

结合 DOEC 极小化策略的 SAT 求解极小碰集方法

王荣全 欧阳丹彤 王艺源 刘思光 张立明
(吉林大学计算机科学与技术学院 长春 130012)
(符号计算与知识工程教育部重点实验室(吉林大学) 长春 130012)
(wangrongquanjl@163.com)

Solving Minimal Hitting Sets Method with SAT Based on DOEC Minimization

Wang Rongquan, Ouyang Dantong, Wang Yiyuan, Liu Siguang, and Zhang Liming
(College of Computer Science and Technology, Jilin University, Changchun 130012)
(Key Laboratory of Symbolic Computation and Knowledge Engineering (Jilin University), Ministry of Education, Changchun 130012)

Abstract In the model-based diagnosis, the minimal conflict sets is employed to find all corresponding minimal hitting sets. Therefore, how to improve the efficiency of obtaining all minimal hitting sets is a great important issue. In this paper, we propose a new method called SAT-MHS, which is mainly based on the transform method and the set-degree of element coverage(DOEC) strategy. There are two main innivations of this paper. Firstly, SAT-MHS transforms a hitting set problem into the SAT problem, which is a new direction to solve this problem. All the minimal conflict sets are expressed in the form of clauses as the input CNF of the SAT. Secondly, compared with the previous sub-superset detecting minimization (SSDM) strategy, the proposed DOEC strategy can effectively reduce both of solution space and the number of iterations. In details, the time consumption of DOEC is along with the number of all minimal conflict sets, not depending on the size of the given problem. Experimental results show that SAT-MHS outperforms the previous state-of-the-art method and the time speed ratio of SAT-MHS rises to 10-20 times, especially for some large instances. Moreover, we also carry out extensive experiments to demonstrate that the performance of DOEC strategy is better than SSDM, even up to about 40 times.

Key words model-based diagnosis; minimal hitting sets; satisfiability problem (SAT); hitting minimization; set-covering

摘 要 在基于模型诊断中,诊断解通常是根据极小冲突集合簇进行相应的计算得到所有的极小碰集,所以提高极小碰集的求解效率是模型诊断的核心问题.因此提出结合基于元素覆盖集合度(degree of element coverage, DOEC)极小化策略的 SAT 求解极小碰集的方法 SAT-MHS(satisfiability problem-minimal hitting sets).首先,方法 SAT-MHS 将碰集求解问题转换成 SAT 问题,即把所有的冲突集合以子句形式表示成 SAT 的输入 CNF 进行迭代求解.其次,提出比现有的基于子超集检测极小化策略(sub-superset detecting minimization, SSDM)更为高效的 DOEC 极小化策略进行极小化处理.由实验数据可见,与 SSDM 极小化策略相比,其优点是缩减了求解空间和迭代求解次数,尤其当求解规模较大问题时,其极小化效率越高.主要是因为其极小化不会随着待求解问题规模的增加而增加,而是只与冲

收稿日期:2016-11-10;修回日期:2017-07-04
基金项目:国家自然科学基金项目(61672261,61502199,61402196,61373052);浙江省自然科学基金项目(LY16F020004)
This work was supported by the National Natural Science Foundation of China (61672261, 61502199, 61402196, 61373052) and the Natural Science Foundation of Zhejiang Province of China (LY16F020004).
通信作者:张立明(limingzhang@jlu.edu.cn)

冲突集合簇的大小相关,因此时间复杂度较低.实验结果表明,对于一些较大的实例,与目前效率最好的 Boolean 方法相比,SAT-MHS 方法高效且易于实现,求解速度能提高 10~20 倍,DOEC 极小化策略对比传统 SSDM 极小化策略能达到 40 倍左右.

关键词 基于模型诊断;极小碰集;可满足性问题;碰集极小化;集合覆盖

中图法分类号 TP18

人工智能专家 Console 等人^[1]指出基于模型诊断(model based diagnosis, MBD)的研究对整个人工智能领域的研究有着重要的作用.学者 Struss^[2]称基于模型诊断是人工智能领域极富有挑战性和具有检验性的课题.基于模型诊断的核心步骤是根据系统的描述和观测,得到极小冲突集合簇^[3],然后以其为输入并利用碰集(hitting set, HS)的求解方法计算得到系统所对应的极小诊断解即极小碰集(minimal hitting set, MHS),因此求解 MHS 问题是模型诊断中比较核心的问题之一^[4].此外很多的实际和理论问题都可以转化为 MHS 问题^[5-6],如极小集合覆盖和极小顶点覆盖等极小覆盖问题均是极小碰集的特殊形式、师生选课问题、蕴涵推理问题以及智能规划的核心问题.

碰集(hitting set, HS)问题最早是由加州大学伯克利分校的计算机专家 Karp 在 1972 年提出,随后 MBD 专家 Reiter^[7]首次提出了一种求解极小碰集的方法 HS-Tree,但该方法基于枚举树并遍历整棵枚举树导致效率较低,且会因剪枝丢失解.因此, Greiner 等人^[8]提出 HS-DAG 方法,优化并改进了 HS-Tree 方法的剪枝策略,虽然解决因剪枝而丢失解,但因新的剪枝规则的应用导致计算复杂度大幅度增加.随着对极小碰集问题的深入探索研究,许多新的求解方法陆续被提出.2001 年, Wotawa^[9]通过改进 HS-Tree 方法提出了 HST-Tree 方法,虽然效率提升,但其通用性降低,并未彻底解决丢解的问题,空间复杂度较高;2003 年,姜云飞等人^[10]提出布尔代数方法,从解的完备性和方法的高效性来看,该方法是目前较好的求解方法之一,因为其不需要建立较复杂的树结构,只需递归计算字符串操作,比之前所有的方法在空间复杂度和时间复杂度都有降低,在提高求解效率的同时简化了数据结构;2004 年,欧阳丹彤等人^[11]改进了 HS-DAG 方法,通过剪枝无解空间达到提高效率,但因数据结构复杂使方法实现繁琐;2006 年,赵相福等人^[12]提出了 HSSE-Tree 方法,其利用与元素相关联的冲突集个数来计算碰集并在标准的 SE-tree 上对节点种类进行标

