Junior Level Self-Teaching Web-Book for

Optimization Models For Decision Making: Volume 1

Katta G. Murty
Dept. Industrial & Operations Engineering
University of Michigan, Ann Arbor
Mi-48109-2117, USA
Phone: 734-763-3513, Fax: 734-764-3451
e-Mail: murty@umich.edu
URL: http://www-personal.engin.umich.edu/~murty/

Contents

This is Chapter 0 of "Junior Level Web-Book for Optimization Models for decision Making" by Katta G. Murty.

PREFACE

Importance of Decision Making Skills for Engineering and Business Professionals

The daily work of an engineering or a business professional involves making a series of decisions. In fact, the human world runs on systems designed by engineers and business people. That's why the quality of decisions made by these two professionals is of critical importance to the health of the world we live in, and should be of great concern to every human being.

These decisions are made by looking at the relevant data and making a manual judgement, usually without the help of quantitative analysis based on an appropriate mathematical model; that's why we can call this the "manual method of making decisions".

Making decisions on issues with important consequences has become a highly complex problem due to the many competing forces under which the world is operating today, and the manual method very often leads to decisions quite far from being optimal. In fact many bad decisions are being made daily due to this.

Many companies have become aware of this problem, and have made efforts to use mathematical models for decision making, and even spent considerable sums of money to acquire software systems to solve these models. However, often the software sits unused because the people who make the decisions are not trained in using it properly. Or, the results obtained by the software may not be practical due to an inappropriate mathematical model being used for the problem. **Intelligent modeling is essential to get good results.** After a disappointing experience with modeling, companies usually go back to their traditional practice of manual decision making.

This points out the importance of developing good mathematical modeling skills in engineering and business students. Some knowledge of algorithms used to solve these models, their implementations, how they work, and their limitations, is equally important in order to make the best use of the output from them. That's why mathematical modeling, computational, and algorithmic skills are very important for engineering and business students today, so that they can become good decsion makers.

Books in Self-Teaching Style

Present day undergraduate population in engineering and business schools find textbooks with a theoretical flavor unappealing as they do not help them acquire the important "mathematical modeling" skill. Also, students are demanding books that discuss the intricacies in applying the methods successfully, in a "self-teaching" style that they can use to learn the subject and its applications mostly by themselves. They want the textbook designed to help carry out a major portion of the learning process by her/himself outside the classroom.

As an example of the possibility of self-learning, I can mention the following historical incident. This incident is the strangest of "mathematical talks" ever given. It took place in October 1903 at the American Mathematical Society meeting. The "speaker" was Professor Cole, and the title of the talk was "Mersenne's claim that $a = 2^{67} - 1$ is a prime number".

This claim has fascinated mathematicians the world over since the 1600s. Not a single word was spoken at the seminar by anyone including the speaker. He started by writing on the blackboard $a =$ 147573952589676412927. Then he wrote 761838257287, and underneath it 193707721. Without opening his mouth, he multiplied the two numbers by hand to get a, and then sat down.

Everyone there instantly understood the result obtained by Cole, and the very short and silent seminar ended with a wild applause from the audience.

The Purpose of this Book

The purpose of this book is to serve as a text for developing the mathmatical modeling, computational, and algorithmic skills of optimization, and some of their elementary applications at the junior level following a linear algebra course.

It has many modeling and numerical examples and exercises to illustrate the use of introductory level modeling techniques, how the algorithms work and the various ways in which they can terminate, the types of problems to which they are applicable, what useful planning conclusions can be drawn from the output of the algorithm, and the limitations of these models and algorithms. Hopefully the many worked out examples and illustrations, and simple explanations, make it possible to study and understand most of the material by oneself, and the rest with occasional help from the instructor.

The wide variety and large number of exercises in the book help the students develop problem solving skills.

Why Web-Book?

The web-format makes it much easier and convenient to deliver the content of the book to the students directly without any middle-men, and thus at a much lower price compared to the hard copy format. Each chapter is prepared with its separate index and kept in a separate file, this way they need to print only those chapters covered in the course (may not need to print the whole book). Among the end of chapter exercises in each chapter, only the ones most likely to be used frequently are kept in the chapter, the rest are included in a final chapter called "Additional Exercises", arranged chapterwise. Also, when students print something, they usually print 8 pages per sheet. These formats help save a lot of paper. In fact some students may prefer reading the book on their screen, but I hope that all will print only the most essential parts, and thus conserve paper for making which we are killing a lot of trees.

