罗马命中函数

Henning Fernau[a,1]，Samuel German[b]，Kevin Mann[a]

[a]第4系—计算机科学部，特里尔大学，德国特里尔 54286。[b]计算机科学与工程系，加利福尼亚大学圣迭戈分校，Gilman Drive 9500号，邮件编码 0404，拉霍亚，CA 92093-0404。

arXiv:2611.32313v1 [cs.CC] 24 Nov 2026

摘要

罗马支配将一种可追溯至君士坦丁大帝时期的军事策略形式化。在这种策略中，军队驻扎在不同区域。如果某个区域至少驻扎了一支军队，或其某个邻近区域驻扎了两支军队，那么该区域便受到保护。这种简单的策略可以轻易转化为一个图论问题。军队的部署由一个函数描述，该函数将每个顶点映射为 0、1 或 2。若每个取值为 0 的顶点都有一个取值为 2 的邻点，则称该函数为罗马支配函数。

罗马支配是少数几个相关的（所谓）扩展问题即使在原判定问题为NP完全问题时仍可在多项式时间内求解的例子之一。这一点很有意思，因为它使得我们能够建立多项式延迟枚举算法，用于列出所有极小罗马支配函数；而对于一个图的所有极小支配集，或（等价地）一个超图的所有极小击中集是否能够以多项式延迟枚举，甚至能否在输出多项式时间内枚举，四十多年来一直悬而未决。为了找出其中的原因，我们将击中集的思想与罗马支配的思想结合起来。由此，我们提出并研究一个名为罗马命中函数的新问题，将罗马支配推广至超图。这样，我们便能界定多项式延迟可枚举性的边界。

我们的主要关注点是该问题的扩展版本，因为在处理罗马支配函数时，这正是关键问题。在研究过程中，我们找到了使扩展罗马击中函数问题成为NP完全问题的一些条件。随后，我们将参数化复杂性作为工具，以更好地理解扩展罗马击中函数为何会表现出这种特性。换个角度来说，我们运用参数化的思想来研究：究竟是什么让某些枚举问题如此困难。

此外，我们还讨论了扩展罗马支配的另一种推广：在这种推广中，给定了所求极小罗马支配函数的下界和上界。额外的上界使该问题变得困难

（同样如此），而所采用的参数化复杂性分析仅能给出困难性结果。

此外，从参数化复杂性的角度来看，对扩展问题的研究也颇为有趣，因为它们不断提供新的例子，表明参数化问题可以是W[3], 完全的；而在几年前，这一复杂性类已知的自然成员还寥寥无几。

我们运用所得结论，为罗马击中函数提出多项式延迟和固定参数可处理延迟枚举算法，同时划定多项式延迟与横截硬度之间的界限。

关键词：枚举问题、多项式延迟、支配问题、击中集、罗马支配

1. 引言

四十多年来，所有极小击中集是否都能以多项式延迟进行枚举，一直是一个悬而未决的问题。这个横截超图问题等价于这样一个问题：所有极小支配集是否都能以多项式延迟进行枚举。[2] 从应用角度来看，找到肯定的答案十分重要：没有用户愿意“永远”等待下一个解，也不愿意等到最后才知道不存在更多解。为了探索这一问题，人们已经在许多特殊图类中研究了极小支配集的枚举问题。目前已经确定，在若干图类中，这一枚举问题可以通过多项式延迟解决。本文从另一个角度研究这一问题。具体而言，文献 [4]表明，所有极小罗马支配函数[3] 都可以通过多项式延迟进行枚举。这一发现颇为出人意料，因为在大多数其他复杂性方面，罗马支配与支配集的表现都非常相似。更具体地说，列举其中一些结果如下：

- 罗马支配是NP完全的，即使在特殊图类中也是如此；参见 [25, 46]。
- 最小罗马支配可以在对数因子范围内近似，但除非P = NP，否则无法获得更好的近似；参见 [1, 46, 45]。
- 在标准参数化下（以军队数量的上界k 为参数），罗马支配是W[2]完全的；参见 [27]。然而，在对偶参数化下，罗马支配属于FPT；参见 [2, 8, 9]。

因此，我们以此为基础，尝试将罗马支配推广为一种击中集变体，以了解多项式延迟这一性质何时或在何处消失。顺带一提，我们也大幅推广了 [4]中的枚举结果。正如下文将要说明的，我们将罗马支配推广到超图；我们可以相当准确地描述其可处理性边界（就多项式延迟而言）。这也有助于理解为何在枚举方面，罗马支配变体与经典情形表现得如此不同。

我们先来说明，罗马支配的各种变体为何本身就很有趣。几个世纪以来，人们一直从历史军事的角度研究罗马帝国的防御策略，后来也开始从更偏图论的视角进行研究，这一视角始于 Arquilla[4] 和 Fredricksen [7]的一篇文章。本文同样讨论的这种策略如今称为罗马支配，其提出见于 [48]

[1]通讯作者。电子邮箱：fernau@uni-trier.de [2]这种枚举是否能在输出多项式时间内完成，也仍是一个开放问题。 [3]这些技术概念将在下文定义。[4]Arquilla 因其对网络战的分析而颇负盛名，参见 [6]。

这与 Arquilla 和 Fredricksen 的讨论一脉相承，将一种可追溯至君士坦丁大帝时期的军事策略形式化。罗马支配是图上支配的一种研究充分的变体，近年出版的一本教材 [35], 中甚至为其专设了长达 45 页的一章，该书由 Chellali 等人撰写。 在此模型中，军队驻扎在不同区域。若某个区域驻有至少一支军队，或其某个邻区驻有两支军队，则该区域受到保护。数学上，我们将地图建模为一个图，其中区域对应顶点，区域之间的连接对应边；此外，军队的部署由一个函数描述，该函数将每个顶点映射到 0、1 或 2。若每个取值为 0 的顶点都有一个取值为 2 的邻点，则称该函数为罗马支配函数。这一务实思路是：投入一支军队即可保护一个区域，而投入两支军队还能保护其所有邻区。目标是保护整个帝国。显然，我们希望找到所需军队数量最少的罗马支配函数，以尽量降低开支；相应的判定问题称为罗马支配问题。

除了已取得的许多罗马支配的组合（图论）结果（ [35], 对此有很好的综述）之外，判定问题 Roman Domination 也已从多个方面得到研究。尽管在各种情形下，Roman Domination 与 Dominating Set 的复杂度表现相同，但这种平行关系在两个（相互关联的）任务上却出乎意料地不再成立：

- 我们能否以多项式延迟枚举（即无重复地列出）给定实例的所有极小解?
- 给定解的某一部分，我们能否判定是否存在一个扩展给定预解的极小解?

第一类问题也称为输出敏感枚举问题。即使是要求较低的任务——以输出多项式时间枚举所有极小支配集——目前也尚未解决。 [5] 正如 [4] 所证明并在 [3]中应用的那样，罗马支配函数的相应枚举问题（针对两种自然的极小性概念）可以通过多项式延迟解决。关于罗马支配的这一结果基于另一项结果：该结果为第二项中描述的扩展问题给出了多项式时间算法。具体而言，其思路是在进一步进行分支之前先调用扩展测试。这一策略在枚举算法领域已广为采用，其历史可追溯至 Read 和 Tarjan [47], ; 然而，已知的具体实例寥寥无几，我们仅请读者参阅 [43, 49]中的讨论。因此，扩展罗马支配是少数几个扩展问题可在多项式时间内求解、而原问题却是NP完全的问题之一。关于扩展问题的更多详情，请参阅综述 [15] ; 该综述还提出了描述此类问题的一般框架，并进一步阐述了相关动机与应用。

本文想要探讨的简单科学问题是“为什么”：是什么使罗马支配在枚举和扩展方面具有可处理性？为了弄清扩展罗马支配（简称 Ext RD）为何表现得如此特别，以及它与扩展支配集（简称 Ext DS

[5]关于本讨论中仅随意使用的一些输出敏感枚举基本概念的正式定义，请参阅 [33, 36, 49]。

）有何不同，我们将推广罗马支配的概念，并尝试寻找可处理性边界。具体来说，我们要探讨的是，能否以多项式延迟枚举极小击中集。这个所谓的横截超图问题 [26] 已经开放了四十年。它相当重要，因为它出现在许多应用领域；尤其在数据库 [20, 26, 32, 38], 中，存在不少有趣的等价问题，或已被证明是横截困难的问题，正如 [33]中所称；我们仅举出两篇近期文献，并请读者参阅其中引用的论文： [11, 12]。本文试图理解哪类问题是横截困难的，并进一步描述何种情况下存在多项式延迟枚举算法。以往关于极小支配集枚举算法的研究，往往试图考察能否在特殊图类上实现多项式延迟枚举。例如，对于分裂图， [38],证明了极小支配集可以用多项式延迟枚举，[6] 而在假设横截困难的前提下，极小连通支配集则无法如此枚举，见 [37]。这种方法可以看作是通过研究特殊图类，对一个已知的（横截困难）枚举问题进行特化。我们的方法则有所不同：我们从一个支配类型问题出发，该问题在一般图上已有已知的多项式延迟枚举算法；我们试图通过推广来拓展这一结果，以了解这项枚举任务何时会变得横截困难。

众所周知，击中集（简称 HS）可以通过用闭邻域超图对图进行建模，视为支配集（简称 DS）的一种推广。图中的支配集与击中集之间的主要区别之一在于，在后者中，可以进行支配的对象（超图的顶点）与应被支配的对象（即超边，也就是顶点集）之间存在明确区分。尽管击中集和罗马支配都是支配集的既有推广概念，但据我们所知，文献中似乎尚未发表将这两个概念结合起来的研究。实际上，尝试定义这种结合会带来一些问题。如果希望一个击中集实例通过闭邻域超图表示一个支配集实例，那么给定图的顶点集就是超图的顶点集，而所有闭邻域构成（超）边集。忽略孪生顶点后，这意味着论域与（超）边集之间存在双射。但对于一般超图而言，超边的数量与论域中元素的数量是相互独立的。因此，我们必须思考如何解释“值为一的设置”，使其与罗马支配的定义相关联；在该定义中，恰好有一支军队被部署在某个顶点上。我们建议基于以下思想，在超图中对这一效果进行建模：如果某个顶点在罗马支配函数下的值为 1，那么它只击中“属于自身的”闭邻域超边。对于一般超图，我们必须明确表达一个顶点“拥有”哪条超边。因此，我们需要一个称为 对应关系 的函数，将一个顶点映射到一条与其关联的超边，使得当该顶点的值为 1 时，这条超边即被支配。下一节将对这个超图问题作出更形式化的定义。

最后，我们提一下本文存在一些不同版本。首先，本文是将技术报告 [28] 拆分后形成的。此外，技术报告还包含了 [42] 的一些结果。相反，

[6]Kanté等人用直接论证证明了这一多项式延迟枚举结果。还可以证明，在分裂图中，极小支配集的扩展问题可在多项式时间内求解。

本文还增加了一项 W[3]-完全性结果（定理12 在 [28] 中是一个 W[1]-困难性结果），以及更多输出敏感枚举分析。这包括横截完全性和参数化枚举算法。本文的扩展摘要已在 IPEC 2024 上发表，见 [29]。此版本不仅包含完整证明，还新增了一个专门章节，详细阐述此前在 IPEC 上未受到特别关注的可枚举性结果。此外，在会议版本中，我们提出了以下问题：我们也不知道，所讨论的多项式延迟可枚举性问题是否真的等价于极小击中集的多项式延迟可枚举性。此后，Samuel German 已解决了这个问题，我们也将他的论证（以及他的名字）加入了期刊版本。

2. 定义与记号

本文将直接使用复杂性理论中的标准概念，而不在此处加以定义。这包括参数化复杂性中的概念，涉及固定参数可处理 以及W-层级中更低的层次，直至W[3], ，相关内容可参见 [30, 24] 等教材。

2.1. 一般概念、图与超图

一般概念。令 $\mathbb{N}$表示所有非负整数（包括 0）组成的集合。对于 $n \in \mathbb{N}$，我们使用记号$[n] := \{1, \ldots\ldots, n\}$。对于有限集 A 以及某个满足 $n \in \mathbb{N}$ 且 $n \leq |A|$的 n，A 的基数记为 $\binom{A}{n}$，表示 A 中基数为 n 的所有子集构成的集合；而 2^A 表示 A 的幂集。对于两个集合 A、B，B^A 表示所有映射 $f : A \to B$ 构成的集合。若 $C \subseteq A$，则 $f(C) = \{f(x) \mid x \in C\} \subseteq B$。我们用 $\chi_C \in \{0,1\}^A$ 表示特征函数，其中$\chi_C(x) = 1$ 成立，当且仅当 $x \in C$。对于两个函数 f、$g \in \mathbb{N}^A$，当且仅当对所有 $a \in A$，$f(a) \leq g(a)$ 成立时，我们记作 $f \leq g$。此外，定义 f 的权重为 $\omega(f) = \sum_{a \in A} f(a)$。

超图。我们关注不一定是简单超图的超图 $H = \left(X, \hat{S} = (s_i)_{i \in I}\right)$，其具有一个有限论域 X（也称为顶点集）和一个有限索引集 I，其中对于每个 $i \in I$，$s_i \subseteq X$ 是一条超边[7]。我们用集合 S 表示包含族 $\hat{S}$ 中所有超边的集合，即 $S = \{s_i \mid i \in I\}$。注意，同一条超边可能在族 $\hat{S}$ 中出现多次。如果禁止这种情况，我们称其为简单超图。[8] 对于所有 $x \in X$，定义 $S(x) = \{s_i \in S \mid x \in s_i\}$，表示被顶点 x 击中的超边集合，从而定义函数 $S : X \to 2^S$。集合 $D \subseteq X$ 是击中集（简称 hs），当且仅当 $S(D) = S$，其中 $S(D) = \bigcup_{x \in D} S(x)$。类似地，定义 $I : X \to 2^I$，其中 $I(x) = \{i \in I \mid x \in s_i\}$，并将其扩展到 $A \subseteq X$，令 $I(A) = \bigcup_{x \in A} I(x)$。我们称函数 $\tau : X \to I$ 为对应关系，如果对于所有 $x \in X$ 都有 $x \in s_{\tau(x)}$，或者换言之，$\tau(x) \in I(x)$ 对所有 $x \in X$ 都成立。

[7]在证明中，我们发现这种指标记法比可能更常见的多重集记法更方便。[8]在这里，我们遵循简单图的类比，即不允许出现重边。不过，此前“简单超图”这一概念有着不同的含义。Ouvrard [44] 要求更强的条件：任意两条超边都不能相互包含。他建议将我们所说的简单超图称为没有重复超边的超图。Frieze 和 Mubayi [31] 则允许简单超图中的两条边至多共享一个顶点。

考虑一个（简单无向）图 G = (V, E)，将其视为超图 G = (V, ˆE)，其中每条超边恰好包含两个元素。现在，我们将这些超边称为边。在讨论简单图时，可以将 E 视为索引集。对于每个顶点 v ∈ V，定义其邻域为 N(v) = {u | {v, u} ∈ E}，闭邻域为 N[v] = {v} ∪ N(v)。对于顶点集 U ⊆ V，我们用 N[U] = ⋃v∈U N[v] 表示 U 的闭邻域。图 G = (V, E) 的支配集（简称 ds）是一个集合 D ⊆ V，使得 N[D] = V。

下面，我们将正式介绍有关罗马支配和罗马命中函数的更具体概念。

2.2. 罗马支配函数

设$G = (V, E)$是一个图。函数$f\colon V \to \{0, 1, 2\}$是罗马支配函数（简称 Rdf），当且仅当对于每个满足$f(v) = 0$的顶点$v \in V$，都存在一个$u \in N(v)$，使得$f(u) = 2$。若对于每个满足$g \leq f$的罗马支配函数g，都有$f = g$成立，则罗马支配函数f是极小 的，由此引出以下问题。

问题名称： 罗马支配（RD）**给定：** 一个图$G = (V,E)$和$k \in \mathbb{N}$**问题：** 是否存在一个罗马支配函数f，满足$\omega(f) \leq k$? **问题名称：** 扩展罗马支配(ExtRD)**给定：** 一个图$G = (V, E)$和一个函数$f\colon V \to \{0, 1, 2\}$**问题：** 对于$G$，是否存在一个极小罗马支配函数$g$，满足$f \leq g$?

注意这与支配集及其扩展变体之间的类比：输入是一个顶点集合X，问题是是否存在一个包含$X \subseteq D$的极小支配集D。对于罗马支配，考虑的是函数的逐点序；而对于支配，采用的是集合的包含序。考虑特征函数而非集合，可以进一步统一这两种情况。有些出人意料的是，文献 [4]证明 Ext RD 可以在多项式时间内求解。这意味着可以用多项式延迟枚举极小罗马支配函数。该算法的基础是对极小罗马支配函数的组合刻画。为了表述前面提到的极小罗马支配函数的组合刻画，我们还需要引入一个概念。对于$D \subseteq V$且$v \in D$，定义*private neighborhood* 为$v \in V$相对于D的$P_{G,D}(v) := N[v] \setminus N[D \setminus \{v\}]$，其中的元素是$v$的私有邻点。

定理 1. *[4]* 设 $G = (V, E)$ 是一个图，且 $f\colon V \to \{0, 1, 2\}$ 是一个函数。简记 $G' := G\left[f^{-1}(0) \cup f^{-1}(2)\right]$。那么，$f$ 是一个极小罗马支配函数，当且仅当满足以下约束：

1. $N\left[f^{-1}(2)\right] \cap f^{-1}(1) = \emptyset$ (1-顶点与 2-顶点的不干扰性)，2. $\forall v \in f^{-1}(2) : P_{G',\, f^{-1}(2)}(v) \not\subseteq \{v\}$，也称为隐私条件，以及3. $f^{-1}(2)$ 是 G 的极小支配集′ (2-顶点的极小性)。

为了探究这些结果为何能够成立，我们将以两种方式把这些概念和问题推广到超图上，并明确聚焦于第二种可能性。

第一种可能是最自然的，类比于（简单）超图中的支配集概念构造而成；参见 [5]。设 $H = (V, E)$ 是一个（简单）超图，即 $E \subseteq 2^V$。那么，$f: V \to \{0, 1, 2\}$ 是一个罗马支配函数，如果对于所有满足 $v \in V$ 且 $f(v) = 0$ 的顶点，都存在一个顶点 $u \in V$，满足 $f(u) = 2$，且其为 v 的邻点，即它与 v 共享一条边；更形式化地说，存在某条 $e \in E$，满足 $\{u、v\} \subseteq e$。随后，可以像对图那样定义这些问题。我们将其称为超图罗马支配，简称 HRD；以及扩展超图罗马支配，简称 Ext HRD。

然而，正如下文所示，我们基本上通过使用相同的归约，就可以将罗马支配在图情形下的所有重要性质转移到超图情形。更具体地说，我们可以证明：

命题1。 HRD 是*NP*-完全的。

命题2。 Ext HRD可以在多项式时间内求解。

命题3。 可以用多项式延迟枚举超图的所有极小罗马支配函数 .

也就是说，如果$G = (V, E)$是一个简单超图，我们可以通过如下方式构造图$G' = (V, E')$：令 $\{x,y\} \in E'$ 当且仅当存在超边$e \in E$，满足 $\{x,y\} \subseteq e$。这一构造也称为G的盖夫曼图 。于是，$f: V \to \{0, 1, 2\}$ 是超图G的罗马支配函数，当且仅当f是图G'的罗马支配函数。反过来，我们只需将给定的图解释为一个简单超图即可。

因此，我们提出了将罗马支配进一步推广到超图上的一种方式，使我们能够探究罗马支配在扩展版本中为何表现出如此特殊的性质，以及极小罗马支配函数的计数问题。

2.3. 罗马命中函数

现在我们介绍本文最重要的新概念。设 $H = (X, \hat{S} = (s_i)_{i \in I})$ 是一个超图，且 $\tau : X \to I$ 是一个对应关系。回到罗马支配这一概念背后的初衷，现在用超边的指标来建模区域，并且每个区域至多可部署一支军队。此时，花费一单位即可守护所部署军队的区域，花费两单位则可守护所有与之关联的区域。若函数 $f: X \to \{0, 1, 2\}$ 满足：对每个 $i \in I$，都存在一个 $x \in s_i$，使得 $f(x) = 2$；或者存在一个 $x \in X$，使得 $\tau(x) = i$ 且 $f(x) = 1$，则称 f 为罗马击中函数（简称 Rhf）。在这些情形下，我们称 x 击中 i 或 s_i。对于函数 $f: X \to \{0, 1, 2\}$，定义划分 $P(f) = \{f^{-1}(0), f^{-1}(1), f^{-1}(2)\}$。现在我们定义两个与罗马击中函数相关的判定问题。

问题名称： 罗马击中函数，简称 RHF**给定：** 一个有限集 X、一个超边族 $\hat{S} = (s_i)_{i \in I}$，构成超图 $(X, \hat{S})$，一个对应关系 $\tau : X \to I$，以及$k \in \mathbb{N}$**问题：** 是否存在一个罗马击中函数 f，使得 $\omega(f) \leq k$?

