第7卷 第2期　　自 动 化 学 报　　Vol. 7, No. 2
1981年4月　　ACTA AUTOMATICA SINICA　　April, 1981

求非完全描述状态表最小闭覆盖的图解法*

张 贤 达
（北京长城计量测试研究所）

摘　　要

目前，在综合一个时序数字系统时，尚无求非完全描述状态表最小闭覆盖的系统方法．本文提出的正是这样一种方法．

本文首先介绍如何画最大相容类隐含图，尔后给出了两种简化隐含图的方法．最后对化简的隐含图运用阱点的递归选择及回路分解，能够由最大相容类集合求出最小闭覆盖．对于比较复杂的非完全描述状态表，本图解法较之目前被广泛采用的试验法具有更为明显的优点．

一、引　　言

目前在需要综合一个时序数字系统时，对于非完全描述的状态表，尚无一种由最大相容类集合求其最小闭覆盖的系统方法．问题在于，最小闭覆盖往往并非最大相容类全体的集合，也不要求每个相容类都必须是最大的．因此，从已求出的最大相容类集合中，应该分裂与剔除哪些最大相容类，以组成一个最小闭覆盖，目前便只有采用试验性的方法了．

本文提出了解决这一问题的图解法．本方法以“最大相容类隐含图”为基础．这种图用节点表示最大相容类，用支路表示相容类之间的隐含相容关系．对经过简化的隐含图

	I_1		I_2	
	0	1	0	1
1	3,0	—,—	2,—	—,0
2	—,—	4,0	6,—	—,—
3	5,1	—,—	—,0	—,—
4	—,—	1,1	6,—	1,0
5	1,—	—,—	6,—	6,0
6	4,—	5,—	6,—	—,—
	$N(t)$, $Z(t)$			

(a) 时序机 M_1 的非完全描述状态表

2	26				
3	×	V			
4	26	×	V		
5	× 13 26	V	×15	16	
6	34 26	45	45	×15	14
	1	2	3	4	5

(b) 隐含表及其化简

图 1

* 本文修改稿于1980年11月29日收到．

应用阱点的递归选择及回路分解，可以比较简便地求得非完全描述状态表的最小闭覆盖.

最大相容类隐含图是根据非完全描述状态表的隐含表及最大相容类集合作出的. 假定某时序机 M_1 具有图 1(a) 所示的非完全描述状态表，其中，I_1、I_2 代表输入信号，序号 1—6 代表时序机内部状态. 表内的 $N(t)$、$Z(t)$ 分别表示在相应的输入作用下转入后的次态及输出值，短杠代表该项内容不确定，换言之，$N(t)$ 填短杠表示可任意取六个状态中的任一个，$Z(t)$ 填短杠表示可任意取 0 或 1. 下面简要介绍求隐含表及最大相容类集合的步骤.

