QIP 2021

COMPILATION OF FAULT-TOLERANT QUANTUM HEURISTICS FOR COMBINATORIAL OPTIMISATION

YUVAL.SANDERS@MQ.EDU.AU

Yuval Sanders

MACQUARIE UNI, SYDNEY, AU

QUADRATIC SPEEDUPS APPEAR

INSUFFICIENT

FOR EARLY QUANTUM COMPUTERS

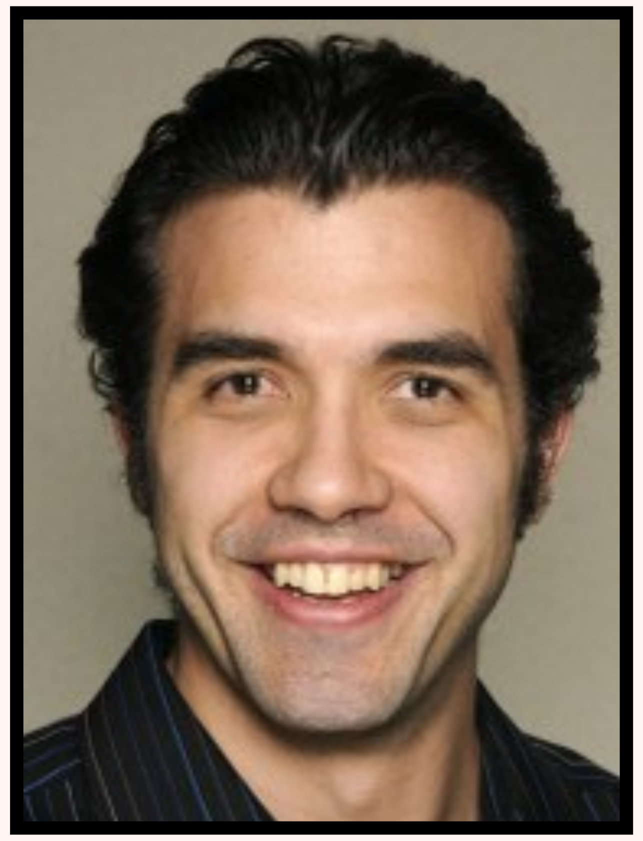
TO BEAT CLASSICAL

AT COMBINATORIAL OPTIMISATION

THE TEAM

Yuval Sanders (Macquarie)

Dominic Berry (Macquarie)



Pedro Costa (Macquarie)

Louis Tessler (Macquarie)

Nathan Wiebe (Toronto)

Craig Gidney (Google)

Hartmut Neven (Google)

Ryan Babbush (Google)

OVERVIEW

Aimed at people who haven't seen our work.

Motivation & Background

Approach & Results

QROM-based function evaluation

Conclusions & Future Work

Aimed at people who want a deep dive into our techniques.

MOTIVATION & BACKGROUND

COMBINATORIAL OPTIMISATION

- **Roughly:** Given diagonal $2^N \times 2^N$ Hamiltonian, find a ground state.
- **Key examples:** {travelling salesman, minimum spanning tree, knapsack} problem.
- In practice, expect to be satisfied with approach that **probably** returns **near-optimal** solution.
  - The travelling salesman does not need an optimal route, only a good one.
  - Prototypical classical method: simulated annealing, an **heuristic** approach.
- **Many practical applications:** logistics, supply-chain optimisation, water distribution, …
  - Natural place to look for useful quantum algorithms!

HAMILTONIAN FAMILIES

L-term spin model: $H_L = \sum_{\ell=1}^{L} w_\ell \prod_{i \in q_\ell} Z_i$ ($w_\ell \in \mathbb{R}$, $q_\ell \subseteq \{1,\ldots,N\}$, $\ell \mapsto q_\ell$ is injective)

Quadratic Unconstrained Binary Optimisation (QUBO): $H_{\text{QUBO}} = \sum_{i \leq j} w_{ij} \left(I - Z_i\right)\left(I - Z_j\right)$ (NP-Hard subproblem)

Sherrington-Kirkpatrick (SK): $H_{\text{SK}} = \sum_{i<j} w_{ij} Z_i Z_j$, $w_{ij} = \pm 1$ (subproblem of QUBO)

Low Autocorrelation Binary Sequences (LABS): $H_{\text{LABS}} = \sum_{k=0}^{N-1} \left(\sum_{i=1}^{N-k} Z_i Z_{i+k} \right)^2$

(hard subproblem of N^3-term spin model; best classical algorithm is $\Theta(1.73^N)$ and only solved for $N \leq 66$)

QUANTUM HEURISTICS

- **Direct amplitude amplification:** Grover-like approach, assume known degeneracy and a threshold energy value.
  - Dumb (treats problem as unstructured), but good reference point.
- **Quantum Approximate Optimisation Algorithm (QAOA):** alternate between evolution under problem Hamiltonian and "driver" Hamiltonian, which superposes solutions.
  - More asymptotically efficient classical approach known [Barak et al., arXiv:1505.03424], but maybe QAOA has better constant factors and can be executed on NISQ hardware.
- **Adiabatic algorithms and quantum-simulated annealing:** 
  - Entire textbooks could be written on this. Potentially fruitful research direction!
  - We focus on compiling ideas already found in the literature. Ongoing work to develop better methods.

