arXiv:2602.16335v1 [cs.LO] 18 Feb 2026

全称量词与未解释函数符号的归纳可满足性证明

Stefan Ratschan[1],Anggha Nugraha[1,2], Mikol´a˘s Janota[3], 和 Marek Dan˘co[3]

[1] 捷克科学院计算机科学研究所，布拉格，捷克 stefan.ratschan@cs.cas.cz[2] 布拉格捷克理工大学信息技术学院，布拉格，捷克[3] 布拉格捷克理工大学捷克信息学、机器人与控制论研究所，捷克

摘要。 未解释函数符号与全称量词的组合出现在许多自动推理应用中，例如，它们能够用于推理数组。然而，这类公式的可满足性通常是不可判定的。在实践中，SMT 求解器往往借助启发式方法成功处理不可满足的情形。但在可满足的情形下，它们依赖显式构造模型；对于最小模型也不够小的公式，这种方法会失效。我们提出了一种替代方法，通过归纳论证证明可满足性，并将其应用于线性整数算术。由此得到的算法能够证明当前 SMT 求解器无法处理的公式是可满足的。

1 引言

当前 SMT 求解器主要用于证明不可满足性。因此，它们检测不属于已知可判定类别的公式的可满足性的能力有限。然而，检测可满足性的能力对于识别反例十分重要，而这在许多应用中不可或缺 [11]。

本文针对包含全称量词和未解释函数符号的公式，探讨了这一问题。我们主要关注与线性算术的组合，对应于不可判定的 SMTLIB 逻辑 UFLIA [12,17]。然而，此类公式出现在重要的应用中，例如程序验证；在程序验证中，未解释函数符号在数组推理方面发挥着重要作用 [6]。当前的 SMT 求解器主要依靠启发式方法 [28] 来证明此类公式的不可满足性。为了证明可满足性，它们会显式构造一个模型 [15,32]。对于像 $f(0) = 0 \wedge \forall x \,.\, f(x+1) = f(x) + 1$ 这样简单的公式，这种方法会失效，因为它们不存在有限模型；而对于最小模型虽为有限模型但规模很大的公式，这种方法的代价也很高。

本文探讨了一种替代方法：我们不显式构造模型，而是通过归纳论证证明可满足性。更具体地说，我们

– 提出可满足性证书这一概念，用于表示此类公式可满足性的归纳论证，– 提出一种算法，通过为一类 UFLIA 公式综合此类可满足性证书来证明其可满足性，– 证明在输入公式满足某个支持归纳推理的语义条件这一假设下，该算法是可靠且完备的，– 提供一种 SMT 编码，用于表示该算法处理的公式，并–通过计算实验表明，我们的方法能够证明当前 SMT求解器无法处理的公式的可满足性。

2 相关工作

我们的工作涉及自动推理的多个领域：基于模型的量词实例化、数组的判定过程、综合以及证明证书。

基于模型的方法和实例化方法。 Reynolds 等人 [30,32] 提出了针对量化公式的有限模型发现技术。该方法要求用户确保可容许性，并且可能引入许多分类讨论（“合并引理”）以满足基数约束。Ge 和 de Moura [15] 提出了基于模型的量词实例化（MBQI），并指出：“对于某些包含数组索引偏移的情况，我们的过程会产生无限多个实例，而线性整数算术的判定过程则会终止。如何在基于实例化的方法中对此类公式进行推理，仍是一个有待未来研究解决的开放问题。”我们的工作试图解决这一开放问题。

数组和量化公式的判定过程。Bradley 等人 [7]提出了一种用于数组性质的判定过程，但明确禁止对全称量化变量进行算术运算。Habermehl 等人 [16] 允许带有差值约束的形式 $a[i+c]$，但仍局限于偏移有界索引。Alberti 等人 [2] 的研究仅限于含有一个全称量化变量的平坦公式。这排除了诸如 $a[i+1]$ 这样的项，而这正是本文关注的重点。Elad 等人 [13] 证明了一个不含算术的一阶片段的可判定性，但该片段无法表达我们的约束。

综合。（程序）综合已得到广泛研究。相关技术通常依赖候选项的枚举，例如反例引导的方法（反例引导归纳综合 [33]），其中一些方法支持递归 [14,20]。语法引导综合（SyGuS） [3] 提供了一个统一框架：给定一个上下文无关文法和一个逻辑规约，找到一个符合要求的表达式。其性能在很大程度上取决于所选文法。语法引导综合已得到广泛研究

在可满足性模理论（SMT）的背景下 [31,1,29]。模型查找可以表述为一个综合问题 [27]。

此外，也有人通过为待综合对象提供模板 来解决综合问题，这种方法也用于模型查找 [10,13,8]。

人们还对满足单次调用性质的规格进行了深入研究，该性质意味着待综合函数始终应用于完全相同的参数列表。这意味着，实际上可以对该函数进行去斯科伦化，并调整量词消去过程来判定见证 [21,22,19]。在此背景下，Hozzov´a 等人 [18]利用归纳证明综合递归函数。我们的方法也适用于单次调用片段之外的情况，例如，$\forall x.\, f(x) + f(x) > x$ 可以去斯科伦化，但 $\forall x.\, f(x) + f(x+1) > 1$ 则不能。

证明证书和模型表示。 反驳证明和模型的表示与认证，一直是自动推理领域长期存在且仍在持续面临的挑战。近期，TPTP [34]和可满足性模理论 [4,5,23,26] 领域都在应对技术和理论方面的挑战 [9]（请注意，一阶逻辑公式的解释并非递归可枚举）。在某些理论中（例如，包含超越函数符号 [24]的实数理论），证明可满足性也可能是一个非平凡的理论特定问题。

3 问题陈述

我们介绍一种用于证明带有未解释函数符号的 SMT 公式可满足性的方法，其中这些函数符号的定义域是与某个具有可数规范模型的理论相关联的排序。在该理论为线性整数算术的情况下，所得公式属于 SMT-LIB 理论 UFLIA。我们考虑的公式形式为

$$F \wedge \forall \bar{x}.\, Q \tag{1}$$

其中，F 和 Q是无量词的，$\forall \bar{x}$ 表示对任意多个变量进行全称量化，并且所有未解释函数符号的参数均不包含元数非零的未解释函数符号（即参数中只能包含未解释常量）。

基于可满足性证明的一般方法，我们将在以下附加假设下设计一种用于检测 UFLIA 中可满足性的算法：

– 所有未解释函数符号的最大元数为 1。– 量化变量的数量为 1。– 对于量化部分 Q 中任意给定的未解释函数符号，该量化变量在其所有参数项中的系数都相同。

在满足能够进行归纳的附加语义条件（定义6）时，算法保证终止。下面通过两个示例说明该算法能够处理的公式类别。

示例*1*。

$$f(0) = 0 \wedge \forall x \,.\, f(-2x+3) = g(x) + f(-2x+c+1) - c,$$

其中 c 是一个未解释常量，且量化变量 x 的系数对于 f的所有参数均为 -2 ，对于 g的所有参数均为1。

例 *2*（无效）。

$$\forall x \,.\, f(2x+3) = f(x+1)$$

该公式不属于我们的片段，因为 f 出现时其参数具有不同的系数（2和1）。

请注意，某些含有多个量化变量的公式可以通过变量变换轻松转换到这一类。例如，公式

$$f(0) = 0 \wedge \forall y, z \,.\, f(2y-2z+3) = g(-y+z) + f(2y-2z+c+1) - c,$$

可化简为例1 ，通过变量变换 $x = -y + z$。

据我们所知，目前没有结果表明这一类具有可判定性。不过，我们也不知道有任何结果证明它是不可判定的。事实上，我们的目标不是对可判定性进行分类，而是提出一种用于证明可满足性的算法，使其能够轻松地在 SMT 求解器中实现，并且在实践中高效。

4 符号与术语

按照 SMT 语境中的惯例，我们将使用一种带排序的一阶语言。我们还假设，变量只出现在受量词约束的位置。我们已经使用符号 $\forall \bar{x}$ 作为对多个变量进行全称量化的简写。在这种情况下，我们用 $|\bar{x}|$表示这些变量的数量。对于公式 F 和未解释函数符号 f， $args(F,\ f)$表示 f在 F中的参数项集合。我们用 $s[v \leftarrow t]$表示在项或公式 s中以项 t 代入 v 的结果。这里， v 也可以表示多个变量；此时，我们要求t表示一个由项构成的 $|v|$向量，并假设代入是并行进行的。

有时我们会使用集合 $\{+, -\}$ 中的符号，其中对于任意两个符号 s 和 s'， ss' 当且仅当 $s = s'$，否则为 $+$。我们将使用这类符号来切换大于和小于关系：对于两个整数 z和 z'， $z \lesseqgtr^{+} z'$ 当且仅当 $z < z'$，且 $z \lesseqgtr^{-} z'$ 当且仅当 $z > z'$。对于区间 $B = [b_{\min}, b_{\max}]$，我们定义 $\mathsf{bd}(B, -) = b_{\min}$ 和 $\mathsf{bd}(B, +) = b_{\max}$。

