FLOW-3D: Flow-Based Computing on 3D Nanoscale Crossbars with Minimal Semiperimeter

Sven Thijssen
University of Central Florida
Orlando, FL, USA

Sumit Kumar Jha
University of Texas at San Antonio
San Antonio, TX, USA

Rickard Ewetz
University of Central Florida
Orlando, FL, USA

Overview

- Big data and deep learning
- Motivation for in-memory computing
- In-memory computing using non-volatile resistive devices
- In-memory computing paradigms
- Flow-based computing
- 3D nanoscale memristor crossbars
- Problem definition
- Analogy between BDDs and 3D crossbars
- The FLOW-3D framework
- Experimental results
- Conclusion

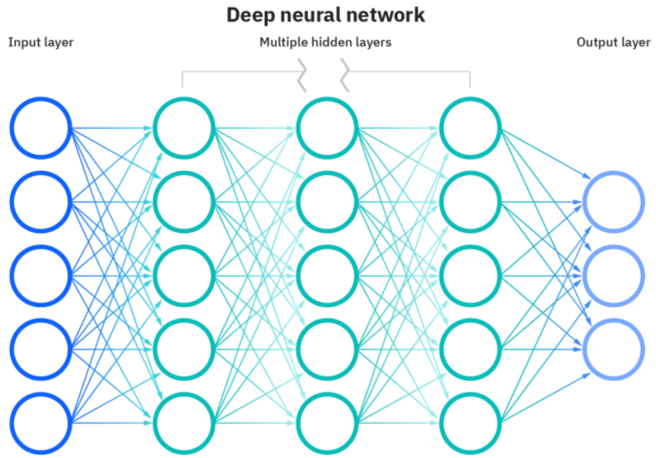
Big data and deep learning

Internet of Things + 5G

→ big data

→ deep learning

Internet of Things [1]



Deep neural network [2]

Tesla neural network [3]

[1] Image from https://news.mit.edu/2020/iot-deep-learning-1113. Accessed on January 13, 2023.
[2] Image from https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks. Accessed on January 12, 2023.
[3] Video from https://www.tesla.com/AI. Accessed on January 12, 2023.

Motivation for in-memory computing

Limitations of current computer architectures

- Von Neumann Bottleneck [1]
- End of Dennard Scaling [2]
- End of Moore's Law [3]

How to improve energy and latency?

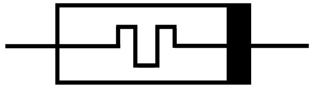
Solution: merge memory and computing units = in-memory computing

Von Neumann bottleneck

[1] Backus, J. (1978). Can programming be liberated from the von Neumann style? A functional style and its algebra of programs. Communications of the ACM, 21(8), 613-641.
[2] Esmaeilzadeh, H., Blem, E., Amant, R. S., Sankaralingam, K., & Burger, D. (2011, June). Dark silicon and the end of multicore scaling. In 2011 38th Annual international symposium on computer architecture (ISCA) (pp. 365-376). IEEE.
[3] Theis, T. N., & Wong, H. S. P. (2017). The end of moore's law: A new beginning for information technology. Computing in Science & Engineering, 19(2), 41-50.

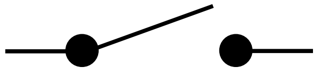
In-memory computing using non-volatile resistive devices

Memristor [1]



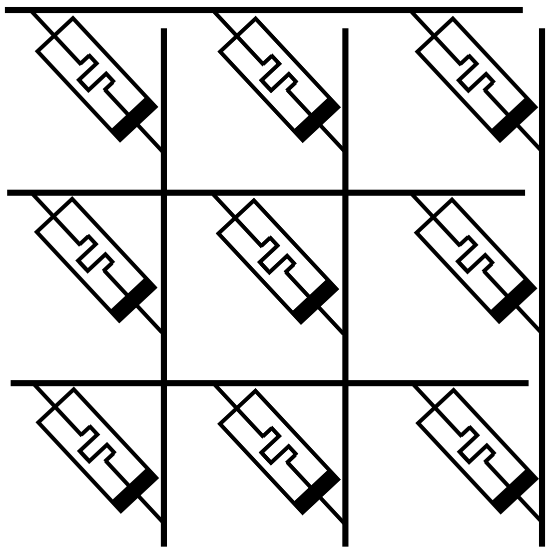
- Non-volatile resistive device
- Behavior of a switch
- Two states in digital in-memory computing: ON/OFF

Low resistive state (ON)



High resistive state (OFF)

Nanoscale memristor crossbar



- 2D array of memristors between wordlines and bitlines
- Share peripheral circuitry (DAC, ADC)

How do we perform computations (Boolean function evaluations)?

[1] Chua, L. (1971). Memristor-the missing circuit element. IEEE Transactions on circuit theory, 18(5), 507-519.

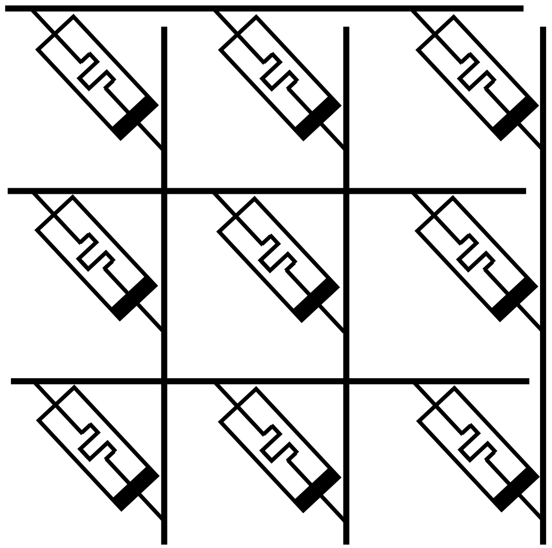
In-memory computing paradigms

Overview

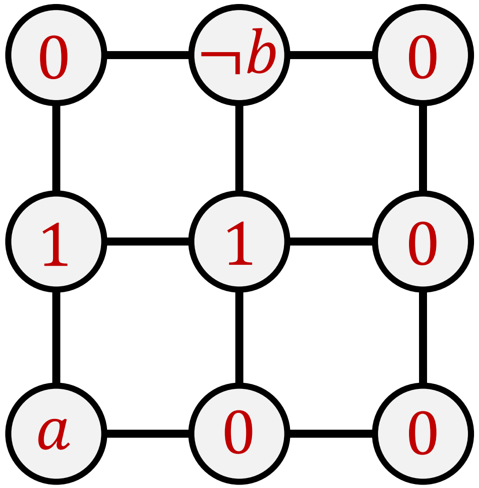
- IMPLY [1]
- MAGIC [2]
- FLOW [3]

How do we evaluate

$$\boldsymbol{f = a \land \neg b?}$$



What if we assign Boolean literals (variables and their negations), ON (1) and OFF (0) to the memristors?



Terminology: crossbar design

[1] Borghetti, J., Snider, G. S., Kuekes, P. J., Yang, J. J., Stewart, D. R., & Williams, R. S. (2010). 'Memristive'switches enable 'stateful'logic operations via material implication. Nature, 464(7290), 873-876.

