上海交通大学硕士学位论文

几个参数化问题的近似困难性研究

姓　　名：刘裕炜

学　　号：123033910011

导　　师：陈翌佳教授

院　　系：计算机学院

学 科/专 业：计算机科学与技术

申 请 学 位：工学硕士

2026 年 4 月 13 日

A Dissertation Submitted to
Shanghai Jiao Tong University for the Degree of Master

ON THE HARDNESS OF APPROXIMATION OF SEVERAL PARAMETERIZED PROBLEMS

Author: Yuwei Liu
Supervisor: Prof. Yijia Chen

School of Computer Science
Shanghai Jiao Tong University
Shanghai, P.R. China

April 13th, 2026

上海交通大学

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全知晓本声明的法律后果由本人承担。

学位论文作者签名：

日期：　　年　　月　　日

上海交通大学

学位论文使用授权书

本人同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。

本学位论文属于：

□**公开论文**

□**内部论文**，保密□1 年/□2 年/□3 年，过保密期后适用本授权书。

□**秘密论文**，保密____ 年（不超过 10 年），过保密期后适用本授权书。

□**机密论文**，保密____ 年（不超过 20 年），过保密期后适用本授权书。

（请在以上方框内选择打“✓”）

学位论文作者签名：　　指导教师签名：

日期：　　年　　月　　日　　日期：　　年　　月　　日

摘 要

近似困难性是计算复杂性以及参数复杂性领域的一个重要的研究方向。在这一领域中，我们期望证明一些问题的近似求解和精确求解的困难程度是相当的，从而完整刻画这些问题的计算复杂性。

本文进一步发展了纠错码的碰撞数这一技术工具，并将其与几个具体的参数化问题结合，以证明这些问题的任意常数比近似困难性，并探讨了未来可能的研究方向。具体来说，本文的贡献主要在于以下方面：

- 构造了 k-近似解码问题从精确求解到近似求解的自归约，并进一步在给出了一系列编码问题和格问题的常数比近似算法在指数时间假设下更紧的运行时间下界；
- 证明了任意常数比近似 2-约束满足问题的可满足多赋值的总大小都是 W[1]-困难的，且进一步证明了任意常数比近似 k-精确覆盖问题的 W[1]-困难性。

关键词： 参数化近似困难性，编码问题，格问题，约束满足问题，时间下界，W[1]-困难性

Abstract

Hardness of approximation is an important research direction in computational complexity and parameterized complexity area. In this area, we expect to prove the equivalence between the hardness of approximately solving some problem and the hardness of exactly solving them, hence giving a complete characterization to the computational complexity of these problems.

In this article, we further develop the collision number of error-correcting code as a technical tool, and combine it with several concrete parameterized problems to prove the hardness of approximating these problems within any constant factor, and discuss some possible directions for future research. To be more precise, the contribution of this article is as follow:

- we construct a self-reduction from exactly solving the k-Maximal Likelihood Decoding problem to approximately solving it, and further give tighter running time lower bounds for constant approximating a series of coding problems and lattice problems under the exponential time hypothesis.
- we prove that constant approximating the total size of a satisfying multi-assignment to a 2CSP instance is $\mathsf{W}[1]$-hard, and further proved the $\mathsf{W}[1]$-hardness of constant approximating k-Exact Cover problem.

Key words: Parameterized Inapproximability, Coding Problems, Lattice Problems, Constraint Satisfaction Problem, Time Lower Bound, W[1]-hardness

目 录

插 图

表 格

符号对照表

$\mathbb{N}$	自然数集
$\mathbb{N}^+$	正整数集
$\mathbb{Z}$	整数集
$\mathbb{R}$	实数集
$\mathbb{F}$	数域
Σ	（编码或约束满足问题的）字母表
$\mathbb{E}$	数学期望
C	编码
Col_ε	ε 碰撞数
$\|\|\cdot\|\|_p$	p-范数
Φ	（约束满足问题的）约束集
σ	（用于约束满足问题时）赋值
$\hat{\sigma}$	（用于约束满足问题时）多赋值
$\dot{\cup}$	不交并
$O, o, \Omega, \omega, \Theta$	渐近记号

第 1 章 绪论

1.1 研究背景

计算复杂性（computational complexity）是理论计算机科学的一个核心研究领域。上世纪 70 年代初，铁幕两侧的研究者们各自独立对求解问题的难度进行了研究，产生了里程碑式的结果——库克-列文定理[1-2]（Cook-Levin Theorem）。该定理建立了 NP-完全性的概念。随后，Karp 证明了二十余个计算问题都是 NP-完全的[3]。自此开始至今的五十余年间，研究者证明了大量计算问题的计算复杂性都至少是 NP-困难的。研究者普遍相信 $\mathsf{P} \neq \mathsf{NP}$，也自然认为这些困难的问题不存在多项式时间的算法，一些更强的假设[4-5]甚至认为很多 NP-困难问题，如 3-布尔可满足性问题（即每个子句中项的个数不超过 3 的合取范式可满足性问题，下称 3SAT 问题）不存在亚指数时间的算法。

很多计算问题自然地存在对应的优化问题。由于精确求解的困难性已被证明，研究者们开始考虑寻找问题的近似解。对优化问题中的最小化问题（最大化问题），假设其最优解的值是 OPT，对任意 $c > 1$，该问题的一个 c-近似解是满足优化值不超过 $c \cdot \mathrm{OPT}$（OPT/c）的可行解。后续研究表明，近似求解很多优化问题的困难性依旧是 NP-困难的，著名的 PCP 定理[6-8]即为一个典例，它表明存在某个常数 c，使得求解约束满足问题（comstraint satisfaction problem，下称 CSP）的 c-近似解依旧是 NP-困难的。PCP 定理的重要意义在于，它给出了计算复杂性领域中最基本的问题——CSP 问题的一个“创造间隔（gap-creating）”的归约。这个归约过程可以将两个最优解差距很小的实例转化为两个最优解差距极大的新实例，或者说，它将一个没有间隔的优化问题转化为一个有常数间隔的问题。后续有大量近似困难性的研究结果即依赖于 PCP 定理所创造出的间隔，例如最大团问题[9]、集合覆盖问题[10]、编码问题和格问题[11]等。

另一个应对 NP-困难问题的角度是参数化（parameterization）。一个参数化的问题给原问题配上了一个可以多项式时间计算的函数 $\nu : \{0,1\}^* \to \mathbb{N}$，使得该问题的求解时间不仅与输入长度有关，还和该函数所表示的参数有关。例如，对参数化的 k-顶点覆盖问题（k-VertexCover），输入 (G,k)，判定图 G 中是否存在大小不超过 k 的顶点覆盖。由于参数 k 可以任意地小于输入长度，在参数化视角中，我们定义一个问题是*固定参数可解* (fixed-parameter tractable，下称 FPT) 的，如果输入 (x,k) 下它可以在

$f(k)|x|^{O(1)}$ 时间内计算，此处 f 可以是任意可计算函数。前述的 k-VertexCover 问题即为 FPT 的。相应地，参数复杂性中对应于 NP 类的复杂性类是 W[1]，而 W[1] ≠ FPT 假设认为 W[1] 中的一些参数问题被认为不是 FPT 的，如 k-团问题（k-Clique）。

同理，对于一些困难的参数化问题，研究者也考虑了近似求解它们的计算复杂性。但与经典的计算复杂性领域不同，参数复杂性中没有类似 PCP 定理这样普适的结果，因此对参数问题近似困难性的研究主要分为两类：一类是直接从自带间隔的困难性假设出发（例如[12-14]，另一类是为每个问题设计创造出间隔的归约方法和框架。后者的主要问题是方法较为分散，例如关于 k-团问题的一系列工作是基于并行编码的[15-19]，关于 k-集合覆盖问题（等价地，支配集问题）的系列工作既有基于编码的[20]，又有基于组合方法的[21-24]。关于参数问题近似困难性的研究现状，读者可参阅 Feldmann 等人的综述论文[25]。

1.2　本文贡献

本文进一步发展了参数问题近似困难性研究领域中的组合方法，首次提供或简化了几个问题的参数近似困难性的证明。具体而言，本文的工作涉及以下两方面。

第一，本文给出了几个参数化编码问题和格问题近似困难性的简化证明，该证明使用的技术非常初等，且大大缩短了之前结果[26-27]所需的归约链，给出了更好的参数，从而在合理的困难性假设下给出了这几个编码问题和格问题，不论是精确算法还是常数比近似算法都适用的更紧的时间下界。

第二，本文首次证明了对参数化 2-约束满足问题（2CSP）多赋值量的常数比近似是 W[1]-困难的。在此基础之上，本文给出了从这一问题到参数化精确覆盖问题的归约，由此证明了精确覆盖问题的任意常数比近似也是 W[1]-困难的。

这两方面结果的核心构造都依赖于之前关于临界图和编码问题的研究[23-24,28]，使用的方法非常初等简单，且绕开了近似困难性领域之前极其依赖的 PCP 定理技术。本文使用的技术为将来证明更多参数问题近似困难性提供了新的思路。

1.3　文章结构

本文在第 2章对要用到的背景知识进行简要介绍。第 3章介绍本文两方面结果所依赖的技术工具，即编码的碰撞数分析。第 4章介绍本文中关于参数化的编码问题和格问题的近似困难性证明，包括最近编码问题、最短距离问题、最近向量问题、最短向量问题等。第 5章介绍本文中关于参数化 2-约束满足问题（2CSP）的多赋值量的近

似困难性证明，以及这一证明的推广和应用。第 4章和第 5章间较为独立，可以分别阅读。最终，在第 6章中对本文结果进行总结与展望。

第 2 章 预备知识

本章介绍本文中所需的符号约定和预备知识。

我们分别使用 $\mathbb{N}, \mathbb{N}^+, \mathbb{R}, \mathbb{Z}$ 表示自然数集、正整数集，实数集和整数集。对素数幂 p，我们使用 $\mathbb{F}_p$ 表示有 p 个元素的伽罗瓦域。对正整数 n，我们使用 $[n] = \{1, 2, \cdots, n\}$。对集合 S，使用 2^S 表示 S 的幂集。对向量 $\vec{v}$ 或字符串 x，$\vec{v}[i]$ 和 $x[i]$ 分别表示 $\vec{v}$ 和 x 的第 i 个位置的内容。

对于向量 $\vec{v} \in \mathbb{R}^d$，令 $||\vec{v}||_0$ 为 $\vec{v}$ 中非零位置的数量（即 0-范数）。对任意 $p \geq 1$，定义向量 $\vec{v}$ 在实数意义下的 p-范数为

$$||\vec{w}||_p = \left(\sum_{i \in [d]} \vec{w}[i]^p \right)^{1/p}$$

对字母表 Σ 上长度为 m 的字符串 $x, y \in \Sigma^m$，用 $\mathsf{dist}(x, y)$ 表示 x 和 y 的相对汉明距离，即

$$\mathsf{dist}(x, y) = \frac{|\{i \in [m] : x[i] \neq y[i]\}|}{m}。$$

用 $||x - y||_0$ 表示 x 和 y 的汉明距离，即

$$||x - y||_0 = |\{i \in [m] : x[i] \neq y[i]\}|。$$

注意到当 $p = 0$ 时，实向量的 0-范数就等价于它与全零向量的汉明距离，因此我们使用 $||x - y||_0$ 表示。

我们使用一些约定俗成的渐近记号，包括 $O, o, \Omega, \omega, \Theta$。对函数 $f, g : [0, +\infty) \to [0, +\infty)$，

- $f = O(g)$ 代表存在常数 $c > 0$，
$$\lim_{n \to +\infty} \frac{f(x)}{g(x)} \leq c;$$
- $f = o(g)$ 代表
$$\lim_{n \to +\infty} \frac{f(x)}{g(x)} = 0;$$
- $f = \Omega(g)$ 代表存在常数 $c > 0$，
$$\lim_{n \to +\infty} \frac{g(x)}{f(x)} \leq c;$$

- $f = \omega(g)$ 代表
$$\lim_{n\to+\infty} \frac{g(x)}{f(x)} = 0;$$
- $f = \Theta(g)$ 代表 $f = O(g)$ 且 $f = \Omega(g)$。

2.1　参数复杂性

我们简述文中所需的参数复杂性知识，关于该领域的详细介绍，读者可参阅相关书籍[29-31]。对于基本的计算复杂性领域介绍，可参阅相关中外文书籍[32-33]。

定义 2.1 (参数化问题)　一个参数化问题 (L, κ) 由语言 $L \subseteq \{0,1\}^*$ 和多项式时间可计算函数 $\kappa : \{0,1\}^* \to \mathbb{N}$ 构成的有序对，其中 κ 被称为参数。

参数复杂性中的“高效计算”定义如下：

定义 2.2 (固定参数可解)　一个参数问题的 (L, κ) 算法 A 被称作固定参数可解（FPT）当且仅当存在一个可计算函数 f，使得对任意输入 $x \in \{0,1\}^*$，$A(x)$ 的运行时间不超过 $f(\kappa(x)) \cdot |x|^{O(1)}$。

一个参数问题的 (L, κ) 被称作固定参数可解（FPT）当且仅当存在一个 FPT 的算法 A 求解该问题。

接下来，我们定义参数复杂性中的归约方式。

定义 2.3　一个从问题 (L, κ) 到问题 (L', κ') 的 FPT 归约是一个满足如下性质的函数 $f : \{0,1\}^* \to \{0,1\}^*$：

- f 可在 FPT 时间内计算；
- $x \in L$ 当且仅当 $f(x) \in L'$；
- 存在可计算函数 $g : \mathbb{N} \to \mathbb{N}$ 满足对任意 $x \in \{0,1\}^*$，$\kappa'(f(x)) \le g(\kappa(x))$。

在引入参数复杂性中的困难类之前，我们先给出一个经典问题——k-团问题（以下记为 k-Clique）的定义。

k-Clique	
输入实例：	图 $G = (V, E)$ 和整数 k，V, E 分别为 G 的顶点集和边集。
参数：	k
问题：	区分以下两种情况：
(YES)	G 中存在一个顶点数为 k 的完全子图。
(NO)	G 中的任意完全子图的顶点数均小于 k。

与 FPT 相对应，参数复杂性中被认为是“困难”的复杂性类是 W[1]。该复杂性类有完整的逻辑和电路刻画[30]。在此为简便起见，我们引用已有的结果如下。

定理 2.1 ([30]) k-团问题是 W[1]-完全的，即：k-Clique 在 W[1] 中，且 W[1] 中的所有问题均可 FPT-归约到 k-Clique。

2.2 近似算法

近似算法是在高效精确算法（基于某些合理的计算复杂性假设下）被证明不存在时的放松。本节我们简单介绍近似算法和近似比的概念，更多背景知识可参阅 V. Vazirani 的专著[34]。首先我们引入优化问题及其最优解的简要定义，严格的定义可参阅上述专著[34]的附录 A。

定义 2.4 (优化问题) 优化问题 Π 是一个最小化或最大化问题。每个问题 Π 都有一个多项式时间可计算的目标函数 $f : \{0,1\}^* \to \mathbb{R}$，每个实例 I 存在一个可行解的集合。对于最小化问题的实例 I，其最优解的目标函数值（记为 $OPT(I)$）即为所有可行解的目标函数值中的最小值（最大化问题反之）。

下面我们给出近似比的定义。

定义 2.5 (近似比) 对一个最小化（最大化）问题 Π 的算法 A，若存在某个函数 $\gamma : \mathbb{N} \to [1, +\infty)$ 使得对任意实例 I，$A(I) \le \gamma(|I|) \cdot OPT(I)$（若为最大化问题，则 $A(I) \ge OPT(I)/\gamma(|I|)$），则称 A 为问题 Π 的 γ-近似算法，γ 被称为算法 A 的近似比。

注 近似比 γ 为常数时是易于理解的。而当 γ 是一个与输入规模有关的函数时，近似解的质量会随着输入规模大小增加而变差。例如，最小化问题中的集合覆盖问题（Set Cover）最优的近似算法的近似比为 $\gamma = O(\log |I|)$（参见 Vazirani 的专著[34]的第 2 章）。

2.3 纠错码

纠错码是计算机科学和信息论中的核心工具之一。本文中讨论的问题和使用的构造方法均与纠错码紧密相关。本小节给出纠错码的一般定义。

定义 2.6 (纠错码) 字母表 Σ 上长度为 m，相对距离为 $\delta > 0$ 的纠错码是 Σ^m 的满足如下性质的非空子集 C：对任意 $x, y \in C$，若 $x \ne y$，则 $\mathsf{dist}(x, y) \ge \delta$。

在考虑编码问题的解码或距离时，我们常将问题限制在线性纠错码上，其定义如下。

定义 2.7 (线性纠错码)　$C \subseteq \Sigma^m$ 是字母表 Σ 上长度为 m 的纠错码，且 Σ^m 为线性空间，C 为其子空间，则 C 为线性纠错码。

2.4　问题定义

本小节定义本文中讨论的几个计算问题。为表一般性，本节中我们考虑对任意 $\gamma \geq 1$，定义所讨论的问题的 γ-近似版本，当近似比 $\gamma = 1$ 时即退化为精确求解的版本。

2.4.1　编码问题与格问题

我们定义 k-最近邻编码问题（k-Nearest Codeword Problem，下文中当在 $\mathbb{F}_p$ 上讨论该问题时，将其简记为 k-NCP$_p$）。

γ-Gap-k-NCP$_p$	
输入实例：	整数 d，多重向量集合 $V \subseteq \mathbb{F}_p^d$，向量 $\vec{t} \in \mathbb{F}_p^d$，整数 k。
参数：	k
问题：	区分以下两种情况：
(YES)	存在系数 $\alpha_1, \cdots, \alpha_n \in \mathbb{F}_p \backslash \{0\}$ 和向量 $\vec{w} \in \mathbb{F}_p^d$，$\|\|w\|\|_0 \leq k$，使得 $\alpha_1\vec{v}_1 + \cdots + \alpha_n\vec{v}_n + \vec{w} = \vec{t}$。
(NO)	对任意系数 $\alpha_1, \cdots, \alpha_n \in \mathbb{F}_p \backslash \{0\}$，令 $\vec{w} = \vec{t} - (\alpha_1\vec{v}_1 + \cdots + \alpha_n\vec{v}_n)$，则 $\|\|\vec{w}\|\|_0 > \gamma k$。

与 k-NCP$_p$ 等价的是 k-近似解码问题（k-Maximum Likelihood Decoding，下文中当在 $\mathbb{F}_p$ 上讨论该问题时，将其简记为 k-MLD$_p$）。

γ-Gap-k-MLD$_p$	
输入实例：	整数 d，多重向量集合 $V \subseteq \mathbb{F}_p^d$，向量 $\vec{t} \in \mathbb{F}_p^d$，整数 k。
参数：	k
问题：	区分以下两种情况：
(YES)	存在系数 $\alpha_1, \cdots, \alpha_k \in \mathbb{F}_p \backslash \{0\}$ 及 V 中不同向量 $\vec{v}_1, \cdots, \vec{v}_k \in V$ 使得 $\alpha_1\vec{v}_1 + \cdots + \alpha_k\vec{v}_k = \vec{t}$。
(NO)	对任意 $\ell < \gamma k$，任意 $\alpha_1, \cdots, \alpha_\ell \in \mathbb{F}_p \backslash \{0\}$ 及 V 中不同向量 $\vec{v}_1, \cdots, \vec{v}_\ell \in V$，$\alpha_1\vec{v}_1 + \cdots + \alpha_\ell\vec{v}_\ell \neq \vec{t}$。

注 k-NCP$_p$ 和 k-MLD$_p$ 本质上是给定一个目标 $\vec{t}$ 和一个线性码 $C \subseteq \mathbb{F}_p^d$，分别以线性码的生成矩阵（generating matrix）和奇偶校验矩阵（parity-check matrix）为视角讨论码字与 $\vec{t}$ 的距离。为了使正实例中的各系数均不为零，我们将输入的向量集合定义为多重集，并假设其中存在 $k-1$ 个全零向量。等价地，我们可以定义为简单集合，去除所有全零和重复向量，并将正实例规定为系数不全为零。

为了方便后面的归约过程，我们给出 k-MLD$_p$ 的一个特殊版本。容易证明该特殊版本和原版本是等价的。

γ-Gap-k-ColoredMLD$_p$

输入实例：整数 d，k 个多重向量集合 $V_1, \ldots, V_k \subseteq \mathbb{F}_p^d$，向量 $\vec{t} \in \mathbb{F}_p^d$，整数 k。

参数：k

问题：区分以下两种情况：

(YES) 存在 $\vec{v}_1 \in V_1, \ldots, \vec{v}_k \in V_k$ 使得 $\vec{v}_1 + \cdots + \vec{v}_k = \vec{t}$。

(NO) 对任意 $\ell < \gamma k$，任意 $\alpha_1, \cdots, \alpha_\ell \in \mathbb{F}_p \backslash \{0\}$ 及 $\vec{v}_1, \ldots, \vec{v}_\ell \in V_1 \cup \cdots \cup V_k$，$\alpha_1\vec{v}_1 + \cdots + \alpha_\ell\vec{v}_\ell \neq \vec{t}$。

注 注意到可以将 k-MLD$_p$ 的实例中每个向量 $\vec{v}$ 变为 $\vec{v}, 2\vec{v}, \cdots, (p-1)\vec{v}, \vec{0}$，并将向量集合复制 k 份即可得到等价的 γ-Gap-k-ColoredMLD$_p$ 实例。

反之，向 γ-Gap-k-ColoredMLD$_p$ 实例中 V_i 内的向量末尾添加一个 $\vec{e}_i$（只有第 i 位是 1，其余位全 0），再向 $\vec{t}$ 末尾添加长为 k 的全 1 向量，即可得到等价的 k-MLD$_p$ 实例。

前述 NCP 的齐次版本即为参数化最短距离问题（Minimum Distance Problem，定义如下。

γ-Gap-k-MDP$_p$

输入实例：整数 d，向量集合 $V = \{\vec{v}_1, \cdots, \vec{v}_n\} \subseteq \mathbb{F}_p^d$，整数 k。

参数：k

问题：区分以下两种情况：

(YES) 存在非全零的系数 $c_1, \cdots, c_n \in \mathbb{F}_p$ 使得 $||c_1\vec{v}_1 + \cdots + c_n\vec{v}_n||_0 \leq k$。

(NO) 对任意非全零的系数 $c_1, \cdots, c_n \in \mathbb{F}_p$，$||c_1\vec{v}_1 + \cdots + c_n\vec{v}_n||_0 > \gamma k$。

以下为两个和上述编码问题紧密相关的格问题。首先是参数化的最近向量问题（Closest Vector Problem，当使用 p-范数时简记为 CVP_p），该问题考虑给定的格和给定的向量在 p-范数意义下是否接近。

γ-Gap-k-CVP_p

输入实例：整数 d，整数向量集合 $V = \{\vec{v}_1, \cdots, \vec{v}_n\} \subseteq \mathbb{Z}^d$，目标向量 $\vec{t} \in \mathbb{Z}^d$，整数 k。

