

# 実用的な平面配置アルゴリズムと計算時間の検討

—PLANEX: プランニング・エキスパート・システム—

山田 邦夫  
(技術研究所)

## § 1. はじめに

本研究は、長方形を基本とする平面ユニットをユニット相互の連結条件によって生成し、敷地や方位の条件を用いて枝刈りを行ないながら平面配置図を実用的に求めるアルゴリズムの提案と、その計算時間について考察したものである。

コンピュータによる平面配置の方法には連結型と分割型とがある<sup>1)</sup>。前者は与えられた平面ユニットを何らかの条件によって連結させながら平面配置を求める方法であり、後者は与えられた長方形平面を幾つかの平面ユニットに分割する方法である。また、平面配置をあるモデルの最適解として解析的に求める場合と、一定の条件のもとに平面解を網羅的に求める場合とがある。ここで新しく提案するモデルは連結型であり、かつ条件を満たす平面解を発見的手法によって尽くすものである。

このような観点での最初の論文は「平面型の電算処理に関する研究」(太田)<sup>2)</sup>である。ここで用いられた平面配置パターン探求の方法は、一定の敷地の中に平面ユニットを順次にはめ込むものであった。また、格子状に分割された敷地の評価点・建物の結びつきなどの条件から平面配置を探求するもの<sup>3)</sup>や、太田の方法を平面ユニットとしてではなく室・通路・出入口などに拡張した配置モデル<sup>4)</sup>の提案もある。

しかし、このような提案においては配置モデルが可能な限りの平面配置を求めるアルゴリズムであり、その方法に幾つかの問題があった。また、平面配置を検討するための条件にも様々な提案があるが実際の設計上の条件とは考えにくいものもある。平面解を尽くす方法では、与えられた条件や平面配置を探求するアルゴリズムによって平面解の数や計算時間が大きく変化する。ルーズな条件のもとでは膨大な数の解が出力され、このような無数の解をチェックすることは設計者にとって困難な作業となる。したがって、チェックの困難な数の平面解を求める意味は実用的にはない。そのため、設計者が評価可

能な範囲に解を制限し、しかも計算時間が実用的な範囲に限定できることが重要となる。

以上のことから、平面配置を与えられた条件から自動的に求めるシステムには次の課題が残されている。

- (1)条件が実務の設計で十分通用するものであること。
- (2)容易に条件を変化させながら平面解を絞り込めるように配慮すること。
- (3)評価可能な数の平面解を求めるための計算時間が実用的な限度内であること。

以下には、上記の課題を解決するための入力条件、新しい平面配置の方法、および計算時間の検討結果について述べる。

## § 2. 平面配置のための設計条件

コンピュータを利用した平面配置手法を実用的なシステムとするためには、実務の設計で使われる設計条件の検討が必要である。分析に用いた設計例は、実施設計を前提とした基本設計に限りて依頼された学校の計画であった<sup>5)</sup>。この設計例の設計条件設定段階では機能・規模・敷地の3つの条件が整理され、この設計条件のもとに具体的な平面型が検討された。また、平面型の検討段階では日照や採光の条件が配置に大きく影響を与えた。平面配置をコンピュータによって自動的に生成するシステムの提案に当たっては、実務の設計から配置に影響を及ぼす主要な条件として次の4つの条件を抽出した。

- (1)機能の条件: 機能図によって表現
- (2)規模の条件: 規模算定表によって表現
- (3)敷地の条件: ゾーニング図によって表現
- (4)方位の条件: 施設配置図によって表現

ところが、これらの条件は空間構成のための拠り所となるが、むしろ発注者の抽象的な要求や条件を視覚的に整理し、それを確認するための共通言語的な性格をもっている。そのため、これらの設計条件はコンピュータの

入力条件として必ずしも十分とはいえない。そこで、これらの条件の基本的な意味を損なうことなく、どのように具体的なコンピュータへの入力条件とするかの検討が必要となる。

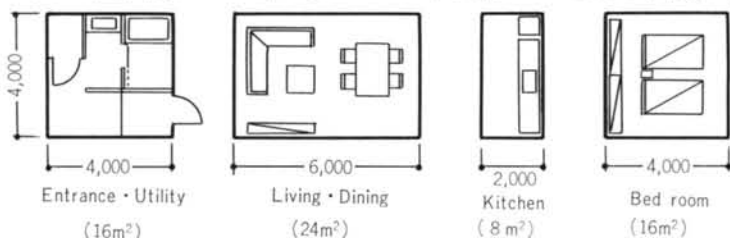
## 2.1 機能の条件

基本設計の実施例では、設計条件設定段階において空間の機能的な関係を機能図によって条件化した。しかし機能図は空間相互の何らかの関係を明らかにするが、具体的な空間関係を明確に示すとはいいいにくい。そのため空間と空間を結ぶ線の意味が動線上のつながり、組織上のまとまり、あるいは動線の交錯を避けるためのかなどは明確に区別されない<sup>3)</sup>。この点で、空間の具体的な配置を決める条件としては冗長といえる。しかし、機能図は発注者と設計者との条件レベルにおける確認媒体としての有効性は認められていることから、機能図に変わる条件を求めるのではなく、ここでは機能図をもとに、より空間配置に役立つ新しい空間との関係を表わす条件を示す。

