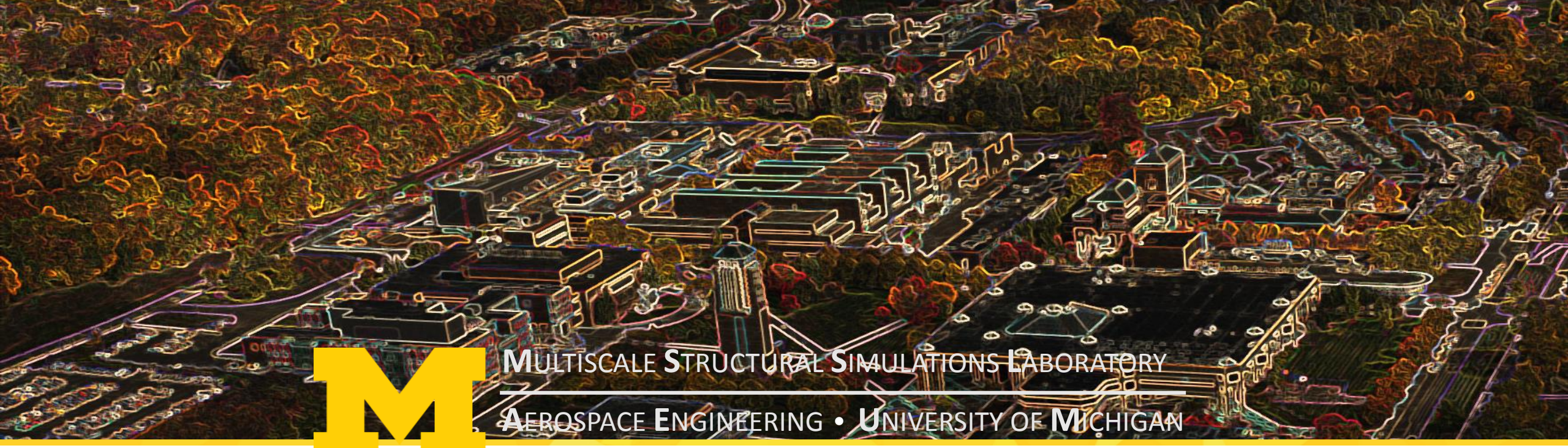


Solving differential equations on quantum computers

Prof. Veera Sundararaghavan
Department of Aerospace Engineering, University of Michigan

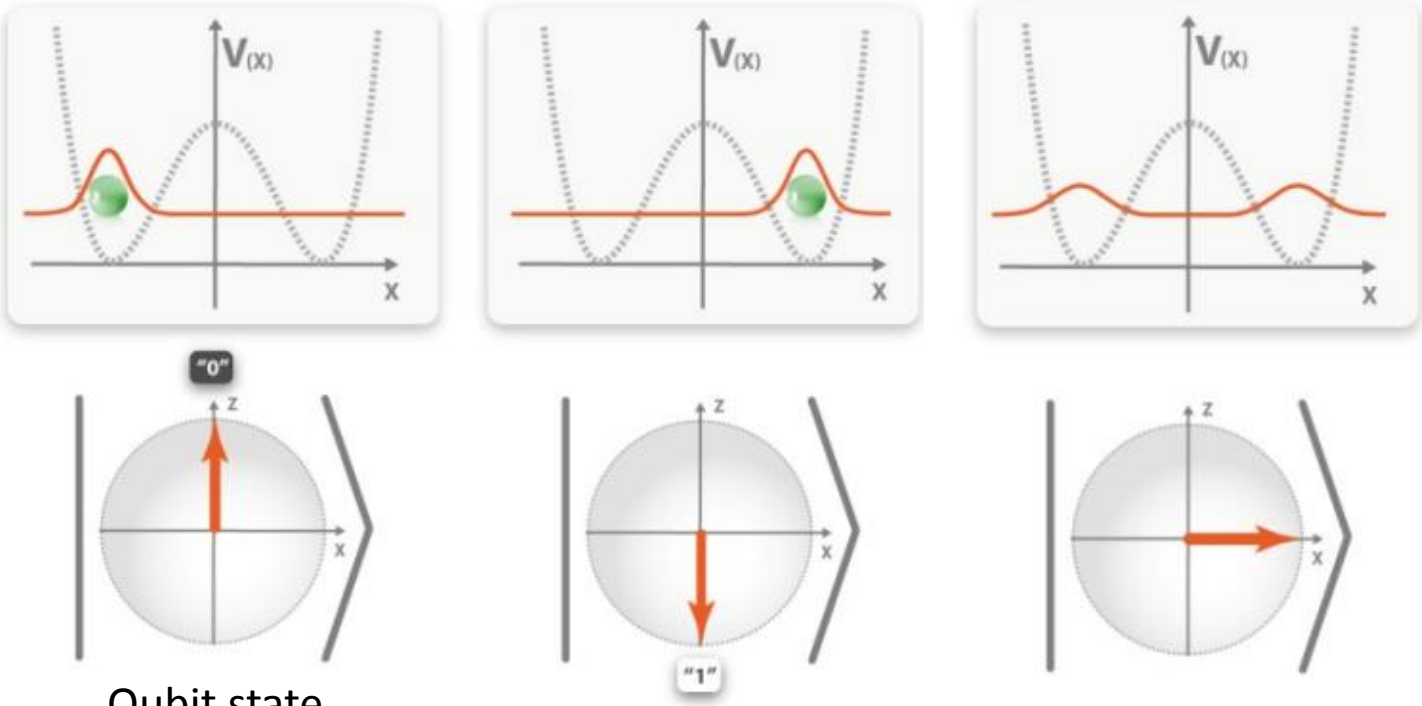
Sid Srivastava (PhD candidate)

Keynote Talk: Modeling and Computation session
16th Pan-American Congress of Applied Mechanics

May 23, 2019

Acknowledgments: USRA Quantum information Sciences Program

What is Quantum computing?



Qubit state

Quantum computers employ superposition of states

Dur and Heusler, Arxiv 2013

"It's not just a question of moving more quickly. It's a question of moving in different ways."

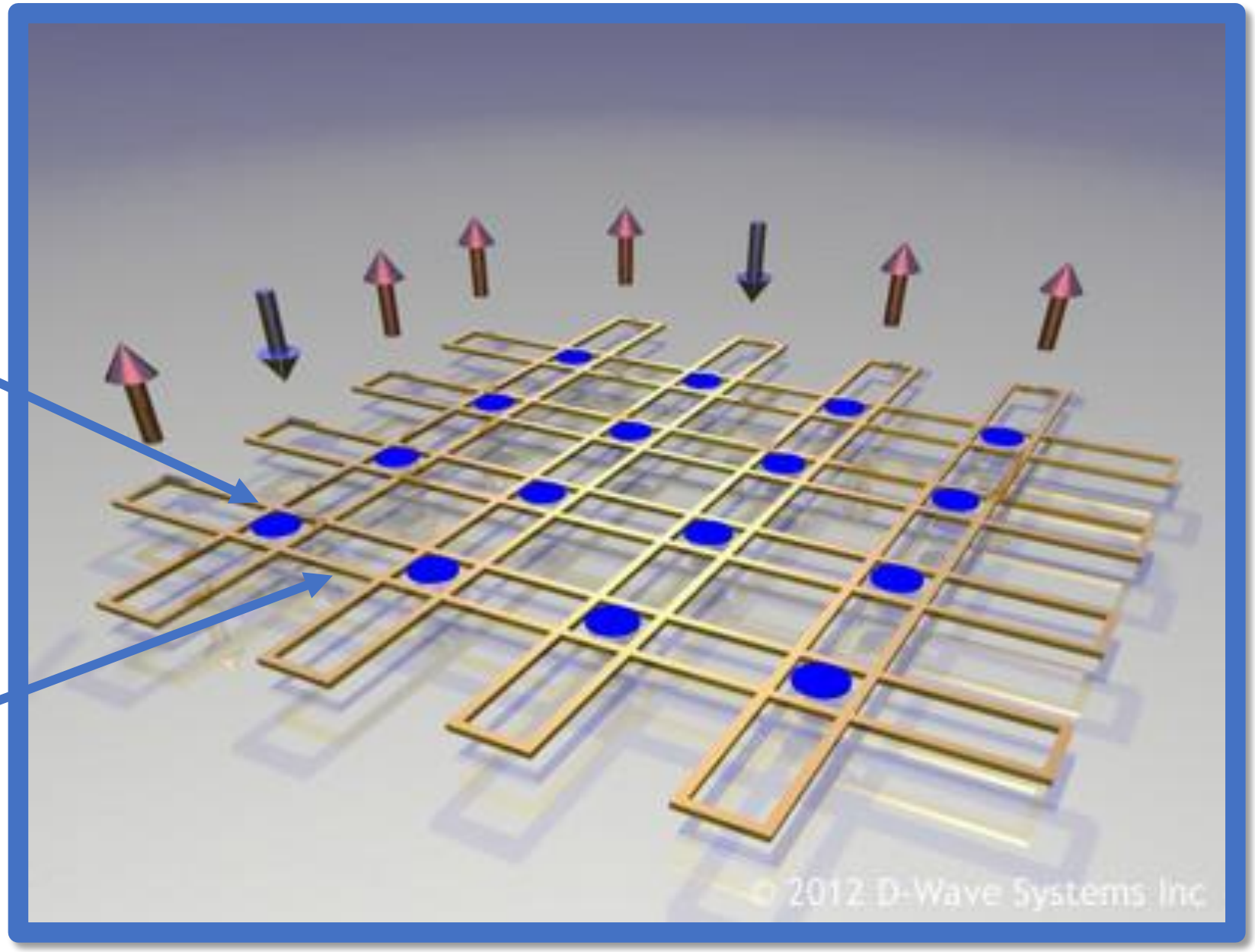
"It's as though you're Houdini trying to pick a lock and escape from an underwater cabinet. If you were free to move your hands wherever you'd like, you could do so much more efficiently than if you were handcuffed."

Aephraim Steinberg
Professor of physics at the University of Toronto and Centre for Quantum Information and Quantum Control

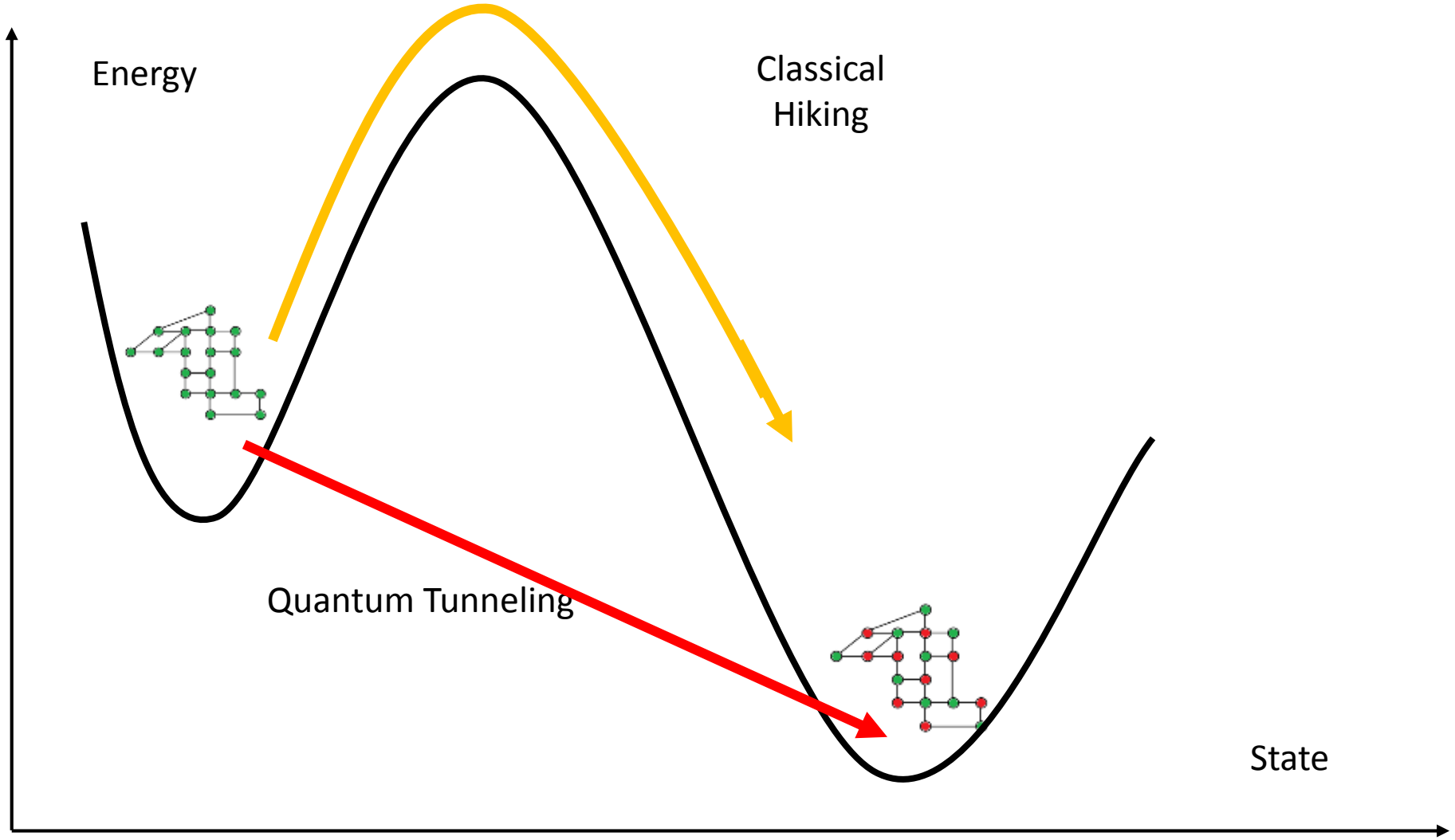
The Quantum annealer

Tunable field on the qubit

Tunable interaction between qubits



Optimization using quantum annealers



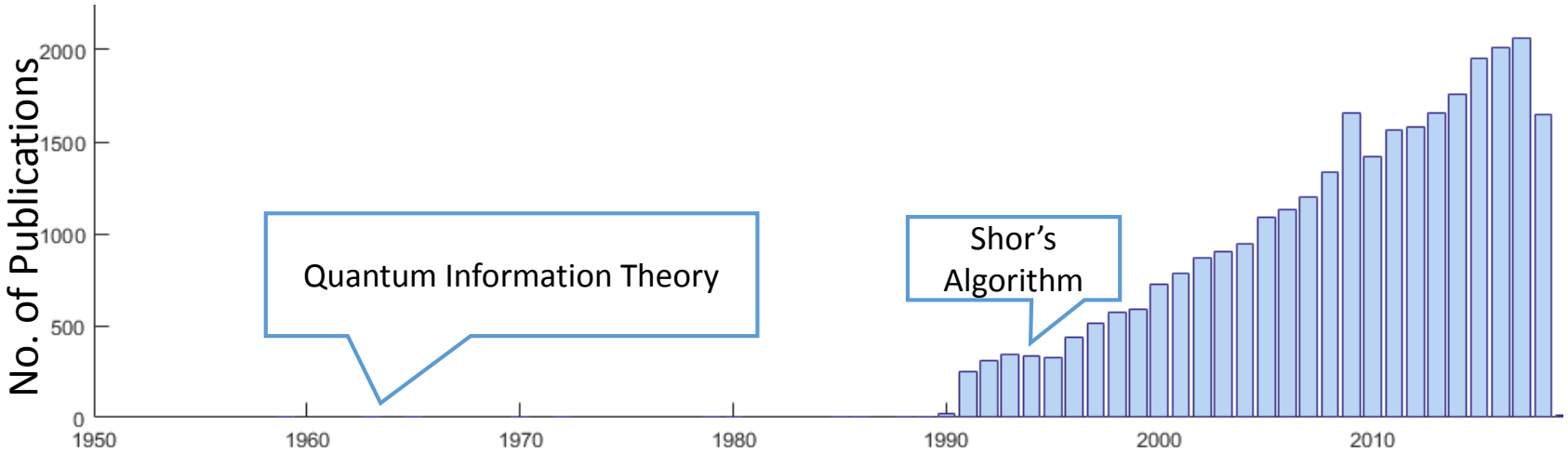
Algorithms were developed before the machines arrived

