arXiv:2304.13422v1 [cs.AI] 26 Apr 2023

用于特征模型配置的基于合取查询的约束求解

Alexander Felfernig[0000−0003−0108−3146]、Viet-Man Le[0000−0001−5778−975X]和Sebastian Lubos[0000−0002−5024−3786]

软件技术研究所，格拉茨技术大学，奥地利格拉茨{
alexander.felfernig,vietman.le,slubos}@ist.tugraz.at

摘要。 特征模型配置可以基于各种类型的推理方法来支持，例如布尔可满足性求解、约束求解和答案集编程（ASP）。使用这些方法需要具备如何定义并求解底层配置问题的专业技术知识。本文展示了如何利用当今关系数据库系统通常支持的合取查询来求解约束满足问题（CSP），更具体地说，如何用于特征模型配置任务。这种方法使广泛应用的数据库技术能够用于求解配置任务，并为识别和解决不一致性提供新的算法方法。

关键词： 约束求解 · 基于知识的配置 · 特征模型 · 合取查询 · 关系数据库。

1 引言

特征模型（FM）可用于表示高可变软件和硬件系统的共性与可变性 [1,2,5,11,17]。这些模型可以通过图形方式进行定义，然后转换为相应的形式化表示，从而支持特征模型配置及相关推理任务，例如冲突检测（如由用户需求引发的冲突 [16]）和冲突解决（如解决空特征模型中现有的冲突 [3,15,18,24,26]）。

目前有多种推理方法可用于支持特征模型配置 [9,23]。首先，布尔可满足性求解 [14] 能够将各个特征直接转换为一组相应的布尔变量，用于指定特征的纳入或排除。事实证明，布尔可满足性求解是一种高效的推理方法，可支持特征模型分析和配置 [19]。其次，可以使用约束满足问题（CSP）来表示特征模型配置任务 [6]。与布尔可满足性求解相比，CSP 能够更“直接”地表示约束，从而表达逻辑等价和蕴含关系等典型约束类型。最后，答案集编程（ASP）支持定义面向对象的配置问题 [22]。在推理层面，这些

问题会被转换为相应的、基于布尔可满足性（SAT）的表示形式，随后可由 SAT 求解器处理。

上述所有知识表示都要求具备额外的技术专业知识，以表示特征模型配置知识并识别相应的配置（在推理层面上，具体形式可以是 SAT 求解、约束求解或答案集编程）。此外，还必须通过相应的应用程序接口纳入由此产生的知识库。除了 SAT 求解、约束满足问题和 ASP 之外，另一种方法是使用合取查询来表示特征模型配置问题。

本文的主要贡献如下：(a) 展示如何使用合取查询表示并求解配置任务；(b) 报告相应性能评估的初步结果；(c) 概述如何利用这类方法提高相关一致性管理任务（例如特征模型诊断）的效率。

本文其余部分的组织结构如下。在第 2 节中，我们介绍一个示例特征模型，全文将以此作为工作示例。在第 3 节中，我们展示如何基于合取查询支持配置任务，并介绍配置任务和特征模型配置的相应定义。第 4 节讨论冲突解决的替代方法。随后，我们给出性能分析的初步结果，将基于合取查询的配置与约束求解的效率进行比较（见第 5 节）。第 6 节讨论潜在的效度威胁。最后，在第 7 节中，我们通过讨论未来工作的开放问题总结全文。

2 示例特征模型

为便于演示，我们引入一个简化的调查软件特征模型（见图 1）。通常，这类模型中的特征以层次结构组织 [5] ，特征之间的关系决定了模型的整体结构：(a) 强制关系表示配置中必须包含特定特征，例如，每种调查软件都必须包含相应的支付；(b) 可选关系表示配置中可以包含特定特征，例如，在调查软件配置中，A/B 测试特征被视为可选特征；(c)互斥关系表示给定的一组子特征中必须且只能有一个包含在配置中（前提是包含其父特征），例如，支付要么需要许可证，要么无需许可证；(d) 或关系表示，在配置中包含父特征的前提下，可以从给定的一组子特征中选择性地包含特征，例如，问题（特征 QA）除单选题外，还可以包含多选题和多媒体式问题。

此外，所谓的跨树约束规定了与所讨论关系正交的其他属性：(a) 互斥两个特征之间的关系，规定这两个特征不得同时纳入：例如，无许可证特征不允许纳入 A/B 测试，反之亦然；