[2] Kvatinsky, S., Belousov, D., Liman, S., Satat, G., Wald, N., Friedman, E. G., ... & Weiser, U. C. (2014). MAGIC—Memristor-aided logic. IEEE Transactions on Circuits and Systems II: Express Briefs, 61(11), 895-899.

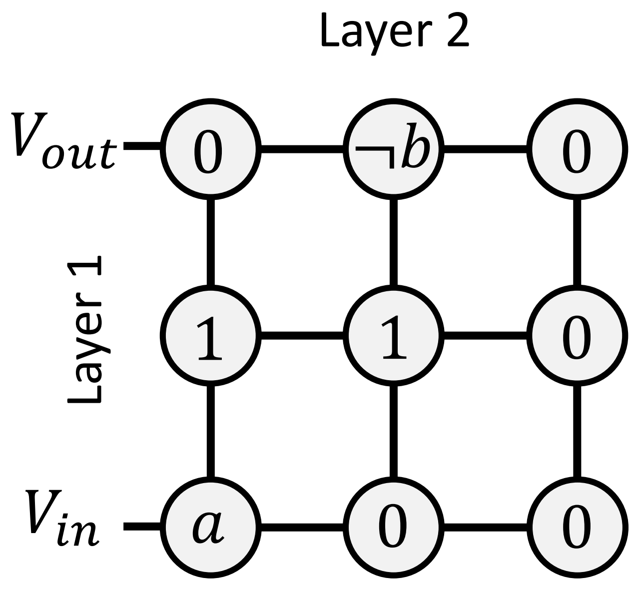
[3] Jha, S. K., Rodriguez, D. E., Van Nostrand, J. E., & Velasquez, A. (2016). U.S. Patent No. 9,319,047. Washington, DC: U.S. Patent and Trademark Office.

Flow-based computing

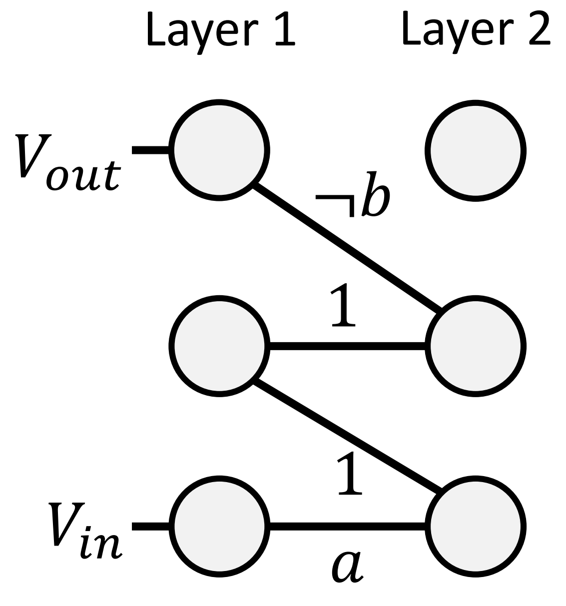
A Boolean function f evaluates to true if and only if there is a path from the input nanowire to the output nanowire along memristors in low resistive state.

$$f = a \wedge \neg b$$

a	b	f
0	0	0
0	1	0
1	0	1
1	1	0



Crossbar design



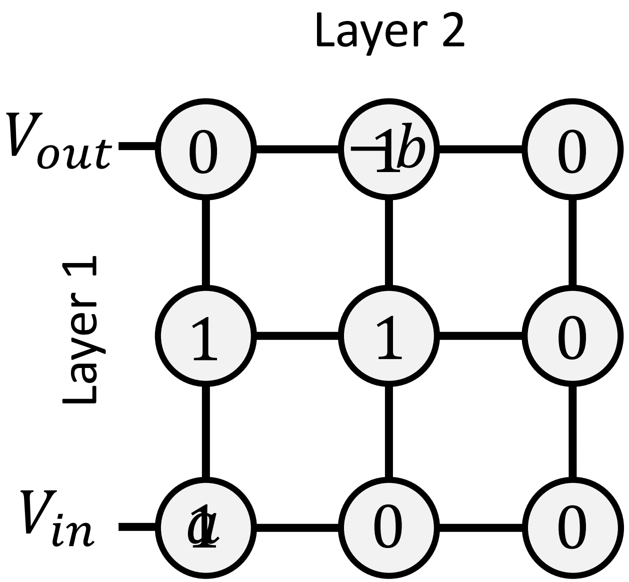
Evaluation

Flow-based computing

A Boolean function f evaluates to true if and only if there is a path from the input nanowire to the output nanowire along memristors in low resistive state.

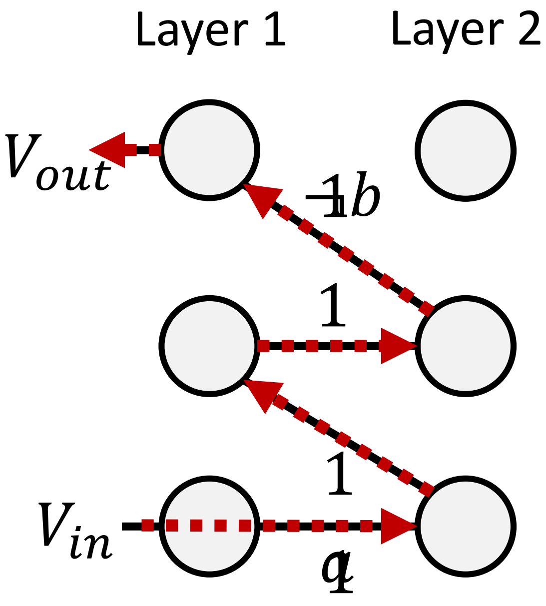
$$f = a \land \neg b$$

a	b	f
0	0	0
0	1	0
1	0	1
1	1	0



Crossbar design

$$f = 1$$



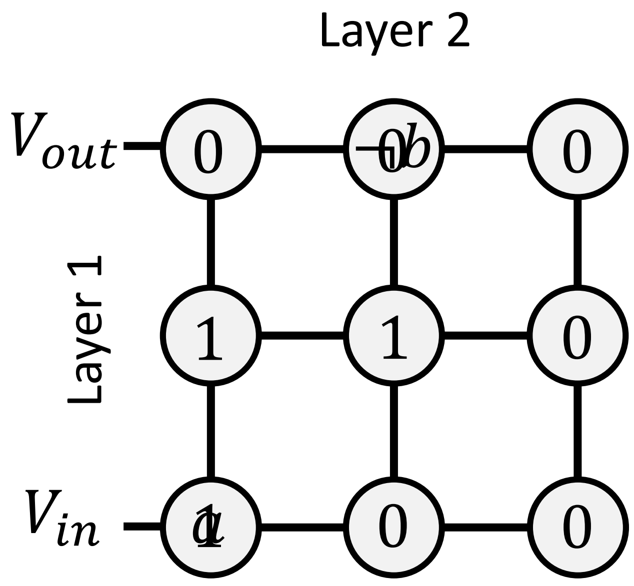
Evaluation

Flow-based computing

A Boolean function f evaluates to true if and only if there is a path from the input nanowire to the output nanowire along memristors in low resistive state.

