

1. まえがき

ナップサック問題とは、容量 M のナップサックがひとつと、 N 個の荷物（それぞれ、重さ W_i 、価値 V_i ）が与えられたとき、ナップサックの容量 M を超えない範囲でいくつかの荷物をナップサックに詰め、ナップサックに入れた荷物の価値の総和を最大にする問題である。NP 困難であることが知られている。

2. 研究方法

2.1.1. C 言語によるプログラムの作成

2.1.2. 荷物の重さと価値はランダム（平均がナップサックの最大容量の 2 倍になるように）で発生させ、動的計画法、貪欲法ともに同じ入力で実験。荷物の個数とナップサックの最大容量をいろいろと変化させて行う

2.1.3. 価値の総和と実行時間を調べ比較

2.2. 動的計画法

容量「 w 」のナップサックに「 i 」番目の荷物まで考慮したときに入る荷物の価値の総和の最大値を $C[i][w]$ とする。

・ $W_i > w$ のとき $C[i][w] = C[i-1][w]$

・ $W_i \leq w$ のとき

$$C[i][w] = \max(C[i-1][w], C[i-1][w - W_i] + V_i)$$

2.3. 貪欲法

各荷物の単位重さあたりの価値の大きい順に、ナップサックがいっぱいになるまで

荷物を詰め込む。

3. 結果(上段：動的計画法 下段：貪欲法)

3.1 価値の合計

	M=10	M=100	M=1000	M=10000
N=10	62.17	199.16	876.84	26674
	46.74	162.36	627.69	16746
N=100	133.29	1281.4	16486	*
	127.72	1237.8	15687	*
N=1000	272.68	5572.8	*	*
	271.18	5494.7	*	*
N=10000	397.93	*	*	*
	397.93	*	*	*

3.2 処理時間

(単位：秒)

	M=10	M=100	M=1000	M=10000
N=10	0.000	0.000	0.002	0.019
	0.000	0.001	0.000	0.000
N=100	0.002	0.000	0.010	*
	0.003	0.003	0.001	*
N=1000	0.003	0.011	*	*
	0.040	0.038	*	*
N=10000	0.033	*	*	*
	3.741	*	*	*

4. 考察

解の精度は動的計画法の方が良く（ほぼ最適解）、処理時間は貪欲法の方が速いと予想されていたが、プログラムの設計上、一部予想とは違った結果が得られた。解の精度に関しては、荷物の個数を増やすことで単位重さあたりの価値の高い荷物が発生しやすくなったことによるもの、処理時間に関しては、貪欲法のプログラム中の N の 2 重ループによるものだと考えられる。