arXiv:2301.01844v1 [math.OC] 4 Jan 2023

利用决策图求解不可拆分网络流问题

HosseinaliSalemi, Danial Davarnia工业与制造系统工程系，艾姆斯，爱荷华州立大学，艾姆斯，爱荷华州 50011，hsalemi@iastate.edu，davarnia@iastate.edu

在不可拆分网络流问题中，某些节点必须满足一项组合要求：入弧上的流经过出弧进行路由时，不得拆分或合并。这一所谓的 *no-splitno-merge* 要求出现在单元列车调度中：在缺少必要设备和人力、无法进行列车编组连接或拆解的车站，列车编组必须保持完整。由于需要大量二元变量来确定不拆分、不合并约束下节点入弧与出弧之间的匹配对，采用标准混合整数规划公式求解不可拆分网络流问题在计算上十分困难。本文研究需求不确定情况下单元列车调度问题的随机变体。我们提出一种新颖的基于决策图（DD）的框架，将底层的两阶段 formulation 分解为一个包含组合要求的主问题，以及一个对连续网络流问题建模的子问题。主问题通过在维度更低的变量变换空间中构建决策图来建模，从而显著缩短求解时间。与 Benders 分解技术类似，子问题会生成用于细化主决策图的割平面。计算实验表明，与标准方法相比，决策图框架显著缩短了求解时间。

Key words：决策图；网络优化；混合整数规划；单元列车；运输*History*：

1. 引言

过去几十年来，铁路货运一直是大宗商品运输的主要方式，并持续发展。与其他运输方式相比，铁路运输具有可靠、安全、成本效益高且环境可持续等优势。从规模上看，2018年，铁路网络承运的货运量占美国货运总吨英里数的 27.2%（Furchtgott-Roth 等，2021年）；见图 1。美国联邦公路管理局估计，到 2040 年，美国货运总量将达到 241 亿吨，比 2018 年 186 亿吨的运输总量增长 30%。为满足市场的这一增长需求，美国货运铁路公司已投入近

7400 亿美元用于资本支出和维护（1980年至2020年）（美国铁路协会，2021年）。

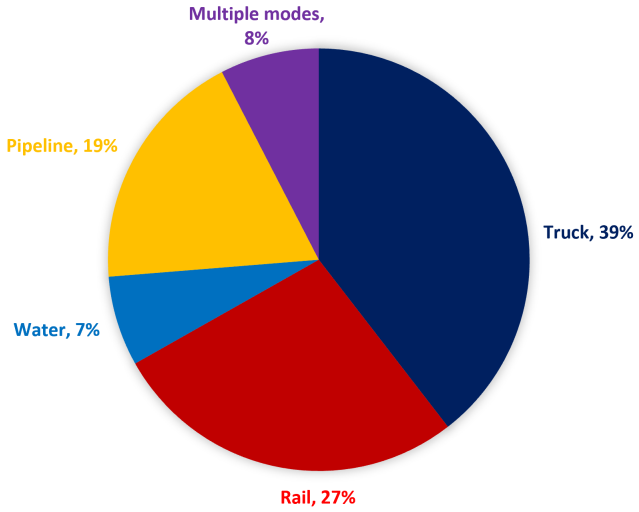


图1 2018年美国境内各运输方式货运量吨英里数饼图。多种运输方式包括邮件。占比为0.1% **的航空和卡车—航空运输方式未列出。**

为降低铁路货运的运输成本和货物延误，铁路公司为大批量产品提供单元列车 服务。单元列车运输单一类型的货物，货运列车从起点驶往目的地的途中不会挂接或摘下任何车厢，除非在配备了所需人力和机械设备的特定地点进行操作。这类列车通常全天运行，使用专用设备，并可在24小时内完成装卸。它们被认为是铁路运输中速度最快、效率最高的方式之一。（美国铁路协会，2021年）。传统上，单元列车用于运输煤炭、谷物、水泥和岩石等散装货物。原油等散装液体，以及小麦和玉米等食品也通过单元列车运输。根据美国联邦铁路管理局的数据，散装商品占美国铁路货运量的91%。美国通过铁路运输的煤炭几乎全部由单元列车承运。此外，这类列车在原油运输中也发挥着重要作用，因为每列单元列车可运载85,000桶原油（美国铁路协会，2021年）。从运营层面来看，单元列车模型的核心内容可描述如下：给定铁路网络中的一组供应点、中间地点和需求点，单元列车调度问题旨在寻找单元列车的最优路线，将流从供应点运送至需求点，同时在满足客户需求、遵守轨道容量限制并满足特定地点不得挂接或摘下车厢的要求的前提下，使运输总成本最小化。因此，制定编组计划 以确定地点

在这些地点需要将车厢从一列列车转至另一列列车，这一点与调度其他类型列车的问题不同（Davarnia 等，2019），但与本问题无关。

尽管单元列车调度意义重大，但用于求解相关问题的精确优化方法仍然不多，部分原因在于其结构复杂。在单元列车建模中，一项主要挑战是：列车编组经过缺少必要解编设备的车站时，必须保持完整。在优化领域，这一要求称为禁止拆分与合并 （NSNM），它确保进入或离开单元列车网络中某些节点的流不能拆分或合并。将这一要求纳入典型的运输网络模型后，便得到所谓的广义不可分流问题 （GUFP），其目标是确定满足给定需求的最低成本单元列车调度方案。大量研究表明，采用确定性需求可能导致运输调度彻底失败（Demir 等，2016 年；Layeb 等，2018 年），这促使研究人员关注需求不确定情况下的单元列车调度问题随机变体。因此，本文考虑 GUFP 的一种随机变体，称为 SGUFP，并将其建模为一个两阶段优化问题。第一阶段确定铁路网络节点的入弧与出弧之间的匹配，第二阶段则确定应通过网络匹配弧发送多少流量，以满足由若干需求情景所表示的不确定需求。我们提出了一种新颖的精确求解框架，用于在运营层面求解该问题。

我们提出的方法基于决策图 （DD），这是一种紧凑的图形数据结构。决策图最初用于表示布尔函数，并应用于电路设计。过去十年间，研究人员通过开发基于决策图的算法来求解不同应用领域中的离散优化问题，成功拓展了决策图的应用范围。由于决策图在结构上仅适用于对整数规划建模，因此从未用于求解本质上包含连续变量的运输问题。本文通过提出一种新框架，拓展了决策图的应用范围。该框架能够像 SGUFP 一样，对同时包含整数和连续部分的网络问题进行建模。

1.1. 列车调度文献综述

文献中已经提出并广泛研究了许多不同变体的列车路径规划和调度问题，这些问题在确定性和随机条件下具有不同的目标函数和约束集合；关于不同问题的分类和结构，可参见 Cordeau、Toth 和 Vigo（1998 年）、Harrod 和 Gorman（2010 年）、Lusby 等（2011 年）、Cacchiani 和 Toth（2012 年）以及 Turner 等（2016 年）的综述。混合整数线性和非线性规划模型是用于刻画这些问题不同类别的最常见精确方法之一（Jovanovi´c 和 Harker 1991年、Huntley 等 1995年、Sherali 和 Suharko 1998年、Lawley 等，

2008年、Haahr 和 Lusby 2017年、Davarnia 等 2019年）。所提出的求解技术包括但不限于分支定界方法（Jovanovi´c 和 Harker 1991年、Fuchsberger 和 L¨uthi 2007年）、分支切割框架（Zwaneveld、Kroon 和 Van Hoesel 2001年、Ceselli 等 2008年）、分支定价方法（Lusby 2008年、Lin 和 Kwan 2016年）、图着色算法（Cor-nelsen 和 Di Stefano 2007年）以及启发式算法（Carey 和 Crawford 2007年、Liu 和 Kozan 2011年、I.cy¨uz 等 2016年）。机车车辆调度（Abbink 等 2004年、Alfieri 等 2006年、Haahr 等 2016年、Bornd¨orfer 等 2016年）将机车车辆分配给给定的时刻表；乘务员调度（Kwan 2011年、Shen 等 2013年、Heil、Hoffmann 和 Buscher 2020年）则通过为相关作业分配乘务员来覆盖列车活动。这些都是铁路规划领域中出现的其他主要问题。

由于不同类型的列车调度与路径规划问题本身具有不确定性，许多研究者探讨了供需存在不确定性的随机变体。Jordan 和 Turnquist（1983年）提出了一种铁路货车分配模型，其中货车的供应和需求均存在不确定性。Jin 等人（2019年）研究了随机需求下列车停站规划问题的机会约束规划模型。Ying、Chow 和 Chin（2020年）提出了一种深度强化学习方法，用于解决乘客需求不确定情况下的列车调度问题。近期，Gong 等人（2021年）提出了一种随机优化方法，用于求解乘客需求不确定的列车时刻表编制问题。另请参阅 Meng 和 Zhou（2011年）、Quaglietta、Corman 和 Goverde（2013年）以及 Larsen 等人（2014年）的研究，这些研究探讨了随机环境下的列车调度问题。

在单元列车调度的背景下，Lawley 等（2008）研究了一种时空网络流模型，用于安排单元列车的铁路大宗货物运输。在他们的模型中，作者考虑了底层铁路网络的特征、客户需求，以及轨道和车站的容量与装卸要求。他们提出了一种混合整数规划（MIP）公式，旨在最大限度地满足需求，同时缩短车站的等待时间。Lin 和 Kwan（2014）（参见 Lin 和 Kwan（2016））提出了一种列车调度问题模型，能够涵盖禁止编组与解编的位置。他们设计了一种受列生成启发的分支定价算法，以求解相关问题。Lin 和 Kwan（2018）还提出了一种启发式分支定界方法，以减少编组与解编的冗余。I.cy¨uz 等（2016）研究了煤炭整列列车的规划问题，其中包括列车编组、流路由和调度。正如作者所指出的，由于规模庞大，他们提出的 MIP 公式无法直接求解该问题。为此，他们设计了一种高效启发式方法，能够生成高质量的解。最近，Davarnia 等人（2019）提出并研究了 GUFP，并将其应用于单元列车调度。具体而言，作者展示了如何在网络优化问题中施加 NSNM限制。他们进行了多面体研究，并提出一种 MIP 公式，用于建模经过简化的单元列车调度问题变体。本文采用他们的公式（见第 3.1 节）作为求解框架的基础。

不可分流问题（UFP）最初由 Kleinberg（1996年）提出，是对不相交路径问题的推广。给定一个具有弧容量的网络，以及一组带有相应需求和奖励的源点-终端点顶点对，UFP 的目标是选择其中一部分源点-终端点顶点对，并为每一对通过一条单一路径路由流，以满足相应需求，从而使总收益最大化。然而，在 GUFP 中，某些节点可以不遵守 NSNM 要求，且需求可以通过多条路径上的流来满足。众所周知，UFP 的不同变体均为 NP 难问题（Baier、K¨ohler 和 Skutella，2005年；Kolman 和 Scheideler，2006年；Chakrabarti 等，2007年）。自提出以来，UFP 结构已应用于多个领域，从异构网络中的带宽分配（Kolman 和 Scheideler，2006年），到可生存的面向连接网络（Walkowiak，2006年）以及虚电路路由问题（Hu、Lan 和 Wan，2009年）。鉴于该问题的求解难度，近似算法已成为文献中处理 UFP 不同变体的常用方法（Baier、K¨ohler 和 Skutella，2005年；Chakrabarti 等，2007年）。

1.2. 决策图文献综述

决策图是有向无环图，包含一个源点和一个终端节点，其中每条源点-终端路径都编码了一个优化问题的可行解。在决策图中，从源点到终端节点的每一层都表示一个决策变量，而弧上的标签则表示该变量的取值。Had˘zi´c 和 Hooker（2006年）提出使用决策图对离散优化问题的可行域建模，并将其用于后最优性分析。随后，Andersen 等人（2007）提出了松弛决策图，以应对在对大型离散优化问题建模时决策图规模的指数增长。Bergman 等人（2016年b）提出了一种分支定界算法，通过迭代使用松弛决策图和受限决策图来寻找最优解。文献中有许多在不同领域成功应用决策图的案例；例如，Bergman 和 Cire（2018年）、Serra 和 Hooker（2019年）、Davarnia 和 Van Hoeve（2020年）、Gonzalez 等人（2020年）以及 Hosseininasab 和 Van Hoeve（2021年）的研究。

直到最近，决策图的应用还仅限于离散问题，如何利用决策图求解含连续变量的优化问题仍未得到解答。为克服这一局限，Davarnia（2021）提出了一种称为弧缩减的技术，用于生成表示底层连续问题松弛的决策图。在后续研究中，Salemi 和 Davarnia（2022a）建立了一般混合整数规划可由决策图表示的充要条件。他们表明，通过采用专门的 Benders 分解技术，可以对有界混合整数规划重新建模，并利用决策图求解。本文在这一框架的基础上，设计了一种新颖的基于决策图的方法来求解 SGUFP。

1.3. 研究贡献

尽管文献中有多项研究专门探讨单元列车问题，但能够严谨处理单元列车模型核心的 NSNM 要求的精确方法仍然少见。本文设计了一种新颖的精确决策图框架，以求解 SGUFP；与该类问题相比，SGUFP 是一种更现实、也更具挑战性的变体。据我们所知，这是首篇从精确求解角度研究 SGUFP 的工作，也是决策图在运输问题中的首次应用。我们提出的框架在变量的变换空间中表述该问题，其维度小于 SGUFP 的标准混合整数规划表述。这样的表述减轻了由混合整数规划表述规模导致的计算困难，为求解大规模网络问题提供了可行的方法。决策图框架的核心原理也可用于建模结构相似的其他运输问题，作为传统网络优化技术的替代方案。

本文其余部分安排如下。第 2 节给出基本定义，并简要概述离散和连续决策图模型，包括用于求解有界混合整数规划的 DD-BD 方法。第 3 节将 DD-BD 方法改造为求解 SGUFP。我们提出算法，在变换空间中构建精确和松弛决策图，以求解该问题。第 4 节通过计算实验评估 DD-BD 方法在求解 SGUFP 时的性能。第 5 节给出总结性意见。

