arXiv:2602.12350v1 [cs.CC] 12 Feb 2026

# 由 NP 导出的许多自然问题在多项式层次和 PSPACE 中的完备性

**Christoph Grüne**
德国亚琛工业大学计算机科学系 gruene@algo.rwth-aachen.de

**Berit Johannes**Flix，德国
慕尼黑 berit@mit.edu

**James B. Orlin**
麻省理工学院斯隆管理学院，美国马萨诸塞州剑桥 jorlin@mit.edu

**Lasse Wulf**
理论计算机科学系，哥本哈根信息技术大学，丹麦 lasw@itu.dk

**摘要**

许多源自 NP 的自然优化问题都具有双层和多层扩展，其中多个目标函数相互冲突的玩家依次做出决策，例如阻断、对抗性选择和可调鲁棒优化。这类问题可以自然地用交替量词建模，因此超出了 NP 的范围，通常位于多项式层次或 PSPACE 中。尽管这些问题类别已得到广泛研究，但在这些更高层级上已知的自然完备性结果仍相对较少。我们提出一个通用框架，用于证明源自 NP 的问题在多项式层次和 PSPACE 中的完备性。我们的方法基于对 NP 的一种细化，我们称之为*NP with solutions*（NP-S）；在该类中，解是显式的组合对象。此外，我们还引入一类受限的归约——*解嵌入归约*——它们能够保留解结构。我们定义*NP-S*完全性，并证明大量经典 NP完全问题，包括团、顶点覆盖、背包和旅行商问题，都是 NP-S 完全问题。

利用这一框架，我们建立了一般性的元定理，表明如果一个问题是 NP-S- 完全的，那么其自然的两层扩展是 $\Sigma_2^p$ -完全的，三层扩展是$\Sigma_3^p$ -完全的，而 $k$ 层扩展是 $\Sigma_k^p$ -完全的。当层数无界时，由此产生的问题是 PSPACE 完全的。我们的结果涵盖了此前已知的、几乎所有从 NP 衍生的多层优化问题的完全性结果，并同时得出许多新结果，表明高计算复杂性是 NP完全问题多层扩展的一种普遍特征。

**2012 年 ACM 主题分类**计算理论 → 问题、归约与完全性

**关键词与短语**计算复杂性、双层优化、鲁棒优化、斯塔克尔伯格博弈、极小极大遗憾、最关键节点、最关键顶点、最关键边、阻断问题、阻断问题、两阶段问题、多项式层次、$\Sigma_2$、$\Sigma_3$

**资助** *Christoph Grüne*：由德国研究联合会（DFG）资助——GRK 2236/2。*Lasse Wulf*：由奥地利科学基金（FWF）资助：W1230；并获嘉士伯基金会资助（CF21-0302“具有几何应用的图算法”）。

## 1 引言

许多自然的判定问题和优化问题，是通过对 NP 中的经典问题进行扩展而得到的，扩展方式包括引入顺序决策、对抗性决策或多层决策。这类扩展例如出现在阻断、对抗性选择和多阶段优化中，其中一名玩家先行动，另一名玩家则作出最优应对。从复杂性理论的角度来看，这些问题已不再属于 NP 中的判定问题，而是位于多项式层次的更高层级或 PSPACE 中。本文旨在系统地理解这类源自 *NP*的问题的复杂性，并解释为何其多层变体通常是这些更高复杂度类中的完全问题。

近年来，NP 中问题的扩展受到多个领域的广泛关注，包括双层优化 [DZ20、KLLS21], 鲁棒优化 [BGN09、GH24],网络阻断 [SPG13], 斯塔克尔伯格博弈 [LS17], 攻击者—防御者博弈 [HZ23],以及许多其他双层问题。这些研究领域对于帮助我们了解网络中哪些部分最容易受到攻击、防止恐怖袭击、理解经济过程，以及了解并改善许多系统的鲁棒性至关重要。

这些领域的共同点在于其固有的多层结构：多个参与者依次做出决策，且其目标可能彼此冲突——这通常通过交替量词或博弈论交互来建模。例如，在双层优化和斯塔克尔伯格博弈中，领导者在决策时会预判跟随者的最优响应；在网络阻断和攻击者—防御者博弈中，一方在预算约束下试图扰乱或保护系统功能，同时预判对手采取的反制措施。这种多层特性使此类问题上升至多项式层次的更高层级，其中存在量词（“存在一种策略”）与全称量词（“对于所有可能的攻击”）之间的交替反映了现实世界中的对抗动态。因此，这些问题可以刻画为两个玩家之间的一种抽象两回合博弈：存在玩家和全称玩家。先手玩家（以下简称 Alice）开始博弈，并采取行动以实现某个目标。随后，后手玩家（以下简称 Bob）对 Alice 作出响应。通常但并非总是如此，Alice 和 Bob 的目标彼此相反。

我们以最大团阻断问题 [PBP14, PPR16, FLMS19, Paj20]为一个典型例子。在这个问题中，给定一个图 $G=(V,\ E)$、一个子集 $V\prime\subseteq V$、某个预算 $\Gamma\in\mathbb{N}$以及某个阈值 $t\in\mathbb{N}$。Bob 的目标是在图中找到一个大小为 $t$的团。然而，在 Bob 行动之前，Alice 可以从图中删除至多 $\Gamma$ 个顶点，以损害 Bob 的目标。因此，这个博弈是一个双层问题，可描述为

$$\exists B\subseteq V'\text{ with }|B|\leq\Gamma:\forall C\subseteq V:\text{If }C\text{ is a clique in }G\text{ of size }t,\text{ then }B\cap C\neq\emptyset.$$

我们指出，这个问题遵循一种自然的模式，研究人员经常据此提出新问题：首先，选取某个名义问题（此处为团），然后通过添加一个额外的组成部分（此处为 Alice 可以进行阻断）将其改造成更复杂的多层问题。

经典阻断问题的一种自然扩展是保护-阻断博弈，它们为多项式层次和 PSPACE 中的判定问题提供了典型来源。在这类博弈中，Alice（攻击者）和 Bob（防御者）展开对决；双方各有自己的预算（Alice 的预算为$\Gamma_A$，Bob 的预算为 $\Gamma_B$），用于攻击或保护全集中的个别元素。博弈按回合进行：首先，Bob 保护一些元素；然后，Alice 阻断未受保护的元素；双方如此交替，直到到达第 $k$ 回合，此时 Bob 必须仅使用幸存的元素找到一个有效解。如果 Bob 赢得

能够构造出一个解，则 Bob 获胜；否则，Alice 获胜。交替行动的次数直接决定了相应判定问题的复杂度，使其处于多项式层次的更高层级；若回合数属于输入的一部分，则问题属于 PSPACE。

## 1.1 NP 归约的结构性质

本文的一个关键洞见是，许多经典的 NP 完全性归约所保留的结构，远超证明 NP 困难性所需的程度。在许多标准归约中，源问题解中的元素与目标问题解中的元素一一对应。因此，通常可以从目标问题的解中谨慎地选取一部分元素，重构出源问题的解；这些元素与源问题解中的元素一一对应。这种现象在教科书中的归约里十分常见。例如，在从 Sat 到顶点覆盖、独立集或团等问题的归约中，满足赋值通过从变量构件中选取顶点而被直接编码；反过来，每个极小解都会导出一个有效赋值。尽管 NP 完全性并不要求具备这种结构，但它在实践中无处不在。在双层和多层优化中，这种额外结构至关重要。在阻断和多阶段模型中，对手影响的不只是可行性，还会直接作用于可能出现在解中的各个元素。因此，计算问题不仅取决于解是否存在，也取决于解由哪些部分组成。这些观察促使我们提出一类受限归约，并由其显式保留解结构。粗略地说，这类归约将一个问题的解中的元素嵌入另一个问题的解中的元素，并以此确保两个方向上的可行性均得以保留。我们将这类归约称为解嵌入归约。它们在我们的元定理中发挥核心作用，使我们能够系统地将 NP 完全性证明提升为多项式层次更高层上的完全性结果。

利用保结构归约，将 NP 完全性结果提升至多项式层次更高层级的这一思路，最早由 Johannes 在其博士论文 [Joh11]，中提出，并应用于多种对抗性问题类和多层问题类。Grüne 和 Wulf [GW24] 在 Johannes 的工作基础上，证明了 Johannes 的思路可以在一个形式化框架内表述，并展示了该框架适用于多项式层次第二或第三阶段的许多常见问题类。本文可视为对 [Joh11, GW24] 在多个方向上的拓展。我们将全面介绍这一框架，并从复杂性角度阐释它与 NP 类的确切关系。随后，我们将结果拓展至多项式层次的任意多个阶段以及 PSPACE，并介绍一种新颖的证明技巧。

## 1.2 带解的 NP

NP 的标准表述将解视为不透明的证书；一旦能够在多项式时间内验证可行性，证书的内部结构便无关紧要。这种抽象足以处理单层判定问题，但当解本身会被操作、限制或对抗性修改时——例如双层和多层优化中的情形——这种抽象就不再适用。为此，我们提出对 NP 的一种细化：在这一类别中，解是由从明确定义的全集中取出的元素构成的显式组合对象。我们将这一类称为含解的 *NP* （*NP-S*）。直观地说，NP-S 中的问题不仅包括一组实例，还包括对何为解以及哪些元素构成该解的规定。解的不同表示方式——例如同一图问题的基于顶点表述与基于边表述——被视为不同的问题，

这反映了它们会产生不同的双层和多层变体。NP-S 的作用并非在 NP 之外引入额外的计算能力，而是明确呈现许多经典归约已隐式利用的解结构。在这一框架下，可以精确定义解嵌入归约，并严格分析其对双层和多层问题的影响。在本文中，我们证明至少以下 26 个经典问题是 NP-S 完全的：

可满足性、3-可满足性、顶点覆盖、支配集、集合覆盖、击中集、反馈顶点集、反馈弧集、无容量设施选址、p-中心、p-中位数问题、三维匹配、独立集、团、子集和、背包、划分、调度问题、有向/无向哈密顿路径、有向/无向哈密顿回路、旅行商问题、两条有向顶点不相交路径、$k$-顶点有向不相交路径、斯坦纳树。

所有这些问题的形式化描述见 A 节。此外，只需从任何一个已经是 NP-S 完全的问题出发，找到一个解嵌入归约，就可以轻而易举地将新问题加入 NP-S 类。由于我们很容易证明许多经典问题都是 NP-S 完全的，因此预计未来会有更多问题加入其中。网站 reductions.network [GP25] 收录了 NP-S 类中的问题，目前登记了 54 个问题。

**提升至多项式层次及更高层次。**

一旦解结构得以保留，多层构造便可以从计算复杂性理论视角进行系统分析。我们元定理背后的核心思想是，多层优化问题可以诠释为一种博弈：玩家在最终解确定之前，轮流限制或选择全集中的元素。由于解嵌入归约保留了部分解，因此也保留了这些多层交互所产生的影响。

这一观察使我们能够以统一的方式提升困难性结果。我们首先证明，在多项式层次的目标层级上，某个可满足性问题是完备的。随后，解嵌入确保相同的复杂性也适用于所有 NP-S 完全问题。由此，阻断和对抗性选择等自然的双层问题继承 $\Sigma_2^p$ -完备性，两阶段可调优化等三层问题继承 $\Sigma_3^p$ -完备性，而更一般地， $k$-层问题继承 $\Sigma_k^p$ -完备性。

该框架并不局限于常数个决策阶段。当交替行动的次数允许随输入规模增长时，相应的博弈便成为 PSPACE 完全问题。这一推广自然源于相同的结构性质，也体现了多项式有界交替与 PSPACE 之间的经典对应关系。因此，NP-S 和解嵌入为多项式层次及 PSPACE 中的困难性提供了统一解释。

这一洞见使我们能够用统一的论证取代冗长且针对特定问题的困难性证明。与其为每个多层问题构造专门的归约，只需验证其基础 NP问题存在解嵌入归约即可。一旦完成这一验证，其双层和多层变体的复杂性便可系统地推导出来。从这个意义上说，复杂性从 NP 上升到多项式层次更高层级，直接源于现有 NP完全性证明的结构。

## 1.3 我们的贡献

本文的核心贡献是一系列元定理，它们能够系统地将 NP 完全性结果提升为多项式层次中的完全性结果，并在更一般的情形下提升为 PSPACE 中的完全性结果。

我们将这些元定理应用于若干重要的多层优化问题类别，包括阻断和多阶段可调优化。通过这样做，我们将这些领域中几乎所有已知的困难性结果作为特例重新推导出来，并同时建立了许多新的完全性结果。这表明，高计算复杂性并非例外，而是 NP完全问题的多层扩展所普遍具有的特征。

我们指出，早期论文仅针对单个问题证明了 $\Sigma_k^p$ -完全性。相比之下，我们的元定理将网络阻断和两阶段可调鲁棒优化领域中几乎所有已知的 $\Sigma_k^p$ -完全性结果都作为特例涵盖在内（即阻断最大团 [Rut93], 阻断背包 [CCLW14、TCCH24],以及两阶段可调旅行商问题、顶点覆盖、独立集 [GLW24]。

虽然上述每一项早期工作都只证明了一个特定问题是 $\Sigma_k^p$ -困难的，但我们的元定理同时证明了一大类问题是 $\Sigma_k^p$ 完全或 PSPACE 完全的，其中包括 A 节中提到的所有问题，以及未来可能加入的其他问题。此外，我们还回答了 [GLW24、GW24]提出的开放问题：对于阻断博弈或多阶段可调鲁棒优化等多层问题，是否存在通用的元归约。

我们具体确立了以下结果。

**对抗性选择博弈**。在第 4.2 节中，我们研究对抗性选择博弈，它是将量化布尔可满足性自然推广到多层优化框架下任意 NP-S 问题的一种形式。对抗性选择博弈定义在 NP-S 类中的一个名义问题上。两名玩家 Alice 和 Bob 轮流从该名义问题的全集中选择元素。Alice 希望构造一个解，而 Bob 则希望阻止 Alice 做到这一点。本节中，我们介绍一种专为解嵌入归约设计的证明技巧，并称之为“对偶模仿”。利用对偶模拟证明，我们证明 NP-S 完全问题上的对抗性选择博弈是$\Sigma_k^p$ 完全的；如果 $k$ 是输入的一部分，则为 PSPACE 完全。

**保护-干预博弈**。在第 5 节中，我们研究保护-干预博弈。在这种博弈中，有两名玩家：一名保护者和一名阻断者。他们各自拥有一定的预算：保护者最多可以保护 $\Gamma_P$ 个元素，而阻断者最多可以阻断 $\Gamma_I$ 个元素。随后，两名玩家轮流选择元素进行保护或阻断。阻断者无法阻断受保护的元素，保护者也无法保护被阻断的元素。在最后阶段，保护者需要仅使用未被阻断的元素，为名义问题找到一个解（例如哈密顿回路、顶点覆盖、最大团等）。如果保护者能够做到这一点，他就获胜；否则，阻断者获胜。我们的主要结果是，对于每个 $\Pi \in$ NP-S 问题，相应的 $k$ 阶段保护-干预问题是 $\Sigma_k^p$ 完全的；或者，如果 $k$ 是输入的一部分，则为 PSPACE 完全。

**多阶段可调鲁棒优化**。在第6节中，我们研究具有离散预算不确定性的多阶段可调鲁棒优化（也称为离散 $\Gamma$-不确定性）。

决策者必须在第一阶段做出决策，但此时尚未完全了解未来决策的成本，甚至可能还不知道未来有哪些决策可供选择（即期决策）。后续阶段的决策则是在部分不确定性

揭示之后做出的（等待观察）。在这一范式中，每个阶段都会出现新的决策元素。阶段开始时，对手可以封锁部分可用的决策元素。随后，决策者对剩余元素做出决策。

对于 NP-S问题， $k$阶段可调鲁棒版本属于 $\Sigma^p_{2k-1}$。

我们在第6节中的主要结果是：对于每个 $\Pi \in$NP-S问题，相应的$k$阶段可调问题在采用离散预算不确定性时是 $\Sigma^p_{2k-1}$完全的；或者，如果 $k$ 属于输入的一部分，则是 PSPACE 完全的。

**替代NP-S 模型**。在第7节中，我们考察对 NP-S问题进行建模的替代方式，并讨论我们的框架相对于这些选择的稳健性。特别是，我们研究不同的解表示方式——例如替代全集，或由不同验证器诱导的编码——如何产生不同的 NP-S 问题，即使它们对应于同一个底层 NP决策问题。我们证明，解嵌入的概念及由此得到的元定理能够自然地扩展到这些替代 NP-S模型。这表明，本文所发展的框架并不局限于单一的规范表示，而是捕捉到了一个在广泛的自然解模型中持续存在的结构性现象。

**汇编NP-S完全问题**。A 节汇集了 NP-S 完全问题，既可作为参考，也可作为应用我们元定理的实用指南。对于每个问题，我们都指定了一个自然的全集和解结构，并说明如何通过解嵌入归约推出其 NP-S 完全性。这份汇编将大量经典 NP 完全问题整合到一个统一框架中，并明确指出哪些问题表示适合提升到多项式层次的更高层级。因此，读者可以轻松识别出新的多层问题——例如对抗性选择、阻断或多阶段变体——这些问题是 $\Sigma^p_k$ 或 PSPACE 完全的，无需进行额外的针对特定问题的归约。

## 1.4 文献综述

由于篇幅所限，不可能在一篇论文中涵盖多层优化的所有子领域。本文将重点放在阻断和可调鲁棒优化领域。我们选择这些领域，是因为它们是多层优化中备受关注的子领域，而且我们能够在保持简洁的同时，展示适用于这些不同情形的多种技术。事实上，这些热门领域各自都有数百篇论文，因此不可能给出完整的综述。我们将这篇文献综述分为两部分：一般领域概述，以及专门针对 $\Sigma^p_k$ 完备性的相关工作概述。

### 一般领域

在网络阻断领域，人们关注的是网络中哪些部分最容易受到攻击或故障的影响。这类问题可以表述为一个极小极大问题，也可以设想为 Alice 和 Bob 之间的一场博弈：Alice 从网络中阻断有限个元素，以干扰 Bob 的目标函数。

网络阻断问题的一些轻微变体也称为最关键节点/最关键边问题以及顶点/边阻断问题。网络阻断已在大量标准优化问题中得到研究，其中包括 P类中的问题，例如最短路径 [MMG89, BNKS98, KBB+08]、匹配 [Zen10],、最小生成树 [LC93], 以及最大流 [Woo93]。这些文献证明了相应阻断问题变体的 NP 困难性（但 [MMG89] 证明，

单一最关键弧问题属于 P类）。此外，针对 NP完全的阻断问题也开发了相应算法，例如顶点覆盖 [BTT10, BTT11], 、独立集 [BTT10, BTT11, BBPR15, PPR17, HLW23], 、着色 [BBPR15, PPR16, PPR17],、团 [PBP14, PPR16, FLMS19, Paj20], 、支配集 [PWBP15], , 以及 1- 和 $p$-中心[BTV10, BTV13], 1- 和 $p$-中位数 [BTV10, BTV13]。通常，对这些问题的复杂性分析不会超出 NP 困难性。Smith、Prince 和 Geunes 提供了一份综合综述 [SPG13]。多层阻断通常称为保护-干预或加固—阻断博弈。在 [TCCH24], 中，分析了背包问题的不同变体，其中包括阻断背包、加固—阻断背包和多层加固—阻断背包。此外， [NCH22]分析了多层关键节点问题。

在多阶段可调鲁棒优化领域中，问题涉及不确定情景。决策者必须做出多步决策，在每一步中，都必须在不了解情景全部信息的情况下做出部分决策，直到最后一个阶段，才能在充分了解情景的情况下做出部分校正决策。对于大量标准优化问题，人们已经研究了两阶段可调鲁棒优化，例如 [KZ15、KZ16、KZ17、CGKZ18、GLW22]。Yanıkoğlu、Gorissen 和 den Hertog [YGdH19] 提供了一份综合综述。此外，以下论文也讨论了多阶段可调鲁棒优化。 [GLW24] 分析了顶点覆盖、独立集和旅行商问题的复杂性。此外，多阶段可调鲁棒性与带估计的在线优化密切相关。相关结果见于 [GGI+15、GLR24、BGL+25]（针对背包问题）， [GKN25]（针对旅行商问题）以及 [GU25] （针对装箱问题）。

**密切相关的工作**

尽管人们对双层优化抱有极大兴趣，且尽管复杂度类$\Sigma_k^p$ 是此类判定问题的自然复杂度类，但据我们所知，关于这些领域中的 $\Sigma_k^p$ -完备性的出版物寥寥可数。Rutenburg [Rut93] 证明了最大团阻断问题具有 $\Sigma_2^p$ -完备性。Caprara、Carvalho、Lodi 和 Woeginger [CCLW14] 证明了由 DeNegre [Den11]最初引入的背包问题阻断变体具有 $\Sigma_2^p$ -完备性。Fröhlich 和 Ruzika[FR21] 证明了两个版本的选址阻断问题具有 $\Sigma_2^p$ -完备性。Nabli、Carvalho 和 Hosteins [NCH22]研究了多层关键节点问题，并证明其具有$\Sigma_3^p$ -完备性。Goerigk、Lendl 和 Wulf [GLW24] 证明了旅行商问题、独立集和顶点覆盖的两阶段可调变体具有 $\Sigma_3^p$ -完备性。Grüne [Grü24] 提出了一个归约框架，用于所谓的宇宙小工具归约，以证明典型优化问题的可恢复鲁棒版本具有 $\Sigma_3^p$ -完备性。Tomasaz、Carvalho、Cordone 和 Hosteins [TCCH24] 证明了一个阻断背包问题具有 $\Sigma_2^p$ -完备性，并证明一个 $k$ 层强化阻断背包问题具有$\Sigma_k^p$ -完备性。

Umans 和 Schaefer 编写的汇编 [SU08] 收录了许多 $\Sigma_2^p$ -完全问题，但其中很少有与双层优化相关的问题。Jeroslow 的开创性论文 [Jer85] 证明了一般多层规划的 $\Sigma_k^p$ -完备性。

据我们所知，这补全了已知的双层和多层优化中的 $\Sigma_k^p$ 完全性结果列表。尽管这一领域作为一个活跃的研究领域已经持续了二十多年，但现有结果却如此之少，实在令人惊讶。相比之下，关于大量问题的 NP完全性证明已经广为人知。根据 Woeginger [Woe21], 的说法，这可能是因为“目前，建立 $\Sigma_2^p$ 完全性通常十分繁琐，而且主要是通过冗长的归约来完成的，这些归约一路追溯到 2-量化

可满足性’。

我们以简要讨论复杂性理论中的一个相关概念结束本小节。从复杂性理论的角度来看，我们所说的解嵌入归约在思想上与关于保数归约和 #P 完全性的经典工作相关 [Val79]。保数归约保持解的确切数量，在计数问题的研究中发挥着核心作用。虽然这两种概念都通过保持额外结构而超越了标准的多对一归约，但它们关注的是计算复杂性的不同方面。具体而言，#P 完全性关注的是对解进行计数，而我们的框架则聚焦于多层决策问题中解在限制和交替下的行为。因此，解嵌入归约适合将 NP 完全性结果提升到多项式层次和 PSPACE，而不是用于计数复杂性。

## 1.5 技术概述

我们简要概述用于证明主要定理的技术和思想。为此，有必要正式说明元定理适用于哪一类问题。复杂性类 NP-S 中的问题是形如 $(\mathcal{I},\ \mathcal{U},\ \mathcal{S})$ 的三元组。这里，$\mathcal{I} \subseteq \{0,\ 1\}^*$ 是问题的输入实例集合，这些实例以二进制字符串编码。对于每个输入实例 $I \in \mathcal{I}$，我们假设存在一个全集 $\mathcal{U}(I)$，以及一个解集 $\mathcal{S}(I) \subseteq 2^{\mathcal{U}(I)}$，其中包含构成解的全集的所有子集。

通过一个例子最能说明我们的想法。考虑顶点覆盖问题。给定一个实例 $I$，我们有一个无向图 $G = (V,\ E)$ 和一个阈值 $t \in \mathbb{Z}_{\geq 0}$。图 $G$ 的顶点覆盖是一个顶点集合 $C \subseteq V$，使得每条边 $\{u,\ v\} \in E$ 都有一个端点属于 $C$，即 $u \in C$ 或 $v \in C$。问题是是否存在大小至多为 $t$ 的顶点覆盖。按上述含义将顶点覆盖问题解释为一个 NP-S 问题，意味着：输入实例是一个以二进制编码的元组 $I = (G,\ t)$。与某个输入实例 $I$ 关联的全集为 $\mathcal{U}(I) = V$，解集为 $\mathcal{S}(I) = \{C \subseteq V\colon\ C$ 是 $G$ 中大小至多为 $t$ 的顶点覆盖$\}$。问题是 $\mathcal{S}(I) \neq \emptyset$ 是否成立。

此外，我们考虑一种从某个问题 $\Pi_1$ 到另一个问题 $\Pi_2$ 的 NP 完全性归约，它具有一种特殊性质，我们称之为解嵌入性质（简称 SE 性质）。该性质指出，$\Pi_1$ 的全集可以单射地嵌入 $\Pi_2$ 的全集，并满足以下两个性质：**(P1)** $\Pi_1$ 的每个解都对应于 $\Pi_2$的一个部分解；**(P2)** 将 $\Pi_2$的每个解限制到该嵌入的像后，都对应于 $\Pi_1$的一个解。

我们证明，对历史上第一个 NP 完全性归约（即 Cook 和 Levin 用来证明可满足性是 NP 完全的归约）进行一种自然的修改后，该归约具有 SE 性质（定理 7）。我们还证明，文献中已知的大量 NP 完全性归约实际上都具有 SE 性质（A 节）。

我们通过考察从 3-可满足性（3-SAT）到顶点覆盖的归约来说明 SE 性质。

3-SAT 的输入是一个布尔公式 $\varphi$，其子句为 $c_1,\ \ldots\ldots,\ c_m$。我们考虑 3-SAT 的 NP-S 版本，其中全集是所有文字构成的集合 $L = \{\ell_1,\ \ldots\ldots,\ \ell_n\} \cup \{\bar{\ell}_1,\ \ldots\ldots,\ \bar{\ell}_n\}$。$\varphi$ 的解集是所有由文字的子集构成、编码了满足解的集合，即

$$\mathcal{S}(\varphi) = \{L' \subseteq L : |L' \cap \{\ell_i, \bar{\ell}_i\}| = 1\ \forall i \in \{1, \ldots, n\}, L' \cap c_j \neq \emptyset\ \forall j \in \{1, \ldots, m\}\}.$$

以下是 Garey 和 Johnson 的著作中从 3-SAT 到顶点覆盖的经典 NP 完全性归约 [GJ79]。

给定一个由文字 $\{\ell_1, \dots, \ell_n\} \cup \{\bar{\ell}_1, \dots, \bar{\ell}_n\}$ 和子句 $C$ 组成的 3-SAT 实例，该归约按如下方式构造顶点覆盖的图 $G = (V, E)$：该图包含顶点 $W := \{v_{\ell_1}, \dots, v_{\ell_n}\} \cup \{v_{\bar{\ell}_1}, \dots, v_{\bar{\ell}_n}\}$，使得每个顶点 $v_{\ell_i}$ 通过一条边与顶点 $v_{\bar{\ell}_i}$ 相连。此外，对于每个子句，我们向图中添加一个新的三角形（包括三个新顶点），并添加三条额外的边，使得该三角形的三个顶点与该子句中出现的文字所对应的顶点相连。该归约如图 1 所示。

注意，$G$ 的每个顶点覆盖的大小至少为 $|L|/2 + 2|C|$，因为它必须包含 $\{v_{\ell_i}, v_{\bar{\ell}_i}\}$ 中的至少一个顶点，以及每个三角形中的至少两个顶点。以下事实易于验证：$G$ 具有大小为 $|L|/2 + 2|C|$ 的顶点覆盖，当且仅当该 3-SAT 实例为是实例。我们可以如下看出这一点。要使顶点覆盖的基数为 $|L|/2 + 2|C|$，它必须恰好包含 $\{v_{\ell_i}, v_{\bar{\ell}_i}\}$ 中的一个顶点，以及每个三角形中恰好两个顶点。此外，每个三角形中的第三个顶点必须与一个不在该三角形中的已选顶点相关联。

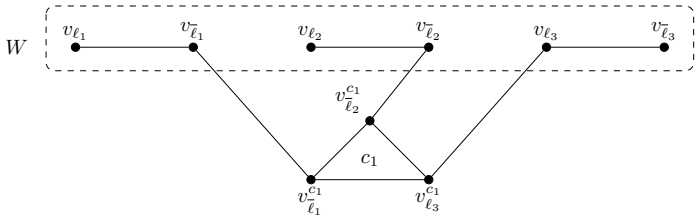


**图 1** 针对 $\varphi = (\ell_1 \vee \ell_2 \vee \ell_3)$ 的从 3-SAT 到顶点覆盖的经典归约。

我们现在描述一种将 3-SAT 的解映射为顶点覆盖的解的方式。设 $f\colon \mathcal{U} \to \mathcal{U}'$，其中 $f(\ell_i) = v_{\ell_i}$ 且 $f(\bar{\ell}_i) = v_{\bar{\ell}_i}$。函数 $f$ 将全集 $\mathcal{U}$ “嵌入”全集 $\mathcal{U}'$。它将 $\mathcal{U}$ 中的每个文字映射到 $\mathcal{U}'$ 中的一个唯一顶点。函数 $f$ 满足以下两个性质：假设 $W = f(L)$。

**(P1)** 如果 $S \subseteq L$ 是某个满足赋值对应的文字，则存在一个大小为 $|L|/2 + 2|C|$ 的顶点覆盖，其限制到 $W$ 等于 $f(S)$。
**(P2)** 如果 $V'$ 是大小为 $|L|/2 + 2|C|$ 的顶点覆盖，则 $f^{-1}(V' \cap W)$ 是满足赋值对应的文字的集合。

接下来，我们给出一种更简洁的方式来表述上述两个性质。令 $\mathcal{S}(\varphi) \subseteq 2^{\mathcal{U}}$ 表示 3-SAT 的解（即所有编码满足赋值的文字子集所构成的集合）。令 $k = |L|/2 + 2|C|$，并令 $\mathcal{S}'(G, k) \subseteq 2^{\mathcal{U}'}$ 表示顶点覆盖的解（即所有大小至多为 $k$ 的顶点覆盖所构成的集合）。那么，$f$ 的上述两个性质可以用以下等价方式表示：

$$\{f(S) : S \in \mathcal{S}(\varphi)\} = \{S' \cap f(\mathcal{U}) : S' \in \mathcal{S}'(G, k)\}. \tag{1}$$

如果从 $\Pi_1$ 到 $\Pi_2$ 的 NP 困难归约具有一个满足性质 (1) 的关联函数 $f$，我们就称其为解嵌入 (SE) 归约，并记作 $\Pi_1 \leq_{\text{SE}} \Pi_2$。我们的元定理基于 SE 归约。

我们引入 NP-S 类，作为 NP 类的对应概念。若问题 $\Pi \in$ NP-S 满足：对于每个其他的 $\Pi' \in$ NP-S 问题，都存在从 $\Pi'$ 到 $\Pi$ 的 SE 归约，则称该问题为 NP-S 完全问题。

## 10 源自NP的PH完备性

A 节列出了一系列 NP-S 完全问题。由于解嵌入归约具有传递性（我们稍后将证明这一点），只需从列表中选取一个 NP-S 完全问题 $\Pi'$，然后证明 $\Pi' \leq_{\mathrm{SE}} \Pi$，即可证明 $\Pi$ 是 NP-S 完全问题。

