Computers & Operations Research 64 (2015) 262–273

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Operating room scheduling with Generalized Disjunctive Programming

Pedro M. Castro [a,b,*], Inês Marques [a,b,c]

[a] Centro de Investigação Operacional, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal
[b] Departamento de Estatística e Investigação Operacional, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal
[c] Universidade Lusófona de Humanidades e Tecnologias, ECEO, Campo Grande, 376, 1749-024 Lisboa, Portugal

ARTICLE INFO

Available online 16 June 2015

Keywords:
Optimization
Mathematical modeling
Logic
Algorithms
Health care

ABSTRACT

This paper addresses the short-term scheduling problem involved in the selection of a subset of elective surgeries from a large waiting list. In order to overcome the combinatorial complexity, a decomposition algorithm is proposed that relies on two continuous-time Generalized Disjunctive Programming (GDP) models. More specifically, there is an upper-level planning model to select surgical assignments to operating rooms and a lower-level constrained scheduling model to synchronize surgeons operating in different rooms on a given day. The GDP models are reformulated using standard convex hull and big-M techniques so as to generate the most efficient set of integer or mixed-integer linear programming constraints. Through the solution of a set of real-life instances from the literature, we show that the new algorithm outperforms a full-space discrete-time formulation and a genetic algorithm, improving the total surgical time as well as the number of performed surgeries by 5%.

1. Introduction

Mixed-integer linear programming models (MILP) are based on algebraic formulations that are not unique, leading to either solvable or intractable problems [1]. Scheduling formulations are no exception and most feature variables and constraints derived from intuition and by trial and error [2] so as to maximize computational performance. Disjunctive programming [3] is an alternative representation of integer programs that is the most natural and straightforward way of stating many problems involving logic conditions. Bounded disjunctive programming models can then be reformulated [4] as integer programming models that exhibit strong continuous relaxations, which often translates into shorter computational times. Generalized Disjunctive Programming (GDP) [5] goes one step further by linking Boolean variables to the different terms in a disjunction so that logical conditions expressing relationships between the disjunctive sets can be included. Logic propositions can then easily be transformed into a set of MILP constraints using a truth table. GDP also includes algebraic equations that need to be enforced regardless of the discrete decisions.

Scheduling formulations are commonly classified based on their underlying time representation, either discrete or continuous-time [6]. In discrete-time formulations, the time horizon is typically divided into uniform intervals of known length with all resources reporting to a single grid, greatly facilitating the modeling of shared resources availability. The drawback is related to the accuracy of processing times that need to be approximated to integer multiples of the chosen length, a particularly relevant issue when dealing with time-related objective functions. Continuous-time formulations allow for accurate data in a natural way, with hybrid models relying both on multiple time grids and sequencing variables being sometimes the best approach [7–9]. These references deal with problems where: (i) a single robot is constantly moving to transfer lots between different chemical and water baths in a semiconductor plant; and (ii) a single crew is being used to perform preventive maintenance on multiple gas engines on a power plant; which are closely related to the current problem, where surgeons move between operating rooms to perform surgeries.

The modeling burden of deriving the timing constraints that avoid a shared resource being in different places at the same time can be greatly reduced by first developing a GDP model and then reformulating it into a MILP [7]. Research has also shown that the most efficient MILP model is obtained through a convex hull reformulation of disjunctions linked to multiple time grids [10,11] and a big-reformulation of disjunctions involving sequencing variables [7,10]. GDP allows developing a structured model with nested decisions. As an example, one must first decide on the surgeons that are assigned to a room before selecting the surgeries to execute. Nested decisions, represented as embedded disjunctions [11,12] that can be transformed into simple disjunctions and

* Corresponding author at: Centro de Investigação Operacional, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal.
E-mail address: pmcastro@fc.ul.pt (P.M. Castro).

http://dx.doi.org/10.1016/j.cor.2015.06.002
0305-0548/

Nomenclature

Sets/Indices

C/c	elective surgeries
C_d	surgeries that can be performed on day d
C_h	surgeries to be performed by surgeon h
C_j	surgeries of specialty j
C_p	surgeries of priority level p
C_r	surgeries that can be performed in operating room r
$C_{r,d}$	surgeries planned for room r in day d
D/d	working days of operating rooms
D_w	days belonging to week w
H/h	surgeons
H_c	single surgeon that can perform surgery c
$H_{r,d}$	surgeons operating on room r during day d
$H_{r,r',d}$	surgeons moving around rooms r and r' during day d
J/j	surgical specialties
J_c	specialty of surgery c
P/p	surgery priority levels
P_c	priority level of surgery c
$R/r,r'$	operating rooms
R_d	operating rooms featuring at least one surgeon that also operates on another room on day d
$T/t,t'$	time slots
$T_{r,d}$	time slots of the grid linked to room r and day d
W/w	planning weeks

