



维数对随机几何 SAT 的实证影响

维数对随机几何可满足性问题的实证影响

Flora Rädiker

为获得以下学术学位而撰写的大学学士论文

理学学士 *(B. Sc.)*

IT 系统工程专业

于 2025-09-16 提交至波茨坦大学数字工程学院算法工程专业

评审人 Prof. Dr. Tobias Friedrich**指导教师** Dr. Andreas Göbel Nadym Mallek

摘要

布尔可满足性问题（SAT）或许是理论计算机科学中最广为人知的问题之一。一方面，它已被证明是 NP 完全问题，这意味着通常认为它很难求解。另一方面，SAT问题有许多实际应用，由此产生了所谓的工业实例，而 SAT 求解器往往能够高效地找到这类实例的解。

弥合理论与实践之间的差距是当前的研究课题。一种方法是找出能够使 SAT 实例易于处理的性质。为此，人们提出了生成 SAT 实例的模型，以模拟工业实例的性质。迄今为止，构建此类模型的尝试大多未获成功：生成的实例要么太容易或太难求解，要么缺少工业SAT实例的重要性质。

在这项工作中，我们分析了由 Giráldez-Cru和Levy [GL17] 提出的一种颇具前景的 SAT 模型，该模型基于底层几何结构。我们通过实证分析了这种几何结构的维度对 SAT 实例的三项性质所产生的影响：可满足性阈值的位置、求解器运行时间，以及不可满足性证明的大小。在 Bläsius 等人[Blä+23]的理论研究基础上，我们发现低维几何实例始终非常易于处理。随着维度增加，几何模型生成的实例似乎会逐渐趋近于困难的均匀实例，这意味着该几何模型能够表示从简单到困难的全部实例范围。我们还观察到，低维几何实例的可满足性阈值出现在较低的密度下。此外，与均匀实例非常不同的是，低维实例在可满足性阈值处的求解器运行时间不会出现难度峰值。这与证明大小的情况相符；我们发现，在低维度下，证明大小与求解器运行时间并不相关。

摘要

命题逻辑可满足性问题（SAT）可能是理论计算机科学中最著名的问题之一。一方面，SAT 已被证明是 NP完全问题，因此通常认为它很难求解。另一方面，SAT 有许多实际应用场景，由此产生了所谓的工业实例，而 SAT求解器往往能够高效地找到这类实例的解。

弥合理论与实践之间的差距是当前研究的课题。一种方法是寻找 SAT实例中能使其更容易求解的特性。为此，人们开发了 SAT实例生成模型，以模拟工业实例的特性。迄今为止，开发此类模型的尝试大多未能成功，因为生成的实例要么过于简单，要么难以求解，或者缺少工业实例的重要特性。

在本文中，我们分析了一种前景广阔的 SAT 模型。该模型由 Giráldez-Cru 和 Levy [GL17] 提出，并利用了底层几何结构。我们通过三项特性，实证分析了该几何结构的维度对 SAT实例的影响：可满足性阈值的位置、求解时间和不可满足性证明大小。在补充Bläsius 等人 [Blä+23] 的理论研究后，我们发现，低维几何实例始终很容易求解。随着维度增加，几何模型生成的实例似乎会逐渐趋近于难以求解的均匀分布实例。这意味着，该几何模型能够涵盖从简单到困难的完整实例范围。我们还观察到，低维实例的可满足性阈值出现在较低密度处。此外，低维实例与均匀分布实例的表现截然不同：在可满足性阈值处，其求解时间并未出现难度峰值。与此相符，我们证明，在维度较低时，不可满足性证明大小与求解时间并不相关。

目录

摘要 iii

摘要 v

目录 vii

1 引言 1

1.1 可满足性背景.1 111 2. . 均匀SAT .1.2 SAT 随机模型 .315 13 4 6. 在几何图中量化几何 .. 我们的贡献.

2 预备知识 9

2.1 随机 SAT 模型 .92.2 可满足性阈值.11

3 实验设置 13

3.1 随机 SAT 实例的生成.133.1.1 几何实例的高效生成.143.2 SAT 实例基准测试.15 321 15. . 证明大小 .

4 实证结果 17

417 41 2 . 维数的影响 .. 可满足性阈值 .18 43 21. 求解器运行时间.4.4 证明大小.23424 441 42 27. . 对证明大小的影响 .. . 证明大小与求解器运行时间的关系 .4.5 推广到其他实例规模.30

5 结论与展望 33

参考文献 35

1 引言

给定一个布尔公式，布尔可满足性问题（SAT）要问的是：是否存在一种对该公式变量的赋值，使公式的求值结果为真。它是首个被证明为 NP完全的问题 [Coo71, Lev73]，并且常用于理论计算机科学中，通过归约证明其他问题的 NP困难性。值得注意的是，计算机科学中最著名的未解问题之一，就是是否存在一种确定性算法，能在多项式时间内求解任意 NP完全问题，例如 SAT。目前尚未发现此类算法（而且人们认为它并不存在），因此 SAT 问题通常被认为难以求解。尽管如此，SAT求解算法往往能够快速找到某些实例的解。因此，SAT问题已在形式验证 [Bie+03] 和芯片设计 [Heu+24, TN 17], 等领域得到实际应用，这些领域会产生称为 工业 SAT 实例的实例。

为了弥合实践中易解实例与理论困难性之间的差距，当前研究的一种方法是找出某些 SAT 实例所固有的、使其易于求解的性质。为此，研究者提出了一些模型，用以模拟工业SAT实例的行为。接下来，我们将先介绍可满足性，再概述随机模型方面的相关研究，最后陈述我们的贡献。

1.1 关于可满足性的背景

为了定义 SAT公式，考虑 n 布尔变量 x_1、…、x_n。这些变量中的每一个都可能以其否定形式出现，记作 x_i。我们将可能被否定的变量称为一个文字。就我们目前的讨论而言，多个这样的文字以析取方式组合成一个子句，例如 $x_3 \vee x_1 \vee x_4$。SAT公式由若干这样的子句构成，这些子句以合取方式连接；例如：

$$(x_3 \vee x_1 \vee \overline{x_4}) \wedge (x_2 \vee x_3) \wedge (\overline{x_2} \vee x_3 \vee \overline{x_1} \vee x_4) \wedge (\overline{x_2} \vee \overline{x_4} \vee \overline{x_5})$$

要满足一个 SAT公式，必须使每个子句中至少有一个文字的取值为真。例如，将$x_3 = \text{true}$ 和 $x_1 = x_2 = x_4 = \text{false}$设为真即可满足上述公式。

《可满足性手册 [Bie+09] 概述了 SAT 的历史，追溯其根源至古希腊的逻辑学。不过，我们不妨从20世纪70年代说起，当时 Cook 和 Levin 独立证明了 SAT 是 NP完全问题[Coo71, Lev73]。在这方面，一个值得关注的例子是2-SAT，其公式中的子句始终恰好包含两个文字：已证明它可以在多项式时间内确定性地求解。从历史上看，研究越来越多地聚焦于由定长子句构成的这类实例[FM09, p. 36]。通常子句长度较短，其中3-SAT 尤其受到关注，因为它是 SAT 达到 NP完全所需的最短子句长度。

1.1.1 均匀SAT

如前所述，SAT 已出现了各种模型。给定一组参数，这些模型会生成随机 SAT 实例，可用于评估参数化 SAT 公式的平均行为，并衡量 SAT 求解器的性能。其中最简单的模型或许是均匀*SAT*。给定变量数、子句数和子句长度，均匀SAT模型会从所有指定规模的实例中均匀随机地抽取一个实例。尽管均匀模型十分简单，却能生成一些最难的实例。

当子句长度和变量数固定、而子句数变化时，可以观察到均匀 SAT 实例的行为随之发生变化。与变量数相比，子句较少的实例往往可满足，而子句较多的实例往往不可满足[HAB09]。对此的解释很简单：子句较少的实例约束不足，也就是说，满足条件的变量赋值有很多。另一方面，增加子句数会减少可能的变量赋值数量，最终导致实例不可满足。

这两个极值的存在表明，均匀 SAT 实例必定会经历从可满足到不可满足的转变。这种*SAT* 相变一直是 SAT 社区广泛关注的话题。存在一个所谓的可满足性阈值，SAT 实例会在此处从可满足转变为不可满足[Ach09]。在子句长度固定时，人们观察到阈值位置与实例规模无关；它似乎只取决于子句数与变量数之比，这一比值称为公式的密度。

SAT 中一些最有趣的结果与可满足性阈值有关。首先，求解难度似乎与阈值相吻合。最难求解的均匀实例出现在阈值处，而使用回溯来求解 SAT 实例的 DPLL算法，已被证明需要指数级的步数才能求解这类实例[FP83]。相较之下，低于或高于阈值的实例通常更容易求解。这种现象被称为 SAT 的“易—难—易”模式[HAB09]。需要注意的是，这一术语并未完整描述实际情况：阈值以下的实例似乎始终都很容易求解，到了阈值处难度便骤然上升；而阈值以上的难度似乎只是逐渐降低。[MSL92] 对此“易—难—易”模式给出了一种解释：阈值处的实例之所以难，是因为它们既未受到约束不足，也未受到约束过度，因此只有少数满足赋值，或者根本不可满足，而这很难判定。

需要注意的是，上述内容主要基于经验观察。关于阈值，目前尚无定论，例如其确切位置。对于 2-SAT，已知其阈值恰好位于密度 1 [CR92], ，而对于更长的子句长度，目前只有经验估计 [MMZ06]。

人们也仅仅猜想可满足性阈值是尖锐的 [Ach09]。如果阈值以下的每个实例渐近几乎必然[1]可满足，而阈值以上的每个实例渐近几乎必然不可满足，我们就称该阈值是尖锐的。 Friedgut [Fri99] 证明了尖锐阈值的存在性，但该阈值可能取决于变量的数量。最近，Ding 等人 [DSS22] 证明了在固定密度下可满足性阈值存在，并且对于较长的子句长度，该阈值是尖锐的。

1.2 SAT 随机模型

遗憾的是，均匀SAT模型无法充分再现真实工业实例的特性。例如，其中的实例太难求解；而在真实实例中，一个变量出现的子句数量通常并非均匀分布。因此，人们提出了不同的模型，以更好地模拟工业实例的特性。这些模型生成随机的参数化SAT实例，从而能够基于这些细粒度参数分析SAT求解器的性能。这有助于

