Omega 93 (2020) 102043

Contents lists available at ScienceDirect

Omega

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

Reformulation, linearization, and decomposition techniques for balanced distributed operating room scheduling☆

Vahid Roshanaei[a,*], Curtiss Luong[b], Dionne M. Aleman[b,c,d], David R. Urbach[e]

[a] Rotman School of Management, University of Toronto, 105 Street George, Toronto, ON M5S 3E6, Canada
[b] Department of Mechanical and Industrial Engineering, University of Toronto, 5 King's College Road, Toronto, ON M5S 3G8, Canada
[c] Institute of Health Policy, Management and Evaluation, University of Toronto, 155 College Street, Suite 425, Toronto, ON M5S 3E3, Canada
[d] Techna Institute at University Health Network, 124-100 College Street, Toronto, ON M5G 1P5, Canada
[e] Division of General Surgery, Toronto General Hospital, University Health Network, 200 Elizabeth Street, Toronto, ON M5G 2C4, Canada

ARTICLE INFO

Article history:
Received 10 November 2017
Accepted 1 March 2019

Keywords:
Healthcare
Operating room scheduling
Balanced location-allocation
Large scale optimization
Multi-level decomposition
Logic-based Benders balancing cuts
Mixed-integer nonlinear programming

ABSTRACT

We study the balanced distributed operating room (OR) scheduling (BDORS) problem as a location-allocation model, encompassing two levels of balancing decisions: (i) daily macro imbalance among collaborating hospitals in terms of the number of allocated ORs and (ii) daily micro imbalance among open ORs in each hospital in terms of the total caseload assigned. BDORS is formulated as a novel mixed-integer nonlinear programming (MINLP) in which the macro and micro imbalance are penalized using absolute value and quadratic functions. We develop various reformulation-linearization techniques (RLTs) for the MINLP models, leading to three mathematical modelling variants: (i) a mixed-integer quadratically constrained program (MIQCP) and (ii) two mixed-integer programs (MIPs) for the absolute value penalty function and an MIQCP for the quadratic penalty function. Two novel exact techniques based on reformulation-decomposition techniques (RDTs) are developed to solve these models: a uni- and a bi-level logic-based Benders decomposition (LBBD). We motivate the LBBD methods with an application to BDORS in the University Health Network (UHN), consisting of three collaborating hospitals: Toronto General Hospital, Toronto Western Hospital, and Princess Margaret Cancer Centre in Toronto, Ontario, Canada. The uni-level LBBD method decomposes the model into a surgical suite location, OR allocation, and macro balancing master problem (MP) and micro OR balancing sub-problems (SPs) for each hospital-day. The bi-level approach uses a relaxed MP, consisting of a surgical suite location and relaxed allocation/macro balancing MP and two optimization SPs. The primary SP is formulated as a bin-packing problem to allocate patients to open operating rooms to minimize the number of ORs, while the secondary SP is the uni-level micro balancing SP. Using UHN datasets consisting of two datasets, hard MP/easy SPs and easy MP/hard SPs, we show that both LBBD approaches and both MIP models solved via Gurobi converge to $\approx$2% and $\approx$1–2% optimality gaps, on average, respectively, within 30 minutes runtime, whereas the MIQCP solved via Gurobi could not solve any instance of the UHN datasets given the same runtime. The uni- and bi-level LBBD approaches solved all instances of hard MP/easy SPs dataset to $\approx$11% and $\approx$2% optimality gaps, on average, respectively, within 30 minutes runtime, whereas MIQCP solved via Gurobi could not solve any of these instances. Additionally, we show that convergence of each LBBD varies depending on where in the decomposition the actual computational complexity lies.

1. Introduction

We study balanced distributed operating room scheduling (BDORS) problem, which is an extension of the distributed operating room scheduling (DORS) problem that schedules patients and operating rooms (ORs) collaboratively across a coalition of multiple hospitals in a strategic network [41]. Given a pool of patients, DORS decides whether or not a patient will be operated on in the current planning horizon based on his/her wait time and health status score. The selected patients for the current planning horizon are optimally allocated to a date, hospital, and an OR. The allocation minimizes the openings of surgical suites (i.e., hospitals with any open ORs) in the network, and minimizes open ORs within each surgical suite.

☆ Handled by editor O. Prokopyev.

* Corresponding author.
E-mail addresses: vahid.roshanaei@rotman.utoronto.ca (V. Roshanaei), curtiss@mie.utoronto.ca (C. Luong), aleman@mie.utoronto.ca (D.M. Aleman), david.urbach@uhn.on.ca (D.R. Urbach).

https://doi.org/10.1016/j.omega.2019.03.001
0305-0483/

BDORS incorporates caseload balancing into DORS by additionally minimizing the caseload imbalance among collaborating hospitals (macro imbalance) and also among opened ORs within the surgical suite of each hospital-day (micro imbalance). Macro imbalances in caseload distribution may undermine the collaborative spirits of hospitals, while micro imbalances in a hospital may yield periods of nurse and surgeon overloading, which can result in stress and poor quality of care [28]. Imbalance can be penalized linearly (absolute value) or quadratically. We penalize macro imbalance with absolute value and examine both absolute value and quadratic penalties for micro imbalance. We apply BDORS to the University Health Network (UHN), consisting of Toronto General Hospital, Toronto Western Hospital, and Princess Margaret Cancer Centre in Toronto, Ontario, Canada.

We formulate BDORS as a novel multi-period location-allocation with caseload balancing mixed-integer nonlinear programming (MINLP) model. To solve this MINLP with the absolute value penalty function, we develop two reformulation-linearization techniques (RLTs), leading to a partially linearized mixed-integer quadratically constrained program (MIQCP) and two fully linearized mixed-integer programs (MIP), the structures of which allow them to be solved via Gurobi. We also develop an RLT to reformulate the MINLP with quadratic penalty function into a new MIQCP model that can also be solved by Gurobi.

We develop two novel exact techniques based on reformulation-decomposition techniques (RDTs): a uni- and a bi-level logic-based Benders decomposition (LBBD) to solve BDORS with absolute value penalty functions. These exact techniques are capable of solving BDORS with both the MIQCP and MIP RLTs. We then slightly modify these LBBD approaches, enabling them to handle the quadratic micro balancing penalty function. The contributions of this paper are as follows:

- We introduce BDORS as a new problem in the area of OR scheduling and formulate it as an MINLP with absolute value penalties for macro imbalance and both absolute value and quadratic penalties for micro imbalance.
- We develop different linearization techniques for the BDORS MINLP model, leading to partially (MIQCP) and fully (MIP) linearized models.
- We develop novel RDT-based LBBD approaches to solve the MIQCP and MIP models, which remove the nonlinearity of the MINLP models. We show that our Benders balancing cuts can handle both the absolute value and quadratic penalty functions in the same way.
- We develop and prove the validity of novel Benders balancing optimality cuts that connect the master problem (MP) and sub-problems (SPs) of BDORS and facilitate the development of LBBD approaches for other balancing optimization problems.

Summary of results. Using UHN datasets [41], we show that the MIQCP solved via Gurobi with absolute value penalty function does not solve any of the UHN instances, whereas both MIP models solved via Gurobi and the LBBD approaches converge to $\approx$1–2% and $\approx$2% average optimality gaps, respectively, within 30 minutes runtime. Using a quadratic penalty function, we show that the MIQCP solved via Gurobi cannot solve any of the UHN instances while the uni- and bi-level LBBD approaches can solve all instances of hard MP/easy SPs to $\approx$11% and $\approx$2% average optimality gaps, respectively, within 30 minutes runtime. We show that the bi-level LBBD produces lower average optimality gaps than the uni-level LBBD for both absolute value and quadratic micro balancing penalty functions. We finally show that RDT-based LBBD approaches are more robust in solving BDORS problems than RLT-based techniques.

2. Literature review

We provide a review of studies focused on scheduling problems in distributed environments, with specific emphasis on OR scheduling. We additionally review solution techniques developed for MINLP models and LBBD approaches and address the challenges and shortcomings of existing methods.

2.1. Operating room scheduling

The process of surgical suite management consists of strategic, tactical, and operational levels [22,43]. At the strategic level, OR times are allocated among different surgical specialties based on surgical demands (case mix planning) [1,2,50]. At the tactical level (master surgical scheduling), the total OR time allotted to each surgical specialty is divided among surgeons of that specialty [3,44,45]. At the operational level (surgical case scheduling), patients are allocated to OR-days and a priority position within an OR [4,15,17,23,27,32,33,37,40,49].

The benefits of distributed job scheduling in manufacturing have recently garnered significant attention [6,7,25,34–36]. However, the problem of human (surgeons, nurses, and anesthesiologists) and physical (ORs and recovery beds) resource pooling across multiple collaborating hospitals (facilities) is a new topic that has attracted limited attention, with only distributed master surgical scheduling [44] and distributed surgical case scheduling [40,41,51] being previously investigated. Roshanaei et al. [40] developed an LBBD approach for a collaborative operating room planning and scheduling problem, and used a game theoretic approach to show that the hospital coalition is stable and results in substantial cost-savings. Roshanaei et al. [41] developed an integer program (IP) for a multi-hospital priority-based OR scheduling problem, solved via multiple variants of LBBDs, resulting in two orders of magnitude computational savings over the IP model. Using real UHN data, Wang et al. [51] used simulation to demonstrate the robustness of the deterministically-optimized OR schedules constructed via the DORS IP of Roshanaei et al. [41] in the presence of emergency surgery arrivals, uncertain use of downstream units, and stochastic duration of surgeries, thereby demonstrating that deterministically-optimized OR schedules can provide significant efficiency gains in real-world scenarios.

According to three recently published surveys on OR scheduling problems [12,22,43] and also to the best of our knowledge, BDORS, consisting of simultaneous macro and micro balancing among multiple collaborating hospitals, has not been previously studied, though hospital balancing problems of smaller scope have received some attention. Marcon et al. [30] studied the problem of caseload balancing among ORs using a quadratic function in a single hospital under a fixed number of open ORs and uncertain surgical durations via an optimization-simulation approach. Galvao et al. [19] studied a bi-objective load balancing hierarchical assignment problem among neonatal clinics, and found optimal solutions using CPLEX and a Lagrangian heuristic for small instances and showed that their heuristic was capable of producing high-quality sub-optimal solutions. The load balancing nurse-to-patient allocation problem in downstream units of ORs has been modeled as a constraint integer and a mixed-integer quadratic programming model solved via CPLEX [28]; such approaches are unable to solve BDORS due to its bi-linear nonlinear structure, emanating from opening and closing decisions of ORs, which requires a tailor-made decomposition strategy.

2.2. Solution techniques for mixed-integer nonlinear programs

MINLP models can be convex or nonconvex. Convex MINLP models can be tackled in a variety of ways [8], using nonlinear-

programming-based branch-and-bound (B&B), multi-tree methods (outer approximation, Generalized Benders Decomposition, extended cutting-plane method), single-tree methods (LP/NLP-based B&B), and presolve techniques (coefficient tightening for MINLP and constraint disaggregation). Cutting-plane methods, including mixed-integer rounding cuts, perspective cuts, disjunctive cuts, Gomory cuts, and lift-and-project cuts, are popular for solving convex MINLP models [8]. A comprehensive review of techniques applied to convex MINLP models can be found in [21].

Nonconvex MINLP models are challenging due to nonconvexities in objective functions and constraints. Therefore, even when the integrality restrictions on integer decision variables are relaxed, the resulting feasible region may be nonconvex, requiring more effort to obtain an efficiently solvable convex relaxation that can be used in a B&B framework [8]. Despite the additional effort in approximating an efficient convex relaxation of the feasible region of an MINLP, most techniques specialized for MINLPs yield many local optima without guaranteeing global optimality [8]. Popular techniques to solve nonconvex MINLP models [21] are piecewise linear modelling, generic relaxation strategies, spatial B&B, and relaxation of structured nonconvex sets. Also, heuristic techniques are widely used to solve nonconvex MINLP models [8], including mixed-integer-based rounding, feasibility pump, undercover, relaxation enforced neighborhood search, and diving.

While RLTs are widely used for MINLP models [8,21], RDT-based approaches have received much less attention. The RDT-based approaches require fewer variables and constraints and lend themselves well to the logic-based decomposition approach with linear components, whereas RLT-based approaches require more variables and constraints and are solved as a single optimization problem. An RDT-based LBBD approach has already been developed for an MINLP model to solve the optimal synthesis of process networks using generalized disjunctive modelling with nonlinear SPs [48]. To the best of our knowledge, no study has developed RDT-based LBBD approaches as we propose.

2.3. LBBD

LBBD, first proposed by Hooker [24] and Hooker and Ottosson [26], is an exact row-generation technique for solving large-scale combinatorial optimization problems [24–26,38,47,52]. LBBD has been applied successfully to location-allocation [5], parallel machine scheduling with sequence-dependent setup times [47], travelling purchaser problem [11], and multi-level OR planning and scheduling [39] problems. The integration of Benders-type cut generation within cutting plane algorithms has also been studied [10,18,20]. None of these LBBD approaches can solve BDORS due to its bi-linear nonlinear structure and the need for balancing Benders cut development.

3. Mathematical models

BDORS extends the IP non-balanced DORS in [41] to include daily macro and micro balancing objective functions. DORS considers two sets of patients: mandatory and optional. Mandatory patients must be operated on during the current planning horizon, but optional patients are scheduled only if capacity permits. DORS allocates the optional patients to hospitals, days, and ORs in such a way that patients with higher health status scores, calculated from their wait time in the wait list and their urgency score, are scheduled at the beginning of the planning horizon until the capacity of ORs is exhausted. Patients with lower health status scores may be scheduled for the next planning horizon if there is not enough OR capacity. DORS trades off the required resources for scheduled patients with the rewards obtained from their scheduling. The macro balancing objective function term in BDORS minimizes the daily difference in the number of ORs allocated to hospitals, whereas the micro balancing objective function term ensures caseload balancing among open ORs in each hospital-day. Micro imbalance can be penalized using both absolute value and quadratic functions, requiring different treatments to be solvable efficiently. Mathematical models with absolute value penalty function and quadratic penalty functions are differentiated with subscripts AVPF and QPF, respectively. See Table 1 for notation.

Similar to many recent studies in OR scheduling literature [23,38,40,41,49], we assume deterministic surgical duration in the BDORS for two reasons: (i) the OR (nurse) manager must use the surgeon's provided OR time estimate for scheduling purposes even if it deviates from historical realizations; and (ii) the recent work of [51] on a similar research topic (DORS [41]) using both booked and realized surgical times reveals that surgeons' underestimation and overestimation of surgical durations are mostly counterbalanced with each other on the day of surgery, demonstrating through simulation that stochasticity in surgical durations culminates in only a 1.4% cancellation rate and on average 8 minutes of OR overtime. Thus, it is reasonable to study BDORS from a deterministic perspective.

