設計自動化 62－31
(1992. 5.28)

配置問題に対する分枝限定法の階層的適用とその評価

栗島 亨　金子 一哉　戸川 望
佐藤 政生　大附 辰夫

早稲田大学 理工学部

あらまし

分枝限定法に基づいた階層的配置手法と，その計算機実験による評価について報告する．本手法は，大規模なモジュール配置問題をクラスタリング手法を用いて階層化し，各階層レベルで分枝限定法を適用するものである．まず，階層化する際のクラスタ数とクラスタサイズの上限の設定法について検討する．次に，各階層レベルの最終処理として multi-partitioningによる反復改良を適用する．実験の結果，クラスタ数，およびクラスタサイズを12以下に設定し，問題を2階層に分割することによって，モジュール数80程度の問題における近似的最適解が比較的短時間で得られた．

Hierarchical Branch-and-Bound Method for Module Placement Problems and its Estimation

Toru Awashima, Kazuya Kaneko, Nozomu Togawa,
Masao Sato, and Tatsuo Ohtsuki

School of Science and Engineering, Waseda University
3-4-1 Ohkubo, Shinjuku-ku, Tokyo 169, Japan

Abstruct

Experimental estimation of the hierarchical branch-and-bound method for module placement problems is presented as well as a description of the algorithm. This method consists of hierarchical partitioning based on clustering of modules and branch-and-bound placement at each hiearachical level. The proposed method yields a suboptimal solotion of 80 module problem within a reasonable time domain where both the number of clusters and size of clusters are controlled not to exceed 12.

1.まえがき

電子回路のレイアウト設計において，複数のモジュールをスロットに最適配置するというスロット配置問題は，最も基本的かつ重要な問題の1つである．この問題は，適当な仮定のもとで2次割当て問題として定式化できる[1],[2]．2次割当て問題に対しては，現在まで数多くの解法が提案されてきた[1]~[13]．しかし，最適解の導出が可能なものとしては，分枝限定法に基づいた手法しか知られていない[4]~[9]．

分枝限定法は，単なる列挙法に比べれば非常に効率の高い手法だが，多項式時間内の処理を保証するものではない．2次割当て問題はNP完全問題であり[14]，分枝限定法によって許容時間内に解き得る問題規模は，要素数15程度が事実上の限界であるといわれてきた．したがって，分枝限定法に基づいた2次割当て手法は，主に近似解法としての側面から評価されることが多かった[12],[13]．

我々は計算機実験により，配置問題に対する分枝限定法を近似解法としてだけでなく，最適解を求める厳密解法としても詳細に評価した[16],[17]．実験の結果，近似解法については，比較的良い初期解を与え，解の評価値の上界を考慮にいれた場合，発見的探索関数に基づいた深さm探索と許容誤差法を用いることにより，20モジュール程度の規模の問題ならば，比較的短時間で良好な解が得られることがわかった．一方，厳密解法については，許容時間内に最適解を求めることができたのは12モジュール程度の問題までであった．

本稿では，より大規模な配置問題を扱うため，モジュールのクラスタリングによって問題を階層化し，階層の各レベルで分枝限定法を適用する手法を提案する．次に計算機実験によって手法を評価する．具体的には各階層レベルにおけるクラスタ数と処理時間，解の評価値との関係，確率的手法であるシミュレーティド・アニーリング法との比較結果等について報告する．

2.配置モデル

本稿では，スロットモデルに基づいた配置問題を扱う．スロットモデルとは，格子状にn個のスロットが並んでおり，対応するn個のモジュールはこれらのスロット上にのみ配置可能であるようなモデルである（図1参照）．モジュール間に存在する接続要求をネットと呼ぶ．目的関数は仮想総配線長（マンハッタン長）の最小化である．

配置問題は，全てのネットが2点間ネットであるという仮定のもとでは，2次割当て問題として定式化できる．また，このような仮定ができないとしても，1組のモジュール対が共通に属するネットの重みの総和をそのモジュール対を接続する2点間ネットの重みとみ

