HUB RESEARCH PAPER
Economics & Management

Using mixed integer programming to win a cycling game

Jeroen Beliën, Dries Goossens, Daam Van Reeth and Liesje De Boeck

HUB RESEARCH PAPER 2009/10
MEI 2009

ASSOCIATIE K.U.LEUVEN
EHSAL
HUB
partners in
Hogeschool-Universiteit Brussel

Using mixed integer programming to win a cycling game

Jeroen Beliën†;‡, Dries Goossens§, Daam Van Reeth†, Liesje De Boeck†;‡,

†Hogeschool-Universiteit Brussel, Campus Economische Hogeschool, Stormstraat 2, B-1000 Brussel, Belgium, jeroen.belien@hubrussel.be, daam.vanreeth@hubrussel.be, liesje.deboeck@hubrussel.be

‡Affiliated Researcher Katholieke Universiteit Leuven, Faculty of Business and Economics, Department of Decision Sciences and Information Management, Research Center for Operations Management, Naamsestraat 69, B-3000 Leuven, Belgium,

§PostDoc researcher for Research Foundation - Flanders, Katholieke Universiteit Leuven, Faculty of Business and Economics, Department of Decision Sciences and Information Management, ORSTAT, Naamsestraat 69, B-3000 Leuven, Belgium, dries.goossens@econ.kuleuven.be

Abstract

This paper presents an application of optimization modeling to win a popular cycling game. The application involves real-life data of today's cyclists and challenges the students because of the competition aspect. Since the developed optimization model contains features of knapsack problems, multiperiod problems and integer programming modeling, it is perfectly suited as a concluding case study in an undergraduate operations research/management science course. Moreover, the application also sharpens the understanding of the working of stock exchange markets and is, therefore, also interesting for finance courses. The application was originally developed for an MBA operations research course focusing on spreadsheet modeling skills, but it can also be used in courses that focus on algebraic modeling of optimization problems.
Keywords: Cycling, game, spreadsheet modeling, mixed integer programming, knapsack problems, multiperiod problems, stock market

1 Introduction and literature review

The best way to make business students enthusiastic about operations research/management science (OR/MS) is to let them experience how useful the gained skills can be in their personal life. Ask your students to model a fictive business problem and only a minority will be motivated to find the correct solution. Ask your students to model a real-life business problem and maybe, if you are lucky, half of them will do a serious effort. Do

the same, but use an application that fascinates students, like sports (e.g., basket, football, baseball in the US; cycling or soccer in Europe) and the majority is eager to find the solution. Finally, combine a popular application with a competition aspect (play a game) and almost all of the students are determined to find the optimal solution (and win the competition).
This is in line with the literature. Cochran (2004) explains why and how sports can help instructors of applied quantitative methods achieve several important pedagogical goals. The special issue on 'SpORts in the OR Classroom', INFORMS Transactions on Education (2004, volume 5, nr 1), presents several articles documenting the successful classroom use of sports contexts, including examples from each of the major North American team sports (baseball, hockey, basketball, soccer, and American football). In that issue, Trick (2004) describes how to teach integer programming using sports scheduling applications. As far as we know, our paper is the first OR/MS education paper in the context of cycling.
As has been argued by Sniedovich (2002), games are a valuable source for developing educationally rich material for OR/MS courses. More specifically, games are very effective for teaching LP and IP modeling to students with motivation problems and/or less quantitative background. For instance, Chlond and Toase (2002) propose several IP models for chessboard placement and closely related puzzles for use in a classroom. Another example of a logical puzzle can be found in Chlond (2008) or in Chlond (2005) in which the popular Su Doku and Log Pile Game are being solved by integer programming.

The game presented in this paper is suited for a mixed integer programming approach involving three important classes of optimization problems and corresponding modeling techniques: (1) knapsack problems, (2) multiperiod problems and (3) integer programming modeling.
(1) The knapsack problem is in many textbooks used to introduce integer programming problems (e.g.,Winston (2004)) and dates back to the early days of linear programming (e.g., Dantzig (1957)). Since then, many papers and even whole books have been written on applications and algorithms for knapsack problems (e.g., Martello and Toth (1990) and Kellerer et al (2005)).
(2) Multiperiod or dynamic models have also been widely studied in literature. Mixed integer programming formulations have been proposed for, amongst other, dynamic inventory (lot sizing) problems (e.g., Stadtler (1996)), multiperiod financial models (e.g., Rohn (1987)) and multiperiod work scheduling problems (e.g., Franz and Miller (1993)).