1960-80: Quantum Information theory

1980-90: Feynman's talk – Cannot simulate Quantum system efficiently on a classical computer
Benioff proposes a theoretical framework for QC
Deutsch describes the first universal quantum computer
Ekert invents entangled based secure communication

1990-98: Shor's algorithm for factorization
Grover's search algorithm

1998: 2-qubit NMR quantum computer

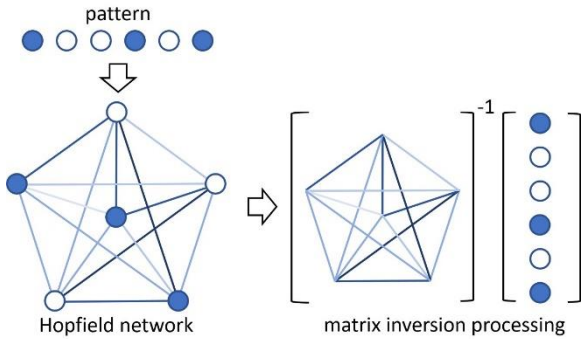


D-Wave IBM Q IONQ rigetti

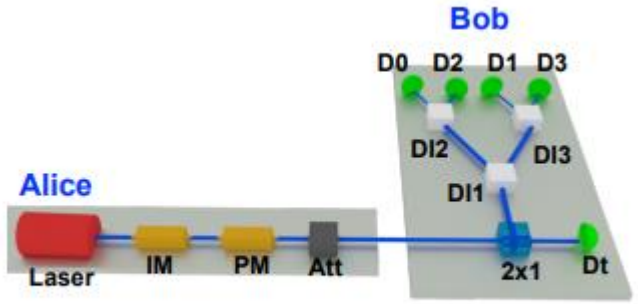
(Present)

Data from Web of science

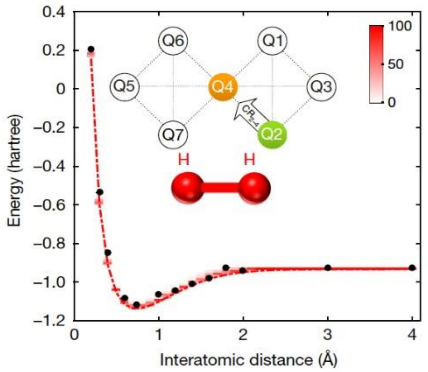
Recent developments



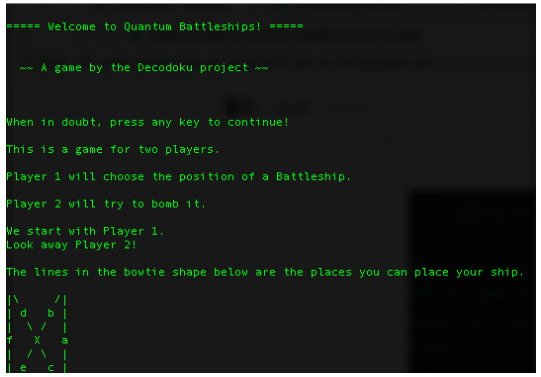
Neural Network[1]



Quantum key distribution [3]



Molecular energy estimation[2]



Playing Battleship [4]

Primary focus: Identify and solve problems which are very hard to be solved on Classical supercomputers (e.g. NP hard combinatorial problems).

Secondary focus: Solve well-established classical problems on a Quantum system more efficiently.

National Quantum Initiative Act (Jan 2019) at a funding level of $1.2 billion impacts growth of quantum computational sciences

[1] Rebentrost, Patrick, et al. "Quantum Hopfield neural network." *Physical Review A* 98.4 (2018): 042308.
[2] Kandala, Abhinav, et al. "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets." Nature 549.7671 (2017): 242.
[3] Islam, Nurul T., et al. "Securing quantum key distribution systems using fewer states." Physical Review A 97.4 (2018): 042347.
[4] https://www.research.ibm.com/ibm-q/ (Last accessed on 20 Feb 2019)

Some issues with near-term quantum computers

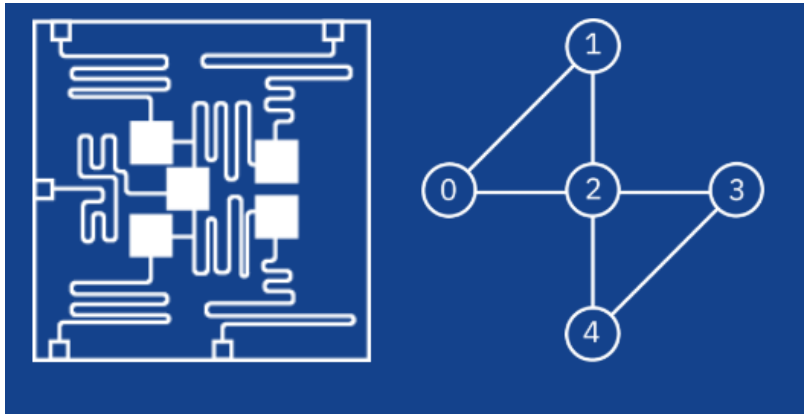
Limited Qubits

In classical computers, with similar binary (0/1) encoding:

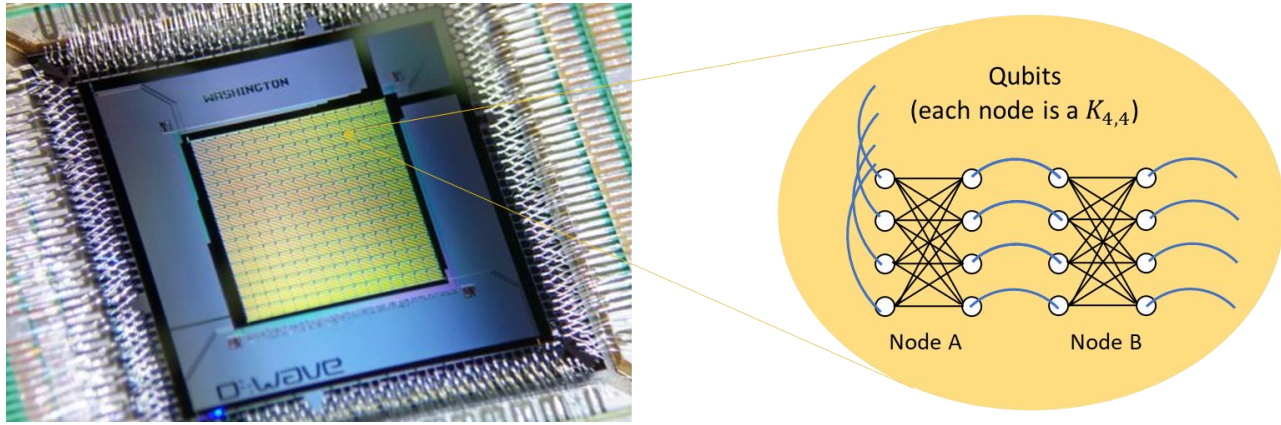
Data type size : 32 bits (float) to 80 bits (long double)
1 GB memory = 12 million high precision variables

In contrast, currently available quantum annealers have a limited number of physical qubits.

Limited connectivity (Gate operations):



Schematic of IBM Q5 processor

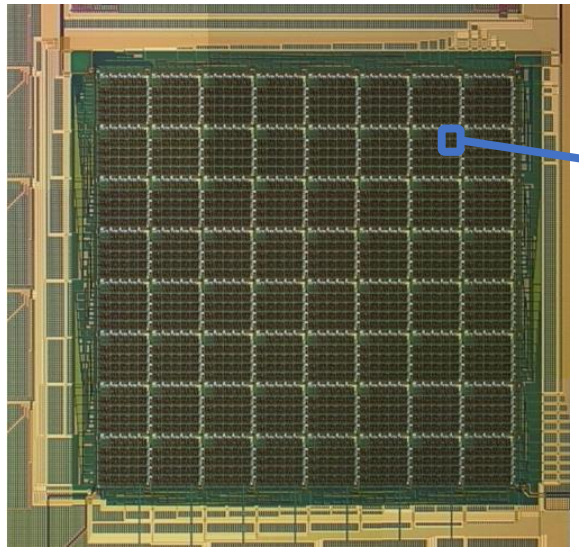


Unit graph structure for D-Wave 2000Q

Quantum annealing: Ising spin systems

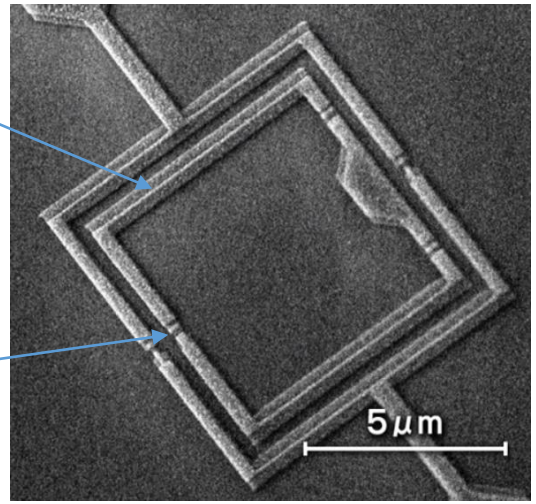
How does a Quantum annealer work?

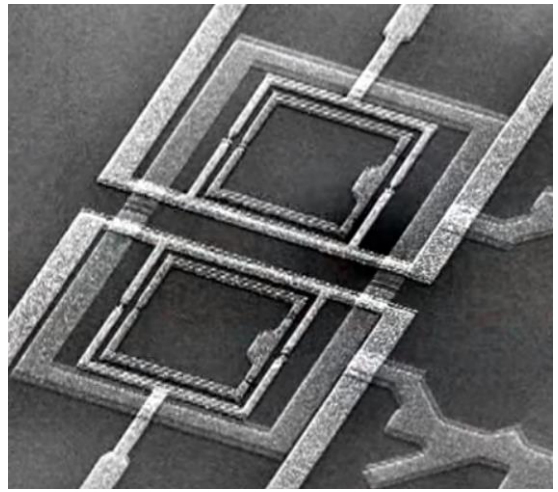
D-Wave Processor



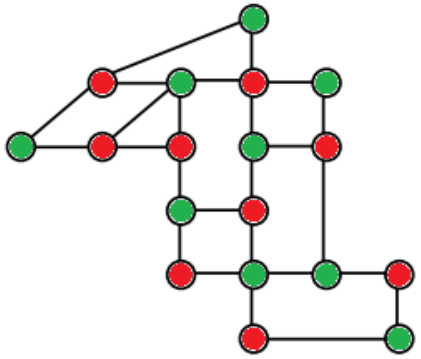
Aluminum loop

Josephson junction





NTT Basic Research Laboratories (2005)



Energy in spin systems is dependent on 3 things:

1. Topology of graph.
2. Parameters H (Field strength) and J (Coupling strength)
3. Labeling of vertices

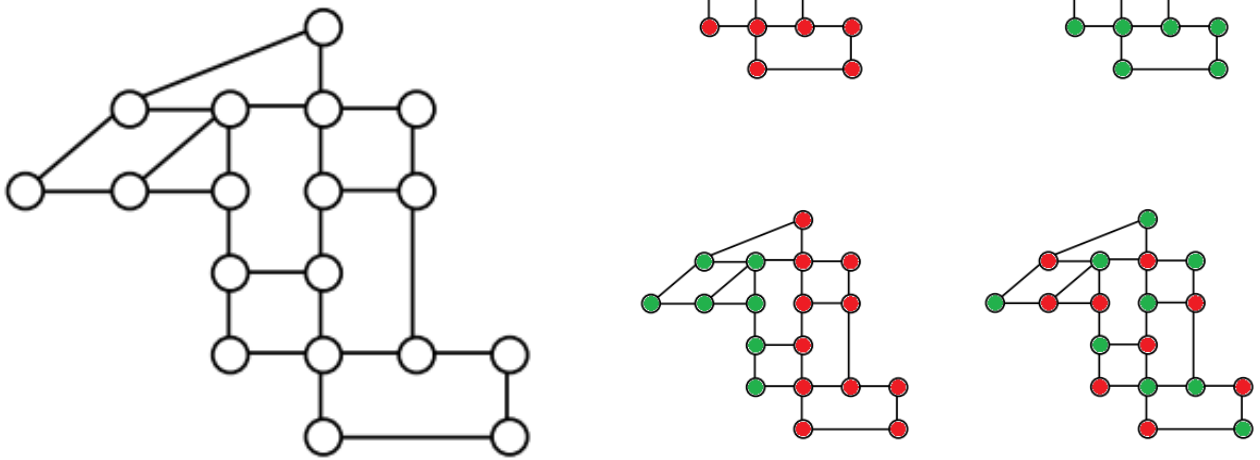
Ising spin systems

Spin systems

- A spin model defines a Hamiltonian (Energy) on a **simple undirected Graph** for a given set of **labelling.**
- An **Undirected Graph** G(V,E) is a set of vertices (V) and edges (E) with no orientation. It is **'simple'** if it does not contain any multi-edge or self loop.
- A **Vertex labeling** is a function of *V* to a set of *labels* ({+1,-1} in our case)