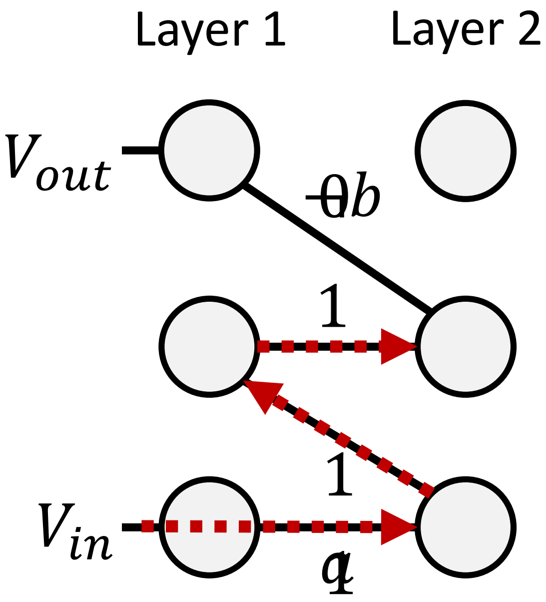
$$f = a \wedge \neg b$$

a	b	f
0	0	0
0	1	0
1	0	1
1	1	0



Crossbar design

$$f = 0$$



Evaluation

Flow-based computing

Synthesis consists of two phases:

- Initialization phase (once)
- Execution phase (many times)

```
module f (a, b, f);
    input a, b;
    output f;
    assign f = a & b;
endmodule
```

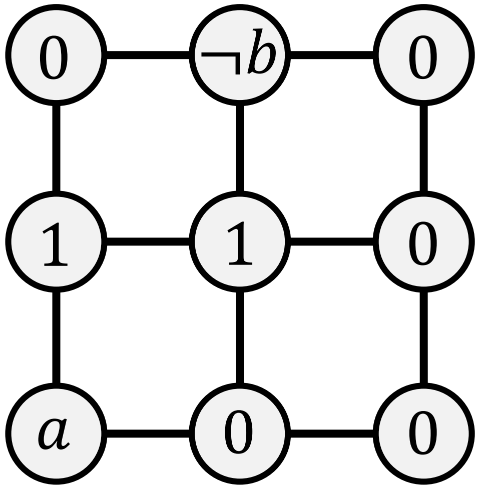
Verilog

[1] Brayton, R., & Mishchenko, A. (2010, July). ABC: An academic industrial-strength verification tool. In *International Conference on Computer Aided Verification* (pp. 24-40). Springer, Berlin, Heidelberg.

Flow-based computing

Synthesis consists of two phases:

- Initialization phase (once)
- Execution phase (many times)

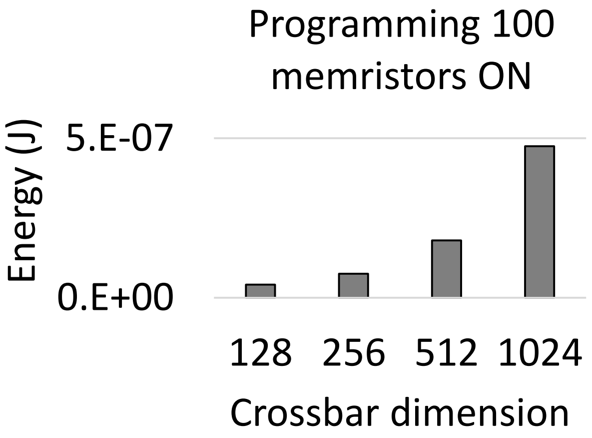


Crossbar design

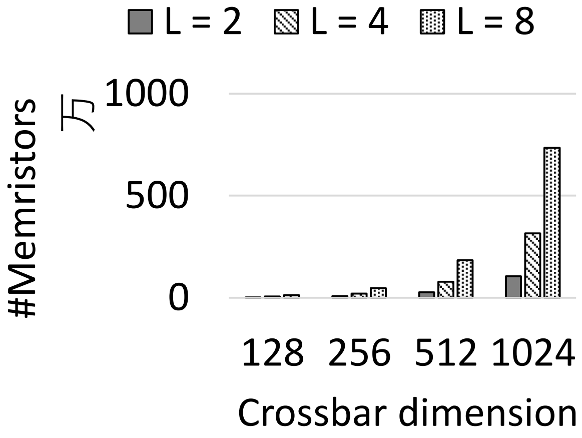
[1] Brayton, R., & Mishchenko, A. (2010, July). ABC: An academic industrial-strength verification tool. In *International Conference on Computer Aided Verification* (pp. 24-40). Springer, Berlin, Heidelberg.

3D nanoscale memristor crossbars

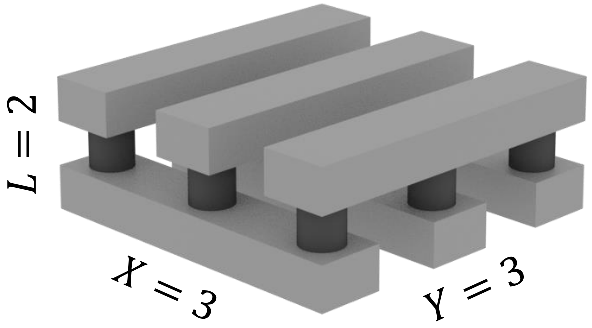
1. Shorter metal wires
   → mitigation of crossbar parasitics
   → improvement of READ/WRITE latency
   → energy reduction
2. Higher density per unit area