我们采用经典的良序数学概念。集合 S 上的良序是 S 上的一种全序，使得 S 的每个非空子集都有最小元素。任何良序都允许进行良基归纳；要证明性质 P 对所有 x 都成立，只需证明：对于每个 $x \in S$，若对所有 $y \prec x$ 都有 $P(y)$，则也有 $P(x)$。

我们按以下含义使用证书这一概念 [25]：给定一个通常难以检查的性质 $P(x)$，该性质的一个证书是

一个对象 Π，它满足性质 $P'(x, \Pi)$，且有证明表明 $P'(x, \Pi)$ 蕴含 $P(x)$；并且检查 $P'(x, \Pi)$ 比检查 $P(x)$ 更容易。本文不会始终严格界定“易于检查”的含义。不过，最终目标是获得一个证书 Π，使得借助某种意义上比用于检查 $P(x)$ 的算法更简单的算法，能够比检查原始性质 $P(x)$ 更高效地检查 $P'(x, \Pi)$。因此，这种证书能够让人们高效地独立验证用于检查 $P(x)$ 的算法所得结果是否正确。

5 可满足性证书

本节定义一些对象，用于证明形如 $F \wedge \forall \bar{x}.\, Q$ 的公式可满足。我们假设量化变量 $\bar{x}$ 的排序为 $\mathbb{D}$，并且所有未解释函数符号的定义域为 $\mathbb{D}$，值域为 $\mathbb{D}'$。我们还假设 $\mathbb{D}$ 和 $\mathbb{D}'$ 都有一个同名的典范模型，且 $\mathbb{D}$ 是可数的。最后，我们假设对于每个元素 $z \in \mathbb{D}$，语言中都有一个对应的解释常量 $\hat{}z$ 表示 z。

下一节中，我们将介绍一种算法，在 $\mathbb{D}$ 和 $\mathbb{D}'$ 均为整数（更具体地说，是 UFLIA 中的公式），且满足问题陈述中所述附加限制的情况下，计算此类证书。本节中，我们也在所有示例中使用整数进行说明。一般而言，我们将相对于底层理论（即未解释函数符号、 $\mathbb{D}$ 和 $\mathbb{D}'$ 的组合理论）来使用所有语义概念（例如符号 $\models$）；而在所有示例中，底层理论均为 UFLIA。

传统上，逻辑公式的可满足性由一种解释来证明，该解释为每个非逻辑符号赋予一个值，使公式为真。然而，无限定义域上的未解释函数符号的存在，引出了如何表示此类解释的问题。我们的方法不是一次性为未解释函数符号的无限定义域中的所有元素赋值，而是分别为该定义域中的每个元素赋值：

定义 1（单元）。单元是形如 $f(n_1, \ldots, n_k)$ 的项，其中 f 是一个未解释函数符号，且 $n_1, \ldots, n_k \in \mathbb{D}$。
我们用 $C(\mathbb{D})$ 表示单元集合。单元的例子有 c 和 $f(7)$。项 $f(c + 7)$ 不是单元。

定义 2（单元解释）。单元解释是 $C(\mathbb{D}) \rightharpoonup \mathbb{D}'$中的一个部分函数。若单元解释 I 与解释 $\mathcal{I}$ 在 I 有定义的所有单元上取值一致，即 I 且 $\mathcal{I}$，则称二者相容。

给定一个单元解释 I，我们用 $def(I)$ 表示I 有定义的单元集合。给定一个项 t 和一个单元解释 I，我们定义 $I(t) := \mathcal{I}(t)$，如果对每一个与 I相容的解释 $\mathcal{I}$，$\mathcal{I}(t)$ 都是唯一的；否则$I(t) := \bot$。例如，对于 $I = \{f(0) \mapsto 7\}$，$I(f(0)) = 7$，但 $I(f(1)) = \bot$。还需注意 $I(f(1) - f(1)) = 0 \neq \bot$。类似地，对于公式 ϕ，我们定义

单元解释 I 满足 ϕ，当且仅当对每一个与 I相容的解释 $\mathcal{I}$， $\mathcal{I}$ 都满足 ϕ，此时我们也记作 $I \models \phi$。

为了支持单元解释的计算机表示，我们将始终使用仅在有限多个单元上有定义的单元解释。不过，我们要求定义的单元足够多，以便 (i) 检查公式的无量词部分是否满足，以及 (ii) 对全称量词的每一种实例化，确定全称量化部分中未解释函数符号的所有参数项的值。

定义 3（预可满足性证书）。给定一个形式为 $F \wedge \forall \bar{x} \,.\, Q$ 的公式 ϕ，我们称满足以下条件的单元解释 I 为

– $I \models F$，且 – 对所有 $z \in \mathbb{D}^{|\bar{x}|}$，对于每个未解释函数符号 f 及其参数项 $t \in args(Q, f)$， $I(t[\bar{x} \leftarrow \hat{} z]) \neq \bot$

一个 ϕ的预可满足性证书。

例如， $I = \{c \mapsto 5\}$ 是 $c \geq 5 \wedge \forall x \,.\, f(x + c + 1) = f(x + c) + 1$ 的预可满足性证书。

预可满足性证书将 F 中未解释函数符号的所有参数解释为定义域 $\mathbb{D}$ 中的一个值，并且连同量化变量的一个取值，也将它们解释为 Q 中的一个值。这样，我们就能针对每个实例化，确定哪些单元会变得相关。

定义 4（相关单元）。对于任意预可满足性证书 I 和 $z \in \mathbb{D}^{|\bar{x}|}$，相关单元的集合 相关单元 $\Gamma_{I,\, z}(F \wedge \forall \bar{x} \,.\, Q) :=$

$$\{f(I(t[\bar{x} \leftarrow \hat{z}])) \mid t \in args(Q, f), f \text{ is an uninterpreted function symbol}\}.$$

继续这个例子，由于 I 将 c 的值设为 5， $\Gamma_{I,\, 10}(\phi) = \{f(15)\ f(16)\}$。

给定一个预可满足性证书后，我们现在可以通过传播预可满足性证书给出的值来证明公式的可满足性，使得对于全称量化变量的每个取值，相应的量化子公式都成立。

定义 5（可满足性证书）。给定一个形式为 ϕ $F \wedge \forall \bar{x} \,.\, Q$ 的公式 ϕ，其可满足性证书为

– 一个预可满足性证书 I，对应于 ϕ，–
在 $\mathbb{D}^{|\bar{x}|}$ 上的良序 $\preceq$，– 对于每个 $z \in \mathbb{D}^{|\bar{x}|}$

- 一组 $X_z \subseteq \Gamma_{I,\, z}(\phi)$ （传播单元），使得

$$X_z \cap def(I) = \emptyset \text{ and } X_z \cap \bigcup_{z' \prec z} \Gamma_{I,z'}(\phi) = \emptyset,$$

and

- 一个证书（可满足性传播器），用于证明：对于 $\Gamma_{I,\ z}(\phi)\ \setminus\ \mathit{def}(I)\ \setminus X_z$ 中单元的所有取值，都存在 X_z 中单元的取值，使得单元解释 I_{X_z} 将这些取值赋给 X_z，$I \cup I_{X_z} \models Q[\bar{x} \leftarrow \hat{}z]$。

直观而言，可满足性证书编码了一个归纳证明，其中良序 $\preceq$ 集合确定归纳方向，而归纳的每一步都表明，在强制选择另一个 z 值来满足 $\bar{x}$ 时，普遍量化部分 Q 是可满足的。这样的步骤可以为传播单元 X_z 选择值。选择这些值时，必须确保传播单元既未由预可满足性证书 I 赋值，也未在更早的归纳步骤中赋值。可满足性传播器只需证明存在适当的值可赋给这些传播单元。实际上，它通常会为这些单元赋值。

示例 *3.* 公式的一个可满足性证书

$$f(3,a) \geq 4 + g(b) \wedge \forall x, y\ .\ f(x,y) < x + g(y) \wedge g(y) = g(y+b) + 2b$$

由预可满足性证书 $I = \{a \mapsto 0, b \mapsto 1, f(3,\ 0) \mapsto 4,\ g(1) \mapsto 0\}$ 以及下表给出，其中第一列表示次序 $\preceq$:

	$X_{(x,y)}$	$propagators$
$x = 3, y = 0$	$\{g(0)\}$	$g(0) \mapsto 2$
$x = 3, y = 1, 2, \ldots$	$\{f(3,\ y),\ g(y+1)\}$	$g(y+1) \mapsto g(y) - 2,\ f(x,\ y) \mapsto y + g(y) - 1$
$x = 3,\ y = -1,\ -2,\ \ldots$	$\{f(3,\ y),\ g(y)\}$	$g(y) \mapsto g(y+1) + 2, f(x,y) \mapsto y + g(y+1) + 1$
$x \neq 3,\ y$ 任意	$\{f(x,y)\}$	$f(x,y) \mapsto y + g(y) - 1$

