Questionnaire "Logic and Computability"

Summer Term 2024

Contents

3 Binary Decision Diagrams **1**

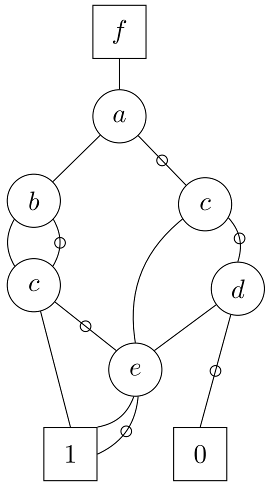
3.1 Reduced Ordered Binary Decision Diagrams 1

3.2 Construction of Reduced Ordered BDDs 10

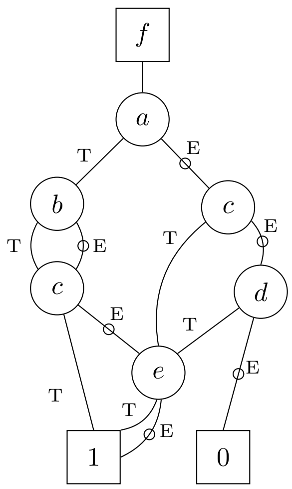
3 Binary Decision Diagrams

3.1 Reduced Ordered Binary Decision Diagrams

3.1.1 Given the *Binary Decision Diagram (BDD)* below, label and explain the different elements of the diagram.

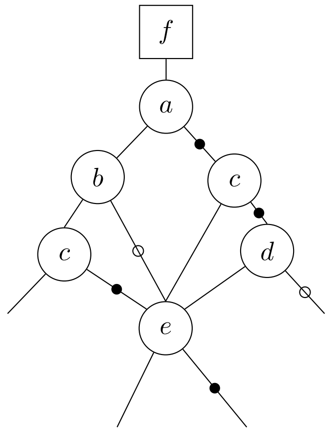


Solution



A binary decision diagram represents a Boolean formula f. It is a DAG with two terminal nodes that are labelled with 0 and 1. The internal nodes are labelled with the Boolean variables of the formula (here a, b, c, d and e). Each internal node has exactly two outgoing edges: one edge labeled with a T (the then-edge), and another edge that is labeled with an E (the else-edge) or marked with a circle. There is a unique initial node called the function node labeled with f that does not have any incoming edges and one outgoing edge to the internal variable node on the first level.

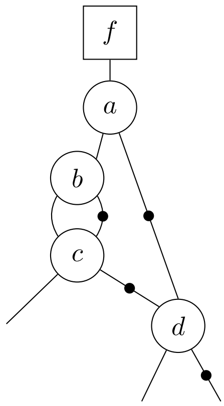
3.1.2 Given the following Binary Decision Diagram that represents the formula f. Compute its disjunctive normal form DNF(f).



Solution

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

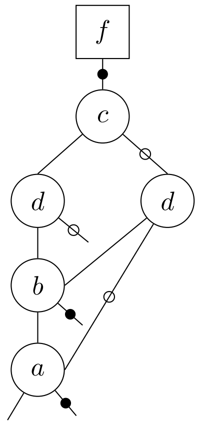
3.1.3 Given the following Binary Decision Diagram that represents the formula f. Compute its disjunctive normal form DNF(f).



Solution

There is no solution available for this question yet.

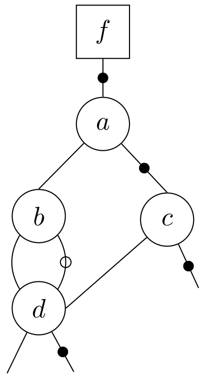
3.1.4 Given the following Binary Decision Diagram that represents the formula f. Compute its disjunctive normal form DNF(f).



Solution

There is no solution available for this question yet.

3.1.5 For the following binary decision diagram:



Check if the following models are satisfying:

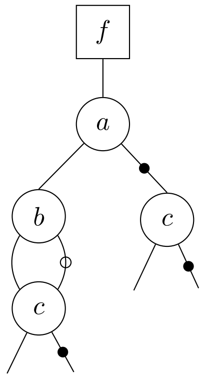
$$\mathcal{M}_1 = \{a = \top, b = \top, c = \bot, d = \bot\},$$
$$\mathcal{M}_2 = \{a = \bot, b = \bot, c = \top, d = \top\}, \text{ and}$$

compute DNF(f).

