Estimating the efficiency of backtrack procedures for nonnegative integer programming problems

Marcos Daniel Baroni

September 24, 2014

During the last decades much effort has been devoted to the search of efficient algorithms to solve some well-known NP-hard problems Those efforts have led to substantial improvement on heuristics and approximate approaches, but no efficient algorithm has been found yet and exponential time algorithms is still needed for exact solution.

Despite the exponential time behavior expected for worst cases, empirical observations has reported efficient time on average for backtrack algorithms for several problems [1, 7, 5, 3, 6]. An intersting problem presenting this intriguing behavior (and moreover generalizes other several NP-complete problems) is the multidimensional knapsack problem (MKP).

The MKP may be defined as follows:

$$\begin{aligned} \text{maximize} & \sum_{j=1}^{n} p_j x_j \\ \text{subject to} & \sum_{j=1}^{n} w_{ij} x_j \leq b_i \quad i \in \{1, \ldots, m\} \\ & x_j \in \{0, 1\}, \quad j \in \{1, \ldots, n\}. \end{aligned}$$

The MKP can be considered as a generalization of other three well-known NP-Complete problems [2]:

- KNAPSACK PROBLEM (KP): special case of MKP where $m = 1$;
- SUBSET SUM PROBLEM (SSP): special case of KP where $p_j = w_j$;
- PARTITION PROBLEM (PP): special case of SSP where $b = \lfloor \frac{1}{2} \sum_{i=1}^{n} w_i \rfloor$.

All four problems above are NP-complete problems and they will be considered in this work as Standard Nonnegative Binary Programing Problems (SNBPPs). Efficient average time on backtrack algorithms has been observed in practice for those problems but few related theoretical foundation is known.

Dynamic programming is an exact approach which, at first, sounds most attractive for SNBPPs since it promises polynomial time for instances with coefficients bounded by a constant. However:

(a) all theoretical results proving NP-Completeness for SNBPPs presents instances with exponential coefficients;

(b) eventually small instances having large coefficients (scaled instances from real problems with rational coefficients, for example) may appear for which a backtrack algorithm is more suitable;

(c) backtrack algorithms are used for others combinatorial problems.

For those reasons the analysis of backtrack procedures seems to be of great relevance.

The main goal of the work is to investigate the expected efficiency of backtrack procedures for SNBPPs. This will be done by considering at first simpler versions of backtrack algorithms. Afterwards more efficient (hence more complex) versions can be analyzed on a step-by-step manner.

The paper by Donald Knuth [4] seems to give helpful directions on how to perform the analysis.

References

[1] P. Cheeseman, B. Kanefsky, and W. Taylor. Where the really hard problems are. In *Proceedings of IJCAI*, volume 91, pages 163–169, 1991.

[2] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. 1979.

[3] D. S. Johnson. The np-completeness column: an ongoing guide. *Journal of Algorithms*, 5(2):284–299, 1984.

[4] D. E. Knuth. Estimating the efficiency of backtrack programs. *Mathematics of computation*, 29(129):122–136, 1975.

[5] L. Pósa. Hamiltonian circuits in random graphs. *Discrete Mathematics*, 14(4):359–364, 1976.

[6] P. W. Purdom Jr. Search rearrangement backtracking and polynomial average time. *Artificial intelligence*, 21(1):117–133, 1983.

[7] H. S. Wilf. Backtrack: an o (1) expected time algorithm for the graph coloring problem. *Information Processing Letters*, 18(3):119–121, 1984.