BUSINESS OPTIMISATION USING MATHEMATICAL PROGRAMMING
Business Optimisation Using Mathematical Programming

JOSEF KALLRATH

and

JOHN M. WILSON
Dedication

To those who increased my¹ pleasure in mathematics:

Wilhelm Braun (1970-1975)
Klaus Reusch and Wilhelm Gieselmann (1976-1979)

To² Helen, Alex, Tim and Jack.

¹ Josef Kallrath
² John M. Wilson
# Contents

List of Figures x

Preface xi

1 Optimisation: Using Models, Validating Models, Solutions, Answers 1
   1.1 Introduction: Some Words on Optimisation . . . . . . . . . . 1
   1.2 The Scope of this Book . . . . . . . . . . . . . . . . . . . . 6
   1.3 The Significance of Models . . . . . . . . . . . . . . . . . . 7
   1.4 Mathematical Optimisation . . . . . . . . . . . . . . . . . . 9
       1.4.1 A Linear Programming Example . . . . . . . . . . . . 9
       1.4.2 A Typical Linear Programming Problem . . . . . . . 14
   1.5 Using Modelling Systems and Software . . . . . . . . . . . . . 16
       1.5.1 Implementing a Model . . . . . . . . . . . . . . . . . 18
       1.5.2 Obtaining a Solution . . . . . . . . . . . . . . . . . . 21
       1.5.3 Interpreting the Output . . . . . . . . . . . . . . . . . 23
   1.6 Benefiting from and Extending the Simple Model . . . . . . . 25
   1.7 A Survey of Real-World Problems . . . . . . . . . . . . . . . 28
   1.8 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31
   1.9 Appendix to Chapter 1 . . . . . . . . . . . . . . . . . . . . . 31
       1.9.1 Notation . . . . . . . . . . . . . . . . . . . . . . . . . 31
       1.9.2 A Brief History of Optimisation ⊖ . . . . . . . . . . . 32

2 The Scope of Problem Formulation and How to Formulate the Problem 37
   2.1 How to Model and Formulate a Problem . . . . . . . . . . . . 37
   2.2 Variables, Indices, Sets and Domains . . . . . . . . . . . . . 39
       2.2.1 Indices, Sets and Domains . . . . . . . . . . . . . . . 42
       2.2.2 Summation . . . . . . . . . . . . . . . . . . . . . . . . 43
   2.3 Constraints . . . . . . . . . . . . . . . . . . . . . . . . . . . 46
       2.3.1 Types of Constraints . . . . . . . . . . . . . . . . . . 47
       2.3.2 Example . . . . . . . . . . . . . . . . . . . . . . . . . . 50
   2.4 Objectives . . . . . . . . . . . . . . . . . . . . . . . . . . . . 50
   2.5 Building More Sophisticated Models . . . . . . . . . . . . . . 52
2.5.1 A Simple Production Planning Exercise ....... 52
  2.5.1.1 The Model Background ............... 52
  2.5.1.2 Developing the Model ............... 53
2.6 Mixed Integer Programming .................. 54
  2.6.1 Example: A Farmer Buying Calves and Pigs .... 56
2.7 Interfaces - Spreadsheets and Databases ....... 57
  2.7.1 Example: A Blending Problem .......... 59
  2.7.2 Developing the Model ............... 61
  2.7.3 Re-running the Model with New Data .... 63
2.8 Creating a Production System ................ 64
2.9 Collecting Data .......................... 66
2.10 Modelling Logic .......................... 67
2.11 Practical Solution of LP Models ............. 67
  2.11.1 Problem Size ..................... 68
  2.11.2 Ease of Solution ............... 68
2.12 Summary ................................ 70
2.13 Exercises ............................. 70