此处定义的证书概念是抽象的，因为它没有固定一种具体的有限表示，以便独立检查给定对象是否确实是可满足性证书。要得到具体且可独立检查的证书，只需固定一种良序，并确定传播单元和可满足性传播器的表示方式，使其能够接受这种独立检查。例如，所有这些定义都可以用线性整数算术的语言来表述，从而可以使用相应的判定过程进行独立检查。下一节将针对 第3节中描述的特殊公式类对此进行说明。

还需注意，对于满足 $z \in \mathbb{D}^{|\bar{x}|}$ 且 $\Gamma_{I,\ z}(\phi) \subseteq \mathit{def}(I)$ 的情况，根据定义，传播单元 X_z 集合必须为空。因此，对于此类 z，可满足性传播器只需证明 $I \models Q[\bar{x} \leftarrow \hat{}z]$。换言之，作为预可满足性证书的单元解释 I 已经起到了可满足性传播器的作用。

定理1。 任何具有可满足性证书的 ϕ 形式的公式 $F \wedge \forall \bar{x}\ .\ Q$ 都是可满足的。

证明。 假设给定形式的公式具有可满足性证书。相应的预可满足性证书 I 满足 ϕ的量词自由部分 F。

我们首先证明，对于每个 $z \in \mathbb{D}^{|\bar{x}|}$，都存在 I 的一个扩展 I'，使得 I' 满足在 z 之前出现的所有量词实例化（相对于顺序 $\preceq$），也就是说，$I' \models \{Q[\bar{x} \leftarrow \hat{}z'] \mid z' \preceq z\}$。为此，我们对良序进行归纳：令 $z \in \mathbb{D}^{|\bar{x}|}$，并令 Φ 为 $\{Q[\bar{x} \leftarrow \hat{}z'] \mid z' \prec z\}$。假设存在$I$的一个扩展 I_Φ，满足 $I_\Phi \models \Phi$。我们证明存在 I的一个扩展 I'，满足 $I' \models \Phi \cup \{Q[\bar{x} \leftarrow \hat{}z]\}$。根据定义 5，传播单元 X_z 与实例化 $z' \prec z$中出现的所有单元互不相交，因此不会出现在 Φ的任何元素中。令 I' 为 I_Φ 的扩展，它将传播单元 X_z 赋值为可满足性传播器所保证存在的值。由于这些单元不会出现在 Φ中，I' 与 I_Φ 在 Φ中出现的所有单元上取值相同。因此，I' 同时满足Φ 和 $Q[\bar{x} \leftarrow \hat{}z]$，所以它满足 $\Phi \cup \{Q[\bar{x} \leftarrow \hat{}z]\}$。

现在，对于每个 $z \in \mathbb{D}^{|\bar{x}|}$，记满足 $Q[\bar{x} \leftarrow \hat{}z]$ 的单元解释为 I_z，其构造如上所述。注意，对于每个单元 u，以及所有 $z,\ z' \in \mathbb{D}^{|\bar{x}|}$，若 $I_z(u) \neq \bot$ 且 $I_{z'}(u) \neq \bot$，则 $I_z(u) = I_{z'}(u)$。令 I_ϕ 为如下定义的（经典）解释：对于任意单元 u，若存在某个 z 使得 $I_z(u) \neq \bot$，则 $I_\phi(u) = I_z(u)$；否则 $I_\phi(u) = 0$。显然 $I_\phi \models F$。此外，对于每个 $z \in \mathbb{D}^{|\bar{x}|}$，$I_\phi \models Q[\bar{x} \leftarrow \hat{}z]$，因为在 I_z 有定义的所有单元上，I_ϕ 与 I_z 一致。因此 $I_\phi \models \phi$。 ■

可满足性证书只能表明一个公式是可满足的，并不提供相应模型的具体取值。现在，我们说明如何从给定的、包含预可满足性证书 I 的可满足性证书中计算出这些取值。为此，我们要求可满足性传播器具有构造性。因此，对于 $z \in \mathbb{D}^{|\bar{x}|}$ 以及每个 $u \in X_z$，我们假设存在一个函数$prop_{u,\ z}$，使得对于单元解释 I'，若 定义 (I') $= \Gamma_{I,\ z}(\phi)$ \定义 (I) $\setminus X_z$，则$I \cup I' \cup \{u \mapsto prop_{u,z}(I \cup I') \mid u \in X_z\} \models Q[\bar{x} \leftarrow \hat{}z]$。

注意，对于任意单元 u，至多存在一个满足 $u \in X_z$ 的 $z \in \mathbb{D}^{|\bar{x}|}$（若存在两个不同的实例化，由于 $\preceq$是良序，因此存在一个最小的实例化，记为 $z_{\min}$；于是对于任意 $z' \succ z$，对 $X_{z'}$ 的要求都会排除 $X_{z'}$ 同时包含 u 的可能性）。因此，对于任意单元 u，令 实例(u) 为满足 $u \in X_z$ 的 $z \in \mathbb{D}^{|\bar{x}|}$；若不存在这样的实例化，则令其为 $\bot$。

那么，对于定理 I_ϕ 1 的证明中给出的赋值，它满足给定公式 ϕ和给定单元 u，$I_\phi(u)$ 的值可通过以下递归函数 $val(u)$ 计算：$C(\mathbb{D}) \to \mathbb{D}'$:

如果 I 将值 x 赋给 u，**则返回** x**如果** $inst(u) = \bot$，**则返回** $\mathbb{D}$ 中的任意元素**返回** $prop_{u,\ inst(u)}(I \cup \{v \mapsto val(v) \mid v \in \Gamma_{I,\ inst(u)}(\phi) \setminus def(I) \setminus X_{inst(u)}\})$

这里，递归调用按照以下意义上的顺序递减 $\prec$:

性质 *1*。设 u 是满足 $inst(u) \neq \bot$ 的单元。那么，对于任意其他单元 $v \in \Gamma_{I,\ inst(u)}(\phi) \setminus def(I) \setminus X_{inst(u)}$，若满足 $inst(v) \neq \bot$，则有 $inst(v) \prec inst(u)$。

因此，程序 val 的每次递归调用要么导致$inst(u)$减小，要么被 $\bot$。后一种情况下，程序会立即终止；前一种情况不可能无限次发生，因为所用的序是良基的。因此我们有:

性质 *2*。 程序 val 终止。

请注意，一般来说，无法通过计算 $X_z, z \in \mathbb{D}^{|\bar{x}|}$的值来计算 $val(u)$：先将 z 初始化为关于 $\prec$的最小 z，然后使用一个循环，按照关于 $\prec$ 的后继关系，让 z 从一个后继递增到下一个。例如，如果 $\prec$ 将所有偶数排在所有奇数之前，那么在计算某个奇数的值之前，这个循环就必须遍历无限多个偶数。

6 算法

现在，我们从一般的证明框架转向具体计算。本节中，我们假设公式 ϕ 具有 $F \wedge \forall \bar{x}.\, Q$的形式，但还需满足问题陈述（第3节）中所述的额外限制。具体而言，我们只允许一个量化变量 x。这些限制还要求任何未解释函数符号 f 在 Q 中的所有参数都具有相同的系数，我们用 $\sigma(f)$ 表示该系数的符号。

我们提出一种算法，通过计算特定形式的可满足性证书来检查 ϕ的可满足性。计算得到的证书明确证明量词自由部分 F 可满足，并证明量化部分 Q在由实例化构成的有限区间 $B = [b_{\min}, b_{\max}]$ 上可满足。对于 B之外的取值，该算法使用可满足性传播器，并按从 $b_{\max}+1$向上、从 $b_{\min}-1$向下传播的顺序进行传播。我们证明了该算法的可靠性，并在确保此类传播能够进行的条件下证明其完备性。

我们现在形式化如下条件，以确保从基区间 B 向外传播是可能的。设 $\mathcal{F}$ 为 ϕ的量化部分 Q 中的未解释函数符号集，并设 $\mathcal{T}$为Q中形如 $f(t)$的项集。对于传播方向 $s \in \{+, -\}$（其中 $+$ 表示递增的 x， $-$表示递减的 x），我们选取项的一个子集 $\mathcal{S} \subseteq \mathcal{T}$，这些项是极值的，即它们彼此相等，并且根据系数的符号以及传播方向，相对于同一函数符号的其他参数项处于适当的严格序关系中。该可传播性条件 $\Phi_{\text{prop}}(\mathcal{S})$保证：无论非极值出现项被赋予何种值，我们都能找到值 $v_1, \ldots, v_r$，使得将 v_i 赋给 $\mathcal{S}$中的所有出现项时，Q为真。不同的函数符号可以取不同的值。相反方向上的传播条件是对称的，使用另一个子集$\mathcal{S}' \subseteq \mathcal{T}$。

定义6（请求枢轴 **条件**）。对于任意选择的子集 $\mathcal{S} \subseteq \mathcal{T}$和符号 $s \in \{-, +\}$，定义：