APPROACH & RESULTS

COMPILATION STRATEGY

- Our results derive from **exhaustive compilation** of these quantum heuristics.
- Compilation consists of three steps:
  1. Design "oracle" for the cost function (encode result into register or as phase).
  2. Design quantum circuit that uses oracle to perform single step of heuristic.
  3. Estimate runtime on surface code quantum computer by estimating # Toffolis.
- Our results therefore tell us **how many heuristic steps can be performed per unit time.**
  They do **not** tell us how many steps are needed to achieve approximate success with some probability.
  Therefore, we cannot directly compare the results for different heuristics. Our results are **performance indicators**.

SAMPLE OF RESULTS

ಠ_ಠ

cost function	algorithm primitive	Toffoli (* or T) count	total ancilla qubits
L-term	amplitude amplification step	$2\,L\,b_{\rm dir} + N + \mathcal{O}(b_{\rm dir})$	$2\,b_{\rm dir} + \mathcal{O}(1)$
Spin	QAOA/Trotter step*	$1.15L(b_{\rm pha} + \log L) + \mathcal{O}(N + \log L + b_{\rm pha}^2)$	$3\log N + b_{\rm pha} + \mathcal{O}(\log b_{\rm pha})$
Model	Hamiltonian walk step	$3\,L + 2b_{\rm LCU} + \mathcal{O}(\log L)$	$3\log L + 2\,b_{\rm LCU} + \mathcal{O}(1)$
H_L	Szegedy walk annealing step	$2(N+1)Lb_{\rm dir} + 2N(b_{\rm sm}^2 + b_{\rm dif} + \log N) + \mathcal{O}(Nb_{\rm sm}\log b_{\rm sm})$	$Nb_{\rm dif} + 2Nb_{\rm sm} + \mathcal{O}(N\log b_{\rm sm})$
	LHPST walk annealing step	$4Lb_{\rm dif} + 2(b_{\rm sm} + b_{\rm fun})^2 + 2b_{\rm dif} + N + 9\log N + \mathcal{O}(b_{\rm sm}\log b_{\rm sm})$	$3b_{\rm sm} + 2b_{\rm dif} + b_{\rm fun} + \log N + \mathcal{O}(\log b_{\rm sm})$
	gap amplified walk step	$4Lb_{\rm dif} + 2b_{\rm sm}^2 + 2b_{\rm dif} + N + 14\log N + \mathcal{O}(b_{\rm rot})$	$3b_{\rm sm} + 2b_{\rm dif} + 2\log N + \mathcal{O}(\log b_{\rm sm})$
Quadratic	amplitude amplification	$N^2 b_{\rm dir} + \mathcal{O}(Nb_{\rm dir})$	$2\,b_{\rm dir} + \mathcal{O}(1)$
Unconstrained	QAOA/Trotter step*	$0.575\,N^2(b_{\rm pha} + 2\log N) + \mathcal{O}(N^2)$	$3\log N + b_{\rm pha} + \mathcal{O}(\log b_{\rm pha})$
Binary	Hamiltonian walk step	$N(b_{\rm LCU} + 2\log N) + \mathcal{O}(N)$	$7\log N + 2\,b_{\rm LCU} + \mathcal{O}(\log b_{\rm LCU})$
Optimization	Szegedy walk annealing step	$2N^2 b_{\rm dif} + 2N(b_{\rm sm}^2 + b_{\rm dif} + \log N) + \mathcal{O}(Nb_{\rm sm}\log b_{\rm sm})$	$Nb_{\rm dif} + 2Nb_{\rm sm} + \mathcal{O}(N\log b_{\rm sm})$
$H_{\rm QUBO}$	LHPST walk annealing step	$2Nb_{\rm dif} + 2(b_{\rm sm} + b_{\rm fun})^2 + 2b_{\rm dif} + N + 9\log N + \mathcal{O}(b_{\rm sm}\log b_{\rm sm})$	$3b_{\rm sm} + 2b_{\rm dif} + b_{\rm fun} + \log N + \mathcal{O}(\log b_{\rm sm})$
	gap amplified walk step	$2Nb_{\rm dif} + 2b_{\rm sm}^2 + 2b_{\rm dif} + N + 14\log N + \mathcal{O}(b_{\rm rot})$	$3b_{\rm sm} + 2b_{\rm dif} + 2\log N + \mathcal{O}(\log b_{\rm sm})$
Sherrington–	amplitude amplification step	$2\,N^2 + N + \mathcal{O}(\log N)$	$6\log N + \mathcal{O}(1)$
Kirkpatrick	QAOA/Trotter step	$2N^2 + 4N + b_{\rm pha}^2 + \mathcal{O}(b_{\rm pha}\log b_{\rm pha})$	$6\log N + b_{\rm pha} + \mathcal{O}(\log b_{\rm pha})$
Model	Hamiltonian walk step	$6\,N + \mathcal{O}(\log^2 N)$	$5\log N + \mathcal{O}(1)$
$H_{\rm SK}$	Szegedy walk annealing step	$4N^2 + 2N(b_{\rm sm}^2 + 2\log N) + 8Nb_{\rm sm} + 18b_{\rm sm}^2 + \mathcal{O}(Nb_{\rm sm}\log b_{\rm sm})$	$N\log N + 2Nb_{\rm sm} + \mathcal{O}(N\log b_{\rm sm})$
	LHPST walk annealing step	$5N + 2(b_{\rm sm} + b_{\rm fun})^2 + 11\log N + \mathcal{O}(b_{\rm sm}\log b_{\rm sm})$	$4\log N + 3b_{\rm sm} + b_{\rm fun} + \mathcal{O}(\log b_{\rm sm})$
	gap amplified walk step	$5N + 2b_{\rm sm}^2 + 16\log N + \mathcal{O}(b_{\rm rot})$	$5\log N + 3b_{\rm sm} + \mathcal{O}(\log b_{\rm sm})$
Low	amplitude amplification step	$5\,N(N+1)/2 + N + \mathcal{O}(\log N)$	$5\log N + \mathcal{O}(1)$
Autocorrelation	QAOA/Trotter step	$8N^2/5 + \min\left(Nb_{\rm pha}^2/2, 9N^2/10\right) + \mathcal{O}(Nb_{\rm pha}\log b_{\rm pha})$	$5\log N + b_{\rm pha} + \mathcal{O}(\log b_{\rm pha})$
Binary	Hamiltonian walk step	$4\,N + \mathcal{O}(\log N)$	$5\log N + \mathcal{O}(1)$
Sequences	Szegedy walk annealing step	$5N(N+1)^2/2 + 2N(b_{\rm sm}^2 + 3\log N) + \mathcal{O}(Nb_{\rm sm}\log b_{\rm sm})$	$2N\log N + 2Nb_{\rm sm} + \mathcal{O}(N\log b_{\rm sm})$
$H_{\rm LABS}$	LHPST walk annealing step	$5N^2 + 2(b_{\rm sm} + b_{\rm fun})^2 + 6N + 13\log N + \mathcal{O}(b_{\rm sm}\log b_{\rm sm})$	$6\log N + 3b_{\rm sm} + b_{\rm fun} + \mathcal{O}(\log b_{\rm sm})$
	gap amplified walk step	$5N^2 + 2b_{\rm sm}^2 + 6N + 18\log N + \mathcal{O}(b_{\rm rot})$	$7\log N + 3b_{\rm sm} + \mathcal{O}(\log b_{\rm sm})$