3 Mathematical Solution Techniques ............... 73
  3.1 Introduction ................................ 73
  3.1.1 Standard Formulation of Linear Programming Problems 73
  3.1.2 Slack and Surplus Variables ............. 75
  3.1.3 Underdetermined Linear Equations and Optimisation 76
3.2 Linear Programming .......................... 77
  3.2.1 Simplex Algorithm — A Brief Overview ..... 77
  3.2.2 Solving the Boat Problem with the Simplex Algorithm 78
  3.2.3 Interior-Point Methods — A Brief Overview .... 84
  3.2.4 LP as a Subroutine ................ 86
3.3 Mixed Integer Linear Programming ............. 87
  3.3.1 Solving the Farmer’s Problem with Branch & Bound 87
  3.3.2 Solving Mixed Integer Linear Programming Problems 90
  3.3.3 Cutting-Planes and Branch & Cut ........ 94
3.4 Interpreting the Results ..................... 96
  3.4.1 LP Solution ........................ 96
  3.4.2 Outputing Results .................. 97
  3.4.3 Dual Value (Shadow Price) ............. 97
  3.4.4 Reduced Costs .................... 98
  3.4.5 Report Writing ..................... 99
3.5 Duality ................................ 99
  3.5.1 Constructing the Dual Problem ........ 100
  3.5.2 Interpreting the Dual Problem ......... 102
  3.5.3 Duality Gap and Complementarity ....... 103
3.6 Summary ................................ 106
3.7 Exercises ............................. 106
3.8 Appendix to Chapter 3 ...................... 107
Contents

3.8.1 Linear Programming — A Detailed Description . . . . 107
3.8.2 Computing Initial Feasible LP Solutions . . . . . . . . 114
3.8.3 LP Problems with Upper Bounds . . . . . . . . . . . . 116
3.8.4 Dual Simplex Algorithm . . . . . . . . . . . . . . . . . . 120
3.8.5 Interior-Point Methods — A Detailed Description . . 120
    3.8.5.1 A Primal-Dual Interior-Point Method . . . . 123
    3.8.5.2 Predictor-Corrector Step . . . . . . . . . . . 126
    3.8.5.3 Computing Initial Points . . . . . . . . . . . 127
    3.8.5.4 Updating the Homotopy Parameter . . . . 127
    3.8.5.5 Termination Criterion . . . . . . . . . . . . . 128
    3.8.5.6 Basis Identification and Cross-Over . . . . 129
    3.8.5.7 Interior-Point versus Simplex Methods . . . 129
3.8.6 Branch & Bound with LP-Relaxation . . . . . . . . . . 130

4 Problems Solvable Using Linear Programming 135
4.1 Trimloss Problem . . . . . . . . . . . . . . . . . . . . . . . . 135
    4.1.1 Example: A Trimloss Problem in the Paper Industry . 136
    4.1.2 Example: An Integer Trimloss Problem . . . . . . . 138
    4.1.3 Other Applications . . . . . . . . . . . . . . . . . . 139
4.2 The Food Mix Problem . . . . . . . . . . . . . . . . . . . . . 139
    4.2.1 Case Study: Manufacturing Foods . . . . . . . . . . . 140
4.3 Transportation and Assignment Problems . . . . . . . . . . 140
    4.3.1 The Transportation Problem . . . . . . . . . . . . . 140
    4.3.2 The Transshipment Problem . . . . . . . . . . . . . 143
    4.3.3 The Assignment Problem . . . . . . . . . . . . . . . 144
    4.3.4 Transportation and Assignment Problems occurring
        as Subproblems . . . . . . . . . . . . . . . . . . . 147
    4.3.5 Matching Problems . . . . . . . . . . . . . . . . . . . 147
4.4 Network Flow Problems . . . . . . . . . . . . . . . . . . . . 148
    4.4.1 Illustrating a Network Flow Problem . . . . . . . . . 148
    4.4.2 The Structure and Importance of Network Flow Models . 150
    4.4.3 Case Study: A Telephone Betting Scheduling Problem . 150
    4.4.4 Other Applications of Network Modelling Technique . . 153
4.5 Unimodularity . . . . . . . . . . . . . . . . . . . . . . . . . 153
    4.5.1 A Unimodular Transportation Matrix ⊕ . . . . . . . . 153
4.6 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . 154
4.7 Exercises . . . . . . . . . . . . . . . . . . . . . . . . . . . . 154

5 How Optimisation is Used in Practice: Case Studies in Linear Programming 155
5.1 Optimising the Production of a Chemical Reactor . . . . . . 155
5.2 An Apparently Nonlinear Blending Problem . . . . . . . . . 157
    5.2.1 Formulating the Direct Problem . . . . . . . . . . . 158
    5.2.2 Formulating the Inverse Problem . . . . . . . . . . . 160
    5.2.3 Analysing and Reformulating the Model . . . . . . . 161
## Contents