最后，我们解释如何利用 SE 归约得到一般的元归约。两个问题之间存在 SE 归约，意味着它们具有密切相关的解结构。我们利用这一性质，通过一个通用的修改算子 $\Phi$，将 NP-S 中的问题提升到多项式层次的更高层级；该算子会将 NP-S 中的问题转化为多层决策问题，例如对抗性选择、保护-干预或可调鲁棒优化。

关键思想是，只需证明 $\Phi(\Pi)$ 对某一个 NP-S 完全问题 $\Pi$（通常为 Sat）是完备的，然后将这一结果推广到所有 NP-S 完全问题。具体而言，我们首先考察修改后的问题 $\Phi$(Sat)，并利用标准技术证明它通过从典型完全问题 $k$-Quantified-Sat 出发的归约是 $\Sigma_k^p$ 完全的。接下来，我们考察任意一个 NP-S 完全问题 $\Pi'$。由于 $\Pi'$ 是 NP-S 完全的，因此存在一个从 Sat 到 $\leq_{\mathrm{SE}} \Pi'$ 的 SE 归约。

这一 SE 归约表明，对于任意 Sat 实例 $I$，我们可以构造一个 $\Pi'$ 实例 $I'$，使得 $I$ 可被视为 $I'$ 的子实例。特别地，存在一个单射，将全集 $\mathcal{U}(I)$ 映入全集 $\mathcal{U}'(I')$，同时保持解的拓扑结构（如 (1) 所给出）。我们证明，这一关系自然地延伸至提升问题，从而实例 $\Phi$(Sat) 可作为 $\Phi(\Pi')$ 的子实例嵌入。

基于这一嵌入，我们证明，由 $\Phi$(Sat) 所定义的博弈中玩家的任意最优策略，都可以借助第4.2小节所述的“对偶模仿”论证，在由 $\Phi(\Pi')$ 所定义的博弈中加以模仿。由于 $\Phi$(Sat) 是 $\Sigma_k^p$ -困难的，故 $\Phi(\Pi')$ 也是 $\Sigma_k^p$ -困难的。

## 2 预备知识

语言是一个集合 $L \subseteq \{0,1\}^*$。从语言$L$到语言 $L'$的多对一归约或 Karp归约是一个映射 $f\colon \{0,1\}^* \to \{0,1\}^*$，使得对所有$w \in \{0,1\}^*$，$w \in L$当且仅当 $f(w) \in L'$。图灵机 $M$ 连同判定问题 $L$ 的输入 $x$ 关联到谓词 $M(x)$，使得若 $x \in L$，则 $M(x) = 1$，若 $x \notin L$，则 $M(x) = 0$。

布尔变量 $x$ 是取值为 0 或 1 的变量。设 $X = \{x_1, \ldots, x_n\}$ 为一个变量集合。对应的文字集为 $L = \{x_1, \ldots, x_n\} \cup \{\overline{x_1}, \ldots, \overline{x_n}\}$。子句是文字的析取。若布尔公式是子句的合取，则称其为合取范式（CNF）。若其是文字合取式的析取，则称其为析取范式（DNF）。按照惯例，子句可表示为文字的一个子集，而 CNF 公式 $\varphi$ 可表示为子句的一个集合。例如，集合 $\{\{x_1, x_2\}, \{\overline{x_2}, x_3\}\}$ 对应于公式 $(x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3)$。我们用 $\varphi(X)$表示公式 $\varphi$ 仅依赖于 $X$。有时我们关注变量被划分的情形。我们用 $\varphi(X, Y, \ldots)$ 表示这种情形。变量赋值是一个映射 $\alpha\colon X \to \{0,1\}$。公式 $\varphi$ 在赋值 $\alpha$ 下的取值记为 $\varphi(\alpha) \in \{0,1\}$。对于变量经划分的公式 $\varphi(X, Y)$，我们也用 $\varphi(\alpha, Y)$ 表示其在 $X$ 上的部分赋值 $\alpha$。此外，如果 $\alpha$满足公式 $\varphi$，即 $\varphi(\alpha) = 1$，我们记为 $\alpha \vDash \varphi$。

集合 $U$ 将作为基集的记号，我们称之为全集。定义在 $U$上的成本函数是一个映射 $c\colon U \to \mathbb{R}$。对于每个子集 $U\prime \subseteq U$，我们将子集 $U'$的成本定义为 $c(U\prime) := \sum_{u \in U'} c(u)$。对于映射 $f\colon A \to B$及其某个子集 $A' \subseteq A$，我们将子集 $A'$的像定义为 $f(A') = \{f(a) : a \in A'\}$。对于某个 $B' \subseteq B$，我们将其

原像记为 $f^{-1}(B') = \{a \in A\colon f(a) \in B'\}$。在本文中，函数是否为成本函数显而易见，因此该记号的含义可根据上下文明确判断。如果 $A_1, \ldots, A_n$ 是一个集合族，我们将 $\widehat{A}_i = \bigcup_{j=1}^{i} A_j$.记为

## 3 复杂性类 NP-S

对于 NP 类中的问题，解通常是隐式描述的。本节中，我们考虑一种将解显式表示的替代方案。这里我们定义类 *NP* 带解，简称 NP-S。这个类是对著名的 NP 类的扩展，通过语义方式为实例添加了解。NP-S 类是本文后续结果的基础。我们将建立问题的 NP 完全性与其在多项式层次中的变体之间的内在联系。我们将为 NP-S 类中的问题提供两种替代编码。在其中一种编码中，NP-S 中的每个问题也都是 NP 中的问题，由此表明 NP-S 可以视为 NP 的子集。

我们首先回顾 NP 的标准定义，然后说明如何对其进行修改以定义 NP-S。理解 NP 类的一种常见方式，是将其视为由判定问题构成的集合，其中包含是实例和否实例。按照这种观点，复杂性类 NP（非确定性多项式时间）由所有这样的判定问题构成：其解（也称为证书或见证）可以由确定性图灵机在多项式时间内验证。形式上，NP 被定义为一组语言。

▶ **定义 1**（非确定性多项式时间）。 **</style}语言 {0,1}*** 属于 *NP*，当且仅当存在一台确定性图灵机 $V$，其运行时间为多项式时间，并且存在一个多项式 $p(n)$，使得对于每个字符串 $I \in \{0,1\}^*$，

$$I \in L \iff \exists y \in \{0,1\}^* \text{ with } |y| \leq p(|I|) \text{ such that } V(I,y) = 1.$$

在这种表述中，解 $y$ 的编码是一个长度至多为多项式的二进制向量。若有需要，可以将证书的长度限制为恰好 $p(|I|)$；这类细节不会影响复杂度类的定义。此外，每个解的编码 $y$ 都可以由预先指定的确定性多项式时间图灵机 $V$ 验证是否为正确的解，我们也称之为“验证器”。不失一般性，我们只考虑接受长度恰好为 $p(|I|)$ 的字符串 $y$ 的验证器。

我们很快将通过为每个实例关联一个解集合，正式定义复杂度类带解的非确定性多项式时间（NP-S）。首先，NP-S 是 NP 类的细化，其中实例 $I$ 的每个证书都表示为全集 $\mathcal{U}(I)$ 的一个子集，该全集由全集元素构成。不失一般性，我们可以将证书表示为全集的子集，而不是$\{0,1\}^*$中的字符串。对于任意二进制向量$y \in \{0,1\}^{p(|I|)}$，我们可以定义相应的解 $S(y) = \{u_j \in \mathcal{U}(I)\colon y_j = 1\}$。如果我们能使用一台验证 $V$ 的确定性图灵机，在多项式时间内验证 $y$ 是一个证书，那么我们也能识别 $\mathcal{U}(I)$ 的子集，并使用同一台图灵机 $V$（或对 $V$稍作修改的版本），在多项式时间内验证 $S(y)$ 是一个解。这种基于子集的表述有助于技术上的清晰性，也有助于表达问题之间的变换和归约。此外，我们用$\mathcal{S}(I)$表示解的集合。也就是说，

$$\mathcal{S}(I) = \{S(y) \subseteq \mathcal{U}(I) : V(I,y) = 1\}.$$

在每种情况下，问题都是“$\mathcal{S}(I) = \emptyset$ 吗？”

全集由验证器 $V$ 隐式定义。对于每个实例 $I$，我们可以将全集表示为 $\mathcal{U}_V(I)$。同样，每个实例 $I$ 的解集合 $\mathcal{S}(I)$ 也由验证器 $V$ 隐式定义。因此，我们可以将其表示为 $\mathcal{S}_V(I)$。

## 12 源自NP的PH完备性

描述 NP-S 类中的判定问题时，我们会指定全集并描述解集合。以下是两个以 NP-S 问题形式表述的 NP 判定问题。

哈密顿回路**实例** $I$**：** 图 $G = (V,\ E)$**全集**$\mathcal{U}(I)$**：** 边集 $E$**解集**$\mathcal{S}(I)$**：** 边 $S \subseteq E$ 的子集，其中包含 $G$ 中的哈密顿回路。

团

**实例** $I$**：** 图 $G = (V,\ E)$，数量为 $k \in \mathbb{N}$。**全集**$\mathcal{U}(I)$**：** 顶点集 $V$**解集**$\mathcal{S}(I)$**：** 满足 $|S| \geq k$ 的子集 $S \subseteq V$，使得 $\forall i,\ j \in S$ 满足 $i \neq j$，我们有 $\{i,\ j\} \in E$。

如果一个 NP 问题可以用两个不同的全集来表达，那么在 NP-S 中它们被视为不同的问题，尽管它们在 NP 中表示的是同一个问题。例如，可以将团问题表示为全集是边集而非顶点集的形式。与上述团问题相比，这种团问题的版本在 NP-S 中是一个不同的问题。在本文余下部分，为了表述清晰，我们假定对于给定问题的所有实例，全集的语义都是一致的。例如，在考虑团问题的 NP-S 变体时，每个实例的全集要么是顶点子集，要么是边子集。

我们通过为语言 $L \in$NP 关联一个验证器 $V$，来表示将其提升为一个属于 NP-S 的问题 $\Pi$。随后，我们定义 $\Pi = (L, \mathrm{U}_V, \mathrm{S}_V)$，其中 $\mathrm{S}_V$ 是解集，使得对于所有 $S(y) \in \mathrm{S}_V$，$I$、$y$ 满足 $V(I, y) = 1$（否则 $V(I, y) = 0$）。为了抽象掉实际编码中的不必要细节，我们用三元组 (I, U, S) 表示该问题 $\Pi$，其中 $\mathrm{I} = L$ 是问题实例的集合，$\mathrm{U} = \mathrm{U}_V$ 是全集的集合，而 $\mathrm{S} = \mathrm{S}_V$ 是解集的集合。我们注意到，当验证器已知或根据上下文显而易见时，会省略 $\mathrm{U}_V$ 和 $\mathrm{S}_V$ 中的下标 $V$。只有在整个讨论过程中验证器保持固定时，我们才会这样做。此外，如果存在两个不同的验证器 $V$ 和 $V'$，使得 $\mathrm{U}_V = \mathrm{U}_{V'}$ 且 $\mathrm{S}_V = \mathrm{S}_{V'}$，我们将相应的 (I, $\mathrm{U}_V$, $\mathrm{S}_V$) 和 (I, $\mathrm{U}_{V'}$, $\mathrm{S}_{V'}$) 视为同一个问题。

▶ **定义 2** (带解的非确定性多项式时间)。 **</style}问题 (***I, U, S***)**属于 *NP-S*，当且仅当存在一台确定性多项式时间图灵机 $V$，使得

- $I \subseteq \{0,1\}^*$ 是一个语言。我们称 $I$ 为 $\Pi$ 的实例集合。对于每个实例 $I \in I$，都存在某个集合 $U(I)$，称为与实例 $I$ 关联的全集，使得 $|U(I)| = |I|^{O(1)}$，其中 $|I|$ 表示输入字符串的长度。对于每个实例 $I \in I$，都存在某个（可能为空的）集合 $S(I) \subseteq 2^{\mathcal{U}(I)}$，称为与实例 $I$ 关联的解集。对于每个 $C \subseteq U(I)$，$V(I, C) = 1$ 当且仅当 $C \in S(I)$。

尽管解集 $\mathcal{S}(I)$ 是问题表示的一部分，但它并不是 NP-S 问题的输入。相反，验证器 $V$ 对决定 $\mathcal{S}(I)$ 的约束进行编码。NP-S 问题要回答的是：给定实例 $I \in \mathcal{I}$，$\mathcal{S}(I) \neq \emptyset$ 吗?

定义 2 是我们将在整篇论文中使用的定义。我们使用这一定义，将全集和解集从其编码中抽象出来。这本质上使得论证这些问题的性质，并以更易理解的方式处理它们之间的归约变得更加容易。因此，当我们表示 NP-S 中的一个问题时，我们将

描述全集和解集。在这种表示中，NP-S 是 NP 类的扩展。

我们还考虑 NP-S 中问题的另一种表示方式。在这种表示方式中，每个问题实例都表示为一对 $(I, V)$，其中 $V$ 是多项式时间图灵机。由于我们从元组 $(\mathcal{I}, \mathcal{U}_V, \mathcal{S}_V)$ 导出每个问题 $(\mathcal{I}, \mathcal{U}, \mathcal{S})$，其中验证器 $V$ 指定了全集 $\mathcal{U}_V$ 和解集 $\mathcal{S}_V$，因此我们只需将验证器 $V$ 与每个实例关联起来，就能表示每个问题。在这种表示方式下，NP-S 类也是 NP 类的扩展。不过，基于这种表示方式，我们可以利用图灵创造通用图灵机时的一个特点。在这一过程中，图灵展示了如何为每台图灵机关联一个唯一的整数，即标准描述数。将标准描述数嵌入实例后，我们便能证明以下引理。

▶ **引理 3**。*NP-S 中的每个问题 $\Pi$ 都可以转化为 NP 中的一种语言 $L_\Pi$。*

**证明**。我们构造一个语言 $L_\Pi$，该语言包含于 NP，方法是对每个实例 $I \in \mathcal{I}$、相关全集 $\mathcal{U}(I)$以及相关解集 $\mathcal{S}(I)$进行编码。

为此，我们为每个实例 $I$构造一个单词 $w$。首先，我们使用现有编码直接对实例 $I \in \mathcal{I} \subseteq \{0,1\}^*$ 进行编码。其次，我们通过使用验证图灵机 $V$的标准描述数$\langle V\rangle$，隐式地对解集$\mathcal{S}(I) = \mathcal{S}_V(I)$进行编码。根据定义，该验证器 $V$ 只接受对应于解集中全集元素子集的证书。因此，我们也通过验证器 $V$的标准描述数$\langle V\rangle$ 隐式地对全集 $\mathcal{U}(I)$进行编码。对于任意固定的问题 $\Pi$及其对应的验证器 $V$，我们有 $|\langle V\rangle| = O(1)$。现在，我们定义与实例 $I$相关联的单词 $w = I\#\langle V\rangle$，其中 # 是分隔符号，从而得到语言

$$L_\Pi = \{I\#\langle V\rangle \mid I \in \mathcal{I},\ \mathcal{S}(I) \neq \emptyset\}$$

作为 NP-S 问题 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 的编码。

为了证明 $L_\Pi \in$ NP，只需证明存在一个确定性的多项式时间验证图灵机 $V_\Pi$，用于语言 $L_\Pi$，使得

$$w \in L_\Pi \iff \exists y \in \{0,1\}^{|\mathcal{U}(I)|}\ V_\Pi(w, y) = 1.$$

为了识别语言 $L_\Pi$，机器 $V_\Pi$ 按如下方式运行。给定一个候选字串 $w \in \{0,1\}^*$ 和一个候选证书 $y \in \{0,1\}^*$，$V_\Pi$ 首先检查 $w$是否确实具有 $I\#V$ 的形式。由于根据 NP-S 的定义，问题 $\Pi$与固定机器 $V$相关联，因此 $V$ 是一个全局固定的字符串，也就是说，它独立于实例 $I$，且长度为 $O(1)$；因此这项检查是可行的。其次，$V_\Pi$ 使用通用图灵机作为子程序，在给定的实例字串 $I$ 和证书 $y$上模拟 $V$。当且仅当模拟的图灵机满足 $V(I, y) = 1$ 时，$V_\Pi$ 才接受输入。换言之，证书 $y$ 对应于一个属于解集的全集元素子集。对于固定实例 $I$，存在一个能使 $V_\Pi$ 接受的证书 $y$，当且仅当存在一个能使 $V$接受的子集 $S \subseteq U(I)$，当且仅当 $w \in L_\Pi$。由此可知，$V_\Pi$ 是 $L_\Pi$ 的多项式时间验证器。因此 $L_\Pi \in$NP。

请注意，如果我们有两个不同的验证器 $V$ 和 $V'$，它们定义了相同的解集$\mathcal{S}_V = \mathcal{S}_{V'}$，那么我们可以在至多线性的时间内，将具有 $\mathcal{S}_V$ 的 $L_\Pi$ 归约为具有 $\mathcal{S}_{V'}$ 的 $L_\Pi$，方法是用 $V'$ 的标准描述数替换 $V$ 的标准描述数。 ◀

## 14 源自NP的多项式层次完备性

### 不同的宇宙会在 NP-S 中诱导出不同的问题

在我们的框架中，每个问题实例都附带一个全集$\mathcal{U}(I)$，而解是$\mathcal{U}(I)$的子集。正如引言中提到的，如果有两种不同的方式来表达全集，就会产生 NP-S 中两个不同的问题。

例如，如果我们将 NP 中的一个判定问题建模为 0-1整数规划（IP），那么就可以将全集 $\mathcal{U}(I)$ 与该整数规划的决策变量关联起来。也就是说，对于给定的二进制整数变量向量 $x_1, \ldots, x_n$，全集可以表示为 $\{1, \ldots, n\}$。如果同一个问题可以用不同的决策变量集合，以两种不同的方式建模为 0-1整数规划，那么这两种表述就会产生 NP-S 中两个不同的问题。如果判定问题以两种不同的方式建模为布尔公式，情况也类似。这一区别在将 NP-S 中的问题提升为多项式层次或 PSPACE 中的问题时起着关键作用。如果将 NP 中的一个问题扩展到 NP-S 的两种方式具有不同的全集，那么提升后得到的问题也不同，甚至可能具有不同的复杂度。

示例：团。团问题有两种自然的全集选择，下面我们将对此进行说明。我们首先将最大团表述为一个整数规划。它并非基于顶点的表述，而是一种同时包含边变量和顶点变量的表述。我们令 $x_i = 1$ 在顶点 $i$ 属于团时取值为 1。此外，若边 $(i, j)$ 属于团，则令 $x_{ij}$ 取值为 1。目标函数是最大化团中的边数，不过，最大化团中的顶点数也是可以接受的。

最大团

$$
\begin{aligned}
\max \quad & \sum_{(i,\ j)\in E} x_{ij} & \\
\text{s.t.} \quad & x_i + x_j \le 1 & \forall \{i, j\} \notin E \\
& x_{ij} \le x_i & \forall (i, j) \in E \\
& x_{ij} \le x_j & \forall (i, j) \in E \\
& x_{ij} \in \{0,1\} & \forall i, j \in E \\
& x_i \in \{0,1\} & \forall i \in V
\end{aligned}
$$

若以变量 $x_i$ 作为全集，便得到问题团（基于顶点），其中$\mathcal{U} = V$。若使用变量 $x_{ij}$，便得到问题团（基于边），其中$\mathcal{U} = E$。也可以将 $x_i$ 与 $x_{ij}$ 的组合用作全集$\mathcal{U} = V \cup E$。

示例：可满足性。为加以说明，考虑布尔公式的可满足性问题。可满足性问题是理论计算机科学的基础，并在本论文后续各节中发挥重要作用。至少有两种自然的方式来定义全集：

可满足性（基于变量）（简称 Sat-V）。全集由变量组成$\mathcal{U} = \{x_1, \ldots, x_n\}$。一个解由满足赋值中被设为真的变量组成；即，若真值赋值 $\alpha$ 是满足的，则 $S(\alpha) = \{x_i : \alpha(x_i) = 1\}$。

可满足性（基于文字）（简称 Sat-L）。全集由文字构成，即 $\mathcal{U} = \{\ell_1, \bar{\ell}_1, \ldots, \ell_n, \bar{\ell}_n\}$。解由满足赋值对应的文字构成；也就是说，如果真值赋值 $\alpha$ 是满足赋值，那么 $S(\alpha) = \{\ell_i : \alpha(x_i) = 1\} \cup \{\bar{\ell}_i : \alpha(x_i) = 0\}$。

由于它们的全集（以及解集）不同，Sat-V 和 Sat-L 是 NP-S 中不同的问题。

**不同的验证器可能会在 NP-S 中导出不同的问题：划分**

设$\mathcal{I}$ 为 NP 中某种语言的实例。假设 $V$和 $V'$ 是针对这组实例$\mathcal{I}$的两个不同验证器，并且假设存在一个共同全集。也就是说，$\mathcal{U}_V = \mathcal{U}_{V'}$。令$\mathcal{U}$表示这个共同全集。如果$\mathcal{S}_V \neq \mathcal{S}_{V'}$，那么 NP-S 中的 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S}_V)$ 和 $\Pi' = (\mathcal{I}, \mathcal{U}, \mathcal{S}_{V'})$ 是 NP-S 中的不同问题。以下两个划分问题的变体说明了这种区别可能非常重要。

划分问题 1
**实例** $I$**：** 数 $\{a_1, \ldots, a_n\} \subseteq \mathbb{N}$。**全集** $\mathcal{U}(I)$**：** $\{a_1, \ldots, a_n\}$。**解集** $\mathcal{S}_1(I)$**：** 所有子集 $S \subseteq \mathcal{U}(I)$ 的集合，使得 $\sum_{a_i \in S} a_i = \sum_{a_j \notin S} a_j$。

划分问题 2
**实例** $I$**：** 数 $\{a_1, \ldots, a_n\} \subseteq \mathbb{N}$。**全集** $\mathcal{U}(I)$**：** $\{a_1, \ldots, a_n\}$。**解集** $\mathcal{S}_2(I)$**：** 所有子集 $S \subseteq \mathcal{U}(I)$ 的集合，使得 $\sum_{a_i \in S} a_i = \sum_{a_j \notin S} a_j$且 $a_n \in S$。

注意，只要划分问题 1的答案是“是”，划分问题 2的答案也必然是“是”。当将它们视为 NP 中的问题描述时，它们是同一个问题。两者之间唯一的区别在于，$\mathcal{S}_2(I)$ 中的解必须包含元素 $a_n$，而 $\mathcal{S}_1(I)$ 中的解则不要求包含该元素。因此，当将它们视为 NP-S 中的问题描述时，这两个问题是不同的。

这一差别带来了一个重要后果。在下一个第 3.1 小节中，我们将定义 NP-S 完全性的概念。我们将在第7节中证明，划分问题 2 是 NP-S 完全的，而划分问题 1 不是 NP-S 完全的。

## 3.1 NP-S 完全性

作为一个核心问题，我们希望对 NP-S 类中最难的问题进行分类。但我们所说的最难的问题究竟是什么意思?

为此，请回顾一下，NP 完全性是基于多项式时间多对一归约定义的；这些归约可以如下适用于 NP-S 问题。

给定 NP-S 中的两个问题，分别记为 $\Pi = (\mathrm{I}, \mathrm{U}, \mathrm{S})$ 和 $\Pi' = (\mathrm{I}', \mathrm{U}', \mathrm{S}')$。那么，从 $\Pi$ 到 $\Pi'$ 的多项式时间多对一归约是一个函数 $g : \mathrm{I} \to \mathrm{I}'$，可在输入规模为 $|I|$时用多项式时间计算，且满足：$I$ 是 $\Pi$ 的一个是实例，当且仅当 $g(I)$ 是 $\Pi'$的一个是实例（即 $\mathrm{S}(I) \neq \emptyset \iff \mathrm{S}'(g(I)) \neq \emptyset$）。$\Pi$ 的全集 U 和解集 S 通过 $g$ 隐式映射到 $\Pi'$的相应全集 $\mathrm{U}'$ 和解集 $\mathrm{S}'$。

我们的目标是为多层优化问题以及其他从 NP-S 到多项式层次甚至 PSPACE 的映射建立完备性证明。为实现这一目标，我们接下来定义一类特殊的多项式时间多对一归约，并称之为“解嵌入”。解嵌入归约满足以下附加性质，使我们能够将 NP-S 层次上的归约提升为多项式层次或 PSPACE 中提升后问题之间的归约：解嵌入归约将问题 $\Pi$ 归约到问题 $\Pi'$，使得问题 $\Pi$的某个实例的所有解都被转换为问题 $\Pi'$ 的相应实例中的一个部分解；另一方面，实例$\Pi'$的所有解都会被转换回 $\Pi$中相应实例的一个解。我们进一步将一个问题定义为 NP-S 完全，当且仅当该问题属于 NP-S，且从 NP-S 中所有问题到该问题都存在多项式时间解嵌入归约。我们将在后续章节探讨这一概念的含义。

▶ **定义 4** (解嵌入归约)。 设 $\Pi = (I,U,S)$ 和 $\Pi' = (I', U', S')$ 是 NP-S 中的两个问题。我们称存在一个从 $\Pi$ 到 $\Pi'$的解嵌入归约，并记作 $\Pi \leq_{\mathrm{SE}} \Pi'$，如果

- 存在一个函数 $g : I \to I'$，可在关于输入规模 $|I|$ 的多项式时间内计算，使得 $S(I) \neq \emptyset$ 当且仅当 $S'(g(I)) \neq \emptyset$。存在函数 $(f_I)_{I \in \mathcal{I}}$，可在 $|I|$ 的多项式时间内统一计算，使得
- 对于所有实例 $I \in I$，$f_I : U(I) \to U'(g(I))$ 都是从实例 $I$ 的全集到实例 $g(I)$ 的全集的单射，并且满足以下条件：

$$\forall S \in \mathcal{S}(I) : \exists S' \in \mathcal{S}'(g(I)) : f_I(S) = S' \cap \mathcal{U}'_{emb} \tag{P1}$$
$$\forall S' \in \mathcal{S}'(g(I)) : \exists S \in \mathcal{S}(I) : f_I^{-1}(S' \cap \mathcal{U}'_{emb}) = S, \tag{P2}$$

其中 $U'_{emb} = f_I(U(I))$ 和 $U'_{aux} = U'(g(I)) \setminus U'_{emb}$。

解嵌入归约将第一个问题 $\Pi$ 的全集嵌入第二个问题 $\Pi'$ 的全集，同时保持解结构不变。也就是说，在解的层面上，$\Pi$ 中的全集元素 $u \in \mathcal{U}(I)$ 与 $\Pi'$ 中相应的解元素 $f(u)$ 一一对应。按照命名惯例，我们将 $\mathcal{U}'_{emb} = f_I(\mathcal{U}(I))$ 称为嵌入全集，它由问题 $\Pi$ 的实例中已经出现的所有元素构成。此外，不属于问题 $\Pi$ 实例的元素，为了确保 $\mathcal{U}(I)$ 中的元素与 $\mathcal{U}'_{emb} = f_I(\mathcal{U}(I))$ 中的元素之间的一一对应关系能够扩展为实例 $\Pi'$ 的正确完整解而引入，这些元素称为辅助全集，定义为 $\mathcal{U}'_{aux} = \mathcal{U}' \setminus \mathcal{U}'_{emb}$。图 2 示意了两个 NP-S 问题的全集在 SE 归约下的关系。

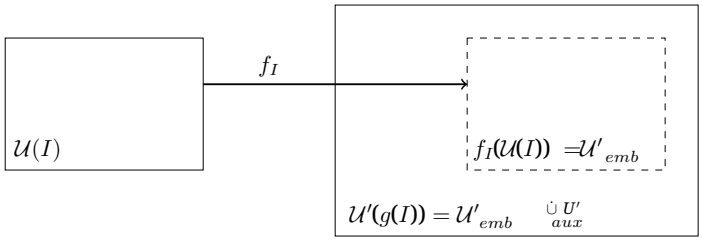


图 2 对给定实例 $I \in$ I，在问题 $\Pi = (\mathrm{I}, \mathrm{U}, \mathrm{S})$ 和 $\Pi' = (\mathrm{I}', \mathrm{U}', \mathrm{S}')$ 之间应用 SE 归约后，宇宙之间的关系。设 $I \in$ I 为 $\Pi$ 的该实例，则 SE 归约 $(g, (f_I)_{I \in \mathcal{I}})$ 将宇宙 $\mathrm{U}(I)$ 嵌入问题 $\Pi'$ 的宇宙 $\mathrm{U}'(g(I))$，使得 $f_I(\mathrm{U}(I)) = \mathrm{U}'_{emb} \subseteq \mathrm{U}'(g(I))$。注意，$g(I)$ 是由通常的归约映射 $g$ 定义的实例。函数 $f_I$ 在 $\mathrm{U}(I)$ 和 $\mathrm{U}'_{emb}$ 的元素之间保持一一对应。

可以用下面这个单一方程等价地表达方程（P1）和方程（P2）：

$$\{f_I(S) : S \in \mathcal{S}(I)\} = \{S' \cap \mathcal{U}'_{emb} : S' \in \mathcal{S}'(g(I))\}. \tag{SE Property}$$

我们将这个方程（SE 性质），或者等价地将 P1 与 P2 合称为解嵌入性质，简称*SE* 性质。

定义 4描述了以 $\Pi$和 $\Pi'$的解为依据的 SE 归约，但也可以用部分解来表述。

设 $I$ 是问题 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S}) \in$ NP-S的一个实例。若 $S$ 满足以下条件，则称其为相对于 $T$的部分解： (1) $S \subseteq T \subseteq \mathcal{U}(I)$，且 (2) 存在一个子集 $S' \subseteq \mathcal{U}(I) \setminus T$，使得$S \cup S' \in \mathcal{S}(I)$。

▶ **引理5.** 设 $\Pi = (I, U, S)$ 和 $\Pi' = (I', U', S')$ 是 *NP-S* 中的两个问题，且$(g, f)$是从 $\Pi$ 到 $\Pi'$ 的解嵌入归约。那么对于每个实例 $I \in I$，以及所有子集 $S \subseteq T \subseteq U(I)$，以下结论成立。子集 $S$ 是相对于 $T$的 $S(I)$ 的部分解，当且仅当 $f(S)$ 是相对于 $f(T)$的 $S'(I')$ 的部分解。

证明。首先假设 $S$是相对于 $T$的 $I$的部分解。选取 $S^* \in \mathrm{U}(I) \setminus T$，使得 $S \cup S^* \in \mathrm{S}(I)$。由于该归约是解嵌入归约，因此存在某个子集 $V \subseteq \mathrm{U}'_{aux}(I)$，使得 $f(S \cup S^*) \cup V \in \mathrm{S}'(I')$。等价地， $f(S) \cup (f(S^*) \cup V) \in \mathrm{S}'(I')$，因此 $f(S)$是相对于 $f(T)$的 $I'$的部分解。

反过来，假设 $f(S)$ 是相对于 $f(T)$ 的 $I'$ 部分解。选取 $V_1 \subseteq f(\mathcal{U}(I)) \setminus f(S)$ 和 $V_2 \subseteq \mathcal{U}'_{aux}(I')$，使得 $f(S) \cup V_1 \cup V_2 \in \mathcal{S}'(I')$。由此可得$f^{-1}(f(S) \cup V_1) \in \mathcal{S}(I)$。等价地， $S \cup f^{-1}(V_1) \in \mathcal{S}(I)$，因此 $S$是相对于$\mathcal{U}(I)$ 的$I$ 部分解。◀