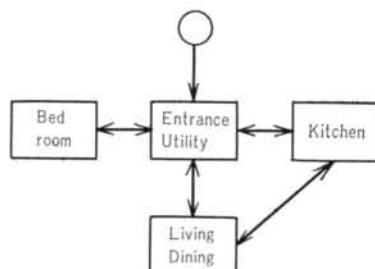
図一1は簡単な住宅のモデルであり、機能図には各空間の種類とその関係が線によって表現されている。条件の変換は、この関係を空間相互の連結関係に置き変えて表わそうとするものである。この例では、空間と空間が線で結ばれている同士を連結させ、そうでないものを空間上離して配置するものとして考える。しかし、このような2値の条件では制限が厳し過ぎ現実的でないため、ルーズな条件を導入し次の3つの連結関係を定義する。このことから、機能図をもとに空間と空間との関係を(0)非連結、(1)連結、(2)未定義の3条件で表わすと、例では図一2のような表現ができる。

## 2.2 規模の条件

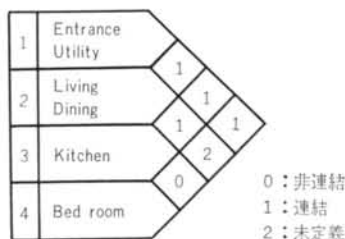
同様に、実務の設計においては各空間(諸室)の規模をユニットプランや現状施設の面積などをもとに分析し、その結果を規模算定表に示した。規模算定表は諸室別の面積を表わす。これらは数値データであるため、コンピュータへの入力条件としては扱い易い。しかし、面積だ



図一3 住宅における規模の検討



図一1 住宅の例における機能図



図一2 機能図に基づく空間相互の連結表

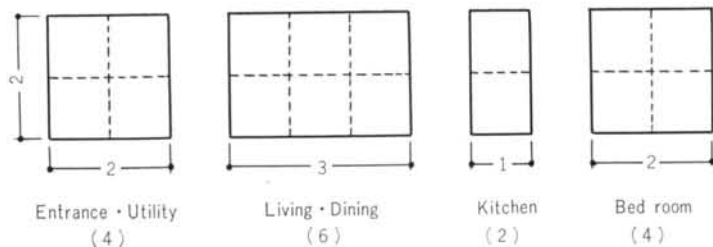
けの情報では室の大きさは定まるが、形は定まらない。そして、形は新たな条件を与えなければ無数に考えられる。実際の設計においては、モジュールという手法を用いて形の効率的な検討を行なうことが多い。

そこで、ここでは実際のユニットプラン等の検討をふまえて、諸室の規模を単位寸法の整数倍の大きさを辺にもつ長方形の平面ユニットと定義する。この場合、xおよびy方向の各単位寸法は異なってもよい。以上の検討から、前述の住宅の例における規模の条件は図一3に示すようなユニットプランによる検討に基づいて、図一4のような平面ユニットの規模の入力条件として表わされる。

## 2.3 敷地の条件

設計例における敷地の条件は、設計対象が学校計画であったため校内の敷地の使い分けをゾーニング図で示した。ゾーニング図は、その施設の建設可能な敷地の領域を区分するために用いた。ここでは、住宅の例によって建設可能な敷地の使い分けが図一5のように表現できるとすれば、前節の規模の条件との関係から図一6のよう

No.	室名	面積
1	Entrance·Utility	16m <sup>2</sup>
2	Living·Dining	24
3	Kitchen	8
4	Bed room	16



No.	平面ユニット名	規模	
		X	Y
1	Entrance·Utility	2	2
2	Living·Dining	3	2
3	Kitchen	1	2
4	Bed room	2	2

図-4 規模検討に基づく入力条件

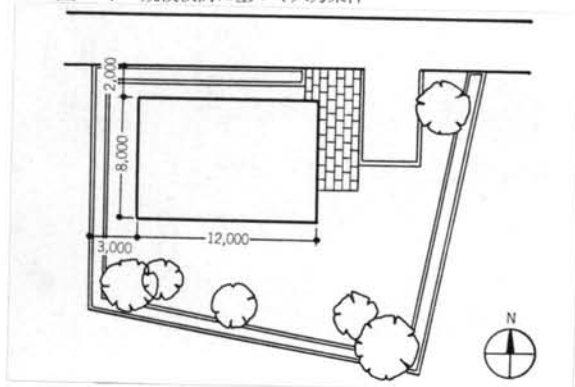


図-5 敷地配置図による条件の検討

な対応する単位寸法の整数倍をもつ長方形の敷地として条件化できる。

#### 2.4 方位の条件

方位の条件は、平面配置を検討する上で重要な役割りを果たす。実務の設計では、設計条件設定段階において整理された機能・規模・敷地の条件に基づいて平面型の検討を行なった。この段階において、日照や採光等の方位の条件に関する検討が進められた。しかも、こうした条件は平面配置を検討する上での一般的な条件であると考えられる。ここでは、以上の点をふまえ各平面ユニットの4つの方位に対して連結させない面を定義する。

住宅の例において、「入口は東側にとり、ユーティリティは北側採光、リビング・ダイニングは南側採光をとりたい」という場合には、連結させたくない面を図-7のように表わすことができる。これらは、図中の表のようなコンピュータへの入力条件として整理できる。

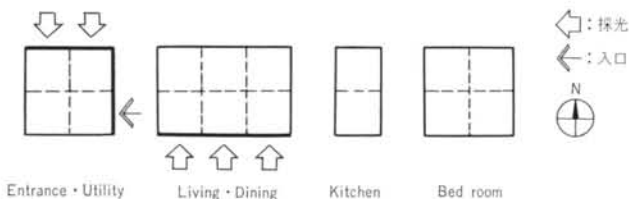


図-7 住宅における方位の検討と入力条件

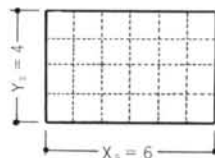


図-6 敷地の入力条件

#### 2.5 平面配置のための入力条件のまとめ

平面配置のための入力条件は実務の設計で用いられる条件をもとに、建物の用途や機能にあまり影響を受けない一般的なものを抽出した。今までの条件を住宅の例で整理して示すと表-1のようにになる。これらの入力条件は理解が容易であり、入力情報も少なく、条件の変更も簡便に行なうことができる。したがって、平面配置を行なうための入力条件は設計の実務で用いられる基本的な条件を備え、操作性も高いといえる。