5.3 Data Envelopment Analysis (DEA) ........................................ 163
  5.3.1 Example Illustrating DEA ....................................... 164
  5.3.2 Efficiency ......................................................... 166
  5.3.3 Inefficiency ....................................................... 166
  5.3.4 More than one Input ............................................. 167
  5.3.5 Small Weights .................................................... 168
  5.3.6 Applications of DEA ............................................ 168
  5.3.7 A General Model for DEA ....................................... 169
5.4 Vector Minimisation and Goal Programming ............................. 170
  5.4.1 A Case Study Involving Soft Constraints ....................... 173
5.5 Limitations of Linear Programming ..................................... 175
  5.5.1 Single Objective .................................................. 175
  5.5.2 Assumption of Linearity ......................................... 175
  5.5.3 Satisfaction of Constraints ...................................... 176
  5.5.4 Structured Situations ............................................ 177
  5.5.5 Consistent and Obtainable Data ................................ 177
5.6 Summary ........................................................................... 178
5.7 Exercises ......................................................................... 178

6 Modelling Structures Using Mixed Integer Programming 181
  6.1 Using Binary Variables to Model Logical Conditions ............. 181
    6.1.1 General Integer Variables and Logical Conditions ......... 182
    6.1.2 Transforming Logical Expressions into Arithmetical Expressions .............................................. 183
    6.1.3 Logical Expressions with Two Arguments ................... 184
    6.1.4 Logical Expressions with More than Two Arguments .... 186
  6.2 Logical Restrictions on Constraints .................................... 188
    6.2.1 Bound Implications on Single Variables .................... 189
    6.2.2 Bound Implications on Constraints ......................... 189
    6.2.3 Disjunctive Sets of Implications .............................. 192
  6.3 Modelling Non-Zero Variables ........................................... 194
  6.4 Modelling Sets of All-Different Elements ............................ 195
  6.5 Modelling Absolute Value Terms ⊖ .................................. 196
  6.6 Modelling Products of Binary Variables ............................ 198
  6.7 Special Ordered Sets ................................................... 198
    6.7.1 Special Ordered Sets of Type 1 .............................. 199
    6.7.2 Special Ordered Sets of Type 2 .............................. 201
    6.7.3 Linked Ordered Sets ............................................ 205
    6.7.4 Families of Special Ordered Sets ............................ 207
  6.8 Improving Formulations by Adding Logical Inequalities .......... 208
  6.9 Summary ........................................................................ 210
  6.10 Exercises ..................................................................... 210
8.4.2 A Yield Management Example .......................... 264
8.5 Post-Optimal Analysis ................................... 266
  8.5.1 Getting Around Infeasibility ......................... 266
  8.5.2 Basic Concept of Ranging ......................... 268
  8.5.3 Parametric Programming ......................... 270
  8.5.4 Sensitivity Analysis in MILP Problems ............... 272
8.6 Summary ................................................. 273

9 User Control of the Optimisation Process and Improving Efficiency 275
  9.1 Preprocessing ........................................... 275
    9.1.1 Presolve ............................................. 276
      9.1.1.1 Arithmetic Tests ............................... 276
      9.1.1.2 Tightening Bounds ......................... 278
    9.1.2 Disaggregation of Constraints .................... 279
    9.1.3 Coefficient Reduction ............................... 280
    9.1.4 Clique Generation ................................. 282
    9.1.5 Cover Constraints ................................. 283
  9.2 Efficient LP Solving .................................... 285
    9.2.1 Warm Starts ....................................... 285
    9.2.2 Scaling .............................................. 285
  9.3 Good Modelling Practice ................................. 286
  9.4 Choice of Branch in Integer Programming ............... 290
    9.4.1 Control of the Objective Function Cut-off .......... 290
    9.4.2 Branching Control .................................. 291
      9.4.2.1 Entity Choice .................................. 291
      9.4.2.2 Choice of Branch or Node .................... 292
    9.4.3 Priorities ............................................ 292
    9.4.4 Branching for Special Ordered Sets ................ 293
    9.4.5 Branching on Semi-Continuous and Partial Integer Variables ........................................... 294
  9.5 Summary ................................................. 296
  9.6 Exercises ............................................... 296

10 How Optimisation is Used in Practice: Case Studies in Integer Programming 299
  10.1 What Can be Learned from Real-World Problems .......... 299
  10.2 Three Instructive Solved Real-World Problems .......... 300
    10.2.1 Contract Allocation ............................... 300
    10.2.2 Metal Ingot Production ............................ 302
    10.2.3 Project Planning .................................... 303
    10.2.4 Conclusions ........................................ 305
  10.3 A Case Study in Production Scheduling .................... 306
  10.4 Optimal Worldwide Production Plans ⊕ .................. 311
    10.4.1 Brief Description of the Problem ............... 311