– 极值条件 $\Phi^{s}_{ext}(\mathcal{S})$**:**

$$\bigwedge_{f \in \mathcal{F}} \left(\bigwedge_{f(t), f(t') \in \mathcal{S}} t = t' \;\wedge \bigwedge_{f(t) \in \mathcal{S}, f(t') \in \mathcal{T} \setminus \mathcal{S}} t' \lesseqgtr^{\sigma(f)s} t \right) [x \leftarrow \hat{0}].$$

– 可传播性条件 $\Phi_{prop}(\mathcal{S})$：

$$\forall x \forall \bar{u} \exists \bar{v}.\ Q'(\mathcal{S})$$

其中，$\bar{u}$ 对应变量 $u_{f(t)}$，其 $f(t) \in \mathcal{T} \setminus \mathcal{S}$，而 $\bar{v}$ 对应变量 $v_{f(t)}$，其 $f(t) \in \mathcal{S}$；Q' 由 Q 通过同时替换而得到

- 将每个 $f(t) \in \mathcal{T} \setminus \mathcal{S}$ 替换为 $u_{f(t)}$，并
- 将每个 $f(t) \in \mathcal{S}$ 替换为 $v_{f(t)}$。

我们称 Q 满足子集 $\mathcal{S} \subseteq \mathcal{T}$ 和 $\mathcal{S}' \subseteq \mathcal{T}$ 的 *ReqPivot* 条件，当且仅当 $\Phi^{+}_{ext}(\mathcal{S})$、$\Phi_{prop}(\mathcal{S})$、$\Phi^{-}_{ext}(\mathcal{S}')$ 以及 $\Phi_{prop}(\mathcal{S}')$ 均可满足。此外，当且仅当存在这样一对 $(\mathcal{S}, \mathcal{S}')$ 时，Q 满足 *ReqPivot* 条件。

例 4（ReqPivot 示例）。考虑

$$\forall x.\ f(x+c+3) + f(x+4) + f(x+1) = g(x+2) + h(x),$$

其基项常量为 c。所有系数均为 $+1$，因此 $\sigma(f) = \sigma(g) = \sigma(h) = +$。项集为

$$\mathcal{T} = \{f(x+c+3),\ f(x+4),\ f(x+1),\ g(x+2),\ h(x)\}.$$

向上传播（$s = +$）。令 $\mathcal{S} = \{f(x+4),\ g(x+2)\}$。对于 f，由于 $\sigma(f)s = +$ 且 $\lesseqgtr^{+}$ 是 $<$，括号内的极值条件部分为

$$x + c + 3 < x + 4 \ \wedge\ x + 1 < x + 4,$$

将 x 替换为 0 后，得到 $c < 1$。对于 g 和 h，$\mathcal{T} \setminus \mathcal{S}$ 不包含相应的项，因此该条件是平凡的。

可传播性变为

$$\forall x\ \forall u_{f(x+c+3)},\ u_{f(x+1)},\ u_{h(x)}\ \exists v_{f(x+4)},\ v_{g(x+2)} \text{ such that}$$
$$u_{f(x+c+3)} + v_{f(x+4)} + u_{f(x+1)} = v_{g(x+2)} + u_{h(x)}.$$

这是可满足的，例如选择 $v_{g(x+2)} = 0$ 和 $v_{f(x+4)} = u_{h(x)} - u_{f(x+c+3)} - u_{f(x+1)}$。

向下传播（$s = -$）。令 $\mathcal{S}' = \{f(x+1),\ h(x)\}$。由于 $\sigma(f)s = -$ 且 $\lesseqgtr^{-}$ 是 $>$，括号内的极值条件部分为

$$x + c + 3 > x + 1 \ \wedge\ x + 4 > x + 1,$$

将 x 替换为 0 后，得到 $c > -2$。h 的条件是平凡的。可传播性变为

$$\forall x\ \forall u_{f(x+c+3)},\ u_{f(x+4)},\ u_{g(x+2)}\ \exists v_{f(x+1)},\ v_{h(x)} \text{ such that}$$
$$u_{f(x+c+3)} + u_{f(x+4)} + v_{f(x+1)} = u_{g(x+2)} + v_{h(x)},$$

这是可满足的，例如通过选择 $v_{h(x)} = 0$ 和 $v_{f(x+1)} = u_{g(x+2)} - u_{f(x+c+3)} - u_{f(x+4)}$。

结论。Q 满足 $(\mathcal{S}, \mathcal{S}')$ 的 *ReqPivot* 条件，当且仅当 $c \in (-2, 1)$。

现在，给定一个具体区间 $B = [b_{\min}, b_{\max}]$，我们需要额外的“冲突条件”，以防止传播过程中生成的单元与基项部分 F发生冲突。

定义 7（区间扩展公式）。对于区间 $B \subseteq \mathbb{Z}$、符号 $s \in \{-, +\}$ 和子集 $\mathcal{S} \subseteq \mathcal{T}$，定义 $\Psi^s(B, \mathcal{S}) :=$

$$\Phi^s_{ext}(\mathcal{S}) \wedge \Phi_{prop}(\mathcal{S}) \wedge \bigwedge_{f(t)\in\mathcal{S}} \bigwedge_{a\in args(F,f)} a \lesseqgtr^{\sigma(f)s} t[x \leftarrow bd(B, s)]$$

其中 $\Phi^s_{ext}(\mathcal{S})$ 和 $\Phi_{prop}(\mathcal{S})$ 如定义6所定义。

因此，对于任意符号 $s \in \{-, +\}$ 和子集 $\mathcal{S} \subseteq \mathcal{T}$，$\Psi^s(B, \mathcal{S})$ 表示$\mathcal{S}$ 满足方向 s 的 $ReqPivot$ 条件，且还满足区间 B的冲突条件。该冲突条件确保，对于 F中出现的 f的每个基项 a，传播后的单元 $f(t[x \leftarrow \mathrm{bd}(B, s)])$（其中 $f(t) \in \mathcal{S}$）不同于基项单元 $f(a)$。

定理 2（区间可满足性）。设 B 为非空整数区间，并假设

$$\left(F \wedge \bigwedge_{x\in B} Q[x \leftarrow \hat{x}]\right) \wedge \left(\bigvee_{\mathcal{S}\subseteq\mathcal{T}} \Psi^+(B, \mathcal{S})\right) \wedge \left(\bigvee_{\mathcal{S}'\subseteq\mathcal{T}} \Psi^-(B, \mathcal{S}')\right)$$

可满足。那么 $F \wedge \bigwedge_{x\in B} Q[x \leftarrow \hat{}x] \wedge \forall x \,.\, x \notin B \Rightarrow Q$ 有一个可满足性证书，因此 ϕ 可满足。

定理 2前提中的公式属于 Presburger 算术（即 SMTLIB LIA）的语言，因此可以通过算法检查其可满足性。因此，满足该公式的区间 $[b_{\min}, b_{\max}]$ 是一个可通过算法检查的证书，用于证明 ϕ的可满足性。为了提高算法检查的效率，还可以纳入证明 $F \wedge \bigwedge_{x\in B} Q[x \leftarrow \hat{}x]$ 可满足性所需的单元解释，以及集合 $\mathcal{S}$ 和 $\mathcal{S}'$。

定理 2 的证明保证了这些可满足性传播器的存在。其算法实现通常可以从对可传播性条件 $\Phi_{\mathrm{prop}}(\mathcal{S})$ 和 $\Phi_{\mathrm{prop}}(\mathcal{S}')$ 进行量词消去时产生的副产品——见证函数——中获得。随后，可以使用这些传播器，在 val 中定义的递归求值函数第 5 节来计算模型为未解释函数符号所赋予的函数值。

该定理立即给出一种算法，可通过搜索区间 $[b_{\min}, b_{\max}]$来检查可满足性。这一点可见于 算法 1。对该算法进行扩展，使其在返回值为 SAT时也返回相应的证书，这很简单。

该结果的正确性 SAT 是 算法 1的直接推论。该结果的正确性 UNSAT 源于以下事实：while 循环的终止条件是输入公式的逻辑推论。然而，算法也可能永远运行下去。在某些条件下，我们可以排除这种情况：

算法 1 基于区间的可满足性检查

输入： $\phi = F \wedge \forall x.\, Q$ 在片段中
输出： SAT 或 UNSAT

$b_{\min} \leftarrow 0$, $b_{\max} \leftarrow 0$
当 $F \wedge \bigwedge_{x=b_{\min}}^{b_{\max}}$ **时** $Q[x]$ 可满足 **执行**
  向上 $\leftarrow \bigvee_{\mathcal{S} \subseteq \mathcal{T}} \Psi^{+}([b_{\min}, b_{\max}],\ \mathcal{S})$
  向下 $\leftarrow \bigvee_{\mathcal{S}' \subseteq \mathcal{T}} \Psi^{-}([b_{\min}, b_{\max}],\ \mathcal{S}')$
  如果 $\left(F \wedge \bigwedge_{x=b}^{b_{\max}}{}_{\text{最小}}\ Q[x]\right) \wedge$向上 $\wedge$向下 可满足 **则**
    返回 SAT
  $b_{\min} \leftarrow b_{\min} - 1$, $b_{\max} \leftarrow b_{\max} + 1$ ▷ 对称扩展区间
返回 UNSAT

