

樹脂建材生産における板取り

甲斐 良隆, 加藤 直樹

1. 生産業務の流れ

エンジニアリング・プラスチックの用途が急拡大し、その生産量は年を追うごとに顕著な伸びを示している。

特に、その一種である樹脂建材は、軽量・透明で耐衝撃性に富むという優れた特質から、従来ガラスや鉄板が使われていた分野において、急速に代替が進んでいる。たとえば、アーケード、ショーケース、建物の側壁等が代表的なものである。

(写真1)

次に製品が製造段階から最終顧客に届くまでの流れを追ってみると以下のようになる。(図1)

まず、工場で生産された樹脂板(シート)は、通常、1m×2m程度の一定形状をしており、定期的に、かつ大量に販売店へ送られる。なお、このシートは原板と呼ばれている。さらに、販売店では、顧客のさまざまな注文に応じて、そのシートを細かく切断し、各地へ配送している。

2. 切断上の問題

ところで樹脂建材の用途はさまざまであり、当然、その形状もいろいろと異なっている。まさに多品種少量の見本のようなものである。また概し

て注文の納期は厳しく、いかに迅速に顧客の手元に届けられるかが大きな問題となっている。

さて、原板を切断するさい、必要部分を切り取った残りは、通常ロスとして処分される。このロスの出具合いは採算に大きく影響し、すぐに、10~20%の利益差となって現われてくる。そこでどの企業でも、かなりの経験を積んだベテランを切断計画業務に配し、歩留りの向上に努めているのが現実である。

しかし、それでも、注文を受け取る期間が不定期であり、しかも、その形状、寸法がほとんど予知できないという事情から、常に仕事に追われている状態である。なお、注文に応じて、原板を生産すれば良いのではないかと考える方もあろうが、工場設備の問題やタイミングの点で非現実的

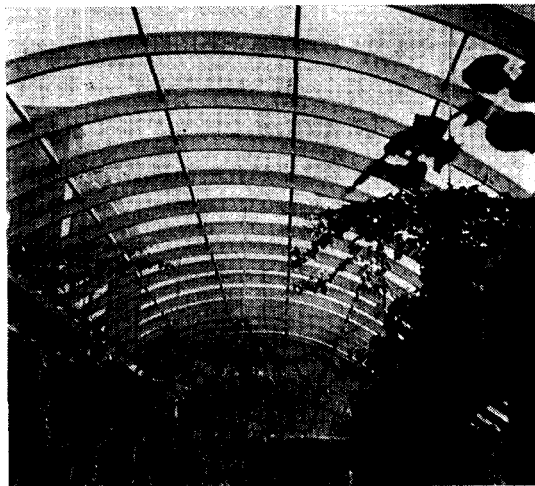


写真1 樹脂建材の使用例①

かい よしたか 帝人システムテクノロジー(株)
〒100 千代田区内幸町2-1-1 飯野ビル
かとう なおき 神戸商科大学 管理科学科
〒655 神戸市垂水区星陵台4-3-3