10.4.2 Mathematical Formulation of the Model ........................ 312
  10.4.2.1 General Framework .................................... 312
  10.4.2.2 Time Discretisation .................................. 314
  10.4.2.3 Including Several Market Demand Scenarios ........... 314
  10.4.2.4 The Variables ........................................ 315
  10.4.2.5 The State of the Production Network .................. 316
  10.4.2.6 Exploiting Fixed Setup Plans ......................... 316
  10.4.2.7 Keeping Track of Mode Changes ....................... 316
  10.4.2.8 Coupling Modes and Production ....................... 318
  10.4.2.9 Minimum Production Requirements ...................... 319
  10.4.2.10 Modelling Stock Balances and Inventories .......... 320
  10.4.2.11 Modelling Transport ................................. 321
  10.4.2.12 External Purchase .................................... 321
  10.4.2.13 Modelling Sales and Demands ......................... 322
  10.4.2.14 Defining the Objective Function ...................... 322

10.4.3 Remarks on the Model Formulation ............................ 323
  10.4.3.1 Including Minimum Utilisation Rates .................. 323
  10.4.3.2 Exploiting Sparsity ................................... 324
  10.4.3.3 Avoiding Zero Right-Hand Side Equations ............. 326
  10.4.3.4 The Structure of the Objective Function .............. 326

10.4.4 Model Performance ........................................... 327

10.4.5 Reformulations of the Model .................................. 328
  10.4.5.1 Estimating the Quality of the Solution ................ 328
  10.4.5.2 Including Mode-Dependent Capacities .................. 329
  10.4.5.3 Modes, Change-Overs and Production ................... 330
  10.4.5.4 Reformulated Capacity Constraints .................... 332
  10.4.5.5 Some Remarks on the Reformulation .................... 333

10.4.6 What can be Learned from this Case Study? .................. 333

10.5 A Complex Scheduling Problem ◀ ............................... 334
  10.5.1 Description of the Problem ............................... 334
  10.5.2 Structuring the Problem .................................. 334
  10.5.2.1 Orders, Procedures, Tasks and Jobs ................. 335
  10.5.2.2 Labour, Shifts, Workers and their Relations ........ 336
  10.5.2.3 Machines ............................................ 337
  10.5.2.4 Services ............................................ 337
  10.5.2.5 Objectives .......................................... 337

10.5.3 Mathematical Formulation of the Problem .................... 338
  10.5.3.1 General Framework .................................... 338
  10.5.3.2 Time Discretisation .................................. 338
  10.5.3.3 Indices .............................................. 339
  10.5.3.4 Data .................................................. 339
  10.5.3.5 Main Decision Variables ............................... 339
  10.5.3.6 Other Variables ...................................... 340
  10.5.3.7 Auxiliary Sets ....................................... 340

10.5.4 Time-Indexed Formulations .................................. 340
10.5.4.1 The δ Formulation 341
10.5.4.2 The α Formulation 342
10.5.5 Numerical Experiments 343
  10.5.5.1 Description of Small Scenarios 343
  10.5.5.2 A Client’s Prototype 345
10.5.6 What can be Learned from this Case Study? 351

10.6 Telecommunication Service Network ϖ 352
10.6.1 Description of the Model 353
  10.6.1.1 Technical Aspects of Private Lines 353
  10.6.1.2 Tariff Structure of Private Line Services 353
  10.6.1.3 Demands on Private Line Services 355
  10.6.1.4 Private Line Network Optimisation 355
10.6.2 Mathematical Model Formulation 355
  10.6.2.1 General Foundations 356
  10.6.2.2 Flow Conservation Constraints 359
  10.6.2.3 Edge Capacity Constraints 360
  10.6.2.4 Additional Constraints 361
  10.6.2.5 Objective Function of the Model 363
  10.6.2.6 Estimation of Problem Size 363
  10.6.2.7 Computational Needs 364
10.6.3 Analysis and Reformulations of the Models 365
  10.6.3.1 Basic Structure of the Model 365
  10.6.3.2 Some Valid Inequalities: Edge Capacity Cuts 365
  10.6.3.3 Some Improvements to the Model Formulation 367
  10.6.3.4 A Surrogate Problem with a Simplified Cost Function 368
  10.6.3.5 More Valid Inequalities: Node Flow Cuts 370
  10.6.3.6 Some Remarks on Performance 370