Parameters

cg	duration of cleaning and disinfecting protocols between surgeries
dh	working hours of operating rooms
$dmax_{h,d}$	daily operating time limit for surgeon h for day d
$or_{h,d}$	number of operating rooms surgeon h is planned to work on day d
pt_c	time required to perform surgery c
st	planned surgical time
st^{OK}	validated planned surgical time
$wmax_h$	weekly operating time limit for surgeon h
$x_{c,r,d}$	optimal values from planning model of variables $X_{c,r,d}$
$z_{h,r,d}$	optimal values from planning model of variables $Z_{h,r,d}$

Variables

$O^H_{h,d}$	occupation time of surgeon h during day d
$O^R_{r,d}$	occupation time of operating room r during day d
$\hat{O}^R_{j,r,d}$	occupation time due to specialty j of operating room r during day d
$Pt_{t,r,d}$	duration of surgery allocated to slot t of room r during day d
$Ts_{t,r,d}$	starting time of slot t of room r during day d
St	total surgical time
$X_{c,r,d}$	binary variable indicating if surgery c is performed at room r during day d
$X^s_{c,t,r,d}$	binary variable indicating if surgery c is executed during time slot t of room r during day d
X^{wait}_c	binary variable indicating if surgery c remains in waiting list for subsequent period
$X^{free}_{t,r,d}$	binary variable indicating if slot t of room r is free from surgeries during day d
$W_{t,r,t',r',d}$	binary variable indicating if the surgery assigned to slot t of room r ends before the start of the surgery assigned to slot t' of room r' in day d
$W^1_{h,t,r,t',r',d}$	binary variable indicating if the surgery assigned to slot t of room r ends before the start of the surgery assigned to slot t' of room r' in day d and both are performed by surgeon h
$W^2_{h,t,r,t',r',d}$	binary variable indicating if the surgery assigned to slot t' of room r' ends before the start of the surgery assigned to slot t of room r in day d and both are performed by surgeon h
$W^3_{h,t,r,t',r',d}$	binary variable indicating that surgeon h does not perform both surgeries assigned to slot t of room r and slot t' of room r' in day d
$Y_{j,r,d}$	binary variable indicating if specialty j is assigned to room r in day d
$Z^{free}_{h,d}$	binary variable indicating if surgeon h is free from surgeries in day d
$Z_{h,r,d}$	binary variable indicating if surgeon h operates in room r during day d
$Z^s_{h,t,r,d}$	binary variable indicating if surgeon h is assigned to slot t of room r during day d

logic propositions, may help to identify interesting decomposition strategies when dealing with large-scale problems.

Decomposition algorithms are particularly interesting when relying on a relaxation of the full-space problem since they allow computing a rigorous optimality gap after a feasible solution has been found. One such algorithm combining the complementary strengths of MILP and constraint programming techniques has been successfully applied to a single-stage scheduling problem with dissimilar parallel machines [13]. The upper decision level takes care of job–machine assignments by solving an optimization problem featuring the full-space scheduling formulation stripped of the complicating sequencing constraints that do not affect the objective function. Then, the lower decision level checks if all jobs can be sequenced on the corresponding machines. When feasibility is confirmed, an optimal solution of the full-space problem has been found (the same as the optimal solution of the relaxed problem). The drawback is that there is no feasible solution before the optimal one, which may take some time to obtain.

The focus of heuristic decomposition algorithms is on obtaining good feasible solutions in a short computational time. They are often used for more complex production environments and may include modules for: (i) dividing the full set of jobs into groups of similar characteristics; (ii) constructing a feasible schedule by considering one or a few jobs at a time; and (iii) improving the schedule locally by removing and reinserting jobs in different positions of the processing sequence or reassigning them to different machines. Successful approaches have been reported for the steel [14], paper [15], pharmaceutical [16,17] and consumer goods industries [18].

In this paper, we address the solution of a large-scale problem concerning the schedule of elective surgeries at a central and university hospital. Previous work dealing with the same problem [19,20] has shown that a full-space discrete-time integer programming formulation is unable to find good quality solutions when dealing with over a couple thousand surgeries in the waiting list. The objective function of maximizing total surgical time over a week is time-related, and so a continuous-time approach is desired. We propose a decomposition algorithm where the main element is a planning model that is a relaxation of the full-space continuous-time model, thus allowing associating an optimality gap to the obtained solutions. By first developing the Generalized