Combining Cubic Dynamical Solvers with Make/Break Heuristics to Solve SAT

Anshujit Sharma, Matthew Burns, Michael Huang
University of Rochester
Department of Electrical and Computer Engineering

July 6, 2023

Outline

Classical computing slowdown:
"The End of Moore's Law"
Particularly acute for SAT

- Application-specific accelerators [Sohanghpurwala et al., 2017, Davis et al., 2008]
- Parallel computing [Hamadi et al., 2010, Osama et al., 2021]
- Dynamical computing [Gabor et al., 2019, Choi, 2010, Bashar et al., 2023]

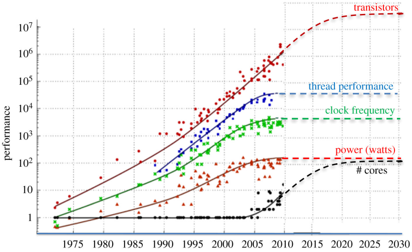


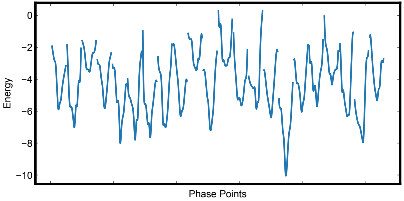
Figure: Predicted scaling of Moore's law. From [Shalf, 2020]

The use of a naturally evolving system to complete a computational task

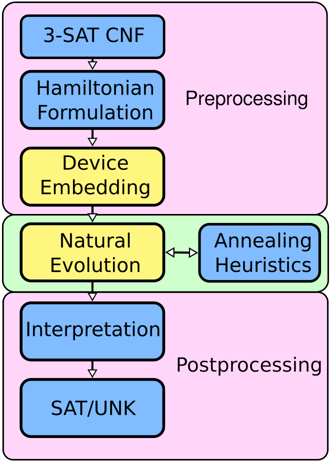
Dynamical Computing Workflow

$$F = \bigwedge_i^M (\ell_{i,1} \bigvee \ell_{i,2} \bigvee \ell_{i,3})$$

$$\mathcal{H} = \sum_i^M g(\ell_{i,1}) g(\ell_{i,2}) g(\ell_{i,3})$$



$$x_n = \mathrm{STEP}(\tilde{x}_n)$$
$$x_n \in \{0, 1\},\ \tilde{x}_n \in [-1, 1]$$
$$\min \mathcal{H} \leftrightarrow \mathrm{SAT}\ F$$



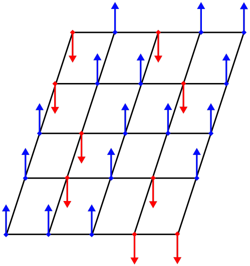
Hamiltonian Formulations: The Ising Model

Model of ferromagnetic metal behavior

- Each *physical* dipole is a "spin" $\sigma_i \in \pm 1$ on d-dimensional lattice
- Nearest-neighbor spins "couple" with strength $J > 0$
- "External" magnetic field h aligns the spins

System "Hamiltonian" $\mathcal{H}(\boldsymbol{\sigma})$:

$$\mathcal{H}(\boldsymbol{\sigma}) = -J \sum_{\langle i,j \rangle} \sigma_i \sigma_j - h \sum_i \sigma_i$$



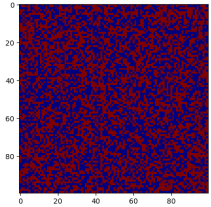


Figure: 100×100 2D lattice at fixed T [Hammel].

Hamiltonian Formulations: Generalized Spin Glasses

Models with "quenched disorder"

- Generalize $J \to \boldsymbol{J} = \{J_{ij}\} \in \mathbb{M}_{N\times N}$
- Arbitrary couplings
- Deep connections to SAT/Complexity Theory[1]

Finding "ground state" $\boldsymbol{\sigma}^* = \mathrm{argmin}_{\boldsymbol{\sigma}} \mathcal{H}(\boldsymbol{\sigma}) \to$ *NP-Hard* [Barahona, 1982, Lucas, 2014]

Suppose we know $\mathcal{H}(\boldsymbol{\sigma}^*) = 0$
$\to$ *NP-Complete*

[1] See [Percus et al., 2006]