1. 隐含表的作出

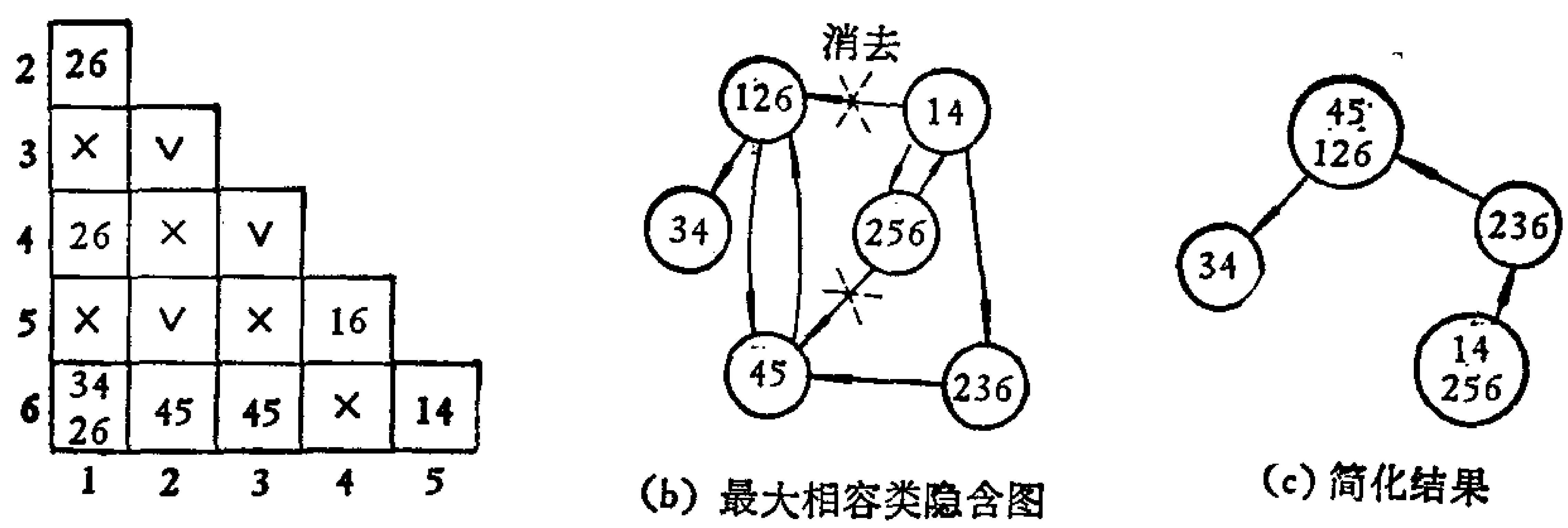
从第一行开始依次同下列各行逐一比较. 每比较两行，就在隐含表的对应方格中根据以下原则填入适当的记号.

1) 任意两个状态，如果在状态表中输出值 $Z(t)$ 不相同，则记“×”号，如 13.

2) 任意两个状态，如果 $Z(t)$ 相同或者其中之一为任意，而且对每个固定的输入所对应的二个次态也是相同的或仍为原状态对或其中之一为任意，则在相应方格中记“V”号，如 23、25 等.

3) 如果两个状态在状态表中的 $Z(t)$ 相同，而对应的次态不相同，则将这些次态填入对应的小方格. 如在 15 对应的方格内记入 13 和 26.

得到隐含表后，再检查所有的小方格. 若某一小方格内记有“×”，例如 13 和 24 对应的方格，则在所有填有 13 或 24 的小方格都记以“×”，如 15 对应的小方格. 再重复上述过程，直到所有要打“×”的小方格都打上“×”为止. 化简后的隐含表经整理后如图 2(a) 所示.



(a) 最大相容类 126
14、236、256、45、34

(b) 最大相容类隐含图

(c) 简化结果

图 2

2. 求最大相容类集合

所谓最大相容类就是不包含在别的相容类中的相容类. 求最大相容类集合时，只需关心那些相容状态对，如 56 和 34 等. 至于相容状态对对应方格内记入的那些状态对，我们暂不感兴趣，在画最大相容类隐含图时才会用到它们. 由隐含表求最大相容类集合的步骤如下：

1) 从右向左，寻找第一个相容状态对 (56).

