Knowledge Before Solutions: Some Reflections on a Successful O.R. Case Study

Giovanni Righini and Pier Giorgio Villani

Abstract This paper describes how a capacity planning problem arising in health care services design and optimization was successfully tackled with mathematical programming techniques. What made the project successful was not the design of a sophisticated algorithm providing optimal solutions, but rather the iterative development of an integer linear programming model of the problem, solved by a general-purpose MILP solver. This approach was made possible by the characteristics of the mathematical model itself and the user-friendly tools that were used. As a result, the problem expert could autonomously challenge and improve the model and the data in a countless number of iterations with little or no intervention of the O.R. expert. This allowed to reduce the development cost to zero and the development time to a few days.

Keywords Decision science · Mathematical modelling · Capacity planning

1 Introduction

On 18/02/2020 a 37-year-old man in apparent good health and with no pathological history came to the emergency room of Codogno hospital for fever, dyspnea and productive cough, on X-ray evidence of right basal pneumonia. Excluding the most common causes of pneumonia, the alarm bell, which lead to the execution of the swab test for COVID19, was his distant connection with China, linked to the visit of an acquaintance who had recently returned from the East. On February 20th at 9:30 pm confirmation of the positivity at the swab test for Sars-Cov-2 arrived [3]. The regional administration of Lombardy gave the crisis unit of Lodi hospital complete

G. Righini (✉)
Department of Computer Science, University of Milan, Milan, Italy
e-mail: giovanni.righini@unimi.it

P. G. Villani
Department of Emergency and Critical Care, Hospital of Cremona, Cremona, Italy
e-mail: piergiorgio.villani@asst-cremona.it

L. Amorosi et al. (eds.), *Optimization in Artificial Intelligence and Data Sciences*, AIRO Springer Series 8, https://doi.org/10.1007/978-3-030-95380-5_9

power in decision-making for managing the emergency [2]. On February 21st, the crisis unit decided to close the Codogno hospital to concentrate resources in a single hospital, that of Lodi. For three months the operating room activity in Codogno and Lodi was interrupted and all physicians of the anesthesia and resuscitation service were engaged in facing the health emergency in Lodi.

At the end of May 2020 the heads of the two hospitals decided to reopen the first aid and intensive care service in Codogno together with the operating rooms of Lodi and Codogno, at the beginning of June. Therefore, the problem arose of ensuring coverage of all work shifts in the two hospitals, while at the same time guaranteeing annual leave periods in the summer for all physicians of the anesthesia and intensive care service.

The problem was tackled and solved thanks to a mathematical optimization model, developed in collaboration between a physician and an O.R. expert. The aim of this short paper is to illustrate and discuss the use of mathematical optimization models for decision-support in a real situation, highlighting the importance of knowledge generation compared with optimal solution computation as well as the flexibility and ease-of-use of models explicitly described in mathematical terms.

2 The Problem

The problem was a capacity planning problem: a given number of physicians with non-identical skills have to be assigned to six different types of work shifts for a given number of weeks, allowing some of them to be on annual leave in each week. The goal was not to define a complete work schedule for the physicians, taking into account, for instance, preferences, additional activities, balance requirements in work shifts assignments in holidays and other details (a problem of this kind is described in [1]). The goal was rather to understand how the strategic decision of reopening the health services could be implemented. A formal description of the problem is given hereafter.

Data The problem data are the following:

1. a set P of physicians;
2. a set T of types of work shifts; namely: Morning, Afternoon, Night, On-call availability, Operating room, Clinic service, Rest (compulsory day of rest following a night work shift);
3. a set D of days of the week (1=Monday, 7=Sunday);
4. a set W of weeks (the planning horizon);
5. a set S of hospitals (Lodi and Codogno).

Additional data appear only in some constraints and therefore they are described when these constraints are introduced in the remainder.