2. 决策图背景

本节将介绍与决策图分析相关的基本定义和结果。

2.1. 概述

一个决策图 $\mathcal{D}=(\mathcal{U},\mathcal{A},l)$，其节点集为 $\mathcal{U}$、弧集为 $\mathcal{A}$，弧标签映射为 $l\colon \mathcal{A}\to\mathbb{R}$是一个有向无环图，包含 $n\in\mathbb{N}$ 个弧层 $\mathcal{A}_1,\mathcal{A}_2,\dots,\mathcal{A}_n$和 $n+1$ 个节点层 $\mathcal{U}_1,\mathcal{U}_2,\dots,\mathcal{U}_{n+1}$。节点层$\mathcal{U}_1$和 $\mathcal{U}_{n+1}$分别包含根节点 r和终端节点 t，其中 $|\mathcal{U}_1|=|\mathcal{U}_{n+1}|=1$。在任一弧层 $j\in[n]:=\{1,2,\dots,n\}$中，弧$(u,v)\in\mathcal{A}_j$ 从尾节点 $u\in\mathcal{U}_j$ 指向头节点$v\in\mathcal{U}_{j+1}$。 $\mathcal{D}$ 的宽度定义为其最大 $\mathcal{U}_j$ 的大小。决策图可以对有界整数集 $\mathcal{P}\subseteq\mathbb{Z}^n$建模，使得每个形如 $(a_1,\dots,a_n)\in\mathcal{A}_1\times\dots\times\mathcal{A}_n$的 r-t弧序列（路径）都编码一个点 $\boldsymbol{y}\in\mathcal{P}$，其中 $l(a_j)=y_j$，对 $j\in[n]$均成立；也就是说， $\boldsymbol{y}$是 $\mathcal{P}$中的一个 n维点，其j坐标等于弧 a_j 的标签值 $l(a_j)$。对于这样的决策图，我们有 $\mathcal{P}=$Sol($\mathcal{D}$)，其中 Sol($\mathcal{D}$) 表示所有 r-t 路径构成的有限集合。

可以利用决策图的图形性质，在离散集合 $\mathcal{P}$ 上优化目标函数。为此，对决策图弧进行加权，使得编码解 $\boldsymbol{y}\in\mathcal{P}$ 的 r-t路径的累积权重等于在 $\boldsymbol{y}$ 处计算的目标函数值。然后，

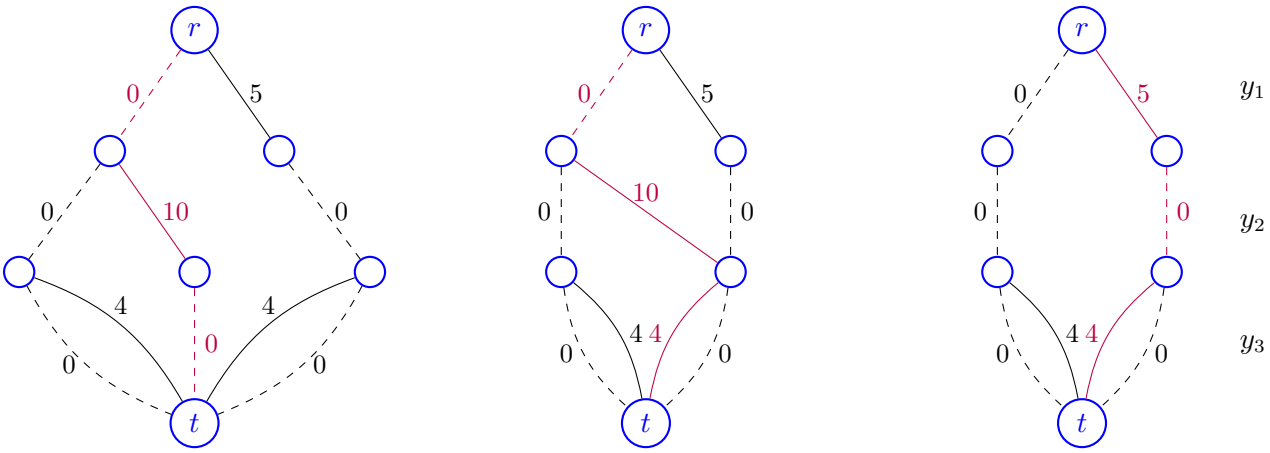
对于相应的最小化（最大化）问题，求出最短（最长）的 r-t 路径；这一操作可以在多项式时间内完成。

如上所述，构造精确决策图在计算上难以承受，因为其规模呈指数增长。为缓解这一难题，提出松弛和受限决策图，以控制决策图的规模。在松弛决策图中，节点以特定方式合并，使所得图的宽度不超过预先设定的宽度限制。这一节点合并过程确保原始集合的所有可行解都由所得决策图中所有 r-t 路径的一个子集编码。对该松弛决策图进行优化，可为原问题的最优解提供对偶界。在受限决策图中，该决策图的所有 r-t 路径所构成的集合编码了原始集合可行解的一个子集。对该受限决策图进行优化，可为原问题的最优解提供原始界。通过分支定界方案迭代细化受限决策图和松弛决策图，可使它们的原始界和对偶界逐步收敛，从而求得问题的最优值。以下示例展示了一个离散优化问题的精确、松弛和受限决策图。

EXAMPLE 1. 考虑离散优化问题 $\max\{5y_1 + 10y_2 + 4y_3 \mid \boldsymbol{y} \in \mathcal{P}\}$ 其中$\mathcal{P}$ = {(1,0, 0),(1, 0,1),(0,1,0), (0,0,1),(0, 0, 0)}。图 2(a) 中宽度为 3 的精确决策图 $\mathcal{D}$对可行域 $\mathcal{P}$进行了建模。每条弧 $a \in \mathcal{A}_j$的权重（其中 $j \in \{1,2,3\}$）表示变量 y_j的取值赋给目标函数所带来的贡献。编码最优解的最长路径 r-t (y_1^*, y_2^*, y_3^*) =(0, 1, 0) 的长度为 10，即该问题的最优值。将宽度限制降至 2 后，我们可以为 $\mathcal{P}$ 构建如下松弛决策图和受限决策图。图 2(b) 中的松弛决策图 $\mathcal{D}$ 为最优解提供了一个上界，其中长度为 14 的最长路径对应一个不可行点(y_1, y_2, y_3) =(0, 1, 1)。最后，图 2(c) 中的受限决策图 $\mathcal{D}$ 为最优解提供了一个下界，其中长度为 9 的最长路径编码了一个可行解 (y_1, y_2, y_3) =(1, 0, 1)。

2.2. 连续决策图模型

虽然前一节所述的框架可以应用于解决不同类别的离散优化问题，但将其扩展到包含连续变量的集合模型需要采用根本不同的方法。传统 DD结构之所以不适用于连续集合，是因为通过弧表示连续变量的定义域需要无穷多个弧，覆盖连续区间内的所有值，而这在决策图中会造成结构上的不可行。幸运的是，可以通过将底层集合分解为某些矩形结构来克服这一障碍，而这些矩形结构又可以通过决策图中的节点序列来表示。下文将概述与我们的分析相关的这些结果。



(a) 精确决策图 $\mathcal{D}$　(b) 松弛决策图 $\overline{\mathcal{D}}$　(c) 受限决策图 $\underline{\mathcal{D}}$

图 2 示例 1 中表示 $\mathcal{P}$ 的精确决策图、松弛决策图和受限决策图。实线弧和虚线弧分别表示弧标签为 1 和 0。弧旁的数字表示权重。

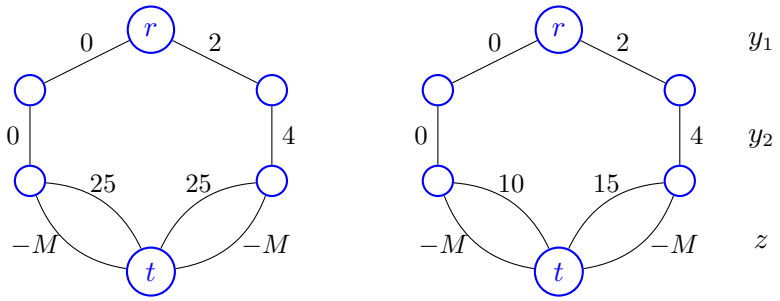
考虑一个有界集合 $\mathcal{P} \subseteq \mathbb{R}^n$。Salemi 和 Davarnia（2022a）给出了 $\mathcal{P}$ 满足所需矩形分解的充要条件。如果存在一个决策图 $\mathcal{D}$，使得对于变量 $\boldsymbol{x}_I$ 所构成空间中为凸的每个函数 $f(\boldsymbol{x})$，均有 $\max\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in \mathcal{P}\} = \max\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in \text{Sol}(\mathcal{D})\}$，则称该集合关于固定指标集 $I \subseteq [n]$ 是 决策图可表示 的。下面给出决策图可表示集合的一个特例。

PROPOSITION 1. 任何具有形式 $\mathcal{P} \subseteq \mathbb{Z}^n \times \mathbb{R}$ 的有界混合整数集合均可关于 $I = \{n+1\}$ 由决策图表示。□

这一结果带来了一种新颖的基于决策图的框架，可用于求解一般的有界混合整数规划，具体如下。考虑一个有界混合整数规划 $\mathcal{H} := \max\{\boldsymbol{cy} + \boldsymbol{dx} \mid A\boldsymbol{y} + G\boldsymbol{x} \leq \boldsymbol{b},\ \boldsymbol{y} \in \mathbb{Z}^n\}$。利用 Benders 分解（BD），公式 $\mathcal{H}$ 等价于 $\max_{\boldsymbol{y} \in \mathbb{Z}^n}\{\boldsymbol{cy} + \max_{\boldsymbol{x}}\{\boldsymbol{dx} \mid G\boldsymbol{x} \leq \boldsymbol{b} - A\boldsymbol{y}\}\}$，而后者可重构为 $\mathcal{M} = \max\{\boldsymbol{cy} + z \mid (\boldsymbol{y};\ z) \in \mathbb{Z}^n \times [l, u]\}$，其中 l，$u \in \mathbb{R}$ 是根据 z有界性推导出的有效界。这里，$\mathcal{M}$ 是主问题，而对于任意给定的 $\bar{\boldsymbol{y}}$，z 表示子问题 $\max_{\boldsymbol{x}}\{\boldsymbol{dx} \mid G\boldsymbol{x} \leq \boldsymbol{b} - A\bar{\boldsymbol{y}}\}$ 的目标值，其中 $\bar{\boldsymbol{y}}$ 是主问题的一个最优解。子问题的结果要么是最优性割，要么是可行性割，这些割将被加入主问题。随后，将重新求解主问题。命题 1 表明，公式 $\mathcal{M}$ 可以直接用决策图建模并求解。对于此决策图，我们为整数变量 y_1，y_2，…，y_n分配 n个弧层，并为连续变量 z分配一个弧层，该层只有两个弧标签，分别表示该变量的下界和上界。为找到最优解，需要计算最长路径，并用它来求解子问题。请注意，由于 $\mathcal{M}$ 是最大化问题，相应决策图中的最长路径编码了一个最优解，其长度即为最优值；参见例2。随后，将子问题生成的可行性割和最优性割加入决策图，以对其进行细化，并重新计算其最长路径。

细化技术包括移除决策图中会导向违反新增不等式的解的弧，同时拆分会导向不同后续部分赋值的决策图节点；有关决策图细化技术的详细介绍，请参见 Bergman 等人（2016a）。我们将在例2中说明这种方法。

EXAMPLE 2. 假设 $\max\{2y_1+4y_2+z \mid \boldsymbol{y}\in\mathcal{P}, z\le 25\}$ 构成 Benders 分解算法倒数第二次迭代的主问题，其中 $\mathcal{P}=\{(0,0),(1,1)\}$。该问题由图3(a)中的决策图 $\mathcal{D}$ 表示，其中 $-M$ 是 z 的有效下界。 $\mathcal{D}$的最长路径编码了该解（$\hat{}y_1$，$\hat{}y_2$，$\hat{}z$）=(1，1，25)。假设在关联的子问题中使用点（$\hat{}y_1$，$\hat{}y_2$）=(1，1)生成一个最优性割 $z\le 3y_1+2y_2+10$，用于 Benders 分解算法的最后一次迭代。根据该割细化决策图 $\mathcal{D}$，得到图3(b)中的新决策图。其最长路径表示最优解（y_1^*，y_2^*，z^*）=(1，1，15)，长度为21，即最优值。



(a) 倒数第二次迭代　　(b) 最后一次迭代

图3 例2中求解主问题的最后两次迭代

如上所述，使用决策图框架可能会面临计算上的挑战，因为精确决策图的规模呈指数级增长。为缓解这一困难，可以如算法1所示，在 Benders 分解框架中采用受限/松弛决策图。我们将这种求解方法称为 DD-BD（Salemi 和 Davarnia 2022a）。

在解释算法1的步骤时，令点 $\hat{}\boldsymbol{y}\in\mathbb{Z}^k$（$k\le n$）为变量 $\boldsymbol{y}$ 前 k 个坐标的部分取值赋值，即对所有 $i\in[k]$，$y_i=\hat{}y_i$。我们将所有部分取值赋值的集合记为 $\hat{}\mathcal{Y}=\{\hat{}\boldsymbol{y}\in\mathbb{Z}^k \mid k\in[n]\}\cup\{\ominus\}$，其中 $\ominus$ 表示 $\boldsymbol{y}$ 的所有坐标均未固定的情形。集合 $\mathcal{C}$ 包含算法执行过程中生成的所有 Benders 割，我们将这些割所描述的可行域记为 $\mathcal{F}^{\mathcal{C}}$。此外，定义 $\mathcal{M}^{C}(\hat{}\boldsymbol{y})=\max\{\boldsymbol{cy}+z \mid (\boldsymbol{y};z)\in \mathbb{Z}^n\times[l,u]\cap\mathcal{F}^{\mathcal{C}},\ y_i=\hat{}y_i,\ \forall i\in[k]\}$，表示通过加入 $\mathcal{C}$ 中的割并固定部分赋值 $\hat{}\boldsymbol{y}$ 得到的受限主问题 $\mathcal{M}$。在此定义中，$\mathcal{C}=\emptyset$ 且 $\hat{}\mathcal{Y}=\{\ominus\}$ 的情形记为 $\mathcal{M}^{\emptyset}(\ominus)=\mathcal{M}$，它是算法1的一个输入。