识,从而提高搜索效率;2010 年,陈晓梅等人^[13]提出了 BNB-HSSE 方法,采用分支定界法与集合枚举法相结合的方法来降低求解规模;2011 年,张立明等人^[14]提出了基于动态极大度的 DMDSE-Tree 方法,其通过结点数概念较早地生成极小碰集,减少了生成结点的数量;2012 年, Pill 等人^[15]对布尔代数方法改进了部分规则,有效提升了该方法对特定问题的求解效率;2015 年,王艺源等人^[16]提出了利用 CSP 求解器求解极小碰集的 CSP-MHS 方法,其思想是通过将求解极小碰集问题转为约束满足问题,然后调用 CSP 求解器进行求解,并根据约束可满足问题的特点去求解具有某些特征的极小碰集.

以上是求解 MHS 的完备方法,随着近似求解策略的不断产生和完善,众多的非完备极小碰集求解方法也陆续被提出^[17-18].其中大多采用的是随机搜索策略,其特点是牺牲了解的完备性从而能针对规模较大的问题可以在较短时间进行求解.此外,随着并行和分布式的发展,很多学者也开始投入到用并行或分布式的方法去求解极小碰集问题^[19-20].

本文在对 SAT 问题的深入理解与分析的基础上并与极小冲突集合簇求解极小碰集过程的本质相结合,提出利用目前较为成熟的 SAT 求解器进行碰集求解,把极小冲突集合簇转换表示成 SAT 求解器的可求解的 CNF 形式.在每次求解出一个碰集之后利用首次提出的 DOEC 极小化策略进行极小化,在每次极小化的过程中,先把碰集通过元素覆盖集合度的势找出其中所有的冗余元素,并对其进行删除,从而高效地实现碰集极小化.最后把已经极小化产生的极小碰集以取反的隔离子句的形式添加到 CNF 中并继续调用 SAT 求解器进行求解,直到 SAT 求解器返回不满足,即问题求解.由于每个求出的碰集都被极小化过,并以取反的隔离子句的形式添加到求解器的 CNF 中,所以在下次求解的过程中不会求解极小化后的极小碰集的求解空间,从而实现了对部分求解空间的屏蔽.但该方法不会屏蔽极小解,从而保证求出所有的解,保证了解的完备性.这种方法的主要优点是:1)因采用宽度优先求解方式,与上述

提到的方法相比能较快地得到解;2)减少了对大量非极小化碰集的无用求解,大幅提高方法效率;3)根据 SAT 求解器的是否满足即可判断求解是否结束;4)仅使用树结构来描述分析问题,实现过程中仅使用数据结构简单的数组描述,所以实现容易。

1 预备知识

在本节,首先介绍下基于模型诊断的基本概念及其相关的基础知识。

1.1 基于模型诊断的相关概念

定义 1^[7]. 一个系统可定义为一个三元组 $(SD, COMPS, OBS)$, 其中:

1) SD 为系统描述, 是一阶谓词公式集合, SD 描述了待诊断系统的相关背景知识;

2) $COMPS$ 是系统组成部件的集合, 是一个有限常量集;

3) OBS 为观测集, 是一阶谓词公式的有限集。

定义 2^[7]. 冲突集 CS 是一个部件集 $\{c_1, c_2, \dots, c_n\} \subseteq COMPS$ 当且仅当 $SD \cup OBS \cup \{\neg AB(c_1), \neg AB(c_2), \dots, \neg AB(c_n)\}$ 是不一致的. 其中 AB 为一元谓词, 表示“abnormal”. $AB(c)$ 为真, 当且仅当 c 异常, 且 $c \in COMPS$.

称冲突集是极小冲突集 (MCS), 当且仅当该冲突集的任意真子集都不是冲突集。

定理 1^[7]. Δ 是 $(SD, COMPS, OBS)$ 的一个极小诊断, 当且仅当 Δ 是 $(SD, COMPS, OBS)$ 的极小冲突集的极小碰集。

1.2 SAT 问题的相关定义

可满足问题的一些基本定义如下:

定义 3^[21]. 文字. 对于变量集合 $X = \{x_1, x_2, \dots, x_n\}$, 文字 l_j 是变量 x_i 或者它的否定 $\neg x_i$.

定义 4^[21]. 子句. 子句 C_j 是文字 l_j 的析取, $C_i = l_1 \vee l_2 \vee \dots \vee l_m$.

定义 5^[21]. 合取范式 (conjunctive normal form, CNF). CNF C 是子句的合取, $C = C_1 \wedge C_2 \wedge \dots \wedge C_k$.

定义 6^[21]. SAT 问题的定义如下: 对于 n 个布尔变量 $X = \{x_1, x_2, \dots, x_n\}$, 找到一组变量的赋值, 满足在各个子句中都存在一个文字取真, 并且这组赋值使得所有的子句都能取真. 若找到这样的赋值, 则称该 SAT 问题是可满足的 (SAT); 如果没有任何一组变量赋值使得 SAT 可满足, 则称该 SAT 问题是不可满足的 (UNSAT).

