计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(2): 408−417, 2025

# 融合基数约束与单次入队的基于模型诊断方法

青　杨　欧阳丹彤　周慧思　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）
（qingyang3696@163.com）

# A Model-Based Diagnosis Method for Integrating Cardinality Constraints and Enqueueing at Once

Qing Yang, Ouyang Dantong, Zhou Huisi, and Zhang Liming
（*College of Computer Science and Technology*, *Jilin University*, *Changchun* 130012）
（*Key Laboratory of Symbol Computation and Knowledge Engineering* (*Jilin University*), *Ministry of Education*, *Changchun* 130012）

**Abstract**　Model-based diagnosis (MBD) finds a growing number of uses in different settings. It includes software fault localization, debugging of spread sheets, Web services, and hardware designs, and also the analysis of biological systems, among many others. Motivated by these different uses, there have been significant improvements made to MBD algorithms in recent years. Nevertheless, the analysis of larger and more complex systems motivates further improvements to existing approaches. Since computing diagnosis is computationally challenging, some MBD algorithms by compacting the model are presented successively, such as dominated-based compacted model with multiple observations (D-CMMO) approach. In this paper, we propose one new diagnosis model, namely, cardinality-constrained compacted (CCM) model, to solve the problem in which a considerable amount of time is needed when multiple observations are given and more than one fault is injected. CCM uses two methods to optimize the process of solving MBD. Firstly, we propose to utilize relationship of faulty system-outs and faulty components to limit the scope of target solution. Secondly, the performance of the MaxSAT (maximum satisfiability) solver is effectively improved by enqueueing all assumptions at once. Furthermore, experiment evaluations on ISCAS85 and ITC99 benchmarks show that compared with D-CMMO, the latest encoding algorithms for MBD, the above two optimization methods effectively reduce the scope of MBD problem, reduce the difficulty of searching for the target solution by the MaxSAT solver, and thus return diagnostic solution in a shorter time. On average, the solving efficiency of CCM method is improved by 64.5% and 92.8% compared with D-CMMO method respectively.

**Key words**　model-based diagnosis; compacted model; cardinality-constraints-based algorithm; top-level diagnosis; enqueueing all assumptions at once

**摘　要**　基于模型诊断（MBD）方法在不同的环境中有越来越多的用途，包括软件故障定位、电子表格的调试、Web服务和硬件设计，以及生物系统的分析等. 受这些不同用途的启发，近年来MBD算法改进成效显著. 然而，对体系庞大、结构复杂的系统，需要对现有方法进一步改进. 由于求解诊断解在计算上具有挑

**收稿日期**：2023-06-24；**修回日期**：2023-12-07
**基金项目**：国家自然科学基金项目（62076108, 61872159）；吉林省教育厅项目（JJKH20211106, JJKH20211103KJ）
This work was supported by the National Natural Science Foundation of China (62076108, 61872159) and the Project of Education Department of Jilin Province (JJKH20211106, JJKH20211103KJ).
**通信作者**：张立明（limingzhang@jlu.edu.cn）

战性，因此相继提出了一些通过压缩模型的 MBD 算法来提高诊断效率，如基于统治的多观测压缩模型（dominated-based compacted model with multiple observations, D-CMMO）算法. 对于给定多个观测值且注入 1 个以上错误需要大量时间的诊断问题，提出了一个新的诊断模型 CCM（cardinality-constrained compacted model）来解决. 基于基数约束的压缩模型算法使用 2 种方法对求解过程进行优化：首先，利用系统观测的故障输出和故障组件数量之间的约束关系来限制目标解的范围；其次，通过对假设集采用单次入队方法，进而有效提升 MaxSAT（maximum satisfiability）求解器的性能. 此外，在 ISCAS85 和 ITC99 基准测试用例上的实验结果表明，与目前最新的 MBD 求解方法 D-CMMO 相比，上述 2 种优化方法有效缩小了 MBD 问题的求解范围，降低 MaxSAT 求解器搜索目标解的难度，进而能在更短的时间内返回一个诊断解. 在平均状况下，CCM 方法相比 D-CMMO 方法求解效率分别提升 64.5% 和 92.8%.

**关键词** 基于模型诊断；压缩模型；基于基数约束的算法；顶层诊断解；单次入队

**中图法分类号** TP391

**DOI:** 10.7544/issn1000-1239.202330546 **CSTR:** 32373.14.issn1000-1239.202330546

基于模型诊断（model-based diagnosis, MBD）[1]是分析复杂系统并解释系统故障原因的重要方法，这个方法已经在人工智能领域被研究了多年. 一般来说，MBD 方法要么使用系统中每个组件都是已知和定义的溯因系统[2]，要么使用系统中所有组件都未被定义的一致性模型[3]. 这些算法被广泛应用于各个领域，包括质量模型[4]、Web 服务的调试[5]、电子表格调试[6]和软件故障定位[7]等领域.

本文的主要贡献包括 2 个方面：

1）提出通过传播系统输入得到诊断解的上界，将硬子句作为基数约束编码加入到 MaxSAT 编码中. 这种基数约束方法可以缩小诊断的搜索范围，进而降低求解诊断的难度.

2）提出利用单次入队方法将假设集中的所有文字入队到第 1 决策层，然后再进行单元传播，从而避免出现冲突后回溯到第 0 决策层导致的对假设集重复赋值的问题. 同时能够有效避免假设集中的否定文字被频繁替换到子句的看守列表中，排除对子句中的文字重复访问的代价.

## 1 相关工作

MBD 算法通过给定的系统模型和 1 个或多个观测值来计算诊断. 目前用于解决单个观测下的 MBD 问题的算法有：随机搜索算法[8]、基于编译的算法[9]、基于广度优先搜索的算法[10]、基于可满足性（satisfiability, SAT）问题的算法[11-12]、基于冲突导向的算法[13]等. 事实上，相比于单观测 MBD 问题，使用关于诊断系统的多个观测值有助于提高诊断过程的效率，因为它们在计算诊断时可以提供更多的诊断系统信息[14].

近年来，已经提出了许多考虑到多观测的基于模型诊断的方法. DC 算法[15]旨在计算关于每个观测的每组诊断的所有覆盖集. DC*[16]通过减少冗余诊断的数量来降低 DC 算法的运行时间. 然而，DC 和 DC* 都不能保证返回最小势的诊断. 蕴含碰集对偶（hitting set dualization, HSD）算法[14]专注于使用蕴含碰集和最大可满足性[17-18]算法，在诊断和冲突之间计算最小校正集（minimal correction subsets, MCS）[19-20]，其效果在注入单故障模型时优于 DC 和 DC*算法. 尽管蕴含碰集对偶算法的性能在大多数情况下优于 DC 和 DC*算法，但当注入更多故障时，它表现出较差的性能. 针对其进行改进的 IHSD 算法[21]，在推导具有多个观测的诊断时，首次使用了统治组件的概念. One SAT 算法[22]将所有观测结果汇编成可满足性公式[23]，并使用 SAT 求解器导出所有组件的健康指定. 基于统治的多观测压缩模型（dominated-based compacted model with multiple observations, D-CMMO）方法[24]通过新型变量映射方法来压缩子系统，并将统治组件编码为硬子句来计算顶层诊断解（top-level diagnosis, TLD）[25]. 作为一种压缩模型的方法，D-CMMO 方法利用了系统的结构属性，有效地缩减了 MBD 问题编码后子句集的规模. 虽然 D-CMMO 保证返回基数最小的诊断，但它没有考虑故障输出与故障组件数量之间的关系.