(3) The specific use of integer programming to model particular constraints has been illustrated by Chlond and Toase (2003) who focus on the modeling of logical conditions in puzzles by IP.

The remainder of this paper is organized as follows. Section 2 introduces the Gigabike game and explains which aspects make it interesting for use in a classroom. Section 3 presents the optimization models for solving the Gigabike optimization problem. Finally, Section 4 discusses some classroom experiences and Section 5 concludes this paper.

2 The Gigabike game

2.1 The origin of Gigabike

Betting on the outcome of future events has always been a favorite pastime. These events historically ranged from the totally unpredictable random roll of dice to the brutal and sometimes more predictable confrontation of an army battle. Betting on sports results today could be considered a modern equivalent of this Ancient History betting on the outcome of war. For instance, in Western culture betting on horse races and boxing matches has been very popular in the second part of the 20th century.

The outcome of a sports competition is not totally random, but depends on the qualities of the sportsmen and teams involved. As a result, in contrast to taking part in a dice game or a lottery, sports betting allows participants to exploit information advantage, e.g. on prior results, to predict the final outcome. However, uncertainty remains important enough to still create sufficient difference in opinion among spectators and allowing sports bets to take place. It is this unique combination of predictability and uncertainty that makes sports betting particularly appealing.

Halfway through the nineties, the popular Belgian newspapers 'Het Laatste Nieuws' and 'Het Nieuwsblad' started to use sports betting games as a marketing instrument to boost their sales. Interest in popular sports, especially soccer and road cycling, was combined with a game element of predicting which sportsmen would do well. Participants in these games typically had to act as some sort of managers of a team of riders selected by themselves. The goal was to select a fixed number of real soccer players (Megascore) or professional cyclists (Megabike) scoring the most points in selected sports events. By paying a small fee, entrants in the game could win interesting (money) prizes if they selected the best team.

For road cycling, newspapers organized games around the popular spring classic races such as the Tour of Flanders and Paris-Roubaix, and the widely viewed Tour de France. Unfortunately for die-hard cycling fans, less popular races, such as the Giro d'Italia or the autumn classic races, were not included in these newspaper games. Peter Samoy and Mark Vanderwegen, two workmates with a common interest in cycling, saw an opportunity and developed a more elaborated game concept. In the year 2000 they started "Gigabike", a non-commercial year-long Internet cycling game including all major road cycling races (www.gigabike.be). Although no prizes could be won, the game became an instant success amongst Belgian and Dutch cycling fans because of its originality. The number of players in the game rose steadily from initially 40 to almost 500 in 2009.

2.2 The principles of the Gigabike game

The Gigabike game basically goes as follows. At the start of the road cycling season, each Gigabike player has to select a team of 30 riders from a world ranking of all professional road cyclists. In this ranking, each cyclist is assigned a value, the so-called Cycling Quotient (CQ) value, based on his performances during the last 12 months (see www.cqranking.com). This value thus reflects the past quality of the rider and might be an indication for his future performance. The total CQ value of a team selection is required to be smaller than a particular limit. This limit is further referred to as the CQ budget or, simply, budget. Observe that this 'budget restriction' makes it impossible to select only the best riders, i.e. the professional cyclists with the highest CQ value. During the season, at five moments, it is possible to adapt the team formation by replacing at most five cyclists. However, the sum of the CQ values of the incoming riders cannot exceed the sum of the values of the outgoing riders plus the remaining (unused) CQ budget of the preceding period (if any). The overall winner of the game is the person whose selection of riders gained the most points over the course of the whole season, consisting of over 60 races and more than 120 racing days.

Two dynamic characteristics make Gigabike very distinct from other internet or newspaper cycling games. First, the transfer opportunities during the season allow for strategic decision making during the season. Second, the CQ value of the riders changes throughout the year as their results from 12 months ago are deleted and replaced by new results. After each race, the riders' CQ values are updated as follows: each rider loses the CQ points won in the last year edition of that race and gains the CQ points earned in the current edition. To give an example: suppose a particular rider won the Tour of Flanders of last year. Then, this rider will lose the CQ points after the Tour of Flanders, however, he might win them back if he wins the Tour of Flanders again this year. Hence, in the best case, this rider keeps his current value after the Tour of Flanders. Other riders, who did not score a single CQ point in the last year edition of the Tour of Flanders, cannot decrease their CQ-value: if they score any CQ points in this year's edition of the Tour of Flanders, their CQ value will increase. Especially this dynamic aspect makes the game appealing. Not only luck, but also strategic considerations and good knowledge of the racing schedule of certain cyclists determines the final result of a team. For instance, although a certain cyclist may be a pre-season favorite for winning the Tour de France, from a strategic point of view it could be better not to have him in the initial team as his value may decrease before the start of the Tour de France. Therefore, it is not always in the best interest of having the best riders in a team right from the start. However, since the number of transfer opportunities is limited and some riders will be in the team for the whole season, a well-balanced start selection of cyclists for all types of races is crucial to have any chance of winning the game.