算法1： DD-BD

数据： 混合整数规划 $\mathcal{H}$，为 $\mathcal{M}$ 构建受限决策图和松弛决策图的构造方法

结果： 一个最优解（$\boldsymbol{y}^*$，z^*）以及 $\mathcal{H}$ 的最优值 w^*

1 初始化部分赋值集合 $\hat{\mathcal{Y}} = \{\ominus\}$，Benders 割集合 $\mathcal{C} = \emptyset$，以及 $w^* = -\infty$
2 **if** $\hat{\mathcal{Y}} = \emptyset$ **然后**
3 　终止并返回 $(\boldsymbol{y}^*, z^*)$ 和 w^*
4 **否则**
5 　选择 $\hat{\boldsymbol{y}} \in \hat{\mathcal{Y}}$ 并更新 $\hat{\mathcal{Y}} \leftarrow \hat{\mathcal{Y}} \setminus \{\hat{\boldsymbol{y}}\}$
6 　创建与 $\mathcal{M}^C(\hat{\boldsymbol{y}})$ 相关联的受限决策图 $\mathcal{D}$
7 　**如果** $\mathcal{D} \neq \emptyset$，**则**
8 　　查找 $\mathcal{D}$ 中具有编码点 $(\boldsymbol{y}, z)$ 且长度为 w 的最长 r-t 路径
9 　　使用 $\boldsymbol{y}$ 求解 BD 子问题，以获得 Benders 割 C
10 　　**如果** $C \in \mathcal{C}$，**则**
11 　　转到第 17 行
12 　　**else**
13 　　更新 $\mathcal{C} \leftarrow \mathcal{C} \cup C$ 并针对 C 细化 $\mathcal{D}$
14 　　转到第 8 行
15 　**else**
16 　　转到第 2 行
17 　**如果** $w > w^*$ **则**
18 　　更新 $w^* \leftarrow w$ 并 $(\boldsymbol{y}^*, z^*) \leftarrow (\boldsymbol{y}, z)$
19 　**如果** $\mathcal{D}$ 为 $\mathcal{M}^C(\hat{\boldsymbol{y}})$，**则**
20 　　转到第 2 行
21 　**else**
22 　　创建与 $\mathcal{M}^C(\hat{\boldsymbol{y}})$ 关联的松弛决策图 $\mathcal{D}$
23 　　查找 $\mathcal{D}$ 中长度为 w 的最长 r-t 路径
24 　　**如果** $w > w^*$，**则**
25 　　使用 $\boldsymbol{y}$ 求解 BD 子问题，以获得 Benders 割 C
26 　　**如果** $C \in \mathcal{C}$，**那么**
27 　　转到第 31 行
28 　　**else**
29 　　更新 $\mathcal{C} \leftarrow \mathcal{C} \cup C$ 并根据 C 细化 $\mathcal{D}$
30 　　转到第 23 行
31 　　**对所有** u，在 $\mathcal{D}$ 的最后一个精确层中**执行**
32 　　更新 $\hat{\mathcal{Y}} \leftarrow \hat{\mathcal{Y}} \cup \{\tilde{\boldsymbol{y}}\}$ 其中 $\tilde{\boldsymbol{y}}$ 表示 $\mathcal{D}$ 的最长 r-u 路径
33 　转到第 2 行

算法首先构造一个受限决策图 $\mathcal{D}$，对应于 $\mathcal{M}^C(\hat{\boldsymbol{y}})$，其 C 和$\hat{y}$的初始值为空。随后，我们找到 $\mathcal{D}$ 中一条最长的 r-t 路径，该路径编码了解 $\boldsymbol{y}$、z。接着，利用 $\boldsymbol{y}$求解相应的子问题，得到一个可行性/最优性割 C。我们将此割加入 $\mathcal{C}$，据此细化 $\mathcal{D}$，并找到一条新的最长 r-t 路径。重复这些步骤，直到不再生成新的可行性/最优性割。此时，$\mathcal{D}$中一条最长的 r-t 路径的长度记为 w，它给出主问题 $\mathcal{M}$的一个下界，该下界对原问题 $\mathcal{H}$也有效。w 的值可用于更新 w^*，即终止时 $\mathcal{H}$的最优值。接下来，我们构造一个松弛决策图 $\mathcal{D}$，对应于 $\mathcal{M}^C(\hat{\boldsymbol{y}})$。我们找到 $\mathcal{D}$ 中一条最长的 r-t 路径，它为 $\mathcal{M}$提供上界 w。如果上界 w 严格大于 w^*的当前值，我们便仿照处理 $\mathcal{D}$ 时的步骤，通过求解子问题，利用可行性/最优性割迭代细化 $\mathcal{D}$，直到不再生成新的割。接下来，我们执行一种专门的分支定界过程，通过扩展决策图中合并的层来改进界。为此，我们将D 最后一个精确层（即没有节点被合并的最后一个节点层）中各节点对应的所有部分赋值加入集合 $\mathcal{Y}$。需要进一步探索 $\mathcal{Y}$ 中与部分赋值对应的节点，以检查 w^* 的值能否得到改进。也就是说，在新的受限/松弛决策图中固定 v 的 r-v 路径后，对 $\mathcal{Y}$ 中部分赋值对应的每个节点重复上述过程。算法在 $\mathcal{Y}$ 变为空时终止；此时，w^* 即为最优值。

3. 面向 SGUFP 的 DD-BD 公式

本节将第 2.2 节所述的 DD-BD 框架进行调整，以求解 SGUFP。

3.1. MIP formulation

我们研究 SGUFP 的混合整数规划公式，该公式基于 Davarnia 等人（2019）给出的其确定性对应问题的公式。考虑一个网络 $G=(V,\ A)$，其节点集为 $V:=V'\cup\{s,t\}$，弧集为A，其中 s 和 t 分别为源节点和汇节点。源节点与 $S\subseteq V'$中的所有供应节点相连，汇节点与 $D\subseteq V'$中的所有需求节点相连。图 4 展示了该网络的一般结构。对于节点 $q\in V$，令 $\delta^-(q):=\{i\in V\mid(i,\ q)\in A\}$ 和 $\delta^+(q):=\{j\in V\mid(q,\ j)\in A\}$ 分别表示节点 q 的入邻居集和出邻居集。定义 $V\subseteq V'$ 为必须满足 NSNM 要求的顶点子集。对于每个节点 $q\in V$，令二元变量 $y_{ij}^q\in\{0,1\}$ 表示流是否通过弧 $(i,\ q)$ 进入节点 $q\in V$，并通过弧 $(q,\ j)$ 离开节点 q。SGUFP 的第一阶段确定不可分流节点的入弧与出弧之间的匹配对，具体如下：

$$\max z \tag{1a}$$

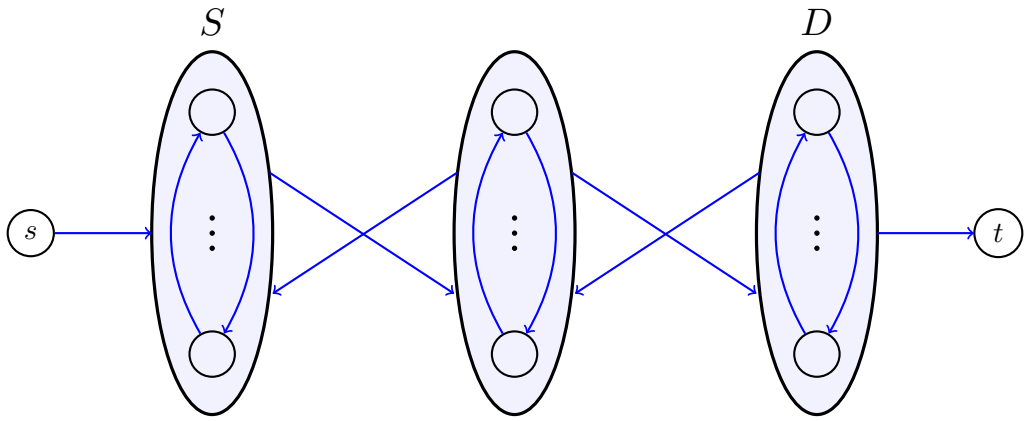


图 4 网络示意图 $G = (V' \cup \{s,t\},\ A)$

$$\text{s.t.} \quad \sum_{j\in\delta^+(q)} y_{ij}^q \le 1 \qquad \forall i \in \delta^-(q),\ \forall q \in \bar{V} \tag{1b}$$

$$\sum_{i\in\delta^-(q)} y_{ij}^q \le 1 \qquad \forall j \in \delta^+(q),\ \forall q \in \bar{V} \tag{1c}$$

$$y_{ij}^q \in \{0,1\} \qquad \forall (i,j) \in \delta^-(q) \times \delta^+(q),\ \forall q \in \bar{V}, \tag{1d}$$

其中，约束 (1b) 确保每条进入具有 NSNM 要求的节点的弧至多分配给一条出弧，约束 (1c) 则保证从此类节点引出的每条弧至多与一条入弧匹配。

在 (1a)–(1d) 中，变量 z 表示 SGUFP 第二阶段的目标值，其中考虑了需求不确定性。需求不确定性通过需求向量 $\boldsymbol{d}^\xi$ 的一组情景 Ξ 建模，每个情景 $\xi \in \Xi$的发生概率为 $\Pr^\xi$。令连续变量 $x_{ij}^\xi \in \mathbb{R}_+$ 表示在情景$\xi \in \Xi$下通过弧 (i,j) 从节点 i 流向节点 j 的流量。此外，我们为通过弧 (i,j) 路由的每单位流量赋予奖励 r_{ij}。由此可得 $z = \sum_{\xi\in\Xi} \Pr^\xi z^\xi$，其中 z^ξ 是每个情景 $\xi \in \Xi$下 SGUFP 第二阶段的目标值。给定一个 $\boldsymbol{y}$ 向量后，该子问题可表述如下：

$$\max \quad \sum_{q\in V} \sum_{j\in\delta^+(q)} r_{qj} x_{qj}^\xi \tag{2a}$$

$$\text{s.t.} \quad \sum_{i\in\delta^-(q)} x_{iq}^\xi - \sum_{j\in\delta^+(q)} x_{qj}^\xi = 0 \qquad \forall q \in V' \tag{2b}$$

$$\ell_{iq}^\xi \le x_{iq}^\xi \le u_{iq}^\xi \qquad \forall i \in \delta^-(q),\ \forall q \in V \tag{2c}$$

$$x_{iq}^\xi - x_{qj}^\xi \le u_{iq}^\xi (1 - y_{ij}^q) \qquad \forall (i,j) \in \delta^-(q) \times \delta^+(q),\ \forall q \in \bar{V} \tag{2d}$$

$$x_{qj}^\xi - x_{iq}^\xi \le u_{qj}^\xi (1 - y_{ij}^q) \qquad \forall (i,j) \in \delta^-(q) \times \delta^+(q),\ \forall q \in \bar{V} \tag{2e}$$

$$x_{iq}^\xi \le u_{iq}^\xi \sum_{j\in\delta^+(q)} y_{ij}^q \qquad \forall i \in \delta^-(q),\ \forall q \in \bar{V} \tag{2f}$$

$$x^{\xi}_{qj} \le u^{\xi}_{qj} \sum_{i \in \delta^-(q)} y^{q}_{ij} \qquad \forall j \in \delta^+(q),\ \forall q \in \bar{V} \tag{2g}$$

$$x^{\xi}_{ij} \ge 0 \qquad \forall (i,j) \in A. \tag{2h}$$

在上述公式中，目标函数表示通过在网络中（从源点 s 到汇点 t）路由流量以满足需求而获得的总奖励。流平衡要求由 (2b) 表示。约束 (2c) 对每条弧上的流量设置上下界。为满足每个情景 $\xi \in \Xi$ 的需求要求，我们对所有需求为 d^{ξ}_{q} 的需求节点 $q \in D$ 固定 $\ell^{\xi}_{qt} = u^{\xi}_{qt} = d^{\xi}_{q}$，并保持所有 $^-$ 其他弧的上下界不变。约束 (2d)–(2g) 对每个节点 $q \in V$的 NSNM 要求进行建模。具体而言，(2d) 和 (2e) 确保匹配弧 (i,q) 和 (q,j) 的流量相等。约束 (2f) 和 (2g) 保证不具有匹配弧的弧不承载任何流量。需要指出的是，在假设 $\boldsymbol{y}$ 满足主问题 (1a)–(1d) 的可行性的前提下，约束 (2b) 可由上述子问题的其他约束推出。然而，我们仍在子问题中保留该约束，因为基于DD的方法中的主问题 formulation（将在第3.2节中介绍）可能会产生一个不满足 (1a)–(1d) 的解。因此，加入约束 (2b) 将使子问题公式更加紧致。

如第 2.2 节所述，使用 DD-BD 算法的第一步是将基础问题分解为一个主问题和一个子问题。SGUFP 的上述两阶段 formulation 很容易适用于 Benders 分解，因为第一阶段问题 (1a)-(1d) 可视为主问题，并结合由混合整数规划 formulation 的有界性导出的关于 z 的有效上下界 $-\Gamma$ and Γ。对于从主问题获得的给定 $\boldsymbol{y}$ 值以及一个情景 $\xi \in \Xi$，第二阶段问题 (2a)-(2h) 可视为所需的子问题。然后，如第 3.3 节所述，通过聚合将各基于情景的子问题所得的最优性割/可行性割添加到主问题中。

3.2. DD-BD：主问题 formulation

尽管 DD-BD算法1 为任意有界混合整数规划提供了通用的求解框架，但其 DD组件具有问题特定性，也就是说，应根据基础问题的具体结构精心设计。本节将为 SGUFP 设计这样一个预言机，以表示主问题 (1a)-(1d) 的可行域 $\{$(1b) –(1d) $z \in [-\Gamma, \Gamma]\}$。若要在 $(\boldsymbol{y}, z)$ 变量的原始空间中对该可行域建模，决策图需要 $\sum_{q \in \bar{V}} |\delta^-(q)| \times |\delta^+(q)|$ 个弧层来表示二元变量 $\boldsymbol{y}$，并需要一个弧层来编码连续变量 z。然而，由于弧层数量庞大，构建这样的决策图在计算上会十分繁重。为缓解这一困难，我们利用决策图在表示不规则 变量类型方面的结构灵活性，而这类变量无法用于标准 MIP模型。其中一种类型是

指标集，弧层表示的是指标而非定义域取值。接下来我们将展示，通过在定义于指标集上的变量变换空间中重新表述主问题，可以显著减少决策图的弧层数量。

¯考虑一个节点 $q \in V$。下面，我们定义映射，为 q 的每条入弧和出弧分配一个指标。这些映射使我们能够定义新变量，以减少决策图的弧层数。令 $\text{ind}^-(i,q)$ 为从入弧 (i,q) 到指标集 $\{1,2,\ldots,|\delta^-(q)|\}$ 的一一映射，其中 $i \in \delta^-(q)$。类似地，令 $\text{ind}^+(q,j)$ 为从出弧 (q,j) 到指标集 $\{1,2,\ldots,|\delta^+(q)|\}$ 的一一映射，其中 $j \in \delta^+(q)$。对于指标为 $h = \text{ind}^-(i,q)$ 的每条入弧 (i,q)，我们定义一个整数变量 $w_h^q \in \{0,1,\ldots,|\delta^+(q)|\}$，使得 $w_h^q = 0$ 表示该入弧未与任何出弧配对，而 $w_h^q = k > 0$ 表示该弧与指标为 $k = \text{ind}^+(q,j)$ 的出弧 (q,j) 配对。