本文提出一种基于基数约束的诊断方法 CCM，并通过单次入队策略进一步提升 MaxSAT 求解器的性能，该方法在求解最小势诊断时，能有效减少求解时间. 此外，本文在 ISCAS85 和 ITC99 标准测试实例上，分别将 CCM 方法与 D-CMMO 方法进行了比较.

实验结果表明, CCM 算法有效提高了 MBD 问题的求解效率.

## 2 基于模型诊断问题

在本节中, 主要介绍基于模型诊断的相关定义及概念. 本文使用标准的基于模型诊断的定义[1]以及最新的参考文献[9,24-25], 主要讨论多观测的基于模型诊断问题. 更具体而言, 本文针对于计算一个弱故障模型的极小势诊断.

### 2.1 基本定义

基于模型诊断问题由一个三元组( *SD*, *Comps*, *Obs*)组成, 其中 *SD*(system description)表示诊断问题的系统描述, *Comps*(components)代表组件的集合, *Obs*(observations)表示系统输入和系统输出的观测值.

根据 *SD* 和 *Comps*, 每一个组件可以被定义为是健康的或是故障的. 对于每一个组件 $c \in Comps$, 如果 $c$ 是故障的, 则 $Ab(c)=1$; 否则 $Ab(c)=0$. 假设每一个组件 $c \in Comps$ 都是健康的, 记为 $\forall c \in Comps$, $Ab(c)=0$, 并且观测值是确定的, 即没有传感器噪声造成的误差, 诊断问题产生于 *SD* 和 *Comps* 不一致时:

$$SD \wedge Obs \wedge \{\neg Ab(c) | c \in Comps\} \vDash \bot . \tag{1}$$

**定义 1.** 诊断[1]. 给定一个诊断问题〈*SD*, *Comps*, *Obs*〉, 诊断被定义为一组组件的集合 $M \subseteq Comps$,

$$SD \wedge Obs \wedge \{Ab(c) | c \in M\} \wedge \{\neg Ab(c) | c \in Comps \backslash M\} \nvDash \bot . \tag{2}$$

**定义 2.** 极小诊断[25]. 一个诊断 $M$ 是极小子集诊断, 当且仅当任意 $M_0 \subset M$ 都不是一个诊断. $M$ 是一个极小势诊断, 当且仅当对于任意的诊断 $M_0$ 都有 $|M_0| \geqslant |M|$.

当对一个诊断问题给定了多于 1 个观测时, 需要返回一组组件的赋值使之能解释全部的观测.

**定义 3.** 多观测诊断[14]. 给当一个多观测的诊断问题〈*SD*, *Comps*, *Obs*〉, 假设每一个组件 $c \in Comps$ 都是健康的, 记为 $\neg Ab(c)$, 一个多观测的诊断被定义为一组组件的子集 $M \subseteq Comps$, 当

$$SD \wedge \bigwedge_{i=1}^{m} Obs_i \wedge \{Ab(c) | c \in M\} \wedge \{\neg Ab(c) | c \in Comps \backslash M\} \nvDash \bot . \tag{3}$$

### 2.2 将诊断问题编码为 MaxSAT

许多诊断方法通过将诊断问题编码为 MaxSAT 来求解诊断, 本节将介绍 MaxSAT 的编码过程.

标准的基于模型诊断问题的编码方法[26]将每一个组件 $c \in Comps$ 与一个命题范式 $\phi$ 联系起来, 代表了它的正常行为, 并且用一个布尔变量 $H_c$ 来代表 $c$ 是否为健康的. 将观测值看作一个命题式声明, MaxSAT 编码的硬子句部分由具体化式(4)得到:

$$\phi = Obs \wedge \bigwedge_{c \in Comps} H_c \Rightarrow \varphi_c . \tag{4}$$

而 MaxSAT 的软子句部分由所有代表组件健康状况的变量 $H_c$ 组成. 由于 MaxSAT 可以求解满足最大权重和的软子句, 因此将 $H_c$ 设置为软子句可以保证得到最大数量健康组件的诊断解, 即极小势诊断. 在一组满足 $\varphi$ 的赋值中, 被指派为 false 的健康变量 $H_c$ 的集合决定了一个诊断 *M*. 图 1 给出 c17 的电路结构.

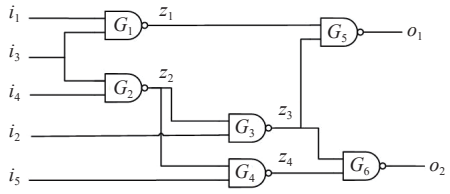



Fig. 1 c17 circuit diagram

图 1 c17 电路图

例如, 考虑图 1 中的基于模型诊断问题, 使 *nand*(*A*, *B*, *C*) 描述与非门正常行为的命题范式, 其中, *A* 和 *B* 是与非门的输入, *C* 为与非门的输出. 因此

$$\varphi = \begin{pmatrix} i_1 \wedge -i_2 \wedge i_3 \wedge i_4 \wedge -i_5 \wedge H_{N_1} \Rightarrow nand(i_1, i_3, z_1) \wedge \\ H_{N_2} \Rightarrow nand(i_3, i_4, z_2) \wedge H_{N_3} \Rightarrow nand(i_2, z_2, z_3) \wedge \\ H_{N_4} \Rightarrow nand(i_5, z_2, z_4) \wedge H_{N_5} \Rightarrow nand(z_1, z_3, o_1) \wedge \\ H_{N_6} \Rightarrow nand(z_3, z_4, o_2) \end{pmatrix} . \tag{5}$$

软子句由以下部分组成:

$$\varphi' = \left(H_{N_1} \wedge H_{N_2} \wedge H_{N_3} \wedge H_{N_4} \wedge H_{N_5} \wedge H_{N_6}\right) . \tag{6}$$

WCNF 中的子句 *cl* 的权重用 $\omega(cl)$ 表示, 我们分别设置硬子句和软子句的权重为:

1)对于 $\varphi$ 中的子句被设置为硬子句, $\omega(cl) := num(H_c)$,

2)所有的 $H_c$ 变量被设置为软子句, $\omega(cl) := 1$,

其中 $num(H_c)$ 代表健康变量的数量, 也就是组件的数量.

