Solution to Exercises - Week 8
Intelligent Systems Programming

Exercise 1

The recursive call structure of constructing the ROBBD of $\neg x_1 \wedge (x_2 \Leftrightarrow \neg x_3)$ using the ordering $x_1 < x_2 < x_3$ is shown in Figure 1 and the resulting ROBBD is shown in Figure 2.

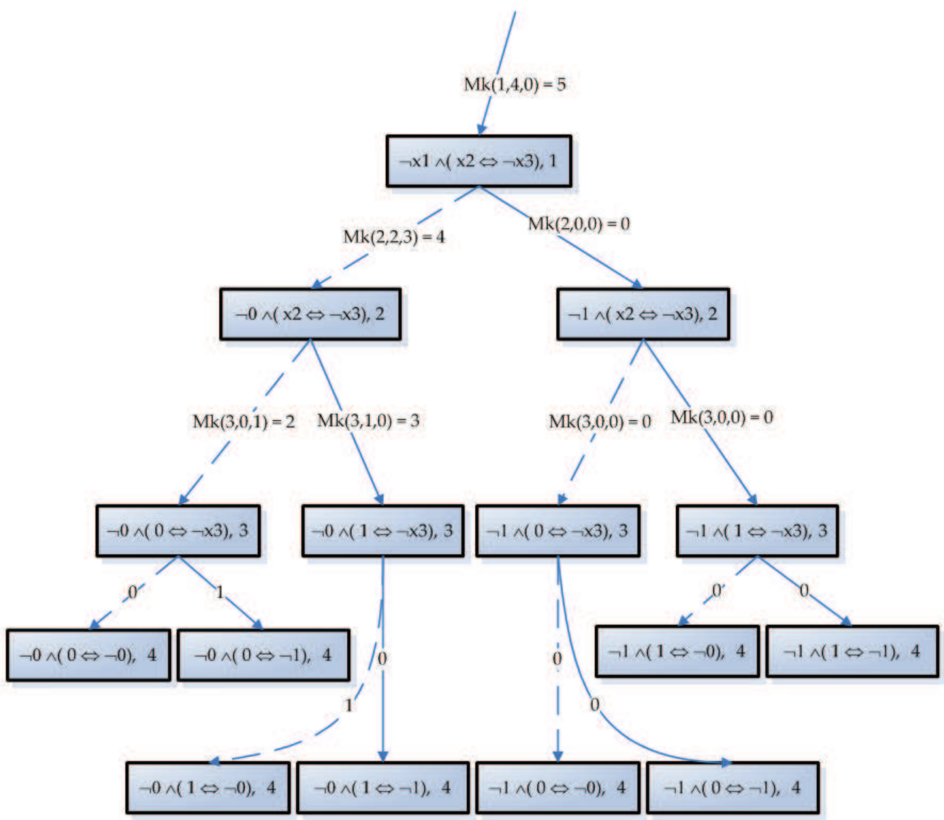

Figure 1: The recursive call structure of BUILD for $\neg x_1 \wedge (x_2 \Leftrightarrow x_3)$

The final content of the unique table is given in Figure 3.

Exercise 2

When we are looking at an expression t which is a conjunctions of variables and negated variables, we know that if the evaluation of the variable we are branching

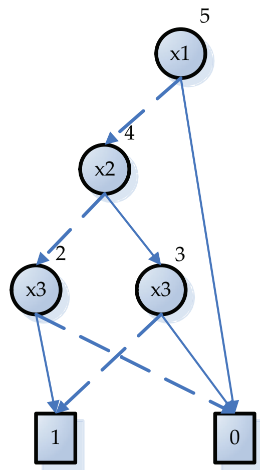

Figure 2: The ROBDD for $\neg x_1 \wedge (x_2 \Leftrightarrow x_3)$

ID	variable	low	high
0	4	-	-
1	4	-	-
2	3	0	1
3	3	1	0
4	2	2	3
5	1	4	0

Figure 3: The table for $\neg x_1 \wedge (x_2 \Leftrightarrow x_3)$

on results in false, then the whole expression will evaluate to false. To improve the BUILD we can therefore do the following:

BUILDCONJ(t)
  return BUILD'CONJ(t, 1)

where BUILD'CONJ(t, i) is described in the following pseudo code:

BUILD'CONJ(t, i)
  if $i > n$
    if t is false
      return 0
    else
      return 1
  else
    if x_i is unnegated in t
      $v_0 \leftarrow 0$
      $v_1 \leftarrow$ BUILD'CONJ$(t[1/x_i], i+1)$
    else
      $v_0 \leftarrow$ BUILD'CONJ$(t[0/x_i], i+1)$
      $v_1 \leftarrow 0$
    return MK(i, v_0, v_1)

Exercise 3

The ROBBDs of x and $x \Rightarrow y$ is shown in Figure 4 and the table of nodes after these have been build is shown in Figure 5.

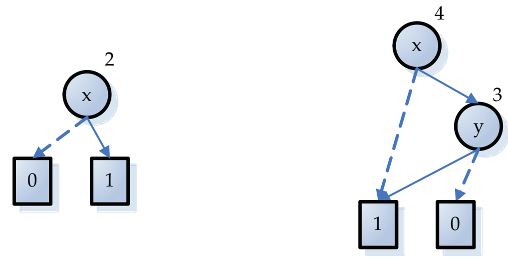

Figure 4: The ROBDDs for x and $x \Rightarrow y$

ID	variable	low	high
0	3	-	-
1	3	-	-
2	1	0	1
3	2	0	1
4	1	1	3

Figure 5: The table after building ROBBDs for x and $x \Rightarrow y$

Running the APPLY results in the structure in Figure 6. The table after running

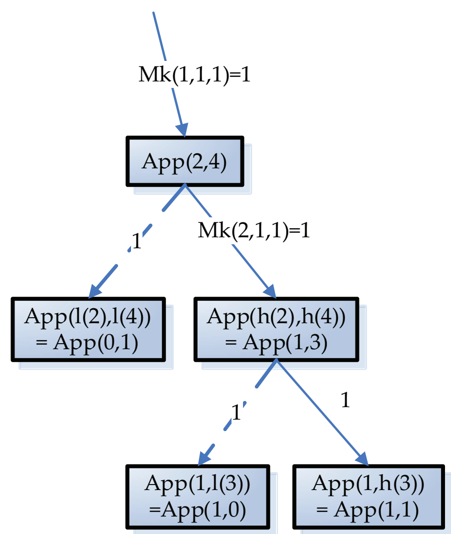

Figure 6: The recursive calls of Apply

the algorithm is still the one shown in Figure 5. The resulting ROBBD is shown in Figure 7; it is easily seen that

$$x \vee (x \Rightarrow y) \equiv x \vee \neg x \vee y \equiv \top.$$

Figure 7: Resulting ROBBD from $x \vee (x \Rightarrow y)$

Exercise 4

The recursive call structure of constructing the ROBBDs of $\neg(x_1 \wedge x_3)$ and $x_2 \wedge x_3$ using the ordering $x_1 < x_2 < x_3$ is shown in Figure 10 and the resulting ROBBDs is shown in Figure 8.

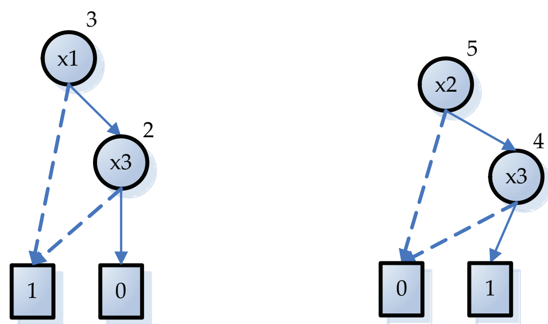

Figure 8: The ROBDDs for $\neg(x_1 \wedge x_3)$ and $x_2 \wedge x_3$

ID	variable	low	high
0	4	-	-
1	4	-	-
2	3	1	0
3	1	1	2
4	3	0	1
5	2	0	4

Figure 9: The table after construction of ROBBDs for $\neg(x_1 \wedge x_3)$ and $x_2 \wedge x_3$

The content of the unique table after constructing these ROBBDs is given in Figure 9.