2) 然后往左一列，如果有某一状态与前已得到的相容类中的每个状态都成对相容，则将该状态加进原相容类，组成一个更大的相容类. 如果一状态与前面已有的某个相容

类中的部分状态成对相容，则将该状态与上述部分状态一起单独列入一个新相容类．其它不包含在上述情况的相容类也列入相容类集合．

3）将步骤（2）自右向左逐列进行，直至最左一列，即得到所有最大相容类的集合．注意，最后要把与任何别的状态不相容的状态也列入最大相容类集合内．

根据图 2(a) 隐含表求最大相容类集合的上述过程可书写如下：

5 列：56

4 列：45，56

3 列：34，36，45，56

2 列：236，256，34，45

1 列：126，14，236，256，34，45

二、最大相容类隐含图

隐含表全面体现了状态表中状态对之间的相容关系．小方格内记入的“×”、“V”和状态对，分别表示该方格所对应的两个状态对之间的简单不相容、简单相容及隐含相容关系．一个富有启发性的问题是，状态对之间的相容关系既然可以用表格形式表示，那么，它们也应该可以用图的形式来体现．并且图往往比表格更直观．考虑到这种图越简洁越好，求最小闭覆盖的基础又是最大相容类集合，所以我们不拟用文献 [2] 中机械反映隐含表的“相容状态对隐含图”和模仿组合逻辑电路中质隐含概念的“质相容隐含图”，而采用“最大相容类隐含图”．

最大相容类隐含图仅由节点及支路组成．用圆圈表示节点，内记最大相容类．用带方向的支路表示隐含关系．例如，某支路从节点 A 指向节点 B，则称 A 稳含 B．

下面，我们通过一个实例介绍画最大相容类隐含图的方法．

考虑图 2(a) 所示的隐含表及最大相容类集合 (MC)．

首先，画出表示各个最大相容类的节点．

再从隐含表最右一列开始向左寻找方格中有状态对记入的第一个列．如果方格中的状态对记入为 q_mq_n，方格所对应的行为 q_i，列为 q_j，则表明 q_iq_j 隐含 q_mq_n，从而含有 q_iq_j 的最大相容类隐含含有 q_mq_n 的最大相容类．为此，从含有 q_iq_j 的节点向含有 q_mq_n 的节点画一支路．图 2(a) 中的第5列表明，状态对 56 隐含状态对 14，故从节点 256 向节点 14 画一支路．依照上述方法对各列依次进行处理．

这里，有几种特殊情况需要加以注意：

1）可能有多个节点都包含 q_iq_j（或 q_mq_n），此时应把这些节点之间的所有隐含关系都画出来．例如，图 2(a) 中状态对 14 隐含状态对 26，故从节点 14 向节点 126、236、256 各画一支路．

2）如果 q_iq_j 和 q_mq_n 包含在同一节点内，那末，即使其它节点还包含有 q_mq_n，亦不用考虑该节点与这些节点之间的隐含关系．例如，状态对 12 隐含 26，但它们均包含在节点 126 内，所以从 126 不再向节点 236、256 画支路了．这是因为若它们之间不存在着其它隐含关系的话，将 126 选入覆盖内已满足闭的要求，而不管其它两项是否选入．当然，如

果尚有其它隐含关系，则在考虑这些关系时，还应照样画出相应的支路. 注意到这一点，可避免使隐含图变得繁杂，否则将给图解增加困难.

凡标有"∨"的方格一律无须考虑. 因为简单相容关系是用从某节点出发，又返回到本节点的自回路表示的，根据回路化简的原则，自回路可消去. 从另一观点看，闭只决定于隐含相容，而与简单相容无关.

如果从一节点已有了指向另一节点的一条支路，那么，在以后需要画此两节点之间相同指向的支路时，不必重画.

在依照上述方法把所有记有状态对的方格所对应的隐含关系无一遗漏地画完之后，便得到一完整的最大相容类隐含图，见图 2(b).

三、隐含图的简化

除了注意画法可使隐含图简单外，尚有二种简化隐含图的方法.

