计算机问题求解--论题1-13 --布尔代数

2015年12-31

问题1：这个是布尔代数的定义，你有没有一种熟悉的感觉?它和另一个定义有什么异同之处？你能想到什么？

Let B be a nonempty set with two binary operations $+$ and $*$, a unary operation $'$, and two distinct elements 0 and 1. Then B is called a *Boolean algebra* if the following axioms hold where a, b, c are any elements in B:

[B$_1$] Commutative laws:
(1a) $a + b = b + a$ (1b) $a * b = b * a$

[B$_2$] Distributive laws:
(2a) $a + (b * c) = (a + b) * (a + c)$ (2b) $a * (b + c) = (a * b) + (a * c)$

[B$_3$] Identity laws:
(3a) $a + 0 = a$ (3b) $a * 1 = a$

[B$_4$] Complement laws:
(4a) $a + a' = 1$ (4b) $a * a' = 0$

Axioms Defining a Lattice

Let L be a nonempty set closed under two binary operations called *meet* and *join*, denoted respectively by $\wedge$ and $\vee$. Then L is called *lattice* if the following axioms hold where a, b, c are elements in L:

[L$_1$] Commutative law:
(1a) $a \wedge b = b \wedge a$ (1b) $a \vee b = b \vee a$

[L$_2$] Associative law:
(2a) $(a \wedge b) \wedge c = a \wedge (b \wedge c)$ (2b) $(a \vee b) \vee c = a \vee (b \vee c)$

[L$_3$] Absorption law:
(3a) $a \wedge (a \vee b) = a$ (3b) $a \vee (a \wedge b) = a$

We will sometimes denote the lattice by $(L, \wedge, \vee)$ when we want to show which operations are involved.

“这个”代数和格

Theorem 15.2: Let a, b, c be any elements in a Boolean algebra B.

(i) Idempotent laws:

(5a) $a + a = a$ (5b) $a * a = a$

(ii) Boundedness laws:

(6a) $a + 1 = 1$ (6b) $a * 0 = 0$

(iii) Absorption laws:

(7a) $a + (a * b) = a$ (7b) $a * (a + b) = a$

(iv) Associative laws:

(8a) $(a + b) + c = a + (b + c)$ (8b) $(a * b) * c = a * (b * c)$

问题2：显然，这个代数一定是个格！那么：多出来的那些特性（由公理描述），有什么用呢？

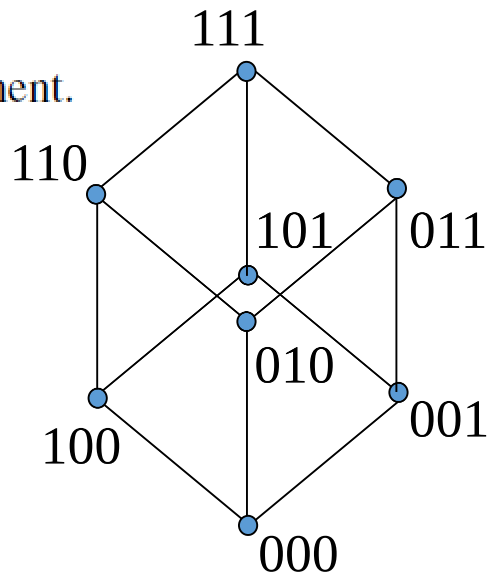
我们可以很容易的验证B3是布尔代数

(b) Let $\mathbf{B}^n = \mathbf{B} \times \mathbf{B} \times \cdots \times \mathbf{B}$ (n factors) where the operations of $+$, $*$, and $'$ are defined componentwise using Fig. 15-1. For notational convenience, we write the elements of $\mathbf{B}^n$ as n-bit sequences without commas, e.g., $x = 110011$ and $y = 111000$ belong to $\mathbf{B}^n$. Hence

$$x + y = 111011, \quad x * y = 110000, \quad x' = 001100$$

Then $\mathbf{B}^n$ is a Boolean algebra. Here $0 = 000 \cdots 0$ is the zero element, and $1 = 111 \cdots 1$ is the unit element.

