ECE 510 OCE
BDDs and Their Applications

Lecture 17.
Implicit Multi-Output Decomposition

May 23, 2000
Alan Mishchenko

Overview

- Implicit representation of partitions
  - equivalence relations as char functions of partitions
  - operations on partitions and their implementation
- Approaches to functional decomposition
  - Implicit, multi-output decomposition (Wurth/Eckl/Legl,DAC'95)

Types of Decomposition

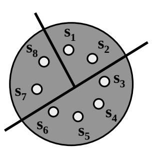
- Simple (Ashenhurst) - intermediary signal is binary / "Complex" (Curtis) - intermediary signal is multi-valued
- Completely specified / Incompletely specified functions
- Single output / Multi-output functions
- Binary / Multi-valued functions
- Multi-valued functions / relations
- Specialized types
  - Bi-decomposition
  - Decomposition with feedback (combinational loops)

Partitions

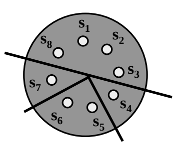
- Partition π on a set of elements S is a collection of disjoint subsets of S, whose set union is S
- It is written: $\pi = \{B_a\}$ such that
  - $B_a \cap B_b = \emptyset, \ \forall\, a \neq b$
  - $\cup\{B_a\} = S$
- Often used are partitions of states of FSM $M=\{S,I,O,\delta,\lambda\}$ satisfying certain properties, for example, substitution property:

$$\forall s,t \in S: s \equiv t(\pi) \Rightarrow \forall a \in I\ \ \delta(s,a) \equiv \delta(t,a)(\pi)$$

Examples

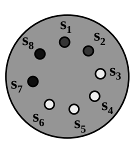


$\pi_1 = \{ \{s_1,s_2\},\{s_3,s_4,s_5,s_6\}, \{s_7,s_8\} \}$

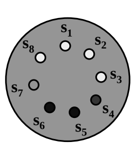


$\pi_2=\{ \{s_1,s_2,s_3,s_8\},\{s_4\},\{s_5,s_6\},\{s_7\} \}$

Examples



$\pi_1 = \{ \{s_1,s_2\},\{s_3,s_4,s_5,s_6\}, \{s_7,s_8\} \}$



$\pi_2=\{ \{s_1,s_2,s_3,s_8\},\{s_4\},\{s_5,s_6\},\{s_7\} \}$

Equivalence Relation

- Equivalence relation is a boolean function $E(x_1, x_2)$ such that $E(x_1, x_2) = 1$, iff x_1 and x_2 are equivalent.
- Equivalence relation is
  - reflexive: $\forall x \; E(x,x) = 1$
  - symmetric: $\forall x,y \; E(x,y) = E(y,x)$
  - transitive: $\forall x,y,z \; [E(x,y) \,\&\, E(y,z)] \Rightarrow E(x,z)$

Equivalence Relations and Partitions

- There is a one-to-one correspondence between equivalence relations and partitions
- Each equivalence relation corresponds to a partition of elements, such that each block of the partition consists of objects equivalent with respect to the given equivalence relation, and vice versa
- Therefore, BDD representing the equivalence relation can be considered a characteristic function of the partition

Characteristic Functions

- The characteristic function of the set is a function which is 1 for those minterms that are used to encode the elements of the set
- The characteristic function of the set depends on one range of BDD variables (WHO variables)
- The characteristic function of the relation is a function which is 1 for those minterms that are used to encode the pairs of related elements of the set
- The characteristic function of the relation depends on two ranges of variables (WHO variables and WITH WHOM variables)

Example

- Problem: Given the set $\{p_0, p_1, p_2, p_3, p_4, p_5\}$ and its encoding:

  $p_0 - \bar{x}_2\bar{x}_1\bar{x}_0$ $p_2 - \bar{x}_2 x_1\bar{x}_0$ $p_4 - x_2\bar{x}_1\bar{x}_0$

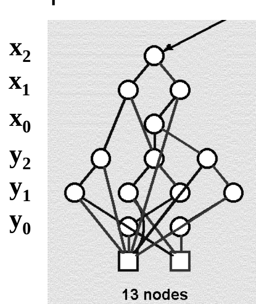
  $p_1 - \bar{x}_2\bar{x}_1 x_0$ $p_3 - \bar{x}_2 x_1 x_0$ $p_5 - x_2\bar{x}_1 x_0$

  find characteristic function of partition $\pi = \{ \{p_0,p_1\}, \{p_2,p_3,p_4\}, \{p_5\} \}$ and represent it using BDDs
- Solution: Define an equivalence relation $E(x_0,x_1,x_2, y_0,y_1,y_2)$ such that it is equal to 1 only for those pairs of minterms that correspond to codes of elements belonging to the same block.