1. 根据关系合成简化

设 ρ 为从集合 A 到集合 B 的一个关系，γ 为集合 B 到集合 C 的另一个关系，则 ρ 和 γ 的合成关系是从 A 到 C 的一个关系，其定义式为:

$$\rho\gamma=\{(a,c)|a\in A,c\in C,\exists b\in B,\text{ 使 }(a,b)\in\rho,(b,c)\in\gamma\}$$

将上述关系合成的定义推广到最大相容类隐含图，于是有:

若存在着下列隐含关系: $N_1\to N_2\to\cdots\to N_{k-1}\to N_k$ 及 $N_1\to N_k$，则称 $N_1\to N_k$ 是 $N_1\to N_2\to\cdots\to N_{k-1}\to N_k$ 关系的合成. 表示隐含关系合成的支路可消去，因为消去这一支路后，节点 N_1 隐含 N_k 的关系仍然得以保留. 换言之，若从某节点到另一节点有二条通路，一条是直接的单一支路，另一条是若干指向顺接的支路组成，则单一支路可消去. 例如，图 2(b) 中可消去节点 14 指向节点 126 及节点 256 指向节点 45 的二条直接支路.

2. 回路的简化

闭合的通路称为回路. 当一回路是由二个支路组成时，可将此回路中的二个节点合并为一个节点，圆圈内分别记入原来的二个最大相容类. 凡从原二个节点发出或指向这二个节点的支路均改接到合并后的节点，如图 2(b) 中，126 与 45 可合并，14 与 256 可合并. 但是，对于多支路组成的复杂回路，一般地不宜合并. 相反地，是对其进行分解.

值得注意的是，消去关系合成的支路，往往能把回路变成非闭合的，如图 2，而这对于应用"阱点的递归选择"是很有利的.

经过简化之后的隐含图有两种特殊节点是我们最感兴趣的. 一种是阱点即只有箭头指入而无箭头指出的节点. 另一种是某状态只出现一次的节点，我们在其边上用星号标出以便区别.

四、最小闭覆盖的图解求法

最大相容类隐含图被简化后，即可着手求最小闭覆盖.

设$(C_1, C_2, \cdots C_L)$为时序机M的最大相容类集合，$(C'_1, C'_2, \cdots C'_{L'})$为$M$的最小闭覆盖，则$(C'_1, C'_2, \cdots C'_{L'})$必须满足以下三个条件：

(1) 最小：即L'最小；

(2) 覆盖：并集 $C'_1 \cup C'_2 \cup \cdots \cup C'_{L'} = \bigcup_{i=1}^{L'} C'_i$ 包含M的全部状态.

(3) 闭：并集 $\bigcup_{i=1}^{L'} C'_i$ 内任何一个相容类 C'_i 所隐含的每一个子相容类仍是该并集中的另一个相容类或者是另一个相容类的子相容类。

命题：如果某状态表M具有一个包含最多为m个状态的最大不相容类，且其最大相容类的个数为n，则最小的L'必须满足$m \leqslant L' \leqslant n$.

证明：最大不相容类中任何两两状态都不相容，因此它的每一个状态只能且必须被覆盖M的任一状态表中的一个特定状态所覆盖，故覆盖M的任一状态表至少应具有m个相容类．又，最大相容类集合本身就是一个闭覆盖，所以L'的上限不会超过最大相容类的个数n.

最大不相容类的求法与最大相容类的求法类似，二者的区别仅在于前者着眼于记"×"的方格，而后者则如前述关心记有"∨"和状态对的方格．恕不再赘述.

从化简的隐含图求最小闭覆盖的主要准则是"阱点的递归选择"．为此，有必要先给出后继和前继的定义．从某节点N_i通过一任意长度的通路可以抵达的节点N_j称为N_i的后继．反之，N_i叫做N_j的前继.

以下三个法则可以帮助选择适当的阱点.

法则 1：如一阱点是所有其它节点的后继，则此阱点所代表的相容类必须选入到闭覆盖.

证明：显然，此相容类为所有其它相容类所隐含，因此，为使覆盖是闭的，必须选取其到闭覆盖内．否则覆盖不可能是闭的.

法则 2：阱点有多个时，应从标有＊号的节点之诸后继中，优先选取与之相距最近的阱点.

法则 3：应尽量选择这样的阱点，其前几个前继包含有较多的状态.

有趣的是，满足法则1的阱点是唯一的．因而，当图中只有一个阱点时，必须将其选入闭覆盖．其它两个法则属于技巧性的.

