Questionnaire “Logic and Computability”

Summer Term 2022

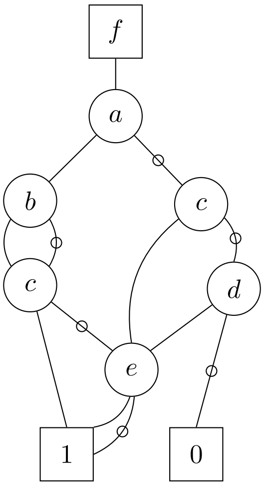
Contents

6 Binary Decision Diagrams

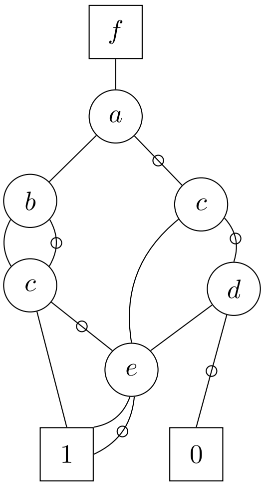
6.1 Lecture

6.1.1 Binary Decision Diagram

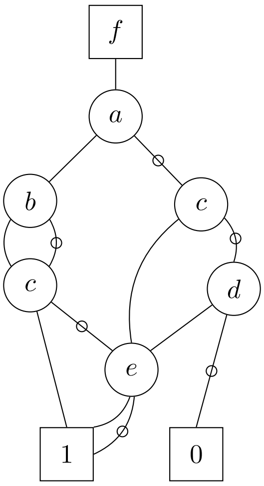
1. [Lecture] Given the *Binary Decision Diagram (BDD)* below, label and explain the different elements of the diagram.



2. [Lecture] Given the *Binary Decision Diagram (BDD)* below. (a) Find a satisfying model $\mathcal{M}_1$, i.e., $\mathcal{M}_1 \models f$. (b) Find a falsifying model $\mathcal{M}_2$, i.e., $\mathcal{M}_2 \not\models f$.



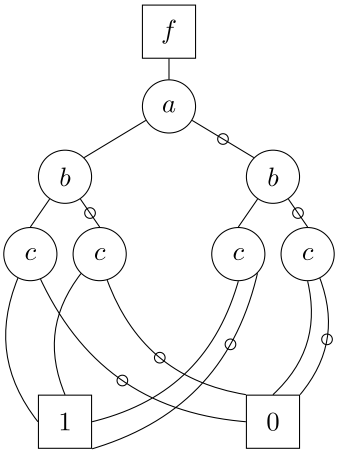
3. [Lecture] Given the *Binary Decision Diagram (BDD)* below. Construct the formula f in disjunctive normal form (DNF) that is represented by the BDD.



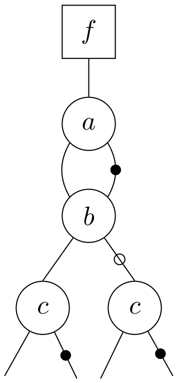
6.1.2 Reduced Ordered BDDs

In the following examples, we have the following convention: Else-edges are marked with circles. Filled circles represent the *complemented* attribute. Dangling edges are assumed to point to the constant node `true`.

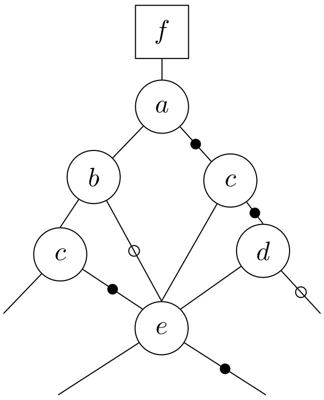
4. [Lecture] Transform the given Binary Decision Diagram (BDD) into a reduced ordered BDD (ROBDD) using the variable order $a < b < c$.



5. [Lecture] Transform the given Binary Decision Diagram (BDD) into a reduced ordered BDD (ROBDD) using the variable order $a < b < c$.



6. [Lecture] A *Reduced and Ordered Binary Decision Diagram (ROBDD)* is a canonical representation of a Boolean formula. Explain what this means and why this is the case.
7. [Lecture] Given the *Binary Decision Diagram (BDD)* below. Construct the formula f in disjunctive normal form (DNF) that is represented by the BDD.