参数：k

问题：区分以下两种情况：

(YES) 存在系数 $c_1, \cdots, c_n \in \mathbb{Z}$ 和 $\vec{w} \in \mathbb{Z}^d$，$||w||_p^p \leq k$，使得 $c_1\vec{v}_1 + \cdots + c_n\vec{v}_n + \vec{w} = \vec{t}$。

(NO) 对任意 $c_1, \cdots, c_n \in \mathbb{Z}$，令 $\vec{w} = \vec{t} - (c_1\vec{v}_1 + \cdots + c_n\vec{v}_n)$，则 $||\vec{w}||_p^p > \gamma k$。

CVP 的齐次版本即为参数化最短向量问题（Shortest Vector Problem，简记为 SVP），定义如下。

γ-Gap-k-SVP_p

输入实例：整数 d，整数向量集合 $V = \{\vec{v}_1, \cdots, \vec{v}_n\} \subseteq \mathbb{Z}^d$，整数 k。

参数：k

问题：区分以下两种情况：

(YES) 存在非全零的系数 $c_1, \cdots, c_n \in \mathbb{Z}$ 使得 $||c_1\vec{v}_1 + \cdots + c_n\vec{v}_n||_p^p \leq k$。

(NO) 对任意非全零的系数 $c_1, \cdots, c_n \in \mathbb{Z}$，都有 $||c_1\vec{v}_1 + \cdots + c_n\vec{v}_n||_p^p > \gamma k$。

一些文献（例如[27]）中对 p 范数下格问题的定义为

$$||w||_p =\leq k,$$

这一定义与本文中使用的定义

$$||w||_p^p =\leq k,$$

在数值上有 p 次方的差异，但在其它方面性质一致，所以也是等价的。

2.4.2 约束满足问题

本小节介绍本文所讨论的参数化的 2-约束满足问题（2-Constraint Satisfaction Problem，简记为 2CSP）及其变种。

2CSP

输入实例： 变量集合 X，集合 $\Sigma = \dot{\bigcup}_{x\in X}\Sigma_x$，约束集合 Φ。每个 Σ_x 为变量 $x \in X$ 允许的取值，Φ 中的每个约束为 $\varphi_j = (x_{j_1}x_{j_2}, C_j)$，其中 x_{j_1}, x_{j_2} 为 X 中变量，C_j 为 $\Sigma_{x_{j_1}} \times \Sigma_{x_{j_2}}$ 的子集。

参数: $|X|$

问题: 区分以下两种情况：

(YES) 存在赋值 $\sigma : X \to \Sigma$ 使得

1. 对任意 $x \in X$，$\sigma(x) \in \Sigma_x$；
2. 对任意 $\varphi_j = (x_{j_1}x_{j_2}, C_j)$，$(\sigma(x_{j_1}), \sigma(x_{j_2})) \in C_j$

同时满足。

(NO) 所有满足条件 1 的赋值 $\sigma : X \to \Sigma$ 均不能同时满足所有约束。

对任意 $0 < \varepsilon \leq 1$，定义上述问题的近似版本如下

ε-Gap-2CSP

输入实例： 与 2CSP 相同。

参数: $|X|$

问题: 区分以下两种情况：

(YES) 与 2CSP 相同。

(NO) 所有满足条件 1 的赋值 $\sigma : X \to \Sigma$ 均只满足少于 ε 比例的约束。

上述近似问题是在满足的约束数量上的放松。我们还可以在“赋值”的定义上作放松。将前文中的赋值记为“单赋值”，我们定义多赋值（multi-assignment）及其满足性如下：

定义 2.8 (多赋值) 一个 2CSP 问题实例 $\Pi = (X, \Sigma, \Phi)$ 的多赋值是一个函数 $\hat{\sigma} : X \to 2^{\Sigma}$，且满足对任意 $x \in X$，$\hat{\sigma}(x) \subseteq \Sigma_x$。对约束 $\varphi_j = (x_{j_1}x_{j_2}, C_j)$，若存在 $a \in \hat{\sigma}(x_{j_1}), b \in \hat{\sigma}(x_{j_2})$ 使得 $(a, b) \in C_j$，则称 $\hat{\sigma}$ 满足约束 φ_j。

一个多赋值 $\hat{\sigma}$ 的个体大小定义为 $\max_{x\in X} |\hat{\sigma}(x)|$，总大小定义为 $\sum_{x\in X} |\hat{\sigma}(x)|$。

令 $r \geq 1$。对于一个 2CSP 实例 $\Pi = (X, \Sigma, \Phi)$，若存在一个个体大小不超过 r 的多赋值 $\hat{\sigma}$ 满足 Π，则称 Π 是*r-列表可满足的*（可以将多赋值看作一个列表）；若存在一个总大小不超过 $r \cdot |X|$ 的多赋值 $\hat{\sigma}$ 满足 Π，则称 Π 是*r-平均列表可满足的*。注意到满足 Π 的多赋值永远存在，因此我们的目标是优化满足 Π 的多赋值的大小。

我们定义判定这两种可满足性的问题如下：

r-List-2CSP	
输入实例：	与 2CSP 相同。
参数:	$\|X\|$
问题:	区分以下两种情况：
(YES)	该实例存在满足的单赋值。
(NO)	该实例不是 r-列表可满足的。

r-AvgList-2CSP	
输入实例：	与 2CSP 相同。
参数:	$\|X\|$
问题:	区分以下两种情况：
(YES)	该实例存在满足的单赋值。
(NO)	该实例不是 r-平均列表可满足的。

注　r-AvgList-2CSP 在文献[35]中也被写作 Avg-r-Gap-2CSP。我们使用这个名称是为了强调其多赋值属性，避免与 ε-Gap-2CSP 混淆。

2.4.3　其它相关问题

对 $r \geq 1$，定义 k-精确覆盖问题（k-ExactCover）的近似版本如下：

r-Gap-k-ExactCover	
输入实例：	整数 k，集合 U，一族 U 的子集 $\mathcal{S}$。
参数:	k
问题:	区分以下两种情况：
(YES)	$\mathcal{S}$ 中存在不超过 k 个不相交的集合，它们组成 U 的一个划分。
(NO)	U 不是任意 $r \cdot k$ 个 $\mathcal{S}$ 中集合的并。

我们给出下文中几个指数时间假设里所讨论的 3SAT 问题的定义。这也是计算复杂性和相关领域中一个核心问题。

3SAT	
输入实例：	一个含有 n 个变量，m 个子句的 3-合取范式 φ（即 φ 的每个子句包含 3 个变量）。
问题：	判定 φ 是否可满足。

2.5 复杂性假设

假设 2.1 (指数时间假设[4-5]，Exponential Time Hypothesis，以下简记为 ETH) 存在常数 $c_0 > 0$，有 n 个变量的 3SAT 实例无法在 $O(2^{c_0 n})$ 时间内解决。

文献[5]（参见[30]第 16.3 节）给出了一个算法，将任意 SAT 实例转化为等价但子句数量线性于变量数量的合取范式的并，称为“稀疏化引理”（Sparsification Lemma）。这一引理在 3SAT 实例的背景下可以如下表述。

引理 2.1 (稀疏化引理) 存在一个可计算函数 $g : \mathbb{N} \to \mathbb{N}$ 使得，对任意 $k, n \in \mathbb{N}$ 和任意含有 n 个变量的 3-合取范式 φ，存在一个公式

$$\beta = \bigvee_{i \in [t]} \varphi_i$$

满足：

1. β 与 φ 等价；
2. $t \leq 2^{n/k}$；
3. 每个 φ_i 中的子句都由原公式 φ 中某个子句至多删除一些变量得到，且每个变量至多在 φ_i 中出现 $g(k)$ 次。

另外，存在一个算法以整数 k, n 和公式 φ 为输入，在 $2^{n/k} \cdot |\varphi|^{O(1)}$ 时间内输出 β。

如果假设 2.1对于常数 $c_0 > 0$ 成立，我们可以取满足 $k > \frac{1}{c_0}$ 和 $2^{n/k} \cdot |\varphi|^{O(1)} \cdot 2^{c'n} < 2^{c_0 n}$ 的常数 $c > 0, k > 1$，应用引理 2.1得到至多 $t = 2^{n/k}$ 个 3-合取范式 $\{\varphi_i\}_{i \in [t]}$，且每个 φ_i 中至多有 $g(k) \cdot n = O(n)$ 个子句。由此，我们得到下面的等价假设。

假设 2.2 (稀疏化后的 ETH) 存在常数 $c > 0$，n 个变量，$O(n)$ 个子句的 3SAT 实例无法在 $O(2^{cn})$ 时间内解决。

同理，应用引理 2.1，我们可以给出随机算法下的指数时间假设如下。

假设 2.3 (随机指数时间假设) 存在常数 $c > 0$，n 个变量，$O(n)$ 个子句的 3SAT 实例无法被随机算法在 $O(2^{cn})$ 时间内解决。

假设 2.4 (参数不可近似假设[36]，Parameterized Inapproximability Hypothesis，**以下简记为** PIH)　存在常数 $\epsilon > 0$，ϵ-Gap-2CSP 没有 FPT 算法。

2.6　概率不等式

定理 2.2 (一致界，Union Bound)　对 $0 \le \epsilon_1, \cdots, \epsilon_n \le 1$，事件 $X_1, \cdots, X_n$ 满足

$$\Pr[X_i] \le \epsilon_i ,$$

有

$$\Pr[X_1 \wedge \cdots \wedge X_n] \le \sum_{i \in [n]} \epsilon_i 。$$

定理 2.3 (切尔诺夫不等式)　考虑一列独立同分布的随机变量 $X_1, X_2, \cdots, X_n \in \{0, 1\}$。记 $X = \sum_{i \in [n]} X_i$，$\mu = \mathbb{E}[X]$。对任意 $0 < \delta < 1$，

$$\Pr\left[X \le (1-\delta)\mu\right] \le \exp\left(-\frac{\mu\delta^2}{2}\right) 。$$

第 3 章 技术工具：纠错码的碰撞数

本章介绍后文中使用的技术工具：纠错码的碰撞数[23-24]（collision number）。

令 $C \subseteq \Sigma^m$ 为字母表 Σ 上长度为 m 的编码，x, y 是 C 的两个不同的码字。若对于某个 $i \in [m]$，$x[i] = y[i]$，则称 x, y 在第 i 位上碰撞（*collide*）。对 C 的非空子集 S，若存在不同的 $x, y \in S$ 且 $x[i] = y[i]$，则称 S 在第 i 位上碰撞。我们称 S 的碰撞集 $\mathrm{ColSet}(S)$ 为

$$\mathrm{ColSet}(S) = \{i \in [m] : S \text{ 在第 } i \text{ 位上碰撞}\}$$

定义 3.1 (编码的碰撞数) 对任意 $0 < \varepsilon \leq 1$，编码 C 的 ε-碰撞数 $\mathrm{Col}_\varepsilon(C)$ 定义为不超过 $|C| + 1$ 且满足以下条件的最大正整数 s：

对任意大小严格小于 s 的 C 的子集 S'，$|\mathrm{ColSet}(S')| \leq \varepsilon m$。

直观而言，对 $0 < \varepsilon \leq 1$，编码 C 的 ε-碰撞数 $\mathrm{Col}_\varepsilon(C)$ 指的是有超过 ε 比例的位置都存在不同的两个码字碰撞的码字集合所需要的大小。

3.1 相对距离与碰撞数

直觉上来说，对于相对距离为 δ 的编码 C，任两个码字 x, y 所能产生的碰撞位数至多是 $\delta \cdot m$。当 $\delta \ll \varepsilon$ 时，需要收集非常多的码字才能在 ε 比例位置产生碰撞。这一简单的直觉被下述定理证实，参见[23-24]。为了维护论文的完整性，本文在此附上证明。

定理 3.1 对任意常数 $0 < \varepsilon \leq 1$，相对距离为 δ（$0 < \delta < \varepsilon$）的纠错码 C 的碰撞数 $\mathrm{Col}_\varepsilon(C) \geq \sqrt{\frac{2\varepsilon}{1-\delta}}$。

证明 令 $S \subseteq C$ 满足 $|\mathrm{ColSet}(S)| > \varepsilon m$。对任意 $x, y \in S$，若 $x \neq y$，则

$$L_{x,y} = |\{i \in [m] : x[i] = y[i]\}| < (1 - \delta) \cdot m$$

但同时，由 S 的定义知

$$\sum_{x,y \in S, x \neq y} L_{x,y} > \varepsilon \cdot m$$

由于上述和式中至多有 $\binom{|S|}{2}$ 项，我们有

$$\binom{|S|}{2} \cdot (1-\delta)m > \varepsilon \cdot m$$

解之得

$$|S| > \sqrt{\frac{\varepsilon}{1-\delta}}$$

■

我们考虑常见的 Reed-Solomon 编码[37]如下。给定有限域 Σ，输入长度 r，输出长度 m，且 $r < m \leq |\Sigma|$，首先固定 Σ 中的 m 个不同元素 $\alpha_1, \alpha_2, \cdots, \alpha_m$。定义 $C^{RS} : \Sigma^r \to \Sigma^m$ 为：对任意输入 $x = (x_1, \cdots, x_r) \in \Sigma^r$，

$$C^{RS}(x) = \left(\sum_{j=1}^{r} x_j \cdot (\alpha_1)^{j-1}, \sum_{j=1}^{r} x_j \cdot (\alpha_2)^{j-1}, \cdots, \sum_{j=1}^{r} x_j \cdot (\alpha_m)^{j-1} \right) \in \Sigma^m$$

定理 3.2 (Reed-Solomon **编码的距离**[37])　$C^{RS} : \Sigma^r \to \Sigma^m$ 的相对距离至少为 $1 - \frac{r}{m}$。

定理 3.1和定理 3.2可以推出 Reed-Solomon 编码的碰撞数如下：

定理 3.3　对任意 $0 < \varepsilon \leq 1$，有限域 Σ，整数 r, m 满足 $r < m \leq |\Sigma|$，任意 Reed-Solomon 编码 $C^{RS} : \Sigma^r \to \Sigma^m$ 满足 $\mathrm{Col}_\varepsilon(C) \geq \sqrt{\frac{2\varepsilon m}{r}}$。

3.2　随机码的碰撞数

在后面的归约中，我们希望所用到的编码的碰撞数更大的同时编码长度更短。因此我们分析随机码的碰撞数，给出更优的参数，并分析上一节中从距离得到碰撞数这一方法的瓶颈所在。首先给出随机码的定义。

定义 3.2　令 $m > r > 0$ 为整数，对每个输入 $x \in \Sigma^r$，独立且均匀随机地选取 $y_x \in \Sigma^m$。定义随机码 $C^R : \Sigma^r \to \Sigma^m$ 为：

$$C^R(x) = y_x$$

引理 3.1　对任意常数 $0 < \varepsilon \leq 1$ 和随机码 $C^R : \Sigma^r \to \Sigma^m$，若 $m \geq 16\frac{1}{\varepsilon^2}|\Sigma|^{1/3} r \ln |\Sigma|$ 且 $|\Sigma| = \omega(1)$，则以概率 $1 - o(1)$，C^R 满足 $\mathrm{Col}_\varepsilon(C^R) > |\Sigma|^{1/3}$。

证明　我们证明 $\mathrm{Col}_\varepsilon(C^R) \leq |\Sigma|^{1/3}$ 的概率可以忽略不计。注意到事件

$$\text{“}\mathrm{Col}_\varepsilon(C^R) \leq |\Sigma|^{1/3}\text{”}$$

等价于事件

“存在大小为 $|\Sigma|^{1/3}$ 的集合 $S \subseteq C^R$，S 在超过 εm 个位置存在碰撞”。

接下来我们对该事件发生的概率的上限作估计。

首先，我们说明对随机码 C^R 的任意大小为 $|\Sigma|^{1/3}$ 的子集 S 以及任意位置 $i \in [m]$，S 不在第 i 位碰撞的概率很高。将 S 中元素列出为

$$S = \{x_1, \cdots, x_{|S|}\}。$$

注意到 “S 不在第 i 位碰撞” 意味着 $x_1[i], \cdots, x_{|S|}[i]$ 均不相同。对 $1 \le j \le |S|$，定义事件 E_j 为

$$“x_j[i] \text{ 不在 } \{x_c[i]\}_{1 \le c < j} \text{ 中”},$$

则前述事件等价于

$$E_1 \wedge \cdots \wedge E_{|S|}。$$

该事件发生的概率下限为：

$$\begin{aligned}
&\Pr[x_1[i], \cdots, x_{|S|}[i] \text{均不相同}] \\
=&\Pr[E_1 \wedge \cdots \wedge E_{|S|}] \\
=&\Pr[E_1] \cdot \Pr[E_2 \mid E_1] \cdots \Pr[E_S \mid E_1 \wedge \cdots \wedge E_{|S|-1}] \\
=&1 \cdot \frac{|\Sigma| - 1}{|\Sigma|} \cdots \frac{|\Sigma| - (|S| - 1)}{|\Sigma|} \qquad \text{(条件概率中每一项减少一种选择)} \\
\ge&\left(\frac{|\Sigma| - |\Sigma|^{1/3}}{|\Sigma|}\right)^{|\Sigma|^{1/3}} \\
=&\left(1 - \frac{1}{|\Sigma|^{2/3}}\right)^{|\Sigma|^{1/3}} \\
=&\left(1 - \frac{1}{|\Sigma|^{2/3}}\right)^{|\Sigma|^{2/3} \cdot \frac{1}{|\Sigma|^{1/3}}} \\
\ge&\left(\frac{1}{4}\right)^{\frac{1}{|\Sigma|^{1/3}}} \\
=&1 - o(1)
\end{aligned}$$

其中最后一个不等式成立的原因是对 $n \ge 2$，

$$(1 - \frac{1}{n})^n \ge \frac{1}{4}。$$

记上述概率为 Δ。

之后，我们计算大小为 $|\Sigma|^{1/3}$ 的 S 在超过 εm 个位置均存在碰撞的概率上限。令 B_i 为表示事件“S 不在第 i 位碰撞”的指示变量，且记 $B = \sum_{i=1}^{m} B_i$ 为 S 不产生碰撞的位置数。则 S 在超过 εm 个位置碰撞等价于

$$B < (1-\varepsilon)m。$$

由期望的线性性质，

$$\mathbb{E}[B] = \sum_{i=1}^{m} \mathbb{E}[B_i] = \Delta m.$$

由随机码的性质可知，$B_1, \cdots, B_m$ 相互独立。应用切尔诺夫不等式（定理 2.3）有：

$$\begin{aligned}
\Pr[B \le (1-\varepsilon)m] &= \Pr[B \le \Delta m - (\Delta - 1 + \varepsilon)m] \\
&\le \exp\left(-\frac{(\Delta - 1 + \varepsilon)^2 \Delta m}{2\Delta^2}\right) \\
&= \exp\left(-\frac{(\Delta - 1 + \varepsilon)^2}{2\Delta} m\right) \\
&\le \exp\left(-\frac{(\Delta - 1 + \varepsilon)^2}{2} m\right) && (\text{由于} \Delta \le 1) \\
&\le \exp\left(-\frac{1}{8}\varepsilon^2 m\right) && (\text{由于} \Delta - 1 + \varepsilon = \varepsilon - o(1) \ge \frac{1}{2}\varepsilon).
\end{aligned}$$

至多存在

$$(|\Sigma|^r)^{|\Sigma|^{1/3}}$$

个大小为 $|\Sigma|^{1/3}$ 的 C^R 的子集，对所有这些子集应用一致界（定理 2.2）得：

$$\begin{aligned}
\Pr[\mathrm{Col}_\varepsilon(C_R) \le |\Sigma|^{1/3}] &\le (|\Sigma|^r)^{|\Sigma|^{1/3}} \cdot \exp\left(-\frac{1}{8}\varepsilon^2 m\right) \\
&= e^{|\Sigma|^{1/3} \ln |\Sigma| r - \frac{1}{8}\varepsilon^2 m} \\
&\le e^{-|\Sigma|^{1/3} \ln |\Sigma| r} \\
&= o(1)
\end{aligned}$$

其中最后一个不等式因为

$$m \ge 16 \frac{1}{\varepsilon^2} |\Sigma|^{1/3} \ln |\Sigma| r$$

成立。由此我们论证了以 $1 - o(1)$ 的概率，

$$\mathrm{Col}_\varepsilon(C^R) > |\Sigma|^{1/3}。$$

■

通过调整引理 3.1中的参数，我们得到如下的高碰撞数编码构造。

引理 3.2 对任意常数 $c > 0$ 和 $0 < \varepsilon < 1$，存在随机算法满足：输入正整数 n, k，算法构造出一个编码 $C^R \subseteq \Sigma^m$，参数满足 $|C| = n, |\Sigma| = O(k^3), m = O(k \log n)$，且以高概率满足 $\mathrm{Col}_\varepsilon(C^R) > ck$。该算法的运行时间为 $O(nm|\Sigma|)$。

证明 算法即为从 Σ^m 中独立随机挑选 $|C| = n$ 个字符串作为编码（也等价于编码的每一位独立随机挑选）。因此，运行时间显然为

$$O(nm|\Sigma|)。$$

令

$$|\Sigma| = ck^3 = O(k^3),$$

且记参数

$$r = \log n / \log |\Sigma|$$

使得 $|\Sigma|^r = n$。令

$$m = 16\frac{1}{\varepsilon^2}|\Sigma|^{1/3} \ln |\Sigma| r = O(k \log n),$$

如此生成一个随机码

$$C^R : \Sigma^r \to \Sigma^m。$$

由引理 3.1，以概率 $1 - o(1)$，随机码 C^R 满足

$$\mathrm{Col}_\varepsilon(C^R) > |\Sigma|^{1/3} = ck。$$

■

注 注意到使用几乎相同的分析方式，我们可以将引理 3.1推广到对任意大于等于 3 的整数 t，若 $m > \Omega(|\Sigma|^{1/t} \log |\Sigma| r)$ 及 $|\Sigma| = \omega(1)$，则以很高概率满足

$$\mathrm{Col}_\varepsilon(C^R) > |\Sigma|^{1/t}。$$

同理，对大于 3 的常数 t，令 $|\Sigma| = \Omega(k^t)$，则引理 3.2可以被推广到参数一致但字母表更大的编码上。

接下来是我们后文归约中使用到的一个 “合并” 的步骤。这一步使我们可以枚举字母表很小的编码中的多个位置，以降低归约过程中的参数增长。

引理 3.3　对任意常数 $c > 1$ 和 $0 < \varepsilon < 1$，存在随机算法满足：输入正整数 n, k，算法构造出一个编码 $C \subseteq \Sigma^m$，参数满足 $|C| = n, |\Sigma| = O(n^{1/k}), m = O(k^2 \log k)$，且以高概率满足 $\mathrm{Col}_\varepsilon(C) > ck$。该算法的运行时间为 $O(n^{1+1/k} \cdot k^2 \log k)$。

