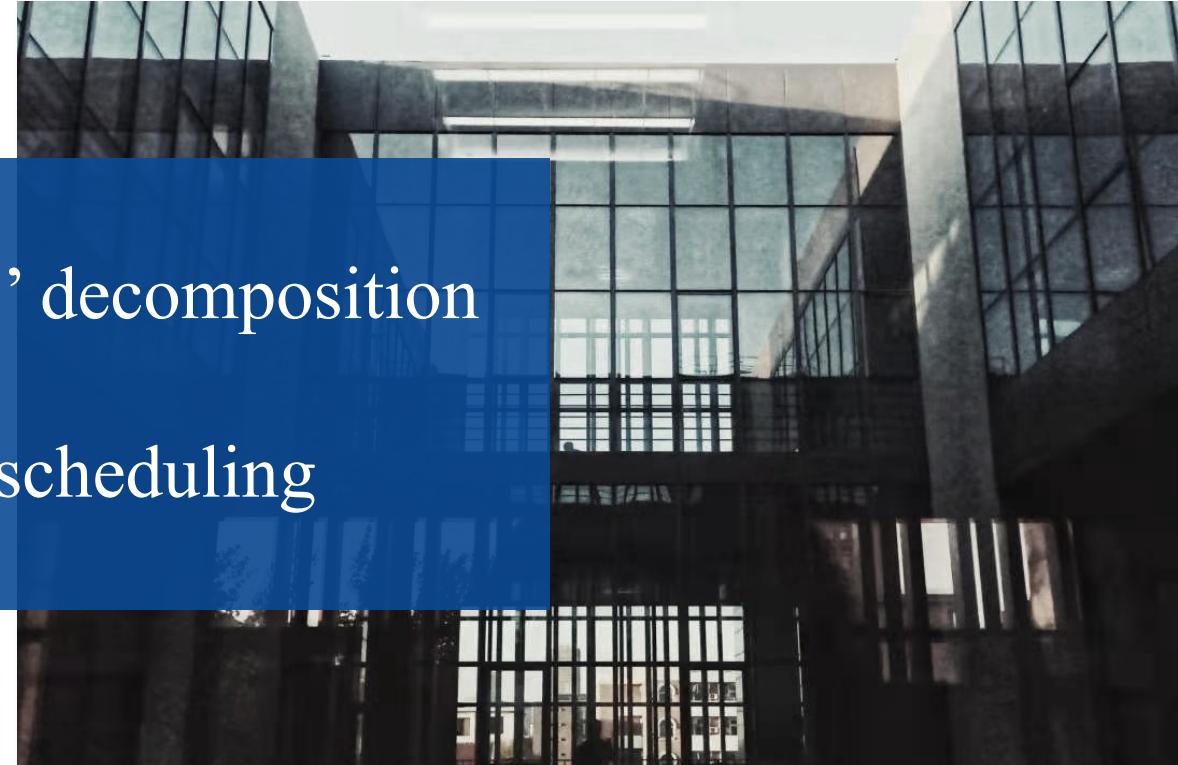


Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling



目录 Contents

- 1 问题介绍**
问题背景、问题描述
- 2 model**
Model介绍、model实现
- 3 算法**
算法介绍、算法实现
- 4 实验结果**
论文实验、结果复现
- 5 算法展望**
前景、个人总结



问题介绍

- ✓ 问题背景
- ✓ 问题描述

介绍——问题背景

sandiago



[Download full issue](#)

[Outline](#)

[Highlights](#)

[Abstract](#)

[Keywords](#)

[1. Introduction](#)

[2. Literature review](#)

[3. Problem definition](#)

[4. Logic-based Benders' decomposition](#)

[5. Computational results](#)

[6. Discussion](#)

[7. Conclusion](#)

[Appendix A](#)

[References](#)

[Show full outline](#)



European Journal of Operational Research

Volume 257, Issue 2, 1 March 2017, Pages 439-455



Discrete Optimization

Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling

Vahid Roshanaei^a , Curtiss Luong^a , Dionne M. Aleman^{a b c} , David Urbach^d

[Show more](#) ▾

+ Add to Mendeley Share Cite

<https://doi.org/10.1016/j.ejor.2016.08.024> ▾

[Get rights and content](#) ▾

Vahid Roshanaei, Curtiss Luong, Dionne M. Aleman, David Urbach, Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling, European Journal of Operational Research, Volume 257, Issue 2, 2017, Pages 439-455, ISSN 0377-2217,
<https://doi.org/10.1016/j.ejor.2016.08.024>.

Copyright© Qianlong Wang, Mails: wangqianlong_nk@foxmail.com

- DORS 是一个考虑患者优先级的多医院集中手术安排调度问题，涉及概念如下：
- 一组协作医院 ($h \in H$)、每家医院有多个手术室 ($r \in R_h$)、规划期 ($d \in D$)，网络中 OR 的数量为 $|H| \times |D| \times |R_h|$ ，每个 OR 的每日可用时间 B_{hd} 。
- 手术室的开放成本（surgical suites and ORs）：套间开放成本、手术室的开放成本（医护人员成本等）
- 患者 ($p \in P$)。考虑了患者手术的等待时间 (α_p) 和患者手术紧急程度评分 (ρ_p)。
- 患者等待成本：等待时间 * 紧急程度评分
- DORS 安排医院等候名单中的全部或部分现有患者。任何未安排在当前计划范围内的患者都将添加到要安排在下一个计划范围内的患者池中。
- 安排在当前规划期内手术的患者分等待成本权重和未安排在当前规划期内手术的患者的等待权重不同（患者手术预备成本）。



MODEL

- ✓ Model介绍
- ✓ Model实现

- (1) 所有 OR 和外科医生在每个外科专业中的功能都相同;
- (2) 主医师以及护士和麻醉师分配到 ORs 已执行;
- (3) 每个 OR 在一天内仅分配给一名外科医生（患者到 OR 的分配等同于患者到外科医生的分配）;
- (4) 医院拥有相同数量的手术室（参考真实数据集）;
- (5) 仅在一个外科专科内进行协同调度;
- (6) 为简化计算，手术持续时间被认为是确定性的;
- (7) 单次手术持续时间 (T_p) 小于最大 OR 开放时间(Bhd)。

MODEL介绍——参数、目标及约束条件OP

sandiago

Table 1
IP notation.

