Proceedings of the 2012 Winter Simulation Conference
C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher, eds.

APPOINTMENT SCHEDULING USING OPTIMISATION VIA SIMULATION

Paulien Koeleman
Ger Koole

Department of Mathematics
VU University Amsterdam
De Boelelaan 1081
1081 HV Amsterdam, THE NETHERLANDS

ABSTRACT

In this study we consider the optimal scheduling of a certain number of appointments in a given number of time slots. Given a set of appointment slots, we assume that customers can arrive early or late according to a known distribution around the scheduled arrival time. Analytical methods exist for this problem when all customers are assumed to be punctual, but evaluating methods when this assumption is relieved do not yet exist. The reason why this is difficult, is that the order of service is no longer fixed when possible arrival times of two consecutive customers overlap. Therefore we use simulation to evaluate schedules, and optimisation via simulation techniques to optimize schedules. We develop and compare several strategies, among which random local search and nested partitions. Numerical experiments show that significant improvements can be achieved compared to standard scheduling practice.

1 INTRODUCTION

Appointment scheduling is an area of health care operations management that has received considerable attention in the scientific literature. See, for example, Cayirli and Veral (2003) and Gupta and Denton (2008) for overviews of the literature. The objective is to determine appointment times for a given number of patients such that certain, often conflicting, objectives are each satisfied to a certain extent. These objectives usually include patient waiting times and doctor idle time and lateness, which is the time the doctor is still working after the end of the scheduled time. The canonical formulation is mathematically attractive and challenging, explaining the attention it received. It is also practically relevant, although its use in practice is still limited.

Part of the reason for the lack of applications is that most of these studies use assumptions that are unrealistic from a practical point of view. One notable assumption is the punctuality of patients, where only in some cases no-shows are allowed but never late (or early) arrivals. Most papers that obtain numerical or structural results assume punctuality, with Jouini and Benjaafar (2012) as an exception. The reason is that, when patients can arrive late, overtaking can occur, which makes the problem technically more complicated. Another feature that is hard to deal with analytically is the inclusion of unplanned emergency arrivals, which usually have some form of priority over the scheduled arrivals. No analytical technique exists that can handle all possible features.

It is the objective of this paper to present a technique that requires no unrealistic assumptions and that allows to find (nearly) optimal appointment schedules. In our opinion the only candidate method is optimization via simulation. In this paper we show the use of this method for appointment scheduling. In the next section we define the problem in detail. In Section 3 we explain the optimization methods we used, to be followed by our numerical results in Section 4. We finish the paper by our conclusions and directions for future work.

978-1-4673-4781-5/12/$31.00 ©2012 IEEE

2 PROBLEM DESCRIPTION

The basic problem consists of finding an optimal way to schedule N patients into T slots of d time units each. Think of this as finding an optimal schedule for a clinical session from 9 to 12am, divided into 5-minute intervals, in which 12 patients need to be planned. Then $N = 12$, $d = 5$, and $T = 36$. A schedule is denoted with $x \in \mathbb{N}_0^{\mathrm{T}}$, $x \geq 0$, $x^{\mathrm{T}}e = N$ with e the unit vector and where superscript T means transposed. Thus x_i denotes the number of appointments scheduled to arrive at the beginning of interval i. Note that this number can be higher than 1, as is not uncommon in the literature (Welch and Bailey 1952). The service time distribution can be completely arbitrary. The service times are assumed to be identical for all patients and independent of the scheduled appointment times and other patients' service times.

Our overall objective $W(x)$ is a convex combination of three objectives: expected average waiting time of the patients, the expected total idle time of the doctor before the start of the last patient, and the expected time that the session exceeds the planned finish time (which is Td). The number of possible schedules is

$$\binom{T+N-1}{N},$$

equal to 5.2×10^{10} in the example, illustrating the necessity for an efficient optimization procedure.

Because overtime/lateness and idle time are strongly related, we choose one of the two, in our situation overtime. Note that the overtime is 0 when we finish early. Thus, if t is the time we finish with the last patient, then the overtime is defined as $(t - Td)^+$. Another issue is how we deal with early and late arrivals and the possibility of overtaking which is a consequence of that. In fact, given our objective, the order does not matter, as this does not influence the average waiting time. However, we assume that the work policy is work-conserving, i.e., whenever a patient is available the doctor does not idle, even when the next scheduled patient is not there yet. Other choices are of course possible and easily implemented in our simulation.

3 OPTIMISATION METHOD

The problem is discrete in nature, so we will use methods suitable for discrete problems. Because the structure of the problem and the solution is not known, we will not be able to use this information in designing a good algorithm. From previous work (Koeleman and Koole 2012) we know that local search works well for the same problem when all patients are assumed to be punctual; for a specific neighborhood it is guaranteed to find the optimal schedule within a reasonable amount of time. With this knowledge, we decided to try a random search algorithm with a similar neighbourhood.