Preview

Chapter 1 introduces mathematical modeling using a simple one variable example. This chapter also explains the classification of decision making problems into Category 1, and Category 2.

Chapter 2 discusses MCDM (multi-characteristic decision making) problems. It explains the commonly used Scoring Method for solving

Category 1 decision making problems when there are several important characteristics that need to be optimized simultaneously, with many simple examples.

Chapter 3 deals with elementary modeling techniques for modeling continuous variable decision making problems in which linearity assumptions hold to a reasonable degree of approximation, as linear programs (LPs), in a variety of applications. The geometric method for solving two variable LP models is discussed along with the concept of marginal values and their planning uses.

Chapter 4 discusses the simplest version of the primal simplex method for solving LPs using full canonincal tableaus, which students at this level can follow easily; and explains it with many worked out examples.

Chapter 5 gives the derivation of the dual problem of an LP using economic arguments, and the marginal value interpretation of the dual variables. It discusses the optimality conditions (primal and dual feasibility, and complementary slackness) for an LP, and the role they play in the simplex method. Marginal analysis, and a few important coefficient ranging and sensitivity analysis techniques are also discussed.

Chapter 6 treats the simplified version of the primal simplex algorithm for the transportation model using transportation arrays.

Chapter 7 presents techniques for modeling integer and combinatorial optimization problems. It shows that many different combinatorial constraints that appear frequently in applications, can be modeled using linear constraints in binary variables. The importance of 0-1 integer programming models is highlighted with interesting examples drawn from puzzle literature and the classics, which students at this age find very engaging.

Chapter 8 discusses the branch and bound approach for solving integer and combinatorial optimization problems, and its advaltages and limitations.

The amount of computer time needed for solving discrete and combinatorial optimization problems with branch and bound or other exact methods available today grows rapidly as problem size increases. So, at present it is practical to solve only moderate sized problems of this type exactly. Consequently, when faced with large scale versions of these problems, most practitioners use heuristic approaches to obtain the best possible approximate solution within a reasonable time. Surprisingly, well designed heuristic methods seem to produce satisfactory solutions to many hard and complex problems. So, heuristic methods are now mainstream for decision making, and the exact methods developed in theory have become tools for designing good heuristics. Chapter 9 discusses the principles for designing good heuristic methods (greedy methods, local search methods, simulated annealing, genetic algorithms) for different problems with many examples.

Chapter 10 explains the recursive technique for solving deterministic dynamic programming problems. Chapter 11 deals with the very important critical path methods for project scheduling and management, using the dynamic programming algorithm for optimal chains in networks.

There is a wide gulf between the mathematical models for solving which we have efficient algorithms, and real world decision making problems. The brief Chapter 12 explains how heuristic approaches, approximations, substitute objective function techniques, and intelligent modeling techniques are helping to bridge this wide gap.

Finally the last chapter, Chapter 13, contains additional end of the chapter exercises for earlier chapters.

Contributions Requested

No funding could be obtained for my effort in preparing this book. Also, everyone who uses this web-book, saves the cost of buying an expensive paper-book containing this material. I request each such user to honestly contribute about US$15 (or more if you like) of the amount you save to my address:

Professor Katta G. Murty
Department of Industrial and Operations Engineering
University of Michigan
Ann Arbor, MI-48109-2117, USA

to partly compensate for my time in preparing it. The money received will be used to maintain and make improvements in this website, and in preparing Volume 2 of this book at the Master's level.

If you are a faculty member using this book in a course, please encourage your students to contribute. In your first class you may select a student to collect from everyone in the class, and then mail the amount collected to my above address.

Numbering Scheme for Equations, Exercises, Etc.

Equations, results, theorems, some examples and tables, within-section exercises, are all numbered serially in each section; so an entity like this with number $i.j.k$ refers to the kth of this entity in Section $i.j$.

End of the chapter exercises at the end of each chapter are nubered serially in each chapter. So Exercise $i.j$ referes to the jth exercise at the end of Chapter i. Similarly figures are numbered serially in each chapter, so Figure $i.j$ refers to the jth figure in Chapter i.

References

Exercises based on material discussed in published papers from journals are included in several chapters. In some of these chapters the reference is cited right at the end of that exercise. In others where there are a lot of such exercises, these references are listed at the end of the chapter in alphabetical order of the fiirst authors last name.