### § 3. 平面配置アルゴリズム

ここでは、前章で検討した入力条件をもとに平面配置を求める新しいアルゴリズムを提案する。平面配置を求めるアルゴリズムは、平面ユニット間の連結関係をもとに解を網羅的に探求しながら、無効な解を方位や敷地の条件で枝刈りするものである。以下には、まず平面ユニットの連結のアルゴリズムを明らかにし、次いで方位や敷地の条件によって枝刈りを行なうアルゴリズムについて述べる。

#### 3.1 2つの平面ユニットの連結

平面配置探求のアルゴリズムは、前述の入力条件から

No.	平面ユニット名	方位			
		N	E	S	W
1	Entrance·Utility	0	0	1	1
2	Living·Dining	1	1	0	1
3	Kitchen	1	1	1	1
4	Bed room	1	1	1	1

0:連結不可 1:連結可

No	平面ユニット名	入力条件		規模		方位				連結				敷地	
		X	Y	N	E	S	W	1	2	3	4	X <sub>s</sub>	Y <sub>s</sub>		
1	Entrance・Utility	2	2	0	0	1	1	0	1	1	1	6	4		
2	Living・Dining	3	2	1	1	0	1	1	0	1	2				
3	Kitchen	1	2	1	1	1	1	1	1	0	0				
4	Bed room	2	2	1	1	1	1	2	0	0	0				

表-1 入力条件のまとめ

出発して、この条件を満足する全ての平面配置パターンを求めることである。その手続きを前述の住宅の例で示すと以下ようになる。まず、与えられた平面ユニットの配置順位を決める。配置順位は入力条件で与えられる順と連結関係から求める。最初に先頭の平面ユニット  $U_1$  が配置され、このときの  $U_1$  の左下角の頂点の座標を  $U_1(0,0)$  とする。以後の平面ユニットの配置はここを原点として表わす。次いで、配置された平面ユニット  $U_1$  に連結される平面ユニットを連結表から求める。連結表から選ばれる平面ユニットは  $U_1$  の連結行列について左から順に行を走査し、最初に1(連結)となる平面ユニットであり、 $U_2$  となる。こうして選択された  $U_1$  と  $U_2$  の2つの平面ユニットの連結パターンを求めることになる。さて、 $U_1$  に連結する  $U_2$  の全ての配置パターンは、ユニット座標の条件から図-8に示す14通りとなる。このとき、平面ユニット  $U_1$  の座標を  $(0,0)$  としたときの平面ユニット  $U_2$  の座標の軌跡は、図-8から図-9のようなプロットとして表現できる。

平面ユニット  $U_1$  に対して連結可能な  $U_2$  の全ての座標を求める手続きは、平面ユニット  $U_1$  の座標を  $(x_1, y_1)$ 、その大きさを  $X_1 \cdot Y_1$  とし、求める平面ユニット  $U_2$  の座標を同様に  $(x, y)$ 、その大きさを  $X_2 \cdot Y_2$  とすれば以下のようなになる。

(a)  $U_1$  に対して北側に面する  $U_2(x, y)$  の座標：  

$$U_2(x_1 - X_1 + k, y_1 + Y_1) \quad k=1, \dots, X_1 + X_2 - 1$$
 ……(1)

(b)  $U_1$  に対して東側に面する  $U_2(x, y)$  の座標：  

$$U_2(x_1 + X_1, y_1 - Y_2 + k) \quad k=1, \dots, Y_1 + Y_2 - 1$$
 ……(2)

(c)  $U_1$  に対して南側に面する  $U_2(x, y)$  の座標  

$$U_2(x_1 - X_2 + k, y_1 - Y_2) \quad k=1, \dots, X_1 + X_2 - 1$$
 ……(3)

(d)  $U_1$  に対して西側に面する  $U_2(x, y)$  の座標：  

$$U_2(x_1 - X_2, y_1 - Y_2 + k) \quad k=1, \dots, Y_1 + Y_2 - 1$$
 ……(4)

このことから、連結可能な組合せ総数  $N$  は式(1)~(4)

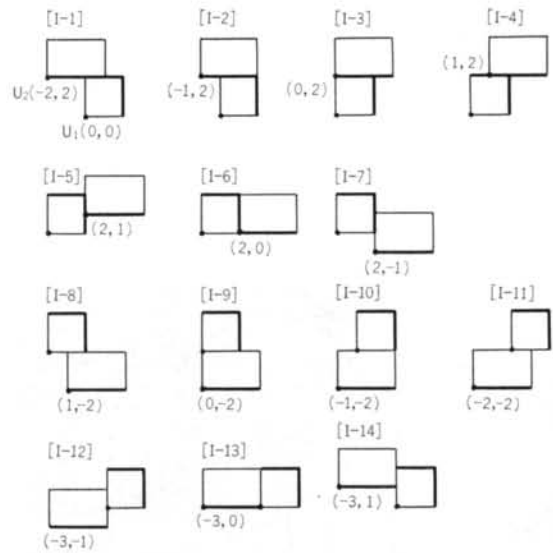


図-8 平面ユニット  $U_1$  および  $U_2$  の全連結パターン

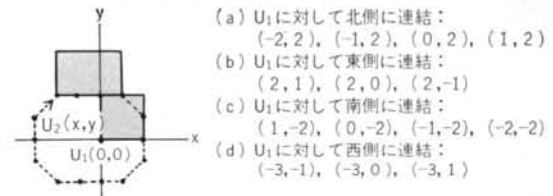


図-9 平面ユニット  $U_1$  に対する  $U_2$  の座標の変化

から次のように表わすことができる。

$$N = 2 \cdot (X_1 + X_2 - 1) + 2 \cdot (Y_1 + Y_2 - 1) = 2 \cdot (X_1 + X_2 + Y_1 + Y_2) - 4 \quad \dots\dots(5)$$

