

Adiar 1.1 : Zero-suppressed Decision Diagrams in External Memory

Steffan Christ Sølvesten and Jaco van de Pol

NFM 2023









Adiar

Binary Decision Diagrams
in External Memory

`github.com/ssoelvsten/adiar`

Adiar

Multi-terminal Decision Diagrams
in External Memory

`github.com/ssoelvsten/adiar`

Adiar

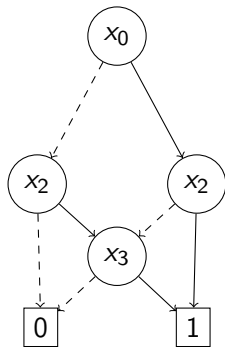
Quantum Multi-valued Decision Diagrams
in External Memory

`github.com/ssoelvsten/adiar`

Adiar

Zero-suppressed Decision Diagrams
in External Memory

`github.com/ssoelvsten/adiar`







BDD: $f : \mathbb{B}^n \rightarrow \mathbb{B}$



ZDD: $A \subseteq \mathbb{B}^n$

```
bdd bdd_apply(bdd f, bdd g, bool_op o)
```

```
bdd bdd_apply(bdd f, bdd g, bool_op o)
```

```
zdd zdd_binop(zdd A, zdd B, bool_op o)
```

```
bdd bdd_apply(bdd f, bdd g, bool_op o)  {  
    return prod2<bdd_policy>(f, g, o);  
}
```

```
zdd zdd_binop(zdd A, zdd B, bool_op o)  {  
    return prod2<zdd_policy>(A, B, o);  
}
```

```
bdd bdd_apply(bdd f, bdd g, bool_op o)  {  
    return prod2<bdd_policy>(f, g, o);  
}
```

```
zdd zdd_binop(zdd A, zdd B, bool_op o)  {  
    return prod2<zdd_policy>(A, B, o);  
}
```




—○— Adiar —◆— CUDD

Running time for *3D Tic-Tac-Toe* with 300 GiB of RAM.



Running time for *3D Tic-Tac-Toe* with 300 GiB of RAM.



Running time for *3D Tic-Tac-Toe* with 300 GiB of RAM.