问题名称： 扩展罗马击中函数，简称扩展罗马击中函数**给定：** 一个有限集 X、一个超边族 $\hat{S} = (s_i)_{i\in I}$，构成超图 $(X, \hat{S})$，一个对应关系 $\tau : X \to I$，以及 $f: X \to \{0, 1, 2\}$**问题：** 是否存在一个极小罗马击中函数 g，使得 $f \leq g$?

为理解这一设定以何种方式推广了罗马支配，请注意，将图表示为超图有多种方式；也就是说，与图 $G = (V, E)$ 相关联的闭邻域超图 G_{nb} 可以表示为 $G_{nb} = (V, (N[v])_{v\in V})$。显然，$D \subseteq V$ 是支配集，当且仅当 D 是 G_{nb} 的击中集。由于 $v \in N[v]$, ，该恒等关系可以视为一种对应关系。在这一解释下，$f : V \to \{0, 1, 2\}$ 是 G 的罗马支配函数，当且仅当它是 G_{nb} 的罗马击中函数。

2.4. 论文的组织结构与主要结果

在第3节中，我们将证明我们的优化问题是NP完全的；更有趣的是，问题k-罗马击中函数是W[2]完全的。随后，我们将注意力转向扩展问题。回顾一下，罗马支配情形下的算法结果基于一些基本的组合洞见；为方便读者，我们实际上会在本节末尾再次介绍这些洞见。遵循这一思路，我们首先在第4节中给出极小罗马击中函数的组合刻画，并在第5节中加以利用，证明具有满射对应的扩展罗马击中函数可以在多项式时间内求解。在第6节中，我们回到罗马支配，考察扩展问题的一个变体：对于我们所寻找的极小罗马支配函数，同时给出下界和上界条件。与原始问题（仅提供下界）不同，正如我们将证明的那样，这个双侧扩展问题是NP完全的。此外，以下界函数的权重对该问题进行自然参数化（其动机来自距平凡性距离原理 [34]）并没有真正带来帮助；相反，它产生了一个属于W[3] 类且为完全的参数化问题。随后，我们在第7节中利用这些结果证明：一般而言，以给定函数的权重为参数时，罗马击中函数的扩展问题是W[3]完全的。在本节中，我们研究该扩展问题的其他参数化方式，有时证明其困难性结果，但有时也得到FPT结果。在第8节中，我们展示如何将我们的结果应用于开发罗马击中函数的枚举算法。我们重点进行输出敏感分析，划分横截硬度与多项式延迟之间的界限。我们还表明，我们的所有枚举算法——无论具有多项式延迟还是FPT延迟——都只使用多项式空间（第16种情形除外；该情形给出一种有条件的多项式延迟枚举算法，但不一定使用多项式空间）。

这里先提一下，采用我们的方法，以多项式延迟枚举极小解意义下的可行性，与相应扩展问题在多项式时间内可解意义下的可行性是相辅相成的。但即使在支配类问题中，情况也未必如此。正如 [14], 所示，边支配集的扩展问题是NP完全的，而极小边支配集则可以用多项式延迟列举，如 [39]所示。 [41]还展示了一个与罗马支配相关的例子。

3. RHF 优化问题

本节将讨论 RHF 优化问题的（参数化）复杂度。该问题最自然的参数化方式，或许是以罗马击中函数权重的上界k 作为参数。对于我们的结果，将用到 [27]中证明的罗马支配的W[2]完全性（以k 为参数）。其困难性源于罗马支配函数可以解释为罗马击中函数。对于成员性，我们构造一个分裂图，其中团表示论域中的元素，独立集则由超边构成。如果超边相对于 τ 的原像为空，就在独立集中将该超边加入两次。这样一来，与其将这两个顶点赋值为 1，不如将一个邻点赋值为 2。

定理 2。 RHF 是*NP*完全的。参数化问题 k-RHF 是W[2]完全的。

证明。 如第 2 节所述，罗马支配函数可以解释为罗马击中函数，因此所有困难性命题都可从罗马支配问题转化而来。

因此，现在给出其归属性的证明。设 $H = (X, \hat{S} = (s_i)_{i\in I})$ 是一个超图， $\tau : X \to I$ 是一个对应关系（因此，对于每个 $x \in X$，有 $\tau(x) \in \mathbf{I}(x)$），且 $k \in \mathbb{N}$。定义图 $G = (V,E)$，其中

$$
\begin{aligned}
V &:= \{a,b,c\} \cup \overbrace{\{v_x \mid x \in X\}}^{=:V(X)} \cup W \cup U, \quad \text{where} \\
W &:= \{w_i \mid i \in I\} \quad \text{and} \\
U &:= \{u_i \mid i \in I, \tau^{-1}(i) = \emptyset\}, \\
E &:= \{\{a,b\},\{a,c\}\} \cup \{\{a,v_x\} \mid x \in X\} \cup \{\{v_x,w_i\} \mid i \in I, x \in s_i\} \\
&\cup \{\{v_x,u_i\} \mid i \in I, x \in s_i, u_i \in U\} \cup \{\{v_x,v_y\} \mid x,y \in X\}.
\end{aligned}
$$

注意到 G 是一个分裂图，其中 $V(X) \cup \{a\}$ 构成一个团，而 $V \setminus (V(X) \cup \{a\})$ 是独立的。

命题。 带有对应关系 τ 的 $H = (X, \hat{S})$ 存在权重至多为 k 的罗马击中函数，当且仅当 G 存在权重至多为 $k + 2$ 的罗马支配函数。

证明。 我们首先证明从左到右的蕴含关系。设$f : X \to \{0,1,2\}$是H上的一个极小罗马击中函数，其对应关系为 τ。定义$g : V \to \{0,1,2\}$如下：

$$
v \mapsto \begin{cases} 2, & \text{if } v \in \{a\} \cup \{v_x \mid x \in X : f(x) = 2\}, \\ 1, & \text{if } v \in \{w_i \in W \mid \exists x \in \tau^{-1}(i) : f(x) = 1\}, \\ 0, & \text{if } v \in \{b,c\} \cup \{w_i \in W \mid \forall x \in \tau^{-1}(i) : f(x) \neq 1\} \cup U. \end{cases}
$$

由于$g(a) = 2$，$N[a] = \{a、b、c\} \cup V(X)$ 受到支配。还有两类顶点，g 将其赋值为零。(a) 设$u_i \in U$，其中$i \in I$。由于 $\tau^{-1}(i) = \emptyset$，且f 是罗马击中函数，因此存在$x \in X \cap s_i$，使得

$f(x) = 2$。根据定义，$v_x \in V(X) \cap N(u_i)$，且 $g(v_x) = 2$。因此，U 中的每个顶点都受到支配。(b) 设 $w_i \in W$，其中 $i \in I$，且 $g(w_i) = 0$。因此，对于每个 $x \in \tau^{-1}(i)$，$f(x) \neq 1$。由于 f 是罗马击中函数，因此存在 $x \in X \cap s_i$，使得 $f(x) = 2$。因此，w_i 受到某个 v_x 的支配，且 g 是罗马支配函数。最后，

$$\sum_{v \in V} g(v) = 2 + \sum_{x \in X \text{ with } f(x)=2} g(v_x) + \sum_{i \in I, (\exists x \in \tau^{-1}(i):\, f(x)=1)} g(w_i)$$
$$\leq 2 + \sum_{x \in f^{-1}(2)} f(x) + \sum_{x \in f^{-1}(1)} f(x) \leq k + 2\,.$$

因此，g 是一个权重至多为$k + 2$的罗马支配函数。

对于反方向，假设$f\colon V \to \{0,1,2\}$是G上的一个权重至多为$k + 2$的最小罗马支配函数。首先，我们推导出两个组合性质；由于f是最小的，不失一般性，这些性质对f均成立。

(A) 不失一般性，若存在 $v, w \in V$，使得 $N[w] \subseteq N[v]$ 且 $f(v) + f(w) \geq 2$，则 $f(v) = 2$ 且 $f(w) = 0$。我们可以作此假设，因为其他每种情况（即假设 $f(v) = 1$ 且 $f(w) \geq 1$，或 $f(v) = 0$ 且 $f(w) = 2$）所支配的顶点至多相同。例如，对于 $x \in s_i$，$N[w_i] \subseteq N[v_x]$。因此，我们可以假设 $\{v \in V \mid f(v) = 2\} \subseteq \{a\} \cup \{v_x \mid x \in X\}$ 且 $W \cap f^{-1}(2) = \emptyset$。

（B）不失一般性，若存在顶点 u、v、$w \in V$，使得 $N[u] \cup N[w] \subseteq N[v]$ 且 $f(u) + f(v) + f(w) \geq 2$，则 $f(u) = f(w) = 0$ 且 $f(v) = 2$。也就是说，设 u、v、$w \in V$，使得 $N[u] \cup N[w] \subseteq N[v]$ 且 $f(u) + f(v) + f(w) \geq 2$。假设 $f(v) \neq 2$。若 $f(v) = 1$，则 $f(u) + f(v) + f(w) \geq 2$ 使我们回到情况（A）的证明。因此，我们假设 $f(u) = f(w) = 1$ 且 $f(v) = 0$。定义 $h : V \to \{0,1,2\}$，其中

$$v \mapsto \begin{cases} f(x), & \text{if } x \notin \{u, v, w\}, \\ 0, & \text{if } x \in \{u, w\}, \\ 2, & \text{if } x = v\,. \end{cases}$$

$N[u] \cup N[w]$ 相对于 h 被 v 支配。其他顶点的支配方式与罗马支配函数 f 相同。此外，$\sum_{v \in V} h(v) \leq \sum_{v \in V} f(v)$。因此，我们可以假设 $f(a) = 2$，并且对于 $x \in \{b, c\} \cup \{w_i, u_i \mid i \in I :\ \tau^{-1}(i) = \emptyset\}$，有 $f(x) = 0$。对于每个满足 $f(w_i) = 1$ 的 $w_i \in W$，$\tau^{-1}(i) \neq \emptyset$。

现在，我们定义 H 上基于 τ 的罗马击中函数。定义 $g : X \to \{0,1,2\}$，其中若 $f(v_x) = 2$，则 $g(x) = 2$；若存在某个 $w_i \in W$，使得 $f(w_i) = 1$，则 $g(x) = 1$，对 $x \in \tau^{-1}$ 定义；否则，$g(x) = 0$。考虑满足以下条件的 $i \in I$：不存在 $x \in X$，使得 $g(x) = 1$ 且 $\tau(x) = i$。因此，$f(w_i) = 0$，并且存在一个 $v_x \in V(X) \cap N(w_i)$，使得 $f(v_x) = 2$。于是，存在一个 $x \in s_i$，使得 $g(x) = 2$。因此，g 是一个罗马击中函数。此外，

$$\sum_{x \in X} g(x) = \sum_{x \in X:\, f(v_x)=2} g(x) + \sum_{x \in X:\, \tau(x)=i, f(w_i)=1} g(x) = \sum_{v \in V} f(v) - \underbrace{2}_{=f(a)} \leq k.$$

◊

至此，我们完成了对该多项式时间多对一归约运作方式的描述，并证明了其既属于NP，也属于W[2]。

□

文献中多次提到，即使在分裂图上，罗马支配问题仍然是NP-完全的，例如 [19, 40], 。但据我们所知，这一事实尚未有已发表的证明。下面我们将给出一个更强的结论。回顾一下，若图中的两个顶点u、v 满足 $N[u] = N[v]$，则称它们为真孪生顶点。

引理 1. 罗马支配即使在无真孪生顶点的分裂图上也是*NP*-完全的。同样，*k*-罗马支配在无真孪生顶点的分裂图上是*W*[2]-完全的。

证明。定理 2 中给出的构造说明了如何根据罗马击中函数的一个实例，构造一个图 G，作为罗马支配问题的实例。不难看出，这个图 G 确实是一个分裂图，因为（沿用定理 2 证明中的记号）$V(X) \cup \{a\}$ 构成一个团，而 $\{b, c\} \cup W \cup U$ 是一个独立集。因此，任何真孪生顶点都必须属于 $V(X) \cup \{a\}$。正如定理 2 证明中所述，$g(a) = 2$ 由 $b, c \in N(a)$ 强制确定。根据定理 1（1-顶点与 2-顶点的不干扰性），这确保了 $g(v_x) \neq 1$，否则便会推出存在一个权重更小的罗马击中函数。由于 $b, c \notin N(V(X))$，不存在 $x \in V(X)$ 使得 $N[x] = N[a]$, ，也就是说，真孪生顶点必须位于 $V(X)$ 中。因此，真孪生关系将 $V(X)$ 划分为若干等价类。从每个类 $C \subseteq V(X)$ 中选取一个代表元 v_C，并令 $V'(X) \subseteq V(X)$ 为所有代表元构成的集合。令 $G' = G[V'(X) \cup \{a, b, c\} \cup W \cup U\}]$。图 G' 是一个无孪生顶点分裂图，当且仅当图 G 存在一个权重至多为 k 的罗马支配函数。

□

定理 2 的证明中，关于困难性的部分（隐含地）利用了这样一个事实：由于原图中可能存在孪生顶点（具有相同闭邻域的顶点），因此超边族中可能会多次包含同一个超边。考虑一个具有 ≥ 2 个顶点的完全图 $K[n], E_n$。由于闭邻域总是相同的，若用普通集合而非超边族表示超边，那么对任意顶点 v，最佳解都会是 $\chi_{\{v\}}$。这甚至不是一个罗马支配函数。具体而言，对任意顶点 v，极小罗马支配函数的形式会是 $2 \cdot \chi_{\{v\}}$，或者它们恒为 1。尽管如此，重新考察引理 1，可得以下结论。

推论 1. 罗马命中函数即使在简单超图上也是 *NP*-完全的。此外，*k*-罗马命中函数是*W*[2]-完全的。

该优化问题的NP完全性也促使我们分析扩展问题；这或许有助于加快求解该判定问题的精确分支算法。让我们暂且将罗马击中函数视为一个最小化问题。我们已经提到，最小 RD 可在对数因子范围内近似。 [46] 的证明思路很容易推广到罗马击中函数，并得出下一个结果；我们还结合补充的困难性结果给出该结果。

定理 *3.* 存在一个多项式时间算法，对于给定的最小罗马击中函数实例 $(H = (X, (s_i)_{i\in I}), \tau)$，该算法输出一个罗马击中函数 $f: X \to \{0,1,2\}$，使得 $\omega(f) \le 2\cdot(\ln(|I|)+1)\cdot opt(H)$。此外，除非 $P = NP$，否则不存在能够达到优于 $1\,2(\ln(|I|) - o(1))\cdot opt(H)$ 的近似比的多项式时间算法。

证明。所述算法将在底层简单超图上运行针对击中集（或在对偶解释下的集合覆盖；参见 [18]）的知名贪心算法，并得到一个覆盖 $S = \{s_i \mid i \in I\}$ 的集合族 $C \subseteq X$。与最优覆盖 C^* 相比，Chvátal [18] 证明了 $|C| \le (\ln(|S|)+1)\cdot|C^*|$。显然，$f := \chi_C$ 是一个有效的罗马击中函数，且 $\omega(f) = 2|C|$。反过来，设 f^* 是一个最优罗马击中函数解，即 $\omega(f^*) = \mathrm{opt}(H)$。我们可以通过定义 $C_{f^*} = (f^*)^{-1}(1) \cup (f^*)^{-1}(2)$，得到将 H 解释为击中集实例时的一个有效解。显然，$|C^*| \le |C_{f^*}| \le \omega(f^*) \le 2\cdot|C_{f^*}|$。将所得到的不等式结合起来，最后注意到 $|S| \le |I|$，即可得到所述贪心算法近似因子的命题：

$$\omega(f) = 2|C| \le 2\cdot(\ln(|S|)+1)\cdot|C^*| \le 2\cdot(\ln(|I|)+1)\cdot\omega(f^*)\,.$$

现在，考虑一个用于最小罗马击中函数的多项式时间近似算法 A，其近似比为 $r(|I|)$，给定一个索引集为 I 的实例。设 $H = (X, S)$ 是一个简单超图，作为最小击中集问题的一个实例，其最优击中集为 $C^* \subseteq X$。我们将其视为最小罗马击中函数问题的一个实例，并在其上运行 A，得到一个解 $f: X \to \{0,1,2\}$。与这个最小罗马击中函数实例的最优解 f^* 相比，我们知道 $\omega(f) \le r(|S|)\cdot\omega(f^*)$。令 $C_f = f^{-1}(1) \cup f^{-1}(2)$，与上文类似。同样，$|C_f| \le \omega(f) \le 2\cdot|C_f|$。如果 C_R 能达到优于 $(\ln(|S|) - o(1))\cdot|C^*|$ 的近似比，那么根据 Dinur 和 Steuer 的著名结果，有 $P = NP$ [22]。但如果 $r(|S|) \le \frac{1}{2}(\ln(|S|) - o(1))$，就会出现这种情况，因为此时我们得到：

$$|C_f| \le r(|S|)\cdot\omega(f^*) \le r(|S|)\cdot\omega(\chi_{C^*}) \le r(|S|)\cdot 2|C^*| \le (\ln(|S|) - o(1))\cdot|C^*|\,,$$

因为我们可以再次将最小 HS 解 C^* 解释为一个权重为 $2|C^*|$ 的解 χ_{C^*}，显然它不可能优于最优的最小 RHF 解 f^*。 □

上一个定理中的下界与上界之间显然存在一个小的差距，但关于 [46] 中最小罗马支配的（不可）近似性结果也存在一个差距，只是没有明确指出。然而，我们的推理结合本节开头给出的归约证明，即使在这种情况下，乘法间隙仍为四。

结合引理 1 中给出的构造，我们可以为无真孪生顶点分裂图上的最小 RD 推导出相同的不可近似结果，从而强化此前发表的此类结果。

4. 极小罗马击中函数的组合性质

在本节中，我们将证明极小罗马命中函数的组合性质。这将帮助我们分析罗马命中函数扩展问题的复杂性。

定理 4. 设 X 为顶点集，$\hat{S} = (S_i)_{i\in I}$ 为超边族，且 $\tau : X \to I$ 为对应关系。则，函数 $f : X \to \{0,1,2\}$ 是极小罗马击中函数，当且仅当满足以下约束项：

0. $\forall x, y \in f^{-1}(1) : x \neq y \Rightarrow \tau(x) \neq \tau(y)$ (τ-1-顶点上的单射性)，1. $\forall x \in f^{-1}(1) : S_{\tau(x)} \cap f^{-1}(2) = \emptyset$ (1-顶点与 2-顶点的不干扰性)，2. $\forall x \in f^{-1}(2)\ \exists i \in I \setminus \{\tau(x)\} : S_i \cap f^{-1}(2) = \{x\}$ (隐私条件)，以及 3. $f^{-1}(2)$ 是 $S' := \{S_i \in S \mid i \in I$ 上的极小击中集，$\tau^{-1}(i) \cap f^{-1}(1) = \emptyset\}$ (2-顶点的极小性)。

鼓励读者将约束1、2、3与定理 1中的相应条目进行比较。这也解释了为什么我们在这里从条目0开始。这些形式上的相似性并非巧合；参见注记1。

证明。我们先证明从左到右的蕴含。设 f 是定义在 X、$\hat{S}$ 和 τ上的极小罗马击中函数。我们首先证明约束 0。假设存在 x、$y \in f^{-1}(1)$，其中 $x \neq y$，但 $\tau(x) = \tau(y) = i$。定义 $\tilde{f} = f - \chi_{\{y\}}$。显然，$\tilde{f} \leq f$ 且 $\tilde{f} \neq f$。由于 $f^{-1}(2) = \tilde{f}^{-1}(2)$ 且 $\tau(\tilde{f}^{-1}(1)) = \tau(f^{-1}(1) \setminus \{y\}) = \tau(f^{-1}(1))$ 成立，$\tilde{f}$ 是一个罗马击中函数。因此，f 不是极小的，这与假设矛盾。