Variables The main variables of the model are the following:

1. Binary variables x representing the selected assignments have five indices: $x_{twdsp} = 1$ if and only if physician $p \in P$ is assigned a work shift of type $t \in T$ in day $d \in D$ of week $w \in W$ in hospital $s \in S$.
2. Additional binary variables y represent non-working days besides days on leave: $y_{wdp} = 1$ if and only if physician $p \in P$ does not work in day $d \in D$ of week $w \in W$.
3. Variables $h' \geq 0$ and $h'' \geq 0$ count the number of work shifts not covered by the available staff of physicians, but assigned to external resources, which implies an extra cost for the administration: h'_{wd} is the number of operating room work shifts assigned to external resources in day $d \in D$ of week $w \in W$; h''_{wd} is the number of night work shifts assigned to external resources in day $d \in D$ of week $w \in W$. Owing to the integrality of the x variables and the right-hand-sides of the assignment constraints (see below), it is not necessary to impose integrality requirements on these variables.

The model also includes other variables that appear only in some constraints and are thus described when needed in the remainder.

Constraints

Assignment Constraints for Work Shifts in Lodi ($s = 1$)

1. Five work shifts must be covered every day: morning, afternoon, night, on-call availability and rest.

$$\sum_{p \in P} x_{twd1p} = 1 \ \forall w \in W, d \in D, t \in \{1, 2, 3, 4, 7\}$$

2. A given number ω_w of operating rooms ($t = 5$) must be active in Lodi in the working days ($d = 1, \ldots, 5$) of each week. Operating rooms are not active in the week-ends.

$$\sum_{p \in P} x_{5wd1p} + h'_{wd} = \omega_w \ \ \forall w \in W, d \in \{1, \ldots, 5\}$$

$$\sum_{p \in P} x_{5wd1p} = 0 \ \ \forall w \in W, d \in \{6, 7\}$$

3. A clinic service work shift ($t = 6$) is required every Tuesday and Thursday ($d = 2, 4$) and not in the other days of the week.

$$\sum_{p \in P} x_{6wd1p} = 1 \ \ \forall w \in W, d \in \{2, 4\}$$

$$\sum_{p \in P} x_{6wd1p} = 0 \ \ \forall w \in W, d \in \{1, 3, 5, 6, 7\}$$

Assignment Constraints for Work Shifts in Codogno ($s = 2$)

1. Two work shifts must be covered every day: morning and on-call availability ($t = 1, 4$).

$$\sum_{p \in P} x_{twd2p} = 1 \; \forall w \in W, d \in D, t \in \{1, 4\}$$

2. Afternon and clinic service work shifts ($t = 2, 6$) are not required.

$$\sum_{p \in P} x_{twd2p} = 0 \; \forall w \in W, d \in D, t \in \{2, 6\}$$

3. A night work shift ($t = 3$) must be covered every day, either by the available staff or by external resources.

$$\sum_{p \in P} x_{3wd2p} + h''_{wd} = 1 \; \forall w \in W, d \in D$$

4. A single work shift for the operating room ($t = 5$) must be covered on Wednesdays and Fridays ($d = 3, 5$). The operating room is not active in the other days of the week.

$$\sum_{p \in P} x_{5wd2p} = 1 \;\; \forall w \in W, d \in \{3, 5\}$$

$$\sum_{p \in P} x_{5wd2p} = 0 \;\; \forall w \in W, d \in \{1, 2, 4, 6, 7\}$$

Skills Not all physicians can be assigned to each work shift: incompatibilities are easily forbidden by fixing the corresponding binary variables x to 0.

Compulsory Pairings Every night work shift ($t = 3$) must be immediately followed by a rest day ($t = 7$).

$$x_{3,w,d,s,p} = x_{7,w,d+1,s,p} \forall w \in W, d \in \{1, \ldots, , 6\}, s \in S, p \in P$$

$$x_{3,w,7,s,p} = x_{7,w+1,1,s,p} \forall w \in W, s \in S, p \in P.$$

Similar constraints were also introduced to force the correct pairing of the rest day in day 1 of week 1 with the last night work shift of the previous planning period in each hospital.

