# Enumerations Methods for Repeatedly Solving Multidimensional Knapsack Sub-Problems

University of Canterbury Ross J.W. James
01402374 Kansai University *Yuji Nakagawa

## 1. Introduction

We present techniques for enumerating a sub-problem of the Nonlinear Knapsack problem that can then be combined with other optimisation techniques in order to solve large Nonlinear or Multidimensional knapsack problems.

The Multidimensional Non-linear Knapsack Problem can be defined as:

$$\text{maximize} \quad f(\mathrm{x}) = \sum_{i=1}^{n} f_i(x_i)$$

$$\text{subject to} \quad g_j(\mathrm{x}) = \sum_{i=1}^{n} g_{ji}(x_i) \le b_j \quad \text{for} \quad i \in M$$

$$x_i \in K_i \quad \text{for} \quad i \in N$$

where $\mathrm{x} = (x_1, x_2, ..., x_n)$, $M = \{1,2,...,m\}$, $K_i = \{1,2,...,k_i\}$, and $N = \{1,2,...,n\}$.
If $k_i = 2 \ \forall\, i$, then the problem reduces to a 0/1 Multidimensional Knapsack problem. Both the Non-linear and Multidimensional knapsack problems are known to be NP-hard and therefore heuristics are primarily used when solving problems where the number of items is large.

Nakagawa (1990) proposed an approach to solving Non-linear Knapsack Problems using what was termed the "Modular Approach" (MA). Essentially this approach is an extension of a hybrid Dynamic Programming/Branch and Bound method. It solves problems by repeatedly fathoming the current decision space, as in a branch and bound algorithm eliminating branches that will not meet a preset target value. This reduces the size of the decision space. It then integrates two variables into a new variable, reducing the overall number of variables in the system, similar to Dynamic Programming where the new variable represents the state space in the network at a particular level.

The problem that is considered in this paper solves the 0/1 Multidimensional Knapsack problem repeatedly as a sub-problem of a Non-linear Knapsack problem that has all 0/1 variables except for one variable, the last variable, that has many different possible states with varying objective and constraint requirements.

## 2. Computational Experience

### Problem Data

We carried out the approach using the Multi-dimensional Knapsack test problems published by Chu and Beasley (1998) and available via the world wide web from the OR library (http://mscmga.ms.ic.ac.uk/jeb/orlib/mknapinfo.html). Problems with a size (number of variables x number of constraints) 100x5, 250x5 and 100x10 were solved optimally by Chu and Beasley (1998) using a CPLEX solver. We therefore chose to focus on the next set of 30 problems with 500 variables and 5 constraints. The optimal solutions to these problems were previously unknown.

### Computational Results

All problems were solved on a combination of two machines. The MA was run on a Fujitsu Unix Workstation with 18G RAM for generating the final problem (as the state spaces were generally below 1,000,000 elements). The enumeration was carried out on Gateway PCs with 1.7Ghz Intel Pentium 4 processors and a minimum of 128MB memory running Windows Me. The same computer was used to process the same problem for both enumeration techniques. We used both (1) the Constraint Based approach and (2) the State Search approach to enumerate the solution spaces in order to assess how these approaches compared.

Table 1 shows the optimal values of the 500 variable, 5 constraint problems of Chu and Beasley (1998). The table shows the problem number, the Optimal objective value, the value previously found by Chu and Beasley (C&B) and the percentage difference of the Chu and Beasley solution from the optimal. The next two columns of the table show how many states were in the last variable for the final MA problem that needed to be enumerated and the total number of variables in the problem. The next two columns show the number of hours required to enumerate the states using the Constraint Based approach and the State Search approach respectively. The final column shows the percentage improvement of the State Search approach over the Constraint Based Approach. Where there are no entries in the last five columns indicates that the MA was able to solve the final problem completely without the aid of either of the enumeration techniques. In all cases the enumerations were started with a lower bound of zero. This was in order that the times shown would be worst-case scenarios. If the best-known values were used, then clearly the processing times would improve.

| Problem | Objective | | % of C&B | Final Problem from MA | | Constraint | State | % |
|---|---|---|---|---|---|---|---|---|
| Number | Optimal | C&B | from Optimal | States | Variables | Time (Hrs) | Time (Hrs) | Difference |
| 0 | 120148 | 120130 | 0.0150% | 844703 | 25 | 16.81 | 10.21 | 39.3% |
| 1 | 117879 | 117837 | 0.0356% | 802008 | 17 | 0.19 | 0.10 | 45.3% |
| 2 | 121131 | 121109 | 0.0182% | 692654 | 22 | 6.07 | 2.33 | 61.6% |
| 3 | 120804 | 120798 | 0.0050% | 859695 | 20 | 3.88 | 1.52 | 60.9% |
| 4 | 122319 | 122319 | 0.0000% | 565439 | 23 | 1.25 | 1.39 | -11.8% |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| 25 | 302571 | 302560 | 0.0036% | 712883 | 17 | 0.24 | 0.09 | 63.6% |
| 26 | 301339 | 301322 | 0.0056% | | | | | |
| 27 | 306454 | 306430 | 0.0078% | | | | | |
| 28 | 302828 | 302814 | 0.0046% | | | | | |
| 29 | 299910 | 299904 | 0.0020% | 992552 | 21 | 4.44 | 3.43 | 22.8% |

Table 1. Optimal Solutions to Chu and Beasley (1998) 500 variable, 5 constraint problems

From the table we can conclude that computational times for each problem varied greatly, depending on the number of variables in the problem and the problem itself. The differences in the computational times between the two searches are also of interest. In most cases the State Search approach clearly dominates the Constraint Based Search with computational times that are significantly faster. The average percentage speed increase of the State Search Approach over the Constraint Based Approach is 42%.

## References

P.C. Chu and J.E. Beasley (1998), “A Genetic Algorithm for the Multidimensional Knapsack Problem”, *Journal of Heurisitics*, 6 : 63-86

Y. Nakagawa (1990), “A New Method for Discrete Optimization Problems”, *Electronics and Communications in Japan*, 73 : 99-106.

Y. Nakagawa and A. Iwasaki (1999), “Modular Approach for Solving Nonlinear Knapsack Problems”, *ICICE Transaction Fundamentals*, E82-A : 1860-1864

**Acknowledgements**

This research was carried out while the first author was at Kansai University as an invited researcher.