；(b) 依赖两个特征之间的关系，规定如果纳入某个特定特征 f_1，则必须纳入另一个特征 f_2，即 f_1 依赖于 f_2。例如，纳入 A/B 测试就必须同时纳入统计。

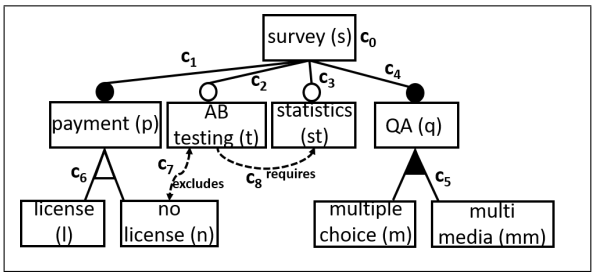

图 1。一个示例特征模型（$survey\ software$）。

为了支持对特征模型属性的推理操作，必须将这些模型转换为形式化表示。这类表示的示例包括布尔可满足性问题 [13,19]，约束满足问题 [7,12]，和答案集程序 [22]。下面将展示如何基于合取查询表示并求解特征模型配置任务。

3 基于合取查询的配置

将特征模型配置任务表示为合取查询，可以应用关系数据库技术来支持配置任务。下面将基于合取查询介绍特征模型配置任务的概念（见定义 1）和特征模型配置的概念（见定义 2） $F_{[c]}S$ 其中 F 可以是：（a）表示所有可能特征模型配置的表（即所有可能特征模型配置的枚举）；（b）一组单个表，其中每个表表示特征模型中的一个特征；以及（c）一组表示与特征模型的单个约束一致的元组的单个表。此外，$[c]$ 是一组必须满足的标准，其中包括特征模型约束以及配置器用户（客户）定义的要求。最后，S 表示所谓的投影，即应包含在查询结果中的属性（及其值）。

定义 1（特征模型配置任务）。特征模型（FM）配置任务可定义为合取查询 $F_{[c]}S$，其中 F 以表格形式（显式或隐式）表示可用特征，$[c]$ 是合取查询的选择部分，表示特征模型配置需要考虑的标准，S 是一组投影属性（特征），

即其设置应作为所展示配置结果的一部分予以包含的特征。此外， $c = cr \cup cf$ 其中 cr 表示一组客户需求（即特征包含和排除），而 cf 表示从特征模型推导出的一组约束。

在我们的示例中， $cf = \{c_0..c_8\}$ （c_0 是根约束，假设我们对空特征模型不感兴趣）。此外，假设特征模型配置器的用户既不希望支付许可证费用，也不需要基于多媒体的问题支持，则有 $cr = [s = 1, p = 1, n = 1, mm = 0]$。将特征模型表示转换为相应的基于合取查询的表示如表 1 所示。为简单起见，在此形式化过程中，我们假设采用单表表示，其中包含整个配置空间的完整枚举。

表 1。 将特征模型约束转换为查询（$cf = \{c_0..c_8\}$）。

约束	基于合取查询的表示
c_0	$F_{[s=1]}\{s..mm\}$
c_1	$F_{[s=1 \wedge p=1 \vee s=0 \wedge p=0]}\{s..mm\}$
c_2	$F_{[\neg t=1 \vee s=1]}\{s..mm\}$
c_3	$F_{[\neg st=1 \vee s=1]}\{s..mm\}$
c_4	$F_{[s=1 \wedge q=1 \vee s=0 \wedge q=0]}\{s..mm\}$
c_5	$F_{[q=1 \wedge (m=1 \vee mm=1) \vee q=0 \wedge m=0 \wedge mm=0]}\{s..mm\}$
c_6	$F_{[((\neg l=1 \vee n=0 \wedge p=1) \wedge (\neg (n=0 \wedge p=1) \vee l=1)) \wedge ((\neg n=1 \vee l=0 \wedge p=1) \wedge (\neg (l=0 \wedge p=1) \vee n=1))]}\{s..mm\}$
c_7	$F_{[\neg t=1 \vee \neg n=1]}\{s..mm\}$
c_8	$F_{[\neg t=1 \vee st=1]}\{s..mm\}$