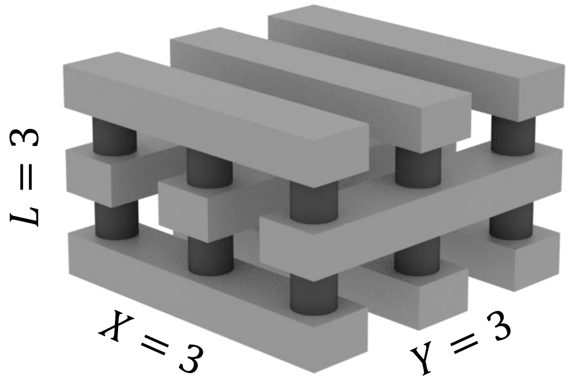
1. WRITE latency



2. Density

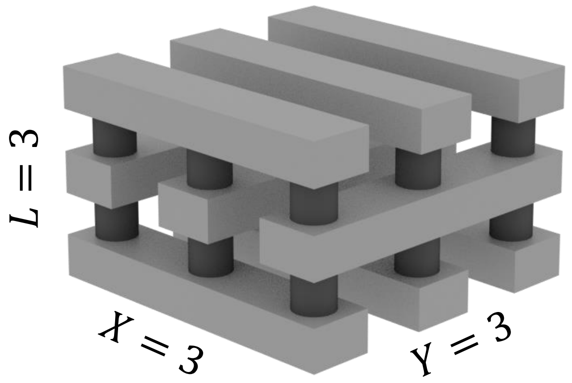


2D crossbar



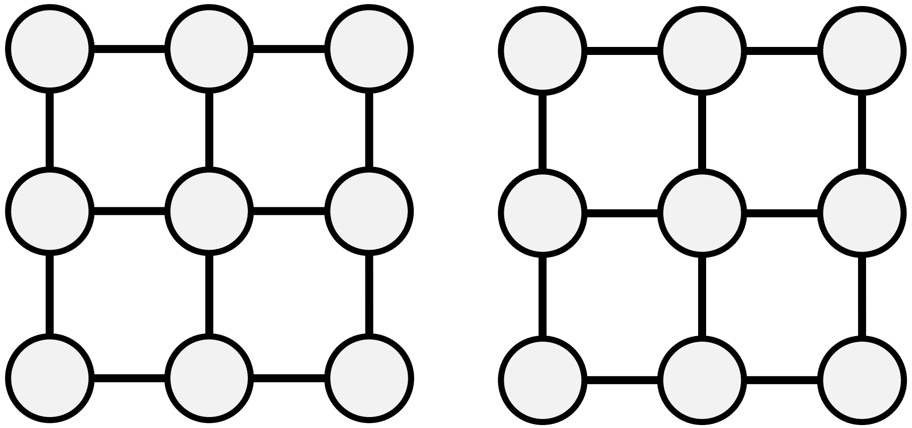
3D crossbar

3D nanoscale memristor crossbar



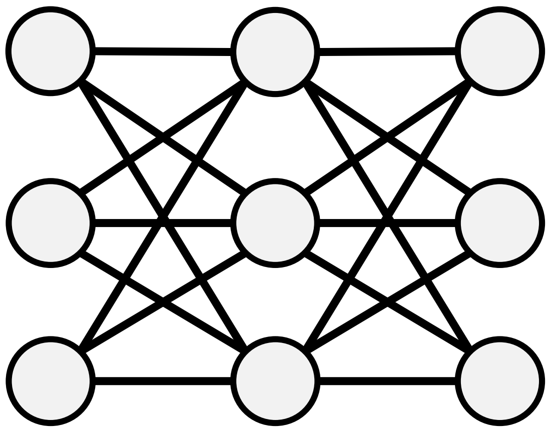
Reference crossbar

Layer 1 to Layer 2 Layer 2 to Layer 3



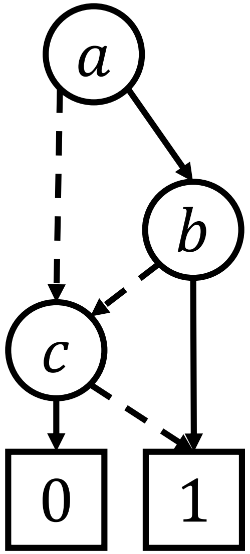
Matrix model

Layer 1 Layer 2 Layer 3



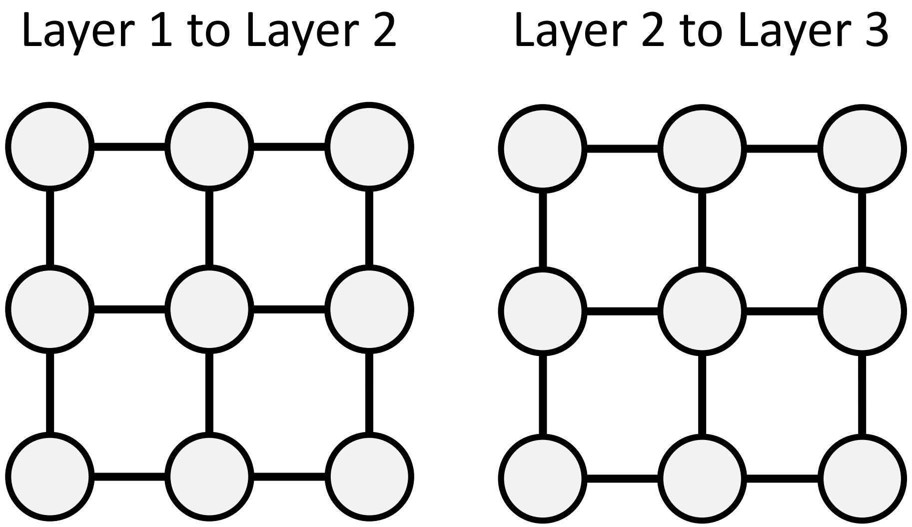
Crossbar model

How do we find a design for a Boolean function f on a 3D nanoscale crossbar?



BDD

?

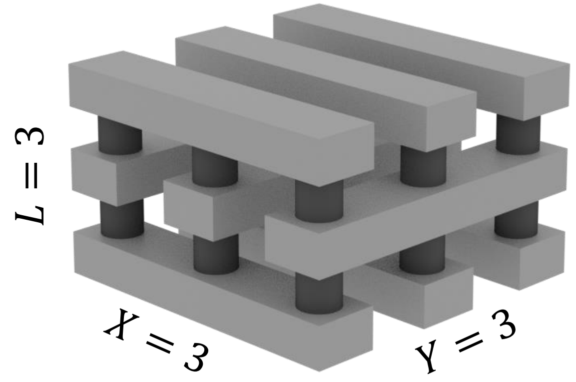


Crossbar design

Problem definition

Given the BDD of a Boolean function f, minimize the semiperimeter $X+Y$ of the crossbar design given a fixed number of layers L.