接下来，我们给出一个在 $\boldsymbol{w}$ 变量空间中的公式，用于描述¯对所有 $q \in V$的入弧与出弧之间的匹配。下文中，sign(.)表示符号函数：当其参数严格为正时返回 1，为零时返回 0，否则返回 -1 。此外，运算符 $|.|$作用于集合时表示集合大小；作用于实数时表示绝对值。

PROPOSITION 2. 公式

$$\sum_{i \in \delta^-(q)} \text{sign}\left(\left|w^q_{\text{ind}^-(i,q)} - \text{ind}^+(q,j)\right|\right) \geq \left|\delta^-(q)\right| - 1 \qquad \forall j \in \delta^+(q),\ \forall q \in \bar{V} \tag{3a}$$

$$w^q_{\text{ind}^-(i,q)} \in \left\{0,1,\ldots,\left|\delta^+(q)\right|\right\} \qquad \forall i \in \delta^-(q),\ \forall q \in \bar{V} \tag{3b}$$

¯对节点 $q \in V$的入弧与出弧之间的匹配进行建模。

¯证明。我们针对单个节点 $q \in V$ 证明该结果。由于每个节点的匹配问题彼此独立，多节点情形的扩展是直接的。对于直接蕴含，假设 M^q 是 q 的入弧与出弧之间的匹配，其中元素形式为 (i,j)，表示入弧 (i,q) 与出弧 (q,j) 之间的匹配。我们证明，与 M^q 中匹配对相关联的变量 $\boldsymbol{w}$ 满足约束 (3a) 和 (3b)。根据 $\boldsymbol{w}$ 的定义，对于每个 $(i,j) \in M^q$，有 $w^q\text{ind}^-(i,q) = \text{ind}^+(q,j)$。此外，对于任意在 M^q 中没有匹配对的 $i \in \delta^-(q)$，有 $w^q\text{ind}^-(i,q) = 0$。这些赋值表明，$\boldsymbol{w}$ 满足 (3b)，因为 ind^+ 映射的像为 $\{1,\ldots,|\delta^+(q)|\}$。对于每个 $i \in \delta^-(q)$ 和 $j \in \delta^+(q)$，有 $\left|w^q\text{ind}^-(i,q) - \text{ind}^+(q,j)\right| \geq 0$，当 $(i,j) \in M^q$ 时等号成立。对于每个 $j \in \delta^+(q)$，有两种情况。第一种情况下，假设对于任意 $i \in \delta^-(q)$，$(i,j) \notin M^q$。因此，对于所有 $i \in \delta^-(q)$，$\left|w^q\text{ind}^-(i,q) - \text{ind}^+(q,j)\right| > 0$。对这些项应用 sign(.) 函数，得到 $\text{sign}\left(\left|w^q\text{ind}^-(i,q) - \text{ind}^+(q,j)\right|\right) = 1$，这意味着

$\sum_{i\in\delta^-(q)}$ sign $\left(\left|w^q \text{ind}^-(i,q) - \text{ind}^+(q,j)\right|\right) = |\delta^-(q)|$，满足 (3a)。第二种情况下，假设对于某个 ( )，($i^*\ j\ \in M^q$ $i^* \in \delta^-\ q$ $\sum_{i\in\delta^-\ q}$ $\left(\left|w^q\ _{-\ i\ q} - \ ^+ q\ j\ \right|\right) =$,)。因此，我们有 $|\delta^-\ q\ | - 1$ $_-$ $-$ $^+ q\ j$ $_-$ $-$ $^+ q\ j$ 3 ( ) sign ind (,) ind (,) ( )，因为 sign $\left(\left|w^q \text{ind}\ (i^*,q)\ \text{ind}\ (\ ,\ )\right|\right) = \left|w^q \text{ind}\ (i^*,q)\ \text{ind}\ (\ ,\ )\right| = 0$，满足 (a)。

对于逆向蕴含，假设 $\boldsymbol{w}$ 是 (3a)–(3b) 的可行解。我们证明，由这些变量编码的 (i,j) 形式的配对构成了 q 的入弧与出弧之间的可行匹配，即：(i) 每条弧 (i,q) 至多与一条弧 (q,j) 匹配；(ii) 每条弧 (q,j) 至多与一条弧 (i,q) 匹配。由约束 (3b) 可知，对于每个 $i \in \delta^-(q)$，变量 $w^q \text{ind}^-(i,q)$ 的取值介于 $\{0,1,\ldots,|\delta^+(q)|\}$ 之间。如果 $w^q \text{ind}^-(i,q) = 0$，则 (i,q) 未与任何出弧匹配；否则，它将与弧 (q,j) 匹配，且 $\text{ind}^+(q,j) = w^q \text{ind}^-(i,q)$。这确保了上述匹配集合满足条件 (i)。此外，对于每个$j \in \delta^-(q)$，约束 (3a) 意味着，符号 $\left(\left|w^q \text{ind}^-(i,q) - \text{ind}^+(q,j)\right|\right)$ 对于至多一个 $i \in \delta^-(q)$ 可以等于零。在这种情况下，集合中至多存在一个 (i,j) 形式的匹配对，从而表明上述匹配集合满足条件 (ii)。□

由命题 2 可知，在主问题 (1a)-(1d) 中，可用约束 (3a)-(3b) 替代 (1b)-(1d)，从而得到以下变量变换空间中的主问题。

$$\max_{\boldsymbol{w};z} \{z \mid (3\text{a}) - (3\text{b}), z \in [-\Gamma, \Gamma]\}. \tag{4}$$

请注意，公式 (4) 是一个整数非线性规划（INLP），其约束函数非凸且不连续。对于传统的混合整数非线性规划技术和求解器而言，这种公式极难处理。然而，由于决策图在表示整数非线性规划方面具有结构灵活性，这个问题可以很容易地通过决策图建模；关于使用决策图对整数非线性规划建模的详细介绍，请参见 Davarnia 和 Van Hoeve（2020）。下面，我们介绍一种算法，用于在 $(\boldsymbol{w};z)$ 变量空间中为含有一个$^-$节点 $q \in V$的主问题 (4) 构建决策图。通过复制决策图结构，即可将其扩展到含有多个节点的情况。算法 2 的输出是一个包含 $|\delta^-(q)| + 1$ 个弧层的决策图，其中前 $|\delta^-(q)|$层表示 $\boldsymbol{w}$ 变量，最后一层对变量 z进行编码。在该算法中，s_u 表示决策图节点 u的状态值。算法的核心思想是，将 q 未配对的出弧用作每个决策图层的状态值，该层表示与 q入弧的匹配。

接下来，我们证明由算法 2 构造的决策图的解集表示(4) 的可行域。需要注意的是，如第 2.2 节所述，混合整数规划集合的决策图表示并不意味着对该集合所有解的编码，而是对所有解的一个子集进行编码，该子集包含该集合的所有极点。对于目标函数关于连续变量为凸函数的情形，这种表示足以求解该集合上的优化问题，而 (4) 正属于这种情形。

算法 2： 为带有节点 $q \in$ 的 SGUFP 主问题构造决策图 $\bar{V}$

数据： 节点 $q \in \bar{V}$，参数 Γ

结果： 精确决策图 $\mathcal{D}$

创建根节点 $r \in \mathcal{U}_1$，其状态为 $s_r = \{0, 1, \ldots, |\delta^+(q)|\}$

对于 $i \in \{1, 2, \ldots, |\delta^-(q)|\}$ 并且 $u \in \mathcal{U}_i$ **执行**

  对于 $\ell \in s_u$ **do**

    创建一个状态为（$s_u \setminus \{\ell\}$）的节点 $v \in \mathcal{U}_{i+1} \cup \{0\}$，并创建一条弧 $a \in \mathcal{A}_i$，连接 u 和 v 其标签为 $l(a) = \ell$

对于 $u \in \mathcal{U}_{1+|\delta^-(q)|}$ **执行**

  创建两条弧 $a_1, a_2 \in \mathcal{A}_{1+|\delta^-(q)|}$，连接 u 到终端节点，并分别标记为 $l(a_1) = \Gamma$ 以及 $l(a_2) = -\Gamma$。

THEOREM 1. 考虑一个满足 $V = \{q\}$ 的 *SGUFP*。设 $\mathcal{D}$ 为由算法 *2* 构造的决策图。则 $\mathrm{Sol}(\mathcal{D})$ 表示(4) 的可行域。

证明。($\subseteq$) 考虑一条 $\mathcal{D}$ 的 r-t 路径，该路径编码了一个解（$\tilde{\boldsymbol{w}}^q, z$）。根据算法 2，该路径前 $|\delta^-(q)|$ 条弧的标签属于 $\{0, 1, \ldots, |\delta^+(q)|\}$，这表明 $\tilde{\boldsymbol{w}}^q$满足约束 (3b)。反设 $\tilde{\boldsymbol{w}}^q$ 不满足约束 (3a)，即对于某个 $j \in \delta^+(q)$，$\sum_{i \in \delta^-(q)} \mathrm{sign}\left(\left|w^q \mathrm{ind}^-(i,q) - \mathrm{ind}^+(q,j)\right|\right) \leq |\delta^-(q)| - 2$。这意味着，对于两个不同的 $i', i'' \in \delta^-(q)$，有 $\tilde{w}^q \mathrm{ind}^-(i',q) = \tilde{w}^q \mathrm{ind}^-(i'',q) = \mathrm{ind}^+(q,j)$。换言之，所选 r-t 路径在层 $\mathrm{ind}^-(i',q)$ 和 $\mathrm{ind}^-(i'',q)$ 上的弧具有相同的标签值 $\mathrm{ind}^+(q,j)$。根据算法 2 第 3 行，r-t 路径在层 $\mathrm{ind}^-(i',q)$ 和 $\mathrm{ind}^-(i'',q)$ 上的节点状态值都必须包含 $\mathrm{ind}^+(q,j)$。这与算法 2 第 4 行的状态更新策略矛盾，因为决策图每一层的正弧标签都会从后续节点的状态值中排除。

($\supseteq$) 考虑（4）的一个可行解点（$\tilde{\boldsymbol{w}}^q$；$\tilde{z}$）。设 $\tilde{\boldsymbol{w}}^q = (\ell_1, \ell_2, \ldots\ldots, \ell_{|\delta^-(q)|})$。根据约束（3a），$\tilde{\boldsymbol{w}}^q$ 中任意两个坐标的正值都不相同。$\mathcal{D}$中根节点的状态值包含所有索引值 $\{0, 1, \ldots, |\delta^+(q)|\}$。根据算法 2，$\mathcal{D}$第一层存在一条标签为 ℓ_1 的弧。因此，该弧头节点的状态值包含 $\ell_2 \in \{0, 1, \ldots, |\delta^+(q)|\} \setminus \{\ell_1\}$，这保证了这条路径第二层存在一条标签为 ℓ_2 的弧。按照类似的方法，我们可以追踪一条从根节点到第 $|\delta^-(q)|$ 层的路径，其弧标签与 $\tilde{\boldsymbol{w}}^q$ 的值相匹配。注意，对于最后一层，$\tilde{z} \in [-\Gamma, \Gamma]$，它包含在 $\mathcal{D}$ 最后一层弧标签所构成的区间内。因此，（$\tilde{\boldsymbol{w}}^q$；$\tilde{z}$）由 $\mathcal{D}$ 中一条 r-t 路径表示。□

使用对主问题 (4) 建模的决策图，而非对 (1a)-(1d) 建模的决策图，主要是为了减少表示变量 $\boldsymbol{w}$ 的弧层数量，相较于表示变量

y。事实证明，这种空间变换能够显著缩短决策图方法的求解时间。感兴趣的读者可参阅附录 A，了解这些优势的详细讨论以及初步计算结果。

如算法 2 所述，构造精确决策图对于规模较大的问题而言计算成本可能很高。如第 2.2 节所述，松弛决策图和受限决策图可用于克服这一困难。构造受限决策图很直接，只需从精确决策图中选取满足预设宽度限制的部分 r-t路径。另一方面，构造松弛决策图则需要谨慎地调整决策图结构，以合并节点，使其编码精确决策图中所有 r-t 路径的超集。我们在算法 3 中展示了一种构造此类松弛决策图的方法。与算法 2 类似，该算法针对单个 NSNM 节点给出，但可通过重复此过程扩展至多个节点。

算法 3： 为含一个节点的 SGUFP 主问题构造松弛决策图 $q \in \bar{V}$

数据： 节点 $q \in \bar{V}$，参数 Γ

结果： 松弛决策图 $\bar{\mathcal{D}}$

创建根节点 $r \in \mathcal{U}_1$，其状态为 $s_r = \{0, 1, \ldots, |\delta^+(q)|\}$

对于所有 $i \in \{1, 2, \ldots, |\delta^-(q)|\}$ 以及 $u \in \mathcal{U}_i$，**执行**

  对于所有 $\ell \in s_u$ **do**

    创建一个状态为 $(s_u \setminus \{\ell\}) \cup \{0\}$ 的节点 $v \in \mathcal{U}_{i+1}$，并创建一条弧 $a \in \mathcal{A}_i$ 连接 u 和 v 标签为 $l(a) = \ell$

  选择一组节点 $v_1, v_2, \ldots, v_k \in \mathcal{U}_{i+1}$，并将它们合并为状态为以下内容的节点 v' $s_{v'} = \bigcup_{j=1}^{k} s_{v_j}$

对于 $u \in \mathcal{U}_{1+|\delta^-(q)|}$，**执行**

  创建两条弧 $a_1, a_2 \in \mathcal{A}_{1+|\delta^-(q)|}$，将 u 连接到终端节点，标签为 $l(a_1) = \Gamma$ 以及 $l(a_2) = -\Gamma$。

THEOREM 2. *考虑一个具有 $V = \{q\}$ 的 SGUFP。令 $\bar{\mathcal{D}}$ 为通过算法 3 构造的决策图。则，$\bar{\mathcal{D}}$ 表示式 (4) 可行域的一个松弛。*

