



Demand-driven scheduling of movies in a multiplex

Jehoshua Eliashberg^a, Quintus Hegie^b, Jason Ho^c, Dennis Huisman^d, Steven J. Miller^{e,1}, Sanjeev Swami^f, Charles B. Weinberg^g, Berend Wierenga^{h,*}

^a The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania 19104, United States

^b School of Economics, Erasmus University Rotterdam, The Netherlands

^c Faculty of Business Administration, Simon Fraser University, Burnaby, Canada BC V5A 1S6

^d Econometric Institute, Erasmus University Rotterdam, The Netherlands

^e Department of Mathematics and Statistics, Williams College, Williamstown, MA 01267, United States

^f Department of Management, Faculty of Social Sciences, Dayalbagh Educational Institute, Agra – 282005, UP, India

^g Sauder School of Business, University of British Columbia, Vancouver, Canada BC V6T 1Z2

^h Rotterdam School of Management, Erasmus University, The Netherlands

ARTICLE INFO

Keywords:

Movie modeling
Optimization of movie schedules
Integer programming
Column generation
Demand forecasting

ABSTRACT

This paper is about a marketing decision support system in the movie industry. The decision support system of interest is a model that generates weekly movie schedules in a multiplex movie theater. A movie schedule specifies, for each day of the week, on which screen(s) different movies will be played, and at which time(s). The model integrates elements from marketing (the generation of demand figures) with approaches from operations research (the optimization procedure). Therefore, it consists of two parts: (i) conditional forecasts of the number of visitors per show for any possible starting time, and (ii) a scheduling procedure that quickly finds a near optimal schedule (which can be demonstrated to be close to the optimal schedule). To generate this schedule, we formulate the “movie scheduling problem” as a generalized set partitioning problem. The latter is solved with an algorithm based on column generation techniques. We tested the combined demand forecasting/schedule optimization procedure in a multiplex in Amsterdam, generating movie schedules for fourteen weeks. The proposed model not only makes movie scheduling easier and less time consuming, but also generates schedules that attract more visitors than current “intuition-based” schedules.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

The motion picture industry is a prominent economic activity with total worldwide box office revenue of \$ 26.7 billion in 2007, of which \$ 9.6 billion is in the U.S.A.² Movie forecasting and programming in practice tend to be associated with intuition rather than formal analysis, and this also characterizes the tradition of decision-making in the film industry. However, many problems in the film industry are actually quite amenable to model building and optimization, as movie executives have increasingly recognized. In this paper, we focus on one such problem: the detailed scheduling of a movie theater in Amsterdam. Movie marketing and modeling is a new area for application of marketing decision support systems, originally developed for the fast-moving consumer goods industry (Wierenga & Oude Ophuis, 1997). Developing models that deal with real problems of

decision-makers in practice has been an issue of continuing concern in marketing (Eliashberg & Lilien, 1993; Leeflang & Wittink, 2000). In this project, we have followed a “market-driven” approach, where we have designed a model with capabilities that are custom-designed to managerial needs (Roberts, 2000).

A movie program or schedule in a theater is designed for a week. In the Netherlands, for example, a new movie week starts each Thursday. Therefore, a movie theater has to prepare a new movie schedule at the beginning of every week. This is particularly complex for the multiplex theaters, the increasingly dominant movie theater format around the world. In the Netherlands, multiplexes with eight or more screens represent 24% of all movie theater seats and 34% of the total box office. It is clear that programming such large cinema facilities is not an easy matter.

For each week’s movie program, management must determine what movies will be shown, on which screens, on which days, and at what times. Typically, on each screen, a theater can accommodate three to five showings per day, where a “showing” is defined as the screening of one movie, including trailers and advertisements. This means that a 10-screen theater needs to program around 280 showings per week. Presently, this programming is mostly done manually with pencil and paper by specialists in the theater company.

* Corresponding author.

E-mail addresses: eliashberg@wharton.upenn.edu (J. Eliashberg),

quintus@quintushegie.com (Q. Hegie), jason_ho_3@sfu.ca (J. Ho), huisman@few.eur.nl (D. Huisman), Steven.J.Miller@williams.edu (S.J. Miller), sswami1853@gmail.com (S. Swami), weinberg@sauder.ubc.ca (C.B. Weinberg), bwierenga@rsm.nl (B. Wierenga).

¹ Partially supported by NSF Grant DMS0600848.

² MPA (Motion Picture Association of America) Annual Report 2007.

Despite the combination of a programmer's analytical mind and a broad knowledge about movies and about the audience of individual theaters, often based on many years of experience, it is our belief that an analytical system can help in movie programming. This would not only relieve theaters from a repetitive labor-intensive task, but also achieve a better performance in scheduling than the current mental, manual procedure.

The programming problem for an individual movie theater consists of two stages: (i) the selection of the list of movies, i.e., the movies to be shown over the course of the particular week and (ii) the scheduling of these movies over screens, days, and times. Stage (i) (macro-scheduling) includes making agreements with movie distributors and is completed before stage (ii). In this study, we develop a solution for the second stage (micro-scheduling), which involves constructing detailed schedules for where (which screen(s)) and when (days, hours) the different movies will play. For analytical procedures that deal with the first stage, (i.e., deciding on the movies to be shown in a particular week), see Swami, Eliashberg, and Weinberg (1999), and Eliashberg, Swami, Weinberg, and Wierenga (2001).

Our scheduling problem has two sub-problems. First, we need to answer the question: if a particular movie were shown on a particular day at a particular time, how many visitors would attend? Inputs for making these forecasts are based on numbers of observed visitors in previous weeks (for existing movies), various characteristics of the movie (for newly released movies), coupled with information about variables such as specific events (holidays) and the weather. Making conditional forecasts is not an easy task, and it belongs to the realm of marketing. Second, given this demand assessment, we have to find the schedule that maximizes the number of visitors for the week, given constraints such as theater capacity and run-times of the movies. This is also a non-trivial problem, for which the discipline of operations research is useful. The specific solution approach employed here is column generation, a method designed to solve (integer) linear optimization problems with many variables/columns with relatively few constraints. The remainder of this paper describes how, by combining the two disciplinary approaches, a solution to the movie micro-scheduling problem was obtained. The structure of the movie scheduling problem of this paper is similar to the problem of optimal scheduling of TV programs, which has been addressed in the marketing/OR literature (Danaher & Mawhinney, 2001; Horen, 1980; Reddy, Aronson, & Stam, 1998). In the TV program scheduling problem, there is also a demand forecasting module that predicts the size of the television audience, and an optimization procedure that finds the best schedule. Several new and unique features arise in our problem, however, as the TV scheduling problem has no constraint for audience size and also the same program cannot be consecutively scheduled many times.

In this paper, we first take a closer look at the movie scheduling problem (Section 2). In Section 3, we describe the column generation approach to the optimization problem. In Section 4, we discuss the method that we developed to conditionally forecast the number of visitors of a show. In Section 5, we apply the complete procedure through a model we call SilverScheduler to fourteen movie weeks of the De Munt theater, a multiplex with 13 screening rooms located in central Amsterdam. The De Munt theater is owned by Pathé Nederland, which has the largest share of the movie exhibition business in Holland. With over 1 million visitors a year, the De Munt theater is the second largest theater in the Netherlands. The last part of the paper (Section 6) discusses the results and places them in perspective, and discusses issues for future research.

2. Problem description and the research project

As movie theaters have evolved from single screen theaters to multiplexes and even megaplexes, the problems of scheduling movies

on screens has become increasingly complex. Here, we will now discuss some of the key reasons for that complexity.

First, there are a large number of different movies that the theater wants to show in a typical week. This number is typically larger than the number of screens. Moreover, these movies have different run-times. For example, among the different movies running in the De Munt theater in Amsterdam during our observation period, the run-time was as short as 71 min for *Plop en Kwispel* (a kids movie) and as long as 240 min for *Ring Marathon*.

Second, the number of seats per screening room differs. In the De Munt theater, the smallest room has 90 seats and the largest room has 382. For some movies, there are contractual arrangements with distributors about the specific screening room in which a movie will be shown. This is typically the case for newly released movies, which distributors like to have shown in large rooms. For other movies, the theater management is completely free to decide where to screen them.

Third, there is variability in demand for movies. In our sample, for example, in May 2005, the most popular movie drew 8027 visitors in a week, and the least popular drew only 133 visitors. Additionally, movie demand varies by time of day and day of week. For example, we found that Saturday evening at 8 pm is the most appealing time for moviegoers. The expected demand for a movie influences in what screening room it will be shown. Moreover, the demand for some movies may be so large that it could be double or triple booked.

Fourth, there are different genres of movies, and this also has implications for their scheduling. For example, children's movies should preferably be shown at times when children are free from school (weekend and Wednesday afternoon³), and will not be shown during evenings.

Fifth, there are many constraints posed by the logistics of a movie theater. An obvious limitation is the time that the theater is open. The closing time is often determined in part by schedule of public transit. Non-drivers attending the last show need to be able to get home. Other logistical constraints are the time needed for cleaning after a showing (which depends on the size of the room), and capacity limitations of the ticket office, corridors, staircases, and concession sales counters. To provide high consumer satisfaction with the theatrical experience, crowding should be avoided as much as possible and major movies with many visitors should not start at the same time, especially if they are on the same floor. The screening rooms of the De Munt theater are on two different floors, with rooms 1 to 8 on one floor and rooms 9 to 13 on the other. To avoid overcrowding, a rule was introduced that, at peak times (evenings and Saturday and Sunday afternoons), at most one movie can start at any given time on screens 1 to 8, with the same applying to screens 9 to 13. This spread in movie start times is also favorable for concession counter sales, which thus have a more evenly distributed demand. Furthermore, management wanted to limit the numbers of screen changes, i.e., the number of times that another movie has to be mounted on the projection system for a particular room. Movie reels are physically quite large and difficult to handle and there is a (small) probability that the movie reel will fall and be damaged.

Finally, management may impose specific requests that are expected to add to the theater's revenues. For example, in our study, management specified that the time between two movie start times should not be more than 20 min. In that case, an impulse movie visitor, looking for entertainment without strong preference for a particular movie, never has to wait more than 20 min.

Regarding the objective function, management asked us to maximize the number of visitors, which is consistent with their own internal policies (e.g., the reward system for their theater managers). Another possible objective is profit maximization. However, the cost

³ Primary school children in the Netherlands have free time on Wednesday afternoons.

and revenue structure behave such that attendance maximization and profit maximization are likely to provide similar results. Admission prices of movies are practically flat (very close to € 7.50). There is a slight differentiation in margin, implying that older movies (having run for more weeks) have a higher margin for the theater. However, the audiences are typically small for older movies such that there is not much of an incentive for the theater to schedule more of these shows because of the larger margin. We tested attendance maximization versus profit maximization for a particular week and found only one difference: one showing of a movie with a lower margin was replaced by a showing of a movie with a higher margin. The effect on profit was negligible.

Consequently, the problem to be addressed here is how to generate a schedule of different movies on different screens during a given week, which obeys all managerial constraints/requests and maximizes the number of visitors.