2 集合极小化方法

集合极小化方法在模型诊断中有着举足轻重的作用并且是必备的求解步骤. 在模型诊断中的诊断解的求解过程中, 其占据大部分求解时间. 目前对该问题的求解策略还主要停留在求解效率和时间较为复杂的 SSDM 极小化策略, 因此提出新的极小化策略会在很大程度上提高诊断求解时间和效率. 本节提出 DOEC 极小化策略来进行极小化处理, 该策略的优点是其求解复杂性和效率不会随着要求解的问题的碰集个数增加而导致极小性判定时间急剧增加, 而是只与待求解的初始极小冲突集合簇的大小相关。

2.1 相关定义

首先给出一些与极小化策略相关的定义。

定义 7^[10]. 设 F 是一个集合簇, $F = \{S_1, S_2, \dots, S_i, \dots, S_n\}$, 称 H 为 F 的一个碰集 (HS), 如果 H 满足 2 个条件:

1) $H \subseteq \bigcup_{S_i \in F} S_i$;

2) 对于任意一个 $S_i \in F$, 都有 $HS \cap S_i \neq \emptyset$.

称 F 的一个碰集 H 为极小碰集 (MHS), 当且仅当 H 的任意真子集都不是 F 的碰集。

定义 8. 若 $num \in S_i$, 则称元素 num 覆盖集合 S_i . 元素 num 覆盖集合 S_i 记为 $Cover(num, S_i) = \{num | num \in S_i, num \in H\}$. 含有元素 num 的碰集 H 和冲突集合 S_i 取交集后共有的元素个数称为元素 num 覆盖集合 S_i 的度, 简称元素覆盖集合度, 记为 $DOEC(Cover(num, S_i)) = \{num | num \in (S_i \cap H)\}$.

定义 9. 对于碰集 H 中的元素 num , 如果其覆盖的所有的 S_i , 均有 $DOEC(Cover(num, S_i)) \geq 2$, 则称此 num 是该 H 的冗余元素。

根据以上定义, 我们易得以下推论:

推论 1. 如果碰集中不存在冗余元素, 则该碰集是极小碰集。

证明. 假设极小碰集中存在冗余元素, 则根据冗余元素定义, 可知元素覆盖的所有 S_i , 均有 $DOEC(Cover(num, S_i)) \geq 2$, 即此元素所覆盖的集合也被极小碰集中其他元素覆盖, 即当前极小碰集删除此元素后仍为碰集, 与极小碰集定义矛盾. 证毕。

针对上面的定义和推论, 我们通过实例对提到的元素集合覆盖度、冗余元素、碰集和极小碰集的概念进行进一步说明和阐述。

例 1. 冲突集合簇 $CS = \{\{2, 4, 5\}, \{1, 2, 3\}, \{1, 3, 5\}, \{2, 4, 6\}, \{2, 4\}, \{2, 3, 5\}, \{1, 6\}\}$, 要求求其所有极小碰集. 首先要极小化, 将 CS 中为某些冲突集

合的超集的集合删除,即删除 $\{2,4,5\}$ 和 $\{2,4,6\}$,因为它们是集 $\{2,4\}$ 的超集,这样得到一个新的极小化的冲突集合簇 $CS = \{\{1,6\}, \{2,4\}, \{1,2,3\}, \{1,3,5\}, \{2,3,5\}\}$. 通过去除超集,这样可以减少方法处理的节点数目和冲突集的数量,从而进一步提高求解效率. 该集合簇的所有的碰集为 $HS = \{\{1,2\}, \{1,3,4\}, \{1,4,5\}, \{2,3,6\}, \{2,5,6\}, \{2,3,4,5,6\}, \{2,4,5,6\}, \{3,4,6\}, \{3,4,5,6\}\}$. 其中 $\{2,3,4,5,6\}, \{2,4,5,6\}, \{3,4,5,6\}$ 为非极小碰集,其余的为极小碰集.

例如以是碰集但是是非极小碰集的 $\{2,4,5,6\}$ 来说明解析,首先对该非极小碰集进行元素集合覆盖度初始化并计算,计算的覆盖度如下: $\{1,6\}:1, \{2,4\}:2, \{1,2,3\}:1, \{1,3,5\}:1, \{2,3,5\}:2$,然后对 $\{2,4,5,6\}$ 的每个元素进行覆盖度访问.

对元素 2 来说,它覆盖的集合有 $\{2,4\}, \{1,2,3\}, \{2,3,5\}$ 其中 $\{1,2,3\}$ 的覆盖度为 1 小于 2,所以元素 2 不是冗余元素,不能将其删除;接着是 4,它覆盖的集合只有 $\{2,4\}$ 且覆盖度为 2,所以根据冗余元素定义,其是冗余元素,可以将其删除;接着是 5,它覆盖的集合有 $\{1,3,5\}:1, \{2,3,5\}:2$,其中 $\{1,3,5\}$ 的覆盖度为 1 小于 2,所以元素 2 不是冗余元素,不能将其删除;最后是元素 6,它覆盖的集合有 $\{1,6\}$ 且覆盖度为 1,所以不能将其删除. 所以将 $\{2,4,5,6\}$ 中唯一的冗余元素 4 删除之后,该碰集为极小碰集.

2.2 节对结合元素覆盖集合度的 DOEC 碰集极小化策略进行详细介绍,然后在第 4 节对这 2 种极小化策略实验结果进行比较.

2.2 结合 DOEC 的碰集极小化策略

本节将给出 DOEC 的极小化策略,该方法的主要思想是:检查待检测碰集中的每个元素的集合覆盖情况,如果它覆盖的所有原始冲突集合已经都被该碰集中的其他元素覆盖过了,即它所覆盖集合的覆盖度均大于 1,则该元素是冗余元素,可以将其删除. 不断的继续检查碰集中的每个元素覆盖集合的覆盖度情况,直到被检测的碰集的所有元素都被检测完毕,剩下元素构成新的碰集一定是极小碰集.

算法 1. DOEC.