2.3 Using Gigabike in a classroom

The Gigabike game can be used in a classroom to illustrate certain economic problems. In essence, the game very much reflects the situation of a stock market in which an investor can buy a number of shares for a fixed amount of money. These shares render dividends, and when share prices go up they can be sold with a profit to invest in other shares. In the game, the shares are the cyclists, the share prices are the CQ values and the dividends are the points scored by a cyclist in the races he participates in. The transfer opportunities in Gigabike are similar to the rearrangement of a stock portfolio in the real world.

Some particular characteristics of investing in shares on a stock market are also present in the strategic decision making during the game. One of the key strategies in Gigabike is to select those promising cyclists with currently low CQ values, for instance due to injury or bad luck, and sell them when they have a much higher CQ value. One of the best ways to make money on the stock market is also to pick out the underrated shares and sell them in time. However, sometimes the most promising shares raise ethical questions. Should an investor buy shares from arms or tobacco companies? An almost similar problem arises in Gigabike. To win the game, it is sometimes necessary to select questionable riders or have faith in riders that return from a doping suspension. Finally, the fact that the highest valued riders do not necessarily take the most points is analogous to the warning financial companies give to their investing clients that 'past performances of stocks or financial products are no guarantee for future profits'.

The focus of this paper, however, is on another classroom use that deals with an a posteriori game component of Gigabike. At the end of each season, some Gigabike players engage in a new competition. Given the fact that all results are now known, they try to find the optimal team, i.e. the team that would have collected the most points if all results had been known (or perfectly forecasted) in advance. The optimal team therefore reflects the hypothetic situation in which all the right decisions would have been taken. As we will show, the problem of finding this optimal team can be modeled as a mixed integer linear optimization problem involving characteristics of several basic problems, i.e. knapsack problems, multiperiod problems and integer programming modeling.

A spreadsheet (see the file Gigabike_OPT_2008.xls) greatly simplifies the search for the optimal cycling team, as it contains all the costs (CQ values) and all the points collected by the riders for each of the six periods. Moreover, given a particular selection, it calculates automatically the budget spent and the number of transfers done in each period. The user only has to indicate for each of the six periods which 30 riders are selected in the team. A constraint violation (negative budget, more than five transfers between two periods, or not exactly 30 riders) is clearly indicated in red. To further simplify things, the spreadsheet only contains the 112 best scoring riders of the past cycling season. Finally, clear instructions on how to use the spreadsheet for finding the optimal team are added in an extra sheet 'instructions'. We recommend distributing this spreadsheet file to the students, at least one week before the actual class on the Gigabike modeling problem. Ask them to find the best team possible starting from this file. If you promise a reward for the highest found score (e.g., a coke works well), many of your students will be extra motivated to find a good solution and become familiar with the problem. Some of them might even try to solve the puzzle by modeling it as a mixed integer programming (MIP) problem, however, the number of variables needed is too large for Frontline's standard Solver© for MS Excel©, further referred to as Solver. The actual class starts with congratulating the winner and presenting the optimal solution, which will, typically, be far better than the best score found by the students. In 2007 and 2008, the optimal score was also better than the best score found by the Gigabike participants who puzzled for almost 1 month. Being fascinated by the proven power of optimization modeling, students will be curious about the model.