接下来，我们证明约束 1。假设存在一个 $x \in f^{-1}(1)$，满足 $S_{\tau(x)} \cap f^{-1}(2) \neq \emptyset$。定义 $\tilde{f} = f - \chi_{\{x\}}$。显然，$\tilde{f} \leq f$，$f \neq \tilde{f}$，且 $f^{-1}(2) = \tilde{f}^{-1}(2)$。因此，$S_{\tau(x)} \in S(\tilde{f}^{-1}(2))$。对于 $i \in I \setminus \{\tau(x)\}$，$S_i$ 以与被 f 击中相同的方式被 $\tilde{f}$ 击中˜，因此 f 是一个罗马击中函数。这与 f 的极小性矛盾。

最后，我们证明约束 3。假设 $f^{-1}(2)$ 不是 S' 上的极小击中集。如果存在一个未被 $f^{-1}(2)$ 击中的 $s \in S'$，那么 f 就不是罗马击中函数，这与我们的假设矛盾。因此，我们可以假设 $f^{-1}(2)$ 不是极小的。更明确地说，我们假设存在一个 $x \in f^{-1}(2)$，使得对于每个 $i \in I \setminus \{\tau(x)\}$，$S_i \cap f^{-1}(2) \neq \{x\}$。（注意，这一假设是否定约束 2 的命题，因此我们也在此证明约束 2。）于是存在一个 $x \in f^{-1}(2)$，使得对于每个 $s \in S'(x) \setminus \{\tau(x)\}$，都存在一个 $y \in \left(f^{-1}(2) \setminus \{x\}\right) \cap s$。定义 $\tilde{f} = f - \chi_{\{x\}}$。令 $i \in I$。根据定义，$\tau(f^{-1}(1)) \cup \{\tau(x)\} = \tau(\tilde{f}^{-1}(1))$ 成立，因此我们只需考虑 $i \in I \setminus (\tau(f^{-1}(1)) \cup \tau(x))$。对于满足 $S_i \in S(f^{-1}(2)) \setminus S(x)$ 的 i，显然 $S_i \cap \tilde{f}^{-1}(2) \neq \emptyset$。若 $S_i \in S(x) \setminus \{\tau(x)\}$，如前所述，存在一个 $y \in \left(f^{-1}(2) \setminus \{x\}\right) \cap S_i = \tilde{f}^{-1}(2) \cap S_i$。因此，$\tilde{f}$ 是一个罗马击中函数，而 f 不是极小的，这与我们的假设矛盾。因此，四个条件均成立。

对于 if 部分，假设 f 满足这些约束。根据约束 3，我们知道，对于所有 $i \in I$，要么 $S_i \cap f^{-1}(2) \neq \emptyset$，要么存在一个 $x \in f^{-1}(1)$，使得 $\tau(x) = i$。因此，f 是一个罗马击中函数。现在我们证明 f 的极小性。设 $g : X \to \{0,1,2\}$ 是一个极小罗马击中函数，且 $g \leq f$。因此，$g^{-1}(2) \subseteq f^{-1}(2)$，并且 $S(g^{-1}(2)) \subseteq S(f^{-1}(2))$ 成立。此外，$g^{-1}(1) \subseteq f^{-1}(1) \cup f^{-1}(2)$。由于对每个 $x \in X$，$\{S_{\tau(x)}\} \subseteq S(x)$，对于每个 $i \in \tau(g^{-1}(1))$，$i \in \tau(f^{-1}(1))$ 或 $S_i \in S(f^{-1}(2))$。为导出矛盾，设 $x \in X$ 是一个满足 $g(x) < f(x)$ 的元素。

情形1： $g(x) = 0 < 2 = f(x)$。这意味着，对于每个 $i \in \mathbf{I}(x)$，都存在一个 $y \in s_i$，使得 $2 = g(y) \leq f(y) = 2$ 或 $y \in \tau^{-1}(i) \cap g^{-1}(1) \subset \tau^{-1}(i) \cap (f^{-1}(1) \cup f^{-1}(2)$。这要么与约束 1 矛盾，要么与约束 2 矛盾。

情形2： $g(x) = 1 < 2 = f(x)$。这种情形的处理方式类似 gously，不知为何相 $\mathrm{p}_{\text{需}}^{\text{对更简单。我们只}}$ y 排除$i = \tau(x)$。

情形 3： $f(x) = 1$。这意味着 $g(x) = 0$。由于 g 是一个罗马击中函数，要么 $s_{\tau(x)} \cap g^{-1}(2)$ 非空，要么存在一个$y \in g^{-1}(1)$，使得 $\tau(x) = \tau(y)$。

情形 3.1： $\tau(x) \cap g^{-1}(2) \neq \emptyset$。由于 $g^{-1}(2) \subseteq f^{-1}(2)$，这与约束 1 矛盾。

情形 3.2： 存在一个$y \in g^{-1}(1)$，使得 $\tau(x) = \tau(y)$。因此，要么存在一个$y \in f^{-1}(1) \setminus \{x\}$，使得 $\tau(x) = \tau(y)$（这与约束0矛盾），要么$f(y) = 2$（这与约束 1 矛盾）。

因此，$g = f$ 成立。因此，f 是极小的。 □

注记1。 现在我们将比较定理 *4* 与定理 *1*。设 $G = (V, E)$ 是一个图，且 $G_{nb} = (V, (N[v])_{v \in V})$是其闭邻域超图。函数 $\tau : V \to V$ 是恒等函数。设 $f : V \to \{0, 1, 2\}$。

由于 τ 是双射，对所有 f，约束*0*均成立。接下来我们将证明，f 满足定理 *4* 的约束 *1*，当且仅当 f 满足定理 *1* 的约束 *1*。假设存在 $v \in f^{-1}(1)$，使得 $N[v] \cap f^{-1}(2) \neq \emptyset$。因此，$N[f^{-1}(2)] \cap \{v\} \subseteq N[f^{-1}(2)] \cap f^{-1}(1)$ 非空。这与定理 *1* 的约束 *1* 矛盾。反之，若存在元素 $v \in N[f^{-1}(2)] \cap f^{-1}(1) w$，则意味着存在 $w \in f^{-1}(2) \cap N[v]$。这与定理 *4* 的约束 *1* 矛盾。

对于每个 $v \in f^{-1}(2)$，存在一个 $u \in V \setminus \{v\}$，使得 $N[u] \cap f^{-1}(2) = \{v\}$，当且仅当 $f^{-1}(2)$ 中的每个顶点都有一个不等于自身的私有邻居。我们可以将定理 *4* 最后一个约束中出现的集合改写为 $\{N[v] \mid v \in V, \tau^{-1}(v) \cap f^{-1}(1) = \emptyset\} = \{N[v] \mid v \in V, v \notin f^{-1}(1)\}$。这是超图 $(f^{-1}(\{0, 2\}), (N[v])_{v \in V \setminus f^{-1}(1)})$ 的超边集合。$f^{-1}(2)$ 是该超图的一个极小击中集，当且仅当它是 $((f^{-1}(\{0, 2\}), (N[v] \setminus f^{-1}(1))_{v \in V \setminus f^{-1}(1)}))$ 的一个极小击中集；后者是 $G' = G[f^{-1}(\{0, 2\})]$ 的闭邻域超图。因此，在这种情况下，定理 *1* 的最后一个约束等价于定理 *4* 的最后一个约束。

我们称超图 $H = (X, \hat{S})$ 上的函数 $f : X \to \{0, 1, 2\}$ 可扩展，若存在一个极小罗马击中函数 g，使得$f \leq g$。以下两个结果由定理 4 推出。

引理 2. 设 $H = (X, \hat{S} = (s_i)_{i \in I})$ 是一个具有对应关系 τ 的超图，且 $f : X \to \{0, 1, 2\}$ 是一个函数，其中 $x \in f^{-1}(2)$，$y \in f^{-1}(1)$ 且 $x \in s_{\tau(y)}$。那么，f 可扩展当且仅当 $f + \chi_{\{y\}}$ 可扩展。

证明。由于 $\leq$ 在 $\mathbb{N}^X$ 上是传递的，若 $f + \chi_{\{y\}}$ 可扩展，则 f 也可扩展。反过来，假设存在一个极小罗马击中函数 g，使得 $f \leq g$。因此，对于 $x \in f^{-1}(2)$，有 $g(x) = f(x) = 2$，且对于 $y \in f^{-1}(1)$，有 $g(y) \in \{1, 2\}$。根据定理 4 的约束 1，由于 $x \in s_{\tau(y)}$，$y \in g^{-1}(2)$。对于其余的 $z \in X \setminus \{y\}$，均有 $(f + \chi_{\{y\}})(z) = f(z) \leq g(z)$。这意味着 $f + \chi_{\{y\}} \leq g$，因此 $f + \chi_{\{y\}}$ 可扩展。

□

引理 3。 设 $H = (X, \hat{S} = (s_i)_{i \in I})$ 是一个具有对应关系 τ 的超图，且 $f : X \to \{0, 1, 2\}$ 是一个函数，满足 $x, y \in f^{-1}(1)$，$x \neq y$ 且 $\tau(x) = \tau(y)$。那么，当且仅当 $f + \chi_{\{x,y\}}$ 是可扩展的时，f 才是可扩展的。

证明。由于 $\leq$ on $\mathbb{N}^X$ 是传递的，如果 $f + \chi_{\{x,y\}}$ 是可扩展的，那么 f 也是可扩展的。反之，假设存在一个极小罗马击中函数 g，满足 $f \leq g$。因此，$g(x)$、$g(y) \in \{1,2\}$。根据定理 4 的约束0，$g(x) = 2$ 或 $g(y) = 2$。这意味着，如果 f 是可扩展的，那么 $f + \chi_{\{x\}}$ 或 $f + \chi_{\{y\}}$ 是可扩展的。在这两种情况下，我们都可以使用引理 2，并得出结论：如果函数 f 是可扩展的，则 $f + \chi_{\{x,y\}}$ 是可扩展的。

□

利用引理 2 和引理 3，我们得到以下结果。

引理4. 设 $H = (X, \hat{S} = (s_i)_{i\in I})$ 是一个具有对应关系 τ 的超图，且 $f : X \to \{0,1,2\}$ 是一个函数。那么，$\tilde{}$存在一个函数 f，使其满足以下条件：

1. 不存在两个元素 $x, y \in X$，使得 $f(x) = f(y) = 1$ 且 $\tau(x) = \tau(y)$。2. 不存在两个元素 $x, y \in X$，使得 $f(x) = 2, f(y) = 1$ 且 $x \in s_{\tau(y)}$。$\tilde{}$3. 当且仅当 f 可扩展时，f 可扩展。

这使我们能够提出一个关于可扩展性的刻画定理，该定理稍后将变得相当重要。

定理 5. 设 $H = (X, \hat{S} = (s_i)_{i\in I})$ 是一个具有对应关系 $\tau, \tau : X \to I$ 的超图。设 $f : X \to \{0,1,2\}$ 是一个函数，使得对每个 $x, y \in f^{-1}(1)$，若 $x \neq y$，则 $\tau(x) \neq \tau(y)$。那么，f 可扩展，当且仅当存在一个集合 R_2，满足 $f^{-1}(2) \subseteq R_2 \subseteq f^{-1}(1) \cup f^{-1}(2)$，以及一个映射 $\rho : R_2 \to I$，且满足以下约束。

1. $\forall x \in R_2 : \rho(x) \neq \tau(x)$. 2. $\forall x \in R_2 : s_{\rho(x)} \cap R_2 = \{x\}$. 3. $\forall x \in f^{-1}(1) \setminus R_2 : s_{\tau(x)} \cap R_2 = \emptyset$. 4. $\forall i \in I$ such that $\tau^{-1}(i) = \emptyset : s_i \subseteq \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)}\right) \implies s_i \cap R_2 \neq \emptyset$.

其思路是，R_2 包含那些从 1-顶点“提升”为 2-顶点的顶点，以及 2-顶点（与 f 相比），而且 ρ 随后确保“新”2-顶点具有隐私性。约束 4 确保可以命中所有超边，同时不命中那些私有顶点。

证明。定义 $I' := \{i \in I \mid \tau^{-1}(i) = \emptyset\}$。首先，假设 f 是可扩展的。令 $g : X \to \{0,1,2\}$ 为一个极小罗马击中函数，且 $f \leq g$。根据定理4的约束3，$g^{-1}(2)$ 是定义在 $\{s_i \in S \mid i \in I, \tau^{-1}(i) \cap g^{-1}(1) = \emptyset\}$ (∗) 上的极小击中集。由定理 4 的约束 2 可知，对于每个 $x \in g^{-1}(2)$，都存在一个指标，我们记为 $\rho(x) \in I \setminus \{\tau(x)\}$，使得 $s_{\rho(x)} \cap g^{-1}(2) = \{x\}$。定义 $R_2 := (f^{-1}(1) \cup f^{-1}(2)) \cap g^{-1}(2)$。显然，$f^{-1}(2) \subseteq R_2 \subseteq f^{-1}(1) \cup f^{-1}(2)$。我们可以将指标函数 ρ 解释为一个映射 $R_2 \to I$。我们需要检验 R_2 和 ρ 所满足的四个约束。第一个约束由 ρ 的构造直接成立。根据 (∗)，第二个约束以稍微更一般的形式成立。现在来看第三个约束。如果存在一个 $y \in f^{-1}(1) \setminus R_2 \subseteq g^{-1}(1)$，使得 $\emptyset \neq s_{\tau(y)} \cap R_2 \subseteq s_{\tau(y)} \cap g^{-1}(2)$，那么这将与定理 4 的约束 2 矛盾。现在来看第四个也是最后一个约束。令 $i \in I'$。由于 $\tau^{-1}(i) = \emptyset$，必存在一个 $y \in g^{-1}(2) \cap s_i$。如果 $y \in R_2$,，则该约束成立。因此，我们可以假设 $y \in g^{-1}(2) \setminus R_2 = g^{-1}(2) \cap f^{-1}(0)$。考虑 $x \in f^{-1}(1) \setminus R_2$。由于 $f \leq g$ 且 $x \notin R_2$,，我们有

$g(x) = 1$。根据定理 4 的约束 1，$y \notin s_{\tau(x)}$。如果$s_i \subseteq \left(\bigcup_{x \in f^{-1}(1) \setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x \in R_2} s_{\rho(x)}\right)$和$s_i \cap R_2 = \emptyset$成立，那么这将与$s_{\rho(x)} \cap R_2 = \{x\}$矛盾。因此，所有约束均成立。

现在设 f、R_2 和 $\rho : R_2 \to I$ 满足该定理的约束。在证明的这一部分，我们将定义一个超图 H'，其中包含 I' 中尚未被击中的边，即不存在 $x \in X$，使得 $x \in R_2$ 或 $\tau(X)$ 是该边。我们将证明，H' 上存在一个极小击中集 D，其中不包含 $s_{\rho(x)}$ 中任意一个顶点（$x \in R_2$或 $s_{\tau(x)}$ 对应 $f^{-1}(1) \setminus R_2$）。$R_2 \cup D$ 将表示取值为 2 的顶点集合。我们将通过给某些顶点赋值 1 来击中其余顶点。因此，我们定义超图 $H' = (X', (s'_i)_{i \in I''})$，其中

$$I'' := I' \cap \{i \in I \mid s_i \cap R_2 = \emptyset\},$$
$$X' := \left(\bigcup_{i \in I''} s_i\right) \setminus \left(\left(\bigcup_{x \in f^{-1}(1) \setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x \in R_2} s_{\rho(x)}\right)\right)$$

以及 $s'_i := s_i \cap X'$。如果对于某个 $i \in I''$，s'_i 为空，那么 H' 上就不存在击中集。因此，我们需要确保不存在这样的指标。设 $i \in I''$。于是，$\tau^{-1}(i) = \emptyset$ 且 $s_i \cap R_2 = \emptyset$。约束 4 中蕴含关系的逆否命题表明

$$s_i \not\subseteq \left(\bigcup_{x \in f^{-1}(1) \setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x \in R_2} s_{\rho(x)}\right).$$

因此，对于每个 $i \in I''$，$s'_i \neq \emptyset$。于是，H'上存在一个极小击中集 D。此外，H' 和 D 的构造意味着，对于每个 $i \in I \setminus I(R_2 \cup D)$，$\tau^{-1}(i) \neq \emptyset$。对于每个 $i \in I \setminus I(R_2 \cup D)$，$x_i$ 表示$\tau^{-1}(i)$中的任意一个顶点，除非存在一个 $x_i \in \left(f^{-1}(1) \setminus R_2\right) \cap \tau^{-1}(i)$（根据对 f 的假设，这样的元素至多有一个）。在这种情况下，我们选择这个 x_i。

定义 $g : X \to \{0, 1, 2\}$，其中 $g^{-1}(1) = \{x_i \mid i \in I \setminus I(R_2 \cup D)\}$，且 $g^{-1}(2) = D \cup R_2$。现在我们将使用定理 4 来证明 g 是一个极小罗马击中函数。根据 $g^{-1}(1)$ 的构造，定理 4 的前两个约束，即约束 0 和约束 1，均得到满足。由于 $g^{-1}(1)$ 击中 $I \setminus I(D \cup R_2)$ 中的每条边，$\{s_i \mid i \in I, \tau^{-1}(i) \cap g^{-1}(1) = \emptyset\}$ 中的每条超边都由 $g^{-1}(2)$ 击中。由于 D 是极小的，且 $D \cap \left(\bigcup_{x \in R_2} s_{\rho(x)}\right) = \emptyset$，约束 2 意味着 $D \cup R_2$ 也是 $I(D \cup R_2)$ 上的一个极小击中集。定理 4 的其余约束可由 f 的前两个约束，以及 H' 和 D 的定义推出，因为 H' 仅包含满足 s'_i where $\tau^{-1}(i) = \emptyset$ 的超边。

□

注 2. 设 $H = (X, (s_i)_{i \in I})$ 是一个具有对应关系 τ 的超图。结合引理 *3* 和定理 *5*，我们可以判断函数 $f : X \to \{0, 1, 2\}$ 是否可扩展（因此也可以判断(H, τ, f)是否为 是-实例，属于扩展罗马罗马命中函数函数问题）。首先，我们穷尽地使用引理 *3*，然后便可使用定理 *5*。本文将利用这一思路证明$W[3]$-归属。

备注 3。 定理 *5* 已经说明了为什么具有满射对应的扩展 罗马击中函数（因此也包括扩展 罗马支配 函数）可以在多项式时间内运行。令 $H = (X, (s_i)_{i \in I})$ 为一个超图，并令 $f : X \to \{0, 1, 2\}$ 为具有满射对应 τ 的函数。因此，我们可以忽略约束 *4*，因为

$\tau^{-1}(i) \neq \emptyset$ 对每个 $i \in I$。接下来，我们可以使用引理 *2* 和引理 *3*。我们设 $R_2 := f^{-1}(2)$。根据约束 *3*，我们必须将每个 $x \in f^{-1}(1)$ 且 $s_{\tau(x)} \cap R_2 \neq \emptyset$ 的元素添加到 R_2 中。这可以在多项式时间内完成（我们将在算法 *1* 中说明具体方法）。现在，我们可以在多项式时间内检查：对于每个 $x \in R_2$，是否存在一个 $i_x \in I$ 满足约束 *1* 和约束 *2*。这也将是我们在下一节中采用的策略。

5. 扩展 展罗马击中函数与满射对应及扩展罗马支配

本节中，我们将介绍一种多项式时间算法，用于处理具有满射对应函数 τ的扩展罗马击中函数实例。在本节末尾，我们将说明如何将该算法视为对先前在 [4] 中研究的扩展罗马支配的一种自然推广。