Using APPLY to construct the ROBBD for $\neg(x_1 \wedge x_3) \vee (x_2 \wedge x_3)$ results in the recursive call structure given in Figure 11. The resulting ROBBD is shown in Figure 12.

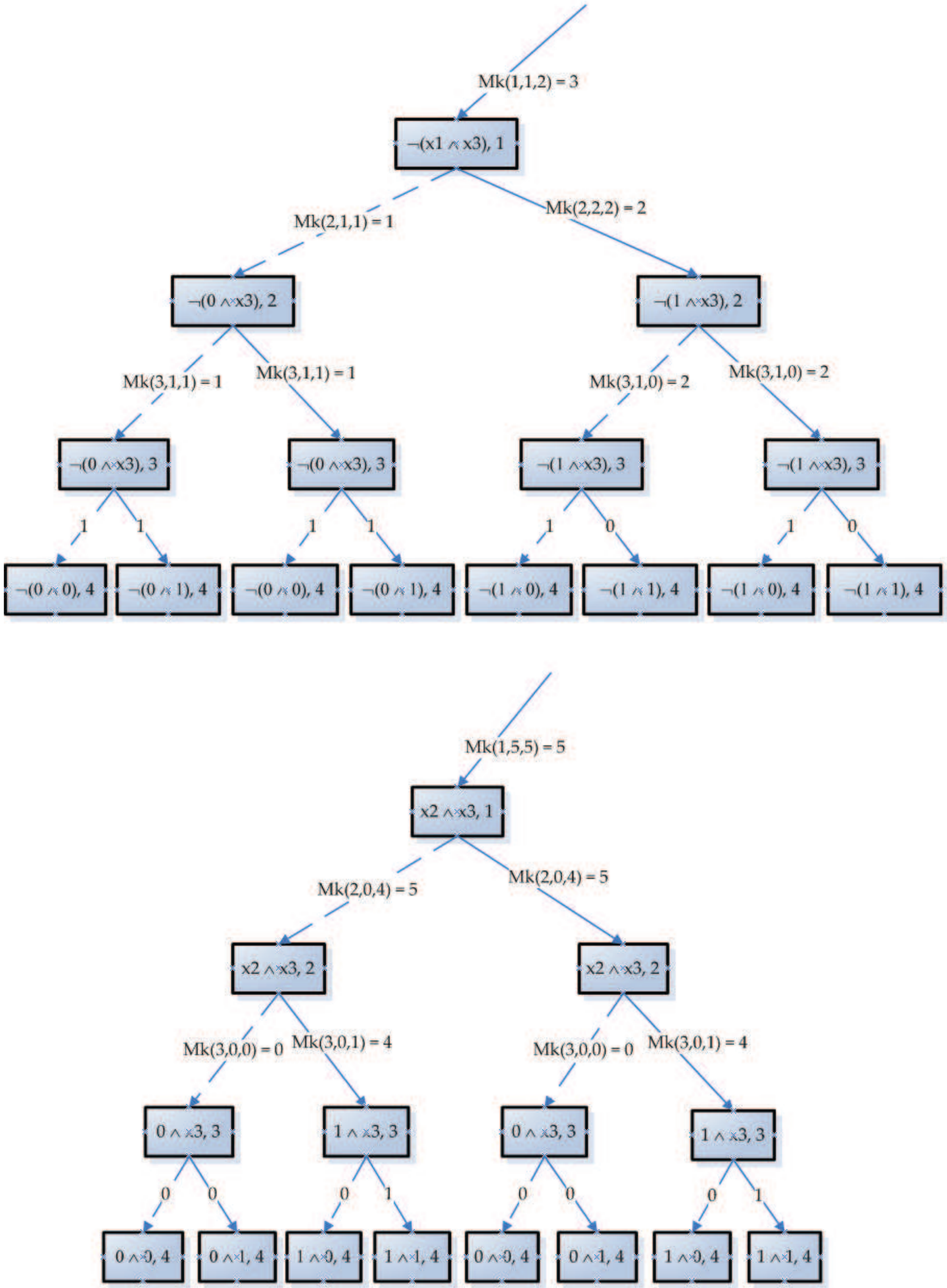

Figure 10: The recursive call structure of BUILD for $\neg(x_1 \wedge x_3)$ and $x_2 \wedge x_3$