1 渐近几乎必然（*a.a.s.*）指随着变量数量趋于无穷大，概率趋于 1

更深入地理解SAT，并提升求解器性能。与工业实例相比，参数化模型的优势在于无需依赖具有固定特性的现成实例 [ABL09a]。随机生成的实例常与工业实例和人工构造的实例一同用作基准测试，参与SAT竞赛。SAT竞赛自2002年起每年举办，SAT求解器在其中相互角逐[SAT25]。

Ansótegui 等人[ABL09b] 观察到，在许多真实世界SAT实例中，变量出现频次遵循幂律分布，即少数变量出现在许多子句中，而许多变量仅出现在少数子句中。这一性质被称为异质性。它已被纳入无标度 *SAT*模型[ABL09a]，该模型包含一个用于控制异质性的参数。然而，实证分析表明，SAT 求解器在无标度实例上的表现仍然不如在某些大型工业实例上[ABL09a, BFS19]。此外，Bläsius 等人[Blä+23] 已表明，从幂律模型中抽取的不可满足实例很可能需要超多项式时间才能求解。因此，无标度模型似乎并不能充分刻画真实世界SAT的性质。

人们还观察到，现实世界中的实例具有局部性，这意味着共同出现在同一子句中的变量很可能还会共享其他子句。这无法用幂律分布来建模，因为幂律的影响仅限于变量的出现频率，而不会影响它们之间的关系。相反，一种自然的选择是借助底层几何结构来对局部性进行建模，这一做法最早由Bradonjić和Perkins [BP14]采用。该方法已被Giráldez-Cru和Levy [GL17]应用于一种很有前景的SAT 模型：该模型将实例的子句和变量放置在几何底空间上，并依据它们之间的距离进行关联，从而引入几何概念。这样，在底层空间上彼此更接近的变量更有可能共享相同的子句。此外，还可以为变量分配遵循幂律分布的权重。因此，该几何SAT模型既能刻画异质幂律分布，也能刻画局部性。人们认为，这种混合模型能够充分刻画从易到难的SAT 实例谱系。事实上，Bläsius 等人[Blä+23] 使用了 [GL17] 中所引入几何模型的一个推广版本，证明几何实例渐近几乎必然易于求解。然而，他们假设几何底空间具有常数维数。我们认为，维数的影响本身值得进一步研究，因为维度很可能影响可满足性和难度等性质。我们将在下一节讨论这一点。

1.3 量化几何图中的几何

SAT 模型并非底层几何结构及其维度受到重点关注的唯一研究领域。关于几何之影响的类似观察也出现在随机图中，这促使我们开展关于几何 SAT 的研究。

一种经典的随机图模型是 Erdős–Rényi 图 [ER59], ，其中任意两个顶点之间的每条边都以某个固定概率独立于其他边而存在。研究表明，Erdős–Rényi图的各种性质也会发生相变，这与均匀SAT中的相变十分相似。例如，根据顶点数量的不同，存在一个边概率阈值，超过该阈值时，图几乎必然形成一个巨型连通分量；而低于该阈值时，几乎必然不会形成[ER60]。

与 SAT 一样，人们也提出了能够生成具有各种性质的随机图的模型。与真实世界SAT实例类似，现实世界图网络中顶点的度通常服从幂律分布。这可以用非齐次随机图（独立随机图）[BJR07], 来表示，它是 Chung-Lu 图的推广[ACL01]。与无标度 SAT 类似，独立随机图通过幂律分布对顶点度进行参数化。

人们还观察到，现实世界图网络具有局部性的概念。以朋友图为例：一个人的两位朋友彼此也很可能是朋友。从形式上讲，这被称为聚类系数，它衡量一个顶点的两个邻居彼此相邻的概率。这一性质被随机几何图（RGGs）[DD23,Pen03], 所纳入——随机几何图等价于几何 SAT 模型：它们将顶点置于几何底空间中，并根据某种度量绘制边。因此，底层空间中彼此距离较近的顶点更有可能相连。

最后，几何非齐次随机图（GIRGs）由Bringmann 等人[BKL19], 提出，同时涵盖了独立随机图和随机几何图。它们以几何底空间为基础，此外还可以为顶点分配服从幂律分布的权重，正如可以为几何SAT模型中的变量分配权重一样。

与 SAT 相比，人们目前对这些图模型的性质了解得深入得多，尤其是对两种几何模型——随机几何图和几何独立随机图——中维数影响的了解。首先，已有证明表明，在某些条件下，随着维数趋于无穷，几何图会失去其几何属性，并且它们

会收敛到一种非几何模型。具体而言，当顶点数量足够大时，使用球面空间的随机几何图已被证明会随着维数趋于无穷而收敛到 Erdős–Rényi图 [DD23，定理 1.(i)]。此外，随着维数增加，几何独立随机图会失去其几何属性；因此，如果将环面用作底层空间，它们已被证明会收敛到独立随机图 [Fri+24，定理 1。1]。 Friedrich 等人[Fri+24] 还表明，随机几何图要收敛到其对应的非几何模型，必须具有齐次底空间，而球面和环面都满足这一条件。例如，超立方体就不是齐次的。由于存在边，超立方体上顶点之间的距离并不相互独立。因此，超立方体上的随机几何图确实会收敛到某种非几何模型，但其中的边不会变得相互独立。

如图所示，维数对几何图模型有很大影响。尽管几何图与几何 SAT 具有许多共同属性，但目前尚无针对几何 SAT 模型的可比研究，其维数对模型行为的影响在很大程度上仍是未知的。

1.4 我们的贡献

我们对维数对几何 SAT 的影响进行了实证分析。我们从Bläsius 等人 [Blä+23]所使用的几何SAT模型中抽取随机 SAT 实例。该模型是对Giráldez-Cru和Levy [GL17]提出的模型的推广。为了分析维数在低维和高维情况下的影响，我们抽取了维度最高达 1000 的实例。同时，我们还从均匀SAT模型中抽取实例进行比较。

为了更好地理解底层底层空间中的维数如何影响 SAT 实例，尤其是 SAT 求解器的行为，我们在生成的实例上运行不同的 SAT 求解器，并记录每个实例的度量指标（详见第3章）。基于这些度量指标，我们考察以下属性：

可满足性 我们分析可满足实例的分布、可满足性阈值的位置及其陡峭程度。

求解器运行时间 对于每个求解器，我们分析其求解实例所花费的时间，并考察几何实例在多大程度上呈现出均匀实例中已知的“易-难-易”模式。

证明大小 一些 SAT 求解器能够为不可满足实例输出不可满足性证明；详细说明请参见 第3.2.1节。我们分析这些证明的大小，并依据一个非常宽泛的经验法则：证明越大，实例就越难求解。

回想一下，随着维度趋于无穷，随机几何图会收敛到均匀非几何图。鉴于随机几何图模型与几何 SAT 的相似之处，我们猜想几何 SAT 也具有同样的性质，具体如下。

▶ **猜想 1.1**：**随机几何 SAT 的收敛**。 随着维度趋于无穷，几何SAT模型会收敛到均匀SAT模型，高维下的实例将无法区分。 ◀

事实上，我们通过经验发现，在上述所有性质中，几何 SAT 实例都会随着维度增加而逐渐趋近于均匀情形。

此外，基于几何实例渐近几乎必然易于求解这一理论结果 [Blä+23]（参见 第4.1节，其中给出了我们的推理），我们猜想低维下的几何 SAT 实例更容易处理，也更有可能不可满足，具体如下。

▶ **猜想 1.2**：**低维随机几何 SAT.** 在低维度下，几何SAT模型中的实例容易处理得多，它们需要更小的证明，并且与均匀SAT相比，在更低的密度处出现可满足性阈值。 ◀

我们的发现再次支持了这一猜想。我们观察到，在不超过6个维度时，SAT 求解器在几何实例上并不表现出“易-难-易”模式。相反，它们能够在求解同等规模均匀实例所需时间的一小部分内求解所有实例。在更高维度下，实例确实会在可满足性阈值处出现不那么明显的难度峰值，并缓慢收敛到均匀 SAT的相应表现。

我们发现，低维几何实例的可满足性阈值出现在低得多的密度下；但在维度约为 10 时，其密度已接近（但并不完全等同于）均匀实例的密度。我们认为，这是因为几何 SAT 实例中的局部性会导致底层空间中彼此距离较近的变量之间产生矛盾（关于这一点的推理，另请参见第4.1节）。此外，我们发现，低维情况下的可满足性阈值要宽泛得多。

在无标度 SAT 模型 [BFS19], 中也观察到了可满足性阈值的偏移及其宽泛程度，但总体而言，这些特征不如在我们的几何模型中明显。这支持了当前的观点：几何 SAT 能够表示范围更广的工业SAT实例。

比较维数对难度和可满足性的影响后，我们发现，阈值的收敛速度快于求解器运行时间。尽管阈值很快就达到类均匀密度，但在 250 维几何实例与均匀实例之间，求解器运行时间仍存在明显差异。

请注意，[Blä+23] 关于几何 SAT 难度的理论结果是基于实例的证明大小得出的。因此，我们也分析了证明大小。我们发现，低维几何实例确实需要小得多的证明。人们通常会认为求解器运行时间至少在一定程度上与证明大小相关；对于均匀实例而言，情况确实如此。然而，对于维度不超过 6 的几何实例，我们发现证明大小与求解器运行时间并不相关。总体而言，我们发现，从 6 维到 7 维的变化是一个转折点：几何实例从此开始逐渐表现出已知的均匀实例特性。

接下来，我们将介绍第2章中的关键概念，然后第3章中对实验设置的描述。我们将在第4章中陈述我们的结果。

2 预备知识

本章将正式定义关键概念。我们首先定义 SAT问题，这是我们研究的核心。SAT问题有多种变体；我们的研究采用 k-SAT，其中（a）子句长度始终为 k，且（b）公式以合取范式（CNF）表示，即由通过合取连接的子句组成。

▸ **定义 2.1**：**k-SAT**。设 x_1、…、x_n 是 n 布尔变量，其值可以为真或假。k-SAT 公式 Φ 是由 m 子句$c_1 \wedge \cdots \wedge c_m$合取而成。每个子句都是若干$k$ 文字 $l_1 \vee \cdots \vee l_k$的析取。文字表示变量之一，也可以是其否定形式，记作 x_i。