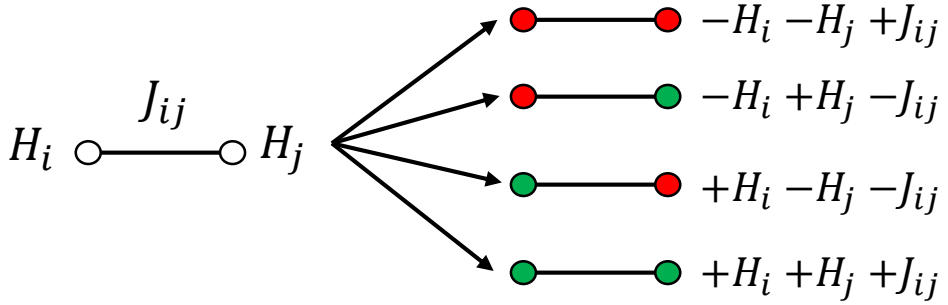
Label = +1

Label = -1

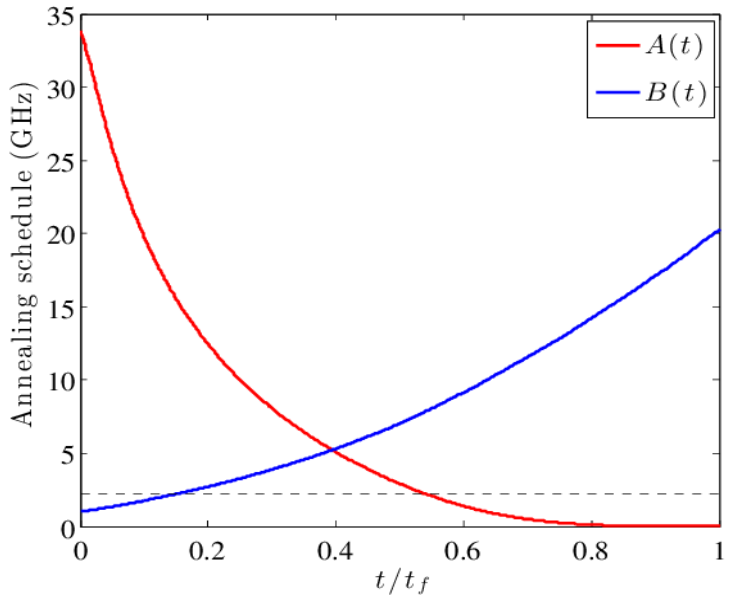


The energy (E) for a given labeling (S) :

$$E(S) = \sum_i H_i S_i + \sum_{\langle i,j \rangle} J_{ij} S_i S_j$$



Annealing procedure

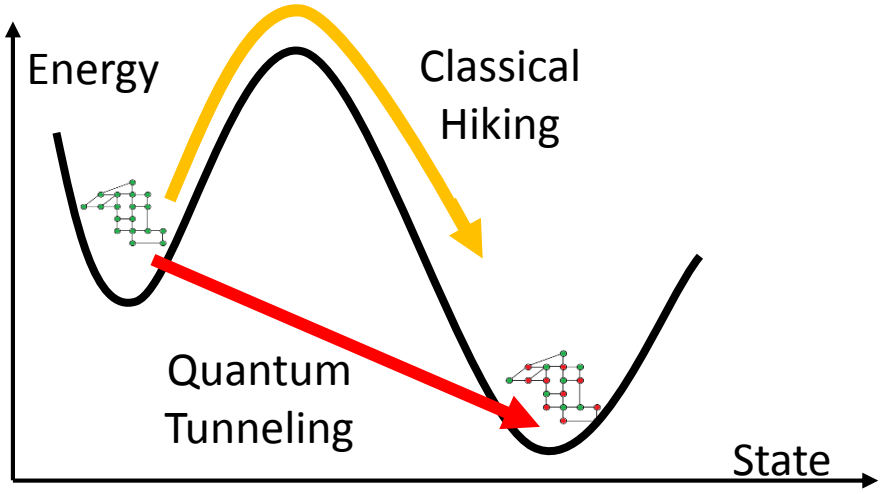


- The annealing procedure is conducted by varying the field as

$$E(t) = A(t)\sum_i S_i^x + B(t)\left(\sum_i H_i S_i^z + \sum_{<i,j>} J_{ij} S_i^z S_j^z\right)$$

- The fridge temperature used in D-Wave Vesuvius processor is 12mK.
- The total annealing time is in range of 5 µs - 2000 µs

Why is this better than classical computing?

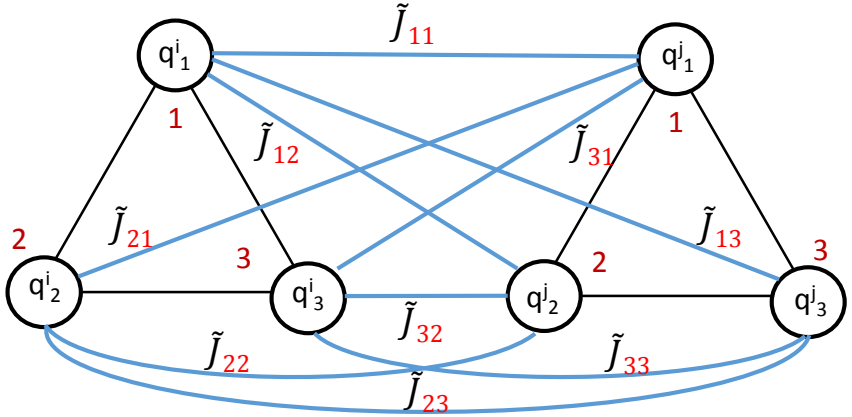


*Mishra, A., Albash, T. and Lidar, D.A., 2018. Finite temperature quantum annealing solving exponentially small gap problem with non-monotonic success probability. *Nature communications*, *9*(1), p.2917.

Key topic of this talk:
Mapping physics to Ising models

$$\Pi[u] = \frac{1}{2}\int_0^1 u'^2 dx$$

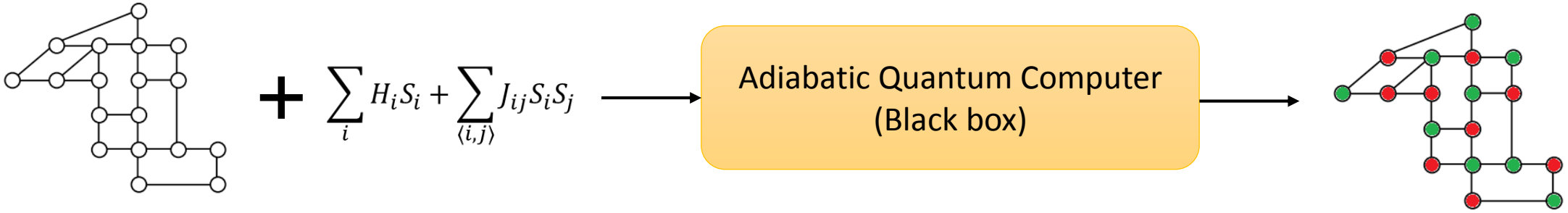
Element



Gate based quantum computers or
Quantum annealers

Scope of this talk

<u>Primary objective:</u> Formulate and test Quantum annealing based algorithms for differential equations.



For most part, we will treat Quantum annealer as a black box which solves graph labeling in one step.

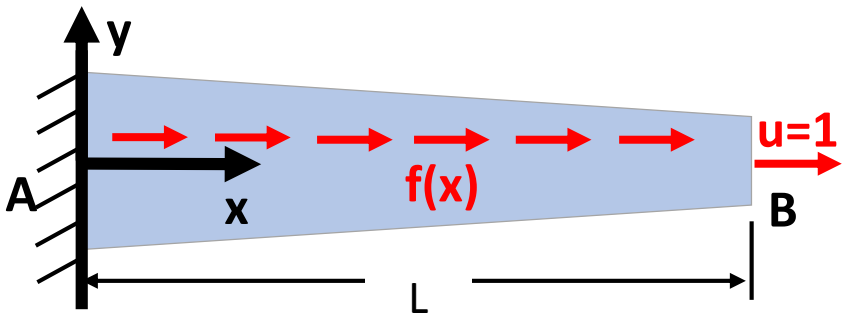
<u>Also in scope:</u> Quantum approximate optimization on gate based quantum computers

<u>Not in Scope:</u> Quantum linear solver-based procedure
A great amount of work has been based on QLSA solver developed by Seth Lloyd [1,2]. QLSA is very promising but is not as robust to noise which becomes important in near term quantum computers.

[1] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd, Quantum algorithm for linear systems of equations, Physical Review Letters (2009), 103, no. 15, 150502, arXiv:0811.3171. (QLSA)
[2] Childs, Andrew M., and Jin-Peng Liu. "Quantum spectral methods for differential equations." arXiv preprint arXiv:1901.00961 (2019).

A simple example

Case Study I: 1-D truss problem



$$\frac{d}{dx}\left(EA(x)\frac{du}{dx}\right) + f(x) = 0 \qquad 0 < x < L$$

Dirichlet boundary conditions: $u(0) = 0$, $u(L) = 1$

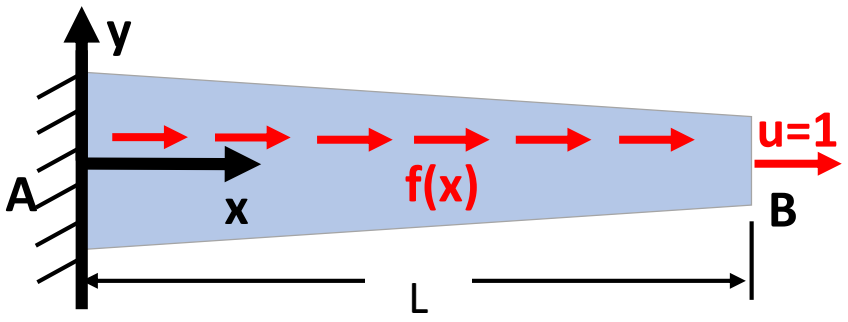
Goals

- Introducing Box algorithm for solving differential equations*
- Familiarizing with the D-wave quantum annealing architecture

*Srivastava, Siddhartha, and Veera Sundararaghavan. "Box algorithm for the solution of differential equations on a quantum annealer." accepted for publication in Physical Review A.

Solving differential equation

Case Study I: 1-D truss problem



$$\frac{d}{dx}\left(EA(x)\frac{du}{dx}\right) + f(x) = 0 \qquad 0 < x < L$$

Dirichlet boundary conditions: $u(0) = 0$, $u(L) = 1$

Energy methods:

Solution obtained by minimizing the potential energy given as:

$$\min\ \pi(u) = \int_0^L \frac{1}{2} EA\left(\frac{du}{dx}\right)^2 - f\,u\,dx$$

Discretization and compact basis

Finite element approximation:

Pick some appropriate finite dimensional space, V_h with basis $\{\phi_1, \phi_2, \dots, \phi_n\}$:

$$u = \sum_{i=1}^{n} a_i \phi_i \text{ with } a_i \in \mathbb{R}$$

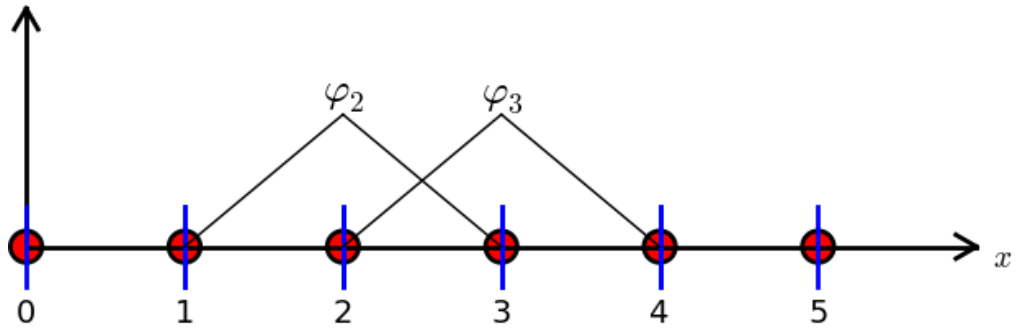
Solve $\min \pi(a)$ to get the best approximation of u

$$\pi(a) = \int_0^L \frac{1}{2} EA \left(\sum_{i=1}^{n} a_i \phi'_i \right)^2 - f \sum_{i=1}^{n} a_i \phi_i \, dx$$

Choice of ϕ:

'Hat functions' (Compact support)

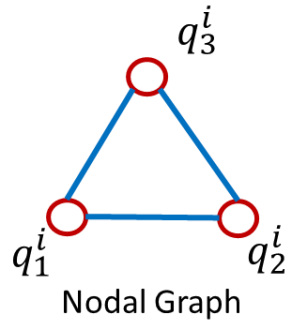
This gives a sparse structure to the minimization problem



Nodal graph

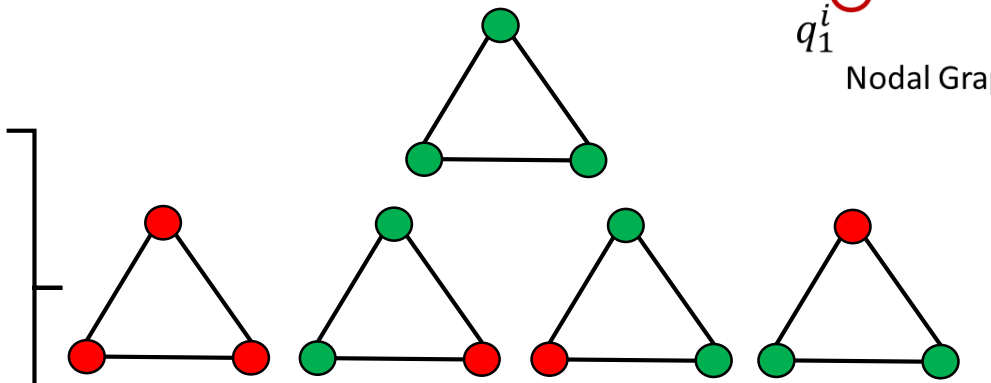
We want to ensure that there is only **'ONE' +1** and **'TWO' -1** on each node

q_1^i	q_2^i	q_3^i	Energy	H=1, J=1
1	1	1	3H+3J	6
-1	-1	-1	-3H+3J	0
-1	1	1	H-J	0
1	-1	1	H-J	0
1	1	-1	H-J	0
-1	-1	1	-H-J	-2
-1	1	-1	-H-J	-2
1	-1	-1	-H-J	-2

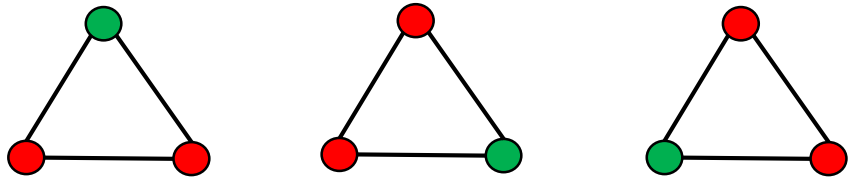


Nodal Graph

$H_i = a$ on all nodes
$J_{ij} = b$ on all edges



Higher energy states

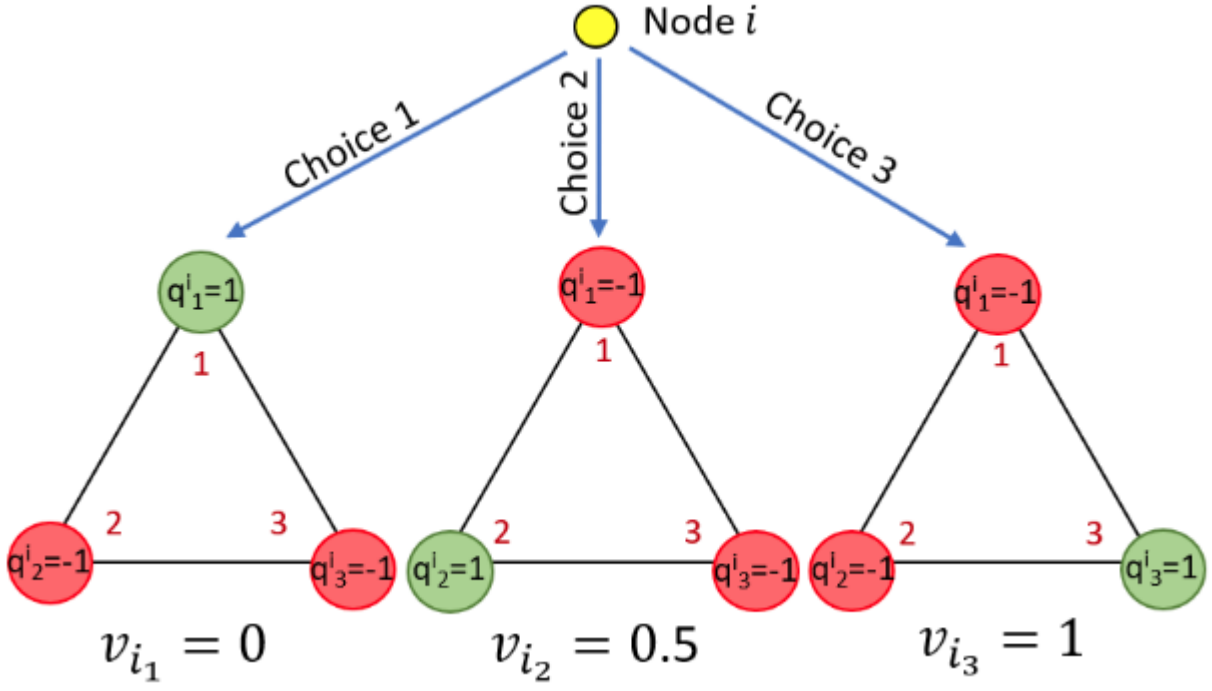


3 Minimum energy degenerate states

Representation of Solution space

Three energy minimizers (symmetric) for each node

⇒ Three values of a_i (coefficient of linear expansion) for i^{th} node

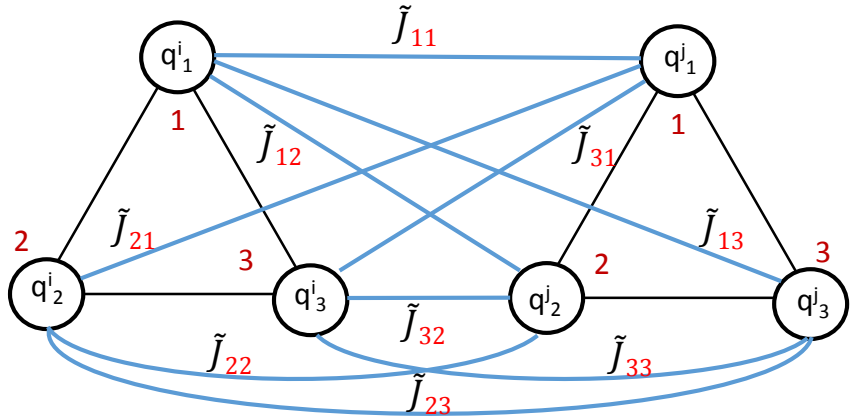


Element graph

Element graph encodes the physics of the problem

Element

- Each node can take 3 values
- Each element can have one of 9 states (a_i, a_{i+1})