证明。设 $\bar{\mathcal{D}}$ 是算法 2 为含有单个节点 $q \in V$的主问题 (4) 构造的决策图。只需证明，$\bar{\mathcal{D}}$ 的解集是 $\mathcal{D}$ 解集的一个松弛。取 $\bar{\mathcal{D}}$ 的一条根终端路径 P，其编码点为 $(\dot{\boldsymbol{w}}^q; \dot{z})$。我们证明，存在 $\mathcal{D}$ 的一条根终端路径 P，其编码点为 $(\boldsymbol{w}^q; z)$，满足 $\boldsymbol{w}^q = \dot{\boldsymbol{w}}^q$ 且 $z = \dot{z}$。给定一个决策图，定义 P_k 为由前 k 层中的弧组成的子路径，其中 $1 \le k \le |\delta^-(q)|$。我们证明，对于 $\mathcal{D}$中任意编码点为 $\dot{\boldsymbol{w}}_k^q = (\dot{w}_1^q, \ldots, \dot{w}_k^q)$的子路径 P_k，都存在 $\mathcal{D}$中的一条子路径 P_k，其编码

点为 $\overline{\boldsymbol{w}}_k = (\overline{w}_1, \ldots, \overline{w}_k)$，使得对于 $h = 1, \ldots, k$，$\boldsymbol{w}_h = \dot{\boldsymbol{w}}_h$。注意，我们只需证明 $k \le |\delta^-(q)|$ 对应的匹配值，因为 $\mathcal{D}$ 和 $\mathcal{D}$ 在节点层 $|\delta^-(q)|+1$ 上的每个节点，都通过两条标签分别为 $-\Gamma$ 和 Γ 的弧连接到终端节点，因此最后一层总存在具有相同标签的匹配弧，即 $z = \dot{z}$。我们对 k 使用归纳法证明该结论。$k = 1$ 的基础情形显然成立，因为 $\mathcal{D}$ 在第一层中包含标签为 $\{0, 1, \ldots, |\delta^+(q)|\}$ 的弧，其中包括 P_1 上第一条弧的标签值。作为归纳假设，假设该命题对 $k = d$ 成立，即对于标签值为 $\dot{\boldsymbol{w}}_d^q = (\dot{w}_1^q, \ldots, \dot{w}_d^q)$ 的子路径 P_d，$\mathcal{D}$ 中存在一条具有匹配弧标签的子路径 P_d。我们证明该命题对 $d+1$ 也成立。设 $u \in A_{d+1}$ 和 $v \in A_{d+1}$ 分别为 P_d 和 P_d 的终点。由算法 2 可知，表示节点 u 处状态值的指标集包含 $\dot{w}_{d+1}^q$，即 $\dot{w}_{d+1}^q \in \dot{s}_u = \{0\} \cup \{1, \ldots, |\delta^+(q)|\} \setminus \{\dot{w}_1, \dot{w}_2, \ldots, \dot{w}_d\}$。另一方面，算法 3 第 5 行的合并步骤意味着 $s_v \supseteq \{0\} \cup \{1, \ldots, |\delta^+(q)|\} \setminus \{w_1, w_2, \ldots, w_d\} = \{0\} \cup \{1, \ldots, |\delta^+(q)|\} \setminus \{\dot{w}_1, \dot{w}_2, \ldots, \dot{w}_d\} = \dot{s}_u$；该包含关系源于路径 P_d 上节点的状态值包含各条独立路径的状态值这一事实，这是由合并操作导致的；第一个等式则由归纳假设成立。因此，s_v 必定包含 $\dot{w}_{d+1}^q$，这意味着存在一条弧，其 $\dot{w}_{d+1}^q$ 连接到 P_d 上的节点 v。将这条弧接到 P_d 上，即得到所需的子路径 P_{d+1}。□

3.3. DD-BD：子问题表述

在 DD-BD 算法的每次迭代中，都会将主问题的一个最优解代入子问题，以获得可行性/最优性割。对于 SGUFP 表述，这一过程对应于在 $\boldsymbol{w}$ 变量空间中求解（4）以获得一个最优解，并用它来求解子问题（2a）-（2h）。然而，子问题的公式定义在原始二元变量 $\boldsymbol{y}$ 上，并且由此生成的可行性/最优性割也在这一空间中。为了解决主问题与子问题中的变量空间不一致这一问题，我们需要找到变量 $\boldsymbol{w}$ 和 $\boldsymbol{y}$ 之间的一一映射，如下所述。

Proposition 3. 考虑一个节点 $q \in V$。令 $\boldsymbol{y}^q$ 为 (1b)-(1d) 的一个可行解。那么，$\boldsymbol{w}^q$ 可表示为

$$w^q_{\mathrm{ind}^-(i,q)} = \sum_{j \in \delta^+(q)} \mathrm{ind}^+(q,j)\, y^q_{ij} \qquad \forall i \in \delta^-(q), \tag{5}$$

是 (3a)-(3b) 的可行解。反之，设 $\boldsymbol{w}^q$ 是 (3a)-(3b) 的可行解。那么，$\boldsymbol{y}^q$ 可按如下方式获得

$$y^q_{ij} = 1 - \mathrm{sign}\left(\left| w^q_{\mathrm{ind}^-(i,q)} - \mathrm{ind}^+(q,j) \right|\right) \qquad \forall (i,j) \in \delta^-(q) \times \delta^+(q), \tag{6}$$

是 (1b)-(1d) 的可行解。

证明。对于直接陈述，设 $\boldsymbol{y}^q$ 是 (1b)-(1d) 的可行解，并根据 (5) 构造向量 $\boldsymbol{w}^q$ 。我们证明 $\boldsymbol{w}^q$ 满足所有约束 (3a)-(3b)。首先，我们证明约束 (3a) 得到满足。反设存在 $j' \in \delta^+(q)$，使得 $\sum_{i\in\delta^-(q)} \text{sign}\left(\left|w^q \text{ind}^-(i,q) - \text{ind}^+(q,j')\right|\right) \leq |\delta^-(q)| - 2$。这意味着，对于某些 $i', i'' \in \delta^-(q)$，有 $w^q \text{ind}^-(i',q) = w^q \text{ind}^-(i'',q) = \text{ind}^+(q,j')$。于是，我们可以写成

$$w^q_{\text{ind}^-(i',q)} = \sum_{j\in\delta^+(q)} \text{ind}^+(q,j) y^q_{i'j} = \text{ind}^+(q,j') = \sum_{j\in\delta^+(q)} \text{ind}^+(q,j) y^q_{i''j} = w^q_{\text{ind}^-(i'',q)},$$

其中，第一个和最后一个等式由 (5) 成立。上述关系链中的第二个和第三个等式意味着 $y^q_{i'j'} = y^q_{i''j'} = 1$，因为 $\text{ind}^+(q, j') > 0$。这违反了约束 (1c)，从而得出矛盾。接下来，我们证明约束 (3b) 成立。根据 $\boldsymbol{w}^q$ 的构造和约束 (1b)，该证明可直接得出。

对于逆命题，设 $\boldsymbol{w}^q$ 是 (3a)-(3b) 的一个可行解，并根据 (6) 构造向量 $\boldsymbol{y}^q$ 。我们证明 $\boldsymbol{y}^q$ 满足所有约束 (1b)-(1d)。为证明每个约束 (1b) 都成立，考虑 $i \in \delta^-(q)$。我们可以写成

$$\sum_{j\in\delta^+(q)} y^q_{ij} = |\delta^+(q)| - \sum_{j\in\delta^+(q)} \text{sign}\left(\left|w^q_{\text{ind}^-(i,q)} - \text{ind}^+(q,j)\right|\right) \leq |\delta^+(q)| - \left(|\delta^+(q)| - 1\right) = 1,$$

其中，第一个等式由 $\boldsymbol{y}^q$ 的构造得出，而不等式由 (3b) 成立，因为 $\left|w^q{}_{\text{ind}^-(i,\ q)} - \text{ind}^+(q,\ j)\right| = 0$ 至多对一个指标 $j \in \delta^+(q)$ 成立。为证明每个约束 (1c) 都成立，选取 $j \in \delta^+(q)$。我们有

$$\sum_{i\in\delta^-(q)} y^q_{ij} = |\delta^-(q)| - \sum_{i\in\delta^-(q)} \text{sign}\left(\left|w^q_{\text{ind}^-(i,q)} - \text{ind}^+(q,j)\right|\right) \leq 1,$$

其中，等式由 $\boldsymbol{y}^q$ 的构造得出，不等式则是因为约束 (3a) 成立。最后，由于 $1 - \text{sign}(|.|) \in \{0,1\}$，每个约束 (1d) 均得到满足。

□

PROPOSITION 4. 由(5) 和(6) 描述的映射在各自的定义域上均为一一映射。

证明。 注意，由 (5) 描述的映射是一个线性变换，其形式为 $\boldsymbol{w}^q = B\boldsymbol{y}^q$，系数矩阵为 $B \in \mathbb{Z}^{|\delta^-(q)|\times(|\delta^-(q)||\delta^+(q)|)}$。显然，根据 B 的单位块结构，它是满行秩的，因为每一列都只包含一个非零元素，而每一行至少包含一个非零元素。因此， B 的零空间为起点，这意味着仅当 $\hat{\boldsymbol{y}}^q = \tilde{\boldsymbol{y}}^q$ 时，$\hat{\boldsymbol{w}}^q = \tilde{\boldsymbol{w}}^q$ 。

对于由 (6) 描述的映射，令不同的点 $\hat{\boldsymbol{w}}^q$ 和 $\tilde{\boldsymbol{w}}^q$ 满足 (3b)。利用 $\hat{\boldsymbol{w}}^q$ 和 $\tilde{\boldsymbol{w}}^q$，根据 (6) 构造向量 $\hat{\boldsymbol{y}}^q$ 和 $\tilde{\boldsymbol{y}}^q$ 。由于 $\hat{\boldsymbol{w}}^q$ 和 $\tilde{\boldsymbol{w}}^q$ 不同，必定

存在 $i \in \delta^-(q)$，使得 $\hat{w}^q \text{ind}^-(i,q) \neq \tilde{w}^q \text{ind}^-(i,q)$。这意味着这些变量中至少有一个非零，例如 $\hat{w}^q \text{ind}^-(i,q)$。由 (3b) 可知，对于某个 $j' \in \delta^+(q)$，有 $\hat{w}^q \text{ind}^-(i,q) = \text{ind}^+(q,j')$，且 $\hat{w}^q \text{ind}^-(i,q) \neq \text{ind}^+(q,j')$。根据 (6)，我们写作 $\hat{y}_{ij'} = 1 - \text{sign}\left(\left|\hat{w}^q \text{ind}^-(i,q) - \text{ind}^+(q,j')\right|\right) = 1$，以及 $\tilde{y}_{ij'} = 1 - \text{sign}\left(\left|\tilde{w}^q \text{ind}^-(i,q) - \text{ind}^+(q,j')\right|\right) = 0$，表明 $\hat{\boldsymbol{y}}^q \neq \tilde{\boldsymbol{y}}^q$。□

利用命题3和4的结果，我们可以将 DD-BD算法1 完整地应用于 SGUFP。具体而言，在算法的每次迭代中，我们可以通过映射（6），将由表示主问题（4）的决策图得到的最优解（$\bar{\boldsymbol{w}}$, $\bar{z}$）转换为解（$\bar{\boldsymbol{y}}$, $\bar{z}$）。给定一个最优第一阶段解 $\bar{\boldsymbol{y}}$，我们可以求解 $|\Xi|$ 个独立子问题，每个子问题对应第二阶段的一种需求实现。子问题得到的可行性割处于 $\boldsymbol{y}$ 变量空间中，通过映射（5）将其转换回 $\boldsymbol{w}$ 变量空间，并添加到主问题中。此外，如果所有子问题都产生最优性割，则可以将这些割聚合，生成一个处于（$\boldsymbol{y}$, z）空间中的最优性割；将其转换到（$\boldsymbol{w}$, z）变量空间后，再添加到主问题中。主决策图将根据由此得到的不等式进行细化，并返回一个最优解供下一次迭代使用。

在本节剩余部分，我们将详细介绍如何从子问题 (2a)-(2h) 推导最优性割/可行性割。考虑将弧集合 A 划分为以下子集

$$A_1 := \left\{(i,j) \in A \,\middle|\, \delta^-(i) = \emptyset,\ \delta^+(j) \neq \emptyset\right\},\ A_2 := \left\{(i,j) \in A \,\middle|\, \delta^-(i) \neq \emptyset,\ \delta^+(j) = \emptyset\right\},$$
$$A_3 := \left\{(i,j) \in A \,\middle|\, \delta^-(i) \neq \emptyset,\ \delta^+(j) \neq \emptyset\right\},\ A_4 := \left\{(i,j) \in A \,\middle|\, \delta^-(i) = \emptyset,\ \delta^+(j) = \emptyset\right\},$$

并令 $\boldsymbol{\theta}^\xi = (\boldsymbol{\beta}^\xi,\ \boldsymbol{\gamma}^\xi,\ \boldsymbol{\delta}^\xi,\ \boldsymbol{\phi}^\xi,\ \boldsymbol{\lambda}^\xi,\ \boldsymbol{\mu}^\xi)$ 表示与情景 $\xi \in \Xi$ 下约束 (2a)-(2h) 相关联的对偶变量向量。进一步，定义双函数

$$f(\boldsymbol{y};\boldsymbol{\theta}^\xi) = \sum_{q\in V}\sum_{j\in\delta^+(q)}\left(-\ell_{qj}\beta^\xi_{qj} + u_{qj}\gamma^\xi_{qj}\right) + \sum_{q\in\bar{V}}\sum_{(i,j)\in\delta^-(q)\times\delta^+(q)}\left(u_{iq}(1-y^q_{ij})\lambda^\xi_{iqj} + u_{qj}(1-y^q_{ij})\mu^\xi_{iqj}\right)$$
$$+\sum_{q\in\bar{V}}\sum_{i\in\delta^-(q)}\left(u_{iq}\sum_{j\in\delta^+(q)} y^q_{ij}\sigma^\xi_{iq}\right) + \sum_{q\in\bar{V}}\sum_{j\in\delta^+(q)}\left(u_{qj}\sum_{i\in\delta^-(q)} y^q_{ij}\phi^\xi_{qj}\right).$$