Physically Realizing Disordered Spin Model

Goal: Implement Ising[a] Hamiltonians in a programmable environment

- Quantum Annealing [Albash and Lidar, 2018, Ushijima-Mwesigwa et al., 2017]
- Coherent Ising Machines (CIM) [Honjo et al., 2021]
- L-C Oscillator Ising Machines (OIM) [Wang and Roychowdhury, 2019]
- Bistable Resistively coupled Ising Machines (BRIM) [Afoakwa et al., 2021]

[a] Generalized spin glass

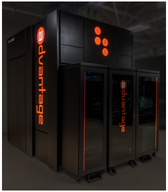


Figure: D-Wave Advantage [D-Wave]

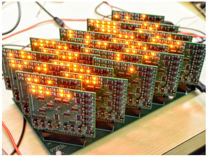


Figure: OIM [Wang et al., 2019]

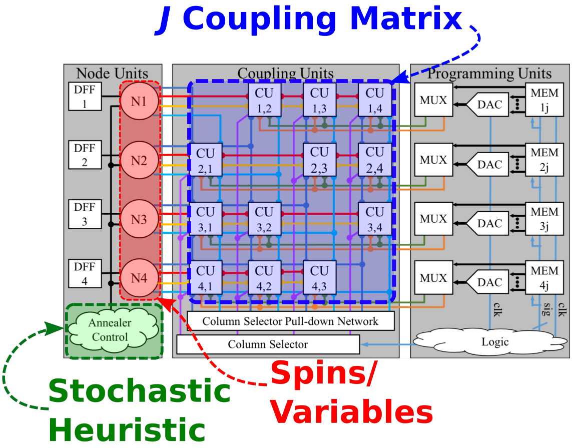


Figure: From [Afoakwa et al., 2021]

Quantized spin states (Clamp $\tilde{x}_n \rightarrow x_n$)
"Physical Stochastic Local Search Solver"

Beyond Quadratic Hamiltonians

Existing dynamical solvers limited to 2-spin interactions

What if the problem has higher degree terms?

Consider a k-SAT CNF

$$F = \bigwedge_{i}^{M} \left(\bigvee_{j}^{k} \ell_{ij} \right) \text{ where } \ell_{i,j} \in \{x_m, \overline{x_m}\}$$

"Natural" k-SAT Ising Hamiltonian for $\sigma \in \{-1, +1\}$:

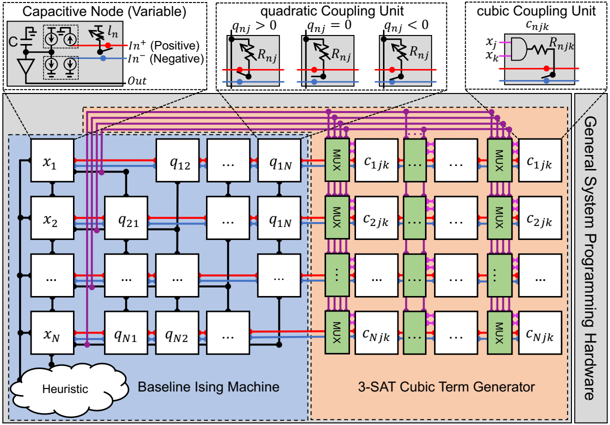
$$\mathcal{H}_{\mathrm{SAT}}(\boldsymbol{\sigma}) = \sum_{i}^{M} \prod_{j}^{k} g(\ell_{ij})$$

$$g(\ell_{ij}) = \begin{cases} \frac{1}{2}(1 - \sigma_m) & \ell_{ij} = x_m \\ \frac{1}{2}(1 + \sigma_m) & \ell_{ij} = \overline{x_m} \end{cases}$$

$$g_b(\ell_{ij}) = \begin{cases} 1 - x_m & \ell_{ij} = x_m \\ x_m & \ell_{ij} = \overline{x_m} \end{cases}$$

Quadratic reductions perform poorly [Gabor et al., 2019, Bashar and Shukla, 2022]

Hardware Support: "Cubic BRIM"(cBRIM)



AKA "High Order Interaction"

