

シーケンスペアに基づくフロアプランニングの報告

村上 耕平

1 はじめに

超大規模集積回路(VLSI)のレイアウト設計における設計行程の一つに,モジュールの集合を与えられた制約の下で,チップ面積が最小となるように配置するフロアプランニングがある.フロアプランニングは,チップの面積最小化とパフォーマンス向上のために重要な工程であるが,最適な配置を求めることは容易でない.そのため,一般にシミュレーテッドアニーリング(Simulated Annealing:SA)や遺伝的アルゴリズム(Genetic Algorithm:GA)などの進化的計算アルゴリズムを用いて近似解が求められる.これらのアルゴリズムにおける解の表現方法の一つとして,シーケンスペア(Sequence-Pair)が利用される.シーケンスペアを用いると,与えられた矩形の集合に対し,有限な解空間でフロアプランを求めることが可能となる.

そこで本報告では,シーケンスペアによる解の表現の方法を報告する.その後,シーケンスペアを解構造とした SA の実装に向けての課題について示す.

2 フロアプランニング

本報告では,より実問題へ近い対象として複数の矩形のモジュールを最小面積で配置するフロアプランニングを対象問題として用いた.具体的には,複数の矩形のモジュールを配置する際に,配置空間内に各モジュールを重なりなく配置し,必要とされるチップの面積の最小化を目的とする.

2.1 データ構造表現におけるシーケンスペア

フロアプランニング問題において最も重要な問題の一つは解の表現方法,すなわちデータ構造の表現である.フロアプランニング問題では,このデータ構造の表現によりモジュールの配置方法が決まるだけでなく,実際の探索空間,最終的に得られる解の精度といった問題の本質的な部分までが決定する.従来は配置領域を水平,垂直の線分で再起的に分割して得られるスライシング構造を2分木で表したスライシング木が用いられてきたが,非スライシング構造も表現可能なデータ構造としてシーケンスペアが提案された. Fig.1 は非スライシング構造とスライシング構造の例である. LSI のフロアプランニングにおいてシーケンスペアを用いることで,非スライシング構造の有限な解空間においてプランニングを行うことが可能となった.シーケンスペアの手順について述べる.

2.2 シーケンスペアのコンセプト

シーケンスペアは,配置の対象となるモジュール名からなる2つの系列(Γ_-, Γ_+)の並びに基づいて矩形配置

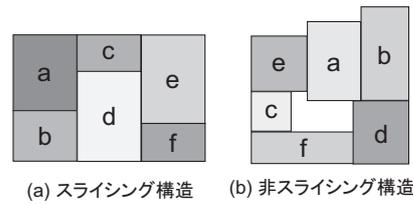


Fig.1 スライシング構造と非スライシング構造 (出典:自作)

を表現する方法である.シーケンスペアでは(Γ_-, Γ_+)の系列に基づいて,任意の2つのモジュールの相対的な位置関係が決まる. Fig.2 にシーケンスペアの概要を示す.

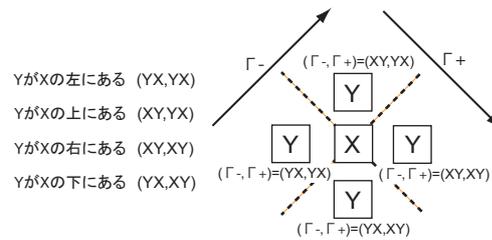


Fig.2 シーケンスペアの概要 (出典:参考文献 [2])

Fig.2 は任意の2つのモジュール X, Y に関する系列(Γ_-, Γ_+)とそれに対応するモジュール X, Y の位置関係を表している.例えば,系列(Γ_-, Γ_+)=(XY, XY)は系列 Γ_- および系列 Γ_+ において X が Y よりも左に位置し,このような場合にはモジュール Y は X の右に位置することを意味する.また,シーケンスペアでは任意の系列(Γ_-, Γ_+)に対してこれらの位置関係を満たす矩形配置は必ず存在する. Fig.3 に,シーケンスペアによる解の表現の例を示す.

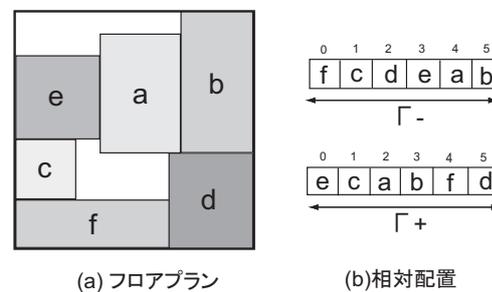


Fig.3 シーケンスペアによる表現 (出典:参考文献 [1][2])

Fig.3(b) の系列 Γ_- と系列 Γ_+ において,未配置のモジュールをチップに配置する際は,すべての配置済みのモジュールとの相対位置を求めることで,新たに配置するモジュールの位置を決定する.この操作を繰り返すことで, Fig.3(a) のフロアプランを求めることが出来る.フロアプランが求まるとチップに必要な横幅と縦幅が決定し,チップ面積が求めることが可能となる.新たにモジ

ルールを配置する際の例を Fig.4 に示す .