我们称 Φ 可满足，当且仅当存在一个对 x_1、…、x_n 的赋值，使得 Φ 的值为真；否则，我们称其为 不可满足。 ◂

我们将 m/n（子句数与变量数之比）称为 SAT 公式的**密度**。

2.1 随机 SAT 模型

我们介绍用于生成随机 SAT 实例的两种模型。我们的生成过程始终会生成一个具有固定 k、m 和 n 的实例 Φ。

首先，我们定义均匀模型，其中 SAT 实例是从所有可能的公式中均匀随机抽取的。

▸ **定义 2.2**：**随机均匀 k-SAT**。设 $\mathscr{U}_k(n, m)$ 是一个随机均匀k-SAT 公式，含有 n 个变量和 m 个子句。我们将公式的 m 个子句彼此独立地抽取。每个子句中包含的 k 个变量均从所有可能的 $\binom{n}{k}$ 个变量中均匀随机且不重复地抽取。随后，每个文字所对应的变量均以概率 $1/2$ 被独立取反。 ◂

其次，我们生成具有底层几何结构的 SAT 实例。因此，该模型还采用一个额外的固定参数$d \in \mathbb{N}^+$ 来指定维度。该模型取自[Blä+23]。

在定义模型本身之前，我们先定义提供底层几何结构的几何底空间。请回想一下，Friedrich 等人[Fri+24] 已经证明，几何独立随机图要收敛到边相互独立的均匀图模型，需要一个齐次底空间。因此，为了让我们的几何SAT模型具有相同的行为，我们选择环面作为底层空间，因为它是齐次的。

▶ **定义 2.3**：**几何底空间**。令 $\mathbb{T}^d$ 为 d 维环面。这样的环面是一个 d 维超立方体 $[0,1]^d$，其中相对的边界被视为同一边界，这意味着坐标 0 与坐标 1 相同。也就是说，我们可以设定坐标 $[0,1)^d$。

为了定义环面上的欧几里得距离，令 p、$q \in [0,1)^d$ 为环面上的两个点。我们将 p 和 q 的第 i 个坐标之间的圆周差定义为

$$|p_i - q_i|_\circ = \min\big(|p_i - q_i|, 1 - |p_i - q_i|\big).$$

于是，p 与 q 之间的欧几里得距离为

$$|p - q| = \sqrt{\sum_{i\in[d]} |p_i - q_i|_\circ^2}.$$ ◀

我们使用这个 d 维环面的定义来生成几何 SAT 实例，如下所示。

▶ **定义2.4**：**随机几何 k-SAT**。如果环面 $\mathbb{T}^d$ 上的点 $p \in [0,1)^d$ 的每个坐标都是从 $[0,1)$ 中均匀随机抽取的，我们就称该点是随机的。

要生成几何 SAT 实例，我们为 m 个子句各随机采样一个点，并为 n 个变量各采样一个点。然后，为每个子句分配 k 个欧几里得距离最近的变量。每个文字中的变量都以 $1/2$ 的概率独立地随机取反。◀

图 2.1展示了如何生成这样的实例。

请注意，我们只关注分析不采用幂律分布的几何 SAT 模型的性质。因此，我们省略了[Blä+23]中使用的变量权重和温度。此外，选择否定文字时，我们不会刻意避免重复选择同一个子句（参见第 3.1.1 节）。

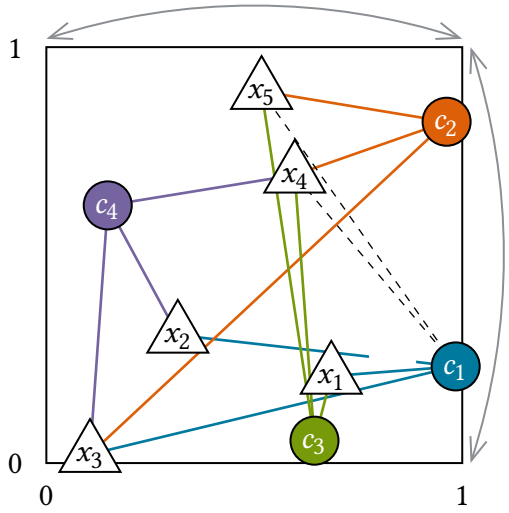


图 2.1：二维几何实例的底层几何结构示例。请注意，环面被展平成一个二维正方形；箭头表示相互识别的边界。此实例包含 $n = 5$ 个变量 x_1、…、x_5 和$m = 4$ 个子句 c_1、…、c_4。连接到每个子句的三条线表示其$k = 3$ 个最近变量。此外，还显示了子句 c_1 到全部五个变量的距离。

2.2 可满足性阈值

如果某事件发生的概率在 $n \to \infty$ 趋近于 1，我们就称该事件**渐近几乎必然**（渐近几乎必然）发生。

如上所述，人们观察到，随着密度增加，均匀实例会经历尖锐相变：从渐近几乎必然可满足转变为渐近几乎必然不可满足。基于[Ach09]，这一现象被形式化为以下猜想。

▶ **猜想 2.5**：**可满足性阈值猜想**。设 $\mathscr{U}_k(n,m)$ 为一个包含 $k \geq 3$的随机均匀 SAT 实例。那么，存在一个常数密度$r_k > 0$，使得对于每个满足 $r \neq r_k$的 $r > 0$，

$$\lim_{n\to\infty} \Pr[\mathscr{U}_k(n, r \cdot n) \text{ is satisfiable}] = \begin{cases} 1, & \text{if } r < r_k \\ 0, & \text{if } r > r_k. \end{cases}$$ ◀

r_k 是阈值所在的位置，我们也将其称为**临界密度**。

3 实验设置

本章将介绍我们为取得研究结果所采用的实验设置。为分析随机 SAT 实例，我们分两步进行了实验：

1. 生成随机 SAT 实例。
2. 使用 SAT 求解器求解这些实例，并记录每个实例的各项度量。

第一步中，我们开发了一个定制生成工具，并在第 3.1 节中对其进行了介绍。请注意，我们可以使用不同的 SAT 求解器执行第二步，从而顺带获得有关求解器性能的结果，并比较不同类型的求解器。我们将在第 3.2 节中介绍相应的实验设置。

我们选择 Rust 编程语言 [Rus] 来实现生成和基准测试。它使我们能够编写高级代码，同时兼顾高性能、内存安全和易读性，从而确保代码的正确性（尤其是生成代码的正确性）易于验证。

3.1 随机 SAT 实例的生成

我们开发了一个命令行界面，可轻松生成均匀和几何 SAT 实例。输入包括要生成的实例类型、一组参数组合，以及每种组合要生成的实例数量。如定义 2.2（随机均匀 k-SAT）和定义 2.4（随机几何 k-SAT）所述，这两种实例类型都以子句长度k、变量数量n和子句数量m 作为参数。此外，几何生成还需要指定维度d。该工具为每个实例输出一个文件，采用广为人知的 DIMACS 格式；这种格式以文本文件形式存储合取范式中的实例，并且尤其用于SAT 竞赛[SAT11]。

为加快生成过程，我们采用了若干优化。请注意，对于均匀实例，子句彼此独立地生成。因此，我们可以按子句高效地写入输出，而无需在子句之间保留状态。遗憾的是，对于几何实例，这并不那么容易，下面将对此进行讨论。

3.1.1 几何实例的高效生成

对于几何实例，如第 2.1 节所述，我们不会避免多次生成同一个子句，因为这需要存储所有先前生成的子句，从而大幅增加计算时间和空间。避免子句重复可简化理论分析[Blä+23]，但由于不同子句的数量随n而增加，因此对实证分析的影响微乎其微。

因此，与均匀实例一样，几何实例生成不需要在生成子句之间保留状态。不过，生成几何实例确实需要以下初始设置步骤。

回想一下，根据定义2.4（随机几何 k-SAT），几何实例是通过在 d维环面上放置点来生成的。令点$v_1, \ldots, v_n \in [0,1)^d$ 分别表示n 个变量，并令$c_1, \ldots, c_m \in [0,1)^d$ 表示m 个子句。对于每个c_i，现在我们需要找到k 最近的变量点。

一种朴素的方法是分别计算每一对c_i, v_j 之间的圆周差。假设计算d维欧几里得距离需要线性时间[2]（相对于d），那么这项计算的运行时间为$\mathcal{O}(mnd)$。接着，对于每个子句c_i，我们按到c_i 的距离对所有变量点排序，并选出最近的$k \leq n$个，因此总运行时间为$\mathcal{O}(mnd + mn \log n)$。我们可以利用 Introselect 来改进$k$近邻选择[Mus97]。它可以在线性时间内选出数组中第k小的元素，从而也能通过比较在线性时间内找到k 个最小元素。这样总运行时间就会降至$\mathcal{O}(mnd)$。

然而，我们转而可以通过使用k维树来大幅提升性能，如[Ben75]中所介绍的那样。它非常适合求解所有最近邻居问题，平均运行时间为$\mathcal{O}(\log n)$，插入与搜索皆然。[3] 经过一个初始设置步骤，将n 个变量点插入树中，该步骤耗时为$\mathcal{O}(n \log n)$，因此为每个子句确定k 个最近变量的平均运行时间变为$\mathcal{O}(n \log n + mk \log n)$。

实际上，这意味着生成大型几何实例的扩展性要好得多。例如，我们测得生成一个 a$k = 3, n = m = 10^5, d = 3$ 实例的速度提升了 200 倍，在我们的硬件上耗时约 350 ms 。

[2] 我们务实地预期，在真实硬件上对浮点数求平方耗时为常数。[3] 请注意，[Ben75] 假设维度为常数，且其平均搜索运行时间是经验性的。

3.2 SAT 实例基准测试

在第二步中，我们让 SAT 求解器求解之前生成的实例，并记录每个实例的以下度量：

- 该实例被判定为可满足还是不可满足，
- 求解该实例所花费的时间，
- 证明大小（参见 第3.2.1节）。

我们选择的实例规模较小，可在 15 分钟的超时时限内求解。所有求解器均在一台 AMD EPYC 7742 处理器上运行，运行位置为 2.25 GHz。

我们为实验选择了三种 SAT 求解器（见 表 3.1）。`minisat`作为一种简单且非常流行的 SAT 求解器，因其历史意义而入选，它曾赢得 SAT 竞赛 2005年。其次，我们选择了`glucose`，它基于`minisat`，但加入了自身的功能。对我们的工作而言，最值得注意的是，它支持输出不可满足性证明。最后，我们选择了`kissat` 作为当前最先进的求解器。`kissat` 以及相关求解器 `CaDiCal`自2020 [SAT25]以来，在所有 SAT 竞赛的主赛道中均名列前茅。

这三种求解器都是完备的冲突驱动子句学习（CDCL）求解器，也就是说，它们既能求解不可满足实例，也能求解可满足实例。我们没有采用不完备的随机局部搜索（SLS）求解器；这类求解器在可满足性阈值附近效率更高，但只适用于可满足实例，因为我们的重点是几何实例，不论其可满足性如何。