算法 1 ExtRHF 算法

过程ExtRHF Solver(X, $\hat{S}$, τ, f)
输入： 集合 X，$\hat{S} := (s_i)_{i \in I}$，$\tau$对应函数，$f$：$X \to \{0,1,2\}$其中 $\{i \in I \mid \tau^{-1}(i) = \emptyset\} \subseteq \mathbf{I}\left(f^{-1}(2)\right)$。
输出： 是否存在一个极小罗马击中函数g，使得$f \leq g$?
$M_2 := f^{-1}(2)$
$M_1 := f^{-1}(1)$
对于$x \in M_1$ **执行**
  对于$y \in M_1 \setminus \{x\}$ **执行**
    若 $\tau(x) = \tau(y)$**则**
      将x、y 加入M_2，并从M_1中删除它们。
      继续处理下一个x。
$M := M_2$ {以下考虑所有$x \in g^{-1}(2)$。}
当$M \neq \emptyset$**时**
  选择$x \in M$。
  对 $y \in \tau^{-1}(\mathbf{I}(x))$ **执行**
    如果 $y \in M_1$，**则**
      将 y 添加到 M_2 和 M中。将y 从 M_1中删除。
  将 x从 M中删除。
对 $x \in M_2$ **执行**
  如果 $\mathbf{I}(x) \setminus \mathbf{I}(M_2 \setminus \{x\}) \subseteq \{\tau(x)\}$，**则**
    返回 否
对于 $i \in I \setminus (\mathbf{I}(M_2) \cup \tau(M_1))$ **执行**
  向 M_1 中添加一个任意元素 $x \in \tau^{-1}(i)$。
返回 是 $\{g^{-1}(0) = X \setminus (M_1 \cup M_2),\ g^{-1}(1) = M_1, g^{-1}(2) = M_2\}$

定理6。 算法 *1* 对满足 $\{i \in I \mid \tau^{-1}(i) = \emptyset\} \subseteq \mathbf{I}\left(f^{-1}(2)\right)$ 的实例 $(X, \hat{S}, \tau, f)$，可在多项式时间内求解 Extension Roman Hitting Function 。

证明。假设算法返回 yes。定义 $g: X \to \{0,1,2\}$ 使得 $g^{-1}(0) = X \setminus (M_1 \cup M_2)$，$g^{-1}(1) = M_1$ 且 $g^{-1}(2) = M_2$。由于我们只将顶点从 M_1 放入 M_2，并从 $X \setminus (M_1 \cup M_2)$ 放入 M_1，因此 $f \leq g$ 成立。由于对每个 $i \in I \setminus I(M_2) \subseteq I \setminus I\left(f^{-1}(2)\right)$，$\tau^{-1}(i)$ 都非空，g 是一个罗马击中函数。while 循环结束后，$\tau(M_1) \cap I(M_2)$ 不包含任何元素。由于从第 19 行开始的 for 循环仅将满足 $s_{\tau(x)} \cap M_2 = \emptyset$ 的元素 $x \in X$ 放入 M_2，

$\tau(M_1) \cap \mathbf{I}(M_2)$ 在算法结束时为空。第一个 for 循环确保 M_1 中两个不同元素在 τ 下的取值不同。只有在最后一个 for 循环中，我们才会将一个具有唯一 $\tau(x)$ 且尚未被击中的元素 x 放入 M_1, 。如果存在一个顶点 $x \in M_2$，满足 $\mathbf{I}(x) \setminus \mathbf{I}(M_2 \setminus \{x\}) \subseteq \{\tau(x)\}$，那么算法将返回 no。定理 4 意味着$g$是一个极小罗马击中函数。

假设该算法返回否，但存在一个满足 $f \leq h$ 的极小罗马击中函数。因此，定理 4 的约束对 h 成立。这意味着，对于每个满足 $x \neq y$ 的 x、$y \in h^{-1}(1)$，$I\left(h^{-1}(2)\right) \cap \tau(h^{-1}(1)) = \emptyset$ and $\tau(x) \neq \tau(y)$ 均成立。因此，在第 16 行之前，$M_1 \subseteq h^{-1}(1)$ 和 $M_2 \subseteq h^{-1}(2)$ 均已满足。因此，对于每个 $x \in M_2$，$I(x) \setminus I\left(h^{-1}(2) \setminus \{x\}\right) \subseteq I(x) \setminus I(M_2 \setminus \{x\}) = \{\tau(x)\}$ 均成立。由此，h 不是一个极小罗马支配函数。因此，算法 1 在 $I \setminus \tau(X) \subseteq I\left(f^{-1}(2)\right)$的条件下解决了扩展罗马击中函数问题。

显然，for 循环在多项式时间内运行。由于从第 10 行开始的 while 循环在每次迭代中都会删除 M 中的一个顶点，并且不再将该顶点放回其中，因此该 while 循环只需多项式时间。于是，算法 算法 1 在多项式时间内运行。

□

这意味着以下推论。

推论 2。扩展罗马命中函数在 τ 为满射时是多项式时间可解的。

设 $G = (V, E)$ 为一个图。回顾与 G 相关联的闭邻域超图 G_{nb}，即 $G_{nb} = (V, (N[v])_{v \in V})$。如果我们在 G_{nb} 上运行算法 1，并将其与 [4] 中在 G 上运行的算法 1 进行比较，那么这两个算法的工作方式将相同。由于对应关系 τ 是双射，对于每个满足 $x \neq y$ 的 x、$y \in X$，都有 $\tau(x) \neq \tau(y)$。因此，从第 4 行开始的 for 循环不会改变任何内容。在 while 循环中，我们遍历所有 $x \in M_2$，并寻找 $y \in \tau^{-1}(I(x)) \cap M_1$。ˆS 的定义表明，我们会将这样的 $y \in N[x]$ 的值更新为 $f(y) = 2$。这也会发生在 [4]的 ExtRDF 算法中。下一步，两个算法都会检查每个值为 2 的顶点是否具有一条私有边，且该私有边不是 τ-超边（即不是该顶点自身的私有邻居）。最后，算法 1 取一个未被命中的指标 i，并为一个顶点 $x \in \tau^{-1}(i)$设置 $f(x) = 1$。由于 τ 是单射，顶点 x 是唯一的。因此，两个算法的工作方式相同。由此，我们提出的算法推广了扩展罗马支配的多项式时间可解性。

6. 有界扩展罗马支配

本节将讨论扩展罗马支配的一个双边有界版本，这也是我们一位同事提出的。

问题名称： 有界扩展罗马支配（有界扩展罗马支配）**给定：** 一个图$G = (V, E)$及函数 f和h：$V \to \{0, 1, 2\}$。**问题：** 是否存在一个极小罗马支配函数g：$V \to \{0, 1, 2\}$，使得 $f \leq g \leq h$?

我们将在推论 3 中证明，有界扩展罗马支配是NP-完全的。在这种情况下，一种标准做法是寻找FPT算法。对于该问题，一种自然的参数化方式可以是 $\omega(2 - h)$，因为当 $2 - h = 0$ 时，即 $\omega(2 - h) = 0$，我们就回到了扩展罗马支配这一特殊情形，而该问题

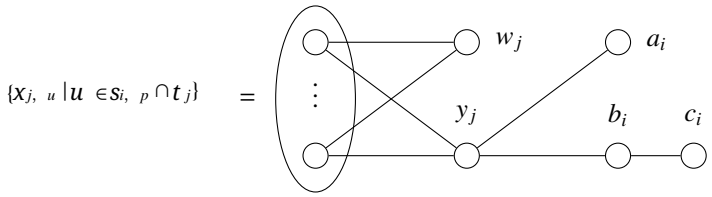


图 1：定理 7 的构造，其中 $i \in [k], p \in [l_i]$ 和 $[j \in l_T]$。

已知可在多项式时间内求解。因此，这种参数化可以被视为一个“距平凡情形的距离”参数 [34]。然而，正如我们将在定理 7 中证明的那样，这种参数化策略是行不通的。在定理 7 的证明中，我们采用了 MultIndFam（多色独立族）关于参数k的、众所周知的W[3]-完全性，并据此给出了归约。

问题名称： 多色独立族（MultIndFam）**给定：** 公共全集 2^U 上的一个由子集组成的（$k+1$）元组（S_1，.....，S_k，T）。**问题：** 是否存在超边 $s_1 \in S_1, \ldots\ldots, s_k \in S_k$，使得不存在 $t \in T$ 是 $\bigcup_{i=1}^{k} s_i \subseteq U$ 的子集？在该定理中，我们实际上讨论的是一种略有不同的参数化，即 $\kappa_{h^{-1}(0)}(G, f, h) := |h^{-1}(0)|$.

定理 7。 $\kappa_{h^{-1}(0)}$-有界-E扩展 R罗马 D支配 问题在二分图上仍然是$W[3]$-难的。

证明。令 $k \in \mathbb{N}$，且 $(S_1, \ldots, S_k, T)$ 为 2^U 的一个 $(k+1)$ 元子集组，它们具有公共全集 U。为简化记号，令 $T = \{t_1, \ldots, t_{\ell_T}\}$，且对 $i \in [k]$，令 $S_i = \{s_{i,1}, \ldots s_{i,\ell_i}\}$。定义 $X_j := \{x_{j,u} \mid u \in t_j\}$（$j \in [l_T]$），$Y_i := \{y_{i,1}, \ldots, y_{i,\ell_i}\}$，以及 $G = (V, E)$，其中

$$V := \left(\bigcup_{i=1}^{k} \{a_i, b_i, c_i\} \cup Y_i\right) \cup \left(\bigcup_{j=1}^{\ell_T} \{w_j\} \cup X_j\right),$$

$$E := \{\{a_i, y_{i,p}\}, \{b_i, y_{i,p}\}, \{b_i, c_i\} \mid i \in [k], p \in [\ell_i]\} \cup \{\{w_j, x_{j,u}\} \mid j \in [\ell_T], u \in t_j\}$$

$$\cup \{\{y_{i,p}, x_{j,u}\} \mid i \in [k], p \in [\ell_i], j \in [\ell_T], u \in s_{i,p} \cap t_j\}.$$

显然，G 是二分的，因为$V = A \cup B$ 将V 分解为两个不相交的独立集，其中

- $A = \left(\bigcup_{i=1}^{k} \{a_i, b_i\}\right) \cup \left(\bigcup_{j=1}^{\ell_T} X_j\right)$ and
- $B = \left(\bigcup_{i=1}^{k} \{c_i\} \cup Y_i\right) \cup \left(\bigcup_{j=1}^{\ell_T} \{w_j\}\right)$.

此外，我们需要映射f、$h \in \{0,1,2\}^V$，其中$f = 2\chi_{\{w_1,\ldots,w_{\ell_T}\} \cup \{b_1,\ldots,b_k\}}$，且$h = 2(1 - \chi_{\{a_1,\ldots,a_k\}})$。

命题。 $S_1, \ldots, S_k$，T 是 是-多色独立集族问题的一个实例，当且仅当在 G 上存在一个极小罗马支配函数 g，使得$f \leq g \leq h$。

证明。设 g 是 G 上的极小罗马支配函数，且 $f \leq g \leq h$。考虑 $i \in [k]$。由于 $g(a_i) = h(a_i) = 0$，$Y_i \cap g^{-1}(2) = N(a_i) \cap g^{-1}(2) \neq \emptyset$。由于 $N(y_{i,p}) \setminus N[g^{-1}(2)] = \{a_i\}$, $|Y_i \cap g^{-1}(2)| = 1$。对每个 $i \in [k], p_i \in [l_i]$定义，使得 $\{y_{i,p_i}\} = Y_i \cap g^{-1}(2)$。假设存在 $j \in [l_T]$ 使得 $t_j \subseteq \bigcup_{i=1}^{k} s_{i,p_i}$。那么，对每个 $u \in t_j$，都有一个

$q_u \in [k]$, 使得 $u \in s_{q_u,\ p_{q_u}}$ 且 $x_{j,\ u} \in N(y_{q_u,\ p_{q_u}})$。因此，$w_j$除了自身以外没有私有邻居。这与$g$是极小的事实矛盾。

对于仅当方向，假设 $s_{1,d_1} \in S_1, \ldots, s_{k,d_k} \in S_k$ 是多色独立族实例的一个解。定义 $g := f + 2 \cdot \chi_{\{y_{1,d_1}, \ldots, y_{k,d_k}\}}$。显然，$f \le g \le h$。由于 $a_i \in N(y_{i,d_i})$，对于每个 $i \in [k]$，$Y_i \cup \{c_i\} = N(b_i)$，并且对于 $j \in [l_T]$，$X_j \subseteq w_j$，$g^{-1}(0) \subseteq N[f^{-1}(2)]$。由于 $g^{-1}(1) = \emptyset$，我们只需证明 $g^{-1}(2)$ 中的每个顶点都有一个除自身以外的私有邻居。显然，对于每个 $i \in [k]$，，c_i 是 b_i 的私有邻居，而 a_i 是 y_{i,d_i} 的私有邻居。现在假设存在某个 $j \in [l_T]$，使得 w_j 除自身以外没有私有邻居。那么，对于每个 $x_{j,u} \in N(w_j)$，存在某个 $q_u \in [k]$，满足 $x_{j,u} \in N(y_{q_u,d_{q_u}})$，这意味着 $u \in s_{q_u,d_{q_u}}$。这与 $t_j \notin \bigcup_{i=1}^{k} s_{i,d_i}$ 矛盾。因此，g 是一个极小罗马支配函数。◊

由于$k = |\{a_1, \ldots, a_k\}| = |h^{-1}(0)|$, 这是一个固定参数可处理归约。 □

由于在定理 7 的证明中给出的归约里，h 没有将任何顶点映射为 1，因此，有界扩展罗马支配关于 $\kappa_{2-h}(G, f, h) := \sum_{v \in V}(2 - h(v))$ 是W[3]-难的。具体而言，在定理 7 的构造中，$\kappa_{2-h}(G, f, h) = 2 \cdot \kappa_{|h^{-1}(0)|}(G, f, h)$。

另一种参数化可以是 $\omega(f)$：如果 $\omega(f) = 0$，则存在一个极小罗马支配函数g：$V \to \{0, 1, 2\}$，满足$f \le g \le h$，当且仅当h是罗马支配函数（这可以在多项式时间内检查）。如果存在这样的g，那么h 也是罗马支配函数。如果h 是罗马支配函数，那么我们可以降低各顶点的值，直到无法在不丧失罗马支配函数性质的情况下继续降低任何顶点的值为止。为了理解该参数的复杂性，我们需要以下击中集的扩展版本。

问题名称： 扩展击中集（Ext HS）**给定：** 一个简单超图 $H = (X, S = (s_i)_{i \in I})$ 和一个集合 $U \subseteq X$。**问题：** 是否存在一个极小击中集$Y \subseteq X$，使得 $U \subseteq Y$?

文献 [10], 证明了，当以 $|U|$ 为参数时，扩展击中集是W[3]-完全的。

定理 8. 有界扩展罗马支配在分裂图上，以下界函数 f 的权重$\omega(f)$ 为参数时是W[3]-困难的。

证明。 设 $((X, (s_i)_{i \in I}), U)$ 是一个扩展击中集实例，其中 $X = \{x_1, \ldots\ldots, x_n\}$ 且 $U \neq \emptyset$。定义 $G = (V, E)$，其中

$$V := X \cup I, \quad E := \{\{x, j\} \mid x_i \in S_j\} \cup \binom{X}{2}.$$

显然，G 是一个分裂图。

设Y 是一个极小击中集，且$U \subseteq Y$。定义

$$g_Y : V \to \{0, 1, 2\}, x \mapsto \begin{cases} 0, & \text{if } x \in I \cup (X \setminus Y), \\ 2, & \text{if } x \in Y. \end{cases}$$

U ⊆Y 且 $g_Y(T) = \{0\}$蕴含 $f(v) \le g_Y(v) \le h(v)$，对每个 v ∈V 都成立。由于 U 非空，g_Y已经支配 X。由于 Y 是一个击中集，对于每个 $s_j \in S$，都存在一个 $x_i \in S_j \cap Y$。因此，对于每个 j ∈I，都存在一个 $x_i \in N(j)$，使得 $g_Y(x_i) = 2$。因此，g_Y是一个罗马支配函数。假设存在一个 $x_i \in X$，使得 $g_Y(x_i) = 2$，且 $g_Y - \chi_{\{x_i\}}$也是一个罗马支配函数。因此，对于每个 $j \in N(x_i)$，都存在一个 $x_q \in N(j) \cap g_Y^{-1}(2)$。于是，对于每个满足 $x_i \in s_j$ 的 $s_j \in S$，都存在一个 $x_q \in (s_j \cap Y) \setminus \{x_i\}$。这与 Y 的极小性矛盾。因此，$g_Y$是一个极小罗马支配函数。由此，如果 (X, S, U) 是扩展击中集的一个是实例，那么 (G, f, h) 就是有界扩展罗马支配的一个是实例。

设 g 是 G 上的一个极小罗马支配函数，且 $f \le g \le h$。定义 $Y_g = g^{-1}(2)$。U 是 Y_g 的一个子集，因为对每个 v ∈V，都有 $f(v) \le g(v)$。由于 $I \subseteq g^{-1}(0)$，必定存在一个 $x_i \in N(j)$，使得 $g(x_i) = 2$。因此，Y_g 是一个击中集。假设存在一个 $x_i \in Y_g$，使得对每个满足 $x_i \in S_j$ 的 $S_j \in S$，都存在一个 $x_q \in Y_g \subseteq S_j$。因此，对每个 $j \in N(x_i) \cap T$，都存在一个 $x_q \in N(j) \setminus \{c_i\}$，使得 $g(x_q) = 2$。这样的 x_q 的存在意味着 x_i 的私有邻居不可能属于 C。这与 g 的极小性矛盾。因此，若 (G, f, h) 是有界扩展罗马支配的一个是实例，则 (X, S, U) 是扩展击中集的一个是实例。

□

由于本节中的归约都是多项式时间归约，而且通过猜测并验证很容易看出该问题属于NP，因此我们可以得出：

推论 3. 即使限定在分裂图或二分图上，有界扩展Extension Roman Domination 也是*NP*完全的。

注 4. 这一结果颇有意思，因为我们已经证明，Extension Dominating Set 在分裂图上可在多项式时间内求解*[42]*。如上所述，Extension RomanDomination 可以在多项式时间内求解，但ExtensionDominating Set是*NP*完全的。现在我们有了一个图类，在该图类上，有界Extension RomanDomination 是*NP*完全的，但Extension Dominating Set可以在多项式时间内求解。

如果顶点集*T* 是一个团，我们最后的证明同样成立。这一改动不会影响证明，因为对每个$t \in T$，都有$h(t) = 0$。此时得到的图是余二部图。因此，我们可以陈述以下结果。

推论 4. 有界扩展Extension RomanDomination 在余二部图上是*NP*完全的。参数化问题 $\omega(f)$-有界扩展Extension Roman Domination 在余二部图上是*W*[3]-难的。

在下一节讨论扩展罗马击中函数的（参数化）复杂性时，我们将使用这一参数化归约来证明W[3]-困难性。那里给出的归约还将表明，对于某些参数化方式，有界扩展罗马支配是W[3]-完全的。因此，本节中证明的W[3]-困难性结果将转化为W[3]-完全性结果。

引理 5。 设$(G = (V, E), f, h)$是一个有界扩展罗马支配E扩展 R罗马 D支配 实例。如果以下三个条件中的任意一个成立，则它是否-实例：

1. $f \not\le h$。2. 存在一个 $v \in V$，使得 $h(v) = 0$ 并且对于所有 $u \in N(v)$，$f(u) \ne 2$。

证明。 1. 这显然成立，因为 ≤是传递的。

2. 假设存在一个 v ∈V，使得 h(v) = 0，并且对于所有 u ∈N(v)，f (u) ≠ 2，则 (G, f, h) 是一个否实例。因此，不存在满足 f ≤ g ≤ h 的罗马支配函数 g。特别地，如果 h(v) = 0，由于 g ≤ h，因此 g(v) = 0；而由于 g 应当是一个罗马支配函数，所以对于某个 u ∈ N(v)，g(u) = 2，从而 h(u) = 2，因为 g ≤h。

□

7. 扩展罗马击中函数的复杂度

在本节中，我们将证明，扩展罗马击中函数还存在一些实例，在考虑其不同参数化的情况下，这些实例分别是 NP-完全和 W[3]-完全的。对于 W[3]隶属性，我们再次使用多色独立族。我们没有找到其他适合进行归约的 W[3]-完全参数化问题。因此，我们使用多色独立族来证明其属于 W[3]。遗憾的是，这一归约相当技术性。