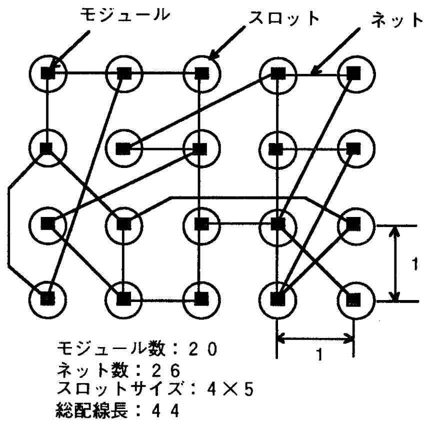

図1　スロット配置モデル

なして，2次割当て問題に変換することが可能である[1],[2]．ただし，この変換によって得られた2次割当て問題の最適解が，原配置問題の最適解となる保証はないことに注意が必要である．以下では，配置問題を2次割当て問題と考えて議論を進める．

2次割当て問題

コスト行列を$C=[c_{ij}]$，距離行列を$D=[d_{kl}]$，iが割り当てられた場所を$p(i)$と表すとすると，2次割当て問題は以下のように定式化できる．

$$\text{minimize } G=\sum_{i,j} c_{ij} d_{p(i)p(j)} \tag{2.1}$$

この式の配置問題における直観的意味は，n個のモジュールとn個のスロットがあって，モジュールiとjの接続の重みがc_{ij}であり，i,jが割り当てられたスロット$p(i),p(j)$間の距離が$d_{p(i)p(i)}$で表されているというものである．

次節では，式(2.1)で定義された2次割当て問題の最適解を求める分枝限定配置手法について述べる．なお，以下では，最適化問題は，目的関数の最小化を意味するものとする．また，本節で定義した2次割当て問題には制約条件が課せられていないので，いかなる解も許容解となる．

3.分枝限定配置手法

組合せ最適化問題においては，NP完全問題のように，列挙法以外に最適解を求める方法が知られていないものが多数存在する．そこで，極力無駄な探索を抑えながら解の列挙を行おうというのが分枝限定法の基本原理である．以下では，配置問題における分枝限定法（分枝限定配置手法と呼ぶ）について述べる．

3.1 基本操作

分枝限定法は文字通り分枝操作と限定操作という２つの基本操作から成り立っている．以下にその詳細を述べる．

・分枝操作

原問題P_0を複数の部分問題P_iに分解する操作である．この操作は再帰的に行われ，全ての部分問題が解かれた時点で原問題も解かれたことになる．分枝操作は，原問題の適当な変数（分枝変数と呼ぶ）の値を固定することによって行われる．配置問題の分枝変数としては，モジュールを採用した．したがって，配置問題の探索図は図２のようになる．探索図とは，実際に生成された部分問題の関係を図示したものである．

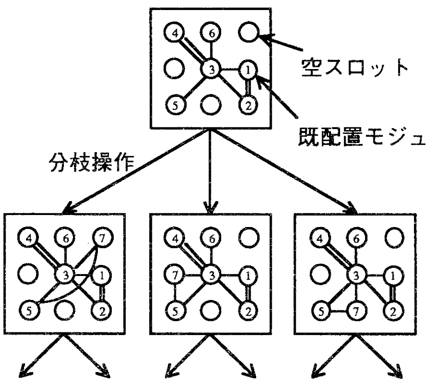

図２　配置問題の探索図

・限定操作

前述の分枝操作だけでは，原始的な列挙法と何ら変わりがない．そこで，分枝操作によって生成された部分問題の最適解が直ちに求まる場合や，それ以上分解しても原問題の最適解が得られないことが明らかな場合には，その部分問題を終端し以後列挙の考慮外とする．このような操作を限定操作という．