3.2.1 证明大小

SAT 求解器找到的满足赋值可以轻易验证，但不可满足实例则需要采用更复杂的验证方法。4 有些

表 3.1：我们实验中使用的 SAT 求解器。

求解器	Year	版本	参考文献	证明支持
minisat	2010	2.2.0	[SE05]	no
glucose	2018	4.2.1	[AS09]	yes
kissat	2025	4.0.2	[Bie+24]	yes

4 否则，不可满足实例就无法获得任何满足[Rol65]。

SAT 求解器能够输出不可满足性证明，通常采用 DRAT 格式，这是 SAT 竞赛主赛道所要求的[Heu+24, SAT23]。

DRAT 格式的证明由一系列子句组成。每个子句要么添加到原始 SAT 实例中，要么从中删除。其思路是以这样的方式修改 SAT 实例：在每一步中，修改后的实例至少与原始实例一样可满足（就满足变量赋值而言）[BCH21]。删除子句总是显然满足这一性质，而添加子句则不得违反这一性质。

SAT 求解器在尝试求解实例时，会学习可添加到原始实例中的子句。事实上，较早期的求解器尤其如此，它们会直接将学习到的子句作为证明输出[SAT23]。现代 SAT 求解器还会指示删除无关子句，以加快验证速度。

应用所有子句添加和删除操作后，证明验证器可以使用单元传播检查证明是否有效；单元传播利用从单文字子句中获得的信息来删除和简化子句。假设修改后的实例包含一个仅由文字 x组成的子句。那么，为了满足该公式，$x = \text{true}$ 必须成立。因此，任何包含 x 的子句都已满足，可以删除（单文字子句本身除外）；而任何出现的 x都可以移除，因为它们不再有助于可满足性。如果移除x 后子句变为空子句，该子句便无法满足，因此可以证明该实例不可满足。为启动这一过程，证明中总会包含一个单文字子句。

在我们的实验中，我们记录了所有不可满足性证明的以下度量：

- 证明子句总数，
  - 其中添加或删除的子句数，
- 所有子句中的文字总数，
  - 其中出现在添加或删除的子句中的文字数，以及
- 任一证明子句的最大长度。

我们的分析表明，添加操作和删除操作的数量大多与子句或文字的总数成正比。同样，文字总数也随子句总数而变化。因此，我们后续的分析仅针对证明子句总数以及任意子句的最大长度。

4 实证结果

本章将展示我们的结果。这些结果是通过生成随机实例、使用三种 SAT 求解器求解这些实例，并将实例的各种属性可视化得到的。

我们生成了固定 $n = 300$ 个变量的 3-SAT 实例，这大约是最难求解的均匀实例仍可行的最大实例规模。我们将子句数量 m 从 300 变为 3 000，以达到密度 $m/n \in [1, 10]$，涵盖均匀实例和几何实例的重要密度。几何实例是在维度 $d \in [1, 50]$ 下生成的，此外还在最高达 $d = 1\,000$ 的特定更高维度下生成。对于每种不同的参数组合，我们生成了 100 个实例，以便对记录的指标取平均值。

我们还通过生成少量其他规模的实例，考察了我们的结果是否适用于子句长度或变量数量不同的实例。我们将在末尾的第 4.5 节讨论这一点。

4.1 维数的影响

我们将先阐述猜想，然后介绍我们的结果。回顾引言中的以下猜想。

▸ **猜想 1.1**：**随机几何 SAT 的收敛**。随着维度趋于无穷，几何 SAT 模型将收敛于均匀 SAT 模型，并且在高维度下，实例之间无法区分。◂

▸ **猜想 1.2**：**低维随机几何 SAT**。在低维度下，几何SAT模型中的实例更容易处理，所需的证明更短；与均匀SAT相比，它们在更低的密度下就会出现可满足性阈值。◂

为了理解为什么我们认为几何实例更容易求解，回顾一下Bläsius 等人[Blä+23]已证明，几何实例渐近几乎必然易于求解。具体来说，他们证明，当 n 趋于无穷时，一个

几何实例渐近几乎必然包含一个大小为常数的不可满足子公式，可在 $\mathcal{O}(n \log n)$ 时间内找到。

还要记得，密度较低的均匀实例很可能可满足，而密度较高的实例很可能不可满足。处于中间阈值附近的实例很难求解，因为它们既没有约束不足，也没有约束过度[MSL92]。几何实例中普遍存在的局部性可能会影响子句数量使实例变得约束过度的速度，因为在环面上彼此距离较近的子句可能共享部分或全部变量，从而形成不可满足子结构。因此，我们预计几何实例的可满足性阈值会出现在较低的密度下。此外，正如[Blä+23]所发现的那样，小型子结构直接意味着几何实例所需的证明大小更小。我们还预计求解器能更快地找到此类证明，因为它们能更高效地找到较小的子结构。

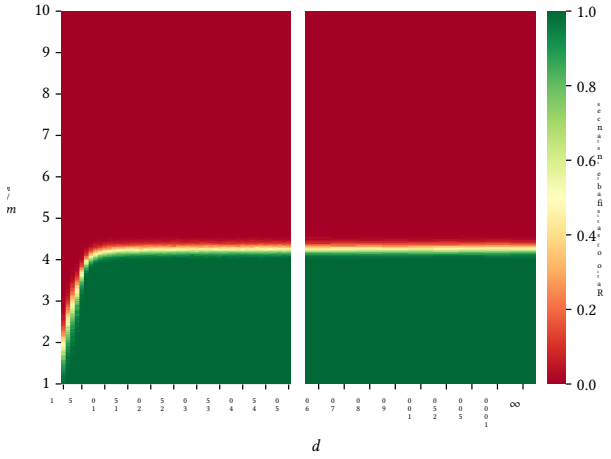
图4.2概述了我们的结果：图中展示了可满足性阈值和求解器运行时间随维度的变化。显然，临界密度和求解器运行时间都会随维度增加。此外，在$d = 1\,000$ 维度下，几何实例与均匀实例在可满足性和求解器运行时间方面都没有明显差异。后续许多图表展示的是相同数据，但侧重点不同，粒度也更细。

接下来的章节中，我们将深入探讨所分析的三项性质。

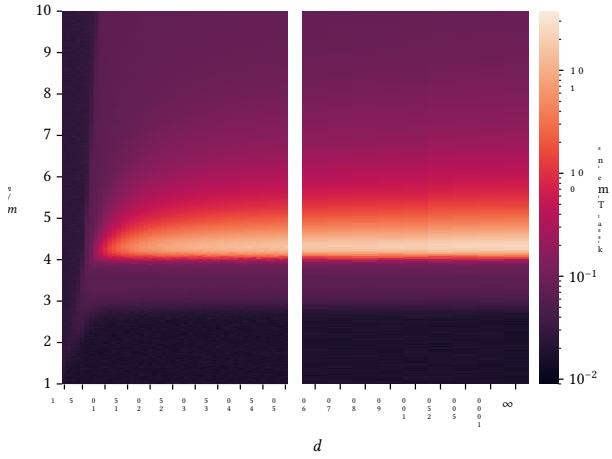
4.2 可满足性阈值

对于均匀 SAT 实例，[MMZ06] 的实证结果表明，临界密度位于 $r_3 \approx 4.267$。从 图4.3可以看出，低维几何实例的可满足性阈值通常出现在低得多的密度下，在 $m/n = 2$ 时低于 $d = 1$。临界密度随维度增加而升高，并很快达到 r_3，其中 $m/n \approx 4.1$ 时为 $d = 7$。这与我们的猜想一致：由于局部性会诱发较小的不可满足子结构，低维几何实例更有可能不可满足。

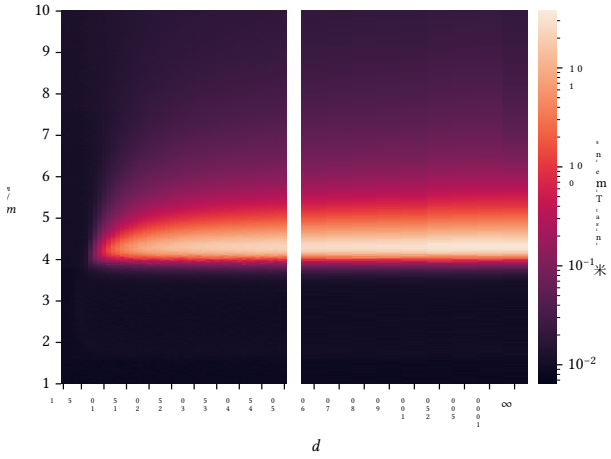
此外，在低维度下，阈值似乎更加粗略，这在无标度 SAT [BFS19] 中也有观察到（参见 第1.4节）。当 $d = 1$时，密度为 $m/n = 1.0$ 的实例大多可满足，而直到 $m/n = 3.0$才大多不可满足。相比之下，当 $d = 7$时，可满足实例的比例会在密度差约为 0.6的范围内更快下降。



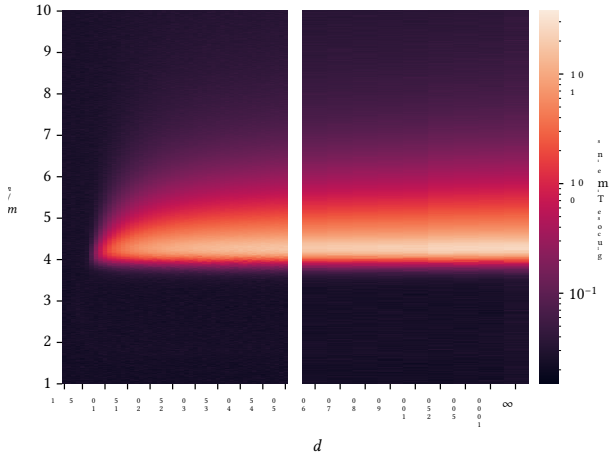
(a) 可满足实例比例。
红色区域仅包含不可满足实例，
而绿色区域仅包含可满足实例。



(b) 按kissat求解的平均时间。
注意到额外的平台期
约为10^{-1} s 求解器运行时间。



(c) 按minisat求解的平均时间。



(d) 按glucose求解的平均时间。

图4.2：显示可满足实例比例(a)和平均求解器运行时间的热图(b)、(c)、(d)。

纵轴上，每个图都显示密度 m/n。

横轴表示维度。每个图的左侧区域显示所有维度 $d \in [1, 50]$，右侧区域显示非等距维度$d \in \{60, 70, ..., 100, 250, 500, 1\,000\}$。此外，图末还显示了均匀实例的数值，以便进行比较（“∞”）。