3 Discussion of the models

This section presents the mixed integer programming model for the Gigabike optimization game. Since student versions of most optimization packages are limited to a small number of decision variables (e.g., up to 200 changing cells in Solver), we propose to develop the model for a smaller instance of the problem; i.e., we only consider the 18 best performing riders, a team consisting of 10 riders (instead of 30), a budget limited to 12000 CQ points (instead of 20000) and a maximum of three transfers between two periods (instead of five). Using these dimensions, the model can be solved by a standard version of Solver. Although the model is not difficult at all, we strongly recommend to apply a divide-and-conqueror approach, certainly for students with a limited quantitative background. After all, the aim of this case is to increase the students' modeling skills, not to overwhelm them with a brilliant optimization model. Fortunately, the modeling process can be very effectively divided into three stages with an increasing difficulty level. Moreover, these three stages each correspond to an important class of optimization problems. In this way, the link between theory and practice becomes very clear to the students. The three stages are:

1. Knapsack problem modeling
   See 'Gigabike 1 template.xls' for a template Excel Solver model
   See 'Gigabike 1 solution.xls' for the full model

2. Multiperiod modeling
   See 'Gigabike 2 template.xls' for a template Excel Solver model
   See 'Gigabike 2 solution.xls' for the full model

3. Integer programming modeling
   See 'Gigabike 3 template.xls' for a template Excel Solver model
   See 'Gigabike 3 solution.xls' for the full model

We now briefly discuss these three stages and present the corresponding optimization models.

3.1 Knapsack problem modeling

In a first stage, let us assume that the six periods are independent. In this simplified case, for each period, an identical, independent problem needs to be solved. Given the costs (CQ values) and the gains (CQ points obtained) of each rider in that period, the problem is to find the 10 riders within a total CQ value of 12000 (=budget constraint) who collect, in total, the most

CQ points. Since the periods are considered to be independent, we do not have to bother about the remaining budget (unused budget is simply lost) and about the transfer restrictions (we can even select 10 new riders in each period). In this way, the problem in each period reduces to a knapsack problem with one extra constraint, namely that the knapsack must consist of exactly 10 items. Let x_{ij} be 1 if rider i is selected in period j, 0 otherwise, p_{ij} be the points obtained by rider i in period j and c_{ij} be the cost (CQ value) of rider i in period j. Then, the formulation of the first stage problem is as follows:
maximize

$$\sum_{i=1}^{18}\sum_{j=1}^{6} p_{ij}x_{ij} \tag{1}$$

subject to

$$\sum_{i=1}^{18} c_{ij}x_{ij} \leq 12000, \qquad \forall j = 1, ..., 6 \tag{2}$$

$$\sum_{i=1}^{18} x_{ij} = 10, \qquad \forall j = 1, ..., 6 \tag{3}$$

$$x_{ij} \in \{0,1\}, \qquad \forall i = 1, ..., 18 \qquad \forall j = 1, ..., 6 \tag{4}$$

Constraints (2) ensure that the total value of the team in each period does not exceed the budget. Constraints (3) limit the number of riders chosen to be exactly 10. Applying this model results in the optimal set of 10 riders for each of the six periods. However, the model is a simplification of the real game. First, it does not take into account that unused budget in the first period, can be used in the second period, unused budget in the second period can be used in the third period, etc. Second, the formulation of the budget constraint is not entirely correct. Indeed, in the real game, a team can have a larger value than 12000 CQ points after the first transfer moment. This will be the case when the selected riders increase their total CQ value, i.e., if they perform better than the previous year during the finished period(s). Obviously, the reverse (a budget decrease) could also occur. Hence, implying a budget constraint of 12000 CQ points in each period is not entirely correct: the budget in the succeeding periods depends on the team performance in the preceding periods. Finally, the solution to (1)-(4) does not guarantee that there are at most three transfers between two periods. In other words, it is possible that, from the 10 riders selected in period 2, there are six or less common with the 10 riders selected in period

1. Therefore, the model has to be extended taking these three issues into account. We start with the first two issues, which both can be solved by transforming the model from a static model to a multiperiod (or dynamic) model.

3.2 Multiperiod modeling

In order to allow for the future use of remaining budget and to have a correct calculation of the available budget in the periods after period 1, balance constraints must be added to the model. Balance constraints are typically for multiperiod problems, e.g., dynamic inventory, work scheduling or investment problems. A balance constraint defines the link between two succeeding periods, i.e., it states that the output of a given period must be equal to the input of the succeeding period. Let r_j denote the remaining budget of period j. Then, the Gigabike a posteriori optimization model extended with the remaining budget issue, is as follows:
maximize

$$\sum_{i=1}^{18}\sum_{j=1}^{6} p_{ij}x_{ij} \tag{5}$$

subject to

$$\sum_{i=1}^{18} c_{i1}x_{i1} + r_1 = 12000, \tag{6}$$

$$r_{j-1} + \sum_{i=1}^{18} c_{ij}x_{i,j-1} - \sum_{i=1}^{18} c_{ij}x_{ij} = r_j, \qquad \forall j = 2, ..., 6 \tag{7}$$