3.1. Absolute value penalty function (AVPF)

The BDORS $\text{MINLP}_{\text{AVPF}}$ formulation is as follows:

$$\underset{\mathbf{u,y,x,w}}{\text{minimize}} \quad \overbrace{\sum_{h\in\mathcal{H}}\sum_{d\in\mathcal{D}} G_{hd}u_{hd}}^{\text{Cost of surgical suite opening}} + \overbrace{\sum_{h\in\mathcal{H}}\sum_{d\in\mathcal{D}}\sum_{r\in\mathcal{R}_h} F_{hd}y_{hdr}}^{\text{Cost of OR opening}} \qquad \text{MINLP}_{\text{AVPF}}$$

$$+ \overbrace{\kappa_1 \sum_{h\in\mathcal{H}}\sum_{d\in\mathcal{D}}\sum_{p\in\mathcal{P}}\sum_{r\in\mathcal{R}_h}\left[\rho_p(d-\alpha_p)x_{hdpr}\right]}^{\text{Cost of scheduling patients}} + \overbrace{\kappa_2 \sum_{p\in\mathcal{P}\setminus\mathcal{P}'}\left[\rho_p(|\mathcal{D}|+1-\alpha_p)w_p\right]}^{\text{Cost of rejecting patients}}$$

$$+ \overbrace{C_1 \sum_{h=1}^{|\mathcal{H}|-1}\sum_{h'=h+1}^{|\mathcal{H}|}\sum_{d\in\mathcal{D}}\left|\frac{\sum_{r\in\mathcal{R}_h} y_{hdr}}{|\mathcal{R}_h|} - \frac{\sum_{r\in\mathcal{R}_{h'}} y_{h'dr}}{|\mathcal{R}_{h'}|}\right|}^{\text{Cost of macro imbalance}}$$

$$+ \overbrace{C_2 \sum_{h\in\mathcal{H}}\sum_{d\in\mathcal{D}}\sum_{r=1}^{|\mathcal{R}_h|-1}\sum_{r'=r+1}^{|\mathcal{R}_h|}\left|y_{hdr'}\sum_{p\in\mathcal{P}} T_p x_{hdpr} - y_{hdr}\sum_{p\in\mathcal{P}} T_p x_{hdpr'}\right|}^{\text{Cost of micro imbalance}}$$

$$\text{subject to} \quad \sum_{h\in\mathcal{H}}\sum_{d\in\mathcal{D}}\sum_{r\in\mathcal{R}_h} x_{hdpr} = 1 \qquad \forall p\in\mathcal{P}' \tag{1}$$

$$\sum_{h\in\mathcal{H}}\sum_{d\in\mathcal{D}}\sum_{r\in\mathcal{R}_h} x_{hdpr} + w_p = 1 \qquad \forall p\in\mathcal{P}\setminus\mathcal{P}' \tag{2}$$

$$\sum_{p\in\mathcal{P}} T_p x_{hdpr} \le B_{hd} y_{hdr} \qquad \forall h\in\mathcal{H};\ d\in\mathcal{D};\ r\in\mathcal{R}_h \tag{3}$$

$$x_{hdpr} \le y_{hdr} \qquad \forall h\in\mathcal{H};\ d\in\mathcal{D};\ p\in\mathcal{P};\ r\in\mathcal{R}_h \tag{4}$$

$$y_{hdr} \le u_{hd} \qquad \forall h\in\mathcal{H};\ d\in\mathcal{D};\ r\in\mathcal{R}_h \tag{5}$$

$$y_{hdr} \le y_{hdr-1} \qquad \forall d\in\mathcal{D};\ h\in\mathcal{H};\ r\in\mathcal{R}_h\setminus\{1\} \tag{6}$$

$$u_{hd}, y_{hdr}, x_{hdpr} \in \{0,1\} \qquad \forall h\in\mathcal{H};\ d\in\mathcal{D};\ p\in\mathcal{P};\ r\in\mathcal{R}_h \tag{7}$$

$$w_p \in \{0,1\} \qquad \forall p\in\mathcal{P}\setminus\mathcal{P}' \tag{8}$$

Table 1
BDORS notation.

Sets:	
$\mathcal{P}$	Set of patients, $p \in \mathcal{P}$
$\mathcal{P}'$	Set of patients to be scheduled this planning horizon, $\mathcal{P}' = \{\rho_p(\lvert\mathcal{D}\rvert - \alpha_p) \leq -\Gamma\}$
$\mathcal{H}$	Set of hospitals, $h \in \mathcal{H}$
$\mathcal{D}$	Set of days belonging to the first planning horizon, $d \in \mathcal{D}$
$\mathcal{R}_h$	Set of ORs in a hospital's surgical suite, $r \in \mathcal{R}_h$
$\mathcal{I}$	Set of incumbent solutions explored during optimization, $i \in \mathcal{I}$
Parameters:	
G_{hd}	Fixed cost associated with opening the surgical suite in hospital h on day d
F_{hd}	Fixed opening cost of ORs in hospital h on day d
B_{hd}	Regular operating hours of each OR in hospital h on day d
C_1	Daily cost of macro imbalance among hospitals
C_2	Daily cost of micro imbalance among ORs within a hospital
T_p	Total preparation, surgery, and cleaning time of patient p
ρ_p	Urgency score of patient p
α_p	Days elapsed from the referral date of patient p
κ_1	Urgency scaling factor for scheduled patients
κ_2	Urgency scaling factor for unscheduled patients
Γ	Health status threshold above which patients have to be operated
Variables:	
u_{hd}	1 if the surgical suite in hospital h is opened on day d, 0 otherwise
y_{hdr}	1 if OR r in hospital h is opened on day d, 0 otherwise
x_{hdpr}	1 if patient p is assigned to OR r in hospital h on day d, 0 otherwise
$v_{hdprr'}$	1 if patient p is assigned to OR r in hospital h on day d when OR r' $(r \neq r')$ is also open, 0 otherwise
w_p	1 if patient p is not scheduled this horizon, 0 otherwise
$b_{hdrr'}$	Amount of micro imbalance among any pair of open ORs $(r < r')$ in hospital h on day d
b_{hd}	Total amount of micro imbalance among open ORs in hospital h on day d
$z_{hh'd}$	Amount of macro imbalance between any pair of hospitals $(h < h')$ on day d

In the objective function, the cost of micro imbalance between each pair of opened ORs is computed such that there is no cost of imbalance between a closed OR and opened ORs. Constraint (1) enforces that critical patients ($\mathcal{P}'$) be operated on in the current planning horizon, while Constraint (2) allows optional patients to be scheduled. Constraint (3) ensures that no OR is over-capacitated. Constraints (4) and (5) ensure consistency among the variables. Constraint (6) breaks the symmetry among ORs in each hospital-day. For notational simplicity, we denote the first, second, third, and fourth terms in the objective function of MINLP$_{\text{AVPF}}$ as ***U, Y, X***, and ***W***, respectively. These four terms correspond to the objective functions used in DORS [41].

3.2. Quadratic penalty function (QPF)

Similar to workload balancing among ORs [30] and nurses in downstream units of ORs [28] using QPF, we extend the mathematical models to treat micro OR balancing within each hospital-day quadratically. Using QPF instead of AVPF places significant emphasis on micro caseload imbalance. The MINLP model with quadratic micro balancing cost (MINLP$_{\text{QPF}}$) is as follows:

$$\underset{\mathbf{u,y,x,w}}{\text{minimize}} \quad \boldsymbol{U+Y+X+W}$$
$$+ C_1 \sum_{h=1}^{|\mathcal{H}|-1} \sum_{h'=h+1}^{|\mathcal{H}|} \sum_{d\in\mathcal{D}} \left| \frac{\sum_{r\in\mathcal{R}_h} y_{hdr}}{|\mathcal{R}_h|} - \frac{\sum_{r\in\mathcal{R}_{h'}} y_{h'dr}}{|\mathcal{R}_{h'}|} \right| \qquad \text{MINLP}_{\text{QPF}}$$
$$+ C_2 \sum_{h\in\mathcal{H}} \sum_{d\in\mathcal{D}} \sum_{r=1}^{|\mathcal{R}_h|-1} \sum_{r'=r+1}^{|\mathcal{R}_h|} \left(y_{hdr'} \sum_{p\in\mathcal{P}} T_p x_{hdpr} - y_{hdr} \sum_{p\in\mathcal{P}} T_p x_{hdpr'} \right)^2$$
$$\text{subject to} \quad \text{Constraints } (1) - (8)$$

We discuss in Section 10 as to why macro imbalance is not penalized quadratically.

4. Linearization

We show that MINLP$_{\text{AVPF}}$ can be partially linearized (MIQCP$_{\text{AVPF}}$), and that this linearization leads to two fully linearized formulations (MIP$_{\text{AVPF1}}$ and MIP$_{\text{AVPF}_2}$). We construct a partial linearization for MINLP$_{\text{QPF}}$ (MIQCP$_{\text{QPF}}$).

4.1. Linearization schemes for absolute value penalty function

We develop linearization schemes for MINLP$_{\text{AVPF}}$, leading to an MIQCP and two MIPs. We show that MIP models can be linearized with and without the big-M modelling paradigm.

4.1.1. Partial linearization

To linearize the absolute value terms in the MINLP$_{\text{AVPF}}$ objective function, we use two new continuous variables, $z_{hh'd}$ and $b_{hdrr'}$, to contain the costs of macro and micro imbalance, respectively. The partially linearized MINLP (MIQCP$_{\text{AVPF}}$) is as follows:

$$\underset{\mathbf{u,y,x,w,z,b}}{\text{minimize}} \ \boldsymbol{U+Y+X+W}$$
$$+ C_1 \sum_{h=1}^{|\mathcal{H}|-1} \sum_{h'=h+1}^{|\mathcal{H}|} \sum_{d\in\mathcal{D}} z_{hh'd} + C_2 \sum_{h\in\mathcal{H}} \sum_{d\in\mathcal{D}} \sum_{r=1}^{|\mathcal{R}_h|-1} \sum_{r'=r+1}^{|\mathcal{R}_h|} b_{hdrr'} \qquad \text{MIQCP}_{\text{AVPF}}$$

subject to Constraints (1)-(8)

$$z_{hh'd} \geq \frac{\sum_{r\in\mathcal{R}_h} y_{hdr}}{|\mathcal{R}_h|} - \frac{\sum_{r\in\mathcal{R}_{h'}} y_{h'dr}}{|\mathcal{R}_{h'}|} \quad \forall (h,h') \in \mathcal{H} \mid h < h'; d \in \mathcal{D} \tag{9}$$

$$z_{hh'd} \geq -\left(\frac{\sum_{r\in\mathcal{R}_h} y_{hdr}}{|\mathcal{R}_h|} - \frac{\sum_{r\in\mathcal{R}_{h'}} y_{h'dr}}{|\mathcal{R}_{h'}|} \right) \quad \forall (h,h') \in \mathcal{H} \mid h < h'; d \in \mathcal{D} \tag{10}$$

$$b_{hdrr'} \geq y_{hdr'} \sum_{p\in\mathcal{P}} T_p x_{hdpr} - y_{hdr} \sum_{p\in\mathcal{P}} T_p x_{hdpr'} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{11}$$

$$b_{hdrr'} \geq -\left(y_{hdr'} \sum_{p\in\mathcal{P}} T_p x_{hdpr} - y_{hdr} \sum_{p\in\mathcal{P}} T_p x_{hdpr'}\right) \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{12}$$

$$z_{hh'd} \geq 0 \quad \forall (h,h') \in \mathcal{H} \mid h < h';\ d \in \mathcal{D} \tag{13}$$

$$b_{hdrr} \geq 0 \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{14}$$

Continuous variables $z_{hh'd}$ with Constraints (9) and (10) linearize the nonlinear macro balancing absolute terms in the $\text{MINLP}_{\text{AVPF}}$ objective function. Continuous variables $b_{hdrr'}$ with Constraints (11) and (12) linearize the micro balancing absolute term in the $\text{MINLP}_{\text{AVPF}}$ objective function, but do not linearize the inherent bi-linear nature of Constraints (11) and (12) caused by the product of binary variables y_{hdr} and x_{hdpr}.

4.1.2. Full linearization

The $\text{MIP}_{\text{AVPF1}}$ uses the concept of big-M to linearize $\text{MINLP}_{\text{AVPF}}$:

minimize $\text{MIQCP}_{\text{AVPF}}$ objective function $\text{MIP}_{\text{AVPF1}}$

subject to $\text{MIQCP}_{\text{AVPF}}$ constraints without Constraints (11) and (12)

$$b_{hdrr'} \leq M y_{hdr} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{15}$$

$$b_{hdrr'} \leq M y_{hdr'} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{16}$$

$$b_{hdrr'} \geq \sum_{p\in\mathcal{P}} T_p x_{phdr} - \sum_{p\in\mathcal{P}} T_p x_{phdr'} - M(2 - y_{hdr} - y_{hdr'}) \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{17}$$

$$b_{hdrr'} \geq -\left(\sum_{p\in\mathcal{P}} T_p x_{phdr} - \sum_{p\in\mathcal{P}} T_p x_{phdr'}\right) - M(2 - y_{hdr} - y_{hdr'}) \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{18}$$

where M represents the maximum amount of imbalance between any pair of ORs in each hospital-day (B_{hd}). The actual value of the continuous variable $b_{hdrr'}$ can be computed when $y_{hdr} = y_{hdr'} = 1 \mid r < r'$. The order of binary variables in $\text{MIP}_{\text{AVPF1}}$ is $\mathcal{O}(|\mathcal{H}| \times |\mathcal{D}| \times |\mathcal{P}| \times |\mathcal{R}|)$.

The $\text{MIP}_{\text{AVPF}_2}$ reformulates $\text{MINLP}_{\text{AVPF}}$ without using the big-M parameter at the expense of adding a set of new five-indexed binary variables $v_{hdprr'} \in \{0,1\}$:

minimize $\text{MIQCP}_{\text{AVPF}}$ objective function $\text{MIP}_{\text{AVPF}_2}$

subject to $\text{MIQCP}_{\text{AVPF}}$ constraints without Constraints (11) and (12)

$$v_{hdprr'} \leq x_{hdpr} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P};\ (r,r') \in \mathcal{R}_h \mid r \neq r'$$

$$v_{hdprr'} \leq y_{hdr'} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P};\ (r,r') \in \mathcal{R}_h \mid r \neq r'$$

$$v_{hdprr'} \geq x_{hdpr} + y_{hdr'} - 1 \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P};\ (r,r') \in \mathcal{R}_h \mid r \neq r'$$

$$b_{hdrr'} \geq \sum_{p\in\mathcal{P}} T_p v_{hdprr'} - \sum_{p\in\mathcal{P}} T_p v_{hdpr'r} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P};\ r \in \mathcal{R}_h \mid r < r'$$

$$b_{hdrr'} \geq -\left(\sum_{p\in\mathcal{P}} T_p v_{hdprr'} - \sum_{p\in\mathcal{P}} T_p v_{hdpr'r}\right) \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P};\ r \in \mathcal{R}_h \mid r < r'$$

$$v_{hdprr'} \in \{0,1\} \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P};\ (r,r') \in \mathcal{R}_h \mid r \neq r'.$$

$\text{MIP}_{\text{AVPF}_2}$ is quadratically sensitive to the number of ORs within each hospital and has more binary variables than $\text{MIP}_{\text{AVPF1}}$ in $\mathcal{O}\left(|\mathcal{H}| \times |\mathcal{D}| \times |\mathcal{P}| \times |\mathcal{R}|^2 - |\mathcal{R}|\right)$ due to new binary variable $v_{hdprr'} \in \{0,1\}$.