证明　根据引理 3.2，使用相同的参数 n, k，我们构造编码 $C' \subseteq (\Sigma')^{m'}$，其中

$$|C| = n, |\Sigma'| = O(k^3), m' = O(k \log n),$$

且以概率 $1 - o(1)$ 满足

$$\mathrm{Col}_\varepsilon(C') > ck。$$

令 g 为一待定整数。我们的主要思路是将 C' 中每 g 个位置合并为一个更大的字母表中的单独位置。具体来说，令 $\Sigma = (\Sigma')^g$ 以及 $m = m'/g$，我们构造新的编码 $C \subseteq \Sigma^m$ 如下。

对每一个 $w' \in C'$，我们令 C 中包含一个码字 $w \in \Sigma^m$。对 $i \in [m]$，w 的第 i 位定义为

$$w[i] = (w'[ig], w'[ig + 1], \ldots, w'[ig + g - 1]).$$

若 $\mathrm{Col}_\varepsilon(C') > h$，我们接下来证明 $\mathrm{Col}_\varepsilon(C) > h$。由碰撞数的定义（定义 3.1），我们只需证明对任意 C 的子集 S，若 S 在超过 εm 个位置产生碰撞，则 $|S| > h$。令 S 为在超过 εm 个位置产生碰撞的子集。对任意 S 产生碰撞的位置 i，存在 $w_1, w_2 \in S$ 且

$$w_1[i] = w_2[i]。$$

由于 C 由 C' 中的码字合并中间位置而来，令 S' 为 S 中码字在编码 C' 中对应的码，$w_1', w_2' \in S'$ 为 w_1, w_2 对应的 C' 的码字，则对所有 $ig \le j < (i + 1)g$，有

$$w_1'[j] = w_2'[j]。$$

由上述分析，我们得到 S' 在这些位置 j 上产生碰撞。由于 S 在超过 ε 比例的位置产生碰撞，S' 自然也如此，从而

$$|S'| \ge \mathrm{Col}_\varepsilon(C') > h。$$

由 C 的构造我们知道，S 和 S' 中的码字可以一一对应，从而 $|S| = |S'| \ge h$，即 $\mathrm{Col}_\varepsilon(C) > h$。结合引理 3.2，以概率 $1 - o(1)$ 满足

$$\mathrm{Col}_\varepsilon(C) > ck。$$

最后，我们将待定常数 g 取为

$$g = \frac{\log n}{k \log |\Sigma'|},$$

则

$$|\Sigma| = (|\Sigma'|)^g = O(n^{1/k}), \quad m = m'/g = O(k^2 \log k).$$

算法的运行时间由引理 3.2代入相应参数即得。 ■

3.3 相关工作中碰撞数分析的瓶颈

现有两种方法分析一个随机编码有好的的碰撞数，第一种即为前文中的分析路径，第二种是先证明随机码有很好的相对距离，再使用相关工作[23-24]中对碰撞数分析的方法（即本文定理 3.1）论证碰撞数的下界。第一种方法给出的编码长度为 $O(k^2 \log k)$，本节简要分析第二种方法瓶颈所在，并指出应用此类方法的编码长度是 $\Omega(k^3)$。

后文中我们需要一个碰撞数是 $\Omega(k)$ 的编码 C。由定理 3.1，C 的相对距离至少应该是

$$\delta \geq 1 - \frac{1}{\Omega(k^2)}.$$

我们引入编码理论中的辛格尔顿界（Singleton Bound）如下：

定理 3.4 任意相对距离为 δ 的编码 $C : \Sigma^r \to \Sigma^m$ 须满足 $r \leq m - \delta m + 1$。

注意到定理 3.4适用于任意编码。关于它的详细讨论和证明可参见 Guruswami 等人的专著[38]的第 4 章第 3 节部分。对我们使用的编码参数应用定理 3.4，有

$$m - (1 - 1/\Omega(k^2))m + 1 \geq r,$$

即

$$m \geq \Omega(k^2) r.$$

后文中对 MLD 的归约需要给每个输入向量指派一个唯一的码字，因此要求 $C \geq n$，即 $|\Sigma|^r \geq n$，则

$$r \geq \frac{\log n}{\log |\Sigma|}.$$

最后，考虑引理 3.3中的“合并”过程，编码被合并到 m' 个位置中，每个新的位置包含原来的 $g = \frac{m}{m'}$ 个位置，则 k 个新字母表的所有可能数量是

$$\begin{aligned}(|\Sigma|^g)^k =& |\Sigma|^{k\frac{m}{m'}} \\ =& 2^{k\frac{m}{m'}\log|\Sigma|} \\ \geq& 2^{\Omega(k\cdot k^2\frac{\log n}{\log|\Sigma|}\log|\Sigma|)/m'} \\ =& n^{\Omega(k^3)/m'}.\end{aligned}$$

为了高效枚举所有 k 种组合，上述的大小需要是 n 的多项式，则 $m' = \Omega(k^3)$。该下界是紧的，因为注意到 Reed-Solomon 码已经可以达到 $m' = O(k^3)$。

3.4　本章小结

本章介绍了本论文的核心技术工具——纠错码的碰撞数。这一概念直观来源于需要“覆盖”住常数比例的位置所需要选择的不同码字对的数量，且由此在定理 3.1中给出了编码的距离与碰撞数的关系。之后，我们在第 3.2节探索了直接分析随机码的碰撞数的方式，并在第 3.3节简单讨论了从距离导出碰撞数这一方法在编码理论上的局限性，从而显示出随机码这一方法的优越性。

第 4 章 编码与格问题

本章讨论几个参数化编码问题和格问题的近似困难性。我们首先给出一个对 k-近似解码问题的创造间隔的归约，建立起近似该问题的困难性，再使用已有的归约将困难性推广到其他问题上。

4.1 k-MLD 的近似困难性

我们先给出一个初步的归约构造，该构造可以清晰地展示我们归约的思路。该构造将输入的 k-ColoredMLD$_p$ 实例转化为一个特殊的 k-MLD$_p$ 实例。输出实例的向量被分为两部分，一个可行解需要在两部分中各选取一定数量的向量，且在坏的情况下，任意可行解需要在其中一部分里选择常数比例的向量。由于两部分内向量数量不一致，在真正的归约中我们将对它进行一些修改以符合要求。

引理 4.1 存在算法满足：输入整数 n, k, d, m，k 个大小为 n 的向量集合 $V_1, \cdots, V_k \subseteq \mathbb{F}_p^d$，目标向量 $\vec{t} \in \mathbb{F}_p^d$ 以及编码 $C \subseteq |\Sigma|^m$，要求 C 满足 $|C| = n$ 且 $\mathrm{Col}_\varepsilon(C) \geq ck$，输出向量集合 $A = A_1 \dot\cup \cdots \dot\cup A_k \subseteq \mathbb{F}_p^D$ 和 $B = B_1 \dot\cup \ldots \dot\cup B_m \subseteq \mathbb{F}_p^D$ 及目标向量 $\vec{t}' \in \mathbb{F}_p^D$，其中参数 $D = O(d + mk|\Sigma|)$。算法运行时间是 $O(dm^2k^2|\Sigma|(n + |\Sigma|^k))$，且满足：

1. 若存在 $\vec{v}_1 \in V_1, \ldots, \vec{v}_k \in V_k$ 满足 $\sum_{i\in[k]} \vec{v}_i = \vec{t}$，则存在向量 $\vec{a}'_1 \in A_1, \cdots, \vec{a}'_k \in A_k$ 及 $\vec{b}'_1 \in B_1, \cdots, \vec{b}'_m \in B_m$ 求和为 $\vec{t}'$；
2. 若对任意向量 $\vec{v}_1 \in V_1, \ldots, \vec{v}_k \in V_k$ 和系数 $\alpha_1, \ldots \alpha_k \in \mathbb{F}_p^+$，其线性组合 $\alpha_1\vec{v}_1 + \cdots + \alpha_k\vec{v}_k \neq \vec{t}$，则对任意向量的子集 $X \subseteq A \dot\cup B$ 和系数 $\lambda : X \to \mathbb{F}_p^+$，若 $\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x} = \vec{t}'$，则至少满足下述两条之一：
   - $|X \cap A| \geq ck$ 且 $|X \cap B| \geq m$；
   - $|X \cap A| \geq k$ 且 $|X \cap B| \geq 2(1-\varepsilon)m$。

证明 输出的向量的维度（即长度）为 $D = d + mk|\Sigma| + k + m$。我们把输出向量的维度分为四部分，各自的长度分别是 $d, mk|\Sigma|, k, m$。形式化地说，对输出的向量 $x \in \mathbb{F}_p^D$，我们记：

- $\vec{x}^{(1)} \in \mathbb{F}_p^d$ 为第 1 部分；
- $\vec{x}^{(2)} \in \mathbb{F}_p^{mk|\Sigma|}$ 为第 2 部分；
- $\vec{x}^{(3)} \in \mathbb{F}_p^k$ 为第 3 部分；

- $\vec{x}^{(4)} \in \mathbb{F}_p^m$ 为第 4 部分。

进一步地，我们将长为 $mk|\Sigma|$ 的第 2 部分分为 m 个小部分，每个小部分长为 $k|\Sigma|$，即记作

$$\vec{x}^{(2)} = \vec{x}^{(2,1)} \circ \cdots \circ \vec{x}^{(2,m)}。$$

令 $\vec{e}_i$ 为第 i 位内容是 1，其它位置内容均为 0 的向量。为简化记号，$\vec{e}_i$ 的维度（长度）不再单独写出，而是取决于具体的语境。再令 $\iota : \Sigma \to [|\Sigma|]$ 为一个多项式时间计算的双射，且对任意 $\sigma \in \Sigma$，我们记

$$\vec{e}_\sigma = (\overbrace{0, \ldots, 0}^{\iota(\sigma)-1}, 1, \underbrace{0 \ldots, 0}_{|\Sigma|})。$$

在开始正式描述我们对向量的构造前，我们先给出一个简化的图示，帮助读者直观地理解我们的构造，见图 4.1。

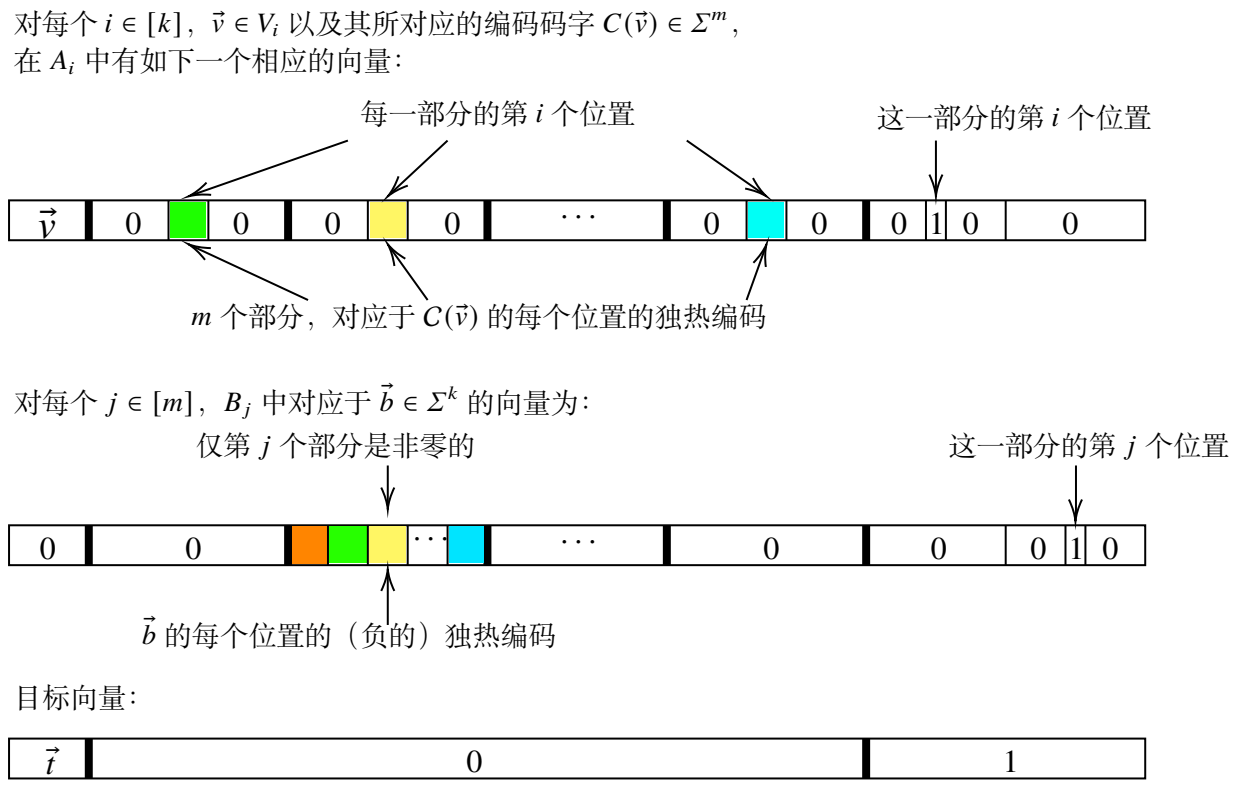

图 4.1　主要构造的简化图示。具体详细的构造请参见图 4.2.

构造向量集合 A：我们描述从原来的 $V_1, \cdots V_k$ 构造向量集合 A 的过程。对每个 V_i，注意到 $|V_i| = n$ 且 $|C| = n$，我们向 V_i 中的每个向量 $\vec{v}$ 指派一个唯一的码字，记作 $C(\vec{v})$。对任意 $i \in [k]$ 和 $\vec{v} \in V_i$，我们构造一个向量 $\vec{a}_{i,\vec{v}} \in \mathbb{F}_p^D$ 如下：

- $\vec{a}_{i,\vec{v}}^{(1)} = \vec{v}$;
- 对每个 $j \in [m]$，$\vec{a}_{i,\vec{v}}^{(2,j)} = (\overbrace{\vec{0},\ldots,\vec{0}}^{(i-1)},\underbrace{\vec{e}_{C(\vec{v})[j]},\vec{0},\ldots,\vec{0}}_{k})$;
- $\vec{a}_{i,\vec{v}}^{(3)} = \vec{e}_i$;
- $\vec{a}_{i,\vec{v}}^{(4)} = \vec{0}_m$。

最后，令

$$A_i = \{\vec{a}_{i,\vec{v}} \mid \vec{v} \in V_i\} \text{ 且 } A = A_1 \cup \cdots \cup A_k.$$

构造向量集合 B： 对每个 $j \in [m]$ 和 $\vec{\sigma} = (\sigma_1,\ldots,\sigma_k) \in \Sigma^k$，构造向量 $\vec{b}_{j,\vec{\sigma}} \in \mathbb{F}_p^D$ 如下：

- $\vec{b}_{j,\vec{\sigma}}^{(1)} = \vec{0}_d$;
- $\vec{b}_{j,\vec{\sigma}}^{(2,j)} = (-\vec{e}_{\sigma_1},\cdots-\vec{e}_{\sigma_k})$;
- 对每个 $j' \in [m]\backslash\{j\}$，$\vec{b}_{j,\vec{\sigma}}^{(2,j')} = \vec{0}_k$;
- $\vec{b}_{j,\vec{\sigma}}^{(3)} = \vec{0}_k$;
- $\vec{b}_{j,\vec{\sigma}}^{(4)} = \vec{e}_j$。

最后，令

$$B_j = \{\vec{b}_{j,\vec{\sigma}} \mid \vec{\sigma} \in \Sigma^k\} \text{ 且 } B = B_1 \cup \cdots \cup B_m$$

最终，令目标向量 $\vec{t}'$ 为：

- $\vec{t}'^{(1)} = \vec{t}$;
- $\vec{t}'^{(2)} = \vec{0}_{mk|\Sigma|}$;
- $\vec{t}'^{(3)} = \vec{1}_k$;
- $\vec{t}'^{(4)} = \vec{1}_m$。

时间复杂度： 构造 A 中的每个向量需要

$$O(d + mk|\Sigma| + km) = O(d + mk|\Sigma|)$$

步，因此构造集合 A 的时间是

$$O(dkn + mk^2n|\Sigma|)。$$

构造 B 中的每个向量也需要

$$O(d + mk|\Sigma|)$$

步，因此构造集合 B 的时间是

$$O(dm|\Sigma|^k + m^2k|\Sigma|^{k+1})。$$

最终整个算法的运行时间是

$$O(dm^2k^2|\Sigma|(n+|\Sigma|^k))。$$

性质 1 的证明：假设存在 $\vec{v}_1 \in V_1, \cdots, \vec{v}_k \in V_k$ 满足

$$\sum_{i\in[k]} \vec{v}_i = \vec{t}。$$

对每个 $i \in [k]$，我们选择向量 $\vec{a}_{i,\vec{v}_i} \in A_i$。对每个 $j \in [m]$，我们选择向量 $\vec{b}_{j,\vec{\sigma}_j} \in B_j$，其中

$$\vec{\sigma}_j = (C(\vec{v}_1)[j], \ldots, C(\vec{v}_m)[j]) \in \Sigma^k。$$

我们将论证

$$\sum_{i\in[k]} \vec{a}_{i,\vec{v}_i} + \sum_{j\in[m]} \vec{b}_{j,\vec{\sigma}_j} = \vec{t}'。$$

- 在第 1 部分中，

$$\sum_{i\in[k]} \vec{a}^{(1)}_{i,\vec{v}_i} + \sum_{j\in[m]} \vec{b}^{(1)}_{j,\vec{\sigma}_j} = \sum_{i\in[k]} \vec{v}_i + \sum_{j\in[m]} \vec{0}_d = \vec{t} = \vec{t}'^{(1)}.$$

- 在第 2 部分中，对每个 $j \in [m]$，考虑第 $(2, j)$ 部分，

$$\begin{aligned}
&\sum_{i\in[k]} \vec{a}^{(2,j)}_{i,\vec{v}_i} + \sum_{j'\in[m]} \vec{b}^{(2,j)}_{j',\vec{\sigma}_{j'}} \\
=& \sum_{i\in[k]} \vec{a}^{(2,j)}_{i,\vec{v}_i} + \vec{b}^{(2,j)}_{j,\vec{\sigma}_j} \\
=& \sum_{i\in[k]} (\overbrace{\vec{0}, \ldots, \vec{0}}^{i-1}, \vec{e}_{C(\vec{v}_i)[j]}, \vec{0}, \ldots, \vec{0}) + (-\vec{e}_{C(\vec{v}_1)[j]}, \ldots, -\vec{e}_{C(\vec{v}_k)[j]}) \\
=& (\vec{e}_{C(\vec{v}_1)[j]}, \ldots, \vec{e}_{C(\vec{v}_k)[j]}) + (-\vec{e}_{C(\vec{v}_1)[j]}, \ldots, -\vec{e}_{C(\vec{v}_k)[j]}) \\
=& \vec{0}_{k|\Sigma|} \\
=& \vec{t}'^{(2,j)}.
\end{aligned}$$

- 在第 3 部分中，

$$\sum_{i\in[k]} \vec{a}^{(3)}_{i,\vec{v}_i} + \sum_{j\in[m]} \vec{b}^{(3)}_{j,\vec{\sigma}_j} = \sum_{i\in[k]} \vec{e}_i + \sum_{j\in[m]} \vec{0}_k = \vec{1}_k = \vec{t}'^{(3)}.$$

- 在第 4 部分中，

$$\sum_{i\in[k]} \vec{a}^{(4)}_{i,\vec{v}_i} + \sum_{j\in[m]} \vec{b}^{(4)}_{j,\vec{\sigma}_j} = \sum_{i\in[k]} \vec{0}_m + \sum_{j\in[m]} \vec{e}_j = \vec{1}_m = \vec{t}'^{(4)}.$$

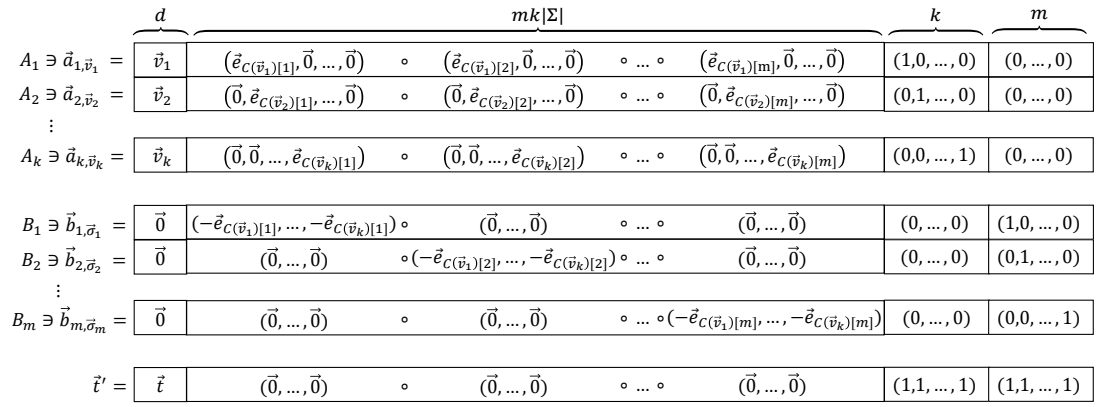

图 4.2 引理 4.1所构造向量的图示。在性质 1 的假设下，可以选择 $\vec{b}_{j,\vec{\sigma}_j}$ 为 $\vec{\sigma}_j = (C(\vec{v}_1)[j], \cdots, C(\vec{v}_k)[j])$。

向量集合 A, B 以及目标向量 $\vec{t}'$ 的直观构造，以及性质 1 的直观证明参见图 4.2。

性质 2 的证明： 对任意 $X \subseteq A \cup B$ 和 $\lambda : X \to \mathbb{F}_p^+$，若它们满足 $\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x} = \vec{t}'$，可以注意到该向量等式的第 3 部分向量满足：

$$\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x}^{(3)} = \sum_{i\in[k]} \sum_{\vec{x}\in X\cap A_i} \lambda(\vec{x})\vec{e}_i = \vec{1}_m = \vec{t}'^{(3)}$$

对每个 $i \in [k]$，$X \cap A_i$ 必然非空，因为 $\sum_{\vec{x}\in X\cap A_i} \lambda(\vec{x}) = 1$。同理，观察等式的第 4 部分可得对每个 $j \in [m]$，$X \cap B_j$ 也必然非空。因此

$$|X \cap A| \geq k \quad \text{且} \quad |X \cap B| \geq m。$$

下面讨论性质 2 中的假设“对任意向量 $\vec{v}_1 \in V_1, \dots, \vec{v}_k \in V_k$ 和系数 $\alpha_1, \dots \alpha_k \in \mathbb{F}_p^+$，其线性组合 $\alpha_1\vec{v}_1 + \cdots + \alpha_k\vec{v}_k \neq \vec{t}$” 成立的情况。令 $I \subseteq [m]$ 为满足 $X \cap B_j$ 中只有一个向量的下标 j 的集合，即

$$I = \{j \in [m] : |X \cap B_j| = 1\}$$

由于对每个 $j \in [m]$，$|X \cap B_j| \geq 1$，若 $|I| \leq \varepsilon m$，则

$$|X \cap B| \geq \sum_{j\in[m]\setminus I} |X \cap B_j| \geq 2(1-\varepsilon)m$$