一俟选定了第一个阱点(一次阱点)后，下一步便只要考虑该阱点的全部前继，而不必去管其余的节点．考虑的重点则是一次阱点的相邻前继．不妨设想将此阱点移去，那末，在这些相邻前继中又可能出现新的阱点．按照上述三法则选出二次阱点．"移去"这二次阱点，又可以从它的相邻前继中选出三次阱点．如此递归地选择下去，直到恰获得一个最小闭覆盖为止．必须指出：我们只能逆着一条通路的方向依次作相邻前继的选择，中间不允许有跳跃．这样做可以自动地保证覆盖是闭的.

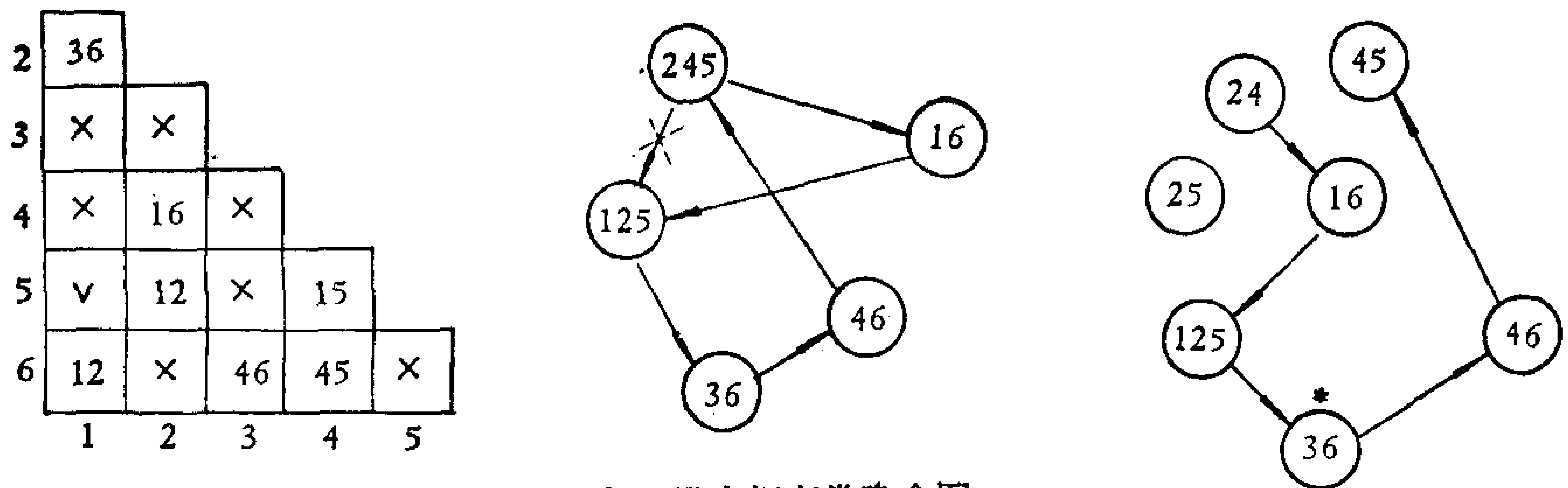
然而，由于最小闭覆盖中每个相容类不一定都是最大的，故在图解过程中，常会遇到一开始就无阱点存在，或者在某次阱点的相邻前继中不再有新的阱点的情况．也就是说，我们此时面临的只是回路.

为了使 L' 最小，我们往往不能把回路中的全部节点都选上. 此时，唯一的方法是分裂回路中的某个节点，把回路变成非闭合的，这就是所谓"回路分解".

对分解的要求显然是应该至少能产生出一个一次阱点或 k 次阱点，此 k 次阱点又恰好是分解前被选取的 $k-1$ 次阱点的相邻前继. 如果对某节点分解达不到要求，可另找其它节点分解. 由于最小闭覆盖必定存在，故分解节点总是可行的.