电路系统的直接统治者是一个用来约束 CNF 编码被广泛使用的技术[9,27]. 令 *o* 代表一个连接所有系统输出的虚拟节点. 节点 *v* 是节点 *u* 的统治者如果从 *u* 到 *o* 的所有路径都通过 *v*[28]. 节点 *v* 是 *u* 的直接统治者如果 *u* 的所有其他统治者也是 *v* 的统治者. 一个电

路门是被统治节点如果它的直接统治者不是 $o$，否则它不是被统治节点.

如果一个被统治的电路门出现在某个极小势诊断中，那么它可以被它的直接统治者取代[9]. 因此，求解极小势诊断时可以只分析不包括被统治节点的诊断. 这种诊断被称作顶层诊断. 利用系统的结构属性计算顶层诊断是一种可行的方法，这种方法在许多基于 SAT 的诊断算法中得到了应用[29-31].

**定义 4.** 顶层诊断[25]. 一个极小势诊断是一个顶层诊断如果它不包括被统治的电路门.

以图 1 为例，由于 $N_1$ 到达系统输出的路径是唯一的，且通过 $N_5$，所以 $N_5$ 是 $N_1$ 的统治者. 对于观测 $\{i_1, -i_2, i_3, i_4, -i_5, o_1, o_2\}$，$N_3$ 是一个顶层诊断，因为它不包括任何被统治节点. 通过将被统治组件的健康变量添加到硬子句中(即声明这些组件是健康的)，可以计算顶层诊断.

## 3 压缩电路模型的 D-CMMO 方法

为了简化 MBD 到 MaxSAT 的编码过程，缩减被编码的电路门生成的子句的规模，D-CMMO 方法[24]对每一个观测构造一个子系统，并利用子系统之间的结构关系压缩电路模型. 压缩后的模型是一个布尔范式，我们通过单次调用 MaxSAT 求解器即可得到多观测的诊断. 一些曾在 D-CMMO 中提出的概念如传播边及传播节点，将在以下定义中给出解释.

**定义 5.** 差别输入(*DI*)[24]. *DI*($SysIns_x$, $SysIns_y$)，记作 $Di_{x,y}$，是系统输入的一个子集，由 $SysIns_x$ 和 $SysIns_y$ 中不同的变量组成.

例如，给定 2 组系统输入 $SysIns_1=\{-i_1, i_2, i_3, -i_4, i_5\}$，$SysIns_2=\{-i_1, i_2, -i_3, i_4, -i_5\}$，$DI_{1,2}=\{i_3, i_4, i_5\}$.

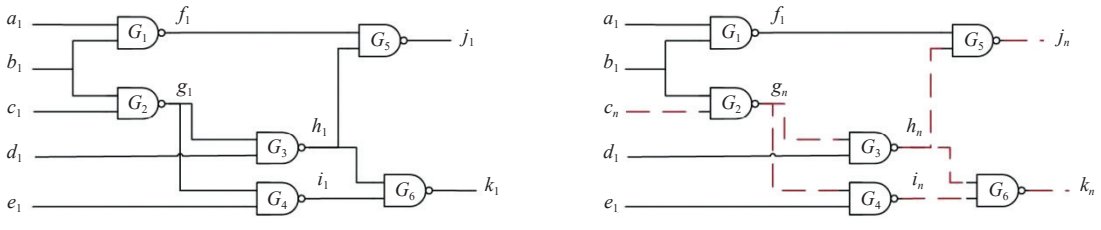
**定义 6.** 传播边(*P-Edges*)[24]. 给定一个系统输入 $i$，*P-Edges*($i$) 是系统变量的一个子集，代表从 $i$ 到所有可达的系统输出所经过的边. 给定一个系统输入的子集，记作 *SubSysIns*，*P-Edges*(*SubSysIns*) 是系统变量的一个子集，代表从 $i$ 到所有可达的系统输出所经过的边，且 $i \in SubSysIns$.

**定义 7.** 传播节点(*P-Nodes*)[24]. 给定一个系统输出 $i$，*P-Nodes*($i$) 是 *Comps* 的一个子集，代表从 $i$ 到所有可达的系统输出所经过的节点. 给定一个系统输入的子集，记作 *SubSysIns*，*P-Nodes*(*SubSysIns*) 是系统变量的一个子集，代表从 $i$ 到所有可达的系统输出所经过的节点，且 $i \in SubSysIns$.

例如，考虑图 1 中的系统变量 $i_1$，*P-Edges*($i_1$)=$\{i_1, z_1, o_1\}$，*P-Nodes*($i_1$)=$\{N_1, N_5\}$. 考虑图 1 中的系统输入的一个子集 $\{i_1, i_2\}$，*P-Edges*($\{i_1, i_2\}$)=$\{i_1, z_1, o_2, i_2, z_3, o_2\}$，*P-Nodes*($\{i_1, i_2\}$)=$\{N_1, N_3, N_5, N_6\}$.

在将 MBD 问题编码为 MaxSAT 时，D-CMMO 方法利用定义 4~6 来简化电路模型. 图 2 展示了第 1 组观测和第 $N$ 组观测之间的差别，图 2(a)展示了与第 1 组观测匹配的系统变量的整型数；图 2(b)展示了与第 $N$ 组观测所匹配的系统变量的整型数，不同的变量代表了不同的整型数.

式(7)列出了利用简化电路模型得到的系统描述的命题逻辑范式. 如式(7)中所示，变量的匹配过程记为 *VarMapping*，工作如下：对于图 2(a)，我们将组件或线路相应地匹配为一个新的整型数并构建第 1 个子系统. 对于图 2(b)，我们计算 *DI*($SysIns_1$, $SysIns_n$)，记作 $DI_{1,n}$，并且重新为 *P-Edges*($D_{1,n}$) 中的系统变量赋值一个新的整型数，除了差别变量之外，保留其他变量的匹配. 迭代地，通过编码 *P-Nodes*($DI_{1,i}$) 中的组件来构建所有子系统. 在图 2 中，假设 $SysIns_1$ 和 $SysIns_n$ 的差别变量是 $\{c_n\}$，在 *VarMapping* 的过程中，*P-Nodes*($c_n$) 中的系统变量被重新赋值了整型数，这些系统变量用虚线代表. 当编码第 $N$ 个子系统时，只有 *P-Nodes*($c_n$) 中的系统变量和连接它的线路需要被重新编码，也就是说，$\{N_1\}$和连接它的线路不需要被重新编码.