Forbidden Pairings By contrast, in some other cases pairs of work shifts were declared incompatible, thus forbidding their assignment to the same person. For instance, on-call availability work shifts are incompatible with morning and afternoon shifts in the next day.

$$\sum_{s \in S} \left(x_{4,w,d-1,s,p} + x_{1,w,d,s,p}\right) \leq 1 \ \ \forall w \in W, d \in D, p \in P$$

$$\sum_{s \in S} \left(x_{4,w,d-1,s,p} + x_{2,w,d,s,p}\right) \leq 1 \ \ \forall w \in W, d \in D, p \in P$$

Similar constraints were introduced at the boundaries of the planning horizon.

Joint Work Shifts Some work shifts can be joined, i.e. they can be assigned to the same physician in the same day. The constraint that forbids multiple assignments of shifts to physicians has the form $\sum_t x_{twdsp} \leq 1 \ \forall w, d, s, p$. The possibility of joining two work shifts was introduced by assigning them coefficient $1/2$ in the left-hand-side of the constraint. These constraints were one of the main issues that were examined, to explore the boundary between feasible and infeasible instances. Here is a sample set of constraints among the many that were tested.

1. In the working days ($d = 1, \ldots, 5$) in Lodi $s = 1$ it is allowed to join morning and afternoon shifts $t = 1, 2$ as well as clinic service and on-call availability shifts ($t = 4, 5$).

$$\frac{1}{2} \sum_{t \in \{1,2,4,5\}} x_{t,w,d,1,p} + \sum_{t \in \{3,6,7\}} x_{t,w,d,1,p} + y_{w,d,p} \leq 1 \ \ \forall w \in W, d \in \{1, \ldots, 5\}, p \in P$$

2. On Saturdays ($d = 6$) in Lodi ($s = 1$) the morning shift and the on-call availability shift ($t = 1, 4$) can be joined.

$$\frac{1}{2} \sum_{t \in \{1,4\}} x_{t,w,6,1,p} + \sum_{t \in \{2,3,7\}} x_{t,w,6,1,p} + y_{w,6,p} \leq 1 \ \ \forall w \in W, p \in P$$

3. On Sundays ($d = 7$) in Lodi ($s = 1$) the morning shift and the afternoon shift ($t = 1, 2$) can be joined.

$$\frac{1}{2} \sum_{t \in \{1,2\}} x_{t,w,7,1,p} + \sum_{t \in \{3,4,7\}} x_{t,w,7,1,p} + y_{w,7,p} \leq 1 \ \ \forall w \in W, p \in P$$

4. When operating rooms are active in Codogno ($s = 2$) on Wednesday and Friday ($d = 3, 5$), the operating rooms shifts $t = 5$ can be joined with on-call availability shifts ($t = 4$).

$$\frac{1}{2} \sum_{t \in \{4,5\}} x_{t,w,d,2,p} + \sum_{t \in \{1,3,7\}} x_{t,w,d,2,p} + y_{w,d,p} \leq 1 \ \ \forall w \in W, d \in \{3, 5\}, p \in P$$

5. In the other days in Codogno no work shifts can be joined.

$$\sum_{t \in \{1,3,4,7\}} x_{t,w,d,2,p} + y_{w,d,p} \leq 1 \;\; \forall w \in W, d \in \{1, 2, 4, 6, 7\}, p \in P$$

6. No two work shifts can be joined if they belong to different hospitals.

$$x_{t',w,d,1,p} + x_{t'',w,d,2,p} \leq 1 \;\; \forall t' \in T, t'' \in T, w \in W, d \in D, p \in P$$

Forced Joined Work Shifts In some cases two work shifts are mandatorily joined, i.e. they must be assigned to the same physician in the same day.