A selected list of textbooks for further reading is given at the end of Chapter 12.

Each Chapter in a Separate file

In paperbooks all chapters are always put together between two covers and bound. But for a webbook I feel that it is convenient to have each chapter in a separate file to make it easier for users to download. That's why I am putting each chapter as a separate file, beginning with its individual table of contents, and ending with its own index of terms defined in it.

Acknowledgements

Figures, and Suggestions to make material easier to read: On a Marian Sarah Parker Scholarship during Summer 2004, Priti Shah helped me by drawing all the figures in the book. She read Chapters 7 to 12 very carefully and provided several suggestions to make this portion easier to understand by Junior level students. Earlier on another Marian Sarah Parker Sholarship during Summer 2003, Shital Thekdi read Chapters 1 to 6 very carefully and made suggestions for improving the exposition in them. I am grateful to Priti and Shital, and to the University of Michigan Marian Sarah Parker Scholarship Administration for this help.

My heartfelt thanks to A. Ravi Ravindran, and Robert Bordley for providing examples, and exercises in Chapter 2.

Suggestions, corrections, and many other kinds of help have been received from several others too numerous to mention by name, and I express my heartfelt thanks to all of them.

Some portions in this book are revised versions of those in my 1995 book *Operations Research: Deterministic Optimization Models* published by Prentice Hall Inc. I received special permission (PE Reference #104672, dated 12 August 2004) to include these in this book from Pearson Education. I thank them for giving me this permission.

Katta G. Murty
Ann Arbor, MI, December 2005.

Glossary of Symbols and Abbreviations

Equations, results, theorems, some examples and tables, within-section exercises, are all numbered serially in each section; so an entity like this with number $i.j.k$ refers to the kth of this entity in Section $i.j$.

End of the chapter exercises at the end of each chapter are nubered serially in each chapter. So Exercise $i.j$ referes to the jth exercise at the end of Chapter i.

Similarly figures are numbered serially in each chapter, so Figure $i.j$ refers to the jth figure in Chapter i.

Abbreviations in alphabetical order

AOA	Activity-on-arc project network, also called the arrow diagram for the project.
AON	Activity-on-node project network.
B&B	Branch and bound approach or algorithm.
BFS	Basic feasible solution for a linear program.
BV	A branching variable used in the branching operation in a B&B. However, in earlier linear programming chapters, this abbreviation is used for either *basic vector* or *basic variables.*
CP	Candidate problem in a B&B.
CPM	Critical path method for project scheduling.
CS	Complementary slackness optimality conditions for a linear program.
DP	Dynamic programming.
EF(i, j), ES(i, j)	Early finish (early start) times associated with job (i, j) in project scheduling.
FIFO	First in first out strategy for selecting objects from a queue.
GA	Genetic algorithm.
GJ	Gauss-Jordan (pivot step, algorithm).

iff	If and only if.
I/O	Input-output coefficients in a linear program.
IP	Integer program.
LA	Linear algebra.
LB	Lower bound for the minimum objective value in a CP.
LIFO	Last in first out strategy for selecting objects from a queue.
LP	Linear program.
LF(i,j), LS(i,j)	Late finish (late start) times associated with job (i,j) in project scheduling.
MCDM	Multi-characteristic decision making problem.
MDR	Minimum daily requirement for a nutrient in a diet model.
MIP	Mixed integer program.
NLP	Nonlinear programming.
Oc.R	Octane rating of gasoline.
OR	Operations Research.
OVF	Optimum value function in DP.
PC	Pivot column (for a GJ pivot step, or in the simplex method).
POS	A partially ordered set.
PR	Pivot row (for a GJ pivot step, or in the Simplex method).
PMX	Partially matched crossover operation in a GA for permutation or tour problems.
RHS	Right hand side (constants, or vector of constants in an LP).
RO	Relaxed optimum (in the LB strategy in B&B).
SA	Simulated annealing algorithm.
TSP	Traveling salesman problem.
WRT	With respect to.