				one hour runtime		one day runtime	
algorithm applied to LABS problem	problem size, N	logical qubits	Toffolis per step	maximum steps	physical qubits	maximum steps	physical qubits
amplitude amplification	64	98	9.8×10^3	2.1×10^3	3.0×10^5 (1.8×10^5)	5.1×10^4	3.6×10^5 (2.0×10^5)
	128	167	3.7×10^4	5.6×10^2	4.1×10^5 (2.1×10^5)	1.3×10^4	5.1×10^5 (2.3×10^5)
	256	300	1.5×10^5	1.4×10^2	7.1×10^5 (3.0×10^5)	3.3×10^3	8.0×10^5 (3.0×10^5)
	512	561	6.1×10^5	3.4×10^1	1.2×10^6 (4.3×10^5)	8.2×10^2	1.4×10^6 (4.3×10^5)
	1024	1078	2.3×10^6	9.0×10^0	2.2×10^6 (6.9×10^5)	2.2×10^2	2.9×10^6 (8.8×10^5)
QAOA / 1st order Trotter e.g., for population transfer or adiabatic algorithm	64	114	1.0×10^4	2.1×10^3	3.3×10^5 (1.9×10^5)	5.0×10^4	4.0×10^5 (2.1×10^5)
	128	183	3.8×10^4	5.5×10^2	4.4×10^5 (2.1×10^5)	1.3×10^4	5.5×10^5 (2.4×10^5)
	256	316	1.5×10^5	1.4×10^2	7.4×10^5 (3.1×10^5)	3.4×10^3	8.4×10^5 (3.1×10^5)
	512	577	5.0×10^5	4.2×10^1	1.2×10^6 (4.4×10^5)	1.0×10^3	1.4×10^6 (4.4×10^5)
	1024	1094	1.7×10^6	1.2×10^1	2.2×10^6 (7.0×10^5)	2.9×10^2	2.9×10^6 (8.9×10^5)
Hamiltonian walk e.g., for population transfer or adiabatic algorithm	64	94	2.6×10^2	8.1×10^4	3.0×10^5 (1.8×10^5)	2.0×10^6	3.5×10^5 (2.0×10^5)
	128	163	5.1×10^2	4.1×10^4	4.1×10^5 (2.1×10^5)	9.8×10^5	5.0×10^5 (2.3×10^5)
	256	296	1.0×10^3	2.0×10^4	7.0×10^5 (3.0×10^5)	4.9×10^5	8.0×10^5 (3.0×10^5)
	512	557	2.0×10^3	1.0×10^4	1.2×10^6 (4.3×10^5)	2.4×10^5	1.4×10^6 (4.3×10^5)
	1024	1074	4.1×10^3	5.1×10^3	2.2×10^6 (6.9×10^5)	1.2×10^5	2.9×10^6 (8.7×10^5)
LHPST walk quantum simulated annealing	64	132	2.0×10^4	1.0×10^3	3.6×10^5 (2.0×10^5)	2.5×10^4	4.4×10^5 (2.1×10^5)
	128	202	7.5×10^4	2.8×10^2	5.3×10^5 (2.5×10^5)	6.7×10^3	5.9×10^5 (2.5×10^5)
	256	336	3.0×10^5	6.9×10^1	7.8×10^5 (3.2×10^5)	1.7×10^3	8.8×10^5 (3.2×10^5)
	512	598	1.2×10^6	1.7×10^1	1.3×10^6 (4.5×10^5)	4.1×10^2	1.5×10^6 (4.5×10^5)
	1024	1116	4.6×10^6	5.0×10^0	2.2×10^6 (7.1×10^5)	1.1×10^2	3.0×10^6 (9.0×10^5)
spectral gap amplified walk based quantum simulated annealing	64	131	2.0×10^4	1.1×10^3	3.6×10^5 (2.0×10^5)	2.5×10^4	4.3×10^5 (2.1×10^5)
	128	202	7.5×10^4	2.8×10^2	5.3×10^5 (2.5×10^5)	6.7×10^3	5.9×10^5 (2.5×10^5)
	256	337	3.0×10^5	6.9×10^1	7.8×10^5 (3.2×10^5)	1.7×10^3	8.8×10^5 (3.2×10^5)
	512	600	1.2×10^6	1.7×10^1	1.3×10^6 (4.5×10^5)	4.1×10^2	1.5×10^6 (4.5×10^5)
	1024	1119	4.6×10^6	5.0×10^0	2.2×10^6 (7.2×10^5)	1.1×10^2	3.0×10^6 (9.0×10^5)