(a) 第1组观测　　(b) 第$N$组观测

Fig. 2 Example diagram for c17 circuit with multiple observations

图 2 多观测的 c17 电路示例图

$$M=\begin{pmatrix} H_{N_1}\Rightarrow nand(a_1,b_1,f_1) & \wedge & \cdots & \wedge & H_{N_1}\Rightarrow nand(a_1,b_1,f_1) & \wedge & \cdots \\ H_{N_2}\Rightarrow nand(b_1,c_1,g_1) & \wedge & \cdots & \wedge & H_{N_2}\Rightarrow nand(b_1,c_n,g_n) & \wedge & \cdots \\ H_{N_3}\Rightarrow nand(g_1,d_1,h_1) & \wedge & \cdots & \wedge & H_{N_3}\Rightarrow nand(g_n,d_1,h_n) & \wedge & \cdots \\ H_{N_4}\Rightarrow nand(g_1,e_1,i_1) & \wedge & \cdots & \wedge & H_{N_4}\Rightarrow nand(g_n,e_1,i_n) & \wedge & \cdots \\ H_{N_5}\Rightarrow nand(f_1,h_1,j_1) & \wedge & \cdots & \wedge & H_{N_5}\Rightarrow nand(f_1,h_n,j_n) & \wedge & \cdots \\ H_{N_6}\Rightarrow nand(h_i,i_1,k_1) & \wedge & \cdots & \wedge & H_{N_6}\Rightarrow nand(h_n,i_n,k_n) & \wedge & \cdots \end{pmatrix}. \tag{7}$$

在编码完系统描述之后, D-CMMO 方法将所有的观测编码为硬子句, 将所有的健康变量编码为软子句.

算法 1 总结了利用 D-CMMO 计算极小势的多观测诊断的方法. 令 $M$ 代表最终的命题逻辑范式. 在行①中, 我们通过将电路系统中的每一个组件和线路匹配为一个整型数得到第 1 个子系统模型 $M_1$, 同时将 $Obs_1$ 编码为相应的常量. 行②~⑧中, 每一次迭代, 算法计算 $SysIns_1$ 和 $SysIns_i$ 的差别变量, 然后传播 $D_{1i}$ 的值得到 $P\text{-}Nodes(D_{1i})$ 和 $P\text{-}Edges(D_{1i})$, 将 $P\text{-}Nodes(D_{1i})$ 中的系统变量编码为新的整型数. 在行⑨⑩中, 算法将每一个子系统编码为硬子句. 行⑪~⑮中, 根据顶层诊断的概念, 算法 1 将被统治组件编码为硬子句(设定其为健康的), 将其他组件编码为软子句. 行⑯⑰中, 算法通过调用 MaxSAT 求解器得到多观测的诊断.

**算法 1.** D-CMMO 算法.

输入: 系统描述 $SD$, 系统组件 $Comps$, 观测值 $Obs_i$;

输出: 诊断 $\Delta$.

① $M_1 \leftarrow$ 将 $SD$,$Comps$,$Obs_1$ 编码为硬子句;

② for $i \in \{2, 3, \cdots, m\}$ do

③ 找到 $DI_{1i}$ 的 $P\text{-}Nodes$;

④ 找到 $DI_{1i}$ 的 $P\text{-}Edges$;

⑤ for $c \in P\text{-}Noeds(DI_{1i})$ do

⑥ $M_c \leftarrow WCNF(c.Fan\text{-}in, c.Fan\text{-}out, c.GateType)$; /*将扇入值、扇出值、门类型编码为 $WCNF$ 赋值给 $M_c$*/

⑦ $M_i = M_i \wedge M_c$;

⑧ end for

⑨ $M = M \wedge M_i \wedge HardCls(DI_{1i})$; /*将 $DI_{1i}$ 编码为硬子句加入到 $M$ 中*/

⑩ end for

⑪ if $c \in DominatedComps$ then

⑫ $M = M \wedge HardCls(c)$; /*如果是统治组件则编码为硬子句*/

⑬ else

⑭ $M = M \wedge SoftCls(c)$; /*否则编码为软子句*/

⑮ end if

⑯ $\Delta \leftarrow MaxSAT(M)$;

⑰ return $\Delta$.

## 4 CCM 方法

文献 [25] 中的实验表明了基数约束方法在求解基于 SAT 的诊断时能够提升算法的效率. 本节将介绍 CCM 方法及该方法为了改进基于 MaxSAT 的多观测诊断用到的基数约束和假设集的概念.

**定义 8.** 故障输出[32]. 假设电路系统中所有组件都是健康的, 对系统输入进行传播得到传播输出, 故障输出变量等同于传播输出与系统输出之间变量之差.

例如, 给定图 1 中的一组观测 $\{i_1, -i_2, i_3, i_4, -i_5, -o_1, o_2\}$, 其中系统输入为 $\{i_1, -i_2, i_3, i_4, -i_5\}$, 系统输出为 $\{-o_1, o_2\}$, 在全部组件都健康的假设下, 传播输出为 $\{o_1, o_2\}$, 故障输出为 $\{o_1\}$.

大量实验表明, 在随机注入多故障的复杂电路中, 单观测的故障输出数量常常小于多观测诊断的基数. 我们利用故障输出的数量作为多观测诊断的下界, 通过引入下界对多观测诊断的基数进行约束, 从而缩小诊断的搜索空间, 提高复杂诊断问题的搜索效率.

**定义 9.** 假设集[33]. 假设集 $A$ 是由文字组成的一个集合. 给定一个 CNF 范式 $F$, 基于假设的 SAT 求解器或返回一个赋值使得 $F \wedge A$ 可满足, 或返回不可满足.

在求解 MaxSAT 问题时, 基于假设的 SAT 求解器经常被用来计算最小不可满足子集和最小校正子集. 在基于假设集的 SAT 求解器中, 最广泛使用的处理假设集的技术是将假设集纳入 SAT 求解器的自身决策中, 这些 SAT 求解器包括 MiniSAT[34] 和 Glucose[35]. 然而, 这项技术会带来不必要的代价, 这些代价包括在加入被蕴含子句时会回溯到假设集决策之前从而导致重新对假设集进行决策, 以及基于 CDCL 的 SAT 求解器在单元传播时会重复访问同一子句中的文字 $O(n^2)$ 次. 而通过单次入队的方法可以避免这些代价. 下面将介绍标准的处理假设集的方法和改进的单次入队方法.

令输入范式为 $F$, 假设集为 $A=\{a_1, a_2, \cdots, a_k\}$. 在大多数先进的 SAT 求解器中, 支持假设集的技术需要 SAT 求解器选择 $a_i$ 作为决策文字, 当它开始第 $i$

个决策层时，其中 $1\leqslant i\leqslant k$. 如果 $a_i$ 已经被赋真值，SAT 求解器增加决策层并继续进行第 $i$+1 个决策，即一个空的决策层被加入到路径中. 如果 $a_i$ 已经被赋假值，那么可以说明前 $n-1$ 个假设决策拥有充分的条件去蕴含$\neg a_i$，因此 $C=(\neg a_1, \neg a_2, \cdots, \neg a_i)$ 是 $A$ 的冲突子句；否则 $a_i$ 还未被赋值，它会被入队到一个新的决策层中. 当假设集中所有的文字被入队后，SAT 求解器能够根据它自身的启发式算法选择决策层文字.