既然我们已经证明 NP-S 是 NP 的子集，自然也可以将完备性的概念从 NP-S 扩展到 NP。

▶**引理6**。 每个 *NP-S* 完全问题都是 *NP*完全问题。

证明。根据引理 3，我们知道，可以对一个 NP-S问题 $\Pi$ 进行编码，使得相应的语言 $L_\Pi$ 属于 NP。接下来需要证明，从 NP-S问题 $\Pi = (\mathrm{I}, \mathrm{U}, \mathrm{S})$ 到 NP-S问题 $\Pi' = (\mathrm{I}', \mathrm{U}', \mathrm{S}')$ 的解嵌入归约，也能导出相应语言 $L_\Pi = \{I \# V \mid I \in \mathrm{I}, \mathrm{S}(I) \neq \emptyset\}$ 和 $L'_\Pi = \{I' \# V' \mid I' \in \mathrm{I}', \mathrm{S}'(I') \neq \emptyset\}$ 之间的归约。设 $(g, (f_I)_{I \in \mathcal{I}})$ 是从 $\Pi$到 $\Pi'$ 的这样一个 SE 归约。现在，可以令 $I' = g(I)$。根据 SE 归约的第一条性质，$\mathrm{S}(I) \neq \emptyset$ 当且仅当 $\mathrm{S}(I') \neq \emptyset$。验证器 $V$ 和 $V'$ 的标准描述数是固定的，根据 NP-S 的定义，这些数都存在。由于 $\Pi$ 是一个给定的 NP-S问题，它们可以在 $O(1)$ 时间内计算出来。

注意，我们并不需要函数族 $f_I$ 来证明NP 完全性。尽管如此，函数 $f_I$ 被隐式地用于从验证器 $V$ 到验证器 $V'$的映射中。更确切地说，被 $V$ 接受的证书 $y$ 是 $V'$的部分证书。也就是说，每个被 $V$ 接受的证书 $y \in \{0,1\}^{|\mathcal{U}(I)|}$ 都可以通过一个字 $z \in \{0,1\}^{|\mathcal{U}'(g(I))| - |\mathrm{U}(I)|}$ 扩展为被 $V'$接受的证书 $y' = yz \in \{0,1\}^{|\mathcal{U}'(g(I))|}$。◀

作为NP-S类的另一项良好性质，我们可以通过复用库克-莱文定理的证明来表明，相应的可满足性版本是NP-S 完全的。他们的构造使我们能够识别出一个布尔变量集合，这些变量对应于所考虑问题的相应实例的证书。因此，我们能够证明，所有NP-S 问题确实都可以通过解嵌入归约归约到可满足性（带有文字全集或变量全集）。

▶ **定理 7** (针对 NP-S 调整的库克-莱文定理)。 可满足性 (文字全集为 *or* 变量全集) 关于多项式时间 *SE* 归约是 *NP-S* 完全的。

**证明**。 我们考察 Cook 的原始证明，并说明它实际上是一个多项式时间的 SE 归约。令 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 为 NP-S 中的任意问题，其全集为$\mathcal{U}(I)$，且对于每个实例 $I \in \mathcal{I}$， $\Pi$都有与之关联的解集 $I$) 。我们需要证明$\Pi \leq_{\mathrm{SE}}$ 可满足性。下面，我们证明 SAT-V 的完备性——证明

## 18 源自NP的PH完备性

Sat-L 的完备性证明通过相应地调整全集即可类似地得到。回忆一下，Sat-V $=(\mathcal{I}', \mathcal{U}', \mathcal{S}')$，其中 $\mathcal{I}'$ 是 Sat实例的集合；对于每个公式 $\varphi \in \mathcal{I}'$，集合 $\mathcal{U}'(\varphi)$ 是其变量集，而 $\mathcal{S}(\varphi)$ 是这样一个变量子集的集合：将这些变量设为真后，它们对应于满足赋值。

由于 $\Pi$ 属于 NP-S，根据定义 2，存在一个确定性多项式时间图灵机 $V$，称为验证器，使得对于每个 $X \subseteq \mathcal{U}(I)$，$V(I, X) = 1$ 当且仅当 $X \in \mathcal{S}(I)$。更具体地说，我们可以假设该图灵机 $V$ 接收作为输入的二进制变量 $n$ $x_1$，……，$x_n$，它们编码集合 $X \subseteq \mathcal{U}(I)$，以及二进制变量 $n$ $i_1$，……，$i_n$，它们描述实例 $i$。（为简单起见，不失一般性地假设实例和全集描述的大小相同。）

现在，Cook 的证明表明，我们可以引入多项式数量的辅助变量 $z_1$，……，$z_{p(n)}$，并在多项式时间内计算出一个布尔表达式 $\varphi(x, i, z)$，使得 $\varphi(x, i, z)$ 恰好刻画图灵机 $V$ 的行为。设 $u_j$ 为全集 $\mathcal{U}(I)$ 中的第 $j$ 个元素。核心思路是，解嵌入函数定义为 $f(u_j) := x_j$ 对所有 $j \in \{1, \ldots, n\}$，也就是说，我们将全集 $\mathcal{U}(I)$ 中的第 $j$ 个元素映射到 Cook 构造中表示其非确定性 Sat 证书的变量。无需重述 Cook 构造的所有细节，只需指出以下关键细节即可。

- 对于给定的实例 $I$ 和某个集合 $X \subseteq \mathcal{U}(I)$，令 $i_I$ 表示 $I$ 的编码，$x_X$ 表示 $X$ 的编码。令 $z$-变量为自由变量。则部分公式‘$\varphi(x_X, i_I, z)$’可满足，当且仅当 $V$ 接受 $(I, X)$。部分公式‘$\varphi(x, i_I, z)$’的大小关于 $|I|$ 为多项式级，且可在多项式时间内由 $I$ 构造。

我们现在声称，Cook 的这一归约可立即导出一个多项式时间 SE 归约 $(g, (f_I)_{I \in \mathcal{I}})$。形式上，令 $g(I) := \varphi(x, i_I, z)$，其中 $i_I$ 是 $i_I$ 的编码，而 $x, z$ 是自由变量。注意，根据 Cook 归约的性质，$I$ 是 $\Pi$ 的是实例，当且仅当 $\exists x, z\ \varphi(x, i_I, z)$ 可满足。因此，这是一个正确的多对一归约。对于 SE 性质，我们定义 $f_I(u_j) := x_j$，其中 $u_j$ 是表示 $\mathrm{U}(I)$ 的第 $j$ 个全集元素的变量，$j \in \{1, \ldots, n\}$，而 $x_j$ 是表示它的二进制变量。由上述等价关系可知，这是一个 SE 归约：若 $S \in \mathrm{S}(I)$，则 $V$ 在 $|I|^{O(1)}$ 步后接受 $(I, S)$，并且对于相应的赋值 $f_I(S)$，它可以扩展为 $\varphi$ 的一个满足赋值。另一方面，我们有 $\mathrm{U}'_{emb} = \{x_1, \ldots, x_n\}$，并且每个满足赋值 $S' \in \mathrm{S}(\varphi)$ 限制在正文字集合 $\mathrm{U}'_{emb}$ 上后，都会编码一个集合 $S = f_I^{-1}(S' \cap \mathrm{U}'_{emb})$，使得 $S \in \mathrm{S}(I)$。这证明了 SE 性质，因而也证明了 $\Pi \leq_{\mathrm{SE}}$ 可满足性。◀

作为最后一个重要性质，我们还可以证明 SE 归约具有传递性。因此，我们处于一个便利的情形：只需考虑手头的两个问题，就能扩展归约链，而不必一路追溯到可满足性问题。

▶ **引理 8.** *SE 归约具有传递性，即对于 NP-S 问题 $\Pi_1$、$\Pi_2$、$\Pi_3$，若满足 $\Pi_1 \leq_{\mathrm{SE}} \Pi_2$ 和 $\Pi_2 \leq_{\mathrm{SE}} \Pi_3$，则有 $\Pi_1 \leq_{\mathrm{SE}} \Pi_3$。*

证明。考虑 $i = 1$、2、3 对应的三个 NP-S 问题 $\Pi_i = (\mathrm{I}^i, \mathrm{U}^i, \mathrm{S}^i)$。存在一个从 $\Pi^1$ 到 $\Pi^2$ 的 SE 归约 $(g_1, f_1)$，以及一个从 $\Pi^2$ 到 $\Pi^3$ 的 SE 归约 $(g_2, f_2)$。我们描述一个从 $\Pi^1$ 到 $\Pi^3$ 的 SE 归约。为此，我们需要一个元组 $(g, (f_I)_{I \in \mathcal{I}})$。

对于第一个函数 $g$，我们令 $g := g_2 \circ g_1$。由于 $\mathcal{S}^1(I) \neq \emptyset \Leftrightarrow \mathcal{S}^2(g_1(I)) \neq \emptyset \Leftrightarrow \mathcal{S}^3((g_2 \circ g_1)(I)) \neq \emptyset$，且 $g$ 是多项式时间可计算的，这已经足够了。

令 $I^1 := I$ 为 $\Pi^1$的初始实例，令 $I^2 := g_1(I^1)$为 $\Pi^2$ 的实例，并令$I^3 := g_2(I^2)$为 $\Pi^3$ 的实例。对于第二个函数 $f_I$，我们对每个实例$I \in \mathcal{I}_1$ 定义映射 $f_I := (f_2)_{I^2} \circ (f_1)_I$。注意，对于每个实例 $I \in \mathcal{I}^1,$，函数 $f_I$是单射，并映射到 $\mathcal{U}^3_{emb}$，且是多项式时间可计算的。剩下需要证明的是，$f$ 具有所需的 SE 性质。由于 $\Pi^1 \leq_{\text{SE}} \Pi^2$ 和 $\Pi^2 \leq_{\text{SE}} \Pi^3$，并且由于（对于单射函数） $f(A \cap B) = f(A) \cap f(B)$，我们有

$$\begin{aligned}\{f(S^1) : S^1 \in \mathcal{S}^1\} &= \{f_2(f_1(S^1)) : S^1 \in \mathcal{S}^1\} \\ &= \{f_2(S^2 \cap \mathcal{U}^2_{emb}) : S^2 \in \mathcal{S}^2\} \\ &= \{f_2(S^2) \cap \mathcal{U}^3_{emb} : S^2 \in \mathcal{S}^2\} \\ &= \{S^3 \cap \mathcal{U}^3_{emb} : S^3 \in \mathcal{S}^3\}.\end{aligned}$$

◀

现在我们回到划分问题的不同版本。在解嵌入归约下，问题划分问题 1（$a_n \in S$ 或 $a_n \notin S$）不是 NP-S 完全的，而划分问题 2（$a_n \in S$）是 NP-S 完全的。原因可概括如下：存在一些可满足性实例，其解集并非解对称；也就是说，存在一个解 $S \in \mathcal{S}$，使得$\mathcal{U} \setminus S \notin \mathcal{S}$。任何从可满足性出发的 SE 归约都会得到一个非解对称的实例。然而，划分问题 1 是解对称的。如果 $S \in \mathcal{S}$，那么$\mathcal{U} \setminus S \in \mathcal{S}$。因此，不存在从可满足性到划分问题 1 的 SE 归约，划分问题 1 也就不可能是 NP-S 完全的。

关于同一问题采用不同全集和解集的进一步讨论，请参阅第7节，其中将更详细地讨论各种推论。

### 弱 NP-S 完全性与强 NP-S 完全性

强弱 NP-S 完全性的性质直接继承自强弱 NP 完全性的性质。如果一个 NP-S 完全问题 $\Pi$ 嵌入 NP 后是弱 NP 完全的，则称其为弱*NP-S* 完全。例如，背包问题的 NP-S 版本是弱 NP-S 完全的。同样，如果一个 NP-S 完全问题$\Pi$ 嵌入 NP 后是强 NP 完全的，则称其为强*NP-S* 完全。例如，旅行商问题的 NP-S 版本是强 NP-S 完全的。

## 4 多项式时间层次

定义多项式时间层次（多项式层次）有几种方法。定义多项式层次最常见的方法之一是使用交替量词。

▶ **定义9**（多项式时间层次（多项式层次））。 如果存在一个多项式 $p$ 和一个多项式时间可计算谓词 $R$，使得语言 $L$ 属于 $\Sigma^p_k$：

$$x \in L \iff \exists y_1 \forall y_2 \exists y_3 \cdots Q_k y_k \; R(x, y_1, \ldots, y_k),$$

其中每个 $y_i$ 的取值范围为 $\{0,1\}^{p(|x|)}$，且当 $k$ 为奇数时， $Q_k$ 是存在量词；当 $k$ 为偶数时， $Q_k$ 是全称量词。类 $\Pi^p_k$ 的定义类似，只是量词顺序从全称量词开始：

$$x \in L \iff \forall y_1 \exists y_2 \forall y_3 \cdots Q_k y_k \; R(x, y_1, \ldots, y_k).$$

多项式层次由所有 $k \in \mathbb{N}$ 对应的 $\Sigma_k^p$ 和 $\Pi_k^p$ 的并集构成，即

$$PH = \bigcup_{k \in \mathbb{N}} (\Sigma_k^p \cup \Pi_k^p).$$

这一定义突出了多项式层次基于逻辑的本质： $\Sigma_1^p$ 对应存在性陈述（NP）， $\Sigma_2^p$ 对应存在—全称陈述，依此类推。

与 NP 类和 coNP 类一样，我们可以为多项式层次的每一层定义困难性和完备性的概念。如果对于多项式层次的某一层中的每个问题，都可以通过多项式时间多对一归约将其归约到问题 $L$，则称问题 $L$ 是 $\Sigma_k^p$ 困难的。此外，如果 $L$ 属于 $\Sigma_k^p$，则称 $L$ 为 $\Sigma_k^p$ 完全问题。类似地，如果多项式层次某一层中的每个问题都能归约到一个问题，则称该问题是 $\Pi_k^p$ 困难的；如果它还属于 $\Pi_k^p$，则称其为 $\Pi_k^p$ 完全问题。多项式层次更高层级的完全性结果意味着很强的难处理性。特别是，如果一个 $\Sigma_k^p$ -完全问题可以在多项式时间内求解，那么整个多项式层次都会坍缩到 $k$ 层。因此，PH 内的完备性是表明其计算困难性超越 NP 的有力证据。

关于多项式层次及其类 $\Sigma_k^p$ 的介绍，可参阅 Papadimitriou 的著作 [Pap94], 、Si pser 的著作 [Sip97] ，或 Jeroslow 的文章 [Jer85]。关于双层优化背景下这一主题的专门介绍，可参阅 Woeginger 的文章 [Woe21]。

## 4.1 多项式层次的博弈论解释

多项式层次的交替量词定义自然地适用于博弈论解释。我们可以将属于语言 $\Sigma_k^p$ 或 $\Pi_k^p$ 视为双人博弈中存在获胜策略：博弈双方分别是一个 $\exists$-玩家 （下文称为 Alice）和一个 $\forall$-玩家（下文称为 Bob）。在 $\Sigma_k^p$ 情形下，Alice 先行动，然后与 Bob 交替行动，直至完成 $k$ 次行动。

我们首先对标准交替量词框架作几处调整。这种修改后的方法可以归约到标准方法。对于每个实例 $I$，我们假设存在一个全集$\mathcal{U}(I)$，使得 $|\mathcal{U}(I)| = p(|I|)$，其中某个多项式为 $p$；并且第 $i$回合由玩家选择$\mathcal{U}(I)$的一个子集 $S_i \subseteq$。其次，存在多项式时间可计算谓词 $R_1$、 $R_2$、 ……、 $R_k$。第 i 回合 $S_i$ 若满足允许 条件，则称为允许回合；若$R_i(I$、 $S_1$、 $\ldots S_i) = 1$。否则，该回合称为禁止回合。如果每个回合都允许，且 $R(I$、 $S_1$、 ……、 $S_k) = 1$，则 Alice 赢得博弈；否则，Bob 赢得博弈。

我们注意到， $k$回合博弈可以用通常的交替量词形式表示。可以用单个谓词 $R'$ 替换 $R$ 和 $R_1$， $\ldots R_k$，其中，若 Bob 是第一个做出违规行动的玩家，或者所有行动都允许且$R(I$， $S_1$， ……， $S_k) = 1$，则 $R'(I$， $S_1$， ……， $S_k) = 1$。因此，输入 $x$ 属于语言 $L$ 当且仅当 Alice 有一种无论 Bob 如何行动都能确保胜利的策略；也就是说，Alice 可以确保 $R'(x$， $I$， $S_1$， ……， $S_k) = 1$。

在 $\Pi_k^p$ 问题中，量词同样交替出现，但以全称量词开头。这意味着 Bob 先行动，玩家的角色也随之互换：当 i 为奇数时，Bob 选择 $S_i \subseteq \mathcal{U}(I)$；当 i 为偶数时，Alice 选择 $S_i \subseteq \mathcal{U}(I)$。和之前一样，谓词$R'(I$， $S_1$， ……， $S_k)$ 决定胜者。若 $R'(I$， $S_1$， ……， $S_k) = 1$，则 Alice 获胜；否则 Bob 获胜。输入 $I$ 属于语言 $L$，当且仅当 Bob 没有能迫使 Alice 落败的策略。等价地，若 Alice 有一种应对策略，能够确保对于 Bob 可能做出的每一种 选择，$R(I$， $S_1$， ……， $S_k) = 1$，则 $I \in L$。

这种双人视角有助于理解多项式层次中问题的复杂性：量词每增加一次交替，就对应于对立双方之间博弈中多增加一回合，从而增加策略深度和计算难度。

## 4.2 k回合对抗选择博弈

在本节中，我们将考察一类非常自然的博弈，称之为对抗性选择博弈。类 NP-S 中的每个博弈都可以转化为任意 $k$ 的 $k$ 回合对抗性选择博弈。在这种对抗性选择博弈中，Alice 和 Bob 竞相选择元素，使其能够或不能构成一个解。下面给出一个简单的例子，其中基础博弈是哈密顿回路问题。

$k$回合哈密顿回路对抗选择博弈**实例：** 图 $G = (V, E)$、一个数 $k$、$E$ 的一个划分 $E_1 \cup \cdots \cup E_k$。**博弈：** Alice 与 Bob 轮流交替行动，为 $i = 1, \ldots, k$ 选择某个集合 $S_i \subseteq E_i$。进行最后一回合的玩家获得胜利，当且仅当 $\bigcup_{i=1}^{k} S_i$ 是哈密顿回路。

我们可以将此示例推广到 NP-S 中的所有问题。设 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 是 NP-S 中的一个问题。则 $\Pi$ 上的 $k$ 回合对抗选择博弈定义如下。

实例由 $(I, \mathcal{P})$ 定义，其中 $I$ 是 $\Pi$ 的一个实例，且 $\mathcal{P} = \{U_1, \ldots, U_k\}$ 是全集 $\mathcal{U}(I)$ 的一个划分。两名玩家 Alice 与 Bob 轮流对弈，由 Alice 先开始。对所有 $1 \leq j \leq k$：若 $j$ 为奇数，则 Alice 选择一个子集 $S_j \subseteq U_j$；若 $j$ 为偶数，则 Bob 选择一个子集 $S_j \subseteq U_j$。在全部行动 $1 \leq j \leq k$ 之后，我们得到一个集合 $S = \bigcup_{j=1}^{k} S_i$。进行最后一回合的玩家在 $S \in \mathcal{S}(I)$ 时获得该博弈的胜利。否则，另一名玩家获得该博弈的胜利。问题在于先行动的 Alice 是否获得该博弈的胜利。我们将根据 $k$ 为奇数且 Alice 最后行动，或 $k$ 为偶数且 Bob 最后行动这两种情形加以区分。若 Alice 最后行动，则当她能够得到一个解时她获得胜利。若 Bob 最后行动，则当 Bob 无法得到一个解时 Alice 获得胜利。

这一描述给出了如下定义。

▶ **定义 10** ($k$回合对抗性选择博弈)。 给定一个问题 $\Pi = (\mathcal{I},\mathcal{U},\mathcal{S})$属于 *NP-S* 。我们定义在 $k$ 上的 回合对抗性选择博弈如下：输入是一个实例 $I \in \mathcal{I}$ 以及全集 $\mathcal{U}(I) = U_1 \cup \ldots \cup U_k$的一个划分。**1.** 如果 $k$ 为奇数，那么问题是判断是否

$$\exists S_1 \subseteq U_1 \; \forall S_2 \subseteq U_2 \; \cdots \; \forall S_k \subseteq U_k : \bigcup_{i=1}^{k} S_i \in \mathcal{S}(I).$$

**2.** 如果 $k$ 为偶数，那么问题是判断是否

$$\exists S_1 \subseteq U_1 \; \forall S_2 \subseteq U_2 \; \cdots \; \exists S_k \subseteq U_k : \bigcup_{i=1}^{k} S_i \notin \mathcal{S}(I).$$

我们观察到如下结论。

▶ **引理 11**。 对于所有 $\Pi \in$ *NP-S*，问题 $k$-回合对抗性选择博弈 on $\Pi$属于 $\Sigma_k^p$ 。

**证明**。 博弈论构造给出了一个属于 $\Sigma_k^p$的问题：实例 $(I, \mathcal{P}_U)$ 是实例，当且仅当 Alice 有一种能够确保胜利的策略。如果 $k$ 为奇数，那么

$$(I, \mathcal{P}_U) \in L \iff \exists S_1 \subseteq U_1 \; \forall S_2 \subseteq U_2 \; \cdots \; \forall S_k \subseteq U_k : \bigcup_{i=1}^{k} S_i \in \mathcal{S}(I).$$

如果 $k$ 是偶数，则

$$(I, \mathcal{P}_U) \in L \iff \exists S_1 \subseteq U_1 \; \forall S_2 \subseteq U_2 \; \cdots \; \exists S_k \subseteq U_k : \bigcup_{i=1}^{k} S_i \notin \mathcal{S}(I).$$

## 22 源自NP的多项式层次完备性

由于 $\Pi \in$ NP-S，可以在多项式时间内检查 $\bigcup_{i=1}^{k} S_i \stackrel{?}{\in} \mathcal{S}(I)$是否成立，并且所有 $U_i$ 都是$\mathcal{U}(I)$的子集，因此它们可以用多项式大小进行编码。◄

$\Pi$ = SAT-V 的情形已得到充分研究。Sat-V 上的 k 步对抗选择博弈有时被称为 $k$-量化布尔公式（$QBF_k$）或$k$-量化可满足性（$QSAT_k$）。Stockmeyer 和 Wrathall 将其称为 $B_k$。

Stockmeyer [Sto76] 和 Wrathall [Wra76] 证明了以下完备性结果。

►**定理 12**。 $k$ 步对抗选择博弈 on Sat-V 是 $\Sigma_k^p$ 完全的。如果 $k$ 允许随 $n$ 增长，则 Sat-V 上的对抗选择博弈是*PSPACE* 完全的。

接下来，我们将上述定理推广到每一个 NP-S 完全问题。

►**定理 13**。 令 $\Pi$ 表示任意一个 *NP-S* 完全问题。则 $k$ 步对抗选择博弈 on $\Pi$是 $\Sigma_k^p$ 完全的。如果 $k$ 允许随 $n$增长，则 在 $\Pi$ 上的对抗选择博弈是 *PSPACE* 完全的。

**证明**。 我们证明 $k$ 为奇数时的定理。基本上，同样的论证也能证明 $k$ 为偶数时的定理。

设 $I$ 是一个 SAT-V 实例，其全集为 $U$ ，解集为$\mathcal{S}$。令 $(I,\mathcal{P})$ 为 SAT-V 上的 $k$ 回合对抗选择博弈的一个实例，其中$\mathcal{P} = \{U_1, ..., U_k\}$ 是全集 $U$的一个划分。

由于 $\Pi$ 是 NP-S 完全的，因此存在一个从 Sat-V 到 $\Pi$的解嵌入归约 $(g, f)$。令 $g(I) = I'$。因此， $I'$ 是 $\Pi$的一个实例。令 $U'$ 和$\mathcal{S}'$ 分别表示其全集和解集。回顾一下， $f : \mathcal{U} \to \mathcal{U}'$ 是一个单射，将 Sat-V 的全集嵌入 $\Pi$的全集。这里，像 $f(U) =: \mathcal{U}'_{emb}$是嵌入全集，而$\mathcal{U}'_{aux} = U' \setminus f(U)$是辅助全集。注意， $U' = \mathcal{U}'_{emb} \cup \mathcal{U}'_{aux}$。

然后，我们在 $\Pi$ 上构造一个 $k$-回合对抗选择博弈的实例（$I'$，*P*类′），其中*P*类′ $= \{U'_1, ....., U'_k\}$定义如下：(i) 对于 $i \in [k-1]$， $U'_i = f(U_i)$；(ii) $U'_k = f(U_k) \cup \mathcal{U}_{aux}$。

假设 Alice 和 Bob 在 SAT 上进行对抗性选择博弈的实例 $I$，而 Adam 和 Barbara 在 $\Pi$ 上进行对抗性选择博弈的实例 $I'$。为了证明该定理，我们需要确立： $I$ 是 Alice 的胜利，当且仅当 $I'$ 是 Adam 的胜利。

为了确立这一充要条件，我们采用一种新的证明技术，将其称为对偶模仿。

我们首先证明，如果 Alice 能保证胜利，那么 Adam 也能保证胜利。我们设想 Alice 和 Barbara 分别是各自博弈的专家，并且她们都采用完美策略。我们还设想 Adam 在模仿 Alice 的行动，而 Bob （帮助 Adam 获胜）在模仿 Barbara 的行动。

对于满足 $i < k$ 的每个奇数值 $i$，如果 Alice 选择 $S_i$，那么 Adam 就选择 $S'_i = f(S_i)$。Adam 行动后，Barbara 选择一个子集 $S'_{i+1}$。然后 Bob 选择 $S_{i+1} = f^{-1}(S'_{i+1})$。

回想一下， $S_{k-1} = \bigcup_{i \in [k-1]} S_i$。我们已经假设 Alice 能在博弈的第 $k$ 回合获胜。因此， $S_{k-1}$ 是关于 $U_{k-1}$的 $\mathcal{S}$ 的部分解。在另一个博弈中，$S'_{k-1} = f(S_{k-1})$。由于从 $\Pi$ 到 $\Pi'$ 的归约是解嵌入，根据引理5，$f(S_{k-1})$是关于 $f(U_{k-1})$的 $\mathcal{S}'$ 的部分解。因此，Adam 能在自己的第 $k$ 回合获胜。

由于我们假设 Adam 的对手采用完美策略， $I'$ 对 Adam 来说必定是必胜的。

反过来，假设 Bob 能保证在实例 $I'$ 中获胜。为了证明 Barbara 能保证获胜，我们交换专家和模仿者的身份。

现在，Adam 和 Bob 是专家，并采用完美策略。我们设想 Barbara 在模仿 Bob 的行动，而 Alice（她正在帮助 Barbara 获胜）在模仿 Adam 的行动。

对于 $i$ 的每个奇数值，只要 $i < k$，如果 Adam 选择 $S'_i$，那么 Alice 就选择 $S_i = f^{-1}(S_i)$。Alice 行动后，Bob 选择一个子集 $S_{i+1} \subseteq U_{i+1}$。然后 Barbara 选择$S'_{i+1} = f(S_{i+1})$。

我们假设 Alice 无法在第 $k$ 回合获胜。因此，$S_{k-1}$ 并非相对于 $U_{k-1}$ 的部分解。根据引理5，$f(S_{k-1})$ 并非相对于 $f(U_{k-1})$ 的部分解。因此，Adam 无法在最后一回合获胜。由于假设 Adam 采用了完美策略，$I'$ 必定是 Barbara 的必胜策略。◀

**关于对偶模拟证明。**

我们将使用对偶模拟证明来证明如下形式的命题：Adam 在他的博弈中有必胜策略，当且仅当 Alice 在她的博弈中有必胜策略。

为证明“若”方向，我们假设 Alice 在与 Bob 的博弈中有必胜策略，且 Alice 和 Barbara 都采取最优策略。假设 Adam 模仿 Alice 的行动，而 Bob 模仿 Barbara 的行动。接下来我们证明 Adam 赢得了他的博弈。由于 Barbara 采取了最优策略，Adam 只有在拥有必胜策略时才能获胜。

为证明“仅当”方向，我们假设 Bob 在与 Alice 的博弈中有必胜策略，且 Bob 和 Adam 都采取最优策略。假设 Alice 模仿 Adam 的行动，而 Barbara 模仿 Bob 的行动。接下来我们证明 Barbara 赢得了她的博弈。由于 Adam 采取了最优策略，Barbara 只有在拥有必胜策略时才能获胜。

## 5 阻断问题

本节将通过把我们的 NP-S 框架应用于最小成本阻断问题（也称为最关键顶点／最关键边问题或阻断问题），展示该框架的强大能力。我们还将考察这些问题的多阶段变体。这是一大类自然且经过充分研究的问题，其关注点是寻找一个规模较小的元素集合，使其同时与给定名义问题的所有解相交。以下是一个自然的阻断问题示例：

团阻断问题
**实例：** 图 $G = (V, E)$，顶点集 $D \subseteq V$，预算 $\Gamma \in \mathbb{N}$，数量 $k \in \mathbb{N}$。**问题：** 是否存在一个大小为 $|B| \leq \Gamma$ 的集合 $B \subseteq D$，使得图 $G$ 中的所有团\ $B$ 大小都不超过 $k-1$?

