

ナップサック問題については、容量に関するさらなる枝刈り条件や、分割ナップサック問題を用いた枝刈り条件など、さらに洗練された枝刈り条件を用いることが可能である。これらについて興味のある人は、巻末の「さらなる勉強のために」で紹介する書籍や最適化手法に関する書籍を参照してほしい。

第9章のポイント

1. バックトラック法とは、解の列挙が必要な問題に対して効率良く系統的に解を列挙しチェックを行う方法である。このバックトラック法における解の列挙は、列挙木とよばれる木を用いて表される。
2. 入力サイズが n の部分和问题に対するバックトラック法を用いたアルゴリズムの時間計算量は、 $O(n2^n)$ である。
3. 分枝限定法はバックトラック法に加えて用いられる手法であり、枝刈りという操作により不必要な列挙の操作を行わないようにする手法である。ここで枝刈りとは、バックトラック法により作成される列挙木の各節点が表す選択肢において、暫定解などからその選択により問題の出力となる解が得られないと判定される場合に、それ以上の列挙の操作を中止し、後戻りする操作である。
4. 部分和问题や0-1 ナップサック問題に対する分枝限定法を用いたアルゴリズムの最悪時間計算量は、漸近的には分枝限定法を用いないバックトラック法と同じである。しかし、分枝限定法を用いたアルゴリズムの実際の実行時間は、単なるバックトラック法のアルゴリズムと比較して、非常に高速である。

演習問題

9.1 以下の文章の①～③について、それぞれ正しい記号を下から選べ。正しい記号が複数存在する場合はすべて列挙せよ。

バックトラック法は(①)というアルゴリズムである。また、分枝限定法は、バックトラック法に(②)という性質を追加した方法である。この分枝限定法では(③)。

- ①: a. すべての解を効率良く列挙する
 b. アルゴリズムの実行途中において全体的なことは考えず、局所的に最良の解を選択する
 c. 入力をいくつかの部分問題に分割し、各部分問題を再帰的に解く
 d. 問題を部分問題から解き、その解を記録しておいて再利用する
- ②: a. 解の列挙をさらに増やす
 b. 不必要な解の列挙を省略する
 c. 入力そのものを限定する
 d. 近似的な解を求める
- ③: a. 入力によりアルゴリズムの実行時間は大きく異なる
 b. どのような入力に対してもアルゴリズムの実行時間は同じである
 c. 枝刈りの条件の決め方により実行時間が変化する
 d. 暫定解を記録しなくても実行時間は変わらない

9.2 重さが 0 より大きく W より小さい n 個の荷物があり、これらの荷物の重さを w_1, w_2, \dots, w_n とする。また、容量が W の袋が 3 つあるものとする。この n 個の荷物を 3 つの袋にすべて詰めたいのだが、そのような詰め方があるかどうかを判定するアルゴリズムをバックトラック法を用いて作成せよ。

9.3 A 君がスーパーにバーベキュー用の肉を買いに行ったところ、スーパーの開店記念ということで、以下の 5 種類の肉の中からちょうど 8 kg になるように選べたら、肉を無料でもらえるというキャンペーンが開催されていた。

- ① 牛カルビ(6 kg) ② 牛ロース(2.5 kg) ③ 牛ハラミ(5 kg)
④ 地鶏(4 kg) ⑤ 豚カルビ(3 kg)

どのように選べば、肉の総重量が 8 kg になるかを分枝限定法で求めよ。

9.4 運送業者が美術館から以下の 4 種類の絵を運搬するように依頼を受けた。それぞれの絵の重さと価値は表 9.2 のとおりである。

表 9.2

| 番号 | 種類 | 重さ(kg) | 価格(百万円) | 単位重さあたりの価値(百万円/kg) |
|----|-----|--------|---------|--------------------|
| 1 | 絵 1 | 16 | 24 | 1.5 |
| 2 | 絵 2 | 20 | 22 | 1.1 |
| 3 | 絵 3 | 24 | 12 | 0.5 |
| 4 | 絵 4 | 30 | 12 | 0.4 |

運送業者のもっている運搬ケースは絵を運ぶために十分大きいのだが、50 kg より多くの物を入れると壊れてしまう。どのように絵を運搬すれば運ぶ絵の価値が最大になるかを分枝限定法で求めよ。