Solution

There is no solution available for this question yet.

3.1.6 For the following binary decision diagram:



Check if the following models are satisfying:

$$\mathcal{M}_1 = \{a = \top, b = \top, c = \bot\},$$
$$\mathcal{M}_2 = \{a = \bot, b = \bot, c = \bot\}, \text{ and}$$

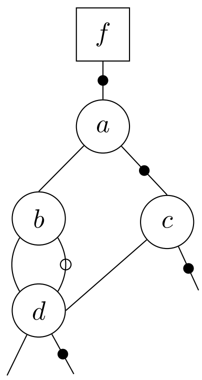
compute DNF(f).

Solution

There is no solution available for this question yet.

3.1.7 For the following binary decision diagram:

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



Check if the following models are satisfying:

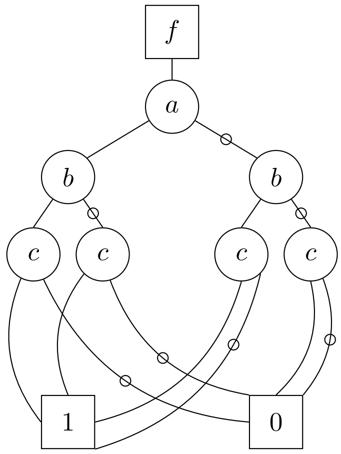
$$\mathcal{M}_1 = \{a = \top, b = \top, c = \bot, d = \bot\},$$
$$\mathcal{M}_2 = \{a = \bot, b = \bot, c = \top, d = \top\}, \text{ and}$$

compute DNF(f).

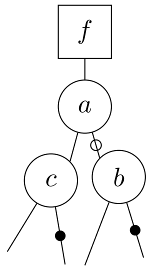
Solution

There is no solution available for this question yet.

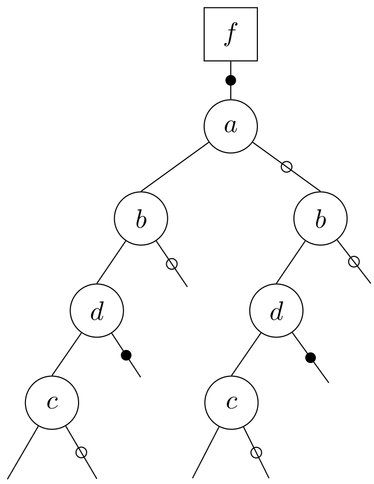
3.1.8 Transform the given Binary Decision Diagram into a reduced and ordered BDD.



Solution



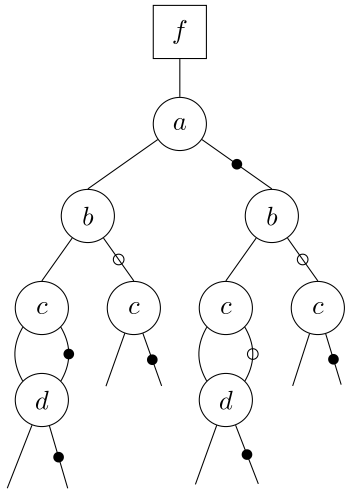
3.1.9 Transform the given Binary Decision Diagram into a reduced and ordered BDD.



Solution

There is no solution available for this question yet.

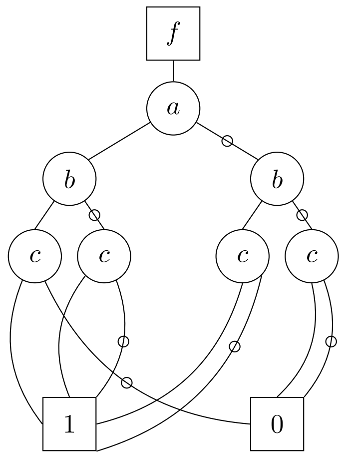
3.1.10 Transform the given Binary Decision Diagram into a reduced and ordered BDD.



Solution

There is no solution available for this question yet.

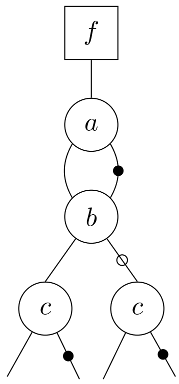
3.1.11 Transform the given Binary Decision Diagram into a reduced and ordered BDD.