问题“团阻断问题”可以看作阻断者（Adam）和保护者（Barbara）之间的两回合博弈。保护者希望找到一个较大的团（大小至少为 $k$），但阻断者的目标恰恰相反。他可以删除至多 $\Gamma$ 个顶点 $B \subseteq D$，以阻挠保护者的目标，也就是摧毁所有大小至少为 $k$ 的团。我们称集合 $D$ 为易受攻击元素集合。

如引言中所述，过去人们针对许多名义问题的各种变体研究了阻断问题，这些问题分属运筹学、理论计算机科学和纯图论等互不相干的领域。这些已研究的问题有什么共同特征？几乎在所有情况下，目标都是找出少量元素，使其能够干扰每一个解（例如，找出少量边，使其与每一个

哈密顿回路相交）。因此，以下对阻断问题的抽象定义显得极其自然。

▶ **定义14** (阻断问题)。 给定一个*NP-S*问题 $\Pi = (I, U, S)$ 。与 $\Pi$ 相关的阻断问题记为 *Interdiction*-$\Pi$ ，定义如下：输入是一个实例 $I \in \mathcal{I}$，以及一个集合 $D \subseteq U(I)$ 和一个预算 $\Gamma \in \mathbb{Z}$。问题是判断

$$\exists B \subseteq D \ with \ |B| \leq \Gamma : \forall S \in \mathcal{S}(I) : B \cap S \neq \emptyset.$$

本节的主要结果是

▶**定理15**。 *Interdiction*-$\Pi$ 是 $\Sigma_2^p$ -完全的，适用于所有 *NP-S* 完全问题 $\Pi$。

定理15是定理23的一个特例，我们将在本节稍后证明该定理。在文献中，阻断有时会结合阻断成本进行研究，即为每个元素指定不同的阻断成本。这个问题也称为最小成本阻断问题。

▶ **定义16** (最小成本阻断)。 给定一个*NP-S*问题 $\Pi = (I, U, S)$ 。与 $\Pi$ 相关的最小成本阻断问题定义如下：输入是一个实例 $I \in \mathcal{I}$，以及一个成本函数 $c : U(I) \to \mathbb{Z}$ 和一个阈值 $t \in \mathbb{Z}$。问题是判断

$$\exists B \subseteq \mathcal{U}(I) \ with \ c(B) \leq t : \forall S \in \mathcal{S}(I) : B \cap S \neq \emptyset.$$

定义16将定义14包含为一个特例：只需对每个元素设置 $c(e) \in \{1, |\mathcal{U}(I)| + 1\}$ 即可。因此，以下结论是定理15的一个推论。对于每个 NP-S 完备问题 $\Pi$，与 $\Pi$ 相关的最小成本阻断问题是$\Sigma_2^p$-完备的。

## 5.1 技术备注

下面我们给出一系列技术备注，以澄清定义14。从整体上看，这些备注可以跳过。

首先需要指出，在我们的问题 Interdiction-$\Pi$ 中，底层实例不会改变。具体而言，我们不会删除全集中的元素（例如图的顶点）。对于某些问题，删除元素与禁止元素出现在解中可能存在微妙差别。顶点覆盖问题就是一个例子：删除顶点 $v$ 及其关联边，与禁止 $v$ 出现在解中、但仍要求覆盖所有关联于 $v$ 的边，两者有很大区别。第一种情况下，可以证明顶点覆盖阻断问题属于 coNP，因此我们无法期望得到一般性的 $\Sigma_2^p$ -完备性结果。本文只考虑第二种情况。

作为第二点说明，我们注意到，文献中最小成本阻断问题与最关键节点/边问题通常略有不同：最小成本阻断问题要求找到一个成本最低的阻断者，使目标值降低指定的幅度。另一方面，最关键节点问题则是在给定阻断者一定成本预算的情况下，求目标函数能够降低的最大幅度。我们在此将阻断问题表述为判定问题，因此能够刻画这两种变体的 $\Sigma_2^p$ 完备性。由标准论证可知，在我们的设定中，上述两个问题都成为 $\Sigma_2^p$ 完全问题。

最后要指出的是，Interdiction-$\Pi$可以理解为 Alice（$\exists$玩家，试图找到阻断者）与 Bob（$\forall$玩家，试图找到一个解）之间的博弈。

## 5.2 多阶段保护-干预博弈

我们还可以进一步推广主要结果：不仅证明阻断问题的 $\Sigma_2^p$ -完备性，还证明一般 $\Sigma_k^p$ -完备性以及 $k$ 回合多阶段阻断博弈的 PSPACE 完全性。这些博弈的灵感来自对初始阻断场景的扩展：对于 $k=3$，现在保护者可以先保护 至多 $\Gamma_P$，之后阻断者才能阻断至多 $\Gamma_B$ 个元素。阻断者不得选择任何已受保护的元素。文献中将此类问题称为保护-阻断问题 。对于一般的 $k \in \mathbb{Z}_{\geq 2}$，我们可以扩展这一思路，从而得到一套具有以下广义规则的博弈：

- 我们假设保护者总共拥有 $\Gamma_P$ 保护令牌，阻断者总共拥有 $\Gamma_B$ 干预令牌。在总共 $k$ 个回合中，保护者和阻断者交替将各自的令牌放置在集合 $\mathcal{U}(I)$ 中。如果 $k$ 为奇数，则由保护者先行动；如果 $k$ 为偶数，则由阻断者先行动。输入包含一系列嵌套集合 $C_1 \subseteq C_2 \subseteq \cdots \subseteq C_{k-1} \subseteq \mathcal{U}(I)$。在第 $i$ 个回合中，玩家只能在 $C_i$ 内放置令牌。换句话说，玩家一开始不能访问整个集合 $\mathcal{U}(I)$，而只能访问一个随每个回合逐渐扩大的区域。令牌不能放置在已被干预令牌或保护令牌占据的任何元素上。每个回合中，保护者可以随意放置任意数量的 $\Gamma_P$ 令牌，但游戏结束时，令牌总数不得超过 $\Gamma_P$。阻断者和 $\Gamma_B$ 也适用同样的规则。在最后一个回合中，保护者不得再放置令牌，而是要选择一个与所有干预令牌互不相交的子集 $S \subseteq \mathcal{U}(I)$。如果 $S \in \mathcal{S}(I)$，则保护者获胜；否则，阻断者获胜。注意， $S$ 可以包含保护令牌，但并非必须包含。

对于 $k=2$，我们重新讨论上一节中的问题 Interdiction-Π 。更严谨地说，我们可以将博弈规则表述如下。

对于 NP-S 中给定的问题 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$，定义两种不同的 $k$-回合阻断博弈：一种中 $k$ 为偶数，另一种中 $k$ 为奇数。为简化定义，我们用 $A_i$ 表示保护者和阻断者各自前几个集合的并集。更准确地说，设 $A_1$，……， $A_n$ 为一组集合，我们用 $\widehat{A}_i = \bigcup_{j=1}^{i} A_j$. 表示。按照惯例，如果 $A_j$ 未进一步指定，则 $A_j = \emptyset$ 。特别地，在下文中，术语 $P_k$ 表示在第 $k$ 回合中放置的保护令牌集合，而术语 $P_k$表示前 $k$ 回合后放置的所有保护令牌的总集合。类似地，我们定义阻断令牌的 $B_k$ 和 $\widehat{B}_k$ 。对于每个 $k$，要么是 $P_k = \emptyset$ ，要么是 $B_k = \emptyset$。

与本文全文一致，我们从 Alice 的视角表述每个判定问题，询问 Alice 是否拥有获胜策略。若行动次数为偶数，则当且仅当保护者（Bob）未能在最后一回合获得可行解时，Alice 获胜。

▶ **定义 17**（组合 $k$-回合封锁博弈，在 $\Pi$）。 给定一个属于 NP-S 的问题 $\Pi =$（$I$、$U$、$S$）。我们定义如下的在 $\Pi$ 上的组合 $k$-回合封锁博弈：输入是一个实例 $I \in \mathcal{I}$，$U$ 的一个嵌套子集族 $C_1$、……、$C_{k-1}$（$I$），即由可保护元素和可阻断元素组成的 $C_i \subseteq C_{i+1}$、$1 \leq i \leq k-2$ 和 $C_{k-1} \subseteq U$（$I$），保护者的预算 $\Gamma_P \in \mathbb{Z}$，以及阻断者（阻断者）的预算 $\Gamma_B \in \mathbb{Z}$ 。我们区分 $k$ 为奇数和偶数的情形。

1. 保护-干预（$k$ 为奇数）。问题是是否

$$\exists P_1 \subseteq C_1, \forall B_2 \subseteq (C_2 \setminus \widehat{P}_1), \exists P_3 \subseteq (C_3 \setminus \widehat{B}_2), \ldots, \forall B_{k-1} \subseteq (C_{k-1} \setminus \widehat{P}_{k-2}) :$$
$$\exists S \in \mathcal{S}(I) : \textit{If } |\widehat{B}_{k-1}| \leq \Gamma_B, \textit{ then } |\widehat{P}_{k-2}| \leq \Gamma_P \textit{ and } \widehat{B}_{k-1} \cap S = \emptyset,$$
$$\textit{i.e., Bob can obtain a feasible solution.}$$

2. 干预-保护（$k$ 为偶数）。问题是：

$$\exists B_1 \subseteq C_1, \forall P_2 \subseteq (C_2 \setminus \widehat{B}_1), \exists B_3 \subseteq (C_3 \setminus \widehat{P}_2), \ldots, \forall B_{k-1} \subseteq (C_{k-1} \setminus \widehat{P}_{k-2}) :$$
$$\forall S \in \mathcal{S}(I) : \textit{If } |\widehat{P}_{k-2}| \leq \Gamma_P, \textit{then } |\widehat{B}_{k-1}| \leq \Gamma_B \textit{ and } \widehat{B}_{k-1} \cap S \neq \emptyset,$$
$$\textit{i.e., Bob cannot obtain a feasible solution.}$$

请注意，行动次数为奇数的博弈与行动次数为偶数的博弈有所不同，因为问题始终是从先行动的玩家的视角提出的。在这两种情况下，保护者都是最后行动。行动次数为奇数时，先手玩家是保护者，因此若他能在最后一回合选出一个解 $S \subseteq \mathcal{S}$，就能获胜。行动次数为偶数时，先手玩家是阻断者，因此若他能阻止保护者在最后一回合选出一个解 $S \subseteq \mathcal{S}$，就能获胜。

如果考虑 $\Pi$ =Sat，那么该公式在奇数 $k$ 时对应一个合取范式公式，在偶数 $k$ 时对应一个析取范式公式（通过应用否定）。事实上，这也对应于 ∃量化 QBF$_k$-Sat 的自然定义：如果 $k$ 为奇数，那么 QBF$_k$-CNF-Sat 是规范的 $\Sigma_k^p$ -完备问题。相反，如果 $k$ 为偶数，那么 QBF$_k$-DNF-Sat 是规范的 $\Sigma_k^p$ -完备问题。

最后，我们考虑保护-干预博弈的最后一种变体。在如上所述的博弈中，双方玩家都有预算 $\Gamma_B$，$\Gamma_P$ 可在博弈过程中的任意时刻使用，但总额不得超过。我们称之为全局预算。相比之下，我们也可以修改博弈规则，使得 $\Gamma_B$，$\Gamma_P$ 成为博弈每一回合的局部约束，即在每一回合中，保护者（相应地，阻断者）最多可放置 $\Gamma_P$ 个保护令牌（相应地，$\Gamma_B$ 个干预令牌）。博弈的所有其他规则保持不变。我们将此变体称为局部预算。

## 5.3 保护-干预博弈的复杂性

本小节我们首先证明其包含关系。

▶**引理 18.** 设 $\Pi$ 属于*NP-S*。若 $k \in \mathbb{N}$ 为常数，则组合 $k$回合封锁博弈（关于 $\Pi$ ）包含于 $\Sigma_{2k-1}^p$。若 $k$ 是输入的一部分，则其包含于*PSPACE*。

**证明。** 设 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 属于 NP-S，且 $I \in \mathcal{I}$ 是其中一个实例。定义 17 通过交替量词直接 $\widehat{\ }$ $\widehat{\ }$描述了该问题。此外，由于集合 $S_i$、$P_i$、$P_i$、$B_i$ 和 $B_i$ 都是 $\mathcal{U}(I)$ 的子集，因此均可用多项式空间编码。最后，由于 $\Pi$ 属于 NP-S，条件$\widehat{B} \cap S ? = \emptyset$ 可在多项式时间内检验。◀

本节的主要结果如下一个定理。

▶ **定理 19. </style}设 是一个 NP-S 完全问题。如果 是常数，那么 组合**$k$-回合封锁博弈（在 $\Pi$ 上）是 $\Sigma_k^p$ - 完全的。如果 $k$ 属于输入的一部分，那么该问题是 *PSPACE* 完全的。（对于全局预算变体和局部预算变体，这一结论均成立。）

我们首先证明，若 $\Pi = \text{Sat-V}$，则组合 $k$-回合封锁博弈（在 $\Pi$ 上）是 $\Sigma_k^p$ -完全的。为此，我们将从 Sat-V 上的 $k$-回合对抗选择博弈进行归约。我们将在 $k$ 为奇数的情形下给出归约和证明。证明之后，我们会说明在 $k$ 为偶数时如何修改归约和证明。在证明了 $\Pi = \text{Sat-V}$ 的 $\Sigma_k^p$ -完备性之后，我们将证明，若 $\Pi$ 是任意 NP-S 完全问题，则组合 $k$-回合封锁博弈（在 $\Pi$ 上）是 $\Sigma_k^p$ -完全的。

从 Sat-V 上的 $k$ 回合对抗选择博弈到保护-干预博弈的归约相当复杂。为了说明这一归约，我们将变换分为两部分。第一部分中，我们把 Sat-V 上 $k$ 回合对抗选择博弈的任意实例（$I$、$U_1$、$U_2$、.....、$U_k$）归约为一个受限干预博弈，其中保护者 Adam 和阻断者（干预者）Barbara 必须遵守四条附加规则。我们将证明，对于这个受限干预博弈的实例，Adam 必胜，当且仅当在原始对抗选择博弈的实例中 Alice 必胜。在证明的第二部分，我们将说明如何放宽这四条附加规则。也就是说，我们将把受限干预博弈的实例归约为（原始）干预博弈的一个实例。我们将证明，对于这个干预博弈的实例，Adam 必胜，当且仅当他在受限干预博弈中获胜。最后，我们证明保护-干预博弈对所有NP-S 完全问题都是完备的。

## 5.4 保护-干预 Sat-V 的复杂性

在本小节中，我们将确立组合 $k$ 回合阻断 Sat-V 的复杂性。需要指出的是，证明相当复杂。读者可以跳过本小节，只了解其主要结果。

▶**定理 20.** 组合 $k$回合阻断博弈在 Sat-V 上对于所有常数 $k$ 都是 $\Sigma_k^p$ 完全的。如果 $k$ 是输入的一部分，则该问题是 *PSPACE* 完全的。

### Sat-V 上的受限干预博弈

本节主要讨论采用全局预算的变体，因为它的困难性证明更加复杂。采用局部预算的变体则可轻松推得。我们先介绍 Sat-V 上的受限干预博弈。我们的困难性归约从 Sat-V 上的 $\Sigma_k^p$ 完全 $k$ 回合对抗选择博弈开始，该博弈定义于第 4.2 节。下面针对 $k$ 为奇数的情况给出变换。

设 $I_{\text{sel}} :=$ (*I*，*P*类) 是 Sat-V 上 $k$-回合对抗选择博弈的一个实例，其中 $I$ 是相应的 Sat 实例，且 *P*类 $= \{U_1, \ldots\ldots, U_k\}$ 是全集$\mathcal{U}$(*I*) 的一个划分。不失一般性，我们可以假设，对于从 $i = 1$ 到 $k$ 的所有情况，$|U_i| = n$。（这可以通过向较小的 $U_i$ 添加虚拟元素来实现。）令 $U_i = \{u_{i1}, \ldots\ldots, u_{in}\}$。回顾对抗选择博弈的规则：Alice 与 Bob 对弈，由 Alice 先行。对于 $i \equiv 1$ (模 2)，Alice 选择 $S_i \subseteq U_i$。对于 $i \equiv 0$ (模 2)，Bob 选择 $S_i \subseteq U_i$。若 $\bigcup_{i=1}^k S_i \in \mathcal{S}$，则 Alice 获胜；否则，Bob 获胜。给定实例 $I_{\text{sel}}$，我们定义 Sat-V 上受限干预博弈的一个新实例 $I_{\text{restr}}$，如下所述。

- 对于每个元素 $u_{ij}$，我们创建另一个布尔变量 $w_{ij}$，称为 $u_{ij}$的伪补集。对于从 $i = 1$ 到 $k$的每个情况，我们令 $W_i := \{w_{ij} : j \in [n]\}$。$k$回合封锁博弈的嵌套集合为 $C_1, \ldots,$
- $C_k$，其中对于每个 $i$，$C_i = \frown \quad \frown U_i \cup W_i$。$I_{\text{restr}}$ 的子句与 $I_{\text{sel}}$中的子句相同。请注意，没有任何子句包含来自 $W = \bigcup_{i=1}^k W_i$的变量。
-

- 预算为 $\Gamma_P = n(k-1)/2$ 和 $\Gamma_B = n(k-1)$。

现在，我们对 Adam 和 Barbara 在这个受限封锁博弈中的行为施加额外约束。我们制定了四条规则，旨在确保玩家使用对偶模仿时，以下对应关系成立。对于从 $i = 1$ 到 $k-1$ 的情况以及从 $j = 1$ 到 $n$的情况，在对抗选择博弈中，Alice 或 Bob 在 $u_{ij}$ 上放置标记，当且仅当 Barbara 没有在封锁博弈的 $u_{ij}$ 上放置标记。我们将利用这种对应关系证明：Alice 能在对抗选择博弈中获胜，当且仅当 Adam 能在封锁博弈中获胜。

- **规则1**。 在第 $i$ 回合（其中 $i$ 为偶数）中，阻断者必须对每个 $j \in \{1, ..., n\}$，恰好阻断 $u_{ij}$ 或 $w_{ij}$ 中的一个。**规则2**。 在第 $i$ 回合（其中 $i$ 为奇数且 $i < k$）中，保护者必须对每个 $j \in \{1,..., n\}$，恰好保护 $u_{ij}$ 或 $w_{ij}$ 中的一个。**规则3**。 到第 $k-1$ 回合结束时，阻断者必须阻断集合 $\bigcup_{i \text{ 奇数},\ i \neq k} U_i \cup W_i$ 中所有未受保护的元素。**规则4**。 在第 $k$ 回合选择元素时，保护者必须为每个 $i < k$ 选取 $U_i \cup W_i$ 中所有未被阻断的元素。

尽管我们将上述内容规定为应当遵守的规则，但我们也考虑违反规则的可能性。如果某条规则被违反，则第一个违反规则的先手玩家输。如果没有违反任何规则，那么若 Adam 能够选出满足规则4的元素集合 $S'$，使得 $S' \cap U$ 是原始 Sat 公式的一个解，即 $S' \cap U \in \mathcal{S}(I)$，则 Adam 获胜。否则，Barbara 获胜。我们将带有上述附加规则的 $k$ 回合封锁博弈称为 $k$ 回合受限干预博弈。

▶**引理 21.** *Adam* 对实例 $I_{restr}$ 上的 $k$回合受限干预博弈（Sat-V）拥有获胜策略，当且仅当 *Alice* 对实例$I_{sel}$ 上的 $k$回合对抗选择博弈（Sat-V）拥有获胜策略。

**证明**。 我们不失一般性地假设 Adam 与 Barbara 满足全部四条规则。

设 $M_P = \{i :\ i$ 为奇数且 $1 \leq i \leq k-2\}$ 为 Adam 保护元素的那些迭代。设 $M_B = \{i :\ i$ 为偶数且 $1 \leq i \leq k-1\}$ 为 Barbara 阻断元素的那些迭代。首先考虑 $I_{\text{sel}}$ 为 Alice 的胜利这一情形。我们现在证明，即使 Barbara 采取最优策略，$I_{\text{restr}}$ 仍是 Adam 的胜利。我们的证明使用对偶模仿策略论证。假设 Alice 采取最优策略，因而保证获得胜利。Adam 与 Bob 在第 1 到 $k-1$ 次行动上的对偶模仿策略如下。

- **Adam：** 对于每个 $i \in M_P$和 $j \in \{1, ..., n\}$，如果 Alice 在第 $i$回合选择了 $u_{ij}$，那么 Adam 就保护 $u_{ij}$。如果 Alice 没有选择 $u_{ij}$，那么 Adam 就保护 $w_{ij}$。（该规则保证 Adam 会满足规则2，并且恰好保护 $u_{ij}$ 和 $w_{ij}$中的一个。）**Bob：** 对于每个 $i \in M_B$和 $j \in \{1, ..., n\}$，如果 Barbara 阻止了 $u_{ij}$，那么 Bob 就不选择 $u_{ij}$。如果 Barbara 阻止了 $w_{ij}$，那么 Bob 就选择 $u_{ij}$。（根据规则1，Barbara 恰好阻止 $u_{ij}$ 和 $w_{ij}$中的一个。）

令 $S_i \subseteq U_i$ 表示在对抗选择博弈的第 $i$ 回合中由 Alice 或 Bob 选出的元素子集。对于 $i \in M_P$，令 $S'_i \subseteq (U_i \cup W_i)$表示由 Adam 保护的元素。根据规则3，对于 $i \in M_P$，Barbara 阻止 $T'_i = (U_i \cup W_i) \setminus S'_i$。对于 $i \in M_B$，令$T'_i \subseteq (U_i \cup W_i)$表示由 Barbara 阻止的元素。

现在我们断言，对于每个 $i = 1$到 $k-1$, $S_i = U_i \setminus T_i$。如果这个断言为真，那么 Adam 就能满足规则4，并在第 $k$ 回合选择一个子集 $S^*$，其中包含所有未阻塞元素

$\bigcup_{i=1}^{k-1}(U_i \cup W_i)$（从而满足规则4），并且满足 $S^* \cap U = \bigcup_{i=1}^{k} S_i \in \mathcal{S}(I)$（从而赢得博弈）。要证明该断言对偶数 $i$ 为真，注意 Bob 选择了$S_i = U_i \setminus T_i$。要证明该断言对奇数 $i$ 为真，注意 Adam 选择了 $S'_i$，使得 $S'_i \cap U = S_i$。根据规则3，Barbara 阻塞了 $T'_i = S'_i \setminus (U_i \cup W_i)$。因此， $S_i = U_i \setminus T_i$。由于该断言为真，Adam 必胜。

现在考虑这样的情形： $I$ 对 Alice 来说是失败，对 Bob 来说是胜利。我们将证明，即使 Adam 采用最优策略，$I'$ 对 Adam 来说也是失败，对 Barbara 来说则是胜利。我们假设 Adam 和 Bob 在各自的博弈中采用最优策略。Alice 和 Barbara 使用以下模仿策略。

- **Alice：** 如果 $i \in M_P$ 且 Adam 选择 $u_{ij}$，那么 Alice 就选择 $u_{ij}$。如果 Adam 选择 $w_{ij}$，那么 Alice 就不选择 $u_{ij}$。**Barbara：** 如果 $i \in M_B$ 且 Bob 选择 $u_{ij}$，那么
- Barbara 就阻止 $w_{ij}$。如果 Bob 不选择 $u_{ij}$，那么 Barbara 就阻止 $u_{ij}$。此外，Barbara 还会阻止 Adam 在上一回合选择的每个元素的伪补。

和之前一样，我们用 $S_i$ 表示 Alice 或 Bob 选取的 $U_i$ 中的元素。对于 $i \in M_P$，我们用 $S'_i$ 表示 $U_i \cup W_i$ 中受到 Adam 保护的元素。对于 $i \in$ （$M_P \cup M_B$），我们用 $T'_i$ 表示 $U_i \cup W_i$ 中被 Barbara 阻止的元素。

现在我们声称，对于每个 $i = 1$ to $k-1$, $S_i = U_i \setminus T'_i$。如果该断言为真，那么 Adam 就无法赢得博弈。原因如下：在博弈的最后一回合，根据规则4，对于 $i \in \{1, ..., k-1\}$，Adam 必须选择 $(U_i \cup W_i) \setminus T'_i$，因此 Adam 选择的 $\bigcup_{i=1}^{k-1} U_i$ 元素为 $\bigcup_{i=1}^{k-1} S_i$。由于不存在满足 $S = \bigcup_{i=1}^{k} S_i \in$ S($I$) 的 $S_k \subseteq U_k$，Adam 无法赢得博弈。要证明该断言对 $i \in M_B$ 成立，请注意 Barbara 的模仿策略确保了 $U_i \setminus T'_i = S_i$。要证明该断言对 $i \in M_P$ 成立，请注意 Alice 的模仿策略确保了 $S_i = S'_i \cap U_i$，而规则3确保了$S'_i \cap U_i = U_i \setminus T'_i$。和之前一样，对于 $i \in M_P$，我们假设 Alice 选择 $S_i \subseteq U_i$，Adam 保护 $S'_i \subseteq U_i \cup W_i$。对于 $i \in M_B$，我们假设 Bob 选择 $S_i \subseteq U_i$，Barbara 阻塞 $T'_i \subseteq U_i \cup W_i$。对于 $i \in M_B$，令 $S'_i$ 表示未阻塞元素。由于 Bob 有一个获胜游戏，不存在满足 $S = \bigcup_{i=1}^{k} S_i \in$ S($I$) 的 $S_k \subseteq U_k$。

由于 Alice 的模仿策略，对于 $i \in M_P$, $S_i = S'_i \cap U_i$。由于 Barbara 的模仿策略，对于每个 $i \in M_B$, $S'_i \cap U_i = S_i$。◀

现在考虑 $k$ 为偶数的情况。Alice 和 Adam 分别先在各自的博弈中行动，而 Bob 和 Barbara 最后行动。这意味着 Adam 是阻断者，Barbara 是保护者。这会导致问题实例和证明中出现一些小的变化。

**1.** $\Gamma_P = n(k-2)/2$； $\Gamma_B = n(k-1)$。**2.** 如果 Bob 无法从未阻塞元素中选出一个子集 $S$，使得 $S \in \mathcal{S}(I)$，那么 Adam 获胜。**3.** 当迭代次数 $i$ 为奇数时，阻断者行动；当 $i$为偶数时，保护者行动。

和之前一样，我们将证明：当且仅当 $I_{\text{sel}}$ 对 Alice 来说是胜局时， $I_{\text{restr}}$ 对 Adam 来说是胜局。证明考虑到，当 $k$ 为偶数时，我们需要证明 Alice 获胜时，Adam（阻断者）获胜。这个证明与证明当 $k$ 为奇数时，Bob 获胜时 Barbara（阻断者）获胜的证明相似。除这几处改动外，证明基本相同。

### 为规则引入构件

证明了受限版本的困难性后，我们可以将结果扩展到非受限版本，证明其困难性。

▶**定理 22.** 在 *Sat-V* 上的组合 $k$ 回合阻断博弈对所有常数 $k$ 都是 $\Sigma_k^p$ -完全的；当 $k$ 是输入的一部分时，该博弈是 *PSPACE* 完全的。

**证明**。 我们证明存在一个从 $k$-回合 Sat-V 对抗选择博弈 到 $k$-回合 Sat-V 阻断博弈的归约，其中受限阻断博弈的所有附加约束都可以通过归约中的特定构件来建模。因此，结合引理 21，即可证明该定理。

我们再次先假设 $k$ 为奇数，并在最后讨论 $k$ 为偶数的情形。给定 Sat-V 上的 $k$-回合对抗选择博弈的一个实例 $I_{\text{sel}} := (I, \{U_1, \ldots\ldots, U_k\})$，我们用它定义受限阻断博弈的一个实例 $I_{\text{restr}}$，方法与上文相同。随后，我们进一步将实例 $I_{\text{restr}}$ 转换为非限制阻断博弈的一个实例 $I_{\text{int}}$。核心思路是将实例 $I_{\text{restr}}$ 的初始合取范式可满足性公式 $\varphi$ 转换为实例 $I_{\text{int}}$ 的一个新合取范式可满足性公式 $\psi$，使得每当 Adam 和 Barbara 在 $\psi$ 上进行非限制博弈时，在任何最优策略中，Adam 和 Barbara 都会遵守规则 1–4。因此，对于这一选择的 $\psi$，受限博弈与非限制博弈是等价的。

我们首先给出 $\psi$ 的形式化定义，然后说明其不同组成部分背后的直观含义。给定变量 $u_{ij}$ 上的初始公式 $\varphi$，其中 $i \in [k]$，$j \in [n]$，我们引入新变量 $w_{ij}$，用于表示受限博弈中 $u_{ij}$ 的伪补，其中 $i \in [k]$，$j \in [n]$。此外，我们还引入额外的辅助变量 $s_{ij}$，适用于所有偶数的 $i \in [k]$，$i$，$j \in [n]$；适用于所有奇数的变量 $t_{ij}$，其中 $i \in [k-1]$、$i$ 为奇数，$j \in [n]$；以及两个最后的辅助变量 $s$，$t$。

这里，我们以两种不同的方式使用记号 $U$、$W$、$u_{ij}$ 和 $w_{ij}$。当它们用于诸如 $\varphi(U)$ 这样的布尔表达式时，$U$（以及隐含的 $u_{ij}$ 和 $w_{ij}$）被视为布尔向量和变量。当它们被视为布尔变量时，文字 $u_{ij}$ 和 $w_{ij}$ 都是定义良好的。写作 $S \in \mathcal{S}$ 时，我们按如下方式解释：若 $u_{ij} \in S$，则令 $u_{ij} = true$；否则令 $u_{ij} = false$。于是，当每个子句都满足时，$\varphi(U) = true$。公式 $\psi$ 定义如下：

**公式 $\psi$:**

$$(\varphi(U) \vee s \vee t) \tag{$\ell_1$}$$

$$\wedge \left( \bigwedge_{i \in M_P} \bigwedge_{j=1}^{n} (u_{ij} \vee w_{ij} \vee s) \right) \wedge \left( \bigwedge_{i \in M_B} \bigwedge_{j=1}^{n} (u_{ij} \vee w_{ij} \vee s \vee t) \right) \tag{$\ell_2$}$$

$$\wedge \left( \bigwedge_{i \in M_B} \bigwedge_{j=1}^{n} (s_{ij} \to (u_{ij} \wedge w_{ij})) \right) \wedge \left( \bigwedge_{i \in M_P} \bigwedge_{j=1}^{n} (t_{ij} \to (u_{ij} \wedge w_{ij})) \right) \tag{$\ell_3$}$$

$$\wedge (\, \overline{t} \vee \bigvee_{i \in M_P} \bigvee_{j=1}^{n} t_{ij}) \wedge (\overline{s} \vee \bigvee_{i \in M_B} \bigvee_{j=1}^{n} s_{ij}) \tag{$\ell_4$}$$

$$\wedge \bigwedge_{i=1}^{k} \bigwedge_{j=1}^{n} ((\overline{u}_{ij} \leftrightarrow w_{ij}) \vee s \vee t). \tag{$\ell_5$}$$

◀

无限制封锁博弈的嵌套集，以及保护者和阻断者的预算，与受限干预博弈中完全相同，即，

$\frown$ $\frown C_i = U_i \cup W_i$ 对于 $i \in [k-1]$，以及 $\Gamma_P = n(k-1)/2$ 和 $\Gamma_B = n(k-1)$。我们澄清，在四个文字 $u_{ij}, \overline{u}_{ij}, w_{ij}, \overline{w}_{ij}$中，只有两个正文字 $u_{ij}, w_{ij}$ 包含在 $C_i$ 中，对于$i \leq k-1$。因此，对于所有 $i \in [k-1]$，在博弈的第 $i$回合，保护者和阻断者被允许将其令牌放置在正文字集合 $C_i$上。仅在最后一回合，即第 $k$回合，保护者被允许自由选择 $\psi$的一个赋值，该赋值表示为所有未被封锁标记占据的文字（包括对应于辅助变量的文字）的一个子集。我们现在解释该归约为何正确。回顾受限博弈中的规则1–4，为方便起见我们在此重复。

- **规则1**。 在第 $i \in M_B$ 回合中，阻断者（Barbara）必须针对每个 $j \in$，恰好阻断 $u_{ij}$ 或 $w_{ij}$ 中的一个{1、.....、$n$}。**规则2**。 在第 $i \in M_P$ 回合中，保护者（Adam）必须针对每个 $j \in$，恰好保护 $u_{ij}$ 或 $w_{ij}$ 中的一个{1、.....、$n$}。**规则3**。 在第 $k-1$ 回合结束时，阻断者必须阻断集合 $\bigcup_{i \in M_P} (U_i \cup W_i)$ 中每个未受保护的元素。**规则4**。 在第 $k$ 回合选择元素时，保护者必须选择 $\bigcup_{i=1}^{k-1} (U_i \cup W_i)$ 中每个未被阻断的元素。

核心思想是引入所谓的 作弊检测变量 $s$，$t$。封锁实例中的变量和子句经过设计，使得第一个作弊者（即违反四条规则之一的人）将在封锁博弈中失败。作弊检测机制的工作原理大致如下（稍后我们会对此进行更详细的解释）。如果 Barbara（阻断者）首先作弊，并且她在第 $i \in M_B$ 回合通过违反规则1作弊，那么 Adam 可以在第 $i+1$ 回合保护一些元素，从而确保自己能够在第 $k$ 回合让 $s = 1$。而如果 $s = 1$，那么 Adam 就获胜。如果 Adam 先通过在第 $i \in M_P$ 回合违反规则2作弊，那么 Barbara 可以在第 $i+1$ 回合阻断一些变量，从而阻止 Adam 赢得博弈。如果 Barbara 先在第 $k-1$ 回合结束时违反规则3作弊，那么 Adam 可以在第 $k$ 回合设置 $t = 1$，并确保获胜。最后，如果规则1、规则2和规则3在第1至 $k-1$ 回合中均得到满足，那么任何满足布尔表达式都会满足规则4。因此，要有获胜的可能，Adam 必须满足规则4。