Example (continued)

X\Y	000	001	010	011	100	101	110	111
000	**1**	**1**	0	0	0	0	0	0
001	**1**	**1**	0	0	0	0	0	0
010	0	0	**1**	**1**	**1**	0	0	0
011	0	0	**1**	**1**	**1**	0	0	0
100	0	0	**1**	**1**	**1**	0	0	0
101	0	0	0	0	0	**1**	0	0
110	0	0	0	0	0	0	0	0
111	0	0	0	0	0	0	0	0

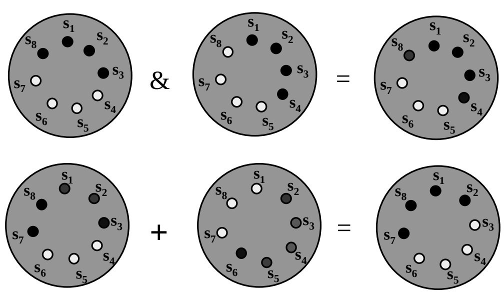
Implicit Representation of Partition π



Operations on Partitions

- The product of partitions π_1 and π_2 on S is the partition $\pi_1 \cdot \pi_2$ on S such that $s \equiv t(\pi_1 \cdot \pi_2)$ iff $s \equiv t(\pi_1)$ and $s \equiv t(\pi_2)$.
- The sum of partitions π_1 and π_2 on S is the partition $\pi_1+\pi_2$ on S such that $s \equiv t(\pi_1+\pi_2)$ iff there is a sequence in S, $s = s_0, s_1, \ldots, s_n$, such that $s_n = t$ and either $s_i \equiv s_{i+1}(\pi_1)$ or $s_i \equiv t_{i+1}(\pi_2)$, $0 \leq i \leq n-1$.

Interpretation of Operations



Implicit Computation of Product/Sum

- Similar to how sets are manipulated using their characteristic functions, partitions can be manipulated using their characteristic functions
- The product of partitions is the product of their char functions
- The sum of partitions is the sum of their char functions followed by computation of transitive closure

Example: Product of Partitions

	00	01	11	10
00	1	0	0	0
01	0	1	1	1
11	0	1	1	1
10	0	1	1	1

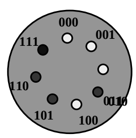
&

	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	0	0	1	1
10	0	0	1	1

=

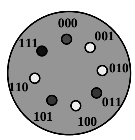
	00	01	11	10
00	1	0	0	0
01	0	1	0	0
11	0	0	1	1
10	0	0	1	1

	000	001	010	011	100	101	110	111
00	0	0	0	1	0	1	1	1
01	1	1	1	1	1	1	1	0
10	1	1	1	1	1	1	1	0
11	0	0	0	1	0	1	1	0

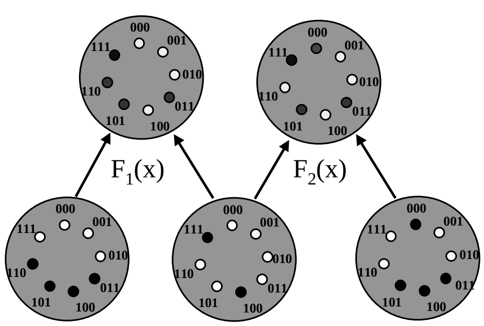


$F_1(x)$

	000	001	010	011	100	101	110	111
00	0	0	0	1	0	1	0	1
01	0	1	1	1	1	1	1	0
10	0	1	1	1	1	1	1	0
11	1	1	1	0	1	0	1	0



$F_2(x)$



$D_2(x)$ $D_1(x)$ $D_3(x)$

Decomposition Condition

- Decomposition exists iff the product of all decomposition functions $D_i(x)$ refines the product of all output functions $F_k(x)$

$$\pi_{Di(x)} \leq \pi_{Fk(x)}$$

Decomposition Algorithm

- Find the global partition of all outputs
- Find the characteristic function of all preferable (constructable and assignable) decomposition functions
- Use Lmax algorithm to select the best preferable function (function which can be used to decompose the most outputs)
- Iterate the above steps until the decomposition is selected