Sets:

\mathcal{P}	Set of patients, $p \in \mathcal{P}$
\mathcal{P}'	Set of mandatory patients, $\mathcal{P}' = \{p \mid \rho_p(\mathcal{D}) - \alpha_p \leq -\Gamma\}$
\mathcal{H}	Set of hospitals, $h \in \mathcal{H}$
\mathcal{D}	Set of days in the planning horizon, $d \in \mathcal{D}$
\mathcal{R}_h	Set of ORs in each hospital's surgical suite, $r \in \mathcal{R}_h$

Parameters:

G_{hd}	Cost of opening the surgical suite in hospital h on day d
F_{hd}	Cost of opening an OR in hospital h on day d
B_{hd}	Regular operating hours of each OR on day d in hospital h
T_p	Total booked time (preparation time + surgery time + cleaning time) of patient p
ρ_p	Health status score assigned to patient p
α_p	Number of days elapsed from the referral date of patient p
κ_1	Waiting cost for scheduled patients
κ_2	Waiting cost for unscheduled patients
Γ	Health status threshold above which patients have to be operated

Variables:

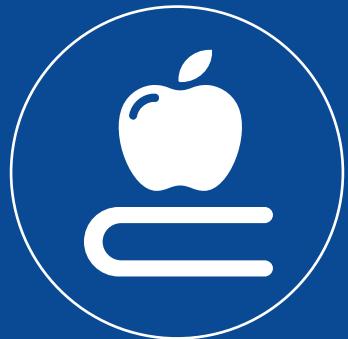
x_{hdpr}	1 if patient p is assigned to hospital h on day d , 0 otherwise
u_{hd}	1 if the surgical suite in hospital h is opened on day d , 0 otherwise
y_{hdr}	1 if OR r of surgical suite of hospital h is opened on day d , 0 otherwise
x_{hdpr}	1 if patient p is assigned to OR r of hospital h on day d , 0 otherwise
w_p	1 if patient p is not scheduled this horizon, 0 otherwise

$$\begin{aligned}
 & \text{minimize} && \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} G_{hd} u_{hd} + \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} \sum_{r \in \mathcal{R}_h} F_{hd} y_{hdr} \\
 & && + \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} \sum_{p \in \mathcal{P}} \sum_{r \in \mathcal{R}_h} \kappa_1 [\rho_p(d - \alpha_p) x_{hdpr}] \\
 & && + \sum_{p \in \mathcal{P} \setminus \{\mathcal{P}'\}} \kappa_2 [\rho_p(|\mathcal{D}| + 1 - \alpha_p) w_p] \\
 & \text{subject to} && \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} \sum_{r \in \mathcal{R}_h} x_{hdpr} = 1 \quad \forall p \in \mathcal{P}' \tag{1} \\
 & && \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} \sum_{r \in \mathcal{R}_h} x_{hdpr} + w_p = 1 \quad \forall p \in \mathcal{P} \setminus \{\mathcal{P}'\} \tag{2} \\
 & && \sum_{p \in \mathcal{P}} T_p x_{hdpr} \leq B_{hd} y_{hdr} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; r \in \mathcal{R}_h \tag{3} \\
 & && y_{hdr} \leq y_{hd,r-1} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; r \in \mathcal{R}_h \setminus \{1\} \tag{4} \\
 & && y_{hdr} \leq u_{hd} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; r \in \mathcal{R}_h \tag{5} \\
 & && x_{hdpr} \leq y_{hdr} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; p \in \mathcal{P}; r \in \mathcal{R}_h \tag{6} \\
 & && u_{hd}, y_{hdr}, x_{hdpr} \in \{0, 1\} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; p \in \mathcal{P}; r \in \mathcal{R}_h \\
 & && w_p \in \{0, 1\} \quad \forall p \in \mathcal{P} \setminus \{\mathcal{P}'\}.
 \end{aligned}$$

Table 2
Parameter values.

κ_1	50 dollars
κ_2	5 dollars
Γ	500 (6–10% of all patients identified as mandatory)
ρ_p	Uniform distribution [1, 5], where 1 is least urgent 5 is the most urgent
B_{hd}	Uniform distribution [420, 480] minutes in 15-minutes intervals
α_p	Uniform distribution [60, 120] days
F_{hd}	Uniform distribution [4000, 6000]
G_{hd}	Uniform distribution [1500, 2500]

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).



算法

✓ 算法介绍

✓ 算法实现

分支定价法：分支定界+列生成算法

列生成算法：通过求解子问题（sub problem）来找到可以进基的非基变量，该非基变量在模型中并没有显性的写出来（可以看成是生成了一个变量，每个变量其实等价于一列，所以该方法被称为列生成算法）。如果找不到一个可以进基的非基变量，那么就意味着所有的非基变量的检验数（Reduced Cost, RC）都满足最优解的条件，也就是说，该线性规划的最优解已被找到。

分解算法

Benders Decomposition算法（BD）：将问题划分为Master Problem（MIP）和Subproblem，但是Subproblem必须为线性规划模型。算法核心：optimal cut 和 feasible cut。算法常使用于便于分解为两阶段的问题。

Logic-based Benders Decomposition算法（LBBD）：由于BD分解算法对问题的结构要求、Subproblem的要求较高，进而提出了基于问题逻辑的BD分解算法。在LBBD算法中，Master problem 和 Subproblem 的要求降低：MP仍然需要为MIP，但是SUBP可为：LP、MIP、CP、可行性检查等。算法核心：问题拆解、optimal cut 和 feasible cut及其证明。