定理3（相对完备性）。如果输入公式 ϕ是可满足的，并且满足 ReqPivot 条件，则算法 *1* 终止并返回SAT。

虽然证明不可满足性并非本文的主要目标，算法 1 在某些情况下也能检测出不可满足性。

定理4（有限不可满足公式的终止性）。如果存在一个有限集合 $Z \subseteq \mathbb{Z}$ 使得 $F \wedge \bigwedge_{z \in Z} Q[x \mapsto \hat{}z]$ 不可满足，则算法 *1* 将终止并返回 UNSAT。

然而，存在一些在标准模型中不可满足的公式，对于这些公式，该算法不会终止。

示例 *5*（需要归纳的公式）。考虑以下公式：

$$f(0) = 0 \wedge f(c) \neq 0 \wedge \forall x.\, f(x) = f(x+1)$$

该公式在标准整数算术中不可满足（在两个方向上应用归纳可得 $\forall n.\, f(n) = 0$，这与 $f(c) \neq 0$ 相矛盾）。然而，任何有限的实例化集合都是可满足的，只需选择 c 为一个足够大的整数。因此，算法 1 将无限扩展该区间，而无法检测出不可满足性。

7 命题编码

迄今所提出的算法依赖于针对所有可能的子集选择 $\mathcal{S} \subseteq \mathcal{T}$ 与 $\mathcal{S}' \subseteq \mathcal{T}$ 的大型析取式。尽管在理论上十分优雅，这些析取式会导致公式随集合 $\mathcal{T}$ 的规模呈指数增长，从而在计算上不切实际。我们现在给出一种紧凑的合取范式编码，以避免这一问题。

我们不再枚举所有可能的子集选择，而是引入辅助布尔变量来表示元素是否属于集合 $\mathcal{S}$ 和 $\mathcal{S}'$。对于每个

项 $f(t) \in \mathcal{T}$，我们创建一个命题变量 $p_f^+(t)$，当且仅当 $f(t) \in \mathcal{S}$ 时该变量为真；另一个变量 $p_f^-(t)$，当且仅当 $f(t) \in \mathcal{S}'$ 时该变量为真。

接下来，我们使用这些变量对极值条件 Φ_{ext}^s、可传播性条件Φ_{prop}和冲突条件进行编码，从而为每个符号 s 得到 $\Psi^{s,e}(B)$ 的紧凑表示。

定义 8（区间扩展公式的合取范式编码）。对于区间 $B = [b_{\min}, b_{\max}]$ 和符号 $s \in \{+, -\}$，定义：

$$\Psi^{s,e}(B) \equiv \Phi_{ext}^{s,e} \wedge \Phi_{prop}^{e} \wedge \mathit{ClashEnc}^s$$

其中：

- 极值条件编码 $\Phi_{ext}^{s,e}$：

$$\bigwedge_{f \in \mathcal{F}} \bigwedge_{f(t), f(t') \in \mathcal{T}} \Big((p_{f(t)}^s \wedge p_{f(t')}^s) \Rightarrow t = t' \wedge (p_{f(t)}^s \wedge \neg p_{f(t')}^s) \Rightarrow t' \lesseqgtr^{\sigma(f)s} t \Big)[x \leftarrow \hat{0}]$$

- 传播编码 Φ_{prop}^e：

$$\bigwedge_{\mathcal{T}_- \subseteq \mathcal{T}} \left(\bigwedge_{f(t) \in \mathcal{T}_-} \neg p_{f(t)}^s \wedge \bigwedge_{f(t) \in \mathcal{T} \setminus \mathcal{T}_-} p_{f(t)}^s \Rightarrow \forall x \, \forall \bar{u} \, \exists \bar{v} \; Q^{\mathcal{T}_-} \right),$$

  其中 $\bar{u} = (u_{f(t)})_{f(t) \in \mathcal{T}_-}$ 和 $\bar{v} = (v_{f(t)})_{f(t) \in \mathcal{T} \setminus \mathcal{T}_-}$，而 $Q^{\mathcal{T}_-}$ 由Q通过同时将每个 $f(t) \in \mathcal{T}_-$ 替换为 $u_{f(t)}$，并将每个 $f(t) \in \mathcal{T} \setminus \mathcal{T}_-$替换为 $v_{f(t)}$而得到。

- 冲突条件编码 $\mathit{ClashEnc}^s$：

$$\bigwedge_{f(t) \in \mathcal{T}} \bigwedge_{a \in args(F,f)} \Big(p_{f(t)}^s \Rightarrow a \lesseqgtr^{\sigma(f)s} t[x \leftarrow bd(B, s)] \Big)$$

注意，如果对于某个 $f \in \mathcal{F}$，其参数项的基项部分（通过代入 $x = 0$ 得到）两两不同，那么在 $\Psi^{s,e}(B)$ 的任何满足赋值下，至多只有一个 $p_{f(t)}^s$ 可以为真。这是由极值条件推出的；否则就会要求两个不同的基项相等。

性质 *3*（编码正确性）。对于任意区间 $B = [b_{\min}, b_{\max}]$ 和符号 $s \in \{+, -\}$，$\Psi^{s,e}(B)$ 可满足，当且仅当存在 $\mathcal{S} \subseteq \mathcal{T}$ 使得 $\Psi^s(B, \mathcal{S})$ 成立。

8 实验

我们的实验检验了如下猜想：前两节所介绍的算法能够在证明所考虑类别中的公式可满足这一能力上，优于当前最先进的 SMT 求解器。我们

准备了一个由28个人工构造的可满足问题组成的测试集[4]，并且此外还满足 $ReqPivot$ 条件。我们集合中的所有问题最多包含两个常量、一个或两个元数为1的函数符号、简单的基约束以及一个全称量化断言。我们集合中问题的两个例子是

$$f(4) = 7 \ \wedge \ \forall x.\, f(x+1) = f(x) + 1 \tag{2}$$

and

$$d > 1 \wedge f(0) = 0 \wedge g(2) > f(d+1) \wedge \forall x\,.\, f(x-1) = f(x) + g(x) + g(x+d). \tag{3}$$

为简化实现，我们使用了带有额外守卫条件的版本，将全称量化变量的取值范围限制为自然数而非整数。所得公式的形式为 $F \wedge (\forall x. 0 \leq x \Rightarrow Q)$。为研究求解器在模型规模增大时的行为，我们还使用了守卫条件将范围限制为上界为 10^c 的有限区间的版本，其中 $c \in \{1, \ldots, 5\}$。所得公式的形式为 $F \wedge \forall x\,.\, (0 \leq x \leq 10^c \Rightarrow Q)$。

除上界 10^c之外，我们的问题在函数参数中并不包含大整数，否则便需要对区间 $[b_{\min}, b_{\max}]$ 进行更智能的初始化，以避免算法 1中 while 循环的大量循环迭代。

我们使用 z3 Python API 实现了我们的算法。我们的实现仅使用 Ψ^+ 检查向上归纳，而省略向下归纳；由于上一段所述的守卫条件，这样做是可靠的。结果表明，我们的算法能在可忽略的时间内在所有问题上终止。此外，参数 c 的取值对测试问题有界版本的运行时间没有明显影响。我们发现前一观察结果出人意料，而后一结果则在意料之中，这是因为使用了归纳。

我们还将 SMT 求解器 z3 和 CVC5 直接应用于这些问题，并明确启用基于模型的量词实例化 [15]。此外，我们以 SYGUS 推理模式运行了CVC5 。为研究这三种 SMT 求解器直接应用于这些问题时的表现，以及参数 c 对其运行时间的影响，我们进行了实验，时间限制设为 10 分钟。设置$c = 1$时， SMT 求解器会立即在这些问题上终止。对于问题 (2)，在 $c = 3$条件下， CVC5 用 54 秒返回 SAT；对于 $c \geq 4$，求解器则超时。在 $c \geq 2$条件下， z3 在 12 秒后返回 UNKNOWN 。对于问题 (3)，随着 c 增加，两个求解器都无法求解：当 $c \geq 3$时， CVC5 超时；而 z3 会为 $c \geq 2$ 返回 UNKNOWN（通常耗时 44–50 秒）。图 1 展示了两个求解器在这些问题上随着 c 增加时的表现。我们还测试了这三种求解器在无界问题上的表现，即移除界限 $0 \leq x \leq 10^c$。 z3 解决了其中 4 个问题，而 CVC5 一个也没有解决。