单次入队方法为：当所有的文字在第 0 决策层中被单元传播后，SAT 求解器增加决策层并将假设集中所有的文字入队到第 1 决策层，之后进行单元传播. 如果在处理假设集时，SAT 求解器发现 $a_i$ 已经被赋真值，那么 $a_i$ 在第 0 决策层已经被弄真，这说明 $a_i$ 已经在路径中的第 0 层而我们可以跳过对 $a_i$ 的入队操作. 相似的，如果 $a_i$ 已经被赋假值，那么$\neg a_i$ 已经在第 0 决策层被弄真，所以 $C=(\neg a_i)$ 是 $F$ 的冲突子句并且被 SAT 求解器返回. 当假设集中的一个新的蕴含单子句被发现后，单次入队方法会回溯到第 1 决策层，并将蕴含子句加入到第 1 决策层的末尾，从而避免重新赋值假设集的代价. 因为假设集文字在单元传播之前被赋值，所以假设集中的否定文字不会被频繁替换到子句的看守列表中，从而避免了相同子句中的文字被重复访问的情况. 实验结果表明，单次入队方法比标准方法更能提升 MaxSAT 求解器的效率，并且能够在相同的时间限制内求解出更多的用例.

**算法 2.** CCM 算法.

输入：系统描述 *SD*，系统组件 *Comps*，观测值 $Obs_i$；

输出：诊断 $\Delta$.

① $M_1 \leftarrow$ 将 *SD*,*Comps*,$Obs_1$ 编码为硬子句；

② for $i \in \{2, 3, \cdots, m\}$ do

③ 找到 $DI_{1i}$ 的 *P-Nodes*；

④ 找到 $DI_{1i}$ 的 *P-Edges*；

⑤ 传播 $Obs_i$ 的系统输入得到故障输出；

⑥ for $c \in P\text{-}Noeds(DI_{1i})$ do

⑦ $M_c \leftarrow WCNF(c.Fan\text{-}in, c.Fan\text{-}out, c.GateType)$; /*将扇入值、扇出值、门类型编码为 *WCNF* 赋值给 $M_c$*/

⑧ $M_i = M_i \wedge M_c$;

⑨ end for

⑩ $M = M \wedge M_i \wedge HardCls(DI_{1i}) \wedge Constraint(\max(Faultouts))$; /*将 $DI_{i1}$ 编码为硬子句加入到 $M$ 中，另外加入基数约束对 $M$ 进行限制*/

⑪ end for

⑫ if $c \in DominatedComps$ then

⑬ $M = M \wedge HardCLs(c)$; /*如果是统治组件则编码为硬子句*/

⑭ else

⑮ $M = M \wedge SoftCls(c)$; /*否则编码为软子句*/

⑯ end if

⑰ $\Delta \leftarrow reinforced_MaxSAT(M)$; /*使用改进的 MaxSAT 求解器搜索诊断解*/

⑱ if $|\Delta| = \max(Faultouts)$/*当诊断解的基数和故障输出数量相同时，进行验证*/

⑲ for $i \in \{|\Delta|-1, |\Delta|-2, \cdots, 0\}$

⑳ $\Omega_i \leftarrow SAT(M_{\text{hard}} \wedge Comps \wedge Constraint(i))$; /*调用 SAT 求解器继续向下搜索*/

㉑ if unsatifiable $\in \Omega_i$

㉒ return $\Omega_{i+1}$;

㉓ end if

㉔ end for

㉕ end if

㉖ return $\Delta$.

具体的 CCM 方法如算法 2 所示，其中被改进的地方为行⑤⑩，⑰~㉔. 在行①中，通过将电路系统中的组件和线路匹配为整型数得到子系统模型 $M_1$. 行②~⑨中，迭代计算 $SysIns_1$ 和 $SysIns_i$ 的差别变量，得到 $P\text{-}Nodes(D_{1i})$ 和 $P\text{-}Edges(D_{1i})$，编码为新的整型数，并对每组观测计算其故障输出. 在行⑩⑪中，将子系统编码为硬子句，同时将最高故障输出作为基数约束加入到硬子句中，其中，基数约束采用 Totalizer[36] 的编码方法. 在行⑫~⑯中，将被统治组件编码为硬子句（设定其为健康的），将其他组件编码为软子句. 行⑰中，采用单次入队方法对 MaxSAT 求解器进行改进后，通过调用 MaxSAT 求解器得到多观测的诊断. 在行⑱之后，加入验证：如果得到的多观测诊断基数等于故障输出最大值，则调用 SAT 求解器继续向下搜索，直至 SAT 求解器返回 unsatisfiable，则极小势诊断为 SAT 最后一次的返回结果 $\Omega_{i+1}$.

## 5 实验结果

在本节中，我们将 CCM 方法与目前最好的处理多观测 MBD 问题的方法 D-CMMO[24] 进行了对比. 在编码为 MaxSAT 后求解诊断问题时，选择与 D-CMMO 相同的 MaxSAT 求解器 RC2[14]，并且使用单次入队方法对 RC2 进行了改进，其中，RC2 是 2018 年的 MaxSAT 评估中加权表现最好的 MaxSAT 求解器. 实验建立

在 ISCAS85 基准用例[37]和 ITC99 基准用例[26]上，这 2 组诊断用例也在文献 [14, 24, 38] 中使用. 本文使用文献 [24] 中提及的方法生成测试用例，即对于每一个电路，通过建立 3 种故障模型来评估实验，分别为：固定为 0(Stack-At-0)、固定为 1(Stack-At-1) 和翻转(FLIP)等故障. 对于 ISCAS85 基准用例，在每种故障模型上分别生成 330 个故障电路结构，对于 ITC99 基准用例，在每种故障模型上分别生成 810 个故障电路结构. 在注入故障时，随机选择 $K$ 个组件注入故障，且 $K$ 的范围为 10~50. 实验在 Ubuntu 20.04.5 LTS 和 11th Gen Intel® Core™ i7-11700KF@3.60 GHz × 2, 12 GB RAM 上进行. 图 3~5 分别给出 D-CMMO 方法和 CCM 方法在 3 种故障模型下，求解 ISCAS85 基准用例时，得到一个极小势诊断解的运行时间.