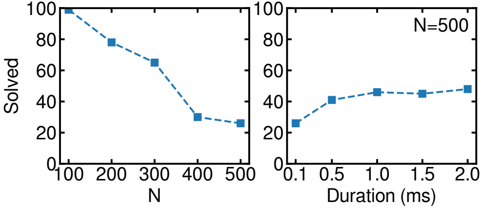
Similar models implemented in oscillator-based systems [Bashar and Shukla, 2022, Bashar et al., 2023]

Problem-Agnostic Heuristic Performance

Cubic BRIM (cBRIM) equations: Gradient descent with noise $\eta \sim \mathcal{N}(0, \sigma)$

$$\frac{d\tilde{x}_n}{dt} \propto -\frac{\partial \mathcal{H}_{\mathrm{C}}(\boldsymbol{x})}{\partial x_n} + \eta$$

- 500 Uniform Random 3-SAT problems [Ansótegui et al., 2009, Ansótegui et al., 2009]
  - N from $100 - 500$
  - $\alpha = 4.25$
- Decreasing solution quality
- Success rate plateau



Cubic interactions alone aren't enough!

"Problem Agnostic" Gaussian Noise $\rightarrow$ stochastic SAT heuristics

Focus on probSAT-like Make/Break distributions[Balint and Schöning, 2012]

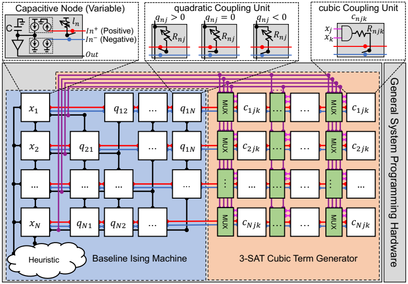
For assignment $a : \boldsymbol{x} \rightarrow \{0, 1\}^N$:

Definition (Make Count ($\mathscr{M}_n$(a)))

The number of *newly SAT* clauses from flipping $a(x_n) = \overline{a(x_n)}$

Definition (Break Count ($\mathscr{B}_n$(a)))

The number of *newly UNSAT* clauses from flipping $a(x_n) = \overline{a(x_n)}$



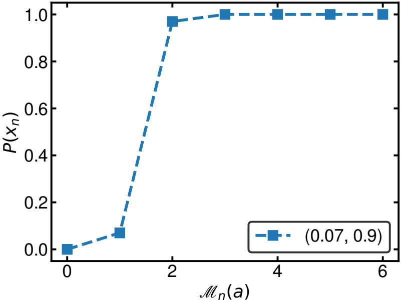
$\mathscr{M}_n$(a) $-$ $\mathscr{B}_n$(a) $\rightarrow$ Net change in SAT clauses from flipping x_n

Make Only Heuristics: Biased Random Walk

First proposal: "Nonlinear biased random walk" (BRW) using make counts [Shim et al., 2018]
Define constants $p_{init},\ p_{step} \in [0,1]$

$$P(x_n) = \begin{cases} \min(1,\ p_{init} + p_{step}(\mathscr{M}_n(\mathtt{a}) - 1)) & \mathscr{M}_n(\mathtt{a}) \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

- Parallel updates
  $x_n \sim \mathrm{Bernoulli}(P(x_n))$
- Additional analog hardware $\rightarrow$ recover variable make counts Shim et al. [2018]



Industrial-Like Problem Structure

Goal: industrial-like benchmarks. Focus on graph-like properties

Definition (Variable Degree [Friedrich et al., 2017])

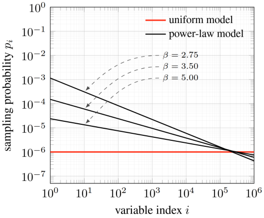
The *degree* of SAT variable x_i is the number of occurances of x_i in the formula

Definition (Degree Distribution [Friedrich et al., 2017])

The *degree distribution* $P(z)$ of a SAT instance gives the fraction of variables with degree z

Power Law distributions: $P(z) \propto z^{-\beta}$

Industrial instances exhibit Scale-Free (SF) structure ($\beta \in [2, 3]$)