CVC5在 SYGUS 模式下，无论断言边界如何，都解决了我们 15 个问题中的 15 个。 CVC5 在 SYGUS 模式下未解决的一个问题是

$$f(0) = 1 \ \wedge \ \forall x\,.\, f(x+1) = 2 * f(x).$$

[4] 我们实验的所有数据均可在我们的公共代码仓库中获取：
https://github.com/MarekDanco/f-definitions

合理地假设函数 $\lambda x.2^x$ 不属于 CVC5 默认 SYGUS语法。此外，图 1 清楚地表明， SYGUS 方法快速且稳定，但仅适用于函数符合预定义语法的问题。

我们还测试了三个求解器在 1 分钟时间限制内（以图1中的 -min 后缀表示）的性能。 z3 的性能相同。 CVC5 在 $c = 2$ 和 $c = 3$上少求解出一个问题。此外， CVC5 在SYGUS 模式下，在无界问题中少求解出一个问题。

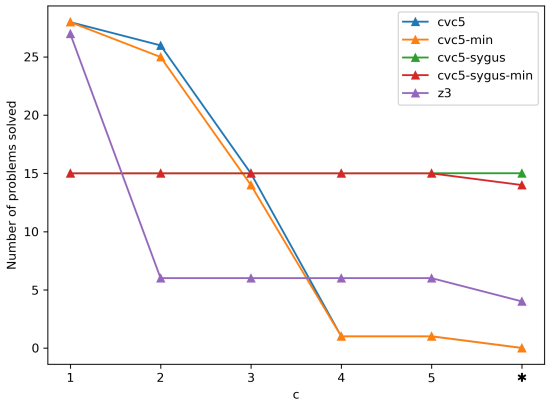

图 1：随着 c 增大而求解的问题。 $*$ 表示无界问题。

9 结论

我们提出了一种方法，用于证明含有全称量词和未解释函数符号的公式可满足；同时，还针对 SMT-LIB 理论 UFLIA 中的一类特定公式，提出了相应的算法。该方法依赖归纳，因此对于没有有限模型的公式，以及只有不具备闭式表示的函数才能使其可满足的公式，也能奏效。除输入片段的语法限制外，该算法还要求满足语义上的 $ReqPivot$条件，才能进行归纳扩展。在未来的工作中，我们希望放宽这一限制，例如采用更强形式的归纳。我们的长远目标是开发一种通用算法，它以未解释函数符号的定义域和值域所关联的底层理论为参数，并且只依赖于一个谓词，该谓词定义了应用归纳的顺序。

参考文献

1. Abate, A., Barbosa, H., Barrett, C.W., David, C., Kesseli, P., Kroening, D., Polgreen, E., Reynolds, A., Tinelli, C.：合成具有非平凡常量的程序。《自动推理杂志》67(2)，19（2023）。https://doi.org/10.1007/ s10817-023-09664-4 2. Alberti, F., Ghilardi, S., Sharygina, N.：平坦数组性质的判定过程。载于：系统构造与分析工具和算法——第20届国际会议 TACAS 2014。计算机科学讲义，第8413卷，第15–30页。施普林格（2014）3. Alur, R., Bod´ık, R., Juniwal, G., Martin, M.M.K., Raghothaman, M., Seshia, S.A., Singh, R., Solar-Lezama, A., Torlak, E., Udupa, A.：语法引导综合。载于：计算机辅助设计中的形式化方法，FMCAD 2013，美国俄勒冈州波特兰，2013年10月20—23日。第1–8页。电气电子工程师学会（2013），https://ieeexplore.ieee.org/document/ 6679385/ 4. Barbosa, H., Barrett, C., Cook, B., Dutertre, B., Kremer, G., Lachnitt, H., Niemetz, A., N¨otzli, A., Ozdemir, A., Preiner, M., Reynolds, A., Tinelli, C., Zohar, Y.：生成并利用自动推理证明证书。《美国计算机学会通讯》66 3587692 5 10 86 95 2023 10 1145 ( )，– ( )。https://doi.org/ . / . Barbosa, H., Reynolds, A., Ouraoui, D.E., Tinelli, C., Barrett, C.W.：将 SMT 求解器扩展到高阶逻辑。载于：Fontaine, P.（编），自动演绎——CADE 27——第27届自动推理国际联合会议，巴西纳塔尔，2019年8月27—30日，论文集。《计算机科学讲义》，第11716卷，第35 6 3 6 54 2019 10 1007 978 3 030 29436 – . 施普林格（）。https://doi.org/ . / - - - - . Bradley, A.R., Manna, Z.：计算演算——应用于验证的判定过程。施普林格（2007）。https://doi.org/10.1007/ 978-3-540-74113-8 7. Bradley, A.R., Manna, Z., Sipma, H.B.：数组有哪些性质是可判定的？载于：Emerson, E.A., Namjoshi, K.S.（编），验证、模型检测与抽象解释：第7届国际会议 VMCAI 2006，美国南卡罗来纳州查尔斯顿，2006年1月8—10日，论文集。《计算机科学讲义》，第3855卷，第 427 442 2006 10 1007 11609773 28 –页。施普林格（）。https://doi.org/ . / . Brown, C.E., Chvalovsky, K., Janota, M., Olsak, M., Ratschan, S.：实数上的 SMT 与函数方程求解：来自国际数学奥林匹克的挑战。载于：Barrett, C.W., Waldmann, U.（编），自动演绎——CADE 30——第30届自动推理国际联合会议，德国斯图加特，2025年7月28—31日，论文集。《计算机科学讲义》，第15943卷，第150–163页。施普林格（2025）。https://doi.org/10 0 9 9 1007 978 3 031 99984 . / - - - - . Brown, C.E., Janota, M., Kaliszyk, C.：摘要：高阶 SMT 证明的挑战与解决方案。载于：D´eharbe, D., Hyv¨arinen, A.E.J.（编），第20届可满足性模理论内部研讨会论文集；该研讨会与第11届自动推理国际联合会议（IJCAR 2022）同期举行，属于第8届联合逻辑会议（FLoC 2022）的一部分，以色列海法，2022年8月11—12日。CEUR 研讨会论文集，第3185卷，第128页。CEUR-WS.org（2022），https://ceur-ws.org/Vol-3185/abstract697.pdf 10. Brown, C.E., Janota, M., Ols´ak, M.：让符号计算乐趣无穷。载于：Brown, C.W., Kaufmann, D., Nalon, C., Steen, A., Suda, M.（编），第9届自动推理的实践方面研讨会（PAAR）与第9届可满足性检查和符号计算研讨会（SC-Square）联合论文集，2024年；该会议与第12届自动推理国际联合会议（IJCAR 2024）同期举行，法国南锡，2024年7月2日。

CEUR 研讨会论文集，第3717卷，第111–121页。CEUR-WS.org（2024），https://ceur-ws.org/Vol-3717/paper6.pdf 11. Clarke, E.M.：模型检测的诞生。载于：模型检测25年（2008）。https://doi.org/10 0 1 12 1007 978 3 540 69850 . / - - - - 。Downey, P.J.：带单个一元谓词字母的Presburger算术的不可判定性。技术报告，哈佛大学计算机技术研究中心（1972）13. Elad, N., Padon, O., Shoham, S.：有限干草堆中的无限细针：在演绎验证中寻找无限反模型。《美国计算机协会编程语言汇刊》8(POPL)，970–1000（2024）14. Farzan, A., Nicolet, V.：面向递归函数综合的反例引导的部分界定。载于：计算机辅助验证，第33届国际会议，CAV 2021。第832–855页。施普林格国际出版（2021）。https: //doi.org/10 8 39 15 1007 978 3 030 81685 . / - - - - 。Ge, Y., de Moura, L.M.：可满足性模理论中量化公式的完全实例化。载于：计算机辅助验证，第21届国际会议，CAV。第306 4 25 16 320 2009 10 1007 978 3 642 02658 – （）。https://doi.org/ . / - - - - 。Habermehl, P., Iosif, R., Vojnar, T.：关于整数数组还有哪些内容是可判定的？载于：Amadio, R.M.（编）《软件科学与计算结构基础》，第11届国际会议，FOSSACS 2008，作为欧洲软件理论与实践联合会议ETAPS 2008的一部分举行，匈牙利布达佩斯，2008年3月29日至4月6日。会议论文集。《计算机科学讲义》，第4962卷，第474–489页。施普林格（2008）。https://doi.org/10.1007/978-3-540-78499-9 33 17. Horbach, M., Voigt, M., Weidenbach, C.：带一元未解释谓词的Presburger算术全称片段是不可判定的。CoRR abs/**1703 01212** 2017 1703 01212 。（），http://arxiv.org/abs/ . 。Hozzova, P., Amrollahi, D., Hajdu, M., Kovacs, L., Voronkov, A., Wagner, E.M.：在饱和中综合递归程序。载于：Benzm¨uller, C., Heule, M.J.H., Schmidt, R.A.（编）自动推理——第12届国际联合会议，IJCAR 2024，法国南锡，2024年7月3日至6日，会议论文集，第一部分。《计算机科学讲义》，第14739卷，第154–171页。施普林格（2024）。https://doi.org/10. 1007 978 3 031 63498 7 10 / - - - - 。Hozzova, P., Bjørner, N.：Synthesiz3 this：一种针对含不可计算符号的综合的基于SMT的方法。载于：Irfan, A., Kaufmann, D.（编）第25届计算机辅助设计中的形式化方法会议论文集，FM- CAD。维也纳工业大学学术出版社（2025）。https://doi.org/10.34727/2025/ISBN. 978-3-85448-084-6 28 20. Kneuss, E., Kuraj, I., Kuncak, V., Suter, P.：模递归函数的综合。载于：2013年美国计算机协会SIGPLAN面向对象编程系统、语言与应用国际会议论文集。第407–426页。SPLASH '13，美国计算机协会（2013年10月 2509555 21 10 1145 2509136）。https://doi.org/ . / . 。Kuncak, V., Mayer, M., Piskac, R., Suter, P.：完全函数综合。载于：Zorn, B.G., Aiken, A.（编）2010年美国计算机协会SIGPLAN编程语言设计与实现会议论文集，PLDI 2010，加拿大多伦多，安大略省，2010年6月5日至10日。第316–329页。美国计算机协会（2010）。https://doi.org/10. 1145 1806632 22 1806596/ . 。Kuncak, V., Mayer, M., Piskac, R., Suter, P.：面向线性算术与集合的函数综合。《软件工具技术转化国际期刊》15(5-6)，455–474（2013）。https://doi.org/10.1007/S10009-011-0217-7