定理9。 参数化问题 $\omega(f)$-扩展 罗马 击中 函数属于 $W[3]$。

证明。设 H = (X, ˆS = $(s_i)_{i\in I}$) 是一个带有对应关系 τ : X → I 的（不一定是简单的）超图，并设 f: X → {0, 1, 2} 是某个函数，构成 ω(f)-扩展罗马击中函数的一个实例。根据引理4，我们可以假设不存在两个元素 x, y ∈X，使得 f (x) = f (y) = 1 且 τ(x) = τ(y)，也不存在 f (x) = 2、f (y) = 1 且 x ∈ $s_{\tau(y)}$。

接下来我们将构造一个等价的多色独立族实例。我们将其论域定义为 $U := X \cup \{x' \mid x \in f^{-1}(\{1,2\})\} \cup \{r_{i,x} \mid x \in f^{-1}(\{1,2\}),\ i \in I\} \cup \{\tau_x \mid x \in f^{-1}(1)\}$。对于超图构造，我们需要定义一些额外的（辅助）集合，其意义将在下一段中解释:

- 对于 $x \in f^{-1}(\{1,2\})$，将 $i \in \mathbf{I}(x)$ 简记为 $\tilde{s}_{x,i} := s_i \cup \{r_{i,x},\ x'\}$。
- 定义 $t_i := s_i \cup \{\tau_x \mid x \in f^{-1}(1) \cap s_i\}$，对于满足 $\tau^{-1}(i)$ 且 $s_i \cap f^{-1}(2)$ 为空的 $i \in I$。
- 对于每个 x ∈f $^{-1}$(2)，令 S x := {˜sx,i | τ(x) ≠i}；对于每个 x ∈f $^{-1}$(1)，令 S x := {s_τ(x) ∪{τx}} ∪ {˜sx,i | τ(x) ≠i}。
- 此外，我们需要目标集合 $T = T' \cup T''$，其中

$$T' := \{ t_i \mid i \in I \wedge \tau^{-1}(i) = \emptyset \wedge s_i \cap f^{-1}(2) = \emptyset \} \quad \text{and}$$

$$T'' := \left\{ \{r_{i,x}, y'\} \mid i \in I \wedge \{x, y\} \subseteq (s_i \cap f^{-1}(\{1,2\})) \wedge x \neq y \right\} \cup \left\{ \{\tau_x, y'\} \mid x \in g^{-1}(1) \wedge y \in s_{\tau(x)} \wedge x \neq y \right\}.$$

现在我们来解释每个元素的含义。请务必牢记，我们要使用可扩展性刻画定理 5：如果元素 x' 属于某条选定的超边，那么我们就在极小

罗马击中函数中为 x 赋值 2。元素 $r_{i,x}$ 为我们提供了有关映射 ρ的信息：当且仅当 $\rho(x) = i$ 成立时，$r_{i,x}$ 才属于一条选定的边。因此，集合 $\{r_{i,x}, y'\}$ 满足定理 5 的约束 2。只有当我们为 x 赋值 1 时，τ_x 才会属于我们选定的某个集合。因此，集合 $\{\tau_x, y\}$将满足约束 3。T′中的集合对应于我们在约束 4 中考虑的集合。这也是将 τ_x 包含在 t_i中的原因。由于存在一个 τ_x，$f(x) = 1$。如果 $x \in R_2$，，那么 $s_i \cap R_2 \neq \emptyset$。在多色独立族实例中，这对应于：$\tau_x$ 不会属于我们的集合，因此 t_i 不会被完全覆盖。

命题。(H, τ, f)是是实例，当且仅当扩展罗马击中函数的$(U, (S_x)_{x\in f^{-1}(\{1,2\})}, T)$是是实例。该实例属于多色独立族问题。

证明。首先，假设 (H, τ, f) 是扩展罗马击中函数的一个是实例，即我们需要证明 $(U, (S_x)_{x\in f^{-1}(\{1,2\})}, T)$ 是多色独立族的一个是实例。为此，我们将使用定理 5（如上所述，先利用引理4简化该实例）。由于 (H, τ, f) 是扩展罗马击中函数的一个是实例，因此存在一个集合 $f^{-1}(2) \subseteq R_2 \subseteq f^{-1}(\{1,2\})$，以及一个映射 $\rho: R_2 \to I$，它们满足定理 5 的全部四个约束。对于每个 $x \in R_2$，，任选一个 $\tilde{s}_{x,\rho(x)} \in S_x$；对于每个 $x \in f^{-1}(1) \setminus R_2$，，任选 $s_{\tau(x)} \cup \{\tau_x\} \in S_x$。我们将证明，从 $(S_x)_{x\in f^{-1}(\{1,2\})}$ 中选出的这些超边表明 $(U, (S_x)_{x\in f^{-1}(\{1,2\})}, T)$ 是多色独立族的一个是实例。反设 T 中存在一条超边，它是以下集合的子集

$$\mathcal{S} := \left(\bigcup_{x\in R_2} \tilde{s}_{x,\rho(x)}\right) \cup \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)} \cup \{\tau_x\}\right).$$

注意，对于每个 $x \in f^{-1}(\{1,2\})$，$x' \in \mathcal{S}$ 当且仅当存在一个 $i \in \mathbf{I}(x)$，使得 $\tilde{s}_{x,i}$ 被选中。因此，x必须属于R_2。

如果T'' 中存在一个包含于 $\mathcal{S}$ 的超边，就会与约束 1 或约束 2 矛盾。现在，考虑T'中的一个超边t_i。因此，存在某个$i \in I$，满足$t_i \subseteq \mathcal{S}$，$s_i \cap f^{-1}(2) = \emptyset$ 以及 $\tau^{-1}(i) = \emptyset$。这意味着

$$s_i \subseteq \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)}\right).$$

此外，$f^{-1}(1) \cap R_2 \cap s_i = \emptyset$，否则将存在一个 $\tau_x \in t_i$，而它不属于 $\mathcal{S}$。这将与约束 4 矛盾。因此，$(U, (S_x)_{x\in f^{-1}(\{1,2\})}, T)$ 是一个是-实例。

对于“若”部分，考虑 MultIndFam 的实例 $(U、(S_x)_{x\in f^{-1}(\{1,2\})}、T)$，以及对于构成该实例解的每个 $x \in f^{-1}(\{1,2\})$，都有 $q_x \in S_x$；也就是说，它们的并集不包含来自 T 的任何超边。定义

$$R_2 := f^{-1}(2) \cup \{x \in f^{-1}(1) \mid \tau_x \notin q_x\}$$

并且，对于每个 $x \in R_2$，，确定 $\rho(x)$，使得 $\tilde{s}_{x,\rho(x)} = q_x$。我们再次使用定理 5 以及刻画定理中提到的四个约束。约束 1 通过构造 R_2 得以满足。接下来的两个约束也得到满足，

否则，一个 T'' 的集合将被包含在 $\bigcup_{x\in f^{-1}(\{1,2\})} q_x$ 中。假设存在一个 $j \in I$，满足 $\tau^{-1}(j) = \emptyset$，并且

$$s_j \subseteq \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)} \right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)} \right).$$

由于对于每个 $i \in I$，t_i 都不是 $\bigcup_{x\in f^{-1}(\{1,2\})} q_x$ 的子集，因此必定存在某个 $\tau_x \in t_i$，使得 $x \in f^{-1}(1) \cap s_j$ 不属于 $\bigcup_{x\in f^{-1}(\{1,2\})} q_x$。于是，$R_2 \cap s_i \neq \emptyset$。因此，约束 4 也得到满足。所以，$(H, f, \tau)$ 是扩展罗马击中函数的一个是实例。

◊

由于 $|U| \le |X| \cdot (|I| + 4)$、$|T| \le |I| + |X|^2 \cdot (|I| + 1)$ 和 $|S_x| \le |I|$，多色独立族可以在多项式时间内构造。此外，所构造的多色独立族实例的参数为 $|f^{-1}(1) \cup f^{-1}(2)| \le \omega(f)$。因此，扩展罗马击中函数属于 W[3]。

□

如前面的证明中所述，上述归约也是一个多项式时间归约。因此，扩展罗马击中函数属于NP，不过根据NP的猜测与检验刻画，这一点也很容易看出。对于困难性结果，我们将使用上一节中描述的有界扩展罗马支配。

定理10。参数化问题 $\omega(f)$-扩展罗马击中函数是W[3]-困难的，即使对应函数是单射。此外，$\omega(f)$-有界-扩展罗马支配是W[3]-完全的。

证明。我们将使用定理 8，从有界扩展罗马支配进行归约。设 (G, f, h) 是有界扩展罗马支配的一个实例，其中 $G = (V, E)$。根据引理 5，我们可以假设 (1) $f \le h$，并且此外，(2) $h(v) = 0$ 蕴含对于某个 $u \in N(v)$，有 $h(u) = 2$。我们以 $\omega(f)$ 为参数。对于 $v \in X := V \setminus h^{-1}(0)$，定义 $T_v := \big(N(v) \setminus h^{-1}(\{0, 1\})\big) \cup \{v\}$；对于 $v \in h^{-1}(0)$，定义 $T_v := \big(N(v) \setminus h^{-1}(\{0, 1\})\big)$。此外，令 $\hat{S} := (T_v)_{v\in V}$，并定义 τ 为满足 $\tau(v) = v$ 的对应关系；令 $f : X \to \{0, 1, 2\}$，$v \mapsto f(v)$，即 $f = f|_X$。令 $H = (X, \hat{S})$。那么，(H, τ, f) 描述了扩展罗马击中函数的一个实例。该实例的参数为 $\omega(f)$。由于 $f \le h$，$h(v) = 0$ 蕴含 $f(v) = 0$。因此，$\omega(f) = \omega(f)$，所以从有界扩展罗马支配实例转移到 $\omega(f)$-扩展罗马击中函数实例时，参数值不会改变。显然，所描述的构造可以在多项式时间内完成。显然，对应关系 τ 是单射。剩下需要证明的是以下命题。

命题。(G, f, h) 是有界扩展罗马支配的是-实例，当且仅当 (H, τ, f) 是扩展罗马命中函数的是-实例。

证明。设 g 是 G 上的一个极小罗马支配函数，满足 $f \le g \le h$。定义 $g : X \to \{0, 1, 2\}, v \mapsto g(v)$，即 $g = g|_X$。因此，$f \preceq g$。令 $v \in V$。若 $g(v) \in \{1, 2\}$，则 T_v 被 v 击中。若 $g(v) = 0$ 成立，则必存在 $w \in N(v)$，使得 $g(w) = 2$。这个 w 会击中 T_v。因此，g 是一个罗马击中函数。假设 g 不是极小的。由于 τ 是单射，定理 4 的约束0成立。因此，定理 4 的约束1、约束2或约束3的极小性必有一项被违背

。如果约束1被违背，这意味着存在$v,w \in V$ with$g(v) = g(v) = 2, g(w) = g(w) = 1$ and $v \in T_w = \left(N(w) \setminus h^{-1}(\{0, 1\})\right) \cup \{w\} \subseteq N[w]$。这与$g$ 的极小性矛盾，因为我们可以令$g(w) = 0$，同时仍得到一个罗马支配函数。

考虑约束 2。假设存在一个 $v \in g^{-1}(2)$，使得对每个 $w \in V \setminus \{v\}$，$T_w \cap g^{-1}(2) \neq \{v\}$ 成立。因此，对每个 $w \in N(v)$，都存在一个 $u \in (T_w \cap N[w] \cap g^{-1}(2)) \setminus \{v\} \subseteq (N[w] \cap g^{-1}(2)) \setminus \{v\}$。这意味着 g 不是极小的。

关于约束 3，考虑 $\mathcal{T} := \{T_v \mid v \in V, \tau^{-1}(v) \cap g^{-1}(1) = \emptyset\} = \{(N(v) \cap h^{-1}(2)) \cup \{v\} \mid v \in V, g(v) \neq 1\}$。若 $g(v) = 2$，则 v 显然击中 $(N(v) \cap h^{-1}(2)) \cup \{v\}$。若 $g(v) = 0$，则存在某个 $w \in N(v)$，使得 $g(w) = 2$，因为 g 是 G 的罗马支配函数。若 $g(w) = 2$，则 $h(w) = 2$，因此 v 击中 $(N(v) \cap h^{-1}(2)) \cup \{v\}$。所以，$g^{-1}(2)$ 是 $\mathcal{T}$ 的一个击中集。假设存在某个 $z \in g^{-1}(2)$，使得 $g^{-1}(2) \setminus \{z\}$ 仍是 $\mathcal{T}$ 的一个击中集。因此，对于每个满足 $z \in T_v$ 的 T，都存在某个 $y \in g^{-1}(2) \setminus \{z\}$，使得 $y \in T_v$。若 $v = z$，则 $y \in (N(v) \cap h^{-1}(2))$。因此 $h(y) = 2$，进而 $g(y) = 2$。在这种情况下，由于 $y \in N(z)$，我们可以令 $g(z) = 0$，这可能会得到一个更小的罗马支配函数。现在假设 $v \neq z$。那么，$z \in (N(v) \cap h^{-1}(2))$。考虑 $y \in T_v$。若 $y = v$，我们同样可以论证令 $g(z) = 0$ 可能会得到一个更小的罗马支配函数，因为我们找到了某个 $y \in N(z)$，使得 $g(y) = 2$。若 $y \neq v$，则 $\{y, z\} \subseteq (N(v) \cap h^{-1}(2))$。由于 $h(v) = 2$ 意味着 $g(v) = 2$，因此在这第三种情况下，我们也可以令 $g(z) = 0$。总之，在每种情况下，我们都可以通过令 $g(z) = 0$，找到一个比 g 更小的 G 的罗马支配函数。这与 g 的极小性矛盾。根据定理 4，g 是极小的。

设g是一个极小罗马击中函数，是实例（H，τ，f）的一个解。定义

$$g: V \to \{0, 1, 2\}, v \mapsto \begin{cases} \overline{g}(v), & \text{if } v \in X, \text{ i.e., if } h(v) \neq 0, \\ 0, & \text{if } v \notin X, \text{ i.e., if } h(v) = 0. \end{cases}$$

由于 g 扩展了 f，因此 $f \leq g$。根据构造，$f \leq h$，因此对 $v \notin X$，有 $f(v) = 0$。于是 $f \leq g$。考虑某个 $y \in h^{-1}(1)$。我们证明此时 $g(y) \leq 1$，从而证明 $g \leq h$。由于 $y \in h^{-1}(1)$，在 $\hat{S}$ 中的所有集合里，只有集合 $T_y = T_{\tau(y)}$包含 y。若 $g(y) = 2$，即 $g(y) \equiv 2$，这将与定理 4 的约束 2 矛盾。现在证明 g 是一个罗马支配函数。考虑某个 $v \in g^{-1}(0) = g^{-1}(0) \cup h^{-1}(0)$。若 $v \in h^{-1}(0)$，则必定存在一个 $u \in g^{-1}(2) \cap T_v \subseteq g^{-1}(2) \cap N[v]$，因为不存在满足 $\tau(u) = v$ 的 $u \in X$。若 $v \in g^{-1}(0)$，则 $\tau^{-1}(v) \cap g^{-1}(1)$ 中没有顶点，因为 $\tau(v) = v$。因此，存在一个 $u \in g^{-1}(2) \cap T_v \subseteq g^{-1}(2) \cap N[v]$。

接下来，我们说明为什么 g 是极小的。假设存在 v、$w \in V$，使得 $g(v) = \overline{g}(v) = 2$，$g(w) = g(w) = 1$ 且 $\{v, w\} \in E$，这违反了定理 1 的第一条性质。这会与 g 的极小性矛盾，因为 $v \in \left(N(w) \setminus h^{-1}(\{0, 1\})\right) \cup \{w\} = T_w$，而 $g(v) = 2$ 意味着由 $g \leq h$ 可知 $h(v) = 2$。假设存在 $v \in g^{-1}(2) = \overline{g}^{-1}(2)$，使得 $N[v] \setminus N[g^{-1}(2) \setminus \{v\}] \subseteq \{v\}$，这违反了定理 1 的隐私性质。因此，对于每个 $w \in N(v)$，都存在 $t \in (N[w] \cap g^{-1}(2)) \subseteq (N(w) \setminus h^{-1}(\{0, 1\})) \cup \{w\}$，且 $t \neq v$。这与定理 4 的约束 2 矛盾。考虑 $G' := G\left[g^{-1}(0) \cup g^{-1}(2)\right]$。由于 g 是罗马支配函数，$D := g^{-1}(2)$ 是 G' 的支配集。若 D 不是极小支配集（因而违反定理 1 的最后一项约束），则必定存在某个 v，使得 $g(v) = 2$，且 $D \setminus \{v\}$ 也是一个

支配集。因此，对于每个 $u \in N[v]$,，都存在某个 $v_u \in N[u] \cap (D \setminus \{v\})$。根据构造，对于每个这样的 v_u，都有 $g(v_u) = 2$，从而 $g(v_u) = h(v_u) = 2$，且 $T_{v_u} \neq \{v_u\}$ (*)。另一方面，由于 g-是极小罗马击中函数，根据定理 4 的最后一项约束，$D = g^{-1}(2) = g^{-1}(2)$ 是 $\mathcal{T} \coloneqq \{T_x \mid x \in V \setminus g^{-1}(1)\}$ 的极小击中集。由于对所有 $x \in V$ 都有 $T_x \subseteq N[x]$，$(D \setminus \{v\})$ 不仅是 G' 的支配集，也是 $\mathcal{T}$ 的击中集（这是由 (*) 得出的），这与定理 4 所述的极小性矛盾。既然已经证明，定理 1 中表述的极小罗马支配函数刻画的所有可能违反情形都会与我们的假设矛盾，我们必须得出结论：g 确实是一个极小罗马支配函数。◊

我们的证明到此结束。 □

由于这也是一个多项式时间归约，因此可推出以下推论。

推论 5. 扩展罗马击中集函数是*NP*-完全的。

我们已经知道，当对应函数是满射时，扩展罗马击中函数可以在多项式时间内求解。由于对应函数为单射的扩展罗马击中函数，难度不亚于一般的击中集问题，因此引出了这样一个问题：对于每个实例 $\mathcal{I} = (H, \tau, f)$，其中 $H = (V, (s_i)_{i \in I})$，$\tau : V \to I, f : V \to \{0, 1, 2\}$，$\kappa_1(\mathcal{I}) = |\{i \in I | \tau^{-1}(i) = \emptyset\}|$ 能否成为该问题的一个良好参数，并以某种方式再次衡量与平凡性的距离？换言之，我们尝试利用参数化复杂性来研究这样一种现象：诸如满射性这样的经典函数性质，似乎是找到扩展罗马击中函数多项式时间算法的关键。

定理 11. κ_1-扩展 罗马 击中 函数 是*W*[3]-困难问题。

证明。 注意，定理 10 的证明中的归约也是一个固定参数可处理归约，它将参数化问题 $\kappa_{h^{-1}(0)}$-有界扩展罗马支配归约到 κ_1-扩展罗马击中函数，所用参数为$\kappa_{h^{-1}(0)}(G, f, h) = |h^{-1}(0)|$，因此我们可以利用定理 7 来证明这一命题。

□

定理 12. κ_1-扩展 罗马 击中 函数和 $\kappa_{h^{-1}(0)}$-有界扩展 罗马 支配均为*W*[3]-完全问题。

证明。对于归属性，我们再次使用多色独立族问题。设 $H = (X, (s_i)_{i \in I})$ 是一个超图，其对应函数为 $\tau : X \to I$，且有两个函数 $f : X \to \{0, 1, 2\}$。我们假设已将引理 2 和引理 3 对 f 穷尽地应用。定义关系 $R = \{(x, y) \in X^2 \mid y \in f^{-1}(1), x \in s_{\tau(y)}\}$。令 R^* 为其传递闭包，并令 $R_x \coloneqq \{x\} \cup \{y \in X \mid (x, y) \in R_x\}$。通过对引理 2 作归纳论证可以证明：对于任意极小罗马击中函数 $g : X \to \{0, 1, 2\}$，若 $f \leq g$ 且 $g(x) = 2$，则对所有满足 $y \in R_x$ 的 y，也有 $g(y) = 2$。