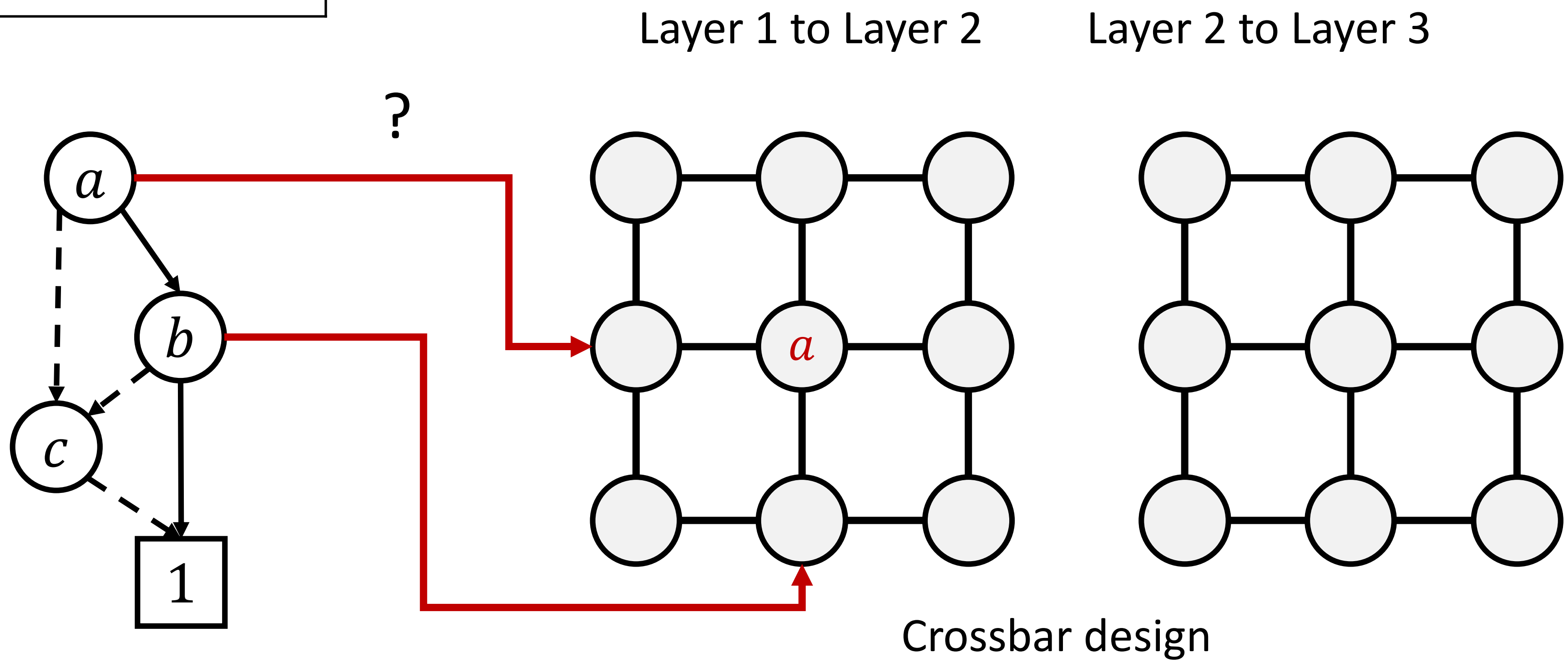
$$\min \ X+Y$$



Reference crossbar

Analogy between BDDs and 3D crossbars

BDD	Crossbar
Nodes	Metal wires
Edges	Memristors



Analogy between BDDs and 3D crossbars

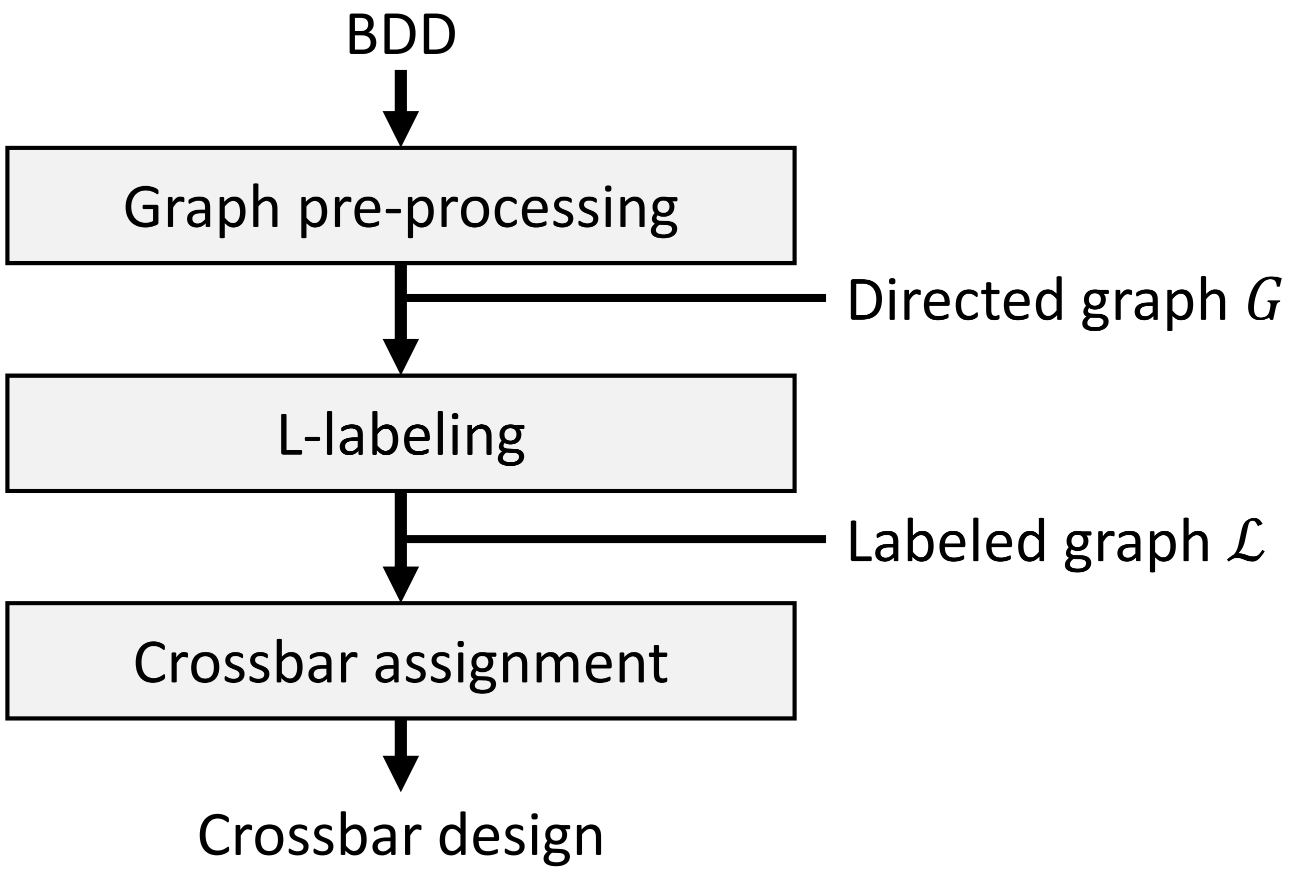
Analogy

BDD	**Crossbar**
Nodes	Metal wires
Edges	Memristors

Constraints

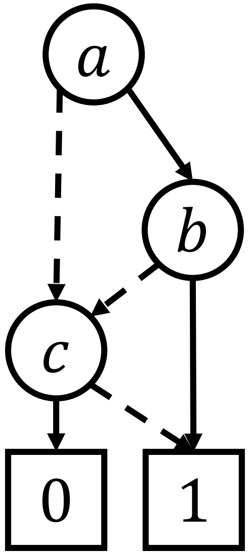
1. Edge constraints
   Two nodes in a BDD connected by an edge must be assigned to metal wires in adjacent layers in the crossbar
2. Node constraints
   Nodes can be assigned to multiple layers and must be connected