これに住宅の例を適用すれば式(5)から  $N=14$  となる。これは図-8で示した結果と当然一致する。この14通りの全配置パターンを求めるプロセスが、ここでは重要である。この特徴は、2つの平面ユニットの連結関係が試行錯誤によるのではなく、一定のアルゴリズムによって一義的に決定できるという点である。

### 3.2 方位の条件による枝刈り

方位の条件による平面解の枝刈りは、前述のアルゴリズムによれば容易に可能である。図-8に示された14種類の連結パターンのうち、平面ユニット  $U_1$  の北側および東側に連結させないという方位の条件から [I-1]~[I-4] ( $U_1$  の北側) および [I-5]~[I-7] ( $U_1$  の東側) の7種類のパターンを取り除くことができる。同様に  $U_2$  の平面ユニットの南側に連結させないという方位の条件から [I-1]~[I-4] の配置パターンは枝刈りされることになる。しかし、この場合には最初の条件によってす

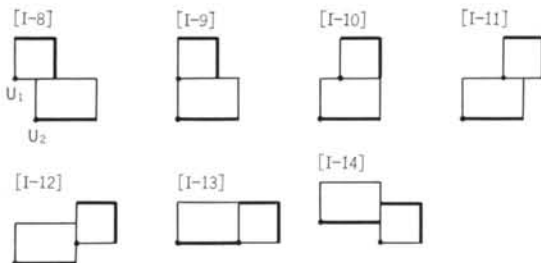


図-10  $U_1$  および  $U_2$  の全連結パターンのうち方位の条件を満たすパターン

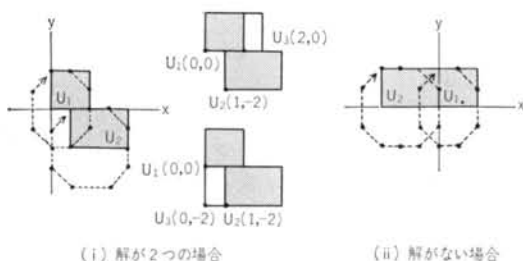


図-11 2つの平面ユニットに同時に連結するパターンに枝刈りが行なわれていることから、方位の条件を満たす配置パターンは図-10に示す7通りとなる。

この枝刈りのアルゴリズムは、平面ユニットが決まるとそれに対応する方位の条件が明らかになるので、始めから余分な連結パターンを求めなくて済む。 $U_1$ の平面ユニットの場合には、北側と東側に連結されるパターンを求める必要がないことから、式(1)および式(2)に基づく連結パターンの発生を行なわなければならないことになる。このように、方位の条件は連結ユニットが決まれば連結パターンを生成しないで平面解の絞り込みができるというメリットがある。こうした枝刈りの手法は、設計者が任意に条件を設定し、平面解の数や計算時間を制限するのに有効である。

### 3.3 2つの平面ユニットに同時に連結する配置パターン

図-10には、2つの平面ユニットの連結可能な全パターンのうち、条件を満足する7つのパターンを示した。ここでは、この有効な配置パターンに新しい平面ユニットを連結させる。前述した手続きによって平面ユニット  $U_3$  が選ばれる。 $U_3$ は連結の条件から  $U_1$  および  $U_2$  に連結し、 $U_4$ に連結しないというものである。ところで  $U_4$ は未配置であり、この段階では  $U_1$  および  $U_2$  に同時に連結するパターンを探索することになる。この連結条件を満足する  $U_3$ の左下角の座標は、図-11から明らかのように  $U_1$  および  $U_2$ の平面ユニットに対して  $U_3$

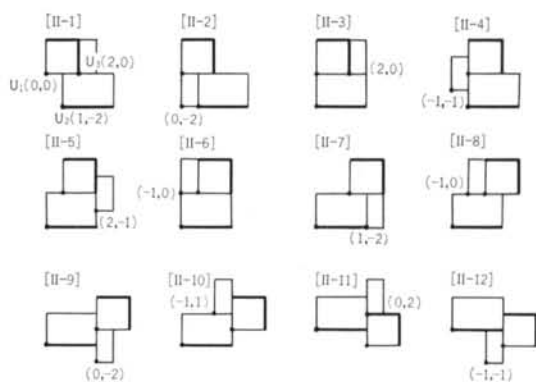


図-12 2つの平面ユニットに同時に連結する全パターン

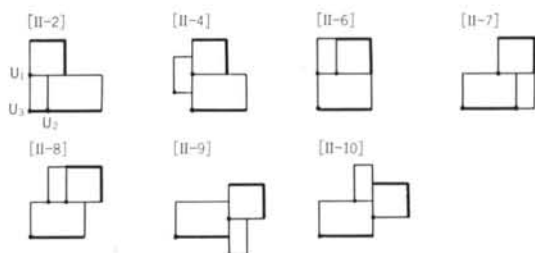


図-13  $U_1$ ,  $U_2$  および  $U_3$  の全連結パターンのうち方位の条件を満たすパターン

の平面ユニットの連結可能な座標の軌跡の交点で表わされる。

すなわち、 $U_1$ に対して連結可能な  $U_3$ の座標の集合を  $A_{1,3}$ 、 $U_2$ に対して連結可能な  $U_3$ の座標の集合を  $B_{2,3}$  とし、 $U_1$  および  $U_2$  に同時に連結可能な  $U_3$ の座標の集合を  $C_{1,2,3}$  とすれば、 $A_{1,3}$  および  $B_{2,3}$ の要素のうち  $a=b$ なる要素のみがこの条件を同時に満たす連結可能な座標となる。これらは式(6)のように表わすことができる。