2.1. The study

Most theaters solve this scheduling problem by “head and hand.” “Movie programmers” work with hard-copy planning sheets to fill the capacity of the theater, accounting for the various constraints to the extent possible. In doing so, programmers follow certain procedures (e.g., choose the size of the screening room for a movie based on the expected attendance), and use a mixture of hard facts (typically attendance figures from the past week) and intuition (e.g., the effect of an important soccer match on TV on the number of theater visitors for a movie). It takes a lot of experience to become a skilled movie programmer. Even then, the solution is never perfect. For a human mind, it is practically impossible to find the best solution while simultaneously honoring all constraints. Moreover, the schedules are made under time pressure. As mentioned, in a movie theater, every Thursday/Friday⁴ a new movie program schedule starts running, and it has to be finalized on the preceding Monday. The critical time window is the Monday morning after attendance figures from the most recent weekend become available, and before information about the new schedule has to be sent to newspapers and posted online.

The purpose of this study is to develop an efficient procedure that makes it possible to automate the movie programming process as described above. We developed a mathematical procedure that produces the (almost) optimal schedule, given the demand assessment and various requests/constraints. The basic requirements for such an algorithm are: (i) it should work quickly, (ii) it should deliver the output in a format that can be directly entered into the theater's planning procedure, (iii) it should also be easy for a manager to make last-minute changes to the schedule recommended by the procedure. In decision situations like this one, there can always be some new information (e.g., a strike or a sudden change in the weather) not included in the model that may induce the manager to make last-minute changes in the schedule. While the generation of a recommended schedule can be automated, human judgment still remains vital for the implementation of a schedule.

We were asked to work on this problem by Pathé, the largest movie exhibitor in the Netherlands. They view the current manual programming procedure as being too cumbersome, not always consistent, and time consuming. Our purpose was not to just save the time of the programmer, but also to improve in terms of generating movie schedules that attract more visitors. It was decided to take the setting of one particular Pathé location, the De Munt Theater, as our empirical environment. Management provided complete access to the internal data they had, and closely monitored the project and its results.

A key input to the scheduling algorithm is demand information about the movies in the movie list. For each movie in the list, we need to forecast the number of visitors that this movie will attract in any given showing at the particular facility. This estimate has to be available for each day of the week and for each different possible starting time of the movie. For this purpose, we developed a forecasting procedure with two modules. The first module is for movies that have already been running. The observed numbers of past visitors are used to estimate a forecasting model. The second module is for newly released movies, where the numbers of visitors are forecasted using the characteristics of a movie as predictor variables. These forecasting procedures are described in more detail after the discussion of the scheduling algorithm. The complete model, the scheduling algorithm integrated with the demand assessment procedure, *SilverScheduler*, and its results applied to the De Munt theater are discussed after the description of its scheduling and the forecasting components.

3. A column generation approach to solve the movie scheduling problem

To produce a movie program for a certain week, we need to find schedules for the different days in that week. We define the *movie scheduling problem* (MSP) as the problem of finding the optimal movie program for a single day given the list of movies to be shown (the “movie list”), the run-times of these movies, forecasted demand, capacities of different screening rooms, and information about contractual agreements with distributors about screening rooms for particular movies, accounting for the different constraints mentioned in Section 2. In Section 5, we describe a procedure to prepare a week's schedule by solving a sequence of MSPs.

The MSP has many similarities to other well-known scheduling problems, which are often formulated as set partitioning or covering problems (eventually with additional constraints). We also use a set partitioning type of formulation. A drawback of this kind of formulation is the large number of variables, which can be overcome by using column generation techniques. See Barnhart, Johnson, Nemhauser, Savelsbergh, and Vance (1998), Lübbecke and Desrosiers (2005), and Desaulniers, Desrosiers, and Solomon (2005) for a general introduction to column generation.

3.1. Mathematical formulation

Before we present the mathematical formulation of the MSP, we first introduce some notation. Analogous to “time-space networks” in transportation problems such as vehicle and crew scheduling, here we use *time-movie networks*. These networks are acyclic directed graphs denoted by $G^s = (N^s, A^s)$, and are defined for each screen s . Furthermore, let M and S be the set of movies and screens, respectively. Recall that, due to capacity restrictions and contractual agreements, not all movies can be shown on each screen; therefore, we define M^s as the subset of movies that can be shown on screen s . We denote for each movie m its duration and cleaning time, dr_m and cl_m , respectively. Cleaning time is dependent on the number of visitors that a movie is expected to attract. Define T as the set of possible time points at which a movie can start, where t_i is the time corresponding to time point i . In each graph, a node (i, m) corresponds to starting movie m at time point i on screen s . We also define a source and a sink. There are arcs from the source to all intermediate nodes and arcs from the nodes to the sink. If, on a screen only one movie is allowed to play throughout the day, an arc is defined between each pair of nodes (i, m) and (j, n) if $t_j \geq t_i + dr_m + cl_m$ with $m = n$. In Fig. 1, the time-movie network is depicted for this particular case. For instance, there is an arc between nodes $(1, 1)$ and $(3, 1)$, because the duration and cleaning time of movie 1 is not longer than the time between time points 1 and 3. A path in the network corresponds to a feasible schedule for the whole day on

⁴ In the Netherlands the new movie week starts on Thursday, in the USA and Canada this is on Friday.

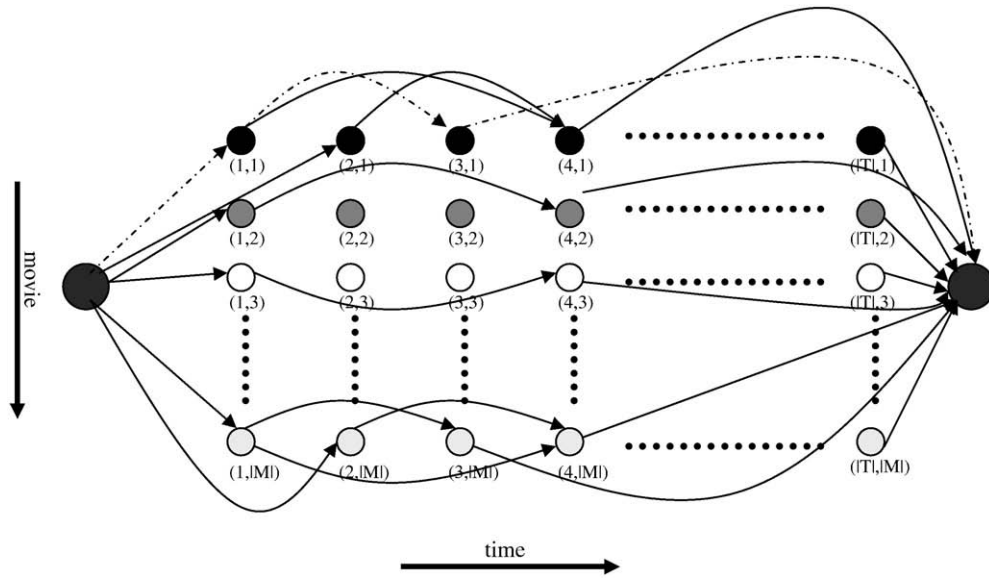


Fig. 1. A time-movie network for one screen (with one movie per screen).

one screen. The path corresponding to showing movie 1 at time points 1 and 3 is illustrated in the figure by the dotted arcs.

If two movies can be shown on the same screen, the network is extended to two layers. In each individual layer, there are only arcs between nodes corresponding to the same movie, while arcs between the layers are between nodes with different movies. Since we want to limit movie switching on one screen (this takes time and effort), a penalty Q is introduced when the arc between the different layers is chosen. In these networks, each path from source to sink is a feasible schedule for one screen. The cost of a path is defined as c_p , which equals the sum of the costs of the individual arcs in the path. Each arc (i,m,j,n) has as costs $-\min\{d_{im},cap_s\}$, which is the minimum of the expected demand for movie m at time point i and the capacity of screen s if (i,m) and (j,n) are in the same layer, and $Q - \min\{d_{im},cap_s\}$ if the nodes are in different layers.⁵ The basic movie scheduling problem, in which every movie can only be shown on one screen, can now be seen as finding a path from source to sink in each network such that the total cost of the paths is minimized and each movie is in exactly one of the paths. In mathematical terms, this can be directly formulated as a set partitioning problem with decision variables x_p^s , which is 1 if path p is selected in network G^s and 0 otherwise:

$$\min \sum_{s \in S} \sum_{p \in P^s} c_p x_p^s \tag{1}$$

$$\text{s.t.} \sum_{s \in S} \sum_{p \in P^s} a_{mp}^s x_p^s = 1 \quad \forall m \in M \tag{2}$$

$$\sum_{p \in P^s} x_p^s = 1 \quad \forall s \in S \tag{3}$$

$$x_p^s \in \{0, 1\} \quad \forall s \in S, p \in P^s. \tag{4}$$

Here P^s is the set of all paths in G^s , and the parameter a_{mp}^s is 1 if movie m is in path p corresponding to screen s and 0 otherwise. Note that the formulation can be extended to the situation where the same movie can be shown on different screens during the day. Because this

⁵ Consistent with standard practice in mathematical programming literature, we formulate the problem as a minimization problem. Here, minimizing “cost” is equivalent to maximizing demand.

situation is not the case in our application, we do not take it into consideration.

Of course, not all practical aspects of the Pathé management problem have been taken into account in this mathematical formulation. In the remainder of this section, we discuss three important aspects that we can take into account by adding extra sets of constraints.

The first aspect deals with the fact that only one movie can start at the same time on a floor during crowded periods. This is coded by the following set of constraints:

$$\sum_{s \in S} \sum_{p \in P^s} b_{ip}^s e_f^s x_p^s \leq 1 \quad \forall f \in F, i \in T_1, \tag{5}$$

where F is the set of floors and T_1 is the set of time points over which this condition should hold. Parameters b_{ip}^s and e_f^s are 1 if at time point i a movie starts in path p belonging to screen s and if screen s is on floor f , respectively, and 0 otherwise.

As mentioned earlier, it is important for management that in certain pre-specified intervals (in our case every 20 min), there is at least one movie starting. However, this extra requirement can result in an excessive reduction in the number of visitors. Therefore, we add a 0/1 decision variable y_l indicating whether these constraints are satisfied (it is 1 if the constraint is not satisfied). The following set of constraints is added to the formulation:

$$\sum_{s \in S} \sum_{p \in P^s} (b_{ip}^s + \dots + b_{jp}^s) x_p^s + y_l \geq 1 \quad \forall l \in L, i, j \in T_1, \tag{6}$$

where L is the set of intervals and T_1 are the time points in interval L . Moreover, we add the term $\sum_{l \in L} R y_l$ in the objective function, where R is a penalty for violating one of these restrictions.

The third aspect that we explicitly consider deals with the theater's closure. Management wants to start the cleaning process of the theater before closing time. Therefore, in a certain number of screens r , the last movie has to be finished before a certain time:

$$\sum_{s \in S} \sum_{p \in P^s} h_p^s x_p^s \geq r, \tag{7}$$

where h_p^s is 1 if the last arc in path p starts from a node (i,m) where $t_i + dr_m$ is less than a certain pre-specified time.