The FLOW-3D framework: overview



The FLOW-3D framework

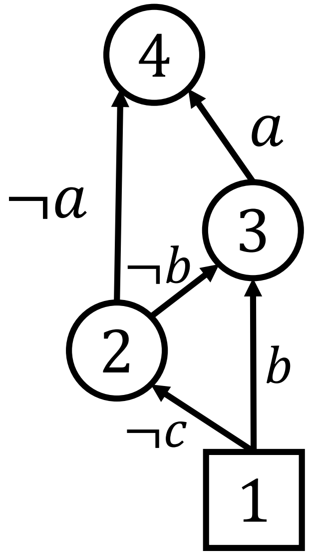
Step 1: graph pre-processing



BDD

The FLOW-3D framework

Step 2: L-labeling



Graph $\mathcal{G}$

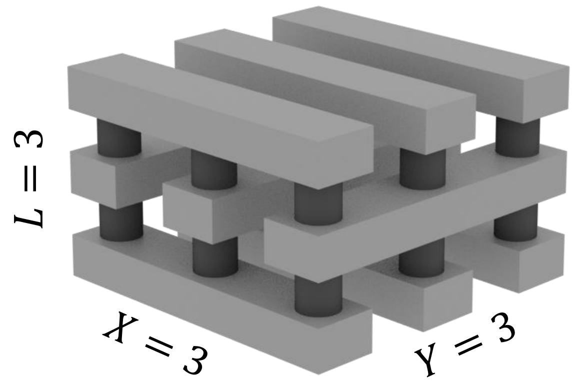
The FLOW-3D framework: L-labeling

$$
\begin{aligned}
\min \quad & X + Y & (1)\\
\text{s.t.} \quad & \sum_{v \in G} x_v^l \le X, \qquad \forall l \in L_{even} & (2)\\
& \sum_{v \in G} x_v^l \le Y, \qquad \forall l \in L_{odd} & (3)\\
& \sum_{l \in \{1, L-1\}} s_{v,u}^{l,l+1} + s_{u,v}^{l,l+1} = 1, \qquad \forall (v,u) \in G & (4)\\
& x_u^l + x_v^{l+1} \ge 2 s_{u,v}^{l,l+1}, \qquad \forall (v,u) \in G, \forall l \in \{1, L-1\} & (5)\\
& x_v^l + x_u^{l+1} \ge 2 s_{v,u}^{l+1,l}, \qquad \forall (v,u) \in G, \forall l \in \{1, L-1\} & (6)\\
& \sum x_v = d_v, \qquad \forall v \in G & (7)\\
& M(1 - (x_v^{l_v^{lb}} + x_v^{l^{ub}} - 1)) + d_v \ge l_v^{ub} - l_v^{lb} + 1, & (8)\\
& \qquad M \gg 0, \forall v \in G, \forall l_v^{lb}, l_v^{ub} \in L, l_v^{lb} < l_v^{ub} &
\end{aligned}
$$

The objective is to minimize the semiperimeter $X + Y$

where X is the maximum dimension of the even layers

and Y is the maximum dimensions of the odd layers



Reference crossbar

The FLOW-3D framework: L-labeling

$$
\begin{aligned}
\min \quad & X + Y && (1)\\
\text{s.t.} \quad & \sum_{v \in G} x_v^l \leq X, \qquad \forall l \in L_{even} && (2)\\
& \sum_{v \in G} x_v^l \leq Y, \qquad \forall l \in L_{odd} && (3)\\
& \sum_{l \in \{1, L-1\}} s_{v,u}^{l,l+1} + s_{u,v}^{l,l+1} = 1, \qquad \forall (v,u) \in G && (4)\\
& x_u^l + x_v^{l+1} \geq 2 s_{u,v}^{l,l+1}, \qquad \forall (v,u) \in G, \forall l \in \{1, L-1\} && (5)\\
& x_v^l + x_u^{l+1} \geq 2 s_{v,u}^{l+1,l}, \qquad \forall (v,u) \in G, \forall l \in \{1, L-1\} && (6)\\
& \sum x_v = d_v, \qquad \forall v \in G && (7)\\
& M(1 - (x_v^{l_v^{lb}} + x_v^{l^{ub}} - 1)) + d_v \geq l_v^{ub} - l_v^{lb} + 1, && (8)\\
& \qquad M \gg 0, \forall v \in G, \forall l_v^{lb}, l_v^{ub} \in L, l_v^{lb} < l_v^{ub}
\end{aligned}
$$

For each node v in the BDD and for each layer l in the crossbar, we introduce a binary variable x_v^l.

If $x_v^l = 1$, then node v is assigned to layer l. Otherwise, not.

→ defines a range of layers to which a node v is assigned.

The FLOW-3D framework: L-labeling

$$\min \quad X + Y \tag{1}$$

$$\text{s.t.} \quad \sum_{v \in G} x_v^l \leq X, \qquad \forall l \in L_{even} \tag{2}$$

$$\sum_{v \in G} x_v^l \leq Y, \qquad \forall l \in L_{odd} \tag{3}$$

$$\sum_{l \in \{1, L-1\}} s_{v,u}^{l,l+1} + s_{u,v}^{l,l+1} = 1, \qquad \forall (v,u) \in G \tag{4}$$

$$x_u^l + x_v^{l+1} \geq 2 s_{u,v}^{l,l+1}, \qquad \forall (v,u) \in G, \forall l \in \{1, L-1\} \tag{5}$$

$$x_v^l + x_u^{l+1} \geq 2 s_{v,u}^{l+1,l}, \qquad \forall (v,u) \in G, \forall l \in \{1, L-1\} \tag{6}$$

$$\sum x_v = d_v, \qquad \forall v \in G \tag{7}$$

$$M(1 - (x_v^{l_v^{lb}} + x_v^{l_v^{ub}} - 1)) + d_v \geq l_v^{ub} - l_v^{lb} + 1, \tag{8}$$

$$M \gg 0, \forall v \in G, \forall l_v^{lb}, l_v^{ub} \in L, l_v^{lb} < l_v^{ub}$$

1. Edge constraints
   Two nodes in a BDD connected by an edge must be assigned to metal wires in adjacent layers in the crossbar

2. Node constraints
   Nodes can be assigned to multiple layers and must be connected

The FLOW-3D framework: L-labeling

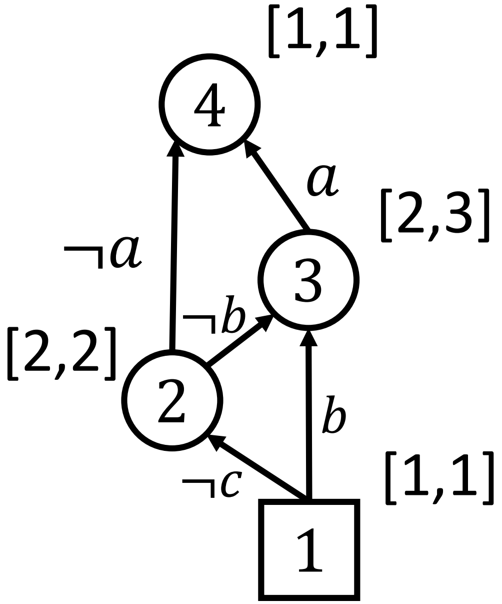
$$
\begin{aligned}
\min \quad & X + Y && (1)\\
\text{s.t.} \quad & \sum_{v \in G} x_v^l \le X, \quad \forall l \in L_{even} && (2)\\
& \sum_{v \in G} x_v^l \le Y, \quad \forall l \in L_{odd} && (3)\\
& \sum_{l \in \{1, L-1\}} s_{v,u}^{l,l+1} + s_{u,v}^{l,l+1} = 1, \quad \forall (v,u) \in G && (4)\\
& x_u^l + x_v^{l+1} \ge 2 s_{u,v}^{l,l+1}, \quad \forall (v,u) \in G, \forall l \in \{1, L-1\} && (5)\\
& x_v^l + x_u^{l+1} \ge 2 s_{v,u}^{l+1,l}, \quad \forall (v,u) \in G, \forall l \in \{1, L-1\} && (6)\\
& \sum x_v = d_v, \quad \forall v \in G && (7)\\
& M(1 - (x_v^{l_v^{lb}} + x_v^{l_v^{ub}} - 1)) + d_v \ge l_v^{ub} - l_v^{lb} + 1, && (8)\\
& \quad M \gg 0, \forall v \in G, \forall l_v^{lb}, l_v^{ub} \in L, l_v^{lb} < l_v^{ub}
\end{aligned}
$$