给定特征模型配置任务 $F_{[c]}S$ 的定义，我们可以引入相应的特征模型配置的定义（见定义 2）。

定义 2（特征模型配置）。 针对特征模型（FM）配置 的特征模型配置任务 $F_{[c]}S$ 是一个元组，表示执行相应合取查询的结果。

遵循显式表示整个配置空间这一理念的“单表”方法，配置结果的一个示例是元组 $\{s = 1, p = 1, l = 0, n = 1, t = 0, st = 1, q = 1, m = 1, mm = 0\}$。

（*a*）所有可能的特征模型配置。表示特征配置空间的一种方法是明确枚举所有可能的配置[1] ，并将特征模型配置视为单表上的基本查询F （见表 2）。只有当配置空间较小，且相应的合取查询能够高效执行时，这种表示方式才有意义 F。在此背景下，返回某个配置的合取查询可以是：
$F_{[s=1,p=1,n=1,mm=0]}\{s, p, l, n, t, st, q, m, mm\}$ 结果为，

[1] 此枚举也可以基于合取查询执行，其中 [c] 表示特征模型约束集。

例如，某个特征模型配置为 $\{s = 1, p = 1, l = 0, n = 1, t = 0,\ st = 1,\ q = 1,\ m = 1,\ mm = 0\}$。在结构化查询语言（SQL）层面上，相关查询为： SELECT *s,p,l,n,t,st,q,m,mm* FROMF WHERE$s=1$ 和 $p=1$和 $n=1$ 和 $mm=0$。[2]

查询优化示例。提升对 F 的查询性能的一种方法，是假设 F 中的根特征（在我们的例子中是特征 调查 (s)）始终为 真 (1)，从而减少 F 中的元组数量。

表 2. 在一个表 F 中的显式配置空间描述，包括属性 s .. mm。

s	p	l	n	t	st	q	m	mm	m
1	1	1	0	0	0	1	1		0
1	1	1	0	0	1	1	1		0
.									..

(b) "每个特征一张表"表示。随着配置空间规模不断增大，需要采用隐式表示，因为枚举所有可能的特征组合是不可能的，或者会导致严重的性能问题。特征模型配置空间的另一种表示方法如表 3 所示，其中，图 1 中特征模型的每个特征分别由一张独立的表表示。例如，特征 p (支付) 由表 p 表示，该表包含属性 val 以及相应的表项 0 和 1，分别表示特征排除或特征包含。在此背景下，可以将返回特征模型配置的合取查询表述如下：
$F_{[s.val=1,p.val=1,n.val=1,mm.val=0]}\{s.val..mm.val\}$，其中 F 被视为笛卡尔积 $s \times p \times l \times n \times t \times st \times q \times m \times mm$ 的隐式表示。在 SQL 层面上，相应的查询为： SELECT *s.val.. mm.val* FROM *s.. mm* WHERE*s.val=1 and p.val=1 and n.val=1 and mm.val=0*。

查询优化示例。提高此类查询性能的方法包括：(a) 将根特征的域缩小为 $true$ (1)；(b) 将死特征的域缩小为$false$；(c) 将错误可选特征的域缩小为$true$ 。有关特征模型分析的详细信息，请参阅 Benavides 等人 [5]。

表 3. 隐式配置空间描述，其中每个特征由一个包含一个属性的单表表示， val 且表项为 0 和 1。

表	s	p	l	n	t	st	q	m	mm
属性 值 值 值									
域 0	,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1

[2] 为确保仅返回一个元组作为配置，我们假设使用如下查询设置： *LIMIT=1* （这是特定于数据库的）。

*(c)*基于“局部一致性”的表示。“每个特征一张表”表示法的一种替代方案是使用表来表达局部一致性属性。例如，约束*A/B* 测试 (*t*)要求统计 (*st*)，可以表示为相应的一致性表*t-st*（见表 4），该表表达了t和st的所有可能组合，并且仅考虑这两个特征之间的$requires$关系。

表 4。 表示局部一致特征赋值的隐式配置空间（a_i），例如，在表 *t-st* 中。

特征 t st	t	
a_1	1	1
a_2	0	1
a_3	0	0

表 5。 由表 n-t 表示的 $excludes$ 的局部一致特征赋值 a_i，特征 n 与 t 之间的关系。

特征 n t	t	
a_1	0	1
a_2	1	0
a_3	0	0

由于表 4 和表 5 包含特征 t，因此我们需要确保在最终配置中，t 恰好具有一个值——这可以通过将连接条件整合到合取查询中来实现。在 SQL 中，相应的查询为： SELECT n-t.n,n-t.t,t-st.st FROM n-t,t-st WHERE n-t.t=t-st.t。使用这种表示方式，特征名称被视为表属性，例如，特征 t （*A/B* 测试）既被视为表 n-t 的属性，也被视为表 t-st 的属性。