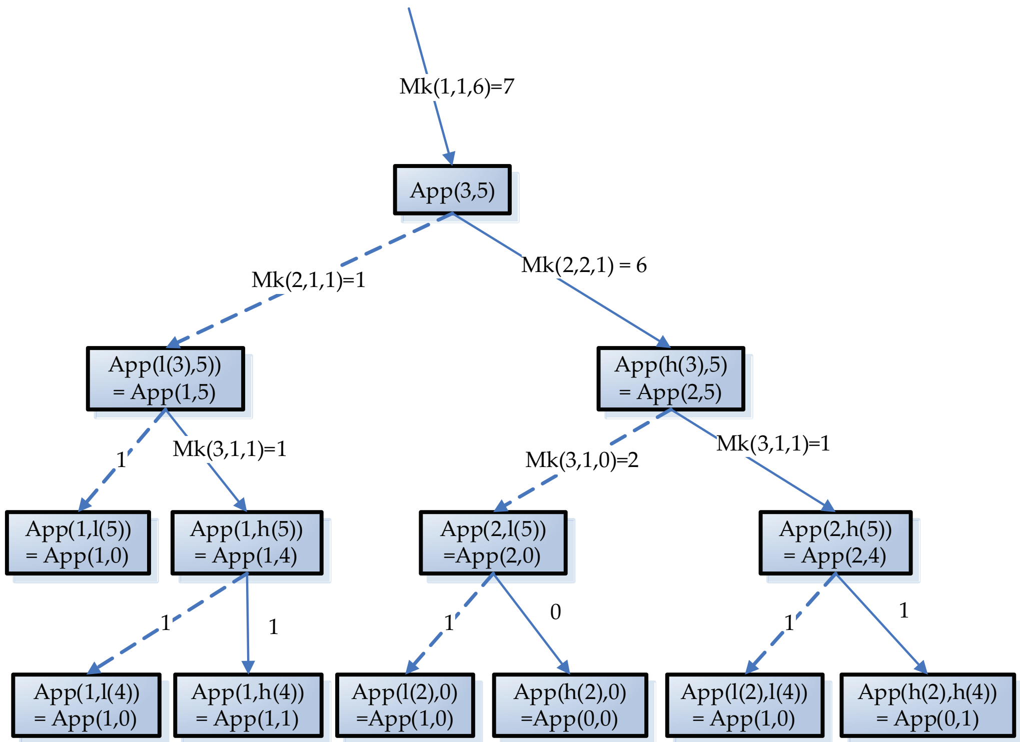

Figure 11: The recursive call structure of APPLY for $\neg(x_1 \wedge x_3) \vee (x_2 \wedge x_3)$

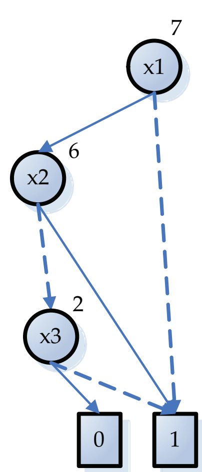

Figure 12: The ROBDD for $\neg(x_1 \wedge x_3) \vee (x_2 \wedge x_3)$