我们定义 $U \coloneqq \{\eta_v \mid v \in f^{-1}(\{0, 1\})\} \cup \{u_{x,i} \mid x \in X, i \in I(x)\}$ 以及 $A \coloneqq \{i \in I \mid \tau^{-1}(i) = \emptyset\}$。对于 $x \in f^{-1}(\{0, 1\})$，定义 $W_x \coloneqq \{u_{y,i} \mid i \in I(y)$，其中 $y \in s_i \setminus \{x\}\}$，且 $r_x \coloneqq \bigcup_{y \in R_x}(\{\eta_y\} \cup W_y)$。此外，对于每个 $x \in X$，我们还需要集合 t_x。对于 $x \in f^{-1}(2)$，令 $t_x \coloneqq \{u_{x,i} \mid i \in I(x) \setminus \left(I(f^{-1}(2) \setminus \{x\}) \cup \{\tau(x)\}\right)\}$。如果 $x \notin f^{-1}(2)$，那么

$t_x := \left\{\eta_x\right\} \cup \{u_{x,i} \mid i \in I(x) \setminus \left(I(f^{-1}(2)) \cup \{\tau(x)\}\right)\}$。现在我们可以定义多色独立族实例 $(S\,i)_{i\in A\setminus I(f^{-1}(2))} \times T$，其中 $T := \{t_x \mid x \in X\}$，且对于 $i \in A \setminus I(f^{-1}(2))$，$S\,i := \{r_x \mid x \in s_i\}$。

现在我们来解释各个元素的含义：集合R_x 包含所有必须取值为 2 的顶点（根据约束 3），前提是x 取值为 2。顶点x 取值为 1，当且仅当 η_x 属于某个选定的超边。如果$u_{x,i}$ 不属于所选超边，那么s_i 不包含任何取值为 2 的元素（可能的x 除外）。因此，t_x 表示x 可能拥有的私有边（除 η_x外）。r_x 描述了x 取值为 2 时将被覆盖的边（可能的η_y除外）。

命题。 $(S_i)_{i\in A\setminus I(f^{-1}(2))} \times T$ 是是-实例，当且仅当(H, τ, f)是是-实例。

证明。设 g 是 H 上相对于 τ满足 $f \le g$ 的一个极小罗马击中函数。因此，对于每个 $i \in A$，都存在一个 $x_i \in s_i \cap g^{-1}(2)$。对于每个 $i \in A' := A \setminus I(f^{-1}(2))$，从 $S\,i$ 中选取 r_{x_i}。假设存在一个 $x \in X$，使得 $t_x \subseteq \bigcup_{i\in A'} r_{x_i}$。若 $x \in f^{-1}(\{0,1\})$，则$\eta_x \in \bigcup_{i\in A'} r_{x_i}$。因此，存在一个 $i \in A'$，使得 $(x_i, x) \in R_{x_i}$。注意，$(x_i, x) \in R_{x_i}$ 与 $g(x_i) = 2$ 合起来意味着 $g(x) = 2$。此外，对于每个满足 $i \in I(y) \setminus (I(f^{-1}(2)) \cup \{\tau(y)\})$ 的 $u_{y,i}$，存在一个 $j \in A'$ 和一个 $z \in R_{x_j}$，使得 $u_{z,i} \in w_z$。这意味着在 g 下，x 没有私有超边，因此 g 不是极小的。于是，$t_x \not\subseteq \bigcup_{i\in A'} r_{x_i}$。对于 $x \in f^{-1}(2)$，同理可证。

现在，令 $(r_{x_i})_{i\in A'}$ 为多色独立族实例的一个解。定义

$$B := \{x \in A' \mid \eta_x \in \bigcup_{i\in A'} r_{x_i}\} \qquad \text{and}$$

$$g\colon X \to \{0,1,2\}, \quad x \mapsto \begin{cases} 2, & \text{if } x \in B, \\ f(x), & \text{otherwise}. \end{cases}$$

我们要证明 $P := g^{-1}(2) = B \cup f^{-1}(2)$ 满足定理 5 的条件。显然，$g^{-1}(2) \subseteq P \subseteq g^{-1}(1) \cup g^{-1}(2)$。根据 P 的定义，对所有 $x \in P$，$\eta_x \notin t_x \setminus \bigcup_{i\in A'} r_{x_i}$（如果存在）。因此，对每个 $x \in P$，都存在一个 $i_x \in I(x) \setminus (I(f^{-1}(2) \setminus \{x\}) \cup \{\tau(x)\})$，使得 $u_{x,i_x} \in t_x \setminus \bigcup_{i\in A'} r_{x_i}$。现在定义 $\rho\colon B \to I, x \mapsto i_x$。这已经意味着 $\rho(x) \neq \tau(x)$，且 $x \in s_{\rho(x)} \cap P$。假设存在 $y \in (s_{\rho(x)} \cap P) \setminus \{x\}$。因此，要么 $y \in f^{-1}(2)$ $(i_x \notin I(x) \setminus (I(f^{-1}(2) \setminus \{x\}) \cup \{\tau(x)\}))$，要么 $\eta_y \in \bigcup_{i\in A'} r_{x_i}$。在后一种情况下，$u_{x,i_x} \in w_y \subseteq \bigcup_{i\in A'} r_{x_i}$。因此，不存在这样的 y，且 $(s_{\rho(x)} \cap P) = \{x\}$。

假设存在一个 $x \in g^{-1}(1) \setminus P = g^{-1}(1) \subseteq f^{-1}(1)$，且 $s_{\tau(x)} \cap R_2 = \emptyset$。那么，$\tau(x) \in I(f^{-1}(2))$，或者存在一个 $y \in s_{\tau(x)}$，满足 $\eta_y \in \bigcup_{i\in A'} r_{x_i}$。因此，我们要么可以使用引理 2，要么 $x \in R_y$。在这两种情况下都会产生矛盾，因此不存在这样的 x。根据 B 和 A' 的定义，对于所有 $A \subseteq I(P)$。因此，条件 4 得到满足，且 g 是可扩展的。 ◊

由于 $|A'| \le |A|$，这是一个固定参数可处理-归约。 □

现在，我们考虑一个用于 κ_1-扩展罗马命中函数的显式XP-算法；属于XP 已可由定理12推出。

引理6. 以下问题属于*XP*：

- κ_1-扩展罗马击中函数，
- $\kappa_{h^{-1}(0)}$-有界扩展罗马支配，以及
- κ_{2-h}-有界扩展罗马支配。

证明。对于 κ_1-扩展罗马击中函数和$\kappa_{h^{-1}}(0)$-有界扩展罗马支配，只需证明 κ_1-扩展罗马击中函数 $\in$属于XP。设 (H, τ, f)是一个实例，其中$H = (X, \hat{S})$，$\hat{S} = (s_i)_{i \in I}$。这个XP算法并不难设计：我们对 $\{i \in I \mid \tau^{-1}(i) = \emptyset\}$ 中的每个指标进行分支，并尝试用X中的一个元素击中边s_i。如果每条边都被击中，我们就利用算法1检查是否存在一个大于该函数的极小罗马击中函数。如果算法1返回是，我们也可以返回是。否则，我们尝试用另一种方式击中$S \setminus \tau(X)$中的所有边。该算法的运行时间为 $O(n^{\kappa_1}(H,\tau, f))$。

现在我们考虑 κ_{2-h}-有界扩展罗马支配。设 (G, f, h) 是一个实例，其中 $G = (V, E)$，且 $f \leq h$。首先，我们希望对每个顶点 $x \in N(f^{-1}(2)) \cap f^{-1}(1)$，令 $g(x) := 2$。然而，如果集合 $N(f^{-1}(2)) \cap f^{-1}(1) \cap h^{-1}(\{0, 1\})$ 非空，那么我们就遇到了一个平凡的否实例：对于我们要寻找的解 g，任何满足 $f(x) = 2$ 的顶点 x 都满足 $g(x) = 2$；此外，若顶点 $y \in N(x)$ 满足 $f(y) = 1$，则 g 必须将其设为 2，但如果 $h(y) \in \{0, 1\}$，这就不可能实现。由于 $f \leq g \leq h$，对于 $x \in f^{-1}(1) \cap h^{-1}(1)$，我们可以令 $g(x) := 1$。完成这些设置后，XP算法会对 $h^{-1}(0)$ 中的每个顶点进行分支，并尝试用 $h^{-1}(2) \setminus N(f^{-1}(1) \cap h^{-1}(1))$ 中的一个邻点支配它。由于 $2 \cdot |h^{-1}(0)| \leq \omega(2 - h)$，我们总共至多有 $n^{\frac{\omega(2-h)}{2}}$ 种可能的设置方式。对于 g 的每一种可能设置，我们将此前尚未定义的任意顶点 x 设为 $g(x) := f(x)$，然后运行 [4]，中用于扩展罗马支配的多项式时间算法，并以 g 作为下界函数。该算法能够正常运行，因为 $h^{-1}(0)$ 中的每个顶点都已被支配（未被支配的顶点被设为 1），并且 $N(f^{-1}(2)) \cap f^{-1}(1) \cap h^{-1}(1)$ 中不存在顶点。

□

下面我们将讨论另外几种参数化，它们要么导致para-NP困难性结果（见定理 13），要么导致固定参数可处理结果（见定理 14）。

定理 13. κ_ζ-扩展 罗马 命中 函数 对每种由 $\zeta \in \{|f^{-1}(0)|, |f^{-1}(1)|, |f^{-1}(2)|, |f^{-1}(\{0, 2\})|\}$ 描述的参数化而言，都是*para-NP*困难的。

证明。对于本证明中展示的所有归约，我们都使用扩展击中集的 NP 完全性。设 $H = (X, \hat{S} = (s_i)_{i \in I})$ 是一个超图，其中 $U \subseteq X$，$I \cap X = \emptyset$，且元素 a、i_b、$i_c \notin X \cup I$。我们定义 $X' = X \cup \{a\}$，$I' = I \cup X \cup \{i_b, i_c\}$。对于每种参数化，我们证明，即使参数取值为零或一，从而固定不变，所得到的每个问题仍可用于求解扩展击中集的任意实例。

首先，我们考虑参数 κ_ζ，其中 $\zeta \in \{f^{-1}(0)、f^{-1}(2)、f^{-1}(\{0,2\})\}$。定义超图 $H' = (X', (s'_i)_{i\in I'})$，以及对应关系 τ，其中

$$\tau : X' \to I', x \mapsto \begin{cases} x, & \text{if } x \in X. \\ i_b, & \text{if } x = a, \end{cases} \qquad s'_i := \begin{cases} s_i, & \text{if } i \in I, \\ \{a, i\}, & \text{if } i \in U, \\ \{i\}, & \text{if } i \in X \setminus U, \\ X', & \text{if } i = i_b, \\ \{a\}, & \text{if } i = i_c, \end{cases}$$

对于$i \in I'$。此外，我们需要

$$f : X' \to \{0,1,2\}, x \mapsto \begin{cases} 2, & \text{if } x = a, \\ 1, & \text{if } x \in X. \end{cases}$$

这意味着 $|f^{-1}(\{0,2\})| = 1 = |f^{-1}(2)|$，且 $|f^{-1}(0)| = 0$，也就是说，参数值是常数。

命题。(H, U) 是是实例，当且仅当扩展 击中 集 (H', τ, f) 是是实例，当且仅当它是扩展罗马击中 函数的一个实例。

证明。首先，我们穷尽地应用引理 2。由于对 $x \in U$，$f(a) = 2$ 且 $a \in s_{\tau(x)}$，因此，(H', τ, f) 是扩展罗马击中函数的一个是实例，当且仅当 (H', τ, f) 满足：对 $x \in U \cup \{a\}$，$f(x) = 2$；对 $x \in X \setminus U$，$f(x) = 1$；并且它是扩展罗马击中函数的一个是实例。假设存在 (H', τ) 的一个极小罗马击中函数 g，满足 $\tilde{f} \leq g$。定义 $D = g^{-1}(2) \setminus \{a\}$。由于对于每个 $i \in I$，$a \notin s'_i = s_i$（即 $\tau^{-1}(i) = \emptyset$），对于这样的 i，有 $s_i \cap D \neq \emptyset$。如果存在 $x \in D$，使得对于每个 $i \in I(x)$，都存在 $y \in s_i \cap (D \setminus \{x\})$，那么这将与定理 4 的约束 2 矛盾。因此，$D$ 是一个极小击中集。

假设 H 上存在一个极小击中集 $D \subseteq X$，且 $U \subseteq D$。定义 $g \in \{0,1,2\}^{X'}$，其中对于 $x \in D \cup \{a\}$，$g(x) = 2$；对于 $x \in X \setminus D$，$g(x) = 1$。我们使用定理 4 证明 g 是一个极小罗马击中函数。由于对每个 $x \in g^{-1}(1) = X \setminus D \subseteq X \setminus U$，都有 $s_{\tau(x)} = \{x\}$，因此前两个约束成立。D 是 H 上的极小击中集这一事实，以及 $s_{i_c} \cap g^{-1}(2) = \{a\}$，蕴含约束 2。约束 3 成立，因为 D 是 H 上的击中集，且超边 $(s'_i)_{i\in I'} \setminus I$ 被 $\{a\} \cup (X \setminus D)$ 击中。因此，g 是 H' 的极小罗马击中函数。◊

接下来，我们来看看由 $|f^{-1}(1)|$ 描述的参数。定义超图 $H'' = (X', (s''_i)_{i\in I'})$。此外，

我们定义$s_i''$for$i \in I'$对应的超边如下：

$$s_i'' \coloneqq \begin{cases} s_i, & \text{if } i \in I, \\ \{a, i\}, & \text{if } i \in X, \\ X', & \text{if } i = i_b, \\ \{a\}, & \text{if } i = i_c. \end{cases}$$

定义$f'\colon X' \to \{0,1,2\}$，其中$f'^{-1}(2) = U \cup \{a\}$，且$f^{-1}(0) = X \setminus U$。因此，$|f'^{-1}(1)| = 0$，即参数值为常数。

命题。(H, U)是扩展H击中集的一个是-实例，当且仅当(H'', τ, f')是扩展罗马击中函数的一个是-实例。

证明。首先，假设H'', τ上存在一个极小罗马击中函数g，且满足$f' \le g$。定义$D \coloneqq g^{-1}(2)\backslash\{a\}$。由于对于每个$i \in I$，$s_i \cap D$非空，且D是H的一个击中集，因此$\tau^{-1}(i) = \emptyset$。如果存在一个$x \in D$，使得对于每个$i \in I(x)$，都存在一个$y \in s_i \cap (D \setminus \{x\})$，那么这将与定理4的约束2矛盾。因此，D是一个极小击中集。

假设H上存在一个极小击中集$D \subseteq X$，且$U \subseteq D$。定义$g : X \to \{0,1,2\}$，其中$g^{-1}(2) = D \cup \{a\}$，且$g^{-1}(0) = X \setminus D$。显然，$f' \le g$成立。我们将使用定理4证明g是一个极小罗马击中函数。由于$g^{-1}(1)$为空，前两个约束，即约束0和约束1，均得到满足。由于D是H的一个极小击中集，且H的每条超边也是H'的超边，因此约束2也成立。对于最后一个约束，只需考虑$i \in I$，因为$g(a) = 2$。由D是H上的一个极小击中集，且s_{i_c}''仅被a击中，可得定理4的最后一个约束成立。因此，g是一个极小罗马击中函数。

◊

至此，所有参数的讨论均告结束。 □

最后，我们介绍一些基于不同参数化的（相当直接的）算法结果。

定理 14. *κ_ζ-扩展罗马击中函数 $\in$固定参数可处理 对于 $\zeta \in \{\omega(2-f), |f^{-1}(\{0,1\})|\}$。*

证明。 我们将给出一个简单的固定参数可处理算法。设$(H = (X, \hat{S}), \tau, f)$是一个实例。其思路是遍历所有函数$g\colon X \to \{0,1,2\}$，满足$f \le g$，并检验$g$是否为极小罗马击中函数。由于对于顶点$x \in X$，若$f(x) = 0$，则有3种选择；若$f(x) = 1$，则有2种选择；而若$f(x) = 2$，则只有1种选择，因此该过程可在固定参数可处理时间内运行。此外，我们可以在多项式时间内检验一个函数是否为极小罗马击中函数（通过以直接的方式修改算法1即可构造出这样的算法）。因此，函数<style id='g'>g有

$$\begin{aligned} \left(\prod_{x \in f^{-1}(0)} 3\right) \cdot \left(\prod_{x \in f^{-1}(1)} 2\right) &\le \left(\prod_{x \in f^{-1}(0)} 2^{2-f(x)}\right) \cdot \left(\prod_{x \in f^{-1}(1)} 2^{2-f(x)}\right) \\ &= 2^{\omega(2-f(x))} \end{aligned}$$

or

$$\left(\prod_{x\in f^{-1}(0)} 3\right)\cdot\left(\prod_{x\in f^{-1}(1)} 2\right) \leq \left(\prod_{x\in f^{-1}(0)} 3\right)\cdot\left(\prod_{x\in f^{-1}(1)} 3\right) = 3^{|f^{-1}(\{0,1\})|}$$

许多种可能性。 □

8. 枚举极小罗马命中函数

本节讨论枚举极小罗马命中函数的输出敏感复杂度。我们首先证明，枚举极小罗马命中函数是横截困难的。随后，我们通过证明其逆向命题来补充横截困难性结果：如果能够以多项式延迟枚举任意超图的所有极小击中集，那么也能够以多项式延迟枚举任意实例 (H,τ) 的所有极小罗马命中函数。结合横截困难性定理，这便确立了这两个枚举问题之间的多项式延迟等价性。此外，我们在证明逆向命题时采用的技巧，经过一些修改和进一步分析后，还可以证明：存在一种FPT延迟算法，能够枚举以 κ_1 为参数的任意实例 (H,τ) 的极小罗马命中函数。

由这些结果可推得一个推论：如果对应函数是满射，那么就能够以多项式延迟枚举极小罗马命中函数。

回顾一下，对于给定的超图 H 和对应函数 τ，$\mu-\mathcal{RHF}(H)$ 表示相对于H 和 τ的极小罗马击中函数集合。

定理 15（横截集难题定理）。 如果存在一种算法，能够以多项式延迟枚举任意实例（H， τ）的所有极小罗马击中函数（即使 τ 是单射），那么就存在一种算法，能够以多项式延迟枚举任意超图 H' 的所有极小击中集。

证明。 设 $H=(X,(s_i)_{i\in I})$ 是一个超图。我们将对超图 $H''=(X',\{s''_i\}_{i\in I'})$ 使用与定理 13 的证明中相同的构造。回顾以下记号：$X':=\{a\}\cup X$，$I':=\{i_b,i_c\}\cup X\cup I$，

$$\tau':X'\to I', x\mapsto\begin{cases}x, & x\in X\\ i_b, & x=a\end{cases},\quad \forall i\in I': s''_i:=\begin{cases}s_i, & \text{if } i\in I,\\ \{a,i\}, & \text{if } i\in X,\\ X', & \text{if } i=i_b,\\ \{a\}, & \text{if } i=i_c.\end{cases}$$

由于 $\tau^{-1}(i_c)=\emptyset$ 且 $s''_{i_c}=\{a\}$，对于每个罗马击中函数 $f:X'\to\{0,1,2\}$，都有 $f(a)=2$。因此，对于每个极小罗马击中函数 $f:X'\to\{0,1,2\}$，以及 $x\in X$，$f(x)\neq 1$，否则将与

定理 4 的约束 1 矛盾。因此，对于 H'' 的每个极小罗马击中函数 f，$f^{-1}(1)$ 为空集。

我们将证明，在 H 的极小击中集与 H'' 的极小罗马击中函数之间存在一个双射函数 T。该函数将每个极小击中集 D 映射为 $g : X' \to \{0, 1, 2\}$，其中 $g^{-1}(0) = X \setminus D$，$g^{-1}(2) = \{a\} \cup D$。在定理 13 的证明中，我们已经证明 g 是 H'' 的一个极小罗马击中函数。逆函数将一个极小罗马击中函数 g 映射为 $g^{-1}(2) \setminus \{a\}$；根据定理 13 的证明，这个集合是一个极小击中集。由于对于极小击中集和极小罗马击中函数 g，我们分别可以在多项式时间内计算 $T(D)$ 和 $T^{-1}(g)$，因此定理得证。

□

现在我们定义一些术语，供本节后续定理使用。若指标 $i \in I$ 满足 $\tau^{-1}(i) \neq \emptyset$，则称其为已拥有；否则称为未拥有。我们记作 $I_{\text{own}} := \tau(X)$ 和 $I_{\text{un}} := I \setminus \tau(X)$。