为了实现上述行为，我们需要两个不同的作弊检测变量 $s$、$t$，其中 $s$ 负责偶数 $i$，而 $t$ 负责奇数 $i$。在这种解释下，$\psi$ 的定义可以解释如下。第（$\ell_1$）行指出，Adam 需要满足 $\varphi(U)$，除非他发现 Barbara 作弊。第（$\ell_2$）行指出，Adam 需要从每对伪补中至少选择一个文字，除非他已发现作弊；对于奇数 $i$ 和偶数 $i$，作弊检测能力略有不同。第（$\ell_3$）行指出，只有当一对伪补变量 $u_{ij}$、$w_{ij}$ 都为真时，才可以将变量 $s_{ij}$ 或 $t_{ij}$ 设为真。第（$\ell_4$）行指出，只有当至少一个变量 $s_{ij}$ 为真时，才能将作弊检测变量 $s$ 设为真；这里针对的是偶数 $i$。同样，只有当至少一个变量 $t_{ij}$ 为真时，才能将 $t$ 设为真；这里针对的是奇数 $i \neq k$。最后，第（$\ell_5$）行指出，除非 Adam 发现作弊，否则他的变量赋值 $w_{ij}$ 需要遵循伪补赋值。

现在我们来论证，在 Adam 和 Barbara 采取最优策略时，规则 1–4 确实会得到遵守。我们根据首先 被违反的规则进行情况分类。

**情况 1：** 规则*1*是第一个被违反的规则。 假设对于某个偶数 $i$，第 1、.....、$i-1$ 回合中的所有规则都得到遵守，但在第 $i$ 回合中，对于某个 $j \in [n]$，Barbara 没有恰好阻止 $u_{ij}$ 和 $w_{ij}$ 中的一个。这意味着她要么两个都阻止，要么两个都不阻止。注意，由于 Adam 到目前为止都按规则行动，此时他已经花费了总共 $\Gamma_P = n(k-1)/2$ 保护预算中的 $n \cdot i/2$。

## 32 源自NP的PH完备性

**情况 1a：** 假设 Barbara 在第 $i$回合中既没有阻止 $u_{ij}$，也没有阻止 $w_{ij}$。那么在情况 $i+1<k$中，Adam 会在紧接着的下一回合 $i+1$中保护 $u_{ij}$和 $w_{ij}$。注意，这是可行的，因为 Adam 还剩下两个单位的预算。在情况 $i+1=k$中，Adam 只需在第 $i+1=k$回合中选择 $u_{ij}$和 $w_{ij}$。在所有情况下，在最后一回合 $k$中，Adam 都可以将作弊检测变量 $s$ 设为真（并将 $t$ 设为假）。由于变量 $s$ 出现在除 ($\ell_4$) 之外的每个子句中，Adam 可以轻松赢得实例 $I_{\text{prot}}$。

**情形 1b：** 假设 Barbara 在第 $i'$ 回合同时封锁了 $u_{i'、j'}$ 和 $w_{i'、j'}$。那么 Adam 采用以下策略：从第 $i'+1$ 回合开始，Adam 仍确保遵循规则1，即对所有 $i\in M_P$ 和所有 $j\in[n]$，都恰好保护 $u_{ij}$ 和 $w_{ij}$ 中的一个。（Adam 可以任意选择保护 $u_{ij}$ 还是 $w_{ij}$。）这是可行的，因为 Adam 还剩下足够的预算。到第 $k-1$ 回合结束时，Alice 至多封锁了 $\Gamma_B=n(k-1)$ 个元素，其中包括配对 $u_{i'、j'}$ 和 $w_{i'、j'}$ 中的两个元素。根据鸽巢原理，在 $S$ 中存在一对未被封锁的伪补：$=\{u_i、_j、w_i、_j：i\in[k-1]、j\in[n]\}$。

这意味着 Adam 可以将作弊检测变量 $s$ 或 $t$ 设为真。于是，Adam 赢得博弈，因为对于 $i\in M_P$，第 ($\ell_2$) 行为真，这是因为 Adam 始终遵循规则1。所有其他子句也都为真，因为 $s$ 或 $t$ 为真。

**情况 2：** 规则*2*是第一个被违反的规则。假设对于某个 $i\in M_P$，第 1、.....、$i-1$ 回合中所有规则均得到遵守，但在第 $i$ 回合中，对于某个 $j\in[n]$，Adam 没有恰好保护 $u_{ij}$ 和 $w_{ij}$ 中的一个。这意味着 Adam 要么两个都保护，要么两个都不保护。由于 Barbara 迄今为止一直遵守规则，注意到她至多花费了预算 $\Gamma_B$ 的 $n(i-1)$。

于是，从第 $i+1$ 回合开始，Barbara 可以采用以下策略：首先假设 Adam 既没有保护 $u_{ij}$，也没有保护 $w_{ij}$。那么在紧接着的第 $i+1$ 回合中，Barbara 可以同时阻止 $u_{ij}$ 和 $w_{ij}$。之后的每个回合中，Barbara 都遵守规则1。（最终，这会导致 Barbara 违反规则3，但她仍然会获胜。）

由于 Barbara 每次都遵守规则1，Adam 将无法将 $s$ 设为真，而 ($\ell_2$) 中的子句 $u_{ij}\vee w_{ij}\vee s$ 也不会得到满足。因此 Adam 输掉这场博弈。最后，我们考虑 Adam 同时保护了 $u_{ij}$ 和 $w_{ij}$ 的情况。鉴于 Adam 的预算为 $\Gamma_P=|M_P|n$，根据鸽笼原理，Adam 会留下某一对未受保护的 $u_{i'、j'}$ 和 $w_{i'、j'}$，分别对应 $i'\in M_P$ 和 $j'\in[n]$。Barbara 可以在每个回合遵守规则1，并在第 $i'+1$ 回合同时阻止这一对 $u_{i'、j'}$ 和 $w_{i'、j'}$。和之前一样，Adam 会输。

**情形 3：** 规则*3*是最先被打破的规则。需注意，规则3的违反只能在回合 $k-1$ 结束时被识别。根据假设，规则 1 和规则 2 在每个回合中均得到满足。由于 Adam 一直维持规则2，因此对所有 $i\in M_P$ 以及所有 $j\in[n]$，$u_{ij}$，$w_{ij}$ 中至少有一个始终受到保护。假设对于 $i\in M_P$，Barbara 没有阻断 $u_{ij}$，$w_{ij}$ 中的至少一个。那么，在回合 $k$ 中，Adam 可以同时选择 $u_{ij}$，$w_{ij}$，从而将作弊检测变量 $t$设为真。于是，Adam 可以平凡地满足除针对 $i\in M_P$ 的子句 ($\ell_2$) 之外的所有子句。但这些子句也可以由 Adam 满足，因为对于所有奇数 $i$、$i\neq k$，每一对 $u_{ij}$，$w_{ij}$ 中至少有一个始终受到保护。总而言之，在这种情况下 Adam 可以平凡地取得胜利。

**情形 4：** 规则*4*是最先被打破的规则。如果规则 1–3 得到维持，则 Adam 必须将作弊检测变量 $s$ 和 $t$ 设为假。

由于已遵循规则 1–3，且 $\Gamma_B=n(k-1)$，对于每个 $i\in[k-1]$ 和$j\in[n]$，$u_{ij},w_{ij}$ 中恰有一个被阻止。对于每个 $i\in[k-1]$ 和 $j\in[n]$，为满足子句 ($\ell_5$)，Adam 必须选择{$u_{ij}$, $w_{ij}$}中未被阻止的元素。因此，Adam 必须遵循规则4，才能满足可满足性公式。

我们终于得出以下结论：Adam 和 Barbara 在非限制博弈 $I_{\text{int}}$ 中的最优策略如下：

- 对于每个 $i\in M_P$，在第 $i$ 回合中，Adam 对每个 $j\in[n]$ 恰好保护 $u_{ij}$，$w_{ij}$ 中的一个元素。

- 对于每个 $i \in M_B$，在第 $i$ 回合中，Barbara 对每个 $j \in [n]$ 恰好阻挡 $u_{ij}$，$w_{ij}$ 中的一个元素。在第 $k-1$ 回合结束时，对于每个 $i \in M_P$ 和 $j \in [n]$，Barbara 阻挡 $\{u_{ij}, w_{ij}\}$ 中未受保护的元素。

- 在回合 $k$ 中，对于所有 $i < k$ 和 $j \in [n]$，Adam 选择 $\{u_{ij}, w_{ij}\}$ 中未被阻断的元素。

由此可知，我们引入的小工具确实将非限制博弈的实例 $I_{\text{in}}$ 归约为受限博弈的实例 $I_{\text{restr}}$。总之，当且仅当 Sat-V 上对抗选择博弈的原始实例是一个是实例时，Adam 才能获胜。

这就完成了对奇数 $k$ 和全局预算 $\Gamma_P$，$\Gamma_B$ 的困难性证明。现在我们讨论局部预算的情形。我们可以照搬全局预算的困难性证明，只需改为设定 $\Gamma_P = n$ 和 $\Gamma_B = 2n$。按照上述证明的逻辑，这并不构成任何限制；也就是说，在全局预算的构造中，保护者在最优策略下每回合恰好放置 $n$ 个令牌，而阻断者每回合恰好放置 $2n$ 个令牌。因此，完全沿用相同的证明，我们可以证明局部预算变体也是困难的。

最后，我们讨论 $k$ 为偶数的情况。在这种情况下，我们保留完全相同的构造和构件，只是现在阻断者是先手玩家，并执行所有奇数序号的行动。同样，保护者后手行动，并执行所有偶数序号的行动。为反映这一变化，规则 1–4 和构件中的“奇数”与“偶数”角色互换。此外，预算修改为$\Gamma_P = n(k-2)/2$ 和 $\Gamma_B = n(k-1)$。证明的其余部分基本保持不变。可以看出，同样的逻辑也适用；而且在 $k$为偶数的情况下，受限博弈也可以归约为非限制博弈。

总之，我们得到：当 $k$ 为常数时，Sat-V 上的组合 $k$-回合阻断博弈是 $\Sigma_k^p$-困难的；当 $k$ 属于输入时，该博弈是 PSPACE-困难的。这些复杂度类中的包含关系由引理 18 得出。

## 5.5 元归约

在上一小节中，我们已经证明，Sat-V 特例下的组合 $k$ 回合阻断博弈是 $\Sigma_k^p$-困难的。本小节中，我们将利用 NP-S 框架的结构性质，将这一结果推广到所有NP-S 完全问题，从而证明定理 23。为方便起见，我们重述该定理。

▶ **定理 23.** 设 $\Pi$ 是一个 *NP-S* 完全问题。若 $k \in \mathbb{N}$ 是常数，则组合$k$回合阻断博弈 $\Pi$ 是 $\Sigma_k^p$-完全的。若 $k$ 是输入的一部分，则该问题是 *PSPACE* 完全的。（这对全局预算变体和局部预算变体均成立。）

**证明。** 属于 $\Sigma_k^p$ 或PSPACE 已由引理 18 证明。

我们通过对偶模拟证明来证明其困难性。由于 $\Pi$ 是 NP-S 完全的，因此存在一个从 Sat-V 到 $\Pi$的解嵌入归约（$g, f$）。给定任意 $k$下 Sat-V 上的组合$k$回合阻断博弈的一个实例。我们定义一个新的、在 $\Pi$上的组合 $k$回合阻断博弈实例。为此，令 $I$ 为 Sat-V 上组合 $k$ 回合阻断博弈中的底层实例。于是，我们可以考虑实例 $g(I) = I'$，它是 $\Pi$ 的一个实例。

我们按以下方式将 $I'$ 转换为组合 $k$ 回合封锁博弈-$\Pi$ 的一个实例。嵌套集合 $C_i'$ 定义为 $C_i' := f(C_i)$，其中所有 $i \in [k-1]$ 均成立。预算 $\Gamma_P$ 和 $\Gamma_B$ 保持完全不变。由此定义了一个关于 $\Pi$ 的组合 $k$ 回合封锁博弈实例。根据我们对 $(f, g)$ 的假设，该实例可在多项式时间内计算得出。回忆一下，$f : \mathcal{U}(I) \to \mathcal{U}(I')$ 是一个单射，它将 Sat-V 的全集嵌入 $\Pi$ 的全集。这里，我们将其像 $f(\mathcal{U}(I)) =: \mathcal{U}_{emb}$ 称为嵌入全集，并且

$\mathcal{U}_{aux} = \mathcal{U}(I') \setminus \mathcal{U}(I)$ 为辅助全集。请注意，根据关于 $\Pi$ 的组合 $k$ 回合封锁博弈实例的构造，博弈的第一部分（即放置令牌的阶段）仅在 $\mathcal{U}_{emb}$ 中进行。

现在我们断言，在这两个实例中，同一位玩家都拥有获胜策略。如果能够证明这一点，我们就证明了该定理。为清楚起见，设Alice和Bob进行Sat-V的组合式拦截博弈，而Adam和Barbara进行 $\Pi$的组合式拦截博弈。

假设Alice在Sat-V的博弈中拥有获胜策略。那么Adam在 $\Pi$的博弈中也拥有获胜策略。他与Bob合作，两人分别在各自的博弈中采用模仿策略。模仿过程如下：每当Barbara在 $\Pi$ 的博弈中进行第 $i$回合，也就是说，她将令牌放在某个集合 $T'_i \subseteq C'_i$上时，Bob就在Sat-V的博弈中执行行动 $T_i = f^{-1}(T'_i)$。随后，Adam观察Alice在Sat-V博弈中的最佳应对 $T_{i+1}$，然后将令牌放在集合$T'_{i+1} = f(T_i)$上。这意味着，Adam实际上窃取了Alice在Sat-V博弈中的策略，而Bob则窃取了Barbara在 $\Pi$博弈中的策略。Adam在第1至 $k-1$回合使用单射 $f$。在最后一回合 $k$中，这样做是不可能的。相反，按照博弈规则，当且仅当保护者能找到一个与所有封锁标记均不相交的 $S \in \mathsf{S}(I')$时，保护者才能赢得 $\Pi$ 上的博弈。令 $T$ 为Sat-V博弈中放置的所有封锁标记的集合，令 $T'$ 为 $\Pi$博弈中放置的所有封锁标记的集合。那么 $T' = f(T)$。根据解嵌入性质，存在某个 $S' \in \mathsf{S}(I')$，使得 $S' \cap T' = \emptyset$，当且仅当存在 $S \in \mathsf{S}(I)$，使得 $S \cap f^{-1}(T') = S \cap T = \emptyset$。由于Alice能够赢得自己的博弈，Adam也就能够赢得自己的博弈。

现在假设 Bob 在 Sat-V 博弈中拥有获胜策略。那么 Barbara 在 $\Pi$ 上的博弈中也拥有获胜策略。这是因为通过类似的论证，她可以应用单射 $f$ 窃取 Bob 的策略。而 Alice 可以应用函数 $f^{-1}$ 窃取 Adam 的策略。在 Sat-V 博弈的第 $k$ 回合中，Alice 将无计可施，因为 Bob 必胜。根据解嵌入性质，在 $\Pi$ 上的博弈的第 $k$ 回合中，Adam 将无计可施。

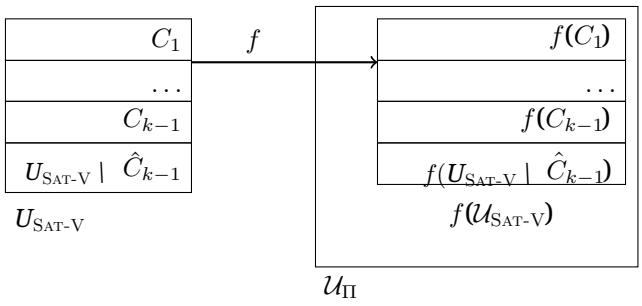


**图3** 将针对 Sat-V 的组合$k$-回合封锁博弈应用元归约到问题 $\Pi$ 的组合 $k$-回合封锁博弈时全集之间的关系。函数 $f$ 在 $C_i$ 与 $f$（$C_i$）的元素之间保持一一对应。

## 6 可调鲁棒问题

在本节中，我们考虑具有离散预算不确定性的可调鲁棒优化问题。我们证明，对于每一个NP-S 完全问题，其对应的

两阶段可调问题都是 $\Sigma_3^p$ 完全。此外，我们将两阶段可调鲁棒性的这些概念推广到多阶段设置，并证明 $k$阶段可调问题是 $\Sigma_{2k-1}^p$完全的；如果 $k$ 是输入的一部分，则为 PSPACE 完全。

在两阶段可调优化中，决策者必须在不确定性下寻找一个解。与通常的鲁棒优化只进行一个阶段的决策不同（这种做法可能过于保守），这里的决策分为两个阶段：第一阶段决策，也称为即刻决策；以及第二阶段决策，也称为观望决策。第一阶段决策必须在不确定性实现之前作出。随后，第二阶段的不确定性得以实现，此后再作出第二阶段决策。

我们将考虑可调鲁棒问题的两个变体。第一种变体中，全集元素上的成本属于不确定集的一部分。第二种变体中，可用决策属于不确定集的一部分。在这种情况下，不可用决策可以视为被阻断，或者也可以认为其成本高到使这些决策变得过于昂贵。我们首先介绍这样一种变体：其不确定集通过成本函数上的离散预算不确定性来定义。

决策者获得第一阶段决策的成本函数 $c_1 : \mathcal{U}(I) \to \mathbb{Z}$ 。第二阶段决策的输入包括成本下界 $\underline{c}_2 \in \mathbb{Z}$ 和成本上界 $\bar{c}_2 \in \mathbb{Z}$ ，以及不确定性参数 $\Gamma \in \mathbb{Z}$。第二阶段的不确定集由所有满足以下条件的情景组成：对于所有 $u \in \mathcal{U}(I)$， $\underline{c}_2(u) \leq c_2(u) \leq \bar{c}_2(u)$，且来自 $u \in \mathcal{U}(I)$ 的元素中，成本高于 $\underline{c}_2$ 的至多有 $\Gamma - 1$ 个。第二阶段的成本由对手设定。每当对手设定成本 $\bar{c}_2(e) > \underline{c}_2(e)$ 时，对手选择 $\bar{c}_2(e) = \underline{c}_2(e)$ 都是最优的。因此，我们可以将成本函数上的离散预算不确定集 $C_\Gamma$ 形式化定义为所有满足以下条件的成本函数：来自 $\mathcal{U}(I)$ 的元素中，成本为 $\bar{c}_2(u)$ 的至多有 $\Gamma$ 个，其余元素的成本均为 $\underline{c}_2(u)$。

$$C_\Gamma := \{c_2 \mid \forall u \in \mathcal{U}(I) : c_2(u) = \underline{c}_2(u) + \delta_u(\bar{c}_2(u) - \underline{c}_2(u)),\ \delta_u \in \{0,1\},\ \sum_{u \in \mathcal{U}(I)} \delta_u \leq \Gamma\}.$$

我们注意到，如果对于所有 $u \in \mathcal{U}(I)$， $c_2(u) \geq c_1(u)$，那么决策者在第一阶段选择完整决策将是最优的，从而使后手玩家的阶段变得无关紧要。然而，对于许多值得关注的问题，存在某个 $u \in \mathcal{U}(I)$，使得 $c_2(u) < c_1(u)$，因此需要第二阶段的决策。

我们将两阶段可调优化建模为一个包含三个行动的博弈，其中一个发生在第一阶段，另外两个发生在第二阶段。在博弈的第一个行动中，决策者选择一个集合 $S_1 \subseteq \mathcal{U}(I)$，其成本为 $c_1(S_1)$。在第二个行动中，对手选择一个成本函数 $c_2 \in C_\Gamma$。在第三个行动中，决策者选择一个额外的子集 $S_2 \subseteq \mathcal{U}(I)$。如果解 $S_1 \cup S_2$ 有效，即 $S_1 \cup S_2 \in \mathcal{S}(I)$，且总成本不超过指定的某个阈值 $t_{TS}$，那么决策者获胜；也就是说，$c_1(S_1) + c_2(S_2) \leq t_{TS}$。否则，对手获胜。

以下是一个源自旅行商问题的两阶段可调鲁棒问题示例。

两阶段可调旅行商问题

**实例：** 图 $G = (V, E)$，第一阶段成本函数 $c_1 : E \to \mathbb{Z}$，第二阶段成本函数 $c_2 : E \to \mathbb{Z}$、$\underline{c}_2 : E \to \mathbb{Z}$，阈值 $t_{TS} \in \mathbb{Z}$，预算 $\Gamma \in \mathbb{Z}$。

**问题：** 是否存在一个边子集 $S_1 \subseteq E$ ，使得对于所有成本函数 $c_2 \in C_\Gamma$，都存在一个额外的边子集 $S_2 \subseteq E \setminus S_1$ ，使得 $S_1 \cup S_2$ 是一个权重为 $c_1(S_1) + c_2(S_2) \leq t_{TS}$的哈密顿回路?

将两阶段可调旅行商问题视为三步双人博弈后，很容易看出两阶段可调旅行商问题属于 $\Sigma_3^p$ 。接下来，我们为 NP-S 中的每个问题定义两阶段可调问题。

▶ **定义 24**（两阶段可调问题）。 给定一个属于 *NP-S* 的问题 $\Pi = (I,U,S)$ 。与 $\Pi$ 相关联的两阶段可调问题记为 *Two-Stage Adjustable*-$\Pi$ ，定义如下：输入是一个实例 $I \in I$，以及一个第一阶段成本函数 $c_1 : U(I) \to \mathbb{Z}$、两个第二阶段成本函数 $c_2 : U(I) \to \mathbb{Z}$ 和 $c_2 : U(I) \to \mathbb{Z}$、一个阈值 $t_{TS} \in \mathbb{Z}$，以及一个不确定性参数 $\Gamma \in \mathbb{Z}$。问题是判断

$$\min_{S_1 \subseteq \mathcal{U}(I)} \max_{c_2 \in C_\Gamma} \min_{\substack{S_2 \subseteq \mathcal{U}(I) \setminus S_1 \\ S_1 \cup S_2 \in \mathcal{S}(I)}} c_1(S_1) + c_2(S_2) \leq t_{TS}.$$

本节的主要结果是以下定理。

▶**定理25**。 对于每个 *NP-S* 完全问题 $\Pi$，两阶段可调变体 *Two-Stage Adjustable*-$\Pi$ 是 $\Sigma_3^p$ -完全的。

我们将定理25的证明推迟到下一小节，在那里我们将证明一个更一般的结果。即使每项成本都为 0 或 1，上述定理仍然成立。这类问题等价于组合两阶段可调问题，我们接下来将对其进行定义。

▶ **定义 26** (组合两阶段可调问题)。 给定一个 *NP-S*问题 $\Pi$ =(*I, U, S*) 。与 $\Pi$ 相关联的组合两阶段可调问题记为 *Comb.* 两阶段可调-$\Pi$。其输入是一个实例 $I \in I$，以及三个集合 $U_1 \subseteq U(I)$、 $U_2 = U(I) \setminus U_1$ 和 $D \subseteq U_2$，还有一个不确定性参数 $\Gamma \in \mathbb{N}_0$。问题是判断

$$\exists S_1 \subseteq U_1 : \forall B \subseteq D \text{ with } |B| \leq \Gamma : \exists S_2 \subseteq (U_2 \setminus B) : S_1 \cup S_2 \in \mathcal{S}(I).$$

我们将 $B$ 中的元素称为脆弱元素，因为对手可以阻止这些元素。在组合两阶段可调问题中，对手可以阻止决策者选择 $B$ 中的元素。该问题可以重新表述为：每项成本不是 0 就是 1，且决策者必须选择一个成本为 0 的解。首先，对于每个元素 $e \in U_1$ ，令 $c_1(e) = 0$ ；对于每个元素 $e' \in U_2$，令 $c_1(e') = 1$ 。其次，对于每个元素 $e \in U_2$，令 $c_2(e) = 0$。此外，对于每个元素 $e \in D$，令 $c_2(e) = 1$ ；对于每个元素 $e' \in U_2 \setminus B$，令 $c_2(e') = 0$。最后，将阈值设为 $t_{TS} = 0$。由于决策者通过选择成本为 $\bar{0}$ 的解获胜，对手可以通过将成本设为 1，阻止 $\Gamma$ 个元素从 $B$ 中被选中。在这些条件下，决策者也能赢得组合版本。

## 6.1 多阶段可调鲁棒问题

在本小节中，我们将两阶段可调鲁棒性的定义推广为多阶段可调鲁棒性的定义。基于这一推广，我们证明 $k$阶段可调鲁棒问题是 $\Sigma_{2k-1}^p$完全的。如果 $k$ 是输入的一部分，我们证明该问题是 PSPACE 完全的。

为了引出我们对多阶段可调鲁棒性的讨论，我们首先介绍一种著名背包问题的在线变体，称为带估计的在线背包问题 [BGL+25]。给定一个容量为 $C \in \mathbb{N}$ 的背包，以及利润为 $p$ 、重量介于 $w : O \to \mathbb{N}$ 和 $w : O \to \mathbb{N}$ 之间的物品 $O = \{1, \ldots\ldots, n\}$，再给定一个不确定性参数 $\Gamma \in \mathbb{N}$，决策者希望完成以下任务。每件物品的重量

已知介于 $w_i$ 和 $w_i$ 之间。最多有 $\Gamma$ 件物品的重量会超过 $w$。随后，物品逐件到达。决策者可以称量到达的物品，并确定其实际重量。得知物品的实际重量后，决策者必须立即决定是将其装入背包还是丢弃，且这一决定不可更改。决策者希望最大化背包内物品的价值。此问题属于一类已得到研究的带估计的在线优化问题：背包问题 [BGL+25]、图探索 [GKN25], 、装箱问题 [GU25]。我们可以将上述问题建模为一个多阶段可调鲁棒问题。因此，本节的定理也为带估计的在线问题提供了复杂性结果。

在本节剩余部分中，我们将这些问题描述为两名玩家之间的博弈：一名决策者和一名对手，后者称为阻断者。我们进一步将复杂性分析聚焦于组合多阶段可调鲁棒问题，这类问题包含一组易受攻击元素 $D$。如上所述，多阶段可调鲁棒性的成本版本将组合版本作为特例。我们在推论30中给出组合版本的一般复杂性。对于每个 $k \in \mathbb{Z}_{\geq 2}$，我们得到一个具有以下规则的博弈：

- 输入包括将全集 $U$ 划分为 $k$ 个互不相交的子集 $U_1$，$U_2$，…，$U_k$，一个易受攻击元素子集 $D \subseteq U$，以及阻断者的预算 $\Gamma$。在第一轮中，代理人选择一个子集 $S_1 \subseteq U_1$。在
- 每一轮 $i \in \{2, \ldots, k\}$开始时，阻断者阻塞一个子集 $B_i \subseteq (U_i \cap D)$。随后，代理人选
- 择一个子集 $S_i \subseteq U_i \setminus B_i$。在每一轮中，阻断者可以阻塞任意数量的元素，但在博弈结
- 束时，被阻塞元素的数量不得超过 $\Gamma$，即 $\sum_{2 \leq i \leq k} |B_i| \leq \Gamma$。如果存在 $S_1$，…，$S_k$ 使得 $\bigcup_{i=1}^{k} S_i \in \mathcal{S}(I)$，则代理人获胜。否则，阻断者获胜。
-

这个博弈与第 5 节中介绍的保护-干预博弈类似。与保护-干预博弈不同，在这个可调鲁棒博弈中，决策者直接选择一个由未被阻断元素构成的子集，而不是通过放置令牌来保护它们。

▶**定义 27 (**组合 $k$阶段可调鲁棒问题)。 给定一个 *NP-S* 问题$\Pi = (I, U, S)$。与 $\Pi$ 对应的组合 $k$ 阶段可调鲁棒博弈记为 组合 $k$阶段可调鲁棒 $\Pi$ 博弈，定义如下：输入是一个实例 $I \in \mathcal{I}$，以及对全集 $U(I)$ 的划分，该划分由子集 $U_1$、……、$U_k$ 组成，另有一个易受攻击元素集合 $D$ 和一个不确定性参数 $\Gamma \in \mathbb{N}_0$。问题是判断

$$\exists S_1 \subseteq U_1, \forall B_2 \subseteq (D \cap U_2), \exists S_2 \subseteq (U_2 \setminus B_2), \ldots, \forall B_k \subseteq (D \cap U_k), \exists S_k \subseteq (U_k \setminus B_k) : \\ \text{if } \sum_{i=1}^{k} |B_i| \leq \Gamma, \text{ then } \bigcup_{i=1}^{k} S_i \in \mathcal{S}(I).$$

## 6.2 多阶段可调鲁棒问题的复杂性

我们先将其属于 $\Sigma^p_{2k-1}$ 表述为一个引理。

▶ **引理 28**。 设 $\Pi$ 是一个 *NP-S* 问题。若 $k \in \mathbb{N}$ 是常数，则 组合 $k$阶段可调鲁棒 $\Pi$ 属于 $\Sigma^p_{2k-1}$。若 $k$ 是输入的一部分，则该问题属于 *PSPACE*。

**证明**。 设 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 属于 NP-S 且 $I \in \mathcal{I}$是其一个实例。定义 27 通过交替量词直接描述了该问题。此外，所有集合 $S_i$ 和 $B_i$ 都可以

在多项式空间内编码，因为它们是$\mathcal{U}(I)$的子集。条件 $\bigcup_{i=1}^{k} S_i \in \mathcal{S}(I)$ 可以在多项式时间内检验，因为 $\Pi$ 属于 NP-S。◀

本节的主要结果是以下两个定理。

▶ **定理29.** 设 $\Pi$ 是一个 *NP-S* 完全问题。若 $k \in \mathbb{N}$ 是常数，则组合$k$-阶段可调鲁棒 $\Pi$是 $\Sigma_{2k-1}^{p}$-完全问题。若 $k$ 属于输入，则该问题是 *PSPACE* 完全问题。

定理29可用于证明相应的多阶段可调优化定理，其中使用成本函数而非易受攻击元素集合。

▶**推论30.** 设 $\Pi$ 是一个 *NP-S* 完全问题。若 $k \in \mathbb{N}$ 是常数，则$k$-阶段可调鲁棒 $\Pi$是 $\Sigma_{2k-1}^{p}$-完全问题。若 $k$ 属于输入，则该问题是 *PSPACE* 完全问题。

证明。与 $k = 2$类似，我们将基于 NP-S 中问题的组合可调鲁棒性推广到多阶段可调鲁棒问题，其中包括每个阶段 $1 \leq i \leq k$的成本函数 $c_i : \mathrm{U} \to \mathbb{Z}$和 $c_i : \mathrm{U} \to \mathbb{Z}$ 。为此，设 $t_{TS} = 0$。一方面，对所有 $u \in U_i$，我们令$c_i(u) = 0$ 。否则， $c_i(u) = 1$。另一方面，对所有$u \in U_i \setminus D$，我们令 $c_i(u) = 0$ 。否则， $c_i(u) = 1$。因此，决策者只能选择当前阶段中未被阻塞的元素。◀

我们对组合 $k$ 阶段可调鲁棒 $\Pi$ 的证明与组合 $k$ 回合保护-干预博弈的证明思路相同。首先，我们将对抗性选择 Sat-V 博弈的一个实例归约为组合 $k$ 阶段可调鲁棒 Sat-V 受限版本的一个实例，该版本包含附加规则。随后，我们通过添加构件来修改这一归约，从而证明原始的组合 $k$ 阶段可调鲁棒 Sat-V 是 $\Sigma_{2k-1}^{p}$ 完全的。最后，我们证明，对于任意 NP-S 完全的 $\Pi$，都存在从组合 $k$ 阶段可调鲁棒 Sat-V 到组合 $k$ 阶段可调鲁棒 $\Pi$ 的元归约。