10.7 Summary 370

10.8 Exercises 371

11 Other Types of Optimisation Problems ϖ 373
11.1 Recursion or Successive Linear Programming 373
  11.1.1 An Example 374
  11.1.2 The Pooling Problem 376
11.2 Stochastic Programming 380
11.3 Quadratic Programming 381
11.4 Mixed Integer Nonlinear Programming 385
  11.4.1 Definition of an MINLP Problem 385
  11.4.2 Some General Comments on MINLP 385
  11.4.3 Deterministic Methods for Solving MINLP Problems 387
  11.4.4 The Program DICOPT 389
11.5 Summary 389
12 Conclusion: The Impact and Implication of Optimisation 391
12.1 What the Users Can Get Out of the Case Studies . . . . . . . 391
12.2 Benefits of Mathematical Programming to Users . . . . . . . 392
12.3 Implementing and Validating Solutions . . . . . . . . . . . . 393
12.4 Communicating with Management . . . . . . . . . . . . . . . 394
12.5 Keeping a Model Alive . . . . . . . . . . . . . . . . . . . . . 394
12.6 Mathematical Optimisation in Small and Medium Size Business395
12.7 On-Line Optimisation by Exploiting Parallelism? . . . . . . . 396
  12.7.1 Algorithmic Components Suitable for Parallelisation . 396
  12.7.2 Non-determinism in Parallel Optimisation . . . . . . . 397
  12.7.3 Platforms for Parallel Optimisation Software . . . . . . 398
  12.7.4 Design Decisions . . . . . . . . . . . . . . . . . . . . 399
  12.7.5 Implementation . . . . . . . . . . . . . . . . . . . . . 400
  12.7.6 Performance . . . . . . . . . . . . . . . . . . . . . . 401
  12.7.7 Acceptability . . . . . . . . . . . . . . . . . . . . . . 401
12.8 Future Developments and Conclusions . . . . . . . . . . . . . 402
12.9 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 406

A Software Related Issues 407
A.1 Accessing Data from Harddisk . . . . . . . . . . . . . . . . . 407
  A.1.1 The DISKDATA statement . . . . . . . . . . . . . . . . 407
  A.1.2 Connecting to Spreadsheets . . . . . . . . . . . . . . . 407
A.2 List of Case Studies and Model Files . . . . . . . . . . . . . 408

B Glossary 409

Index 427
### List of Figures

1.1 Transforming a real-world problem .................. 4  
1.2 Graphical solution of an LP problem in two variables .... 13  
1.3 Initial screen ........................................ 19  
1.4 New problem ........................................... 20  
1.5 Boats model ............................................. 21  
1.6 Changing minimise to maximise ........................ 22  
1.7 Saving the file ......................................... 23  
1.8 Results from model .................................... 24  
1.9 Erroneous model ....................................... 25  
1.10 Extended model ....................................... 27  
1.11 Results from extended model .......................... 28  
2.1 Integer model .......................................... 58  
2.2 Progress to integer solution ............................ 59  
2.3 Simple spreadsheet ..................................... 60  
3.1 Illustration of interior-point methods .................. 85  
3.2 LP relaxation and the first two subproblems of a B&B tree . 88  
3.3 LP-relaxation, convex hull, and a B&B tree ............. 93  
3.4 Illustrating the idea of Branch & Cut ................... 95  
3.5 Feasible region of an LP problem ...................... 108  
3.6 The revised simplex algorithm ........................ 110  
3.7 Logarithmic penalty term .............................. 122  
3.8 The Branch & Bound algorithm ......................... 131  
3.9 Two Branch & Bound trees ............................. 133  
4.1 Trimloss problem ...................................... 136  
4.2 Routes on transportation and transshipment networks .... 145  
4.3 Flows between nodes of a network ..................... 149  
4.4 Partial network [from Wilson and Willis (1983)] ........ 152  
6.1 Using S1 sets to select capacity size .................. 200  
6.2 Using S2 sets to model a nonlinear curve ............... 202  
7.1 Travelling salesman problem with four cities .......... 222
<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.2</td>
<td>A set covering problem</td>
<td>226</td>
</tr>
<tr>
<td>7.3</td>
<td>Gantt chart showing the schedule</td>
<td>240</td>
</tr>
<tr>
<td>8.1</td>
<td>Cost as a function of number of items</td>
<td>261</td>
</tr>
<tr>
<td>8.2</td>
<td>Sensitivity analysis: objective versus optimal value</td>
<td>269</td>
</tr>
<tr>
<td>8.3</td>
<td>Sensitivity analysis: optimal value versus unit profit</td>
<td>270</td>
</tr>
<tr>
<td>8.4</td>
<td>Sensitivity analysis: slope of objective function</td>
<td>271</td>
</tr>
<tr>
<td>10.1</td>
<td>Production network with three sites</td>
<td>313</td>
</tr>
<tr>
<td>10.2</td>
<td>Illustration of a set-up change</td>
<td>318</td>
</tr>
<tr>
<td>10.3</td>
<td>Production plan</td>
<td>325</td>
</tr>
<tr>
<td>10.4</td>
<td>Precedence relations between jobs</td>
<td>347</td>
</tr>
<tr>
<td>10.5</td>
<td>Gantt chart and personnel occupation diagram</td>
<td>349</td>
</tr>
<tr>
<td>10.6</td>
<td>Cost of bandwidth for POP-to-POP private lines</td>
<td>354</td>
</tr>
<tr>
<td>10.7</td>
<td>A possible routing via hub sites for a demand $D_{ij}$</td>
<td>358</td>
</tr>
<tr>
<td>11.1</td>
<td>The pooling problem and a process unit fed by a pool</td>
<td>379</td>
</tr>
<tr>
<td>11.2</td>
<td>Convex and non-convex sets and functions</td>
<td>386</td>
</tr>
<tr>
<td>12.1</td>
<td>Speed-up achieved with eight slaves</td>
<td>397</td>
</tr>
</tbody>
</table>
Preface