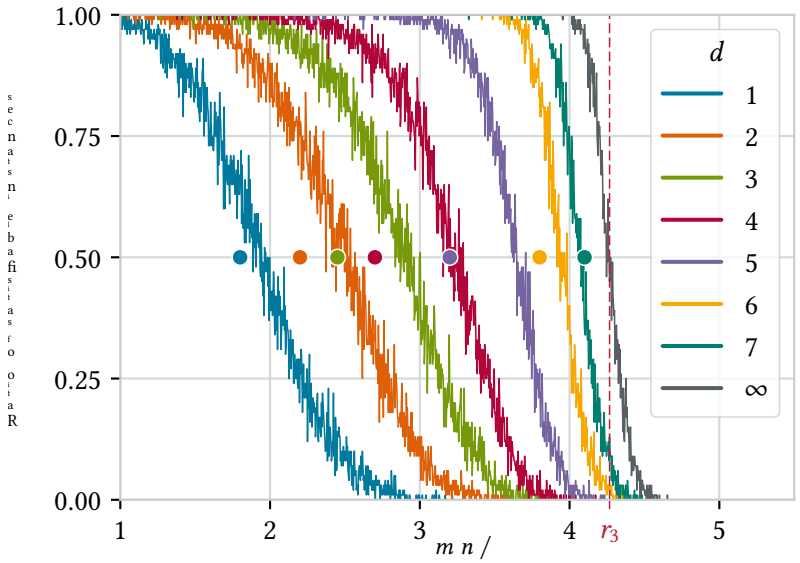


图4.3：低维度下可满足几何实例的比例随密度的变化$d \in [1,7]$。此外，为了进行比较，还展示了可满足均匀实例的比例（“∞”），正如预期，该比例在 1/2 时达到 $r_3 \approx 4.267$ [MMZ06]。

对于每个维度，圆圈表示kissat的求解时间中所观察到的平台期达到最大值时的密度（见 图4.4）。圆圈位于比率1/2 处，因为如果平台期与可满足实例比例有关，则预计最大值应出现在比率 1/2处。

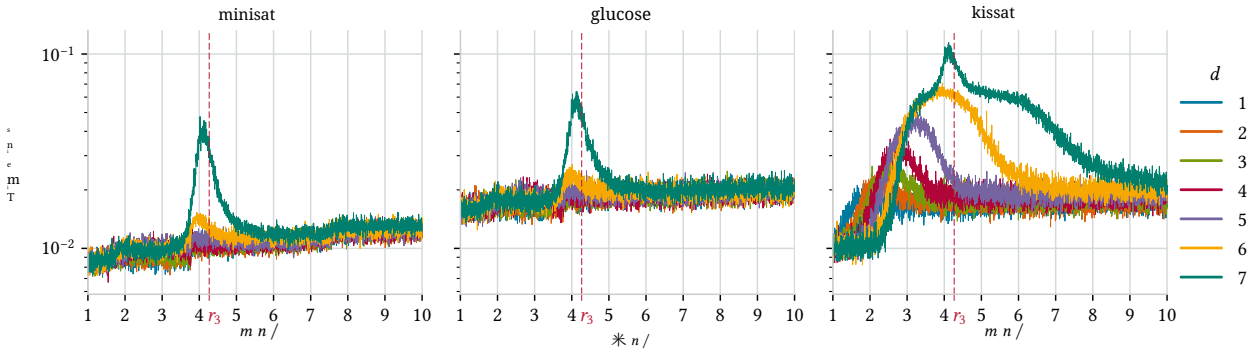


图4.4：低维度下，密度不超过 $d = 7$ 时的平均求解器运行时间。在 $d = 7$处，所有求解器的求解时间都在均匀可满足性阈值 $r_3 \approx 4.267$略低处出现显著峰值。

与另外两个求解器不同，kissat 的求解时间呈现出平台期，且这一现象在每个维度下都会出现。

4.3 求解器运行时间

请回想一下，均匀实例呈现所谓的“易-难-易”模式：对于固定的子句长度k和n 个变量，在低密度下它们很容易求解；随后在可满足性阈值处，求解难度会骤然上升，而在更高密度下又会逐渐变得更容易求解。这一点在我们称之为“日落图”的图4.2中显而易见：图中展示了三种求解器的运行时间。在更高维度下，求解器运行时间会在$r_3 \approx 4.267$ 附近骤然增加，之后只缓慢下降。总体而言，这三种求解器的表现非常相似。

为了进行更细致的分析，图4.4专门展示了低维度下的求解器运行时间。在这些低维度下，`minisat` 速度最快，其次是`kissat` 和`glucose`。相比之下，图4.5展示了均匀实例的求解器运行时间：在均匀可满足性阈值以下和以上，`minisat` 仍然表现最佳，而在阈值附近最困难的实例上，`glucose` 和`kissat` 表现更好。`kissat` 不仅在所有均匀实例上速度最快，而且

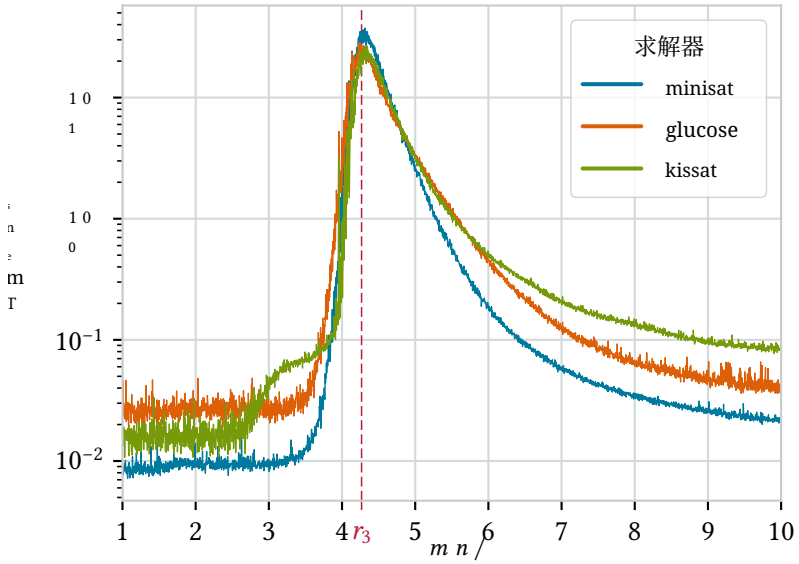


图4.5：均匀实例的平均求解器运行时间随密度的变化。如预期所示，求解器在临界密度 $r_3 \approx 4.267$附近表现最差。在临界密度处，`kissat` margin- ally 表现最佳，其次是`glucose` ，最后是`minisat`。请注意，因此，`kissat` 在所有均匀实例上的总体求解速度最快，尽管对数时间轴可能会让人得出相反的结论。

总的来说，它求解我们生成的所有实例所用时间也最短。尽管在较高密度下，kissat 的表现不如glucose 和minisat，在较低密度下的表现也不如minisat，但仍然如此。对于均匀实例或高维实例的可满足性阈值，kissat 尤其快速。一种可能的解释是，kissat作为最新的求解器，针对临界密度附近的困难实例进行了高度优化，代价是求解较简单的实例时会稍微多花一些时间。

有趣的是，低维几何实例似乎并未呈现“易-难-易”模式；所有实例似乎都相当容易求解。这一点在图4.2中可以观察到，在图4.4中则展现得更为详细。显然，密度对glucose的求解器运行时间影响不大，直到$d = 6$。对于minisat，情况似乎也类似：求解器运行时间确实会随着密度略有增加，但第一个明显的峰值直到$d = 6$才出现。minisat 和glucose 的求解器运行时间都要到$d = 7$才会出现较大的峰值。$d = 7$ 处的峰值出现在3-SAT临界密度略低的位置。如图4.2所示，求解器运行时间的峰值与临界密度的变化相对应；维度越高，峰值出现时的密度也越高。

同样，kissat 在 $d = 7$处呈现出众所周知的“易-难-易”峰值。然而，与glucose 和minisat相反，kissat的求解时间表现出一个很大的

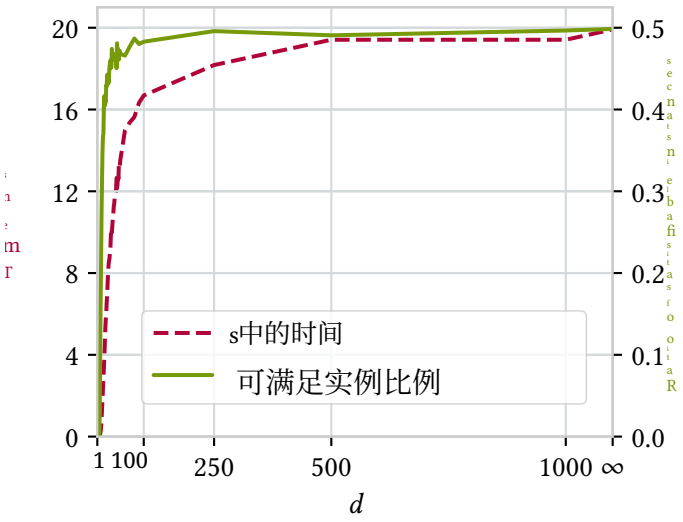


图 4.6：均匀 SAT 阈值下不同维度的求解器运行时间与可满足率对比。每个数值均为密度为 $m/n \in [4.20, 4.33]$ 的所有实例的平均值。

平台期，它似乎独立于可满足性阈值而持续存在。无论维数如何，该平台期均存在，如图4.2 (b)所示。对于高维和均匀实例，kissat的求解时间急剧增加于$m/n \approx 3$。在低维度下，该平台期出现在较低密度处，而求解时间在较高密度处再次下降。为了考察其与通常观察到的“易-难-易”模式之间可能的关系，我们将kissat的平台期达到最大值时的密度添加到图4.3。如果该平台期与可满足性有关，我们会预期其最大值与可满足性阈值相关。可以看出，它与该阈值之间似乎并无直接联系，这表明kissat 呈现出另一种与可满足性阈值无关的“易-难-易”模式。我们目前尚无法解释这一观察结果。

在图 4.6中，我们通过考察临界 3-SAT 密度r_3附近的实例，展示维度增加时求解器运行时间和可满足率的变化。在$d = 100$时，几何实例与均匀实例的求解器运行时间仍存在明显差异。这与可满足率形成鲜明对比：可满足率很快就达到类似均匀情形的数值。显然，尽管可满足率在大约$d = 100$时达到接近均匀情形下可满足率的数值，求解器运行时间却要到大约$d = 500$时才达到。这一点尤其值得注意，因为这表明存在这样的 SAT 实例：其可满足性阈值表现得与均匀情形相似，但求解起来仍然更容易。