23. Lachnitt, H., Fleury, M., Barbosa, H., Jakpor, J., Andreotti, B., Reynolds, A., Schurr, H.J., Barrett, C., Tinelli, C.：改进Isabelle/HOL中的SMT证明重构流程。载于：Forster, Y., Keller, C.（编）第16届交互式定理证明国际会议（ITP 2025）。莱布尼茨国际信息学论文集（LIPIcs），第352卷，第26:1–26:22页。Schloss Dagstuhl – Leibniz-Zentrum f¨ur Informatik，达格施图尔，德国（2025）。https://doi.org/10. 423026 242025/LIPIcs.ITP..。Lipparini, E., Ratschan, S.：作为证书搜索问题的非线性超越算术可满足性。载于：NASA形式化方法研讨会。计算机科学讲义，第13903卷，第472–488页。施普林格（2023）25. McConnell, R., Mehlhorn, K., N¨aher, S., Schweitzer, P.：可认证算法。《计算机科学评论》**5**（2），119–161（2011）26. Mohamed, A., Mascarenhas, T., Khan, H., Barbosa, H., Reynolds, A., Qian, Y., Tinelli, C., Barrett, C.：lean-smt：一种用于在Lean中完成证明目标的SMT策略，第197–212页。施普林格·自然瑞士（2025）。https://doi.org/10.1007/978-3-031-98682-6 1127. Parsert, J., Brown, C.E., Janota, M., Kaliszyk, C.：SMT求解中无限模型发现的实验。载于：Piskac, R., Voronkov, A.（编）LPAR 2023：第24届程序设计、人工智能与推理逻辑国际会议论文集，哥伦比亚马尼萨莱斯，2023年6月4日至9日。计算领域 EPiC 系列，第94卷，第317–328页。EasyChair（2023）。https://doi.org/10.29007/SLRM 28. Reynolds, A.：可满足性模理论中量词实例化的冲突、模型与启发式方法。载于：Vampire 2016。第3届 Vampire 研讨会论文集。计算领域 EPiC 系列，第44卷（2017）29. Reynolds, A., Barbosa, H., Larraz, D., Tinelli, C.：经由枚举式语法引导综合的可扩展溯因算法。载于：Peltier, N., Sofronie- Stokkermans, V.（编）自动推理——第10届国际联合会议，IJCAR 2020，法国巴黎，2020年7月1日至4日，会议论文集，第一部分。《计算机科学讲义》，第12166卷，第141–160页。施普林格（2020）。https: //doi.org/109 9 301007978303051074./----。Reynolds, A., Blanchette, J.C., Cruanes, S., Tinelli, C.：可满足性模理论中递归函数的模型查找。载于：自动推理国际联合会议。第133–151页。施普林格（2016）31. Reynolds, A., Kuncak, V., Tinelli, C., Barrett, C.W., Deters, M.：可满足性模理论中基于反驳的综合。《系统设计中的形式化方法》**55**（2），73–102（2019）。https://doi. org/102 321007107030170270./S---。Reynolds, A., Tinelli, C., Goel, A., Krsti´c, S.：可满足性模理论中的有限模型发现。载于：计算机辅助验证国际会议。第640–655页。施普林格（2013）33. Solar-Lezama, A., Tancau, L., Bod´ık, R., Seshia, S.A., Saraswat, V.A.：面向有限程序的组合草图设计。载于：Shen, J.P., Martonosi, M.（编）第12届编程语言与操作系统体系结构支持国际会议论文集，ASPLOS 2006，美国加利福尼亚州圣何塞，10月211168907 3425, 200640441520061011451168857-。第–页。美国计算机协会（）。https://doi.org/./.。Steen, A., Sutcliffe, G., Fontaine, P., McKeown, J.：带类型一阶逻辑的塔斯基解释的表示、验证与可视化。载于：Piskac, R., Voronkov, A.（编）LPAR 2023：第24届程序设计、人工智能与推理逻辑国际会议论文集，哥伦比亚马尼萨莱斯，2023年6月4日至9日。计算领域 EPiC 系列，第94卷，第369–385页。EasyChair（2023）。https://doi.org/10.29007/1RHX

A 附录

性质的证明1：

假设有一个单元 u with inst(u) $\neq \bot$，一个单元 $v \in \Gamma_{I,inst(u)}(\phi) \setminus \text{def}(I) \setminus X_z$ with inst(v) $\neq \bot$，并假设 inst(v) $\succeq$ inst(u)。那么 $v \in X_{inst(v)}$ and $v \in \Gamma_{I,inst(u)}(\phi)$，因此 $X_{inst(v)} \cap \Gamma_{I,inst(u)}(\phi) \neq \emptyset$。这与要求 $X_{inst(v)} \cap \text{def}(I) = \emptyset$ and $X_{inst(v)} \cap \bigcup_{z' \prec inst(v)} \Gamma_{I,z'}(\phi) = \emptyset$ 相矛盾。 ■

证明定理 2：

假设该合取式可满足，设 I^* 是满足它的一个解释，并设 $\mathcal{S}$满足 $I^* \models \Psi^+(B, \mathcal{S})$，且 $\mathcal{S}'$满足 $I^* \models \Psi^-(B, \mathcal{S}')$。假设 $B = [b_{\min}, b_{\max}]$。我们按照定义 5构造一个证书：

1. **预可满足性证书：**定义单元解释 I 如下：
   – 对于 ϕ中的每个未解释常量 c，设置 $I(c) = I^*(c)$。– 对于每个 $a \in args($ $F \wedge \bigwedge_{x \in B} Q[x \leftarrow \hat{}x], f)$，其中 f是未解释函数符号，$I(f(I^*(a))) = I^*(f)(I^*(a))$
   根据构造，I 是有限的，并且满足 $F \wedge \bigwedge_{x \in B} Q[x \leftarrow \hat{}x]$，因为 I^*满足该公式，且 I 在所有相关单元上都与 I^* 一致。
2. **在 $\mathbb{Z}$上对 $\preceq$进行良序：**

   $$0, 1, -1, 2, -2, 3, -3, \ldots, b_{\max}, b_{\min}, b_{\max}+1, b_{\max}+2, \ldots, b_{\min}-1, b_{\min}-2, \ldots$$

3. **对于 $z \in \mathbb{Z}$，传播单元 X_z：**

   $$X_z = \begin{cases} \emptyset & \text{if } z \in B, \\ \{f(I(t[x \leftarrow \hat{z}])) \mid f(t) \in \mathcal{S}\} & \text{if } z > b_{\max}, \\ \{f(I(t[x \leftarrow \hat{z}])) \mid f(t) \in \mathcal{S}'\} & \text{if } z < b_{\min}. \end{cases}$$

   注意，极值条件 $\Phi^+_{\text{ext}}(\mathcal{S})$ 和 $\Phi^-_{\text{ext}}(\mathcal{S}')$ 确保对于 $z \notin B$、$f \in \mathcal{F}$、$|\{n \mid f(n) \in X_z\}| \leq 1$。
4. **可满足性传播器：**
   – 对于 $z \notin B \Rightarrow$ 在 $Q[x \leftarrow \hat{}z]$ 是一个重言式，因此该传播器是平凡的。– 对于 $z \notin B$：条件 $\Phi_{\text{prop}}(\mathcal{S})$（若 $z > b_{\max}$）或 $\Phi_{\text{prop}}(\mathcal{S}')$（若 $z < b_{\min}$）针对 Γ_I 中所有未传播单元的任意给定取值，$_z(\phi)$ 以及每个 $f(t) \in \mathcal{S}$，提供取值 $v_{f(t)}$，使得将这些取值赋给 X_z 中各自对应的单元后，$Q[x \leftarrow \hat{}z]$ 可满足。