$$\pi_e = \frac{1}{2} EA(a_{i+1} - a_i)^2 - f\frac{(a_{i+1} + a_i)}{2}$$

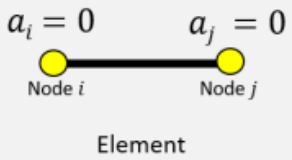
Element graph has 9 edges and 9 valid colorings

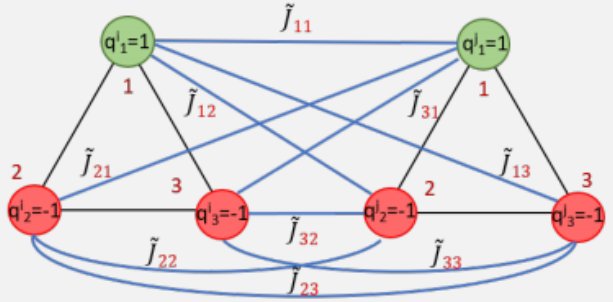
Estimate $\tilde{J}$ (edge strength) such that: $\mathrm{E} = \sum_{ij} \tilde{J}_{ij} S_i S_j = \pi_e(a_i, a_{i+1})$

System of 9 linear equations in 9 variables

Element graph : Example

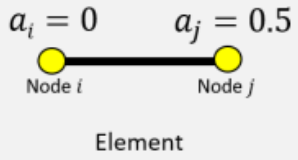
$$\Pi(\mathbf{a}) = (a_1 - a_2)^2$$

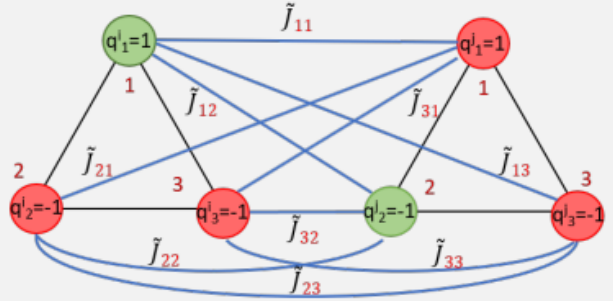




In the above Figure, both nodes take up choice 1 ($a_i = a_j = 0$). The interaction energy for qubits: $E = \tilde{J}_{11} - \tilde{J}_{12} - \tilde{J}_{13} - \tilde{J}_{21} + \tilde{J}_{22} + \tilde{J}_{23} - \tilde{J}_{31} + \tilde{J}_{32} + \tilde{J}_{33} = (a_i - a_j)^2 = 0$

Sample 2:



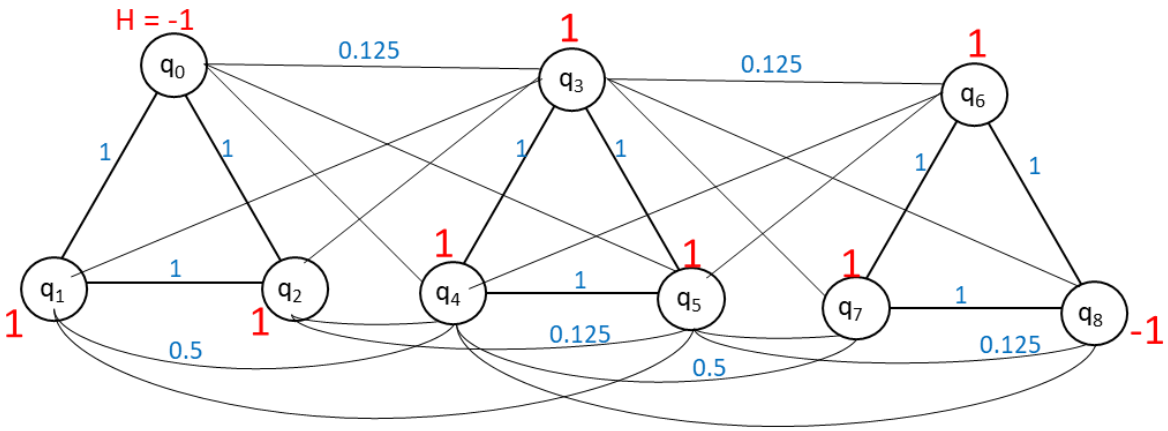


In the above Figure, node i takes up choice 1 ($a_i = 0$), while node j takes up choice 2 ($a_j = 0.5$). The interaction energy for qubits: $E = -\tilde{J}_{11} + \tilde{J}_{12} + \tilde{J}_{13} + \tilde{J}_{21} - \tilde{J}_{22} - \tilde{J}_{23} - \tilde{J}_{31} + \tilde{J}_{32} + \tilde{J}_{33} = (a_i - a_j)^2 = 0.25$

Element graph : Example (contd)

$$\begin{bmatrix} +1 & -1 & -1 & -1 & +1 & +1 & -1 & +1 & +1 \\ -1 & +1 & +1 & +1 & -1 & -1 & -1 & +1 & +1 \\ -1 & +1 & +1 & -1 & +1 & +1 & +1 & -1 & -1 \\ -1 & +1 & -1 & +1 & -1 & +1 & +1 & -1 & +1 \\ +1 & -1 & +1 & -1 & +1 & -1 & +1 & -1 & +1 \\ +1 & -1 & +1 & +1 & -1 & +1 & -1 & +1 & -1 \\ -1 & -1 & +1 & +1 & +1 & -1 & +1 & +1 & -1 \\ +1 & +1 & -1 & -1 & -1 & +1 & +1 & +1 & -1 \\ +1 & +1 & -1 & +1 & +1 & -1 & -1 & -1 & +1 \end{bmatrix} \begin{bmatrix} \widetilde{J}^n_{11} \\ \widetilde{J}^n_{12} \\ \widetilde{J}^n_{13} \\ \widetilde{J}^n_{21} \\ \widetilde{J}^n_{22} \\ \widetilde{J}^n_{23} \\ \widetilde{J}^n_{31} \\ \widetilde{J}^n_{32} \\ \widetilde{J}^n_{33} \end{bmatrix} = \begin{bmatrix} (v_{i_1} - v_{j_1})^2 \\ (v_{i_2} - v_{j_1})^2 \\ (v_{i_3} - v_{j_1})^2 \\ (v_{i_1} - v_{j_2})^2 \\ (v_{i_2} - v_{j_2})^2 \\ (v_{i_3} - v_{j_2})^2 \\ (v_{i_1} - v_{j_3})^2 \\ (v_{i_2} - v_{j_3})^2 \\ (v_{i_3} - v_{j_3})^2 \end{bmatrix}$$

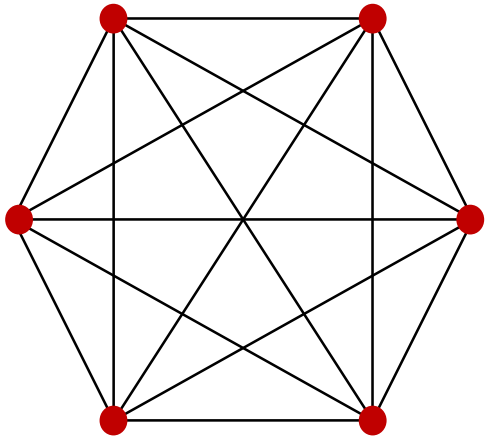
$$\widetilde{J}^1 = \widetilde{J}^2 = \begin{bmatrix} 0.1250 & 0.3750 & 0.3750 \\ 0.3750 & 0.5000 & 0.3750 \\ 0.3750 & 0.3750 & 0.1250 \end{bmatrix}$$

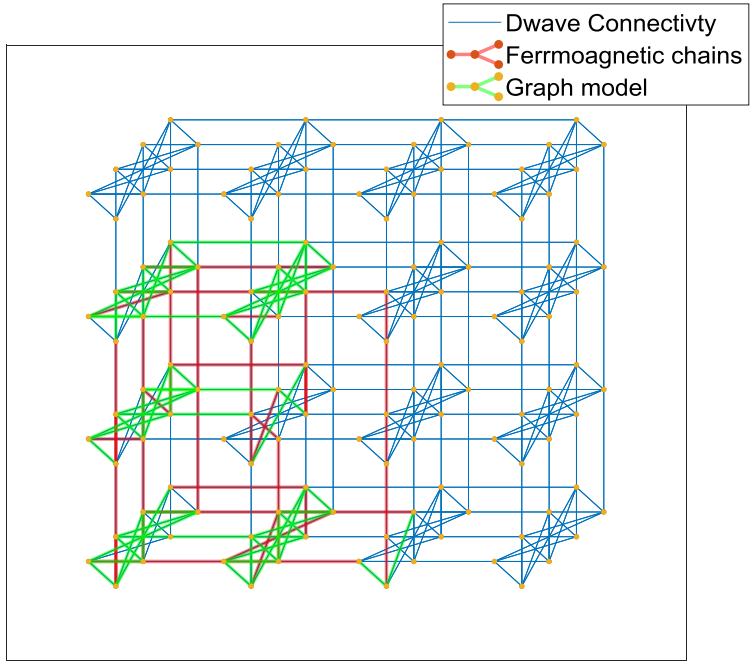


All unmarked links have a weight of 0.375

Graph Embedding

- We want to map all nodes from the graph model onto the physical graph.
- This mapping should preserve the minimum energy states.





Required Connectivity

Physical connectivity

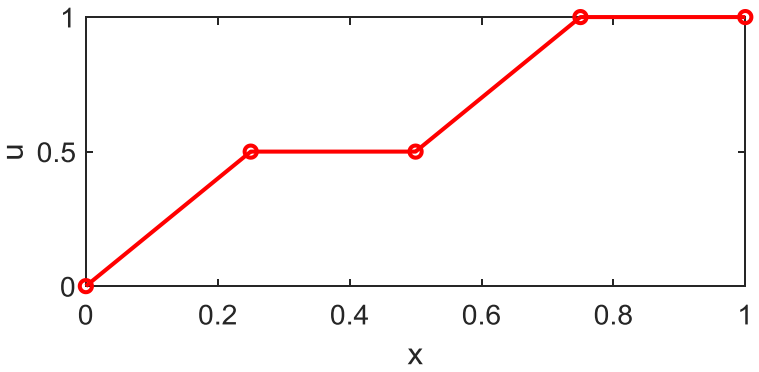
Good News: D-wave provides you with API's to search embedding in a heuristic fashion

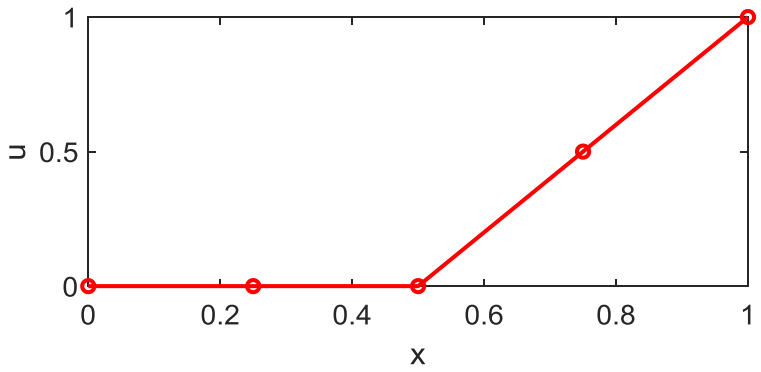
Energy minimization

Boundary Conditions: $a_1 = u_1$ and $a_n = u_3$

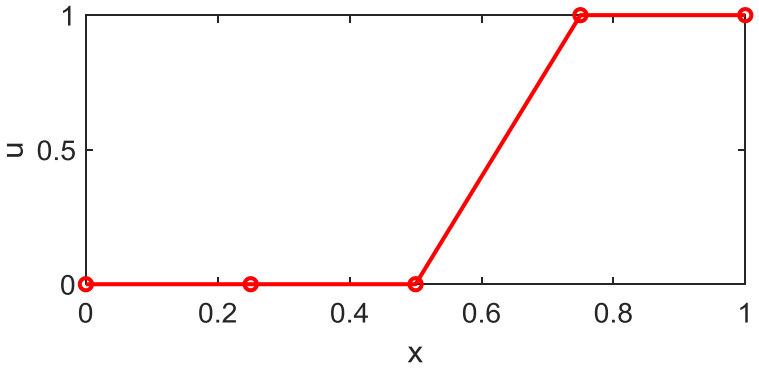
Choose H_i (Field term) corresponding to q_1^1 and q_n^3 as large negative values.