即性质 2 的第 2 种情况成立。我们接下来证明当第 2 种情况不成立时，即 $|I| > \varepsilon m$ 时，$|X \cap A| \geq ck$，第一种情况成立。

首先我们注意到必然存在一个位置 $i \in [k]$ 使得 $X \cap A_i$ 包含多于一个向量。假设不然，即对每个 $i \in [k]$，$|X \cap A_i| = 1$，记

$$\vec{a}_{i,\vec{v}_i} \in X \cap A_i$$

为 $X \cap A_i$ 中唯一的那个向量。注意到 $X \cap B$ 中的向量在第 1 部分全零，所以 X 中向量在第 1 部分求和的结果是

$$\begin{aligned}\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x}^{(1)} &= \sum \lambda(\vec{a}_{i,\vec{v}_i})\vec{a}'^{(1)}_{i,\vec{v}_i} \\ &= \sum_{i\in[k]} \lambda(\vec{a}_{i,\vec{v}_i})\vec{v}_i \\ &= \vec{t} \\ &= \vec{t}'^{(1)}\end{aligned}$$

显然与性质 2 中的假设矛盾。我们记 $i^* \in [k]$ 为满足 $|X \cap A'_{i^*}| > 1$ 的位置（若有多个则任选其一），再记 $\ell = |X \cap A'_{i^*}| > 1$，我们接下来论证 $\ell \geq ck$。记

$$X \cap A_{i^*} = \{\vec{a}_{i^*,\vec{v}_1}, \ldots, \vec{a}_{i^*,\vec{v}_\ell}\},$$

其中 $\vec{v}_1, \ldots \vec{v}_\ell \in V_{i^*}$。我们说明码字集合

$$\{C(\vec{v}_1), \ldots, C(\vec{v}_\ell)\}$$

在每个 $j \in I$ 位置上都产生碰撞。

对任意 $j \in I$，令 $\vec{b}_{j,\vec{\sigma}}$ 为 $X \cap B_j$ 中唯一的那个向量，且 $\vec{\sigma} = (\sigma_1, \ldots, \sigma_k)$。由于向量的第 $(2, j)$ 部分长为 $k|\Sigma|$，我们将它再次分为 k 个小部分，每部分长为 $|\Sigma|$，并考虑第 $(2, j, i^*)$ 部分的和式：

$$\begin{aligned}\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x}^{(2,j,i^*)} &= \lambda(\vec{a}_{i^*,\vec{v}_1})\vec{a}^{(2,j,i^*)}_{i^*,\vec{v}_1} + \cdots + \lambda(\vec{a}_{i^*,\vec{v}_\ell})\vec{a}^{(2,j,i^*)}_{i^*,\vec{v}_\ell} + \lambda(\vec{b}_{j,\vec{\sigma}})\vec{b}^{(2,j,i^*)}_{j,\vec{\sigma}} \\ &= \lambda(\vec{a}_{i^*,\vec{v}_1})\vec{e}_{C(\vec{v}_1)[j]} + \cdots + \lambda(\vec{a}_{i^*,\vec{v}_\ell})\vec{e}_{C(\vec{v}_\ell)[j]} - \lambda(\vec{b}_{j,\vec{\sigma}})\vec{e}_{\sigma_{i^*}} \\ &= \vec{0}_{|\Sigma|} = \vec{t}'^{(2,j,i^*)}.\end{aligned}$$

若 $C(\vec{v}_1)[j], \ldots, C(\vec{v}_\ell)[j]$ 均不相同，则等式

$$\lambda(\vec{a}_{i^*,\vec{v}_1})\vec{e}_{C(\vec{v}_1)[j]} + \cdots + \lambda(\vec{a}_{i^*,\vec{v}_\ell})\vec{e}_{C(\vec{v}_\ell)[j]} - \lambda(\vec{b}_{j,\vec{\sigma}})\vec{e}_{\sigma_{i^*}} = \vec{0}_{|\Sigma|}$$

显然无法成立。因此 $\{C(\vec{v}_1), \ldots, C(\vec{v}_\ell)\}$ 在第 j 个位置产生碰撞。

由于 $|I| > \varepsilon m$, $\{C(\vec{v}_1), \ldots, C(\vec{v}_\ell)\}$ 在超过 εm 个位置, 由碰撞数的定义（定义 3.1）,

$$\begin{aligned} |\{C(\vec{v}_1), \ldots, C(\vec{v}_\ell)\}| &\geq \mathrm{Col}_\varepsilon(C) \\ &\geq ck, \end{aligned}$$

由此可得

$$|X \cap A| > |X \cap A_{i^*}| \geq ck。$$

■

由于我们所构造的具有高碰撞数的编码长度为 $m = O(k^2 \log k)$（参见引理 3.3）, 远大于 k，引理 4.1的构造不能直接给出一个 k-MLD 的创造间隔的归约。直观上来说，我们可以将 A 复制足够多次使得 A, B 的大小达到同一量级。我们将这一直观形式化地写为下面的定理。

定理 4.1 对任意 $0 < \varepsilon < 1$，存在一个随机化的归约满足：输入整数 n, k, d，k 个大小为 n 的向量集合 $V_1, \cdots, V_k \subseteq \mathbb{F}_p^d$，目标向量 $\vec{t} \in \mathbb{F}_p^d$，输出向量集合 $U_1, \cdots, U_{k'} \subseteq \mathbb{F}_p^D$ 及目标向量 $\vec{t}' \in \mathbb{F}_p^D$，其中参数 $k' = O(k^2 \log k)$，$D = O(k'd + k'^2 n^{1/k})$。归约算法运行时间是 $O(d2^{O(k)} n^{1+1/k})$，且满足：

1. 若存在 $\vec{v}_1 \in V_1, \ldots, \vec{v}_k \in V_k$ 使得 $\sum_{i\in[k]} \vec{v}_i = \vec{t}$，则存在向量 $\vec{u}_1 \in U_1, \cdots, \vec{u}_{k'} \in U_{k'}$，它们的求和为 $\vec{t}'$；
2. 若对任意向量 $\vec{v}_1 \in V_1, \ldots, \vec{v}_k \in V_k$ 和系数 $\alpha_1, \ldots \alpha_k \in \mathbb{F}_p^+$，其线性组合 $\alpha_1\vec{v}_1 + \cdots + \alpha_k\vec{v}_k \neq \vec{t}$，则对任意向量的子集 $X \subseteq \bigcup_{i\in[k']} U_i$ 和系数 $\lambda : X \to \mathbb{F}_p^+$，若 $\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x} = \vec{t}'$，则 $|X| \geq (\frac{3}{2} - \varepsilon)k'$。

证明 调用引理 3.3并设定其中的参数 $c = 2$，可以在

$$O(n^{1+1/k} \cdot k^2 \log k)$$

的时间内输出一个随机编码 $C \subseteq \Sigma^m$，其中 $|C| = n, |\Sigma| = O(n^{1/k}), m = O(k^2 \log k)$，且以高概率满足

$$\mathrm{Col}_\varepsilon(C) > 2k。$$

以 C 为要用的编码，调用引理 4.1，我们在

$$O(dm^2k^2|\Sigma|(n + |\Sigma|^k)) = O(d \cdot 2^{O(k)} n^{1+1/k})$$

的时间内得到向量集合

$$A_1, \ldots, A_k, B_1, \ldots, B_m \subseteq \mathbb{F}_p^{D'}$$

和目标向量

$$\vec{t}' \in \mathbb{F}_p^{D'},$$

其中

$$D' = O(d + km|\Sigma|) = O(d + kmn^{1/k})。$$

记 $w = \frac{m}{k}$，我们将目前所得的向量维度再拉长至 w 倍，即

$$D = wD' = O(mk + m^2 n^{1/k})。$$

算法最终输出 $k' = 2m$ 个向量集合 $U_1, \cdots, U_{k'}$ 和目标向量 $\vec{t}''$。为便于后文分析，我们将这 k' 个集合分为两部分：$\{A'_{\ell,i}\}_{\ell\in[w],i\in[k]}$ 和 $\{B'_j\}_{j\in[m]}$。其中

$$U_{(\ell-1)k+i} = A'_{\ell,i} = \{(\overbrace{\underbrace{\vec{0}_{D'}, \ldots, \vec{0}_{D'}}_{\ell-1}, \vec{a}, \vec{0}_{D'}, \ldots, \vec{0}_{D'}}^{w}) \in \mathbb{F}_p^D \mid \vec{a} \in A_i\},$$

$$U_{m+j} = B'_j = \{(\underbrace{\vec{b}, \ldots, \vec{b}}_{w}) \in \mathbb{F}_p^D \mid \vec{b} \in B_j\},$$

以及目标向量

$$\vec{t}'' = (\underbrace{\vec{t}', \ldots, \vec{t}'}_{w}) \in \mathbb{F}_p^D$$

为简化记号，我们对每个 $\ell \in [w]$ 定义

$$A'_\ell = A'_{\ell,1} \cup \cdots \cup A'_{\ell,k},$$

并定义

$$A' = A'_1 \cup \cdots \cup A'_w \ , \ B' = B'_1 \cup \cdots \cup B'_m。$$

这一步重复所需时间线性于输出长度，为

$$O(wD' \cdot k'n) = O(\frac{m^2}{k} nd + m^3 n^{1+1/k})。$$

容易验证最终的时间复杂度为

$$O(d \cdot 2^{O(k)} n^{1+1/k})。$$

性质 1 的证明： 假设存在 $\vec{a}_1 \in A_1, \ldots, \vec{a}_k \in A_k$ 和 $\vec{b}_1 \in B_1, \ldots, \vec{b}_m \in B_m$ 求和为 $\vec{t}'$。对每个 $\ell \in [w], i \in [k]$，我们从 $A'_{\ell,i}$ 中选择

$$(\vec{0}_{(\ell-1)D'}, \vec{a}_i, \vec{0}_{(w-\ell)D'});$$

对每个 $j \in [m]$，从 B'_j 中选择

$$(\vec{b}_j, \ldots, \vec{b}_j)。$$

注意到在第 $\ell \in [w]$ 个长为 D' 的部分，这些所选向量的非零部分求和为：

$$\vec{a}_1 + \cdots + \vec{a}_k + \vec{b}_1 + \cdots + \vec{b}_m = \vec{t}'$$

从而，这些所选向量求和为

$$(\vec{t}', \ldots, \vec{t}') = \vec{t}''。$$

性质 2 的证明： 对任意 $X \subseteq \bigcup_{i\in[k']} U_i$ 和系数 $\lambda : X \to \mathbb{F}_p^+$，假设它们满足

$$\sum_{\vec{x}\in X} \lambda(\vec{x})\vec{x} = \vec{t}''。$$

对每个 $\ell \in [w]$，考虑这些向量的第 ℓ 个部分。注意到在这一部分上，只有属于集合 $A'_{\ell,1}, \ldots, A'_{\ell,k}, B'_1, \ldots, B'_m$ 中的向量以及目标向量 $\vec{t}''$ 的内容是非零的，其余向量均全零，因此不影响求和结果。而存在非全零部分的这些向量的内容和 $A_1, \ldots, A_k, B_1, \ldots, B_m$ 及 $\vec{t}'$ 中的内容完全一致，由引理 4.1可得

$$|X' \cap A'_l| \geq k \qquad |X' \cap B'| \geq m。$$

进一步加上性质 2 的假设。假定任意 $\vec{v}_1 \in V_1, \ldots, \vec{v}_k \in V_k$ 和系数 $\alpha_1, \ldots \alpha_k \in \mathbb{F}_p^+$，都有

$$\alpha_1\vec{v}_1 + \cdots + \alpha_k\vec{v}_k \neq \vec{t}。$$

考虑 $|X' \cap B'|$ 的两种情况。首先，如果

$$|X' \cap B'| < 2(1-\varepsilon)m,$$

则由引理 4.1，对每个 $\ell \in [w]$，都有

$$|X' \cap A'_\ell| \geq 2k。$$

因此，

$$|X'| = |X' \cap A'| + |X' \cap B'| \geq w \cdot (2k) + m = 3m = \frac{3}{2}k'。$$

现在假定

$$|X' \cap B'| \geq 2(1-\varepsilon)m,$$

则可以直接得到

$$|X'| = |X' \cap A'| + |X' \cap B'| \geq w \cdot k + 2(1-\varepsilon)m = (\frac{3}{2} - \varepsilon)k'。$$

由此，在两种情况中都有

$$|X'| \geq (\frac{3}{2} - \varepsilon)k'。$$

■

注　注意到有文献[17]定义过特征为 q 的有限域上的 k-向量和问题（k-VectorSum$_q$），其定义与 k-MLD$_q$ 的唯一差别在于所选向量的系数仅允许是 0 或 1。将本节的归约应用到 k-VectorSum$_q$ 问题上，可以得到 $(q+1)/2$ 的间隔，而非仅仅 k-MLD$_q$ 问题下的 $\frac{3}{2}$。其原因是，如果系数只允许选择 0 或者 1，则对任意解 X，若某个位置 $j \in [m]$ 中有 $|X \cap B_j| > 1$，则必然有

$$|X \cap B_j| = cq + 1$$

（其中 c 为某个正整数）使得 $X \cap B_j$ 中的所有向量直接求和后在该有限域的加法下等于 $\vec{e}_j$。因此，如果某个解 X 只在不超过 ε 比例的位置 $j \in [m]$ 上满足 $|X \cap B_j| = 1$，则必然要求

$$|X \cap B| \geq q(1-\varepsilon)m$$

而非 k-MLD$_q$ 问题中的 $2(1-\varepsilon)m$，由此最终的近似比可以达到

$$(\frac{q+1}{2} - \varepsilon)$$

当 q 被选为超过常数的值时，显然优于 $(\frac{3}{2} - \varepsilon)$ 的近似比。

4.2　近似几个编码与格问题的运行时间下界

本节中，我们应用上一节的归约，给出近似几个编码问题和格问题在 ETH（假设 2.2）下的运行时间下界。为便于阅读，我们将本节主要结论列于表 4.1中，并在后面的小节中证明它们。

4.2.1　k-MLD 和 k-NCP 问题

我们应用已有的归约将 Gap-k-MLD$_p$ 的间隔放大到原来的任意常数倍。该归约来源于 Bhattacharyya 等人的文章[39]。Bhattacharyya 等人的原文只在二元域 $\mathbb{F}_2$ 上证明了下述定理，但可以直观地看出这一定理的证明不依赖于域的特征，因此在所有有限域中均成立。

问题	近似比	时间下界	参数依赖	注记
k-NCP	任意 $\gamma \in (1, \frac{3}{2})$	$f(k)n^{\Omega(\sqrt{k/\log k})}$		任意有限域 $\mathbb{F}_p$
k-NCP	任意 $\gamma > 1$	$f(k)n^{\Omega(k^\epsilon)}$	$\epsilon = \frac{1}{\mathrm{polylog}(\gamma)}$	任意有限域 $\mathbb{F}_p$
k-MDP	任意 $\gamma > 1$	$f(k)n^{\Omega(k^\epsilon)}$	$\epsilon = \frac{1}{p \log \gamma \cdot \mathrm{polylog}(p)}$	任意有限域 $\mathbb{F}_p$
k-CVP	任意 $\gamma > 1$	$f(k)n^{\Omega(k^\epsilon)}$	$\epsilon = \Theta(\frac{1}{\mathrm{polylog}(\gamma)})$	任意 ℓ_p 范数，$p \geq 1$
k-SVP	任意 $\gamma > 1$	$f(k)n^{\Omega(k^\epsilon)}$	$\epsilon = \epsilon(p, \gamma)$	任意 ℓ_p 范数，$p > 1$
k-SVP	任意 $\gamma \in [1, 2)$	$f(k)n^{\Omega(k^\epsilon)}$	$\epsilon = \epsilon(p, \gamma)$	任意 ℓ_p 范数，$p \geq 1$

表 4.1 在 ETH 或 rETH 下近似几个编码和格问题的时间下界

定理 4.2 (文献[39]的定理 4.5 的推广) 对任意素数 p，存在一个多项式时间算法，输入整数 $k_1, k_2, m_1, m_2, n_1, n_2 > 0$，实数 $\gamma_1, \gamma_2 > 1$，以及两个向量集合 $U \subseteq \mathbb{F}_p^{m_1}, V \subseteq \mathbb{F}_p^{m_2}$，其中 $|U| = n_1, |V| = n_2$，两个目标向量 $\vec{t} \in \mathbb{F}_p^{m_1}, \vec{s} \in \mathbb{F}_p^{m_2}$，输出一个向量集合 $W \subseteq \mathbb{F}_p^{m_2+n_1 m_1}$ 和一个目标向量 $\vec{t}' \in \mathbb{F}_p^{m_2+n_1 m_1}$，且满足如下性质：令 $k' = k_2 + k_1 k_2$，$\gamma' \geq \gamma_1 \gamma_2 (1 - \frac{1}{k_1})$，则

- 若 $(U, \vec{t})$ 是 γ_1-Gap-k_1-MLD$_p$ 的一个正实例，且 $(V, \vec{s})$ 是 γ_2-Gap-k_2-MLD$_p$ 的一个正实例，则 $(W, \vec{t}')$ 是 γ'-Gap-k'-MLD$_p$ 的一个正实例；
- 若 $(U, \vec{t})$ 是 γ_1-Gap-k_1-MLD$_p$ 的一个负实例，且 $(V, \vec{s})$ 是 γ_2-Gap-k_2-MLD$_p$ 的一个负实例，则 $(W, \vec{t}')$ 是 γ'-Gap-k'-MLD$_p$ 的一个负实例。

定理 4.2的详细证明可参见文献[39]的第 4.2 节，我们在此只给一个简单的证明概要。我们先给出一个示意图表示其证明思路，以给读者其成立的直观。

定理 4.2的证明概要 考虑形如图 4.3的输出实例 $(W, \vec{t}')$ 的构造。

若 $(U, \vec{t})$ 和 $(V, \vec{s})$ 各自都是正实例，则我们可以在 W 的前 n_1 个向量中选择 $(V, \vec{s})$ 实例中可以线性组合出 $\vec{s}$ 的 k_2 个向量，令他们的系数为该线性组合所需的系数。注意到这样选取后，这些向量线性组合后，在后半段至多存在 k_2 个部分非零，且均为 $-\vec{t}$ 的倍数。对每个有非零 $-\vec{t}$ 的倍数的位置，我们在下方选择 U 中的 k_1 个向量，并相应选取系数使得它们线性组合与该 $-\vec{t}$ 的倍数在 $\mathbb{F}_p$ 下相加为零向量。由此，我们选择的向量和系数组合起来就是所定义的目标向量 $\vec{t}'$，且所选向量的数量不超过 $k_2 + k_1 k_2 = k'$。

若 $(U, \vec{t})$ 和 $(V, \vec{s})$ 各自都是负实例，则在左半部分中，为了线性组合得到 $\vec{s}$，W 的前 n_1 个向量中需要选择至少 $\gamma_2 k_2$ 个向量，以及它们相应的全不为零的系数。注意到这样选取后，这些向量线性组合后，在后半段至少存在 $\gamma_2 k_2$ 个部分非零，且均为 $-\vec{t}$ 的倍数。但是，由于 $(U, \vec{t})$ 也是负实例，对于每个出现 $-\vec{t}$ 的倍数的位置，我们需

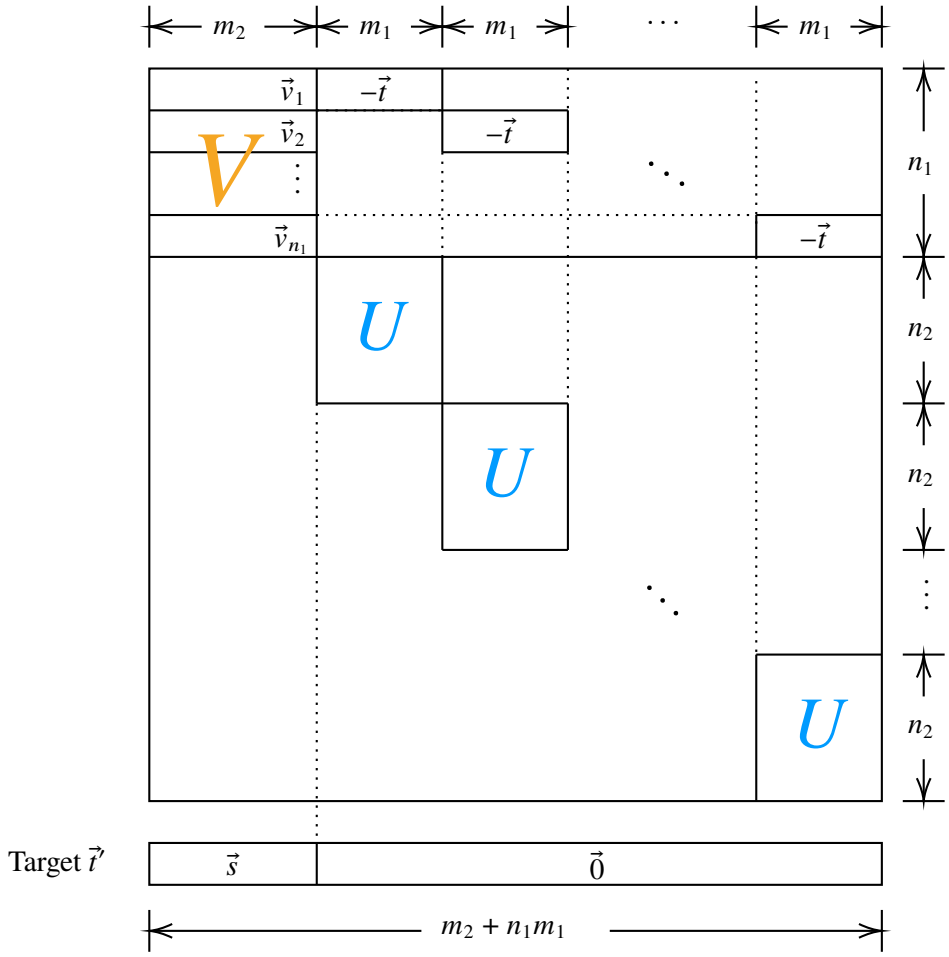

图 4.3　定理 4.2证明的图示。

要在线性组合后将它求和消去，因此在下方必须选择 U 中至少 $\gamma_1 k_1$ 个向量，以及它们的全不为零的系数，才能组合出 $\vec{t}$ 的某个倍数并将这一位置消去成零向量。由此，我们至少需要选择 $\gamma_2 k_2 + \gamma_1 k_1 \gamma_2 k_2 \geq \gamma' k'$ 个向量。