1. Edge constraints
   Two nodes in a BDD connected by an edge must be assigned to metal wires in adjacent layers in the crossbar

2. Node constraints
   Nodes can be assigned to multiple layers and must be connected

The FLOW-3D framework

Step 3: crossbar assignment



Labeled graph $\mathcal{L}$

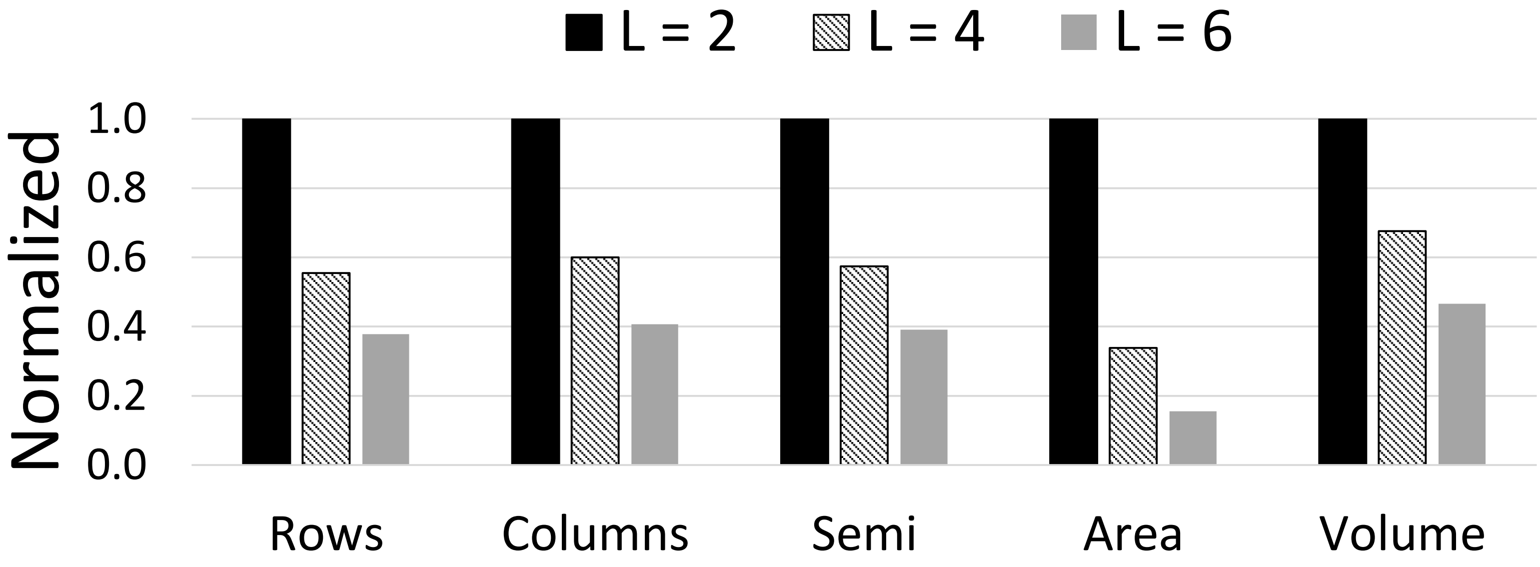
Experimental results

- Source code available on GitHub: https://github.com/sventhijssen/flow-3d
- 15 benchmarks from Revlib [1]
- WRITE latency depends on the voltage drop across the device [2]

[1] Wille, R., Große, D., Teuber, L., Dueck, G. W., & Drechsler, R. (2008, May). RevLib: An online resource for reversible functions and reversible circuits. In *38th International Symposium on Multiple Valued Logic (ismvl 2008)* (pp. 220-225). IEEE.
[2] Xu, C., Niu, D., Muralimanohar, N., Balasubramonian, R., Zhang, T., Yu, S., & Xie, Y. (2015, February). Overcoming the challenges of crossbar resistive memory architectures. In 2015 IEEE 21st international symposium on high performance computer architecture (HPCA) (pp. 476-488). IEEE.

Experimental results

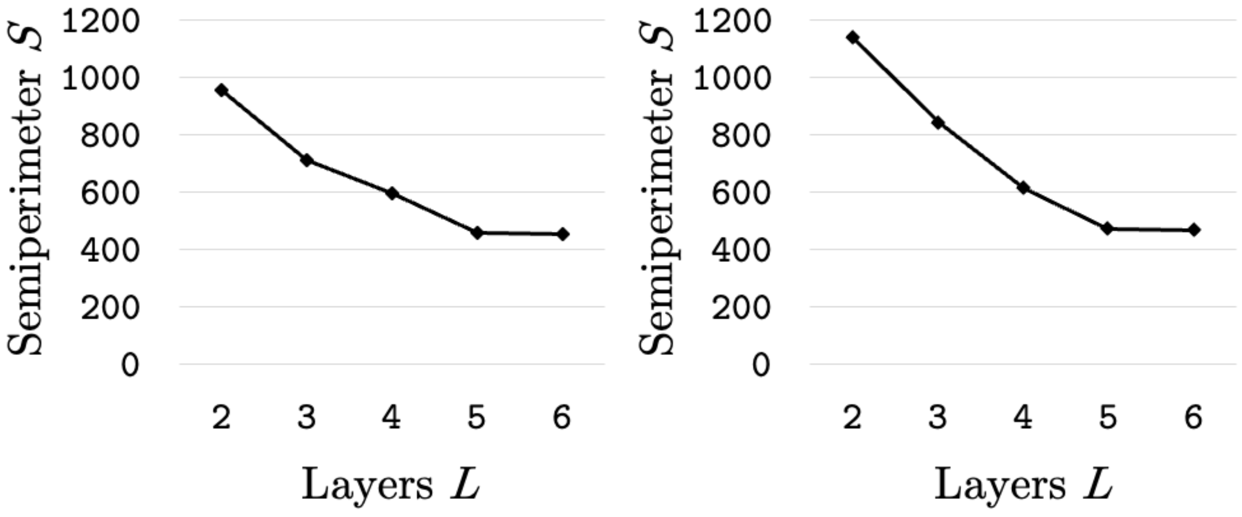
Hardware resources for increasing number of layers L