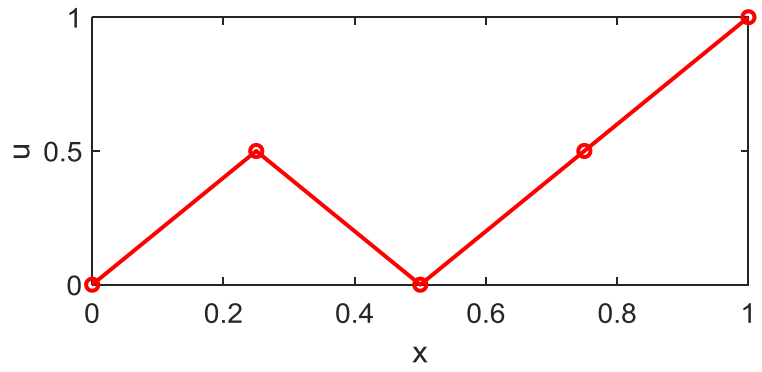
Solve for $EA(x) = 1$ and $f(x) = 0$





Low Energy Solution
High Probability

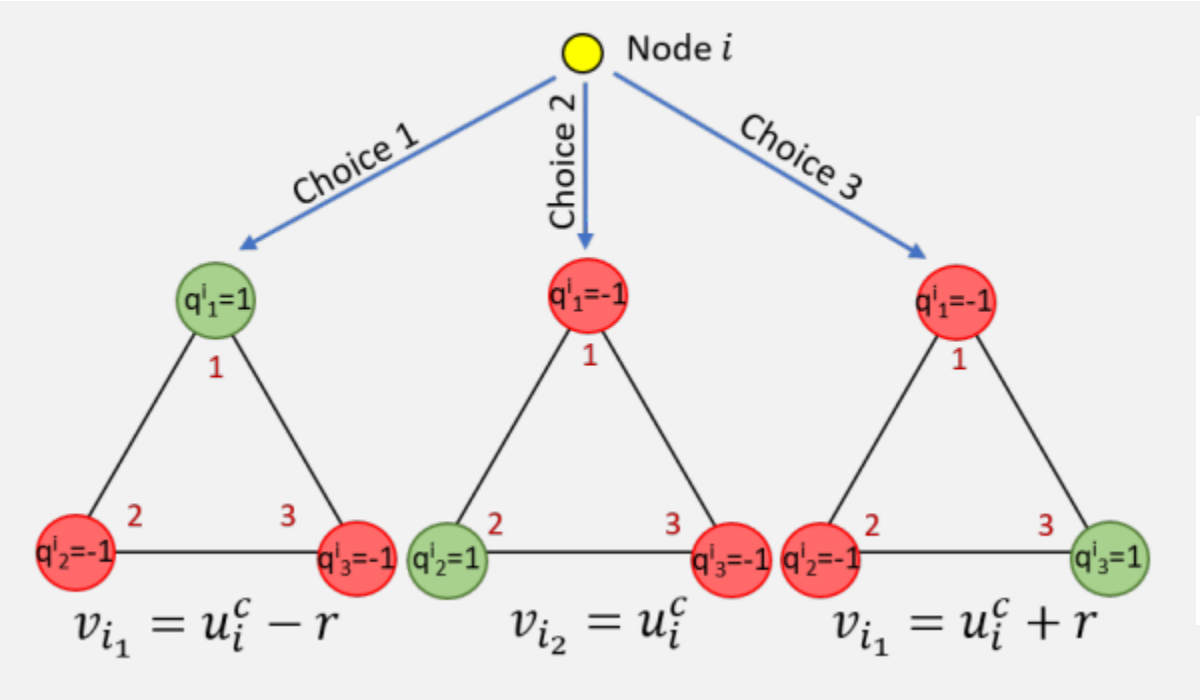


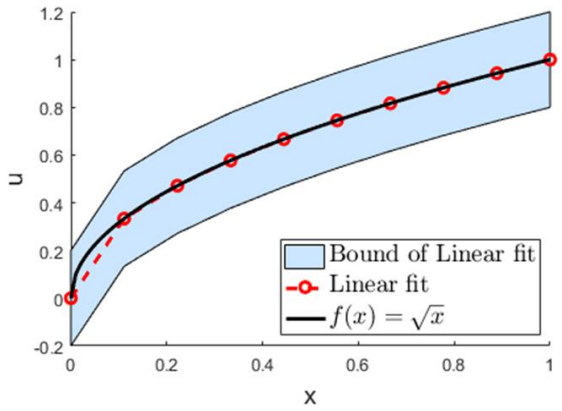


High Energy Solution
Low Probability

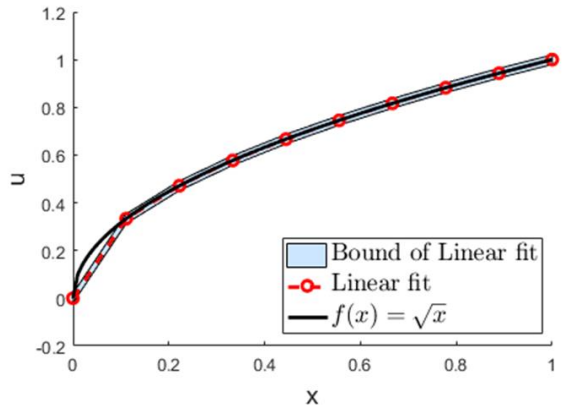
100 labels per node are required to get a precision of 0.01 for a bounded displacement between [0,1].

Introduce slack variables





(a)



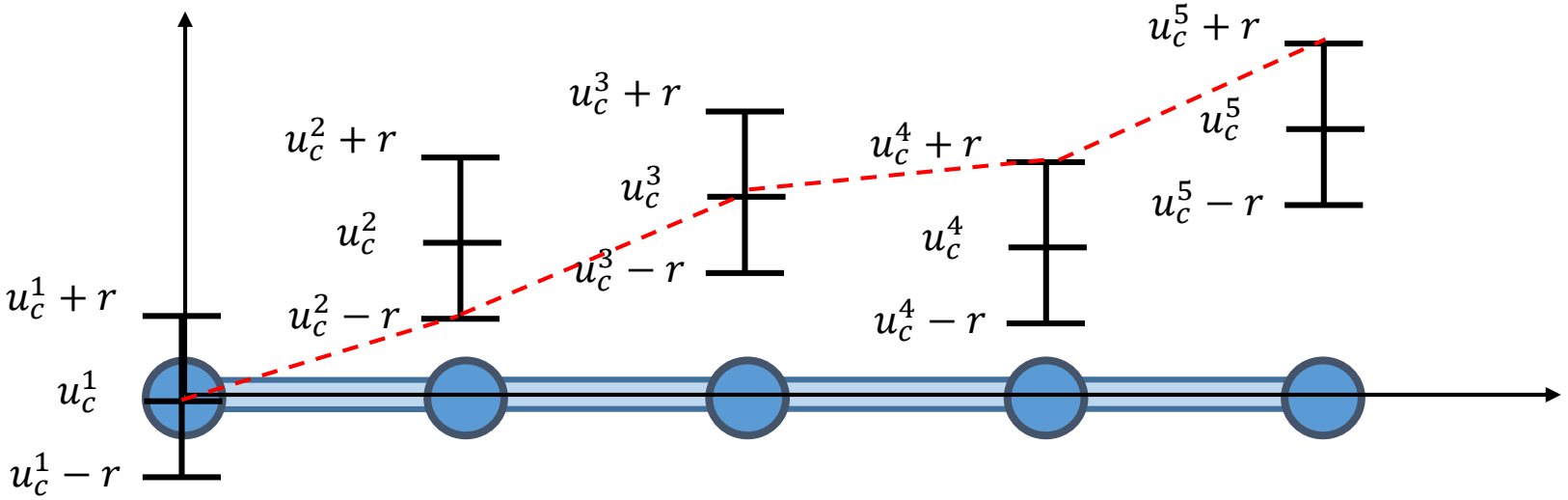
(b)

Box algorithm: Iterative Procedure

- Define u_c^i as the displacement corresponding to center label of i^{th} node.

$$\boldsymbol{u_c} = \{u_c^1, u_c^2, \ldots\ldots, u_c^n\}$$

- And a parameter, 'r' (called slack variable) so that the displacements of i^{th} node for corresponding to labels {1,2,3} are $\{\boldsymbol{u_c^i - r, u_c^i, u_c^i + r}\}$, respectively.



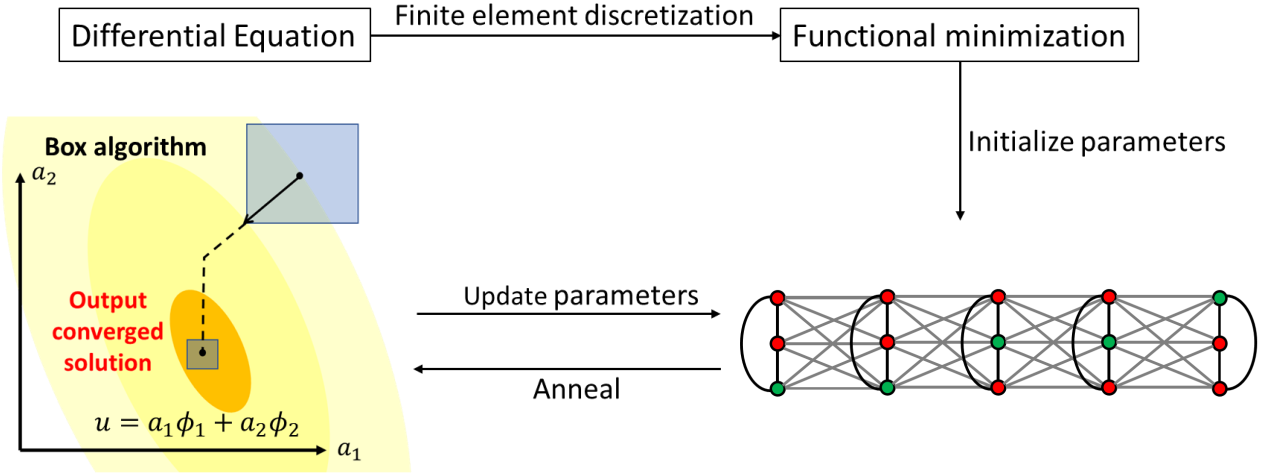
Definition: 'Box' is the high dimensional representation of all possible outcomes. In this case 5D space. Box length = 2r
Box center: $\{u_c^1, u_c^2, \ldots\ldots, u_c^5\}$

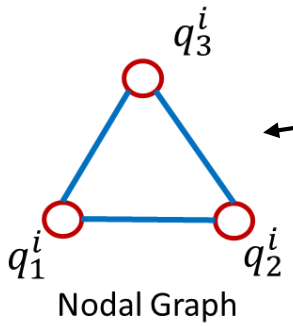
Approach:

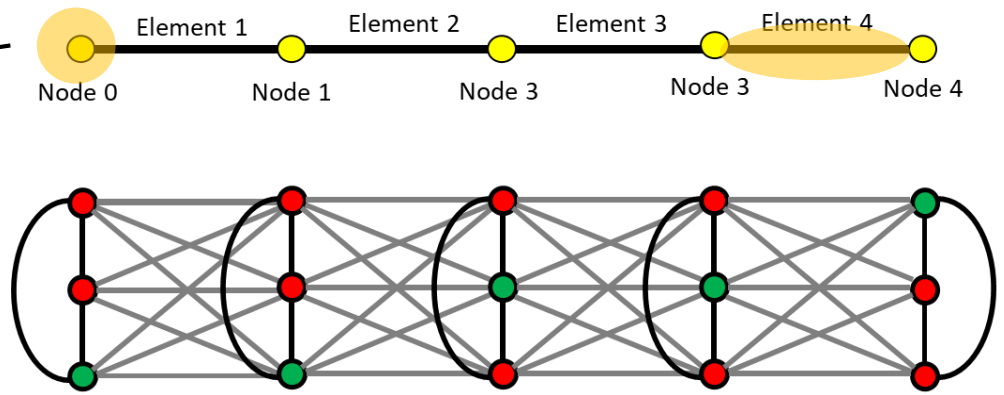
1. **If box corner is chosen, recenter the box to the corner.**
2. **If box center is chosen, shrink the box.**

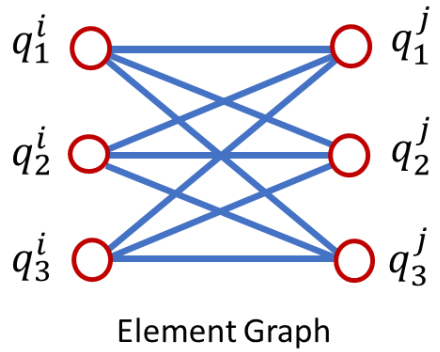
Iterative solution

The link weights are modified based on the current choices of center and slack variable for each node.