$$C_{1,2,3} = A_{1,3} \cap B_{2,3} \quad \dots\dots(6)$$

ただし、 $A_{1,3} = \{a | a \text{は } U_1 \text{ に対して連結可能な } U_3 \text{ の座標}\}$

$B_{2,3} = \{b | b \text{は } U_2 \text{ に対して連結可能な } U_3 \text{ の座標}\}$

図は、有効な7つのパターンのうち2つを典型例として取り出し、 $U_1$  および  $U_2$  に同時に連結する配置パターンを示したものである。図から平面解が必ず求まるわけではなく、(ii)のように交点のない、つまり解のない場合もある。(i)は平面解が2つあり、式(6)を適用すると、

$$C_{1,2,3} = \{(2, 0), (0, -2)\}$$

となる。

以上の手続きを図-10の7つの全パターンについて実施すれば、 $U_1$  および  $U_2$  に対して同時に連結する全

配置パターンは図-12のようになる。さらに、方位の条件で枝刈りを行なって残った配置パターンを示すと、それらは図-13のように表わされる。

### 3.4 敷地の条件による枝刈り

図-12および図-13には、 $U_1, U_2$  および  $U_3$  の3つの平面ユニットの連結条件を満たす12の配置パターン

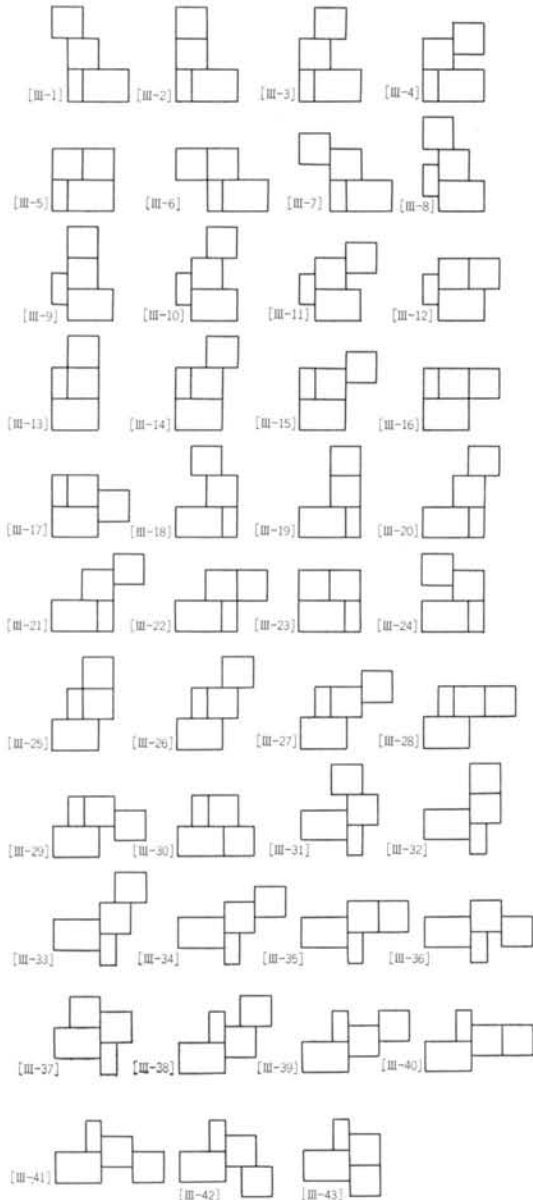


図-14  $U_1 \cdot U_2 \cdot U_3$  および  $U_4$  の全連結パターン

と、そのうち方位の条件を満たす7パターンを示した。最終的に求める配置パターンは、この7パターンに  $U_4$  の平面ユニットを連結したもののうち他の条件を満たすものとなる。図-14は、7パターンの有効解に連結条件をもとに求めた全配置パターンである。さらに、方位の条件によって解を選ぶと、残った平面解は図-15のようになる。

ここでは、こうして残った有効解のうちx方向およびy方向の各方向の長さが、最初に設定した  $X_s, Y_s$  の敷地の条件を満足するかを調べる。有効解の最大長さは、各平面ユニットの座標と平面ユニットの大きさから求められる。そこで、平面解の各最大長さを  $X_{max}$  および  $Y_{max}$  とすれば、これを満たす条件式は(7)のようになる。

$$(X_{max} \leq X_s) \cap (Y_{max} \leq Y_s) \quad \dots\dots(7)$$

さて、図-15の8通りの平面解のうち敷地の条件を満たす最終平面解は、図の各方向の最大長さと敷地の条件から、明らかに図-16で示される3通りとなる。

### 3.5 アルゴリズムのまとめ

図-17は、説明で使用した住宅の例におけるコンピュータの出力である。利用したコンピュータは IBM-3081で、言語は PL/I を用いた。出力図はプリンタープロッタによるものである。また、平面配置パターンの探索のアルゴリズムは図-18のフローチャートで表わされる。フローチャート中の具体的な各手続きは前述したとおりであるが、配置パターンを求める方法はバックトラッキングと呼ばれる木探索の手法を用いている。このシステムにおける配置パターンの探索は、規模の条件と連結の

