

もうひとつ、以下の問題についても、Partition からの reduction を考えることができる (Korte and Vygen [4, p. 442] を参照).[†]

Knapsack

入力: c_1, c_2, \dots, c_n および w_1, w_2, \dots, w_n ならびに W なる $2n+1$ 個の正の整数. ただし, $\sum_j w_j > W$ とする.

出力: $S \subset \{1, 2, \dots, n\}$ とする. $\sum_{j \in S} w_j \leq W$ なる制約条件の下で $\sum_{j \in S} c_j$ を最大にする S .

ここに、Partition からの reduction を $w_j := c_j$ ($j = 1, 2, \dots, n$), $W := \lfloor \sum_j c_j / 2 \rfloor$ として構成する. この時、 $\sum_{j \in S} c_j$ の最大値が $\lfloor \sum_j c_j / 2 \rfloor$ ならば、まさにその S が、Partition において $\sum_{j \in S} c_j = \sum_{j \notin S} c_j$ を実現する. しかし Max-Cut と同様に、 $\lfloor \sum_j c_j / 2 \rfloor$ はいくらでも大きくできるので、Knapsack についても、ここでの議論は残念ながらあてはまらない. 実際、Knapsack には全多項式時間近似スキーム (FPTAS) が提案されており、FPTAS の近似率はいくらでも 1 に近づけることができる (例えば [4, 第 17.3 節] を参照). Knapsack への効率の良い FPTAS の研究は今も続いている. 例えば、Kellerer and Pferschy [3] を参照されたい.

ここで、つけ加えておきたい. 拙稿 [2] では、machine scheduling に関して、dedicated と identical が、あたかも同じ概念であるかのように記述されているが、dedicated とは、各ジョブが実行されるマシンがあらかじめ決まっているということであり、identical (複数のマシンがあり、ジョブの実行時間がマシンに依存しない) とは異なる. 両者を混同してしまったことを、ここに、お詫びして訂正する.

identical parallel machines の近似率が dedicated parallel machines と同じく 1.5 より良くならないことは、Grigoriev et al [1] の証明 (定理 3., p. 193) ほぼそのまま示せる. 以下、蛇足ながら書き留めておく:

[†]Knapsack において $c_j = w_j$ ($j = 1, 2, \dots, n$) としたものを、特に Subset-Sum (部分和问题) という. Subset-Sum については、Knapsack への議論と同様の結果しか得られないので、ここでは言及しない.

各ジョブ $j = 1, 2, \dots, n$ の実行時間を

$$p_{js} := \begin{cases} 1, & s \geq c_j, \\ 3, & s < c_j \end{cases}$$

とし、マシン数 $m := n$ とする (実際には、 n も要らないが). また、 $\sum_j c_j = 2k$ とする. もし、 $\sum_{j \in S} c_j = k$ を満たす $S \subset \{1, 2, \dots, n\}$ があれば、まず S に入るジョブのみを (各々にリソース c_j ちょうどを割り当てて) 一斉に実行し、その終了 (所用時間 1) 後に残りのジョブを一斉に実行すれば、makespan は 2 である; これ以外では、makespan は 3 以上となる. したがって、この identical parallel machines の例について、makespan が 2 か、あるいは 3 以上かを判定できるアルゴリズム (即ち、近似率が 1.5 より良い) は、Partition を解く ($\sum_{j \in S} c_j = \sum_{j \notin S} c_j$ なる S の存在を判定する) アルゴリズムとなる.

参考文献

- [1] A. Grigoriev, M. Sviridenko and M. Uetz, Unrelated parallel machine scheduling with resource dependent processing times. In: M. Jünger and V. Kaibel (Eds.) 11th IPCO Proceedings, LNCS 3509, pp. 182–95, Springer-Verlag, Berlin Heidelberg 2005 [doi:10.1007/11496915-14].
- [2] 飯田, Partition のある風景. Discussion paper series No. 115, 小樽商科大学ビジネス創造センタ, 2008 年 6 月; (<http://hdl.handle.net/10252/918>) から入手可能.
- [3] H. Kellerer and U. Pferschy, Improved dynamic programming in connection with an FPTAS for the knapsack problem. Journal of Combinatorial Optimization **8**(1) 5–11 (March 2004) [doi:10.1023/B:JOCO.0000021934.29833.6b].
- [4] B. Korte and J. Vygen, Combinatorial Optimization: Theory and Algorithms (Algorithms and Combinatorics **21**), 4th edition, Springer-Verlag, Berlin Heidelberg 2008 [doi:10.1007/978-3-540-71844-4].