Symbols dealing with sets:

R^n	The n-dimensional real Euclidean vector space. The space of all vectors of the form $x = (x_1, \ldots, x_n)^T$ (written either as a row vector as here, or as a column vector) where each x_j is a real number.
$\setminus$	Set difference symbol. If D, E are two sets, $D \setminus E$ is the set of all elements of D which are not in E.
$\|\mathbf{F}\|$	Cardinality of the set $\mathbf{F}$

$\in$	Set inclusion symbol. $a \in \mathbf{D}$ means that a is an element of $\mathbf{D}$. $b \notin \mathbf{D}$ means that b is not an element of $\mathbf{D}$
$\subset$	Subset symbol. $\mathbf{E} \subset \mathbf{F}$ means that set $\mathbf{E}$ is a subset of $\mathbf{F}$, i.e., every element of $\mathbf{E}$ is an element of $\mathbf{F}$
$\cup$	Set union symbol
$\cap$	Set intersection symbol
$\emptyset$	The empty set

Symbols dealing with vectors:

$=, \geq, \leq$	Symbols for equality, greater than or equal to, less than or equal to, which must hold for each component in a vector.
$\|\|x\|\|$	Euclidean norm of vector $x = (x_1, \ldots, x_n)$, it is $\sqrt{x_1^2 + \ldots + x_n^2}$. Euclidean distance between two vectors x, y is $\|\|x - y\|\|$.

Symbols dealing with matrices:

(a_{ij})	Matrix with a_{ij} as the general element in it.
x^T, A^T	Transpose of vector x, matrix A.
A^{-1}	Inverse of the nonsingular square matrix A.
$A_{i.}, A_{.j}$	The ith row vector, jth column vector of matrix A.
$\text{rank}(A)$	Rank of a matrix A, same as the rank of its set of row vectors, or its set of column vectors.

Symbols dealing with real numbers:

$\|\alpha\|$	Absolute value of real number α.
$n!$	n factorial.
∞	Infinity.
$\sum$	Summation symbol

Symbols dealing with networks or graphs:

$\mathcal{N}$	The finite set of nodes in a network
$\mathcal{A}$	The set of lines (arcs or edges) in a network
$G = (\mathcal{N}, \mathcal{A})$	A network with node set $\mathcal{N}$ and line set $\mathcal{A}$
(i, j)	An arc (directed line) joining node i to node j
$(i; j)$	An edge (undirected line) joining nodes i and j

Symbols dealing with LPs and IPs:

x_j, x_{ij}, x	x_j is the jth decision variable in an LP or IP. x_{ij} is the decision variable associated with cell (i, j) in an assgnment or a transportation problem, or a TSP. x denotes the vector of these decision variables.
$c_{ij}; c_j, c$	The unit cost coefficient or length or weight of arc (or cell in an array)(i, j) or edge $(i; j)$ is denoted by c_{ij}. c_j is the original cost coefficient of a variable x_j in an LP or IP model. c is the vector of c_{ij} or c_j.
π_i, π	Dual variable associated with the ith constraint in an LP, the vector of dual variables.
$u = (u_i), v = (v_j)$	Vectors of dual variables associated with rows, columns of a transportation array.
$\bar{c}_j$, $\bar{c}_{ij}, \bar{c}$	The reduced or relative cost coefficient of variables x_j, x_{ij} in an LP, or the transportation problem. $\bar{c}$ is the vector of these relative cost coefficients.
n, m	Usually, number of variables, constraints in an LP or IP. Also, the number of sinks (columns in a transportation array), and the number of sources (rows in the transportation array) in a transportation problem. The symbol n also denotes the number of cities in a TSP.
a_i, b_j	In a transportation problem, these are the amounts of material available for shipment at source i, required at sink j respectively.
θ	Usually the minimum ratio in a pivot step in the simplex algorithm for solving an LP or a transportation problem.

B	Usually denotes a basis for an LP in standard form.
x_b, x_D	the vectors of basic (dependent), nonbasic (independent) variables WRT a basis for an LP.
$0-1$ variable	A variable that is constrained to take values of 0 or 1. Also called "binary variable" or "boolean variable".
$1, 2, \ldots, n; 1$	A tour for a TSP beginning and ending at city 1, indicating the order in which the various cities are visited.

Other symbols:

$O(n^r)$	When n is some measure of how large a problem is (either the size (number of digits in the data when it is encoded in binary form), or some quantity which determines the number of data elements), a finitely terminating algorithm for solving it is said to be of order n^r or $O(n^r)$, if the computational effort required by it is bounded above by αn^r, where α is a constant independent of the size and the data in the problem.
$\bowtie$	This symbol indicates the end of the present portion of text (i.e., example, exercise, comment etc.). Next line either resumes what was being discussed before this portion started, or begins the next portion.