Hans Kellerer, Ulrich Pferschy, David Pisinger

# Knapsack Problems

Springer's internal project number, if known

Mathematics – Monograph (English)

October 10, 2003

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

# Preface

Thirteen years have passed since the seminal book on knapsack problems by Martello and Toth appeared. On this occasion a former colleague exclaimed back in 1990: “How can you write 250 pages on the knapsack problem?” Indeed, the definition of the knapsack problem is easily understood even by a non-expert who will not suspect the presence of challenging research topics in this area at the first glance.

However, in the last decade a large number of research publications contributed new results for the knapsack problem in all areas of interest such as exact algorithms, heuristics and approximation schemes. Moreover, the extension of the knapsack problem to higher dimensions both in the number of constraints and in the number of knapsacks, as well as the modification of the problem structure concerning the available item set and the objective function, leads to a number of interesting variations of practical relevance which were the subject of intensive research during the last few years.

Hence, two years ago the idea arose to produce a new monograph covering not only the most recent developments of the standard knapsack problem, but also giving a comprehensive treatment of the whole knapsack family including the siblings such as the subset sum problem and the bounded and unbounded knapsack problem, and also more distant relatives such as multidimensional, multiple, multiple-choice and quadratic knapsack problems in dedicated chapters.

Furthermore, attention is paid to a number of less frequently considered variants of the knapsack problem and to the study of stochastic aspects of the problem. To illustrate the high practical relevance of the knapsack family for many industrial and economic problems, a number of applications are described in more detail. They are selected subjectively from the innumerable occurrences of knapsack problems reported in the literature.

Our above-mentioned colleague will be surprised to notice that even on the more than 500 pages of this book not all relevant topics could be treated in equal depth but decisions had to be made on where to go into details of constructions and proofs and where to concentrate on stating results and refer to the appropriate publications. Moreover, an editorial deadline had to be drawn at some point. In our case, we stopped looking for new publications at the end of June 2003.

The audience we envision for this book is threefold: The first two chapters offer a very basic introduction to the knapsack problem and the main algorithmic concepts to derive optimal and approximate solution. Chapter 3 presents a number of advanced algorithmic techniques which are used throughout the later chapters of the book. The style of presentation in these three chapters is kept rather simple and assumes only minimal prerequisites. They should be accessible to students and graduates of business administration, economics and engineering as well as practitioners with little knowledge of algorithms and optimization.

This first part of the book is also well suited to introduce classical concepts of optimization in a classroom, since the knapsack problem is easy to understand and is probably the least difficult but most illustrative problem where dynamic programming, branch-and-bound, relaxations and approximation schemes can be applied.

In these chapters no knowledge of linear or integer programming and only a minimal familiarity with basic elements of graph theory is assumed. The issue of $\mathcal{NP}$-completeness is dealt with by an intuitive introduction in Section 1.5, whereas a thorough and rigorous treatment is deferred to the Appendix.

The remaining chapters of the book address two different audiences. On one hand, a student or graduate of mathematics or computer science, or a successful reader of the first three chapters willing to go into more depth, can use this book to study advanced algorithms for the knapsack problem and its relatives. On the other hand, we hope scientific researchers or expert practitioners will find the book a valuable source of reference for a quick update on the state of the art and on the most efficient algorithms currently available. In particular, a collection of computational experiments, many of them published for the first time in this book, should serve as a valuable tool to pick the algorithm best suited for a given problem instance. To facilitate the use of the book as a reference we tried to keep these chapters self-contained as far as possible.

For these advanced audiences we assume familiarity with the basic theory of linear programming, elementary elements of graph theory, and concepts of algorithms and data structures as far as they are generally taught in basic courses on these subjects.

Chapters 4 to 12 give detailed presentations of the knapsack problem and its variants in increasing order of structural difficulty. Hence, we start with the subset sum problem in Chapter 4, move on to the standard knapsack problem which is discussed extensively in two chapters, one for exact and one for approximate algorithms, and finish this second part of the book with the bounded and unbounded knapsack problem in Chapters 7 and 8.

The third part of the book contains more complicated generalizations of the knapsack problems. It starts with the multidimensional knapsack problem (a knapsack problem with $d$ constraints) in Chapter 9, then considers the multiple knapsack problem ($m$ knapsacks are available for packing) in Chapter 10, goes on to the multiple-choice knapsack problem (the items are partitioned into classes and exactly one item of each class must be packed), and extends the linear objective function to a quadratic one yielding the quadratic knapsack problem in Chapter 12. This chapter also contains an excursion to semidefinite programming giving a mostly self-contained short introduction to this topic.