6.1.3 Construction of Reduced Ordered BDDs

8. [Lecture] Construct a ROBDD for the formula

$$f = (a \wedge b \vee \neg a) \wedge (\neg c \wedge d) \vee c,$$

using *alphabetic variable order*. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

9. [Lecture] Construct a ROBDD for the formula

$$f = (r \wedge p) \vee (\neg r \wedge \neg p) \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee (\neg r \wedge q),$$

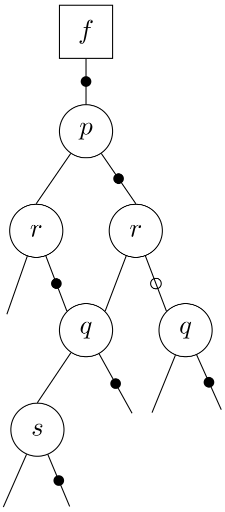
using *variable order* $p < q < r < s$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

10. [Lecture] Construct a ROBDD for the formula

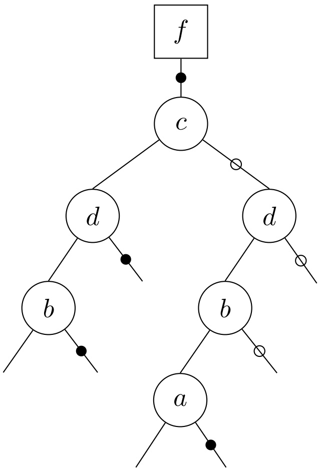
$$f = (r \wedge \neg p) \vee (\neg r \wedge p) \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee (r \wedge q),$$

using *variable order* $q < s < r < p$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

6.2 Practicals

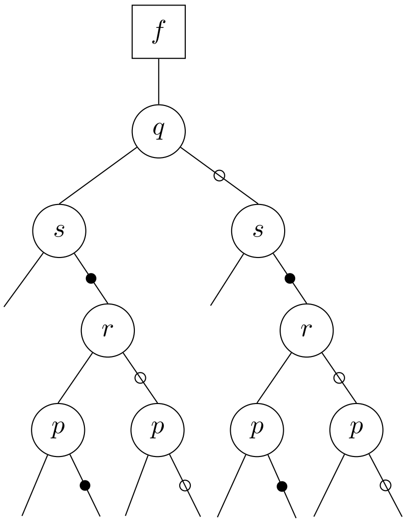


1. [Practicals] **[2 Points]**
   - (a) Use the BDD shown in the figure on the right to check if the formula it represents evaluates to `true` or `false` with the following variable assignments.
     - i. $\mathcal{M}_1 : \ p = \top, r = \bot, q = \top, s = \bot$
     - ii. $\mathcal{M}_2 : \ p = \bot, r = \bot, q = \bot, s = \top$
   - (b) Find the formula f that is represented by the BDD.



2. [Practicals] **[2 Points]**
   - (a) Use the BDD shown in the figure on the right to check if the formula it represents evaluates to `true` or `false` with the following variable assignments.
     - i. $\mathcal{M}_1 : \ a = \bot, b = \top, c = \bot, d = \top$
     - ii. $\mathcal{M}_2 : \ a = \top, b = \top, c = \top, d = \top$
   - (b) Find the formula f that is represented by the BDD.

3. [Practicals] **[2 Points]** Convert the following BDD into a *reduced ordered* BDD.



4. [Practicals] **[3 Points]** Construct a ROBDD for the formula

$$f = (a \land d \land c) \lor (b \land \neg d \land \neg a) \lor (c \rightarrow \neg d) \lor (a \rightarrow \neg b)$$

using *variable order* $b < a < d < c$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

5. [Practicals] **[3.5 Points]** Construct a reduced ordered binary decision diagram (ROBDD) for the formula

$$f = (p \oplus q) \land \neg r$$

using *variable order* $p < q < r$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

6. [Practicals] **[3.5 Points]** Construct a ROBDD for the formula

$$f = (p \leftrightarrow q) \land (r \leftrightarrow s)$$

using *variable order* $r < s < p < q$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

7. [Practicals] **[4 Points]**

   (a) Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for the formula

$$f = (a \lor b \lor c) \land \neg d$$

   using *variable order* $c < a < d < b$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

(b) Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for f with a different variable order. The ROBDD should result in a *smaller* ROBDD, w.r.t. the number of nodes.