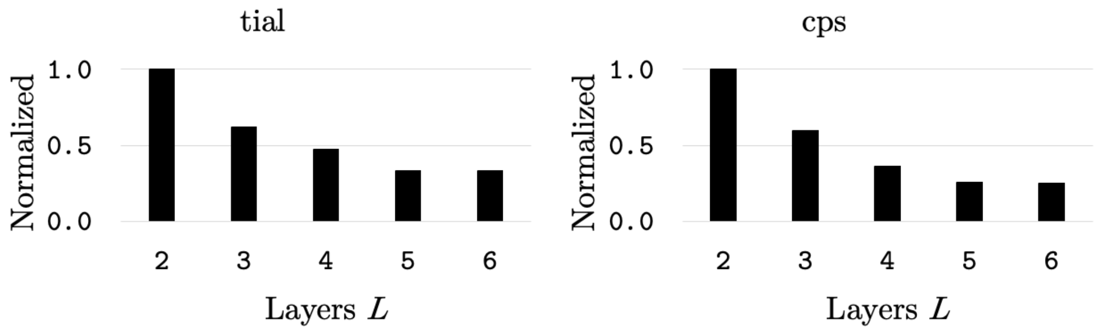
[1] Wille, R., Große, D., Teuber, L., Dueck, G. W., & Drechsler, R. (2008, May). RevLib: An online resource for reversible functions and reversible circuits. In *38th International Symposium on Multiple Valued Logic (ismvl 2008)* (pp. 220-225). IEEE.
[2] Xu, C., Niu, D., Muralimanohar, N., Balasubramonian, R., Zhang, T., Yu, S., & Xie, Y. (2015, February). Overcoming the challenges of crossbar resistive memory architectures. In 2015 IEEE 21st international symposium on high performance computer architecture (HPCA) (pp. 476-488). IEEE.

Experimental results

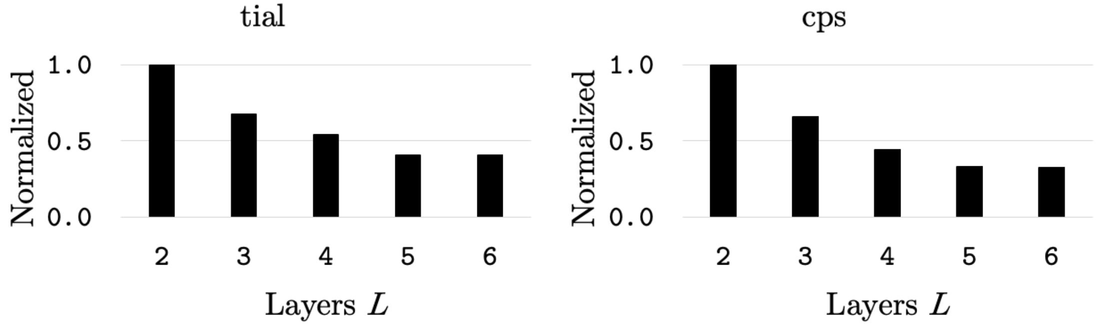
Hardware utilization, energy and latency for different layers $L = 2$, $L = 4$, and $L = 6$



(a) tial (b) cps



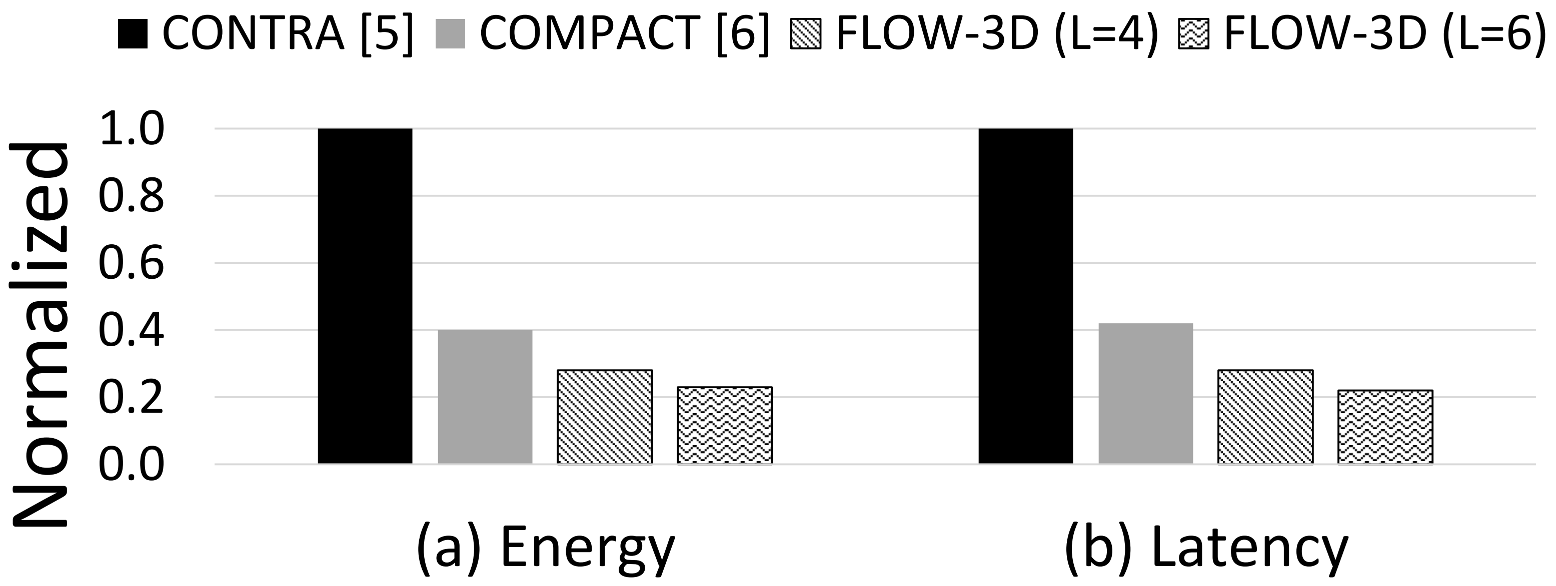
(a) Latency



(b) Energy

Experimental results

Comparison of energy and latency of FLOW-3D with COMPACT [1] and CONTRA [2]



[1] Thijssen, S., Jha, S. K., & Ewetz, R. (2021). COMPACT: Flow-Based Computing on Nanoscale Crossbars with Minimal Semiperimeter and Maximum Dimension. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.
[2] Bhattacharjee, D., Chattopadhyay, A., Dutta, S., Ronen, R., & Kvatinsky, S. (2020, November). Contra: area-constrained technology mapping framework for memristive memory processing unit. In Proceedings of the 39th International Conference on Computer-Aided Design (pp. 1-9).

Conclusion

- Analogy between BDDs and 3D crossbar
- Synthesis framework FLOW-3D using ILP formulation based on L-labeling
- Proposed framework improves semiperimeter, area, energy, and latency up to 61%, 84%, 37%, and 41% compared with COMPACT, the state-of-the-art synthesis tool for flow-based computing on 2D crossbars

Thank you