输入: $HS = \{e_1, e_2, \dots, e_i, \dots, e_n\}$;

/* 待判定的碰集 */

$MHS_s = \{MHS_1, MHS_2, \dots, MHS_m\}$;

/* 已知的极小冲突集合簇 */

输出: 极小化后的碰集 HS .

① $Covers := \emptyset$;

② for each $e_i \in MHS_j$ do

③ $Covers(MHS_j) := 0$;

④ end for

⑤ for each $e_i \in HS$ do

⑥ if ($e_i \in MHS_j$) then

/* 是否在极小冲突集合簇 MHS_j 中 */

⑦ $Covers(MHS_j)++$;

/* 对应的集合覆盖度累加 1 */

⑧ end if

⑨ end for

⑩ for each $e_i \in HS$ do

⑪ if ($\exists Covers(MHS_j) == 1$) then

/* MHS_j 只被 e_i 覆盖,终止 e_i 的检测 */

⑫ break;

⑬ else

⑭ $count = 0$;

⑮ for each $e_i \in MHS_j$ do

⑯ $count = count + 1$;

/* $count$ 记录 e_i 覆盖的集合个数 */

⑰ end for

⑱ $num = 0$;

⑲ for each $e_i \in MHS_j$ do

⑳ if $Covers(MHS_j) > 1$ then

㉑ $num = num + 1$;

/* num 统计被 e_i 覆盖集合的集合覆盖度大于 1 的个数 */

㉒ end if

㉓ end for

㉔ if ($num == count$)

/* 如果 e_i 覆盖的集合均已经被覆盖,则其为冗余元素 */

㉕ $HS := HS / \{e_i\}$;

/* 从 HS 中删除 e_i */

㉖ end if

㉗ for each $e_i \in MHS_j$ do

㉘ $Covers(MHS_j)--$;

/* 把包含该元素的集合的覆盖度减 1 */

㉙ end for

㉚ end if

㉛ end for

㉜ return HS .

该方法使用数组保存每个原始的极小冲突集合 (MHS_j) 被当前 HS 中元素覆盖的情况 (行①~④); 对被检测的 HS 中每个元素的集合覆盖度进行统计,以便后续判断该 HS 中该元素是否为冗余元素 (行⑤~⑨); 然后对被检测的 HS 中每个元素所

覆盖集合度的情况进行统计,根据不同的集合覆盖度情况加以相应的处理(行⑩~⑳).其中行⑪~⑫是对集合覆盖度小于2的元素即非冗余元素进行跳过处理,行⑭~⑯是查找冗余元素并对其从当前待检测碰集中执行删除操作,行⑰~⑱是把对应的冗余元素的相应覆盖的集合的覆盖度进行更新操作,行㉔是输出碰集经过极小化方法后得到的极小碰集.

该方法和以前 SSDM 的极小化策略相比,其优点是碰集极小化判定时间不会随着极小碰集个数的增加,极小化时间随之急剧增加,而是只与输入的极小冲突集合簇的大小有关,大大提高了极小化求解效率.实验数据结果对比见 4.1 节.

3 利用 SAT 求解极小碰集的方法

3.1 将求解碰集问题转换成 SAT 问题

本节将介绍如何将极小冲突集合簇转换成 CNF 的形式.

要想把碰集问题转换成 SAT 问题,首先是如何把要解决的集合形式化表示成 SAT 问题求解器能够读取和处理的 CNF 形式,实质上,只要把所有的冲突集合簇表示成 CNF 形式,然后把每个冲突集合中的元素用 SAT 中的文字来表示,每个冲突集合用一个子句来表示,举例说明如下:

例 2. 以极小冲突集合簇 $MCS = \{\{1, 6\}, \{2, 4\}, \{1, 2, 3\}, \{1, 3, 5\}, \{2, 3, 5\}\}$ 为例,介绍一下如何对极小冲突集合簇进行形式化表示,得到相对应的 CNF 文件描述.

冲突集合簇(MCS)形式化 CNF 描述如下:

- ① p cnf 6 5
- ② 1 6 0
- ③ 2 4 0
- ④ 1 2 3 0
- ⑤ 1 3 5 0
- ⑥ 2 3 5 0

上述 CNF 描述例子语法解释如下:

行①中的“p cnf”是关键字,“6”代表集合变量元素的个数,“5”代表对应的极小冲突集合的个数.行②~⑥中每行是一个极小冲突集合,其中除 0 以外的数字代表极小冲突集合中存在的元素且均为正,每行的结尾数字 0 代表极小冲突集合结束符.

碰集的定义要求碰集覆盖所有的极小冲突集合簇,此点和 SAT 问题中的所有的析取范式都必须为真然后构成一个合取范式为真是相同的.可见碰集问题和 SAT 问题的实质是集合覆盖问题,只是有些

概念的定义范围略有不同,所以可以将一个碰集的求解问题转化为 SAT 问题进行求解.

3.2 方法核心描述

首先简单介绍一下隔离子句^[21],其定义是将可满足性求解器所得到的解中所有变量的逻辑赋值取反,作为新的子句加入到 SAT 子句库中.

例 3. 如用可满足性求解器得到一个解为

$$v_1, -v_2, v_3, -v_4.$$

上式表示,当变量 v_1, v_3 的逻辑取值为 true (1),而变量 v_2, v_4 的逻辑取值为 false(0)时,输入的合取范式能够被满足.此时,我们将这些变量的逻辑值取反作为它们各自对应文字的逻辑符号,就构成了 SAT 求解器所需要的隔离子句:

$$-v_1, v_2, -v_3, v_4.$$

本文 SAT-MHS 方法采用的是循环迭代求解方法,该方法主要思想是在调用 SAT 求解极小碰集的过程中,把之前每次求解的碰集极小化后取反添加到 CNF 中.屏蔽对该解和非极小解及部分无用搜索空间求解,达到缩减求解空间,提高求解效率.每次求解判断是否为可满足,当不可满足则终止求解.