3.2. The proposed algorithm

As mentioned above, column generation techniques have been employed in optimization problems to deal with the large number of variables. The general idea behind column generation, introduced by Dantzig and Wolfe (1960), is to solve a sequence of reduced problems, where each reduced problem only contains a small portion of the set of variables (columns). After a reduced problem is solved, a new set of columns is obtained by using dual information of the solution. The column generation algorithm converges once it has been established that the optimal solution, based on the current set of columns, cannot be improved by adding more columns. Then the optimal solution of the reduced problem is the optimal solution to the overall problem. We will refer to the reduced problem as the *restricted master problem* and to the problem of generating a new set of columns as the *pricing problem*.

Traditionally, column generation for integer programs have been used to solve their LP-relaxations. However, we use it here in combination with Lagrangian relaxation (for the reader unfamiliar with Lagrangian relaxation, see Fisher (1981)). This approach has been chosen for the following reasons.

- For this problem, a set of columns generated to compute the lower bound turns out to be a set from which we can select a reasonably feasible solution.
- Since we compute a lower bound on the optimal solution, we obtain an indication about the quality of the constructed feasible solution.
- Lagrangian relaxation has been shown to provide tight bounds for set partitioning type problems (see e.g., Beasley (1995)).

Combining column generation with Lagrangian relaxation has recently been applied to several other situations, e.g., integrated vehicle and crew scheduling (Huisman, Freling, & Wagelmans, 2005) and integrated airline fleetling and crew-pairing (Sandhu & Klabjan, 2007). See Huisman, Jans, Peeters, and Wagelmans (2005) for a more comprehensive discussion on combining these two techniques. In Fig. 2, we provide a schematic overview of the algorithm.

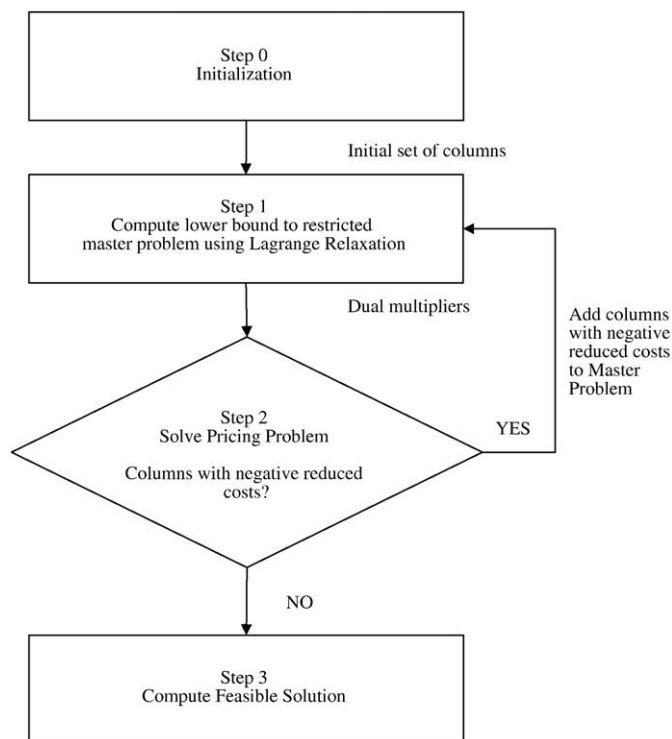


Fig. 2. Schematic picture of the algorithm.

Table 1

Computational results of the movie scheduling algorithm: the gap from the optimum is 1.58% on average.

Day	# Movies	Gap (abs)	Gap (in %)	Cpu ^a Step 1	Cpu Step 2	Total Cpu ^b
20050303	19	42	1.34	60	65	145
20050310	18	23	1.05	42	50	116
20050317	17	27	1.40	87	64	185
20050324	19	30	1.52	56	54	117
20050407	18	28	1.32	56	46	110
20050414	16	33	2.04	49	44	101
20050421	18	18	1.00	86	75	176
20050428	17	18	1.59	44	46	117
20050505	22	25	3.27	105	215	441
20050519	17	28	2.67	62	61	138
20050526	15	6	0.60	43	27	74
20050602	15	22	1.36	58	34	98
20050630	19	23	1.36	44	56	107
20050714	20	21	1.52	102	107	223
Average	18	24.57	1.58	64	67	153

^a The CPU times are measured in seconds on a Pentium III, 1 GHz PC (384 MB RAM).

^b Total CPU time also includes Step 0 and 3.

After an initialization step, we iterate between Steps 1 and 2. In Step 1, a good set of dual multipliers is computed by solving a Lagrangian dual problem. Moreover, a lower bound on the optimal solution given the current set of columns is computed. New columns are then generated in Step 2 by solving a shortest path problem. As long as there are columns with negative reduced costs, they are added to the restricted master problem and we iterate between Steps 1 and 2. When there are no columns left with negative reduced cost, we have found a lower bound on the overall problem, and we go to Step 3. Here, we compute the optimal solution of the problem given all columns at the end of Step 2. Because this set does not necessarily contain all columns in the overall optimal solution, we cannot guarantee that the found solution is the overall optimal solution. However, we can calculate the gap between the lower bound and the value of this solution. If this gap is small, we have either found the optimal solution or we are very close to it. We refer the reader interested in the details of the algorithm to Appendix C.

3.3. Illustrative computational results

To help evaluate the algorithm's performance, we conducted several tests on Pathé's data. We constructed daily schedules for a number of days with varying numbers of movies. After discussing the results with Pathé, we set the values for parameters Q and R (penalties) to 100 and 10, respectively. The choice for Q comes from the fact that setting the parameter to an excessively high value guarantees that no more than 2 movies per screen will be scheduled. Parameter R is chosen in such a way that there is almost always a movie starting within a 20-minute interval. Sensitivity analysis showed that the number of visitors is not very sensitive to the chosen values for these penalties.

In Table 1, for 14 instances (days), we report the number of movies, the absolute and relative gap between upper and lower bounds on the optimal solution, and the computation times (in seconds) for the master problem (Step 1), pricing problem (Step 2), and the complete algorithm. The number of screens is equal to 13 for all instances. The underlying graphs as illustrated in Fig. 1 contain about 2000+ nodes and 300,000+ arcs. As a result, for each screen, there are millions of possible schedules. From these numbers, column generation is clearly required to solve the problem.

As seen in Table 1, the relative gap is between 0.60 and 3.27%. The average gap is 1.58%. The total computation time is very reasonable. On average, it only requires about 2.5 min. The most time, as expected, is spent in the column generation part of the algorithm, where the times to solve the master and pricing problems are almost equal.

4. Conditional demand forecasting

Developing a week's schedule requires forecasts for A_{jt} , the attendance for a screening of movie j starting at time t , for all movies available for showing at all possible starting times (broken down into hourly intervals) for every day of that week. These forecasts are required to be available each Sunday for the scheduling algorithm so that recommendations to management become available on Monday. To use the most recent data possible, the forecasting procedure had to be completed in less than 12 h. This essentially ruled out some simulation-based estimation procedures, such as hierarchical Bayes models (e.g., Ainslie, Dreze, & Zufryden, 2005). We opted for simplicity and efficiency and chose to generate our forecasts using linear regression models, estimated by OLS. We explored the use of weighted least squares, but this did not improve the forecasts.

The notion that the age of a movie is an important factor in determining its appeal to the audience is common knowledge. A number of empirical studies report that moviegoers' demand for most movies typically decreases over time (e.g., Krider & Weinberg, 1998; Lehmann & Weinberg, 2000; Sawhney & Eliashberg, 1996). Thus, we used a two-parameter exponential model to capture these effects. For most movies, these parameters can be estimated from detailed attendance records at De Munt, maintained by Pathé. For movies for which we did not have De Munt Theater attendance data for at least two weeks, we developed a secondary procedure to estimate the two parameters of the exponential model. This procedure is described in more detail below. There are also other effects, such as holidays, which may influence attendance, and we also included control variables for such effects.

In addition to these weekly effects, there are more micro-effects that need to be considered. In particular, we introduce variables that account for day of the week and time within a day (in hourly intervals) at which movies are shown. We also allow for other detailed factors, such as weather (temperature, precipitation) and whether the Dutch national football (soccer) team is playing a major international match. We divided our procedure into three steps.

4.1. Method description

4.1.1. Step 1: demand model estimation

In the first step, we use past attendance figures to estimate a demand model, which separates different time-varying patterns. Formally, we model A_{jt} , the attendance for a showing of movie j starting at time t , as:

$$\begin{aligned} \text{Log}(A_{jt}) = & \theta_j + \lambda_j \cdot \text{AGE}_{jt} + \sum_h \beta_h \cdot I_{\{h\}t} + \sum_d \omega_d \cdot I_{\{d\}t} \quad (8) \\ & + \sum_v \gamma_v \cdot I_{\{v\}t} + \delta_{\text{SATNIGHT}} \cdot \text{SATNIGHT}_t \\ & + \delta_{\text{SUNPM}} \cdot \text{SUNPM}_t + \delta_{\text{NG}} \cdot \text{NG}_t + \delta_{\text{DTEMP}} \cdot \text{DTEMP}_t \\ & + \delta_{\text{DPRECIP}} \cdot \text{DPRECIP}_t + \varepsilon_{jt}, \end{aligned}$$

where

- $\text{AGE}_{jt} \equiv$ number of weeks at time t since movie j 's first regular showing at the theater
- $I_{\{h\}t} \equiv$ indicator variable for the event that starting time t is within hour h , $h \in \{10 \text{ am} - 10 \text{ pm}\}$
- $I_{\{d\}t} \equiv$ indicator variable for the event that starting time t is on day d , $d \in \{\text{Monday, Tuesday, ..., Sunday}\}$ ⁶
- $I_{\{v\}t} \equiv$ indicator variable for the event that starting time t is on an Amsterdam holiday or school vacation v , $v \in \{\text{Spring vacation, May vacation, Ascension Day, Whit Sunday, Whit Monday,}$

Easter weekend, Summer vacation, Fall vacation, Christmas vacation}

- $\text{SATNIGHT}_t \equiv$ indicator variable for the event that starting time t is between 11 pm and 1 am of a Saturday
- $\text{SUNPM}_t \equiv$ indicator variable for the event that starting time t is between 2 pm and 6 pm of a Sunday
- $\text{NG}_t \equiv$ indicator variable for the event that starting time t occurs during the time when the Dutch national team is playing in a major tournament and is televised
- $\text{DTEMP}_t \equiv$ the maximum temperature of the day starting time t is in
- $\text{DPRECIP}_t \equiv$ the indicator variable for the event that starting time t is in a day with nonzero precipitation (e.g., rain or snow).