4.4 证明大小

对于每个实例，我们还记录了glucose和kissat提供的证明相关指标，详见第 3.2 节。这为了解真实求解器行为提供了宝贵的参考。

请注意，[Blä+23] 所做的工作基于几何 SAT 证明大小的一个上界。他们证明，在较大的n下，渐近几乎必然存在一个至少包含2^k 个子句的子公式，且这些子句都包含相同的变量。由于文字是随机取反的，这很可能导致公式不可满足。由于该子公式大小固定，可以很快找到证明，因此这些实例易于处理。因此，分析真实求解器证明可以强化这一结果，或许还能为较小的n提供证据。

另请注意，记录求解器证明大小衡量的是求解器实际所需的证明大小，而非理论上界。求解器优化

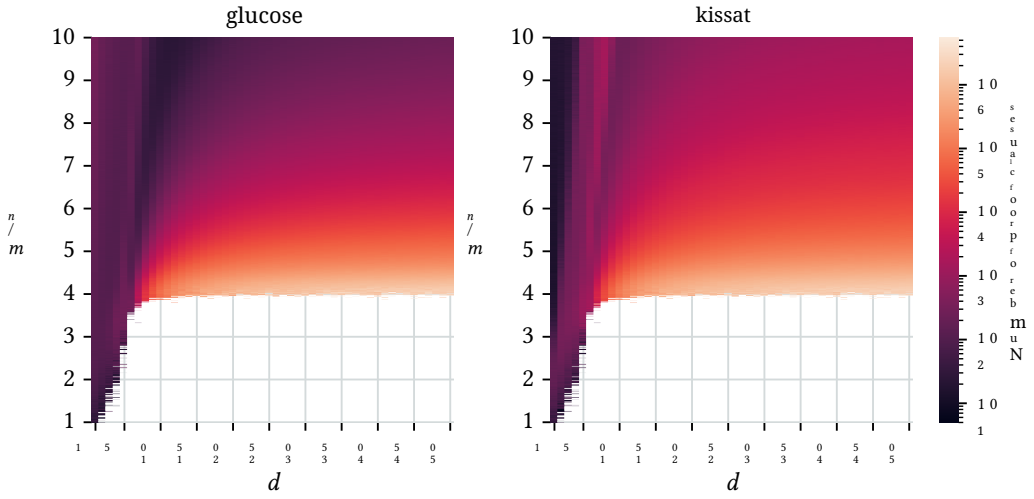
通常侧重于求解速度，而非在合理范围之外追求证明大小（例如 SAT 竞赛对证明检查设置的超时）。因此，不应使用证明大小来比较求解器。

我们的研究结果表明，几何实例确实需要更小的证明，下一节将对此进行讨论。除此之外，低维证明与均匀实例的证明在求解器运行时间方面表现迥异，相关内容将在第4.4.2节中讨论。

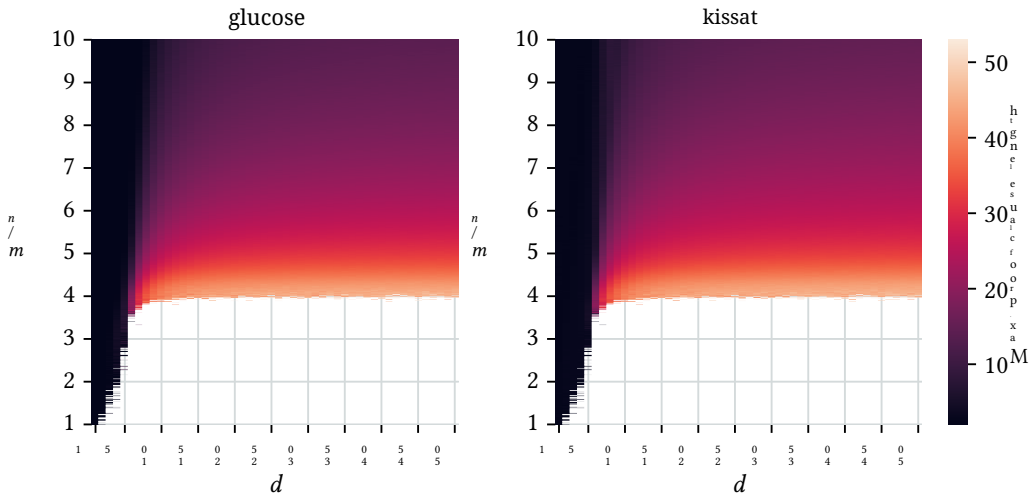
4.4.1 对证明大小的影响

图4.7展示了衡量证明大小的两个指标：证明子句的数量和证明子句的最大长度。请注意，我们在实际应用中得到的证明子句数量大于或等于著名的理论归结规模，后者衡量任何不可满足性证明所需的最少子句数。同样，实际应用中证明子句的最大长度大于或等于归结宽度，后者衡量任何证明所需的子句最大长度的最小值。

显而易见，低维实例所需的证明相对较小。直到 $d = 6$，`kissat` 和 `glucose`的证明都具有相对较小的证明子句最大长度，平均不到10个文字。在两者之中，`kissat` 在低维度下输出的子句尤其小，每个子句只有两个或三个文字，见 $d = 1$ (图4.9)。这并不令人意外，因为大多数低维实例中的矛盾只涉及两个或三个变量，因此求解器证明相关部分中也只包含这些变量。代码清单4.8展示了这样一个实际求解器证明的例子。显然，只需证明两个变量之间存在矛盾即可；然而，`kissat` 会输出更多包含无关变量的子句。对于其他一些实例，`kissat` 会立即找到一个简单的矛盾，并且只输出一个少于10个子句的小型证明。我们认为，这是因为`kissat` 未能立即找到反驳，在这种情况下，它还会输出迄今为止学到的所有子句。不出所料，由于几何局部性导致子句极为相似，在 $d = 1$，包含两个反驳变量的21个子句中有19个也包含同一个第三变量。由于每个子句中的文字都以均匀随机的方式取反，因此实例很可能包含变量相同但取反方式不同的子句。在密度足够高时，也就是说，如果存在



(a) 证明所包含的添加或删除子句的平均数量。



(b) 证明中任意子句的平均最大长度。

图4.7：各求解器的证明大小的两种度量。每幅图展示了证明大小度量随维度（至 $d = 50$ ）和密度的变化。在低密度的空白区域中，所有实例均可满足，因此没有证明。

请注意，由于热图显示的是平均值，因此少数需要较大证明的实例所包含的信息会被忽略；详情请参阅图4.9。

$$
\begin{aligned}
&x_{28} \vee x_{42} \vee x_{281} \\
&\overline{x_{28}} \vee \overline{x_{42}} \vee x_{281} \\
&x_{28} \vee x_{276} \vee x_{281} \quad (\times 3) \\
&x_{28} \vee x_{276} \vee \overline{x_{281}} \\
&x_{28} \vee \overline{x_{276}} \vee x_{281} \quad (\times 3) \\
&x_{28} \vee \overline{x_{276}} \vee \overline{x_{281}} \quad (\times 5) \\
&\overline{x_{28}} \vee x_{276} \vee x_{281} \\
&\overline{x_{28}} \vee x_{276} \vee \overline{x_{281}} \\
&\overline{x_{28}} \vee \overline{x_{276}} \vee x_{281} \\
&\overline{x_{28}} \vee \overline{x_{276}} \vee \overline{x_{281}} \quad (\times 4)
\end{aligned}
$$

(a) 部分实例子句摘录。所有包含任一反驳变量 x_{28} 或 x_{281} 的子句均已显示，其中一些出现了多次。

$$
\begin{aligned}
&x_{281} \vee x_{28} \\
&x_{281} \vee \overline{x_{28}} \\
&\overline{x_{281}} \vee x_{28} \\
&\overline{x_{281}} \vee \overline{x_{28}} \\
&x_{28}
\end{aligned}
$$

(b) 最后五个子句（均为添加操作），出自 kissat的证明。该证明共包含53个子句，但只有这最后五个子句引用了反驳变量x_{28} 和x_{281}。

代码清单4.8：实例摘录(a)以及kissat (b)对该实例不可满足性证明的摘录。该实例的参数为$k = 3$， $n = 300$， $m = 1\,702$， $d = 1$。

由于几何局部性的影响，不存在仅包含反驳子句x_{28} 和x_{281}之一的子句；它们总是同时出现。此外，这两个反驳变量总是伴随着另外两个变量之一：x_{42} 或x_{276}，其中x_{276} 出现在21个子句中的19个里。

虽然 kissat 的证明总共由 53 个子句组成（包括添加操作和删除操作），但这里展示的最后几行中，只有最后几行引用了 x_{28} 或 x_{281}，这表明 kissat 在找到反驳之前就输出了它学到的内容。

存在足够多的子句，这几乎必然会导致冲突。这与以下理论研究结果一致：Bläsius 等人 [Blä+23]。

在均匀可满足性阈值处，困难实例所需的证明尤其大，但只有在超过 $d = 6$ 时才如此。这与求解器运行时间的增长非常相似，而且似乎与我们的观察吻合：求解器运行时间首次大幅增长发生在 $d = 7$。与求解器运行时间一样，证明大小也呈现出“难—易”模式：一达到可满足性阈值，不可满足实例就需要非常大的证明，而随着密度进一步升高，证明大小才逐渐减小。甚至可以说，kissat 再次呈现出“第二阈值”：在维度不超过 $d = 6$ 时，kissat 在较低密度下需要更多证明子句，