4.2. Linearization scheme for quadratic penalty function

The big-M full linearization formulation, $\text{MIP}_{\text{AVPF1}}$, outperforms the formulation without big-M (see Section 9), and we therefore only examine linearization using big-M for the $\text{MIQCP}_{\text{QPF}}$ model. Using big-M eliminates the bilinearity of the $\text{MINLP}_{\text{QPF}}$, but the actual square root of the OR imbalance between any pair of open ORs is needed to compute the micro imbalance cost while using QPF. Note that we transfer the nonlinearity of the micro balancing objective function to constraints via continuous variable $b_{hdrr'}$, similar to $\text{MIP}_{\text{AVPF1}}$. Therefore, the $\text{MINLP}_{\text{QPF}}$ cannot be fully linearized, but its corresponding quadratic constraints become more tractable. The $\text{MIQCP}_{\text{QPF}}$ is constructed by removing Constraints (17) and (18) from $\text{MIP}_{\text{AVPF1}}$ and replacing them with Constraint (19):

minimize $\text{MIP}_{\text{AVPF1}}$ objective function $\text{MIQCP}_{\text{QPF}}$

subject to $\text{MIP}_{\text{AVPF1}}$ constraints without Constraint (17) and (18)

$$b_{hdrr'} \geq \left(\sum_{p\in\mathcal{P}} T_p x_{phdr} - \sum_{p\in\mathcal{P}} T_p x_{phdr'}\right)^2 - M(2 - y_{hdr} - y_{hdr'}) \quad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ (r,r') \in \mathcal{R}_h \mid r < r' \tag{19}$$

5. Decomposition methods

We develop uni-level and bi-level LBBD methods to optimally solve the BDORS problem. Like classical Benders decomposition [9], LBBD partitions the decision variables of the global problem into two vectors, x (primary or master variables) and y (secondary or slave variables). In general, we can view the problem as

$$\begin{aligned} \text{minimize} \quad & f(x,y) && \text{IP} \\ \text{subject to} \quad & (x,y) \in S \\ & x \in D_x, y \in D_y \end{aligned}$$

where f is a real-valued function, S is the feasible set, and D_x and D_y are the domains of x and y, respectively. Problem constraints are decomposed into two sets: (i) constraints whose scope exclusively include x variables (the MP variables), and (ii) constraints whose scope include both y and possibly x variables (SP variables). Since the MP only considers the x variables, the feasible set S is projected onto the x variables, resulting in set $\bar{S}_x$. Formally, the MP can be defined as follows:

$$\begin{aligned} \text{minimize} \quad & z && \text{MP} \\ \text{subject to} \quad & x \in \bar{S}_x \\ & z \geq \beta_{x^{(i)}}(x) && \forall i = 1,\ldots,|\mathcal{I}| \\ & x \in D_x \end{aligned}$$

where z is a real-valued decision variable and $\beta_{x^{(i)}}$ is a Benders cut on the objective function z found when fixing values of x to $x^{(i)}$, where $x^{(i)}$ is the ith MP incumbent. The SP for incumbent i given fixed values of x is as follows:

$$\begin{aligned} \text{minimize} \quad & f(x^{(i)},y) && \text{SP} \\ \text{subject to} \quad & (x^{(i)},y) \in S \\ & y \in D_y \end{aligned}$$

The SP solves for y variables given the trial MP values ($x^{(i)}$). Unlike classical Benders decomposition, there is no linear programming restriction on the form of the SP. LBBD consists of solving SPs for each MP incumbent solution until the MP and SP objective function values converge. The general process is as follows. The MP is solved to feasibility, producing solution $x^{(i)}$ with cost $z^{(i)}$ in iteration i. This solution is then used to formulate one or more SPs, which are each solved producing bounding constraints

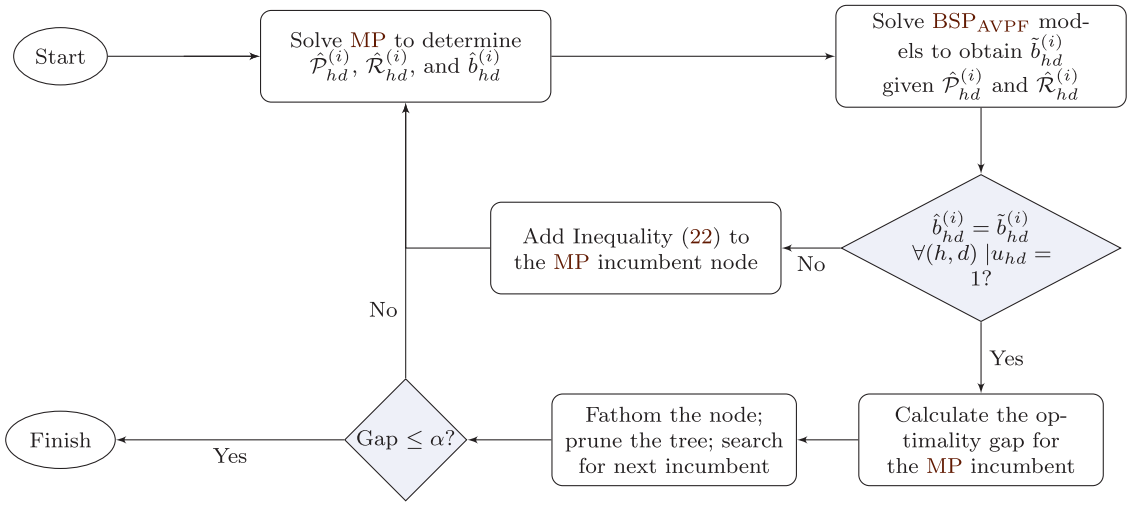


Fig. 1. Uni-level LBBD.

(i.e., Benders cuts) on the global objective function, z. If at iteration i, the MP optimal objective function value is equal to the SP objective function value, the process has converged to a globally optimal solution (i.e., $z^{(i)} = f(x^{(i)}, y^{(i)})$, where $y^{(i)}$ is the SP solution). Otherwise, the Benders cuts are added to the MP and the process repeats. It is noted that the logic-based cuts incorporated into the MP are similar to the ones generated in integer L-shaped methods—a specialized exact algorithm for solving large-scale stochastic integer programs (see, e.g., inequality 10 in [29]).

Unlike most LBBDs that solve SPs when the MP finds an optimum, our LBBD solves SPs whenever the MP finds an incumbent solution. This LBBD variant is called *LBBD cutting plane* or alternatively called branch-and-check (B&C) [5,11,46,47]. Therefore, $x^{(i)}$ and $z^{(i)}$ are interpreted as the B&C's ith incumbent solution and objective value found by the MP during the branch-and-cut search process, respectively. B&C converges to optimality when the MP optimum is verified against the relaxed MP constraints that are present in the SPs. Otherwise, Benders cuts are developed and incorporated into the MP to direct the master search towards optimality for future incumbents. For simplicity, we denote our cutting-plane LBBD as just LBBD throughout the paper.

Each Benders cut must satisfy two properties [14]:

1. It must cut off the current infeasible or sub-optimal solution of the MP or relaxed MP (RMP).
2. It must not cut off any other globally integer feasible solution of MP or relaxed MP (RMP).

6. LBBD approaches for $MIQCP_{AVPF}$ and MIP_{AVPF1}

We note that the uni-level LBBD uses only balancing optimality cuts (novel contribution), whereas the bi-level LBBD uses both packing feasibility and optimality cuts [41] in addition to the balancing optimality cuts (novel contribution). Similar to [13,16,23,40,41], we assume independence among hospital-days (or just days), meaning that the decision made in hospital h on day d does not impact the decisions of other hospital-days. We discuss how relaxing this assumption will structurally impact the LBBD approaches.

6.1. Uni-level LBBD

The uni-level LBBD method (Fig. 1) decomposes MIP_{AVPF1} and $MIQCP_{AVPF}$ into a location-allocation and macro balancing MP (micro balancing constraints are removed from both models) and micro balancing SPs (BSP_{AVPF} models) for the MP-determined open ORs in each hospital-day. We note that the MP solution is always feasible with respect to BSP_{AVPF} models; therefore, the purpose of BSP_{AVPF} models is to find optimal micro imbalance cost. Since BSP_{AVPF} models are solved for each hospital-day whose surgical suite is open ($u_{hd} = 1$), there are a maximum of $|\mathcal{H}| \times |\mathcal{D}|$ separable BSP_{AVPF} models, receiving independent output from the MP. If the cost of micro imbalance in the MP for incumbent i, $\hat{b}_{hd}^{(i)}$, is equal to that of the BSP_{AVPF} models, $\tilde{b}_{hd}^{(i)}$, BSP_{AVPF} is optimal; otherwise ($\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$), we must incorporate Benders balancing cuts (Inequality (22)), defined in Section 6.1.3, into the MP.

6.1.1. MP

The MP is a mixed-integer location-allocation and macro balancing optimization model. The MP is obtained by removing Constraints (11) and (12) from $MIQCP_{AVPF}$ and Constraints (15)–(18) from MIP_{AVPF1}. Removing these constraints from the MIP_{AVPF1} and $MIQCP_{AVPF}$ models lead to an identical MIP MP for the two models. Instead of using continuous variable $b_{hdrr'}$, which captures the individual cost of imbalance between any pair of open ORs in each hospital-day, the MP uses a two-indexed continuous variable b_{hd}, which captures the aggregate cost of imbalance among open ORs in each hospital-day. The MP is as follows:

$$\underset{\mathbf{u,y,x,w,z,b}}{\text{minimize}} \quad MIQCP_{AVPF}/MIP_{AVPF1} \text{ objective function} \qquad \text{MP}$$

$$\text{subject to} \quad \text{Constraints (1)} - \text{(10) and Constraint (13)}$$

$$b_{hd} \geq 0 \qquad \forall h \in \mathcal{H};\ d \in \mathcal{D}$$

The optimized MP output for each BSP_{AVPF} is the set of allocated patients ($\hat{\mathcal{P}}_{hd}^{(i)}$), the set of open ORs ($\hat{\mathcal{R}}_{hd}^{(i)}$), and a lower bound on the minimum micro imbalance cost ($\hat{b}_{hd}^{(i)}$).

6.1.2. Micro balancing SPs (BSPs)

The BSP_{AVPF} at incumbent i is a mixed-integer OR micro balancing optimization problem that re-allocates each patient in $\hat{\mathcal{P}}_{hd}^{(i)}$ via binary variable x_{pr} to one of the MP-determined open ORs, $\hat{\mathcal{R}}_{hd}^{(i)}$ ($|\hat{\mathcal{R}}_{hd}^{(i)}| \leq |\mathcal{R}_h|$), to minimize the actual cost of micro imbalance ($\tilde{b}_{hd}^{(i)}$) while respecting the capacity of each OR (B_{hd}). For each MP incumbent solution i, the MP output is fed into BSP_{AVPF}, which is as follows for hospital h on day d:

$$\underset{\mathbf{x,b}}{\text{minimize}} \quad \tilde{b}_{hd}^{(i)} = \sum_{r=1}^{|\hat{\mathcal{R}}_{hd}^{(i)}|-1} \sum_{r'=r+1}^{|\hat{\mathcal{R}}_{hd}^{(i)}|} b_{rr'} \qquad \text{BSP}_{\text{AVPF}}$$

$$\text{subject to} \quad \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{pr} = 1 \qquad \forall p \in \hat{\mathcal{P}}_{hd}^{(i)}$$

$$\sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr} \le B_{hd} \qquad \forall r \in \hat{\mathcal{R}}_{hd}^{(i)}$$

$$b_{rr'} \ge \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr} - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr'} \qquad \forall (r,r') \in \hat{\mathcal{R}}_{hd}^{(i)} \mid r < r' \qquad (20)$$

$$b_{rr'} \ge -\sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr} - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr'} \qquad \forall (r,r') \in \hat{\mathcal{R}}_{hd}^{(i)} \mid r < r' \qquad (21)$$

$$x_{pr} \in \{0,1\} \qquad \forall p \in \hat{\mathcal{P}}_{hd}^{(i)};\ r \in \hat{\mathcal{R}}_{hd}^{(i)}$$

$$b_{rr'} \ge 0 \qquad \forall (r,r') \in \hat{\mathcal{R}}_{hd}^{(i)} \mid r < r'$$

The possible micro imbalance cost difference in the MP and BSP_{AVPF} is caused by the additional Constraints (20) and (21) that exist in BSP_{AVPF}, but do not exist in the MP. Any objective function value mismatch between $\hat{b}_{hd}^{(i)}$ and $\tilde{b}_{hd}^{(i)}$ ($\hat{b}_{hd}^{(i)} < \tilde{b}_{hd}^{(i)}$) is handled by Benders balancing optimality cuts (Inequality (22)), defined in Section 6.1.3.

6.1.3. Benders cuts

Due to the always-feasible nature of BSP_{AVPF} models ($|\hat{\mathcal{R}}_{hd}^{(i)}| \le |\mathcal{R}_h|$), we only need Benders optimality cuts to connect the MP to the BSP_{AVPF} models. After the MP finds incumbent i, BSP_{AVPF} models are solved given the MP output ($\hat{\mathcal{R}}_{hd}^{(i)}$, $\hat{\mathcal{P}}_{hd}^{(i)}$, and $\hat{b}_{hd}^{(i)}$) for each open surgical suite ($u_{hd} = 1$). If the BSP_{AVPF} optimal micro balancing cost is equal to that of the MP ($\tilde{b}_{hd}^{(i)} = \hat{b}_{hd}^{(i)}$), that BSP_{AVPF} is optimal and no Benders cut is needed. Otherwise, the BSP_{AVPF} optimal cost given MP output must be communicated back to the MP for further refinements via the following valid Benders balancing cut (Theorem 1):

$$b_{hd} \ge \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{r\in\mathcal{R}_h\setminus\hat{\mathcal{R}}_{hd}^{(i)}} y_{hdr}}^{\text{OR increase}} + \overbrace{\sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} \left(1 - y_{hdr}\right)}^{\text{OR decrease}}}^{\text{Resource adjustment}} + \overbrace{\overbrace{\sum_{p\in\mathcal{P}\setminus\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{hdpr}}^{\text{Patient increase}} + \overbrace{\left(\left|\hat{\mathcal{P}}_{hd}^{(i)}\right| - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\mathcal{R}_h} x_{hdpr} \right)}^{\text{Patient decrease}}}^{\text{Load adjustment}} \right) \right) \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)} \qquad (22)$$

where $\bar{\mathcal{K}}_d^{(i)}$ is the set of BSP_{AVPF} models for incumbent i on day d where $\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$. Since this cut is derived when $\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$, the MP has to make a trade-off between accepting the balancing cost difference ($\tilde{b}_{hd}^{(i)} - \hat{b}_{hd}^{(i)}$) or taking the following remedial strategies: (i) open a new OR from the set $\mathcal{R}_h \setminus \hat{\mathcal{R}}_{hd}^{(i)}$; (ii) close an OR from $\hat{\mathcal{R}}_{hd}^{(i)}$; (iii) add patients from other hospital-days to the sub-optimal BSP_{AVPF} ($\mathcal{P} \setminus \hat{\mathcal{P}}_{hd}^{(i)}$); and/or (iv) remove patients from $\hat{\mathcal{P}}_{hd}^{(i)}$.