## 6.3 可调鲁棒 Sat-V 的复杂性

在本小节中，我们将确立组合 $k$ 阶段可调鲁棒 Sat-V 的复杂性。

▶ **定理31**。对于所有常数 $k$， $\Pi = \textit{SAT-V}$ 的组合 $k$ 阶段可调鲁棒博弈 $\Pi$ 是 $\Sigma_{2k-1}^{p}$ 完全的。如果 $k$ 是输入的一部分，则该问题是 *PSPACE* 完全的。

### 受限可调鲁棒 Sat-V

我们首先介绍 Sat-V 上的受限可调鲁棒问题。我们的困难性归约从 Sat-V 上的 $\Sigma_{2k-1}^{p}$ 完全（$2k-1$）回合对抗选择博弈开始，该博弈在第 4.2 节中定义为 $k \in \mathbb{N}$ 。设 Sat-V $= (\mathrm{I}_{\mathrm{SAT}}, \mathrm{U}_{\mathrm{SAT}}, \mathrm{S}_{\mathrm{SAT}})$。此外，设 $I_{\mathrm{sel}} := (I, U_1, \ldots, U_{2k-1})$ 是 Sat-V 上（$2k-1$）回合对抗选择博弈的一个实例， $k \in \mathbb{N}$，其中 $I \in \mathrm{I}_{\mathrm{SAT}}$，且 $U_1 \cup \cdots \cup U_{2k-1}$ 是其变量集合 $\mathrm{U}_{\mathrm{SAT}}(I)$ 的（$2k-1$）划分。不失一般性，我们可以假设对于所有 $i \in \{1, \ldots, 2k-1\}$， $|U_i| = n$ 。最后，设 $U_i = \{u_{i1}, \ldots, u_{in}\}$。

我们回顾一下（$2k-1$）回合对抗选择博弈的规则：Alice 与 Bob 对弈，由 Alice 先行。对于 $i \equiv 1 \pmod 2$，Alice 选择 $S_i \subseteq U_i$。对于 $i \equiv 0 \pmod 2$，Bob 选择 $S_i \subseteq U_i$。如果 $\bigcup_{i=1}^{2k-1} S_i \in \mathcal{S}(I)$，则 Alice 获胜。否则，Bob 获胜。

给定实例 $I_{\text{sel}}$，我们定义一个新的受限 $k$ 阶段可调鲁棒 Sat-V 实例 $I_{\text{restr}}$。该构造是对我们在 Sat-V 上（$2k-1$）回合封锁博弈完备性证明中所用构造的修改。对于每个 $U_i := \{u_{ij} : j \in [n]\}$，令 $W_i := \{w_{ij} : j \in [n]\}$。元素 $w_{ij}$ 称为 $u_{ij}$ 的伪补。令 $W = \bigcup_{i=1}^{2k-1} W_i$。我们按如下方式构造 $I_{\text{restr}}$：

- 该实例 $I_{\text{restr}}$ 包含元素

$$\bigcup_{i=1}^{2k-1} U_i \cup \bigcup_{i=1}^{2k-1} W_i,$$

其中，$U$ 也是 $I_{\text{sel}}$的全集。

- 对于 $i \in [k]$，我们定义 $C_i$ 如下。 $C_1 = U_1 \cup W_1$。对于每个 $i \in \{2, \ldots, k\}$，

$$C_i = (U_{2i-2} \cup W_{2i-2}) \cup (U_{2i-1} \cup W_{2i-1}).$$

- $D = \bigcup_{i=2}^{k} (U_{2i-2} \cup W_{2i-2})$。
- 预算为 $\Gamma = n(k-1)$。

我们将受限可调鲁棒博弈的玩家称为 Adam（选择者）和 Barbara（阻断者）。与受限保护-阻断博弈一样，我们现在对 Adam 和 Barbara 施加附加规则，这将使 Alice 和 Adam 的行动之间形成自然对应关系，同时也使 Bob 和 Barbara 的行动之间形成对应关系。在后一种情况下，Barbara 将阻断 $U_i$ 中 Bob 未选择的元素。

- 规则1. 在第 1 阶段（即第 1 回合），Adam 必须为每个 $j \in \{1, \ldots, n\}$，从 $u_{1,j}$ and$w_{1,j}$ 中恰好选择一个。规则 2. 在阶段 $i \geq 2$ （即第 $2i-2$ 回合），Barbara 必须为每个 $j \in \{1, \ldots, n\}$，从 $u_{2i-2,j}$and $w_{2i-2,j}$ 中恰好阻挡一个。规则3. 在阶段 $i \geq 2$ （即第 $2i-1$ 回合），Adam 必须选择 $U_{2i-2} \cup W_{2i-2}$ 中每个未被阻挡的元素，并为每个 $j \in \{1, ..., n\}$，从 $u_{2i-1,j}$ and $w_{2i-1,j}$ 中恰好选择一个。

虽然我们已将上述内容规定为必须遵守的规则，但我们也考虑违反规则的情况。第一个违反规则的玩家输掉博弈。如果没有规则被违反，则令 $S'$ 表示 Adam 所选物品的并集。若$S' \cap U \in \mathcal{S}(I_{\text{SAT}})$，则 Adam 获胜。否则，Barbara 获胜。

我们将具有上述附加规则的 $k$ 阶段可调鲁棒博弈称为受限 $k$ 阶段可调鲁棒博弈。

▶**引理 32**。 当且仅当 *Alice* 在 *Sat-V* 上的受限可调鲁棒博弈的 阶段实例 $I_{restr}$ 中拥有获胜策略时，*Adam* 在 *Sat-V* 上的（）回合对抗选择博弈实例 $I_{sel}$ 中拥有获胜策略。

**证明**。 令 $S'_1$ 表示 Adam 在第 1 阶段选出的 $U_1 \cup W_1$ 元素。对于 $2 \leq i \leq k$，令 $T'_{2i-2}$ 表示 Barbara 从 $U_{2i-2} \cup W_{2i-2}$ 中选出的元素。此外，令 $S'_{2i-2} = (U_{2i-2} \cup W_{2i-2}) \setminus T'_{2i-2}$ 表示 Adam 从 $U_{2i-2} \cup W_{2i-2}$ 中选出的元素。令$S'_{2i-1}$ 表示 Adam 从 $U_{2i-1} \cup W_{2i-1}$ 中选出的元素。

现在我们证明：如果 $I_{\text{sel}}$ 是 Alice 的胜利策略，那么 $I_{\text{restr}}$ 就是 Adam 的胜利策略。我们首先假设 $I_{\text{sel}}$ 是 Alice 的胜利策略，并证明它也是 Adam 的胜利策略。我们的证明采用对偶模仿策略论证。假设 Alice 和 Barbara 在各自的博弈中都采取最优策略。Adam 和 Bob 在阶段 1 到 $k$ 的模仿策略如下。

- **Adam：** 对于每个从 $i = 1$ 到 $k$的阶段，如果 Alice 在阶段 $i$选择 $u_{2i-1,\ j}$ ，那么 Adam 就选择$u_{2i-1,\ j}$。否则，Adam 选择 $w_{2i-1,\ j}$。此外，对于 $i \geq 2$，Adam 选择 $S'_{2i-2}$ =$(U_{2i-2} \cup W_{i-2}) \setminus T'_{2i-2}$。 **Bob：** 对于每个从 $i = 1$ 到 $k$的阶段，
- $S_{2i-2} = U_{2i-2} \setminus T'_{2i-2}$。

由于 Adam 正在模仿 Alice，因此可得，对于 $i = 1$ 到 $k$, $S'_{2i-1} \cap U_{2i-1} = S_{2i-1}$。由于 Bob 正在模仿 Barbara，因此 $S_{2i-2} = U_{2i-2} \setminus T'_{2i-2}$，从而 $S_{2i-2} = U_{2i-2} \cap S'_{2i-2}$。由此可得 $S' \cap U = S$，因此 Adam 赢得了博弈。现在我们假设 $I_{\text{sel}}$ 是 Bob 的一场胜利，并将证明这同样是 Barbara 的一场胜利。在这种情况下，证明依赖于 Alice 和 Barbara 的模仿策略。

- **Alice：** 对于每个 $i = 1$ 到 $k$，Alice 选择 $S_{2i-1} = S'_{2i-1} \cap U_{2i-1}$。 **Barbara：** 对于每
- 个 $i = 1$ 到 $k$，以及每个 $j = 1$ 到 $n$，如果 Bob 选择 $u_{2i-2,\ j}$，那么 Barbara 阻止 $w_{2i-2,\ j}$。如果 Bob 不选择 $u_{2i-2,\ j}$，那么 Barbara 阻止 $u_{2i-2,\ j}$。

由于 Alice 在模仿 Adam，因此可得 $S'_{2i-1} \cap U_{2i-1} = S_{2i-1}$ for $i = 1$ to $k$。由于 Barbara 在模仿 Bob，因此可得 $U'_{2i-2} \setminus T'_{2i-2} = S_{2i-2}$。由于 Adam 选择的是 Barbara 被阻塞元素的伪补，$S'_{2i-2} = (U_{2i-2} \cup W_{2i-2}) \setminus T'_{2i-2}$，因此 $S'_{2i-2} \cap U_{2i-2} = S_{2i-2}$。由此可得 $S' \cap U = S$。由于该博弈是 Bob 的获胜游戏，$S \notin \mathcal{S}$，因此 $S' \cap U \notin \mathcal{S}$，且这是 Barbara 的获胜游戏。◀

### 为规则引入构件

在证明了受限可调鲁棒 Sat-V 的困难性之后，接下来我们将引入构件，以在一般的可调鲁棒 Sat-V 实例中强制执行规则。为此，我们推导出以下定理。

▶**定理 33.** 组合 $k$ 阶段可调鲁棒 *Sat-V* 对所有常数 $k$ 都是 $\Sigma^p_{2k-1}$ 完全的；若 $k$ 属于输入，则为 *PSPACE* 完全。

**证明。** 由引理 28 可知该问题属于相应复杂度类。为证明其困难性，我们使用构件，将 Sat-V 上的 $(2k-1)$ 回合对抗选择博弈归约到不受限的版本的 $k$ 阶段可调鲁棒 Sat-V，并确保玩家采用最优策略时，受限的 $k$阶段可调鲁棒问题的三个附加约束均得到满足。因此，结合引理 32，即可证明该定理。

给定一个在 Sat-V 上的（$2k-1$）回合对抗性选择博弈实例 $I_{\text{sel}}$:=($I_{\text{SAT}}$，$U_1$，.....，$U_{2k-1}$)，我们定义了一个受限 $k$ 阶段可调鲁棒 Sat-V 实例 $I_{\text{restr}}$。随后，我们进一步将实例 $I_{\text{restr}}$ 转换为一个非限制 $k$ 阶段可调鲁棒 Sat-V 实例 $I_{\text{adj}}$，使得无论 Adam 和 Barbara 如何进行非限制博弈，在任何最优策略中，Adam 和 Barbara 都遵守规则 1–3。因此，受限问题与非限制问题等价。我们先给出 $\psi$ 的形式化定义，然后解释其不同组成部分背后的直觉。给定关于变量 $u_{ij}$ 的初始公式 $\varphi$，其中 $i \in [k]$，$j \in [n]$，我们为 $i \in [k]$，$j \in [n]$ 引入伪补项 $w_{ij}$。此外，对于所有偶数的 $i \in [k]$，$i$，我们引入额外的辅助变量 $s_{ij}$；对于所有奇数的 $i \in [k-1]$，我们引入变量 $t_{ij}$，其中 $i$、$j \in [n]$；并引入两个最终辅助变量 $s$，$t$。公式 $\psi$ 定义如下：

**公式** $\psi$**:**

$$(\varphi(U) \vee s) \tag{$\ell_1$}$$

$$\wedge \bigwedge_{i \text{ even}} \bigwedge_{j=1}^{n} (s_{ij} \to (u_{ij} \wedge w_{ij})) \tag{$\ell_2$}$$

$$\wedge (\overline{s} \vee \bigvee_{i \text{ even}} \bigvee_{j=1}^{n} s_{ij}) \tag{$\ell_3$}$$

$$\wedge \bigwedge_{i=1}^{2k-1} \bigwedge_{j=1}^{n} ((u_{ij} \vee w_{ij} \vee s) \wedge (\overline{u}_{ij} \vee \overline{w}_{ij} \vee s)). \tag{$\ell_4$}$$

划分 $C_1$、.....、$C_{2k-1}$、易受攻击元素集合 $D$ 以及预算 $\Gamma$均与受限博弈中的相同。也就是说，对于 $1 \le i \le k$， $D = \bigcup_{i=2}^{k}(U_{2i-2} \cup W_{2i-2})$ 和 $\Gamma = n(k-1)$，有 $C_1 = U_1 \cup W_1$ 和 $C_i = (U_{2i-2} \cup W_{2i-2}) \cup (U_{2i-1} \cup W_{2i-1})$。请注意，阻断者只能通过阻断集合$D$中包含的变量 $u_{ij}$， $w_{ij}$ 来赋予假值。然而，决策者可以在自己的每个回合中赋予真值或假值。

与封锁博弈的证明一样，我们引入一个作弊检测变量 $s$ ，用于检测阻断者作弊的情况。对于这个证明，无需引入第二个作弊检测变量。回想一下受限博弈的规则。

- 规则1。在第 1 阶段（以及第1回合），Adam 必须为每个 $j \in \{1, \ldots, n\}$，从 $u_{1,j}$ and $w_{1,j}$ 中恰好选择一个。规则2。在第 $i \ge 2$ 阶段（以及第 $2i-2$ 回合），Barbara 必须为每个 $j \in \{1, \ldots, n\}$，从 $u_{2i-2,j}$and $w_{2i-2,j}$ 中恰好阻挡一个。规则3。在第 $i \ge 2$ 阶段（以及第 $2i-1$ 回合），Adam 必须选择 $U_{2i-2} \cup W_{2i-2}$ 中每个未被阻挡的元素，并为每个 $j \in \{1, ..., n\}$，从 $u_{2i-1,j}$ and $w_{2i-1,j}$ 中恰好选择一个。

我们声称，如果一名玩家在某个回合中作弊，那么另一名玩家就必定获胜。我们再次考虑所有规则的重要性，这些规则按照回合构成一个层级。更准确地说，规则1最为重要，因为它适用于第一阶段。然后，对于每个阶段$i \in \{2, \ldots, k\}$，我们认为规则2（Barbara 的行动）比同一阶段 $i$ 中的规则3（Adam 的行动）更重要，而阶段 $i-1$ 的规则比阶段$i$的规则更重要。因此，第一个违反规则的玩家将输掉博弈。

第 $\ell_1$ 行是原始变量 $U$ 以及作弊检测变量 $s$构成的原始公式 $\varphi(U)$。如果 Adam 将 $s$ 设为真，他就能轻松满足这个子公式。第$\ell_2$ 行实现了阻断者 Barbara 对作弊行为的检测。如果她同时阻断 $u_{ij}$和 $w_{ij}$，预算就不足以阻断 $u_{i'j'}$ 和 $w_{i'j'}$中的一个。此时 Adam 可以将 $u_{i'j'}$ 和 $w_{i'j'}$ 赋值为真，并将 $s_{ij}$ 赋值为真。这个结果会传递到第$\ell_3$ 行，使 Adam 能够将 $s$ 设为假。第 $\ell_4$ 行充当 Adam 的作弊检测装置，确保对于所有 $i \in \{1, \ldots, 2k-1\}$ 和$j \in \{1, \ldots, n\}$，他恰好选择 $u_{ij}$ 和 $w_{ij}$ 中的一个。总之，我们可以得出以下几种情况。

**情况 1：** 规则*1*是第一个被违反的规则。因此，对于某个 $j \in \{1, \ldots, n\}$，Adam 将 $u_{1j}$ 和 $w_{1j}$ 同时设为真或同时设为假。显然，Barbara 之前并未作弊。从这一刻起，Barbara 只需遵守规则2，就能轻松赢得博弈，因为 Adam 无法将变量 $s$ 设为真。

**情况 2：** 规则*2*是第一个被违反的规则。 我们考虑 Barbara 既没有阻止 $u_{ij}$ 也没有阻止 $w_{ij}$的第一种情况。此时，Adam 可以将 $u_{ij}$ 和 $w_{ij}$ 都设为真，并通过将 $s_{ij}$ 设为真且将 $s$ 设为真来满足 $\ell_2$。因此， $\ell_1$ 和 $\ell_4$ 都显然得到满足，Adam 赢得了这场博弈。

第二种情况中，Barbara 阻止了 $u_{ij}$ 和 $w_{ij}$。由于 Barbara 的预算只有$\Gamma = n\,(k-1)$，因此存在一些 $i' \geq i$ 和 $j \in \{1,\ldots,n\}$，使得 $u_{i'j'}$ 和 $w_{i'j'}$ 都没有被阻止。然后，Adam 可以遵循规则进行博弈，直到阶段 $i'$，并采用情况 1 中的相同策略来赢得博弈。

**情况3：** 规则*3*最先被违反。Barbara 按规则行事直到阶段 $i$，使得对于某个 $j \in \{1,\ldots,n\}$，Adam 既不选择 $u_{ij}$，也不选择 $w_{ij}$。此时，Barbara 只需继续按规则行事，就有获胜策略。于是，Adam 无法满足 $\ell_4$，因为 $s$ 必须设为假，而 $u_{ij}$ 和 $w_{ij}$ 都设为假，因此他输掉了博弈。

通过这种情况分类，我们得出结论：Adam 和 Barbara 最好都遵循全部三条规则。如果遵循全部三条规则，那么 $\ell_2$、$\ell_3$和 $\ell_4$ 中的子句都得到满足，并且 $s$被迫取值为 0。因此，$\psi(U)$ 可归约为 $\varphi(U)$。总之，Adam 能够在实例 $I_{restr}$ 上赢得受限可调鲁棒 Sat-V，当且仅当他能够在实例 $I_{adj}$上赢得非受限可调鲁棒 Sat-V，从而完成证明。◀

## 6.4 元归约

最后一步，我们希望证明所有基于 NP-S 完全问题的可调鲁棒问题都是完备的。为此，我们展示存在一种从组合可调鲁棒 Sat-V 出发的元归约，该归约使用从 Sat-V 到 $\Pi$ 的底层解嵌入归约。

▶**定理 34**。如果 $\Pi$ 是 *NP-S* 完全的，那么 组合 $k$ 阶段可调鲁棒 $\Pi$ 在 $k$ 为常数时是 $\Sigma^p_{2k-1}$-完全的；如果 $k$ 是输入的一部分，则属于 *PSPACE*。

证明。包含关系的证明与引理 28 类似。我们进一步通过对偶模拟策略证明其困难性。由于 $\Pi$ 是 NP-S 完全的，因此存在一个从 Sat-V 到 $\Pi$ 的解嵌入归约 $(g, f)$。令 $I$ 为 Sat-V 的实例，$g(I) = I'$ 为 $\Pi$的实例。Alice 和 Bob 进行可调鲁棒 Sat-V 博弈。Adam 和 Barbara 进行可调鲁棒 $\Pi$ 博弈。对于 Adam 的每个回合 $i \in \{1,\ldots,2k-1\}$，$i$为偶数：如果 Alice 选择 $S_i \subseteq U_i$，Adam 就选择 $S'_i = f(S_i)$。对于 Barbara 的每个回合 $i \in \{2,\ldots,2k-2\}$，$i$ 为偶数：如果 Bob 选择 $S_i \subseteq U_i$，Barbara 就选择 $B'_i = f(U_i \setminus S_i)$。随后 Adam 选择 $S'_i = f(U_i \setminus B_i)$。这些策略的最优性可以用与定理 23 的证明类似的方法论证。◀

# 7 其他 NP-S模型

在本节中，我们考察 NP-S 问题的不同模型。NP-S 问题的全集及其相应解集，不仅会对多项式层次和 PSPACE 中的相应问题产生重大影响，也会影响这些问题之间可能存在的解嵌入归约。一方面，全集和解隐式地定义了多项式层次中问题的不同版本，例如阻断问题：在这里，我们要寻找一个阻断者 $B \subseteq D$，其中 $D \subseteq \mathcal{U}$，也就是说，易受攻击元素集合 $D$ 直接取决于 NP-S 问题中全集$\mathcal{U}$ 的选择。另一方面，许多 NP 完全性归约（或许是隐式地）被设计成解嵌入归约，A 节中描述的问题便证明了这一点。即使在涉及复杂“构件”的 NP 完全性归约中，通常也存在一个潜在的解嵌入映射 $f$。然而，有些 NP完全问题并非 NP-S 完全问题，也有些 NP 完全性归约并非解嵌入归约。因此，本节将进一步探讨这类问题和

归约，并论证其中大多数都可以在不同模型下解释为 NP-S 完全问题或解嵌入归约。

在本节的第一部分，我们通过分析可满足性的不同版本，讨论在问题 $\Pi \in$NP-S 的全集中使用互补变量，从而开始我们的研究。互补变量是现有变量的对应物，可作为解嵌入归约的一部分发挥作用；我们还可以利用它们定义基于文字的模型。这可以推广到其他组合问题，我们将用对偶问题这一概念加以刻画。所有这些模型还可能使由 NP-S完全问题生成的阻断问题具有理想性质。在第二部分，我们研究属于全集覆盖的 NP-S 问题。这类问题可能是划分问题和装箱问题，在典型的子集解释下，它们并不直接是 NP-S 完全的。事实上，这些问题可以用不同方式建模，从而使它们成为 NP-S 完全的。在第三部分，我们讨论一种称为两步验证 的技术，并将其应用于最小化迟交任务的排序问题，以表明该问题存在一个 NP-S 完全模型。

## 7.1 互补变量与问题的对偶

在第 3 节中，我们定义了可满足性（基于变量），记作 Sat-V，其中全集是变量集合；还定义了可满足性（基于文字），记作 Sat-L，其中全集是文字集合。此外，还可以定义可满足性（基于互补变量）问题，其中全集再次为变量集合，但解被定义为被赋值为假的变量。

由于所有已知的归约都从可满足性出发，因此 NP-S 完全问题的解结构与这些问题中的某一种相似，我们据此对这些概念进行推广。在此基础上，我们讨论互补模型和基于文字的模型的使用如何丰富多项式层次和 PSPACE 中相应问题的建模。

设 $\Pi = (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 是一个 NP-S 问题。我们可以将 $\mathcal{U}(I)$ 解释为某个实例 $I$ 的二进制变量集合，正如第 3 节所讨论的那样。我们将 $\mathcal{U}(I)$ 表示为 $\mathcal{U}(I)$ 的互补变量集合。也就是说，对于 $x_j \in \mathcal{U}(I)$，我们有 $x_j = 1 - x_j$。基于这一观察，我们为 NP-S 问题的模型推导出一般概念。

### 基于补集的模型

我们将 $\Pi$ 的 基于补集的模型定义为 $\Pi_C = (\mathcal{I}, \mathcal{U}_C, \mathcal{S}_C)$。为此，我们定义 $\mathcal{U}_C = \mathcal{U}$，并相应地修改解集 $\mathcal{S}$，即对于每个实例 $I \in \mathcal{I}$：如果 $S \in \mathcal{S}(I)$，那么 $\{u \mid u \in S\} \in \mathcal{S}_C(I)$；如果 $S_C \in \mathcal{S}_C(I)$，那么 $\{u \mid u \in S_C\} \in \mathcal{S}(I)$。

例如，考虑问题 团（基于边），给定一个图 $G = (V, E)$ 和一个数 $k$，要求我们找到一个集合 $S \subseteq E$，使得如果 $\{u, v\}$、$\{v, w\} \in E$，那么 $\{u, w\} \in E$ 且 $|S| \geq k$。此外，考虑问题 独立集（基于边），给定一个图 $G' = (V'E')$ 和一个数 $k'$，要求我们找到一个集合 $S \subseteq E$，使得如果 $\{u, v\}$、$\{v, w\} \in E$，那么 $\{u, w\} \notin E$ 且 $|S| \geq k$。将团（基于边）的边 $E$ 的补集解释为独立集（基于边）中的边 $E' = E$，则问题 团（基于边）就是独立集（基于边）的基于补集的模型。

▶**引理 35**。 SAT-V 的基于补的模型是 *NP-S* 完全。

**证明**。 回想一下，在定理 7 中，我们证明了 SAT-V 是 NP-S完全的。在该证明中，我们可以将每个变量 $x_i$ 替换为一个不同的变量 $y_i$。因此，每次出现 $x_i$

在子句中都会被替换为 $y_i$，而每次出现 $x_i$ 都会被替换为 $\bar{y_i}$。由此得到的证明表明，Sat-V 的基于补的模型是 NP-S 完全的。◄

►**定理 36.** 假设 $\Pi$ 是一个 *NP-S* 完全问题，且其全集由变量构成。那么， $\Pi$ 的基于补的模型也是 *NP-S* 完全的。

证明。从 Sat-V 到 $\Pi$的解嵌入归约，也是从 Sat-V 的基于补的模型到 $\Pi$的基于补的模型的解嵌入归约。为此，设从 Sat-V 到 $\Pi$ 的归约将 $x_i$ 映射到 $f(x_i)$，那么我们可以将 Sat-V 中的一个变量$x_i$ $_C$ 映射到 $\Pi_C$中的 $f(x_i)$。由于 $S \in \mathrm{S}(I)$，因此 $\{u \mid u \in S\} \in \mathrm{S}_C(I)$，反之亦然，所以这是一个正确的解嵌入归约。◄

### 基于文字的模型

$\Pi$ 的基于文字的模型通过重新定义 $\Pi = (\mathcal{I},\mathcal{U},\mathcal{S})$ 为 $\Pi_L = (\mathcal{I},\mathcal{U}_L,\mathcal{S}_L)$ 得到，其中，对于每个实例 $I \in \mathcal{I}$，$\mathcal{U}_L(I) = \mathcal{U}(I) \cup \overline{\mathcal{U}(I)}$。我们进一步将 $S \in \mathcal{S}(I)$ 替换为$S' \in \mathcal{S}_L(I)$，如下所示：(i) $x_j \in S$，当且仅当 $x_j \in S'$；(ii) $x_j \notin S$，当且仅当 $x_j \in S'$。

►**定理 37**。 假设 $\Pi$ 是某个 *NP-S* 完全问题的基于变量的模型。那么，基于文字的变体 $\Pi_L$ 是 *NP-S* 完全的。

证明。在将 $\Pi$ = (I, U, S) 归约为 $\Pi_L$ = (I, $\mathrm{U}_L$, $\mathrm{S}_L$) 的过程中，我们定义 $g(I) = I$ 对所有$I \in$ I 均成立。由于 $\mathrm{U}_L(I)$ 包含集合 U($I$)，我们对 U($I$) 中的每个变量 $x_i \in$ 定义 $f(x_i) = x_i$。此归约是解嵌入。◄

►**引理 38**。 以下命题为真：$\textsc{Sat-V} \leq_{\mathrm{SE}} \textsc{Sat-L}$ 以及 $\textsc{Sat-L} \leq_{\mathrm{SE}} \textsc{Sat-V}$。

证明。Sat-V $\leq_{\mathrm{SE}}$ Sat-L 可由定理 37 得出。为证明 Sat-L $\leq_{\mathrm{SE}}$ Sat-V，我们定义如下归约。对于每对文字 $\ell$、$\ell \in L$，我们创建一个变量对 $x^t$、$\overline{x^f} \in X$。我们进一步定义 $f(\ell) = x^t$和 $f(\ell) = x^f$。此外，我们将某个子句 $c \in \overline{C}$ 中每个 $\ell$ 的出现替换为 $x^t$，并将某个子句 $c \in C$ 中每个 $\ell$ 的出现替换为 $x^f$。最后，我们将子句 $\{x^t \vee x^f\}$ 和 $\{x^t \vee x^f\}$ 加入子句集。◄

### 对偶问题

对于 NP-S 中的每个基于文字的问题，我们定义另一个称为“对偶”的问题。对于基于文字的问题 $\Pi \in$ NP-S，$\Pi = (\mathcal{I},\mathcal{U},\mathcal{S})$ 的对偶问题，记为 $\Pi_D = (\mathcal{I},\mathcal{U},\mathcal{S}_D)$，定义为 $\mathcal{S}_D(I) = \{S \subseteq \mathcal{U}(I) \colon (\mathcal{U}(I) \setminus S) \in \mathcal{S}(I)\}$。

例如，考虑顶点覆盖问题：给定一个图$G = (V, E)$和一个数 $k$，要求找出一个集合 $S \subseteq V$，使得对于所有 $e \in E$ 和 $|S| \leq k$均满足 $|S \cap e| \geq 1$。在顶点覆盖的基于文字的变体中，全集为$\mathcal{U}(I) = V \cup \bar{V}$。对于每个顶点 $v \in V$，根据 $v$ 是否被选中，解中包含 $v$ 或 $\bar{v}$。其对偶问题是独立集的基于文字的版本：给定一个图 $G = (V, E)$ 和一个数 $\ell$，要求找出一个子集 $S \subseteq V$，使得 $|S| \geq \ell$，并且对于每条边 $(v, w) \in E$, $|S \cap \{v, w\}| \leq 1$。

为了看出这种直接对应关系，假设$\mathcal{S}_{VC}$ 是图 $G = (V, E)$、阈值为 $k$ 的实例中顶点覆盖问题的解集。设$\mathcal{S}_{IS}$ 是图 $G = (V, E)$、$\ell = n - k$ 的实例中独立集问题的解集。那么$\mathcal{S}_{IS} = \{S \subseteq (V \sqcup \bar{V}) : ((V \sqcup \bar{V}) \setminus S) \in \mathcal{S}_{VC}\}$。

►**定理 39.** 如果一个基于文字的问题 $\Pi$ 是 *NP-S* 完全的，那么 $\Pi$ 的对偶也同样是 *NP-S* 完全的。

证明。假设 $\Pi = (\mathrm{I}, \mathrm{U}, \mathrm{S})$ 是某个 NP-S 完全问题的基于文字的模型，而 $\Pi_D = (\mathrm{I}, \mathrm{U}, \mathrm{S}_D)$ 是其对偶。由于 $\Pi$ 是一个基于文字的模型，对于每个实例 $I \in \mathrm{I}$，全集是一个划分 $\mathrm{U}(I) = U \cup U$。此外，根据基于文字的模型的定义，对于每个 $S \in \mathrm{S}(I)$，$u \in S$ 当且仅当 $u \notin S$。因此，对于所有实例 $I \in \mathrm{I}$，我们可以映射 $f(u) = u$ 以及 $f(u) = u$，对所有 $u \in \mathrm{U}(I)$ 均成立。由此可得 $\mathrm{U}(I) = f(\mathrm{U}(I))$，且 $S \in \mathrm{S}(I)$ 当且仅当 $(\mathrm{U}(I) \setminus S) \in \mathrm{S}_D(I)$。这个映射是双射，并且也是解嵌入。◀

如前所述，同一个 NP 中问题的不同模型会导出 NP-S 中的不同问题，而它们又会映射到多项式层次中的不同问题。例如，可能会出现 Interdiction-Knapsack 问题。当全集 $\mathcal{U}$ 由变量构成时，阻断者可以在她的回合中阻止物品进入背包。反之，若集合 $\mathcal{U}$ 同时包含变量及其补项，阻断者还可以强制物品进入背包。因此，文字模型和变量模型（无论是否包含补项）会导出不同的阻断博弈。

## 7.2 全集覆盖问题

NP-S 中有一类不可能是 NP-S 完全的问题，它们具有“全集覆盖”性质。如果对于每个满足 $\mathcal{S}(I) \neq \emptyset$ 的实例 $I \in \mathcal{I}$，都有 $\bigcup_{S \in \mathcal{S}(I)} S = \mathcal{U}(I)$，那么我们称问题 $\Pi$ 属于 NP-S 且是全集覆盖的。