$$\sum_{i=1}^{18} x_{ij} = 10, \qquad \forall j = 1, ..., 6 \tag{8}$$

$$x_{ij} \in \{0,1\}, \qquad \forall i = 1, ..., 18 \qquad \forall j = 1, ..., 6 \tag{9}$$

$$r_j \geq 0, \qquad \forall j = 1, ..., 6 \tag{10}$$

Equation (6) now calculates the remaining budget in period 1. Note that the budget cannot be exceeded, as the r_j variables are required to be non-negative. Since a budget violation corresponds to a negative remaining budget ($r_j < 0$), this cannot occur in a feasible solution to (5)-(10). Equations (7) define the balance restrictions, saying that the remaining budget from the previous period plus the incomes of the riders sold minus the costs of the new riders selected must be equal to the remaining budget that can be carried over to the next period. Remark that riders are always bought (sold) at their value of the moment of purchase (sell). Hence, it is possible to sell a rider at a price higher than the price paid when the rider was bought. This will be the case if the rider performed better than he did in the preceding year during the considered periods (between time of buy and time of sell). Obviously, the reverse is also possible. Observe that for riders which are not transferred, i.e., these riders that are in the team in period j, but also in period $j+1$, the net contribution to the budget equals zero. Indeed, these riders have a positive as well as a negative contribution in the left hand side of equation (7), and this at the same cost, i.e., at their current value, c_{ij}.

3.3 Integer programming modeling

The optimal solution to (5)-(10) does not necessarily satisfy the restriction on the maximal number of transfers between two periods. If less than seven riders are common between two succeeding periods, then more than three transfers took place, which is not allowed. In order to model the transfer restriction we need a new binary decision variable, t_{ij}, which equals one if rider i is transferred into the team in period j and zero otherwise. The model is adapted as follows:
maximize

$$\sum_{i=1}^{18}\sum_{j=1}^{6} p_{ij}x_{ij} \tag{11}$$

subject to

$$\sum_{i=1}^{18} c_{i1}x_{i1} + r_1 = 12000, \tag{12}$$

$$r_{j-1} + \sum_{i=1}^{18} c_{ij}x_{i,j-1} - \sum_{i=1}^{18} c_{ij}x_{ij} = r_j, \qquad \forall j = 2,...,6 \tag{13}$$

$$\sum_{i=1}^{18} x_{ij} = 10, \qquad \forall j = 1,...,6 \tag{14}$$

$$t_{ij} \geq x_{ij} - x_{i,j-1}, \qquad \forall j = 2,...,6 \tag{15}$$

$$\sum_{i=1}^{18} t_{ij} \leq 3, \qquad \forall j = 2,...,6 \tag{16}$$

$$x_{ij} \in \{0,1\}, \qquad \forall i = 1,...,18 \qquad \forall j = 1,...,6 \tag{17}$$

$$r_j \geq 0, \qquad \forall j = 1,...,6 \tag{18}$$

$$t_{ij} \in \{0,1\}, \qquad \forall i = 1,...,18 \qquad \forall j = 2,...,6 \tag{19}$$

Having introduced the binary variable t_{ij}, it is easy to state the transfer restriction by requiring that the total sum of t_{ij}'s must be smaller than or equal to three in each period, see inequalities (16). Now, we only have to make sure that t_{ij} equals one for each new rider i, that is, for each rider that is selected in the team in period j but was not yet present in the team in period $j-1$. Modeling this kind of conditional relations is very typical for integer programming formulations. Constraints (15) ensure this relation by requiring t_{ij} to be one, only if the rider is in the team in period j (i.e., when $x_{ij}=1$) and not in period $j-1$ (i.e., when $x_{i,j-1}=0$). Model (11)-(19) is the complete mixed integer programming model for the smaller instance of the ex post Gigabike optimization problem. Obviously, the same model can be used to find the optimal solution to the Gigabike problem including all the riders. Only the dimensions of the problem will change.

4 Class experiences

Since 2007, the Gigabike game is used to teach mixed integer programming modeling to students of the commercial engineering program as well as the MBA program at the University College Brussels. We had a lot of positive reactions on this class. First of all, thanks to the game and the sport element, students are much more motivated to find the optimal solution to the problem (and beat their opponents). Consequently, most students do a serious effort to understand the characteristics of the Gigabike game and particularly the optimization problem. Many students have asked us to get more time to solve the Gigabike puzzle, a request we don't often get when students have to model a (fictitious) business case. Some students even immediately started to formulate the problem as a MIP model, while we only asked them to find the best possible solution by puzzling in the Excel sheet (by a trial-and-error approach). The Gigabike application, hence, provides a good illustration of how a structured 'model formulation' approach eventually outperforms a quick trial-and-error approach that focusses on immediate success. This is an important lesson for students who might have to take complex business decisions in their future careers. Finally, given the positive feedback from many students, we are convinced that the Gigabike game helps to improve the students' MIP modeling skills.