1. On Sundays ($t = 7$) in Lodi ($s = 1$) morning and afternoon work shifts are joined.

$$x_{1,w,7,1,p} = x_{2,w,7,1,p} \;\; \forall w \in W, p \in P$$

2. On Saturdays ($t = 6$) in both hospitals morning and on-call availability work shifts ($t = 1, 4$) are joined.

$$x_{1,w,6,s,p} = x_{4,w,6,s,p} \;\; \forall w \in W, s \in S, p \in P$$

3. In working days ($d = 1, \ldots, 5$) in Lodi ($s = 1$) the physician who is available on-call ($t = 4$) is also assigned an operating room work shift (but not necessarily vice versa).

$$x_{4,w,d,1,p} \leq x_{5,w,d,1,p} \;\; \forall w \in W, d \in \{1, \ldots, 5\}, p \in P$$

4. On Wednesdays and Fridays ($d = 3, 5$) in Codogno ($s = 2$) the physician who is available on-call ($t = 4$) is also assigned an operating room work shift (but not necessarily vice versa).

$$x_{4,w,d,2,p} \leq x_{5,w,d,2,p} \;\; \forall w \in W, d \in \{3, 5\}, p \in P.$$

Days Off A complicating feature of the model is the presence of days off. An integer variable r_{wp} indicates how many days off a physician $p \in P$ must have in week $w \in W$. Days off in week w are assigned to physicians who have been assigned demanding work shifts, such as a joint morning+afternoon shift on Sunday of week $w - 1$ or a night shift on Saturday or Sunday in week $w - 1$.

$$r_{w,p} = x_{1,w-1,7,1,p} + \sum_{s \in S} (x_{3,w-1,6,s,p} + x_{3,w-1,7,s,p}) \;\; \forall w \in W, p \in P$$

Similar constraints are used to make the plan in week 1 consistent with the last work shifts assigned in the previous days.

Annual Leaves All physicians must be on leave for two weeks every year. This period is usually concentrated in two consecutive weeks in the summer. The need for this study was triggered by the question whether leave periods were compatible with the need of covering all services in the two hospitals and it proved that actually this would have been impossible without changing the constraints of the problem.

A binary variable $f_{wp} = 1$ indicates that physician $p \in P$ is on leave in week $w \in W$. These variables occur in several constraints.

1. Days off cannot be taken in leave weeks

$$r_{wp} + f_{wp} \leq 1 \;\; \forall w \in W, p \in P.$$

   Suitable boundary constraints ensure that in the last days of the planning horizon not too many demanding work shifts are assigned to physicians in the subset of those who still have to be assigned leave weeks. This is done to make these constraints feasible in the next planning period.
2. Physicians on leave cannot be assigned any work shift, apart from the rest day following a night shift (a rest day can occur at the beginning of a leave week).

$$x_{t,w,d,s,p} + f_{wp} \leq 1 \;\; \forall t \in \{1, \ldots, 6\}, w \in W, d \in D, p \in P, s \in S.$$

3. The number of non-working days for each physician during a non-leave week is equal to 1 plus the number of required days-off.

$$\sum_{d \in D} y_{wdp} \geq 1 - f_{wp} + r_{wp} \;\; \forall w \in W, p \in P$$

4. A suitable constraint was introduced to force consecutive leave weeks for each physician.

$$f_{w,p} = f_{w+1,p} \;\; \forall w \in W : w \pmod 2 = 1, p \in P$$

5. A different number ϕ_w of physicians on leave was decided for each week $w \in W$, according to the forecasted needs of the hospitals, and it was imposed by suitable constraints.

$$\sum_{p \in P} f_{w,p} \geq \phi_w \;\; \forall w \in W.$$

   It was used as a lower bound to give the model more flexibility.
6. A maximum number of leave weeks is given for each physician in each planning period. Subset P' includes physicians that have not been assigned leave weeks in the previous planning periods.

$$\sum_{w \in W} f_{w,p} \leq 2 \;\; \forall p \in P'.$$

For those who have already been assigned leave weeks, variables f are forced to 0.