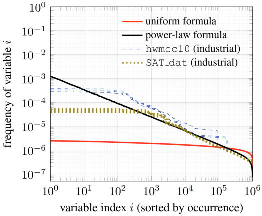
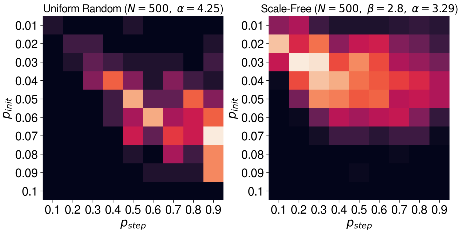
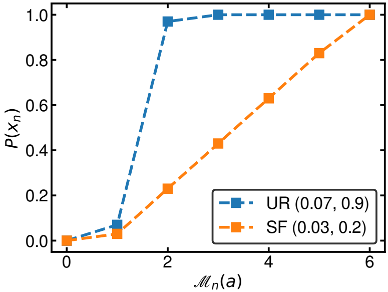


Figure: From [Friedrich et al., 2017]



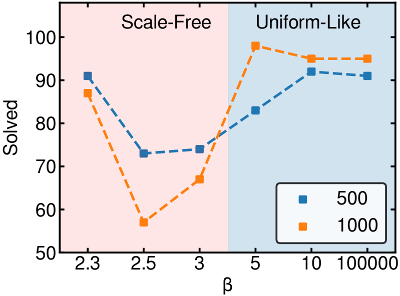
- Different degree distributions require *different functions*
- Tuned separately for UR and SF



Make-Only Performance

Previous BRW results limited to Uniform Random (UR) [Shim et al., 2018]
Performance on Scale-Free (SF)?

- Generated 1200 3-SAT problems [Friedrich et al., 2017, Ansótegui et al., 2009]
  - $N \in \{500, 1000\}$
  - β from $2.3 - 100000$
- ODE Simulation: Dynamical solver with discrete-time heuristics
- Poor SF performance for $\beta \in [2.5, 3]$, worse with $\uparrow N$



Recovering Break Counts

Conjecture: Considering $\mathscr{M}_n(a)$ alone insufficient
Inter-variable dependence→non-trival recovery
From dynamical system equations (with no noise):

$$\frac{d\tilde{x}_n}{dt} \propto -\frac{\partial \mathcal{H}_{\mathrm{C}}(\boldsymbol{x})}{\partial x_n}$$

Can derive:

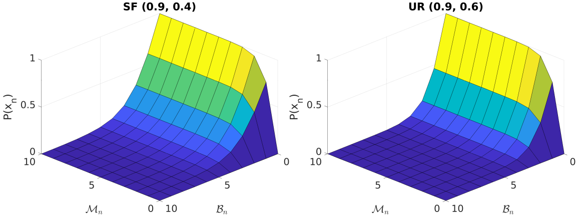
$$\frac{d\tilde{x}_n}{dt} \propto (1 - 2x_n) \times (\mathscr{M}_n(a) - \mathscr{B}_n(a))$$

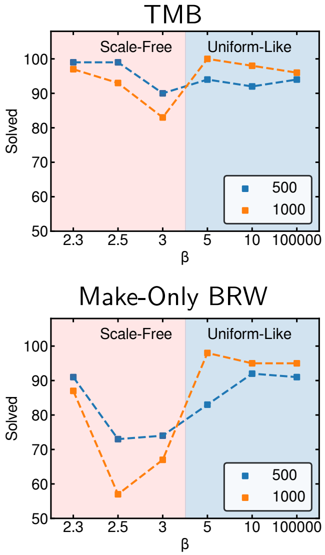
Dynamical model→Break counts recovery

"Tanh Make-Break" (TMB) Heuristic

$$P(x_n) = \tanh(c_m \mathscr{M}_n(a))(1 - \tanh(c_b \mathscr{B}_n(a)))$$

- Lack of range normalization
- Nonlinear threshold functions
- Ground state preservation
- Implementable in current-domain





Testing Methodology

Problem Suite: Generated 200 instances of 1000 variable problems

- 100 SF $\beta = 2.935$, $\alpha = 3.4$
- 100 UR $\alpha = 4.25$

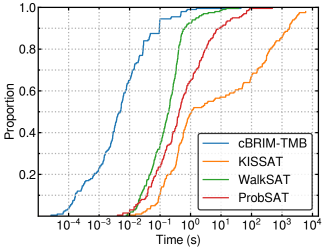
Solvers and Testing Configurations:

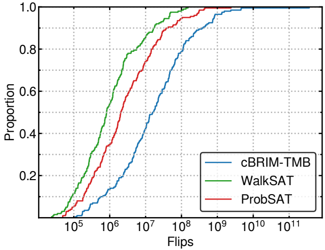
- WalkSAT: 20 restarts, 5 million flips per problem, default configuration ("best" heuristic)
- KISSAT, ProbSAT: 10,000 s timeout, default configuration
- cBRIM-TMB: 20 iterations per problem, 200 ms simulated timeout. Used previous c_m, c_b values

$$\text{Time/Flips to Solution (TTS/FTS)} = \begin{cases} Time/Flips & P(Success) \geq 0.99 \\ Time/Flips \cdot \frac{\log(1-0.99)}{\log(1-P(Success))} & otherwise \end{cases}$$

Classical Solver Comparison

- cBRIM-TMB/SLS solvers able to solve all instances
- KISSAT unable to solve 4 Uniform Random before timeout
- WalkSAT, ProbSAT more "flip-efficient" than cBRIM-TMB
  - cBRIM-TMB flip rate: ~ 3400 flips$/\mu s$
  - WalkSAT, ProbSAT flip rate: $\sim 5-6$ flips$/\mu s$
- **$\sim$ 500$\times$ Superior Flip Rate**
- **$>$ 25$\times$ Speedup vs WalkSAT**





Conclusions and Future Work

Conclusions:

- Augmented dynamical models promise large speedups over software
- Heuristic choice has a large impact on solution quality
- Problem structure is a necessary testing consideration

Future work:

- Hardware validation of results
- Broader heuristic testing/analysis
- Exceeding hardware limitations
- Larger (industrial) problem instances
- Application to other optimization/decision tasks

Thank You
Questions?

References I

[1] Richard Afoakwa, Yiqiao Zhang, Uday Kumar Reddy Vengalam, Zeljko Ignjatovic, and Michael Huang. BRIM: Bistable Resistively-Coupled Ising Machine. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 749–760, February 2021. doi: 10.1109/HPCA51647.2021.00068. ISSN: 2378-203X.

[2] Tameem Albash and Daniel A. Lidar. Adiabatic quantum computation. *Rev. Mod. Phys.*, 90:015002, Jan 2018. doi: 10.1103/RevModPhys.90.015002. URL https://link.aps.org/doi/10.1103/RevModPhys.90.015002.

[3] Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. On the structure of industrial SAT instances. In *Principles and Practice of Constraint Programming-CP 2009: 15th International Conference, CP 2009 Lisbon, Portugal, September 20-24, 2009 Proceedings 15*, pages 127–141. Springer, 2009.

[4] Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. Towards industrial-like random SAT instances. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, IJCAI'09, page 387–392, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[5] Adrian Balint and Uwe Schöning. Choosing probability distributions for stochastic local search and the role of make versus break. In *SAT*, 2012.

References II

[6] F. Barahona. On the computational complexity of Ising spin glass models. *Journal of Physics A: Mathematical and General*, 15(10):3241, October 1982. ISSN 0305-4470. doi: 10.1088/0305-4470/15/10/028. URL https://dx.doi.org/10.1088/0305-4470/15/10/028.

[7] Mohammad Khairul Bashar and Nikhil Shukla. Constructing dynamical systems to model higher order Ising spin interactions and their application in solving combinatorial optimization problems. *arXiv preprint arXiv:2211.05365*, 2022.

[8] Mohammad Khairul Bashar, Antik Mallick, Avik W. Ghosh, and Nikhil Shukla. Dynamical system-based computational models for solving combinatorial optimization on hypergraphs. *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, pages 1–1, 2023. doi: 10.1109/JXCDC.2023.3235113.

[9] Vicky Choi. Adiabatic quantum algorithms for the NP-complete maximum-weight independent set, exact cover and 3SAT problems. *arXiv*, April 2010.

[10] Media Resources | D-Wave. Media Resources | D-Wave. URL https://www.dwavesys.com/company/media-resources/.

References III

[11] John D. Davis, Zhangxi Tan, Fang Yu, and Lintao Zhang. A practical reconfigurable hardware accelerator for boolean satisfiability solvers. In *Proceedings of the 45th Annual Design Automation Conference*, DAC '08, page 780–785, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581156. doi: 10.1145/1391469.1391669. URL https://doi.org/10.1145/1391469.1391669.