3.3 SAT-MHS 方法实现

算法 2. SAT-MHS.

输入: $CONFLICT = \{CF_1, CF_2, \dots, CF_j, \dots, CF_M\}$;

/* 极小冲突集合簇 */

输出: $MHSs$.

/* 极小碰集集合簇 */

① $Trans_CNF(CONFLICT, CNFs)$;

/* 将极小冲突集合簇转化为 CNF 形式 */

② 初始化 $Mark := UNSATISFIABLE$;

③ $Mark := IS_SATISFIABLE(CNFs)$;

/* 第 1 次调用 SAT 求解 */

④ if ($Mark == UNSATISFIABLE$) then

/* 如果无解,停止求解并返回 */

⑤ return;

⑥ else

⑦ while ($Mark == SATISFIABLE$) do

/* 如果 SAT 求解是可满足的,进行循环迭代求解 */

⑧ $New_HS = Picosat_Sat(CNFs)$;

/* 调用 SAT 求解碰集解 */

⑨ $Clause := IS_MHS(New_HS)$;

/* 把解进行极小化为极小碰集 */

⑩ $MHSs \leftarrow MHSs \cup Clause$;

```
/* 把求得的解添加到极小碰集集合簇 */
⑪ CNFs←CNFs∪ $\overline{Clause}$ ;
/* 屏蔽掉本次求得的极小碰集和它的超集的 */
⑫ Mark:=IS_SATISFIABLE(CNFs);
/* 调用 SAT 求解,并判断是否有碰集 */
⑬ end while
⑭ end if
⑮ return MHSs.
```

算法 2 首先把输入极小冲突集合簇转化为 SAT 求解器能够读取的 CNF 文件(行⑪),初始化可满足性变量 *Mark*(行⑫). 然后首次调用 SAT 求解器对 CNF 文件进行可满足性求解并对其值为不可满足时则返回,结束算法(行⑬~⑮);否则,把可满足的赋值进行极小化处理得到的极小碰集添加到极小碰集集合簇中,并对求解的极小碰集以取反的隔离子句的形式添加到原始的 CNF 中,屏蔽掉该极

小碰集对应的求解空间(行⑧~⑪). 最后对是可满足的 CNF 不断循环迭代的求解,直到得到不可满足(行⑦~⑬),结束 while 循环并得到所有的极小碰集 *MHSs*(行⑮).

下面主要对 SAT-MHS 方法最关键的 DOEC 极小化策略举例说明其应用效果.

对于例 1 中的冲突集合簇对应的 *HS*,假设该方法当前得到碰集{2,3,4,5,6},如果不使用 DOEC 极小化策略对其进行处理,而是直接以隔离子句形式将其加入到 SAT 的 CNF 文件中,则只是剪枝屏蔽掉了{1,2,3,4,5,6}这一个解,其剪枝效果如图 1(a)所示. 但如果使用 DOEC 极小化策略对求解出碰集{2,3,4,5,6}进行极小化处理得到极小碰集{3,4,6},然后再把其以隔离子句的形式加入到 SAT 的 CNF 文件中,可以从图 1(b)中看出其把结点{3,4,6}所应的超集剪枝掉,从而有效地避免了对非碰集和非极小碰集的求解,减少调用 SAT 求解的次数,使方法更加高效.

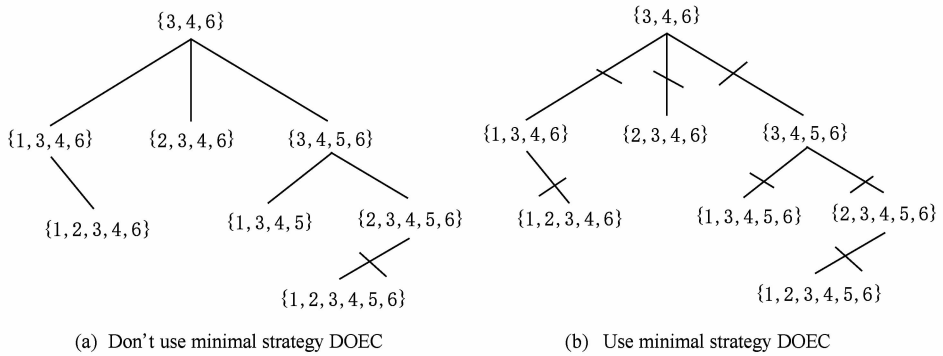


Fig. 1 An example of using the minimization strategy DOEC for pruning the search space
图 1 使用 DOEC 极小化策略搜索空间剪枝情况

4 实验与结果

实验所用的所有实验用例均是随机生成,输入参数有元素个数 *n*、集合簇中集合个数 *m* 以及元素在一个集合中出现的概率 *p*. 在同一个用例中,所有元素的 *p* 均相等,每个集合包含元素的平均个数约等于 *np*. 本文使用的实验用例共有 4 组,分别为 30, 25,20,15. 每组都是包含概率为 *p* 的用例,*p* 为从 0.05~0.94 的 19 个概率分布数据,每组集合个数都是 200. 本文所有实验数据均为使用相同参数下独立生成的 10 个用例所得结果,然后对其取平均值. 本文的实验平台均为:系统硬件配置是 Intel Core I5-4590 CPU@3.30 Hz x4,8 GB RAM,C 语言.

4.1 SSDM 和 DOEC 极小化策略实验结果分析比较

本节使用随机生成测试用例的 25 和 30 这 2 组较难的包含概率为 *p* 的用例,*p* 取值从 0.20~0.80 的 14 个概率分布数据进行实验,实验数据均为使用相同参数下独立生成的 10 个用例以 SAT-MHS 方法为基础进行实验所得结果并对其实验结果取平均值,详细如表 1 所示.