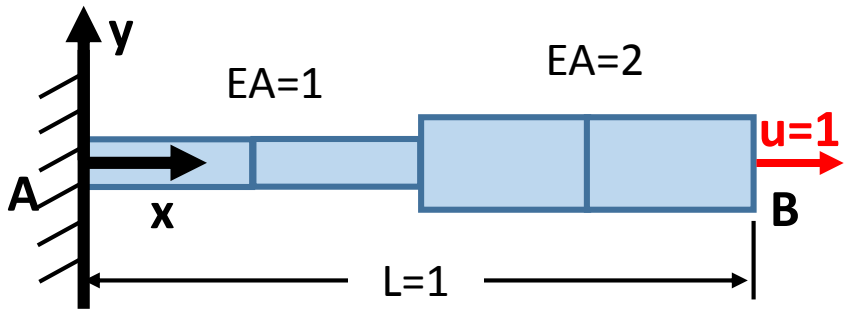




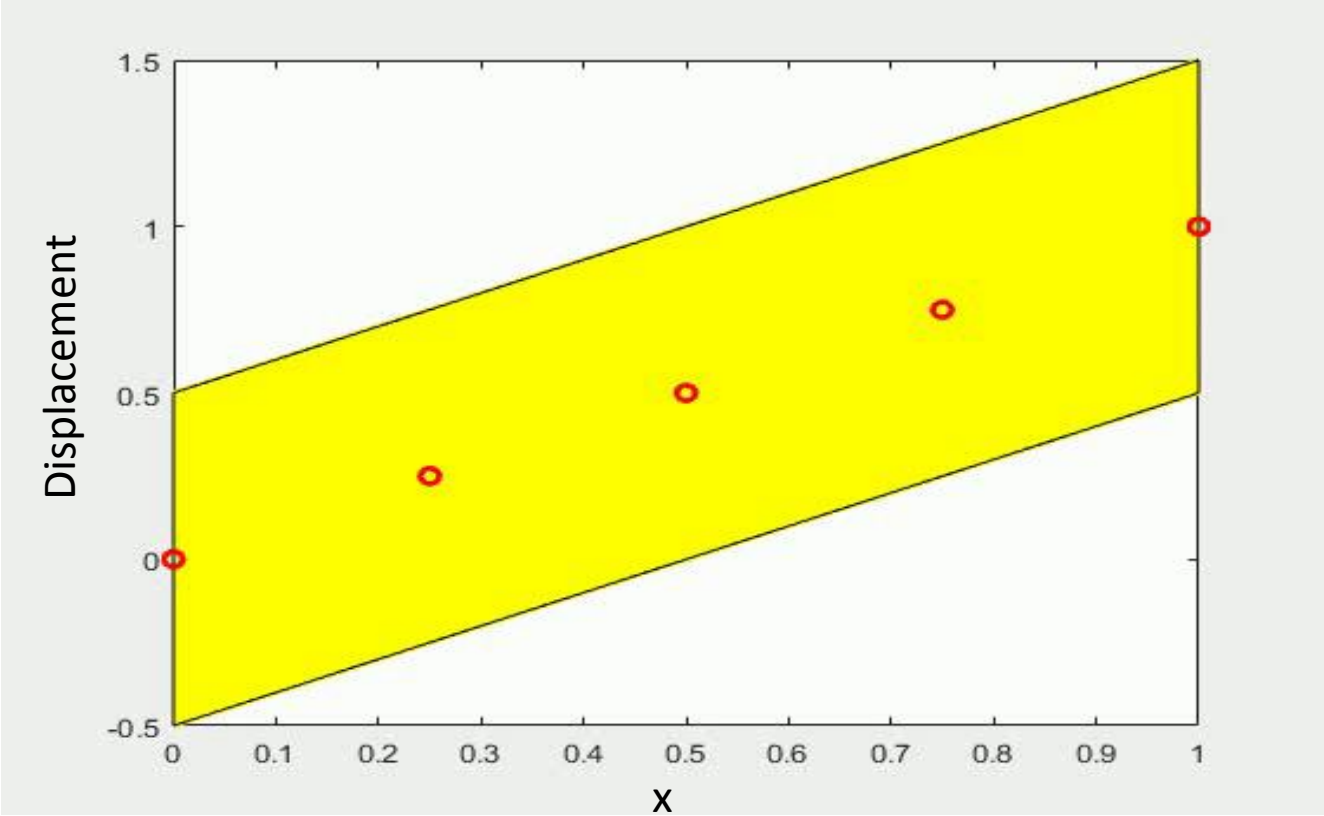




Numerical solution



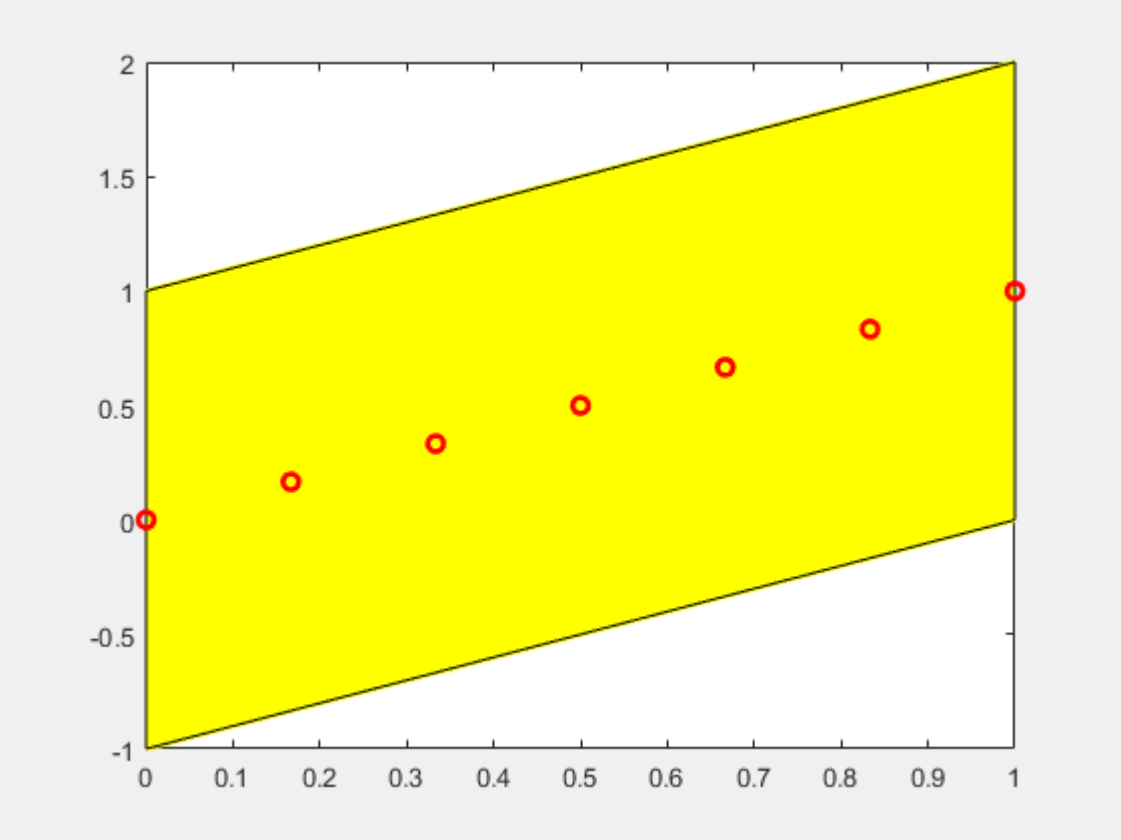
- The yellow region represents the space between $\boldsymbol{u_c} + \boldsymbol{r}$ and $\boldsymbol{u_c} - \boldsymbol{r}$.
- The result converges to the exact solution

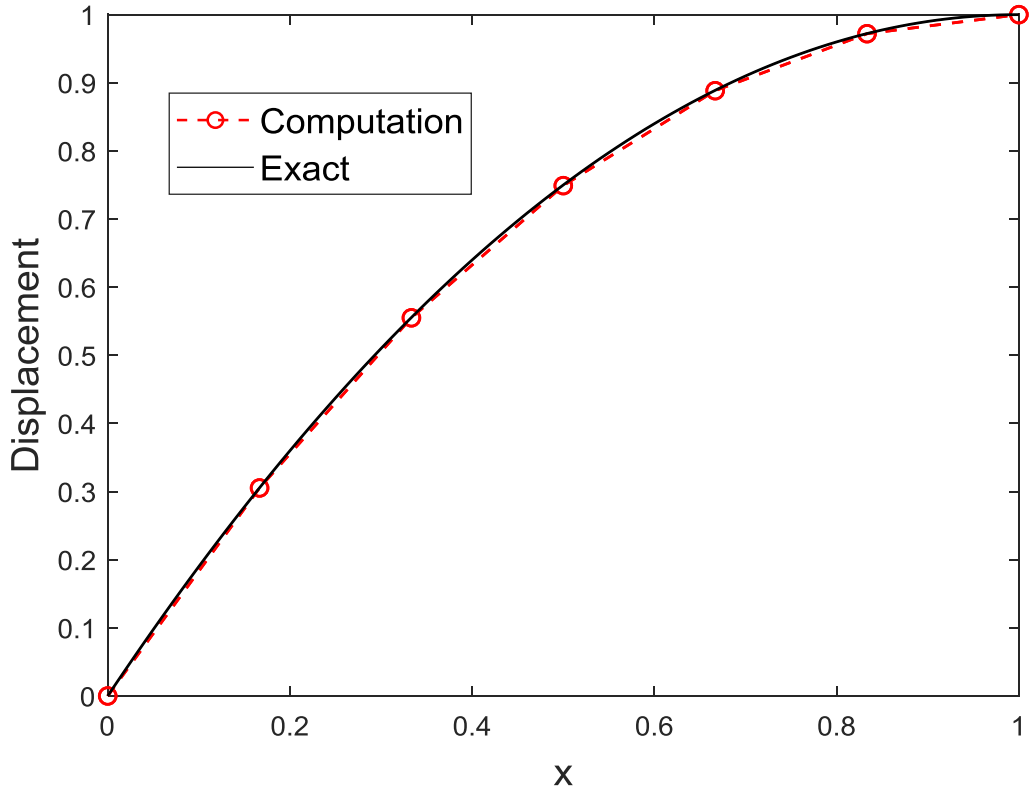


Quantum computer's solution

$$EA(x) = 2 - x$$

$$f(x) = 4x - 6$$

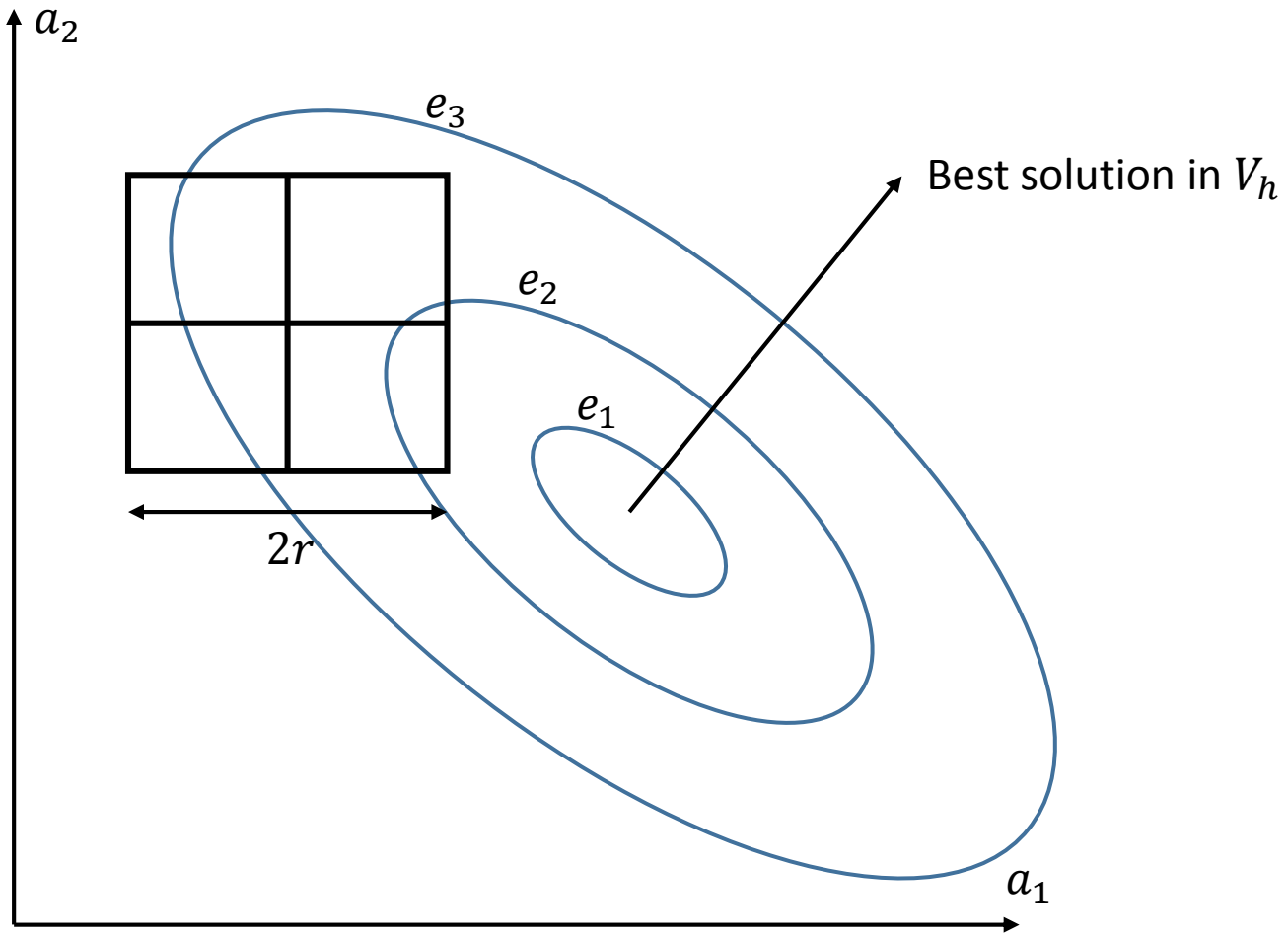




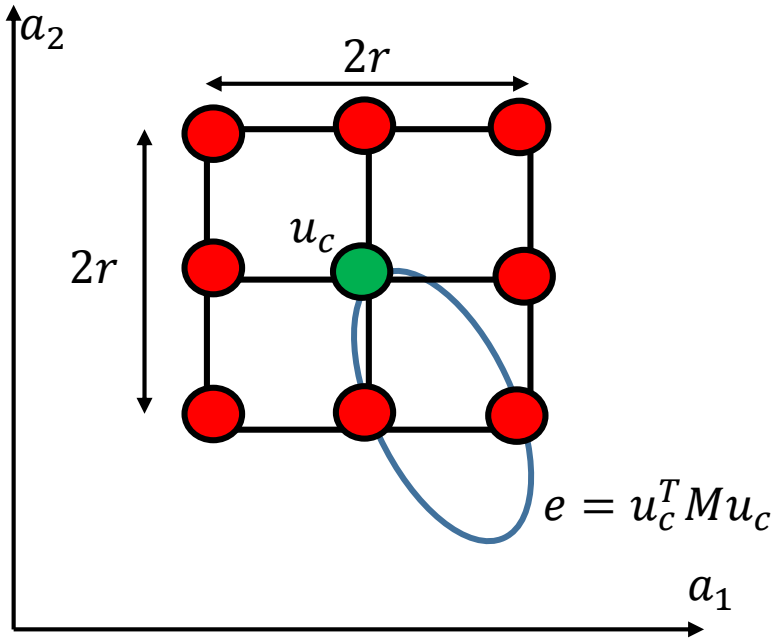
Convergence

$$v = \sum_{i=1}^{n} a_i \phi_i \quad \text{with } \phi_i \in V_h \text{ and } a_i \in \mathbb{R}$$

Now we sample a_i from $\{u_c^i - r, u_c^i, u_c^i + r\}$



Convergence



Consider the iteration, when the minimum energy point corresponds to u_c.

Observe:

- All other points in the sample lie outside the energy contour corresponding to $F(u_c)$
- The length of the major axis is bounded for a given matrix M

Only considering the horizontal and vertical node, you can bound the major axis of the ellipse as:

$$d_{max} = \sqrt{2}(1 + \lambda_{max}/\lambda_{min})r$$

Following the same logic the bound can be extended to $\mathbb{R}^2$: $d_{max} = 2\left(1 + (n-1)\frac{\lambda_{max}}{\lambda_{min}}\right)\frac{r}{\sqrt{n}}$

i.e. for a finite discretization, $\lim_{r \to 0} d_{max} \to 0$

This mean as $r \to 0$, u_c approaches the best approximation for u in V_h

Another example

Case Study II: Advection-Diffusion problem

Homogeneous Advection-diffusion equation with Dirichlet boundary conditions on both ends
Governing Equation:

$$-u'' + vu' = 0 \qquad 0 < x < 10$$

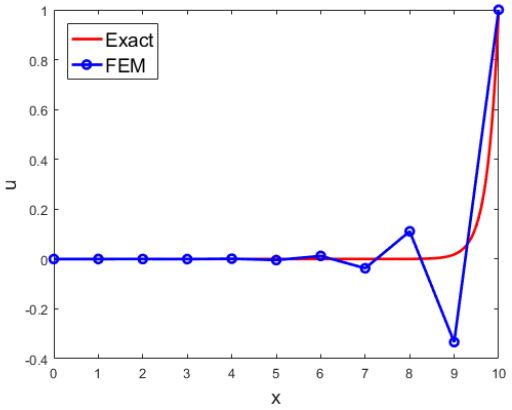
$$u(0) = 0 \quad , \quad u(10) = 1$$

Finite Element Method (Galerkin approach)

Weak-form: $$W(u, \tilde{u}) := \int_{\Omega} (u'\tilde{u}' + vu'\tilde{u})\, dx = 0$$

- Observe that the weak form is non-symmetric

- This results in unstable solution for high values of $'v'$ i.e. in highly advective flows.