We chose local random search over global random search because from a practical viewpoint finding a good schedule in a reasonable running time, even if not optimal, is much more valuable than having guaranteed convergence to the optimal schedule. The algorithm we use works as follows: we start with a schedule chosen randomly from all possible schedules. This schedule is simulated a few times, and a next schedule is chosen randomly from a neighbourhood of that first schedule in which all schedules in the neighborhood have equal probability of being drawn. This is also simulated a few times, and then with a certain (high) probability we choose the best of the two solutions as the next one, and with small probability we choose the other one. This ensures that the algorithm improves steadily, but the small probability of choosing the less good solution gives a way out of a local optimum. In our situation 0 worked well because of the high variability of the simulation outcomes.

As neighbourhood of a schedule we choose all possible schedules made up of a shift of a patient from one interval to the one just before it compared to the current schedule, or if the patient is now scheduled at the first interval, to the last interval. Note that this is not the same neighbourhood as used in Kaandorp and Koole (2007) and Koeleman and Koole (2012), but smaller. We made this choice after some experimentation, in which we found out that the performance of all schedules in the original neighbourhood differed so much that the algorithm started to drift aimlessly. The smaller neighbourhood does not produce that behaviour.

Since it is still possible to reach every possible schedule from any starting point, this restriction does not give any problems.

To decide on the final solution we use all information available up to that point, as it is recommended in Andradóttir (1999). This accelerates the algorithm and leads to better results than just taking the final solution or the schedule visited most often as the outcome of the algorithm. Information about the number of times a solution is simulated and the average objective value over all simulations is needed to compute this of course, but since running time is more restrictive than memory this does not create problems. It is only necessary to keep information on schedules that have actually been visited; there is no need to save information on all possible schedules which could become problematic for real-sized problem instances. If necessary we can restrict the final outcome to only those schedules that have been simulated a minimum number of times, so as to lessen the influence of randomness in the performance evaluations. Because of the higher probability of choosing the better solution in each step of the algorithm these are probably the better solutions in any case.

This algorithm can end up in a local optimum, and it might be very hard to get out of there again if the difference in performance between this local optimum and its neighbours is large. Whether this happens or not depends of course on the randomness in the algorithm, but it can also be influenced by the choice of the starting point. We don't know the structure of the value function over the solution space, so we cannot be sure that this does not happen. This is why we choose to restart the algorithm a few times in a new solution which is again randomly chosen from all possible schedules. This gives a greater chance of finding a global optimum, or, if not, then at least a better local optimum.

4 NUMERICAL EXPERIMENTS

To demonstrate the performance of the algorithm described in the last section, we performed some experiments with a small example. In this example we try to optimally schedule $N = 3$ patients in $T = 6$ time slots of $d = 10$ minutes each. This combination of T and N leads to 52 possible schedules. The reason for choosing such a small example is that we can actually simulate the performance of all schedules often enough to get a reasonably certain idea of which one is the optimal schedule. Then we can compare the outcome of the optimisation algorithm to these results.

For the service time we assume an exponential distribution with a mean of 20 minutes. For the non-punctuality we assume a normal distribution around the scheduled arrival time of the patient, with a mean of 5 minutes earlier and standard deviation 5. The arrival times and service times are assumed to be independent of each other and of those of other patients. The objective function is the sum of the total waiting time with weight $\alpha_W = 3$ and the tardiness with weight $\alpha_T = 1$.

First we simulated each possible schedule many times, to see which results we would like to see from the optimisation algorithm. The best schedules and their performance are shown in Table 1.

There is quite some variability in the performance of the schedules from one simulation run to another. However, we simulated often enough to obtain statistical evidence that 1-0-1-0-0-1 is indeed the optimal schedule.

Next we ran a local random search algorithm ten times. The results from the runs are given in Table 2.

The final schedules are all the same here, and indeed equal to the schedule with the best mean objective value from Table 1.

The optimal schedule in the case where all patients are punctual is 1-0-1-0-0-1, with an objective value of 35.97.

Results for larger more realistic example can be seen in Table 3, with 12 intervals, 6 appointments, and non-punctuality of on average 10 minutes early and standard deviation 5. All other parameters are the same. We see that the algorithm has more difficulty in trying to find the best solution. The most often occuring solution is 3-0-1-1-0-0-0-0-0-1-0-0.

Table 1: The ten best schedules with mean and standard deviation of the objective function after 100000 simulation runs.

Schedule	Mean	Standard deviation
1-0-1-0-0-1	57.4 ± 0.3	52.7
1-0-1-0-1-0	58.3 ± 0.4	56.4
1-0-0-1-0-1	58.5 ± 0.3	51.7
1-1-0-0-1-0	60.6 ± 0.4	58.2
1-1-0-0-0-1	60.9 ± 0.3	54.2
1-0-0-1-1-0	61.4 ± 0.4	55.7
1-0-1-1-0-0	62.5 ± 0.4	59.1
1-0-0-2-0-0	63.9 ± 0.4	57.6
1-1-0-1-0-0	64.0 ± 0.4	61.4
1-0-0-0-1-1	64.3 ± 0.3	51.7

Table 2: The ten schedules and objective values resulting from ten random search runs with random starting points.