If the MP accepts the cost increment in BSP_{AVPF}, the previous MP allocation ($\hat{\mathcal{P}}_{hd}^{(i)}$, $\hat{\mathcal{R}}_{hd}^{(i)}$) remains unchanged; otherwise, the MP is forced to change the allocation of patients and/or ORs for each sub-optimal BSP_{AVPF}. Reducing the number of ORs for each BSP_{AVPF} may be a counterintuitive remedial strategy, but it is a feasible remedial strategy due to Constraint (4), which allows an OR to be open without allocating any patient to it. This linear combinatorial balancing optimality cut functions in lieu of the two nonlinear hard Constraints (11) and (12) in $\text{MIQCP}_{\text{AVPF}}$ and four linear Constraints (15)–(18) in $\text{MIP}_{\text{AVPF1}}$.

Theorem 1. *Inequality (22) is a valid Benders optimality cut.*

Proof. See proof in Appendix A.1. □

6.2. Bi-level LBBD

The bi-level LBBD approach (Fig. 2) decomposes the MP into a (i) location-allocation and relaxed macro balancing MP, which we call the relaxed MP (RMP), and (ii) multiple separable packing SPs (PSPs), one for each open surgical suite ($u_{hd} = 1$). The RMP relaxes the individual capacities of ORs in each hospital-day and only considers the aggregate OR capacity of each hospital-day ($|\mathcal{R}_h| \times B_{hd}$), which results in a lower bound on the number of ORs that should be opened in each hospital-day ($|\hat{\mathcal{R}}_{hd}^{(i)}|$) for the allocated set of patients ($\hat{\mathcal{P}}_{hd}^{(i)}$). PSPs determine the minimum number of ORs ($|\bar{\mathcal{R}}_{hd}^{(i)}|$) for $\hat{\mathcal{P}}_{hd}^{(i)}$. Given the status of each PSP (feasible or optimal), a Benders cut, as defined in Section 6.2.4, is added to the RMP. This give-and-take process between the RMP and PSPs is continued until convergence (first level optimality: $|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|, \forall h,d \mid u_{hd} = 1$). The procedure used for the first-level convergence of the bi-level LBBD resembles the LBBD_1 procedure with maximal implementation in [41], with the exception that their MP is replaced with our RMP that includes both macro and micro balancing. After first-level convergence, BSP_{AVPF} is solved for each open hospital-day with optimal PSP. The bi-level LBBD approach converges to global optimality when the optimality gap of the RMP incumbent i is less than α (optimality tolerance) and $\tilde{b}_{hd}^{(i)} = \hat{b}_{hd}^{(i)}, \forall h,d \mid u_{hd} = 1$. Any objective function value mismatch between $\tilde{b}_{hd}^{(i)}$ and $\hat{b}_{hd}^{(i)}$ is handled with our novel Benders balancing cuts.

6.2.1. Relaxed MP (RMP)

Similar to DORS MP [41], the RMP removes index r from binary variables y_{hdr} and x_{hdpr} from the $\text{MIQCP}_{\text{AVPF}}$ and the $\text{MIP}_{\text{AVPF1}}$ models:

$$\underset{\mathbf{u,y,x,w,z,b}}{\text{minimize}} \quad \text{Objective functions of MP} \qquad \text{RMP}$$

subject to Constraints (7)–(13) of DORS MP [41] and Constraints (9) and (10)

$$z_{hh'd} \ge 0 \qquad \forall (h,h') \in \mathcal{H} \mid h < h';\ d \in \mathcal{D}$$

$$b_{hd} \ge 0 \qquad \forall h \in \mathcal{H};\ d \in \mathcal{D}$$

$$y_{hd} \in \mathbb{Z}^+ \forall h \in \mathcal{H};\ d \in \mathcal{D} \qquad (23)$$

$$u_{hd}, x_{hdp} \in \{0,1\} \qquad \forall h \in \mathcal{H};\ d \in \mathcal{D};\ p \in \mathcal{P} \qquad (24)$$

$$w_p \in \{0,1\} \qquad \forall p \in \mathcal{P} \setminus \{\mathcal{P}'\} \qquad (25)$$

The RMP determines $\hat{\mathcal{P}}_{hd}^{(i)}$ and a lower bound on the minimum number of ORs to accommodate $\hat{\mathcal{P}}_{hd}^{(i)}$, $\hat{\mathcal{R}}_{hd}^{(i)}$, for each open surgical suite.

6.2.2. Packing SPs (PSPs)

The RMP output ($\hat{\mathcal{R}}_{hd}^{(i)}, \hat{\mathcal{P}}_{hd}^{(i)}$) is fed into packing sub-problems (PSP) (one for each open surgical suite). The RMP only assigns patients to hospital-days via binary variable x_{hdp} and relaxes patient-to-OR assignment dimension, leaving it to be determined by the

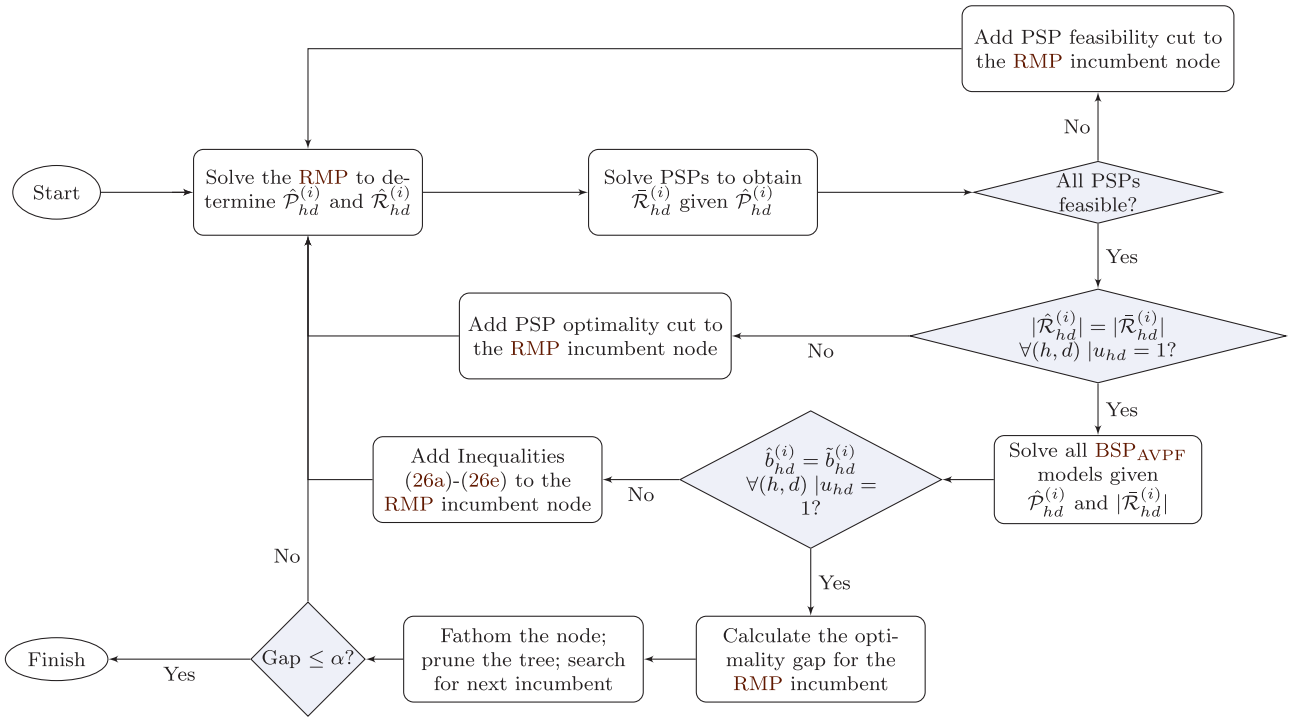


Fig. 2. Bi-level LBBD. PSP feasibility and optimality cuts correspond to the Inequality (19) and (20) of LBBD_1 in [41].

binary variable x_{pr} in PSP (see the PSP model in Section 4.2 in [41]). The packing SP represents a standard bin packing problem with symmetry-breaking constraints among ORs in each hospital-day that ensures each patient is assigned to exactly one OR and respects the availability of ORs in each hospital-day (B_{hd}), minimizing the number of ORs to open, $|\bar{\mathcal{R}}_{hd}^{(i)}|$.

6.2.3. Micro balancing SPs (BSPs)

Once the RMP and PSPs converge and the first-level optimality is ensured: $|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|$, the RMP solution is fed into BSP_{AVPF} to determine the micro imbalance cost for that hospital-day. Optimality is reached when the micro imbalance cost of the RMP optimum is equal to that of the BSP_{AVPF} for each hospital-day. Otherwise, Benders balancing cuts, as defined in Section 6.2.4, are derived and added to the RMP to guide the master search towards optimality.

6.2.4. Benders cuts

Any RMP incumbent solution may simultaneously yield of the following scenarios with respect to PSPs:

1. Infeasible PSP if $\hat{\mathcal{P}}_{hd}^{(i)}$ is not packable within $|\mathcal{R}_h|$: $|\hat{\mathcal{R}}_{hd}^{(i)}| > |\mathcal{R}_h|$
2. Super-optimal RMP and optimal PSPs: $|\hat{\mathcal{R}}_{hd}^{(i)}| < |\bar{\mathcal{R}}_{hd}^{(i)}|$
3. Optimal RMP and optimal PSP (first-level optimality): $|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|$
4. Sub-optimal RMP and optimal PSPs: $|\hat{\mathcal{R}}_{hd}^{(i)}| > |\bar{\mathcal{R}}_{hd}^{(i)}|$

The sets of SPs for Scenarios 1–4 at incumbent i of the LBBD are denoted $\bar{\mathcal{U}}_d^{(i)}$, $\bar{\mathcal{J}}_d^{(i)}$, $\bar{\mathcal{K}}_d^{(i)}$, and $\bar{\mathcal{G}}_d^{(i)}$, respectively.

We note that the LBBD developed for DORS [41] is a uni-level optimization with only MP and PSPs and no BSP_{AVPF} is considered. Therefore, to solve the second level of BDORS (BSP_{AVPF}), we ensure first-level optimality with respect to PSPs (Scenario 3: $|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|, \forall h,d \mid u_{hd}=1$) before solving BSP_{AVPF} models (second level), i.e., we solve all PSPs and no Benders packing feasibility and optimality cuts (Inequalities (19) and (20) in [41]) should exist for PSPs.

Balancing optimality cuts After the first-level optimality, the RMP incumbent solution i is by definition feasible to BSP_{AVPF} models for Scenarios 3–4; therefore, we only need balancing optimality cuts for BSP_{AVPF} models for which $\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$. Up to $|\mathcal{H}| \times |\mathcal{D}|$ BSP_{AVPF} models may be generated for each RMP incumbent depending on $|\mathcal{P}|$. We note that we use new auxiliary binary variables (q_{hd} and a_{hd}) in Benders balancing optimality cuts due to information loss stemming from the removal of index r in the bi-level LBBD approach. These auxiliary variables help reduce the number of binary variables in the model from $\mathcal{O}(|\mathcal{H}| \times |\mathcal{D}| \times |\mathcal{P}| \times |\mathcal{R}_h|)$ to $\mathcal{O}(|\mathcal{H}| \times |\mathcal{D}| \times |\mathcal{P}|)$ variables at the expense of adding $2\mathcal{O}(|\mathcal{H}| \times |\mathcal{D}| \times |\mathcal{I}|)$ new binary variables, where $|\mathcal{I}|$ is the number of potential incumbents that we anticipate the RMP will find. Based on LBBD convergence, we empirically show that the inclusion of auxiliary variables is a worthwhile strategy because branching on binary variables q_{hd} and a_{hd} leads to faster convergence in certain cases.

If $|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|$ (Scenario 3), then despite $|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|$ for hospital h on day d, we may have $\hat{b}_{hd}^{(i)} < \tilde{b}_{hd}^{(i)}$ for that hospital-day. The following valid Benders balancing cut (Theorem 2) is then added to the RMP:

$$b_{hd} \geq \tilde{b}_{hd}^{(i)}\left(1 - \left(M\left(q_{hd}^{(i)} + a_{hd}^{(i)}\right)\right)\right) \tag{26a}$$

$$\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) + \sum_{p \in \mathcal{P} \setminus \hat{\mathcal{P}}_{hd}^{(i)}} x_{hdp} \geq 1 - M\left(1 - q_{hd}^{(i)} + a_{hd}^{(i)}\right) \tag{26b}$$

$$y_{hd} - |\bar{\mathcal{R}}_{hd}^{(i)}| \geq 1 - M\left(1 + q_{hd}^{(i)} - a_{hd}^{(i)}\right) \tag{26c}$$

$$|\bar{\mathcal{R}}_{hd}^{(i)}| - y_{hd} \geq 1 - M\left(2 - q_{hd}^{(i)} - a_{hd}^{(i)}\right) \tag{26d}$$

$$a_{hd}^{(i)}, q_{hd}^{(i)} \in \{0,1\} \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)} \tag{26e}$$

Inequalities (26a)–(26e) communicate to the RMP the same set of remedial strategies as Inequality (22). The value of big M in Inequalities (26a), Inequality (26b), Inequality (26c), and Inequality (26d) is 1, 1, $|\bar{\mathcal{R}}_{hd}^{(i)}|+1$, and $|\mathcal{R}_h| - |\bar{\mathcal{R}}_{hd}^{(i)}| + 1$, respectively. Auxiliary variables $q_{hd}^{(i)}$ and $a_{hd}^{(i)}$ can enforce up to four remedial strategies, but only one active strategy is sufficient to force the RMP to generate a new solution.