Objective The model was initially formulated just to check the satisfiability of all constraints, with no objective. Then it was formulated to minimize the number of work shifts to be assigned to external physicians.

$$\text{minimize} \quad \sum_{w \in W, d \in D} (h'_{w,d} + h''_{w,d}).$$

Finally, it was used in a multi-objective fashion to perform a parametric analysis to explore the trade-off between the number of external work-shifts and some indicators of the quality of service in the two hospitals from the viewpoint of patients and physicians. For instance, one of these indicators was the number of joint morning+afternoon shifts in Lodi (to be minimized) apart from the week-ends (when they are explicitly forced to occur). Such an objective can be expressed as follows:

$$\text{minimize} \quad \sum_{w \in W, d \in \{1,\ldots,5\}} \delta_{w,d}$$

with the additional constraints

$$x_{1,w,d,1,p} + x_{2,w,d,1,p} \leq 1 + \delta_{w,d} \;\; \forall w \in W, d \in \{1, \ldots, 5\}, p \in P,$$

where $\delta_{w,d} \geq 0$ is a non-negative auxiliary variable that is forced to 1 every time the same person p is assigned both the morning and the afternoon shifts ($t = 1$ and $t = 2$) in a working day ($d \in \{1, \ldots, 5\}$) of a week w in Lodi ($s = 1$).

2.1 The Solution Process

Since time constraints did not allow for the development of a customized mathematical programming algorithm and since there was no budget to carry out the analysis, it was mandatory to rely on a free MILP solver. The solver *glpsol* with its *Gusek* interface was selected, both because it is free and because of its ease of use.

Real instances with 12 physicians and a time horizon of 14 weeks turned out out to be by far out of reach for *glpsol*. For this reason the model was solved in a rolling horizon fashion, two weeks at a time. Provably optimal solutions were not always found, depending on the activated and deactivated constraints. However, a five minutes timeout for each run was enough to provide the necessary insight into the problem complexity and explore the boundary between feasibility and infeasibility.

The model, written in MathProg and including many comments, was about two hundreds lines long, plus an additional hundred lines for the data and the commands to produce an easy-to-understand output in a text file. The user-friendliness of the Gusek interface and the MathProg language turned out to be instrumental for the success of the study, because it made possible to the problem expert to directly use the model written and commented by the O.R. expert, thus speeding up the process. The problem expert necessarily had to learn the MathProg language, which is a standard in mathematical programming and is well documented on the web. In this way he could grasp the meaning of each instruction, becoming able to add, remove or modify constraints and objectives autonomously. Hence he modified and run the mathematical model countless times, since each solution (including the answer "No feasible solution found") was used as a starting point to modify either the model (the possible decisions, the constraints to be enforced, the objectives to be optimized) or the data or both. The main model parameters to act upon were the type and number of required services, the different possible definitions of allowed joined shifts, the use of external resources in specific services in either hospital, and the rules to assign days-off.

3 Discussion and Conclusions

The mindset of O.R. experts is instinctively oriented to the computation of optimal solutions through efficient algorithms that suitably exploit the mathematical properties of the models representing critical decision problems. However, one of the main lessons that can be learned from this study concerns the generation of knowledge that comes well before the computation of an optimal solution and can even be treated as an objective by itself.

In natural sciences, knowledge is generated by iteratively comparing abstract models with empirical observations. Every model is challenged by new observations, triggering the search for more general or more refined models. In a similar way, when the object of the study is not a natural phenomenon but rather a complex decision problem, knowledge can be generated by continuously improving the mathematical model of the problem: each solution round provides a feedback that challenges the model and the input data, possibly triggering the development of a more detailed model or the collection or observation of more reliable and precise data. When used in this way, mathematical programming is a powerful tool to generate knowledge, well before providing optimal solutions.