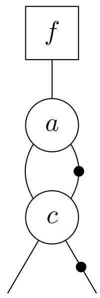
Solution

There is no solution available for this question yet.

3.1.12 Transform the given Binary Decision Diagram into a reduced and ordered BDD.



Solution



3.1.13 A reduced and ordered BDD is a canonical representation of a propositional formula. Explain the term *canonical* in the context of propositional formulas and explain why reduced and ordered BDDs give canonical representations.

Solution

A canonical representation means, that for any two semantically equivalent formulas f_1 and f_2 and a fixed variable order, f_1 and f_2 are represented by the same reduced and ordered BDD.
Since semantically equivalent formulas have the same set of satisfying models and a reduced and ordered BDD does not have redundancies, the representation of f_1 and f_2 are the same.

3.1.14 Give the definition of a cofactor of a formula f with respect to an assignment A.

Solution

There is no solution available for this question yet.

3.1.15 What is the worst-case size of a binary decision diagram? What is the advantage of computing a reduced and ordered BDD to represent a formula compared to using a truth table?

Solution

There is no solution available for this question yet.

3.1.16 Tick all properties that apply to a reduced and ordered binary decision diagram.

☐ A reduced and ordered BDD is a canonical representation of the formula it represents, for any fixed variable order.

- ☐ Since it is reduced, the number of nodes in the reduced and ordered BDD does not exceed $2n^2$, where n is the number of variables.
- ☐ The graph of an BDD may contain cycles.
- ☐ A BDD represents a propositional formula as directed acyclic graph (DAG).
- ☐ Every node with two non-complemented outgoing edges has two distinct child nodes.
- ☐ No two nodes in an reduced and ordered BDD represent the same cofactor.

3.1.17 Given you have computed the reduced and ordered BDD for a formula f. How can you compute the BDD representation for $\neg f$ in *constant* time?

Solution

There is no solution available for this question yet.

3.1.18 How can you compute the propositional formula f represented by a given BDD?

Solution

There is no solution available for this question yet.

3.1.19 Give the definition of redundant nodes in a BDD. Give an example for a BDD that contains at least one redundant node.

Solution

There is no solution available for this question yet.

3.1.20 In the context of *Binary Decision Diagrams (BDDs)*, how does the variable order impact the BDD?

Solution

There is no solution available for this question yet.

3.1.21 Tick all properties that apply to a reduced and ordered BDD.

- ☐ If the *else*-edge of a node is complemented, it may point to the same child node as the *then*-edge.
- ☐ Using the reduced and ordered BDD representation of formula f, it is possible to whether f is valid in constant time.
- ☐ The size of a BDD is independent on the variable order.
- ☐ Using complemented edges, negation can be performed in constant time.
- ☐ The size of a BDD is independent of the variable order.

3.1.22 When do we consider a BDD to be reduced? Explain the types of redudancies that are not allowed to appear in a reduced and ordered BDD.

Solution

There is no solution available for this question yet.

3.1.23 Explain how a reduced and ordered BDD can be used to determine the satisfiability of the formula f it is representing.

Solution

There is no solution available for this question yet.

3.1.24 Explain how a reduced and ordered BDD can be used to determine whether the formula f it is representing is valid.

Solution

There is no solution available for this question yet.

3.2 Construction of Reduced Ordered BDDs

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

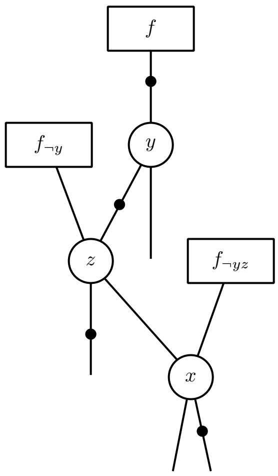
$$f = (\neg x \vee \neg y) \wedge (x \wedge (y \vee z)),$$

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.

Solution

$f = ((\neg x \vee \neg y) \wedge (x \wedge (y \vee z)))$
$f_y = \bot$
$f_{\neg y} = (x \wedge z)$
$f_{\neg y z} = x$
$f_{\neg y z x} = \top$
$f_{\neg y z \neg x} = \bot$
$f_{\neg y \neg z} = \bot$

The final ROBDD:



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

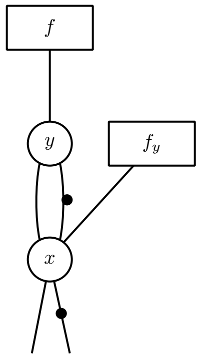
$$f = (\neg x \wedge \neg y) \vee (x \wedge y),$$

using *variable order* $y < x$. 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.