6.3 Self-Assessment

6.3.1 Binary Decision Diagram

11. [Self-Assessment]

    Give the definition of a *Directed Acrylic Graph (DAG)*. Is the relation between a Binary Decision Diagrams (BDDs) and DAGs?

12. [Self-Assessment]

    - Give the definition of a Binary Decision Diagram (BDD).
    - Draw an example and label and explain the different elements of the diagram.
    - Explain the underlying structure of a BDD and explain, why BDDs are not trees.

6.3.2 Reduced Ordered BDDs

13. [Self-Assessment] In the following list tick all items which can be part of a *Reduced Ordered Binary Decision Diagram (ROBDD)*.

    ☐ Function nodes
    ☐ (Complemented) edges
    ☐ Self-Loops
    ☐ Constant ”true”-node
    ☐ Variable pointers

14. [Self-Assessment] Assume that you have already constructed a *Reduced Ordered Binary Decision Diagram (ROBDD)* for a given formula and variable order. What can happen, if you change the variable order and you draw the ROBDD for the same formula with the new order again?

15. [Self-Assessment] What is the worst-case size of a *Reduced Ordered Binary Decision Diagrams (ROBDDs)* with respect to the formula that it represents. What is the advantage of using a ROBDD to represent a formula compared to using a truth table?

16. [Self-Assessment] How many nodes does a *Reduced Ordered Binary Decision Diagrams (ROBDDs)* for a Boolean formula with n variables have, in worst-case?

    ☐ $2n$
    ☐ $\mathcal{O}(n^2)$
    ☐ $\mathcal{O}(2^n)$
    ☐ $2^{n+1}-1$
    ☐ n^2
    ☐ infinitely many

17. [Self-Assessment] For each of the following statements, state whether it constitutes an advantage of *Reduced Ordered Binary Decision Diagrams (ROBDDs)* as a data structure, a disadvantage, or neither. Mark **a**dvantages with "A", **d**isadvantages with "D", and items which are **n**either with "N". (Note: Consequently, items which are factually wrong, or have nothing to do with BDDs, should be marked with "N"!)

    ☐ Some formulas that can be expressed by truth tables cannot be expressed by ROBDDs.

    ☐ The size of a ROBDD may depend significantly on the variable order, which is hard to optimize.

    ☐ Equivalence checks can be performed in constant time (assuming that the ROBDDs for the formulas to check are already available).

    ☐ Logic operations, such as conjunction or disjunction, can be performed in polynomial time.

    ☐ Using complemented edges, negation can be performed in constant time.

18. [Self-Assessment] Tick all properties that apply to a *Reduced Ordered Binary Decision Diagram (ROBDD)*.

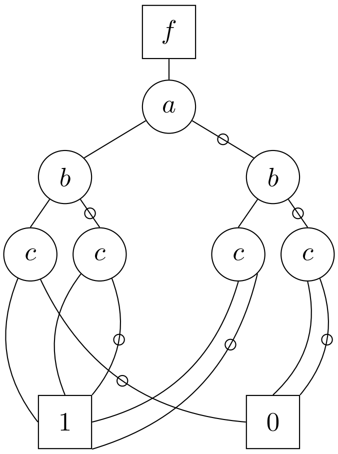
    - ☐ A ROBDD is a canonical representation of its respective formula, for any fixed variable order.
    - ☐ Since it is reduced, the number of nodes in the ROBDD does not exceed $2n^2$, where n is the number of variables.
    - ☐ The graph of an ROBDD may contain cycles.
    - ☐ A ROBDD represents a Boolean formula as directed acyclic graph (DAG).
    - ☐ Every node with two regular outgoing edges has two distinct child nodes.
    - ☐ No two nodes in an ROBDD represent the same formula.