如图 3~5 所示，在实验中，设置 MaxSAT 求解时间的上限为 200 s. 对于大多数测试用例，D-CMMO

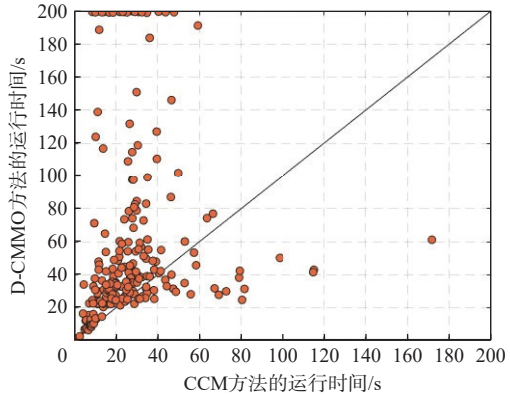



Fig. 3 Comparison of run time of D-CMMO and CCM methods on ISCAS85 benchmark in the mode of Stack-At-0

图 3 D-CMMO 和 CCM 方法在 Stack-At-0 模式下 ISCAS85 基准上的运行时间比较

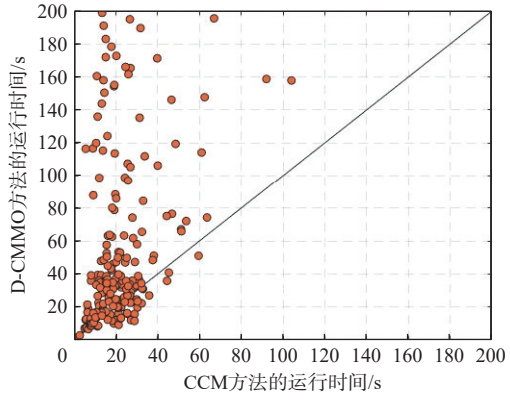



Fig. 4 Comparison of run time of D-CMMO and CCM methods on ISCAS85 benchmark in the mode of Stack-At-1

图 4 D-CMMO 和 CCM 方法在 Stack-At-1 模式下 ISCAS85 基准上的运行时间比较

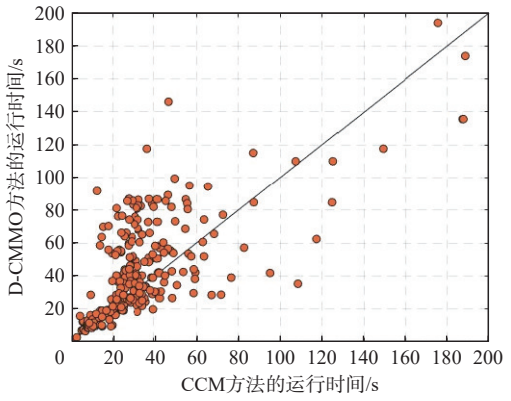



Fig. 5 Comparison of run time of D-CMMO and CCM methods on ISCAS85 benchmark in the mode of FLIP

图 5 D-CMMO 和 CCM 方法在 FLIP 模式下 ISCAS85 基准上的运行时间比较

和 CCM 可以在 200 s 内通过 MaxSAT 求解器返回诊断结果. 并且对于大多数测试实例，CCM 的运行时间要小于 D-CMMO，3 种故障模型中，CCM 运行时间较少，所占的比例分别为 82.5%, 80.9%, 60.9%. 在平均情况下，CCM 求解效率相比 D-CMMO 提升 64.5%. 图 6~8 分别显示了在 3 种故障模型下，D-CMMO 和 CCM 求解 ITC99 用例时，得到一个极小势诊断所用的时间.

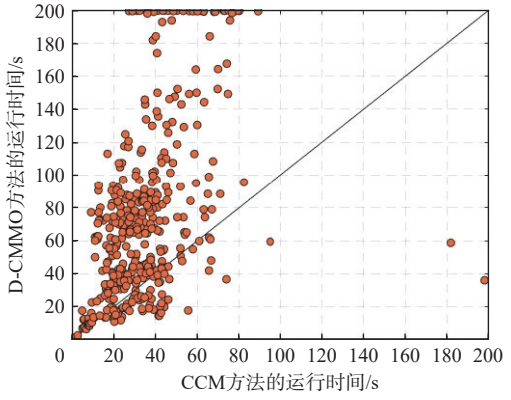



Fig. 6 Comparison of run time of D-CMMO and CCM methods on ITC99 benchmark in the mode of Stack-At-0

图 6 D-CMMO 和 CCM 方法在 Stack-At-0 模式下 ITC99 基准上的运行时间比较

如图 6~8 所示，在坐标轴上的点表示一方在给定时间内无法求解的一些测试用例. 使用 CCM 和 D-CMMO 不能得到一个解的用例个数分别为 117 和 144. 在求解实例的个数上，CCM 优于 D-CMMO. 除此之外，在求解效率上，CCM 的运行速度相比 D-CMMO 平均提升 92.8%.

在 ISCAS85 和 ITC99 实例上的详细实验结果分别如表 1 和表 2 所示.

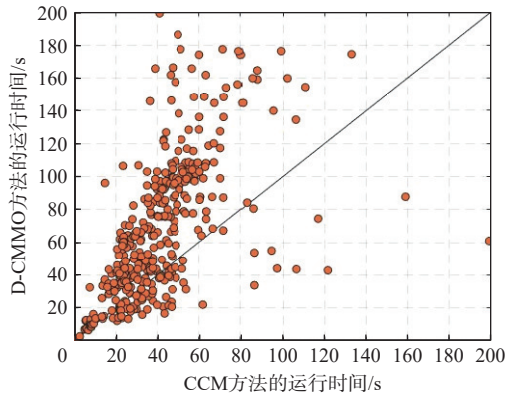



Fig. 7 Comparison of run time of D-CMMO and CCM methods on ITC99 benchmark in the mode of Stack-At-1

图 7 D-CMMO 和 CCM 方法在 Stack-At-1 模式下 ITC99 基准上的运行时间比较

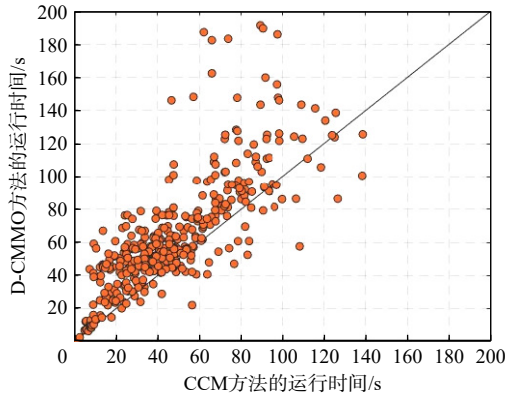



Fig. 8 Comparison of run time of D-CMMO and CCM methods on ITC99 benchmark in the mode of FLIP