而在一个随维度成比例上移的阈值之后，证明子句的数量则较少。相比之下，glucose的平均证明子句数在低维度下相当稳定。

4.4.2 证明大小与求解器运行时间

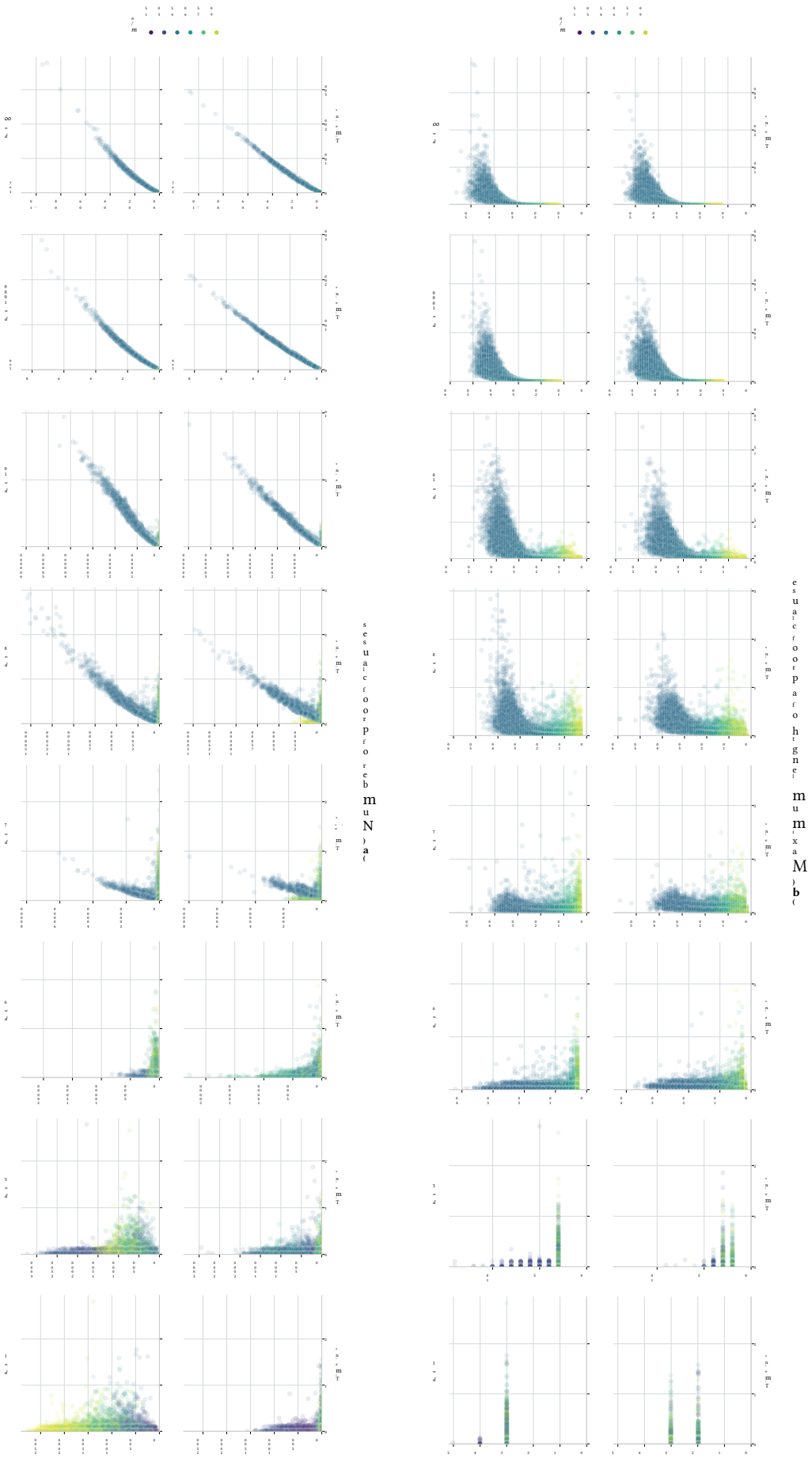
较大的证明与更困难的实例相关联，这种假设是合理的，因为找到较大的证明也需要求解器付出更多工作。事实上，有些求解器会直接将其学习子句作为证明输出 [SAT23]，；在这种情况下，预计证明大小至少与搜索树深度相关。为了比较求解器运行时间与密度，我们将选定维度下的所有实例绘制成散点图，横纵轴分别表示求解器运行时间和证明大小。结果见 图4.9。该图提供了有关求解器行为和几何实例的多方面信息。我们先考察求解器行为，再讨论几何实例。

与 图4.7类似， kissat 和 glucose 在最大证明大小方面表现非常相似，但在证明子句数量上有所不同。在较高维度下，两者的求解器运行时间与证明大小都呈现出同样强的相关性，这符合预期。在较低维度下， kissat 再次表现出色，为大多数实例生成较小的证明，而 glucose总体上需要更长的证明。对于这两种求解器，耗时最长的实例并非证明最大的实例。相反，glucose 在证明大小介于其最小值与最大值之间时耗时最长。另一方面，kissat 有少数实例出现了反直觉的情况：它们所需的证明非常小，但求解却耗时很长。对于大多数实例，kissat 要么输出更大的证明，要么耗时稍长。我们怀疑，这可能是由于 kissat针对特定困难实例所做的优化，这与我们在 kissat 求解均匀实例时观察到的求解时间表现类似。 kissat 能够迅速为大多数实例找到证明，即使证明大小较大；但少数实例则需要更长时间。随着维度增加，两种求解器的行为越来越相似，运行时间与证明大小呈相关性。

如前所述，就证明子句的最大长度而言，两个求解器的表现相似。正如我们在上一节中讨论的那样，证明在低维度下包含非常小的子句是意料之中的。在 $d = 1$， kissat 的证明子句最多由三个文字组成，这可能与其求解的是 3-SAT 实例有关。与此同时， glucose 也提供了非常小的证明

子句，最多包含五个文字。在 $d = 6$ 之前，两个求解器的证明都需要非常小的子句，因此耗时稍长；或者需要较大的子句，但求解时间较短。在更高维度下，两个求解器开始输出更长的证明子句。

考察各个实例后，我们发现似乎存在一个从$d = 6$ 到 $d = 7$的转变。在 $d = 6$之前，所有密度的实例都呈现出上述“要么证明规模大，要么求解时间长”的模式，但不同求解器下，实例按密度的分布各不相同。然而，实例似乎从$d = 6$ 到 $d = 7$ 开始“转变”，其证明大小也开始与求解器运行时间相关。这对于 $r_3 \approx 4.267$附近的实例尤其明显：它们通常需要更长时间才能求解，而这似乎与证明大小有关。正如预期的那样，随着维度增加，图表越来越像均匀实例的图表；在 $d = 1000$ 处的实例与均匀实例几乎无法区分。所有这些观察结果都与我们的发现一致：直到 $d = 6$ 的实例都没有呈现众所周知的“易-难-易”模式。这表明，就不可满足性证明而言，低维几何实例的表现也与均匀实例大不相同。



4.5 推广到其他实例规模

迄今为止，我们只考察了变量数固定的3-SAT 实例$n = 300$。为了研究我们的结果在多大程度上可以推广到其他实例规模，我们生成了低维几何实例，其参数值为$k \in \{2, 3, 4\}$ 且$n,\ m \in [20, 1\,000]$。我们确定了这些实例的临界密度，结果如图4.10所示。可以看出，我们的结果似乎确实适用于其他实例规模。无论变量数和子句长度如何，临界密度都会趋近于均匀SAT模型的临界密度，并随维度增加而略微上升。

需要注意的是，随着变量数增加，临界密度也会发生变化。这在变量数少于100的极小实例中尤为明显。此后，这种变化看起来只是次线性的。这是意料之中的，因为均匀实例也能观察到相同的效应。对于2-SAT，这种现象尤为显著：尽管已证明临界2-SAT密度恰好为$r_2 = 1$ [CR92], ，但我们的均匀实例的临界密度似乎更高，并且会随着变量数增加而下降，这是渐近

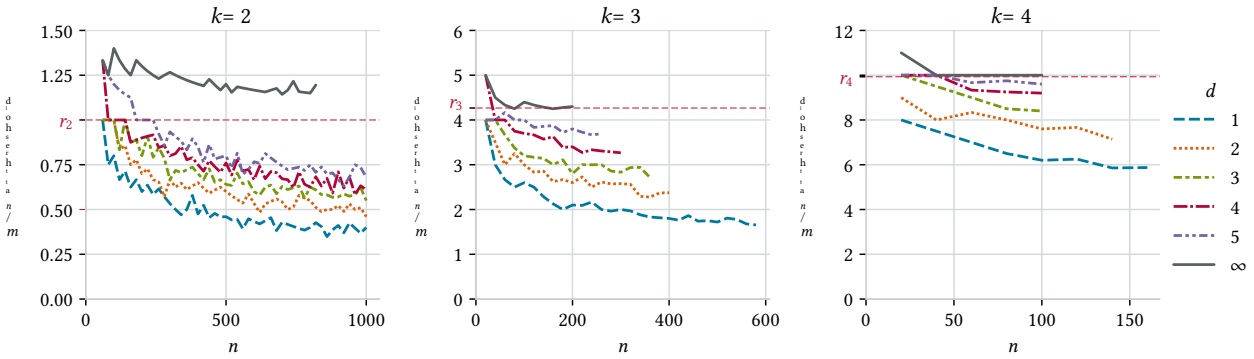


图4.10：对于子句长度$k \in \{2, 3, 4\}$，每个图都显示在不同变量数量m/n下，$1/2$ 不可满足实例达到的临界密度n。我们展示了维度$d \in [1, 5]$ 下的结果，并以均匀实例作为比较（“∞”）。水平线表示均匀SAT的临界密度：$r_1 = 1$ [CR92], 和$r_3 \approx 4.267$, $r_4 \approx 9.931$ 正如[MMZ06]所猜想的那样。

这是通过以 20 为步长生成实例（$n, m \in [20, 1\,000]$），并针对每个 m, n 生成 50 个实例，然后对每个 n 找出可满足实例比例最接近 $1/2$ 时的密度得出的。

需要大得多的实例才能观察到这些效应。在另一项实验中，我们观察到，只有当实例达到 $n \approx 10^7$ 时，观测到的临界密度才会与预期的 r_2 相差不超过 1 %（这些实例仍然很容易处理，因为 2-SAT 可以在多项式时间内求解）。

除此之外，值得注意的是，在维度固定时，几何实例的临界密度会随子句长度增加。这与均匀 k-SAT 的临界密度相符，后者也会随着子句长度增加而提高。

5 结论与展望

我们对Giráldez-Cru和Levy [GL17]提出的几何SAT模型进行了实证分析。我们发现，维数会影响几何实例的重要性质。随着维数增加，几何模型中的实例似乎会逐渐趋近于均匀SAT模型中的实例；高维实例与均匀实例在实际中几乎无法区分。与均匀模型相比，低维几何实例似乎更容易处理，因此SAT求解器所需的证明也更小。此外，由于几何诱导的局部性，几何实例更有可能不可满足。这些观察支持了Bläsius 等人 [Blä+23] 提出的理论结果，即几何实例渐近几乎必然易于求解。