19. [Self-Assessment] Using BDDs, how can you perform a negation of a formula in constant time?

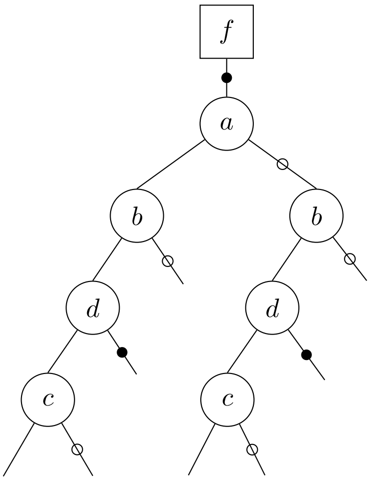
20. [Self-Assessment] Given a *Reduced and Ordered Binary Decision Diagram (ROBDD)*. Explain how you can find the propositional logic formula f that is represented by a given ROBDD?

21. [Self-Assessment] In the context of *Binary Decision Diagrams (BDDs)*, what are redundant nodes? Explain them in a few words and give an example of such a redundancy.

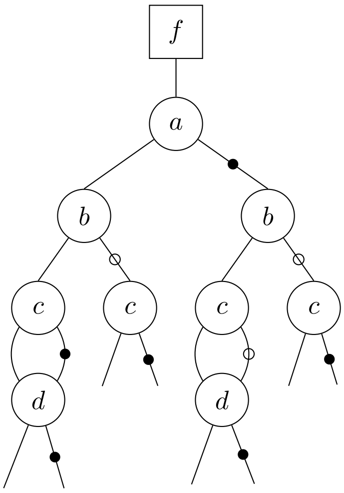
22. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below. Transform the BDD into a *Reduced Ordered Binary Decision Diagram (ROBDD)*.



23. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below. Transform the BDD into a *Reduced Ordered Binary Decision Diagram (ROBDD)*.



24. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below. Transform the BDD into a *Reduced Ordered Binary Decision Diagram (ROBDD)*.



25. [Self-Assessment] In the context of *Binary Decision Diagrams (BDDs)*, how does the variable order impact the BDD?

26. [Self-Assessment] Tick all properties that apply to a *reduced* and *ordered* BDD (ROBDD).

    - ☐ If the *else*-edge of a node is complemented, it may point to the same child node as the *then*-edge.
    - ☐ Using ROBDD, it is possible to check for validity in constant time.
    - ☐ The size of a BDD is independent on the variable order.
    - ☐ Logic operations, such as conjunction or disjunction, can be performed in polynomial time.
    - ☐ The function of the *then*-edge and the *else*-edge of a terminal node is always true.

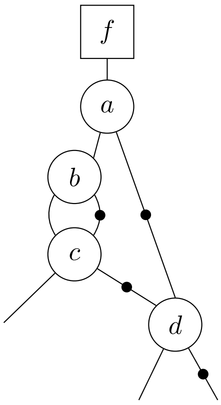
27. [Self-Assessment] Tick all properties that apply to a *reduced* and *ordered* BDD (ROBDD).

    - ☐ Checks for entailment can be done in constant time.
    - ☐ Using complemented edges, negation can be performed in constant time.
    - ☐ Some formulas that can be expressed by truth tables cannot be expressed by BDDs.
    - ☐ Equivalence checks can be performed in constant time (assuming that the BDDs for the formula to check are already available).
    - ☐ The size of a BDD may depend significantly on the variable order, which is hard to optimize.

28. [Self-Assessment] Consider a *Reduced and Ordered Binary Decision Diagram*. Explain the meaning of the terms *reduced* and *ordered* in this context. Moreover, for each of these terms, draw an example of a Binary Decision Diagram that **does <u>not</u> have** the respective property.

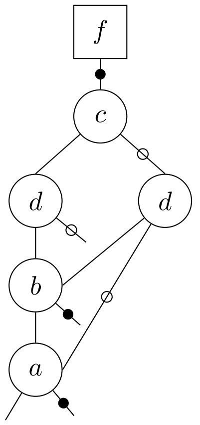
29. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below. State the formula f that is represented by the BDD.

