

Computation of Hitting Sets with Intersection Search Strategy in Model-based Diagnosis

Yaxiong Zhu¹

*Department of Management Engineering
Ordnance Engineering College
Shijiazhuang, Hebei, 050003, China
E-mail: zyx_0406@sina.cn*

Xingxin Li

*Department of Management Engineering
Ordnance Engineering College
Shijiazhuang, Hebei, 050003, China*

Zhimeng Li

*Department of Management Engineering
Ordnance Engineering College
Shijiazhuang, Hebei, 050003, China*

In the diagnosis process of fault diagnosis simulation system based on model, the computation of minimal hitting sets by the minimal conflict sets is a critical step in the entire process. In consideration of the defects of computing the minimal hitting sets with logic array process in term of time and space, the way of intersection search strategy which computes the minimal hitting sets in combination with backtracking is proposed while a frequency sorting is adopted to search the intersection and other ways to improve the operational efficiency and obtain the final minimal hitting sets. An example is given to prove that the algorithm can improve the diagnostic efficiency in terms of time and space with optimization to a certain extent.

*CENet2015
12-13 September 2015
Shanghai, China*

¹Speaker

1. Introduction

When the model-based reasoning is adopted to diagnosis faults, we should generate the minimal conflict sets based on the structure and behavior model of the system under diagnosis, then, compute the minimal hitting sets of the minimal conflict sets. Now, as the algorithm of computing minimal hitting sets has been researched by many scholars, many kinds of algorithms have been found. The HS-tree algorithm was put forward by Reiter[1] in the beginning; therefore, Greiner[2] proposed the HS-DAG algorithm to improve the flaw of HS-tree and Wotawa[3] proposed the HST-tree algorithm. In order to improve the defect of the improved algorithm, Yunfei Jiang[4-5] built the BHS-tree algorithm and boolean formulas algorithm. Lin li[6-7] proposed the RHS-tree algorithm and the logic array algorithm. Shaowen Fu[8] proposed the minimal hitting sets algorithm based on sets iteration in order to improve the efficiency. Xiangfu Zhao[9] proposed the HSSE to avoid missing the hitting sets, but the efficiency of this way was fair. Xiaomeng Chen[10] proposed the BNB-HSSE way that combined HSSE with branch and bound, but this way was very complex. Xiao Wang[11] proposed the CHS-tree method based on cardinality, this method featured higher computational efficiency than HS-tree, BHS-tree and HSSE-tree, its efficiency was higher than the most efficient Boolean method at present in some cases.

The Literature "Calculated based on the model of diagnosis minimal hitting sets with logic array" which used a two-dimensional array of logic "and" and "or" calculation cycle gradually method for the purpose of solving the hitting sets is not strong, and may be relatively large in solving the resulting number superset minimal hitting sets. The results need to be minimized again, leading to time inefficiency. The time and space there still have some room for improvement. In this paper, an improved algorithm based on the logic array algorithm is put forward to optimize the time and space.

2. Computing Flow

Through the minimum conflict sets of $MCS = \{C_1, C_2, \dots, C_N\}$ to calculate the minimal hitting sets of $MHS = \{H_1, H_2, \dots\}$, note $\cup MCS = \{C_1 \cup C_2 \cup \dots \cup C_N\} = \{c_1, c_2, \dots, c_M\}$. If the initial conflict sets are not the minimal conflict sets, the superset of the conflict sets have to be deleted, turning them to the minimal conflict sets, can greatly reduce the computational complexity and the number of operations (all numbers herein M, N, i, j, l are positive integers).

(1) The element w_{c_i} represents the frequency of c_i appearing in the minimum conflict sets of MCS . In accordance with the size of w_{c_i} , from big to small, reorder the element of $\cup MCS$ to get $\cup MCS' = \{C_1 \cup C_2 \cup \dots \cup C_N\} = \{W_1, W_2, \dots, W_M\}$, in which, the element W_i represents that the frequency is equal to i corresponding to the magnitude order of the elements and the element w_{W_i} represents the frequency of W_i .