FEM (Linear elements) calculation for $v = 4$

Goals

- Application of energy minimization in a restrictive way
- Some tweaks in the Box algorithm and speed of convergence
- Error correction measures

Box algorithm for A-D equation

First, we need a functional minimization form

Potential flow assumption: $v = -\nabla\phi$

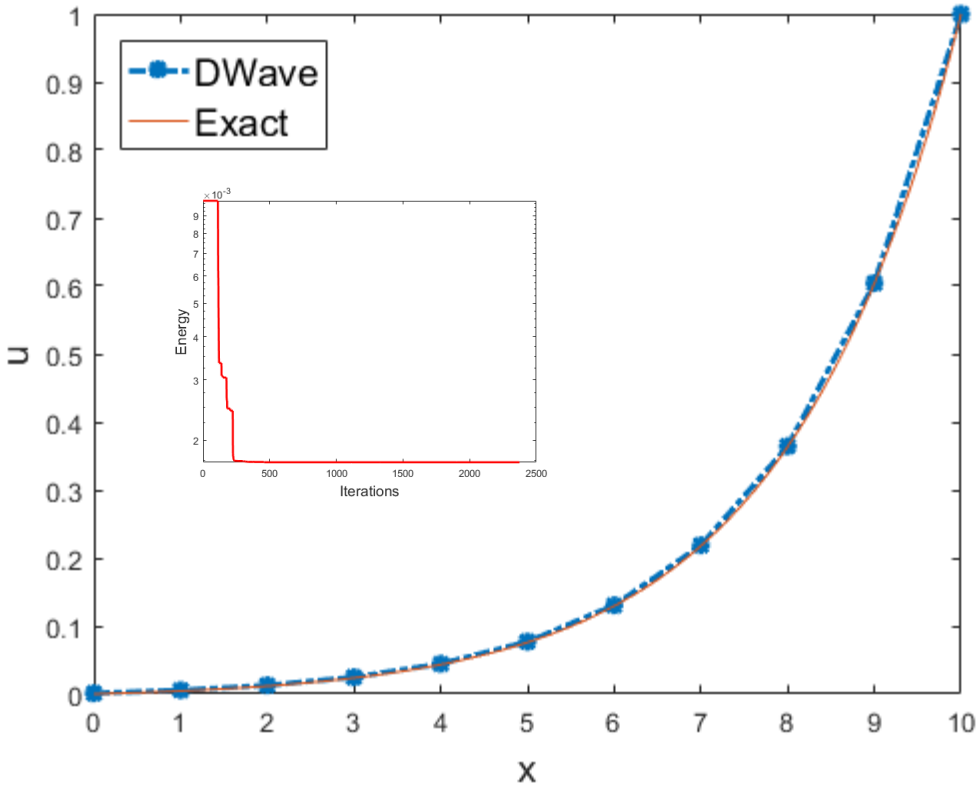
$$F_v[u] = \frac{1}{2}\int_0^L e^{-v.x}(u')^2 dx$$

This energy can be written for n-dimensions with non-homogenous terms with Dirichlet and flux boundary conditions*

Restriction (Necessary condition for $n \geq 2$)

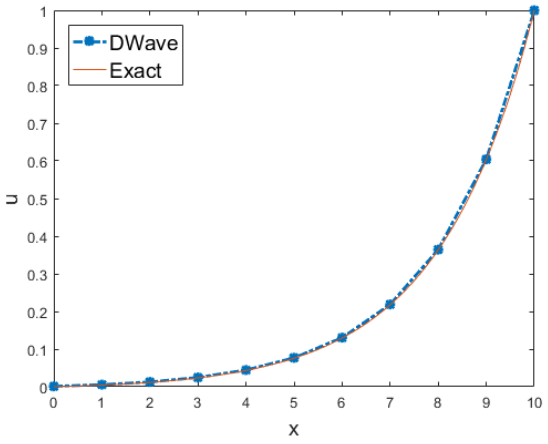
$$\nabla \times v = 0$$

Application of Box algorithm is same as truss problem

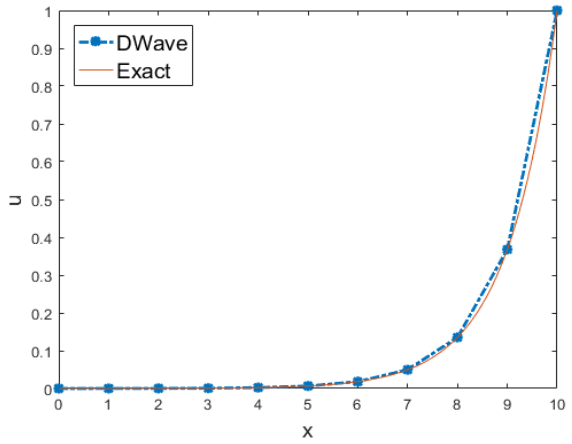


*Auchmuty, Giles. "Variational principles for advection–diffusion problems." *Computers & Mathematics with Applications* 75.6 (2018): 1882-1886.

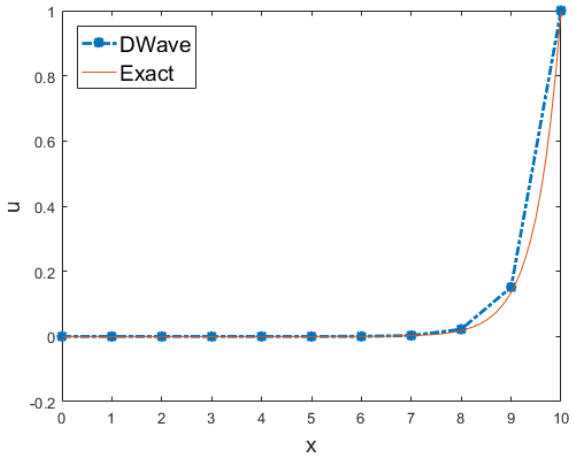
Slow convergence



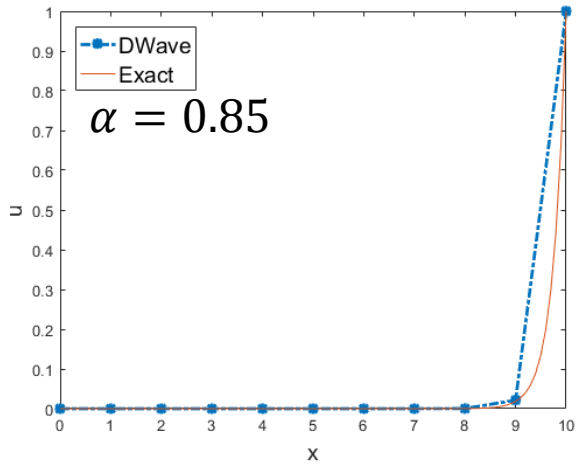
$v = 0.5$



$v = 1$



$v = 2$



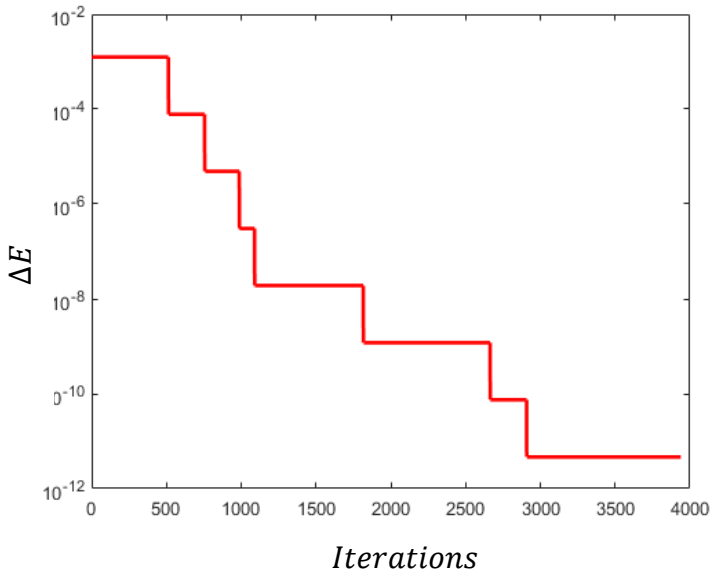
$v = 4$

- Box algorithm performs well for smaller velocities
- For higher velocities we use step size selection

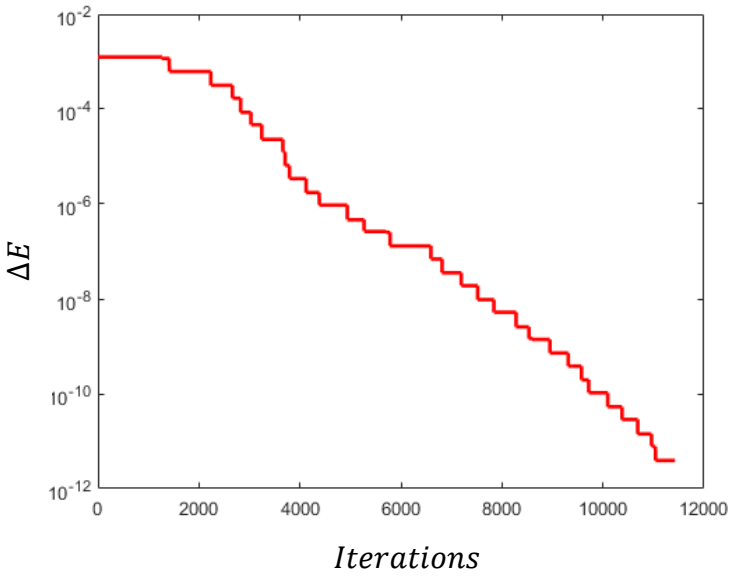
$$r_{new} = \alpha r_{old}$$

$$0 < \alpha < 1$$

Summary of Case Study II



$\alpha = 0.5$



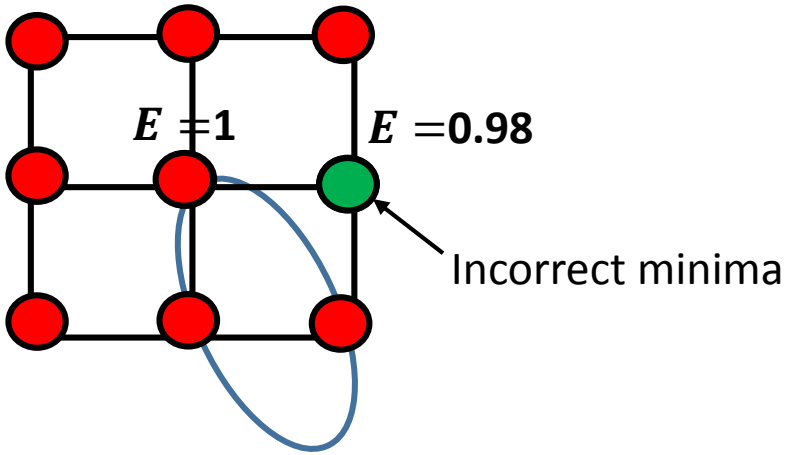
$\alpha = 0.85$

- We formulated and implemented Box algorithm for Advection-Diffusion (w/ Potential flow)
- We showed that step size selection can be used to approach the global minima

Error correction

So far, we have treated the quantum computer as a black box which outputs the correct minima.

When $\delta(E) \ll E$ i.e. the relative energy of all states are same then the solver can output sub-optimal solutions.



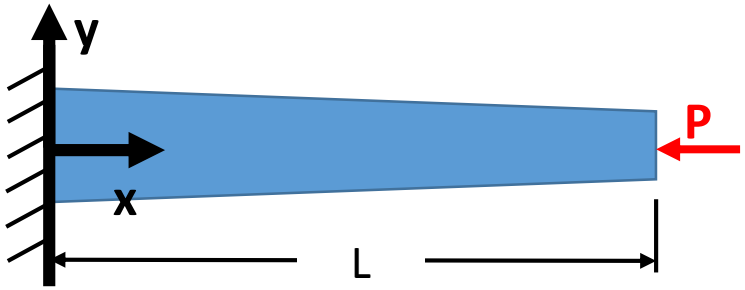
Idea for error correction: Scale energy to maximize the gap between the different states: $(H, J) \rightarrow (H', J')$

$$E' = \frac{1}{a} \sum_{i \in element} (E_i - b_i)$$

Choose a, b_i appropriately

Bifurcation

Case Study III: Beam-buckling problem



4th Order differential equation with critical behavior

Governing Equation:

$$EIw'''' + Pw'' = 0 \qquad 0 < x < L$$

Boundary conditions at $x = x_b$:

$w(x_b) = 0$ (Displacement) or $V(x_b) \sim w'''(x_b) = 0$ (Shear)

$w'(x_b) = 0$ (Slope) or $M(x_b) \sim w''(x_b) = 0$ (Moment)

Energy form

$$F[w] = \frac{1}{2}\int_0^L EI(w'')^2 - P(w')^2\, dx$$

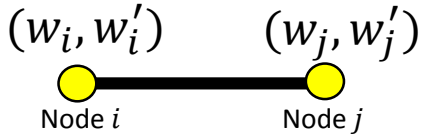
Goals

- Introducing higher order derivatives
- Non convex energy form when $P > P_{cr}$

Bifurcation

FEM discretization:

We enforce continuity of slope on element boundary using Hermite cubic interpolation



Element

At each node there are 2 DOF's : (w_i, w_i')

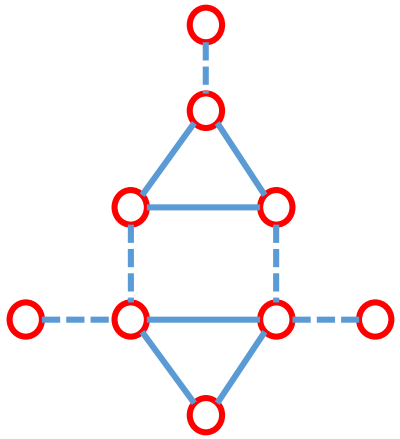
Note: Having 2 DOF's per node means we need to construct new nodal and element graphs

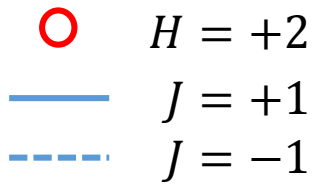
Nondimensionalized form:

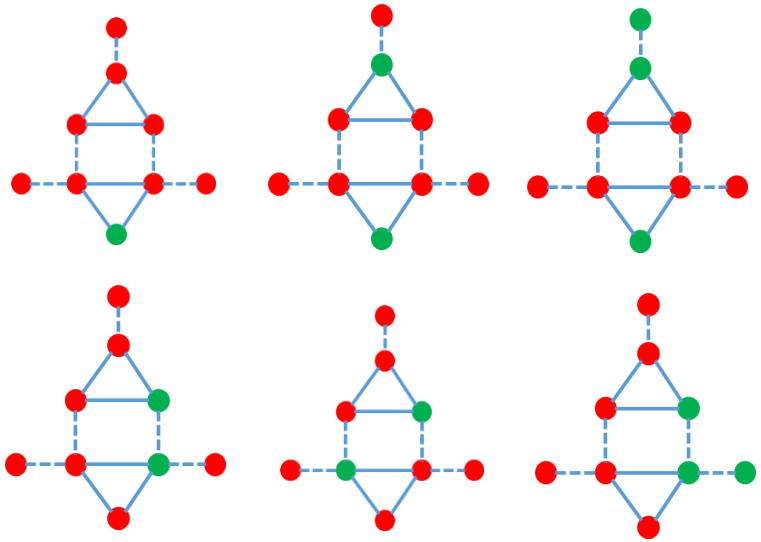
$$\frac{2\pi L^3 F}{EI} = \frac{1}{2}\sum_e \int_0^1 (w'')^2 - \overline{P_c}\left(\frac{l_e}{L}\right)^2 (w')^2\, dz \qquad \text{with} \qquad \overline{P_c} = \frac{\mathrm{PL}^2}{\mathrm{EI}}$$

Bifurcation

Nodal and Element graphs:







9 Energy minimizing states

$\times 2$

(bilateral symmetry)

Each minimizing state of the nodal graph is mapped to one of the 9 solutions of the node

$$\{w_c^i - r, w_c^i, w_c^i + r\} \times \{w'^i_c - r, w'^i_c, w'^i_c + r\}$$

Element graph is constructed as a complete bipartite graph between consecutive nodes.