查询优化示例。遵循约束求解中 *k*-一致性 的概念 [25], ，可以通过考虑更多特征模型约束来减少约束表（例如 n-t）中的元组数量。在极端情况下，这将导致全局一致性 [25] ，从而确保每个约束表只包含能够成为至少一个配置一部分的元组（表示特征值组合）。对于复杂的特征模型（FM）而言，全局一致性是不可能实现的，因为这基本上要求枚举所有可能的配置。

总而言之，将特征模型配置任务表示为合取查询有三种基本方案：(a) 枚举所有可能的配置（见表 2）；(b) 使用隐式表示，为每个特征建立一张表（见表 3）；或者 (c) 使用表来表示局部一致的特征设置。

4 冲突解决的新方法

在配置场景中，合取查询 $F_{[c]}S$ 可能产生空集。例如，如果特征模型为空，即无法识别出任何解，或者给定的客户需求集与特征模型约束 [8,10] 不一致，就可能出现这种情况。在此类情况下，我们需要识别出必须调整的最小约束集（位于特征模型或客户需求中），以恢复一致性。在特征模型配置的背景下，已有一些方法支持相应的特征模型诊断操作 [4,15]。应用诊断算法的另一种方案是直接利用基于表的表示中包含的信息。表 6 展示了一个例子。假设存在一组客户需求 cr，我们可以将这些需求直接与一组潜在配置进行比较。如果 cr 与某个配置 $conf_i$ 不一致，则与 $conf_i$ 不一致的 cr 特征包含（排除）构成一个诊断 Δ，其中 $cr - \Delta$ 与 $conf_i$ 一致。

表 6.诊断的直接推导。

特征	s	p	l	n	t	st	q	m	m	m	
cr	1	1	0	1	1	1	1	1			0
$conf_1$	1	1	0	1	0	1	1	1			0
$conf_2$	1	1	1	0	1	1	1	1			0
..	.										..

通过这种方式，我们无需应用相应的诊断算法即可确定诊断结果。在表 6 所示的示例中，我们能够识别出两个潜在诊断结果：$\Delta_1 = \{t = 1\}$ 和 $\Delta_2 = \{l = 0, n = 1\}$。应用 Δ_1 基本上意味着告知用户可能排除特征 t，这可能进而导向一个一致配置。

5 性能分析

我们基于从 S.P.L.O.T. 特征模型存储库 [20]中选取的四个真实世界特征模型，对比了三种将特征模型配置任务表示为合取查询并结合约束求解的方法的性能。表 7 概述了所选的特征模型。由于空间复杂度，对于 $WebArch$，无法在合理的时间限制内确定所有配置。

对于每个特征模型，我们随机合成[3] 并收集了25,000条用户需求，这些需求涵盖特征模型中40%至60%的叶特征。我们

[3] 为确保结果的可复现性，我们为随机数生成器使用了种子值 141982L。

应用系统抽样技术 [21] 选择了10条无解用户需求，以及10条至少有一个解的用户需求。在表8中，每种设置均展示了相应方法在基于这20条用户需求执行查询后的平均运行时间。我们使用 Choco Solver[4]作为推理求解器，并使用 HSQLDB[5] 作为内存中的关系数据库管理系统。所有报告的实验均在配备Apple M1 Pro（8核）和16 GB内存的设备上运行，HSQLDB的最大缓存大小为4 GB。

表 8 展示了对所选特征模型进行评估的结果，这些特征模型分别表示为：(a) 对<熟悉的术语>所有配置</熟悉的术语>进行显式枚举；(b) 对特征模型配置空间进行隐式表示（每个特征一张表）；(c) 使用单个表表示局部一致的特征赋值的隐式表示（每个约束一张表）；以及 (d) 约束满足问题（CSP）。初步评估表明，对于小型特征模型，各种方法的运行时间相近；而对于更复杂的模型，基于合取查询的方法运行时间则显著更长。总体而言，这些性能评估的初步结果表明，基于合取查询的方法适用于特征模型配置任务。