(2) Two-dimensional array $A_{M \times N}$ represents MCS' , using $B[k]$ to store the hitting sets of $H = \{B[1], B[2], \dots, B[k], \dots\}$ which are calculating. The first column to the N column of Array A represents the conflict sets of C_1, C_2, \dots, C_N , the first line to the M line of Array A which represents the element W_1, W_2, \dots, W_M of conflict sets. Two-dimensional array in which each element is assigned as in Equation 1.

$$\begin{cases} A[i][j] = 1, (1 \leq i \leq M; 1 \leq j \leq N; W_i \in C_j) \\ A[i][j] = 0, (1 \leq i \leq M; 1 \leq j \leq N; W_i \notin C_j) \end{cases} \quad (1)$$

(3) Calculate the value of l which meets $w_{W_M} + w_{W_{M-1}} + w_{W_{M-2}} + \dots + w_{W_{M-l}} < N$, $(1 \leq l < M)$, and calculate the l_{\max} which is satisfied with the condition, the apparent absence of the hitting sets of $B[1] = W_{M-l}$.

$$B[k] = W_i (1 \leq k \leq M; W_i \in \square \cup MCS'; 1 \leq i \leq M; B[1] \in \{W_1, W_2, \dots, W_{(M-l_{\max})}\})$$

(4) Judge the value w_{w_i} , if $w_{w_i} = N$, $\{W_i\}$ is a hitting sets, the output is $\{W_i\}$. If $w_{w_i} < N$, proceed to Step (5).

(5) If $k=1$, $B[k] \notin \{W_1, W_2, \dots, W_{(M-l_{\max})}\}$, end the program; otherwise, proceed to step (6).

(6) If $B[k] \notin \{W_1, W_2, \dots, W_M\}$ when $k > 1$, or $k > N$, remove the last element $B[k]$ from H , $B[k-1] = W_{i+1}$, proceed to step (5); otherwise, proceed to Step (7).

(7) Search the row of j in $B[k]$ which the value is still not equal to 1, if there is no row of j in $B[k]$ which is equal to 1, so $B[k] = W_{i+1}$, proceed to Step (5). If the row j in $B[k]$ exists in the searched value of 1, then add a new element $B[k+1] = W_{i+1}$, proceed to Step (5). Otherwise, proceed to Step (8).

(8) If $\{B[1], B[2], \dots, B[k]\}$ is a hitting sets, output. Calculate the next hitting sets $\{B[1], B[2], \dots, B[k] = W_{i+1}\}$, proceed to Step (5).

The algorithm is completed.

3. Case Analysis

Given $MCS = \{C_1, C_2, C_3, C_4, C_5\} = \{\{1, 3, 4, 6\}, \{1, 2, 3\}, \{1, 4, 5\}, \{1, 5, 6\}, \{1, 3, 5\}\}$, calculate the minimum hitting sets.

$UMCS = \{c_1, c_2, c_3, c_4, c_5, c_6\} = \{1, 2, 3, 4, 5, 6\}$, calculate the minimum number of frequent w_{c_i} of conflict sets of MCS ; and we can get $w_{c_1} = 5$, $w_{c_2} = 1$, $w_{c_3} = 3$, $w_{c_4} = 2$, $w_{c_5} = 3$, $w_{c_6} = 2$, then,

$$UMCS' = \{W_1, W_2, W_3, W_4, W_5, W_6\} = \{1, 3, 5, 4, 6, 2\}. \text{ Two-dimensional array } A_{6 \times 5} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix},$$

$w_{w_6} = 1 < 5$, $w_{w_6} + w_{w_5} = 3 < 5$, $w_{w_6} + w_{w_5} + w_{w_4} = 5 = 5$, so $l_{\max} = 1$, so $B[1] = W_i$ ($i=1$ or 2 or 3 or 4).

Let $B[1] = W_1 = 1$, because $w_{c_1} = N = 5$, so $A[1][1] = 1$, $A[1][2] = 1$, $A[1][3] = 1$, $A[1][4] = 1$, thus $\{W_1\} = \{1\}$ is one of the hitting sets. In accordance with the principle of (4), go to the next operation directly.

Let $B[1] = W_2 = 3$, because $A[2][1] = 1$, $A[2][2] = 1$, $A[2][5] = 1$, $A[2][3] \neq 1$, $A[2][4] \neq 1$, add a new element $B[2] = W_3 = 5$, and search the value of $A[3][3]$ and $A[3][4]$.

Because $A[3][3] = 1$, $A[3][4] = 1$, $\{W_2, W_3\} = \{3, 5\}$ is one of the hitting sets. Keep $B[1] = W_2 = 3$, let $B[2] = W_4 = 4$, search the value of $A[4][3]$ and $A[4][4]$. Because $A[4][3] = 1$,

$A[4][4] \neq 1$ then let $B[3] = W_5 = 6$, search the value of $A[5][4]$. Because $A[5][4] = 1$, $\{W_2, W_4, W_5\} = \{3, 4, 6\}$ is one of the hitting sets. Keep $B[1] = W_2 = 3$ and $B[2] = W_4 = 4$, let