从表 1 可以看出,对于元素个数 *N*=25 的测试用例,DOEC 极小化策略是 SSDM 极小化策略求解速度的 1~40.3 倍;而对于 *N*=30 的测试用例,DOEC 极小化策略是 SSDM 极小化策略求解速度的 21~55.1 倍之间,可见 DOEC 极小化策略对于较大规模测试用例的高效性. 除此之外,DOEC 极小化策略会随着碰集个数的增多和元素个数的增多,

Table 1 Performance Comparison of SSDM Minimization Strategy and DOEC Minimization Strategy

表 1 SSDM 极小化策略和 DOEC 极小化策略实验结果比较

$N=25/p$	Number of MHS_s	DOEC/s	SSDM/s	Ratio of Solving Speed	$N=30/p$	Number of MHS_s	DOEC/s	SSDM/s	Ratio of Solving Speed
20	670.4	0.1	0.1	1.0	20	20 211.8	3.2	176.4	55.1
25	5 114.2	0.3	8.1	27.0	25	105 100.3	275.1	10 546.6	38.3
30	17 456.3	3.1	119.7	38.6	30	254 760.6	1 247.1	50 286.3	40.3
34	34 051.8	9.6	338.5	35.2	34	280 574	1 472.2	51 413.4	34.9
40	37 746.7	10.6	428.2	40.3	40	231 908.4	849.8	30 586.7	35.9
44	34 917.3	7.5	292.9	39.0	44	173 240.6	326.1	11 896.1	36.5
50	31 386.1	6.1	185.3	30.3	50	127 041.4	134.7	6 001.3	44.5
55	23 200.8	3.1	84.6	27.2	55	78 050.6	31.1	1 549.1	49.8
60	16 495.5	1.5	32.3	21.5	60	47 083.5	9.1	400.2	43.9
64	11 412.8	0.7	11.9	17.0	64	30 292.8	3.5	102.7	29.3
69	7 104.7	0.3	4.12	13.7	69	17 809	1.1	31.3	28.4
75	4 872.8	0.1	1.47	14.7	75	10 617	0.4	8.4	21.0
80	3 013.3	0.1	0.47	4.7	80	6 080.3	0.1	2.3	23.0

其求解的效率也会随之增加。所以,DOEC 极小化策略会随着待求解问题规模的增大,其求解效率也会随之增加。

4.2 SAT-MHS 方法与 Boolean 方法对比分析

本节使用 SAT-MHS 方法与目前求解效率较高的 Boolean 方法进行比较,并给出 2 种方法在随

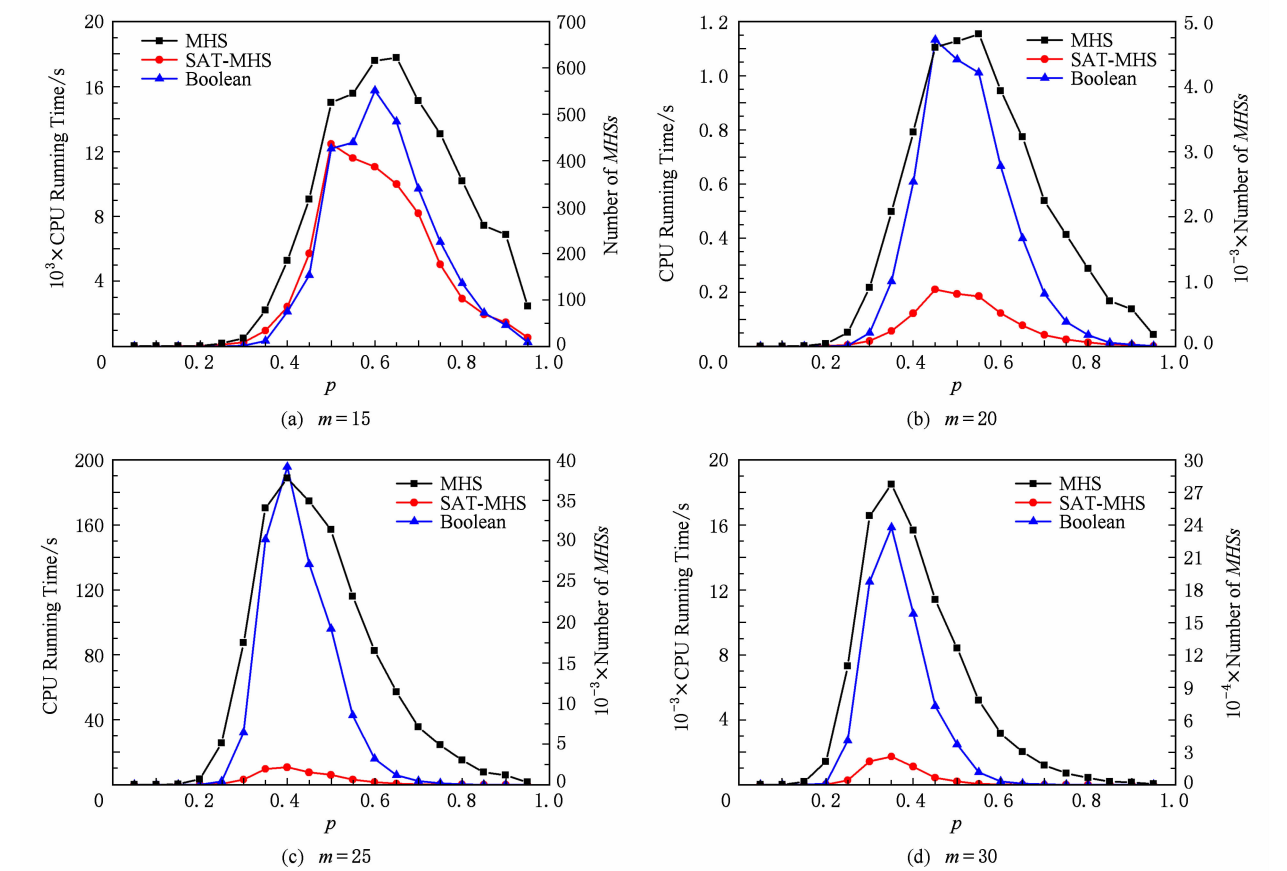


Fig. 2 The performance comparison of SAT-MHS and Boolean

图 2 SAT-MHS 方法与 Boolean 方法实验结果比较

机生成测试用例的实验结果.