This book arose from a realisation that modelling using mathematical programming should be tightly linked with algorithms and their software implementation to solve models. Such linkage is necessary for a full appreciation of the methods used to model problems that will ensure they can be solved successfully. While there exist textbooks concentrating on the pure mathematics aspects of optimisation, and others which just describe applications without providing sufficient technical background, we see our book as trying to provide a link between applications and the mathematics required to solve real-world problems. Few textbooks have integrated modelling with state-of-the-art commercially available software. Our book will also incorporate this missing link and will include the software to solve the models discussed.

Optimisation using mathematical programming is an important subject area as it can determine the dramatic savings available to organisations that could not be achieved by other means. In the book, examples are cited where organisations are saving many millions of pounds (sterling) or dollars (US) by using optimisation methods. Mathematical optimisation models are tools that can help people in the process of making decisions concerning the use of resources and saving costs.

Mathematical programming also provides a way to solve problems that, because of their size or other features, would not otherwise be solvable by other methods. In major cities, for example London, mathematical programming models influence the control of the flow of domestic water through the city as the model is used to determine the most efficient strategy to move water from source to user as peaks and troughs in the usage pattern develop. Thus, the results from mathematical programming models are literally all around many of us.

The need for a source book of material on the subject was recognised while teaching at Heidelberg University and Loughborough University and while planning conference sessions on the practical relevance of mixed integer optimisation.

Although there is an extensive literature on mathematical programming, the paucity of instructional materials in the area of efficient modelling and solving real-world problems is striking. The student, researcher, or indus-
trial practitioner must read between the lines of material, usually only available in journal articles or similar, to glean the details of the modelling process and the “tricks of the trade”. Yet the need is acute: as with many other areas of science, the computer revolution has given many modellers in industry as well as at universities the tools to attempt to solve realistic and complex models. In this work, we endeavour to provide a suitable background as an aid to the novice modeller, a useful reference book for the experienced modeller, and a springboard for the development of new modelling ideas. In particular, by tailoring the book around a commercially available software package we are able to illustrate some of the subtle details that determine the success or failure of the modelling efforts.

Readership
This book has been planned for use by more than one type of readership. Most of the book is designed to be used by readers who possess fairly elementary mathematical skills, i.e., the use of algebraic manipulation, and it is made clear which sections are not of this type. Further mathematical skills required are developed during the course of the book but the presentation should not prove too daunting. The material is suitable for use in courses in Business and Management Studies and operations research environments. Readers with stronger mathematical skills (e.g., linear and matrix algebra) and experienced practitioners in the field will still find much to interest them as the logic of modelling is developed. The book, therefore, will provide appropriate course material for lecture courses, short courses and self-teaching on the topics contained in it.