$B[3] = W_6 = 2$, search the value of $A[6][4]$ and $A[6][4] \neq 1$; thus $\{W_2, W_4, W_6\} = \{3, 4, 2\}$ can not constitute hitting sets. Keep $B[1] = W_2 = 3$, let $B[2] = W_5 = 6$, search the value of $A[5][3]$ and

$A[5][4]$. Because $A[5][3] \neq 1$ and $A[5][4] = 1$, let $B[3] = W_6 = 2$, search the value of $A[6][3]$ and $A[6][3] \neq 1$, so $\{W_2, W_5, W_6\} = \{3, 6, 2\}$ can not constitute the hitting Sets. Keep

$B[1] = W_2 = 3$, let $B[2] = W_6 = 2$, search the value of $A[6][3]$ and $A[6][4]$. Because

$A[6][3] \neq 1$, $A[6][4] \neq 1$, so $\{W_2, W_6\} = \{3, 2\}$ is not a hitting sets; thus it is no longer the hitting sets headed with W_2 and go to the next round of operations.

Similarly, let $B[1]=W_3=5$, $\{W_3, W_4, W_6\}=\{5, 4, 2\}$ and $\{W_3, W_5, W_6\}=\{5, 6, 2\}$ is one of the hitting sets. Let $B[1]=W_4=4$, we can conclude that hitting sets is not currently headed with W_4 .

Because $B[1]=W_i$ ($i=1$ or 2 or 3 or 4), operation is completed.

The sets $HS = \{\{1\}, \{3, 5\}, \{3, 4, 6\}, \{5, 4, 2\}, \{5, 6, 2\}\}$ is the hitting sets of the minimum conflicting sets MCS , then delete the superset of hitting sets and minimize the hitting sets, and judge the hitting sets obtained by this algorithm just for minimal hitting sets, namely $MHS = \{\{1\}, \{3, 5\}, \{3, 4, 6\}, \{5, 4, 2\}, \{5, 6, 2\}\}$.

4. Comparison and Analysis

Calculate according to the logic array algorithm and we can get the hitting sets $HS = \{\{1\}, \{2, 3, 4, 5\}, \{2, 3, 4, 6\}, \{2, 3, 5\}, \{2, 4, 5\}, \{2, 5, 6\}, \{3, 4, 5\}, \{3, 4, 6\}, \{3, 5\}\}$, then delete the superset of hitting sets and minimize the hitting sets, finally get minimal hitting sets $MHS = \{\{1\}, \{3, 5\}, \{3, 4, 6\}, \{2, 4, 5\}, \{2, 5, 6\}\}$. Because the computation certain blindness, it greatly increases the number of operations and hitting sets; and this instance generates 180% hitting sets.

In order to demonstrate the effectiveness of this algorithm, the comparison simulation has been done. The environment of simulations includes: Intel Core (TM) i5-3470, CPU 3.2GHz, memory 3.49GB, windows 07(32 bit), Visual C ++6.0. Random array is used to compare the efficiency of this algorithm with the logic array algorithm. The total number of conflict sets and the number of elements is n , each set is generated randomly without duplicated elements. The comparison of running time is shown in Fig. 1.

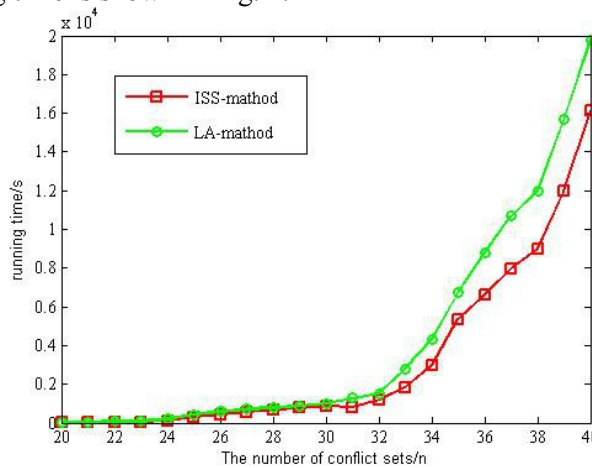


Figure 1: Comparison of Running Time between Two Different Algorithms

From the figure, we can conclude that this method is superior to the logic array method no matter how large the number of conflict sets is. The running time is determined by the number of conflict sets.

The analysis of the case shows that when calculating the minimum hitting sets with logic array method and the intersection search method proposed in this paper, the latter will have superset of minimal hitting sets. Because the method of logic array does not set any processing of conflict, leading to the possibility of superset appear of minimal hitting sets complicated situation, it will inevitably lead to a substantial increase in the number hitting sets.