问题3：从这个B3中，我们能否设计一个“类似”的格？
元素有哪些？偏序关系是什么？
meet和join分别是什么？



B3

其实，布尔代数是一类特别的格：

- 1，我们可以从布尔代数和偏序集两个角度共同定义一个“系统”
- 2，布尔代数和有界有补分配格是“等价”的

Theorem 15.5: The following are equivalent in a Boolean algebra:

$$(1)\quad a+b=b,\qquad (2)\quad a*b=a,\qquad (3)\quad a'+b=1,\qquad (4)\quad a*b'=0$$

Thus in a Boolean alegbra we can write $a \leq b$ whenever any of the above four conditions is known to be true.

问题4：

- 有界有补分配格一定有2^n个元素

因为B1，B2，B3，…,Bn有2^n个元素？

其实，我们还可以这样来观察：
1，这样的格中，原子个数是n个
2,，除0外，所有元素都可以表示为一个或者多个原子的join，所有由一个或者多个原子的join的结果都是格中元素。
3，这样的元素有2^n-1个

问题5：我们为什么要定义sum-of-products form?如何理解form是什么意思？

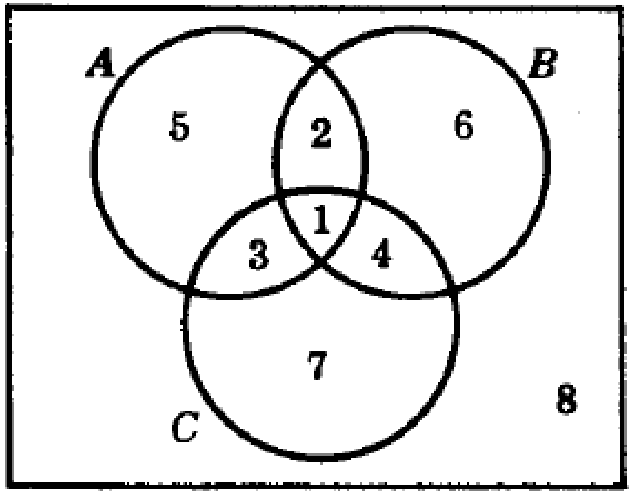


Fig. 15-3

rectangle (universal set) into eight numbered sets which can be represented as follows:

(1)	$A \cap B \cap C$	(3)	$A \cap B^c \cap C$	(5)	$A \cap B^c \cap C^c$	(7)	$A^c \cap B^c \cap C$
(2)	$A \cap B \cap C^c$	(4)	$A^c \cap B \cap C$	(6)	$A^c \cap B \cap C^c$	(8)	$A^c \cap B^c \cap C^c$

问题6：任意的布尔代数运算表达式，是否都可以表达为某个”sum-of-products form”？

$$E_1 = (x + y'z)' + (xyz' + x'y)' \quad \text{and} \quad E_2 = ((xy'z' + y)' + x'z)'$$

回到举重裁判的问题

x	y	z	$f(x,y,z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	[illegible]
1	0	1	[illegible]
1	1	0	[illegible]
1	1	1	[illegible]

相应的布尔表达式:

$(x'\wedge y\wedge z)\vee(x\wedge$ … $)\vee($ … $\wedge z')\vee(x\wedge y\wedge z)$

…什么？

…布尔代数表达式是什么？

$yz+xz+xy$

问题7：有没有科学办法，从任意的一个布尔逻辑表达式出发，得到和它等价的最简表达式？

复习几个概念：

Literal？Fundamental product？Contained in?

“惊艳”的吸收率：

x' is not a literal of $xy'z$. Observe that if P_1 is contained in P_2, say $P_2 = P_1 * Q$, then, by the absorption law,

$$P_1 + P_2 = P_1 + P_1 * Q = P_1$$

Thus, for instance, $x'z + x'yz = x'z$.

