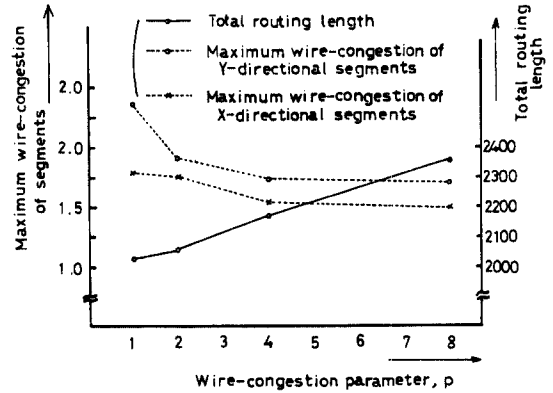


그림 7. P에 의한 總配線長 및 最大 세그먼트 混雜度の 變化

Fig. 7. The relation between the maximum wire-congestion of segments and the total routing length.

컴퓨터에 의한 계산결과, 예를 들면 P=4에서 混雜도가 1.0~1.2사이의 값을 갖는 세그먼트 數는 42개 임을 보여 주고 있다.

그림으로 부터 어떤 P의 값에 대해서도 半数 程度의 세그먼트에 있어서 混雜도가 1을 넘어 서고 있고 $C_{11}^x=C_{15}^y=10$ 으로 했을 때는 100% 結線이 곤란하다는 것이 예상된다. 즉 C_{11}^x , C_{15}^y 의 값을 15前後(이 경우 混雜度の 分布는 그림 8의 횡축의 수치를 1.5로 나눈 것이 된다.)로 하면 1.0 이상의 混雜度를 갖는 세그먼트가 10% 이하가 되며 좋은 配線 알고리즘에 의해 100%의 結線이 기대될 수 있다. P=1,2의 경우에는 세그먼트 混雜도가 平均 分布를 하고 있지만 最大 세그먼트 混雜度를 最小로 하는 의미에 있어서는 좋은 配置를 제공하지 않고 있는 것을 알 수 있다.

3. 計算時間

總配線長을 最小로 하는 프로그램^[6]과 비교해 本 論文中에서 記述한 프로그램은 약 5배의 계산 시간이 걸렸다. 그러나 계산 시간을 단축하기 위해서 總配線長을 最小化하는 프로그램을 최초로 수행시켜 그 뒤에 本 論文中에서 記述한 프로그램을 수행시키면 최초부터 최후까지 本 論文中에서 記述한 프로그램을 수행시킨 것과 비교하면 계산 시간은 1/2~1/3이 된다.

VI. 結 論

LSI의 layout 設計에 있어서 중요한 문제인 모듈의

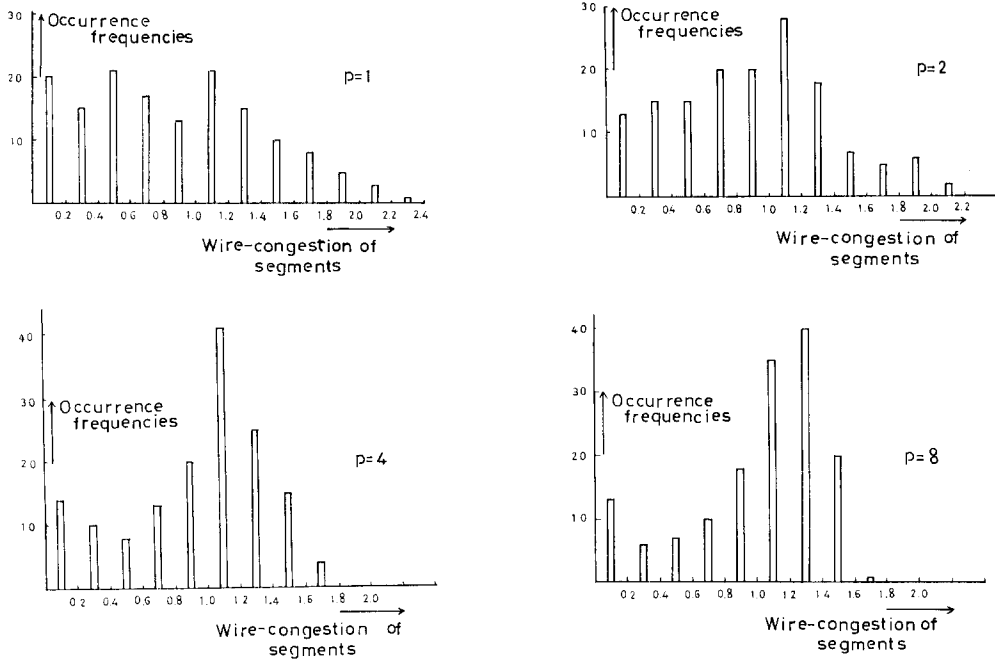


그림8. 세그먼트 混雜度の 發生 回数
 Fig. 8. The relation between wire-congestion of segment and occurrence frequency.

2 차원 配置 問題를 취급하였다. 종래에는 總配線長 또는 通過線數라고 하는 大局的인 목적 함수로써 配置 結果를 평가하여 왔으나 本 論文에서는 局所的인 混雜度가 평가 가능한 목적 함수를 제안했다. 또 종래에는 假想 配線 徑路를 고려하고 있지 않았기 때문에 실용적인 견지에서 응용하기 어려운 점이 있었으나 本 論文에서는 실제의 배선 처리에 가장 가까우며 단시간에 계산이 가능한 방법을 제안하여 混雜度를 구했기 때문에 실용적으로 매우 유용하다고 볼 수 있다.