这一归约的其余参数，包括新实例的向量维度和数量，运行的时间复杂性是 = 可以直接验证来得到。 ■

回顾文献[17]中从 3-SAT 到 k-VectorSum 的归约，可以注意到，将他们的归约稍作修改，即可将可靠性部分加强为

- 若 ϕ 不可满足，则对任意 $\vec{v}_1 \in V_1, \cdots, \vec{v}_k \in V_k$ 和任意系数 $\alpha_1, \cdots, \alpha_k \in \mathbb{F}_p^+$，都有

$$\Sigma_{i=1}^{k} \alpha_i \vec{v}_i \neq \vec{t}。$$

修改方法如下：对每个 V_i 中的向量，我们往它的末尾再添加上一个 $(0^{i-1} \circ 1 \circ 0^{k-i})$。

相应地，目标向量也改变为 $\vec{t} = 0^d \circ 1^k$。这一修改的完备性是显然的。对于其可靠性，我们注意到对任意所选的向量 $\vec{v}_1 \in V_1, \cdots, \vec{v}_k \in V_k$ 和任意系数 $\alpha_1, \cdots, \alpha_k \in \mathbb{F}_p$，若

$$\Sigma_{i=1}^{k} \alpha_i \vec{v}_i = \vec{t}$$

则新加入的部分强迫所有系数满足

$$\alpha_1 = \cdots = \alpha_k = 1。$$

经过上述修改，文献[17]给出了下述的 k-MLD 在 ETH 下的运行时间下界：

定理 4.3 假设 ETH 成立，对任意常数素数 p，k-MLD$_p$ 没有运行时间为 $n^{o(k)}$ 的算法。

由于 k-MLD 和 k-NCP 是等价地，我们即可应用上一节中给出的归约，得到这两个问题的近似算法在 ETH（以及 rETH，见假设 2.3）成立时的运行时间下界如下。

定理 4.4 假设随机化 ETH 成立，对任意常数素数 p，常数 $1 < \gamma < \frac{3}{2}$，γ-Gap-k-MLD$_p$ 和 γ-Gap-k-NCP$_p$ 没有运行时间为 $f(k)O(n^{o(\sqrt{k/\log k})})$ 的算法。

证明 对任意小常数 $\varepsilon > 0$，定理 4.1给出了一个从 k-MLD$_p$ 到 $(3/2-\varepsilon)$-Gap-k'-MLD$_p$ 的归约，其中参数增长幅度为

$$k' = k^2 \log k$$

注意到定理 4.1使用了随机码构造（引理 3.3），因此归约是随机化的。应用定理 4.3，我们可以得到

$$f(k)O(n^{o(\sqrt{k/\log k})})$$

的时间下界。 ■

注 注意到定理 4.4中唯一涉及随机性的部分就是对随机码的使用。若使用非随机的编码，如 Reed-Solomon 码（参见定理 3.3），则可以得到 $k' = O(k^3)$，从而在 ETH 下得到 $f(k)O(n^{o(k^{1/3})})$ 的时间下界。

对任意常数 $\gamma > 1$，将上述实例自身作为输入，迭代应用定理 4.2$O(\log\log\gamma)$ 次，我们可以得到 ETH（rETH）下近似 γ-Gap-k-NCP$_p$ 的时间下界。

推论 4.1　假设 ETH 成立，对任意常数素数 p，常数 $\gamma > 1$, ，γ-Gap-k-MLD$_p$ 和 γ-Gap-k-NCP$_p$ 没有运行时间为

$$O_k(n^{o(k^\epsilon)})$$

的算法，其中

$$\epsilon = \frac{1}{\mathsf{polylog}(\gamma)}$$

为一仅与 γ 相关的常数。

4.2.2　最短距离问题

回顾文献[27]中从 Gap-k-NCP 到 Gap-k-MDP 的归约：

定理 4.5　对任意素数幂 $p \geq 2$，存在一个从 $(4p)$-Gap-k-NCP$_p$ 到 $\frac{4p}{4p-1}$-Gap-k'-MDP$_p$ 的随机化多项式时间归约，且满足 $k' = O(k)$。

结合我们前文的归约，得到：

推论 4.2　假设随机 ETH 成立，对任意常数素数 $p \geq 2$ 和实数 $\gamma > 1$，γ-Gap-k-MDP$_p$ 没有运行时间为 $O_k(n^{o(k^\epsilon)})$ 的算法，其中

$$\epsilon = \Theta(\frac{1}{p \log \gamma \cdot \mathsf{polylog}(p)})。$$

证明　首先调用定理 4.1得到 $(\frac{3}{2} - \varepsilon)$-Gap-$k_1$-MLD$_p$ 的实例，其中

$$k_1 = k^2 \log k$$

再迭代地调用定理 4.2的算法 $\Theta(\log \log p)$ 次，将近似比间隔放大到 $4p$，同时参数 k_2 增长为

$$k_2 = k_1^{\mathsf{polylog}(p)} = k^{\mathsf{polylog}(p)}$$

调用定理4.5得到 $\frac{4p}{4p-1}$-Gap-k_3-MDP$_p$ 实例，参数为

$$k_3 = O(k_2) = k^{\mathsf{polylog}(p)}$$

最终，为将 MDP 的近似比放大到任意常数 $\gamma > 1$，只需首先将实例与自身作张量积 $O(\log p)$ 次得到间隔为 2 的实例，参数增长为

$$k_4 = k_3^{O(p)} = k^{p \cdot \mathsf{polylog}(p)}$$

再与自身做 $O(\log\log\gamma)$ 次张量积，参数增长为

$$k_5 = k_4^{O(\log\gamma)} = k^{p\log\gamma\cdot\mathsf{polylog}(p)}$$

定理 4.3给出了 k-MLD$_p$（在 ETH 和 rETH 下）的 $n^{\Omega(k)}$ 运行时间下界，与本节归约结合后可得在 rETH 成立时，γ-Gap-k-MDP$_p$ 没有

$$f(k)n^{o(k^\epsilon)}$$

时间的算法，其中

$$\epsilon = \Theta(\frac{1}{p\log\gamma\cdot\mathsf{polylog}(p)})$$

■

4.2.3 最近向量问题

回顾文献[26]的定理 7.2 中从 Gap-k-NCP 到 Gap-$2k$-CVP 的归约：

定理 4.6 对任意实数 $\gamma, p \geq 1$ 和素数 $q > 2\gamma$，存在一个从 (2γ)-Gap-k-MLD$_q$ 到 γ-Gap-k'-CVP$_p$ 的多项式时间归约，其中参数 $k' = 2k$。

与我们的结果结合，可得近似 CVP 问题的运行时间下界：

推论 4.3 假设 ETH 成立，存在 $c > 0$，对任意实数 $p, \gamma \geq 1$，γ-Gap-k-CVP$_p$ 都没有运行时间为 $O_k(n^{o(k^\epsilon)})$ 的算法，其中 $\epsilon = \Theta(\frac{1}{\gamma^c})$。

证明 首先调用定理 4.1得到一个 $(\frac{3}{2}-\varepsilon)$-Gap-$k_1$-MLD$_p$ 实例，其中

$$k_1 = k^2\log k$$

再迭代地调用定理 4.2$\Theta(\log\log\gamma)$ 次，将近似比间隔放大到 2γ，同时参数 k_2 也增长到

$$k_1^{\mathsf{polylog}(\gamma)} = k^{\mathsf{polylog}(\gamma)}$$

再应用定理 4.6，得到 γ-Gap-$2k_2$-CVP$_p$ 的实例。结合定理 4.3可得 ETH 下问题 γ-Gap-$2k_2$-CVP$_p$ 的 $f(k)n^{\Omega(k^\epsilon)}$ 的运行时间下界，其中

$$\epsilon = \Theta(\frac{1}{(\log\gamma)^c})$$

且 c 为与 γ 和 p 独立的一个常数。 ■

4.2.4　最短向量问题

文献[27]中给出了几个到 Gap-k-SVP 问题的归约。将我们前文的结果与他们的归约结合，可以得到几个 Gap-k-SVP 问题的时间下界。本节第一部分应用 Gap-k-CVP 到 Gap-k-SVP 的归约，给出后者在任意 $p \geq 1$ 的 ℓ_p 范数下近似比达到某个特定常数的运行时间下界；第二部分应用 Gap-k-NCP 到 Gap-k-SVP 的归约，给出后者在任意 $p > 1$ 的 ℓ_p 范数下任意常数近似比的运行时间下界。

4.2.4.1　从 Gap-k-CVP 出发的归约

定理 4.7 (文献[27]中定理 4.1 及定理 4.3)　对任意实数 $p \geq 1$ 和 $\gamma' \in [1,2)$，存在实数 $\gamma \geq 1$ ① 以及从 γ-Gap-k-CVP$_p$ 到 γ'-Gap-k'-SVP$_p$ 的随机化多项式时间归约，其中 $k' \leq \gamma k$。

推论 4.4　假设 rETH 成立，对任意实数 $p \geq 1$ 和 $\gamma \in [1,2)$，γ-Gap-k-SVP$_p$ 没有运行时间为 $O_k(n^{o(k^\epsilon)})$ 的算法，其中 $0 < \epsilon < 1$ 是依赖于 p 和 γ 的常数。

证明　推论 4.3给出了 ETH (rETH) 下 γ_0-Gap-k-CVP$_p$ 的算法时间下界为 $f(k)n^{\Omega(k^{\epsilon_0})}$，其中

$$\epsilon_0 = \Theta(\frac{1}{\gamma_0^c})$$

且 $c > 0$ 为一个全局的常数。令定理 4.7中的 $\gamma = \gamma_0$，注意到 γ_0 也是依赖于 p 和 γ 的常数，注意到该定理的归约是随机化的，这样就得到了 γ-Gap-k'-SVP$_p$ 的随机化归约，其中

$$k' \leq \gamma k = O(k)$$

这就给出了问题 γ-Gap-k-SVP$_p$ 的算法在 rETH 运行时间下界

$$f(k)n^{o(k^\epsilon)}$$

其中常数 $\epsilon > 0$ 仅依赖于 p 和 γ。 ■

4.2.4.2　从 Gap-k-NCP 出发的归约

定理 4.8 (文献[27]中定理 5.1 及定理 5.2)　存在实数常数 $\mu \geq 1$ 使得，对任意实数 $p > 1$ 和 $\gamma' \geq 1$，存在从 μ-Gap-k-NCP$_2$ 到 γ'-Gap-k'-SVP$_p$ 的多项式时间随机化归约，其中

① $\gamma = (\max\left(12/\varepsilon, \frac{1}{(1+\varepsilon/2)^{1/p}-1}\right))^p$，其中 $\varepsilon = (\gamma')^{-1} - 1/2 > 0$。

$k' = O(k^c)$，$c > 1$ 是仅依赖于 p 和 γ' 的常数[①]。

推论 4.5 假设 rETH 成立，对任意实数 $p > 1$ 和 $\gamma \geq 1$，γ-Gap-k-SVP$_p$ 没有运行时间为 $O_k(n^{o(k^\epsilon)})$ 的算法，其中 $0 < \epsilon < 1$ 是依赖于 p 和 γ 的常数。

证明 为满足定理 4.8中的参数要求，我们需要得到一个 μ-Gap-k_1-NCP$_2$ 的实例，其中 μ 是定理 4.8所需的间隔大小。应用定理 4.1和定理 4.2，前述实例可以由 k-MLD$_2$ 归约得到，参数增长为

$$k_1 = O(k^{\epsilon_0})$$

其中 ϵ_0 为一仅依赖于 μ 的常数。再应用定理 4.8，对任意 $\gamma \geq 1$，可以得到 γ-Gap-k_2-SVP$_p$ 的实例参数增长为

$$k_2 = O(k_1^c) = O(k^{1/\epsilon})$$

其中

$$\epsilon = \Theta(\frac{1}{c})$$

为仅依赖于 p 和 γ（以及 μ，但在此省略，因为它是全局的常数，与 p 和 γ 无关）的常数。注意到定理 4.8的归约是随机化的，可得在 rETH 下，问题 γ-Gap-k-SVP$_p$ 没有时间为

$$f(k)n^{o(k^\epsilon)}$$

的算法。 ■

4.3 本章小结

本章使用第 3章中发展的技术工具，讨论了几个编码问题和格问题的近似困难性。在第 4.1节，我们给出了核心的归约步骤，即使用高碰撞数的编码作为工具，将无间隔的 k-MLD 实例转化为有常数间隔的 Gap-k-MLD 实例。在之后的第4.2节中，我们将这一结果插入到已有的归约中，显著地优化了常数比近似与之相关的下面几个编码与格问题在指数时间假设下的运行时间下界：

- k-MLD 和 k-NCP 问题（第4.2.1节）；

① 此处关于参数增长需要考虑两点，其一是 SVP 的 “Haviv-Regev 张量化[27,40]” 步骤会导致 $k' \leq (\mu k)^{O(1)}$；其二是为达到目标间隔 γ'，NCP 的间隔 μ 需要满足

$$\frac{\mu}{2^p + 1 + \alpha\mu} > \gamma'$$

（其中 $1/2 + 2^{-p} < \alpha < 1$），将 μ 放大到此值也会导致参数有多项式大小的增长。

- k-MLD 问题（第4.2.2节）;
- k-CVP 问题（第4.2.3节）;
- k-SVP 问题（第4.2.4节）。

第 5 章 约束满足问题

本章讨论参数化约束满足问题的多赋值大小的近似困难性。本章的主要结果是证明下述定理 5.1，并讨论这一结果可能的推广与应用。

定理 5.1 对任意常数 $r \geq 1$，r-AvgList-2CSP 是 W[1]-困难的，且当 r-AvgList-2CSP 实例限制为矩形约束时依然是 W[1]-困难的。

5.1 补充预备知识

本节介绍本章内容所需的一些额外假设和注记。

定义 5.1 (矩形约束) 对 2CSP 实例 $\Pi = (X, \Sigma, \Phi)$，若每个约束 $\varphi_j = (x_{j_1}x_{j_2}, C_j) \in \Phi$ 都存在一个集合 Q_j 和映射 $\pi_j, \rho_j : \Sigma \to Q_j$ 使得 $(a, b) \in C_j$ 当且仅当 $\pi_j(a) = \rho_j(b)$，则称 Π 有矩形约束，并将 Q_j 称作约束 φ_j 的基集。

我们对“矩形”这一命名稍作解释。一个集合 $S \subseteq \Sigma \times \Sigma$ 被称作一个组合学的矩形，当且仅当存在 $A, B \subseteq \Sigma$ 使得 $S = A \times B$。容易看出集合 S 满足定义 5.1中的矩形当且仅当 S 是一族 $\Sigma \times \Sigma$ 上互不相交的组合学的矩形的并。

5.2 从 r-List-2CSP 到 r-AvgList-2CSP 的归约

本节给出一个从 r-List-2CSP 到 r-AvgList-2CSP 的 FPT 归约。归约分为两步进行，第一步是将 r-List-2CSP 的实例转化为一个类似于 r-AvgList-2CSP 的实例，区别在于新实例的变量被划分为两部分，且在输入为 r-List-2CSP 的负实例的情况下，输出的新实例的可满足多赋值必然在某一部分变量上产生间隔。这也是整个归约过程的技术难点所在。第二步就是将这一新实例转化为 r-AvgList-2CSP 的实例，其中近似比有所折损，但仍在常数范围内。

为了叙述归约的第一步，我们引入如下定义：

定义 5.2 令 $\Pi = (X, \Sigma, \Phi)$ 为一个二分的 2CSP 实例，即变量集 $X = X_1 \dot\cup X_2$，且每个约束 $\varphi = (x_1x_2, C) \in \Phi$ 均满足 $x_1 \in X_1$ 和 $x_2 \in X_2$。对常数 $r_1, r_2 \geq 1$，我们称 Π 的一个多赋值 $\hat\sigma : X \to 2^\Sigma$ 是(r_1, r_2)-平均多赋值，如果

$$\frac{\sum_{x \in X_1} |\hat\sigma(x)|}{|X_1|} \leq r_1 \quad \text{且} \quad \frac{\sum_{x \in X_2} |\hat\sigma(x)|}{|X_2|} \leq r_2$$

即 $\hat{\sigma}$ 限制在 X_1 上的总大小不超过 $r_1|X_1|$，且限制在 X_2 上的总大小不超过 $r_2|X_2|$（参见定义 2.8）若存在一个 (r_1, r_2)-平均的多赋值满足 Π，则称 Π 是(r_1, r_2)-平均可满足的。

引理 5.1　存在算法 $\mathcal{A}$，其输入为一个 2CSP 实例 $\Pi_0 = (X_0, \Sigma_0, \Phi_0)$，以及参数 $\varepsilon > 0$，$r \geq 1$，输出一个二分的 2CSP 实例 $\Pi = (X_1 \dot{\cup} X_2, \Sigma, \Phi)$，该输出实例满足：

完备性：　若 Π_0 是可满足的，则 Π 也是可满足的；

可靠性：　对任意常数 $r \geq 1$，若 Π_0 不是 $2r$-列表可满足的，则 Π 不是 (r_1, r_2)-平均列表可满足的，其中 $r_1, r_2 \in \mathbb{N}$ 是满足

$$r_1 + r_2 \leq 2(1 - \varepsilon)r.$$

的常数。

矩形约束：　Φ 有矩形约束。

且存在可计算函数 f，$\mathcal{A}$ 的运行时间不超过

$$f(|X_0| + |\Phi_0| + 1/\varepsilon + r)|\Sigma_0|^{O(1)} \tag{5.1}$$

且 Π 的变量数 $|X_1| + |X_2|$ 和约束数 $|\Phi|$ 也不超过 $f(|X_0| + |\Phi_0| + 1/\varepsilon + r)$。

证明　对输入的 2CSP 实例 $\Pi_0 = (X_0, \Sigma_0, \Phi_0)$，我们令

$$k = |X_0| \quad 且 \quad k' = |\Phi_0|$$

我们记变量集 X_0 和约束集 Φ_0 的内容为

$$X_0 = \{x_1, \ldots, x_k\} \quad 和 \quad \Phi_0 = \left\{\varphi_1, \ldots, \varphi_{k'}\right\}$$

令 $C : \mathbb{F}_p^k \to \mathbb{F}_p^{k''}$ 为满足下面条件的 Reed-Solomon 编码：

$$2|\Sigma_0|^{1/k} > p \geq |\Sigma_0|^{1/k} \quad 且 \quad k'' = \left\lfloor \frac{8(1-\varepsilon)^2 r^2}{\varepsilon} k(k')^2 \right\rfloor + 1$$

显然 $|\Sigma_0| \leq p^k$，因此不失一般性我们可以假设

$$\Sigma_0 \subseteq \mathbb{F}_p^k$$

我们可以只考虑 k'' 满足下述条件的情况

$$k'' \leq p \left(= \left|\mathbb{F}_p\right| \right) < 2|\Sigma_0|^{1/k},$$

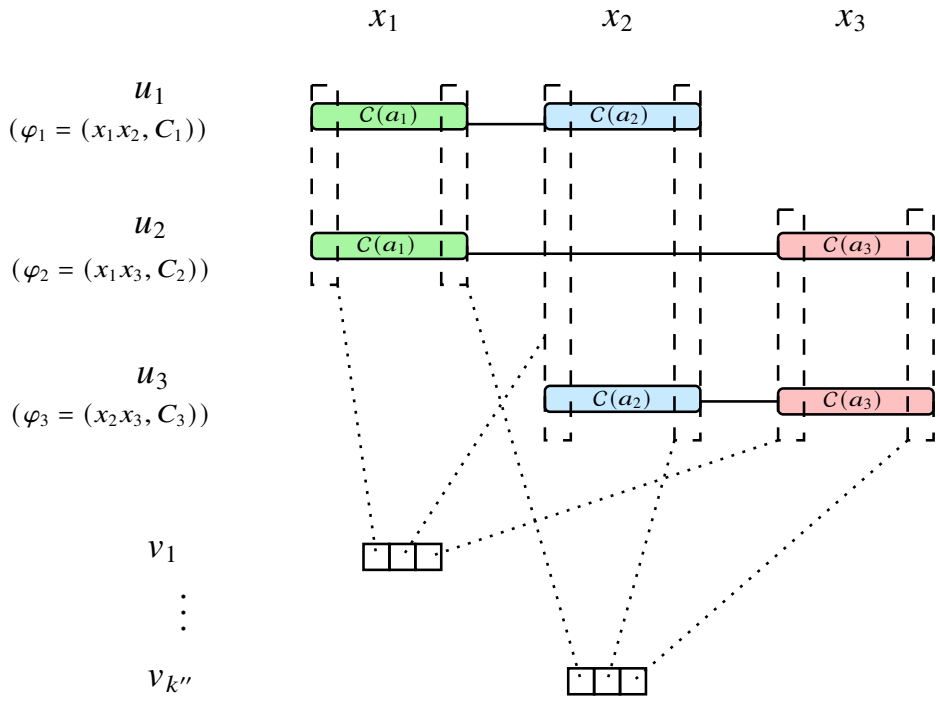

图 5.1 输入实例为 $\Pi_0 = (X_0, \Sigma_0, \Phi_0)$，且 $|X_0| = |\Phi_0| = 3$ 时输出实例的构造图示。

即 Σ_0 远远大于 k 和 k' 的情况[①]。我们调用定理 3.3，设定其中的参数为 $\Sigma \leftarrow \mathbb{F}_p$，$k \leftarrow k$，$m \leftarrow k''$，$\varepsilon \leftarrow \varepsilon$，得到满足如下条件的编码 C：

$$\mathrm{Col}_{\varepsilon}(C(\mathbb{F}_p^k)) \geq \sqrt{\frac{2\varepsilon k''}{k}} > 4(1-\varepsilon)rk', \tag{5.2}$$

其中 k'' 的选择保证了第二个不等式成立。

完成参数设定和编码选取后，算法 $\mathcal{A}$ 构造二分的 2CSP 实例 $\Pi = (X, \Sigma, \Phi)$。为便于读者直观理解，我们先给出一个简单的图示，见图 5.1。该图直观地展示了在输入一个 Π_0 满足 $|X_0| = |\Phi_0| = 3$ 时的构造。

我们的具体构造如下。

变量： $X = X_1 \dot{\cup} X_2$，其中

$$X_1 = \{u_1, \ldots, u_{k'}\} \quad , \quad X_2 = \{v_1, \ldots, v_{k''}\}.$$

字母表： $\Sigma = \bigcup_{u \in X_1} \Sigma_u \cup \bigcup_{v \in X_2} \Sigma_v$，其中：