对于给定的 $\bar{\boldsymbol{y}}$ 和每个情景 $\xi \in \Xi$，子问题 (2a)-(2h) 的对偶问题可写为¯如下，其中节点上的符号 $\star$ 表示该节点属于 V。

$$\text{最小化 } f(\bar{\boldsymbol{y}};\ \boldsymbol{\theta}^\xi) \tag{7a}$$
$$\text{s.t.}\quad \alpha^\xi_{\mathring{q}} - \beta^\xi_{i\mathring{q}} + \gamma^\xi_{i\mathring{q}} + \sum_{j:j\in\delta^+(\mathring{q})}\lambda^\xi_{i\mathring{q}j} - \sum_{j:j\in\delta^+(\mathring{q})}\mu^\xi_{i\mathring{q}j} + \sigma^\xi_{i\mathring{q}} \geq r_{i\mathring{q}} \qquad \forall (i,\mathring{q}) \in A_1 \tag{7b}$$
$$\alpha^\xi_q - \beta^\xi_{iq} + \gamma^\xi_{iq} \geq r_{iq} \qquad \forall (i,q) \in A_1 \tag{7c}$$

$$-\alpha^{\xi}_{\star q}-\beta^{\xi}_{\star qj}+\gamma^{\xi}_{\star qj}-\sum_{i:i\in\delta^-(\star q)}\lambda^{\xi}_{i\star qj}+\sum_{i:i\in\delta^-(\star q)}\mu^{\xi}_{i\star qj}+\phi^{\xi}_{\star qj}\geq r_{\star qj} \qquad \forall(\star q,j)\in A_2 \quad (7d)$$

$$-\alpha^{\xi}_{q}-\beta^{\xi}_{qj}+\gamma^{\xi}_{qj}\geq r_{qj} \qquad \forall(q,j)\in A_2 \quad (7e)$$

$$-\alpha^{\xi}_{\star q}+\alpha^{\xi}_{\star j}-\beta^{\xi}_{\star q\star j}+\gamma^{\xi}_{\star q\star j}+\sum_{i\in\delta^-(\star q)}\left(\mu^{\xi}_{i\star q\star j}-\lambda^{\xi}_{i\star q\star j}\right)+\sum_{i\in\delta^+(\star j)}\left(\lambda^{\xi}_{\star q\star ji}-\mu^{\xi}_{\star q\star ji}\right)+\sigma^{\xi}_{\star q\star j}+\phi^{\xi}_{\star q\star j}\geq r_{\star q\star j} \qquad \forall(\star q,\star j)\in A_3 \quad (7f)$$

$$-\alpha^{\xi}_{\star q}+\alpha^{\xi}_{j}-\beta^{\xi}_{\star qj}+\gamma^{\xi}_{\star qj}+\sum_{i\in\delta^-(\star q)}\left(\mu^{\xi}_{i\star qj}-\lambda^{\xi}_{i\star qj}\right)+\phi^{\xi}_{\star qj}\geq r_{\star qj} \qquad \forall(\star q,j)\in A_3 \quad (7g)$$

$$-\alpha^{\xi}_{q}+\alpha^{\xi}_{\star j}-\beta^{\xi}_{q\star j}+\gamma^{\xi}_{q\star j}+\sum_{i\in\delta^+(\star j)}\left(\lambda^{\xi}_{q\star ji}-\mu^{\xi}_{q\star ji}\right)+\sigma^{\xi}_{q\star j}\geq r_{q\star j} \qquad \forall(q,\star j)\in A_3 \quad (7h)$$

$$-\alpha^{\xi}_{q}+\alpha^{\xi}_{j}-\beta^{\xi}_{qj}+\gamma^{\xi}_{qj}\geq r_{qj} \qquad \forall(q,j)\in A_3 \quad (7i)$$

$$-\beta^{\xi}_{iq}+\gamma^{\xi}_{iq}\geq r_{iq} \qquad \forall(i,q)\in A_4 \quad (7j)$$

$$\alpha^{\xi}_{q}\in\mathbb{R} \qquad \forall q\in V' \quad (7k)$$

$$\beta^{\xi}_{ij},\ \gamma^{\xi}_{ij},\ \sigma^{\xi}_{ij},\ \phi^{\xi}_{ij},\ \lambda^{\xi}_{iqj},\ \mu^{\xi}_{iqj}\geq 0 \qquad \forall i,q,j\in V. \quad (7l)$$

如果上述问题对于所有 $\xi\in\Xi$ 都有最优解 $\hat{\boldsymbol{\theta}}^{\xi}$，那么子问题的输出将是形式为 $\sum_{\xi\in\Xi}\Pr^{\xi} f(\boldsymbol{y};\hat{\boldsymbol{\theta}}^{\xi})\geq z$ 的最优性割。如果上述问题沿着某个 $\xi\in\Xi$ 的射线 $\hat{\boldsymbol{\theta}}^{\xi}$ 无界，那么子问题的输出将是形式为 $f(\boldsymbol{y};\hat{\boldsymbol{\theta}}^{\xi})\geq 0$ 的可行性割。请注意，通过映射 (5) 将上述约束中的变量 $\boldsymbol{y}$ 替换为 $\boldsymbol{w}$，会得到可分离非线性约束。尽管如此，由于这些约束将用于细化主决策图，借助决策图在对此类约束建模时所具有的结构灵活性，将它们纳入模型并不复杂；有关使用决策图对整数非线性规划进行建模的详细说明，请参阅 Davarnia 和 Van Hoeve（2020）。

4. 计算实验

本节中，我们使用三种不同的方法求解 SGUFP，将其作为具有需求随机性的单元列车调度问题的核心模型：(i) 标准混合整数规划公式，它是两阶段模型的确定性等价形式，包含主问题和 $|\Xi|$ 子问题的所有变量和约束；(ii) 第 3.1 节提出的 Benders 重构，由主问题 (1a)-(1d) 和 $|\Xi|$ 子问题 (2a)-(2h) 组成；以及 (iii) 本文提出的 DD-BD 算法。在 Benders 方法中，我们使用从主问题获得的固定向量 $\bar{\boldsymbol{y}}$，分别求解各个子问题。子问题生成的可行性割直接加入主问题的约束集，而最优性割则作为涵盖所有情景的聚合割加入。需要指出的是，只要任一情景存在可行性割，我们就会将其直接加入，以排除当前迭代的解，然后进入下一次迭代。为了获得可为单个 z 变量提供界的有效不等式，我们需要聚合所有情景子问题的有效不等式，因为 z 由所有这些子问题的目标值

组成。因此，只有在所有子问题都有最优性割时，我们才能为 z 变量生成最优性割。对于 DD-BD 方法，我们采用以下算法选择来构建受限决策图和松弛决策图。对于受限决策图，我们选择长度最大的部分 r-t 路径，因为这些路径更有可能包含最优解。对于松弛决策图，我们合并状态值中共有成员数量最多的节点。关于可用于此目的的其他启发式方法，请参阅 Bergman 等人（2016a）。

4.1. 测试实例

在我们的实验中，我们采用第 3.1 节给出的 SGUFP 网络结构。为确保问题始终可行，我们创建一个人工节点 s_0，用于弥补供应不足，并添加一条从人工供应节点 s_0 指向每个需求节点的弧。

我们根据 Davarnia 等人（2019）给出的规范创建测试实例，该规范受到现实模型的启发。具体而言，我们考虑一个基础铁路网络 $G' = (V', A')$，其中 10% 和 30% 的节点分别为供应节点和需求节点。我们假设 50% 的节点必须满足 NSNM 要求。随后，按照上述方式增设供需节点和人工节点，并根据以下设置构建网络 $G = (V, A)$。供应节点处的整数供应值从区间 $[100, 600]$ 中随机选取。连接 s_0 与需求节点的弧的容量视为无界，其他弧的整数容量值则从区间 $[100, 300]$ 中随机选取。对于每个需求情景 $\xi \in \Xi$，需求节点处的整数需求值从区间 $[100, 200]$ 中随机选取。连接 s_0 与需求节点的弧的奖励从区间 $[-10, -5]$ 中生成，以表示未满足需求的成本。连接源点与供应节点的弧的奖励从区间 $[5, 10]$，中随机选取；连接需求节点与汇点的弧的奖励固定为零，因为这些弧上的流量也固定不变。所有其他弧的奖励均从区间 $[-2, 2]$ 中随机生成，其中负值表示流经拥塞弧的成本。我们考虑四类铁路网络，其$|V'| \in \{40, 60, 80, 100\}$。对于每一类，我们针对需求情景数量 $|\Xi| \in \{50, 100, 150, 200, 250\}$ 创建五种情景类别。对于每种网络类别和情景类别，我们根据上述设置创建五个随机实例。测试实例可公开获取（Salemi 和 Davarnia，2022b）。

4.2. 数值结果

本节给出数值结果，对比 DD-BD 公式在 SGUFP 实例上的表现与 MIP 公式（记为“MIP”）及标准 Benders 重构（记为 “BD” ）的表现。所有实验均在一台机器上进行，该机器

运行 Windows 10、x64 操作系统，配备 Intel® Core i7 处理器（2.60 GHz）和 32 GB 内存。求解 MIP 和 BD 模型的实例时，使用 Gurobi 优化求解器（9.1.1 版）。为确保比较公平，使用 Gurobi 求解问题时，所有方法均关闭预求解和割。表 1–4 报告了这些公式在 $|V'| \in \{40, 60, 80, 100\}$ 和 $|\Xi| \in \{50, 100, 150, 200, 250\}$ 上的运行时间，时间限制设为 3600 秒。符号“> 3600”表示问题未能在时间限制内求解。如这些表格所示，DD-BD 公式优于其他方案。具体而言，随着问题规模增大，DD-BD 与 MIP 和 BD 方法之间的求解时间差距也随之扩大。例如，表 1 显示，DD-BD 方法在 275 秒内求解了全部 25 个实例，而 MIP 方法有 10 个实例未能在 3600 秒内求解，其中 80% 涉及 200 或 250 个情景。这表明 DD-BD 明显优于 MIP 方法。此外，对于大多数实例，DD-BD 方法优于标准 BD 方法，因此在这三种方法中表现最佳。图 5–8 使用箱线图比较了 DD-BD、BD 和 MIP 公式在不同网络规模和各情景类别下的表现。为使图示统一，对于未能在时间限制内求解的实例，我们将其运行时间设为 3600 秒。如图所示，在所有情况下，DD-BD 方法的运行时间最小值、中位数和最大值都显著小于 BD 和 MIP 方法。这些结果表明，DD-BD 框架有望求解具有复杂组合结构的网络问题。附录 B 还给出了 DD-BD 方法的其他数值结果，以评估其求解更大规模问题的能力。

表 1 $|V'| = 40$ **的混合整数规划、Benders 分解和 DD-BD 的运行时间（秒）。**

实例编号	模型	场景数量				
		50	100	150	200	250
1	MIP	75.74	512.62	2877.19	> 3600	> 3600
	BD	141.83	313.84	339.81	451.93	565.82
	DD-BD	56.94	129.87	163.43	219.02	274.36
2	MIP	67.59	275.07	906.10	1892.21	2235.53
	BD	63.44	121.25	141.04	230.81	235.87
	DD-BD	42.60	82.65	128.16	164.52	208.94
3	MIP	94.86	753.23	2453.05	> 3600	> 3600
	BD	71.14	139.20	172.86	224.33	244.91
	DD-BD	53.32	93.58	113.93	178.65	217.33
4	MIP	71.46	309.62	> 3600	> 3600	> 3600
	BD	63.55	182.01	267.94	334.74	380.22
	DD-BD	46.61	87.81	130.19	183.23	253.72
5	MIP	380.33	406.73	> 3600	> 3600	> 3600
	BD	123.69	198.73	205.16	231.56	287.24
	DD-BD	67.04	104.78	138.46	195.69	231.74

表2 $|V'| = 60$ **的混合整数规划、Benders 分解和 DD-BD 运行时间（单位：秒）。**

实例编号	模型	场景数量				
		50	100	150	200	250
1	MIP	893.73	> 3600	> 3600	> 3600	> 3600
	BD	241.85	556.18	582.80	758.54	933.05
	DD-BD	176.16	357.06	603.81	719.27	901.02
2	MIP	206.87	811.64	1554.10	> 3600	> 3600
	BD	259.63	351.39	624.08	816.44	1017.95
	DD-BD	189.07	388.85	572.52	764.76	961.35
3	MIP	139.70	702.96	1035.79	> 3600	> 3600
	BD	246.48	569.37	628.84	795.56	978.15
	DD-BD	142.81	284.65	422.52	565.23	725.86
4	MIP	153.16	415.46	938.03	1681.21	2604.25
	BD	238.33	388.19	563.15	732.59	919.08
	DD-BD	131.29	262.36	393.18	521.12	654.71
5	MIP	165.57	706.16	2447.15	> 3600	> 3600
	BD	194.12	244.61	479.32	463.63	617.09
	DD-BD	112.09	221.30	332.25	443.96	556.33

表 3 MIP、Benders 分解和 DD-BD 求解 $|V'| = 80$ **的运行时间（秒）。**

实例编号	模型	场景数量				
		50	100	150	200	250
1	MIP	215.82	860.21	> 3600	> 3600	> 3600
	BD	588.51	806.61	1731.50	1860.12	2051.52
	DD-BD	256.12	500.52	757.68	1025.88	1278.13
2	MIP	479.76	> 3600	> 3600	> 3600	> 3600
	BD	398.29	713.01	861.65	1080.79	1709.04
	DD-BD	184.34	379.04	724.66	1088.21	1587.90
3	MIP	238.79	996.22	> 3600	> 3600	> 3600
	BD	702.18	1236.58	1650.42	1773.63	2227.89
	DD-BD	285.13	518.46	778.97	1046.39	1326.22
4	MIP	404.26	2441.64	2855.29	> 3600	> 3600
	BD	572.83	1219.37	1334.21	1745.91	2089.80
	DD-BD	263.78	665.30	1230.81	1277.93	1444.02
5	MIP	778.50	> 3600	> 3600	> 3600	> 3600
	BD	231.11	481.31	625.91	1310.24	1452.27
	DD-BD	187.34	376.96	564.34	1205.54	1412.94

表4 $|V'| = 100$ **的混合整数规划、Benders 分解和 DD-BD 运行时间（秒）。**

实例编号	模型	场景数量				
		50	100	150	200	250
1	MIP	774.18	> 3600	> 3600	> 3600	> 3600
	BD	1282.59	1728.71	1848.49	2307.74	3309.93
	DD-BD	698.36	1427.38	1731.95	2014.96	3323.54
2	MIP	480.97	> 3600	> 3600	> 3600	> 3600
	BD	781.47	1573.23	1820.79	2672.18	2819.61
	DD-BD	586.89	1171.96	1848.49	2471.49	2635.22
3	MIP	3071.37	> 3600	> 3600	> 3600	> 3600
	BD	1072.14	1322.96	2112.50	2951.55	3412.99
	DD-BD	485.31	703.70	1055.36	1803.66	2269.97
4	MIP	838.79	2585.38	> 3600	> 3600	> 3600
	BD	1548.93	1738.92	2580.53	2616.19	3169.28
	DD-BD	554.89	743.64	1098.82	2052.73	3094.23
5	MIP	714.39	> 3600	> 3600	> 3600	> 3600
	BD	808.48	1013.68	1722.01	2824.14	3282.10
	DD-BD	353.48	700.57	1680.60	2213.81	2907.78