Solution

$f = ((\neg x \wedge \neg y) \vee (x \wedge y))$
$f_y = x$
$f_{yx} = \top$
$f_{y\neg x} = \bot$
$f_{\neg y} = \neg f_y$

The final ROBDD:



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

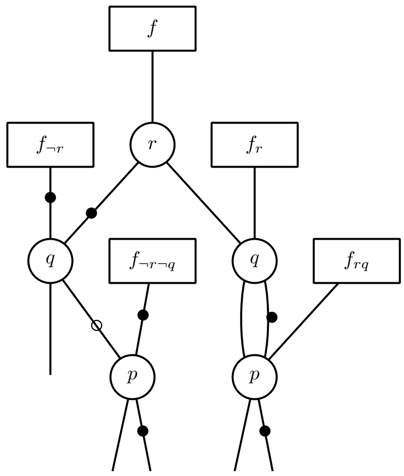
$$f = (\neg p \vee r) \wedge (q \vee \neg p) \wedge (\neg q \vee p)$$

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.

Solution

$f = (((\neg p \vee r) \wedge (q \vee \neg p)) \wedge (\neg q \vee p))$
$f_r = ((q \vee \neg p) \wedge (\neg q \vee p))$
$f_{rq} = p$
$f_{rqp} = \top$
$f_{rq\neg p} = \bot$
$f_{r\neg q} = \neg f_{rq}$
$f_{\neg r} = ((\neg p \wedge (q \vee \neg p)) \wedge (\neg q \vee p))$
$f_{\neg rq} = \bot$
$f_{\neg r\neg q} = (\neg p \wedge \neg p)$
$f_{\neg r\neg qp} = \bot$
$f_{\neg r\neg q\neg p} = \top$

The final ROBDD:



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

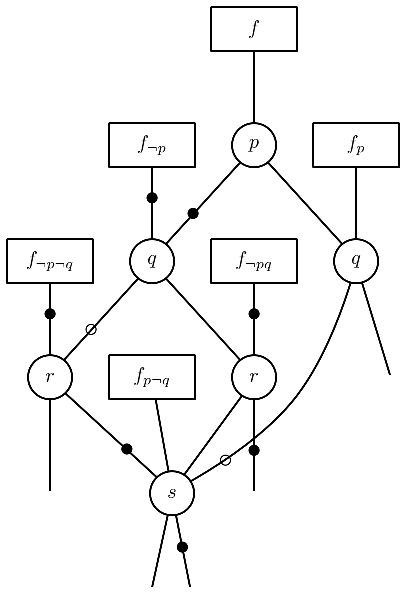
$$f = (q \wedge \neg s) \vee (s \wedge (\neg r \vee p)) \vee (p \wedge q \wedge r)$$

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.

Solution

$f = (((q \wedge \neg s) \vee (s \wedge (\neg r \vee p))) \vee ((p \wedge q) \wedge r))$
$f_p = (((q \wedge \neg s) \vee s) \vee (q \wedge r))$
$f_{pq} = \top$
$f_{p\neg q} = s$
$f_{p\neg qs} = \top$
$f_{p\neg q\neg s} = \bot$
$f_{\neg p} = ((q \wedge \neg s) \vee (s \wedge \neg r))$
$f_{\neg pq} = (\neg s \vee (s \wedge \neg r))$
$f_{\neg pqr} = \neg f_{p\neg q}$
$f_{\neg pq\neg r} = \top$
$f_{\neg p\neg q} = (s \wedge \neg r)$
$f_{\neg p\neg qr} = \bot$
$f_{\neg p\neg q\neg r} = f_{p\neg q}$

The final ROBDD:



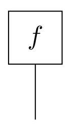
3.2.5 Construct a ROBDD for the formula

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

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.