图 8 D-CMMO 和 CCM 方法在 FLIP 模式下 ITC99 基准上的运行时间比较

**Table 1 Solved Results in the Example of ISCAS85**

**表 1 在 ISCAS85 实例上的求解结果**

| 电路 | 系统输入数量 | 系统输出数量 | 组件数量 | CCM wins | | |
|---|---|---|---|---|---|---|
| | | | | Stack-At-0 | Stack-At-1 | FLIP |
| c17 | 5 | 2 | 6 | 19 | 20 | 10 |
| c432 | 36 | 7 | 160 | 15 | 23 | 19 |
| c499 | 41 | 32 | 202 | 25 | 22 | 28 |
| c880 | 60 | 26 | 383 | 12 | 14 | 11 |
| c1355 | 41 | 32 | 546 | 23 | 20 | 19 |
| c1908 | 33 | 25 | 880 | 22 | 25 | 18 |
| c2670 | 233 | 140 | 1 193 | 27 | 19 | 15 |
| c3540 | 50 | 22 | 1 669 | 21 | 24 | 14 |
| c5315 | 178 | 123 | 2 307 | 19 | 17 | 18 |
| c6288 | 32 | 32 | 2 416 | 16 | 13 | 3 |
| c7552 | 207 | 108 | 3 512 | 13 | 11 | 10 |

**Table 2 Solved Results in the Example of ITC99**

**表 2 在 ITC99 实例上的求解结果**

| 电路 | 系统输入数量 | 系统输出数量 | 组件数量 | CCM wins | | |
|---|---|---|---|---|---|---|
| | | | | Stack-At-0 | Stack-At-1 | FLIP |
| b01 | 2 | 2 | 39 | 35 | 44 | 31 |
| b02 | 1 | 1 | 24 | 30 | 35 | 30 |
| b03 | 4 | 4 | 144 | 41 | 37 | 37 |
| b04 | 11 | 8 | 632 | 37 | 40 | 43 |
| b05 | 1 | 36 | 821 | 38 | 34 | 45 |
| b06 | 2 | 6 | 49 | 30 | 28 | 43 |
| b07 | 1 | 8 | 380 | 44 | 39 | 38 |
| b08 | 9 | 4 | 157 | 45 | 34 | 29 |
| b09 | 1 | 1 | 146 | 44 | 33 | 44 |
| b10 | 11 | 6 | 174 | 40 | 43 | 28 |
| b11 | 7 | 6 | 662 | 41 | 44 | 42 |
| b12 | 5 | 6 | 963 | 35 | 42 | 44 |
| b13 | 10 | 10 | 310 | 33 | 27 | 34 |
| b14 | 32 | 54 | 8 567 | 36 | 39 | 21 |
| b15 | 36 | 70 | 7 922 | 38 | 42 | 29 |
| b20 | 32 | 22 | 17 158 | 28 | 21 | 34 |
| b21 | 32 | 22 | 17 482 | 25 | 24 | 25 |
| b22 | 32 | 22 | 25 460 | 21 | 25 | 24 |

在 2 组表中第 5~7 列显示在不同的故障模型下，CCM 求解诊断时所用时间比 D-MMO 短的实例个数（CCM wins），当 2 种方法的运行时间都超过时间限制时，用例结果不计入任何一方数据. 从表 1 和表 2 可以看出，对每一个电路进行诊断搜索时，CCM 都是可行的，且求解结果显示，相比于 D-CMMO，CCM 具有明显优势. 特别是在 c499 电路上，CCM 在 3 种故障模型下的测试结果都远超 D-CMMO，94.9% 的实例的实验结果优于 D-CMMO.

## 6 总　　结

目前，很多基于 SAT 的多观测 MBD 问题把 MaxSAT 编码作为分析多观测 MBD 问题的一个基本步骤. 本文在面向统治组件的研究基础上，提出一种基于基数约束的求解方法 CCM，显著缩小了 MBD 问题的搜索范围，进而降低了 MaxSAT 问题的求解难度和求解诊断的时间. 本文提出了 2 种方法用于提高 CCM 的效率. 首先，根据电路系统的结构特性，通过传播系统输入得到诊断解的上界，作为硬子句加入到 MaxSAT 编码中. 其次，利用单次入队的方法避免对假设集重复赋值以及对子句中的文字重复访问

的代价,从而优化 MaxSAT 求解器,提高求解的效率. 实验结果表明,通过加入基数约束以及单次入队策略,能有效减少 MaxSAT 求解器搜索诊断的难度,进而提高求解诊断的效率. CCM 在 ISCAS85 和 ITC99 基准用例上求解高效,相对于 D-CMMO 在求解效率上分别提升 64.5% 和 92.8%. 在未来研究中,将探索强故障模型下的 CCM 的扩展算法.

**作者贡献声明**:青杨提出了算法思路和实验方案、负责论文的撰写与修订;欧阳丹彤确定综述选题、指导和督促论文的撰写与修改;周慧思负责文献资料的整理与分析以及部分图表的绘制;张立明提出论文指导意见与修改方案.

## 参 考 文 献

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

[2] Friedrich G, Gottlob G, Nejdl W. Hypothesis classification, abductive diagnosis and therapy[C]//Proc of the Expert Systems in Engineering Principles and Applications. Berlin: Springer, 1990: 69–78

[3] Feldman A, Pill I, Wotawa F, et al. Efficient model-based diagnosis of sequential circuits[C]//Proc of the 34th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2020: 2814–2821

[4] Struss P, Price C. Model-based systems in the automotive industry[J]. AI Magazine, 2004, 24(4): 17−34

[5] Ardissono L, Console L, Goy A, et al. Cooperative model-based diagnosis of web services[C]//Proc of the 16th Int Workshop on Principles of Diagnosis. Paris: CEUR-WS, 2005: 125–130

[6] Jannach D, Schmitz T. Model-based diagnosis of spreadsheet programs: A constraint-based debugging approach[J]. Automated Software Engineering, 2016, 23(1): 105−144

[7] Jose M, Majumdar R. Cause clue clauses: Error localization using maximum satisfiability[J]. ACM SIGPLAN Notices, 2011, 46(6): 437−446

[8] Feldman A, Provan G M, Van Gemund A J C. Computing minimal diagnoses by greedy stochastic search[C]//Proc of the 23rd AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2008: 911–918

[9] Siddiqi S A, Huang Jinbo. Hierarchical diagnosis of multiple faults[C]//Proc of the 20th Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 2007: 581–586

[10] Jannach D, Schmitz T, Shchekotykhin K M. Parallelized hitting set computation for model-based diagnosis[C]//Proc of the 29th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2015: 1503–1510

[11] Ansótegui C, Bonet M L, Levy J. SAT-based MaxSAT algorithms[J]. Artificial Intelligence, 2013, 196(1): 77−105

[12] Feldman A, Provan G, de Kleer J. Solving model-based diagnosis problems with Max-SAT solvers and vice versa[C]//Proc of the 21st Int Workshop on Principles of Diagnosis. New York: PHM Society, 2010: 185–192

[13] Stern R T, Kalech M, Feldman A. Exploring the duality in conflict-directed model-based diagnosis[C]//Proc of the 26th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2012: 828–834