θ_j and λ_j are the two movie-specific parameters capturing the time trend of an individual movie's attractiveness across weeks. In particular, we assume that the attractiveness of each movie title follows a pattern of exponential decay, with θ_j characterizing the scale of the movie's attractiveness (opening strength) and λ_j capturing the weekly decay of its attractiveness. Decay occurs for most movies, but there are exceptions (*The Blair Witch Project* is a famous example).

The other type of time-related variation inherent in attendance is the moviegoers' time preference of watching a movie. These effects are captured by three sets of parameters, β_h , ω_d , and γ_v . The γ_v 's capture the "leisure time of year" effect. Because moviegoers tend to have more free time for leisure activities, such as going to movie theaters during holidays or school vacations vice during normal work or school days, we expect all of these parameters to be positive. Five school vacation periods, namely spring, May, summer, fall, and Christmas vacations are first identified. Three public holidays outside these vacation periods, namely Ascension Day, Whit weekend, and Easter weekend, are then added to capture the other potential holiday effects.

The day of week effect is captured by the ω_d 's. Seven parameters ω_{MON} , ω_{TUE} , ω_{WED} , ω_{THU} , ω_{FRI} , ω_{SAT} , and ω_{SUN} capture this type of demand variation over time. At the more micro-level, the β_h 's capture the time of day effect. We expect the β_h 's corresponding to daytime to be smaller than those for evening are. While normal operating hours of the De Munt end at 11 pm, occasionally some movies are shown after 11 pm on Saturday. Therefore, we use the parameter δ_{SATNIGHT} to capture the effect of this extension on Saturday. We also include a parameter to represent Sunday afternoons (δ_{SUNPM}), as our preliminary data analysis showed that attendance for Sunday was typically higher in the afternoon compared to other days.

We also need to control for three additional systematic shifts of the attendance in our demand model: (1) the existence of a strong competing leisure activity (we focus on major tournament football games played by the Dutch national team), (2) the highest temperature of the day, and (3) the existence of rain or snow during the day. The parameters, δ_{NG} , δ_{DTEMP} , and δ_{DPRECIP} capture these systematic shifts.

Assuming $\varepsilon_{jt} \sim N(0, \sigma^2)$, we estimate Eq. (8) by OLS, using all the available attendance figures up to the Sunday preceding the Thursday on which the new movie program starts. When additional attendance data are added each week, the demand model is re-estimated with the extended dataset.

The current demand model assumes that the attendance of movie j starting at time t is not affected by other specific movies that are shown at the same time. More complex models that capture substitution effects are explored in Ho (2005), but they provide limited improvement in predictive power, while greatly increasing the complexity of the mathematical optimization. Consequently, we retained Eq. (8) as the demand model for the current project.

4.1.2. Step 2: determination of movie-specific parameters

In capturing individual movies' attractiveness, the two vectors of movie-specific parameters, θ_j and λ_j , are crucial to our demand

⁶ For completeness, all possible values of the indicator variable are listed in the text. The base case for all indicator variables is stated in the tables reporting results.

forecasting. We separate our forecasting procedure into two cases: (1) movies with attendance data for 2 or more weeks and (2) newly released movies with no data (as they have not yet been shown), or with one week of data, if they have been shown for just one week at the De Munt theater. When there are two or more weeks of data for a movie title (case 1), we have sufficient information to estimate both θ_j and λ_j from Eq. (8). To make forecasts for such a movie, we use estimates of θ_j and λ_j , obtained from the first step with the most recent data. On the other hand, for movies with either limited or no attendance data (case 2), there are no estimates of θ_j (and/or λ_j) from the first step. To deal with this issue, we first built a regression model that relates θ_j (and/or λ_j) to movie attributes that might explain θ_j (and/or λ_j). We then use estimates obtained from this regression model and the values of the attributes for a new movie to estimate that movie's value of θ_j (and/or λ_j). This approach is consistent with meta-analysis (see, for example, Sultan, Farley, & Lehmann, 1990).

More specifically, we regress, for example, the values of θ_j , the estimates of θ_j from the first step, on various movie attributes using the following model:

$$\hat{\theta}_j = \rho_0 + \rho_{OU} \cdot O - USA_j + \rho_{OD} \cdot O-DUTCH_j + \rho_{OO} \cdot O-OTHER_j \quad (9)$$

$$+ \rho_{LE} \cdot L-ENGLISH_j + \rho_{LD} \cdot L-DUTCH_j + \rho_{LO} \cdot L-OTHER_j$$

$$+ \rho_{DD} \cdot D-DUTCH_j + \rho_{SQ} \cdot SEQUEL_j + \rho_{FC} \cdot FRANCHISE_j$$

$$+ \rho_{BOW1} \cdot BOW1_j + \rho_{TNW1} \cdot TNW1_j + \rho_{BOW1W2} \cdot BOW1W2_j$$

$$+ \rho_G \cdot MPAA-G_j + \rho_{PG} \cdot MPAA-PG_j + \rho_{PG13} \cdot MPAA-PG13_j$$

$$+ \rho_R \cdot MPAA-R_j + \rho_{DRA} \cdot DRAMA_j + \rho_{ACT} \cdot ACTION_j$$

$$+ \rho_{COM} \cdot COMEDY_j + \rho_{RCOM} \cdot RCOMEDY_j + \rho_{MUS} \cdot MUSICAL_j$$

$$+ \rho_{SUS} \cdot SUSPENSE_j + \rho_{SF} \cdot SCIFI_j + \rho_{HOR} \cdot HORROR_j$$

$$+ \rho_{FAN} \cdot FANTASY_j + \rho_{WES} \cdot WESTERN_j + \rho_{ANI} \cdot ANIMATED_j$$

$$+ \rho_{ADV} \cdot ADVENTURE_j + \rho_{DOC} \cdot DOCU_j + \epsilon'_j.$$

See Appendix A1 for definitions of these attributes. For Dutch-made and other movies for which there was no U.S. release, attributes such as US box office revenues are set to zero. The indicator variable is sufficient to capture the fact that no additional information from the US box office is available for estimation purposes.

Using parameter estimates, we then predict θ_j for new movies that have yet to be opened (case 2) by inserting the new movies' characteristics into Eq. (9). For the few movies that opened simultaneously in the US and the Netherlands, averages of all U.S. movies in our database were used as the value of the US box office variables. For movies with one week of data, a similar procedure is used for the decay rate, λ_j .

4.1.3. Step 3: attendance forecasts

In this step, we use Eq. (8) to generate forecasts for all movies available for screening (cases 1–2) at all possible times in the new movie program. Specifically, we take all the parameter estimates from the first step and, when needed, the estimates of θ_j and λ_j from the second step, to forecast the expected attendance for movie j starting at future time t , $E(A_{jt})$:

$$E(A_{jt}) = \exp(\theta_j + \hat{\lambda}_j \cdot AGE_{jt} + \sum_{\forall h} \hat{\beta}_h \cdot I_{\{h\}t} + \sum_{\forall d} \hat{\omega}_d \cdot I_{\{d\}t} \quad (10)$$

$$+ \sum_{\forall v} \hat{\gamma}_v \cdot I_{\{v\}t} + \hat{\delta}_{SATNIGHT} \cdot SATNIGHT_t + \hat{\delta}_{SUNPM} \cdot SUNPM_t$$

$$+ \hat{\delta}_{NG} \cdot NG_t + \hat{\delta}_{DTEMP} \cdot DTEMP_t + \hat{\delta}_{DRECIPI} \cdot DRECIPI_t$$

$$+ \hat{\sigma}^2 / 2).$$

As our parameter estimates are obtained from Eq. (8), which regresses the logarithms of A_{jt} , there will be a downward bias in the forecasts in $E(A_{jt})$ (Hanssens, Parsons, & Schulz, 2003, p.395). We use the correction factor $\hat{\sigma}^2/2$ in Eq. (10) to compensate for this downward bias ($\hat{\sigma}^2$ is the estimate for the variance of ϵ_{jt} in Eq. (8)).

Unlike other variables in Eq. (10), future values of the two weather variables ($DTEMP_t$ and $DRECIPI_t$) are unknown at the time of forecasting. We therefore use government weather forecasts for Amsterdam as the values for $DTEMP_t$ (=forecasted maximum temperature) and $DRECIPI_t$ (= 1 if probability of precipitation >0.5 and = 0 otherwise).

4.1.4. Data management

For our runs of SilverScheduler in the 14 weeks during March–July 2005, we continuously updated our estimates as new data became available. Specifically, we started with 57 weeks (January 29, 2004–February 27, 2005) of attendance data, giving us 23,148 observations for estimation in the first week. For each Sunday, the dataset was

Table 2a
Estimation results of demand model (8).

Movie program: Mar 3–9, 2005					
R-Square	0.6423		TUE	−0.380	<.0001
Adj R-Sq	0.6361		WED	−0.431	<.0001
N	23148		THU	−0.291	<.0001
Variable	Parameter estimate	Pr> t	FRI	−0.165	<.0001
Intercept	5.071	<.0001	SUN	−0.015	0.4333
10 am	−2.011	<.0001	SATNIGHT	−1.033	<.0001
11 am	−1.712	<.0001	SUNPM	0.512	<.0001
12 pm	−1.885	<.0001	National soccer	−1.123	<.0001
1 pm	−1.334	<.0001	Easter weekend	0.452	<.0001
2 pm	−1.063	<.0001	Ascension day	0.891	<.0001
3 pm	−1.066	<.0001	Whit weekend	0.850	<.0001
4 pm	−1.164	<.0001	Spring vacation	0.589	<.0001
5 pm	−1.063	<.0001	May vacation	0.365	<.0001
6 pm	−0.768	<.0001	Summer vacation	0.142	<.0001
7 pm	−0.349	<.0001	Fall vacation	0.616	<.0001
9 pm	−0.237	<.0001	Xmas vacation	0.723	<.0001
10 pm	−0.491	<.0001	Daily max. temp.	−0.003	<.0001
MON	−0.590	<.0001	Precipitation	0.084	<.0001

Base case: Classics screened at 8 pm on Saturday.

Table 2b
Estimation results of θ_j model (9).

Mar 3–9, 2005		θ_j regression			
R-Square		0.3925			
Adj R-Sq		0.2892			
N		173			
Variable	Parameter estimate	Pr> t	Variable	Parameter estimate	Pr> t
Intercept	–0.686	0.0081	Musical genre if US film	–0.937	0.1979
Dutch movies	–0.216	0.6061	Suspense genre if US film	0.081	0.7697
Other origins	–0.488	0.1693	Comedy genre if US film	0.186	0.4105
Sequel	0.034	0.8912	Romantic comedy genre if US film	0.451	0.1097
Franchise	–0.000	0.7843	Action genre if US film	0.164	0.5828
Dutch speaking	0.371	0.2326	Sci-Fi genre if US film	0.153	0.6131
Other language	0.341	0.2890	Horror genre if US film	–0.128	0.7338
Dutch dubbing	–0.225	0.5805	Fantasy genre if US film	0.416	0.3876
US opening week box office revenue if US film	0.009	0.0165	Western genre if US film	0.645	0.3729
US opening week theater numbers if US film	0.000	0.7609	Animated genre if US film	0.852	0.0263
US box office drop in the 2nd week if US film	–0.630	0.1709	Adventure genre if US film	0.872	0.0472
G-rated if US film	–1.242	0.0069	Documentary genre if US film	0.728	0.3148
PG-rated if US film	–0.772	0.0025			
PG13-rated if US film	–0.164	0.3499			

Base case: English-speaking R-rated U.S. film of drama genre.

updated with a new week of data up to that day, adding around 400 new observations (each week has about 400 movie screenings).⁷

Some particular aspects of our data collection process are worth noting. While there were some free admissions to movies, our estimation only included paid admissions. As only 3% of movies reached 90% of capacity and average filled capacity per showing was about 26%, we do not consider capacity constraints (i.e., sold out showings) in either our estimation or programming model. In addition, we do not consider price effects. Although the De Munt theater charges slightly different prices for showings starting in different time slots (e.g., matinees) and for moviegoers of different ages (e.g., senior discounts), prices are the same across different movie titles. The price variation is absorbed by time of day parameters and there is no need to include any price variables.