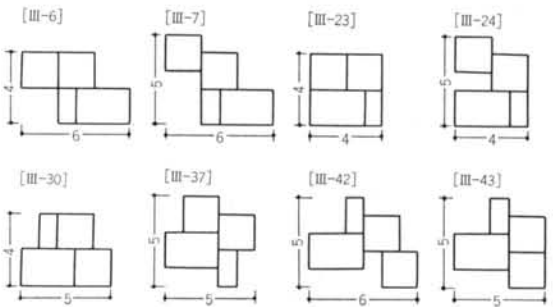


図-15  $U_1 \cdot U_2 \cdot U_3$  および  $U_4$  の全連結パターンのうち方位の条件を満たすパターン

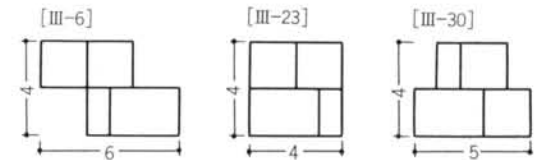


図-16 入力条件を満たす最終平面解

NO.	SPACE ACTIVITY	WIDTH X-UNIT	LENGTH Y-UNIT	ORIENTATION N E S W	ADJACENCY MATRIX										SIZE OF SITE		
					1	2	3	4	5	6	7	8	9	10	X-UNIT	Y-UNIT	
1	ENTRANCE	2	2	0 0 1 1	0	1	1	1								6	4
2	LIVING	3	2	1 1 0 1	1	0	1	2									
3	KITCHEN	1	2	1 1 1 1	1	1	0	0									
4	BED ROOM	2	2	1 1 1 1	1	2	0	0									

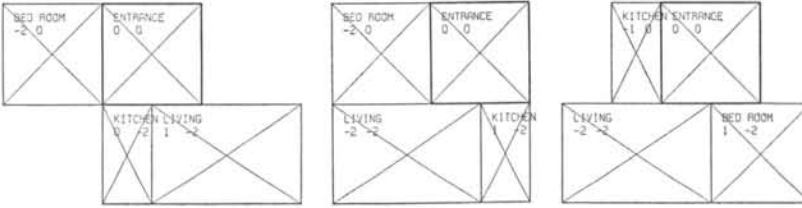


図-17 コンピュータの出力例

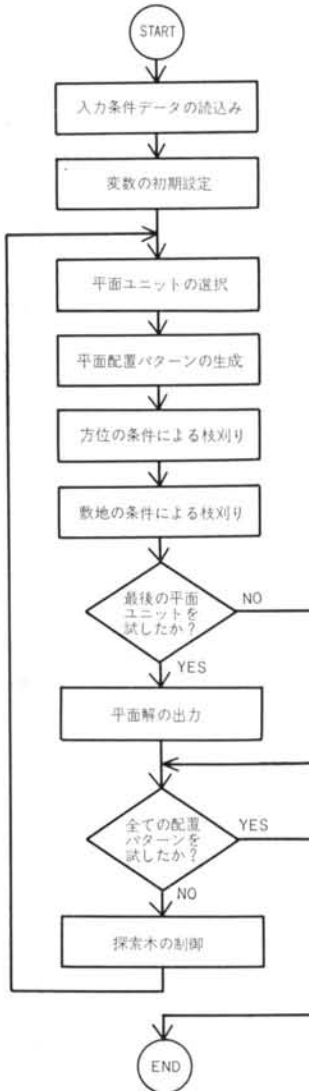


図-18 平面配置のアルゴリズム

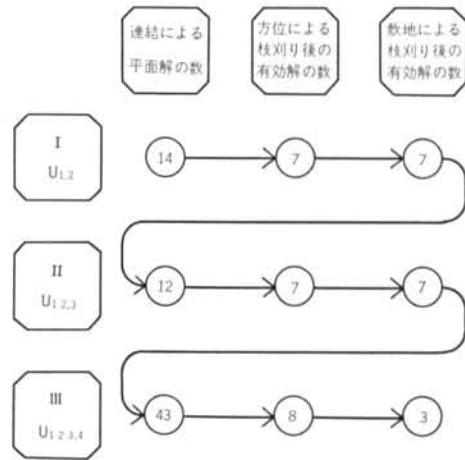


図-19 平面解探索過程における有効解の変化

条件のみで行なうことができるが、連結の条件がルーズであると解が無数に生成されることになる。そのため、システムは方位の条件や敷地の条件によって各探索ステップで枝刈りができるように設計されている。例では、初期の条件を満たす最終平面解を3通りに絞り込むことができた。枝刈りは全ての平面解を求めてから評価するのではなく、探索ステップ毎に方位や敷地についての評価を行なうため計算時間を大幅に短縮できる。このときの計算時間は CPU タイムで0.66秒となった。

図-19は、平面ユニットの連結ステップにおける枝刈りに有効な平面解の数の変化を示したものである。図から、第1ステップでは14通りの解が7通りに、第2ステップでは12通りのものが7通りに、さらに第3ステップでは43通りのものが3通りに絞られ、枝刈りが平面解の数を制限する方法として有効であることが分かる。以下には、こうした条件の変化に対して平面解の数や計算時間がどのように変化するかを検討を行なう。