[14] Ignatiev A, Morgado A, Weissenbacher G, et al. Model-based diagnosis with multiple observations[C]//Proc of the 28th Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 2019: 1108–1115

[15] Lamraoui S, Nakajima S. A formula-based approach for automatic fault localization of imperative programs[C]//Proc of the 16th Int Conf on Formal Engineering Methods. Berlin: Springer, 2014: 251–266

[16] Lamraoui S, Nakajima S. A formula-based approach for automatic fault localization of multi-fault programs[J]. Journal of Information Processing, 2016, 24(1): 88−98

[17] Narodytska N, Bacchus F. Maximum satisfiability using core-guided MaxSAT resolution[C]//Proc of the 28th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2014: 2717–2723

[18] Cai Shaowei, Lei Zhendong. Old techniques in new ways: Clause weighting, unit propagation and hybridization for maximum satisfiability[J]. Artificial Intelligence, 2020, 287: 103354−103354

[19] Marques-Silva J, Heras F, Janota M, et al. On computing minimal correction subsets[C]//Proc of the 23rd Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 2013: 615–622

[20] Previti A, Mencía C, Järvisalo M, et al. Premise set caching for enumerating minimal correction subsets[C]//Proc of the 32nd AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2018: 6633–6640

[21] Zhou Huisi, Ouyang Dantong, Zhang Liming, et al. Model-based diagnosis with improved implicit hitting set dualization. Forthcoming[J]. Applied Intelligence, 2022, 52(2): 2111−2118

[22] Kalech M, Stern R, Lazebnik E. Minimal cardinality diagnosis in problems with multiple observations[J]. Diagnostics, 2021, 11(5): 780−780

[23] Gu Jun, Purdom P W, Franco J, et al. Algorithms for the Satisfiability Problem[M]//Handbook of Combinatorial Optimization. Berlin: Springer, 1999: 379–572

[24] Zhou Huisi, Ouyang Dantong, Zhao Xiangfu, et al. Two compacted models for efficient model-based diagnosis[C]//Proc of the 34th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2022: 3885−3893

[25] Metodi A, Stern R, Kalech M, et al. A novel SAT-based approach to model based diagnosis[J]. Journal of Artificial Intelligence, 2014, 51(1): 377−411

[26] Marques-Silva J , Mikoláš J, Ignatiev A , et al. Efficient model based diagnosis with maximum satisfiability[C]//Proc of the 24th Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 2015: 1966−1972

[27] Amit M, Roni S, Meir K, et al. Compiling model-based diagnosis to Boolean satisfaction[C]//Proc of the 26th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2012: 793−799

[28] Thomas L, Robert E T. A fast algorithm for finding dominators in a flowgraph[J]. ACM Transactions Programming Languages and Systems, 1979, 1(1): 121−141

[29] Liu Meng, Ouyang Dantong, Liu Bowen, et al. Grouped diagnosis approach using the feature of problem[J]. Acta Electronica Sinica, 2018, 46(3): 589−594 (in Chinese)
(刘梦, 欧阳丹彤, 刘伯文, 等. 结合问题特征的分组式诊断方法[J] 电子学报, 2018, 46(3): 589−594)

[30] Ouyang Dantong, Liu Bowen, Liu Meng, et al. A block-based diagnostic method combing with the circuit structure[J]. Acta Electronica Sinica, 2018, 46(7): 1571−1577 (in Chinese)
(欧阳丹彤, 刘伯文, 刘梦, 等. 结合电路结构基于分块的诊断方法[J] 电子学报, 2018, 46(7): 1571−1577)

[31] Zhou Huisi, Ouyang Dantong, Tian Xinliang, et al. A novel encoding for model based diagnosis[J]. Journal of Computer Research and Development, 2023, 60(1): 95−102 (in Chinese)
(周慧思, 欧阳丹彤, 田鑫亮, 等. 基于模型诊断的一种新编码方法[J] 计算机研究与发展, 2023, 60(1): 95−102)

[32] Siddiqi S A, Huang Jinbo. Sequential diagnosis by abstraction[J]. Journal of Artificial Intelligence Research, 2011, 41: 329−365

[33] Eèn N, Sörensson N. Temporal induction by incremental SAT solving[J]. Electronic Notes in Theoretical Computer Science, 2003, 89(4): 543−560

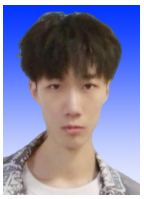
[34] Eèn N, Sörensson N. An extensible SAT-solver[C]//Proc of the 6th Theory and Applications of Satisfiability Testing. Berlin: Springer, 2004: 502−518

[35] Audemard G, Lagniez J M, Simon L. Improving Glucose for incremental SAT solving with assumptions: Application to MUS extraction[C]//Proc of the 16th Theory and Applications of Satisfiability Testing. Berlin: Springer, 2013: 309–317

[36] Ogawa T, Liu Yangyang, Hasegawa R, et al. Modulo based CNF encoding of cardinality constraints and its application to MaxSAT solvers[C]//Proc of the 25th IEEE Int Conf on Tools with Artificial Intelligence. Piscataway, NJ: IEEE, 2013: 9−17

[37] Brglzz F, Fujtwara H. A neutral netlist of 10 combinational benchmark circuits and a target translator in FORTRAN[J/OL]. IEEE International Symposium on Circuits and Systems. 1985[2023-03-20]. https://www.researchgate.net/publication/239562831_A_neutral_netlist_of_10_combinational_circuits_and_a_target_translator_in_fortran

[38] Siddiqi S A. Computing minimum-cardinality diagnoses by model relaxation[C]//Proc of the 22nd Int Joint Conf on Artificial Intelligence. Palo Alto, CA : AAAI, 2011: 1087–1092



**Qing Yang**, born in 2000. Master candidate. His main research interest includes model-based diagnosis.

**青　杨**, 2000 年生. 硕士研究生. 主要研究方向为基于模型的诊断.



**Ouyang Dantong**, born in 1968. PhD, professor, PhD supervisor. Senior member of CCF. Her main research interests include the model-based diagnosis, SAT problem, and model checking.

**欧阳丹彤**, 1968 年生. 博士, 教授, 博士生导师, CCF 高级会员. 主要研究方向为基于模型的诊断、可满足性问题、模型检测.



**Zhou Huisi**, born in 1995. PhD. Her main research interests include MaxSAT, model-based diagnosis.

**周慧思**, 1995 年生. 博士. 主要研究方向为最大可满足性问题、基于模型的诊断.



**Zhang Liming**, born in 1980. PhD, senior engineer. Senior member of CCF. His main research interests include model-based diagnosis and SAT problem.

**张立明**, 1980 年生. 博士, 高级工程师. CCF 高级会员. 主要研究方向为基于模型诊断、可满足性问题.