4.2. Evaluation of forecasting performance

Table 2a shows the estimation results for our main forecasting model (8) and Table 2b shows the estimation results of Eq. (9).⁸ For brevity, we only report the estimation results in the first week. To identify the model (8), we set “classics” screened at 8 pm on Saturday as the base case.

As shown in Table 2a, the results demonstrate high face validity and strong explanatory power. The R^2 for the first week is 0.64. All but one of the estimated coefficients are statistically significant. This is not surprising given the large sample size. More importantly, the values of the coefficients are consistent with expectations.

- (1) The estimates for β_h increase as we go into the evening (8 pm is the most preferred time to watch a movie).
- (2) The estimates for ω_d on Saturday and Sunday are the largest.
- (3) All estimates for γ_v are significantly positive.
- (4) Sunday afternoons are better than afternoons of other days (but other times on Sundays are equivalent to the base case).
- (5) National soccer games negatively affect movie attendance.

⁷ After completion of test runs, we performed an audit of our data management process and found some minor problems. For example, a small number of observations for some movies were dropped. These effects had a small overall impact and led to less accurate forecasts than would otherwise have occurred. We verified this by running our model for an additional week after discovering these problems and found an increase in the R^2 between forecasts sales and attendance compared to our mean results. In this paper, we use the originally generated forecasts as they were used in our interactions with management.

⁸ Estimation results of λ_j regression model are in Appendix A2.

In addition to the coefficients reported in Table 2a, movie-specific parameter values for θ_j and λ_j were obtained for each movie. For example, based on the data available for the week of March 3, the movie *Passion of the Christ* had values of $\theta_j = 0.224$ and $\lambda_j = -0.053$, indicating a strong opening and a slow decay rate.

For movies in their first or second week of showing, characteristics of the movies were used to estimate values of θ_j and λ_j as shown in Eq. (9). The values for R^2 in Table 2b and Appendix A2 show that, for these new movies, the fit of the models for θ and λ is moderate. Interestingly, the attractiveness of a movie (θ) at the De Munt is significantly related to US opening week box office revenue (see Table 2b) and the weekly decay (λ) (the drop in revenue in the second week at US box office – see Appendix A2). As expected, the need to rely on estimates had an effect on the accuracy of our forecasts. For example, across the 14 weeks, the average error for new movies was –19.7 admissions per showing. For movies with just one week of data, the mean error was 12.8 admissions per showing. For movies with two or more weeks of data, the mean error was –6.4 admission tickets per showing.⁹ Future work should examine methods to obtain better estimates for new movies. Market research techniques, such as those described in Eliashberg, Jonker, Sawhney, and Wierenga (2000) or data from new information sources such as the Hollywood Stock Exchange (Spann & Skiera, 2003), could be usefully employed to obtain such results.

Table 2c compares our forecasts to the actual attendance given the movie schedules actually adopted in our test period.

Across the 14 weeks in our run, the mean absolute error, root mean squared error, and Pearson correlation between predicted attendance and actual attendance averaged 22.09, 38.94 and 0.65 respectively. Moreover, as shown in the average error, it appears that there is not any systematic upward or downward bias. Due to the need to update the forecasts each weekend with a great deal of time pressure (as discussed earlier), we chose to estimate Eqs. (8) and (9) using OLS. Although more sophisticated models are available to conduct such analyses (see, for example, Ainslie et al. 2005), in practice such techniques do not always provide an improvement that is sufficiently important to management. For example, Andrews, Currim, Leeflang, and Lim (2008) show that, with both simulated data and actual data for shampoo sales at the store level in Holland, forecast accuracy was not improved for the SCAN*PRO model using either hierarchical Bayes or finite mixture techniques. For our data, we also explored the

⁹ The Mean Absolute Errors were also calculated and were 37.2, 22.6, and 19.0 respectively for new movies, movies with one week of data, and movies with 2 or more weeks of data.

Table 2c
Predictive performance of the forecasting module.

	Mean absolute errors	Root mean squared errors	Pearson correlation between predicted and actual	Average errors
Mar 3–Mar 9, 05	28.29	44.33	0.58	8.91
Mar 10–Mar 16, 05	16.83	24.68	0.84	– 3.13
Mar 17–Mar 23, 05	18.06	28.54	0.76	3.69
Mar 24–Mar 30, 05	27.71	47.20	0.74	9.36
Apr 7–Apr 13, 05	16.95	24.18	0.73	1.30
Apr 14–Apr 20, 05	12.67	23.36	0.81	– 3.43
Apr 21–Apr 27, 05	14.34	23.22	0.78	6.73
Apr 28–May 4, 05	17.52	27.10	0.43	1.26
May 5–May 11, 05	27.65	58.30	0.59	– 24.50
May 19–May 25, 05	29.84	70.59	0.49	– 21.56
May 26–Jun 1, 05	18.20	40.21	0.55	– 11.95
Jun 2–Jun 8, 05	22.55	42.50	0.71	– 15.81
Jun 30–Jul 6, 05	34.36	55.02	0.56	– 23.33
Jul 14–Jul 20, 05	24.24	35.92	0.57	– 2.70

benefits of capturing more complexity in the data by using a weighted least squares (WLS) method for one week of forecasts and found no improvement in our estimation of Eq. (8), the attendance equation in Step 1 (OLS R^2 : 0.64 versus WLS R^2 : 0.61), and only a small improvement in R^2 for our estimations of θ_j (Eq. (9)) (from 0.39 to 0.48) and λ_j (from 0.31 to 0.34). Moreover, we found no improvement in the predicted attendance for the week's schedule of movies in terms of mean absolute deviation, root mean squared error, or Pearson correlation. We agree with Andrews et al. (2008) that further exploration of situations when more sophisticated techniques are appropriate in managerial settings is needed.

5. Applying SilverScheduler to a multiplex in Amsterdam

5.1. Empirical setting

The complete *SilverScheduler* procedure (scheduling algorithm plus conditional forecasting method) was tested and evaluated in the De Munt theater in downtown Amsterdam. The goal was to demonstrate to Pathé's management the efficiency of *SilverScheduler* and the difference between *SilverScheduler* and manually generated schedules for 14 weeks, employing data from the previous year for estimating initial values of the model parameters.

For each of the 14 weeks we received the following information from Pathé:

- The movies to be scheduled in that particular week (i.e., the list of movies) with their running times. The average number of movies to be scheduled was 18, ranging from 15 to 22.
- Contractual agreements with distributors that certain movies will be shown in certain screening rooms. Typically, these agreements are made for newly released movies (1 to 3 per week). These agreements included the specification of a particular screening room or a screening room that meets some capacity requirement.
- Opening and closing times of the theater for weekdays and weekend days and information about the times required for cleaning screening rooms between shows.

For demand input, we used the estimates produced by the demand forecasting procedure described earlier. This means that we have a forecast for the number of visitors for every movie for each possible starting time/day combination in the week for which the schedule is produced. Here we worked with a grid of starting times that were 1 h apart.

5.1.1. From day schedules to a week schedule

Employing the procedures described above, we used *SilverScheduler* to generate schedules for each of the 14 weeks in our dataset. As made clear from the description in Section 3, the scheduling algorithm, in principle, makes a schedule for a day (it solves the

MSP). However, De Munt does not have the same movie schedule on each of the seven days of the week. First, on both weekend days, the theater opens at 10 am (compared to noon on weekdays) and on the weekend, some children movies shown (until 6 pm). Furthermore, Saturday is different from Sunday because Saturday is the only day that the theater is open until 1.30 am. On all other days, it closes at midnight. Finally, Wednesday is different from the other weekdays because children's movies are shown on Wednesday afternoons. To account for these differences, we first applied *SilverScheduler* to generate a schedule for the four days (Thursday, Friday, Monday, and Tuesday) that have the same schedule. Subsequently, this "base schedule" served as the starting point for schedules for the other three days. Here, the movies remain in the same screening room as in the base schedule, but the times are adapted according to the requirements of the specific day. Furthermore, on Saturday, Sunday, and Wednesday, children's movies were inserted into the day, replacing movies in the base schedule that had the lowest number of visitors.

5.2. Results for one day

To illustrate the *SilverScheduler* approach to movie scheduling and to compare it to the corresponding, manually constructed Pathé schedule, we focus on the first day (Thursday, March 3) in our dataset. For a list of movies and name abbreviations, see Appendix B. There are 26 movies in total for this week, including 2 copies of *Constantine*. Sometimes management wants to have a movie double booked (or even triple booked). To get the most out of high potential movies, the starting times of the same movie in different screening rooms is set at least 1 h apart.