限定操作は部分問題P_iの下界値$g(P_i)$によって行われる．いま，現時点までの最良の解を暫定解，その評価値を暫定値Zとし，分解も終端もされていない部分問題のことを活性部分問題と呼ぶことにする（活性部分問題がなくなったときの暫定解が，原問題の最適解となることは明らかであろう）．このとき，下界値による終端条件は次式で表される．

$$g(P_i) \geqq Z \tag{3.1}$$

なぜなら，ある部分問題の下界値が暫定値を上回る場合には，それ以上その部分問題を分解しても暫定解が改良されることはないと結論できるからである．配置問題における下界値の計算には文献[4]の方法を用いる．これは，あるモジュールをスロットに置いたときに得られる評価値の自明な下界値から割当て行列を構成し，線形割当てによって原問題の下界値を得るという方法である．

3.2 上界値による暫定値の更新

本稿で扱う配置問題は制約条件を持たないため，いかなる解も許容解となる．したがって，部分問題P_iの下界値を計算する際に求められる解も１つの許容解とみなすことができる．このような許容解の評価値は部分問題P_iの上界値$u(P_i)$と解釈でき，この値によって暫定値の更新を行うことができる．これは終端条件(式(3.1))の右辺のZが比較的低く抑えられることを意味し，限定操作の効率向上につながる．

3.3 探索法

探索法とは，どのような順序で部分問題を探索していくかを定めた規則で，代表的なものに深さ優先探索法，幅優先探索法，発見的探索法，最良下界探索法がある．理論的には，処理時間の点で最良下界探索が，記憶容量の点で深さ優先探索が最適であることが知られている[19]．しかし，ここでは文献[16],[17]の議論に基づいて，発見的関数による深さm探索法を採用する．ただし，mには比較的小さな値（例えば1）を設定する．

・発見的探索関数

発見的探索関数とは，各部分問題P_iの最適値$f(P_i)$に対する推定値$h(P_i)$のことである．ここでは，部分問題P_iの上界値$u(P_i)$と下界値$g(P_i)$の中間値を適当に選ぶことにする．すなわち$0 \leqq \alpha \leqq 1$なる定数αを用いて，

$$h(P_i) = \alpha g(P_i) + (1-\alpha) u(P_i) \tag{3.2}$$

とする．

・深さm探索

正整数mを導入することによって一般化された深さ優先探索法である．k個に分解された部分問題をリストの最後尾に加える際に，これらk個の部分問題に加えリストの末尾の$m-1$個の部分問題も対象として推定値$h(P_i)$の降順に整列する．したがって，$m=1$のとき深さ優先探索となり，$m=\infty$のとき発見的探索となる．

次節では分枝限定配置手法によって実用時間内で直接解くことができないような規模の問題に対処する方法として，階層的分枝限定配置手法を提案する．

4．階層的分枝限定配置手法

本節では大規模な配置問題に対処する方法として，分枝限定法を階層的に適用する手法を提案する．文献[15]では，ビルディングブロック配置問題を対象に類似のアプローチが提案されている．一方，本手法はスロ

ット配置問題を対象としており，クラスタリングによって原配置問題を階層化し，階層の各レベルではクラスタを１つのモジュールとみなすことで分枝限定配置手法の適用を可能としたものである．配置領域は各クラスタに対応した小領域に分割される．あるレベルでの配置，すなわちクラスタの分割領域への割当てが終了したら，各クラスタを分解して下階層の配置問題を解くという操作を，順次最下階層に達するまで行う．

図３に36モジュールの配置問題をクラスタリングによって２階層の分枝限定配置手法で解く場合の概念図を示す．さらに本手法では，階層化によって局所最適解に陥ることをできるだけ回避する目的から，各階層レベルの最後の処理としてmulti-partitioningによる反復改良を行う．なお，問題が多点間ネットを含んでいた場合でも，分枝限定配置のフェーズでは前述の置き換えにより２次割当てとして扱われる．一方，反復改良フェーズにおいては，多点間ネットの配線長は発見的に計算されたスタイナー長によって評価される．