Schedule	Objective value
1-0-1-0-0-1	57.6
1-0-1-0-0-1	57.7
1-0-1-0-0-1	57.6
1-0-1-0-0-1	57.8
1-0-1-0-0-1	57.6
1-0-1-0-0-1	57.5
1-0-1-0-0-1	57.6
1-0-1-0-0-1	57.4
1-0-1-0-0-1	57.4
1-0-1-0-0-1	57.5

5 CONCLUSIONS AND FUTURE RESEARCH

We can see from the numerical results that scheduling using simulation provides better results than the analytical methods when the assumptions are not met, even when taking into account the uncertainty in the optimal solution. This may have been expected, but what we did not expect was the shorter runtimes of the simulation optimisation algorithms compared to for example the local search methods, for this small problem. Currently we are experimenting with larger, more realistically sized problems.

We think that the methods, as they stand right now, are flexible enough to be implemented in an actual outpatient clinic. There is a real need for this as good scheduling rules are almost never available to hospital personnel at the moment. A fast and easy base schedule could improve the performance in many cases without increasing any costs.

In the future we want to extend our experiments in a number of directions. We would like to include multiple types of patients, as typically occurs in outpatient clinics they make a difference between new and repeat patients. We also would like to experiment with other performance measure such as the probability that patients have to wait longer than a certain number of minutes, similar to the standard service level definition in call centers. Finally, we plan to experiment with other optimization algorithms, such as nested partitions.

Table 3: The ten schedules and objective values resulting from ten random search runs with random starting points for an example with 6 appointments and 12 intervals.

Schedule	Objective value
3-1-0-1-0-0-0-1-0-0-0-0	128.8
3-1-0-0-0-1-0-0-0-1-0-0	128.4
0-0-0-2-1-1-0-1-0-0-1-0	132.2
3-0-0-1-0-0-0-1-0-0-1-0	124.8
2-2-0-1-0-0-0-1-0-0-0-0	124.5
2-1-0-1-0-0-1-0-1-0-0-0	124.8
3-0-2-0-0-0-0-1-0-0-0-0	131.4
3-0-1-1-0-0-0-0-0-1-0-0	127.6
3-0-1-1-0-0-0-0-0-1-0-0	126.5
0-0-3-0-1-0-0-0-0-1-0-1	125.0

REFERENCES

Andradóttir, S. 1999. "Accelerating the convergence of random search methods for discrete stochastic optimization". *ACM Trans. Model. Comput. Simul.* 9 (4): 349–380.

Cayirli, T., and E. Veral. 2003. "Outpatient scheduling in health care: a review of literature". *Production and Operations Management* 12 (4): 519–549.

Gupta, D., and B. Denton. 2008. "Appointment scheduling in health care: Challenges and opportunities". *IIE Transactions* 40 (9): 800–819.

Jouini, O., and S. Benjaafar. 2012. "Queueing systems with appointment-driven arrivals, non-punctual customers, and no-shows". Working paper.

Kaandorp, G., and G. M. Koole. 2007. "Optimal outpatient appointment scheduling". *Health Care Management Science* 10:217–229.

Koeleman, P. M., and G. M. Koole. 2012. "Optimal outpatient appointment scheduling with emergency arrivals and general service times". *IIE Transactions on Healthcare Systems Engineering* 2 (1): 14–30.

Welch, J., and N. Bailey. 1952. "Appointment Systems in Hospital Outpatient Departments". *The Lancet* 259:1105–1108.

AUTHOR BIOGRAPHIES

PAULIEN KOELEMAN obtained masters in both Business Analytics and Classical Languages from VU University Amsterdam. She finished Business Analytics with an internship at VU University medical center. After that she became a consultant at CC Zorgadviseurs, and she started working on her doctorate part-time, both in the area of health operations management. She expects to defend her PhD thesis in the end of 2012. Her email address is paulien@few.vu.nl.

GER KOOLE is full professor at VU University Amsterdam. He graduated in Leiden on a thesis on the control of queueing systems. Since then he held post-doc positions at CWI Amsterdam and INRIA Sophia Antipolis. His current research is centered around service operations, especially call centers, health care and, more recently, revenue management. Dr. Koole is founder of CCmath, a call center planning company, of Adscience, a software company active in the area of online marketing, and of PICA, the VU University/medical center joint knowledge center on health care operations management. He teaches on the theory and applications of stochastic modeling at all levels, from PhD students to professionals in call centers and hospitals. His email address is ger.koole@vu.nl.