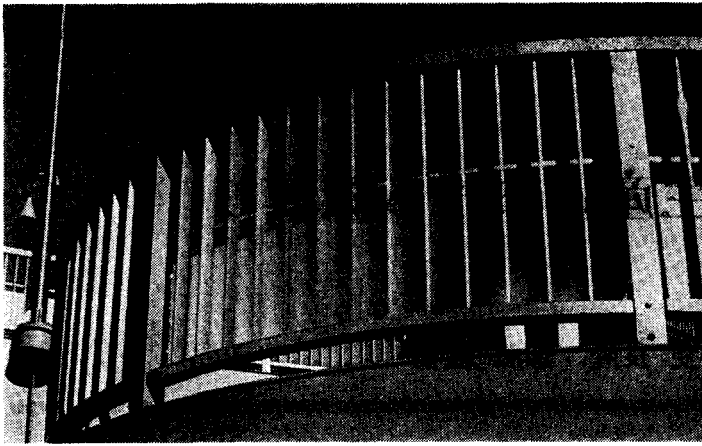


写真1 ②

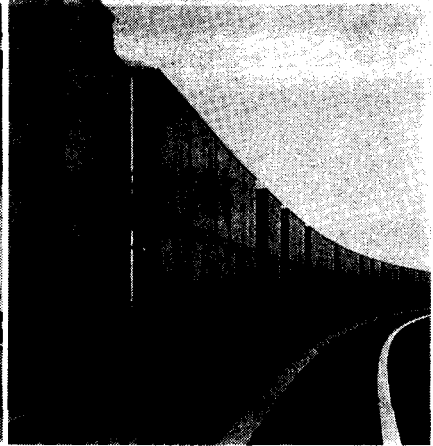


写真1 ③

な方法になってしまう。

それゆえ、最適な切断方法を即座に与えてくれる板取りシステムの実現が従来から望まれていたわけで、その研究、開発に着手した次第である。

3. アルゴリズムの概要

ロスが最小になるさまざまな原板の切り取り方を求めることは、一般に非常に難しいことが知られている [1]。したがって、必要な原板の枚数を最少とまでは言わないまでも、できるだけ枚数を少なくおさえる近似アルゴリズムを作る必要がある。そのためには、実際に発生するデータのパターンについても熟知する必要がある。

そのためアルゴリズムの概要に入る前に、アルゴリズムを開発・評価するために用いたデータについて述べる。表1～表3は、そのデータを示している。ケースごとの特色を簡単に述べるならば、ケース1、ケース2は比較的製品数が少ない

が、原板に対し大きな製品が多い。また、原板は3種類挙げられているが、複合して用いないことにしている。ケース3は、製品数が多くさまざまなサイズであり、なかでも特に代表的なタイプだと言うことができる。そこで実際に表のデータを用いて板取りした例を図2に示す。

樹脂建材（製品）の形状の多様性については先に述べたとおりであるが、包括的にとらえることによって多様性のうちにいくつかの特性を見出すことができる。すなわち、その特性こそがアルゴリズム開発の背景となりうるもので、言いかえれば、その特性をふまえたアルゴリズムを開発しなくてはいけない。もちろんイレギュラーなデータほど多くの特性を持ち得るものであるが、ここで扱ったデータにおける特性としては、以下の3点を挙げることができる。

1. 原板の高さ・幅の比が1:1ではないこと
2. 原板に対し比較的大きな製品が多いこと
3. 製品の中には同型のものが複数個あるものもあること

また、製品の形状は一義的に与えられているが、実際の板取りにおいては、どちらが縦、どちらが横という概念はない。すなわち、製品は90度回転して板取りすることもある。

さてわれわれが開発したアルゴリズム

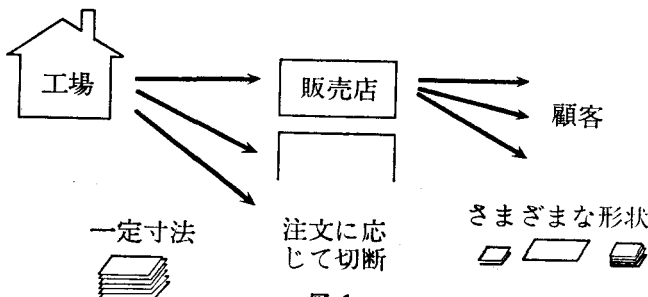


図1

は、神戸商科大学・笠原氏・中谷氏の卒業論文の成果であり従来多くの研究者によって考えられてきた組合せ的アルゴリズムとは違い ([2], [3], [7], [8], [9]), 製品の90度回転を考慮に入れたものとなっている。以下の説明は、理解に重点を置き厳密性に欠けるが、詳しくは[10]を見ていただきたい。

開発したアルゴリズムは、まず先に示した製品のデータがある順序でソートし

たリストを作製し、そのリストの順で原板に割り当てる。なお原板への割り当て方法は、原則としてファースト・フィットを適用する。すなわち、今図3のように入らずで使用された原板がn板あったとすると、次の製品は、1枚目、2枚目と順に見てゆき、最初にフィットする原板に割り当てる。どこにも入らなければ新しく原板を用意して(n+1枚目)割り当てる。

われわれの開発したアルゴリズムは、いわゆる、BL (Bottom-Left) アルゴリズムを応用したものである。BLアルゴリズムは、詳しくは [4], [6]を見ていただき

図2 板取り例 (ケース2, 原板サイズ1000×2000) ▶

表1 ケース1のデータ
原板のサイズ (3種類)
910×1820, 1000×2000,
1220×2440
製品 (54枚)