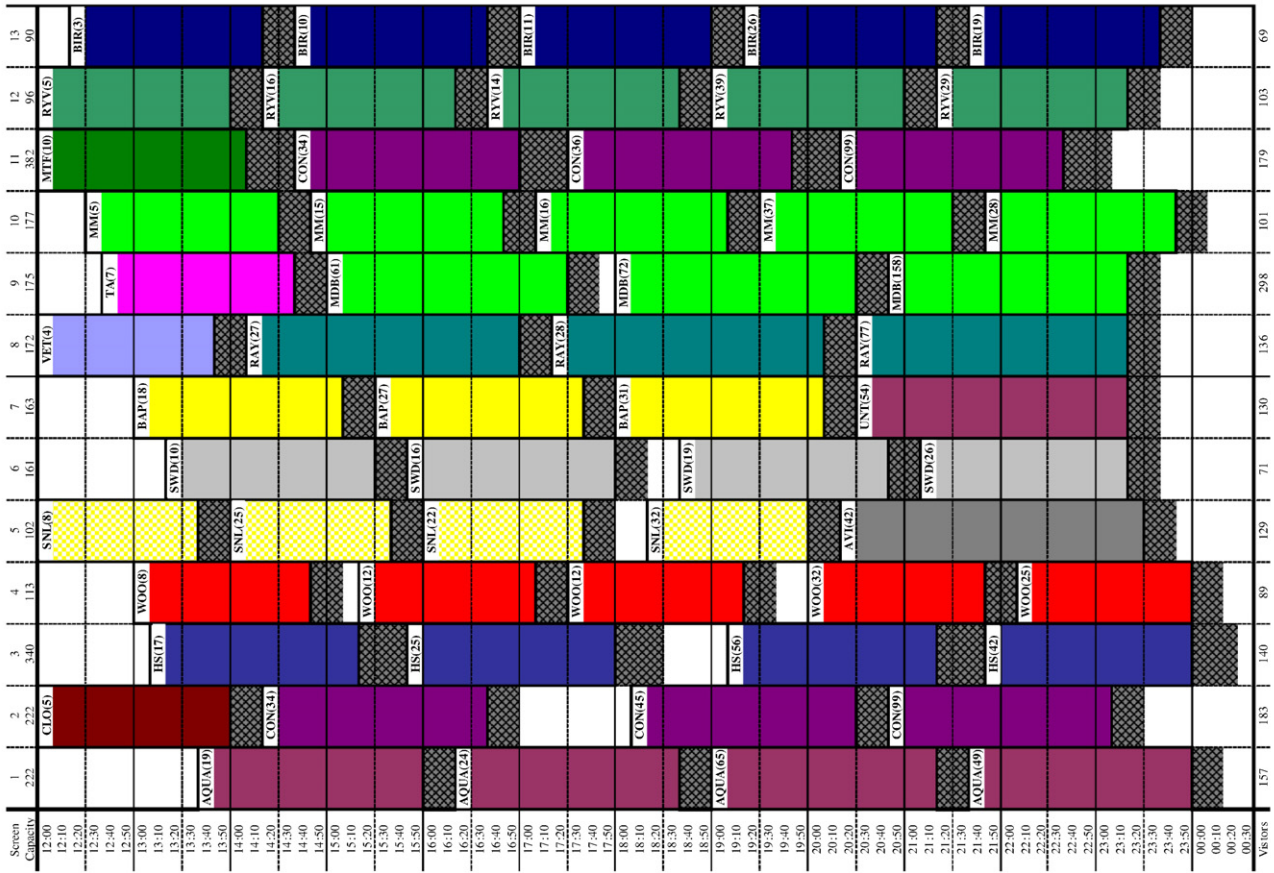
The last 3 movies listed are for dedicated showings on specific screens. There are 4 children's movies (not shown on Thursdays, Fridays, Mondays or Tuesdays), so there are 19 movies that remain to be scheduled.

Fig. 3 shows:

- (1) the actual schedule as produced by Pathé management (left), and
- (2) the schedule generated by *SilverScheduler* (right).

The numbers behind the movie names are the forecasted numbers of visitors for the particular show. For this particular week, there was only one contractual agreement with a distributor, requiring that the movie *Hide and Seek* (HS) should be shown in one of the two largest screening rooms, i.e., either in Screening Room 3 (340 seats) or in Screening Room 11 (382 seats). Fig. 3 shows that Pathé's HS is in Screening Room 11, whereas *SilverScheduler* recommends assigning it to Screening Room 3. Fig. 3 also clearly shows the differences in run-times between the different movies.

Schedule constructed with SilverScheduler



Manual Schedule

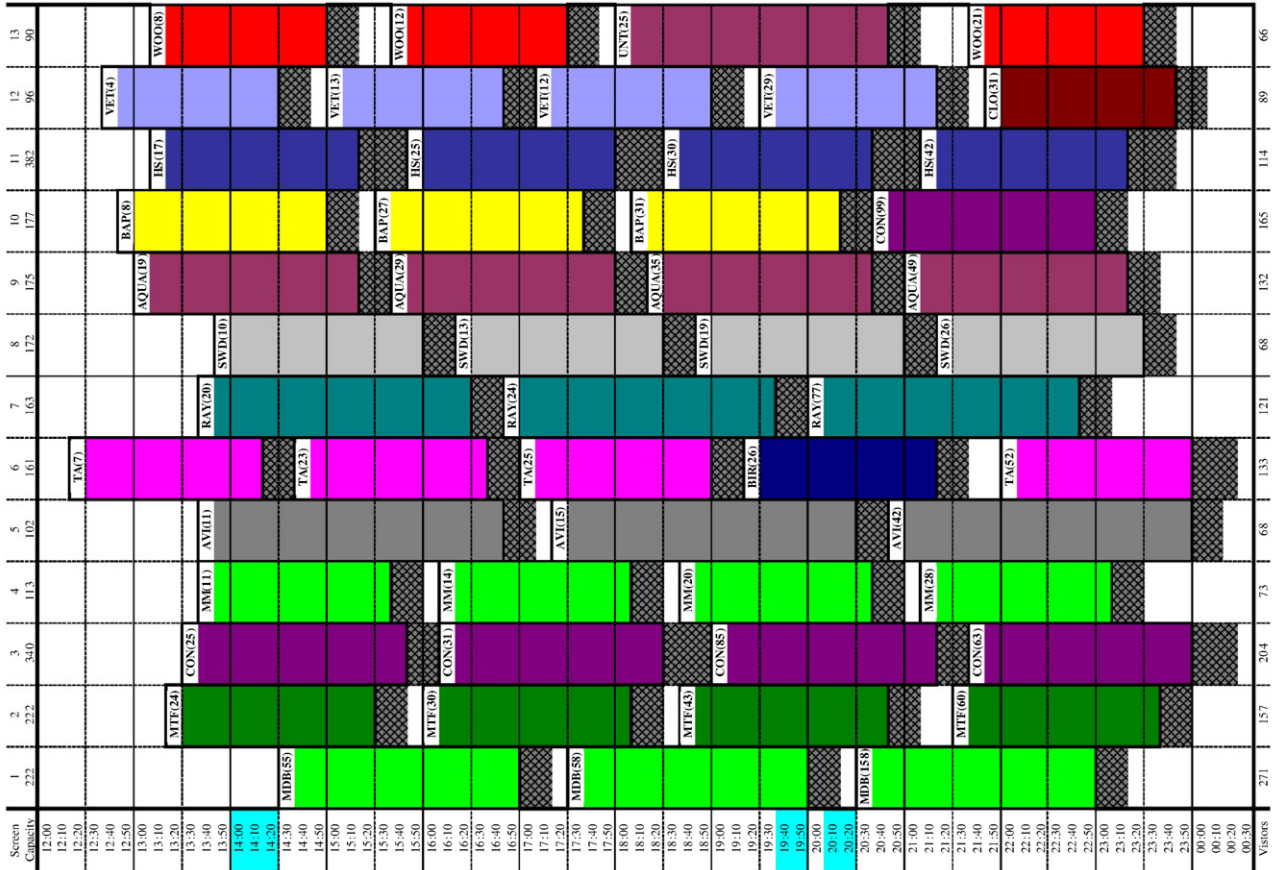


Fig. 3. Movie schedules for Thursday March 3, 2005.

For example, for *Der Untergang* (UNT), the run-time is 165 min, and for *Vet Hard* (VET), the run-time is 105 min. Run-times include advertising and trailers.

After each movie showing, there is a cleaning time before a new movie starts. The cleaning time is 20 min for a small room and 30 min for a large room.

5.2.1. Evaluation of the SilverScheduler recommended schedule

The number of visitors for Thursday March 3, 2005, is estimated to be 1785 in the SilverScheduler schedule compared to 1661 in the manual schedule. For this day only, SilverScheduler generates 124 extra visitors. SilverScheduler manages to schedule more shows on this particular day: 57, compared to 51 shows in the manually constructed schedule. In this case, notwithstanding the extensive set of constraints, SilverScheduler is able to accommodate more shows. The SilverScheduler solution complies with the requirement that a new movie starts at least every 20 min. It is quite difficult to reach this goal when the schedule is made by hand. In the schedule for March 3 (Fig. 3, left) this requirement is violated several times in the manually made schedule. For example, there are no movies that start between 1:50 pm and 2:30 pm, between 7:30 pm and 8 pm, and between 8 pm and 8:30 pm. The other managerial requirements are also violated repeatedly in the manual schedule. For example, on the Saturday of the first week of our dataset (March 5, 2005; not shown here), at several times the manual schedule allowed more than one movie to start on the same floor at the same time. This will produce undesirable (and avoidable) crowding. Additionally, the actual schedule had more than two different movies in the same screening room on the same day, which violates the constraint of no more than two different movies on one screen (i.e., too much screen switching).

5.3. Results for the 14 weeks

We evaluate our overall results in several ways. First, we look at the scheduling efficiency, particularly the number of movies that are shown each week. Second, we examine implications for attendance (ticket sales). Table 3 shows the number of movies (“movie showings”) that would be shown in SilverScheduler schedules compared to Pathé schedules. Although SilverScheduler has to consider a large number of constraints, it still manages to schedule slightly more movies than the programmers at Pathé do (+0.51%). Of course, the main contribution of SilverScheduler is that it schedules “better” movies, which should result in higher ticket sales. When examining this, there is the basic problem that we will never know how many visitors a particular show (showing a specific movie in a particular

week on a particular day at a particular time) recommended by SilverScheduler would have generated if this show did not actually take place. We have to approach this issue in an indirect way, and we follow three different routes. First, we make a comparison within the prediction regime that generated the input for the optimization, as in Table 3. If we compare the Pathé schedules to schedules generated by SilverScheduler, and use, in both cases, Eq. (8) to predict the numbers of visitors, we see that SilverScheduler shows an increase of 10.83%. This improvement reflects the contribution of the scheduling algorithm, given the demand (i.e., the predicted numbers of visitors). In other words, this is the contribution of SilverScheduler with perfect forecasts. However, actual visitor numbers tend to deviate from forecasted numbers. As we saw earlier, the correlation between actual and forecasted numbers is on average 0.65. Of course, with imperfect forecasts, the improvement obtained through SilverScheduler will be less.

Second, we carried out a comparison of the schedules of Pathé and SilverScheduler based on the actual demand manifested from observed visitor data (i.e., ex post). For this purpose, using the actual visitor data for the week, a regression model was estimated to explain the number of visitors per show by the movie, day of the week, and hour of the day. This model was then used to predict the number of visitors per show for the Pathé schedule, as well as the SilverScheduler schedule. Using this approach, the estimated improvement in number of visitors was 2.53% over the 14 weeks. While lower than the level reported in Table 3, this is still a considerable increase. It indicates there would be about 20,000 additional visitors per year, or about € 150,000 in extra revenue (US \$220,000). Third, we can look at the actual number of visitors that the Pathé schedules generated (first column of Table 3). In this data, the week of June 30–July 6 (with over 24,000 visitors) is evidently an outlier, caused by the enormous success of a one-time promotional action where the biggest supermarket in Holland gave their customers free “second tickets” for Pathé (This promotional action was not included in our prediction model.). If we leave this particular week out and compare the totals of actual ticket sales to predicted ticket sales for the SilverScheduler schedule, we see that the total for SilverScheduler is 3.1% higher (199,884 versus 193,774).