▶**引理 40.** 如果问题 $\Pi$ 是全集覆盖问题，那么该问题不可能是 *NP-S* 完全的。

证明。我们采用反证法。假设 $\Pi = (\mathrm{I}_\Pi, \mathrm{U}_\Pi, \mathrm{S}_\Pi)$ 既是全集覆盖问题，又是 NP-S 完全的。那么，必定存在从 Sat-V $= (\mathrm{I}_{\mathrm{SAT}}, \mathrm{U}_{\mathrm{SAT}}, \mathrm{S}_{\mathrm{SAT}})$ 到 $\Pi$ 的 SE 归约。现在，令 $I \in \mathrm{I}_{\mathrm{SAT}}$ 为 Sat-V 的一个实例，并选择该实例使得 $x_1 \in \mathrm{U}_{\mathrm{SAT}}(I)$，且 $x_1$ 不属于任何解 $S \in \mathrm{S}_{\mathrm{SAT}}(I)$。此外，令 $g(I)$ 为通过 SE 归约从 $I$ 得到的实例。由于 $\Pi$ 是全集覆盖问题，因此存在一个解 $S' \in \mathrm{S}_\Pi(g(I))$，使得 $f(x_1) \in S'$。由于该归约是 SE 归约，因此存在一个解 $S = f^{-1}(S' \cap f(\mathrm{U}_\Pi(I))) \in \mathrm{S}_{\mathrm{SAT}}$，其中包含 $x_1$，这与我们的假设矛盾。◀

接下来我们讨论其他一些自然且重要的全集覆盖问题。NP-S 中全集覆盖问题的第一个例子是 3-着色问题，其表述如下。

> 3-着色（基于整数规划的模型）实例 I：无向图 $G = (V, E)$。全集 U：$V \times \{1, 2, 3\}$。解 S：所有 $S \subseteq V \times \{1, 2, 3\}$ 的集合，其中 $(v, i) \in S$ 表示顶点 $v$ 被赋予颜色 $i$，并且若 $(v, i) \in S$ 且 $(w, i) \in S$，则 $\{v, w\} \notin E$。

如果 $S$ 是任意一个可行解，那么通过将颜色 1 与颜色 2 或颜色 3 互换，可以得到另一个解 $S'$。解集中的对称性使该问题成为全集覆盖问题，并使其无法成为 NP-S 完全问题。

NP-S 中全集覆盖问题的另一个例子是下文所述的装箱问题。

装箱问题（基于整数规划的模型）实例 I：集合 $A \subseteq \mathbb{N}$，箱子容量 $B \in \mathbb{N}$，数量 $k \in N$。全集 U：$\{(a,j) : a \in A$ 且 $j \in \{1,\ldots,k\}\}$，表示将 $A$ 中的一个元素分配到箱子 $\{1,\ldots,k\}$ 之一。解 S：所有 $S \subseteq \{(a,j) : a \in A$ 且 $j \in \{1,\ldots,k\}\}$ 的集合，其中对于每个 $a \in A$，恰有一个 $j$ 满足 $(a,j) \in S$，并且对于所有 $j \in \{1,\ldots,k\}$，$\sum_{(a,j)\in S} a \leq B$。

在这个装箱问题的 NP-S 模型中，对于每个可行解 $S$，都有 $k!$ 个不同的解，每个解都可以通过置换箱子的标签得到。根据引理 40，这个装箱问题模型不是 NP-S 完全。3-着色模型和装箱问题模型都是全集覆盖的，因为每个解集 $\mathcal{S}$ 在置换后保持不变。

在本小节中，我们将构建 3-着色和装箱问题的模型，这些模型都是 NP-S 完全的。在第 7.2.1 节中，我们将讨论一种基于等价关系的建模方法，可用于构造 NP-S 完全的划分问题和着色问题。在第 7.2.2 节中，我们将讨论一种建模方法，其中$\mathcal{U}$ 是基集的子集集合。我们将这些模型称为子集生成型模型。我们将证明，4-划分和三角形装箱问题的子集生成型模型都是 NP-S 完全的。

## 7.2.1 着色与划分问题

如果划分问题的基于整数规划的模型是全集覆盖的，那么根据引理 40，该问题的这个模型不可能是 NP-S 完全的。这里，我们介绍一种将各部分（或颜色）建模为等价关系中的等价类的方法。这种建模方式打破了基于整数规划的模型通常具有的对称性。我们证明以下问题都是 NP-S 完全的：3-着色、4-划分和装箱问题。

**等价关系与等价类的建模。**

集合 $S$ 上的等价关系是一种关系 $\sim$，它对所有 $a$、$b$、$c \in S$ 满足以下三个性质：自反性（$a \sim a$）、对称性（如果 $a \sim b$，那么 $b \sim a$）以及传递性（如果$a \sim b$ 且 $b \sim c$，那么 $a \sim c$）。给定集合 $S$ 上的等价关系 $\sim$，元素 $a \in S$ 的等价类定义为 $[a] = \{x \in S : x \sim a\}$。每个元素恰好属于一个等价类，并且任意两个不同的等价类都不相交。因此，$S$ 的等价类构成 $S$ 的一个划分。

我们证明，3着色问题的等价关系模型是 NP-S 完全的。

3-着色（等价关系模型）**实例 $\mathcal{I}$：** 无向图 $G = (V, E)$.**全集 $\mathcal{U}$：** $\{(v, w) : v, w \in V$且$\{v, w\} \notin E\}$。**解 $\mathcal{S}$：** 一个在 $S \subseteq \mathcal{U}$ 上的等价关系，且至多有三个等价类。

▶**定理41。** 3-着色 的等价关系模型是 *NP-S* 完全的。

**证明。** 我们证明，从3-SAT 到 3-着色 Garey 和 Johnson 所描述的归约 [GJ79] 是从 3SAT-V 到3-着色（等价关系模型）的 SE 归约。

令 3Sat-V $= (\mathcal{I}, \mathcal{U}, \mathcal{S})$ 和 3-着色 $= (\mathcal{I}', \mathcal{U}', \mathcal{S}')$。对于每个实例 $I \in \mathcal{I}$，Garey 和 Johnson 中的归约构造了一个实例 $g(I)$，其中存在一个无向

图 $G = (V, E)$。对于 U($I$) 中的每个变量 $x_i$，都有一个顶点 $v_i \in V = \mathrm{U}'(g(I))$。顶点集 $V$还包括另外三个顶点，分别标记为 $T$、$F$ 和 $D$；$T$ 表示“真”，$F$ 表示“假”，$D$ 表示“虚拟”。对于 $I$ 的每个子句，还有三个顶点。该归约正确性的证明表明：$\alpha$ 是 3Sat 的一个满足真值赋值，当且仅当 $G$ 的顶点存在一种 3-着色，使得：(1) 如果 $\alpha(x_i) = 1$，则相应的顶点 $v_i$ 与顶点 $T$ 具有相同的颜色；(2) 如果 $\alpha(x_i) = 0$，则相应的顶点 $v_i$ 与顶点 $F$ 具有相同的颜色。对于 3- 着色的等价关系模型，该归约是解嵌入的。函数 $f$ 定义如下。对于每个变量 $x_i \in \mathrm{U}(I)$，$f(x_i) = (v_i, T)$。这可以解释为：如果 $x_i$ 为真，则顶点 $v_i$ 与顶点 $T$ 具有相同的颜色。◄

接下来，我们定义 4-划分问题的等价关系模型，并证明它是 NP-S 完全的。

4-划分问题（等价关系模型）
实例 I：一个集合 $A = \{a_1, a_2, ..., a_{4n}\}$、整数 $B$，以及每个 $i \in \{1, ..., 4n\}$ 对应的整数大小 $s(a_i)$，满足 $B/5 < s(a_i) < B/3$。全集 U：$\{(a_i, a_j) : i, j \in \{1, \ldots, 4n\}$。
解 S：所有满足 $S$ 是等价关系，且对于 $S$ 的每个等价类 $C$，$\sum_{a_i \in C} s(a_i) = B$ 的 $S \subseteq \{(a_i, a_j) : i, j \in \{1, \ldots, 4n\}\}$ 的集合。

►**定理 42**。 4-划分 的等价关系模型是强 *NP-S* 完全的。

**证明**。 证明基于从三维匹配 到 4-划分的归约，见 Garey 和 Johnson [GJ79]。（我们在 A 节证明NP-S完备性： 三维匹配 。）

该归约将一个三维匹配实例 $I$（$= \mathcal{I}, \mathcal{U}, \mathcal{S}$）转化为一个4-划分实例 $g(I) = (\mathcal{I}', \mathcal{U}', \mathcal{S}')$，其中所有元素的大小均由元素总数的一个多项式函数所界定。特别地，$\max\{a_i : i \in \{1, ..., 4n\}\} < 2^{16}\,|A|^4$。

令 $W = \{w_1, w_2, ..., w_q\}$、$X = \{x_1, x_2, ..., x_q\}$、$Y = \{y_1, y_2, ..., y_q\}$，并令 $M \subseteq W \times X \times Y$ 表示一个任意的三维匹配实例。相应的4-划分实例包含 $4|M|$ 个元素：对于 $W \cup X \cup Y$ 中的每个成员在 $M$ 的某个三元组中的每次出现，设置一个元素；此外，对于 $M$中的每个三元组，另设一个元素。

如果 $z \in W \cup X \cup Y$出现在 $M$ 的 $N(z)$ 个三元组中，那么 $A$ 中对应的元素记为 $z[1], z[2], \ldots, z[N(z)]$。元素 $z[1]$ 称为与 $z$ 对应的“实际”元素。元素 $z[2]$ 到 $z[N(z)]$ 称为与 $z$ 对应的 “ 虚拟 ” 元素。这些元素的大小取决于 $W, X, Y$ 中哪些包含 $z$，也取决于该元素是实际元素还是虚拟元素。下文使用的参数 $r$ 定义为 $32q$，并按如下方式选择 $A$ 中元素的大小：

$$
\begin{aligned}
s(w_i[1]) &= 10^4 i + r + 1, && 1 \le i \le q,\\
s(w_i[l]) &= 11r^4 + ir + 1, && 1 \le i \le q,\ 2 \le l \le N(w_i),\\
s(x_j[1]) &= 10^4 j^2 + 2, && 1 \le j \le q,\\
s(x_j[l]) &= 11r^4 + jr^2 + 2, && 1 \le j \le q,\ 2 \le l \le N(x_j),\\
s(y_k[1]) &= 10^4 + kr^3 + 4, && 1 \le k \le q,\\
s(y_k[l]) &= 8r^4 + kr^3 + 4, && 1 \le k \le q,\ 2 \le l \le N(y_k).
\end{aligned}
$$

如果 $m_\ell = (w_i, x_j, y_k) \in M$，那么 $m'_\ell \in A$ 的大小为 $s(m'_\ell) = 10^4 - kr^3 - jr^2 - ir + 8$。最后，$B = 40r^4 + 15$。

假设 $m'_\ell \in A$ 对应于 $m_\ell = (w_i,\ x_j,\ y_k) \in M$。Garey 和 Johnson 的证明表明，元素 $m'_\ell$ 属于一个总和为 $B$的子集 $T \subseteq A$ 只有两种方式。要么 $T = \{m'_\ell,\ w_i[1],\ x_j[1],\ y_k[1]\}$，要么 T 包含 $m'_\ell$ 以及分别对应于 $w_i,\ x_j$ 和 $y_k$的“虚拟”元素。

为说明该归约是解嵌入，对于每个 $\ell = 1$ 到 $q$，我们令 $f(m_\ell) = \langle m_\ell,\ w_i[1]\rangle$。如果 $M\prime$是一个三维匹配，那么 $f(M\prime)$ 可以扩展为一个 4-划分。同样，如果 $S' \subseteq \mathcal{U}'(g(I))$ 是一个可行的等价关系，那么 $f^{-1}(S' \cap f(\mathcal{U}(I)))$ 是一个三维匹配。◄

本小节最后讨论的问题是装箱问题。我们介绍相应的等价关系模型。

装箱问题（等价关系模型）
**实例 $\mathcal{I}$：** 集合 $A$；整数大小 $s(a)$，其中 $a \in A$ 为正整数箱子容量 $B$，以及一个整数 $k$。**全集 $\mathcal{U}$：** $\{(a,\ a')$：$a,\ a' \in A$ 且 $a \neq a'\}$。**解 $\mathcal{S}$：** 所有等价关系 $S\{(a,\ a')$：$a,\ a' \in A$ 且 $a \neq a'\}$的集合，其中对 $S$ 的每个等价类 $T$，$s(T) \leq B$，并且等价类最多有 $k$ 个。

►**定理 43**。 装箱问题 是强 *NP-S* 完全问题。

**证明**。 通过将四个部分分别解释为一个箱子，可以从 4-划分的等价关系模型得到一个直接的解嵌入归约。◄

## 7.2.2 由基集的子集组成的全集

在某些划分或着色问题中，一个部分（或具有相同颜色的元素）中的元素数量上限可能是常数。当可能的部分数量受到多项式界限约束时，我们可以令全集 $\mathcal{U}$ 由所有可能的部分组成。我们将这种表示类型称为子集生成。

子集生成是打破某些整数规划中固有对称性的另一种方法。我们以三角形覆盖和三角形划分为例。三角形覆盖的每个实例由一个图 $G = (V, E)$ 和一个整数 $k$ 组成。三角形 是 $V$ 中三个两两相邻的顶点所组成的集合。集合 $\mathcal{U}$ 由 $G$ 中的所有三角形组成。

解集$\mathcal{S}$ 是由任意一组三角形构成的集合，这些三角形包含 $V$ 的所有顶点，且三角形数量不超过 $k$ 。三角形划分的定义类似，区别在于其解集$\mathcal{S}$ 是由任意一组三角形构成的集合，这些三角形恰好各包含 $V$ 的所有顶点一次。

三角形覆盖（子集生成模型）实例 I：无向图 $G = (V, E)$，整数 k。全集 U：$\{\{w, v, x\}$：$\{w, v\}$，$\{w, x\}$，$\{v, x\} \in E\}$。解 S：所有满足 $\bigcup_{S_i \in S} S_i = V$ 且 $|S| \leq k$的 $S \subseteq$ U的集合。

三角形划分（子集生成模型）实例 I：无向图 $G = (V, E)$，以及 $n = |V|/3$。全集 U：$\{\{w, v, x\}: \{w, v\}, \{w, x\}, \{v, x\} \in E\}$。解 S：所有满足 $|S| = n/3$且 $\bigcup_{S_i \in S} S_i = V$的 $S \subseteq$ U的集合。

▶**定理 44**。 三角形覆盖 和三角形划分的子集生成模型都是 *NP-S* 完全的。

证明。三角形划分是三角形覆盖的特例。因此，只需证明三角形划分是 NP-S 完全的。令 Π 表示三角形划分的子集生成模型。我们将基于 Garey 和 Johnson[GJ79]中给出的从三维匹配到 Π的归约，证明 Π 是 NP-S 完全的。（实际上，他们的变换是从精确三集合覆盖（X3C）归约而来；不过，从三维匹配出发的变换几乎相同。）令 $W$、$X$和 $Y$ 表示三维匹配实例 $I$中的三个集合。这三个集合中的每一个都包含 $n$ 个互不相同的元素。令 $M$ 表示实例 $I$中 $m$ 个三元组（$w_i$、$x_j$、$y_k$）组成的集合。一个解由 $M$ 中的 $n$ 个三元组组成，这些三元组涵盖了 $W \cup X \cup Y$中的每个元素。

从三维匹配到 Π 的归约构造出一个实例 $I'$，它属于 Π；其中，Π 的实例 $I'$ 是一个图 $G = (V, E)$，且 $V$ 具有以下性质。$|V| = 3n + 9m$；$W \cup X \cup Y$ 中的每个元素在 $V$ 中对应一个顶点，而 $M$ 中的每个三元组在 |V| 中对应 9 个顶点。如果 $m_\ell = (w_i, x_j, y_k) \in M$，那么这 9 个额外顶点标记为 $a_\ell[1], \ldots, a_\ell[9]$。

该证明建立了匹配 $M' \subseteq M$ 与三角形划分之间的一一对应。此外，匹配 $M'$ 包含 $m_\ell = (w_i, x_j, y_k)$，当且仅当对应的三角形划分包含边 $(a_\ell[1], w_i)$。令 $f(m_\ell = (a_\ell[1], w_i)$，即可看出，从三维匹配到三角形划分的子集生成模型的归约具有解嵌入性质。◀

## 7.3 两步验证模型。

描述 NP 中问题的解（或证书）时，通常解都“明确规定”，验证任务也相当直接。对于哈密顿回路问题，解就是一个哈密顿回路。对于团问题，解就是团中的顶点。（边的集合也能明确地规定解。）对于装箱问题，解会说明如何将元素 $A$ 划分到各个箱子中。对于背包问题，解就是要放入背包的元素子集。

我们的最终策略是以一种解仅被部分指定的方式对集合 $\mathcal{S}$ 建模。我们将这种建模方法称为两步验证。该方法依赖于这样一种多项式时间算法：它能将部分指定的解 $S \in \mathcal{S}$ 转换为完全指定的解。我们以最小化迟交任务的排序问题为例进行说明。我们首先给出一个基于整数规划的模型。

**50 源自NP的PH完备性**

最小化迟交任务的排序（基于整数规划的模型） 实例 I：任务集合 $T$， $T$中元素上的偏序 $\prec$；对于每个任务 $t$，给定一个长度 $\ell(t)$、一个截止时间 $d(t)$，以及一个正整数 $k \leq T$。全集 U：$\{(t, j) : \text{对于 } t \in T \text{和 } j \in \{1, ..., n\}\}$，表示任务到开始时间的赋值。解 S：所有 $S \subseteq \{(t, j) : \text{对于 } t \in T \text{和 } j \in \{1, ..., n\}\}$ 的集合，其中任务被排程在 $\{0, ..., n\}$ 中彼此不同的时间开始，并且只要 $t \prec t'$， $t$就先于 $t'$ 开始；此外，完成时间晚于其截止时间的任务数量至多为 $k$。

基于整数规划的模型并非全集覆盖。尽管如此，我们猜想它不是 NP-S 完全。

通过利用以下观察，可以构造一个两步验证模型。对于最小化迟交任务的排序问题的一个实例 $I$，如果存在一组任务 $|T| - k$ 的子集 $S \subseteq T$，能够排程并满足所有截止时间，则该实例存在一个可行调度。可以按如下方式得到 $S$ 的一个可行调度（如果存在）。首先，调整 $S$ 中作业的截止时间，使得如果 $t$， $t' \in S$和 $t \prec t'$，那么 $d(t) \leq d(t') - 1$。修改截止时间后，只需按截止时间顺序排程作业。

以下是针对该问题的修订模型，它利用了上述观察。请注意，在下面的描述中，解由按时排程的作业组成。

最小化迟交任务的排序（两步验证模型）**实例 $\mathcal{I}$：** 任务 $T$ 的集合 $n$，对于每个任务 $t$，其长度为 $\ell(t)$，偏序为 $\prec$ on$T$，对于每个任务 $t \in T$，截止时间为 $d(t)$，以及一个整数 $k \leq n$。**全集 $\mathcal{U}$：** 任务 $T$ 的集合。**解 $\mathcal{S}$：** 所有 $S \subseteq T$ 的集合，使得可以将 $S$ 中的所有作业排程并满足其截止时间，且 $|S| \geq |T| - k$。

给定一个解 $S \in \mathcal{S}$，可以在多项式时间内构造出一个可行调度。我们将这一多项式时间构造视为验证过程中的“第二步”。

▶**定理 45**。 最小化迟交任务的排序的两步验证模型是 *NP-S* 完全的。

证明。该证明基于从团（基于边）到本问题的归约，见 Garey 和 Johnson [GJ79]。（我们在 A 节中证明团（基于边）的 NP-S 完全性。）考虑一个实例 $I$：确定是否存在一个大小为 $k$ 的边集 $S$，使其在图 $G = (V, E)$ 中定义一个团。也就是说，任意解 $S$ 都由一个具有 $j$ 个顶点的团的 $k = j(j-1)/2$ 条边组成。调度问题的变换后的实例 $g(I)$ 如下。对于每个顶点 $v \in V$，有一个截止时间为 $|V| + |E|$ 的任务 $v \in T$。我们用 $V'$ 表示这些任务组成的子集。对于每条边 $\{v, w\} \in E$，有一个截止时间为$k + j$的任务 $vw \in T$。我们用 $E'$ 表示这些任务组成的子集。此外， $v \prec vw$，并且 $w \prec vw$。因此，$T = V' \cup E'$，且 $|T| = n + m$。从边到任务的映射为： $f(\{v, w\}) = vw$。

注意，无论开始时间如何， $V'$ 中的所有任务都可以按时排程。 $E'$ 中至多有 $k$ 个任务可以按时排程。只有当 $f^{-1}(S')$ 是含有 $k$ 条边的 $E$ 的一个团时，才能将 $k$个任务组成的集合 $S' \subseteq E'$ 按时排程。因此，该归约是解嵌入归约。◀

## 8 结论

我们提出了一个通用框架，用于理解源自 NP 的多层优化问题的复杂性。通过引入*NP* 带解类（NP-S），并对解嵌入归约进行形式化，我们提炼出许多经典 NP 完全性归约中的一项结构性质；这一性质对于将困难性结果提升到 NP 之外至关重要。基于这一视角，我们得以定义 NP-S 完全性，并找出一大类符合这一更强完备性概念的自然问题，涵盖可满足性、图问题、背包问题和路由问题等。

这一框架的核心成果是一系列元定理，可系统地将 NP 完全性结果提升到多项式层次的更高层级以及 PSPACE。我们将这些元定理应用于若干重要的多层问题类别，包括对抗性选择博弈、阻断问题和保护-阻断问题，以及多阶段可调鲁棒优化。对于 NP-S 完全问题，我们证明，自然的两层变体（如对抗性选择和阻断）是 $\Sigma_2^p$ 完全的，而三层变体是 $\Sigma_3^p$ 完全的。更一般地，我们证明，这些问题的 $k$阶段扩展是 $\Sigma_k^p$ 完全的；如果允许阶段数随输入规模增长，则问题将达到 PSPACE 完全性。

这些结果涵盖了此前已知的几乎所有由 NP 衍生的多层优化问题的完全性结果，并同时得出了许多新的结果。不同于以往通常通过针对具体问题的归约来证明单个问题的困难性，我们的框架统一解释了整个多层问题族的复杂性。尤其是，对抗性选择、阻断和可调鲁棒优化，体现了同一潜在现象的紧密关联：解结构与序贯决策之间的相互作用。

除了给出新的完全性结果外，我们的工作还有助于解释为何 NP 完全问题的多层扩展往往比其单层对应问题难得多。对 NP-S 中解的显式处理阐明了，即使针对同一个底层 NP决策问题，解的不同表示也可能导致复杂度各异、根本不同的多层问题。这一观察凸显了解结构在复杂性理论中的重要性，并表明仅凭 NP困难性往往不足以理解多层优化问题的真正难度。

未来仍有若干研究方向。尽管 NP-S 涵盖了广泛的组合问题，但进一步完善这一框架，或找出能够刻画其他形式的多层优化或动态优化的替代抽象，都将颇有意义。此外，我们的结果主要立足于复杂性理论；如何运用这些困难性结果来指导对抗性选择、阻断和可调鲁棒优化中的算法设计、近似保证或参数化方法，仍是一个重要的开放方向。更广泛地说，解嵌入、交替与空间有界计算之间的紧密联系表明，类似的结构性技术或许也适用于其他问题从 NP 提升到更丰富计算模型的情形。

总体而言，我们的结果表明，高计算复杂性并非例外，而是 NP 完全问题的多层扩展所普遍具有的特征。我们将对抗性选择、阻断和多阶段可调鲁棒优化统一纳入一个复杂性理论框架，希望本研究能加深对多项式层次及其之外的优化问题的结构性理解。

## A NP-S 完全问题汇编

本节将介绍许多 NP-S 完全问题。对于每个问题，我们给出实例集合$\mathcal{I}$、全集$\mathcal{U}(I)$，以及与每个实例 $I \in \mathcal{I}$相关联的解集$\mathcal{S}(I)$的定义。此外，我们还陈述解嵌入归约并给出参考文献。对于某些问题，我们会在定义下方直接补充说明，描述对原始归约所做的修改。本节介绍和引用的所有归约都是已知的，且可在文献中找到。图 4 展示了本节所介绍的所有归约构成的树，其起点为可满足性（基于变量）。我们大量利用 SE 归约的传递性（引理 8）。

我们指出，网站$reductions.network$ [GP25] 收集了嵌入NP-S中问题归约的解，其中也包括Pfaue [Pfa24] 和Bartlett[Bar25]的结果。

根据图 4 所示的 SE 归约树，我们推导出以下定理。

▶**定理 46**。 以下问题均为 *NP-S* 完全：可满足性（基于文字）、可满足性（基于变量）、*3*-可满足性（基于文字）、*3*-可满足性（基于变量）、顶点覆盖、支配集、集合覆盖、击中集、反馈顶点集、反馈弧集、无容量设施选址、$p$-中心、$p$-中位数问题、三维匹配、独立集、团（基于顶点）、团（基于边）、子集和、背包、划分、调度问题、有向哈密顿路径、有向哈密顿回路、无向哈密顿回路、旅行商问题、两条有向顶点不相交路径、$k$-顶点有向不相交路径、斯坦纳树

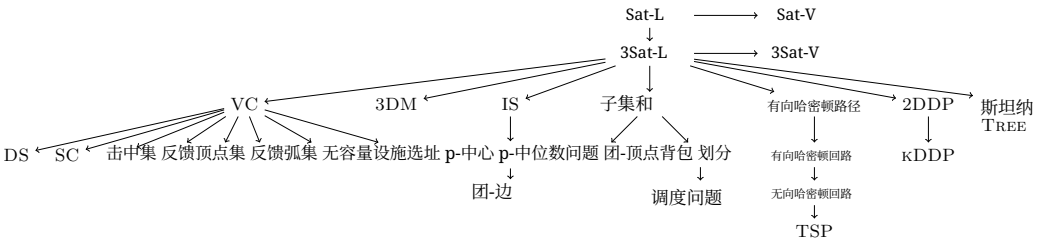


**图 4** 所有所考虑问题的 SSP 归约树。

可满足性（基于文字）实例 I：文字集 $L = \{\ell_1, \ldots, \ell_n\} \cup \{\bar{\ell}_1, \ldots, \bar{\ell}_n\}$，子句 $C \subseteq 2^L$。全集 U：文字集 $L$。解 S：所有满足以下条件的集合 $L' \subseteq L$：对于所有 $i \in \{1, \ldots, n\}$，均有 $|L' \cap \{\ell_i, \bar{\ell}_i\}| = 1$，且 $|L' \cap c_j| \geq 1$ 对于所有 $c_j \in C, j \in \{1, \ldots, |C|\}$。_

参考文献。Cook。 [Coo71]

可满足性（基于变量）**实例** $\mathcal{I}$**:** 变量 $X = \{x_1, \ldots, x_n\}$，子句 $C = \{C_1, \ldots, C_m\}$。**全集** $\mathcal{U}$**:** 变量集合 $X = \{x_1, \ldots, x_n\}$。**解** $\mathcal{S}$**:** $S \subseteq X$ 使得 $S$ 是一个满足 $C$所有子句的真值赋值，并且 $S$ 是 $X$ 中为真的变量集合。

参考文献。Cook。 [Coo71]

对于可满足性（基于文字）和可满足性（基于变量），Cook 关于可满足性 NP 完全性的结果也表明了它们的 NP-S 完全性。引理 38 给出了可满足性（基于文字）与可满足性（基于变量）之间双向归约的存在性。

3-可满足性（基于文字） 实例 I：文字集 $L = \{\ell_1, \ldots, \ell_n\} \cup \{\bar{\ell}_1, \ldots, \bar{\ell}_n\}$，子句 $C \subseteq L^3$。全集 U：文字集 $L$。解 S：所有满足以下条件的集合 $L' \subseteq$ U：对于所有 $i \in \{1, \ldots, n\}$，都有 $|L' \cap \{\ell_i, \bar{\ell}_i\}| = 1$，且 $|L' \cap c_j| \geq 1$ 对于所有 $c_j \in C, j \in \{1, \ldots, |C|\}$。

SEReduction：从可满足性（基于文字）归约。参考文献。
Karp， [Kar72]

与可满足性类似，3-可满足性（基于变量）有一个变体，称为 3-可满足性（基于文字），其全集由文字组成。与可满足性的引理 38 类似，3-可满足性（基于文字）和 3-可满足性（基于变量）可以相互归约。

3-可满足性（基于变量）**实例** $\mathcal{I}$**:** 变量 $X = \{x_1, \ldots, x_n\}$，子句 $C = \{C_1, \ldots, C_m\}$ 大小为 3。**全集** $\mathcal{U}$**:** 变量集 $X$。**解** $\mathcal{S}$**:** 与满足真值赋值对应的 $X$ 子集。

SEReduction：从可满足性（基于变量）归约。参考文献。
Karp， [Kar72]

顶点覆盖
**实例** $\mathcal{I}$**:** 图 $G = (V, E)$，数量 $k \in \mathbb{N}$。**全集** $\mathcal{U}$**:** 顶点集 $V$。**解** $\mathcal{S}$**:** 所有满足以下条件的 $S \subseteq V$ 集合：对所有 $e \in E$， $|S \cap e| \geq 1$ 和 $|S| \leq k$。

SE 归约。来自 3-可满足性（基于文字）。参考文献。
Garey 和 Johnson [GJ79]

支配集
**实例** $\mathcal{I}$**:** 图 $G = (V, E)$，数字 $k \in \mathbb{N}$.**全集** $\mathcal{U}$**:** 顶点集 $V$.**解** $\mathcal{S}$**:** 所有满足以下条件的 $S \subseteq V$ 的集合：对于所有 $v \in V$ 和 $|S| \leq k$， $|S \cap N[v]| \neq \emptyset$。

SEReduction：来自顶点覆盖。参考文献。
民间知识

在从顶点覆盖到支配集的众所周知的归约中，对于每条弧 $(v,w) \in E$，变换后的问题会额外增加一个顶点 $vw$ 和两条边 $(v,vw)$ 和 $(w,vw)$。任何顶点覆盖 $S \subseteq V$也都是变换后的图 $G' = (V\prime E')$ 的支配集。然而，这种归约并非解嵌入，因为可能存在一个支配集 $S' \subseteq V'$，满足 $|S'| \leq k$，且 $vw \in S'$。可以通过以下方式将该变换改为解嵌入：对于每个 $(u,v) \in E$，添加 $k+1$ 个顶点$vw_1, vw_2, \ldots vw_{k+1}$，它们都与 $v$ 和 $w$ 相邻。

集合覆盖
**实例 $\mathcal{I}$：** 集合 $S_i \subseteq \{1, \ldots, m\}$ 取自 $i \in \{1, \ldots, n\}$，数量为 $k \in \mathbb{N}$。**全集 $\mathcal{U}$：** $\{S_1, \ldots, S_n\}$。**解 $\mathcal{S}$：** 所有 $S \subseteq \{S_1, \ldots, S_n\}$ 的集合，使得 $\bigcup_{s \in S} s = \{1, \ldots, m\}$ 且$|S| \leq k$。

SE 归约：　　来自顶点覆盖。
参考文献。 Karp [Kar72]。

击中集
**实例 $\mathcal{I}$：** 集合 $S_j \subseteq \{1,\ldots,n\}$ 对应于 $j \in \{1,\ldots,m\}$，数量为 $k \in \mathbb{N}$。
**全集 $\mathcal{U}$：** $\{1,\ldots,n\}$。
**解 $\mathcal{S}$：** 所有 $H$的集合 $\subseteq \{1,\ldots,n\}$ 满足以下条件： $H \cap S_j \neq \emptyset$ 对于所有 $j \in \{1, \ldots, m\}$ 和 $|H| \leq k$。