从图 2 中可以看出,对于元素个数 $m=15$ 的测试用例,因其碰集个数和元素个数较少,所以其求解耗时较少,但 SAT-MHS 方法还是略比 Boolean 方法用时少. 对于元素个数较多的用例 $m=20, 25, 30$, SAT-MHS 方法的优势就较为明显. 随着所求解的极小碰集的个数不断增加, SAT-MHS 方法的优势越来越明显,求解时间的差距也越来越大,所以对于碰集个数较大的问题其拥有较大优势. 尤其对于耗时较长的用例来说, SAT-MHS 的求解效率为 Boolean 方法的 30 倍左右. 不难发现,对比 Boolean 方法, SAT-MHS 方法在耗时越长的测试用例中其优势越明显.

从图 3 可以看出,对于所求解的极小碰集的个数较少的测试用例,当两者的求解时间在 10^{-3} s 以下, SAT-MHS 方法的求解时间大于 Boolean 方法,因为调用 SAT 求解器需要一些初始化时间;但随着求解规模的增大,当求解时间在 $10^{-3} \sim 10^{-2}$ s 之间时,可以看出两者的求解时间逐渐接近;当求解时间在 $10^{-2} \sim 10^0$ s 之间, SAT-MHS 方法的求解效率明显是 Boolean 方法的 3~7 倍之间;当求解时间在 $10^0 \sim 10^2$ s 之间,其求解效率甚至高达 10~20 倍;当求解时间在 $10^2 \sim 10^4$ s 之间, SAT-MHS 方法的求解效率基本是 Boolean 方法的 10 倍以上. 综合对比图 2 和图 3 可以看出, SAT-MHS 方法与 Boolean 方法相比,在求解极小碰集的规模大及复杂性高的问题,其基本求解效率在 10 倍以上.

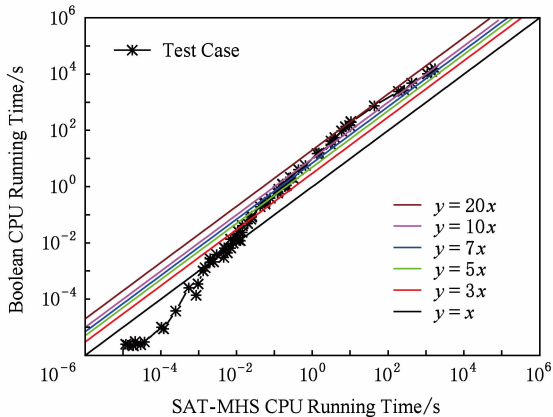


Fig. 3 The comparison results of efficiency between SAT-MHS and Boolean

图 3 SAT-MHS 方法与 Boolean 方法实验求解效率比较

此外,从图 3 可以看出,在求解小规模问题时, Boolean 没有初始化过程,而 SAT-MHS 求解时需要初始化,因此其求解时间比 Boolean 多些. 但当问

题规模越大时,产生的碰集个数增多,需要花费大量的时间处理极小化问题,可见 SAT-MHS 更有优势.

5 总 结

极小碰集在现实生活中有重要的实际应用价值,目前主流完备的碰集极小化策略是 SSDM,但其不足是其极小化效率会随着极小碰集的数量增加而使极小化的耗时也随之急剧增多. 通过把碰集求解问题转化为 SAT 问题,然后在 DOEC 极小化策略的基础上提出了一种新的碰集求解方法 SAT-MHS. 该极小化策略极小化耗时和已求得的极小碰集规模脱离关系,只与最初的极小冲突集合簇的大小有关,不会随着求解的极小碰集数量的增加而迅速增加.

SAT-MHS 方法把极小冲突集合簇转化为 SAT 求解器能处理的 CNF 文件,并调用较为成熟的 SAT 求解器进行循环迭代的碰集求解;每次可满足的文字变量赋值就是 SAT 求解器求解得到的一个碰集. 然后对得到的非极小碰集用 DOEC 极小化策略进行极小化,并把处理得到的极小碰集以隔离子句的形式添加到 CNF 文件中,从而屏蔽待搜索求解空间,提高求解效率. 与目前求解效率较高的 Boolean 方法相比,可以看到该方法的高效性,尤其对求解问题规模和碰集个数较多的问题实例,该方法的求解效率至少在 10 倍以上. 未来的工作是使用启发式求解策略^[22-24]求解更大规模的极小碰集问题.