Theorem 2. *Inequalities (26a)–(26e) are valid Benders optimality cuts.*

Proof. See proof in Appendix A.2. □

Scenario 4 ($|\hat{\mathcal{R}}_{hd}^{(i)}| > |\bar{\mathcal{R}}_{hd}^{(i)}|$) seems unlikely as $|\hat{\mathcal{R}}_{hd}^{(i)}|$ should not be greater than the PSP optimal solution ($|\bar{\mathcal{R}}_{hd}^{(i)}|$) because the RMP is a relaxation of the MP. This rare phenomenon may occur because the RMP may strike macro OR balancing among hospitals at the expense of opening more ORs for hospitals in which fewer ORs have been opened. Consequently, we may have some loosely utilized ORs with high micro imbalance cost. However, the RMP readjusts the number of ORs in each hospital-day after the following Benders balancing optimality cut is added:

$$b_{hd} \geq \tilde{b}_{hd}^{(i)}\left(1 - \left(M\left(q_{hd}^{(i)} + a_{hd}^{(i)}\right)\right)\right) \tag{27a}$$

$$\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) + \sum_{p \in \mathcal{P} \setminus \hat{\mathcal{P}}_{hd}^{(i)}} x_{hdp} \geq 1 - M\left(1 - q_{hd}^{(i)} + a_{hd}^{(i)}\right) \tag{27b}$$

$$y_{hd} \geq |\bar{\mathcal{R}}_{hd}^{(i)}| - M\left(2 - q_{hd}^{(i)} - a_{hd}^{(i)}\right) \tag{27c}$$

$$y_{hd} \leq |\bar{\mathcal{R}}_{hd}^{(i)}| + M\left(2 - q_{hd}^{(i)} - a_{hd}^{(i)}\right) \tag{27d}$$

$$a_{hd}^{(i)}, q_{hd}^{(i)} \in \{0, 1\} \qquad \forall (h, d, i) \in \bar{\mathcal{G}}_d^{(i)} \tag{27e}$$

The value of big M in Inequality (27a), Inequality (27b), Inequality (27c), and Inequality (27d) is 1, 1, $|\bar{\mathcal{R}}_{hd}^{(i)}|$, and $|\bar{\mathcal{R}}_{hd}^{(i)}|$, respectively. Unlike Inequalities (26a)–(26e), OR increase is not part of Inequalities (27a)–(27e) remedial strategies to the RMP because the RMP has already opened more than the required number of ORs to obtain macro caseload balancing among hospitals.

Theorem 3. *Inequalities (27a)–(27e) are a valid Benders cut.*

Proof. See proof in Appendix A.3. □

We note that the Benders cut for Scenario 4 was never used/generated throughout our experiments because we chose to solve BSP_{AVPF} models when all PSPs are optimal. We included this cut for the sake of completeness and possibly different implementation schemes by other researchers.

7. LBBD approaches for $\text{MIQCP}_{\text{QPF}}$

The only change required for the uni- and bi-level LBBD approaches to solve $\text{MIQCP}_{\text{QPF}}$ with quadratic micro imbalance SP (BSP_{QPF}) is to remove Constraints (20) and (21) from BSP_{AVPF} and replace them with Constraint (28):

$$b_{rr'} \geq \left(\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr} - \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr'}\right)^2 \qquad \forall (r, r') \in \hat{\mathcal{R}}_{hd}^{(i)} \mid r < r' \tag{28}$$

8. Implementation

We employ a maximal cut generation approach similar to [41] to execute both the uni- and bi-level LBBD approaches. The rationale behind maximal cut generation implementation for both LBBD methods is that we aim to converge to first-level optimality as quickly as possible, needing all the feedback (packing feasibility and optimality cuts) from PSPs. Unlike the first level of bi-level LBBD where PSPs can be both infeasible and sub-optimal, BSP_{AVPF} models in the second level are by definition feasible and their optimal solutions can help construct a new stronger bound for decomposition approaches.

At any MP and RMP integer (incumbent) solution, we use lazy callbacks to ensure global incumbency. MP and RMP integer solutions are considered global if $\tilde{b}_{hd}^{(i)} = \hat{b}_{hd}^{(i)}, \forall h, d \mid u_{hd} = 1$, i.e., no Benders cut exists for any SP. This callback approach to LBBD implementation is suitable for LBBD approaches with only feasibility/satisfaction SPs, in which the sole purpose is to accept or reject the MP or RMP incumbent solution. In this case, a combinatorial Benders cut is developed for each infeasible SP and the search is continued for next incumbent. This callback approach is unsuitable for LBBD implementations where SPs are by definition feasible and the primary purpose of solving SPs is to find optimal integer solutions. If optimal solutions of all SPs exist, but differ from those of the MP or RMP ($\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$), a new global upper bound can be constructed for the problem. However, Gurobi rejects such MP or RMP integer solutions because a Benders balancing cut is required when $\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$.

Given first-level optimality ($|\hat{\mathcal{R}}_{hd}^{(i)}| = |\bar{\mathcal{R}}_{hd}^{(i)}|$), the following formula can be used to obtain an optimality gap for integer solutions that Gurobi rejects when $\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$:

$$\text{Optimality gap} = \frac{\left(z^{\text{MP}} - z^{\hat{b}_{hd}^{(i)}} + z^{\tilde{b}_{hd}^{(i)}}\right) - z^{\text{LP}}}{\left(z^{\text{MP}} - z^{\hat{b}_{hd}^{(i)}} + z^{\tilde{b}_{hd}^{(i)}}\right)}$$

where z^{LP} is the objective function value of the MP or RMP linear programming relaxation, z^{MP} is the MP or RMP incumbent objective function value, $z^{\hat{b}_{hd}^{(i)}}$ is the MP or RMP micro imbalance cost, and $z^{\tilde{b}_{hd}^{(i)}}$ is the BSP_{AVPF} micro imbalance cost [40]. We note that we save all rejected/accepted incumbents of the bi-level LBBD approach that yielded $\tilde{b}_{hd}^{(i)} > \hat{b}_{hd}^{(i)}$ throughout Gurobi branch-and-cut search, compute the optimality gap for each, and finally choose the incumbent with the lowest optimality gap among existing rejected incumbents as the best incumbent. Gurobi does not automatically apply this strategy, so a new callback is needed to communicate this repaired upper bound to the bi-level LBBD approach. At any MP and RMP integer (incumbent) solution, we use lazy callbacks to ensure global incumbency. MP and RMP integer solutions are considered global if $\tilde{b}_{hd}^{(i)} = \hat{b}_{hd}^{(i)}, \forall h, d \mid u_{hd} = 1$, i.e., no Benders cut exists for any SP. The number of auxiliary variables in the RMP, q_{hd} and a_{hd}, that can be used throughout the optimization must be determined at the beginning of the optimization because callbacks do not allow new variables to be added to the MP/RMP on the fly. Thus, the possible number of auxiliary variables was fixed to 1000 for all instances; however, the maximum number of auxiliary variables was 173. We also implemented a mechanism to raise a flag if the number of required auxiliary variables exceeds 1000, in which case the instance has to be re-solved with a larger number of auxiliary variables.

9. Results

The test cases were run in Python [42] v3.5 with Gurobi Optimizer (Gurobi, Inc.) v6.5 on a 2.4 GHz Intel Core i5 processor with

Table 2
Average optimality gap over 40 trials of each dataset. First superscript: number of unsolved trials; second superscript: number of best solutions; bold: best performance.

Absolute value penalty			
	Three ORs	Five ORs	Average
$MIQCP_{AVPF}$	–	–	–
MIP_{AVPF1}	$0.61\%^{(0,30)}$	$1.44\%^{(0,7)}$	$\mathbf{1.03}\%^{(0,19)}$
MIP_{AVPF_2}	$0.90\%^{(0,9)}$	$2.11\%^{(0,3)}$	$1.50\%^{(0,6)}$
Uni-level LBBD	$2.18\%^{(0,3)}$	$1.91\%^{(0,20)}$	$2.05\%^{(0,12)}$
Bi-level LBBD	$1.60\%^{(1,2)}$	$2.26\%^{(1,10)}$	$1.93\%^{(1,6)}$
Quadratic penalty			
$MIQCP_{QPF}$	–	–	–
Uni-level LBBD	$11.30\%^{(0,25)}$	–	–
Bi-level LBBD	$\mathbf{2.34}\%^{(0,16)}$	–	–

4 GB RAM. We use two UHN datasets [41] with randomly generated surgical times, each with three hospitals, 20–160 patients, and a one-week planning horizon, where there are five days per week. One dataset has three ORs per hospital (yielding easy SPs but a hard MP) and the other has five ORs (yielding hard SPs but an easy MP). Five trials with different random surgical times are generated for each test case using the data generation method in [41]. BDORS is a weighted multi-objective optimization problem with all terms in the objective function are cost related. The values of the first four terms in the objective function are directly obtained from DORS [41]. The value of macro imbalance cost (C_1) is \$1000 and the value of micro imbalance cost (C_2) among opened ORs in each open surgical suite is \$5 per minute. We have empirically tuned these parameters such that a trade-off between opening cost-effective surgical suite/ORs and balancing among hospitals/ ORs is found, though these values can be calibrated in accordance with the decision-maker's preference. However, when conflicting objectives are considered, multi-objective approaches similar to [31,53] can be adopted. We further discuss this issue in Table 7. We ensure a fair comparison among LBBDs and mathematical models by considering a fixed time limit of 1800 s (30-minute). This time limit allows the small instances of BDORS to be solved to near optimality and the larger instances to $\approx$1–2% optimality gap. There is also a real-world utility to being able to quickly resolve the problem as new, potentially more urgent, patients arrive to the system or existing patients cancel or reschedule. Therefore, we posit 30-minute runtime allows both MIPs and LBBDs to find a balance between solution quality and computation time.

Gurobi did not find any integer feasible solutions for the original nonlinear ($MINLP_{AVPF}$ and $MINLP_{QPF}$) or partially linearized ($MIQCP_{AVPF}$ and $MIQCP_{QPF}$) models (Table 2). Otherwise, with absolute value penalties using pure Gurobi, the performance of MIP_{AVPF1} (big-M formulation) is superior to that of MIP_{AVPF_2} (no big-M formulation) in terms of both obtaining lower optimality gaps and better integer solutions on both instances of three and five ORs per hospital-day. The pure Gurobi (MIP_{AVPF1} and MIP_{AVPF_2}) and the LBBD approaches obtained similar optimality gaps for both datasets (range: [0.61, 2.26]), with Gurobi obtaining the lowest optimality gaps and the bi-level LBBD failing to solve one instance in each dataset. Since MIP_{AVPF1} had the lowest average optimality gap and found the best solution of the three approaches most frequently, we deem it the best approach for absolute value penalty formulations in general. However, despite larger optimality gaps of the uni-and bi-level LBBD approaches on five-OR dataset, they both outperform MIP_{AVPF1} and MIP_{AVPF_2} in terms of the number of best integer solutions found. For quadratic penalties, while no approaches could solve the easy MP/hard SP (five ORs per hospital) dataset, the bi-level LBBD significantly outperformed the uni-level LBBD with an optimality gap of 2.34%, similar performance to the absolute value formulation, although the uni-level LBBD found better solutions in more trials.

9.1. Absolute value penalty function

We compare computational times and optimality gap of the LBBD approaches against $MIQCP_{AVPF}$ and MIP_{AVPF1} solved via Gurobi.

9.1.1. $MIQCP_{AVPF}$ versus MIP_{AVPF1} and MIP_{AVPF_2}

$MIQCP_{AVPF}$ did not solve any instances, whereas MIP_{AVPF1} and MIP_{AVPF1} solved all problem instances to $\approx$1% (range [0.61%, 1.44%]) and $\approx$2% (range [0.90%, 2.11%]) optimality gaps for hard MP/easy SPs and easy MP/hard SPs dataset, respectively (Table 3). The average minimum (G_{min}), the average maximum (G_{max}), and the grand average (G_{mean}) of optimality gaps of MIP_{AVPF1} on any instance of five ORs per hospital are all higher than those of three ORs per hospital, demonstrating a small impact of increased MIP size dimensionality on MIP_{AVPF1}'s optimality gap. The average minimum (G_{min}), the average maximum (G_{max}), and the grand average (G_{mean}) of optimality gaps of MIP_{AVPF_2} on instances of both datasets are at least 50% higher than those of MIP_{AVPF1}, further illustrating the performance improvement obtained by using a big-M formulation.

9.1.2. Uni-level LBBD versus bi-level LBBD

For the easy MP/hard SPs (five ORs per hospital) dataset, the uni-level LBBD method outperformed the bi-level LBBD method in terms of optimality gap (1.91% average optimality gap compared to 2.26%) (Table 4).

For the hard MP/easy SPs dataset, the bi-level LBBD outperformed the uni-level LBBD in terms of optimality gap in all but one instance. We examine the CPU breakdown of the uni- and bi-level LBBDs with the absolute value penalty function using the largest instance of each dataset (160 patients, five trials each). We chose exclusively the absolute value penalty function for convergence analysis, because both LBBDs found verifiable integer solutions for all trials of the largest instance of each dataset. There exists a significant number of iterations between MP and BSP_{AVPF} (average 205) and RMP, PSP, and BSP_{AVPF} (average 188) on the hard MP/easy SP dataset (Table 5). The MP average computational time in the uni-level LBBD method (1.49 s) is slightly higher than that of the RMP (0.81 s) in the bi-level LBBD method. Conversely, the BSP_{AVPF} average computational times in the uni-level LBBD (7.32 s) is slightly lower than that of the bi-level LBBD (8.41 s). The bi-level's PSP and BSP_{AVPF} average computational times are nearly identical at 8.41 and 8.34 s, respectively. Interestingly, the bi-level LBBD's visits to the BSP_{AVPF} models are not uniformly spread throughout the iterations; they generally come in clusters (Fig. 3).

The easy MP/hard SP dataset results in fewer iterations for both methods, though the bi-level's RMP average computational time is not impacted by the increase in the number of ORs per hospital-day due to the removal of index r. On the other hand, the uni-level's MP average computational time increases from 1.49 to 9.16 s as SP difficulty increases. Furthermore, the easy MP/hard SP dataset results in at least one order of magnitude higher BSP_{AVPF} computational time (7.32–130.27 s and 8.34–120.89 s for the uni- and bi-level LBBD methods, respectively), but it does not significantly impact the PSP computational difficulty (8.41–9.35 s). This increase in BSP_{AVPF} computation time is the primary cause for the fewer iterations in this dataset for both methods (Fig. 4). The performance prediction of BDORS on larger instances of the problem can be found in Appendix A.4.

Table 3
Average computation times and optimality gap of MIP_{AVPF1} and MIP_{AVPF_2} on the dataset with three ORs per hospital; bold: best performance.