SIMPLIFIED RESULTS

Problem	Algorithm Primitive	(Table VIII and Table IX)		(Table VII)	
		steps per day	physical qubits	Toffoli count	
SK	Amplitude Amplification (§ III A)	4.8×10^3	8.1×10^5	$2N^2 + N$	$+\mathcal{O}(\log N)$
	QAOA / 1st order Trotter (§ III B)	4.7×10^3	8.6×10^5	$2N^2 + 4N$	$+\mathcal{O}(1)$
	Hamiltonian Walk (§ III C)	3.3×10^5	8.0×10^5	$6N$	$+\mathcal{O}(\log^2 N)$
	QSA / Qubitized (§ III E)	3.3×10^5	8.4×10^5	$5N$	$+\mathcal{O}(\log N)$
	QSA / Gap Amplification (§ III F)	3.9×10^5	8.4×10^5	$5N$	$+\mathcal{O}(\log N)$
LABS	Amplitude Amplification (§ III A)	3.3×10^3	8.0×10^5	$5N^2/2 + 7N/2$	$+\mathcal{O}(\log N)$
	QAOA / 1st order Trotter (§ III B)	3.4×10^3	8.4×10^5	$5N^2/2$	$+\mathcal{O}(N)$
	Hamiltonian Walk (§ III C)	4.9×10^5	8.0×10^5	$4N$	$+\mathcal{O}(\log N)$
	QSA / Qubitized (§ III E)	1.7×10^3	8.8×10^5	$5N^2$	$+\mathcal{O}(N)$
	QSA / Gap Amplification (§ III F)	1.7×10^3	8.8×10^5	$5N^2$	$+\mathcal{O}(N)$

- Number of bits of precision treated as unspecified constant in the rightmost column.
- Boxed numbers can be (sort of) compared to classical simulated annealing via Metropolis-Hastings. My laptop beats these numbers by **at least two orders of magnitude** with no code optimisation.