算法介绍——框架

sandiago

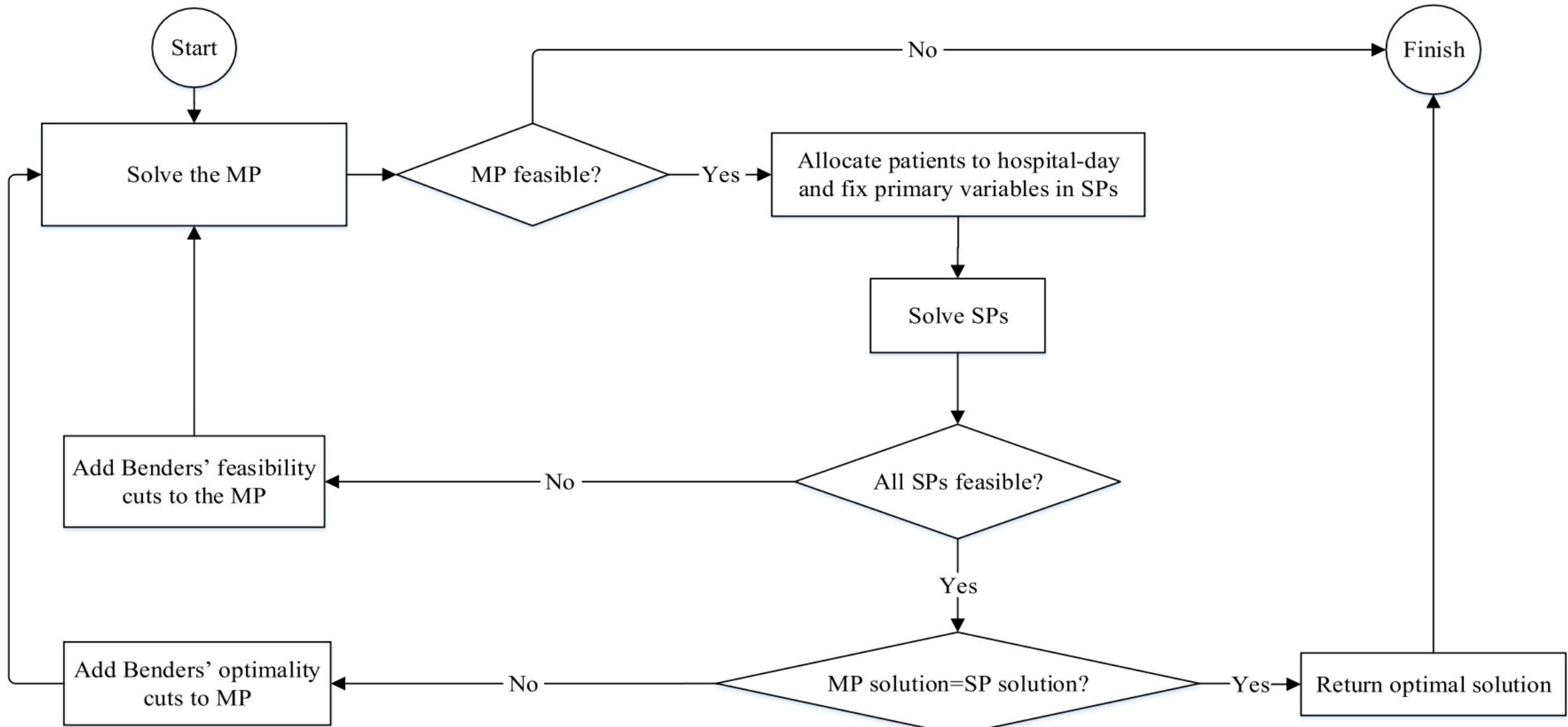


Fig. 1. LBBD flowchart for DORS.

算法介绍—— Location-allocation master problem MP sandiego

Decision: patient → hospital-day, the minimum number of ORs required for each hospital-day.

$$\begin{aligned} \text{minimize} \quad & \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} G_{hd} u_{hd} + \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} F_{hd} y_{hd} \\ & + \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} \sum_{p \in \mathcal{P}} \kappa_1 [\rho_p (d - \alpha_p) x_{hdp}] \\ & + \sum_{p \in \mathcal{P} \setminus \{\mathcal{P}'\}} \kappa_2 [\rho_p (|\mathcal{D}| + 1 - \alpha_p) w_p] \end{aligned} \quad (\text{MP})$$

$$\text{subject to} \quad \sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} x_{hdp} = 1 \quad \forall p \in \mathcal{P}' \quad (7)$$

$$\sum_{h \in \mathcal{H}} \sum_{d \in \mathcal{D}} x_{hdp} + w_p = 1 \quad \forall p \in \mathcal{P} \setminus \{\mathcal{P}'\} \quad (8)$$

$$x_{hdp} \leq u_{hd} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; p \in \mathcal{P} \quad (9)$$

$$\sum_{p \in \mathcal{P}} T_p x_{hdp} \leq |\mathcal{R}_h| B_{hd} u_{hd} \quad \forall h \in \mathcal{H}; d \in \mathcal{D} \quad (10)$$

$$T_p x_{hdp} \leq B_{hd} \quad \forall h \in \mathcal{H}; d \in \mathcal{D}; p \in \mathcal{P} \quad (11)$$

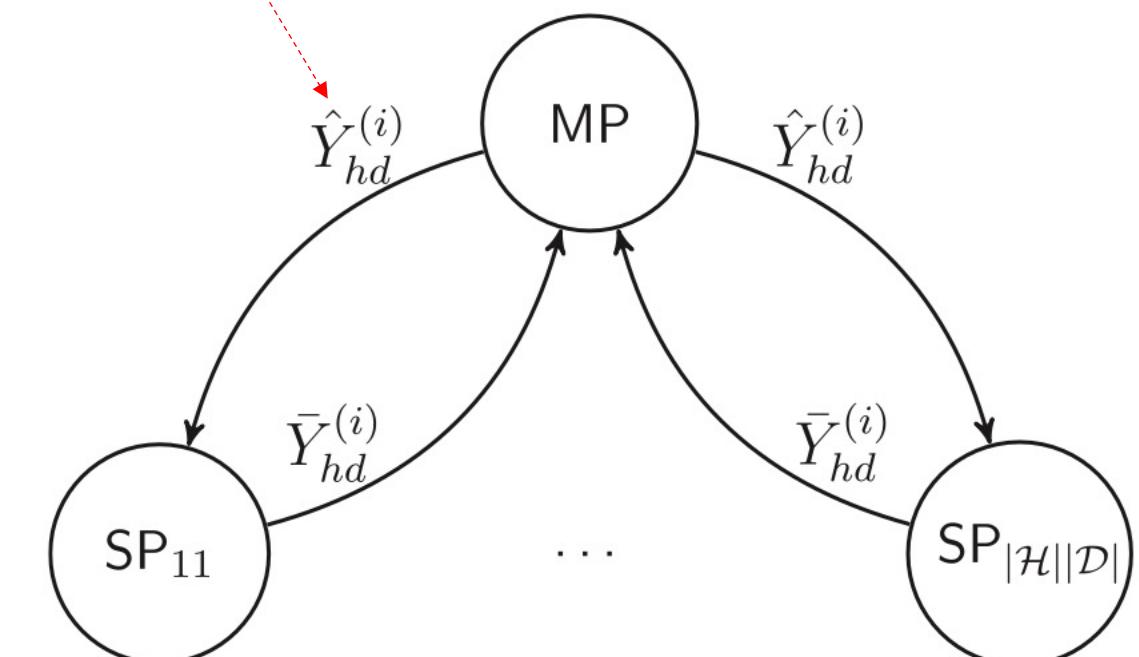
$$y_{hd} \geq \frac{\sum_{p \in \mathcal{P}} T_p x_{hdp}}{B_{hd}} \quad \forall h \in \mathcal{H}; d \in \mathcal{D} \quad (12)$$

$$y_{hd} \leq |\mathcal{R}_h| \quad \forall h \in \mathcal{H}; d \in \mathcal{D} \quad (13)$$

$$y_{hd} \in \mathbb{Z}^+ \quad \forall h \in \mathcal{H}; d \in \mathcal{D}$$

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

$$w_p \in \{0, 1\} \quad \forall p \in \mathcal{P} \setminus \{\mathcal{P}'\}.$$



算法介绍—— OR allocation sub-problems, first-fit sandiego decreasing heuristic algorithm SP

Input : patient → hospital-day

$$\hat{\mathcal{P}}_{hd}^{(i)}$$

Decision : the minimum number of ORs required for each hospital-day, patient → OR (hospital-day)

$$\text{minimize } \bar{Y}_{hd}^{(i)} = \sum_{r \in \mathcal{R}_h} y_r \quad (\text{SP})$$

$$\text{subject to } \sum x_{pr} = 1 \quad \forall p \in \hat{\mathcal{P}}_{hd}^{(i)} \quad (14)$$

$$\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} T_p x_{pr} \leq B_{hd} y_r \quad \forall r \in \mathcal{R}_h \quad (15)$$

$$x_{pr} \leq y_r \quad \forall p \in \hat{\mathcal{P}}_{hd}^{(i)}; r \in \mathcal{R}_h \quad (16)$$

$$y_r \leq y_{r-1} \quad \forall r \in \mathcal{R}_h \setminus \{1\} \quad (17)$$

$$x_{pr}, y_r \in \{0, 1\} \quad \forall p \in \hat{\mathcal{P}}_{hd}^{(i)}; r \in \mathcal{R}_h. \quad (18)$$

此问题可抽象为背包问题，而此问题已经有较好的启发式算法FFD（求解快，效果好），故而可使用FFD快速生成可行解并计算目标。

$$\hat{Y}_{hd}^{(i)} \leq \bar{Y}_{hd}^{(i)} \leq \bar{F}_{hd}^{(i)}$$

若 $\bar{F}_{hd}^{(i)}$ $\hat{Y}_{hd}^{(i)}$ 相等，则说明启发式算法求解至最优解（MP的输出为SP的下界）。不等则使用IP求解SP。

算法介绍—— Benders cut : SP 解的情况讨论

sandiago

1. Infeasible SP if $\hat{\mathcal{P}}_{hd}^{(i)}$ is not packable within $|\mathcal{R}_h|$, implying $\bar{Y}_{hd}^{(i)} > |\mathcal{R}_h|$. $\rightarrow \bar{\mathcal{U}}^{(i)}$
2. Optimal MP and optimal SPs, but $\hat{Y}_{hd}^{(i)} \neq \bar{Y}_{hd}^{(i)}$. $\rightarrow \bar{\mathcal{J}}^{(i)}$
3. Global optimal solution if $\hat{Y}_{hd}^{(i)} = \bar{Y}_{hd}^{(i)}$.

feasible cut

$\bar{\mathcal{U}}^{(i)}$

optimal cut

$\bar{\mathcal{J}}^{(i)}$

$$\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hd p}) \geq 1 \quad \forall (h, d) \in \bar{\mathcal{U}}^{(i)}.$$

$$(19) \quad y_{hd} \geq \bar{Y}_{hd}^{(i)} - \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hd p}) \quad \forall (h, d) \in \bar{\mathcal{J}}^{(i)}. \quad (20)$$

Benders cut 有效性证明

(1) 没有 cut 任何原问题OP的可行解; (2) cut 当前MP的最优解

feasible cut

$\bar{\mathcal{U}}^{(i)}$

$$\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) \geq 1 \quad \forall (h, d) \in \bar{\mathcal{U}}^{(i)}.$$

$$(19) \quad \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) = \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)} \cap \tilde{\mathcal{P}}_{hd}} (1 - x_{hdp}) + \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)} \setminus \tilde{\mathcal{P}}_{hd}} (1 - x_{hdp})$$

Benders cut 有效性证明

- (1) 没有 cut 任何原问题OP的可行解; (2) cut 当前MP的最优解

Proof Feasible Cut (至少从当前MP分配至 SP_{hd} 的患者集合中移除一个患者) :

1) 显而易见, (2) 满足

2) proof (1) : OP在(h, d)上的患者集合(S1)和当前MP在(h, d)上的输出集合(S2)存在四种关系: S1 不等于S2, S1包含S2; S1 不等于S2, S2包含S1; S1 不等于S2, S1交S2不为空; S1 不等于S2, S1交S2为空。易知 (1) 满足

optimal cut $\bar{\mathcal{J}}^{(i)}$

$$y_{hd} \geq \bar{Y}_{hd}^{(i)} - \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) \quad \forall (h, d) \in \bar{\mathcal{J}}^{(i)}.$$

$$(20) \quad y_{hd} - \bar{Y}_{hd}^{(i)} + \left(|\hat{\mathcal{P}}_{hd}^{(i)}| - \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} x_{hdp} \right) \geq 0 \quad \forall (h, d) \in \bar{\mathcal{J}}^{(i)}.$$

Benders cut 有效性证明

- (1) 没有 **cut** 任何原问题**OP**的可行解; (2) **cut** 当前**MP**的最优解

Proof Optiaml Cut (至少从当前MP分配至 S_{Phd} 的患者集合中移除一个患者或增加OR开放数量) :

1) 显而易见, (2) 满足

2) proof (1) : 保持开放OR的数量不变, 证明过程与**feasible cut**证明过程相同。保持患者分配不变:

设OP开放数量为 y_1 , MP输出为 y_2 , 如 y_1 小于等于 y_2 则易知, 此解infeasible。必然有 y_1 大于 y_2 。式

(20) 满足。

算法介绍—— feasible cut增强、Cut propagation for **SBBDiago**

feasible cut

$\bar{\mathcal{U}}^{(i)}$

$$\sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) \geq 1 \quad \forall (h, d) \in \bar{\mathcal{U}}^{(i)}. \quad (19)$$



至少从当前MP分配至 SP_{hd} 的患者集合中移除一个患者或增加OR开放数量

$$y_{hd} \geq (|\mathcal{R}_h| + 1) - \sum_{p \in \hat{\mathcal{P}}_{hd}^{(i)}} (1 - x_{hdp}) \quad \forall (h, d) \in \bar{\mathcal{U}}^{(i)}. \quad (23)$$

Propagation :

If a combination of patients with known surgical times are not feasibly packable within the OR time of a certain hospital-day, it cannot be packed into other hospital-days with less or the same OR time.

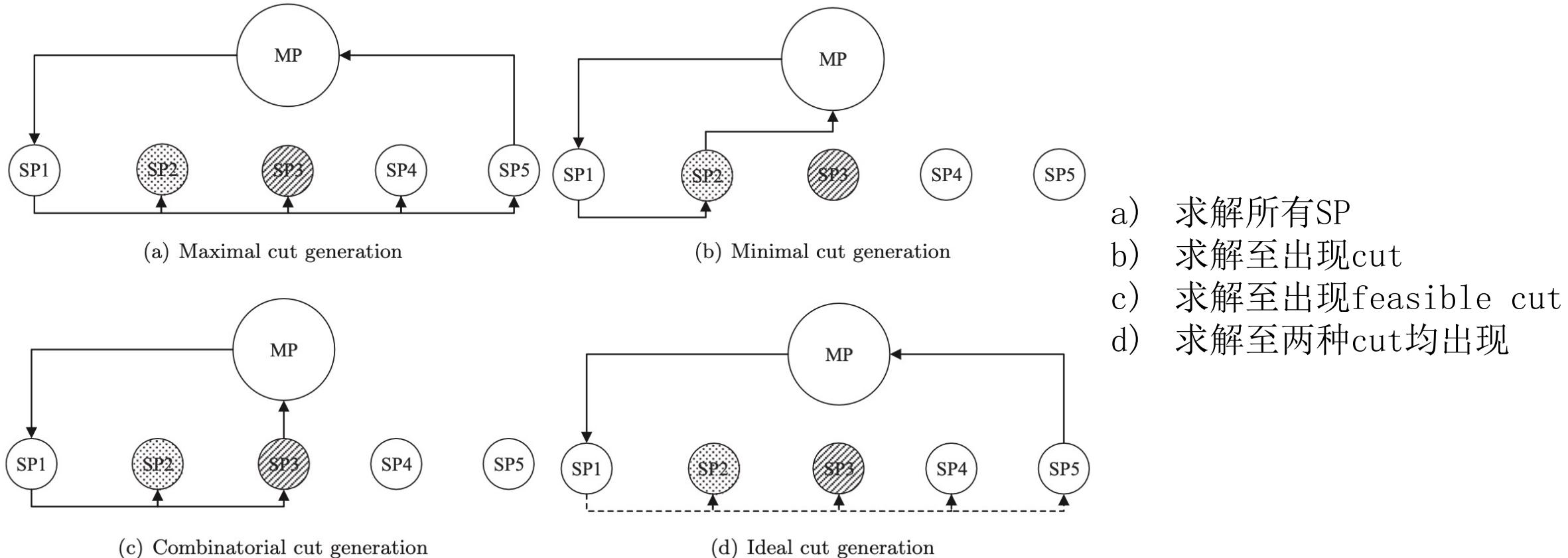


Fig. 3. LBBD implementation techniques. Unshaded SPs are optimally solved, dotted SPs are sub-optimally solved, and striped SPs are infeasible. Dashed lines indicate that SP feasibility is checked before an SP is solved.



实验分析

- ✓ 论文结果
- ✓ 结果复现
- ✓ 算法前景
- ✓ 个人总结

Table 3

Average CPU time (seconds) of LBBD implementations and IP+Gurobi over five trials (bold is the best performance in each test case; superscripts are the average number of unsolved trials per instance; the average CPU time of each implementation is taken over five trials for each of the six LBBDs).

	$ \mathcal{P} $	Maximal	Minimal	Combinatorial	Ideal	LBBD grand average	IP+Gurobi
Five ORs	20	0.39	0.39	0.37	0.39	0.39	13.64
	40	7.04	0.93	6.20	7.08	5.31	68.57 ^(1.00)
	60	7.27	4.03	8.06	7.38	6.69	305.72 ^(3.00)
	80	21.07	183.55	35.07	15.54	63.81	2233.70 ^(3.00)
	100	109.43	103.79	140.83	129.63	120.92	******(5.00)
	120	307.17 ^(0.67)	397.15 ^(0.50)	362.76 ^(0.50)	364.60 ^(0.67)	357.92 ^(0.59)	******(5.00)
	140	1367.39 ^(1.67)	1227.58 ^(1.17)	926.42 ^(1.83)	1132.72 ^(1.67)	1163.53 ^(1.59)	******(5.00)
	160	1263.14 ^(1.50)	1953.05 ^(1.83)	1744.88 ^(1.83)	1496.17 ^(2.17)	1614.31 ^(1.83)	******(5.00)
Three ORs	20	0.49	0.49	0.52	0.45	0.49	14.40
	40	8.79	7.08	9.05	9.14	8.52	5500.00 ^(3.00)
	60	11.17	11.16	11.10	13.98	11.85	2904.60 ^(1.00)
	80	133.38	299.93	285.60	161.05	1219.99	5718.90 ^(4.00)
	100	554.65 ^(0.83)	1451.79 ^(0.33)	1210.53 ^(0.50)	976.55 ^(0.83)	1048.38 ^(0.62)	6500.90 ^(4.00)
	120	711.76 ^(2.00)	1589.68 ^(1.17)	1380.80 ^(1.83)	1755.94 ^(2.00)	1359.55 ^(1.75)	5934.00 ^(4.00)
	140	929.89 ^(3.17)	1282.93 ^(3.67)	731.80 ^(3.83)	1970.31 ^(3.67)	1228.73 ^(3.59)	4099.70 ^(4.00)

实验——论文结果.....

sandiago

Table 4

Five ORs: average CPU time (seconds) of LBBDs over five trials (bold is the best performance in each test case; superscripts are the numbers of unsolved trials; highlight indicates improved performance with cut propagation).

	$ \mathcal{P} $	LBBD1	LBBD1 _P	LBBD2	LBBD2 _P	LBBD3	LBBD3 _P
Maximal	20	0.40	0.38	0.38	0.38	0.40	0.39
	40	0.92	0.97	0.87	8.49	0.87	30.13
	60	5.12	13.10	4.84	3.30	4.45	12.85
	80	62.28	18.11	3.07	5.59	13.34	24.00
	100	126.74	95.26	36.73	44.31	187.17	166.36
	120	309.55	810.01	113.27	176.36 ⁽¹⁾	119.94 ⁽¹⁾	313.88 ⁽²⁾
	140	1770.20 ⁽²⁾	2060.60	883.87 ⁽²⁾	391.04 ⁽¹⁾	2229.10 ⁽³⁾	869.50 ⁽²⁾
	160	1164.30 ⁽¹⁾	1422.80 ⁽¹⁾	1629.10 ⁽¹⁾	372.70 ⁽¹⁾	2339.70 ⁽³⁾	650.22 ⁽²⁾
Average		429.94 ⁽³⁾	552.65 ⁽¹⁾	334.02 ⁽³⁾	125.27 ⁽³⁾	611.87 ⁽⁷⁾	258.42 ⁽⁶⁾
Minimal	20	0.39	0.38	0.38	0.38	0.40	0.40
	40	0.87	0.61	0.85	1.05	0.86	1.31
	60	7.08	1.66	3.15	2.57	6.84	2.88
	80	4.21	1.43	94.08	33.76	726.50	241.34
	100	144.75	65.67	45.00	39.55	247.58	80.21
	120	653.11 ⁽¹⁾	259.64	557.61	421.86	126.07 ⁽¹⁾	364.62 ⁽¹⁾
	140	2336.90 ⁽¹⁾	1325.70 ⁽¹⁾	632.97 ⁽¹⁾	400.28 ⁽¹⁾	1149.20 ⁽²⁾	1520.40 ⁽¹⁾
	160	3548.90	1944.20 ⁽¹⁾	1373.70 ⁽¹⁾	275.40 ⁽²⁾	3907.30 ⁽⁴⁾	665.82 ⁽³⁾
Average		837.03 ⁽²⁾	449.91 ⁽²⁾	338.47 ⁽²⁾	146.74 ⁽³⁾	770.59 ⁽⁷⁾	359.62 ⁽⁵⁾
Combinatorial	20	0.37	0.37	0.37	0.36	0.38	0.37
	40	0.87	0.94	0.86	3.61	0.85	30.04
	60	7.41	17.95	3.17	2.52	4.43	12.88
	80	4.04	41.61	94.13	33.55	13.14	23.96
	100	133.50	272.98	45.05	39.30	187.47	166.70
	120	534.61	897.12	200.85	110.67	119.67 ⁽¹⁾	313.63 ⁽²⁾
	140	655.00 ⁽²⁾	755.93 ⁽¹⁾	627.27 ⁽¹⁾	430.06 ⁽²⁾	2222.10 ⁽³⁾	868.18 ⁽²⁾
	160	3506.00 ⁽¹⁾	2184.10 ⁽¹⁾	1468.80 ⁽²⁾	274.13 ⁽²⁾	2402.60 ⁽³⁾	633.66 ⁽²⁾
Average		605.23 ⁽³⁾	521.38 ⁽²⁾	269.08 ⁽³⁾	111.84 ⁽⁴⁾	618.83 ⁽⁷⁾	256.18 ⁽⁶⁾
Ideal	20	0.40	0.38	0.39	0.41	0.40	0.38
	40	0.92	0.98	0.89	8.64	0.91	30.16
	60	5.16	13.24	4.99	3.43	4.51	12.93
	80	4.30	37.90	6.92	6.40	13.48	24.26
	100	270.79	82.22	31.98	38.24	187.59	166.96
	120	382.33 ⁽¹⁾	303.33	528.36	541.15	119.50 ⁽¹⁾	312.92 ⁽²⁾
	140	517.50 ⁽¹⁾	1443.70 ⁽¹⁾	605.73 ⁽²⁾	1111.10 ⁽¹⁾	2247.40 ⁽³⁾	870.90 ⁽²⁾
	160	2788.9 ⁽²⁾	2487.90 ⁽²⁾	377.90 ⁽¹⁾	253.91 ⁽³⁾	2404.50 ⁽³⁾	663.91 ⁽²⁾
Average		496.29 ⁽⁴⁾	546.33 ⁽³⁾	196.65 ⁽³⁾	245.41 ⁽⁴⁾	622.29 ⁽⁷⁾	260.30 ⁽⁶⁾
Grand average		592.12 ^(3.00)	516.08 ^(2.00)	888.61 ^(2.75)	158.54 ^(3.5)	655.90 ^(7.00)	283.63 ^(5.75)

Table 5

Three ORs: Average CPU time (seconds) of LBBDs over five trials (bold is the best performance in each test case; superscripts are the numbers of unsolved trials; asterisks are scenarios with no solved trials; highlight indicates improved performance with cut propagation).

	$ \mathcal{P} $	LBBD1	LBBD1 _P	LBBD2	LBBD2 _P	LBBD3	LBBD3 _P
Maximal	20	0.64	0.45	0.45	0.47	0.48	0.46
	40	2.56	6.84	10.59	10.73	10.82	11.17
	60	19.18	11.39	9.45	8.94	9.10	8.98
	80	183.22	297.55	53.82	103.97	55.05	106.65
	100	625.15 ⁽²⁾	1581.00 ⁽¹⁾	385.59	135.47 ⁽¹⁾	402.16	138.54 ⁽¹⁾
	120	662.36 ⁽²⁾	673.15 ⁽²⁾	291.47 ⁽²⁾	1175.5 ⁽²⁾	295.80 ⁽²⁾	1172.30 ⁽²⁾
	140	2546.40 ⁽³⁾	2276.30 ⁽³⁾	305.51 ⁽³⁾	51.95 ⁽³⁾	341.57 ⁽³⁾	57.60 ⁽⁴⁾
	Average	577.07 ⁽⁷⁾	692.38 ⁽⁶⁾	150.98 ⁽⁵⁾	212.43 ⁽⁶⁾	159.28 ⁽⁵⁾	213.67 ⁽⁷⁾
Minimal	20	0.65	0.46	0.44	0.45	0.45	0.46
	40	7.89	13.99	6.72	3.49	6.82	3.55
	60	12.69	16.87	9.41	9.22	9.45	9.32
	80	340.15	985.98	52.44	183.01	53.16	184.86
	100	2160.60 ⁽¹⁾	2440.60 ⁽¹⁾	756.49	1300.10	766.33	1286.6
	120	3668.30 ⁽³⁾	3033.50 ⁽²⁾	1033.10 ⁽¹⁾	377.26	1048.80 ⁽¹⁾	376.91
	140	***** ⁽⁵⁾	***** ⁽⁵⁾	1086.40 ⁽³⁾	1476.20 ⁽³⁾	1090.90 ⁽³⁾	1479.10 ⁽³⁾
	Average	1031.71 ⁽⁹⁾	1081.90 ⁽⁸⁾	420.71 ⁽⁴⁾	478.53 ⁽³⁾	425.13 ⁽⁴⁾	477.26 ⁽³⁾
Combinatorial	20	0.60	0.52	0.51	0.50	0.50	0.48
	40	7.90	14.41	7.17	3.71	10.24	10.84
	60	12.96	16.97	9.48	9.23	9.06	8.92
	80	340.22	977.06	53.79	185.93	52.47	104.12
	100	2200.30 ⁽¹⁾	2480.00 ⁽¹⁾	768.24	1289.60	388.91	136.11 ⁽¹⁾
	120	3916.60 ⁽³⁾	1280.10 ⁽³⁾	950.97 ⁽¹⁾	637.10	300.12 ⁽²⁾	1199.90 ⁽²⁾
	140	***** ⁽⁵⁾	1091.40 ⁽³⁾	1488.80 ⁽³⁾	297.87 ⁽³⁾	49.12 ⁽⁴⁾	
	Average	1079.76 ⁽⁹⁾	794.84 ⁽⁹⁾	411.65 ⁽⁴⁾	516.41 ⁽³⁾	151.314 ⁽⁵⁾	215.64 ⁽⁷⁾
Ideal	20	0.56	0.45	0.42	0.43	0.43	0.42
	40	2.97	14.56	10.99	6.52	9.98	9.84
	60	31.13	16.78	9.41	8.83	8.96	8.72
	80	243.60	469.33	53.97	50.55	51.93	96.91
	100	3147.40 ⁽²⁾	1847.10 ⁽¹⁾	106.90 ⁽¹⁾	239.69	386.11	132.12 ⁽¹⁾
	120	4191.20 ⁽²⁾	2853.50 ⁽³⁾	960.99 ⁽²⁾	837.62 ⁽¹⁾	309.02 ⁽²⁾	1383.30 ⁽²⁾
	140	7158.40 ⁽⁴⁾	***** ⁽⁵⁾	1277.30 ⁽³⁾	1046.10 ⁽³⁾	320.84 ⁽³⁾	48.90 ⁽⁴⁾
	Average	2110.77 ⁽⁸⁾	866.95 ⁽⁹⁾	345.71 ⁽⁶⁾	312.82 ⁽⁴⁾	155.32 ⁽⁵⁾	240.03 ⁽⁷⁾
Grand average		1199.83 ^(8.25)	859.02 ^(8.00)	332.26 ^(4.75)	380.05 ^(4.00)	222.76 ^(4.75)	286.65 ^(6.00)

实验——结果复现

sandiago

```
LBBD x origin_problem x
Optimal solution found (tolerance 1.00e-08)
Best objective -4.394050000000e+05, best bound -4.394050000000
Day-Hospital-Room: 0-0-0, Patient: 0, 3, 8,
Day-Hospital-Room: 0-0-1, Patient: 1, 19, 22,
Day-Hospital-Room: 0-0-2, Patient: 17, 27, 33,
Day-Hospital-Room: 0-0-3, Patient: 2, 13, 37,
Day-Hospital-Room: 0-0-4, Patient: 6, 30, 36,
Day-Hospital-Room: 1-2-0, Patient: 12, 14, 34,
Day-Hospital-Room: 1-2-1, Patient: 4, 7, 23,
Day-Hospital-Room: 1-2-2, Patient: 28, 39,
Day-Hospital-Room: 1-2-3, Patient: 9, 11, 21,
Day-Hospital-Room: 1-2-4, Patient: 25, 31, 35,
Day-Hospital-Room: 2-0-0, Patient: 5, 10, 15,
Day-Hospital-Room: 2-0-1, Patient: 16, 32, 38,
Day-Hospital-Room: 2-0-2, Patient: 18, 26,
Day-Hospital-Room: 2-0-3, Patient: 20, 24, 29,
Not Scheduled Patients: cost_suite: 5691.0
cost_room: 61504.0
cost_3: -506600.0
cost_4: 0.0
objective: -439405.0
time: 212.51316499710083
```

```
LBBD x origin_problem x
H 0 0 5.0000000 5.00000 0.00% - 0s
0 0 5.00000 0 5 5.00000 5.00000 0.00% - 0s
Cutting planes:
Gomory: 4
Cover: 5
MIR: 3
StrongCG: 3
Mod-K: 1
RLT: 4

Explored 1 nodes (201 simplex iterations) in 0.01 seconds (0.00 work units)
Thread count was 8 (of 8 available processors)

Solution count 1: 5

Optimal solution found (tolerance 1.00e-08)
Best objective 5.000000000000e+00, best bound 5.000000000000e+00, gap 0.0000%
----- iter: 14, add_cut_num: 0 -----
----- optimal -----
----- add_cut_num_all: 18 -----
LBBD: optimal objective: -439405.0
LBBD time: 13.427375078201294
```

Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling

European Journal Of Operational Research

EI检索

SCI升级版 管理学2区

SCI基础版 管理科学2区

IF 6.4

SWJTU A++

V Roshanaei, C Luong, DM Aleman, D Urbach

European Journal of Operational Research, 2017 • Elsevier

Abstract

We develop three novel logic-based Benders' decomposition (LBBD) approaches and a cut propagation mechanism to solve large-scale location-allocation integer programs (IPs). We show that each LBBD can be implemented in four different possible ways, yielding distinct LBBD variants with completely different computational performances. LBBDs decompose the IP model into an integer location and knapsack-based allocation master problem and multiple packing IP sub-problems. We illustrate the performance of our

展开 ▼

☆ 保存 引用 被引用次数: 94 相关文章 所有 4 个版本 easyScholar文献收藏

算法前景——Recently

sandiago

Find articles with these terms

Search

Advanced search

5,191 results

Set search alert

Refine by:

Years

2024 (413)
2023 (514)
2022 (448)

Show more ▾

Article type ②

Review articles (679)
Research articles (3,241)
Encyclopedia (99)
Book chapters (706)

Show more ▾

Publication title

European Journal of Operational Research (150)
Computers & Operations Research (112)
Computers & Chemical Engineering (97)

[ps://www.sciencedirect.com/science/article/pii/S0305054824001783](https://www.sciencedirect.com/science/article/pii/S0305054824001783)

Download selected articles Export

sorted by relevance | date

1 Logic-based Benders decomposition for order acceptance and scheduling on heterogeneous factories with carbon caps EI检索 Research article Full text access

SCI升级版 工程技术2区 SCI基础版 工程技术3区 IF 4.6 SWJTU A++

Computers & Operations Research, August 2024
Jian Chen, Xudong Ye, ... Dehua Xu

View PDF Abstract Extracts Figures Export

2 A logic-based Benders decomposition solution approach for two covering problems that consider the underlying transport EI检索 SCI升级版 工程技术2区 SCI基础版 工程技术3区 IF 4.6 SWJTU A++

Computers & Operations Research, December 2023
Vera Fischer, Sonne Wöhlk

View PDF Abstract Extracts Figures Export

3 Logic-based Benders decomposition for order acceptance and scheduling in distributed manufacturing EI检索 SCI升级版

Advanced Engineering Informatics, October 2023
Jian Chen, Wenjing Ma, ... Zhiheng Zhao

View PDF Abstract Extracts Figures Export

4 Logic-based Benders decomposition for bi-objective parallel machine selection and job scheduling with release dates and reconsumption

Research article Full text access

SCI基础版 工程技术3区 IF 4.6 SWJTU A++

Computers & Operations Research, July 2023
Alireza Ghahtorani, Ahmed Saif, ... Erick Delage

View PDF Abstract Extracts Figures Export

5 Capacity reservation for humanitarian relief: A logic-based Benders decomposition method with subgradient cut EI检索 Research article Full text access

SCI升级版 管理学2区 SCI基础版 管理科学2区 IF 6.4 SWJTU A++

European Journal of Operational Research, 16 December 2023
Penghui Guo, Jianjun Zhu

View PDF Abstract Extracts Figures Export

6 A logic-based Benders decomposition algorithm for a repair crew routing and drone scheduling problem after a natural disaster EI检索 SCI升级版 工程技术1区 SCI基础版 工程技术2区 IF 7.9 SWJTU A+

Computers & Industrial Engineering, September 2023
Motahareh Safdari Shadlou, Mohammad Ranjbar, Majid Salari

View PDF Abstract Extracts Figures Export

7 Constraint programming and logic-based Benders decomposition for the integrated process planning and scheduling problem EI检索 Research article Full text access

SCI基础版 管理科学1区 IF 6.9 SWJTU A++ SWUFE A

Omega, June 2023
Xuedong Zhu, Junbo Son, ... Jianguo Wu

View PDF Abstract Extracts Figures Export

8 A double-oracle, logic-based Benders decomposition approach to solve the K-adaptability problem EI检索 SCI升级版 工程技术2区

Computers & Operations Research, July 2023
Alireza Ghahtorani, Ahmed Saif, ... Erick Delage

View PDF Abstract Extracts Figures Export

9 Combining optimisation and simulation using logic-based Benders decomposition EI检索 SCI升级版 管理学2区 SCI基础版 管理科学2区

IF 6.4 SWJTU A++

European Journal of Operational Research, 1 February 2024

Feedback

算法前景——Recently

sandiago



Log In | Help |

JOURNALS | MAGAZINES | PUBLICATIONS | PRICING & SUBSCRIPTIONS

logic-based Benders de Advanced Search

logic-based Benders decomposition

SEARCH

Advanced Search



Logic-Based Benders Decomposition for Integrated Process Configuration and Production Planning Problems

Karim Pérez Martínez, Yossiri Adulyasak, Raf Jans

Published Online: INFORMS Journal on Computing (July-August 2022)

Preview Abstract

PDF (824 KB) Supplementary Material Permissions



Critical-Path-Search Logic-Based Benders Decomposition Approaches for Flexible Job Shop Scheduling

Bahman Naderi, Vahid Roshanaei

Published Online: INFORMS Journal on Optimization (Winter 2022)

Preview Abstract

PDF (1264 KB) Supplementary Material Permissions



Solving the Type-2 Assembly Line Balancing with Setups Using Logic-Based Benders Decomposition

Hassan Zohali, Bahman Naderi, Vahid Roshanaei

Published Online: INFORMS Journal on Computing (January-February 2022)

Preview Abstract

PDF (1650 KB) Supplementary Material Permissions



Novel Formulations and Logic-Based Benders Decomposition for the Integrated Parallel Machine

算力增强→精确算法求解规模增加
精确算法和元启发式算法结合（FFD）
+RL/ML

感谢！
THANKS