It is worth remarking that algebraic modeling languages, relying on mathematics as a universal and unambiguous language, allow any user endowed with a sufficient mathematical education to understand the model and to use it as a tool to investigate the problem, not necessarily to solve it. Furthermore, the separation between the logical structure of the model and the numerical values of the data, placed in two separate files or in two separate sections of the same file is instrumental in making the problem expert autonomous in evaluating alternative models.

This remark is especially important in an age in which “solutionism” is heavily criticized (not without some good reason) [4], “artificial intelligence” is often presented as *the* way to solve complex problems and strong emphasis is placed on data, especially “big” data. However, when using an “artificial intelligence” tool, one could only examine solutions, often without any clue about why they have been suggested and how they depend on data, because the model is not explicit. By contrast, mathematical optimization puts emphasis on models, that are represented in an algebraic language. This allows the user to examine the effects of the changes he himself has introduced into the model. In this way mathematical optimization and decision science aim at empowering human intelligence and ability to understand complex problems, in order to formulate them better and better. Solutions come later, almost as a side effect.

This project was no exception. Its main outcome was not a best possible solution, but first of all a good model. This should be remarked, because in general problem experts do not know what is the right model of their problems at the beginning; they perfectly know their needs, but in general this is not enough to translate them into a model. The search for the model should obviously precede the search for the solutions and not rarely when a solution is provided after countless efforts in algorithm development, it turns out that the model is wrong, incomplete, or flawed for some reason. It may be the case that some “constraints” are not constraints but decisions and the same holds for some “data”. Similarly it may be the case that the initially assumed objective turns out not to be the main objective, because different performance indicators have priority. This is why the definition of a model must be challenged by a critical examination of the solutions (not necessarily the optimal ones) obtained from it.

Making the problem expert autonomous in managing this knowledge generation process was extremely beneficial to the development of the project. The feedback from the solution back to the model and the data typically requires the intervention of both the problem expert and the technical expert. On the contrary, in this case after the development and documentation of an initial ILP model, the iterative feedback was completely managed by the problem expert, with just a limited support from the O.R. expert for major changes, such as the steps from constraint satisfaction to optimization and then to multi-objective optimization. This significantly reduced the effort, the time and the amount of interaction needed to carry the study to a positive end under very strict time requirements (about ten days overall).

Acknowledgments The authors acknowledge the role of Enrico Storti, who strongly supported the adoption of a mathematical optimization approach in the hospitals of Lodi and Codogno and encouraged the writing and presentation of this contribution.

The detailed comments of an anonymous referee were very helpful.

An informative article on the same subject recently appeared on OR/MS Today [5].

References

1. Cervesato, E., Righini, G., Rellini, G.L., Cassin, M., Piazza, R., Nicolosi, G.L.: Optimization of shifts and on-call coverage of cardiologists working in a hospital complex structure by using free software. Comput. Cardiol. **41**, 477–480 (2014)
2. Gagliano, A., Villani, P.G., Co', F.M., Manelli, A., Paglia, S., Bisagni, P.A.G., Perotti, G.M., Storti, E., Lombardo, M.: Epidemic in the central province of Northern Italy: Impact, logistics and strategy in the front-line hospital. Disaster Med. Publ. Health Preparedness **14**(3), 372–376 (2020). https://doi.org/10.1017/dmp.2020.51
3. Micheli, V., Mancon, A., Malara, A., Mileto, D., Villani, P.G., Rizzo, A., Pagani, C., Alquati, O., Gismondo, M.R.: What was behind the first recognition and characterization of the indigenous transmission of SARS-CoV-2 in Italy: the impact on European scenario. Clinical Case Reports, Wiley (2021). https://doi.org/10.1002/ccr3.4154
4. Morozov, E.: To Save Everything, Click Here: The Folly of Technological Solutionism. PublicAffairs, New York (2013)
5. Villani, P.G., Righini, G.: O.R. reopens ORs in Italy. OR/MS Today, INFORMS (April 2021)