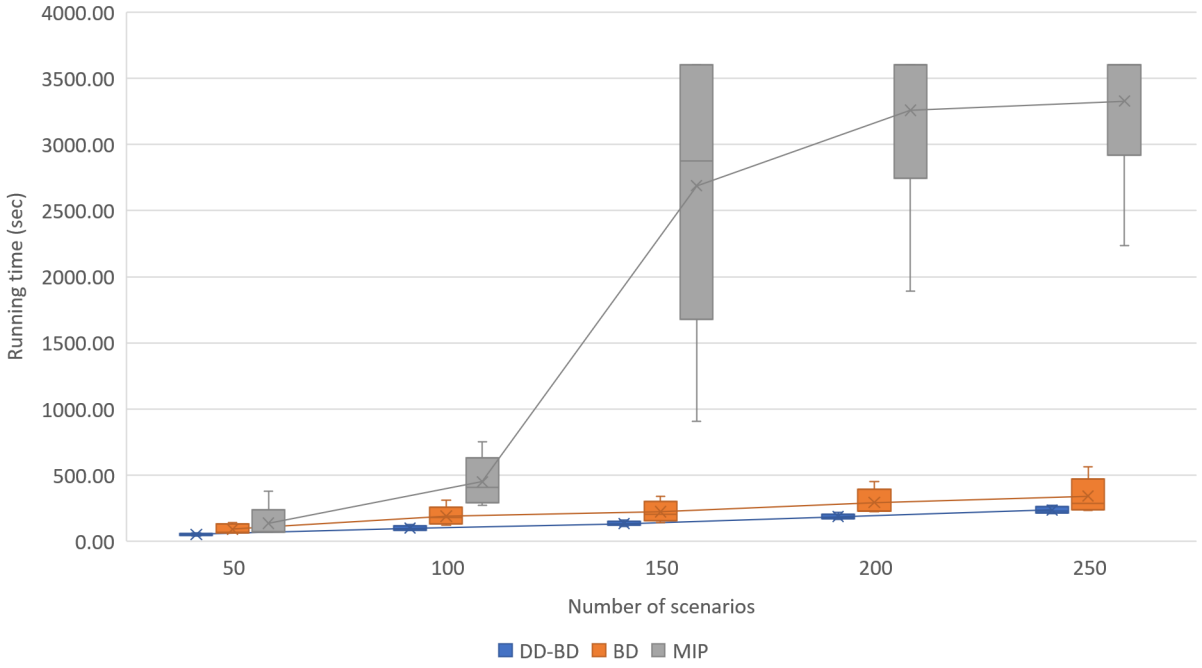


图 5 当 $|V'|=40$ 时 DD-BD、Benders 分解和混合整数规划模型在五种情景下的比较 s

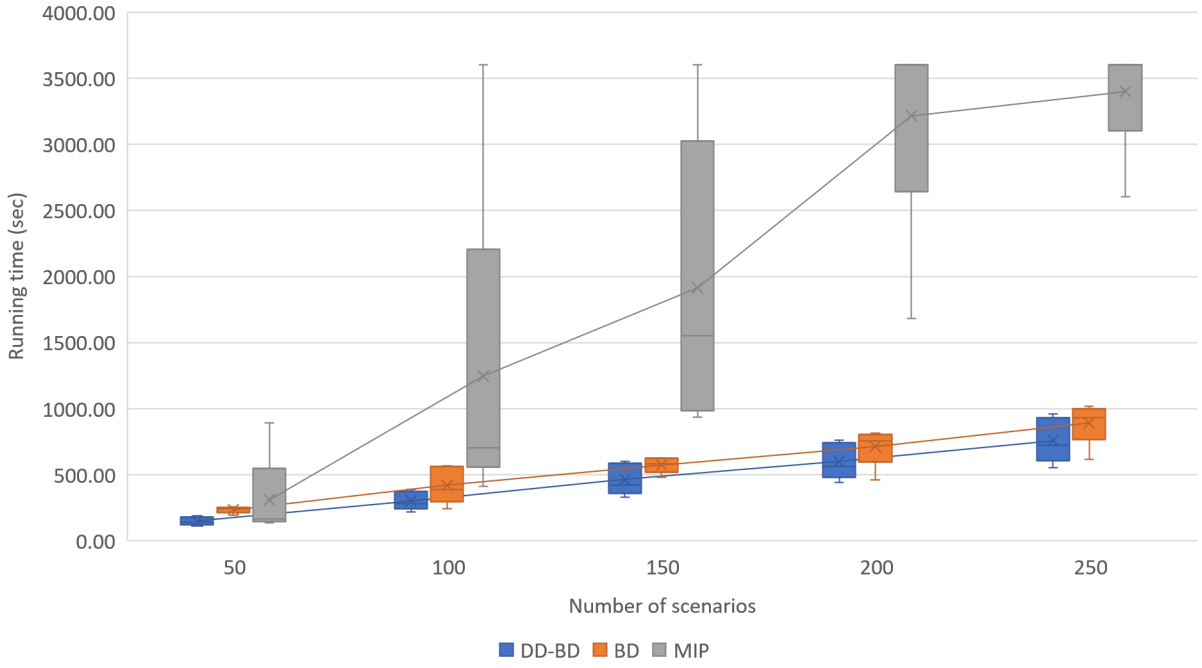


图 6 当 $|V'|=60$ 时 DD-BD、Benders 分解和混合整数规划模型在五种情景下的比较

我们在此指出，尽管本文重点研究的是满足不拆分不合并要求的单元列车问题，但所提出的 DD-BD 框架也可用于建模包含流变量额外侧约束的网络问题，因为这些约束可以在子问题中处理，而主问题中的 DD结构

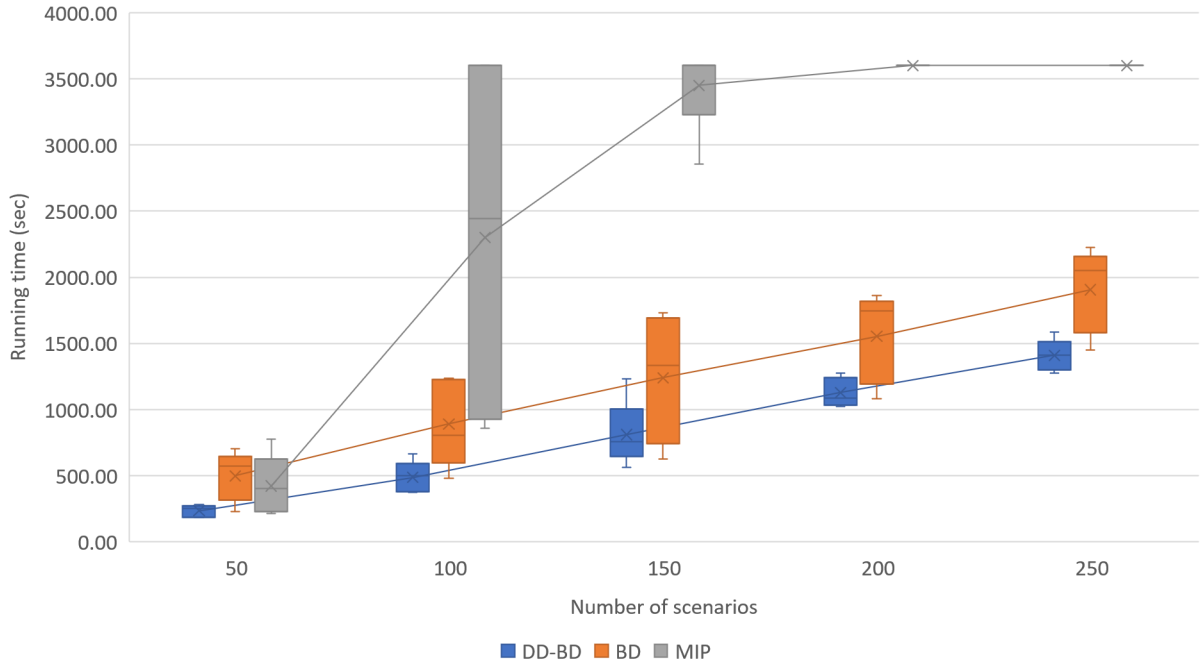


图 7 $|V'| = 80$ **在五种情景下 DD-BD、Benders 分解和混合整数规划模型的比较** **s**

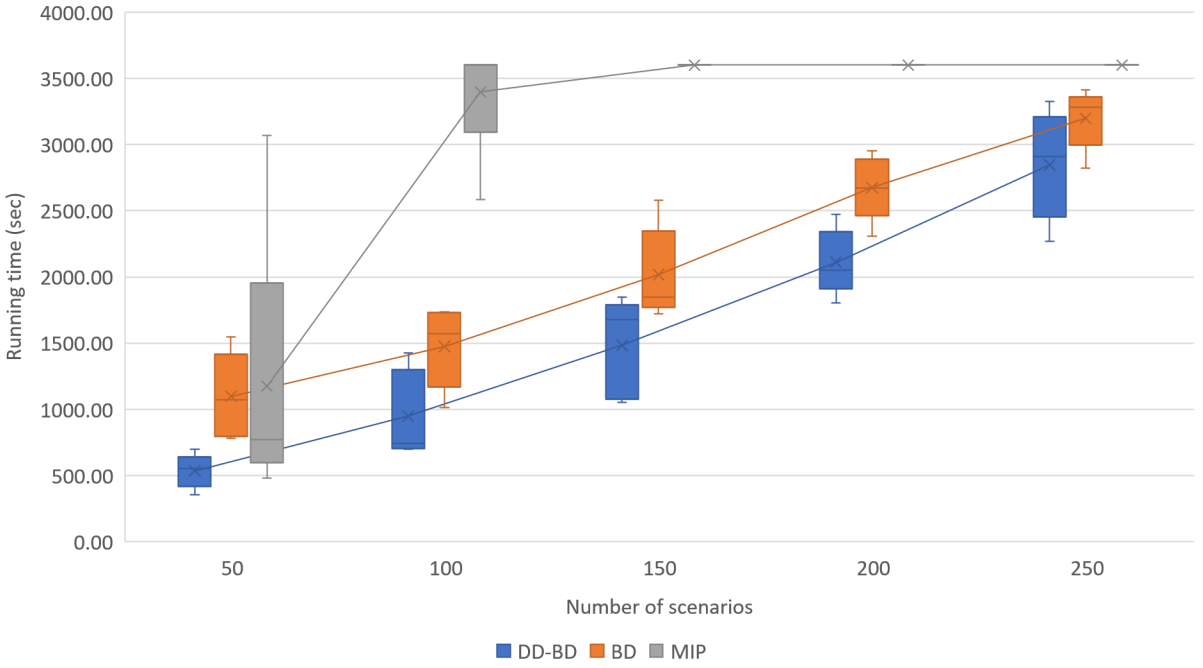


图 8 $|V'| = 100$ **在五种情景下 DD-BD、Benders 分解和混合整数规划模型的比较**

仍保持不变。此类侧约束的例子包括使用费 限制（Holzhauser、Krumke 和 Thielen，2017b）以及流量比 要求（Holzhauser、Krumke 和 Thielen，2017a）。将 DD-BD 方法应用于此类网络模型，并评估其相较于其他方法的有效性，可能是未来研究的一个有趣方向。

5. 结论

本文介绍了一种基于决策图的框架，用于求解 SGUFP。该框架采用 Benders 分解，将 SGUFP 分解为两个部分：由组合型 NSNM约束构成的主问题，以及用于求解连续网络流模型的子问题。主问题由决策图建模，并根据子问题生成的割逐步细化。为评估所提方法的性能，我们将其应用于一个建模为 SGUFP 的单元列车调度问题变体，并将其与该问题的标准混合整数规划及 Benders 重构进行比较。

致谢

本项目部分由爱荷华能源中心、爱荷华州经济发展局及其公用事业合作伙伴资助。我们感谢匿名审稿人和副编辑提出的宝贵意见，这些意见有助于改进本文。

参考文献

Abbink E, Van den Berg B, Kroon L, Salomon M, 2004 客运列车铁路机车车辆的分配。*Transportation Science 38(1):33–41. Alfieri A, Groot R, Kroon L, Schrijver A, 2006* 铁路机车车辆的高效循环。*Transportation Science 40(3):378–391. Andersen HR, Hadzic T, Hooker JN, Tiedemann P, 2007* 基于多值决策图的约束存储库。约束规划原理与实践国际会议，*118–132*（施普林格）。美国铁路协会，*2021* 货运铁路概况。*https://www.aar.org*，访问日期：*06/28/2021*。*Baier G, K¨ohler E, Skutella M, 2005 k-*可分流问题。*Algorithmica 42(3):231–248. Bergman D, Cire AA, 2018* 通过状态空间分解进行离散非线性优化。*Management Science 64(10):4700–4720. Bergman D, Cire AA, Van Hoeve WJ, Hooker J, 2016a* 优化决策图，第*1*卷（施普林格）。*Bergman D, Cire AA, Van Hoeve WJ, Hooker JN, 2016b* 基于决策图的离散优化。*INFORMS Journal on Computing 28(1):47–66. Bornd¨orfer R, Reuther M, Schlechte T, Waas K, Weider S, 2016* 城际铁路机车车辆周转的集成优化。*Transportation Science 50(3):863–877. Cacchiani V, Toth P, 2012* 名义列车时刻表编制问题与鲁棒列车时刻表编制问题。*European Journal of Operational Research 219(3):727–737. Carey M,Crawford I, 2007* 在繁忙复杂车站网络上调度列车。*Transportation Research Part B: Methodological 41(2):159–178. Ceselli A, Gatto M, L¨ubbecke ME, Nunkesser M, Schilling H, 2008* 瑞士联邦铁路货运快运服务的优化。*Transportation Science 42(4):450–465. Chakrabarti A, Chekuri C, Gupta A, Kumar A, 2007* 不可分流问题的近似算法。*Algorithmica 47(1):53–78. Cordeau JF, Toth P, Vigo D, 1998* 列车路径规划与调度优化模型综述。*Transportation Science 32(4):380–404. Cornelsen S, Di Stefano G, 2007* 股道分配。*Journal of Discrete Algorithms 5(2):250–261. Davarnia D, 2021* 基于决策图的连续非线性规划强松弛。*Operations Research Letters 49(2):239–245. Davarnia D, Richard JPP, I.cy¨uz-Ay E, Taslimi B, 2019* 含不可分节点流的网络模型及其在单元列车调度中的应用。*Operations Research 67(4):1053–1068.*