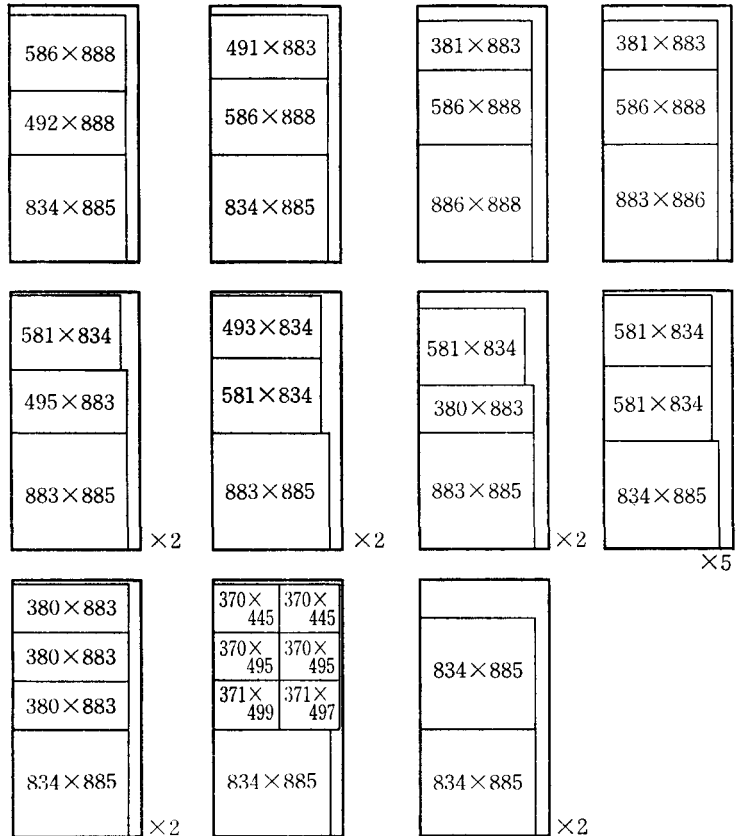
縦	横	枚
799	269	4
784	1667	4
833	269	4
818	1667	4
848	262	4
845	1011	4
621	269	4
606	1667	2
813	1667	2
635	1133	4
156	1176	2
813	790	2
813	669	2
768	817	2
768	366	2
1678	307	2

表2 ケース2のデータ
原板のサイズ (3種類)
910×1820, 1000×2000,
1220×2440
製品 (64枚)

縦	横	枚
888	586	4
888	886	1
888	492	1
371	499	1
497	371	1
883	381	2
883	491	1
883	886	1
495	370	2
883	380	8
883	885	6
883	495	2
834	885	14
834	581	16
834	493	2
444	370	2

表3 ケース3のデータ
原板のサイズ (1種類)
1820×3000
製品 (234枚)

縦	横	枚	縦	横	枚
837	913	8	837	333	16
837	540	8	888	586	2
787	882	12	837	333	12
787	333	12	887	882	6
837	371	12	887	333	6
887	398	24	769	568	6
887	333	24	587	882	6
815	415	3	587	333	6
815	348	3	1117	348	3
815	861	3	1117	415	3
837	864	16	1117	861	3
837	398	16	887	664	24



たいが、簡単に述べるならば、原板に対し製品をなるべく下、なるべく左に割り当ててゆくものである。われわれの実験は、90度回転を許すものであるので、回転をしたもの、しないものをそれぞれBLアルゴリズムで割り当て、より前の原板に割り当てられた方を採用する。また、

同じ原板に割り当てられた場合、アルゴリズムによって処理が違ってくる。

HEIGHT-BLアルゴリズム(高さ優先)—このアルゴリズムは図4のように90度回転したもの、しないものを比べ、より高さが低くなる向きを採用するものである。図4の場合Aを採択する。

LOW-BLアルゴリズム(底優先)—このアルゴリズムは、図5のように回転したもの、しないものを比べ、より低い位置に割り当てられる方を採用するものであり、図5の場合Bが採択される。