## § 4. 平面解の数と計算時間

平面配置の方法は、与えられた条件を満たす平面解を尽くすものである。ここでは、これらの条件によって平面解の数や計算時間を、どの程度実用的な範囲に制限できるかについて考察する。検討の方法は、それぞれの条件について、それを変化させたときの平面解の数と計算時間の変化を実験的に調べて行なう。計算時間は、実際の処理を行なったときのコンピュータの CPU タイムを用いた。この実験では、各条件を変化させたときの平面解の数と計算時間の変化を、以下の4つのケースについて明らかにする。

- (a) 平面ユニット数と平面ユニットの大きさの変化による平面解の数と計算時間の変化
- (b) 連結の条件の変化による平面解の数と計算時間の変化
- (c) 方位の条件の変化による平面解の数と計算時間の変化
- (d) 敷地の条件の変化による平面解の数と計算時間の変化

### 4.1 平面ユニット数と平面ユニットの大きさの変化による平面解の数と計算時間の変化

実験は、平面ユニット数および平面ユニットの大きさを変化させながら各条件毎に実際の計算を行なって、平面解の数と計算時間を求める。このとき、方位および敷地の条件による枝刈りは行なわない。また、連結の条件は最もルーズな条件（1つの平面ユニットしか連結しない

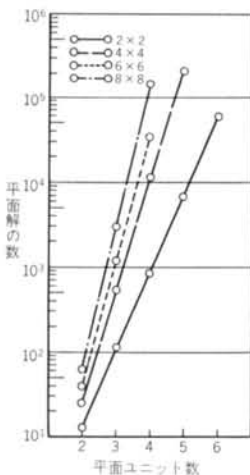


図-20 平面ユニット数の変化による平面解の数の変化

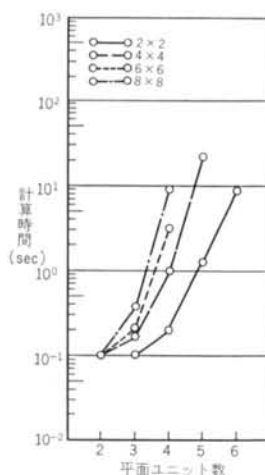


図-21 平面ユニット数の変化による計算時間の変化

い状態)とした。実験結果は図-20 および 図-21 に示すとおりである。

実験結果から、最もルーズな条件下における平面ユニット数の変化による平面解の数と計算時間は平面ユニット数の増加によって幾何級数的に増加している。また、平面ユニットの大きさの変化によって、その増え方は一層顕著である。このように、規模の条件と最もルーズな連結条件のみで平面解を網羅的に求めることは、この結果から結局、平面解の数や計算時間が増えて実用的でないといえる。

### 4.2 連結の条件の変化による平面解の数と計算時間の変化

前述したように条件がルーズであればシステムの機能上、解の数は無数に出力され実用的な意味を失ってしまう。ここでは、連結の条件を変化させたときの平面解の数と計算時間の変化を調べる。条件は、どの平面ユニットも他の1つの平面ユニットに連結するという関係から出発し、実験は以下の3タイプについて実施した。こ

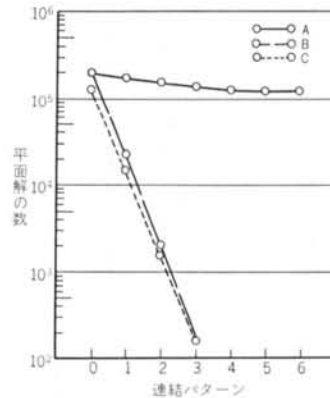


図-22 連結パターンの変化による平面解の数の変化

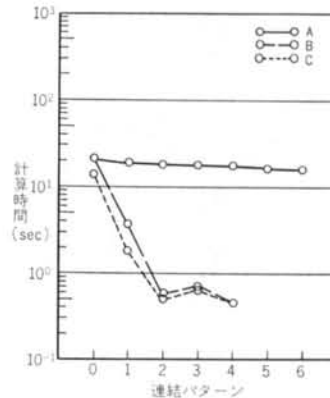


図-23 連結パターンの変化による計算時間の変化



のときの平面ユニットの大きさは  $4 \times 4$  で、平面ユニット数は5とした。

実験タイプAは、連結の最もルーズな連結パターンから出発し、未定義(2)の連結関係の要素を1つずつ非連結(0)の条件に変えたときの平面解の数と計算時間の変化を調べるものである。また、実験タイプBは未定義(2)の連結関係を1つずつ連結(1)の条件に変えながら、同様に平面解の数と計算時間の変化を調べるものである。さらに実験タイプCは、非連結(0)の条件を1つずつ連結(1)の条件に変えたときの同様な変化を調べるものである。実験結果は図-22 および 図-23 に示すとおりとなった。図中の横軸の連結パターンは、条件を1つずつ変化させるプロセスを示している。

2つの図から、平面解の数と計算時間の変化はほぼ同様な傾向を示している。一方、各実験タイプの違いはAの減少の仕方がゆるやかなのに対して、B,Cの減少の変化は著しい。この違いは、明らかにAが非連結(0)の関係を与えるのに対して、B,Cは連結(1)の関係を与えている点である。すなわち、同時に連結するパターンが増えると解の数は著しく減少し、計算時間も解が一義的に決まるため同様な傾向となる。しかし、非連結の条件は連結以外の全てのパターンを調べることになり、減少効果は小さい。このことから、平面解の数および計算時間を減らすためには、平面ユニット相互の連結(1)の条件を加えて行くことが非常に有効であることが分かる。