Solution

$f = (a \wedge d \wedge c) \vee (b \wedge \neg d \wedge \neg a) \vee (c \rightarrow \neg d) \vee (a \rightarrow \neg b)$
$f_b = \top$
$f_{\neg b} = \top$



3.2.6 Construct a reduced ordered binary decision diagram (ROBDD) for the formula

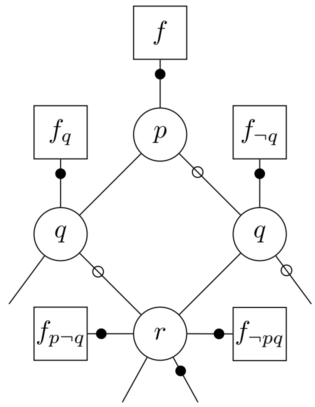
$$f = (p \oplus q) \wedge \neg r$$

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.

Solution

The final ROBDD:

$f = (p \oplus q) \wedge \neg r$
$f_p = \neg q \wedge \neg r$
$f_{pq} = \bot$
$f_{p\neg q} = \neg r$
$f_{p\neg q r} = \bot$
$f_{p\neg q \neg r} = \top$
$f_{\neg p} = q \wedge \neg r$
$f_{\neg p q} = \neg r = f_{p\neg q}$
$f_{\neg p \neg q} = \bot$



3.2.7 Construct a ROBDD for the formula

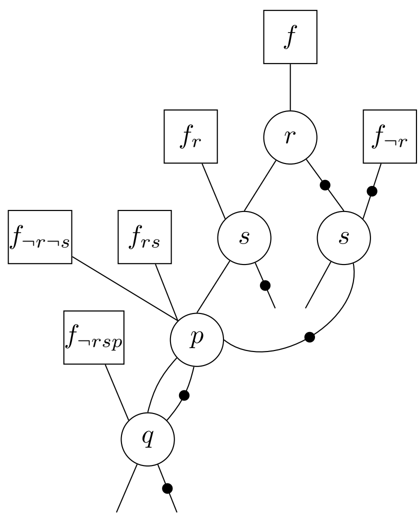
$$f = (p \leftrightarrow q) \wedge (r \leftrightarrow s)$$

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.

Solution

The final ROBDD:

$f = (p \leftrightarrow q) \wedge (r \leftrightarrow s)$
$f_r = (p \leftrightarrow q) \wedge s$
$f_{rs} = (p \leftrightarrow q)$
$f_{rsp} = q$
$f_{rspq} = \top$
$f_{rsp\neg q} = \bot$
$f_{rs\neg p} = \neg q = \neg f_{rsp}$
$f_{r\neg s} = \bot$
$f_{\neg r} = (p \leftrightarrow q) \wedge \neg s$
$f_{\neg r s} = \bot$
$f_{\neg r \neg s} = (p \leftrightarrow q) = f_{rs}$



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

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

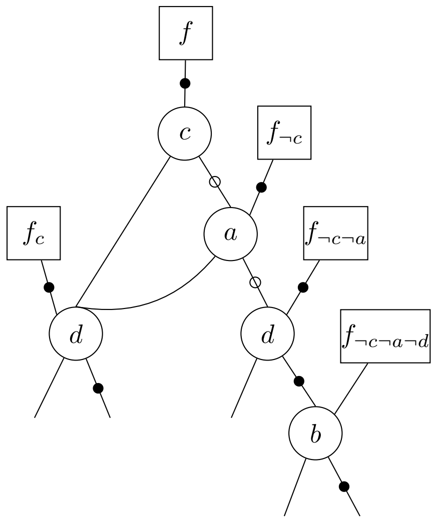
using *variable order* $a < b < c < d$. Use complemented edges and a node for `true` as the only constant node. To simplify drawing, you may assume that *dangling edges* point to the constant node. Write down all cofactors that you compute to obtain the final result and mark them in the graph.

(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.

Solution

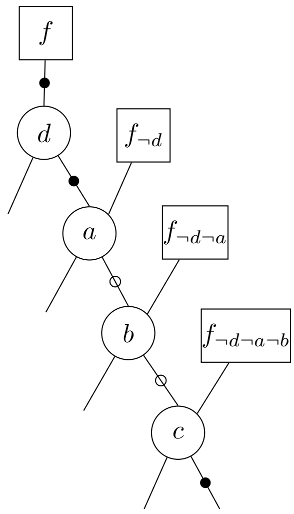
(a) using *variable order* $c < a < d < b$:

$f = (a \lor b \lor c) \land \neg d$
$f_c = \neg d$
$f_{ca} = \neg d$
$f_{c\neg a} = \neg d$
⇒a does not have an influence on the formula.
These cofactors can be skipped.
$f_{cd} = \bot$
$f_{c\neg d} = \top$
$f_{\neg c} = (a \lor b) \land \neg d$
$f_{\neg ca} = \neg d = f_c$
$f_{\neg c \neg a} = b \land \neg d$
$f_{\neg c \neg a d} = \bot$
$f_{\neg c \neg a \neg d} = b$
$f_{\neg c \neg a \neg d b} = \top$
$f_{\neg c \neg a \neg d b} = \bot$



(b) using *variable order* $d < a < b < c$:

$f = (a \lor b \lor c) \land \neg d$
$f_d = \bot$
$f_{\neg d} = a \lor b \lor c$
$f_{\neg da} = \top$
$f_{\neg d \neg a} = b \lor c$
$f_{\neg d \neg ab} = \top$
$f_{\neg d \neg a \neg b} = c$
$f_{\neg d \neg a \neg bc} = \top$
$f_{\neg d \neg a \neg b \neg c} = \bot$



3.2.9 Construct the reduced and ordered BDD for the formula

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

using *alphabetic variable order*. Compute the needed cofactors. You may add function nodes representing all cofactors to the final BDD. Use complemented edges and one terminal node representing the truth value **T**. To simplify drawing, you may assume that dangling edges point to the constant node.

Solution

$f_a = b \vee (c \leftrightarrow d)$

$f_{ab} = \top$

$f_{a\neg b} = c \leftrightarrow d$

$f_{a\neg bc} = d$

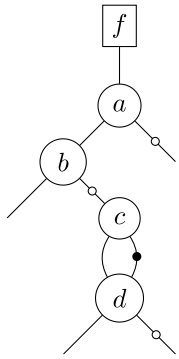
$f_{a\neg bcd} = \top$

$f_{a\neg bc\neg d} = \bot$

$f_{a\neg b\neg c} = \neg d = f_{abc}$

$f_{\neg a} = \top$

The final ROBDD:



3.2.10 Construct the reduced and ordered BDD 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* $p < q < r < s$. Compute the needed cofactors. You may add function nodes representing all cofactors to the final BDD. Use complemented edges and one terminal node representing the truth value **T**. To simplify drawing, you may assume that dangling edges point to the constant node.

Solution

$f_p = r \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee (\neg r \wedge q)$

$f_{pq} = r \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee \neg r = \top$

$f_{p\neg q} = r \vee (s \wedge \neg r) \vee (\neg s \wedge r)$

$f_{p\neg qr} = \top$

$f_{p\neg q\neg r} = s$

$f_{p\neg q\neg rs} = \top$

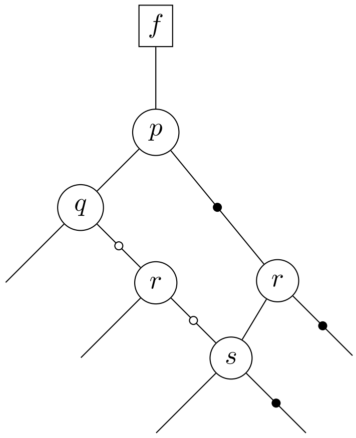
$f_{p\neg q\neg r\neg s} = \bot$

$f_{\neg p} = \neg r \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee (\neg r \wedge q) = \neg r \vee (s \wedge \neg r) \vee (\neg s \wedge r)$

$f_{\neg pr} = \neg s = f_{p\neg q\neg r}$

$f_{\neg p\neg r} = \top$

The final ROBDD:



3.2.11 Construct the reduced and ordered BDD 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* $p < q < r < s$. Compute the needed cofactors. You may add function nodes representing all cofactors to the final BDD. Use complemented edges and one terminal node representing the truth value **T**. To simplify drawing, you may assume that dangling edges point to the constant node.