Fo对于超图 $H = ((X, (s_i)_{i \in I})$ 和对应函数 τ，定义 $P_{H,C}(x) := \mathbf{I}(x) \setminus \mathbf{I}(C \setminus \{x\})$ a nd

$$\mathcal{R}(H, \tau) := \{C \subseteq X \mid \{i \in I \mid \tau^{-1}(i) = \emptyset\} \subseteq \mathbf{I}(C) \wedge \forall x \in C : P_{H,C}(x) \nsubseteq \{\tau(x)\}\}.$$

集合 $P_{H,C}(x)$ 由相对于 x 而言对 C 私有的超边的指标组成；也就是说，这些边仅在 x 处被 C 击中。我们将每个集合 $C \in \mathcal{R}(H, \tau)$ 称为 (H, τ) 的罗马支撑。等价地，罗马支撑会击中每条未归属超边，并且其中的每个顶点都有一条私有超边，其指标不同于该顶点所拥有的指标。这里，“支撑集”特指集合 $f^{-1}(2)$，而不是函数的普通支撑集 $f^{-1}(\{1, 2\})$。根据下面的引理得出的推论 6告诉我们，罗马支撑是极小罗马击中函数的必要支撑集连通分量。

引理 7. 设 $H = (X, (s_i)_{i \in I})$ 是一个超图，τ 是一个对应函数。那么 $\Psi_H : \mu - \mathcal{RHF}(H) \to \mathcal{R}(H, \tau)$，$f \mapsto f^{-1}(2)$ 是一个满射函数，并且对于所有 $C \in \mathcal{R}(H, \tau)$，$\Psi_H^{-1}(C)$ 都可以用多项式延迟和多项式空间枚举。

证明。根据定理 4，Ψ_H 是良定义的。设 $\mathrm{C} \in \mathcal{R}(\mathrm{H}, \tau)$。令 $(\mathrm{x}_i)_{i \in I \setminus \mathbf{I}(C)}$ 为一个序列，使得对于所有 $\mathrm{i} \in \mathrm{I} \setminus \mathrm{I(C)}$，$\mathrm{x}_i \in \tau^{-1}(\mathrm{i})$。定义 $\mathrm{f}_C := 2\chi_C + \chi_{\{x_i \mid i \in \mathrm{I} \setminus \mathrm{I(C)}\}}$。根据定理 4，$\mathrm{f}_C$ 是极小的，因为对于每个 $\mathrm{i} \in \mathrm{I} \setminus \mathrm{I(C)}$，恰有一个元素 $\mathrm{x}_i \in \mathrm{f}_C^{-1}(1)$。因此，$\Psi_H$ 是满射的。

现在，我们要证明，对于所有 $\mathrm{C} \in \mathcal{R}(\mathrm{H}, \tau)$ 和 $\mathrm{f} \in \Psi_H^{-1}(\mathrm{C})$，都存在一个序列 $(\mathrm{x}_i)_{i \in I \setminus \mathbf{I}(C)}$，其中 $\mathrm{x}_i \in \tau^{-1}(\mathrm{i})$ 对所有 $\mathrm{i} \in \mathrm{I} \setminus \mathrm{I(C)}$ 均成立，使得 $\mathrm{f} = 2\chi_C + \chi_{\{x_i \mid i \in \mathrm{I} \setminus \mathrm{I(C)}\}}$。由于 f 是一个罗马击中函数，对于每个 $\mathrm{i} \in \mathrm{I} \setminus \mathrm{I(C)}$，都存在一个 $\mathrm{y}_i \in \tau^{-1}(\mathrm{i})$。因此，$\mathrm{f}' = 2\chi_C + \chi_{\{y_i \mid i \in \mathrm{I} \setminus \mathrm{I(C)}\}} \leq \mathrm{f}$。显然，$\mathrm{f}'$ 是一个罗马击中函数。因此，$\mathrm{f} = \mathrm{f}'$，并且存在一个序列 $(\mathrm{y}_i)_{i \in I \setminus \mathbf{I}(C)}$，满足 $\mathrm{f} = 2\chi_C + \chi_{\{y_i \mid i \in \mathrm{I} \setminus \mathrm{I(C)}\}}$。请记住，对于所有 $\mathrm{i} \in \mathrm{I} \setminus \mathrm{I(C)}$，我们可以从 $\tau^{-1}(\mathrm{i})$ 中任意选择元素组合。因此，可以在多项式时间和至多二次空间内构造下一个组合。由此，我们可以以多项式延迟和多项式空间枚举 $\Psi_H^{-1}(\mathrm{C})$。

□

由于 Ψ_H 是一个定义良好的满射映射，我们得到以下推论。

推论 6. 设 $H = (X,(s_i)_{i\in I})$ 是一个超图，τ 是一个对应函数，且 $C \subseteq X$。那么，存在一个极小罗马击中函数 $f \in \{0,1,2\}^X$，满足 $f^{-1}(2) = C$，当且仅当 $C \in \mathcal{R}(H,\tau)$。

定理 16。 如果可以用多项式延迟枚举任意超图的极小击中集，那么就可以用多项式延迟枚举每个实例(H,τ) 的极小罗马击中函数。

证明。 固定任意超图$H = (X,(s_i)_{i\in I})$以及任意对应关系 $\tau : X \to I$。定义$H_{\text{un}} := (X,(s_i)_{i\in I_{\text{un}}})$。

命题 (删除性质)。 设 B 是一个罗马支撑，且 $C \subseteq B$。若 C 击中每条未归属超边，则 C 是一个罗马支撑。

证明。 根据假设，C 击中每条未归属超边。为了验证使一个集合成为罗马支撑所需的额外私有超边条件，任取$x \in C$。于是$x \in B$，并且由于B是一个罗马支撑，存在某个指标$i_x \in I \setminus \{\tau(x)\}$，使得$s_{i_x} \cap B = \{x\}$。由$C \subseteq B$且$x \in C$，可得$s_{i_x} \cap C = \{x\}$。$i_x \neq \tau(x)$成立，这是由 i_x 的取值集合的性质直接推出的。因此，C 满足成为罗马支撑所需的所有条件。◊

命题。 H_{un} 的极小击中集恰好是 $\mathcal{R}(H,\tau)$ 中 (H,τ) 的包含极小罗马支撑集（若不存在 $C \in \mathcal{R}(H,\tau)$ 使得 $C \subsetneq B$），则罗马支撑集 B (H,τ) 在 $\mathcal{R}(H,\tau)$ 中是包含极小的。

证明。 我们通过证明相关的两个包含关系，来说明 H_{un} 的所有极小击中集构成的集合等于 (H,τ) 的所有包含极小罗马支撑集构成的集合。

对于正向包含，设 T 是 H_{un} 的一个极小击中集。由于 T 是 H_{un} 的击中集，它击中每个满足 s_i with i $\in I_{\text{un}}$ 的超边。现在任取 x $\in$ T。根据 T 的极小性，T $\setminus$ {x} 无法击中某个满足 s_i with i $\in I_{\text{un}}$ 的未归属超边。于是有 $s_i \cap$T = {x}：x 必须属于这个交集，因为正是缺少 x 才导致 T $\setminus$ {x} 无法击中 s_i，这意味着 x $\in s_i$；此外，$s_i \cap$T 不可能包含 x 以外的元素，否则 T $\setminus$ {x} 就确实会击中 s_i。关键的是，i $\in I_{\text{un}} = I \setminus \tau(X)$，因此我们有 i $\neq \tau$(x)。于是 i $\in P_{H,T}$(x) $\setminus \{\tau(x)\}$，且 T 是一个罗马支撑集。T 必须是包含极小罗马支撑集，因为 T 的任何真子集都会无法击中 H_{un}的某条超边。

现在令B为一个包含极小的罗马支撑集。根据定义，B是H_{un}的击中集。反设B 不是H_{un}的极小击中集。那么存在一个集合$C \subsetneq B$，它是H_{un}的击中集。根据前面的删除命题，C 是一个罗马支撑集，这与B的包含极小性矛盾。因此，B 是H_{un}的极小击中集。◊

在X上固定一个任意的全序 $\prec$。对于 $\mathcal{R}(H,\tau)$ 中的每个B，定义

$$D(B) := \{x \in B \mid B \setminus \{x\} \in \mathcal{R}(H,\tau)\}$$

因此，$D(B)$ 是这样一个顶点集合：删除其中任一顶点后，B仍会留下另一个罗马支撑。

若 $D(B) \neq \emptyset$，则将B的父节点定义为

$$p(B) := B \setminus \{\max_{\prec} D(B)\}.$$

由于X 是有限集，且 $\prec$ 是X上的全序，因此只要$D(B)$非空，$\max_{\prec} D(B)$就唯一确定。若$D(B) = \emptyset$，则称B为根。

令 $\mathcal{F}_p$ 为一个有向图，其顶点集为 $\mathcal{R}(H,\tau)$，并且对每个满足$D(B) \neq \emptyset$的$B \in \mathcal{R}(H,\tau)$，都添加一条从 B 到$p(B)$的弧。在下面的命题中，有根森林是指一种无环图，其组成树各有一个节点指定为根。

命题。 有向图 $\mathcal{F}_p$ 是一个有根森林，其根——按上述定义，即(H,τ) 中使 $D(\cdot)$ 为空的元素——恰好是 H_{un}的极小击中集。

证明。 首先我们证明，$D(B) = \emptyset$ 当且仅当B 是 $\mathcal{R}(H,\tau)$中的包含极小元素。根据前一个命题，这意味着 $\mathcal{F}_p$ 的根恰好是H_{un}的极小击中集。

若$D(B) \neq \emptyset$，则对于某个$x \in B$，有$B \setminus \{x\} \in \mathcal{R}(H,\tau)$，且$B$ 在 $\mathcal{R}(H,\tau)$中不是包含极小的。

反过来，假设 B 在 $\mathcal{R}(H,\tau)$中不是包含极小的。那么存在一个 $C \in \mathcal{R}(H,\tau)$，使得 $C \subsetneq B$。令 $x \in B \setminus C$。由于 C 是一个罗马支撑，它命中 H_{un}的每条超边；这意味着 $B \setminus \{x\}$ 也命中 H_{un}的每条超边，因为 $C \subseteq B \setminus \{x\}$。因此，本证明前文的删除命题可推出 $B \setminus \{x\}$ 是一个罗马支撑。由此，$x \in D(B)$，且 $D(B) \neq \emptyset$。

现在我们证明 $\mathcal{F}_p$ 是一个有根森林。每个非根罗马支撑B恰有一个父节点：$D(B) \neq \emptyset$，而全序 $\prec$使$p(B)$唯一。此外，根据$D(B)$的定义，$p(B)$本身也是一个罗马支撑，且$|p(B)| = |B| - 1$。由于 $\mathcal{R}(H,\tau) \subseteq 2^X$ 是有限的，从任意罗马支撑开始反复应用p，必定会终止于某个罗马支撑B^*，满足$D(B^*) = \emptyset$。因此，每个罗马支撑都属于一个包含根的连通分量。反复应用p 最终到达的根是唯一的，因为每个罗马支撑的父节点都是唯一的。

此外，$\mathcal{F}_p$中不可能存在环：$\mathcal{F}_p$ 中的每条有向弧都从一个罗马支撑指向其父节点，因此，这类弧的头部是一个基数严格小于弧尾部的集合。若 $\mathcal{F}_p$ 中存在环，则意味着 $\mathcal{F}_p$ 中的某个罗马支撑有两个基数——“它自身的”基数，以及根据连续有向弧基数递减的性质，沿着这些弧回到该罗马支撑时得到的一个更小的基数。

因此，$\mathcal{F}_p$ 是一个有根森林。 ◊

我们现在使用有根森林$\mathcal{F}_p$来定义一种反向搜索算法；该算法使用枚举H_{un} 的极小击中集的多项式延迟算法作为工具，而本证明假设该算法存在。当极小击中集枚举器输出一个根时，该根在此时并不是反向搜索算法本身的输出；只有在其所采用的递归过程被调用到该根时，反向搜索算法才会随后输出该根。

反向搜索算法执行以下步骤。运行（模拟）枚举H_{un} 的极小击中集的多项式延迟算法，直到它输出H_{un}的一个极小击中集，或报告不存在其他极小击中集。如果发生后一种情况，反向搜索算法终止。在前一种情况下，反向搜索算法暂停极小击中集枚举器，并对所枚举的根运行以下所规定的递归过程。该递归过程以一个罗马支撑B作为输入，并首先输出B。然后，对于每个$y \in X \setminus B$，该过程设置$C := B \cup \{y\}$；如果C是一个罗马支撑且$p(C) = B$，则该过程递归访问C。递归过程终止后，反向搜索算法恢复运行极小击中集枚举器，并重复对所假设的枚举器进行相同处理：如果它输出一个根，反向搜索算法就暂停极小击中集枚举器，并对所枚举的根运行上述递归过程；该过程结束后，它恢复运行极小击中集枚举器；如果极小击中集枚举器终止，反向搜索算法也终止。该过程持续进行，直到极小击中集枚举器以及随之而来的反向搜索算法终止。

对于 $\mathcal{F}_p$ 的顶点 B，将B的深度定义为从B 到其所在树的根的唯一有向路径中的弧数。

命题(正确性：反向搜索))。 反向搜索算法恰好输出每个罗马支撑集 *nce.*

证明。 我们首先证明，每个罗马支撑集都会由反向搜索算法输出。固定一棵 $\mathcal{F}_p$ 的有根树；我们证明，对于树中的所有深度d，该树中深度为d 的每个罗马支撑集都会被输出。我们对d进行归纳。

在深度 0 处，由前述命题可知，这棵树的根是H_{un} 的一个极小击中集；因此，基于假设存在最小击中集枚举器，它将被生成，反向搜索算法随后会对其调用递归过程并输出它。

现在假设反向搜索算法会输出固定树中深度为 d 的每个罗马支撑集。令 C 为同一棵树中深度为 d + 1 的一个罗马支撑集。那么，对某个深度为 d 的树中罗马支撑集 B，有 p(C) = B。由于 p(C) = B，存在唯一的顶点 y ∈ X \ B，使得 C = B ∪ {y}。根据归纳假设，反向搜索算法会输出 B。因此，它会访问 B，考察这个顶点 y，并在 C 确为罗马支撑集且满足 p(C) = B 的条件下输出 C。事实上，C 确为罗马支撑集，且 p(C) = B。因此，在验证这两个条件成立后，反向搜索算法会递归访问 C，进而输出集合 C。

接下来，我们通过对d 进行归纳法证明：对于 $\mathcal{F}_p$ 的有根树中深度为d 的每个罗马支撑B，反向搜索算法输出集合B 的次数不超过一次。

若 $d = 0$，则这是因为每个根都由最小击中集枚举器恰好生成一次，而反向搜索算法会对该枚举器生成的每个根恰好调用一次递归过程。此外，递归过程进行的一系列递归调用不可能返回到最初调用它的根，因为

递归过程访问罗马支撑集 B 时，只会对以 B为父节点的罗马支撑集进行递归，而根没有父节点。

现在假设深度为 d 的每个罗马支撑都至多输出一次。那么，如果C 是深度为 $d+1$的罗马支撑，其唯一的父节点 $p(C)$ 至多输出一次，因此递归过程至多访问它一次。当递归过程访问$p(C)$时，只会考察C 一次，因为存在唯一的顶点$y \in C \setminus p(C)$，使得 $C = p(C) \cup \{y\}$，所以递归过程只会考察y 一次。由于非根罗马支撑只有在反向搜索过程访问其唯一父节点时才能输出，因此反向搜索算法输出集合C 的次数不超过一次。◊

命题。 反向搜索算法以多项式延迟枚举所有 *Roman* 支持集。

证明。 根据假设，可以用多项式延迟枚举 $\mathcal{F}_p$ 的根（即H_{un} 的极小击中集）。因此，反向搜索算法输出第一个罗马支撑之前所需的时间是多项式级的。

现在我们说明：如果反向搜索算法连续两次枚举得到的元素都属于 $\mathcal{F}_p$中的同一棵有根树，那么两次枚举之间的时间有多项式上界。在 $\mathcal{F}_p$中的同一棵有根树内枚举两个元素期间，反向搜索算法只会针对每个通过向正在访问的集合 B 中加入某个 $y \in X \setminus B$ 而形成的 X 的子集 C，检查 C 是否为罗马支撑，且 $p(C) = B$；如果这些检查结果均为否，算法就会回溯到另一个罗马支撑。在此过程中，当反向搜索算法仍在访问 B 时，会考虑多项式数量的此类 C；具体而言，反向搜索算法恰好构造其中的 $|X \setminus B|$ 个。对于满足 $|C| = |B| + 1$ 的单个集合 C，执行反向搜索算法对 C 所做的两项检查只需多项式时间。也就是说，可以在多项式时间内检验 C 是否为罗马支撑：扫描 I_{un} 中的所有指标，验证对于每个 $i \in I_{\text{un}}$，$C \cap s_i \neq \emptyset$；并且对于每个 $x \in C$，扫描所有指标 $i \in I \setminus \{\tau(x)\}$，检查是否存在至少一个这样的指标满足 $s_i \cap C = \{x\}$。检查 $p(C) = B$ 也可以在多项式时间内完成。可以通过对每个 $c \in C$ 检验 $C \setminus \{c\} \in \mathcal{R}(H, \tau)$ 来计算 D(C)；如果 $D(C) \neq \emptyset$，再计算 $\max_{<} D(C)$，并将 $C \setminus \{\max_{<} D(C)\}$ 与 B 比较，即可完成所需的检查。如果反向搜索算法在执行这些检查后回溯到另一个罗马支撑，这一过程也会在多项式时间内完成：$\mathcal{F}_p$ 中有根树的深度至多为 $|X|$，因为由于父节点函数的性质，有向路径上的顶点数沿每条有向弧递减 1。

反向搜索算法完成对一棵有根树中每个 Roman 支持集的枚举后，我们假定存在的多项式延迟最小击中集枚举器会在多项式时间内生成下一个根，或报告没有剩余的根。因此，反向搜索算法能够以多项式延迟枚举所有 Roman 支持集。◊

我们给出的极小罗马命中函数枚举算法A_{enum} 将反向搜索算法作为工具。与反向搜索算法使用最小击中集枚举器的方式相同，A_{enum} 会在内部运行反向搜索算法，并在其输出一个罗马支撑后暂停。具体来说，每当反向搜索

算法生成一个罗马支撑B时，A_{enum} 会暂停反向搜索算法，并使用引理 7 中的多项式延迟过程枚举所有满足$f^{-1}(2) = B$ 的极小罗马击中函数f。随后，A_{enum}会恢复对反向搜索算法的模拟，直到该算法输出一个罗马支撑或终止。反向搜索算法终止时，A_{enum} 也会终止。

反向搜索算法的输出是由于其被用作A_{enum} 的内部工具而产生的，实际上并未由A_{enum}枚举。

首先，A_{enum} 会输出每一个极小罗马击中函数。为此，任取一个极小罗马击中函数f。由引理 7，$f^{-1}(2) \in \mathcal{R}(H, \tau)$。因此$f^{-1}(2)$是一个罗马支撑，反向搜索算法最终会输出$B$，使得$B = f^{-1}(2)$。此时$A_{\text{enum}}$ 会采用引理 7 中的过程，枚举所有满足$g^{-1}(2) = B$的极小罗马击中函数g；特别地，f 会由A_{enum}输出。

现在反设A_{enum} 将某个极小罗马击中函数f 枚举了两次。A_{enum} 对f的两次枚举，不可能发生在调用引理 7 中的多项式延迟过程、枚举所有满足$f^{-1}(2) = B$的极小罗马击中函数时；因为若是如此，引理 7 中所引用的多项式延迟过程就不是有效的枚举算法。因此，必定存在两个罗马支撑B_1 和B_2，反向搜索算法在不同时间枚举了它们，并且$f^{-1}(2) = B_1$ 且$f^{-1}(2) = B_2$。但根据传递性，$B_1 = B_2$ 成立，因此反向搜索算法不可能在不同时间输出B_1 和B_2。

算法 A_{enum} 具有多项式延迟。如果反向搜索算法报告不存在罗马支撑，那么A_{enum} 将在多项式时间内终止，且不产生任何输出。否则，从A_{enum} 初始化到A_{enum} 首次输出之间的时间为多项式级别，因为反向搜索算法会在启动后经过多项式时间产生其第一个输出，而A_{enum} 用作子程序的引理 7 中的枚举算法同样会在多项式时间内产生第一个输出。

由引理 7 可知，对于某个罗马支撑集 B，满足$f^{-1}(2) = B$ 的极小罗马击中函数的两次枚举之间的延迟是多项式级的。根据引理 7 所确立的 Ψ_H 的满射性，对于每个罗马支撑集 B，都存在一个极小罗马击中函数 f，使得$f^{-1}(2) = B$。因此，反向搜索算法输出一个罗马支撑集后，A_{enum} 会在多项式时间内枚举出一个极小罗马击中函数。在一个罗马支撑集的原像中的最后一个函数与下一个罗马支撑集的原像中的第一个函数之间，依次只有引理 7 中枚举器的多项式延迟终止、反向搜索算法进行到下一个罗马支撑集的多项式延迟转换，以及引理 7 中枚举器的多项式延迟初始化。最后，最后一个已枚举的极小罗马击中函数与A_{enum} 终止之间的时间是多项式级的，因为最后一个已枚举的极小罗马击中函数与引理 7 中枚举器终止之间的时间是多项式级的，而反向搜索算法最后一个已枚举的罗马支撑集与反向搜索算法终止之间的时间也是多项式级的。由此证明该定理。

□

定理 15 和定理 16 推出以下推论。

推论 7 (多项式延迟等价性)。 存在一种多项式延迟算法，对于任意给定的实例 (H, τ)，枚举 (H, τ) 的所有极小罗马命中函数，当且仅当存在一种多项式延迟算法，对于任意给定的超图，枚举其所有极小击中集。

由于目前尚无已知的多项式延迟算法可用于枚举极小击中集，我们接下来考虑相应的参数化枚举问题。以下记号基于 [21]。

如果满足以下条件，则 $=(Q,\kappa,\mathrm{Sol})$ 是一个（基于有限字母表 Σ 的）参数化枚举问题：

- $Q \subseteq \Sigma^*$,
- $\kappa\colon \Sigma^* \to \mathbb{N}$，即所谓的参数化，是一个可计算函数，并且
- $\mathrm{Sol} : \Sigma^* \to 2^{\Sigma^*}$ 是一个函数，使得对于所有$x \in \Sigma^*$，$\mathrm{Sol}(x)$都是有限集，并且当且仅当$x \in Q$时，$\mathrm{Sol}(x) \neq \emptyset$.