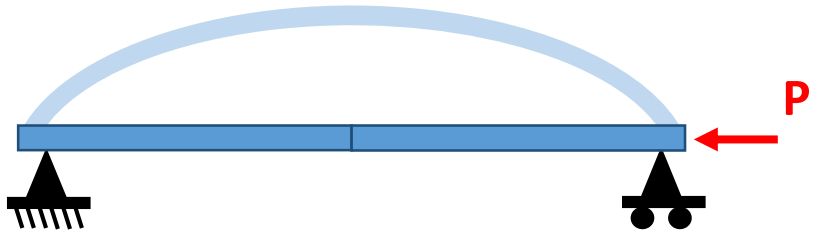
Total connections (element graph) = 81

Total possible states for the element = 81

Exactly solvable weights!!!

Why is non-convexity a problem?

2-element problem example

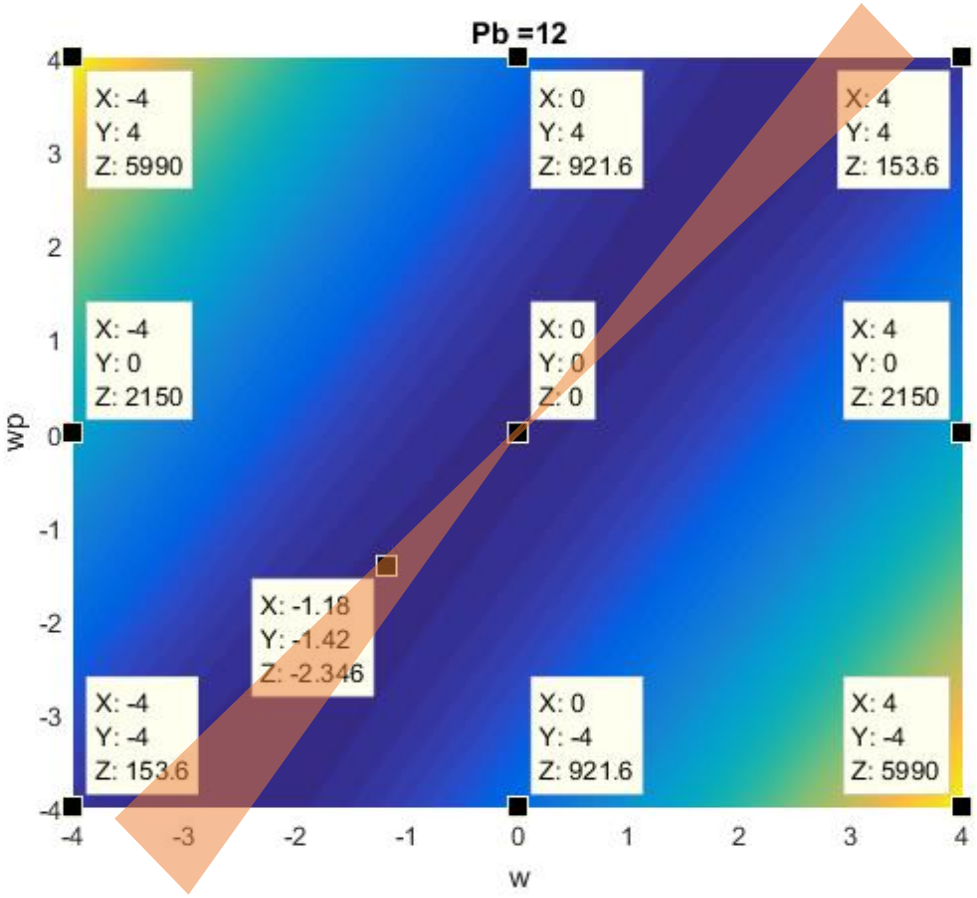


Symmetry of the problem: $w_1' = -w_3'$

$w_2' = 0$

Essentially two degrees of freedom: $w_1' \sim w_p, w_2 \sim w$

Observe single slack variable box algorithm cannot resolve the downward hill of the saddle and gives a false stable point. We augment another slack variable for the slope.



Red region constitutes the downward hill of the saddle

Multiple slack variables

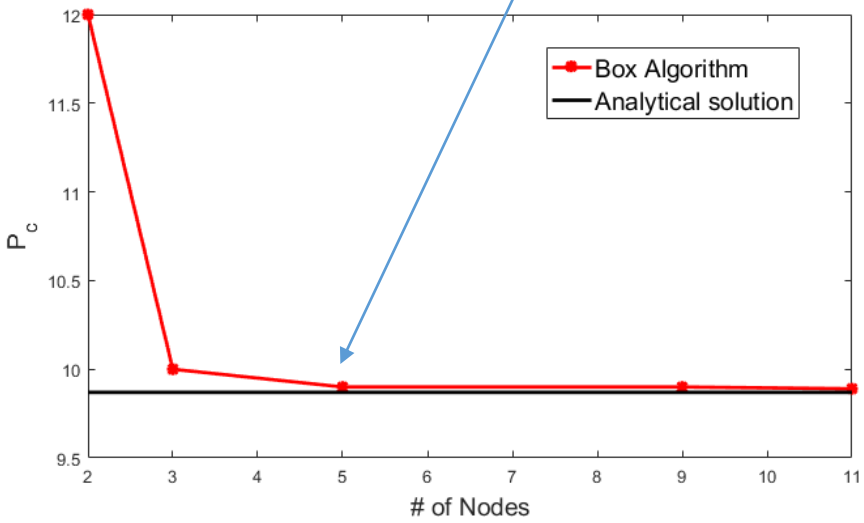
Naturally identifies critical loads via energy minimization

A heuristic remedy:

Consider different box sizes for different variables

$$\{w_c^i - r_1, w_c^i, w_c^i + r_1\} \times \{w'^i_c - r_2, w'^i_c, w'^i_c + r_2\}$$

In the post buckling solution (for 2 element case) the solution tends to choose the up/down solution with similar likelihoods.



More work is needed in this direction to extend the algorithm for non-convex problems.

Summary

- Formulated and implemented an energy-based algorithm for solving differential equations on quantum annealer.
- Showed applications in following different types of equations:
  - Truss mechanics
    - Convergence of the method for convex problems.
  - Advection-Diffusion
    - Discussed an energy formulation
    - Convergence rate depends on the contraction step
    - Error correction strategies
  - Beam buckling problem
    - Introducing higher order derivatives by augmenting nodal and element graph
    - Non-convex: multiple slack variables

Prospective

Comparison to Finite elements:

Complexity: Similar complexities

FEM: Assembly and solve O(N) for 1D problems

Box: Computing element graphs imposes O(N) complexity (per iteration) as well. However, annealing time depends on the ‘energy gap’ rather than the number of unknowns.

Memory: Similar order of memory requirements

FEM: Stiffness matrix requires O(N) floats (sparse structure)

Box: Graph adjacency requires O(N) floats (#Edges for Truss and AD-problem)

Utility: Box algorithm seems more advantageous for certain problems:

- It does not need gradient estimations or inversion so there is no problem of ill-conditioning
- It may be easier to navigate non-convex energy manifolds using this method. However, further development of algorithm is needed for completely spanning the solution space.

Moving to gate-based computing

Quantum Approximate Optimization Algorithm (QAOA)

Introduce a gate-based quantum algorithm that produces approximate solutions for Ising hamiltonians.

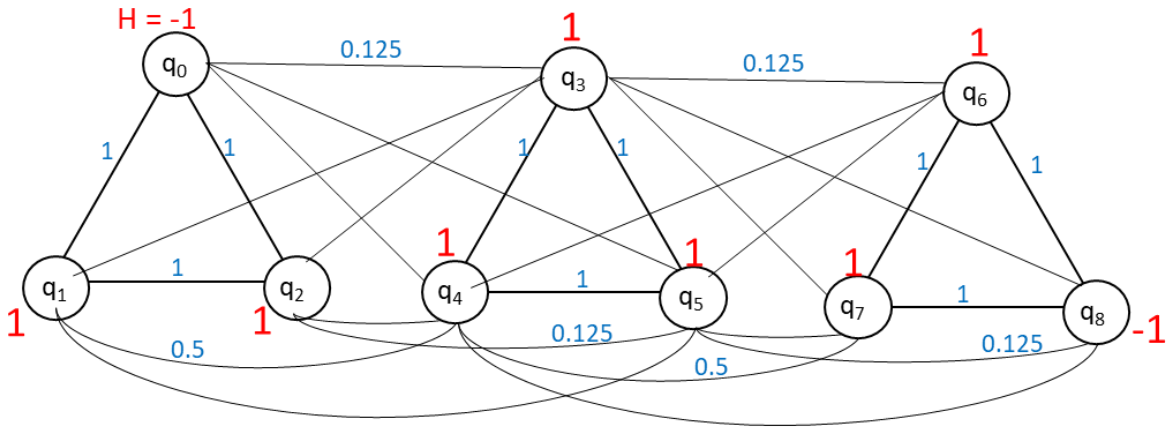
Example: Truss problem

$$F = \frac{1}{2}\int_0^1 u'^2$$

- Variable transformation from -1/+1 (D-Wave) to 0/1 (Quantum Assembly Language)

$$H^{new} = 2H - 2\sum J$$

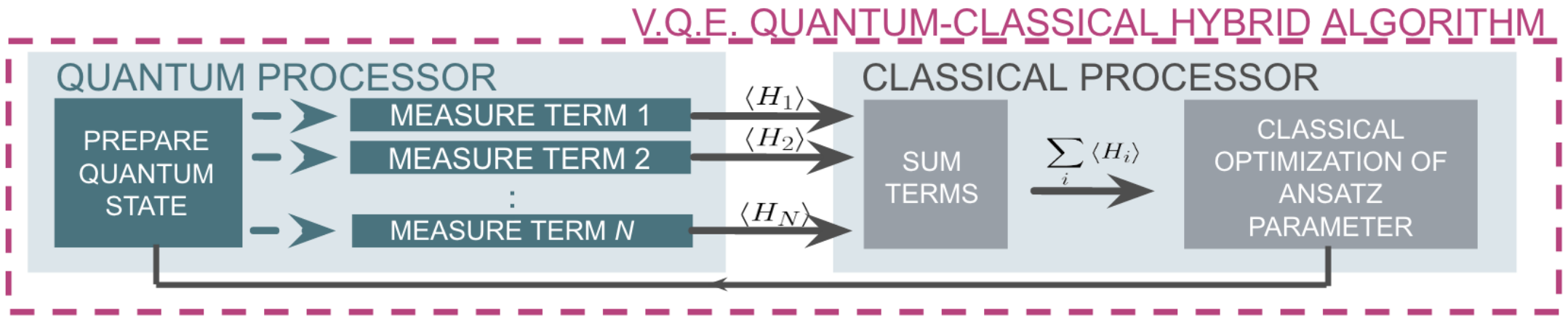
$$J^{new} = 4J$$



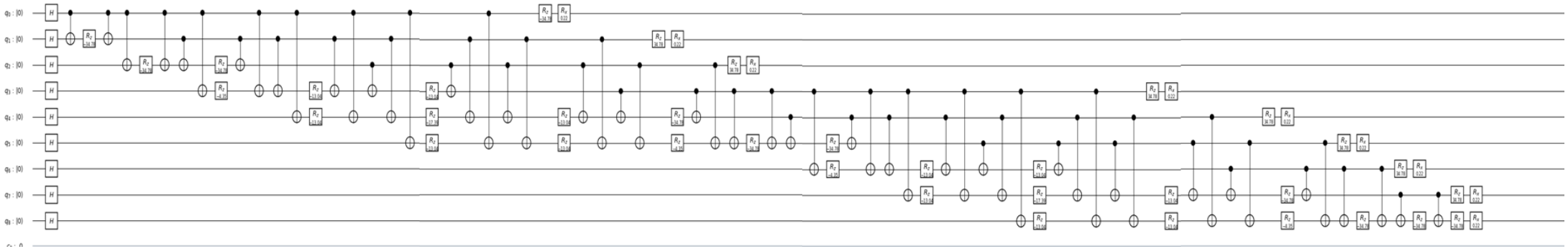
All unmarked links have a weight of 0.375

Farhi, Edward, Jeffrey Goldstone, and Sam Gutmann. "A quantum approximate optimization algorithm." arXiv preprint arXiv:1411.4028 (2014).

Moving to gate-based computing



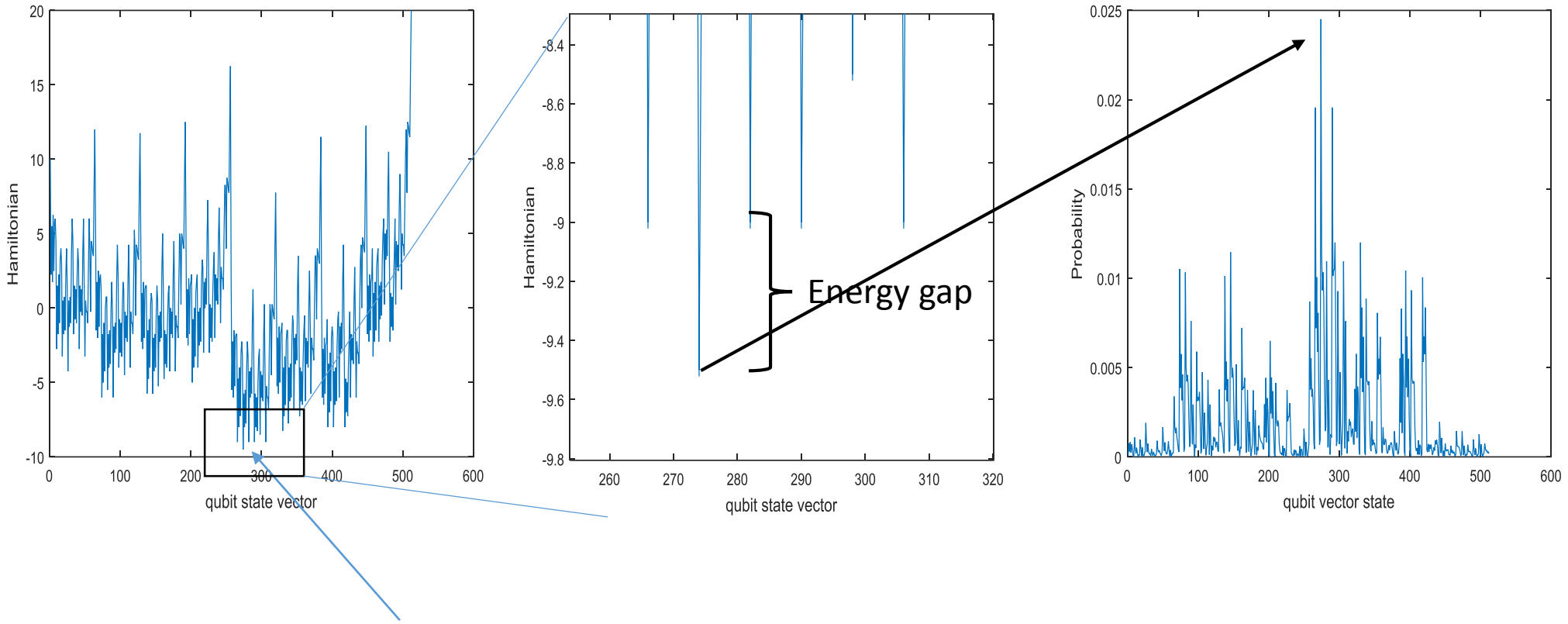
Quantum Assembly Language (**QASM**) based circuit is generated and solved using Qiskit



Farhi, Edward, Jeffrey Goldstone, and Sam Gutmann. "A quantum approximate optimization algorithm." arXiv preprint arXiv:1411.4028 (2014).

Moving to gate-based computing

Solution statistics



Unique minima at [1 0 0 0 1 0 0 0 1]

Thank you