DavarniaD, Van Hoeve WJ, 2020 通过决策图对整数非线性规划进行外逼近。*Mathematical Programming* 1–40.Demir E, Burgholzer W, Hruˇsovsk`y M, ArıkanE, Jammernegg W,Van WoenselT,2016 考虑旅行时间不确定性的绿色多式联运网络设计问题。*Transportation Research Part B: Methodological* 93:789–807.Fuchsberger M, L¨uthiP,2007 通过资源受限时空整数多商品流解决主要车站区域的列车调度问题。苏黎世联邦理工学院运筹学研究所。 Furchtgott-Roth D, Hu PS, Nguyen L, Jahanmir S, Moore WH, Riley D, Beningo S, Chambers M, Smith- Pickel S, Thai H, 等，2021 《*2021* 年交通运输袖珍指南》。 Gong C, Shi J, Wang Y, Zhou H, Yang L, Chen D, Pan H, 2021 考虑动态随机客运需求的列车时刻表编制：一种随机优化方法。*Transportation Research Part C: Emerging Technologies* 123:102963.Gonzalez JE, Cire AA,LodiA, Rousseau LM, 2020 整数规划与决策图搜索树的集成及其在最大独立集问题中的应用。*Constraints* 1–24.Haahr J, Lusby RM, 2017将机车车辆调度与列车单元调车相结合。*European Journal of Operational Research* 259(2):452–468.Haahr JT,Wagenaar JC, Veelenturf LP, Kroon LG, 2016 客运铁路机车车辆（重新）调度的两种精确方法比较。*Transportation Research Part E: Logistics and Transportation Review* 91:15–32.Hadˇzi´c T, HookerJ, 2006 利用二元决策图进行离散全局优化。 全球优化研讨会：整合凸性、优化、逻辑编程与计算代数几何（*GICOLAG*）。维也纳。 Harrod S, Gorman MF, 2010 货运列车路径规划与调度的运筹学。 *Wiley* 运筹学与管理科学百科全书。 Heil J, Hoffmann K, Buscher U, 2020铁路乘务人员调度：模型、方法与应用。 *European Journal of Operational Research* 283(2):405–425.Holzhauser M,KrumkeSO, Thielen C,2017a 广义处理网络中的最大流。 *Journal of Combinatorial Optimization* 33(4):1226–1256.Holzhauser M, KrumkeSO,Thielen C, 2017b 预算约束最小费用流问题的网络单纯形法。*European journal of operational research* 259(3):864–872.Hosseininasab A, Van Hoeve WJ, 2021利用同步决策图进行精确多序列比对。*INFORMS Journal on Computing* 33(2):721–738.Hu Y, Lan J,Wan C, 2009 灵活可重构网络中不可分流问题的算法。 *2009*年第四届计算机科学与技术前沿国际会议，543–547（IEEE）。

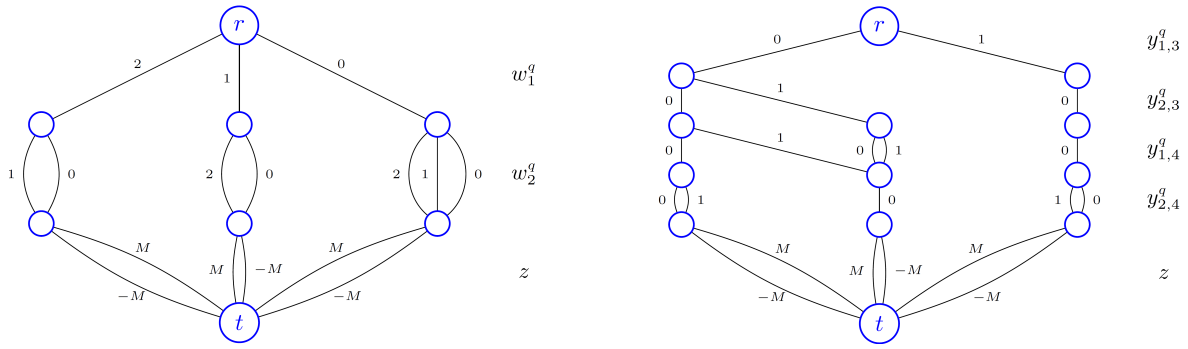
Huntley CL, Brown DE, Sappington DE, Markowicz BP, 1995*CSX*运输公司的货运路径规划与调度。*Interfaces* 25(3):58–71.Lcy¨uz IE, Richard JPP, Eskigun E, Acharya D, 2016月度煤炭列车预留规划问题的双模型求解方法。*Transportation Science* 50(3):926–946.Jin G, He S, Li J, Guo X, LiY, 2019 随机需求下高速铁路可变列车长度与停站时间的列车停站规划方法。*IEEE Access* 7:129690–129708.Jordan WC, Turnquist MA,1983 铁路货车分配的随机动态网络模型。 *Transportation Science* 17(2):123–145.Jovanovi´cD, Harker PT, 1991 铁路运营战术调度：*scan i*系统。 *Transportation Science*25(1):46–64.Kleinberg JM, 1996不相交路径问题的近似算法。博士论文，麻省理工学院。 Kolman P, Scheideler C, 2006不可分流问题的改进界。 *Journal of Algorithms*61(1):20–44.Kwan RS, 2011成功的列车乘务组排班优化案例研究。 *Journal of Scheduling*14(5):423–434.Larsen R, Pranzo M, D'Ariano A, Corman F, Pacciarelli D,2014 最优列车时刻表对作业时间随机扰动的敏感性。*Flexible Services and Manufacturing Journal* 26(4):466–489.Lawley M, Parmeshwaran V, Richard JP, Turkcan A, Dalal M, Ramcharan D,2008 用于优化定期散装铁路货车运输的时空调度模型。*Transportation Research Part B: Methodological*42(5):438–454.Layeb SB,JaouaA, JbiraA,Makhlouf Y,2018 随机货运运输调度的仿真优化方法。*Computers & Industrial Engineering* 126:99–110.Lin Z, Kwan RS,2014 实际列车单元调度的两阶段方法。 *Public Transport* 6(1-2):35–65.Lin Z,KwanRS, 2016 求解列车单元调度问题的分支定价方法。*Transportation Research Part B: Methodological* 94:97–120.Lin Z,Kwan RS, 2018 列车单元调度优化中的冗余连挂与解编。 *Electronic Notes in Discrete Mathematics* 64:45–54.Liu SQ, Kozan E,2011 容量约束下煤炭铁路网络的优化。 *Flexible Services and Manufacturing Journal* 23(2):90–110.Lusby RM, 2008 列车经铁路枢纽运行的优化方法。博士论文，ResearchSpace@ Auckland。

Lusby RM、Larsen J、Ehrgott M、Ryan D，2011年 铁路轨道分配：模型与方法。 *OR Spectrum*33(4):843–883。Meng L、Zhou X，2011年 动态随机环境下鲁棒单线列车调度模型：基于情景的滚动时域求解方法。*Transportation Research Part B: Methodological* 45(7):1080–1102。Quaglietta E、Corman F、Goverde RM，2013年 随机动态环境下铁路调度方案的稳定性。*RailCopenhagen2013*：第五届铁路运营建模与分析国际研讨会（*IAROR*）（交通规划与系统研究所，苏黎世联邦理工学院）。Salemi H、Davarnia D，2022a 决策图可表示混合整数规划的结构及其在机组组合中的应用。 *Operations Research* 网址https://doi.org/10.1287/opre.2022.2353。Salemi H、Davarnia D，2022b *SGUFP*测试实例。 https://doi.org/10.5281/zenodo.6373664。Serra T、Hooker JN，2019年 近似最优整数规划解的紧凑表示。 *Mathematical Programming* 1–34。Shen Y、Peng K、Chen K、Li J，2013年 采用自适应染色体的进化型乘务排班。 *Transportation Research Part B: Methodological* 56:174–185。Sherali HD、Suharko AB，1998年 用于空车管理的战术决策支持系统。*Transportation Science* 32(4):306–329。Turner C 、 Tiwari A 、 Starr A 、 Blacktop K，2016 年 欧洲和英国铁路行业关键规划与调度工作的综述。《机械工程师学会会报*F*辑：铁路与快速交通期刊》230(3):984–998。Walkowiak K，2006年 不可分流问题的新算法。 计算科学及其应用国际会议，1101–1110（施普林格）。Ying Cs、Chow AH、Chin KS，2020年随机需求下考虑铁路车辆周转的地铁列车调度行动者-评论家深度强化学习方法。*Transportation Research Part B: Methodological* 140:210–235。Zwaneveld PJ、Kroon LG、Van Hoesel SP，2001年 基于节点装箱模型的铁路车站列车径路安排。*European Journal of Operational Research* 128(1):14–33。

附录A：主问题表述比较

本节将介绍主问题公式（4）中变量空间 $\boldsymbol{w}$ 的决策图与原始 $\boldsymbol{y}$ 空间中的决策图之间的差异。首先，我们通过例3说明这些决策图的规模差异。

EXAMPLE 3. 考虑一个有向图 $G=(V,A)$，其节点集为 $V=\{1,2,q,3,4\}$，弧集为 $A=\{(1,q),(2,q),(q,3),(q,4)\}$，其中中心节点 q受 NSNM 约束限制。令 $\text{ind}^-(1,q)=\text{ind}^+(q,3)=1$，且 $\text{ind}^-(2,q)=\text{ind}^+(q,4)=2$。那么，图9(a)和图9(b)中展示的分别具有三层和五层弧层的精确决策图，表示主问题 (4) 和 (1a)-(1d) 的可行域，其中$-M$ 和 M 是变量 z的有效界。



(a) 变量 $\boldsymbol{w}$ 空间中的决策图。弧旁的数字表示标签。

(b) 在 $\boldsymbol{y}$ 变量空间中的决策图。弧旁的数字表示标签。

图9 $\boldsymbol{w}$ 和 $\boldsymbol{y}$ 变量空间中决策图的弧层数比较

从上述示例可以明显看出，在 $\boldsymbol{w}$ 空间中使用决策图的主要优势在于减少弧层数，而弧层数是决定决策图计算效率的主要因素。具体而言，尽管这类决策图在各层的节点数更多，但可以构建松弛决策图来限制宽度，从而在较低维度下提供高效的松弛决策图；而在 $\boldsymbol{y}$ 变量空间中构建的决策图，其松弛仍然具有更高的维度。

为评估求解方法相对于决策图空间的计算效率，我们比较了 DD-BD 方法在两种不同设置下的表现：(i) 在 $\boldsymbol{w}$ 变量空间中构建决策图，记为 DD-BD-$\boldsymbol{w}$；(ii) 在 $\boldsymbol{y}$ 变量空间中构建决策图，记为 DD-BD-$\boldsymbol{y}$。我们在表5和表6中报告了这两种实现针对 $|V'|\in\{40,80\}$ 以及五种不同情景的结果。

从这些表格可以看出，随着问题规模（场景数量）的增大，DD-BD-$\boldsymbol{w}$ 求解所有实例的速度都快于 DD-BD-$\boldsymbol{y}$，耗时改善了几个数量级。这些初步计算结果表明，在变量变换空间中为 SGUFP 设计 DD-BD 方法具有优势。

表5 DD-BD-w 和 DD-BD-y 在 $|V'|=40$下的运行时间（秒）。

实例编号	模型	场景数量 50	100	150	200	250
1	DD-BD-w	56.94	129.87	163.43	219.02	274.36
	DD-BD-y	89.68	304.08	432.34	642.70	839.57
2	DD-BD-w	42.60	82.65	128.16	164.52	208.94
	DD-BD-y	68.23	148.76	244.53	344.86	605.04
3	DD-BD-w	53.32	93.58	113.93	178.65	217.33
	DD-BD-y	83.05	157.67	310.07	541.33	658.98
4	DD-BD-w	46.61	87.81	130.19	183.23	253.72
	DD-BD-y	78.11	149.26	325.31	460.73	694.57
5	DD-BD-w	67.04	104.78	138.46	195.69	231.74
	DD-BD-y	109.61	223.78	351.80	532.12	669.78

表6 DD-BD-w 和DD-BD-y 在 $|V'|=80$下的运行时间（秒）。

实例编号	模型	场景数量 50	100	150	200	250
1	DD-BD-w	256.12	500.52	757.68	1025.88	1278.13
	DD-BD-y	483.42	977.03	1642.27	3175.72	4230.29
2	DD-BD-w	184.34	379.04	724.66	1088.21	1587.90
	DD-BD-y	340.13	864.21	1856.96	3010.55	4843.67
3	DD-BD-w	285.13	518.46	778.97	1046.39	1326.22
	DD-BD-y	568.32	1176.44	2401.98	3326.76	4283.58
4	DD-BD-w	263.78	665.30	1230.81	1277.93	1444.02
	DD-BD-y	501.04	1430.77	2868.92	3356.39	4356.48
5	DD-BD-w	187.34	376.96	564.34	1205.54	1412.94
	DD-BD-y	354.37	781.18	1279.73	3001.72	3834.08

附录 B：更多计算实验

本节将展示更多数值结果，以评估 DD-BD 方法求解较大问题实例的能力极限。结果列于表 7 和表 8 中，各列的定义与表 1–4 类似。对于这些实例，时间限制设为 3600 秒，符号“>3600”表示问题未能在此时间限制内求解。

表 7 $|V'|=120$ 的 DD-BD 运行时间（秒）。

实例编号	模型	场景数量 50	100	150	200	250
1	DD-BD	1494.49	2824.58	>3600	>3600	>3600
2	DD-BD	975.47	1892.41	3198.18	>3600	>3600
3	DD-BD	1150.30	2263.09	3454.47	>3600	>3600
4	DD-BD	1261.59	2403.79	>3600	>3600	>3600
5	DD-BD	906.34	1863.15	3050.68	>3600	>3600

表 8 $|V'| = 150$**的 DD-BD 运行时间（秒）。**

实例编号	模型	场景数量 50	100	150	200	250
1	DD-BD 2496.16	> 3600	> 3600	> 3600	> 3600	> 3600
2	DD-BD 2944.20	> 3600	> 3600	> 3600	> 3600	> 3600
3	DD-BD 2321.62	> 3600	> 3600	> 3600	> 3600	> 3600
4	DD-BD 2590.34	> 3600	> 3600	> 3600	> 3600	> 3600
5	DD-BD 2298.36	> 3600	> 3600	> 3600	> 3600	> 3600