表 7。 所选特征模型的属性（IDE=IDE产品线、DVS=数字视频系统、DELL=DELL笔记本电脑特征模型、WebArch=Web架构）。

特征模型	IDE	DVS	DELL	WebArch
#特征	14	26	47	77
#叶特征	9	16	38	46
#层次约束 11		25	16	65
#跨树约束	2	3	105	0
#配置	80	22,680	2,319	-

表 8。合取查询与基于约束的特征模型配置的平均运行时间（$msec$）。

特征模型	IDE	DVS	DELL	WebArch
所有配置	0.78	9.43	6.49	-
每个特征一张表	0.49	0.45	2.53	327.75
每个约束一张表 0.51	0.78		3.32	294.55
CSP	0.73	0.78	1.09	1.1

[4] choco-solver.org [5]
hsqldb.org

6 效度威胁与开放问题

在本文中，我们展示了如何应用合取查询来识别特征模型配置。我们给出了对特征模型的显式和隐式基于表的表示进行初步性能分析的结果，并比较了合取（数据库）查询与相应的基于约束求解器的实现的性能。初步结果令人鼓舞，表明我们的方法在搜索效率方面具有适用性。我们意识到，还需要使用工业数据集和不同类型的知识表示进行更深入的评估，以进一步了解我们方法的适用性。我们还概述了在特征模型配置场景中处理不一致性的替代方法。我们也清楚，仍需开展相关评估，才能详细估计诊断和冲突搜索在运行时性能方面的潜在改进。

7 结论与未来工作

我们提出了一种基于合取查询的特征模型（FM）配置方法。在此背景下，我们比较了三种替代性的基于合取查询的知识表示方法。借此，我们为布尔可满足性求解、约束求解和 ASP 提供了一种替代方案，该方案可以减少新技术集成所带来的开销。我们的性能分析初步结果令人鼓舞，但我们也清楚，诸如 SAT 求解等技术可能优于合取查询。此外，我们还概述了如何利用基于表的表示来实现高效的冲突解决。

未来工作中还有一些开放问题。首先，我们将把性能分析扩展到与更广泛的工业特征模型进行比较。我们还将把 SAT 求解和 ASP 纳入评估范围。此外，我们计划分析融合不同领域的潜力，例如，研究约束求解中的方法（如前向检查）能够在多大程度上帮助提高合取查询的效率。反过来，我们将分析关系数据库（以及机器学习）中的搜索技术能够在多大程度上进一步推动 SAT 求解、约束求解和 ASP 领域的发展。最后，我们希望分析相变的性质，即何时应从显式搜索空间表示转向隐式搜索空间表示。

致谢。 本文所述工作是在研究项目 ParXCel (面向可扩展约束求解的机器学习与并行化)框架内开展的。该项目由奥地利研究促进机构（FFG）资助，项目编号为 880657。

