

値を生成する関数を $key(\alpha) = \sum_{i=1}^{|\alpha|} num(\alpha, i) * 2^{shift(i)}$ とする.

ある節点 c の計算状況を γ とする. γ 中に含まれるピースの集合を S_1 とし, 入力ピースの集合から S_1 を除いた集合を S_2 とする. ハッシュ表 $Hash$ には (S_2 に対する最適解の有無, パターンの列) という形の値を格納する. ここでパターンの列は最適解が有るときには S_2 の最適解であり, 最適解が無いときは値を登録する時に着目している節点 c の計算状態 γ である. ハッシュ表に値が既に登録されているときは次の手順でハッシュ表を利用または更新する.

- $Hash[key(S_1)]$ が最適解を持つならば, その値を利用する
- $Hash[key(S_1)]$ が最適解を持たないならば
 - $cost(Hash[key(S_1)][1]) \leq cost(\gamma)$ ならば子節点の探索を省略する
 - $cost(Hash[key(S_1)][1]) > cost(\gamma)$ ならハッシュ表に次の値を登録する

着目した節点からの探索において

- 葉まで調べるパスが存在しないとき $Hash[key(S_1)] =$ (最適解無し, γ)
- 葉まで調べるパスが存在する $Hash[key(S_1)] =$ (最適解有り, S_2 に対する最適解)

5 実装と実験結果

提案手法以外にもいくつかの改良を加えたアルゴリズムを C# 言語を用いて実装し計算速度を計測した. Intel Core2Quad プロセッサ Q9450 を搭載したコンピュータを用いて野呂らの研究と比較した. ここでは株式会社鈴工より提供された実際に木造住宅を建築する際に切り出すときに必要とされるビン (材木) とピース (柱) のデータを使用した. 提供されたデータ中のピースの種類と総数の平均値はそれぞれ 23 種類と 175 個であった. ビンは 6 種類あり長さはそれぞれ 6096mm, 5486mm, 4876mm, 4267mm, 3657mm, 3048mm, 2438mm のものである. 木造住宅の建築例 50 例中 26 例について最適解を求めることが出来た.

5.1 最適化アルゴリズムによる実験結果

野呂の手法で計算時間を多く要していたピースの数が 76 個であり種類が 9 種類である入力例 9, ピースの数が 79 個でありピースの種類が 7 種類である入力例 13 ([野呂 09] p.19) に対する計算時間と探索節点の数を表.1 に示す.

入力例	従来手法		提案手法	
	節点数	時間	節点数	時間
9	6300 万	115 秒	2100 万	37 秒
13	760 万	10.3 秒	4 万	0.6 秒

表 1: 実験結果 1

5.2 近似アルゴリズムによる実験結果

野呂らの論文において最適解が得られていない 29 例の入力のうち 19 例について近似度を改善した. 近似度は解のコストを入力ピースの総和で割ったものである. 計算時間は概ね数分程度であった. 近似アルゴリズムによる計算結果の一例を表.2 に示す.

入力例	従来手法の近似度	提案手法の近似度
36	1.016	1.013
39	1.008	1.004
42	1.017	1.012
47	1.021	1.012

表 2: 実験結果 2

6 あとがき

本研究では野呂ら [野呂 09] が提案したパターンクラス概念を拡張するとともにいくつかの新しい手法を導入することにより最適木取り問題の計算時間及び探索節点数を低減し, 実用的な時間内で計算可能な入力の範囲を広げた.

また, 提案手法を野呂らと同様の近似アルゴリズムに組み込むことで従来の近似度を改善した.

残された課題としてはビンの長さに対して小さいピースが多く存在するような入力に対して計算時間が多くかかるなど, 入力によってはピースの種類や数が少ない場合であっても計算時間が非常に大きくなる問題があり, そのような入力に対しても効率的に計算可能なアルゴリズムに改善していくことがあげられる.

参考文献

- [FL86] D. K. Friesen and M. A. Langston. Variable sized bin packing. In *SIAM J. COMPUT* Vol.15, pp. 222–229, 1986.
- [野呂 09] 野呂耕三. 建築パーツ切り出しのための可変サイズビンパッキングアルゴリズムについて. 修士論文, 三重大学大学院, 2009.