RAT-BLアルゴリズム(率優先)—このアルゴリズムは、原板の高さ・幅の比が1:1でないことに注目したものである。製品を90度回転した

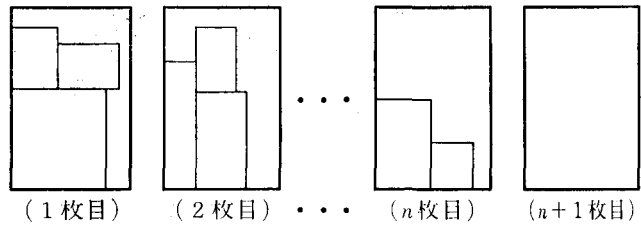


図3 ファースト・フィット

もの、しないもの、それぞれ図6のように製品の右上の頂点の位置に原板に対する比率の和を計算し、より低い方を採用する。図6の場合Aが採択される。

RT2-BLアルゴリズム(率優先)—このアルゴリズムは、割り当てをする前にそれぞれの向きで、原板の高さに対する製品の縦、原板の幅に対する製品の横の比を出し、ともに1/2以下である場合にだけその向きで固定し、それ以外はRAT-BLアルゴリズムを適用するものである。

RT3-BLアルゴリズム(率優先)—このアルゴリズムは同型の製品が複数個あることと、原板の高さと幅の比が1:1でないことに注目したものである。高さ・幅の比が1:1でないことは、

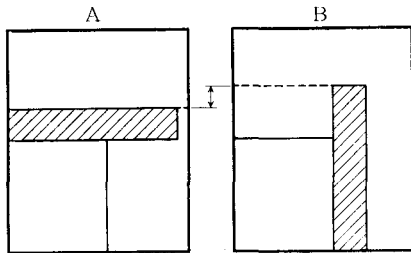


図4 HIGH-BL

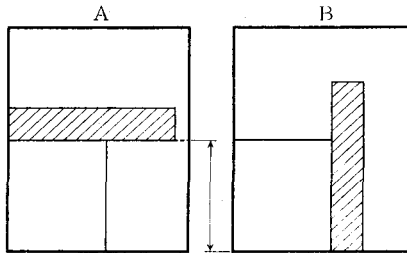


図5 LOW-BL

—高さ>幅と考えると—幅に対し高さ方向の方が多くの製品の割当てが行なえることを意味する。すなわち、高さ方向は無視して、幅方向のすき間ができるだけ少なくなるように製品の向きを考える

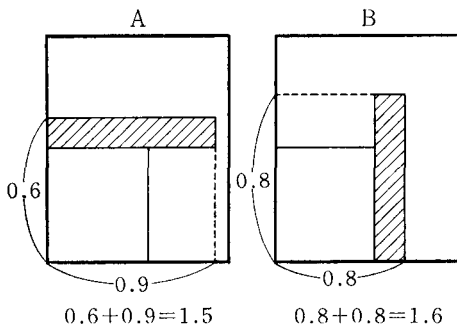


図6 RAT-BL

る。たとえば図7の製品に対しては、BよりAの方が幅方向に良く詰まっていると言え、この場合Aのようにその製品を縦長で詰めるようにする。

以上のような5つのアルゴリズムを開発したが、ケース2においてあまり良い結果が得られなかった。もういちど図2を見ていただきたい。これは最適解の1つである。すなわち、最適解は、原板を横方向に切断(ギロチンカット)できる形で与えられるのである。そこでそのような形で割

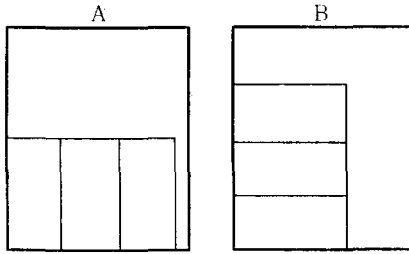


図 7 RT3-BL

り当てるアルゴリズムが望まれる。90度回転を許さない場合は、そのようなアルゴリズムとして Hybrid-Best-Fit アルゴリズムが知られているが(詳しくは[5]参照)、われわれはそれを改良した次のHBM (Hybrid-Best-Matching) アルゴリズムを開発した。

HBMアルゴリズムは、まず原板の高さを考え、既存のアルゴリズムにしたがいブロックという単位を作りながら割り当ててゆく(図8)。次にできたブロックを2ヶ、3ヶと組み合わせてゆき、その高さと同板の高さとの比がある水準以上になったら原板への割り当てを許すというものである。組合せが許されなかったブロックは、Best-Fitで原板に割り当てられる。すなわち、あき面積の少ない原板から順に見てゆき、割り当てを行なう。

アルゴリズムの良し悪しは、データのリストの作製方法にも大いに依存している。ソート方法はいろいろと考えられるが、われわれは以下の6種類を採用した。これらのソート方法は、他の実験においても、おおむね優れていることが実証されている([7, 10])。