① 否则，输入的实例 Π_0 可以在公式 (5.1)所示的时间内被解决，从而我们可以根据 Π_0 的列表可满足性直接输出某个显然的正或负实例 Π。

- 对每个 $j \in [k']$，变量 $u_j \in X_1$ 的字母表为

$$\Sigma_{u_j} = \left\{ \big(C(a_1), C(a_2)\big) \;\middle|\; \varphi_j = (x_{j_1}x_{j_2}, C_j) \text{ 且 } (a_1, a_2) \in C_j \right\} \subseteq \big(C(\mathbb{F}_p^k)\big)^2 \subseteq \big(\mathbb{F}_p^{k''}\big)^2 \tag{5.3}$$

  即 Σ_{u_j} 包含了 φ_j 的所有可满足的部分赋值在编码

$$C : \mathbb{F}_p^k \to \mathbb{F}_p^{k''}$$

  下的码字，这些码字的形式为 $\mathbb{F}_p^{k''}$ 中的向量对（因为 $\Sigma_0 \subseteq \mathbb{F}_p^k$）。
- 对每个 $\ell \in [k'']$，令 $\Sigma_{v_\ell} = \mathbb{F}_p^k$。由于 $p < 2|\Sigma_0|^{1/k}$，我们有

$$|\Sigma_{v_\ell}| \leq 2^k |\Sigma_0|$$

约束： 对每个 $j \in [k']$，约束 $\varphi_j = (x_{j_1}x_{j_2}, C_j)$，以及每个 $\ell \in [k'']$，在 $u_j \in X_1$ 和 $v_\ell \in X_2$ 之间有一个约束，该约束检查 u_j 的赋值 $(w_1, w_2) \in \big(\mathbb{F}_p^{k''}\big)^2$ 和 v_ℓ 的赋值 $s \in \mathbb{F}_q^\ell$ 是否满足

$$w_1[\ell] = s[j_1] \quad \text{且} \quad w_2[\ell] = s[j_2] \tag{5.4}$$

容易看出公式 (5.4)表明的约束是矩形约束[①]。且 Π 中的约束数量为

$$k'k'' = k' \left\lfloor \frac{2(1-\varepsilon)^2 r^2}{\varepsilon} k (k')^2 \right\rfloor + k'.$$

完备性的证明： 上述归约的完备性是容易得出的，我们简述如下。

假设输入的 2CSP 实例 Π_0 是可满足的，令 $\sigma_0 : X_0 \to \Sigma_0$ 为一个它的可满足赋值，我们构造 Π 的可满足赋值 $\sigma : X \to \Sigma$ 如下。

对每个 $u_j \in X_1$，注意到其对应于 Π_0 中的约束

$$\varphi_j = (x_{j_1}x_{j_2}, C_j)$$

令其赋值为

$$\sigma(u_j) = (C(\sigma_0(x_{j_1})), C(\sigma_0(x_{j_2})))$$

由于 φ_j 被 σ_0 满足，$(\sigma_0(x_{j_1}), \sigma_0(x_{j_2}))$ 自然满足了 φ_j，从而 σ 给 u_j 的这一赋值是合法的。

① 可以令 $\pi(u_j) = \pi(w_1, w_2) = (w_1[\ell], w_2[\ell])$，$\rho(v_\ell) = (v_\ell[j_1], v_\ell[j_2])$，则公式 (5.4)和定义 5.1中的 $\pi(u_j) = \rho(v_\ell)$ 完全一致。

在确定了 X_1 的赋值后，我们确定 X_2 的赋值如下。对每个 $v_\ell \in X_2$，赋值为

$$\sigma(v_\ell) = (C(\sigma_0(x_1))[\ell], C(\sigma_0(x_2))[\ell], \cdots, C(\sigma_0(x_k))[\ell])$$

由此，对 Π 中每个约束，即每一对 (u_j, v_ℓ)，σ 对 u_j 的赋值中两部分各自的第 ℓ 个位置的值是

$$C(\sigma_0(x_{j_1}))[\ell] \quad 和 \quad C(\sigma_0(x_{j_2}))[\ell]$$

恰与 v_ℓ 赋值的第 j_1 和 j_2 个位置内容一致，因此 u_j, v_ℓ 间的约束被赋值 σ 满足。

可靠性的证明： 假设输入的 2CSP 实例 Π_0 不是 $2r$-列表可满足的。

我们希望证明对任意的 $r_1, r_2 \in \mathbb{N}$，如果存在一个可满足的 (r_1, r_2)-平均的多赋值 $\hat{\sigma}$，则

$$r_1 + r_2 > 2(1-\varepsilon)r \tag{5.5}$$

令 $\hat{\sigma} : X \to 2^{\Sigma}$ 为 Π 的任意一个可满足的多赋值。我们记集合 $\mathsf{Word}_{\hat{\sigma}}$ 为

$$\mathsf{Word}_{\hat{\sigma}} = \bigcup_{u_j \in X_1} \bigcup_{(w_1, w_2) \in \hat{\sigma}(u_j)} \{w_1, w_2\} \subseteq \mathbb{F}_p^{k''} \tag{5.6}$$

即 $\mathsf{Word}_{\hat{\sigma}}$ 是 $\mathbb{F}_p^{k''}$ 中的码字里被 $\hat{\sigma}$ 指派过给 X_1 中的变量的那一部分。

我们给出如下的断言：

断言 1. 令 $\ell \in [k'']$ 是满足 $|\hat{\sigma}(v_\ell)| \le 2r$ 的位置，则 $\mathsf{Word}_{\hat{\sigma}}$ 在位置 ℓ 产生碰撞。

断言 1的证明 考虑任意约束 $\varphi_j = (x_{j_1} x_{j_2}, C_j) \in \Phi_0$（其中 $j \in [k']$，$j_1, j_2 \in [k]$）。因为 $\hat{\sigma}$ 是 Π 的一个可满足的多赋值，所以存在

$$(w_1, w_2) \in \hat{\sigma}(u_j) \subseteq \Sigma_{u_j} \subseteq \left(\mathbb{F}_p^{k''}\right)^2 \quad 和 \quad s \in \hat{\sigma}(v_\ell) \subseteq \mathbb{F}_p^k$$

使得当 u_j 取值 (w_1, w_2)，v_ℓ 取值 s 时，u_j 与 v_ℓ 间的约束被满足。由于 $(w_1, w_2) \in \Sigma_{u_j}$，回顾公式 (5.3)可知，存在 $a_1, a_2 \in \Sigma_0$ 使得 $w_1 = C(a_1)$，$w_2 = C(a_2)$，且

$$x_{j_1} 取值 a_1 且 x_{j_2} 取值 a_2 时约束 \varphi_j 被满足。 \tag{5.7}$$

我们称原字母表中的 a_1 在 s 下对变量 x_{j_1} 是$(\hat{\sigma}, \varphi_j)$-适宜的，$a_2$ 在 s 下对变量 x_{j_2} 是$(\hat{\sigma}, \varphi_j)$-适宜的。进一步地，由 Π 中约束的定义，即公式(5.4)，

$$C(a_1)[\ell] = s[j_1] \quad 且 \quad C(a_2)[\ell] = s[j_2] \tag{5.8}$$

我们根据 $\hat{\sigma}$ 构造原输入实例 $\Pi_0 = (X_0, \Sigma_0, \Phi_0)$ 的一个多赋值 $\hat{\sigma}_0 : X_0 \to 2^{\Sigma_0}$ 如下。对变量 $x \in X_0$，令

$$\hat{\sigma}_0(x) = \bigcup_{s \in \hat{\sigma}(v_\ell)} \left\{ a \in \Sigma_0 \,\middle|\, \text{存在 } j \in [k']，\ a \text{ 在 } s \text{ 下对变量 } x \text{ 是 } (\hat{\sigma}, \varphi_j)\text{-适宜的} \right\} \quad (5.9)$$

（注意此处我们讨论的是断言中给出的一个特殊的 $\ell \in [k'']$ 以及相应的集合 $\hat{\sigma}(v_\ell)$。）由于每个变量 x 至少会出现在一个约束 $\varphi_j \in \Phi_0$ 中，由性质 (5.7)我们容易看出，$\hat{\sigma}_0$ 是 Π_0 的一个可满足的多赋值。

由于 Π_0 不是 $2r$-列表可满足的，必然存在某个变量 $x_{i^*} \in X_0$（相应地，某个位置 $i^* \in [k]$）使得

$$\left|\hat{\sigma}_0(x_{i^*})\right| \geq 2r + 1$$

注意到由断言 1的假设，

$$|\hat{\sigma}(v_\ell)| \leq 2r,$$

因此由公式 (5.9)，存在某个 $s \in \hat{\sigma}(v_\ell)$ 使得

$$\left| \left\{ a \in \Sigma_0 \,\middle|\, \text{存在 } j \in [k']，\ a \text{ 在 } s \text{ 下对变量 } x_{i^*} \text{ 是 } (\hat{\sigma}, \varphi_j)\text{-适宜的} \right\} \right| \geq 2$$

由此可得，存在 $a_1, a_2 \in \Sigma_0$，$a_1 \neq a_2$，以及 $j_1, j_2 \in [k']$ 使得

- a_1 在 $s \in \hat{\sigma}(v_\ell)$ 下对变量 x_{i^*} 是 $(\hat{\sigma}, \varphi_{j_1})$-适宜的；
- a_2 在 $s \in \hat{\sigma}(v_\ell)$ 下对变量 x_{i^*} 是 $(\hat{\sigma}, \varphi_{j_2})$-适宜的。

再由公式 (5.8)可知

$$C(a_1)[\ell] = s[i^*] = C(a_2)[\ell]$$

也就是说，$C(a_1)$ 和 $C(a_2)$ 在位置 ℓ 上碰撞。显然 $C(a_1), C(a_2) \in \mathsf{Word}_{\hat{\sigma}}$，因此断言 1得证。 □

我们记

$$r_1 = \frac{\sum_{x \in X_1} |\hat{\sigma}(x)|}{|X_1|} = \frac{\sum_{j \in [k']} \left|\hat{\sigma}(u_j)\right|}{k'} \quad \text{以及} \quad r_2 = \frac{\sum_{x \in X_2} |\hat{\sigma}(x)|}{|X_2|} = \frac{\sum_{\ell \in [k'']} \left|\hat{\sigma}(v_\ell)\right|}{k''}$$

现在考虑以下两种情况：

1. 有超过 ε 比例的 X_2 中的位置 $\ell \in [k'']$，这些位置上的变量 x_ℓ 赋值都很少，即 $|\hat{\sigma}(v_\ell)| \leq 2r$, 则断言 1表明集合 $\mathsf{Word}_{\hat{\sigma}}$ 在超过 ε 比例的位置 $\ell \in [k'']$ 上产生碰撞。应用编码的性质，即公式 (5.2)，有

$$\mathrm{Col}_\varepsilon(C(\mathbb{F}_p^k)) \geq \sqrt{\frac{2\varepsilon k''}{k}} > 4(1-\varepsilon) r k'$$

由此，

$$\left|\mathsf{Word}_{\hat{\sigma}}\right| \geq \mathrm{Col}_{\varepsilon}(C(\mathbb{F}_p^k)) > 4(1-\varepsilon)rk'$$

由 $\mathsf{Word}_{\hat{\sigma}}$ 的定理（公式 (5.6)），我们可以推论得到

$$\begin{aligned}\left|\mathsf{Word}_{\hat{\sigma}}\right| &= \left|\bigcup_{u_j \in X_1} \bigcup_{(w_1,w_2)\in\hat{\sigma}(u_i)} \{w_1, w_2\}\right| \\ &\leq \sum_{u_j \in X_1} \left|\bigcup_{(w_1,w_2)\in\hat{\sigma}(u_i)} \{w_1, w_2\}\right| \leq \sum_{u_j \in X_1} 2\left|\hat{\sigma}(u_i)\right|\end{aligned}$$

由此可得

$$r_1 = \frac{\sum_{j\in[k']} \left|\hat{\sigma}(u_j)\right|}{k'} > \frac{4(1-\varepsilon)rk'}{2k'} = 2(1-\varepsilon)r$$

2. X_2 中赋值少的变量（$|\hat{\sigma}(v_\ell)| \leq 2r$）不超过 ε 比例，换言之，有超过 $(1-\varepsilon)$ 比例的 X_2 中的位置 $\ell \in [k'']$，这些位置上的变量 x_ℓ 赋值满足 $|\hat{\sigma}(v_\ell)| \geq 2r+1$。则

$$r_2 = \frac{\sum_{\ell\in[k'']} \left|\hat{\sigma}(v_\ell)\right|}{k''} \geq \frac{(1-\varepsilon)k''(2r+1)+\varepsilon k''}{k''} > 2(1-\varepsilon)r.$$

由此，两种情况下等式 (5.5)都成立，证毕。 ■

将引理 5.1给出的具有非平衡的 (r_1, r_2)-间隔的实例适中不同部分分别复制若干次，可以得到具有平衡间隔的实例，即一个不可以 r-平均列表满足的实例。

引理 5.2 对任意二分的 2CSP 实例 $\Pi = (X_1 \dot\cup X_2, \Sigma, \Phi)$ 以及 $r > 1$，我们可以在多项式时间内输出一个 2CSP 实例 $\Pi' = (X', \Sigma', \Phi')$，其中

$$|X| = 2|X_1||X_2|$$

且满足

完备性： 如果 Π 可满足，则 Π' 也可满足；

可靠性： 令 $r \geq 1$。若对任意 $r_1, r_2 > 1$，$r_1 + r_2 \leq 2r$，Π 都不是 (r_1, r_2)-平均列表可满足的，则 Π' 不是 r-平均列表可满足的。等价来说，若 Π' 是 r-列表可满足的，则存在 $r_1, r_2 \in \mathbb{N}$，$r_1 + r_2 \leq 2r$，且 Π 是 (r_1, r_2)-平均列表可满足的。

更进一步地，如果 Π 有矩形约束，则 Π' 也有矩形约束。

证明 令

$$k_1 = |X_1| \quad 以及 \quad k_2 = |X_2|$$

待输出的实例 $\Pi' = (X', \Sigma', \Phi')$ 构造如下。

变量： X' 包含 k_2 份 X_1 的拷贝，以及 k_1 份 X_2 的拷贝。正式地说，$X' = X'_1 \dot\cup X'_2$，其中

$$X'_1 = \left\{x^{(i)} \,\middle|\, x \in X_1 \text{ 且 } i \in [k_2]\right\} \quad \text{以及} \quad X'_2 = \left\{x^{(i)} \,\middle|\, x \in X_2 \text{ 且 } i \in [k_1]\right\}.$$

注意到 $|X'_1| = |X'_2| = k_1 k_2$，因此 Π' 包含 $2k_1k_2$ 个变量。

字母表： $\Sigma' = \bigcup_{x \in X'} \Sigma'_x$，其中：

- 对每个 $x \in X_1$ 和 $i \in [k_2]$，令 $\Sigma'_{x^{(i)}} = \Sigma_x$。其中 $\Sigma_x \subseteq \Sigma$ 是变量 x 在输入的 2CSP 实例 Π 中的字母表。
- 相应地，对每个 $x \in X_2$ 和 $i \in [k_1]$，令 $\Sigma'_{x^{(i)}} = \Sigma_x$。

约束： 对每个输入实例中的约束

$$\varphi = (x_1 x_2, C) \in \Phi$$

其中 $x_1 \in X_1$，$x_2 \in X_2$ 以及每个 $i_1 \in [k_2]$ 和 $i_2 \in [k_1]$，新实例中有约束

$$\varphi^{i_1,i_2} = \left(x_1^{(i_1)} x_2^{(i_2)}, C\right) \in \Phi'.$$

即 φ^{i_1,i_2} 是 φ 的一份拷贝，但将其中的 x_1 替换为它的第 i_1 份拷贝 $x_1^{(i_1)}$，x_2 替换为它的第 i_2 份拷贝 $x_2^{(i_2)}$。由此我们立即得到：若 Π 有矩形约束，则 Π' 也有矩形约束。

完备性的证明： 假设 Π 可满足，我们证明 Π' 也是可满足的。

令 $\sigma : X_1 \dot\cup X_2 \to \Sigma$ 为 Π 的一个可满足赋值，我们构造 Π' 的可满足赋值 $\sigma' : X' \to \Sigma'$ 如下。

对每个 $x_1 \in X_1$，它在 X'_1 中有 k_2 份拷贝

$$\{x_1^{(1)}, x_1^{(2)}, \cdots, x_1^{(k_2)}\}$$

σ' 给它们均赋值为 $\sigma(x_1)$。同理，对每个 $x_2 \in X_2$，它在 X'_2 中有 k_1 份拷贝

$$\{x_2^{(1)}, x_2^{(2)}, \cdots, x_2^{(k_1)}\}$$

σ' 给它们均赋值为 $\sigma(x_2)$。由于 x 的每份拷贝的字母表均与 x 的字母表一致，上述赋值 σ' 是合法的。

对于每个 Φ' 中的约束 φ^{i_1,i_2}，它与输入实例中的某个约束 $\varphi = (x_1 x_2, C)$ 满足的集合 C 一致，区别只在于 φ^{i_1,i_2} 的变量分别换为 φ 变量的第 i_1 份和第 i_2 份拷贝。由上

述赋值 σ' 的构造，所有拷贝的赋值均和原实例赋值 σ 一致。由于 φ 被 σ 满足，φ^{i_1,i_2} 也被 σ' 满足。

可靠性的证明： 令 $\hat{\sigma}' : X' \to 2^{\Sigma'}$ 为 Π' 的一个可满足的 r-平均多赋值。也就是说，

$$r = \frac{\sum_{x\in X'} \left|\hat{\sigma}'(x)\right|}{|X'|} = \frac{\sum_{x\in X'_1} \left|\hat{\sigma}'(x)\right| + \sum_{x\in X'_2} \left|\hat{\sigma}'(x)\right|}{|X'_1| + |X'_2|} = \frac{\sum_{x\in X'_1} \left|\hat{\sigma}'(x)\right| + \sum_{x\in X'_2} \left|\hat{\sigma}'(x)\right|}{2k_1k_2}$$

我们令

$$r_1 = \frac{\sum_{x\in X'_1} \left|\hat{\sigma}'(x)\right|}{|X'_1|} = \frac{\sum_{x\in X'_1} \left|\hat{\sigma}'(x)\right|}{k_1k_2} \quad \text{以及} \quad r_2 = \frac{\sum_{x\in X'_2} \left|\hat{\sigma}'(x)\right|}{|X'_2|} = \frac{\sum_{x\in X'_2} \left|\hat{\sigma}'(x)\right|}{k_1k_2} \tag{5.10}$$

则有

$$r_1 + r_2 = \frac{\sum_{x\in X'_1} \left|\hat{\sigma}'(x)\right| + \sum_{x\in X'_2} \left|\hat{\sigma}'(x)\right|}{k_1k_2} = 2r.$$

注意到

$$X'_1 = \dot{\bigcup}_{i\in[k_2]} \left\{x^{(i)} \mid x \in X_1\right\} \tag{5.11}$$

由此，

$$\begin{aligned} r_1k_1k_2 &= r_1|X'_1| && \text{(根据 } |X'_1| = k_1k_2\text{)} \\ &= \sum_{x\in X'_1} \left|\hat{\sigma}'(x)\right| && \text{(根据 (5.10))} \\ &= \sum_{i\in[k_2]} \sum_{x\in X_1} \left|\hat{\sigma}'(x^{(i)})\right|. && \text{(根据 (5.11))} \end{aligned}$$

因此，存在某个 $i_1 \in [k_2]$ 使得

$$\sum_{x\in X_1} \left|\hat{\sigma}'(x^{(i_1)})\right| \le r_1k_1, \quad \text{也就是说} \ \frac{\sum_{x\in X_1} \left|\hat{\sigma}'(x^{(i_1)})\right|}{|X_1|} \le r_1$$

（因为 $|X_1| = k_1$）。对另一半变量 X_2 使用同样的论证过程，可以得到存在某个 $i_2 \in [k_1]$ 使得

$$\frac{\sum_{x\in X_2} \left|\hat{\sigma}'(x^{(i_2)})\right|}{|X_2|} \le r_2$$

最后，我们定义原输入实例 Π 的一个多赋值 $\hat{\sigma}$ 如下：

$$\hat{\sigma}(x) = \begin{cases} \hat{\sigma}'(x^{(i_1)}) & \text{如果 } x \in X_1 \\ \hat{\sigma}'(x^{(i_2)}) & \text{如果 } x \in X_2 \end{cases}$$

由上述的讨论，$\hat{\sigma}$ 是 (r_1, r_2)-平均的多赋值，且满足 Π，因为 $\hat{\sigma}'$ 已经满足 Π'。 ■

将前述的各个部分拼起来，我们得到一个从 r-List-2CSP 到 r-AvgList-2CSP 的 FPT 归约如下。

定理 5.2　存在算法 $\mathcal{A}$，对任意常数 $r \geq 1$，算法 $\mathcal{A}$ 的输入为一个 r-List-2CSP 实例 $\Pi_0 = (X_0, \Sigma_0, \Phi_0)$，以及参数 $\varepsilon > 0$，输出一个 r'-AvgList-2CSP 实例 $\Pi = (X, \Sigma, \Phi)$，其中 $r' \geq \frac{1-\varepsilon}{2} r$，$\Pi$ 有矩形约束。

存在可计算函数 f，$\mathcal{A}$ 的运行时间不超过

$$f(|X_0| + |\Phi_0| + 1/\varepsilon + r)|\Sigma_0|^{O(1)} \tag{5.12}$$

且 Π 的变量数 $|X|$ 和约束数 $|\Phi|$ 也不超过 $f(|X_0| + |\Phi_0| + 1/\varepsilon + r)$。

证明　对任意所输入的 r-List-2CSP 的实例

$$\Pi_0 = (X_0, \Sigma_0, \Phi_0)$$

（它也是 2CSP 的实例），由引理 5.1，我们可以从 Π_0 构造一个二分的 2CSP 实例

$$\Pi_1 = (X_1, \Sigma_1, \Phi_1)$$

再由引理 5.2，从 Π_1 构造一个 r-AvgList-2CSP 的实例

$$\Pi = (X, \Sigma, \Phi)$$

显然，当 Π_0 可满足时，Π_1 是可满足的，自然 Π 也是可满足的。

如果 Π_0 不是 r-列表可满足的，则由引理 5.1，对于任意 $r_1, r_2 > 0$ 且 $r_1 + r_2 \geq 2(1-\varepsilon)r$，$\Pi_1$ 不是 (r_1, r_2)-列表可满足的，再由引理 5.2可知，Π 不是 $(1-\varepsilon)r$-平均列表可满足的。

由引理 5.1，Π_1 有矩形约束，再由引理 5.2，Π 也有矩形约束。

这一归约过程的运行时间为

$$f(|X_0| + |\Phi_0| + 1/\varepsilon + r)|\Sigma_0|^{O(1)}$$

其中 f 是一个可计算的函数，且 X 和 Φ 的大小满足

$$|X| + |\Phi| \leq f(|X_0| + |\Phi_0| + 1/\varepsilon + r)|\Sigma_0|^{O(1)}$$

从而说明该归约是 FPT 的。　■

5.3 r-AvgList-2CSP 的困难性

本节我们讨论 r-AvgList-2CSP 这一问题的近似困难性及其推广。

5.3.1 W[1]-困难性

Guruswami 等人[41]证明了:

定理 5.3 对任意常数 $r \geq 1$，r-List-2CSP 是 W[1]-完全的。

将前述的各个部分合起来，可以得到定理 5.1的证明如下。

定理 5.1的证明 由定理 5.3和定理 5.2直接得到。 ■

5.3.2 稀疏和稠密实例的二分性

本节中我们讨论 r-AvgList-2CSP 在不同类型实例上困难性的区别。我们首先给出 2CSP 实例中稠密与稀疏的定义。

定义 5.3 对于一个 2CSP 实例 $\Pi = (X, \Sigma, \Phi)$，如果 $|\Phi| = \omega(|X|)$，则 Π 被称作稠密的，否则被称作稀疏的。

我们将证明如下的二分性结果:

定理 5.4 以下两条均成立:

- 对任意常数 $r \geq 1$，r-AvgList-2CSP 在稠密实例上是 W[1]-困难的;
- 若存在常数 $r > 1$，r-AvgList-2CSP 在稀疏实例上是 W[1] 困难的，则存在常数 $\varepsilon > 0$，ε-Gap-2CSP 是 W[1] 困难的。换言之，PIH（假设 2.4）在 W[1] $\neq$ FPT 下成立。

为增进可读性，我们将定理 5.4的两部分分为下面两个引理分开证明。

引理 5.3 对任意常数 $r \geq 1$，r-AvgList-2CSP 在稠密实例上是 W[1]-困难的。

证明 注意到引理 5.1中的归约给出的二分 2CSP 实例

$$\Pi_1 = (X_1 \dot{\cup} X_2, \Sigma_1, \Phi_1)$$

是完全的，即对每个 $x_1 \in X_1$ 和 $x_2 \in X_2$，都存在一个约束

$$\varphi = (x_1 x_2, C) \in \Phi$$

引理 5.2将 Π_1 中的 X_1 复制 $|X_2|$ 遍，X_2 复制 $|X_1|$ 遍，且在每一对拷贝间都保留一份约束。因此，定理 5.2的输出实例 $\Pi = (X, \Sigma, \Phi)$ 中约束数为

$$|\Phi| = |X_1|^2|X_2|^2 = \frac{|X|^2}{4}$$

考虑任意函数 $h \in \omega(1)$。我们可以构造新实例 $\Pi' = (X', \Sigma', \Phi')$，该实例仅仅只是将 Π 自身复制 t 份，其中 t 是满足下式的最小正整数：

$$h(t|X|) \geq \frac{|X|}{4}$$

注意到在这次复制中，不同的拷贝间不添加约束。因此，新的参数为 $|X'| = t|X|$, 以及

$$|\Phi'| = t|\Phi| = \frac{t|X|^2}{4} = \frac{|X|}{4}|X'| \leq h(|X'|)|X'|$$

显然这一归约运行时间是 FPT 的。如果 Π 是可满足的，则 Π' 也是可满足的。如果 Π 的任意满足的多赋值都需要至少 $r|X|$ 个值，则 Π' 的任意满足的多赋值需要给每一份 Π 的拷贝赋 $r|X|$ 个值，因此总共至少 $r|X'|$ 个值，即归约前后的间隔 r 是不变的。 ■

引理 5.4　若存在常数 $r > 1$，r-AvgList-2CSP 在稀疏实例上是 W[1] 困难的，则存在常数 $\varepsilon > 0$，ε-Gap-2CSP 是 W[1] 困难的。换言之，PIH（假设 2.4）在 W[1] ≠ FPT 下成立。

证明　令 $\Pi = (X, \Sigma, \Phi)$ 为一个 r-AvgList-2CSP 的稀疏的负实例，即 Π 的任意满足的多赋值需要赋超过 $r|X|$ 个值，且存在常数 $c > 0$ 满足

$$|\Phi| \leq c \cdot |X|$$

考虑 Π 的单赋值 $\sigma : X \to \Sigma$。假定 σ 违反了 t 个约束，则我们可以向 σ 中添加至多 $2t$ 个赋值，得到一个满足的多赋值 $\hat{\sigma}$，且该多赋值的大小不超过

$$|X| + 2t$$

由 Π 多赋值的可满足性，我们得到 $|X| + 2t > r|X|$，从而，

$$t > \frac{r-1}{2}|X| = \frac{r-1}{2c \cdot r} \cdot c \cdot r|X| \geq \frac{r-1}{2c \cdot r}|\Phi|.$$

换言之，任意 Π 的单赋值都至多满足 Π 中

$$\varepsilon = 1 - \frac{r-1}{2c \cdot r}$$

比例的约束。由此得出，对上述的 c, r, ε，恒等映射就是一个从 r-List-2CSP 到 ε-Gap-2CSP 的 FPT-归约。 ■

定理 5.4的证明 由引理 5.3和引理 5.4得到。 ■

本节的结果给出了另一种尝试在标准假设 $\mathsf{W}[1] \neq \mathsf{FPT}$ 下下证明 PIH 的可能方法：将 r-List-2CSP 的困难性改进到稀疏的实例上。

5.4 应用：证明 k-ExactCover 问题的近似困难性

我们给出一个从具有矩形约束的 r-AvgList-2CSP 到 r-Gap-k-ExactCover 的 FPT 归约，从而证明任意常数比近似 k-ExactCover 都是 $\mathsf{W}[1]$-困难的。注意到文献[41]也给出了相同的结果，其归约是基于 (T, m)-集合部件[10,42]的归约，而我们的归约基于另一种对编码碰撞数（定义 3.1）的应用。

我们先回顾下述“超立方体分划系统（Hypercube Partition System）”的定义：

定义 5.4 令 A, B 为两个非空集合。定义(A, B)-超立方体分划系统为

- 基集 $\mathcal{M} = A^B \left(= \left\{ z \,\middle|\, \text{函数 } z : B \to A \right\} \right)$，以及
- 上述基集的一个子集的集合 $\left\{ P_{x,y} \right\}_{x \in B, y \in A}$,，其中每个 $P_{x,y} = \left\{ z \in \mathcal{M} \,\middle|\, z(x) = y \right\}$.

超立方体分划系统有如下良好的性质：

引理 5.5 给定非空集合 A, B 的 (A, B)-超立方体分划系统 $(\mathcal{M}, \{P_{x,y}\})$，任何从 $\{P_{x,y}\}$ 中取的 $\mathcal{M}$ 的覆盖 $S \subseteq \{P_{x,y}\}$ 都满足：存在 $b \in B$，对每个 $y \in A$，都有 $P_{b,y} \in C$。

证明 若不然，假设对任意 $x \in B$，都存在一个 $y_x \in A$ 使得

$$P_{x,y_x} \notin C$$

则考虑函数 $z : B \to A$ 如下：

$$\forall x \in B, z(x) = y_x$$

显然 $z \in \mathcal{M}$。但 C 中不存在任何集合包含 z。我们可以将 C 中的集合写为

$$C = \{P_{x_1,y_1}, P_{x_1,y_2}, \cdots, P_{x_m,y_{m'}}\}$$

对每个 $x \in B$，z 只在 $= y_x$ 时被 $P_{x,y}$ 所包含，但由我们的假设 $P_{x,y_x} \notin C$，C 中所有的 $P_{x,y_1}, \cdots, P_{x,y_{m'}}$ 均不包含 z。这一论断适用于所有 $x \in B$，因此我们得到 C 中的集合全部都不包含 z，这与 C 是 $\mathcal{M}$ 的覆盖所矛盾。 ■

定理 5.5 (参见[41]的定理 21)　若 r-AvgList-2CSP 在有矩形约束的实例上是 W[1]-困难的，则 r-Gap-k-ExactCover 也是 W[1]-困难的。准确来说，对任意常数 $r > 1$，输入 k-SetCover 实例 $\Pi = (S, U)$ 和整数 $k \geq 1$，区分下面两种情况是 W[1]-困难的：

- S 中存在 k 个互不相交的集合，它们的并集 U；
- U 不是 S 中任意 rk 个集合的并。

证明　令 $\Pi = (X, \Sigma, \Phi)$ 为一个具有矩形约束的 r-AvgList-2CSP 实例。我们令 $k = |X|$。对每个矩形约束

$$\varphi_j = (x_{i_1} x_{i_2}, C_j) \in \Phi$$

我们记 Q_j 为它的基集，$\pi_j, \rho_j : \Sigma \to Q_j$ 为相应的映射（参见定义 5.1）。由此，对任意 $a, b \in \Sigma$，我们得到 $(a, b) \in C_j$ 当且仅当 $\pi_j(a) = \rho_j(b)$。最后，我们令

$$t = \max_{\varphi_j \in \Phi} |Q_j| \tag{5.13}$$

显然，我们可以不失一般性地假设

$$t \leq |\Pi|$$

我们现在将 Π 归约到一个 k-ExactCover 的实例。为此，我们选择一个大小为素数的新字母表 Δ，且满足

$$\max\left\{\lceil \log t \rceil, 2^{2r^2k^2}\right\} \leq |\Delta| \leq 2\max\left\{\lceil \log t \rceil, 2^{2r^2k^2}\right\}$$

再令

$$d = \left\lceil \frac{2r^2k^2 \log t}{\log |\Delta|} \right\rceil$$

由此我们可以构造出下述的具有很大的相对距离的编码（其本质上还是一个在该参数下的 Reed-Solomon 编码）

$$\mathrm{Enc} : \Delta^{\left\lceil \frac{\log t}{\log |\Delta|} \right\rceil} \to \Delta^d.$$

将参数

$$k \leftarrow \left\lfloor \frac{\log t}{\log |\Delta|} \right\rfloor,\ m \leftarrow d,\ p \leftarrow |\Delta|,^{①} \text{ and } \varepsilon \leftarrow 1/2$$

插入定理 3.3中，可以得到编码 Enc 的 1/2-碰撞数为

$$\mathrm{Col}_{1/2}(\mathrm{Enc}) \geq \sqrt{\frac{d}{\log t / \log |\Delta|}} > rk$$

① 注意到

$$\left\lfloor \frac{\log t}{\log |\Delta|} \right\rfloor < \left\lceil \frac{2r^2k^2 \log t}{\log |\Delta|} \right\rceil \leq \lceil \log t \rceil \leq |\Delta|,$$

定理 3.3所要求的条件 $k < m \leq p$ 得以满足。

注意到式子 (5.13)表明约束集 C_i 中的每一对值都可以表示为 $\Delta^{\left\lceil \frac{\log m}{\log|\Delta|} \right\rceil}$，也就是 Enc 的定义域中的一个字符串。

由此，对每个变量 $x \in X$ 和字母表中每个可能的值 $a \in \Sigma$ 我们定义一个集合 $S_{x,a}$ 如下。对每个矩形约束

$$\varphi_j = (x_{i_1} x_{i_2}, C_j) \in \Phi$$

其基集 Q_j 以及对应的映射 $\pi_j, \rho_j : \Sigma \to Q_j$，以及对每个 $\ell \in [d]$，我们构造一个 $([2], \Delta)$-超立方体分划系统

$$\left(\mathcal{M}^{(j,\ell)}, \{P^{(j,\ell)}_{u,v}\}_{u \in \Delta, v \in [2]} \right) \tag{5.14}$$

对每个 $(a,b) \in C_j$，我们将 $P^{(j,\ell)}_{\mathrm{Enc}(\pi_j(a))[\ell],1}$ 所包含的内容加入集合 $S_{x_{i_1},a}$ 中，将 $P^{(j,\ell)}_{\mathrm{Enc}(\rho_j(b))[\ell],2}$ 所包含的内容加入集合 $S_{x_{i_2},b}$ 中。

最终，令这一 k-ExactCover 实例的待覆盖集合 U 为

$$U = \dot{\bigcup}_{\varphi_j \in \Phi, \ell \in [d]} \mathcal{M}^{(j,\ell)}$$

以及子集族 $\mathcal{S}$ 为

$$\mathcal{S} = \left\{ S_{x,a} \,\middle|\, x \in X \text{ and } a \in \Sigma \right\}$$

我们首先讨论这一归约的完备性。令 $\sigma : X \to \Sigma$ 为实例 Π 的一个可满足的赋值，我们说明 $\{S_{x,\sigma(x)}\}_{x \in X}$ 构成 U 的一个划分。不失一般性，我们任取一个约束 $\varphi_j = (x_{i_1} x_{i_2}, C_j) \in \Phi$，对每个 $\ell \in [d]$，注意到

$$\mathcal{M}^{(j,\ell)} \cap S_{x_{i_1},\sigma(x_{i_1})} = P^{(j,\ell)}_{\mathrm{Enc}(\pi(\sigma(x_{i_1})))[\ell],1} \in \{P^{(j,\ell)}_{u,v}\}_{u \in \Delta, v \in [2]},$$

以及

$$\mathcal{M}^{(j,\ell)} \cap S_{x_{i_2},\sigma(x_{i_2})} = P^{(j,\ell)}_{\mathrm{Enc}(\rho(\sigma(x_{i_2})))[\ell],2} \in \{P^{(j,\ell)}_{u,v}\}_{u \in \Delta, v \in [2]},$$

且由于赋值 σ 是可满足的，由矩形约束的性质

$$\mathrm{Enc}(\pi(\sigma(x_{i_1})))[\ell] = \mathrm{Enc}(\rho(\sigma(x_{i_2})))[\ell],$$

从而由引理 5.5，

$$\mathcal{M}^{(j,\ell)} \cap S_{x_{i_1},\sigma(x_{i_1})} \quad \text{和} \quad \mathcal{M}^{(j,\ell)} \cap S_{x_{i_2},\sigma(x_{i_2})}$$

构成 $\mathcal{M}^{(j,\ell)}$ 的一个划分，这一论证对所有 $\mathcal{M}^{(j,\ell)}$ 都成立，从而 $\{S_{x,\sigma(x)}\}_{x \in X}$ 构成了 U 的一个划分。

再讨论归约的可靠性。假设实例 Π 的每个可满足的多赋值都至少需要 $rk = r|X|$ 个值（参见定义 2.8）。令 $\mathcal{S}' \subseteq \mathcal{S}$ 为 U 的一个覆盖。考虑如下的多赋值：

- 给每个 $x \in X$ 赋集合 $\left\{a \in \Sigma \,\middle|\, S_{x,a} \in \mathcal{S}'\right\}$ 的值。

  若该多赋值可以满足 Π，则由我们的假设可知 $|\mathcal{S}'| \geq rk$。

  现在考虑该多赋值不满足 Π 的情况，这就意味着存在某个约束

$$\varphi_j = (x_{i_1} x_{i_2}, C_j) \in \Phi$$

未被该多赋值满足。注意到该多赋值给变量 x_{i_1} 指派了值的集合

$$E_1 = \left\{a \in \Sigma \,\middle|\, S_{x_{i_1},a} \in \mathcal{S}'\right\},$$

x_{i_2} 指派了值的集合

$$E_2 = \left\{b \in \Sigma \,\middle|\, S_{x_{i_2},b} \in \mathcal{S}'\right\},$$

由于 φ_j 未被满足，任意值对 $(a, b) \in E_1 \times E_2$ 都有

$$\mathrm{Enc}(\pi_j(a)) \neq \mathrm{Enc}(\rho_j(b))$$

然而，对每个 $\ell \in [d]$，超立方体分划系统中的 $\mathcal{M}^{(j,\ell)}$ 是被 $\mathcal{S}'$ 所覆盖的，由引理 5.5，这意味着必然存在 $a \in E_1$ 和 $b \in E_2$ 满足

$$\mathrm{Enc}(\pi_j(a))[\ell] = \mathrm{Enc}(\rho_j(b))[\ell]$$

由此可得，集合

$$\{\pi_j(a)\}_{a \in E_1} \cup \{\rho_j(b)\}_{b \in E_2}$$

在所有位置 $\ell \in [d]$ 上均产生碰撞，由编码 Enc 的碰撞数的定义，这一集合的大小至少应该是

$$\mathrm{Col}_{1/2}(\mathrm{Enc}) > rk$$

由此，我们得到：

$$|\mathcal{S}'| \geq |E_1| + |E_2| \geq \left|\{\pi_j(a)\}_{a \in E_1} \cup \{\rho_j(b)\}_{b \in E_2}\right| \geq \mathrm{Col}_{1/2}(\mathrm{Enc}) > rk$$

也就是说，不论该多赋值可满足与否，$\mathcal{S}'$ 均包含至少 rk 个集合。

接下来我们讨论关于该归约的时间复杂度。注意到在我们构造的每个超立方体分划系统 $(\mathcal{M}^{(j,\ell)}, \{P_{u,v}^{(j,\ell)}\}_{u \in \Delta, v \in [2]})$ 中，

$$\left|\mathcal{M}^{(j,\ell)}\right| = 2^{|\Delta|} \leq 4^{\lceil \log t \rceil} + 4^{2^{2r^2k^2}} \leq |\Pi|^2 + 4^{2^{2r^2k^2}}$$

且我们至多需要构造

$$\binom{k}{2} d \le k^2 r^2 k^2 \log t \le r^2 k^4 \log |\Pi|$$

个这样的系统。待覆盖的集合 U 的大小即为 $g(r,k)|\Pi|^3$，其中 $g : \mathbb{N}^2 \to \mathbb{N}$ 为某个适宜的可计算函数。

从 r-AvgList-2CSP 实例到 k-ExactCover 实例的归约中，我们保持了参数 $k = |X|$。由此，这一归约是 FPT 归约。 ■

将定理 5.1和定理 5.5结合起来，我们得到：

定理 5.6 对任意常数 $r > 1$，r-Gap-k-ExactCover 是 W[1]-困难的。

5.5 应用：r-AvgList-2CSP 与 r-Gap-k-NCP 的时间下界关联

本节我们证明 r-AvgList-2CSP 与 r-Gap-k-NCP 的运行时间下界有着紧密的关联。这一结果实际上是两个保持参数不变的 FPT 归约的复合，因此我们仅简述其证明概要。

定理 5.7 对任意素数 p，可计算函数 g，以及常数 r，如果对任意可计算函数 f_1，拥有 k 个变量的 r-AvgList-2CSP 实例没有运行时间为 $f_1(k) \cdot n^{o(g(k))}$ 的算法，则对任意可计算函数 f_2，r-Gap-k-NCP$_p$ 也没有运行时间为 $f_2(k) \cdot n^{o(g(k))}$ 的算法。

证明概要 注意到定理 5.5（以及文献[41]的定理 22）中给出的归约是 FPT 的，且保证了归约前后参数 k 和近似比 r 均不改变。文献[13]的定理 28（也可参见文献[11]的定理 5）给出了从 r-Gap-k-ExactCover 到 r-Gap-k-NCP$_p$ 的归约，且归约前后参数 k 和近似比 r 也均不改变。将这两个归约结合起来就得到了从 r-AvgList-2CSP 与 r-Gap-k-NCP$_p$ 的保持参数 k 和近似比 r 的归约。由此，r-AvgList-2CSP 的运行时间下界和 r-Gap-k-NCP$_p$ 的运行时间下界是一致的。 ■

定理 5.7的意义在于，它给出了不同于第4章的另外一种证明编码问题近似算法的运行时间下界的方法：证明本章中讨论的近似约束满足问题多赋值总大小的近似算法所需的运行时间下界。

5.6 本章小结

本章使用第3章中发展的技术工具，讨论了 2-约束满足问题的多赋值大小的近似困难性。我们在第 5.2节中给出了从近似可满足多赋值的个体大小到近似其总大小

的 FPT 归约，之后将这一归约与已有的结果结合，在第 5.3节中给出了近似其总大小的困难性的证明，并讨论了这一困难性结果在稀疏与稠密实例上的成立情况。之后，我们在第 5.4节中将这一困难性结果应用于证明常数比近似 k-ExactCover 问题的 W[1]-困难性，并在第 5.5节中讨论了归约的参数增长情况和这几个问题的运行时间之间的紧密关系。

第 6 章 总结

本文研究了几个参数化问题的近似困难性。我们首先讨论了编码的碰撞数这一有用的技术工具，并对其参数做了一定的改进。

通过使用编码的碰撞数这一技术工具，本文做出了下面两部分主要工作：

1. 关于最近编码问题（近似解码问题），我们设计了一个新的创造间隔的自归约，显著地简化了这一问题之前的近似困难性证明，且保证了归约过程中参数的增长在多项式级别，改进了之前工作中的指数级别增长。由此，我们显著地改进了（随机）指数时间假设下近似一系列参数化编码问题与格问题的时间下界，达到了 $n^{\Omega(k^\epsilon)}$，缩小了这一时间下界与现有的暴力穷举算法运行时间的差距。
2. 关于约束满足问题，我们设计了从单个变量上赋值数量的间隔到整体赋值数量的归约。由此，在前文结果的基础上，我们证明了常数比近似参数化约束满足问题的可满足多赋值的大小是 W[1]-困难的。我们更进一步地给出了从上述问题到近似参数化精确覆盖问题的归约，证明了任意常数比近似该问题都是 W[1]-困难的。

本文所使用的证明构造均非常简单，方法也是较为初等的方法，易于理解，且达到了非常好的效果，增加了我们对这几个参数化问题的理解。

6.1 问题展望

在编码与格问题的近似算法与近似困难性方面，注意到我们在第 4章中的结果只给出了（随机）指数时间假设下的常数比近似这些问题的困难性与时间下界，且我们证明的时间下界 $n^{\Omega(k^\epsilon)}$（其中 $\epsilon > 0$ 是一个常数）与暴力穷举算法的运行时间 $n^{O(k)}$ 之间仍有间隙，并不是紧的。在更强的 Gap-ETH 假设（区分 n 个变量的 3SAT 实例是可满足的或任意赋值都只能满足至多常数比例子句需要 $2^{\Omega(n)}$ 时间，参见[43-44]）下，Manurangsi[13]证明了任意常数比近似最近编码问题（以及格问题中的最近向量问题）都没有 $n^{o(k)}$ 时间的算法。一个自然的问题就是：

开放性问题 1. *在弱于 Gap-ETH 的假设下证明：对任意常数 $\gamma > 0$，没有 $n^{o(k)}$ 时间的算法可以以 γ 比例近似 k-NCP 和 k-CVP 问题。*

正如 Manurangsi 在其文章中的评论[13]所言，开放性问题 1的主要瓶颈在于需要

给出一个“单步的”证明，即一次构造就给出任意大常数比例的近似困难性。与之相对的是我们的证明分为创造间隔的归约（定理 4.4）和放大间隔的归约（定理 4.2）两步，这使得参数有了多项式级别的增长，同时由于运行时间也指数地依赖于近似比，这限制了我们达到超过常数近似比的结果。