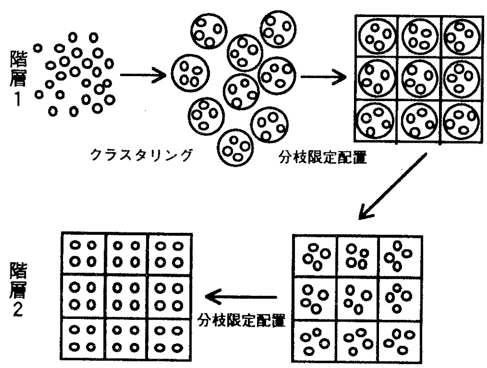

図３　階層的分枝限定配置手法

以下では，クラスタリングおよび反復改良の詳細を述べる．

4.1 クラスタリング

各階層におけるクラスタ数は，分枝限定配置手法の適用可能規模を考慮して設定する．

クラスタリングは，まず個々のモジュールを初期クラスタとみなし，各クラスタ間の接続度を評価してクラスタ同士を併合させることによって行われる．まず，全クラスタに対してそのクラスタと最も接続度が強く，併合したときに，分割領域内のスロット数によって決まるクラスタサイズの上限を超えないようなクラスタを求める．評価値が最大であるようなクラスタの組を実際に併合し，さらに処理を続ける．

クラスタ間の接続度は，２つのクラスタを１つのクラスタに併合したときの目的関数の減少度として定義される．ここで，目的関数は各ネットに関する配線コスト（＝各配線の重み×端子数に対し微係数が単調減少するような単調増加関数）の和である．よって，２点間ネットに接続するモジュール対が最も接続度が強く，多点間ネットになるほど接続度は弱くなる．

4.2 Multi-Partitioningによる反復改良[18]

問題の階層化によって局所最適解に陥ることをできるだけ避ける目的から，各階層レベルでの分枝限定配置手法の解を初期分割としてmulti-partitioningによる反復改良を行う．処理手順を以下に示す．

Step1:配線コストテーブルの作成

可能な全てのモジュール分布に対してそれらを接続するネットのスタイナー長を発見的に計算する．その結果をハッシュテーブルに格納し，モジュールの分布を与えることによって参照できるようにする．

Step2:ゲインリストの作成

全てのモジュールに対し，そのモジュールを他の分割領域に移動したときののゲイン（目的関数の減少分）を計算する．

Step3:モジュール配置の反復改良

有効な配置変更がなくなるまで以下の処理を反復する．

・モジュール選択

ゲインリストから最大のゲインを持つモジュールを選択する．ただし，配置変更によって，ある分割領域に含まれるモジュール数が分割領域内のスロット数を超えてしまう場合には，その分割領域からスロット数に余裕がある領域へのモジュール移動を優先して選択する．したがって，全体のモジュール数とスロット数が等しい場合は，ペア交換のみが行われることになる．

・ゲインリストの更新

必要に応じてゲインリストの内容を更新する．

5.計算機実験による評価

実験に用いたプログラムおよびテストデータの仕様は以下の通りである．

下界値計算法	：文献[4]の方法
探索法	：深さm探索(m=1:深さ優先探索，α=0.5)
計算機	：SUN SPARCstation2 (28.5MIPS)
言語	：C言語
テストデータ	：表１に詳細を示す

表1　テストデータ

データ	モジュール数	ネット数	スロットサイズ
nugent20	20	144	4×5
nugent30	30	293	5×6
random64	64	299	8×8
rd53-hdl	24	21	4×6
rd53	26	23	5×6
misex1	42	35	6×7
z4ml	53	49	8×8
f51m	87	79	10×10

nugent20,30は文献[12]より引用した．random64は乱数により生成した．それ以外のデータはMCNCベンチマークのネットリストから生成したものである．

便宜上，分枝限定配置のフェーズは処理時間による打ち切り法を適用した．具体的には，処理に要するCPU時間が1時間を超えたときに無条件にそのフェーズの処理を打ち切るものとした．実験によって各階層レベルにおけるクラスタ数とクラスタサイズの最大値の各値に対する処理時間，解の評価値に関する評価，およびmulti-partitioningの効果に関する評価を行った．なお，シミュレーティド・アニーリング法との比較も行った．評価値は仮想総配線長である．ただし，多点間ネットの配線長はネットに属する全てのモジュールを囲む矩形の半周囲長によって評価している．なお，全てのデータは2階層の分枝限定配置手法によって処理された．