	Three ORs											
	MIP_{AVPF1} (big-M)						MIP_{AVPF_2} (no big-M)					
	Optimality gap (%)			Time (minutes)			Optimality gap (%)			Time (minutes)		
$\|\mathcal{P}\|$	G_{min}	G_{max}	G_{mean}	T_{min}	T_{max}	T_{mean}	G_{min}	G_{max}	G_{mean}	T_{min}	T_{max}	T_{mean}
20	**0.00**	**0.00**	**0.00**	**0.1**	**1.9**	**0.60**	**0.00**	**0.00**	**0.00**	0.3	5.6	2.2
40	**0.01**	**0.37**	**0.20**	30.0	30.0	30.0	0.04	0.63	0.22	30.0	30.0	30.0
60	**0.44**	**0.74**	**0.60**	30.0	30.0	30.0	0.48	0.91	0.72	30.0	30.0	30.0
80	**0.39**	**0.88**	**0.59**	30.0	30.0	30.0	0.58	1.37	0.95	30.0	30.0	30.0
100	**0.59**	**1.20**	**0.81**	30.0	30.0	30.0	0.99	1.99	1.39	30.0	30.0	30.0
120	**0.67**	**1.09**	**0.77**	30.0	30.0	30.0	0.99	1.87	1.43	30.0	30.0	30.0
140	0.78	**1.31**	**1.00**	30.0	30.0	30.0	**0.53**	1.78	1.08	30.0	30.0	30.0
160	**0.67**	**1.02**	**0.91**	30.0	30.0	30.0	1.2	1.97	1.43	30.0	30.0	30.0
Mean	**0.44**	**0.83**	**0.61**	**26.3**	**26.5**	**26.3**	0.60	1.32	0.90	30.0	27.9	26.5
	Five ORs											
20	**0.00**	**0.11**	**0.05**	0.7	30.0	30.0	**0.00**	0.20	0.09	16.0	30.0	30.0
40	**0.02**	**3.84**	0.94	30.0	30.0	30.0	0.04	3.92	**0.90**	30.0	30.0	30.0
60	**0.13**	**1.95**	**0.68**	30.0	30.0	30.0	0.78	2.37	1.52	30.0	30.0	30.0
80	**0.78**	**1.90**	**1.27**	30.0	30.0	30.0	1.39	3.31	2.39	30.0	30.0	30.0
100	**1.90**	**2.32**	**2.12**	30.0	30.0	30.0	2.53	3.70	2.91	30.0	30.0	30.0
120	**1.85**	**2.64**	**2.30**	30.0	30.0	30.0	2.40	3.60	2.98	30.0	30.0	30.0
140	**1.33**	**2.72**	**2.16**	30.0	30.0	30.0	2.16	3.88	2.99	30.0	30.0	30.0
160	**1.59**	**2.43**	**2.02**	30.0	30.0	30.0	2.46	3.72	3.07	30.0	30.0	30.0
Mean	**0.97**	**2.24**	**1.44**	**26.3**	30.0	**29.3**	1.47	3.09	2.11	28.3	30.0	30.0

Table 4
Average computation times and optimality gaps of LBBDs with absolute value penalties. Superscript: number of unsolved trials; bold: best performance.

		Uni-level LBBD						Bi-level LBBD					
		Optimality gap (%)			Time (minutes)			Optimality gap (%)			Time (minutes)		
	$\|\mathcal{P}\|$	G_{min}	G_{max}	G_{mean}	T_{min}	T_{max}	T_{mean}	G_{min}	G_{max}	G_{mean}	T_{min}	T_{max}	T_{mean}
3 ORs	20	**0.00**	**0.00**	**0.00**	**4.1**	**12.6**	**8.0**	$< 10^{-4}$	1.81	0.60	30.0	30.0	30.0
	40	0.83	3.55	2.00	30.0	30.0	30.0	**0.41**	**1.52**	**0.90**	30.0	30.0	30.0
	60	1.48	4.26	2.87	30.0	30.0	30.0	**0.97**	**1.52**	**1.29**	30.0	30.0	30.0
	80	2.79	4.15	3.38	30.0	30.0	30.0	**0.68**	**1.73**	**1.37**	30.0	30.0	30.0
	100	2.25	6.59	3.97	30.0	30.0	30.0	**1.02**	**2.04**	**1.63**	30.0	30.0	30.0
	120	1.70	3.73	2.94	30.0	30.0	30.0	**1.33**	**3.42**	**2.35**$^{(1)}$	30.0	30.0	30.0
	140	2.30	4.31	2.89	30.0	30.0	30.0	**1.46**	**2.22**	**1.79**	30.0	30.0	30.0
	160	2.32	3.27	2.72	30.0	30.0	30.0	**1.79**	4.09	**2.54**	30.0	30.0	30.0
	Mean	1.71	3.73	2.18	**22.9**	**27.8**	**27.3**	**0.96**	**2.29**	**1.60**$^{(1)}$	30.0	30.0	30.0
5 ORs	20	**0.00**	**0.02**	**0.00**	**0.8**	30.0	**8.7**	$< 10^{-4}$	0.73	0.20	30.0	30.0	30.0
	40	**0.00**	**3.14**	**0.78**	**2.4**	30.0	**19.4**	0.57	3.80	1.90	30.0	30.0	30.0
	60	**0.02**	6.36	**1.42**	30.0	30.0	30.0	1.16	**5.96**	2.32	30.0	30.0	30.0
	80	**0.16**	8.70	**2.69**	30.0	30.0	30.0	0.61	**3.36**	**1.80**	30.0	30.0	30.0
	100	1.28	**3.75**	**2.37**	30.0	30.0	30.0	**0.85**	**2.87**	**1.94**	30.0	30.0	30.0
	120	**1.15**	**5.00**	**2.41**	30.0	30.0	30.0	1.58	11.66	4.40$^{(1)}$	30.0	30.0	30.0
	140	**1.74**	7.68	3.33	30.0	30.0	30.0	1.87	**3.08**	**2.34**	30.0	30.0	30.0
	160	1.39	**3.16**	**2.23**	30.0	30.0	30.0	**1.24**	8.78	3.16	30.0	30.0	30.0
	Mean	**0.72**	**4.73**	**1.91**	**22.9**	30.0	**26.0**	0.99	5.03	2.26$^{(1)}$	30.0	30.0	30.0

Table 5
Average CPU time (seconds) breakdown for LBBDs with absolute value penalties, where one MP/RMP visit is one iteration. The first number under bi-level LBBD BSP_{AVPF} time is the number of times we go from BSP_{AVPF} models to RMP.

		Uni-level LBBD			Bi-level LBBD			
	$\|\mathcal{P}\|$	Iterations	MP time	BSP_{AVPF} time	Iterations	RMP time	PSP time	BSP_{AVPF} time (#, s)
3 ORs	160-1	210	1.32	7.28	190	0.70	8.39	[9, 8.31]
	160-2	194	1.96	7.30	189	0.80	8.31	[11, 8.01]
	160-3	199	1.90	7.20	178	1.04	8.64	[9, 9.24]
	160-4	208	1.34	7.33	183	0.79	8.44	[15, 8.12]
	160-5	215	0.93	7.48	198	0.74	8.27	[4, 8.03]
	Average	205	1.49	7.32	188	0.81	8.41	[10,8.34]
5 ORs	160-1	15	19.27	110.86	87	0.79	8.41	[7, 144.16]
	160-2	13	8.45	130.01	169	0.87	8.45	[4, 58.30]
	160-3	14	4.41	124.15	45	0.63	7.07	[6, 243.49]
	160-4	9	4.05	195.95	96	0.57	14.79	[3, 113.59]
	160-5	18	9.64	90.36	122	0.67	8.01	[10, 74.90]
	Average	14	9.16	130.27	104	0.71	9.35	[6, 120.89]

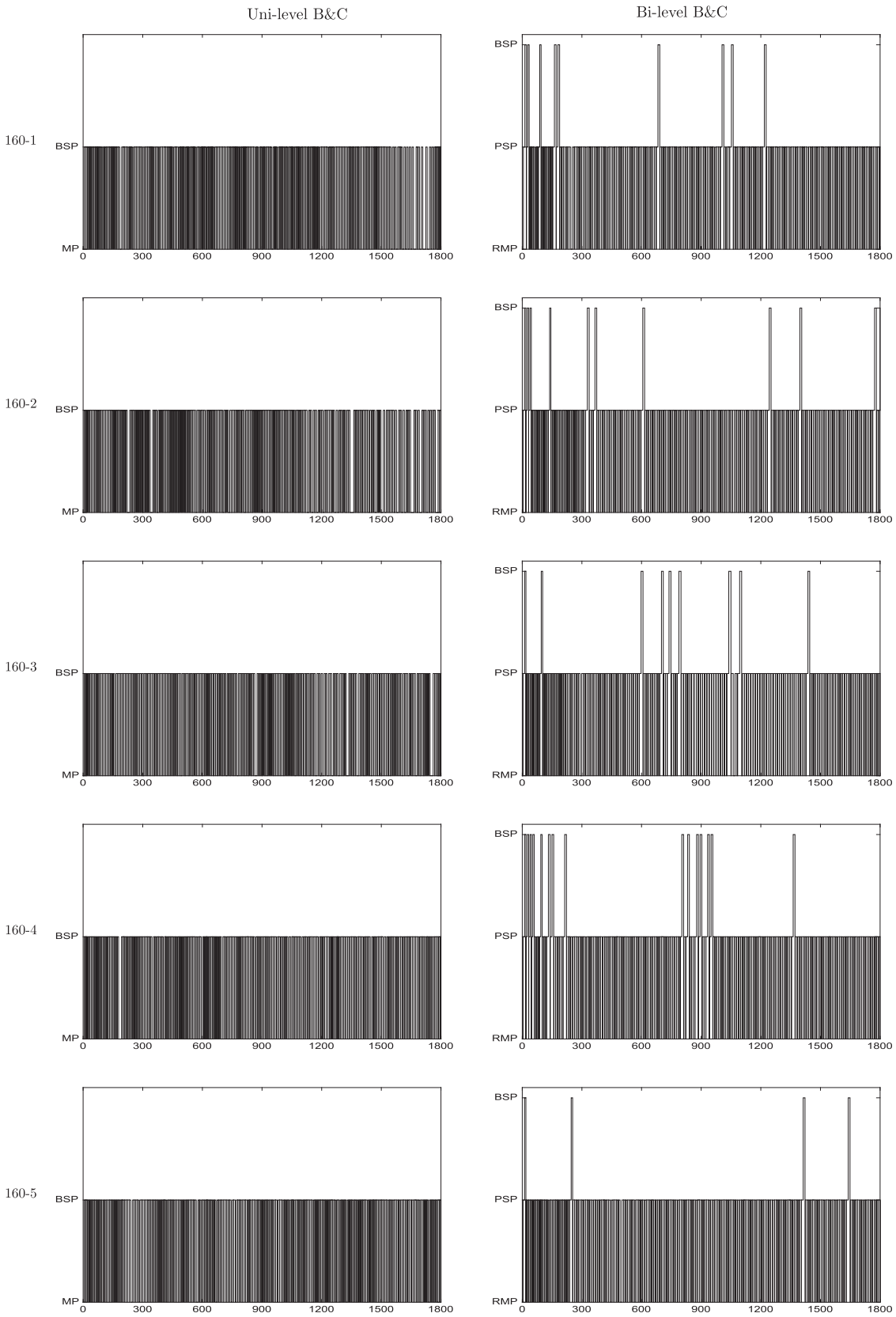


Fig. 3. MP/RMP and SP traversal per CPU time (seconds) for the easy MP/hard SP dataset with absolute value penalties.

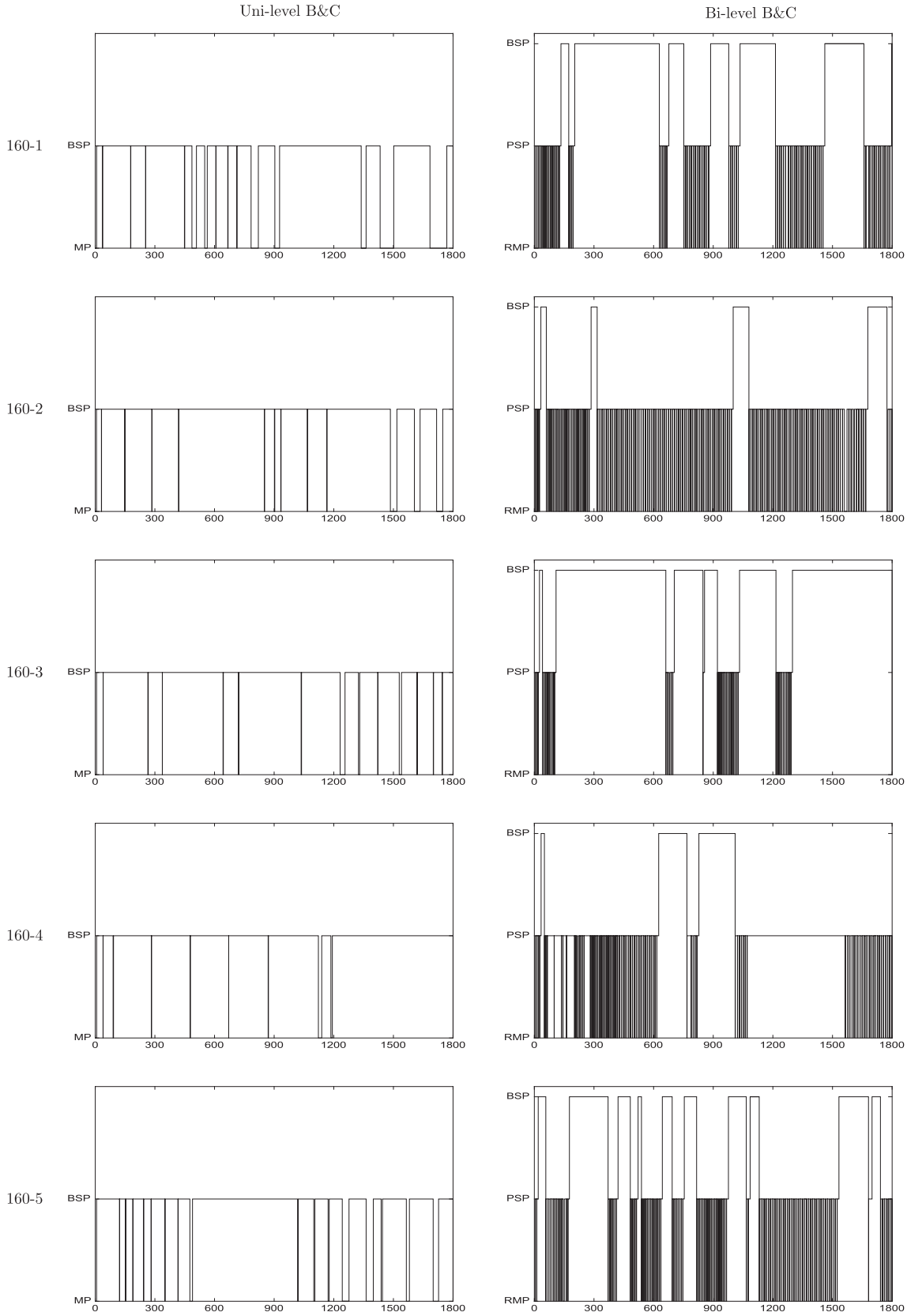


Fig. 4. MP/RMP and SP traversal per CPU time (seconds) for the easy MP/hard SP dataset with absolute value penalties.

Table 6
Average optimality gap of LBBDs with quadratic penalties. Bold: best performance.