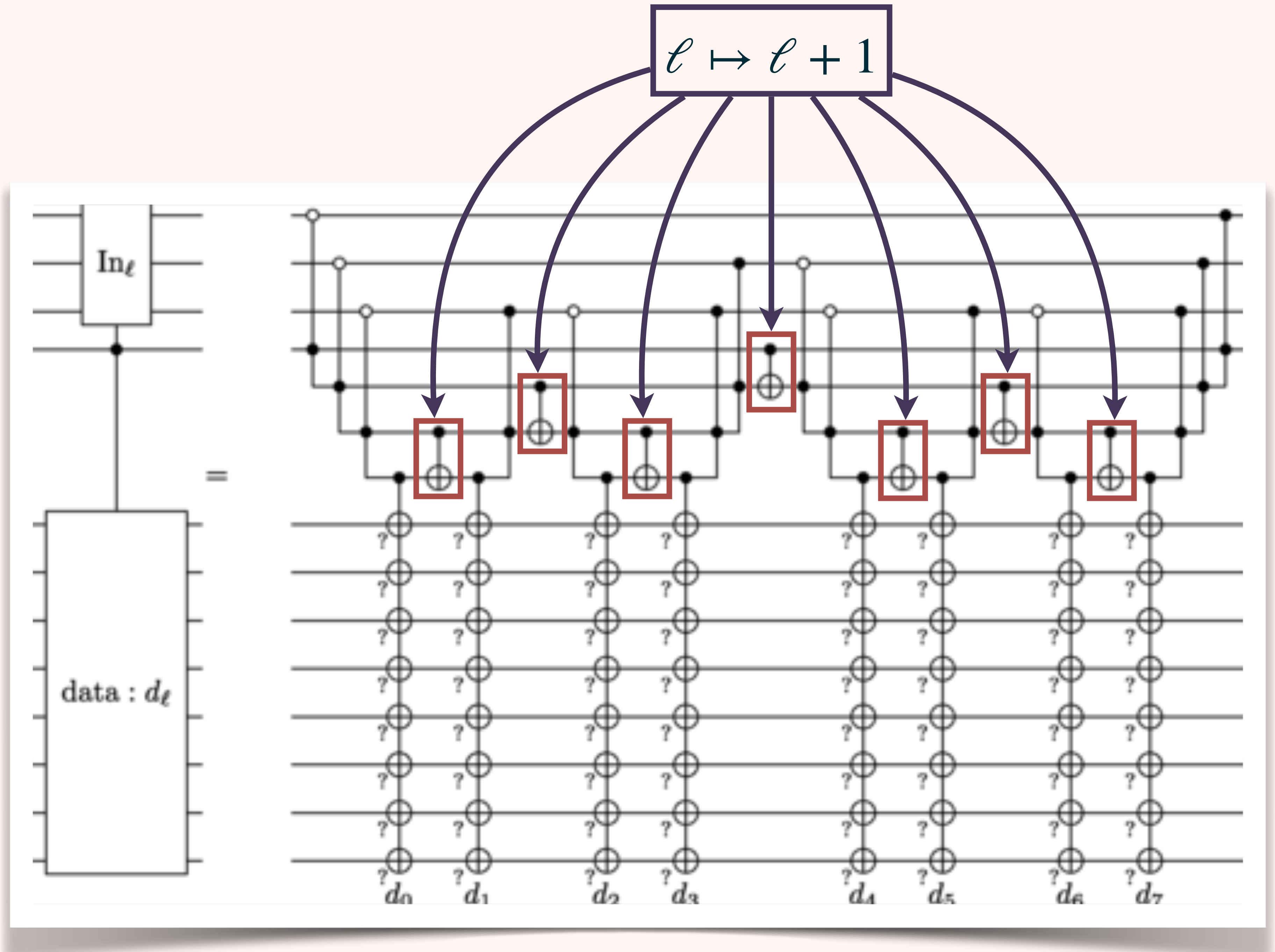
QROM-BASED FUNCTION EVALUATION

MOTIVATION

- Heuristic-based optimisation is bottlenecked by calculating functions of energy.
- But our approach is **heuristic**, meaning we do not need accurate output.
- We therefore want function approximation that trades accuracy for speed.
- Our approach is based on linear interpolation between lookup points.
  - This is a **very general technique** that will be useful for other numerical algorithms.

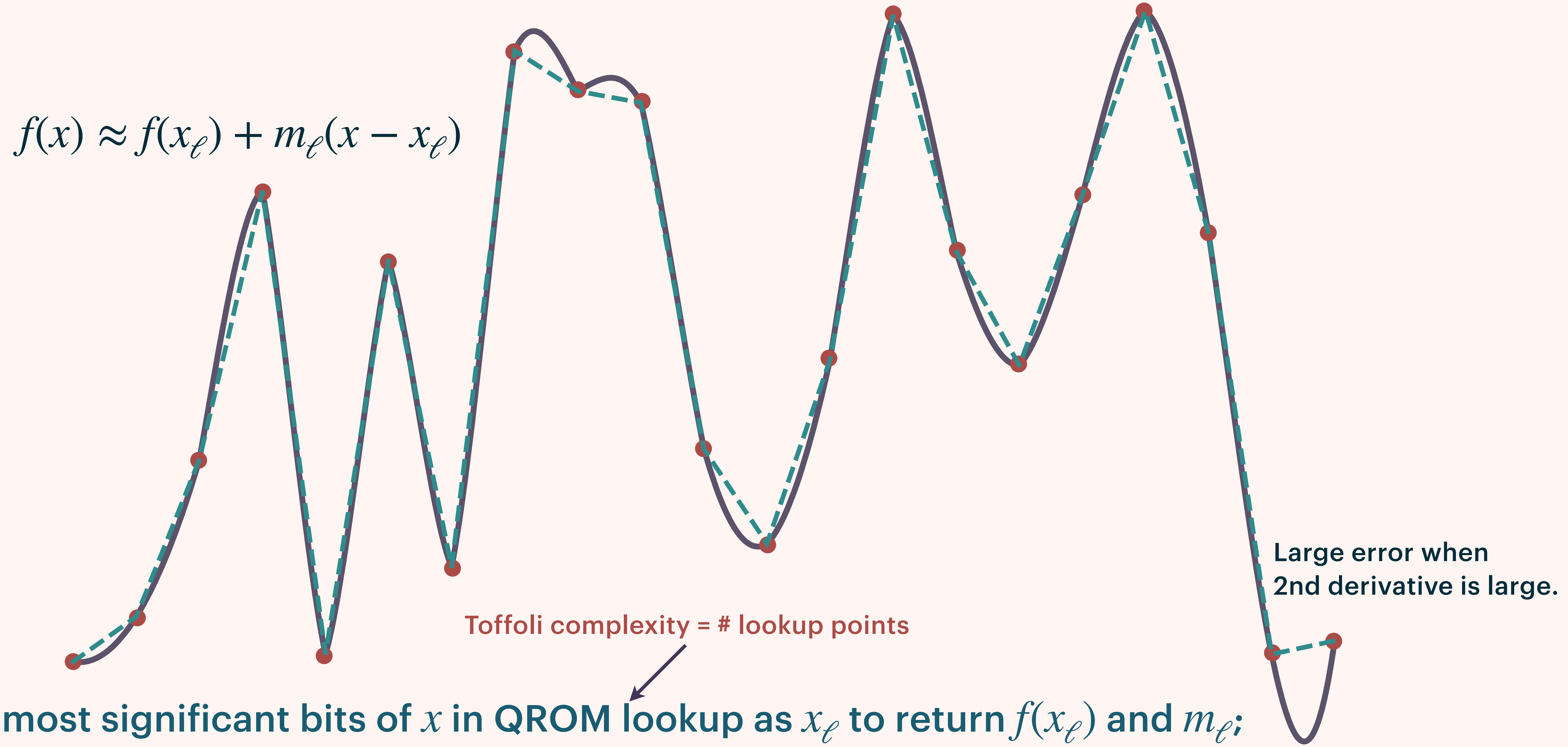
QROM LOOKUP

- **QROM:** a quantum data structure
- Return d_ℓ given classical integer input $\ell = 0,1,\ldots,L-1$
  (costs $\mathcal{O}(L)$ Toffolis; no dependence on size of d_ℓ)
- Can query with arbitrary superposition:
  $$\sum_\ell \alpha_\ell |\ell\rangle|0\rangle \to \sum_\ell \alpha_\ell |\ell\rangle|d_\ell\rangle$$
- **Main trick:** clever iteration through possible inputs.



doi:10.1103/PhysRevX.8.041015

LOOKUP & INTERPOLATION



Idea:

1. use most significant bits of x in QROM lookup as x_ℓ to return $f(x_\ell)$ and m_ℓ;
2. multiply m_ℓ to least significant bits of x (i.e. those of $x - x_\ell$); and
3. add result to $f(x_\ell)$.

THE FUNCTIONS WE NEED

Needed in qubitised Metropolis-Hastings approach to QSA by Lemieux et al. [doi:10.22331/q-2020-06-29-287]

$\arcsin(\exp(-x))$ →

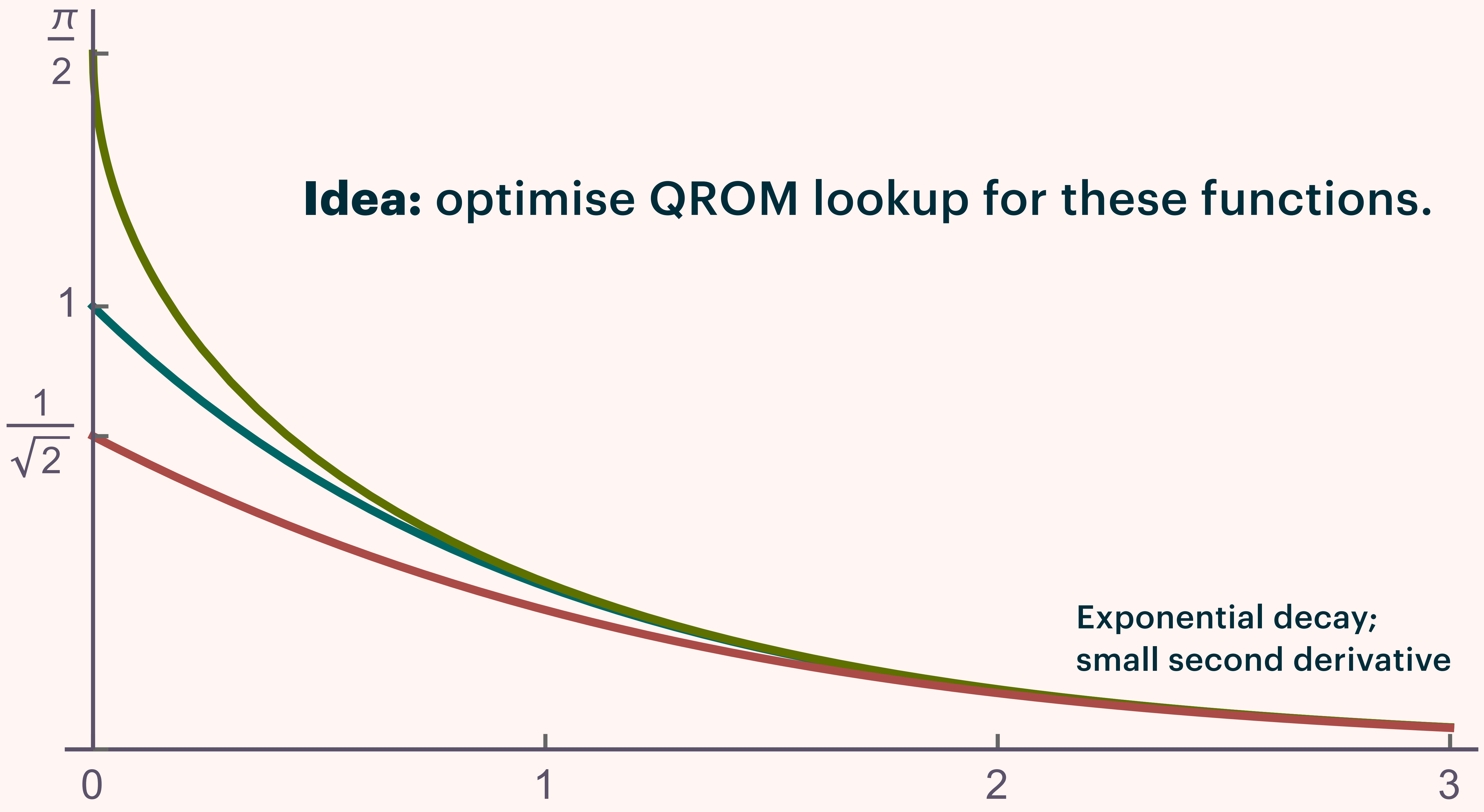
Needed in direct Szegedy walk approach of Somma et al. [doi:10.1103/PhysRevLett.101.130504]

$\exp(-x)$ →

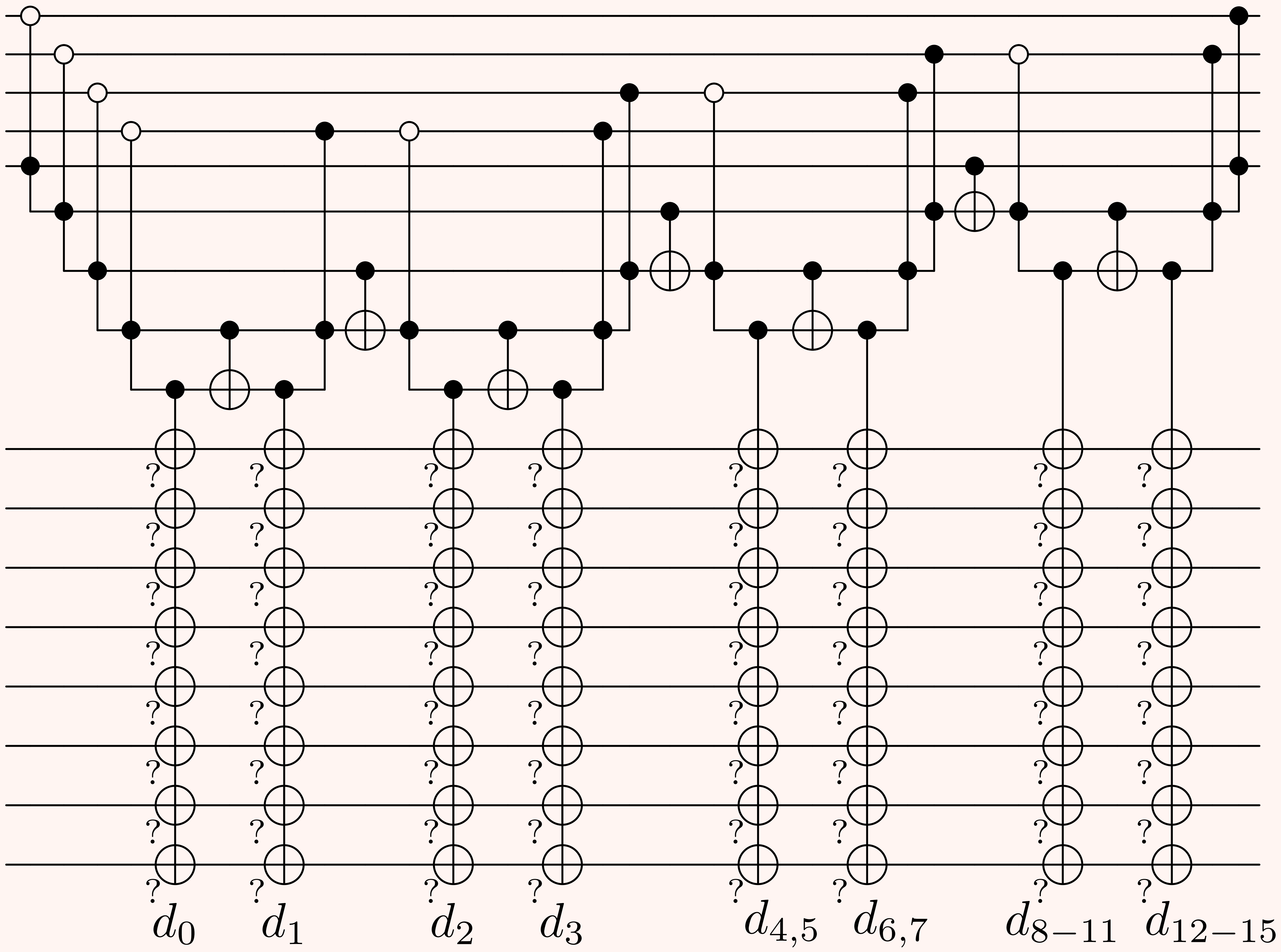
$\frac{\exp(-x)}{\sqrt{1+\exp(-x)}}$ →

Needed in simulation of spectral-gap-amplified Hamiltonian of Boixo et al. [doi:10.1140/epjst/e2015-02341-5]

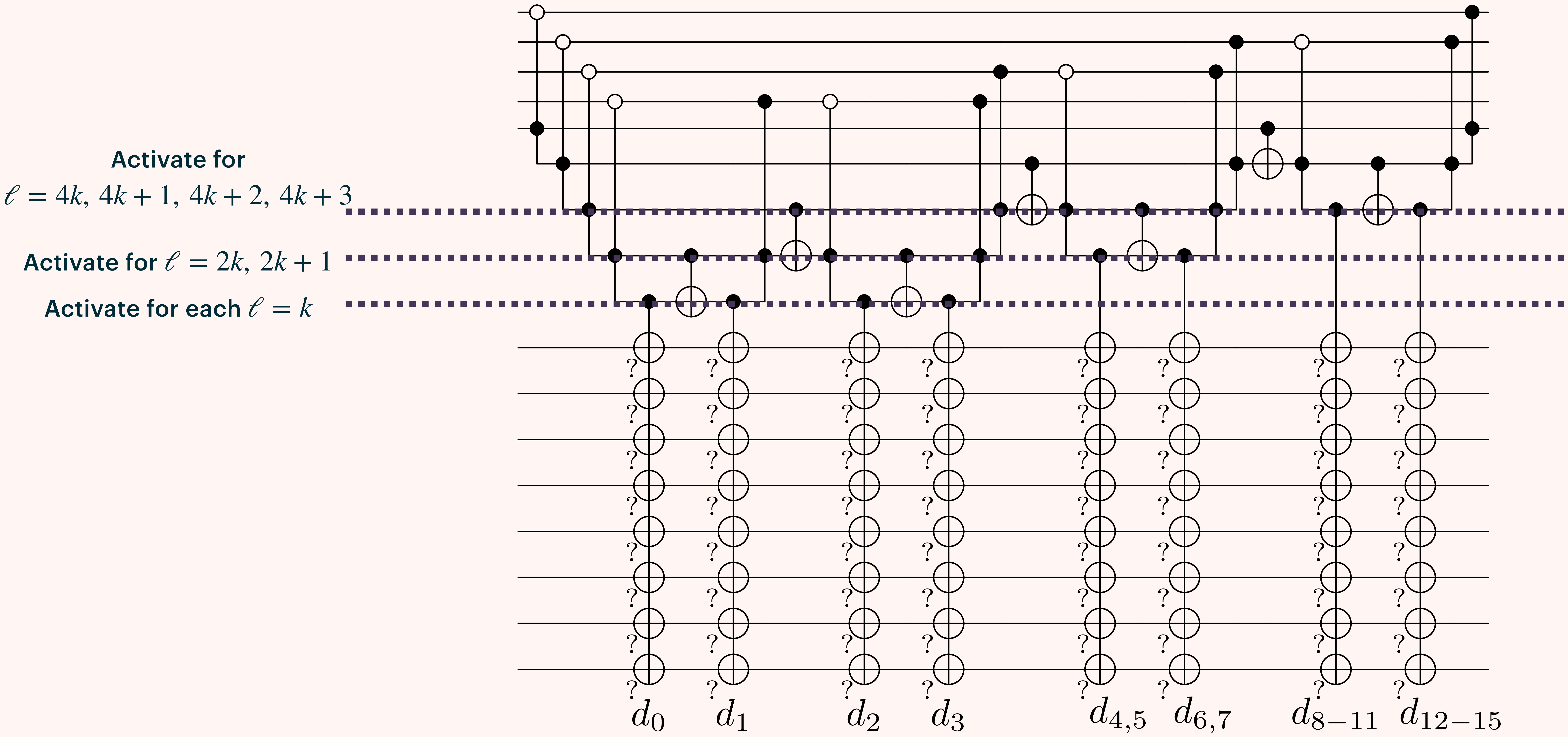
Idea: optimise QROM lookup for these functions.



EXPONENTIALLY-SPACED QROM LOOKUP



EXPONENTIALLY-SPACED QROM LOOKUP



CONCLUSIONS & FUTURE WORK

- Quadratic speedups for heuristic-based optimisation **probably not enough** for early quantum computers to beat even my laptop.
- Numbers mostly follow from overhead due to fault-tolerant architecture.
    - **Possible responses:** more efficient fault-tolerance methods, less hardware noise.
    - … or give up on quantum computer use cases involving quadratic speedups.
- **Some hope remains** for better-than-quadratic speedups for optimisation, but needs deeper research.
    - We attempted to "chain together" quadratic speedups but this is not easy to do effectively.
- Need more careful analysis of relative merits of different quantum heuristics.
    - And need more thorough comparison between classical- and quantum-simulated annealing.

FURTHER READING

- **Our full paper:** arXiv:2007.07391, doi:10.1103/PRXQuantum.1.020312
- More from Google about limitations of quadratic speedups: arXiv:2011.04149
- Similar conclusions on constraint satisfaction: Campbell, Khurana, & Montanaro, arXiv:1810.05582, doi:10.22331/q-2019-07-18-167
- Any good textbook on simulated annealing!
    - Lots of opportunity for follow-up work that is more sophisticated than ours.