5 Conclusion

Recently, (mixed) integer programming has gained in popularity. Indeed, advances in solution methods as well as computer hardware and software make solving such problems no longer an exhaustive challenge with respect to computing times. As a result, also the importance and relevance of (M)IP increased for solving real-life managerial-related problems. However, getting students really excited about (M)IP is not always an easy task. One way to overcome this is to let them experience how MIP can be used in applications taken from their own living environment. In this paper, we use an a posteriori game component of a popular cycling game (called the Gigabike game), involving real-life data of all current cyclists. Further on, we stimulate the students by adding a competitive aspect: the week before the solution is discussed in class acts as a real race among all students in finding the best solution for the game. An Excel sheet facilitates this search process. The game is very much suited for an undergraduate OR/MS course as a case concluding the (M)IP chapter. Indeed, the model can be gradually built up in three clearly distinctive phases, each phase covering an important class of (M)IP optimization methods. In the first phase, we consider the model as made up of independent knapsack problems (one for each period). The second phase introduces dependency between the periods, turning it into a multiperiod model. Finally, the third phase adds a typical IP constraint in order to obtain the right final model. The models of all phases can be solved using Frontline's standard version of Excel Solver©, as such making the models accessible to every student. The class experience with the game fosters our belief that using applications from the student's own world of interest are very valuable in getting the students excited about what they themselves call theoretical models (in this case, MIP models). The competitive component as introduction to the game, has impressively stimulated the student's interest which was clearly visible during the class. We hope that in this way the students learned about the power such models can provide and, more importantly, that they start to get a feeling about the importance and relevance these models might have in real-life management applications.

Acknowledgements

The authors are extremely grateful to Peter Samoy and Mark Vanderwegen for developing the unrivalled Gigabike game. Next to years of game pleasure, Gigabike was our inspiration for the educational tool presented in this paper.

References

Chlond MJ (2005) Classroom exercises in IP modeling: Su doku and the log pile. INFORMS Transactions on Education 5(2), URL http://ite.pubs.informs.org/Vol5No2/Chlond/

Chlond MJ (2008) Puzzle O.R. with the fairies. INFORMS Transactions on Education 8(2), URL http://ite.pubs.informs.org/

Chlond MJ, Toase CM (2002) IP modeling of chessboard placements and related puzzles. INFORMS Transactions on Education 2(2), URL http://ite.informs.org/Vol2No2/ChlondToase/

Chlond MJ, Toase CM (2003) IP modeling and the logical puzzles of Raymond Smullyan. INFORMS Transactions on Education 3(3), URL http://ite.pubs.informs.org/Vol3No3/ChlondToase/

Cochran JJ (2004) Introduction to the special issue: SpORts in the OR classroom. INFORMS Transactions on Education 5(1), URL http://ite.pubs.informs.org/Vol5No1/IntroCochran/

Dantzig GB (1957) Discrete variable extremum problems. Operations Research 5:266–277

Franz LS, Miller JL (1993) Scheduling medical residents to rotations: solving the large-scale multiperiod staff assignment problem. Operations Research 41(2):269–279

Kellerer H, Pferschy U, Pisinger D (2005) Knapsack Problems. Springer Verlag

Martello S, Toth P (1990) Knapsack Problems: Algorithms and Computer Implementations. John Wiley & Sons

Rohn E (1987) A new LP approach to bond portfolio management. Journal of Financial and Quantitative Analysis 22:439–467

Sniedovich M (2002) OR/MS games: 1. a neglected educational resource. INFORMS Transactions on Education 2(3), URL http://ite.informs.org/Vol2No3/Sniedovich/

Stadtler H (1996) Mixed integer programming model formulations for dynamic multi-item multi-level capacitated lotsizing. European Journal of Operational Research 94(3):561–581

Trick M (2004) Using sports scheduling to teach integer programming. INFORMS Transactions on Education 5(1), URL http://ite.pubs.informs.org/Vol5No1/Trick/

Winston W (2004) Operations Research: Applications and Algorithms. Thomson Learning Brooks/Cole