与不可满足概率更高这一情况相符，低维实例的临界密度较低。低维度下的可满足性阈值似乎也更加模糊。

我们发现，几何实例的行为在六维到七维左右发生了转变。维度不超过六维的几何实例并未表现出均匀实例的典型性质：求解器运行时间受密度影响不大，它们没有呈现“易-难-易”模式，证明大小也与求解器运行时间不相关。另一方面，从七维开始，实例的行为越来越像均匀实例：它们的求解器运行时间在可满足性阈值处达到峰值，且求解器运行时间与证明大小相关；不过，这些效应仍不如超高维或均匀实例中那么明显。还需要进一步研究，才能详细理解这种从六维到七维的转变缘由。对实例的子句—变量图进行统计分析和可视化，或许会对此有所帮助。

还需要进一步研究 SAT 求解器，才能解释我们观察到的一些现象，例如 `kissat` 的求解器运行时间出现平台期，以及 `kissat` 在均匀阈值处尤其快，但在其他密度和低维度下比 `minisat` 慢。找出 SAT 求解器中导致这些现象的设计选择，最终或许能推动进一步改进。

总体而言，我们的研究结果表明，几何为 SAT 提供了一种颇具前景的模型。还需要通过分析提供证据，进一步验证这一结论。

工业实例。几何 SAT 模型能够利用局部性再现许多真实实例中的网络结构，并且通过改变维度，可以生成从简单到困难的各种实例。结合变量权重的幂律分布，几何 SAT 模型有望表示范围广泛的工业SAT实例。

参考文献

[ABL09a] C. Ansótegui、M. L. Bonet 和 J. Levy，“面向类工业随机SAT实例”，载于第二十一届国际人工智能联合会议论文集，2009年7月。[在线]。获取地址：https://www.ijcai.org/ Proceedings/09/Papers/072.pdf[ABL09b] C. Ansótegui、M. L. Bonet 和 J. Levy，“工业SAT实例的结构”，载于约束规划原理与实践——*CP 2009*，I. P. Gent 编，柏林、海德堡：施普林格，2009年，第127–141页。DOI：10.1007/978-3-642-04244-7 13._[Ach09] D. Achlioptas，“随机可满足性”，可满足性手册，载于《人工智能与应用前沿》，第185卷。IOS出版社，2009年。[ACL01] W. Aiello、F. Chung 和 L. Lu，“幂律图的随机图模型”，实验数学，第10卷，第1期，第53–66页，2001年1月，DOI：10.1080/10586458.2001.10504428。[AS09] G. Audemard 和 L. Simon，“预测现代 SAT 求解器中学习子句的质量”，载于第二十一届国际人工智能联合会议论文集，2009年7月。[在线]。获取地址：https://www.ijcai.org/ Proceedings/09/Papers/074.pdf[BCH21] S. Baek、M. Carneiro 和 M. J. H. Heule，“用于 SAT求解器与细化器通信的灵活证明格式”，载于系统构建与分析的工具与算法，J. F. Groote 和 K. G. Larsen 编，沙姆：施普林格国际出版公司，2021年，第59–75页。DOI：10.1007/978-3-030-72016-2_4.[Ben75] J. L. Bentley，“用于关联检索的多维二叉搜索树”，《*ACM*通讯》，第18卷，第9期，第509–517页，1975年9月，DOI：10.1145/361002.361007。[BFS19] T. Bläsius、T. Friedrich 和 A. M. Sutton，“相变处无标度3-SAT的经验时间复杂度”，载于系统构建与分析的工具与算法，T. Vojnar 和 L. Zhang 编，沙姆：施普林格国际出版公司，2019年，第117–134页。DOI：10.1007/978-3-030-17462-0_7.[Bie+03] A. Biere、A. Cimatti、E. M. Clarke、O. Strichman 和 Y. Zhu，“有界模型检验”，计算机进展，第58卷，第117–148页，2003年，DOI：10.1184/R1/6603944.V1。

[Bie+09] A. Biere、M. Heule、H. van Maaren与T. Walsh编，可满足性手册，载于《人工智能与应用前沿》，第185卷。IOS出版社，2009年，ISBN：978-1-58603-929-5[Bie+24] A. Biere、T. Faller、K. Fazekas、M. Fleury、N. Froleyks与F. Pollitt，《CaDiCaL、Gimsatul、IsaSAT和Kissat参加2024年SAT竞赛》，载于*2024年SAT竞赛论文集：求解器与基准测试描述*，M. Heule、M. Iser、M. Järvisalo与M. Suda编，计算机科学系报告系列B。赫尔辛基大学，2024年，第8–10页。[BJR07] B. Bollobás、S. Janson与O. Riordan，《非齐次随机图中的相变》，随机结构与算法，第31卷，第1期，第3–122页，2007年，DOI：10.1002/rsa.20168。[BKL19] K. Bringmann、R. Keusch与J. Lengler，《几何非齐次随机图》，理论计算机科学，第760卷，第35–54页，2019年2月，DOI：10.1016/j.tcs.2018.08.014。[Blä+23] T. Bläsius、T. Friedrich、A. Göbel、J. Levy与R. Rothenberger，《异质性与几何对随机可满足性证明复杂度的影响》，随机结构与算法，第63卷，第4期，第885–941页，2023年，DOI：10.1002/rsa.21168。[BP14] M. Bradonjić与W. Perkins，《论随机几何图中的尖锐阈值》，*LIPIcs，第28卷，APPROX/RANDOM 2014*，第28卷，第500–514页，2014年，DOI：10.4230/LIPIcs.APPROX-RANDOM.2014.500。[Coo71] S. A. Cook，《定理证明过程的复杂度》，载于第三届*ACM*计算理论研讨会年会论文集，STOC '71。美国纽约州纽约市：计算机协会，1971年5月，第151–158页。DOI：10.1145/800157.805047。[CR92] V. Chvatal与B. Reed，《Mick有所收获（胜算在他这边）（可满足性）》，载于第*33*届计算机科学基础研讨会年会论文集，1992年10月，第620–627页。DOI：10.1109/SFCS.1992.267789。[DD23] Q. Duchemin与Y. De Castro，《随机几何图：若干最新进展与展望》，载于高维概率*IX*，R. Adamczak、N. Gozlan、K. Lounici与M. Madiman编，沙姆：施普林格国际出版公司，2023年，第347–392页。DOI：10.1007/978-3-031-26979-0_14。[DSS22] J. Ding、A. Sly与N. Sun，《大k情形下可满足性猜想的证明》，数学年刊，第196卷，第1期，第1–388页，2022年7月，DOI：10.4007/ annals.2022.196.1.1。[ER59] P. Erdős与A. Rényi，《论随机图I》，德布勒森数学刊物，第6卷，第3–4期，第290–297页，1959年，DOI：10.5486/PMD.1959.6.3-4.12。

[ER60] P. Erdős与A. Rényi，《随机图的演化》，匈牙利科学院数学研究所出版物，第5卷，第17–61页，1960年。[FM09] J. Franco与J. Martin，《可满足性的历史》，可满足性手册，载于《人工智能与应用前沿》，第185卷。IOS出版社，第3–74页，2009年。[FP83] J. Franco与M. Paull，《求解可满足性问题的Davis-Putnam过程的概率分析》，离散应用数学，第5卷，第1期，第77–87页，1983年1月，DOI：10.1016/0166-218X(83)90017-3。[Fri+24] T. Friedrich、A. Göbel、M. Katzmann与L. Schiller，《高维几何非齐次随机图中的团》，SIAM离散数学杂志，第38卷，第2期，第1943–2000页，2024年6月，DOI：10.1137/23M157394X。[Fri99] E. Friedgut，《图性质的尖锐阈值与 *k*-Sat问题》，美国数学会杂志，第12卷，第4期，第1017–1054页，1999年，DOI：10.1090/S0894-0347-99-00305-7。[GL17] J. Giráldez-Cru与J. Levy，《随机 SAT 实例中的局部性》，载于第26届国际人工智能联合会议论文集，澳大利亚墨尔本：国际人工智能联合会议组织，2017年8月，第638–644页。DOI：10.24963/ijcai.2017/89。[HAB09] K. Henry、S. Ashish与S. Bart，《不完备算法》，可满足性手册，载于《人工智能与应用前沿》，第185卷。IOS出版社，2009年。DOI：10.3233/978-1-58603-929-5-185。[Heu+24] M. J. H. Heule、M. Iser、M. Järvisalo与M. Suda编，《求解器、基准测试与证明检查器描述》，2024年SAT竞赛论文集，2024年，[在线]。可获取：http://hdl.handle.net/10138/584822[Lev73] L. A. Levin，《通用序贯搜索问题》，信息传输问题，第9卷，第3期，第115–116页，1973年。[MMZ06] S. Mertens、M. Mézard与R. Zecchina，《由腔方法得到的随机K-SAT阈值》，随机结构与算法，第28卷，第3期，第340–373页，2006年，DOI：10.1002/rsa.20090。[MSL92] D. G. Mitchell、B. Selman与H. J. Levesque，《SAT问题的困难与简单分布》，载于第10届全国人工智能会议论文集，美国加利福尼亚州圣何塞，1992年7月12–16日，W. R. Swartout编，AAAI出版社／麻省理工学院出版社，1992年，第459–465页。[在线]。可获取：http://www.aaai.org/ Library/AAAI/1992/aaai92-071.php[Mus97] D. R. Musser，《内省排序与选择算法》，软件：实践与经验，第27卷，第8期，第983–993页，1997年8月，DOI：10.1002/ (SICI)1097-024X(199708)27:8<983::AID-SPE117>3.0.CO;2-%23。

[Pen03] M. Penrose, 《随机几何图》，载于《牛津概率研究丛书》。牛津：牛津大学出版社，2003年。[Rus] 《Rust 编程语言》。[在线]。网址：https://www.rust-lang.org/[SAT11] SAT 竞赛，《基准测试提交指南》。访问日期：2025年6月12日。[在线]。网址：https://web.archive.org/web/20230927183712/http://www.satcompetition.org/2011/format-benchmarks2011.html[SAT23] SAT 竞赛，《已认证的不可满足》。访问日期：2025年8月6日。[在线]。网址：https://satcompetition.github.io/2023/certificates.html[SAT25] SAT 竞赛，《国际 SAT 竞赛网页》。访问日期：2025年4月28日。[在线]。网址：https://satcompetition.github.io/[SE05] N. Sorensson 和 N. Een，《MiniSat v1.13——一种带冲突子句最小化功能的 SAT 求解器》，2005, [在线]。网址：http://minisat.se/downloads/MiniSat_v1.13_short.pdf[TN17] P. Taatizadeh 和 N. Nicolici，《一种用于设计片上位翻转检测器的自动化 SAT 方法》，载于*2017* 年 *IEEE/ACM* 国际计算机辅助设计会议（*ICCAD*），2017 年 11 月，第 101–108 页。DOI：10.1109/ ICCAD.2017.8203766。