SE 归约：　　来自顶点覆盖。
参考文献。 Karp [Kar72]。

反馈顶点集
**实例$\mathcal{I}$：** 有向图 $G = (V, A)$，编号 $k \in \mathbb{N}$。**全集$\mathcal{U}$：** 顶点集 $V$。**解$\mathcal{S}$：** 所有满足以下条件的 $S \subseteq V$的集合： $G[V \setminus S]$ 是无环的，并且 $|S| \leq k$。

SE 归约：　　来自顶点覆盖。
参考文献。 Karp [Kar72]。

反馈弧集
**实例 $\mathcal{I}$：** 有向图 $G = (V, A)$，编号 $k \in \mathbb{N}$。
**全集 $\mathcal{U}$：** 弧集 $A$。
**解 $\mathcal{S}$：** 所有满足以下条件的 $S \subseteq A$ 的集合：弧集 $G' = (V, A \setminus S)$ 是无环的以及 $|S| \leq k$。

SE 归约：　　从顶点覆盖归约。
参考文献。 Karp [Kar72]。

Karp [Kar72] 从顶点覆盖到反馈弧集的归约的一种修改是 SSP 归约。设 $I = (G, k) = ((V, E) k)$ 为顶点覆盖实例

以及 $(G', k') = ((V'A')\ k')$ 反馈弧集实例。首先，我们将顶点 $v \in V$转换为两个顶点 $v_0$, $v_1 \in V'$。我们通过将每个顶点 $v \in V$映射到弧 $(v_0, v_1) \in A'$来定义单射嵌入函数$f_I$，该弧也是所有解中的对应元素，即 $f_I(v) = (v_0, v_1)$。最后，我们将每条边 $\{v, w\} \in E$转换为从 $v_1$ 到 $w_0$ 的一次细分弧，以及从 $w_1$到 $v_0$的一次细分弧 $|V + 1|$。最终，我们保持参数 $k = k'$ 不变。通过删除弧 $(v_0, v_1)$（其对应于 $v$ 中的顶点），$G$ 中的所有这些诱导环都被断开。这意味着，$G$ 的每个顶点覆盖都通过函数 $f_I$转换为反馈弧集。

无容量设施选址（UFL）**实例 $\mathcal{I}$：** 候选设施集合 $F = \{1, \ldots, n\}$，客户集合 $C = \{1, \ldots, m\}$，设施启用固定成本函数 $f\colon F \to \mathbb{Z}$，服务成本函数$c\colon F \times C \to \mathbb{Z}$，成本阈值 $k \in \mathbb{Z}$。**全集 $\mathcal{U}$：** 候选设施 $F$。**解 $\mathcal{S}$：** 所有满足 $F\prime \subseteq F$ s.t. $\sum_{i \in F'} f(i) + \sum_{j \in C} \min_{i \in F'} c(i, j) \leq k$ 的集合。

SEReduction：来自顶点覆盖。参考文献。Cornuéjols、Nemhauser 和 Wolsey [CNW83]

p-中心
**实例 $\mathcal{I}$：** 候选设施集合 $F = \{1, \ldots, n\}$，客户集合 $C = \{1, \ldots, m\}$，服务成本函数 $c\colon F \times C \to \mathbb{Z}$，设施数量阈值 $p \in \mathbb{N}$，成本阈值 $k \in \mathbb{Z}$。**全集 $\mathcal{U}$：** 候选设施 $F$。**解 $\mathcal{S}$：** 所有满足 $F\prime \subseteq F$ s.t. $|F\prime| \leq p$且 $\max_{j \in C} \min_{i \in F'} c(i, j) \leq k$ 的集合。

SEReduction：来自顶点覆盖。参考文献。Cornuéjols、Nemhauser 和 Wolsey [CNW83]。

Cornuéjols、Nemhauser 和 Wolsey [CNW83]提出的从顶点覆盖到无容量设施选址的归约的一个修改版本，是一种子集和问题归约。令$I = ((V, E)\ k)$为顶点覆盖实例，$(F, C, c, p, k')$为 p-中心实例。我们将每个 $v \in V$ 映射到 $v \in F$，并将每个 $e \in E$ 映射到 $e \in C$。此外，我们定义 $c(v, e) = 0$，当$v \in e$时；否则，$c(v, e) = |V| + 1$。最后，我们令 $p$ 等于顶点覆盖的大小 $k$，并令 $k' = 0$。注意，这意味着在一个解中，目标函数必须为 0。

p-中位数问题
**实例 $\mathcal{I}$：** 候选设施集合 $F = \{1, \ldots, n\}$，客户集合 $C = \{1, \ldots, m\}$，服务成本函数 $c\colon F \times C \to \mathbb{Z}$，设施数量阈值 $p \in \mathbb{N}$，成本阈值 $k \in \mathbb{Z}$。**全集 $\mathcal{U}$：** 候选设施 $F$。**解 $\mathcal{S}$：** 所有 $F\prime \subseteq F$的集合，其中 $|F\prime| \leq p$且 $\sum_{j \in C} \min_{i \in F'} c(i, j) \leq k$。

SE 归约：来自顶点覆盖。参考文献：Cornuéjols、Nemhauser 和 Wolsey [CNW83]。

Cornuéjols、Nemhauser 和 Wolsey [CNW83]提出的从顶点覆盖到无容量设施选址的归约的一个修改版本，是一种子集和问题归约。它与 p-中心的情况相同。令 $I = ((V, E)\ k)$ 为顶点覆盖实例，并

$(F,\ C,\ c,\ p,\ k')$ 是 p-中位数问题实例。我们将每个 $v \in V$ 映射到 $v \in F$，并将每个 $e \in E$ 映射到$e \in C$。此外，如果满足条件，我们定义 $c(v,\ e)$ 为 $=0$；否则定义 $c(v,\ e)$ 为 $=|V|+1$。最后，我们令$p=k$ 等于顶点覆盖的大小，并令 $k'=0$。

三维匹配
**实例 $\mathcal{I}$:** 集合 $W,X,Y \in \mathbb{N}^q$，以及三元组集合 $M \subseteq W \times X \times Y$。
**全集 $\mathcal{U}$:** 三元组集合 $M$
**解$\mathcal{S}$:** 所有满足 $|S|=q$ 的 $S \subseteq M$ 的集合。

SEReduction。来自 3-可满足性（基于文字）。参考文献。
Garey 和 Johnson [GJ79]

独立集
**实例 $\mathcal{I}$:** 图 $G=(V,\ E)$，数量为 $k \in \mathbb{N}$。**全集 $\mathcal{U}$:** 顶点集 $V$。**解 $\mathcal{S}$:** 所有满足以下条件的 $S \in\subseteq V$ 的集合：对于所有 $i,\ j \in V\prime$，$\{i,\ j\} \notin E$，以及$|S| \geq k$。

SEReduction：来自 3-可满足性（基于文字）。参考文献。Garey 和 Johnson [GJ 79]（归约到顶点覆盖的修改版）

设 $I=(L,\ C)$ 为 3-SAT 实例。我们定义相应的顶点覆盖实例 $((V\prime E')\ k')$。每个文字 $\ell \in\subseteq L$ 都会被转换为一个顶点 $v_\ell \in V\prime$，每对 $(\ell,\ \ell) \in L \times L$ 也会再次被转换为一条边 $\{v_\ell,$ $^{v}\ell\} \in E'$，连接对应的文字顶点。每个子句 $c \in C$也会再次被转换为一个 3-团，其中每个顶点 $v^c_{\ell_{i_1}}$，$v^c_{\ell_{i_2}}$，$v^c_{\ell_{i_3}}$表示该子句中的一个文字。与顶点覆盖归约不同，子句顶点连接的是相反的文字顶点。最后，我们将参数 $k'$定义为 $k' := |L|/2+|C|$。3-可满足性（基于文字）的全集元素被单射映射到 $V\prime$ 中的文字顶点，其中 $f_I(\ell) = v_\ell \in V\prime$。

团（基于顶点）
**实例$\mathcal{I}$:** 图 $G=(V,\ E)$，数量为 $k \in \mathbb{N}$。**全集$\mathcal{U}$:** 顶点集 $V$。**解$\mathcal{S}$:** $S \subseteq V$，使得对于所有 $u,\ v \in V\prime$，$\{u,\ v\} \in E$且 $|S| \geq k$。

SEReduction：来自独立集。参考文献。
Garey 和 Johnson [GJ79]。

团（基于边）
**实例$\mathcal{I}$:** 图 $G=(V,\ E)$，数量为 $k \in \mathbb{N}$。**全集$\mathcal{U}$:** 边集 $E$。**解$\mathcal{S}$:** 所有 $S \subseteq E$ 的集合，使得对于所有点对 $(\{u,\ v\},\ \{u,\ w\}) \in S^2$，$\{v,\ w\} \in S$，且 $|S| \geq k$。

SEReduction：来自团（基于顶点）。

给定一个团（基于顶点）实例，其由图 $G=(V,\ E)$ 和整数 $k$ 组成，我们按如下方式构造一个团（基于边）实例 $G'=(V \cup V\prime E')$ 和整数 $k'=2k$。顶点子集 $V'$ 是 $V$ 的一个副本。对于每个顶点 $v \in V$，存在

顶点 $v$ 和 $v' \in V$，以及一条边 $(v, v') \in E'$。对于每条边 $(v, w) \in G$，都有边 $(v, w), (v', w)$, $(v, w')$ 和 $(v', w') \in E'$。很容易看出，$G$ 中大小为 $k$ 的团与 $G'$ 中大小为 $2k$的团之间存在一一对应关系。该归约是解嵌入：对于 $G$的每个顶点 $v$，$f(v) = (v, v') \in E'$。

子集和
**实例 $\mathcal{I}$：** 数 $\{a_1, \ldots, a_n\} \subseteq \mathbb{N}$，以及目标值 $M \in \mathbb{N}$。**全集 $\mathcal{U}$：** $\{a_1, \ldots, a_n\}$。**解 $\mathcal{S}$：** $S \subseteq \mathcal{U}$ 满足 $\sum_{i \in S} a_i = M$。

SEReduction：从 3-可满足性（基于文字）归约。Sipser [Sip97]

背包
实例 I：具有利润和重量的物品 $\{(p_1, w_1) \ldots, (w_n, p_n)\} \in \mathbb{N}^2$，界限 $W, P \in \mathbb{N}$。全集 U：这些物品 $\{(p_1, w_1) \ldots, (p_n, w_n)\}$。解 S：所有满足 $\sum_{(p_i, w_i) \in S} w_i \leq W$ 且 $\sum_{(p_i, w_i) \in S} p_i \geq P$ 的 $S \subseteq$ U 的集合。

SE 归约：来自子集和。
参考文献。民间知识。

对于归约，设 $p_i = w_i = a_i$ 和 $W = P = M$。

划分
**实例 $\mathcal{I}$：** 数 $\{a_1, \ldots, a_n\} \subseteq \mathbb{N}$。**全集 $\mathcal{U}$：** $\{a_1, \ldots, a_n\}$。**解 $\mathcal{S}$：** $S \subseteq \mathcal{U}$ 使得 $a_1 \in S$，且 $\sum_{i \in S} a_i = \sum_{i \notin S} a_i$。

SEReduction：来自子集和。参考文献。
民间知识。

注意，我们要求第一个元素属于解。若放宽这一要求，则不存在从 NP-S 完全问题到该问题的解嵌入归约。原因如下：对于每个子集 $S \subseteq U$ 使得 $\sum_{i \in S} a_i = b$，也有 $\sum_{i \notin S} a_i = b$ 为真，因此 $U \setminus S \in \mathcal{S}$。因此，若放宽 $1 \in S$这一要求，则不存在从 NP-S 完全问题到该问题的解嵌入归约。

设 $a_1, \ldots, a_n, M$为子集和实例。我们通过设定 $a'_1 = M + 1$、$a'_2 = \sum_i a_i + 1 - M$，并令 $a'_{i+2} = a_i$（对 $i \in \{1, \ldots, n\}$），定义划分实例$a'_1, \ldots, a'_{n+2}$。解嵌入函数 $f$ 定义为 $f(a_i) = a'_{i+2}$。

双机最大完工时间调度**实例 $\mathcal{I}$：** 处理时间 $\{t_1, \ldots, t_n\} \subseteq \mathbb{N}$，阈值 $T \in \mathbb{N}$。**全集 $\mathcal{U}$：** 处理时间 $\{t_1, \ldots, t_n\}$。**解 $\mathcal{S}$：** 所有满足 $t_1 \in S$、$\sum_{t_i \in S} t_i \leq T$ 和 $\sum_{t_i \notin S} t_i \leq T$的 $S \subseteq \{t_1, \ldots, t_n\}$ 的集合。SE 归约：来自划分。参考文献。民间知识。

这里，我们像在划分问题中那样打破两台机器之间的对称性。我们要求第一个任务由第一台机器处理。

对此，令 $I = \{a_1, \ldots, a_n\}$ 为划分实例。我们将划分实例中的每个数字 $a_i$转换为双机最大完工时间调度实例中处理时间为 $a_i$ 的一个作业，并设置阈值 $T = \frac{1}{2}\sum_i a_i$。

有向哈密顿路径**实例** $\mathcal{I}$**:** 有向图 $G = (V, A)$，顶点 $s, t \in V$。**全集** $\mathcal{U}$**:** 弧集 $A$。**解** $\mathcal{S}$**:** 所有满足 $S$ 是一条长度为 $|V| - 1$ 的 $s$-$t$ 路径的 $S \subseteq A$ 的集合。

SEReduction：来自 3-可满足性（基于文字）。参考文献。
AroraandBarak [AB09]。

有向哈密顿回路
**实例** $\mathcal{I}$**:** 有向图 $G = (V, A)$。**全集** $\mathcal{U}$**:** 弧集 $A$。**解** $\mathcal{S}$**:** 所有满足 $S$ 是一条长度为 $|V|$的回路的 $S \subseteq A$ 的集合。

SEReduction：来自有向哈密顿路径。参考文献。
AroraandBarak [AB09]。

无向哈密顿回路
**实例** $\mathcal{I}$**:** 图 $G = (V, E)$.**全集** $\mathcal{U}$**:** 边集 $E$.**解** $\mathcal{S}$**:** 所有满足 $S$ 是长度为 $|V|$的回路的 $S \subseteq E$ 的集合。

SEReduction：从有向哈密顿回路归约而来。参考文献。
Karp [Kar72]。

旅行商问题
**实例** $\mathcal{I}$**:** 完全图 $G = (V, E)$，函数 $w: E \to \mathbb{Z}$，数值 $t \in \mathbb{N}$。**全集** $\mathcal{U}$**:** 边集 $E$.
**解** $\mathcal{S}$**:** 所有满足 $S$ 是哈密顿回路且$w(S) \leq t$的 $S \subseteq E$ 的集合。

SE 归约: 无向哈密顿回路
参考文献。 民间知识。

令 $I = (V, E)$ 为无向哈密顿回路实例，$(V, E', w', k')$ 为旅行商问题实例。每个顶点 $v \in V$ 都映射到自身 $v \in V'$。此外，我们将每条边 $e \in E$ 映射到 $E'$ 中的自身，并添加额外的边以构成完全图。权重函数 $w': E' \to \mathbb{Z}$ 定义在所有 $e' \in E'$ 上，定义如下

$$w(e') = \begin{cases} 0, & \text{if } e' \in E \\ 1, & \text{if } e' \notin E \end{cases}$$

最后，我们设置 $k' = 0$，由此只有来自 $E$ 的边可用。因此，我们保留了与 $f_I(e) = e$ 的边之间的一一对应关系。

有向两点不相交路径
**实例 $\mathcal{I}$:** 有向图 $G=(V,\ A)$，$s_i,\ t_i \in V$，其中 $i \in \{1,\ 2\}$。
**全集 $\mathcal{U}$:** 弧集 A。
**解 $\mathcal{S}$:** 所有满足 $S$ 的 $S \subseteq A$ 的集合，是两条不相交路径的并集 $P_1$ 和 $P_2$，其中 $P_1$ 是一条从 $s_1$ 到 $t_1$ 的路径，而 $P_2$ 是一条从 $s_2$ 到 $t_2$ 的路径。

SE 归约: 来自 3-可满足性（基于文字）。
参考文献。 Fortune、Hopcroft 和 Wyllie [FHW80]。

有向 $k$-顶点不相交路径**实例 $\mathcal{I}$:** 有向图 $G=(V,\ A)$，$s_i,\ t_i \in V$，其中 $i \in \{1,\ \ldots,\ k\}$。**全集 $\mathcal{U}$:** 弧集 $A =: \mathcal{U}$。**解 $\mathcal{S}$:** 所有集合 $S \subseteq A$ 的集合，其中 $S = \bigcup_{i=1}^{k} A(P_i)$，且所有 $P_i$ 都是从 $s_i$ 到 $t_i$ 的两两顶点不相交路径，适用于 $1 \leq i \leq k$。

我们引入 $k-2$ 额外的顶点对 $s_i,\ t_i$ 用于 $i \in \{3, \ldots, k\}$，并通过添加弧 $(s_i,\ t_i)$ 将它们连接起来，其中所有 $i \in \{3, \ldots, k\}$ 均如此。

斯坦纳树
**实例 $\mathcal{I}$:** 图 $G=(S \cup T,\ E)$，斯坦纳顶点集合 $S$，终端顶点集合$T$，边权重 $c: E \to \mathbb{N}$，数量 $k \in \mathbb{N}$。**全集 $\mathcal{U}$:** 边集 $E$。**解 $\mathcal{S}$:** 所有满足以下条件的 $S \subseteq E$ 的集合: $S$ 是一棵连接 $T$ 中所有终端顶点的树，且 $\sum_{e' \in E'} c(e') \leq k$。

从 3-SAT 到斯坦纳树存在一种众所周知的归约，它是一种 SE 归约。首先，指定两个终端顶点 $s$ 和 $t$。对于每个文字 $\ell \in L$，都有一个斯坦纳顶点 $\ell$。此外，对于每个文字对 $(\ell_i,\ \bar{\ell}_i)$，$1 \leq i \leq |L|/2-1$，我们添加一个斯坦纳顶点$v_i$。我们定义 $v_0 := s$。然后，将上述所有顶点连接成一条“菱形链”: 首先令 $s$ 同时与 $\ell_1$ 和 $\bar{\ell}_1$ 相连。顶点 $\ell_1$ 和 $\bar{\ell}_1$ 都与 $v_1$ 相连。接着，顶点 $v_1$ 与顶点 $\ell_2$ 和 $\bar{\ell}_2$ 相连，以此类推。最后，$\ell_{|L|}$ 和 $\bar{\ell}_{|L|}$ 与$-t$ 相连。此外，对于每个子句 $c_j \in C$，我们添加一个对应的终端顶点 $c_j$。然后，顶点 $c_j$ 通过一条长度为 $|L|+1$ 的斯坦纳顶点路径与其对应的文字 $\ell \in c_j$ 相连。每条边的成本均设为 1，阈值设为 $k = |L| + |C| \cdot (|L|+1)$。我们为所有 $\ell \in L$ 定义解嵌入函数 $f_I$，使得$f_I(\ell) = \{v_{i-1}, \ell\}$。

## 60 源自NP的PH完备性

参考文献 AB09 Sanjeev Arora 和 Boaz Barak。《计算复杂性：现代方法》。剑桥大学出版社，2009年。Bar25 Celina Janet Bartlett. 关于具有节约性质的子集搜索问题及其归约的汇编。CoRR，abs/2506.12255，2025年。BBPR15 Cristina Bazgan、Cédric Bentz、Christophe Picouleau 和 Bernard Ries. 关于稳定数和色数的阻断者。《图与组合》，31(1):73–90，2015年。BGL+25 Jakub Balabán、Matthias Gehnen、Henri Lotze、Finn Seesemann 和 Moritz Stocker. 带估计的在线背包问题。CoRR，abs/2504.21750，2025年。BGN09 Aharon Ben-Tal、Laurent El Ghaoui 和 Arkadi Nemirovski.《鲁棒优化》，普林斯顿应用数学丛书第28卷。普林斯顿大学出版社，2009年。BNKS98 Amotz Bar-Noy、Samir Khuller 和 Baruch Schieber. 寻找最关键弧和节点的计算复杂性。1998年。BTT10 Cristina Bazgan、Sonia Toubaline 和 Zsolt Tuza. 与树结构相关图中独立集的最关键节点问题的计算复杂性。载于 Costas S. Iliopoulos 和 William F. Smyth 编，《组合算法——第21届国际研讨会 IWOCA 2010》，英国伦敦，2010年7月26—28日，修订精选论文，计算机科学讲义第6460卷，第154–166页。施普林格，2010年。BTT11 Cristina Bazgan、Sonia Toubaline 和 Zsolt Tuza. 关于独立集和顶点覆盖的最关键节点。《离散应用数学》，159(17):1933–1946，2011年。BTV10 Cristina Bazgan、Sonia Toubaline 和 Daniel Vanderpooten. 确定 1-中位数和 1-中心选址问题中最关键元素的计算复杂性。载于 Weili Wu 和 Ovidiu Daescu 编，《组合优化与应用——第4届国际会议 COCOA 2010》，美国夏威夷州凯卢阿-科纳，2010年12月18—20日，会议录，第I部分，计算机科学讲义第6508卷，第237–251页。施普林格，2010年。BTV13 Cristina Bazgan、Sonia Toubaline 和 Daniel Vanderpooten. 确定 p-中位数问题和 p-中心选址问题中最关键元素的计算复杂性。《组合优化杂志》，25(2):191–207，2013年。CCLW14 Alberto Caprara、Margarida Carvalho、Andrea Lodi 和 Gerhard J. Woeginger. 双层背包问题计算复杂性的研究。《工业与应用数学学会优化杂志》，24(2):823–838，2014年。CGKZ18 André B. Chassein、Marc Goerigk、Adam Kasperski 和 Pawel Zielinski. 关于具有预算不确定性的可恢复选择问题和两阶段鲁棒选择问题。《欧洲运筹学杂志》，265(2):423–436，2018年。CNW83 Gérard Cornuéjols、George Nemhauser 和 Laurence Wolsey. 无容量限制的设施选址问题。技术报告，康奈尔大学运筹学与工业工程系，1983年。Coo71 Stephen A. Cook. 定理证明过程的计算复杂性。载于 Michael A. Harrison、Ranan B. Banerji 和 Jeffrey D. Ullman 编，《第3届年度 ACM计算理论研讨会论文集》，1971年5月3—5日，美国俄亥俄州 Shaker Heights，第151–158页。ACM，1971年。Den11 Scott Denegre. 阻断与离散双层线性规划。博士论文，美国，2011年。AAI3456385。DZ20 Stephan Dempe 和 Alain Zemkoho. 双层优化。载于《施普林格优化及其应用》，第161卷。施普林格，2020年。FHW80 Steven Fortune、John E. Hopcroft 和 James Wyllie. 有向子图同胚问题。《理论计算机科学》，10:111–121，1980年。FLMS19 Fabio Furini、Ivana Ljubic、Sébastien Martin 和 Pablo San Segundo. 最大团阻断问题。《欧洲运筹学杂志》，277(1):112–127，2019年。

FR21 Nicolas Fröhlich 和 Stefan Ruzika。覆盖-阻断问题的困难性。《理论计算机科学》，871：1–15，2021年。GGI+15 Marc Goerigk、Manoj Gupta、Jonas Ide、Anita Schöbel 和 Sandeep Sen。带查询的鲁棒背包问题。《计算机与运筹学》，55：12–22，2015年。GH24 Marc Goerigk 和 Michael Hartisch。鲁棒组合优化导论，运筹学与管理科学国际丛书第361卷。施普林格，2024年。GJ79 M. R. Garey 和 David S. Johnson。《计算机与难处理性：NP完全性理论指南》。W. H. Freeman，1979年。GKN25 Matthias Gehnen、Ralf Klasing 和 Émile Naquin。利用边权重估计进行图探索。CoRR，abs/2501.18496，2025年。GLR24 Matthias Gehnen、Henri Lotze 和 Peter Rossmanith。带有有界预测的在线简单背包。载于 Olaf Beyersdorff、Mamadou Moustapha Kanté、Orna Kupfer- man 和 Daniel Lokshtanov 编，第41届计算机科学理论方面国际研讨会（STACS 2024），2024年3月12日至14日，法国克莱蒙费朗，LIPIcs第289卷，论文37:1–37:20。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2024年。GLW22 Marc Goerigk、Stefan Lendl 和 Lasse Wulf。具有两阶段不确定性的两阶段鲁棒优化问题。《欧洲运筹学杂志》，302(1)：62–78，2022年。GLW24 Marc Goerigk、Stefan Lendl 和 Lasse Wulf。具有离散补救的鲁棒多阶段问题的复杂性。《离散应用数学》，343：355–370，2024年。GP25 Christoph Grüne 和 Femke Pfaue。归约汇编：reductions.network。CoRR，abs/2511.04308，2025年。Grü24 Christoph Grüne。汉明距离可恢复鲁棒问题的复杂度类。载于 José A. Soto 和 Andreas Wiese 编，LATIN 2024：理论信息学——第16届拉丁美洲研讨会，智利瓦拉斯港，2024年3月18日至22日，论文集第一部分，计算机科学讲义第14578卷，321–335页。施普林格，2024年。GU25 Matthias Gehnen 和 Andreas Usdenski。利用物品尺寸估计进行在线装箱。CoRR，abs/2505.09321，2025年。GW24 Christoph Grüne 和 Lasse Wulf。双层优化和鲁棒优化中许多自然问题在多项式层次中的完备性，2024年。HLW23 Hung P. Hoang、Stefan Lendl 和 Lasse Wulf。区间图上的辅助与阻断问题。《离散应用数学》，340：153–170，2023年。HZ23 Kyle Hunt 和 Jun Zhuang。攻击者—防御者博弈综述：现状与未来方向。《欧洲运筹学杂志》，2023年。Jer85 Robert G. Jeroslow。多项式层次与竞争分析的一个简单模型。《数学规划》，32(2)：146–164，1985年。Joh11 Berit Johannes。多项式层次第二层的完全问题新类，2011年。Kar72 Richard M. Karp。组合问题之间的可约性。载于 Raymond E. Miller 和 James W. Thatcher 编，1972年3月20日至22日在纽约州约克敦高地 IBM Thomas J. Watson 研究中心举行的计算复杂性研讨会论文集，IBM研究研讨会丛书，85–103页。Plenum出版社，纽约，1972年。KBB+08 Leonid Khachiyan、Endre Boros、Konrad Borys、Khaled M. Elbassioni、Vladimir Gurvich、Gábor Rudolf 和 Jihui Zhao。短路径阻断问题：总量限制与逐节点限制的阻断。《理论计算机系统》，43(2)：204–233，2008年。KLLS21 Thomas Kleinert、Martine Labbé、Ivana Ljubic 和 Martin Schmidt。双层优化中的混合整数规划技术综述。《EURO计算优化杂志》，9：100007，2021年。

## 62 源自NP的PH完备性

KZ15 Adam Kasperski 和 Pawel Zielinski。鲁棒两阶段网络问题。载于 Karl F. Doerner、Ivana Ljubic、Georg Pflug 和 Gernot Tragler 编，2015年运筹学会议论文集：德国、奥地利和瑞士运筹学学会国际会议（GOR、ÖGOR、SVOR/ASRO）精选论文，维也纳大学，奥地利，2015年9月1日至4日，《运筹学会议论文集》，35–40页。施普林格，2015年。KZ16 Adam Kasperski 和 Paweł Zieliński。离散与区间不确定性下的鲁棒离散优化：综述。《决策辅助、优化与分析中的鲁棒性分析》，113–143页，2016年。KZ17 Adam Kasperski 和 Pawel Zielinski。鲁棒可恢复选择问题与两阶段选择问题。《离散应用数学》，233：52–64，2017年。LC93 Kao-Chêng Lin 和 Maw-Sheng Chern。最小生成树问题中的最关键边。《信息处理快报》，45(1)：25–31，1993年。LS17 Tao Li 和 Suresh P Sethi。动态斯塔克尔伯格博弈模型综述。《离散与连续动力系统-B》，22(1)：125，2017年。MMG89 Kavindra Malik、Ashok K Mittal 和 Santosh K Gupta。最短路径问题中的k条最关键弧。《运筹学快报》，8(4)：223–227，1989年。NCH22 Adel Nabli、Margarida Carvalho 和 Pierre Hosteins。多层关键节点问题的复杂性。《计算机与系统科学杂志》，127：122–145，2022年。Paj20 Foad Mahdavi Pajouh。最小成本边阻断团问题。《运筹学年刊》，294(1)：345–376，2020年。Pap94 Christos H. Papadimitriou。《计算复杂性》。Addison-Wesley，1994年。PBP14 Foad Mahdavi Pajouh、Vladimir Boginski 和 Eduardo L. Pasiliao。最小顶点阻断团问题。《网络》，64(1)：48–64，2014年。Pfa24 Femke Pfaue。探索ssp-NP完全问题之间的归约并开发展示结果的汇编网站。CoRR，abs/2411.05796，2024年。PPR16 Daniël Paulusma、Christophe Picouleau 和 Bernard Ries。通过边收缩和顶点删除降低团和色数。载于 Raffaele Cerulli、Satoru Fujishige 和 Ali Ridha Mahjoub 编，组合优化——第四届国际研讨会（ISCO 2016），意大利维耶特里苏尔马雷，2016年5月16日至18日，修订版精选论文，计算机科学讲义第9849卷，38–49页。施普林格，2016年。PPR17 Daniël Paulusma、Christophe Picouleau 和 Bernard Ries。通过边收缩和顶点删除阻断H-free图中的独立集。载于 T. V. Gopal、Gerhard Jäger 和 Silvia Steila 编，计算模型理论与应用——第十四届年度会议（TAMC 2017），瑞士伯尔尼，2017年4月20日至22日，论文集，计算机科学讲义第10185卷，470–483页，2017年。PWBP15 Foad Mahdavi Pajouh、Jose L. Walteros、Vladimir Boginski 和 Eduardo L. Pasiliao。最小边阻断支配集问题。《欧洲运筹学杂志》，247(1)：16–26，2015年。Rut93 Vladislav Rutenburg。命题真值维护系统：分类与复杂性分析。《数学与人工智能年刊》，10(3)：207–232，1993年。Sip97 Michael Sipser。《计算理论导论》。PWS Publishing Company，1997年。SPG13 J Cole Smith、Mike Prince 和 Joseph Geunes。现代网络阻断问题与算法。载于《组合优化手册》，1949–1987页。Springer New York，2013年。Sto76 Larry J. Stockmeyer。多项式时间层次。《理论计算机科学》，3(1)：1–22，1976年。SU08 Marcus Schaefer 和 Christopher Umans。多项式时间层次中的完备性：一份汇编，2008年。

TCCH24 Alberto Boggio Tomasaz、Margarida Carvalho、Roberto Cordone和Pierre Hosteins。论多项式层次中若干加固—阻断博弈的完备性。 $CoRR$，abs/2406.01756，2024年。
Val79 Leslie G Valiant。计算永久式的复杂性。 理论计算机科学，8(2):189–201，1979年。
Woe21 Gerhard J. Woeginger。第二个量词带来的麻烦。 $4OR$，19(2):157–181，2021年。
Woo93 R.Kevin Wood。确定性网络阻断。 《数学与计算机建模》，17(2):1–18，1993年。
Wra76 Celia Wrathall。完全集与多项式时间层次。 理论计算机科学，3(1):23–33，1976年。
YGdH19 İhsan Yanıkoğlu、Bram L Gorissen和Dick den Hertog。可调鲁棒优化综述。 《欧洲运筹学杂志》，277(3):799–813，2019年。Zen10 Rico Zenklusen。匹配阻断。 《离散应用数学》，158(15):1676–1690，2010年。