Solution

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

$f_p = \neg r \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee (r \wedge q)$

$f_{pq} = \top$

$f_{p\neg q} = r \vee (s \wedge \neg r) \vee (\neg s \wedge r)$

$f_{p\neg qr} = \top$

$f_{p\neg q\neg r} = s$

$f_{p\neg q\neg rs} = \top$

$f_{p\neg q\neg r\neg s} = \bot$

$f_{\neg p} = r \vee (s \wedge \neg r) \vee (\neg s \wedge r) \vee (r \wedge q)$

$f_{\neg pq} = r \vee (s \wedge \neg r) \vee (\neg s \wedge r)$

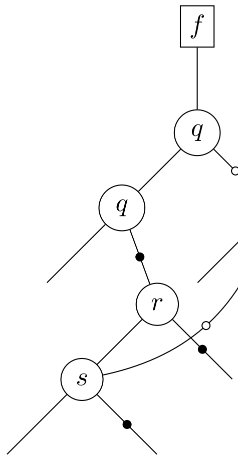
$f_{\neg p\neg q} = r \vee (s \wedge \neg r) \vee (\neg s \wedge r)$

$\Rightarrow q$ does not have an influence on the formula. These cofactors can be skipped.

$f_{\neg pr} = \top$

$f_{\neg p\neg r} = s = f_{p\neg q\neg r}$

The final ROBDD:



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

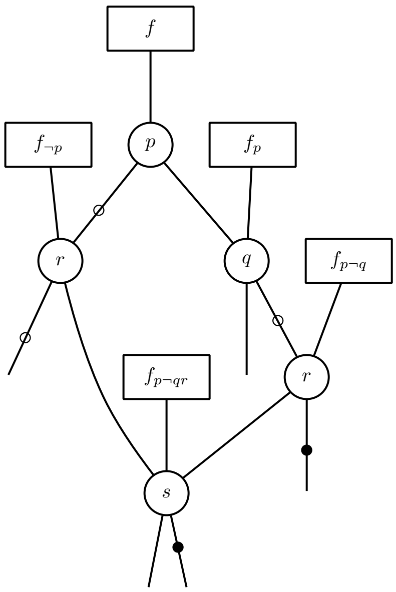
$$f = (p \wedge q) \vee (r \wedge s) \vee (\neg p \wedge \neg r)$$

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.

Solution

$f = (((p \wedge q) \vee (r \wedge s)) \vee (\neg p \wedge \neg r))$
$f_p = (q \vee (r \wedge s))$
$f_{pq} = \top$
$f_{p\neg q} = (r \wedge s)$
$f_{p\neg qr} = s$
$f_{p\neg qrs} = \top$
$f_{p\neg qr\neg s} = \bot$
$f_{p\neg q\neg r} = \bot$
$f_{\neg p} = ((r \wedge s) \vee \neg r)$
$f_{\neg pr} = f_{p\neg qr}$
$f_{\neg p\neg r} = \top$

The final ROBDD:



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

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

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.

Solution

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

$f_p = \neg s$

$f_{ps} = \bot$

$f_{p\neg s} = \top$

$f_{\neg p} = (q \wedge r)$

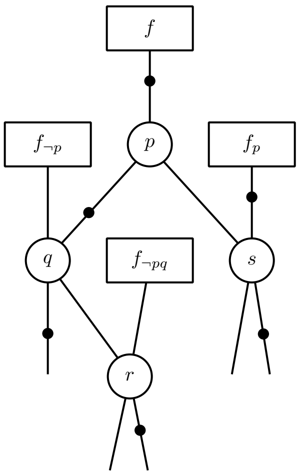
$f_{\neg pq} = r$

$f_{\neg pqr} = \top$

$f_{\neg pq\neg r} = \bot$

$f_{\neg p\neg q} = \bot$

The final ROBDD:



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

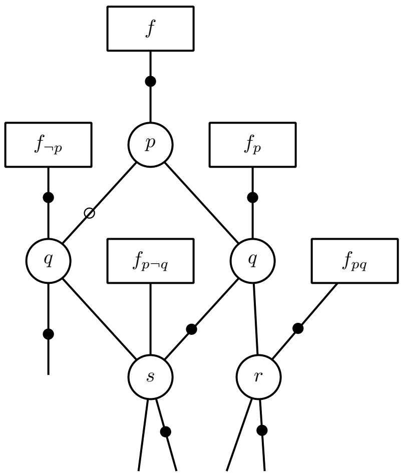
$$f = (p \wedge q \wedge \neg r) \vee (\neg q \wedge s) \vee (\neg p \wedge \neg s)$$

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.