	Uni-level LBBD						Bi-level LBBD					
	Optimality gap (%)			Time (minutes)			Optimality gap (%)			Time (minutes)		
$\|\mathcal{P}\|$	G_{min}	G_{max}	G_{mean}	T_{min}	T_{max}	T_{mean}	G_{min}	G_{max}	G_{mean}	T_{min}	T_{max}	T_{mean}
20	**0.00**	100.00	29.00	**2.0**	**30.0**	**24.4**	0.29	**1.50**	**0.87**	30.0	30.0	30.0
40	**0.00**	34.79	8.23	4.3	30.0	24.9	**0.00**	**2.72**	**1.01**	**3.6**	**30.0**	**30.0**
60	**0.50**	100.00	40.51	30.0	30.0	30.0	1.67	**12.68**	**4.2**	30.0	30.0	24.2
80	**0.57**	35.87	8.40	30.0	30.0	30.0	1.26	**5.65**	**2.66**	30.0	30.0	30.0
100	**0.67**	**1.71**	**1.15**	30.0	30.0	30.0	1.66	3.74	2.47	30.0	30.0	30.0
120	**0.85**	**1.18**	**0.99**	30.0	30.0	30.0	1.33	5.83	3.81	30.0	30.0	30.0
140	**0.83**	**1.68**	**1.21**	30.0	30.0	30.0	1.51	2.08	1.69	30.0	30.0	30.0
160	0.72	**1.16**	**0.92**	30.0	30.0	30.0	**0.20**	5.42	1.94	30.0	30.0	30.0
Mean	**0.52**	34.55	11.30	23.3	30.0	28.7	0.99	**4.95**	**2.34**	26.7	30.0	29.28

Table 7
Impact of considering macro and micro balancing on the DORS [41]/BDORS final solution. Instance of 80 patients with three and five ORs have been chosen for this purpose; bold: more balanced solution with respect to macro (for open surgical suites) and micro (at least two open ORs per hospital-day); $|\bar{\mathcal{P}}_r|$: average number of patients per OR; b_{hd}: total surgical load difference among open ORs within each hospital-day.

Instance						DORS									
						Open ORs ($\|\mathcal{R}_{hd}\|$)					Total daily micro imbalance (b_{hd})				
	$\|\mathcal{P}\|$	$\|\mathcal{R}\|$	$\|\bar{\mathcal{P}}_r\|$	Total weekly b_{hd}		d_1	d_2	d_3	d_4	d_5	d_1	d_2	d_3	d_4	d_5
	80	21	3.33	432	h_1	3	3	0	0	0	**14**	26	0	0	0
					h_2	0	3	3	0	0	0	114	36	0	0
					h_3	0	0	3	3	3	0	0	6	154	82
3 ORs						BDORS									
	80	18	3.44	**104**	h_1	3	**3**	**3**	0	0	18	**24**	16	0	0
					h_2	0	**3**	**3**	0	0	0	**16**	**28**	0	0
					h_3	0	**3**	**3**	0	0	0	0	**2**	0	0
						DORS									
						Open ORs ($\|\mathcal{R}_{hd}\|$)					Total micro imbalance				
	80	24	3.33	684	h_1	5	5	0	0	0	**28**	96	0	0	0
					h_2	0	0	0	0	0	0	0	0	0	0
					h_3	0	0	5	5	4	0	0	102	164	294
5 ORs						BDORS									
	80	24	3.33	**240**	h_1	5	**5**	0	0	0	70	**34**	0	0	0
					h_2	0	**4**	**5**	0	0	0	38	58	0	0
					h_3	0	0	**5**	0	0	0	0	**40**	0	0

9.2. Quadratic penalty function

The $MIQCP_{QPF}$ solved via Gurobi did not find any integer solutions, while both the uni- and bi-level LBBD approaches solved all instances of hard MP/easy SP datasets to 11.30% and 2.34% average optimality gap, respectively (Table 6), they could not solve any instances of easy MP/hard SPs. The interesting pattern is that the uni-level LBBD produces very high optimality gaps for some of the small instances up to 80 patients, but produces lower optimality gaps as the size of problem exceeds 100 patients (Table 6). A possible reason for this phenomenon is the large micro imbalance cost in the small instances of BSP_{QPFs} and a hard MP that is not capable of finding an alternative solution that can be verified by the SPs. As the size of the problem increases, the surgical load assigned to each open OR increases and the cost of micro imbalance decreases. The reason for the unsolvability of LBBD approaches on easy MP/hard SP datasets is the existence of hard BSP_{QPF} (most instances found integer solutions, but lower bound stayed at 100% optimality gap), preventing LBBDs from finding optimal solutions that are needed for Benders balancing cut development.

9.3. The impact of macro and micro balancing on DORS

We investigate the impact of macro and micro balancing on the final solution of BDORS (Table 7). When considering the instance of three ORs per hospital-day, the solution of BDORS without balancing decisions opens the most cost-effective ORs in different days of a week, whereas BDORS opens more ORs within each day in different hospitals to strike macro balancing among hospitals. Additionally, the cost of micro imbalance in BDORS is much lower (aggregate 104 minutes) than that of non-balanced DORS (432 minutes). When considering five ORs, the number of opened ORs is the same, but BDORS produces a better solution than DORS in terms of macro and micro balancing. The solutions of both instances of three and five ORs per hospital-day in BDORS are more balanced in terms of macro and macro balancing than that of the DORS. For example for the case of five ORs per hospital-day, DORS chooses to open five and four ORs on d_4 and d_5, respectively, while BDORS closes those ORs on d_4 and d_5 and opens them on d_2 and d_3, leading to a reduction in the macro imbalance cost on day d_2 and d_3 and no macro imbalance cost on d_4 and d_5. Due to vary-

ing operating room time availability in different hospital-days (B_{hd}) and for solutional representation simplicity, we choose to provide only the total amount of macro imbalance in each hospital-day (b_{hd}).

10. Discussion

We developed a variety of linearization techniques to solve BDORS MINLP models. We chose a linear penalty function for the macro imbalance cost, but both linear and quadratic penalty functions for micro imbalance cost. The quadratic macro imbalance cost (or very high macro imbalance cost in the linear case) causes the model to open unnecessary ORs to strike macro balancing among hospitals, stifling the opportunities that can be created due to more cost-effective surgical suites and ORs openings in the network. It is more realistic in practice to use linear penalty functions than quadratic as the latter leads to difficult optimization problems that cannot be solved within reasonable clinical decision-making timeframes.

We proposed two linearization schemes to make the BDORS MINLP structure amenable to MIP solvers. The first linearization led to an MIQCP which was unsolvable via Gurobi due to the existence of quadratic constraints. $\text{MIP}_{\text{AVPF1}}$ performed well compared to the $\text{MIQCP}_{\text{AVPF}}$ in spite of having a higher number of linear constraints, demonstrating that quadratic constraints should be avoided when possible. We also developed a linearization scheme for the $\text{MIQCP}_{\text{QPF}}$ which removed the inherent bilinear structure of the MINLP model, but still remained nonlinear and unsolvable via Gurobi due to the quadratic penalty function. Gurobi's drastic performance difference on these two models highlights the significance of modelling structure on the solvability of BDORS problem.

We developed a uni-level and a bi-level LBBD method capable of solving BDORS MINLP with both absolute value and quadratic imbalance penalties. These LBBD approaches can be applied to other MINLPs with similar structures. The uni- and bi-level LBBD worked well for hard MP/easy SPs dataset with quadratic penalty function, but failed to solve instances of easy MP/hard SPs, due to the increase of three to 10 quadratic constraints. Overall, if the absolute value penalty function is to be used, $\text{MIP}_{\text{AVPF1}}$ solved via Gurobi is a better choice; however, both the uni-and bi-level LBBDs are viable alternative choices. If the quadratic penalty function is to be used, both the uni-and bi-level LBBDs are better choices than $\text{MIQCP}_{\text{QPF}}$, which did not find any integer solutions for BDORS instances. Additionally, the bi-level LBBD approach is a better choice than the uni-level LBBD to solve BDORS instances based on the optimality gap, but uni-level LBBD is a better choice in terms of quality of integer solutions.

Similar to [13,16,23,40,41], we assumed that the decisions made in one hospital-day do not impact the decisions in other hospital-days. Relaxing this assumption has clinical and methodological implications. The clinical implications have been addressed in [51], but the methodological aspect has not been discussed. Relaxing this assumption and incorporating downstream units of ORs links the OR scheduling decisions of one hospital to those of the same hospital in other days. Since each patient must spend all his/her required time in downstream units in the hospital in which the operation was performed, SP structure changes from a daily decision to a weekly or monthly decision, requiring more patients to be considered in the optimization. As we showed, the bi-level LBBD approach may outperform MIP solved via Gurobi after 450 patients; its smaller size also makes it more likely to fit in the RAM of a desktop computer.

11. Conclusion and future works

We proposed the BDORS problem and presented two novel mixed-integer nonlinear formulations for which we developed two partially linearized models (one for the absolute value penalty function and one for the quadratic penalty function) and two fully linearized model for the absolute value penalty function. We additionally developed uni-level and bi-level LBBD approaches to solve BDORS. We incorporated a novel flexible approach into the LBBDs, allowing them to cast nonlinear objective functions as new constraints and later replace these nonlinear constraints with valid linear logic-based Benders cuts. The new approach worked consistently well under both absolute value and quadratic penalty functions, but failed to solve instances where integer balancing solutions needed to develop Benders cuts were hard to find, i.e., balancing SPs with five ORs per hospital. The new approach is a versatile approach, generalizable to other MIQCP problems with similar mathematical structures. Both LBBDs were competitive with Gurobi with absolute value penalties and significantly outperformed Gurobi with quadratic penalties.

Future work includes designing a balancing SP relaxation into the MP or RMP and cut-strengthening heuristics to ensure faster convergence. BDORS can be extended to an open scheduling setting in which surgeons can alternate among ORs, requiring the inclusion of sequencing decision variables among the starting and finishing times of surgeries, which would require a new exact decomposition technique. Considering downstream units of ORs is another interesting future direction. It would be interesting to further investigate the performance of mathematical models and decomposition techniques using a much longer computation time, as well as examine the solution impact of varying the trade-off weights. Finally, considering BDORS stochastic and robust variations is of both practical and theoretical interest.

Appendix

A.1. Proof of inequality (22)

We first prove Property 1. The current sub-optimal MP solution does not satisfy Inequality (22) and is hence cut off. We reformulate Inequality (22) as follows:

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{r\in\mathcal{R}_h} y_{hdr} - \left|\hat{\mathcal{R}}_{hd}^{(i)}\right|}^{\text{OR increase}} + \overbrace{\sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} \left(1 - y_{hdr}\right)}^{\text{OR decrease}}}^{\text{Resource adjustment}} + \overbrace{\overbrace{\sum_{p\in\mathcal{P}\setminus\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{hdpr}}^{\text{Patient increase}} + \overbrace{\left(\left|\hat{\mathcal{P}}_{hd}^{(i)}\right| - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\mathcal{R}_h} x_{hdpr}\right)}^{\text{Patient decrease}}}^{\text{Load adjustment}} \right) \right) \geq 0 \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)} \tag{29}$$

Instantiating Inequality (29) with the current MP solution ($\hat{\mathcal{R}}_{hd}^{(i)}$, $\hat{\mathcal{P}}_{hd}^{(i)}$, and $\hat{b}_{hd}^{(i)}$) results in

$$\hat{b}_{hd}^{(i)} - \tilde{b}_{hd}^{(i)} < 0 \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which shows the current sub-optimal MP solution does not satisfy Inequality (29) and is hence cut off. We now prove Property 2. No globally integer feasible solution is cut off by Inequality (29). To do

so, we first focus only on future solutions, stemming from resource adjustment strategies and assume that the set of allocated patients ($\hat{\mathcal{P}}_{hd}^{(i)}$) remain unchanged. Thus, Inequality (29) is reduced to

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{r\in\mathcal{R}_h} y_{hdr} - \left|\hat{\mathcal{R}}_{hd}^{(i)}\right|}^{\text{OR increase}} + \overbrace{\sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} \left(1 - y_{hdr}\right)}^{\text{OR decrease}}}^{\text{Resource adjustment}} \right) \right) \geq 0 \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)} \quad (30)$$

Between the two resource adjustment strategies, we first focus on OR increase. We assume the future solution will increase the number of ORs by at least one; therefore, Inequality (30) is reduced to

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\sum_{r\in\mathcal{R}_h} y_{hdr} - \left|\hat{\mathcal{R}}_{hd}^{(i)}\right|}^{>1} \right) \right) \geq 0 \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (30) because $b_{hd} \geq 0$. Now, we focus on OR decrease as a valid strategy. Let's assume the future solution will decrease the number of ORs by at least one; therefore, Inequality (30) is reduced to

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} \left(1 - y_{hdr}\right)}^{>1} \right) \right) \geq 0 \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (30) because $b_{hd} \geq 0$. The simultaneous occurrence of resource adjustment strategies leads to the following Benders balancing cut:

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{r\in\mathcal{R}_h} y_{hdr} - \left|\hat{\mathcal{R}}_{hd}^{(i)}\right|}^{>1} + \overbrace{\sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} \left(1 - y_{hdr}\right)}^{>1}}^{>1} \right) \right) \geq 0 \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (30) because $b_{hd} \geq 0$.

We now focus on future solutions stemming from load adjustment strategies and assume that allocated resources to hospital h on day d are fixed ($\hat{\mathcal{R}}_{hd}^{(i)}$), translating Inequality (22) into the following reduced Benders balancing cut:

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{p\in\mathcal{P}\setminus\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{hdpr}}^{\text{Patient increase}} + \overbrace{\left(\left|\hat{\mathcal{P}}_{hd}^{(i)}\right| - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\mathcal{R}_h} x_{hdpr} \right)}^{\text{Patient decrease}} }^{\text{Load adjustment}} \right) \right) \geq 0 \quad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)} \quad (31)$$

Inequality (31) is reduced to the following Benders cut if patient increase is the only used strategy:

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\sum_{p\in\mathcal{P}\setminus\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{hdpr}}^{>1} \right) \right) \geq 0 \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (31) because $b_{hd} \geq 0$. If patient decrease strategy is used, Inequality (31) is reduced to

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\left(\left|\hat{\mathcal{P}}_{hd}^{(i)}\right| - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\mathcal{R}_h} x_{hdpr} \right)}^{>1} \right) \right) \geq 0 \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (31) because $b_{hd} \geq 0$. The simultaneous occurrence of load adjustment strategies leads to the following Benders balancing cut:

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{p\in\mathcal{P}\setminus\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{hdpr}}^{>1} + \overbrace{\left(\left|\hat{\mathcal{P}}_{hd}^{(i)}\right| - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\mathcal{R}_h} x_{hdpr} \right)}^{>1} }^{>1} \right) \right) \geq 0 \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (31) because $b_{hd} \geq 0$.