In addition to generating more visitors, SilverScheduler also provides direct operational efficiency by automating the preparation of the weekly movie schedule. SilverScheduler saves a considerable amount of managerial time and effort. The current process is cumbersome, often generating a fair amount of managerial frustration. Furthermore, all the expressed management constraints were met. Taking these managerial constraints into account (every 20 min a new movie, less crowding) should over time have a positive effect on

Table 3
Comparison of the schedules of Pathé and SilverScheduler over all 14 weeks.

Week in 2005	Actual ticket sales	Ticket sales predicted by Eqs. (8) and (9)				
		Pathé schedule		SilverScheduler		
		Movie showings	Ticket sales	Movie showings	Ticket sales	Improvement (%)
Mar 03–Mar 09	16,757	394	18,316	423	22,930	25.2
Mar 10–Mar 16	16,167	396	13,903	397	16,379	17.8
Mar 17–Mar 23	14,367	415	15,229	418	16,069	5.5
Mar 24–Mar 30	18,221	424	22,292	414	24,053	7.9
Apr 07–Apr 13	13,258	410	14,378	418	15,531	8.0
Apr 14–Apr 20	13,249	414	12,166	432	13,240	8.8
Apr 21–Apr 27	10,017	429	15,370	434	16,614	8.1
Apr 28–May 04	10,866	416	13,675	404	14,138	3.4
May 05–May 11	17,729	439	9977	412	10,502	5.3
May 19–May 25	16,776	412	9257	425	10,887	17.6
May 26–Jun 01	13,495	406	9675	385	10,099	4.4
Jun 02–Jun 08	16,528	418	11,305	407	11,639	3.0
Jun 30–Jul 06	24,106 ^a	409	13,800	419	15,654	13.4
Jul 14–Jul 20	16,348	419	15,086	443	17,763	17.7
Total	217,884	5801	194,429	5831	215,498	10.8

^a Week with sales promotion (not in model).

the numbers of visitors, which is not considered in the current computations of additional revenue.

6. Conclusions and further development

We have developed a model, *SilverScheduler*, which schedules movies over the days of the week and the times of the day. The model's algorithm, which follows the column generation approach, is able to produce solutions in a reasonable amount of time (on average 2.5 min) with very good performance (on average within 1.57% of the optimum). A forecasting module was also developed where the numbers of visitors are forecasted using a model estimated on data from previous weeks.

We generated and evaluated the scheduling for the De Munt theater in Amsterdam for 14 weeks in 2005. Comparing *SilverScheduler* results with manual schedules, *SilverScheduler* generated better results. Within the constraints set by logistical and managerial considerations, on average, *SilverScheduler* scheduled about the same numbers of shows as management. *SilverScheduler*, however, scheduled more appealing movies, and also took better account of the managerial requirements compared to the manual solution. Under the same forecasting procedure, *SilverScheduler* generated nearly 11% more visitors than the manual schedules. When we compare based on actual demand in the given week, the improvement in visitors through *SilverScheduler* is 2.5%. The latter still amounts to Euro 187,000 in extra revenue for the De Munt, theater on an annual basis.

Additional research should help further improve *SilverScheduler*. The first priority is to increase the accuracy of its forecasts. As we have seen, better forecasts significantly improve the performance of *SilverScheduler*. In particular, there is room for improvement in the attendance forecasts for new movies. One possibility is to find a way to include the intuitive judgment of the management of the theater. These managers have usually seen a pre-screening of a new movie, which gives them an idea of its potential for a particular theater. A Bayesian approach could be explored, where this intuitive managerial knowledge is combined with attendance data as soon as such data become available. Use of input from management will also increase the external/managerial validity of the model and therefore its probability of acceptance (Laurent, 2000). Another possibility is to combine the currently used box office data with market research data, at least for some movies. Furthermore, new methodologies have become available for the early prediction of the success of new movies (e.g., Eliashberg et al., 2000). Another area to examine is the impact of movie scheduling on concession sales. Presently, Pathé management estimates a constant amount of concession sales per visitor. However, concession sales may vary by the times when movies are shown, by the duration of the movie, and sales may be dependent on the amount of queuing that occurs. An extension in this regard would pose interesting technical challenges, but could lead to substantial improvements in profit given the high margins on concession sales. So far, we have assumed personnel costs to be constant, but if *SilverScheduler* is able to attract more visitors, more personnel may be needed for selling and checking the tickets. In this respect, *SilverScheduler* can be refined. In addition, *SilverScheduler* has to be developed further in the direction of a decision support tool, so that it can be easily used by the theater manager. This includes a user-friendly, intuitive interface that is easy to navigate and operate.

Pathé Nederland, the owner of the De Munt theater, considers the demonstration of *SilverScheduler* as an interesting case study worth further consideration for implementation. In that case, they will not just use *SilverScheduler* for De Munt, but for movie scheduling for all of their 12 movie theaters in the Netherlands. Pathé Nederland already has a planning and ticketing software system in place, and *SilverScheduler* can potentially be used to deliver its schedules as the inputs to this system.

While we describe our development of the *SilverScheduler* algorithm in one theater, the problem is widespread. There are 10,000 theaters in Europe and 7000 in the US, all faced with the same problem. Movie theater programming is a time-consuming activity, and as our example illustrates, the required managerial constraints are not always met. The possibility to delegate at least part of this task to a decision support system is clearly an important step forward. Theaters are increasingly moving from manual ticketing to computer based ticketing systems. This should facilitate the adoption of computer based decision support systems. Probably even more important to the future impact of systems such as *SilverScheduler* in the movie industry is the expected growth of digital cinema, which accounted for 6455 screens worldwide (MPAA) in 2008. When this occurs, movies will not be delivered to theaters on huge reels, but on simple diskettes or other electronic media. This will make theaters much more flexible in their scheduling. In this situation, the scheduling possibilities will multiply, making it even more important to have an algorithm like *SilverScheduler* as part of a decision support system to help management find the best schedule for its theaters.

Appendix A

A.1. Definition of movie attributes

$O-USA_j \equiv$	indicator variable for the event that movie j is made in the U.S.
$O-DUTCH_j \equiv$	indicator variable for the event that movie j is made in the Netherlands
$O-OTHER_j \equiv$	indicator variable for the event that movie j is made in other countries
$L-ENGLISH_j \equiv$	indicator variable for the event that movie j is in English
$L-DUTCH_j \equiv$	indicator variable for the event that movie j is in Dutch
$L-OTHER_j \equiv$	indicator variable for the event that movie j is in other languages
$D-DUTCH_j \equiv$	indicator variable for the event that movie j is dubbed Dutch
$SEQUEL_j \equiv$	indicator variable for the event that movie j is a sequel
$FRANCHISE_j \equiv$	average worldwide box office gross of all preceding movies in the franchise of movie j , if movie j is a sequel.
$BOW1_j \equiv$	opening week's box office sales in the U.S., if movie j is made in U.S.
$TNW1_j \equiv$	opening week's theater numbers in the U.S., if movie j is made in U.S.
$BOW1W2_j \equiv$	box office sales percentage change from opening week to the second week in the U.S., if movie j is made in U.S.
$MPAA-G_j \equiv$	indicator variable for the event that movie j is rated G by MPAA and made in U.S.
$MPAA-PG_j \equiv$	indicator variable for the event that movie j is rated PG and made in U.S.
$MPAA-PG13_j \equiv$	indicator variable for the event that movie j is rated PG13 and made in U.S.
$MPAA-R_j \equiv$	indicator variable for the event that movie j is rated R and made in U.S.
$DRAMA_j \equiv$	indicator variable for the event that movie j is classified by Variety.com to be in the drama genre and made in U.S.
$ACTION_j \equiv$	indicator variable for the event that movie j is in the action genre and made in U.S.
$COMEDY_j \equiv$	indicator variable for the event that movie j is in the comedy genre and made in U.S.
$RCOMEDY_j \equiv$	indicator variable for the event that movie j is in the romantic comedy genre and made in U.S.

- $MUSICAL_j \equiv$ indicator variable for the event that movie j is in the musical genre and made in U.S.
- $SUSPENSE_j \equiv$ indicator variable for the event that movie j is in the suspense genre and made in U.S.
- $SCIFI_j \equiv$ indicator variable for the event that movie j is in the sci-fi genre and made in U.S.
- $HORROR_j \equiv$ indicator variable for the event that movie j is in the horror genre and made in U.S.
- $FANTASY_j \equiv$ indicator variable for the event that movie j is in the fantasy genre and made in U.S.
- $WESTERN_j \equiv$ indicator variable for the event that movie j is in the western genre and made in U.S.
- $ANIMATED_j \equiv$ indicator variable for the event that movie j is in the animated genre and is made in U.S.
- $ADVENTURE_j \equiv$ indicator variable for the event that movie j is in the adventure genre and made in U.S.
- $DOCU_j \equiv$ indicator variable for the event that movie j is in the documentary genre and made in U.S.

A.2. Estimation results for the λ_j regression model

Mar 3–9, 2005		λ_j regression			
R-Square	0.3051				
Adj R-Sq	0.1725				
N	157				
Variable	Parameter estimate	Pr> t	Variable	Parameter estimate	Pr> t
Intercept	0.209	0.1458	Musical genre	0.545	0.0566
Dutch movies	-0.409	0.0817	if US film		
Other origins	-0.392	0.0617	Suspense genre	-0.120	0.2756
Sequel	0.001	0.9893	if US film		
Franchise	-0.000	0.4803	Comedy genre	-0.194	0.0360
Dutch speaking	0.065	0.7112	if US film		
Other language	-0.003	0.9883	Romantic comedy genre if US film	-0.133	0.2519
Dutch dubbing	-0.038	0.8492			
US opening week box office revenue if US film	-0.000	0.9414	Action genre if US film	-0.034	0.7788
			Sci-Fi genre if US film	-0.062	0.6118
US opening week theater numbers if US film	0.000	0.3850	Horror genre if US film	0.035	0.8123
US box office drop in the 2nd week if US film	0.855	0.0009	Fantasy genre if US film	0.138	0.4624
			Western genre if US film	-0.413	0.1400
G-rated if US film	0.221	0.2146	US film		
PG-rated if US film	0.243	0.0136	Animated genre if US film	-0.620	<.0001
PG13-rated if US film	-0.133	0.0563	Adventure genre if US film	-0.477	0.0058
			Documentary genre if US film	-0.167	0.5561

Base case: Englishspeaking R-rated U.S. film of drama genre.

Appendix B. List of movies for the movie week of March 3–9 2005

No	Full name	Abbreviation	Kids?	Duration ^a
1	Million Dollar Baby	MDB	No	147
2	Meet The Fockers	MTF	No	130
3	Constantine	CO1	No	136
4	Constantine	CO2	No	136
5	Melinda And Melinda	MM	No	114
6	The Aviator	AVI	No	185
7	Birth	BI	No	115
8	Team America	TA	No	113
9	Ray	RAY	No	167

Appendix B (continued)

No	Full name	Abbreviation	Kids?	Duration ^a
10	Raise Your Voice	RYV	No	118
11	Shall We Dance	SWD	No	121
12	Spongebob	SNL	No	99
13	The Life Aquatic	AQ	No	133
14	Bride And Prejudice	BAP	No	126
15	Hide & Seek	HS	No	122
16	Vet Hard	VH	No	105
17	Closer	CLO	No	119
18	Der Untergang	UNT	No	165
19	The Woodsman	WOO	No	103
20	Lepele	LPL	Yes	104
21	Plop & Kwispel	PK	Yes	71
22	Incredibles	INC	Yes	130
23	Streep Wil Racen	STR	Yes	104
24	Passion Of The Christ ^b	PAS	No	126
25	Goodbye Lenin ^b	GL	No	136
26	Hitch ^c	HIT	No	133

^a In minutes. Advertising and trailers included.