As some material is for the more advanced reader, or for the reader to use on a second pass through the book, certain sections in chapters have been marked as “advanced”. These sections may be omitted on a first pass through the book. The more advanced parts of the book are written in such a way that it is sufficient if the reader is familiar with the basic concepts and techniques of linear algebra. A discussion of some foundations of optimisation is provided at the end of some chapters, where it is helpful if the reader has familiarity with calculus techniques. It is also expected that the later advanced chapters will be read only once the reader has started to build models in earnest. A glossary at the end of the book will provide further help.

Scope
The focus of the book is primarily on models, model applications and individual case studies rather than algorithmic details. However, because the success of solution of complex problems requires efficient problem solving, it is important that models and algorithms are tightly connected. Therefore, we also concentrate on the mathematical formulation of models and the mathematical background of the algorithms. The understanding of the
mathematics involved in a problem or model explains why certain model formulations work well while others do not. We have tried to present in this book a self-contained treatment of the subject where possible. The presentation of the material is not too far away from what real modelling in business looks like. Most of the case studies have a commercial or industrial background. For instance, some of the case studies in Chapter 10 stem from problems recently analysed and solved in a mathematical consultancy group in the chemical industry.

**Organisation**

Chapter 1 gives an introduction and overview of the field. Parts of this chapter, in particular the details on the software used in this book, can be skipped by the experienced practitioner. An overview on the history of optimisation is presented in the appendix to Chapter 1. It is presented as an appendix because it requires some familiarity with the terminology of the subject. This chapter and parts of Chapter 2, illustrating how small linear and integer programming problems may be formulated, are kept on a very elementary level appropriate to the novice without a background in mathematics. In Chapter 3 we provide a systematic overview of mathematical solution techniques on both linear and mixed integer programming. Exercises are included at the ends of chapters. These exercises can be tackled by hand or by using the software, where appropriate, included with the book.

Types of linear programming problems and their modelling are discussed in Chapter 4. Chapter 5 is a collection of case studies in the framework of linear programming. Chapters 6 and 7 cover foundations of integer programming while in Chapter 8 case studies are discussed. In Chapter 9 we consider how practitioners may best set up and solve their optimisation problems and in Chapter 10 we consider examples of large cases. Then in Chapter 11 we consider other types of optimisation, e.g., sequential linear, quadratic and mixed integer nonlinear programming. Chapter 12 reflect the authors’ views on mathematical optimisation and modelling, how it is and should be used, and what is to be expected from it in the future.

Certain sections of chapters may be skipped by readers new to the area of optimisation. They are marked by ⊖ in the section heading. These sections should be read through when required on a subsequent reading.

**Instructors Manual**

An Instructors Manual is available to bona fide lecturers.

**Acknowledgements**

We would like to thank colleagues and mentors who have advised and/or inspired us over the years. These are too numerous to mention but we would like to single out Beate Brockmüller for providing material on the telecommunication network problem, Bob Daniel and Gunter Schnabel for the time they spent with us discussing the manuscript in great detail, Tom Horak,
Gernot Sauerborn, Anna Schreieck, James Tebboth, Christian Timpe and Max Wagner for reading the manuscript (JK), and inspiration from the work of Peter Hammer, Ailsa Land, Gautam Mitra, Paul Williams and the late Martin Beale (JMW). We also offer our thanks to Dash Associates for help, advice and cooperation over the inclusion of the XPRESS-MP software and related discussion material. JK wants to express his special thanks to Marilyn Dalton for her kind hospitality during numerous visits to Blisworth House. Finally, JK wants to thank the clients involved in some of the real-world cases. The interaction and communication with the clients, most of whom were enthusiastic persons with deep knowledge of the business process they wanted to improve using mathematical optimisation, was an important and irreplaceable resource which made the solution of challenging problems possible. Although, after all the years these people might have forgotten the work and the exciting time we spent together and might not be aware how they indirectly contributed to this book I want to mention them: Peter Bassler, David DeSantis, Andy Hayter, Klaus Kindler, Jan Orband, Gunter Schnabel, Hubert Smuda and Eckhardt Schubert.

Ludwigshafen and Loughborough 1997  
Josef Kallrath and John M. Wilson

3Dash holds a copyright on parts of the following sections: 2.7, 10.1, 10.3 and 12.6.
4Of course, I hope they have not forgotten! All the special adventures involved in the time working on their problems fill a book on its own.