相应地，对于剩下几个问题，与 Manurangsi 的论文[13]中已经达到紧的时间下界结果相应，我们可以提出下面的问题：

开放性问题 2. *在弱于 Gap-ETH 的假设下证明：存在常数 $\gamma > 0$，没有 $n^{o(k)}$ 时间的算法可以以 γ 比例近似 k-MDP 问题和 k-SVP 问题。*

特别地，在最短距离问题和最短向量问题的近似困难性方面，归约的随机化与否也是关注的焦点。在传统的计算复杂性领域中，最短距离问题的 NP-困难性的证明最终被成功去随机化（可参见[45-46]），但最短向量问题问题的 NP-困难性的证明仍然严重依赖于随机化的归约。这一问题在参数复杂性中依然存在且更加严重，文献[26-27]中关于最短距离问题和最短向量问题归约均强依赖于随机性。因此，我们提出如下将参数复杂性中这两个问题的近似困难性证明去随机化的问题。

开放性问题 3. *不使用随机性证明存在常数 $\gamma > 0$，γ 比例近似 k-MDP 问题和 k-SVP 问题在 ETH 或 $\mathsf{W}[1] \neq \mathsf{FPT}$ 假设下的困难性。*

在近似比方面，Alon 等人[47]给出了最近编码问题近似比为 $O(n/\log n)$ 的确定性多项式时间近似算法，而我们的近似困难性只针对于常数近似比。因此，一个可探究的方向就是尝试缩小这两方面之间的空白。

开放性问题 4. *是否可以在 ETH 或 $\mathsf{W}[1] \neq \mathsf{FPT}$ 假设下证明超过常数比近似 k-NCP 问题的困难性，或给出这一问题的近似比为 $o(n/\log n)$ 的 FPT 算法？*

在参数化约束满足问题方面，我们所证的困难性结果仍然只适用于任意常数比近似。因此，将结果推广到更大的近似困难性，包括任意的与参数 k 相关的近似比，仍是一个值得探究的问题。

开放性问题 5. *是否可以证明对近似比 $r = r(k) = \omega(1)$，r-AvgList-2CSP 是 $\mathsf{W}[1]$-困难的？*

开放性问题 5的主要瓶颈不在于本文中归约的步骤，而是在于 Guruswami 等人[41]证明的定理 5.3只对常数近似比有效。他们的归约的运行时间以及参数增长情

况指数依赖于要达到的近似比，因此近似比必须为常数。如果能将 r-List-2CSP 的 W[1]-困难性推广到超过常数的近似比 r 上，则我们的归约可以立即给出 r-AvgList-2CSP 在相同量级近似比下的困难性结果。

除此之外，我们还可以探究 r-AvgList-2CSP 的一些加强的变种。第一种变种是要求任意可满足多赋值必须在常数比例的变量上赋多个值。

开放性问题 6. *是否可以证明对任意常数 $r > 2$，存在常数 $c > 0$，判定一个给定的 2CSP 实例 $\Pi = (X, \Sigma, \Phi)$*

- *是可满足的，或*
- *任意可满足的多赋值都要给至少 $c|X|$ 个变量中每个赋至少 r 个值*

是 W[1]-困难的？

本文的证明中，引理 5.1的情况 1只保证了对 X_1 这一整个集合的赋值数量很大，对其中有多少个变量被赋多个值没有规定。开放性问题 6进一步追问能否将这一情况也消去。注意到当 $r = 2$ 时，这一情况是成立的，可以由 k-Clique 问题的近似困难性得出，参见 Chen，Lin 的综述论文[48]中的命题 5.22。

另一种变种即为我们在定理 5.4中所讨论的情况。作为开放性问题，我们希望探究是否能通过这一方法证明 PIH。

开放性问题 7. *是否可以证明r-AvgList-2CSP 在约束数量是变量数量的常数倍的情况下仍然是 W[1]-困难的？*

本文第 3章中对高碰撞数编码进行了讨论。这一编码性质最初被提出是用于构造一类特殊的 “临界图”（参见[23-24]），而此类 “临界图” 在证明一些别的问题的近似困难性方面也起到了较好的效果，例如集合覆盖问题[21-22]，关于这样的编码性质以及本文中证明近似困难性的方法论，我们考虑这样方法的推广或局限性。

开放性问题 8. *是否存在高碰撞数编码的构造满足编码长度 $m = O(k)$，碰撞数为 $\Omega(k)$？或者是否可以证明这样的编码必然不存在？*

如果这样的编码存在，则我们文中归约的参数增长情况可以得到进一步改善，从而得到更紧的运行时间下界结果。

最后，关于证明的方法论部分，本文中证明几个问题困难性的方法有其共通之处，即对于文中所考虑的最小化问题，我们构造出由两部分变量组成的新的实例，且论证在可靠性情况下至少其中一部分变量的优化值需要产生一个较大的间隔。我们

最后的问题是，这一方法论能否被整理为一个完整的框架，从而将更多参数化的最小化优化问题一并处理?

开放性问题 9. 能否构造一个通用的框架来证明参数化的最小化问题的近似困难性？

参考文献

[1] COOK S A. The Complexity of Theorem-Proving Procedures[C/OL]//HARRISON M A, BANERJI R B, ULLMAN J D. Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA. ACM, 1971: 151-158. https://doi.org/10.1145/800157.805047. DOI: 10.1145/800157.805047.

[2] LEVIN L A. Universal sequential search problems[J]. Problems of information transmission, 1973, 9(3): 265-266.

[3] KARP R M. Reducibility Among Combinatorial Problems[C/OL]//MILLER R E, THATCHER J W. The IBM Research Symposia Series: Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA. Plenum Press, New York, 1972: 85-103. https://doi.org/10.1007/978-1-4684-2001-2%5C_9. DOI: 10.1007/978-1-4684-2001-2_9.

[4] IMPAGLIAZZO R, PATURI R. On the Complexity of k-SAT[J/OL]. J. Comput. Syst. Sci., 2001, 62(2): 367-375. https://doi.org/10.1006/jcss.2000.1727. DOI: 10.1006/JCSS.2000.1727.

[5] IMPAGLIAZZO R, PATURI R, ZANE F. Which Problems Have Strongly Exponential Complexity?[J/OL]. J. Comput. Syst. Sci., 2001, 63(4): 512-530. https://doi.org/10.1006/jcss.2001.1774. DOI: 10.1006/JCSS.2001.1774.

[6] ARORA S, SAFRA S. Probabilistic Checking of Proofs: A New Characterization of NP[J/OL]. J. ACM, 1998, 45(1): 70-122. https://doi.org/10.1145/273865.273901. DOI: 10.1145/273865.273901.

[7] ARORA S, LUND C, MOTWANI R, et al. Proof Verification and the Hardness of Approximation Problems[J/OL]. J. ACM, 1998, 45(3): 501-555. https://doi.org/10.1145/278298.278306. DOI: 10.1145/278298.278306.

[8] DINUR I. The PCP theorem by gap amplification[J/OL]. J. ACM, 2007, 54(3): 12. https://doi.org/10.1145/1236457.1236459. DOI: 10.1145/1236457.1236459.

[9] FEIGE U, GOLDWASSER S, LOVÁSZ L, et al. Interactive Proofs and the Hardness of Approximating Cliques[J/OL]. J. ACM, 1996, 43(2): 268-292. https://doi.org/10.1145/226643.226652. DOI: 10.1145/226643.226652.

[10] ALON N, MOSHKOVITZ D, SAFRA S. Algorithmic construction of sets for *k*-restrictions[J/OL]. ACM Trans. Algorithms, 2006, 2(2): 153-177. https://doi.org/10.1145/1150334.1150336. DOI: 10.1145/1150334.1150336.

[11] ARORA S, BABAI L, STERN J, et al. The Hardness of Approximate Optima in Lattices, Codes, and Systems of Linear Equations[J/OL]. J. Comput. Syst. Sci., 1997, 54(2): 317-331. https://doi.org/10.1006/jcss.1997.1472. DOI: 10.1006/JCSS.1997.1472.

[12] MANURANGSI P. Approximation and Hardness: Beyond P and NP[D/OL]. University of California, Berkeley, USA, 2019. https://www.escholarship.org/uc/item/8xg105gg.

[13] MANURANGSI P. Tight Running Time Lower Bounds for Strong Inapproximability of Maximum k-Coverage, Unique Set Cover and Related Problems (via t-Wise Agreement Testing Theorem)[C/OL]//CHAWLA S. Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020. SIAM, 2020: 62-81. https://doi.org/10.1137/1.9781611975994.5. DOI: 10.1137/1.9781611975994.5.

[14] CHALERMSOOK P, CYGAN M, KORTSARZ G, et al. From Gap-Exponential Time Hypothesis to Fixed Parameter Tractable Inapproximability: Clique, Dominating Set, and More[J/OL]. SIAM J. Comput., 2020, 49(4): 772-810. https://doi.org/10.1137/18M1166869. DOI: 10.1137/18M1166869.

[15] LIN B. Constant approximating k-clique is w[1]-hard[C/OL]//KHULLER S, WILLIAMS V V. STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021. ACM, 2021: 1749-1756. https://doi.org/10.1145/3406325.3451016. DOI: 10.1145/3406325.3451016.

[16] Karthik C. S., KHOT S. Almost Polynomial Factor Inapproximability for Parameterized k-Clique [C/OL]//LOVETT S. LIPIcs: 37th Computational Complexity Conference, CCC 2022, July 20-23, 2022, Philadelphia, PA, USA: vol. 234. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022: 6:1-6:21. https://doi.org/10.4230/LIPIcs.CCC.2022.6. DOI: 10.4230/LIPICS.CCC.2022.6.

[17] LIN B, REN X, SUN Y, et al. On Lower Bounds of Approximating Parameterized k-Clique[C/OL]//BOJANCZYK M, MERELLI E, WOODRUFF D P. LIPIcs: 49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France: vol. 229. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022: 90:1-90:18. https://doi.org/10.4230/LIPIcs.ICALP.2022.90. DOI: 10.4230/LIPICS.ICALP.2022.90.

[18] LIN B, REN X, SUN Y, et al. Improved Hardness of Approximating k-Clique under ETH[C/OL]//64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023. IEEE, 2023: 285-306. https://doi.org/10.1109/FOCS57990.2023.00025. DOI: 10.1109/FOCS57990.2023.00025.

[19] CHEN Y, FENG Y, LAEKHANUKIT B, et al. Simple Combinatorial Construction of the $k^{o(1)}$-Lower Bound for Approximating the Parameterized k-Clique[C/OL]//BERCEA I O, PAGH R. 2025 Symposium on Simplicity in Algorithms, SOSA 2025, New Orleans, LA, USA, January 13-15, 2025. SIAM, 2025: 263-280. https://doi.org/10.1137/1.9781611978315.21. DOI: 10.1137/1.9781611978315.21.

[20] Karthik C. S., LAEKHANUKIT B, MANURANGSI P. On the Parameterized Complexity of Approximating Dominating Set[J/OL]. J. ACM, 2019, 66(5): 33:1-33:38. https://doi.org/10.1145/3325116. DOI: 10.1145/3325116.

[21] CHEN Y, LIN B. The Constant Inapproximability of the Parameterized Dominating Set Problem

[J/OL]. SIAM Journal on Computing, 2019, 48(2): 513-533. eprint: https://doi.org/10.1137/17M1127211. https://doi.org/10.1137/17M1127211. DOI: 10.1137/17M1127211.

[22] LIN B. A Simple Gap-Producing Reduction for the Parameterized Set Cover Problem[C/OL]//BAIER C, CHATZIGIANNAKIS I, FLOCCHINI P, et al. LIPIcs: 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece: vol. 132. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019: 81:1-81:15. https://doi.org/10.4230/LIPIcs.ICALP.2019.81. DOI: 10.4230/LIPICS.ICALP.2019.81.

[23] Karthik C. S., NAVON I L. On Hardness of Approximation of Parameterized Set Cover and Label Cover: Threshold Graphs from Error Correcting Codes[C/OL]//LE H V, KING V. 4th Symposium on Simplicity in Algorithms, SOSA 2021, Virtual Conference, January 11-12, 2021. SIAM, 2021: 210-223. https://doi.org/10.1137/1.9781611976496.24. DOI: 10.1137/1.9781611976496.24.

[24] LIN B, REN X, SUN Y, et al. Constant Approximating Parameterized *k*-SETCOVER is W[2]-hard [C/OL]//BANSAL N, NAGARAJAN V. Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023. SIAM, 2023: 3305-3316. https://doi.org/10.1137/1.9781611977554.ch126. DOI: 10.1137/1.9781611977554.CH126.

[25] FELDMANN A E, Karthik C. S., LEE E, et al. A Survey on Approximation in Parameterized Complexity: Hardness and Algorithms[J/OL]. Algorithms, 2020, 13(6): 146. https://doi.org/10.3390/a13060146. DOI: 10.3390/A13060146.

[26] BHATTACHARYYA A, BONNET É, EGRI L, et al. Parameterized Intractability of Even Set and Shortest Vector Problem[J/OL]. J. ACM, 2021, 68(3): 16:1-16:40. https://doi.org/10.1145/3444942. DOI: 10.1145/3444942.

[27] BENNETT H, CHERAGHCHI M, GURUSWAMI V, et al. Parameterized Inapproximability of the Minimum Distance Problem over All Fields and the Shortest Vector Problem in All *l*p Norms [J/OL]. SIAM J. Comput., 2024, 53(5): 1439-1475. https://doi.org/10.1137/23m1573021. DOI: 10.1137/23M1573021.

[28] LIN B. The Parameterized Complexity of the *k*-Biclique Problem[J/OL]. J. ACM, 2018, 65(5): 34:1-34:23. https://doi.org/10.1145/3212622. DOI: 10.1145/3212622.

[29] DOWNEY R G, FELLOWS M R. Parameterized Complexity[M/OL]. New York, USA: Springer, 1999: 1-533. https://doi.org/10.1007/978-1-4612-0515-9. DOI: 10.1007/978-1-4612-0515-9.

[30] FLUM J, GROHE M. Parameterized Complexity Theory[M/OL]. Heidelberg: Springer Berlin, 2006: 1-495. https://doi.org/10.1007/3-540-29953-X. DOI: 10.1007/3-540-29953-X.

[31] CYGAN M, FOMIN F V, KOWALIK Ł, et al. Parameterized algorithms[M/OL]. Cham: Springer Cham, 2015: 1-613. DOI: https://doi.org/10.1007/978-3-319-21275-3.

[32] ARORA S, BARAK B. Computational complexity: a modern approach[M]. Cambridge: Cambridge University Press, 2009: 1-579.

[33] 傅育熙. 计算复杂性理论[M]. 北京: 清华大学出版社, 2023: 1-379.

[34] VAZIRANI V V. Approximation algorithms[M]. Heidelberg: Springer Berlin, 2001: 1-380.

[35] LIU Y, CHEN Y, LI S, et al. On Average Baby PIH and Its Applications[C/OL]//BEYERSDORFF O, PILIPCZUK M, PIMENTEL E, et al. LIPIcs: 42nd International Symposium on Theoretical Aspects of Computer Science, STACS 2025, March 4-7, 2025, Jena, Germany: vol. 327. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025: 65:1-65:19. https://doi.org/10.4230/LIPIcs.STACS.2025.65. DOI: 10.4230/LIPICS.STACS.2025.65.

[36] LOKSHTANOV D, RAMANUJAN M S, SAURABH S, et al. Parameterized Complexity and Approximability of Directed Odd Cycle Transversal[C/OL]//CHAWLA S. Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020. SIAM, 2020: 2181-2200. https://doi.org/10.1137/1.9781611975994.134. DOI: 10.1137/1.9781611975994.134.

[37] REED I S, SOLOMON G. Polynomial Codes Over Certain Finite Fields[J/OL]. Journal of the Society for Industrial and Applied Mathematics, 1960, 8(2): 300-304. eprint: https://doi.org/10.1137/0108018. https://doi.org/10.1137/0108018. DOI: 10.1137/0108018.

[38] GURUSWAMI V, RUDRA A, SUDAN M. Essential coding theory[J]. Draft available at https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/, 2025, 2(1).

[39] BHATTACHARYYA A, GHOSHAL S, C. S. K, et al. Parameterized Intractability of Even Set and Shortest Vector Problem from Gap-ETH[C/OL]//CHATZIGIANNAKIS I, KAKLAMANIS C, MARX D, et al. Leibniz International Proceedings in Informatics (LIPIcs): 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018): vol. 107. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018: 17:1-17:15. https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2018.17. DOI: 10.4230/LIPIcs.ICALP.2018.17.

[40] HAVIV I, REGEV O. Tensor-based Hardness of the Shortest Vector Problem to within Almost Polynomial Factors[J/OL]. Theory of Computing, 2012, 8(23): 513-531. https://theoryofcomputing.org/articles/v008a023. DOI: 10.4086/toc.2012.v008a023.

[41] GURUSWAMI V, REN X, SANDEEP S. Baby PIH: Parameterized Inapproximability of Min CSP[C/OL]//SANTHANAM R. Leibniz International Proceedings in Informatics (LIPIcs): 39th Computational Complexity Conference (CCC 2024): vol. 300. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024: 27:1-27:17. https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CCC.2024.27. DOI: 10.4230/LIPIcs.CCC.2024.27.

[42] LUND C, YANNAKAKIS M. On the hardness of approximating minimization problems[J/OL]. J. ACM, 1994, 41(5): 960-981. https://doi.org/10.1145/185675.306789. DOI: 10.1145/185675.306789.

[43] DINUR I. Mildly exponential reduction from gap 3SAT to polynomial-gap label-cover[J/OL]. Electron. Colloquium Comput. Complex., 2016, TR16-128. ECCC: TR16-128. https://eccc.weizmann.ac.il/report/2016/128.

[44] MANURANGSI P, RAGHAVENDRA P. A Birthday Repetition Theorem and Complexity of Ap-

proximating Dense CSPs[C/OL]//CHATZIGIANNAKIS I, INDYK P, KUHN F, et al. Leibniz International Proceedings in Informatics (LIPIcs): 44th International Colloquium on Automata, Languages, and Programming (ICALP 2017): vol. 80. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017: 78:1-78:15. https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2017.78. DOI: 10.4230/LIPIcs.ICALP.2017.78.

[45] CHENG Q, WAN D. A Deterministic Reduction for the Gap Minimum Distance Problem[J/OL]. IEEE Trans. Inf. Theory, 2012, 58(11): 6935-6941. https://doi.org/10.1109/TIT.2012.2209198. DOI: 10.1109/TIT.2012.2209198.

[46] AUSTRIN P, KHOT S. A Simple Deterministic Reduction for the Gap Minimum Distance of Code Problem[J/OL]. IEEE Trans. Inf. Theory, 2014, 60(10): 6636-6645. https://doi.org/10.1109/TIT.2014.2340869. DOI: 10.1109/TIT.2014.2340869.

[47] ALON N, PANIGRAHY R, YEKHANIN S. Deterministic Approximation Algorithms for the Nearest Codeword Problem[C]//DINUR I, JANSEN K, NAOR J, et al. Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009: 339-351.

[48] CHEN Y, LIN B. Parameterized inapproximability: From Clique to PIH[J/OL]. Computer Science Review, 2026, 59: 100834. https://www.sciencedirect.com/science/article/pii/S1574013725001108. DOI: https://doi.org/10.1016/j.cosrev.2025.100834.

致 谢

两年半的硕士生活即将画上句点，在即将离开学校踏上新的征程之际，我也要向一路上给予我支持和帮助的人致以诚挚的谢意。

首先要感谢我的导师陈翌佳教授。陈老师在研究方面给了我很大的自由，与陈老师的交流讨论总是富有启发性的。在我所欠缺的严谨性方面，陈老师的言传身教也使我受益良多。在日常的学习和科研之外，陈老师还为我争取了很多对外交流访问的机会，拓展了我的视野，让我对理论计算机科学领域的研究不再局限于某一个题目。同时，陈老师也一直叮嘱我不要一味寻求行路的捷径，要脚踏实地，勇于挑战困难。和陈老师一起学习的几年，将会成为我未来宝贵的精神财富。

我也要感谢南京大学的林冰凯教授。林老师自我本科期间就开始指导我在近似困难性领域进行研究，本文的工作也是和林老师一同完成的，林老师在科研方面的直观给了我很大的启发。在学业发展和生活方面，林老师也积极关心我在外访问期间的生活情况，为我提供了很多帮助。

感谢日本国立信息学研究所（NII）的平原秀一副教授。平原老师慷慨接待了我在 NII 的访问，让我了解到了元复杂性和与之相关的结构复杂性领域的广阔内容。在 NII 访问的三个月多时间里，我了解到了和之前课题大不相同的领域，这样的交流大大开拓了我的视野，也为我将来的研究指引了方向。

感谢 BASICS 实验室的傅育熙老师，傅老师的计算复杂性课程使我受益良多。感谢龙环老师，龙老师在实验室为我们解决了大大小小的问题，也在我们各自的学业生活领域给出了很多有益的建议。感谢龙环老师和张驰豪老师作为我的论文评审专家给我提出的建议。感谢尹强老师，杨宽老师，张宇昊老师和李国强老师在我研究生期间的帮助和交流。

感谢本文工作的合作者，南京大学的李双乐和郑欣同学，本文的工作是建立在和他们的密切合作上的。

感谢 BASICS 实验室和 318/327 办公室的刘国航、刘明君、俞逸洋、Carolina Johanna Reiß、周逸洋、叶梓淳、陈郁霖、李至丹、和昱辰、陈和达、陈厚双、吴昊、付博、陈蔚骏、徐崔、薛成峰、郑扬珞、邱国良、王玉林等同学，和你们度过的时光总是愉快的。感谢在这期间相遇的张文博博士，杨启哲博士，任瀚林博士，以及更多无法在此一一列明的朋友们。

感谢我的父母、家人和朋友们在这两年半时间内的支持。

硕士历程已到尾声，是时候走向新的挑战和际遇了。

学术论文和科研成果目录

学术论文

[1] Li S, Lin B, Liu Y. Improved Lower Bounds for Approximating Parameterized Nearest Codeword and Related Problems Under ETH[C]//51st International Colloquium on Automata, Languages, and Programming (ICALP 2024). Tallinn, Estonia: Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024: 107: 1-107: 20.

[2] Liu Y, Chen Y, Li S, et al. On Average Baby PIH and Its Applications[C]//42nd International Symposium on Theoretical Aspects of Computer Science (STACS 2025). Jena, Germany: Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2025: 65: 1-65: 19.

个人简历

基本情况

刘裕炜，2000 年 2 月生于云南省昆明市。

教育背景

- 2023 年 9 月至今，上海交通大学，硕士研究生，计算机科学与技术专业
- 2018 年 9 月至 2023 年 6 月，北京航空航天大学，本科，计算机科学与技术专业（辅修数学与应用数学专业）

研究兴趣

计算复杂性，近似困难性

联系方式

- 地址：上海市闵行区东川路 800 号，200240
- E-mail: yuwei.liu@sjtu.edu.cn