^b One show, Sunday morning.

^c One show, Saturday night.

Appendix C. Technical details of the proposed algorithm

Recall that the MSP can be formulated as follows:

$$(MSP) \quad \min \sum_{s \in S} \sum_{p \in P^s} c_p x_p^s + R \sum_{l \in L} y_l \tag{A1}$$

$$\text{s.t.} \quad \sum_{s \in S} \sum_{p \in P^s} a_{mp}^s x_p^s = 1 \quad \forall m \in M, \tag{A2}$$

$$\sum_{p \in P^s} x_p^s = 1 \quad \forall s \in S, \tag{A3}$$

$$\sum_{s \in S} \sum_{p \in P^s} b_{ip}^s e_{fp}^s x_p^s \leq 1 \quad \forall f \in F, i \in T_1, \tag{A4}$$

$$\sum_{s \in S} \sum_{p \in P^s} (b_{ip}^s + \dots + b_{jp}^s) x_p^s + y_l \geq 1 \quad \forall l \in L, i, j \in T_l, \tag{A5}$$

$$\sum_{s \in S} \sum_{p \in P^s} h_p^s x_p^s \geq r, \tag{A6}$$

$$x_p^s \in \{0, 1\} \quad \forall s \in S, p \in P^s. \tag{A7}$$

In the remainder of this appendix, we give a detailed description of the different steps in the algorithm presented in Fig. 2.

To initialize the procedure, we first construct an initial feasible solution in Step 0. A trivial feasible solution can be obtained by showing each movie once and setting all y -variables equal to 1. Because there are often more movies than screens, on some screens, two movies will have to be shown. These screens are chosen arbitrarily. The initial feasible solution produces an initial set of columns. Note that the procedure is not very sensitive to the choice of initial columns.

Master problem

To calculate a lower bound on the (restricted) master problem, we use a relaxation of the MSP by replacing the “=” signs by “≥” signs and subsequently relaxing the constraints in a Lagrangian way (Step 1). That is, we associate non-negative Lagrangian multipliers $\lambda_m, \mu_s,$

v_{fi} , σ_{ijl} and ω to Constraints (A2)–(A6), respectively. The remaining Lagrangian subproblem can then be rewritten as:

$$\min \sum_{s \in S} \sum_{p \in P^s} \bar{c}_p^s x_p^s + \sum_{l \in L} (R - \sigma_{ijl}) y_l + \sum_{m \in M} \lambda_m + \sum_{s \in S} \mu_s \quad (\text{A8})$$

$$- \sum_{f \in F} \sum_{i \in T_1} v_{fi} + \sum_{l \in L} \sum_{i \in T_1} \sum_{j \in T_1} \sigma_{ijl} + r\omega$$

$$\text{s.t. } x_p^s \in \{0, 1\} \quad \forall s \in S, p \in P^s, \quad (\text{A9})$$

$$y_l \in \{0, 1\} \quad \forall l \in L, \quad (\text{A10})$$

where $\bar{c}_p^s = c_p - \sum_{m \in M} \lambda_m \alpha_{mp}^s - \mu_s + \sum_{f \in F} \sum_{i \in T_1} v_{fi} b_{ip}^s e_f^s - \sum_{l \in L} \sum_{i \in T_1} \sum_{j \in T_1} \sigma_{ijl} (b_{ip}^s + \dots + b_{jp}^s) - \omega$. The Lagrangian subproblem can be solved by inspection, i.e., $x_p^s = 1$ if $\bar{c}_p^s < 0$ and 0 otherwise, and $y_l = 1$ if $R < \sigma_{ijl}$ and 0 otherwise. In this way, we obtain a lower bound for the given set of columns. We apply subgradient optimization to find the best lower bound. For the reader interested in this topic, see, for example, the survey in Beasley (1995). We will not go into further detail here.

Pricing problem

In Step 2, we generate new columns with negative reduced costs, where the reduced costs of a column corresponding to path p on screen s are denoted by \bar{c}_p^s . For each screen, we generate new paths by solving an all-pair shortest path problem (e.g., Freling, 1997) between each pair of nodes (excluding the source and sink node). From all these possible paths, we add the k -smallest ones to the restricted master problem. In our current implementation, we choose for k the value 100. However, tests in Hegie (2004) indicated that values between 75 and 150 results in essentially the same performance. By solving an all-pair shortest path problem, we get a large variety in columns, making it possible that only a few iterations in the column generation procedure are necessary. Furthermore, we calculate a lower bound on the overall problem by adding the reduced costs of all generated columns by the lower bound obtained in Step 1. If this difference is small enough, we stop the column generation procedure and construct a feasible solution. Otherwise, we return to Step 1, and we add all columns generated in Step 2 to the restricted master problem. Note that in the procedure, we never delete columns.

Feasible solution

Finally, we construct a feasible solution for the movie scheduling problem. This is only computed at the end of the algorithm by solving the problem (MSP) with the initial columns and the columns generated during the column generation to optimality. For this purpose, we use the commercial MIP solver Cplex 7.1. Note that this solution is optimal given the set of columns, but it is generally not an overall optimal solution because there are no columns generated during the branching.

References

- Ainslie, A., Dreze, X., & Zufryden, F. (2005). Modeling movie lifecycles and market share. *Marketing Science*, 24(3), 508–517.
- Andrews, R. L., Currim, I. S., Leeflang, P., & Lim, J. (2008). Estimating the Scan*Pro model of store sales: HB, FM or just OLS? *International Journal of Research in Marketing*, 25, 22–33.

- Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P., & Vance, P. H. (1998). Branch-and-Price: Column generation for solving huge integer programs. *Operations Research*, 46, 316–329.
- Beasley, J. E. (1995). Lagrangean relaxation. In C. R. Reeves (Ed.), *Modern heuristic techniques for combinatorial problems* (pp. 243–303). London: McGraw-Hill.
- Danaher, P. J., & Mawhinney, D. (2001). Optimizing television program schedules using choice modeling. *Journal of Marketing Research*, 38(3), 298–312.
- Dantzig, G. B., & Wolfe, P. (1960). Decomposition principles for linear programming. *Operations Research*, 8, 01–11.
- Desaulniers, G., Desrosiers, J., & Solomon, M. M. (Eds.). (2005). *Column generation*. New York: Springer.
- Eliashberg, J., Jonker, J. J., Sawhney, M. S., & Wierenga, B. (2000). MOVIEMOD: An implementable decision support system for prerelease market evaluation of motion pictures. *Marketing Science*, 19(3), 226–243.
- Eliashberg, J., & Lilien, G. L. (1993). Mathematical marketing models: Some historical perspectives and future projections. In J. Eliashberg & G.L. Lilien (Eds.), *Handbooks in operations research and management science/Marketing*, vol. 5 (pp. 3–23). Amsterdam: North Holland.
- Eliashberg, J., Swami, S., Weinberg, C. B., & Wierenga, B. (2001). Implementing and evaluating SilverScreener: A marketing management support system for movie exhibitors. *Interfaces*, 31, S108–S127 (May–June).
- Fisher, M. L. (1981). The Lagrangean relaxation method for solving integer programming problems. *Management Science*, 27(1), 1–18.
- Freling, R. (1997). Models and techniques for integrating vehicle and crew scheduling. Ph.D. thesis, Tinbergen Institute, Erasmus University Rotterdam.
- Hanssens, D. M., Parsons, L. J., & Schulz, R. L. (2003). *Market response models: Econometric and time series analysis (sec. Ed.)*. Boston: Kluwer.
- Hegie, Q. (2004). Movie scheduling using column generation and Lagrangian relaxation, Thesis, Erasmus University Rotterdam.
- Ho, J. (2005). Marketing models of entertainment products. Unpublished doctoral thesis, University of British Columbia.
- Horen, Jeffrey H. (1980). Scheduling of network television programs. *Management Science*, 26, 354–370 (April).
- Huisman, D., Freling, R., & Wagelmans, A. P. M. (2005). Multiple-depot integrated vehicle and crew scheduling. *Transportation Science*, 39, 491–502.
- Huisman, D., Jans, R., Peeters, M., & Wagelmans, A. P. M. (2005). Combining column generation and Lagrangian relaxation. In G. Desaulniers, G. Desrosiers, & M. M. Solomon (Eds.), *Column generation* (pp. 247–270). New York: Springer.
- Krider, R. E., & Weinberg, C. B. (1998). Competitive dynamics and the introduction of new products: The motion picture timing game. *Journal of Marketing Research*, 35(1), 1–15.
- Laurent, G. (2000). Improving the external validity of marketing models: A plea for more qualitative input. *International Journal of Research in Marketing*, 17(no 2–3), 177–182.
- Leeflang, P. S. H., & Wittink, D. R. (2000). Building models for marketing decisions: Past, present and future. *International Journal of Research in Marketing*, 12(no 2–3), 105–126.
- Lehmann, D. R., & Weinberg, C. B. (2000). Sales through sequential distribution channels: An application to movies and videos. *Journal of Marketing*, 64(3), 18–33.
- Lübbecke, M. E., & Desrosiers, J. (2005). Selected topics in column generation. *Operations Research*, 53, 1007–1023.
- Reddy, S. K., Aronson, J. E., & Stam, A. (1998). SPOT: Scheduling programs optimally for television. *Management Science*, 44(1), 83–102.
- Roberts, J. H. (2000). The intersection of modeling potential and practice. *International Journal of Research in Marketing*, 17(2–3), 127–134.
- Sandhu, R., & Klabjan, D. (2007). Integrated airline fleet and crew-pairing decisions. *Operations Research*, 55, 439–456.
- Sawhney, M. S., & Eliashberg, J. (1996). A parsimonious model for forecasting gross box-office revenues of motion pictures. *Marketing Science*, 15(2), 113–131.
- Spann, M., & Skiera, B. (2003). Internet-based virtual stock markets for business forecasting. *Management Science*, 49, 1310–1326.
- Sultan, F., Farley, J., & Lehmann, D. R. (1990). A meta-analysis of applications of diffusion models. *Journal of Marketing Research*, 27(1), 70–77.
- Swami, S., Eliashberg, J., & Weinberg, C. B. (1999). SilverScreener: A modeling approach to movie screens management. *Marketing Science*, 18(3), 352–372.
- Wierenga, B., & Oude Ophuis, P. A. M. (1997). Marketing decision support systems: Adoption, use and satisfaction. *International Journal of Research in Marketing*, 14(3), 275–290.