[12] Tobias Friedrich, Anton Krohmer, Ralf Rothenberger, and Andrew Sutton. Phase transitions for scale-free sat formulas. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[13] Thomas Gabor, Sebastian Zielinski, Sebastian Feld, Christoph Roch, Christian Seidel, Florian Neukart, Isabella Galter, Wolfgang Mauerer, and Claudia Linnhoff-Popien. Assessing solution quality of 3SAT on a quantum annealing platform, 2019. URL https://arxiv.org/abs/1902.04703.

[14] Youssef Hamadi, Said Jabbour, and Lakhdar Sais. ManySAT: a Parallel SAT Solver. *Journal on Satisfiability, Boolean Modeling and Computation*, 6(4): 245–262, January 2010. doi: 10.3233/SAT190070. URL https://content.iospress.com/articles/journal-on-satisfiability-boolean-modeling-and-computation/sat190070. Publisher: IOS Press.

[15] Ben Hammel. bdhammel/ising-model: Python implementation of the Ising model. URL https://github.com/bdhammel/ising-model.

References IV

[16] Toshimori Honjo, Tomohiro Sonobe, Kensuke Inaba, Takahiro Inagaki, Takuya Ikuta, Yasuhiro Yamada, Takushi Kazama, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, Ken ichi Kawarabayashi, and Hiroki Takesue. 100,000-spin coherent ising machine. *Science Advances*, 7(40):eabh0952, 2021. doi: 10.1126/sciadv.abh0952. URL https://www.science.org/doi/abs/10.1126/sciadv.abh0952.

[17] Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2, 2014. ISSN 2296-424X. URL https://www.frontiersin.org/articles/10.3389/fphy.2014.00005.

[18] Muhammad Osama, Anton Wijs, and Armin Biere. Sat solving with gpu accelerated inprocessing. In *Tools and Algorithms for the Construction and Analysis of Systems: 27th International Conference, TACAS 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27–April 1, 2021, Proceedings, Part I 27*, pages 133–151. Springer, 2021.

[19] Allon Percus, Gabriel Istrate, and Cristopher Moore. *Computational complexity and statistical physics*. OUP USA, 2006.

References V

[20] John Shalf. The future of computing beyond Moore's Law. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 378(2166):20190061, January 2020. doi: 10.1098/rsta.2019.0061. URL https://royalsocietypublishing.org/doi/full/10.1098/rsta.2019.0061. Publisher: Royal Society.

[21] Yong Shim, Abhronil Sengupta, and Kaushik Roy. Biased random walk using stochastic switching of nanomagnets: Application to SAT solver. *IEEE Transactions on Electron Devices*, 65(4):1617–1624, 2018. doi: 10.1109/TED.2018.2808232.

[22] Ali Asgar Sohanghpurwala, Mohamed W. Hassan, and Peter M. Athanas. Hardware accelerated SAT solvers - a survey. *J. Parallel Distributed Comput.*, 106:170–184, 2017.

[23] Hayato Ushijima-Mwesigwa, Christian F. A. Negre, and Susan M. Mniszewski. Graph Partitioning using Quantum Annealing on the D-Wave System. In *Proceedings of the Second International Workshop on Post Moores Era Supercomputing*, PMES'17, pages 22–29, New York, NY, USA, November 2017. Association for Computing Machinery. ISBN 978-1-4503-5126-3. doi: 10.1145/3149526.3149531. URL https://dl.acm.org/doi/10.1145/3149526.3149531.

References VI

[24] Tianshi Wang and Jaijeet Roychowdhury. OIM: Oscillator-based Ising Machines for Solving Combinatorial Optimisation Problems, March 2019. URL http://arxiv.org/abs/1903.07163. arXiv:1903.07163 [cs].

[25] Tianshi Wang, Leon Wu, and Jaijeet Roychowdhury. New Computational Results and Hardware Prototypes for Oscillator-based Ising Machines. In *Proceedings of the 56th Annual Design Automation Conference 2019*, pages 1–2, Las Vegas NV USA, June 2019. ACM. ISBN 978-1-4503-6725-7. doi: 10.1145/3316781.3322473. URL https://dl.acm.org/doi/10.1145/3316781.3322473.