Done

BDD ZDD

Doable

MTBDD

LDD

QMDD

Done

BDD

ZDD

(K)FDD

Tagged/Chained BDD

Open

Clock DD

MDD

Doable

MTBDD

LDD

QMDD

Done

BDD

ZDD

(K)FDD

Tagged/Chained BDD

Steffan Christ Sølvsten

✉ soelvsten@cs.au.dk

🌐 ssoelvsten.github.io

Adiar

🔗 github.com/ssoelvsten/adiar

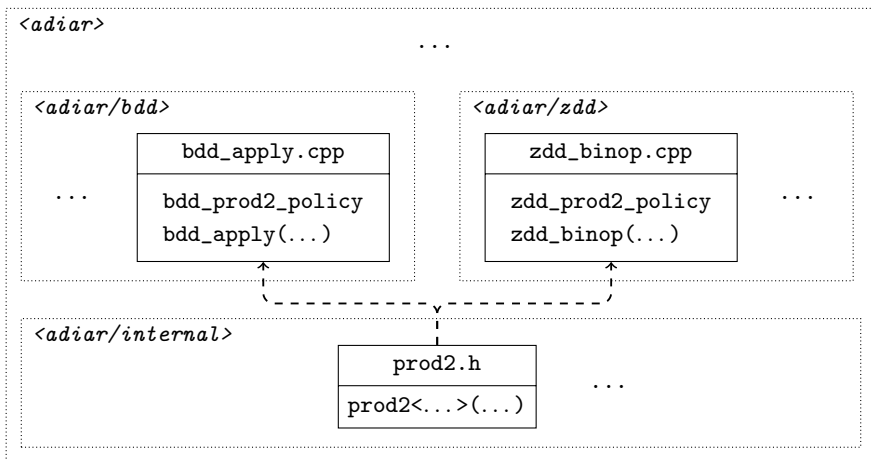
📖 ssoelvsten.github.io/adiar



| Function | Operation Semantics |
|---|---|
| ZDD Constructors | |
| <code>zdd_empty()</code> | \emptyset |
| <code>zdd_null()</code> | $\{\emptyset\}$ |
| <code>zdd_singleton(var)</code> | $\{x_{var}\}$ |
| <code>zdd_vars(vars)</code> | $\{\bigcup_{i \in vars} \{x_i\}\}$ |
| <code>zdd_singletons(vars)</code> | $\{\{x_i\} \mid i \in vars\}$ |
| <code>zdd_powerset(vars)</code> | $\mathcal{P}(vars)$ |
| <code>zdd_sized_set(vars, k, \odot)</code> | $\{s \in \mathcal{P}(vars) \mid s \odot k\}$ |
| ZDD Manipulation | |
| <code>zdd_binop(A, B, \otimes)</code> | $\{x \mid x \in A \otimes x \in B\}$ |
| <code>zdd_change(A, vars)</code> | $\{(a \setminus vars) \cup (vars \setminus a) \mid a \in A\}$ |
| <code>zdd_complement(A, dom)</code> | $\mathcal{P}(dom) \setminus A$ |
| <code>zdd_expand(A, vars)</code> | $\bigcup_{a \in A} \{a \cup v \mid v \in \mathcal{P}(vars)\}$ |
| <code>zdd_offset(A, vars)</code> | $\{a \in A \mid vars \cap a = \emptyset\}$ |
| <code>zdd_onset(A, vars)</code> | $\{a \in A \mid vars \subseteq a\}$ |
| <code>zdd_project(A, vars)</code> | $\bigcup_{a \in A} \{a \cap vars\}$ |

| Function | Operation Semantics |
|---------------------------------|---|
| Counting | |
| <code>zdd_size(A)</code> | $ A $ |
| <code>zdd_nodecount(A)</code> | # ZDD Nodes in A |
| <code>zdd_varcount(A)</code> | # Non-empty Levels in A |
| Predicates | |
| <code>zdd_equal(A, B)</code> | $A = B$ |
| <code>zdd_unequal(A, B)</code> | $A \neq B$ |
| <code>zdd_subseteq(A, B)</code> | $A \subseteq B$ |
| <code>zdd_disjoint(A, B)</code> | $A \cap B = \emptyset$ |
| Set elements | |
| <code>zdd_contains(A, a)</code> | $a \in A$ |
| <code>zdd_minelem(A)</code> | $a \in A \text{ s.t. } \forall a' \in A. a \leq a'$ |
| <code>zdd_maxelem(A)</code> | $a \in A \text{ s.t. } \forall a' \in A. a' \leq a$ |
| Conversion | |
| <code>zdd_from(f, dom)</code> | $\{x \in \mathcal{P}(dom) \mid f(x) = \top\}$ |
| <code>bdd_from(A, dom)</code> | $\vec{x} : \mathcal{P}(dom) \mapsto \vec{x} \in A$ |

Operations provided by Adiar in `<adiar/zdd.h>`.





(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Priority Queue: Q_{count} :

[

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Priority Queue: Q_{count} :

[

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Priority Queue: Q_{count} :

[$((0,0) \xrightarrow{\top} (1,0), 1)$,
 $((0,0) \xrightarrow{\perp} (2,0), 1)$,

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|----------|-----|--------|
| $(1, 0)$ | 0 | 0 |

Priority Queue: Q_{count} :

[$((0, 0) \xrightarrow{\top} (1, 0), 1)$,
 $((0, 0) \xrightarrow{\perp} (2, 0), 1)$,

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|----------|-----|--------|
| $(1, 0)$ | 0 | 0 |

Priority Queue: Q_{count} :

[$((0, 0) \xrightarrow{\top} (1, 0), 1)$,
 $((0, 0) \xrightarrow{\perp} (2, 0), 1)$,

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|----------|-----|--------|
| $(1, 0)$ | 1 | 0 |

Priority Queue: Q_{count} :

[
 $((0, 0) \xrightarrow{\perp} (2, 0), 1)$,
]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|----------|-----|--------|
| $(1, 0)$ | 1 | 0 |

Priority Queue: Q_{count} :