局所的인 混雜度를 解消하는 능률좋은 휴리스틱 알고리즘을 제안하여 最大 세그먼트 混雜도와 總配線長間을 제어하는 파라미터 P를 導入하여 配置 結果를 얻었다.

프로그램 結果에 의하면 P를 크게 함으로써 最大 混雜度가 解消 가능했으나 그것에 따라 總配線長이 증가하여 갔다. $2 \leq P \leq 4$ 에서 總配線長도 크게 되지 않고 동시에 最大 세그먼트 混雜度도 작은 것이 判明되었다. 이후로는 本 論文에서 제안한 配線 混雜度の 타당성 및 가장 最適의 P값의 研究를 계속하지 않으면 안될 것이다.

附 錄

[정리 1]의 證明

그림A에 나타낸 것 같이 직선상에 모듈이 늘어서 있고 i_x 의 장소에 있는 모듈과 모듈 M과는 L_{i_x} 개의 信號線이 있다.

이때 장소 k_x 로부터 모든 점의 거리의 總和를 $g(k_x)$ 로 나타내면

$$g(k_x) = \sum_{i_x=1}^{k_x-1} (k_x - i_x)L_{i_x} + \sum_{i_x=1}^{n_x-k_x} i_x L_{k_x+i_x}$$

로 된다.

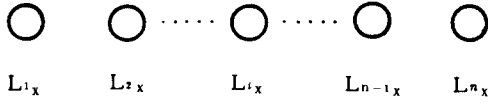
점 q_x 가 重心이라고 하는 것은 모든 K_x 에 대해서 $g(q_x) \leq g(k_x)$ 가 성립한다.

그러므로

$$\begin{aligned} g(k_x+1) - g(k_x) &= \sum_{i_x=1}^{k_x} L_{i_x} - \sum_{i_x=1}^{n_x-k_x} L_{k_x+i_x} \\ &= 2 \sum_{i_x=1}^{k_x} L_{i_x} - N \end{aligned}$$

$$g(k_x - 1) - g(k_x) = N - 2 \sum_{i_x=1}^{k_x-1} L_{i_x}$$

가 성립한다.



그림A. 모듈의 列

Fig. A. A linear placement of modules.

1) 필요 조건

$$\sum_{i_x=1}^{q_x-1} L_{i_x} \leq \frac{N}{2} \leq \sum_{i_x=1}^{q_x} L_{i_x} \text{ 가 성립한다.}$$

$$g(q_x) \leq g(q_x + 1)$$

$$g(q_x) \geq g(q_x - 1) \text{ 가 성립하고 더욱이 } k_x \geq q_x + 1$$

에 대해서

$$\sum_{i_x=1}^{q_x} L_{i_x} \leq \sum_{i_x=1}^{k_x} L_{i_x} \text{ 가 성립하고}$$

$k_x \leq q_x - 1$ 에 대해서

$$\sum_{i_x=1}^{k_x} L_{i_x} \leq \sum_{i_x=1}^{q_x} L_{i_x} \text{ 가 성립함에 의해}$$

$$g(q_x) \leq g(k_x) \text{ 가 성립한다.}$$

단, 여기서 $k_x = 1, 2, \dots, n_x$

2) 충분 조건

$$g(q_x) \leq g(k_x) \text{ 가 성립한다.}$$

단, 여기서 $k_x = 1, 2, \dots, n_x$

$$g(k_x + 1) - g(q_x) \geq g(k_x + 1) - g(k_x) \text{ 로 부터}$$

$$2 \sum_{i_x=1}^{k_x} L_{i_x} - N \geq 0$$

또는 $g(k_x - 1) - g(q_x) \geq g(k_x - 1) - g(k_x)$ 로 부터

$$N - 2 \sum_{i_x=1}^{k_x-1} L_{i_x} \geq 0 \text{ 이 성립한다.}$$

(증명 끝)

參 考 文 獻

- [1] Hanan, M. and Kurtzberg, J. M. , "Placement techniques". Chapt. 5 in Design Automation of Digital systems and Techniques, 1, Breuer, M. A. ed. , pp. 213~282, New Jersey, Prentice Hall 1972.
- [2] Breuer, M. A. , "A class of mincut placement algorithms", Proc. 14th D. A. Conf. , pp. 284 - 290. 1977.
- [3] Hanan, M. , Wolff, P. K. and Anguli, B. J. , "Some experimental result on placement techniques", Proc. 13' th D. A. Conf. , pp. 214 - 224, 1976.
- [4] Sahni, S. and Gonzales, T. , "P-complete approximation problem", J. Assoc. Comput. Mach. , 23, 3, pp. 555 - 565 July 1979.
- [5] Lin, S. and Kernighan, B. , "An effective algorithm for travelling-salesman problem", Oper. Res. , 11, pp. 498 - 516, 1973.
- [6] Goto, S. , "An efficient algorithm for two Dimensional placement problem in electrical circuit Lay-out", Proc. 1979 ISCAS, pp. 850 - 853. 1979.
- [7] Stevens, J. E. , Fast Heuristic Techniques for Placing and Wiring [Printed Circuit Boards. Ph. D. Thesis, Com. Scien. Univ. of Illinois, 1972.