如何“化简”下列布尔代数表达式

$x'yz + xy'z + xyz' + xyz =$

$x'yz + xyz + xy'z + xyz + xyz' + xyz =$

$yz + xz + xy$

这个式子难道就是最简的吗？

Yes！

这个式子有什么特征？

sum-of-products expression

求和E等价的积和表达式

Algorithm 15.1: The input is a Boolean expression E. The output is a sum-of-products expression equivalent to E.

Step 1. Use DeMorgan's laws and involution to move the complement operation into any parenthesis until finally the complement operation only applies to variables. Then E will consist only of sums and products of literals.

Step 2. Use the distributive operation to next transform E into a sum of products.

Step 3. Use the commutative, idempotent, and complement laws to transform each product in E into 0 or a fundamental product.

Step 4. Use the absorption and identity laws to finally transform E into a sum-of-products expression.

如何去寻找（定义）最小（其实是极小）的积和表达式？

复习几个概念：

E is ***simpler*** than *F*

E is ***minim***… …ession

which is sin…

Prime Impl…

$x'yz + xy'z + xyz' + xyz =$

$x'yz + xyz + xy'z + xyz + xyz' + xyz =$

$yz + xz + xy$

Theorem 15.9: A minimal sum-of-products form for a Boolean expression *E* is a sum of prime implicants of *E*.

接下来的问题是：如何找到E的素(蕴含)项

一个比较突兀的概念： **Consensus of Fundamental Products**

Let $E = \boxed{xyz} + x'z' + \boxed{xyz'} + x'y'z + x'yz'$

你能在这个式子中找到某两个基本积的**Consensus** 吗？

Lemma 15.10: Suppose Q is the consensus of P_1 and P_2. Then $P_1 + P_2 + Q = P_1 + P_2$.

Let $E = xyz + x'z' + xyz' + x'y'z + x'yz' + xy$

问题9：你能说说看，这个寻找素项的每一步的理论基础吗？

EXAMPLE 15.8 Let $E = xyz + x'z' + xyz' + x'y'z + x'yz'$. Then:

$$
\begin{aligned}
E &= xyz + x'z' + xyz' + x'y'z && (x'yz' \text{ includes } x'z') \\
&= xyz + x'y' + xyz' + x'y'z + xy && (\text{consensus of } xyz \text{ and } xyz') \\
&= x'z' + x'y'z + xy && (xyz \text{ and } xyz' \text{ include } xy) \\
&= x'z' + x'y'z + xy + x'y' && (\text{consensus of } x'z' \text{ and } x'y'z) \\
&= x'z' + xy + x'y' && (x'y'z \text{ includes } x'y') \\
&= x'z' + xy + x'y' + yz' && (\text{consensus of } x'z' \text{ and } xy)
\end{aligned}
$$

显然，E有两个等价的素项表达，最终结果并不是极小式

怎么办？

EXAMPLE 15.9 We apply Algorithm 15.4 to the following expression E which (by Example 15.8) is now expressed as the sum of all its prime implicants:

$$E = x'z' + xy + x'y' + yz'$$

Step 1. Express each prime implicant of E as a complete sum-of-products to obtain:

$$\begin{aligned} x'z' &= x'z'(y + y') = x'yz' + x'y'z' \\ xy &= xy(z + z') = xyz + xyz' \\ x'y' &= x'y'(z + z') = x'y'z + x'y'z' \\ yz' &= yz'(x + x') = xyz' + x'yz' \end{aligned}$$

Step 2. The summands of $x'z'$ are $x'yz$ and $x'y'z'$ which appear among the other summands. Thus delete $x'z'$ to obtain

$$E = xy + x'y' + yz'$$

作业

- 1，证明布尔代数是有界有补分配格，有界有补分配格是布尔代数
- 2，证明定理15.6
- 3，证明等势的布尔代数均同构