As to the reasons for the proposed intersection search method when calculating the minimum hitting sets, the production of superset minimal hitting sets can be summarized as follows: the current $n(n \geq 2)$ elements still exists is not 1, they are not equal to 1 as to all columns; if it continues to bring about a element and correspond to the column is equal to 1, as contained by the n elements in front of the corresponding column is equal to 1, as shown in

Equation 2, it may produce a superset of minimal hitting sets .

$$\{L_{iS_1}, L_{iS_2}, \dots, L_{iS_t}\} \subset \{L_{jS_1}, L_{jS_2}, \dots, L_{jS_t}\} \subset \{S_1, S_2, \dots, S_t\} \tag{2}$$

in which:

i, j —represents the value of the row of the element.

S_t —represents the value of the column of the number of the first t elements is still not equal to 1 in the former n elements.

L —represents the value of the column of meets the number of the element is equal to 1.

1	1	1	0	0	1
2	0	1	1	0	0
3	0	0	1	1	0
4	1	0	0	0	0
5	0	0	0	1	0

Table 1: Indicates of the Minimum Conflicts with An Array of Sets 0-1

An example such as shown in Table 1, according to the way of intersection search strategy, the hitting sets of $\{1,2,3\}$ is a superset of $\{1,3\}$. The third and fourth column in the element one row is not equal to 1, so $S_1=3, S_2=4$. The third column in the element two row is not equal to 1, but the fourth column is 1, so $L_{23}=3$. The third and fourth column in the element three row is equal to 1, so $L_{33}=3, L_{34}=4$. From what has been discussed above, $\{L_{23}\} \subset \{L_{33}, L_{34}\} \subset \{3, 4\}$, it leads to the creation superset $\{1,2,3\}$ of the hitting sets of $\{1,3\}$.

From the above analysis, it shows that the intersection of the search strategy by adding constraints and determining the conditions to avoid hitting sets of minimum superset as generated, and the minimal hitting sets can be obtained directly.

5. Conclusion

The intersection search strategy to calculate the minimum hitting sets in combination with the backtracking are proposed. Firstly, the frequency of elements has been calculated, the frequency of element reflects this element corresponding to LRU important degree structure in diagnostic subsystem. Higher frequency number indicates higher importance and greater impact on the functional integrity of the subsystem. Secondly, according to the frequency of the order of the elements reorder with series of optimization measures adopted to reduce the generation of a large number of minimal hitting sets superset, reduce the amount of logic operations, but also makes the operation more targeted and purposeful. Finally, the algorithm is illustrated by specific operational steps and ideas with the reasons of superset generating analyzed in comparison with the method of logic array. The algorithm can get the number of operations and the resulting collision clusters are superior to logic array algorithm in terms of time and space, which have been optimized to a certain extent.

References

- [1] Reiter R. *A theory of diagnosis from first principles* [J]. Artificial Intelligence, 1987, 32(1):57-96.
- [2] 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.
- [3] Wotawa F. *A variant of Reiter's hitting-sets algorithm* [J]. Information Processing Letters, 2001, (79):45-51.
- [4] Yunfei Jiang, Lin Li. *With binary tree HS- computing minimal hitting sets* [J]. Journal of Software, 2002, 13 (12): 2267-2274.(In Chinese)
- [5] Yunfei Jiang, Lin Li. *Boolean algebra to calculate the minimum hitting sets* [J]. Journal of Computers, 2003, 26 (8): 919-924.(In Chinese)

- [6] Lin Li. *HS- tree recursively build computing minimal hitting sets* [J]. *Microelectronics & Computer*, 2002, (2): 7-10.(In Chinese)
- [7] Lin Li. *Calculated based on the model of diagnosis minimal hitting sets with logic array* [J]. *Jinan University (Natural Science)*, 2002, 23 (1): 24-27.(In Chinese)
- [8] Shaowen Fu, Jiankang Dong. *Minimal hitting sets algorithm based on sets iteration*[J]. *Journal of Harbin Institute of Technology*, 2004, 36(8): 1084-1086.(In Chinese)
- [9] Xiangfu Zhao, Dantong Ouyang. *A method of combining SE-tree to compute all minimal hitting sets*[J]. *Progress in Natural Science*, 2006, 16(2): 169-174.(In Chinese)
- [10] Xiaomeng Chen, Xiaofeng Meng, Renxiao Qiao. *Method of computing all minimal hitting sets based on BNB-HSSE*[J]. *Chinese Journal of Scientific Instrument*, 2010, 31(1): 61-67.(In Chinese)
- [11] Xiao Wang, Xiangfu Zhao, *Computing minimal hitting sets with CHS-tree based on cardinality*[J]. *Computer Integrated Manufacturing Systems*, 2014, 20(2): 401-406.(In Chinese)