Solution

$f = ((((p \wedge q) \wedge \neg r) \vee (\neg q \wedge s)) \vee (\neg p \wedge \neg s))$
$f_p = ((q \wedge \neg r) \vee (\neg q \wedge s))$
$f_{pq} = \neg r$
$f_{pqr} = \bot$
$f_{pq\neg r} = \top$
$f_{p\neg q} = s$
$f_{p\neg qs} = \top$
$f_{p\neg q\neg s} = \bot$
$f_{\neg p} = ((\neg q \wedge s) \vee \neg s)$
$f_{\neg pq} = \neg f_{p\neg q}$
$f_{\neg p\neg q} = \top$

The final ROBDD:



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

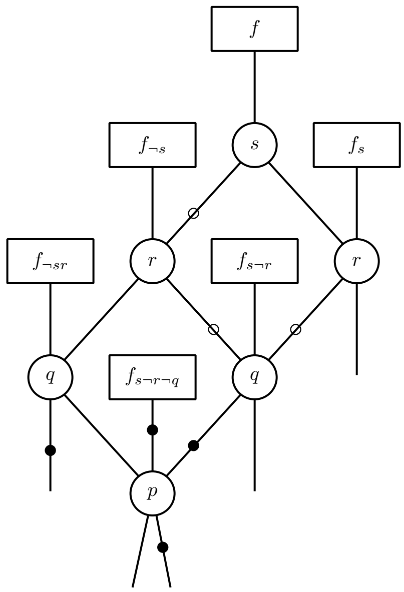
$$f = (p \wedge q) \vee (r \wedge s) \vee (\neg p \wedge \neg r)$$

using *reverse 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.

Solution

$f = (((p \wedge q) \vee (r \wedge s)) \vee (\neg p \wedge \neg r))$
$f_s = (((p \wedge q) \vee r) \vee (\neg p \wedge \neg r))$
$f_{sr} = \top$
$f_{s\neg r} = ((p \wedge q) \vee \neg p)$
$f_{s\neg rq} = \top$
$f_{s\neg r\neg q} = \neg p$
$f_{s\neg r\neg qp} = \bot$
$f_{s\neg r\neg q\neg p} = \top$
$f_{\neg s} = ((p \wedge q) \vee (\neg p \wedge \neg r))$
$f_{\neg sr} = (p \wedge q)$
$f_{\neg srq} = \neg f_{s\neg r\neg q}$
$f_{\neg sr\neg q} = \bot$
$f_{\neg s\neg r} = f_{s\neg r}$

The final ROBDD:



3.2.16 [3 points] Construct a reduced ordered binary decision diagram (ROBDD) for the formula

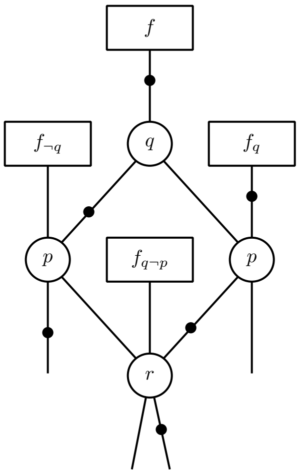
$$f = (p \vee q) \wedge \neg(p \wedge q) \wedge r$$

using *variable order* $q < p < r$. 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.

Solution

$f = (((p \vee q) \wedge \neg(p \wedge q)) \wedge r)$
$f_q = (\neg(p \wedge q) \wedge r)$
$f_{qp} = \bot$
$f_{q\neg p} = r$
$f_{q\neg pr} = \top$
$f_{q\neg p\neg r} = \bot$
$f_{\neg q} = (p \wedge r)$
$f_{\neg qp} = f_{q\neg p}$
$f_{\neg q\neg p} = \bot$

The final ROBDD:



3.2.17 [3 points] Construct a reduced ordered binary decision diagram (ROBDD) for the formula

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

using *variable order* $a < b < c < d$. 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.

Solution

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

$f_a = (\neg(c \vee d) \vee b)$

$f_{ab} = \top$

$f_{a\neg b} = \neg(c \vee d)$

$f_{a\neg bc} = \bot$

$f_{a\neg b\neg c} = \neg d$

$f_{a\neg b\neg cd} = \bot$

$f_{a\neg b\neg c\neg d} = \top$

$f_{\neg a} = (\neg b \wedge \neg(c \vee d))$

$f_{\neg ab} = \bot$

$f_{\neg a\neg b} = f_{a\neg b}$

The final ROBDD: