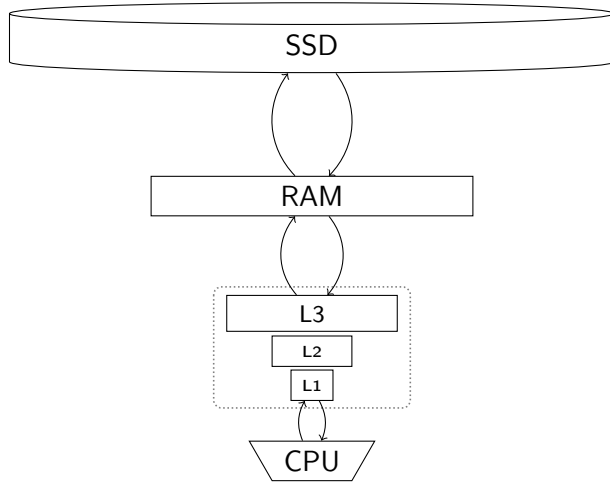


Adiar 1.1 : Zero-suppressed Decision Diagrams in External Memory

Steffan Christ Sølvesten and Jaco van de Pol

18th of May, 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)  
}
```

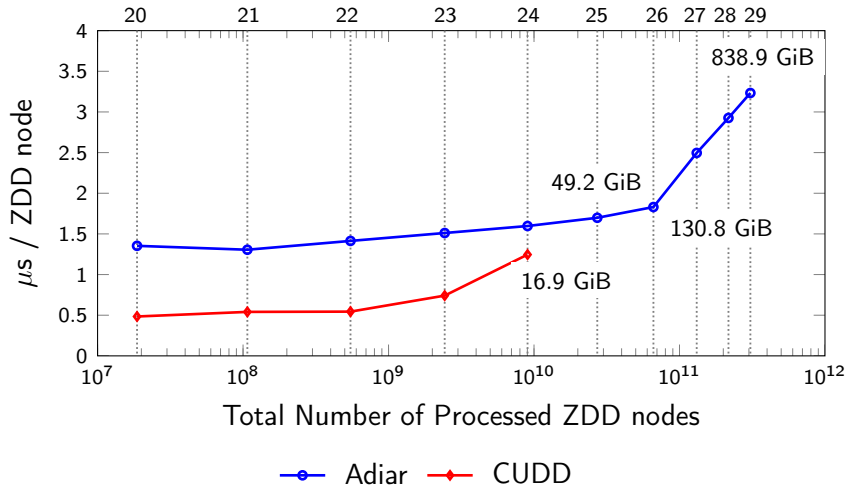
```
zdd zdd_binop(zdd A, zdd B, bool_op o)  {  
    return prod2<zdd_policy>(A, B, o)  
}
```




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

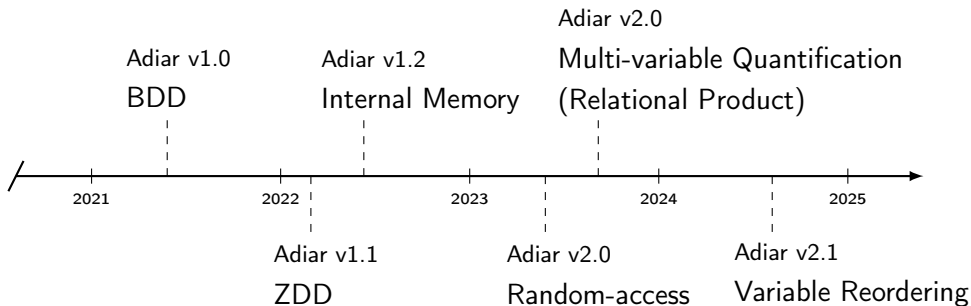
✉ soelvsten@cs.au.dk

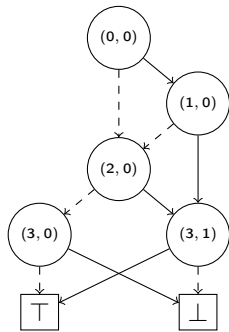
🐦 [@ssoelvsten](https://twitter.com/ssoelvsten)

Adiar

📄 github.com/ssoelvsten/adiar

📖 ssoelvsten.github.io/adiar





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

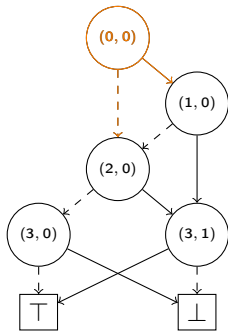


(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Priority Queue: Q_{count} :

[

]

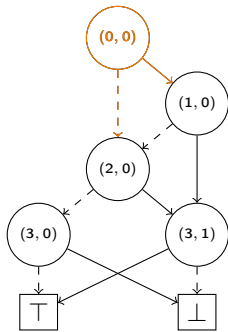


(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

Priority Queue: Q_{count} :

[

]



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

Priority Queue: Q_{count} :

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

]



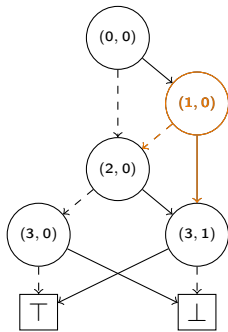
(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

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

Priority Queue: Q_{count} :

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

]



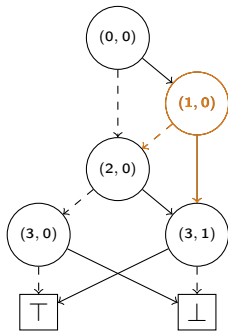
(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

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

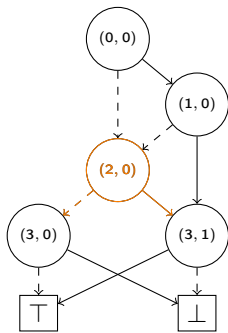
| Seek | Sum | Result |
|----------|-----|--------|
| $(2, 0)$ | 0 | 0 |

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} :

[

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

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

]



(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} :

[

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



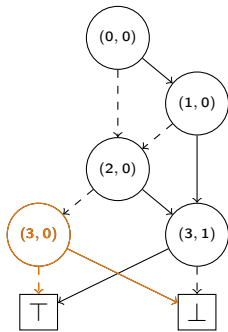
(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

| 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 |] |



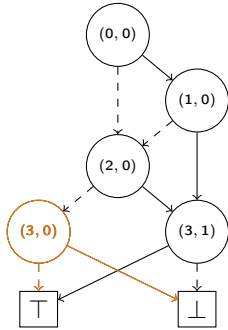
(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

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

[

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



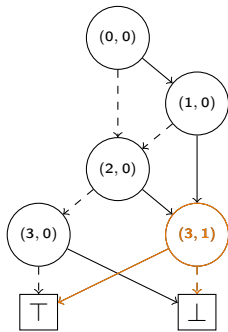
(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

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

Priority Queue: Q_{count} :

[

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



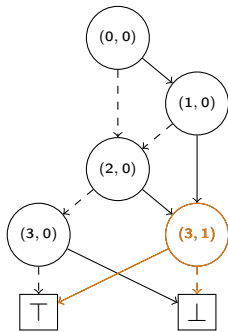
(a) $(x_0 \wedge x_1 \wedge x_3) \vee (x_2 \oplus x_3)$

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

Priority Queue: Q_{count} :

[

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



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

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

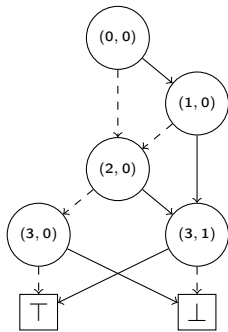
Sum
3

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](https://twitter.com/ssoelvsten)

Adiar

🔗 github.com/ssoelvsten/adiar

📖 ssoelvsten.github.io/adiar