DADW—面積の大きい順にソートし、同じものは横幅の大きな方を優先する。

DWDH—横幅の大きい順にソートし、同じものは高さの高い方を優先する。

TMAX—縦・横どちらでも大きな方を比較の規準とし、その降順にソートする。

TMIN—縦・横どちらでも小さい方を比較の規準とし、その昇順にソートする。

PLUS—縦・横の和を降順にソートする。

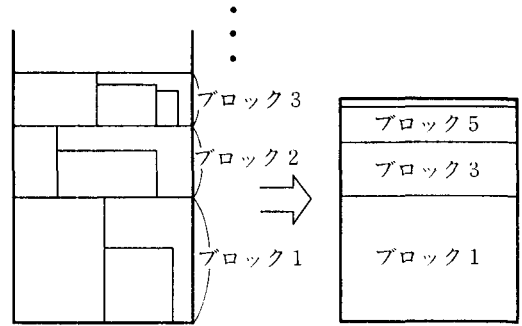


図 8 HBM

SMAX—製品をすべて縦長にし、それから高さで降順にソートする。

4. 主な計算結果

初めにケース3について示し、次にケース1、ケース2について示す。

さてケース3であるが、まず全体として、最適解がわかっていないので断言できないが、充填率がほぼ90%というのは満足できるものであろう。アルゴリズムとしては、RT3がなかでも一番優れており、次にはHEIGH, LOWが優れている。ソートにおいては、TMINが良くないことが挙げられる。また、最も良い結果が得られたのは、RT3でTMAXの時であった。以後ケース1、ケース2においては、RT3, TMAXを中心に実験を行なう。(表4参照)

次にケース1であるが、いずれの原板サイズにおいても最適解を得ることができた。最適解は人手による計算により求めた。製品数が比較的少ないので、アルゴリズムによる違いはわからなかったが、ソートについては、TMAXよりDADW, PLUS ソートの方が比較的良い結果となっている。また、表中の空欄は実験を行っていないことを示す。(表5, 表6参照)

次にケース2であるが、全体としてあまり良い結果が得られていない。今まで最も優れていたRT3が、いずれの原板サイズにおいても最適解より2枚多い結果となっている。しかし、原板のサイズが1000×2000の時、HBMによって最適解が

表 4 ケース 3 実験結果

		PACKING				
S O R T I N G		HEI	LOW	RAT	RT 2	RT 3
	DADW	23 2096568 89.617	23 4538457 89.617	24 1177155 85.883	24 1239426 85.883	23 2096568 89.617
	DWDH	23 2682981 89.617	23 3078252 89.617	24 981684 85.883	24 1110555 85.883	23 1701297 89.617
	TMAX	23 2529052 89.617	23 3402636 89.617	24 586413 85.883	24 586413 85.883	23 1023205 89.617
	TMIN	25 782334 82.448	25 4694004 82.448	26 2347002 79.277	26 1564668 79.277	25 1564668 82.448
	PLUS	23 2816181 89.617	23 4274433 89.617	24 1177155 85.883	24 1501497 85.883	23 1834497 89.617
	そのまま	24 4598208 85.883	24 4598208 85.883	24 4598208 85.883	24 4598208 85.883	24 3065472 85.883

原板のサイズは(高さ)3000×(幅)1820

(上段) 原板の枚数

(中段) 最後の原板に割り当てられた製品の総面積

(下段) 充填率(%)

得られていることは、大変評価できる。(表 7, 表 8 参照)

この結果をみると、現在、人手でやっているものとそれほど変わらないというものであった。これは、計算例に代表的なサイズを採用したことが大きな理由で、人がこのパターンをかなり熟知していたことによるものだろう。

もっとイレギュラーなケースでは、人手よりずっと良い結果が出ると思われるし、何よりも、短時間でコンスタントに良い切断方法を出力してくれるのが魅力である。

5. ま と め

今までの研究を評価してみると、採用したアルゴリズムにより、かなりの精度が得られることが明らかになった。さらに、これを実用化するため

表 5 ケース 1 の実験結果

		原板のサイズ(RT 3 アルゴリズム)		
S O R T I N G		910×1820	1000×2000	1220×2440
	DADW	27 1030292 82.0013	26 719455 70.5170	17 1284540 72.4601
	TMAX	27 1496565 82.0013	27 854295 67.9053	18 1284540 68.4345
	PLUS	27 1000543 82.0013	26 719455 70.5170	17 1284540 72.4601
	最適解	27 82.0013	26 70.5170	17 72.4601

表 6 ケース 1 の実験結果

		原板のサイズ(TMAX-sort)		
P A C K I N G		910×1820	1000×2000	1220×2440
	HEI		27 854295 67.9053	18 1284540 68.4345
	LOW	27 1496565 82.0013		18 1284540 68.4345
	RAT		27 854295 67.9053	18 1284540 68.4345
	RT 2		27 854295 67.9053	18 1284540 68.4345
	RT 3	27 1496565 82.0013	27 854295 67.9053	18 1284540 68.4345
	最適解	27 82.0013	26 70.5170	17 72.4601

には、入出力インターフェースの工夫を重ねたり、レスポンス・タイムの短縮化を計ったりする必要がある。

また、今回手がけたのは、与えられた原板にもとづいて注文サイズを処理するという問題であったが、平均的にロスが小さくなるような原板その

ものの寸法を決定するという問題も重要である。

さて、この種の問題は、樹脂建材の切り取りだけでなく、産業界に数多く存在する。いずれも生産効率に多大の影響を与えており、早急に解決が望まれている。それらの問題は少しずつ前提条件が異なっており、一層強力なアルゴリズムの出現と同時にきめ細かな運用手法の出現が不可欠である。

参考文献

- [1] M. R. Garey and D. S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman & Company, San Francisco, 1979.
- [2] E. G. Coffman, Jr. M. R. Garey, D. S. Johnson and R. E. Tarjan: Performance Bounds for Level-oriented Two-dimensional Packing Algorithm, *SIAM J. Comput.*, 9 (1980), pp. 808-826.
- [3] B. S. Baker, D. J. Brown and H. P. Katseff: A 5/4 Algorithm for Two-dimensional Packing, *J. Algorithms* (1981), pp. 348-368.
- [4] B. S. Baker, E. G. Coffman, Jr. and R. L. Rivest: Orthogonal Packing in Two Dimensions, *SIAM J. Comput.*, 94 (1980), pp. 846-855
- [5] F. R. K. Chung, M. R. Garey and D. S. Johnson: On Packing Two-Dimensional Bins, *SIAM J. Alg. Disc. Meth.*, 3 (1982).
- [6] B. Chazelle: The Bottom-Left BinPacking Heuristic: An Efficient Implementation, *IEEE transaction on computers*. Vol. C-32. No. 8 (1983).
- [7] 栢本雄介, 吉川謙司: 二次元パッキング・アルゴ

表 7 ケース 2 実験結果

		原板のサイズ(TMAX-sort)		
		910× 1820	1000× 2000	1220× 2400
P A C K I N G	HE I	24 164650 84.0646		
	LOW		22 1476278 75.9424	16 1791432 70.1564
	RAT		24 1306878 69.6139	
	RT 2	26 1126499 77.5981		
	RT 3	24 164650 84.0646	22 1065116 75.9424	16 411162 70.1564
	最適解	22 91.707	20 83.537	14 80.179

表 8 ケース 2 実験結果(HBM)
910×1820

				HE I	LOW	RT 3
SMAX	23	23	23	879987	850854	664840
	87.7195	87.7195	87.7195			
DHDW	23	23	23	880729	832354	685319
	87.7195	87.7195	87.7195			
1000×2000						
				HE I	LOW	RT 3
SMAX	20	22	21	1433690	671080	822324
	83.5366	75.9424	79.5587			
1220×2440						
				HE I	LOW	RT 3
SMAX	17	18		671080	822324	
	66.0296	62.3613				
DHDW	17	20	20	671080	595549	778699
	66.0296	56.1251	56.1251			

リズムの計算機による性能評価, 神戸商科大学昭和61年度卒業論文。

- [8] 栢本雄介, 吉川謙司, 加藤直樹: 二次元 Strip-packing Algorithm の実際的评价, 1986年オペレーションズ・リサーチ春期研究発表会アブストラクト集。
- [9] 栢本雄介, 吉川謙司, 加藤直樹: 二次元 Bin-Packing Algorithm の実際的评价, 1986年オペレーションズ・リサーチ春期研究発表会アブストラクト集。
- [10] 笠原浩平, 中谷 透: 二次元パッキング・アルゴリズムの計算機による性能評価, 神戸商科大学昭和62年度卒業論文。