#### 4.3 方位の条件の変化による平面解の数と計算時間の変化

ここでの実験は、方位の条件の要素を1つずつ変化させたときの平面解の数および計算時間の変化を観察するものである。実験は、方位以外の条件を一定にし、方位の条件のみをルーズな条件(1)から出発して、各要素を1つずつ連結不可(0)に条件を変えながら行なった。実験結果は図-24 および 図-25 のとおりとなった。ここで、横軸の実験番号は非連結(0)の面を1つずつ増やして行くプロセスに対応している。

実験結果から、方位の条件を1つずつ非連結の条件に変化させることによって平面解の数および計算時間は、それに対応して減少する傾向を示している。また、平面解の数と計算時間はともに同様な傾向で減少する。方位の条件による枝刈りの機能は、4つの方位に対して1面を連結させないという条件から、正方形ユニットでは組合せ数が必ず減少することになる。また、方位の条件は連結の条件と同様に平面解の数および計算時間の両方も減少させる効果があり、結果から方位の条件が枝刈

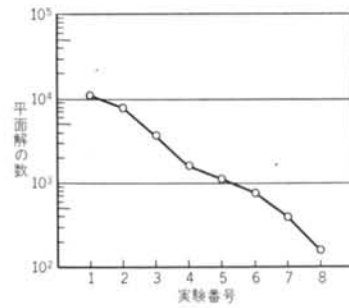


図-24 方位の条件の変化による平面解の数の変化

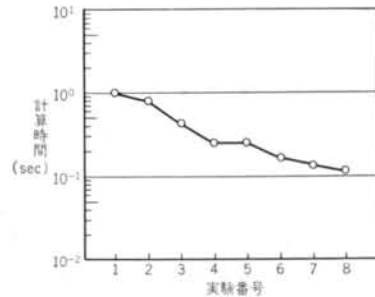


図-25 方位の条件の変化による計算時間の変化  
りの機能として有効なものであるといえる。

#### 4.4 敷地の条件の変化による平面解の数と計算時間の変化

ここでも同様な方法によって、敷地の大きさを変えたときの平面解の数および計算時間の変化を調べた。敷地の条件以外の条件は固定し、敷地の条件は平面ユニットの大きさをもとに実験番号(1)  $12 \times 12$ , 同(2)  $10 \times 10$ , 同(3)  $8 \times 8$  の3種とした。その結果は図-26 および 図-27 のようになった。

実験結果から、平面解の数は敷地の条件を制限するこ

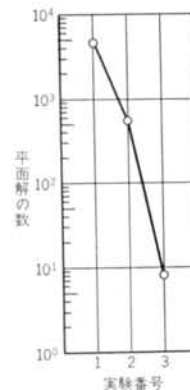


図-26 敷地の条件による平面解の数の変化

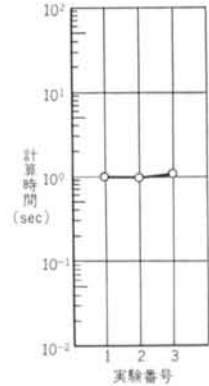


図-27 敷地の条件による計算時間の変化

とによって大幅に減少させることができるが、計算時間は条件を変えても殆んど変化しない。敷地の条件による枝刈りは、全ての平面ユニットを配置したときにX方向およびY方向の長さが最大となることから、最終解に近づくほど多くなる。しかし、計算時間は配置パターン探索の全プロセスにおいて評価が行なわれるため、探索木の規模が同じなら変化しないことになる。したがって、敷地の条件による枝刈りは平面解の数を制限するには有効であるが、計算時間を縮める効果はないといえる。

## § 5. おわりに

今回の報告は、コンピュータによる平面配置手法を実用化するために、問題点の分析を通して新しい配置アルゴリズムを示した。このシステムによって、平面解の数や計算時間を設計者が条件を操作しながら制限することができ、一応実用的な利用が可能となった。システムの利用に当たっては出来るだけ厳しい条件から始め、結果

を見ながら徐々にルーズな条件に変えると能率が良い。また、システムは設計条件に適合する全ての配置パターンを求めるという方法により、設計の質の向上や手戻りを減少させる効果がある。さらに、単なる平面配置の検討だけでなく、施設配置計画への応用や設計案と条件との検証にも利用できる他、敷地条件を一定の外形に置き換え、三次元への計画にも比較的容易に拡張できるメリットがある。一方、システムは汎用性があり、しかも基本的な入力条件を備えているので、こうした建築設計の分野以外の配置問題にも適用可能である。その反面、現状のシステムでは空間を長方形の平面ユニットで表現する点や、通路を1つの平面ユニットとして扱う必要があるなどの問題も残されている。

コンピュータを使った平面配置のシステムは、現在大型コンピュータを利用してプリンタープロッタ上に出力しているため、条件を変更してから結果を得るまでの作業に手間がかかる。今後は、対話型のワークステーション上で条件を変更しながら平面配置を検討できるシステムが非常に効果的となる。

### <参考文献>

- 1) 太田利彦・佐藤庄一：“平面型の電算処理に関する研究”日本建築学会論文報告集 第158号(1969年)
- 2) T.M. Willoughby: “Computer-aided Design of a University Campus” Architects' Journal, 25 March (1970)
- 3) 岡崎甚幸・伊藤明宏：“室・通路・出入口の最適配置モデルの研究”日本建築学会論文報告集 第311号(1982年)
- 4) 山田邦夫：“基本設計におけるCADの課題”清水建設研究所報 第36号(1982年)
- 5) 日本建築学会建築計画委員会：“設計方法・Ⅲ：設計プロセス/道具の提案”彰国社(1974年)