    Note: Else-edges are marked with circles. Filled circles represent the *complemented* attribute. Dangling edges are assumed to point to the constant node `true`.



30. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below. State the formula *f* that is represented by the BDD.

    Note: Else-edges are marked with circles. Filled circles represent the *complemented* attribute. Dangling edges are assumed to point to the constant node `true`.



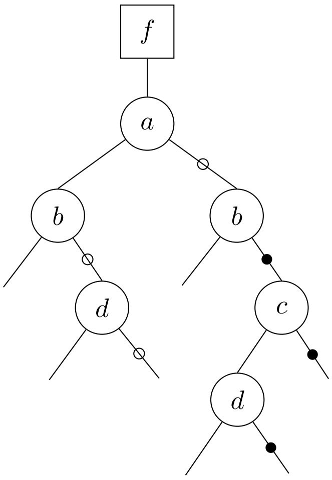
31. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below, ...

    (a) ... check if the following variable assignments evaluate to `true` or to `false`.

    i. $a = \top, b = \bot, c = \bot, d = \top$
    ii. $a = \bot, b = \bot, c = \top, d = \top$

    (b) ... find a propositional formula f, that is represented by the BDD.

    Note: Else-edges are marked with circles. Filled circles represent the *complemented* attribute. Dangling edges are assumed to point to the constant node `true`.



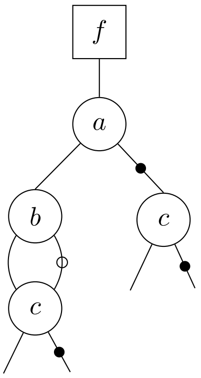
32. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below, ...

    (a) ... check if the following variable assignments evaluate to `true` or to `false`.

    i. $a = \top, b = \top, c = \bot$
    ii. $a = \bot, b = \bot, c = \bot$

    (b) ... find a propositional formula f, that is represented by the BDD.

    Note: Else-edges are marked with circles. Filled circles represent the *complemented* attribute. Dangling edges are assumed to point to the constant node `true`.



33. [Self-Assessment] Given the *Binary Decision Diagram (BDD)* below, ...

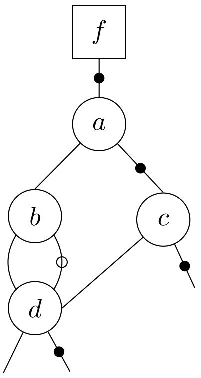
    (a) ... check if the following variable assignments evaluate to `true` or to `false`.

       i. $a = \top, b = \top, c = \bot, d = \bot$

       ii. $a = \bot, b = \bot, c = \top, d = \top$

    (b) ... find a propositional formula f, that is represented by the BDD.

    Note: Else-edges are marked with circles. Filled circles represent the *complemented* attribute. Dangling edges are assumed to point to the constant node `true`.



6.3.3 Construction of Reduced Ordered BDDs

34. [Self-Assessment] What is a cofactor of a formula? Given an example of a propositional logic formula and compute the positive and the negative cofactor for one variable of this formula.

35. [Self-Assessment] Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for the formula

$$f = (\neg a \lor b) \land (a \lor b),$$

    using *alphabetic variable order*. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

36. [Self-Assessment] Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for the formula

$$f = (\neg x \lor \neg y) \land (x \land (y \lor z)),$$

using *variable order* $y < z < x$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

37. [Self-Assessment] Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for the formula

$$f = (\neg x \wedge \neg y) \vee (x \wedge y),$$

using *variable order* $z < x < y$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

38. [Self-Assessment] Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for the formula

$$f = (\neg p \vee r) \wedge (q \vee \neg p) \wedge (\neg q \vee p)$$

using *variable order* $r < q < p$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

39. [Self-Assessment] Construct a Reduced Ordered Binary Decision Diagram (ROBDD) for the formula

$$f = (q \wedge \neg s) \vee (s \wedge (\neg r \vee p)) \vee (p \wedge q \wedge r)$$

using *variable order* $p < q < r < s$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.