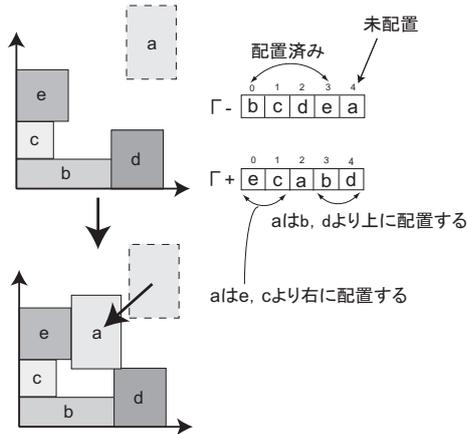


Fig.4 モジュール配置の例 (出典：自作)

2.3 レフトダウンパッキング

本報告では、シーケンスペアを用いてモジュール間の相対位置をフロアプランを反映させる際の、パッキング手法としてレフトダウンパッキングを用いた。⁴⁾ レフトダウンパッキングでは、系列 Γ_- の系列が並ぶ順番に、可能な限り左下にモジュールを配置する。すると Γ_- の要素は先に配置済みモジュールが並び、その後未配置のモジュールが並び、

配置済みモジュールを X 、新たに配置するモジュールを Y とし、 Y と X の Γ_+ 中で出てくる順序をみる。 Γ_+ の中で Y よりも左側に出てくる配置済みモジュール X と Y が $(\Gamma_-, \Gamma_+) = (XY, XY)$ を満たせば、 Y はこの X よりも右に配置される。また Γ_+ 中で Y よりも右側に出てくる配置済みモジュール X と Y が $(\Gamma_-, \Gamma_+) = (XY, YX)$ を満たせば、 Y はこの X よりも上に配置される。この操作を繰り返すことで、系列 (Γ_-, Γ_+) は対応するフロアプランとなる。

3 今後の課題

前節で説明したシーケンスペアを用いて解を表現し SA を用いることでチップの配置の最適化が今後の課題である。これより有効な近傍について考察する必要がある。また、データ構造の表現として、シーケンスペアの他に木構造を用いたものなどが提案されており、他の表現に関しても調査する予定である。本節では、今後設計を行う SA の実装に関して述べる。

3.1 近傍の設定

フロアプランニング問題で解探索を行う際には、モジュールの相対位置と、矩形のモジュールが縦向きか横向きかを変化させる必要がある。このため、モジュールの相対位置に関する近傍と、モジュールの向きに関する近傍を考える。

モジュールの相対位置を変化させる近傍として、現在 2 通りの種類の近傍を考えている。第 1 の近傍は、 Γ_- と Γ_+ のどちらかの系列内の要素をランダムに 2 つ選択し、その順序を入れ替えるものである。どちらの系列で入

れ替えを行うかは確率 0.5 で決定する。また、入れ替えを行う 2 つのモジュールのうち 1 つのモジュールの向きを確率 0.01 で変化させることにより、モジュールの向きも変化させる。

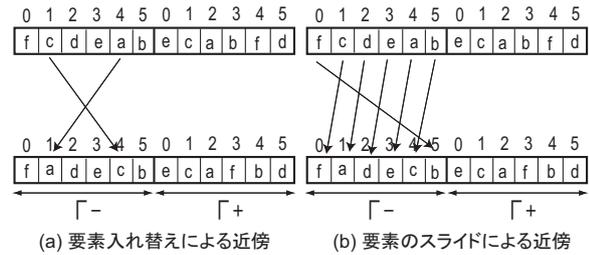


Fig.5 近傍の適用 (出典：自作)

第 2 の近傍は、 Γ_- または Γ_+ の系列内の全ての要素を左、または右に 1 つスライドさせるものである。第 1 の近傍と同様にどちらの系列でスライドさせるかは確率 0.5 で決定される。また、左右どちらにスライドさせるかは確率 0.5 で決定する。また、スライドさせるモジュールのうち 1 つのモジュールの向きを確率 0.01 で変化させることにより、モジュールの向きも変化させる。Fig.5 は、今後適応する近傍を表す。

これらの近傍を用いて、数値実験を行うとともに、文献調査を進めてフロアプランニング問題を SA に適用させる際に有効な他の近傍の調査を行う予定である。

3.2 データ構造

本報告では、シーケンスペアを用いた解の表現に関して述べたが、フロアプランニング問題を解く際のデータ構造の表現には、二分木を用いた表現や、多分木を用いたものなどが多く発表されている。³⁾ 今後は、データ構造の表現に関しても文献調査を進め、フロアプランニング問題を SA に適用させる際に有効なデータ構造の調査を行う予定である。

3.3 配線長の評価

本報告では、チップの面積を最小化する問題を対象問題とした。しかし、フロアプランニングにおいて、モジュール間の配線長も考慮にいれ、チップの面積と配線長を同時に最小化する問題も発表されている。今後、チップの面積と配線長を同時に最小化する問題にも取り組む予定である。

参考文献

- 1) 渡邊真也, 廣安知之, 三木光範. 多目的矩形パッキング問題に対する進化的手法の有効性. 電子情報通信学会論文誌. 2003
- 2) 中矢真吾, 若林真一, 小出哲士. 適応的遺伝的アルゴリズムとシーケンスペアに基づくフロアプランニング手法. システム LSI 設計技術. 1999
- 3) Yun-Chih Chang, Yao-Wen Chang, Guang-Ming Wu, and Shu-Wei Wu. B*-Tree: A New Representation for Non-Slicing FloorPlans. Cite Seer. 2000
- 4) The Logistic Institute - Asia Pacific. Rectangle Packing. http://www.tliap.nus.edu.sg/tliap/Project_BerthManagement/rectangle_packing.htm