5.1 クラスタ数と処理時間および評価値

各階層におけるクラスタリングにより生成されるクラスタ数とクラスタサイズの上限が解の評価値に及ぼす影響について検討する．ここではmulti-partitioningによる反復改良を行っていない．

表2にnugent20,30とrandom64に関する実験結果を示す．処理時間に関して，クラスタ数，クラスタサイズが共に12以下の際に非常に高速に処理が終了することがわかる．また，クラスタ数を最小に設定した場合は，評価値の良い解がコンスタントに得られるが，モジュール数15,16の問題を分枝限定配置手法で計算しなければならないために処理時間が莫大になってしまっている．

表2　クラスタ数と処理時間および評価値

データ	クラスタ数	最終解の評価値	処理時間[sec.]
nugent20	2(10)	1319	73
	4(5)	1396	1
	5(4)	1370	1
	10(2)	1366	30
nugent30	2(15)	3219	14400
	3(10)	3314	225
	5(6)	3210	9
	6(5)	3251	6
	10(3)	3275	17
	15(2)	3270	3605
random64	4(16)	1034	28850
	8(8)	1042	371
	16(4)	1034	3643

*）括弧内はクラスタサイズ

5.2 Multi-Partitioningの効果

Multi-Partitioningステップはクラスタリングにより局所的最適解に陥ることを解消する目的で行うものである．

表3にnugent20,30とrandom64についての結果を示す．

表3　multi-partitioning反復改良の効果

データ	クラスタ数	最終解の評価値	処理時間[sec.]
nugent20	2(10)	1301	71
	4(5)	1349	4
	5(4)	1309	9
	10(2)	1344	32
nugent30	2(15)	3123	14414
	3(10)	3089	176
	5(6)	3132	69
	6(5)	3142	37
	10(3)	3189	52
	15(2)	3103	3673
random64	4(16)	963	29076
	8(8)	978	517
	16(4)	980	4119

*）括弧内はクラスタサイズ

multi-partitioningの効果により，処理時間をそれほど増加させることなく，最終解の評価値が改良されている．この結果，最も高速に処理を行うことができた，クラスタ数クラスタサイズ共に12以下の場合でも比較的良い解が得られていることが解る．

5.3 階層的分枝限定法とシミュレーティド・アニーリング法との比較

各データに対して階層的分枝限定法とシミュレーティド・アニーリング法を適用した．表4に最終的に得られる解の評価値と処理時間を示す．ここで，シミュレーティド・アニーリング法は十分に乱雑な状態になるように初期温度を設定してある．温度降下係数は0.95である．

階層的分枝限定法を適用した場合は，シミュレーティド・アニーリングに比べ非常に短時間で処理を終了する．また，最終解の評価値に関してもシミュレーティド・アニーリングに匹敵する比較的良い解を得ることができる．一方，シミュレーティド・アニーリングは最終的に得られる解の評価値は良好だが，問題の規模が大きくなるにしたがい莫大な処理時間を必要としている．

表4　シミュレーティド・アニーリングと
階層的分枝限定法の比較

データ	適用手法	最終解の評価値	処理時間[sec.]
nugent20	階層分枝限定(5)	1309	9
	ＳＡ法	1287	880
nugent30	階層分枝限定(3)	3089	176
	ＳＡ法	3064	3880
random64	階層分枝限定(8)	978	352
	ＳＡ法	939	9744
rd53-hdl	階層分枝限定(4)	38	3
	ＳＡ法	37	92
rd53	階層分枝限定(5)	59	9
	ＳＡ法	54	205
misex1	階層分枝限定(6)	85	39
	ＳＡ法	83	604
z4ml	階層分枝限定(8)	144	219
	ＳＡ法	125	2046
f51m	階層分枝限定(10)	218	1587
	ＳＡ法	191	8672