参 考 文 献

- [1] Console L, Dressler O. Model-based diagnosis in the real world: Lessons learned and challenges remaining [C] //Proc of the 16th Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 1999: 1393-1400
- [2] Struss P. Knowledge-based diagnosis: An important challenge and touchstone for AI [C] //Proc of the 10th European Conf on Artificial Intelligence. New York: John Wiley & Sons, 1992: 863-873
- [3] Hamscher W. Readings in model-based diagnosis [OL]. 1992 [2017-06-17]. <http://www.citeulike.org/group/14148/article/8042710>
- [4] De Kleer J. Diagnosing multiple persistent and intermittent faults [C] //Proc of the 21st Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 2009: 733-738
- [5] Wang Yiyuan, Yin Minghao, Ouyang Dantong, et al. A novel local search algorithm with configuration checking and scoring mechanism for the set k -covering problem [J]. Int Trans in Operational Research, 2017, 24(6): 1463-1485

[6] Wang Yiyuan, Ouyang Dantong, Zhang Liming, et al. A novel local search for unicost set covering problem using hyperedge configuration checking and weight diversity [J]. Science China Information Sciences, 2017, 60(6): 062103

[7] Reiter R. A theory of diagnosis from first principles [J]. Artificial Intelligence, 1987, 32(1): 57-95

[8] Greiner R, Smith B A, Wilkerson R W. A correction to the algorithm in Reiter's theory of diagnosis [J]. Artificial Intelligence, 1989, 41(1): 79-88

[9] Wotawa F. A variant of Reiter's hitting-set algorithm [J]. Information Processing Letters, 2001, 79(1): 45-51

[10] Jiang Yunfei, Lin Li. The computation of hitting sets with Boolean formulas [J]. Chinese Journal of Computers, 2003, 26(8): 919-924 (in Chinese)
(姜云飞, 林笠. 用布尔代数方法计算最小碰集[J]. 计算机学报, 2003, 26(8): 919-924)

[11] Ouyang Dantong, Ouyang Jihong, Cheng Xiaochun, et al. A method of computing hitting set in model-based diagnosis [J]. Chinese Journal of Science Instrument, 2004, 25(4): 605-608 (in Chinese)
(欧阳丹彤, 欧阳继红, 程晓春, 等. 基于模型诊断中计算碰集的方法[J]. 仪器仪表学报, 2004, 25(4): 605-608)

[12] Zhao Xiangfu, Ouyang Dantong. A method of combing SE-Tree to compute all minimal hitting sets [J]. Progress in Natural Science, 2006, 16(2): 169-174

[13] Chen Xiaomei, Meng Xiaofeng, Qiao Renxiao. Method of computing all minimal hitting set based on BNB-HSSE [J]. Chinese Journal of Science Instrument, 2010, 31(1): 61-67 (in Chinese)
(陈晓梅, 孟晓风, 乔仁晓. 基于 BNB-HSSE 计算全体碰集的方法[J]. 仪器仪表学报, 2010, 31(1): 61-67)

[14] Zhang Liming, Ouyang Dantong, Zeng Hailin. Computing the minimal hitting set based on dynamic maximum degree [J]. Journal of Computer Research and Development, 2011, 48(2): 209-215 (in Chinese)
(张立明, 欧阳丹彤, 曾海林. 基于动态极大度的极小碰集求解方法[J]. 计算机研究与发展, 2011, 48(2): 209-215)

[15] Pill I, Quaritsch T. Optimizations for the Boolean approach to computing minimal hitting sets [C] //Proc of the 20th European Conf on Artificial Intelligence. Amsterdam: IOS Press, 2012: 648-653

[16] Wang Yiyuan, Ouyang Dantong, Zhang Liming, et al. A method of computing minimal hitting sets using CSP [J]. Journal of Computer Research and Development, 2015, 52(3): 588-595 (in Chinese)
(王艺源, 欧阳丹彤, 张立明, 等. 利用 CSP 求解极小碰集的方法[J]. 计算机研究与发展, 2015, 52(3): 588-595)

[17] Yang Kun, Bai Hongjun, Ouyang Qi, et al. Finding multiple target optimal intervention in disease-related molecular network [J]. Molecular Systems Biology, 2008, 4(1): 228

[18] Zhou Gan, Feng Wenquan, Jiang Bofeng, et al. Computing minimal hitting set based on immune genetic algorithm [J]. Int Journal of Modelling, Identification and Control, 2014, 21(1): 93-100

[19] Jannach D, Schmitz T, Shchekotykhin K. Parallelized hitting set computation for model-based diagnosis [C] //Proc of the 24th AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2015: 1503-1510

[20] Zhao Xiangfu, Ouyang Dantong. Deriving all minimal hitting sets based on join relation [J]. IEEE Trans on Systems, Man, and Cybernetics; Systems, 2015, 45(7): 1063-1076

[21] McMillan K L. Applying SAT methods in unbounded symbolic model checking [C] //Proc of Conf on Computer Aided Verification. Berlin: Springer, 2002: 250-264

[22] Wang Yiyuan, Cai Shaowei, Yin Minghao. Local search for minimum weight dominating set with two-level configuration checking and frequency based scoring function [J]. Journal of Artificial Intelligence Research, 2017, 58(1): 267-295

[23] Wang Yiyuan, Cai Shaowei, Yin Minghao. Two efficient local search algorithms for maximum weight clique problem [C] //Proc of the 13th AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2016: 805-811



Wang Rongquan, born in 1992. Master candidate at Jilin University. His main research interests include model-based diagnosis, SAT problem and automated reasoning.



Ouyang Dantong, born in 1968. Professor and PhD supervisor of Jilin University. Senior member of CCF. Her main research interests include model-based diagnosis, model checking and automated reasoning (ouyangdantong@163.com).



Wang Yiyuan, born in 1988. PhD candidate at Jilin University. His main research interests include model-based diagnosis, SAT problem and automated reasoning (yiyuanwangjlu@126.com).



Liu Siguang, born in 1988. Master candidate at Jilin University. His main research interests include model-based diagnosis, SAT problem and automated reasoning (lsgmliss@163.com).



Zhang Liming, born in 1980. PhD, post-doctor in Jilin University. His main research interests include model-based diagnosis, SAT problem and automated reasoning.