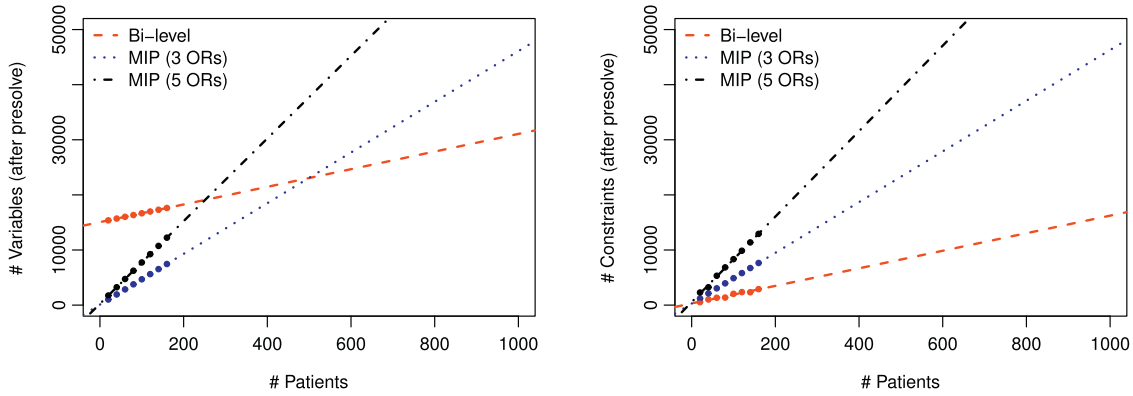
If the future solution is constructed using all the four remedial strategies, Inequality (22) will be as follows:

$$b_{hd} - \tilde{b}_{hd}^{(i)} \left(1 - \left(\overbrace{\overbrace{\sum_{r\in\mathcal{R}_h} y_{hdr} - \left|\hat{\mathcal{R}}_{hd}^{(i)}\right|}^{>1} + \overbrace{\sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} \left(1 - y_{hdr}\right)}^{>1} + \overbrace{\sum_{p\in\mathcal{P}\setminus\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\hat{\mathcal{R}}_{hd}^{(i)}} x_{hdpr}}^{>1} + \overbrace{\left(\left|\hat{\mathcal{P}}_{hd}^{(i)}\right| - \sum_{p\in\hat{\mathcal{P}}_{hd}^{(i)}} \sum_{r\in\mathcal{R}_h} x_{hdpr} \right)}^{>1} }^{>1} \right) \right) \geq 0 \qquad \forall (h,d,i) \in \bar{\mathcal{K}}_d^{(i)}$$

which satisfies Inequality (22) because $b_{hd} \geq 0$. Thus, all potential future feasible solutions satisfy Inequality (22) and hence they are not cut off. □

A.2. Proof of inequalities (26a)–(26e)

The proof is similar to that of Inequality (22). If the future solution keeps the same set of patients and ORs, i.e., $q_{hd}^{(i)}$ and $a_{hd}^{(i)} = 0$, as in the current solution, $b_{hd} < \tilde{b}_{hd}^{(i)}$ in Inequalities (26a)–(26e). Thus, Inequalities (26a)–(26e) eliminate the future solution if it is identical to the current solution (Property 1). Alternatively, Inequalities (26a)–(26e) allow the future solution to choose among all possible remedial strategies: OR increase ($q_{hd}^{(i)} = 0$ and $a_{hd}^{(i)} = 1$), OR decrease ($q_{hd}^{(i)}$ and $a_{hd}^{(i)} = 1$), and patient increase/decrease



(a) Variables (after pre-solve) (b) Constraints (after pre-solve)

Fig. 5. Regression to predict MIP and bi-level LBBD problem size.

($q_{hd}^{(i)} = 1$ and $a_{hd}^{(i)} = 0$). Thus, no future feasible integer solution that can be generated from the combination of remedial strategies is eliminated (Property 2). □

A.3. Proof of inequalities (27a)–(27e)

Property 1 is proven similar to Inequalities (26a)–(26e). Inequalities (27a)–(27e) do not suggest that the RMP opens new ORs because it has opened at least one OR more than the optimal PSP solution ($|\bar{\mathcal{R}}_{hd}^{(i)}|$) to strike macro caseload balancing among hospitals. At the same time, Inequalities (27a)–(27e) do not preclude the RMP from opening a new OR. Assume that the extraneously opened OR remains empty with no assigned patient, leading to possibly maximal micro imbalance cost among partially or totally occupied ORs and the open empty OR(s). This issue can be remedied in the RMP future incumbents with Inequalities (27a)–(27e), which allows patients in one hospital-day to be re-allocated from partially or totally occupied ORs to empty ORs without needing to open a new OR. Alternatively, when $\tilde{b}_{hd}^{(i)}$ is communicated to the RMP, Inequalities (27a)–(27e) have to propose other remedial strategies to eliminate micro imbalance cost difference, among which reducing $\hat{\mathcal{R}}_{hd}^{(i)}$ to $\bar{\mathcal{R}}_{hd}^{(i)}$ is one of the strategies. Thus, opening more ORs than $\hat{\mathcal{R}}_{hd}^{(i)}$ is not a viable strategy given the high cost of opening ORs, which deteriorates the objective function value in terms of costs of opening ORs and macro load balancing. Thus, Inequalities (27a)–(27e) does not cut off a feasible integer solution that can be the optimum. □

A.4. Performance prediction for BDORS larger instances

$\text{MIP}_{\text{AVPF1}}$+Gurobi obtained lower optimality gaps than LBBD methods on both datasets. However, we hypothesize that the bi-level LBBD approach may be more useful than the MIP+Gurobi as problem size increases, since the bi-level problem sizes grow at a slower rate than the full MIP as the number of patients increases (Fig. 5). The bi-level LBBD may be more likely to fit in a computer's memory when the number of patients exceeds $\approx$450. The size dimensionality (number of constraints and variables) of the uni-level MP is very close to that of the MP; therefore, we do not anticipate better performance for the uni-level LBBD as the size of BDORS instances increases.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.omega.2019.03.001.

References

[1] Adan I, Bekkers J, Dellaert N, Vissers J, Yu X. Patient mix optimisation and stochastic resource requirements: a case study in cardiothoracic surgery planning. Health Care Manag Sci 2009;12(2):129–41.
[2] Adan I, Vissers J. Patient mix optimisation in hospital admission planning: a case study. Int J Oper Prod Manag 2002;22(4):445–61.
[3] Banditori C, Cappanera P, Visintin F. A combined optimization-simulation approach to the master surgical scheduling problem. IMA J Manag Math 2013;24(2):155–87.
[4] Batun S, Denton BT, Huschka TR, Schaefer AJ. Operating room pooling and parallel surgery processing under uncertainty. INFORMS J Comput 2011;23(2):220–37.
[5] Beck JC. Checking-up on branch-and-check. In: Cohen D, editor. Principles andpractice of constraint programming CP 2010. Lecture Notes in Computer Science, 6308. Berlin Heidelberg: Springer; 2010. p. 84–98. ISBN 978-3-642-15395-2.
[6] Behnamian J. Multi-cut Benders decomposition approach to collaborative scheduling. Int J Computer Integr Manuf 2015;28(11):1167–77.
[7] Behnamian J, Ghomi SF. The heterogeneous multi-factory production network scheduling with adaptive communication policy and parallel machine. Inf Sci 2013;219(0):181–96.
[8] Belotti P, Kirches C, Leyffer S, Linderoth J, Luedtke J, Mahajan A. Mixed-integer nonlinear optimization. Acta Numer 2013;3:1–131.
[9] Benders JF. Partitioning procedures for solving mixed-variables programming problems. Numer Math 1962;4:238–52.
[10] Bley A. An integer programming algorithm for routing optimization in ip networks. Algorithmica 2011;60(1):21–45.
[11] Booth KEC, Tran TT, Beck JC. Logic-based Benders decomposition methods for the travelling purchaser problem. In: Quimper C-G, editor. Principles and practice of constraint programming – CP 2016. Lecture Notes in Computer Science, 5732. Berlin Heidelberg: Springer; 2016. p. 55–64.
[12] Cardoen B, Demeulemeester E, Belien J. Operating room planning and scheduling: a literature review. Eur J Oper Res 2010;201(3):921–32.
[13] Castro PM, Marques I. Operating room scheduling with generalized disjunctive programming. Comput Oper Res 2015;64:262–73.
[14] Chu Y, Xia Q. Generating Benders cuts for a general class of integer programming problems. In: RĘgin J-C, Rueher M, editors. Integration of AI and OR techniques in constraint programming for combinatorial optimization problems. Lecture Notes in Computer Science, 3011. Berlin Heidelberg: Springer; 2004. p. 127–41. ISBN 978-3-540-21836-4.
[15] Denton BT, Miller AJ, Balasubramanian HJ, Huschka TR. Optimal allocation of surgery blocks to operating rooms under uncertainty. Oper Res 2010;58:802–16.
[16] Fei H, Chu C, Meskens N. Solving a tactical operating room planning problem by a column-generation-based heuristic procedure with four criteria. Ann Oper Res 2009;166(1):91–108.
[17] Fei H, Meskens N, Chu C. A planning and scheduling problem for an operating theatre using an open scheduling strategy. Comput Ind Eng 2010;58(2):221–30.
[18] Fortz B, Poss M. An improved Benders decomposition applied to a multi-layer network design problem. Oper Res Lett 2009;37(5):359–64.
[19] Galvao RD, Espejo LGA, Boffey B, Yates D. Load balancing and capacity constraints in a hierarchical location model. Eur J Oper Res 2006;172(2):631–646.
[20] Gendron B, Lucena A, da Cunha AS, Simonetti L. Benders decomposition, branch-and-cut, and hybrid algorithms for the minimum connected dominating set problem. INFORMS J Comput 2014;26(4):645–57.
[21] Grossmann IE. Review of nonlinear mixed-integer and disjunctive programming techniques. Optim Eng 2002;3(3):227–52.
[22] Guerriero F, Guido R. Operational research in the management of the operating theatre: a survey. Health Care Manag Sci 2011;14(1):89–114.

[23] Hashemi Doulabi SH, Rousseau LM, Pesant G. A constraint-programming-based branch-and-price-and-cut approach for operating room planning and scheduling. INFORMS J Comput 2016;28(3):432–48.
[24] Hooker JN. Logic-based methods for optimization. In: Borning A, editor. Principles and practice of constraint programming. Lecture Notes in Computer Science, 874. Berlin Heidelberg: Springer; 1994. p. 336–49. ISBN 978-3-540-58601-2.
[25] Hooker JN. Planning and scheduling by logic-based Benders decomposition. Oper Res 2007;55(3):588–602.
[26] Hooker JN, Ottosson G. Logic-based Benders decomposition. Math Program 2003;96(1):33–60.
[27] Jebali A, Alouane ABH, Ladet P. Operating rooms scheduling. Int J Prod Econ 2006;99:52–62.
[28] Ku WY, P T, Beck JC. CIP And MIQP models for the load balancing nurse-to–patient assignment problem. In: OSullivan B, editor. Principles andpractice of constraint programming. Lecture Notes in Computer Science, 8656. Springer International Publishing; 2014. p. 424–39. ISBN 978-3-319-10427-0.
[29] Laporte G, Louveaux FV. The integer l-shaped method for stochastic integer programs with complete recourse. Oper Res Lett 1993;13(3):133–42.
[30] Marcon E, Kharraja S, Simonnet G. The operating theatre planning by the follow-up of the risk of no realization. Int J Prod Econ 2003;85(1):83–90.
[31] Marques I, Captivo ME. Bicriteria elective surgery scheduling using an evolutionary algorithm. Oper Res Health Care 2015;7:14–26.
[32] Marques I, Captivo ME, Margarida VP. An integer programming approach to elective surgery scheduling. OR Spectr 2012;34(2):407–27. doi:10.1007/s00291-011-0279-7.
[33] Marques I, Captivo ME, Margarida VP. Scheduling elective surgeries in a portuguese hospital using a genetic heuristic. Oper Res Health Care 2014;3:59–72.
[34] Naderi B, Azab A. Modeling and heuristics for scheduling of distributed job shops. Expert Syst Appl 2014;41(17):7754–63.
[35] Naderi B, Ruiz R. The distributed permutation flowshop scheduling problem. Comput Oper Res 2010;37(4):754–68.
[36] Naderi B, Ruiz R. A scatter search algorithm for the distributed permutation flowshop scheduling problem. Eur J Oper Res 2014;239(2):323–34.
[37] Pham DN, Klinkert A. Surgical case scheduling as a generalized job shop scheduling problem. Eur J Oper Res 2008;185(3):1011–25.
[38] Riise A, Mannino C, Lamorgese L. Recursive logic-based Benders' decomposition for multi-mode outpatient scheduling. Eur J Oper Res 2016;255:719–728.
[39] Roshanaei V. Large-scale decomposition strategies for collaborative operating room planning and scheduling. Department of Mechanical and Industrial Engineering, University of Toronto; 2017. PhD Thesis
[40] Roshanaei V, Luong C, Aleman D, Urbach D. Collaborative operating room planning and scheduling. INFORMS J Comput 2017;29(3):558–80.
[41] Roshanaei V, Luong C, Aleman D, Urbach D. Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling. Eur J Oper Res 2017;257(2):439–55.
[42] Rossum G. Python reference manual. Tech. Rep.; 1995. Amsterdam, The Netherlands, The Netherlands
[43] Samudra M, Van Riet C, Demeulemeester E, Cardoen B, Vansteenkiste N, Rademakers FE. Scheduling operating rooms: achievements, challenges and pitfalls. J Sched 2016;19(5):493–525.
[44] Santibanez P, Begen M, Atkins D. Surgical block scheduling in a system of hospitals: an application to resource and wait list management in a british columbia health authority. Health Care Manag Sci 2007;10(3):269–82.
[45] Tanfani E, Testi A. A pre-assignment heuristic algorithm for the master surgical schedule problem (MSSP). Ann Oper Res 2010;178(1):105–19.
[46] Thorsteinsson ES. Branch-and-check: a hybrid framework integrating mixed integer programming and constraint logic programming. In: Walsh T, editor. Principles and practice of constraint programming – CP 2001. Lecture Notes in Computer Science, 2239. Berlin Heidelberg: Springer; 2001. p. 16–30. ISBN 978-3-540-42863-3.
[47] Tran TT, Araujo A, Beck JC. Decomposition methods for the parallel machine scheduling problem with setups. INFORMS J Comput 2016;28(1):83–95.
[48] Turkay M, Grossmann IE. Logic-based MINLP algorithms for the optimal synthesis of process networks. Comput Chem Eng 1996;20(8):959–78.
[49] Vijayakumar B, Parikh PJ, Scott R, Barnes A, Gallimore J. A dual bin-packing approach to scheduling surgical cases at a publicly-funded hospital. Eur J Oper Res 2013;224(3):583–91.
[50] Vissers J, Adan I, Bekkers J. Patient mix optimization in cardiothoracic surgery planning: a case study. IMA J Manag Math 2005;16(3):281–304.
[51] Wang S, Roshanaei V, Aleman D, Urbach D. A discrete event simulation evaluation of distributed operating room scheduling. IIE Trans Healthc Syst Eng 2016;6(4):236–45.
[52] Weatley D, Gzara F, Jewkes E. Logic-based Benders decomposition for an inventory-location problem with service constraints. Omega 2015;55:10–23.
[53] Xiang W. A multi-objective ACO for operating room scheduling optimization. Nat Comput 2017;16(4):607–17.