一般地说，只要能够达到分解的要求，分解哪一个节点都是可以的. 不过，状态个数是 2 的节点明显地不能分解. 状态个数越少的节点分解起来越简单，但状态个数多的节点分解后达到分解要求的概率要大一些. 建议为使分解简便些，先选状态个数较少的一节点分解.

分解时，首先将该节点两两状态作为一个新节点. 例如对于图 3(b)，先将节点 245 分解成三个节点 24、25 和 45. 然后看原节点 245 的后继 16，找到隐含表 (a) 中内记状态对 16 的小方格，由其对应的行和列可知是 24 隐含 16，故将原节点 245 指向 16 的支路改为由 24 指向 16. 再看原 245 的相邻前继 46，找到隐含表中行列数 46 所对应的方格，可知 46 隐含 45，所以把原 46 指向 245 的支路改为 46 指向 45. 将被分解的节点的全部后继和前继逐一处理完毕后，分解即完成.



(a) 最大相容类 245、125、46、16、36

(b) 最大相容类隐含图

(c) 分解节点 245

图 3

在分解某节点达到要求之后，又可应用上面介绍的法则递归地选择新的高次阱点. 若以后再遇到只有回路的情况，还可以再分解. 如此类推，直至所选取的全部阱点恰好包含了所有状态在内. 顺便指出，若在图解过程中选入的相容类只差一个状态就可全部覆盖时，我们便只要将此状态单个选入即可，而无须舍简就繁再去考虑阱点的选择问题了.

五、结 论

本文提出的方法比较直观，通过观察就可进行阱点的选择. 同时，由于"阱点的递归选择"要求只能逆着一条通路依次相邻进行，不仅使所要考虑的节点数大为减少，简化了求解问题，而且还能自动保证覆盖是闭的. 如果一旦不慎多选了相容类，本方法尚能使您很快从中挑选出最小的闭覆盖. 因为所选取的相容类是按照顺序书写的，只要您从第一项算起，数到那个刚好已覆盖了全部状态在内的相容类为止，就可获得了一个所需要的最小闭覆盖了. 而试验法对于一个不满足最小的闭覆盖内的相容类的舍取则无所适从，只

有重新试验．很容易求得图 2(a) 和图 3(a) 的最小闭覆盖分别为 (34, 126, 45) 和 (45, 46, 36, 125).

诚然，在分解回路时可能一次不会成功，但是，这种分解毕竟不是盲目的，即便较复杂的回路这种分解也能较快获得成功．

与目前被广泛采用的试验性方法相比，本图解法可以系统地由最大相容类集合求出最小闭覆盖．对于越复杂的非完全描述的状态表，本方法的优越性将越发明显．

参 考 文 献

[1] 北京大学计算机科学技术系编著，电子数字计算机原理(第 1 册)，科学出版社，(1979).

[2] A. D. Friedman and P. R. Menon, Theory & Design of Switching Circuits, Computer Science Press, Inc. (1975).

[3] F. H. Edwards, The Principles of Switching Circuits, The Massachusetts Institute of Technology, (1973).

A MAPPING METHOD FOR FINDING A MINIMUM CLOSED COVER OF INCOMPLETELY DEFINED STATE TABLE

ZHANG XIANDA

(*Changcheng Institute of Metrology & Measurement Technology Beijing*)

Abstract

At present, when a sequential digital system is synthesized, there is not any systematic method in respect of finding a minimum closed cover of incompletely defined state table. What this paper presents is such a systematic method.

First, it is described how to draw an implicit map of MC (maximum compatible) terms. Second, two methods of simplifying the map are given. Finally, applying the catch point's recursive selection and the loop decomposition into simplified map, we are able to find out a minimum closed cover from a set of maximum compatible terms.

For a more complex incompletely defined state table this mapping method has more obvious advantage over the try-and-error method applied widely at present.