对于字母表 Σ上的参数化枚举问题$E =(Q,\kappa,\mathrm{Sol})$，$x \in \Sigma^*$ 是一个实例，其参数为 $\kappa(x)$，而$\mathrm{Sol}(x)$是x 的解集（按照适当的“解”概念，例如极小击中集）。目标是不重复地枚举$\mathrm{Sol}(x)$中的所有元素。

对于参数化枚举问题，若存在一个多项式p和一个可计算函数$t\colon \mathbb{N} \to \mathbb{N}$，使得从程序开始运行到输出第一个解的时间、输出最后一个解到程序终止的时间，以及任意两个连续解输出之间的时间，均有$t(\kappa(x))p(|x|)$上界，则该枚举算法具有FPT延迟。

以下推论结合定理 16 的证明和 [11] 的极小命中集枚举算法即可得到。

推论 8 （关于 κ_1 参数化的 FPT 延迟）。 存在一种多项式空间、固定参数可处理延迟的枚举算法，用于枚举关于 $\kappa_1 = |I_{\mathrm{un}}|$ 参数化的极小罗马击中函数。

证明。 设 $H =(X,(s_i)_{i\in I})$ 是一个超图，且 $\tau\colon X \to I$ 是一个对应函数。按照定理 16 的证明，定义 $H_{\mathrm{un}} := (X,(s_i)_{i\in I_{\mathrm{un}}})$。由于我们希望应用 [11]，中的极小命中集枚举器，但该极小命中集枚举器仅适用于边集为集合的超图，而H_{un} 可能包含重复的超边，因此与之不同，我们定义一个辅助超图，从H_{un}中删除重复的超边。令 $\mathcal{S}_{\mathrm{un}} = \{s_i \mid i \in I_{\mathrm{un}}\}$，并定义$H_{\mathrm{un}} = (X,\mathcal{S}_{\mathrm{un}})$。由于重复的超边不会影响极小命中集，枚举$H_{\mathrm{un}}$ 的极小命中集等价于枚举H_{un}的极小命中集。

若 $\emptyset \in \mathcal{S}_{\mathrm{un}}$，则$H_{\mathrm{un}}$ 没有击中集，因此既不存在罗马支撑，也不存在极小罗马击中函数；这种情况可以通过定义一个具有本推论所需性质、且终止时不产生任何输出的极小罗马击中函数枚举器，轻而易举地处理。因此，在本证明余下部分中，我们假设 $\emptyset \notin \mathcal{S}_{\mathrm{un}}$。

这一假设使我们能够将 k^* 定义为 H_{un}的极小击中集的最大基数。我们首先证明 $k^* \leq \kappa_1$。这点很重要，因为 [11] 给出的算法在枚举任意超图的极小击中集时，其延迟界取决于该超图的超边数以及该超图的极小击中集的最大基数 k^*。我们的目标是证明，[11] 中的极小击中集枚举器具有以 κ_1为参数的固定参数可处理延迟，这样我们就能在定理 16 的证明中用它替代 A_{enum} 中假设存在的极小击中集枚举器。

设 T 是 H_{un} 的一个极小击中集。由于 T 是极小的，对于每个 $x \in T$，都存在一个超边 $e_x \in \mathcal{S}_{\text{un}}$，使得 $e_x \cap T = \{x\}$（如果不存在这样的超边——也就是说，对于 H_{un} 的每条超边，如果它与 T 的交集包含 x，就必须还包含除 x 之外的另一个元素——那么 $T \setminus \{x\}$ 就会是 H_{un} 的一个击中集，这与 T 的极小性矛盾）。令 $x \mapsto e_x$ 是从 T 到 $\mathcal{S}_{\text{un}}$ 的任意函数，它将 T 中的一个顶点 $x \in$ 映射到某个超边 $e_x \in \mathcal{S}_{\text{un}}$，使得 $e_x \cap T = \{x\}$。$x \mapsto e_x$ 是单射：如果 $x \neq y$，那么若 $e_x = e_y$，就存在唯一的超边 e_x，同时满足 $e_x \cap T = \{x\}$ 和 $e_x \cap T = \{y\}$，这是不可能的。因此，$x \mapsto e_x$ 的陪域的基数至少与 $x \mapsto e_x$ 的定义域的基数相等；即 $|T| \leq |\mathcal{S}_{\text{un}}| \leq \kappa_1$。由于 T 是任意的，这就得到了 $k^* \leq \kappa_1$，证毕。

通过 [11], ，可以以同时占用多项式空间且延迟为 $O(m^{k^*+1}n^2)$的方式枚举具有 n 个顶点和 m 条超边的超图的极小击中集。我们将 [11] 给出的极小击中集枚举器的延迟表示为 $O(1+m^{k^*+1}n^2)$，以处理退化的边情况。将给定的极小击中集枚举器应用于具有 $n = |X|$、$m = |\mathcal{S}_{\text{un}}| \leq \kappa_1$和 $k^* \leq \kappa_1$ 的 H_{un}，可得到一个枚举 H_{un} 的极小击中集的算法，其延迟为 $O(1+\kappa_1^{\kappa_1+1}|X|^2)$，并占用多项式空间。该延迟可写为 $g(\kappa_1) \cdot |(H,\tau)|^{O(1)}$，其中可计算函数 $g(k) = 1 + k^{k+1}$，而 $|(H,\tau)|$ 表示实例 (H, τ) 的编码长度。因此，存在一个以 κ_1为参数、具有 FPT 延迟的多项式空间极小击中集枚举器。

现在考虑算法 A'_{enum}，它在各方面都与定理 16 的证明中的 A_{enum} 相同，唯一的区别是：该算法调用反向搜索算法时，具体使用了一个以 κ_1 为参数、具有 FPT 延迟的多项式空间极小击中集枚举器，来生成 H_{un} 的极小击中集，而不是使用任意的极小击中集枚举器来枚举 H_{un}的极小击中集。根据本证明第一段中的讨论，枚举 H_{un} 的极小击中集等价于枚举 H_{un} 的极小击中集，因此，A_{enum} 的正确性证明同样适用于 A'_{enum}。

—

如此修改后的反向搜索算法是一个针对所有罗马支撑集、相对于 κ_1 的多项式空间枚举算法。修改后的反向搜索算法只需占用运行多项式空间极小击中集枚举器所需的空间，并维护深度为 $|X|$的递归栈；栈中每个帧只存储多项式大小的信息，即当前访问的罗马支撑集、待检查的下一个候选顶点，以及成员测试和父节点测试所需的数据。

由于引理 7 中的过程同样使用多项式空间，因此可知 A'_{enum} 自身仅使用多项式空间。算法 A'_{enum} 具有固定参数可处理延迟，因为其任意两个输出之间的延迟，至多是最小击中集枚举器部分的固定参数可处理延迟与 A_{enum} 在其任意两个输出之间执行的多项式延迟操作之和。

□

由于 κ_1 度量了对应函数的“满射性距离”，因此我们可以推导出以下结果；该结果可视为扩展问题领域中推论 2 的对应版本。

推论 9. 存在一种针对具有满射对应函数的极小罗马击中函数的多项式延迟、多项式空间枚举算法。

9. 结论

我们通过引入罗马击中函数的定义，将罗马支配的概念推广到了超图。我们证明了，当对应函数为满射时，所有极小罗马命中函数都可以通过多项式延迟枚举。这可以视为对我们所提出问题的一种技术性回答：是什么导致罗马支配在多项式延迟可枚举性方面表现得不同于经典支配？当对应关系不是满射时，罗马击中函数的表现更像支配集；具体而言，以给定预解的权重为参数时，其扩展问题是W[3]-完全的，而且我们观察到，枚举所有极小罗马击中函数是横截困难的。

我们在引言中提到，罗马支配属于固定参数可处理，若采用对偶方式进行参数化，在本例中即以$n - k$为参数，其中n 是图的顶点数，k 是罗马支配函数权重的上界。若能为本文提出的罗马支配的两种推广形式获得类似结果，或许会很有意义。不过，目前还不清楚 $\omega(f)$-参数化的“对偶”应该是什么。

请注意，直到最近，已知只有少数几个（自然）问题是W[3],完全的，而W是参数化复杂性的W层级的第三级。即使在今天，除了我们在本文中一直提到的扩展问题及其相关问题外，我们也只知道 [13, 16, 17]中所列出的问题。如今，越来越多的问题被发现是W[3] 完全的，这使W[3]变得更加令人关注。或许现在是时候解决那个已有25年历史的开放问题了：如果W[3] = W*[3]，见 [23]。根据 [17]，，目前的状态是：W[3] ⊆ W*[3] ⊆W[4]。

参考文献

[1] Aazami, A., Cheriyan, J., Jampani, K.R.：平面图中打包元素不相交施泰纳树的近似算法与困难性结果。Algorithmica**63**(1-2)，425–456（2012）[2] Abu-Khzam, F.N., Bazgan, C., Chopin, M., Fernau, H.：用于改进近似算法的数据归约与组合界。Journal of Computer and System Sciences **82**(3)，503–520（2016）[3] Abu-Khzam, F.N., Fernau, H., Mann, K.：罗马普查：图类上罗马支配函数的枚举与计数。载于：Leroux, J., Lombardy, S., Peleg, D.（编）第48届计算机科学数学基础国际研讨会，MFCS。莱布尼茨国际信息学会议录（LIPIcs），第272卷，第6:1–6:15页。Dagstuhl城堡–莱布尼茨信息学中心（2023）

计算机科学数学基础，MFCS。莱布尼茨国际信息学会议录（LIPIcs），第272卷，第6:1–6:15页。Dagstuhl城堡–莱布尼茨信息学中心（2023）[4] Abu-Khzam, F.N., Fernau, H., Mann, K.：极小罗马支配函数：扩展与枚举。Algorithmica**86**，1862–1887（2024）[5] Acharya, B.D.：超图中的支配。AKCE国际图论与组合数学期刊 **4**（2），117–126（2007）[6] Arquilla, J.：从闪电战到比特战：军事与计算机的交锋。ACM通讯**54**（10），58–65（2011）[7] Arquilla, J., Fredricksen, H.：“绘制”最优大战略。军事运筹学，第3–17页（1995年秋），http://hdl.handle.net/10945/38438[8] Bermudo, S., Fernau, H.：用于更小核的组合数学：图的微分ff。《理论计算机科学》**562**，330–345（2015）

[9] Bermudo, S., Fernau, H., Sigarreta, J.M.：《图的差分数与罗马支配数》。《应用分析与离散数学》**8**, 155–171 (2014)

[10] Bläsius, T., Friedrich, T., Lischeid, J., Meeks, K., Schirneck, M.：高效枚举数据剖析中产生的超图的命中集。载于：《算法工程与实验》（ALENEX），第 130–143 页。SIAM (2019)

[11] Bläsius, T., Friedrich, T., Lischeid, J., Meeks, K., Schirneck, M.：高效枚举数据剖析中产生的超图的命中集。《计算机与系统科学杂志》**124**, 192–213 (2022)

[12] Bläsius, T., Friedrich, T., Schirneck, M.：《关系数据库中依赖关系检测与发现的复杂性》。《理论计算机科学》**900**, 79–96 (2022)

[13] Bodlaender, H.L., Groenland, C., Pilipczuk, M.：《树结构图问题的复杂性》。技术报告 arXiv:2208.12543v3，ArXiv，康奈尔大学（2022）

[14] Casel, K., Fernau, H., Ghadikolaei, M.K., Monnot, J., Sikora, F.：一些边图问题的扩展：标准复杂性与参数化复杂性。载于：Gasieniec, L.A., Jansson, J., Levcopoulos, C.（编）：计算理论基础——第22届国际研讨会，FCT。LNCS，第11651卷，第185–200页。施普林格（2019）[15] Casel, K., Fernau, H., Ghadikolaei, M.K., Monnot, J., Sikora, F.：关于优化问题解扩展的复杂性。《理论计算机科学》**904**, 48–65（2022）[16] Chen, J., Zhang, F.：三级供应链模型中的乘积覆盖：W[3]与W[4]的自然完全问题。《理论计算机科学》**363**（3），278–288（2006）

[17] Chen, Y., Flum, J., Grohe, M.：W*W**层次结构分析。《符号逻辑杂志》**72**（2），513–534（2007）[18] Chvátal, V.：集合覆盖问题的贪心启发式算法。《运筹学数学》**4**（3），233–235（1979）[19] Cockayne, E.J., Dreyer Jr., P., Hedetniemi, S.M., Hedetniemi, S.T.：图中的罗马支配。《离散数学》**278**，11–22（2004）[20] Creignou, N., Kröll, M., Pichler, R., Skritek, S., Vollmer, H.：困难枚举问题的复杂性理论。《离散应用数学》**268**，191–209（2019）[21] Creignou, N., Meier, A., Müller, J., Schmidt, J., Vollmer, H.：参数化枚举范式。《计算系统理论》**60**（4），737–758（2017）[22] Dinur, I., Steurer, D.：平行重复的分析方法。载于：Shmoys, D.B.（编）：计算理论研讨会，STOC。第624–633页。ACM（2014）[23] Downey, R.G., Fellows, M.R.：阈值支配集及W*W*[2]的改进刻画。《理论计算机科学》**209**（1–2），123–140（1998）[24] Downey, R.G., Fellows, M.R.：《参数化复杂性基础》。计算机科学丛书，施普林格（2013）[25] Dreyer, P.A.：图支配的应用与变体。博士论文，罗格斯大学，美国新泽西州（2000）[26] Eiter, T., Gottlob, G.：识别超图的极小横截集及相关问题。《SIAM计算杂志》**24**（6），1278–1304（1995）[27] Fernau, H.：罗马支配：参数化视角。《国际计算数学杂志》**85**，25–38（2008）[28] Fernau, H., Mann, K.：《击中罗马人》。技术报告2302.11417，ArXiv，康奈尔大学，美国（2023）[29] Fernau, H., Mann, K.：罗马击中函数。载于：Rz.a˙zewski, P., Bonnet, E.（编）：第19届参数化与精确计算国际研讨会，IPEC。LIPIcs，第321卷，第24:1–24:15页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik（2024）[30] Flum, J., Grohe, M.：《参数化复杂性理论》。施普林格（2006）[31] Frieze, A.M., Mubayi, D.：简单超图的着色。《组合理论杂志》B辑**103**（6），767–794（2013）

[32] Gainer-Dewar, A., Vera-Licona, P.：《极小击中集生成问题：算法与计算》。SIAM 离散数学杂志 **31**(1), 63–100 (2017)[33] Gogic, G., Papadimitriou, C.H., Sideri, M.：知识的增量重新编译。《人工智能研究杂志》**8**, 23–37 (1998)[34] Guo, J., Hüffner, F., Niedermeier, R.：参数化问题的结构视角：与平凡性的距离。载于：Downey, R., Fellows, M., Dehne, F.（编）《国际参数化与精确计算研讨会 IWPEC 2004》。LNCS，第 3162 卷，第 162–173 页。施普林格 (2004)[35] Haynes, T.W., Hedetniemi, S., Henning, M.A.（编）：《图的支配主题》，《数学进展》，第 64 卷。施普林格 (2020)[36] Johnson, D.S., Papadimitriou, C.H., Yannakakis, M.：关于生成所有极大独立集。《信息处理快报》**27**(3), 119–123 (1988)[37] Kanté, M.M., Limouzy, V., Mary, A., Nourine, L.：极小支配集及其变体的枚举。载于：Owe, O., Steffen, M., Telle, J.A.（编）《计算理论基础》——第 18 届国际研讨会 FCT。LNCS，第 6914 卷，第 298–309 页。施普林格 (2011)

[38] Kanté, M.M., Limouzy, V., Mary, A., Nourine, L.：关于极小支配集及相关概念的枚举。SIAM 离散数学杂志 **28**(4),1916–1929 (2014)

[39] Kanté, M.M., Limouzy, V., Mary, A., Nourine, L., Uno, T.：用于枚举图中极小边支配集的多项式延迟算法。载于：Dehne, F., Sack, J., Stege, U.(编)《算法与数据结构研讨会》，WADS。LNCS，第 9214 卷，第 446–457 页。施普林格 (2015)

[40] Liedloff, M., Kloks, T., Liu, J., Peng, S.L.：若干图类中罗马支配的高效算法。《离散应用数学》**156**(18), 3400–3415 (2008)

[41] Mann, K., Fernau, H.：完美罗马支配：枚举与参数化方面。算法**17**(12:576)（2024）[42] Mann, K., Fernau, H.：罗马击中集。载于：Královic, R., Kurková, V.（编）《SOFSEM 2025：计算机科学理论与实践——第50届国际会议》，第二部分。LNCS，第15539卷，第142–156页。施普林格（2025）[43] Mary, A., Strozecki, Y.：通过闭包运算高效枚举解。《离散数学与理论计算机科学》**21**(3)（2019）[44] Ouvrard, X.：超图：导论与综述。技术报告 2002.05014，ArXiv，康奈尔大学，美国（2020）

[45] Padamutham, C., Venkata Subba Reddy Palagiri：图中罗马支配的算法方面。《应用数学与计算杂志》**64**，89–102（2020）

[46] Pagourtzis, A., Penna, P., Schlude, K., Steinhöfel, K., Taylor, D.S., Widmayer, P.：服务器放置、罗马支配及其他支配集变体。载于：Baeza-Yates, R.A., Montanari, U., Santoro, N.（编）《网络与移动计算时代的信息技术基础》，IFIP第17届世界计算机大会——TC1专题 / 第二届IFIP理论计算机科学国际会议（IFIP理论计算机科学）。第280–291页。Kluwer（2002）；亦以技术报告365的形式提供，苏黎世联邦理工学院理论计算机科学研究所，10/2001。

[47] Read, R.C., Tarjan, R.E.：关于列举圈、路径和生成树的回溯算法的界。Networks**5**，237–252（1975）[48] Stewart, I.：保卫罗马帝国。《科学美国人》**281**(6)，136,137,139（1999年12月）[49] Strozecki, Y.：枚举复杂度。EATCS通报**129**（2019）