A collection of other variants of the knapsack problem is put together in Chapter 13. Detailed expositions are devoted to the multiobjective and the precedence constraint knapsack problem, whereas other subjectively selected variants are treated in a more cursory way. The solitary Chapter 14 gives a survey on stochastic results for the knapsack problem. It also contains a section on the on-line version of the problem.

All these six chapters can be seen as survey articles, most of them being the first survey on their subject, containing many pointers to the literature and some examples of application.

Particular effort was put into the description of interesting applications of knapsack type problems. We decided to avoid a boring listing of umpteen papers with a two-line description of the occurrence of a knapsack problem for each of them, but selected a smaller number of application areas where knapsack models play a prominent role. These areas are discussed in more detail in Chapter 15 to give the reader a full understanding of the situations presented. They should be particularly useful for teaching purposes.

The Appendix gives a short presentation of $\mathcal{NP}$-completeness with the focus on knapsack problems. Without venturing into the depths of theoretical computer science and avoiding topics such as Turing machines and unary encoding, a rather informal introduction to $\mathcal{NP}$-completeness is given, however with formal proofs for the $\mathcal{NP}$-hardness of the subset sum and the knapsack problem.

Some assumptions and conventions concerning notation and style are kept throughout the book. Most algorithms are stated in a flexible pseudocode style putting emphasis on readability instead of formal uniformity. This means that simpler algorithms are given in the style of an unknown but easily understandable programming language, whereas more complex algorithms are introduced by a structured, but verbal description. Commands and verbal instructions are given in Sans Serif font, whereas comments follow in *Italic* letters. As a general reference and guideline to algorithms we used the book by Cormen, Leiserson, Rivest and Stein [92].

For the sake of readability and personal taste we follow the non-standard convention of using the term *increasing* instead of the mathematically correct *nondecreasing* and in the same way *decreasing* instead of *nonincreasing*. Whereever we use the log function we always refer to the base 2 logarithm unless stated otherwise. After the Preface we give a short list of notations containing only those terms which are used throughout the book. Many more naming conventions will be introduced on a local level during the individual chapters and sections.

As mentioned above a number of computational experiments were performed for exact algorithms. These were performed on the following machines:

| | | |
|---|---|---|
| AMD Athlon, 1.2 GHz | SPECint2000 = 496 | SPECfp2000 = 417 |
| Intel Pentium 4, 1.5 GHz | SPECint2000 = 558 | SPECfp2000 = 615 |
| Intel Pentium III, 933 MHz | SPECint2000 = 403 | SPECfp2000 = 328 |

The performance index was obtained from SPEC (`www.specbench.org`). As can be seen the three machines have reasonably similar performance, making it possible to compare running times across chapters. The codes have been compiled using the GNU project C and C++ Compiler gcc-2.96, which also compiles Fortran77 code, thus preventing differences in computation times due to alternative compilers.

## Acknowledgements

The authors strongly believe in the necessity to do research not with an island mentality but in an open exchange of knowledge, opinions and ideas within an international research community. Clearly, none of us would have been able to contribute to this book without the innumerable personal exchanges with colleagues on conferences and workshops, in person, by e-mail or even by surface mail. Therefore, we would like to start our acknowledgements by thanking the global research community for providing the spirit necessary for joint projects of collection and presentation.

The classic book by Silvano Martello and Paolo Toth on knapsack problems was frequently used as a reference during the writing of this text. Comments by both authors were greatly appreciated.

To our personal friends Alberto Caprara and Eranda Cela we owe special thanks for many discussions and helpful suggestions. John M. Bergstrom brought to our attention the importance of solving knapsack problems for all values of the capacity. Jarl Friis gave valuable comments on the chapter on the quadratic knapsack problem. Klaus Ladner gave valuable technical support, in particular in the preparation of figures.

In the computational experiments and comparisons, codes were used which were made available by Martin E. Dyer, Silvano Martello, Nei Y. Soma, Paolo Toth and John Walker. We thank them for their cooperation. Anders Bo Rasmussen and Rune Sandvik deserve special thanks for having implemented the upper bounds for the quadratic knapsack problem in Chapter 12 and for having run the computational experiments with these bounds. In this context the authors would also like to acknowledge DIKU Copenhagen for having provided the computational facilities for the computational experiments.

Finally, we would like to thank the Austrian and Danish tax payer for enabling us to devote most of our concentration on the writing of this book during the last two years. We would also like to apologize to the colleagues of our departments, our friends and our families for having neglected them during this time. Further apologies go to the reader of this book for any errors and mistakes it contains. These will be collected at the web-site of this book at `www.diku.dk/knapsack`.

# Table of Contents