＊）括弧内の数字は，階層分枝限定におけるクラスタ数である．

6．むすび

スロット配置問題に対し，モジュールのクラスタリングによって問題を階層化し，階層の各レベルで分枝限定法を適用する階層的分枝限定配置手法を提案した．階層化によって比較的大規模な問題も処理することが可能となった．また各階層レベルの最終処理としてmulti-partitioningによる反復改良を行うことで，局所最適解に陥ることを抑制している．評価実験により，本手法がシミュレーティド・アニーリングに匹敵する程度の良好な解を非常に高速に求めることを確認できた．

今後は，クラスタ数の設定に関する検討を行うことが重要である．そのために，より大規模な問題に本手法を適用し，評価する必要がある．また，各分割領域間の混雑度平均化や，各種の制約への対処を考えることも有意義であろう．

謝辞

本研究は財団法人電気通信普及財団の助成のもとに行われたものである．

文献

[1]Hanan, M. and J. M. Kurtzberg: " A Review of the Placement and Quadratic Assignment Problems," *SIAM Review*, Vol.14, pp.324-342(1972).

[2]Breuer, M. A. ed.: *Design Automation of Digital System*, Prentice-Hall(1973).

[3]Preas, B. T. and M. J. Lorenzetti ed.: *Physical Design Automation of VLSI System*, Benjamin/Cummings Publishing(1988).

[4]Gilmore, P. C.: " Optimal and Suboptimal Algorithms for the Quadratic Assignment Problem," *Journal of the Society Industrial and Applied Mathematics*, Vol.10, pp.305-313(1962).

[5]Lawler, E. L.: " The Quadratic Assignment Problem," *Management Science*, Vol.9, pp.586-599(1963).

[6]Gavett, J. W. and N. V. Plyter: "The Optimal Assignment of Facilities to Locations by Branch and Bound," *Operations Research*, Vol.14, pp.210-232(1966).

[7]Hillier, F. S. and M. M. Connors: " Quadratic Assignment Problem Algorithms and the Location of Indivisible Facilities," *Management Science*, Vol.13, pp.42-57(1966).

[8]Pierce, J. F. and W. B. Crowston: "Tree Search Algorithm for Quadratic Assignment Problems," *Naval Research Logistics Quarterly*, Vol.18, pp.1-36(1971).

[9]Heider, C. H.: " An N-Step 2-Variable Search Algorithm for the Component Placement Problem," *Naval Research Logistics Quarterly*, Vol.20, pp.699-724(1973).

[10]Steinberg, L.: " The Backboard Wiring Problem: A Placement Algorithm," *SIAM Review*, Vol.3, pp.37-50(1961).

[11]Gaschutz, G. K. and J. H. Ahrens: " Suboptimal Algorithms for the Quadratic Assignment Problem," *Naval Research Logistics Quarterly*, Vol.15, pp.49-62(1968).

[12]Nugent, C. E., T. E. Vollmann and J. Ruml: " An Experimental Comparison of Techniques for the Assignment of Facilities to Locations," *Operations Research*, Vol.16, pp.150-173(1968).

[13]Burkard, R. E. and K-H. Stratmann: " Numerical Investigations on Quadratic Assignment Problems," *Naval Research Logistics Quarterly*, Vol.25, pp.129-148(1978).

[14]Garey, M. R. and D. S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman and Company,(1978).

[15]Onodera, H. et.al.: " Branch-and-Bound Placement for Building Block Layout," *Proc. ACM/IEEE Design Automation Conf.*, pp.433-439(1991).

[16]粟島, 金子 他: "モジュール配置問題における分枝限定法の実験的評価," 情処研報, DA-57-2(1991).

[17]金子，粟島 他: "上界値を考慮した分子限定配置手法とその実験的評価," 信学技報,VLD91-38(1991).

[18]Mayerhofer, S. and U. Lauther: " Congestion Driven Placement Using a New Multi-Partitioning Heuristic," *Proc. ICCAD*, pp.332-335(1990).

[19]茨木: 組合せ最適化～分枝限定法を中心として～, 講座·数理計画法8, 産業図書(1983).