接下来还需验证 定义 5 的不相交条件：

$$X_z \cap def(I) = \emptyset \text{ and } X_z \cap \bigcup_{z' \prec z} \Gamma_{I,z'}(\phi) = \emptyset.$$

对于 $z \in B$、$X_z = \emptyset$，因此这两个条件都显然成立。对于 $z > b_{\max}$，X_z中的任意元素都具有 $f(I(t[x \leftarrow \hat{}z]))$ 的形式，其中 $f(t) \in \mathcal{S}$。

– 为了证明 X_z 中的这个元素不属于 定义 (I)，令 $f'(n)$ 为 定义(I) 中一个元数非零、任意但固定的元素。我们证明 $f'(n) \neq f(I(t[x \leftarrow \hat{} z]))$。如果 f是一个不同于 f'的函数符号，那么这显然成立，因此只需证明 $f(n) \neq f(I(t[x \leftarrow \hat{} z]))$，这就需要证明 $I(n) \neq I(t[x \leftarrow \hat{} z])$。根据单元解释 I的构造， $n = I(a)$，其中 $a \in args(F \wedge \bigwedge_{x \in B} Q[x \leftarrow \hat{} x], f)$。

- 如果 $a \in args(F, f)$，那么冲突条件 $a \lesssim^{\sigma(f)^s} t[x \leftarrow \mathsf{bd}(B, s)]$ in$\Psi^+(B, \mathcal{S})$保证，对于每个基项 $a \in \mathsf{args}(F, f)$:

$$I \models a \lesssim^{\sigma(f)+} t[x \leftarrow \hat{b}_{\max}] \lesssim^{\sigma(f)+} t[x \leftarrow \hat{z}],$$

  其中，严格不等式成立是因为 t 是线性的，而且 $z > b_{\max}$。因此 $I(a) \neq I(t[x \leftarrow \hat{} z])$。
- 如果 $a \notin args(F, f)$，那么存在一个 $z_B \in B$，使得 $a \in args(Q[x \leftarrow \hat{} z_B], f)$。现在，由于 $z_B \leq b_{\text{最大}} < z$，同样有 $I \models a \lesssim^{\sigma(f)+} t[x \leftarrow \hat{} b_{\text{最大}}]$, 并且$I(a) \neq I(t[x \leftarrow \hat{} z])$。

– 为了证明该元素不会出现在任何满足 $z' \prec z$ 的 $\Gamma_{I, z'}(\phi)$ 中，令 f' 为一个未解释函数符号， $t' \in args(Q, f')$，并证明 $f'(I(t'[x \leftarrow \hat{} z'])) \neq f(I(t[x \leftarrow \hat{} z]))$。同样，只需分析 f 和 f' 是相同函数符号的情况，并证明 $I(t'[x \leftarrow \hat{} z']) \neq I(t[x \leftarrow \hat{} z])$。注意，由于 t' 和 t 是同一函数符号的参数， x 在 t 和 t' 中对应的系数相同。还要注意， $z' \prec z$ 和 $b_{\max} < z$ 蕴含 $z' < z$。因此，

$$I \models t'[x \leftarrow \hat{z}'] \lesssim^{\sigma(f)+} t'[x \leftarrow \hat{z}].$$

此外， $\Phi^+_{\text{ext}}(\mathcal{S})$确保

$$I \models t'[x \leftarrow \hat{z}] \lesssim^{\sigma(f)+} t[x \leftarrow \hat{z}].$$

因此 $I(t'[x \leftarrow \hat{} z']) \lesssim^{\sigma(f)+} I(t[x \leftarrow \hat{} z])$，这意味着不等式两边不可能相同。

情况 $z < b_{\min}$ 是对称的，只需使用 $\Psi^-(B, \mathcal{S}')$。请注意，在顺序 $\preceq$中，大于 $b_{\max}$ 的元素的向上传播块出现在小于 $b_{\min}$ 的元素的向下传播块之前，这一事实不会造成问题，因为传播仍然仅限于未出现在前面的单元。

因此，所有证书要求均已满足。根据 定理1， ϕ是可满足的。
■

定理定理3的证明：

假设 ϕ 是可满足的，并且满足 $(\mathcal{S}^*, \mathcal{S}'^*)$ 的 $ReqPivot$ 条件。令 J 为一个满足 $J \models \phi$ 的解释。对于每个函数 $f \in \mathcal{F}$，令 c_f 为 x 在其所有参数中的公共系数，并定义：

$$v^f_{\max} = \max\{\, J(t[x \leftarrow \hat{0}]) \mid f(t) \in \mathcal{T} \,\}, \qquad v^f_{\min} = \min\{\, J(t[x \leftarrow \hat{0}]) \mid f(t) \in \mathcal{T} \,\}.$$

极值条件 $\Phi^+_{\text{ext}}(\mathcal{S}^*)$ 和 $\Phi^-_{\text{ext}}(\mathcal{S}'^*)$ 可确保

– 对于所有 $f(t) \in \mathcal{S}^*$: $J(t[x \leftarrow \hat{0}]) = v^f_{\max}$, 且– 对于所有 $f(t) \in \mathcal{S}'^*$: $J(t[x \leftarrow \hat{0}]) = v^f$ 最小值。

现在选择

$$b_{\max} = \max\left\{ \left\lceil \frac{J(a) - v^f_{\max}}{c_f} \right\rceil \,\middle|\, a \in \text{args}(F, f), f \in \mathcal{F} \right\}$$

$$b_{\min} = \min\left\{ \left\lfloor \frac{J(a) - v^f_{\min}}{c_f} \right\rfloor \,\middle|\, a \in \text{args}(F, f), f \in \mathcal{F} \right\},$$

并设定 $B = [b_{\min}, $ 最大值$]$。验证其满足: J fi

1. **基础公式**: $J \models F \wedge \bigwedge_{x \in B} Q[x]$, 因为 $J \models \phi$。
2. **向上传播条件**: 对于特定的 $\mathcal{S}^*$, 我们有:
   – $\Phi^+_{\text{ext}}(\mathcal{S}^*)$ 根据定义成立, – $\Phi_{\text{prop}}(\mathcal{S}^*)$ 根据 $ReqPivot$ 条件成立, – 对于每个 $f(t) \in \mathcal{S}^*$ 和 $a \in \text{args}(F, f)$: 根据 $b_{\max}$ 的构造, 且由于 $J(t[x \leftarrow \hat{0}]) = v^f_{\max}$, 我们有:

$$J(a) \overset{\leq}{>}^{\sigma(f)+} c_f b_{\max} + v^f_{\max} = c_f b_{\max} + J(t[x \leftarrow \hat{0}]) = J(t[x \leftarrow \hat{b}_{\max}]),$$

   这恰好是 $\Psi^+(B, \mathcal{S}^*)$ 中的冲突条件。
   因此 $J \models \Psi^+(B, \mathcal{S}^*)$。
3. **向下传播条件**: 同样, 对于 $\mathcal{S}'^*$ 我们有:
   – $\Phi^-_{\text{ext}}(\mathcal{S}'^*)$ 成立, – $\Phi_{\text{prop}}(\mathcal{S}'^*)$ 由 $ReqPivot$ 条件成立, –对于每个 $f(t) \in \mathcal{S}'^*$ 以及 $a \in \text{args}(F, f)$: 根据 $b_{\min}$ 的构造,

$$J(a) \overset{\leq}{>}^{\sigma(f)-} c_f b_{\min} + v^f_{\min} = c_f b_{\min} + J(t[x \leftarrow \hat{0}]) = J(t[x \leftarrow \hat{b}_{\min}]),$$

   这是 $\Psi^-(B, \mathcal{S}'^*)$ 中的冲突条件 .
   因此 $J \models \Psi^-(B, \mathcal{S}'^*)$。

因此, J 满足

$$F \wedge \bigwedge_{x \in B} Q[x] \;\wedge\; \Psi^+(B, \mathcal{S}^*) \;\wedge\; \Psi^-(B, \mathcal{S}'^*),$$

因此也满足析取式 $\bigvee_{\mathcal{S}} \Psi^+(B, \mathcal{S})$ 和 $\bigvee_{\mathcal{S}'} \Psi^-(B, \mathcal{S}')$。

令 $N = \max(|b_{\min}|, |b_{\max}|)$。算法 1 从 $[0, 0]$开始对称地扩展区间。当达到第 $n = N$ 次迭代时, 区间变为 $[-N, N]$, 其中包含 B。在这次迭代中, 第 5 行将检查以下公式的可满足性:

$$F \wedge \bigwedge_{x=-N}^{N} Q[x] \;\wedge\; \bigvee_{\mathcal{S}} \Psi^+([-N, N], \mathcal{S}) \;\wedge\; \bigvee_{\mathcal{S}'} \Psi^-([-N, N], \mathcal{S}').$$

由于 $[-N, N] \supseteq B$, J也满足这个公式。因此, 算法以 SAT 终止。 ■