[
 $((0, 0) \xrightarrow{\perp} (2, 0), 1)$,
 $((1, 0) \xrightarrow{\perp} (2, 0), 1)$,
 $((1, 0) \xrightarrow{\top} (3, 1), 1)$,
]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|--------|-----|--------|
| (2, 0) | 0 | 0 |

Priority Queue: Q_{count} :

[
 $((0, 0) \xrightarrow{\perp} (2, 0), 1)$,
 $((1, 0) \xrightarrow{\perp} (2, 0), 1)$,
 $((1, 0) \xrightarrow{\top} (3, 1), 1)$,
]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (2, 0) | 0 | 0 |

Priority Queue: Q_{count} :

[

| | |
|--|---|
| $((0, 0) \xrightarrow{\perp} (2, 0), 1)$ | , |
| $((1, 0) \xrightarrow{\perp} (2, 0), 1)$ | , |
| $((1, 0) \xrightarrow{\top} (3, 1), 1)$ | , |

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|----------|-----|--------|
| $(2, 0)$ | 1 | 0 |

Priority Queue: Q_{count} :

[

$((1, 0) \xrightarrow{\perp} (2, 0), 1)$,

$((1, 0) \xrightarrow{\top} (3, 1), 1)$,

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|--------|-----|--------|
| (2, 0) | 2 | 0 |

Priority Queue: Q_{count} :

[

$((1, 0) \xrightarrow{\top} (3, 1), 1)$,
]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (2, 0) | 2 | 0 |

Priority Queue: Q_{count} :

[

$((2, 0) \xrightarrow{\perp} (3, 0), \quad 2) \quad ,$
 $((1, 0) \xrightarrow{\top} (3, 1), \quad 1) \quad ,$
 $((2, 0) \xrightarrow{\top} (3, 1), \quad 2) \quad]$



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (3, 0) | 0 | 0 |

Priority Queue: Q_{count} :

[

| | | |
|---------------------------------------|---|---|
| $((2, 0) \xrightarrow{\perp} (3, 0),$ | 2 | , |
| $((1, 0) \xrightarrow{\top} (3, 1),$ | 1 | , |
| $((2, 0) \xrightarrow{\top} (3, 1),$ | 2 |] |



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|--------|-----|--------|
| (3, 0) | 0 | 0 |

Priority Queue: Q_{count} :

[

| | | |
|---------------------------------------|---|---|
| $((2, 0) \xrightarrow{\perp} (3, 0),$ | 2 | , |
| $((1, 0) \xrightarrow{\top} (3, 1),$ | 1 | , |
| $((2, 0) \xrightarrow{\top} (3, 1),$ | 2 |] |



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (3, 0) | 2 | 0 |

Priority Queue: Q_{count} :

[

$((1, 0) \xrightarrow{T} (3, 1), 1)$,
 $((2, 0) \xrightarrow{T} (3, 1), 2)$]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (3, 0) | 2 | 2 |

Priority Queue: Q_{count} :

[

$((1, 0) \xrightarrow{T} (3, 1), 1)$,
 $((2, 0) \xrightarrow{T} (3, 1), 2)$]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|--------|-----|--------|
| (3, 1) | 0 | 2 |

Priority Queue: Q_{count} :

[

$((1, 0) \xrightarrow{T} (3, 1), 1)$,
 $((2, 0) \xrightarrow{T} (3, 1), 2)$]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (3, 1) | 0 | 2 |

Priority Queue: Q_{count} :

[

$((1, 0) \xrightarrow{T} (3, 1), 1)$,
 $((2, 0) \xrightarrow{T} (3, 1), 2)$]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|--------|-----|--------|
| (3, 1) | 1 | 2 |

Priority Queue: Q_{count} :

[

$((2, 0) \xrightarrow{\top} (3, 1), \quad 2) \quad]$



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| Seek | Sum | Result |
|---------------|-----|--------|
| (3, 1) | 3 | 2 |

Priority Queue: Q_{count} :

[

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Seek
(3, 1)

Sum
3

Result
5

Priority Queue: Q_{count} :

[

]



(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Result
5

Priority Queue: Q_{count} :

[

]

Steffan Christ Sølvsten

✉ soelvsten@cs.au.dk

🌐 ssoelvsten.github.io

Adiar

🔗 github.com/ssoelvsten/adiar

📖 ssoelvsten.github.io/adiar