算法学习进展

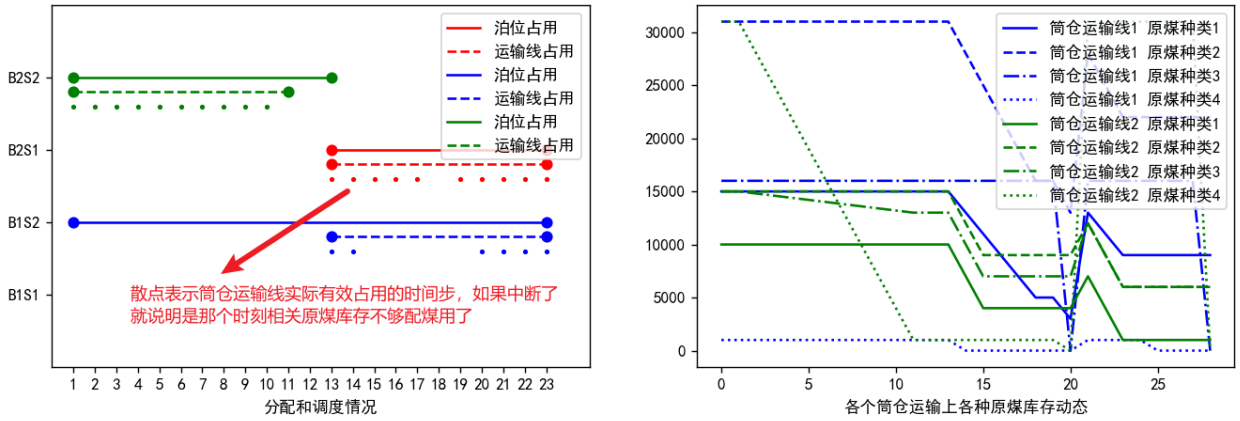
基于逻辑的Benders分解算法

佘远升

大连海事大学交通运输工程学院

2026年3月19日

Multi-Resource Joint Scheduling for Fast-Turnaround Silo Terminals under Soft Coal Quality Constraints（航道调度+泊位分配+筒仓组运营）

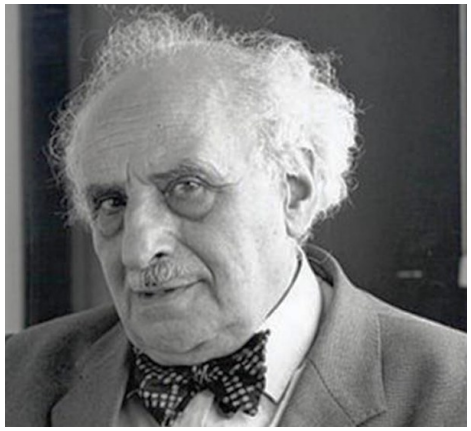


干散货领域：集成模型不好解，精确解算法的文献相对比较少

王帅安等人在其2025年关于散货码头运营的综述文章中提到希望未来的文章能够更多地从模型特征角度定制合理的分解算法：

Undoubtedly, there will always be interest in the integrated optimization of operating such large systems in port sectors. Until there is a leap in computing power that allows these integration problems to be solved easily in industrial-scale scenarios, the development of effective problem decomposition methods and mathematical techniques will remain of great importance. We believe that judiciously considering the synchronization characteristics of DBT operations and finding a reasonable problem decomposition will be more efficient than constantly integrating multiple subproblems into a single model and trying to design an efficient solution algorithm.

搞出来经典Benders的大佬



Jacques F. Benders
(1924.6.1-2017.1.9)

大约40年后

BD进化成了LBBD

搞出来逻辑Benders的大佬



J. N. Hooker
(1949.5.15-)

① Benders分解和行生成是什么关系？Jacques F. Benders 1962年提出的经典Benders分解的思路是什么，经典Benders分解在应用上有什么局限性？

② J. N. Hooker提出的Inference dual是什么东西？他为什么要提出这样一种全新的对偶？LBBD和经典Benders分解有什么区别？LBBD收敛到最优解的充分条件是什么？Inference dual是否是强对偶对LBBD的收敛性有影响吗？

③ 第一篇用LBBD做考虑准备时间的并行机调度问题的论文是怎么用LBBD的？最近几年散货码头运营相关论文是怎么用LBBD的，LBBD的设计难点在哪？

https://www.zhihu.com/collection/975351744

如何理解benders decomposition在混合整数规划中的应用?



rjz elec，电力系统稳定优化与控制

33 人赞同了该回答 ›

也算研究了benders decomposition✦ 3年，自认为没有太多话语权，毕竟是电力系统✦的学生，没有太强的数学功底，但是还是想分享分享硕士三年多对BD的感想。

主要是几个阶段：

刚开始接触BD是因为，电力系统现在更多的问题规模比较庞大，为了更好的解决大规模问题，所以对其进行分解，即刚接触本德斯的理解，大规模问题分层求解；

后来由于在研究本德斯的时候，发现其核心是对偶变换✦，又开始自学运筹学✦的知识，又理解本德斯是的一点，就是他是将原始多约束子问题对偶成少约束复杂目标形式；

后来有一次跟英国的一个lecturer谈论，发现原来所谓的对偶，其实在本德斯里的作用是将子问题转化为约束可以添加到主问题；

▲ 已赞同 33 ▼ 12 条评论 收藏 喜欢 分享 ··· 收起 ^

$$
\begin{aligned}
(\mathrm{P}) \quad \min \quad & c^T x + f^T y \\
\text{s.t.} \quad & Ax + By = b \\
& x \in X \subseteq R^n, y \in Y
\end{aligned}
$$

变量分组写成双层规划的形式

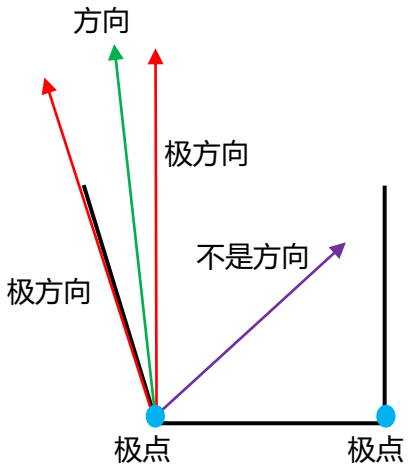
$$
(\mathrm{P}) \quad \min_{y \in Y} \quad \{ f^T y + \underbrace{\min_{x \in X} \{ c^T x | Ax = b - By \}}_{q(y)} \}
$$

$$
q(y) = \max_{\alpha} \{ (b - By)^T \alpha | A^T \alpha \leq c \}
$$

线性规划对偶，强对偶性替换

$$
\begin{aligned}
(\mathrm{P}) \quad \min_{y \in Y} \quad & \{ f^T y + q(y) \} \\
\text{s.t.} \quad & (b - By)^T \alpha_j^r \leq 0, \quad \forall j \in J
\end{aligned}
$$

$$
\begin{aligned}
q(y) &= \max_{\alpha} \{ (b - By)^T \alpha | A^T \alpha \leq c \} \\
&= \min_{q} \{ q | (b - By)^T \alpha_i^p \leq q, \forall i \in I \}
\end{aligned}
$$



$$(\mathrm{P}) \quad \min_{y \in Y, q} \quad \{f^T y + q\}$$

$$\text{s.t.} \quad (b - By)^T \alpha_j^r \leq 0, \quad \forall j \in J$$

经典Benders可行割

$$(b - By)^T \alpha_i^p \leq q, \quad \forall i \in I$$

经典Benders最优割



留德华叫兽

海德堡大学 交叉学科计算中心离散与组合优化实验室研究员

已关注

52 人赞同了该回答 ›

意大利潘多瓦的Matteo Fischetti - DEI是Benders世界级的专家，有幸和他吃过饭，还有和他的得意门生Andrea Lodi合作过。

一种特殊形式的割平面法：

简单得说，约束不等式太多，比如1000个，但是optimal solution里面，bounding/active的不等式可能只有100个，另外900个都是多余的。所以先从很小一部分不等式开始算master problem✦，比如10个，然后有个subproblem✦，用来add constraints on the fly。如此循环。然后加到100个不等式的时候，subproblem找不到新的不等式了，就optimal了。

一般情况下，master problem仍旧是一个integer or mixed integer programming, subproblem是一个LP，用来加Benders' feasibility cut✦ or optimal cut,前者是那个LP问题的extreme ray（换句话说LP unbounded）,后者是LP bounded 情况下的最优解（一般是唯一的？）。当LP unbounded的时候，一般每一次subproblem可以加不止一条feasiblity cut，然后optimality cut✦的质量一般比前者要好。

行生成的思想

$$(\mathrm{MP}) \quad \min_{y \in Y, q} \quad \{f^T y + q\}$$

松弛操作，行生成

J. N. Hooker给Inference Dual下的定义：

Consider a general optimization problem of the form

$$\min_{x} \{ f(x) \mid C(x),\ x \in \mathcal{D}_x \} \tag{2.1}$$

where $C(x)$ represents a constraint set containing variables in $x = (x_1, \ldots, x_n)$, and $\mathcal{D}_x$ is the domain of x (for example, tuples of nonnegative reals or integers). We refer to (2.1) as the *primal* problem. The *inference dual* of (2.1) is the problem of finding the tightest lower bound on the objective function that can be deduced from the constraints. It can be stated

$$\max_{v,P} \left\{ v \;\middle|\; C(x) \overset{P}{\vdash} \big(f(x) \geq v\big),\ v \in \mathbb{R},\ P \in \mathcal{P} \right\} \tag{2.2}$$

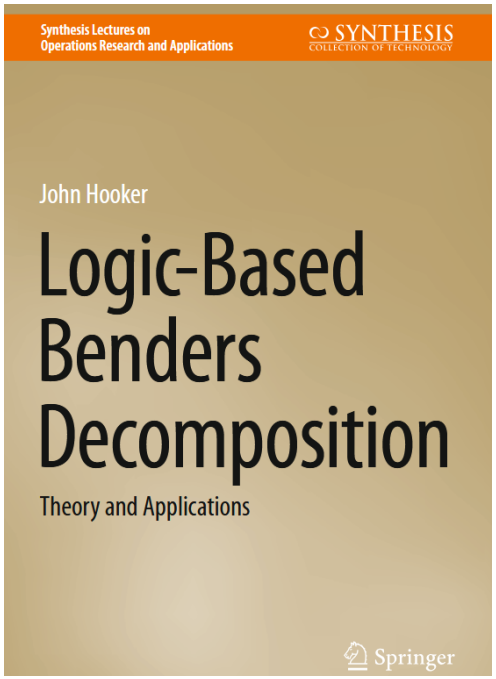
The notation $C(x) \overset{P}{\vdash} (f(x) \geq v)$ indicates that proof P deduces $f(x) \geq v$ from $C(x)$, given that $x \in \mathcal{D}_x$. The domain of variable P is a family $\mathcal{P}$ of proofs, and a solution of the dual is a proof of the tightest lower bound v. The inference dual is always defined with respect to an inference method reflected by the proofs in $\mathcal{P}$.

从实践的角度出发，理不理解推理对偶影响不大，
生成逻辑benders割的方法就是把对问题什么时候不可行，
什么时候下界至少是多少的分析结果用数学约束的形式表达出来

LBBD收敛性到最优解的充分条件：主问题（MP）的可行域是可枚举的就一定能收敛到最优解，如果你的割好，就收敛的快，割不好，就收敛的慢

J. N. Hooker在 *Logic-Based Benders Decomposition Theory and Applications* 这本书中提及的割的类型（部分）：

- 按目的：可行性割 / 最优性割
- 按构造方式：
  - 经典对偶割（理论依据是LP dual）
  - 基于逻辑的割（理论依据是Inference dual）
    - 简单nogood割（只针对主问题某个特定解）
    - 单调割（利用单调性扩大范围）
      - 加强单调割（把不重要的变量移除）
    - 分析割（基于问题结构进行分析设计）
      - 加强分析割（把不重要的变量移除）
      - 通用分析割
    - …



绿色类型的割可以套模板，但是效果可能不是很好

蓝色类型的割设计难度大，需要根据问题结构做分析设计，效果可好可坏

This article was downloaded by: [132.174.255.116] On: 01 February 2016, At: 03:59
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA



INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Decomposition Methods for the Parallel Machine Scheduling Problem with Setups

Tony T. Tran, Arthur Araujo, J. Christopher Beck

To cite this article:
Tony T. Tran, Arthur Araujo, J. Christopher Beck (2016) Decomposition Methods for the Parallel Machine Scheduling Problem with Setups. INFORMS Journal on Computing 28(1):83-95. http://dx.doi.org/10.1287/ijoc.2015.0666

Full terms and conditions of use: http://pubsonline.informs.org/page/terms-and-conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2016, INFORMS

松弛处理



原问题

$$\text{minimize} \quad C_{\max} \tag{21}$$

$$\text{s.t.} \quad \sum_{j \in N} x_{ij} p_{ij} + \xi_i \le C_{\max}, \quad i \in M, \tag{22}$$

$$\sum_{i \in M} x_{ij} = 1, \quad j \in N, \tag{23}$$

$$x_{i0} = 1, \quad i \in M, \tag{24}$$

$$\xi_i = \sum_{j \in N_0} \sum_{k \in N_0} y_{ijk} s_{ijk}, \quad i \in M, \tag{25}$$

$$x_{ik} = \sum_{j \in N_0} y_{ijk}, \quad k \in N_0;\ i \in M, \tag{26}$$

$$x_{ij} = \sum_{k \in N_0} y_{ijk}, \quad j \in N_0;\ i \in M, \tag{27}$$

$$C_k - C_j + V(1 - y_{ijk}) \ge s_{ijk} + p_{ik}, \quad j \in N_0,\ k \in N,\ i \in M, \tag{28}$$

$$C_0 = 0, \tag{29}$$

$$x_{ij} \in \{0; 1\}, \quad j \in N;\ i \in M, \tag{30}$$

$$y_{ijk} \in \{0; 1\}, \quad j, k \in N_0;\ i \in M, \tag{31}$$

主问题

$$\text{minimize} \quad C_{\max} \tag{32}$$

$$\text{s.t.} \quad \text{constraints (22)–(27)},$$

$$cuts, \tag{33}$$

$$x_{ij} \in \{0; 1\}, \quad j \in N;\ i \in M, \tag{34}$$

$$0 \le y_{ijk} \le 1, \quad j, k \in N_0;\ i \in M, \tag{35}$$

where

x_{ij}: 1 if job j is processed on machine i

y_{ijk}: Relaxed representation of job k being processed directly after job j on machine i.

子问题

分配方式固定后并行机调度转换为|M|个单机调度问题，每个单机调度问题的最优值都是原问题的一个上界



固定处理

套模板，设计分配与调度问题LBBD的通用分析割：

(A1) For any set $J' \subseteq J$, removing task j from J' reduces the optimal subproblem value $v^*(J')$ by at most α_j.

Then we have the valid cut

$$z \geqslant v^*(J) - \sum_{j \in J} \alpha_j (1 - x_j) \tag{3.9}$$

$$\max Pre_{hj} = \max_{k \in N_i'; k \neq j} (s_{i'kj})$$
$$\theta_{hi'j} = p_{ij} + \max Pre_{hj}$$

估计一下把原本分配给机器 i' 的工件 j 拿走机器 i' 最多节省多少时间

如果在机器 i' 上的分配有所改变，下界最多降低多少

$$C_{\max} \geq C_{\max}^{hi'*} - \sum_{j \in N_i'^h} (1 - x_{i'j}) \theta_{hi'j}.$$

如果在机器 i' 上重复同样的分配，那么下界至少是这个数

近几年的LBBD大家咋用的?

论文：An exact algorithm for integrated planning of operations in dry bulk terminals
作者：Ozgur Unsal, Ceyda Oguz

- 主问题增强Master problem enhancements(MPEs)
- 提出分析可行割割证明并有效性

论文：Integrated optimization of multi-machine scheduling and berth allocation in automated ore terminals
作者：Zhendi Gao, Mingjun Ji , Lingrui Kong

- 主问题增强Master problem enhancements(MPEs)
- 提出分析可行割割证明并有效性

论文：Joint berth allocation and ship loader scheduling under the rotary loading mode in coal export terminals
作者：Zhen Cao , Wenyuan Wang , Ying Jiang , Xinglu Xu , Yunzhuo Xu , Zijian Guo

- 主问题增强Master problem enhancements(MPEs)
- 介绍子问题生成的可行割和最优割并证明其有效性
- 设计benders割增强启发式，这个设计思路就是将当前割缩紧成不可约割
- 证明算法能够在有限步收敛到最优解

汇报完毕
感谢各位的聆听