参考文献

1. Acher, M., Temple, P., J

´ ´

VARY-

LATEX 12

ezequel, J.M., Galindo, J., Martinez, J., Tiadi, T.：《满足约束的论文变体学习》。载于：软件密集型系统可变性建模国际研讨会。第 83–88 页。西班牙马德里（2018） 2. Apel, S., K¨astner, C.：面向特征的软件开发概述。《对象技术杂志》**8**(5)，49–84（2009） 3. Bakker, R., Dikker, F., Tempelman, F., Wogmim, P.：过度确定约束满足问题的诊断与求解。载于：IJCAI-93 论文集。第 276–281 页。Morgan Kaufmann（1993） 4. Benavides, D., Felfernig, A., Galindo, J., Reinfrank, F.：特征建模与产品配置中的自动化分析。载于：第 13 届软件复用国际会议（ICSR 2013）。第 160–175 页。LNCS 第 7925 卷，比萨，意大利（2013） 5. Benavides, D., Segura, S., Ruiz-Cortes, A.：20 年后的特征模型自动化分析：文献综述。《信息系统》**35**，615–636（2010） 6. Benavides, D., Trinidad, P., Cort´es, A.：使用约束编程对特征模型进行推理。载于：Chu, W.C., Juzgado, N.J., Wong, W.E.（编）第 17 届软件工程与知识工程国际会议（SEKE'2005）论文集，中国台湾台北，2005 年 7 月 14–16 日。第 677–682 页（2005） 7. Benavides, D., Trinidad, P., Ruiz-Cortez, A.：特征模型上的自动推理。载于：高级信息系统工程国际会议。LNCS，第 3520 卷，第 491–503 页。葡萄牙波尔图（2005） 8. Felfernig, A., Boratto, L., Stettinger, M., Tkalcic, M.：《群体推荐系统》。Springer（2018） 9. Felfernig, A., Hotz, L., Bagley, C., Tiihonen, J.：《基于知识的配置——从研究到业务案例》。Elsevier（2014） 10. Felfernig, A., Schubert, M., Zehentner, C.：不一致约束集的高效诊断算法。《工程设计、分析与制造中的人工智能》（AIEDAM）**26**(1)，53–62（2012） 11. Felfernig, A., Le, V.M., Popescu, A., Uta, M., Tran, T.N.T., Atas, M.：特征建模与配置中的推荐系统和机器学习概述。VaMoS ' 21，ACM，美国纽约州纽约市（2021） 12. Francesca Rossi, Peter van Beek, T.W.：《约束编程手册》。Elsevier（2006） 13. Gomes, C., Kautz, H., Sabharwal, A., Selman, B.：可满足性求解器。《知识表示手册》（3），89–134（2008） 14. Gu, J., Purdom, P.W., Franco, J., Wah, B.W.：可满足性（sat）问题的算法：一项调查。载于：DIMACS离散数学与理论计算机科学丛书。第 19–152 页。美国数学学会（1996） 15. Hentze, M., Pett, T., Th¨um, T., Schaefer, I.：用于特征模型缺陷分析的超解释。载于：第 15 届软件密集型系统可变性建模国际工作会议。VaMoS'21，计算机协会，美国纽约州纽约市（2021） 16. Junker, U.： QUICKXPLAIN：过度约束问题的首选解释与松弛。载于：第 19 届全国人工智能会议。第 167–172 页。AAAI 2004，AAAI（2004） 17. Kang, K., Cohen, S., Hess, J., Novak, W., Peterson, S.：面向特征的领域分析（FODA）——可行性研究。技术报告 SEI-90-TR-21（1990）

18. Le, V.M.、Felfernig, A.、Uta, M.、Benavides, D.、Galindo, J.、Tran, T.N.T.：
RECTDEBUG 2021 ：

特征模型的自动化测试与调试。载于：IEEE/ACM第43届软件工程国际会议：新思想与新兴成果（ICSE-NIER）。第81–85页。IEEE/ACM（2021） 19. Mendonca, M.、Wasowski, A.、Czarnecki, K.：基于SAT的特征模型分析很容易。载于：Muthig, D.、McGregor, J.D.（编）：SPLC 2009。第446卷，第231–240页。ACM（2009） 20. Mendonca, M.、Branco, M.、Cowan, D.：S.P.L.O.T.：软件产品线在线工具。载于：第24届ACM SIGPLAN会议伴随论文集：面向对象编程系统、语言与应用。第761–762页。OOPSLA '09，ACM，美国纽约州纽约市（2009） 21. Mostafa, S.A.、Ahmad, I.A.：系统抽样的最新进展：综述。《统计理论与实践杂志》**12**(2)，290–310（2018）。
https://doi.org/ 10 1080 15598608 2017 1353456 。/。。。
22
Myllarniemi, V.、Tiihonen, J.、Raatikainen, M.、Felfernig, A.：使用答案集编程进行特征模型表示与配置。载于：第16届配置国际研讨会。第1–8页。CEUR，塞尔维亚诺维萨德（2014） 23. Popescu, A.、Polat-Erdeniz, S.、Felfernig, A.、Uta, M.、Atas, M.、Le, V.、Pilsl, K.、Enzelsberger, M.、Tran, T.：约束求解中的机器学习技术概述。《智能信息系统杂志》**58**(1)，91–118（2022） 24. Reiter, R.：从第一性原理出发的诊断理论。《人工智能杂志》**32**(1)，57–95（1987） 25. Rossi, F.、van Beek, P.、Walsh, T.：《约束编程手册》。Elsevier，荷兰阿姆斯特丹（2006） 26. White, J.、Benavides, D.、Schmidt, D.、Trinidad, P.、Dougherty, B.、Ruiz-Cortes, A.：特征模型配置的自动化诊断。《系统与软件杂志》**83**(7)，1094–1107（2010）