21th EU/ME meeting x Quantum School

Emerging optimization methods: from metaheuristics to quantum approaches

DE LA RECHERCHE À L'INDUSTRIE

Discussions about High-Quality Embeddings on Quantum Annealers (WIP)

Valentin GILBERT[1] (valentin.gilbert@cea.fr), Julien RODRIGUEZ[1,2] (julien.rodriguez@cea.fr)

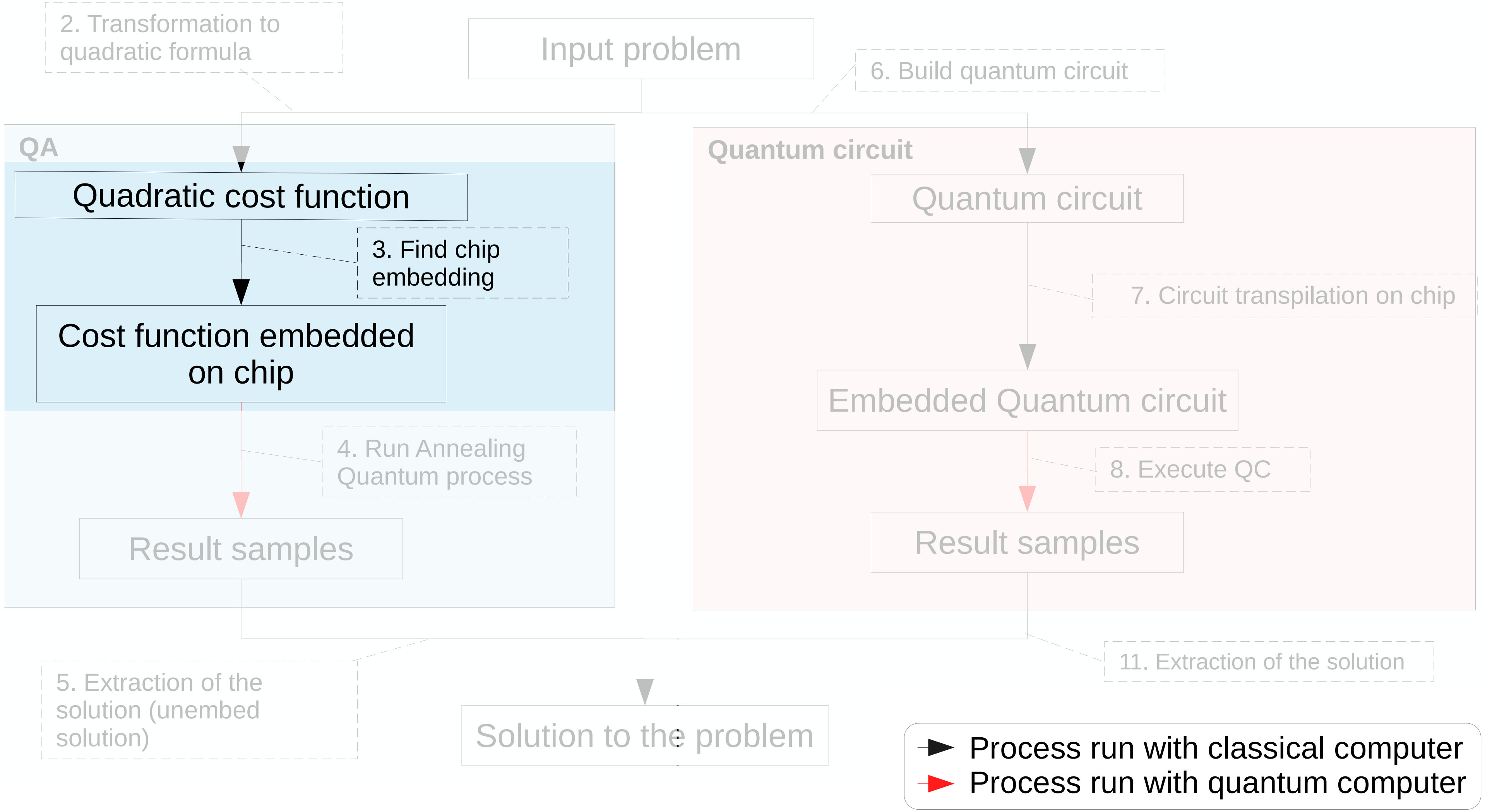
[1] Université Paris-Saclay, CEA, List, France

[2] Université de Bordeaux, Inria

- **Quantum computing is based on 2 main principles:**
  - Quantum superposition
  - Interference

- **What is limiting the Quantum Advantage:**
  - Quantum noise
  - Quantum chip topologies (require qubit mapping (QA) or swapping strategies (QC))

- **Overview:**
  - The Minor-embedding problem and its context
  - Existing work
  - Proposition of first experiments
  - Perspectives

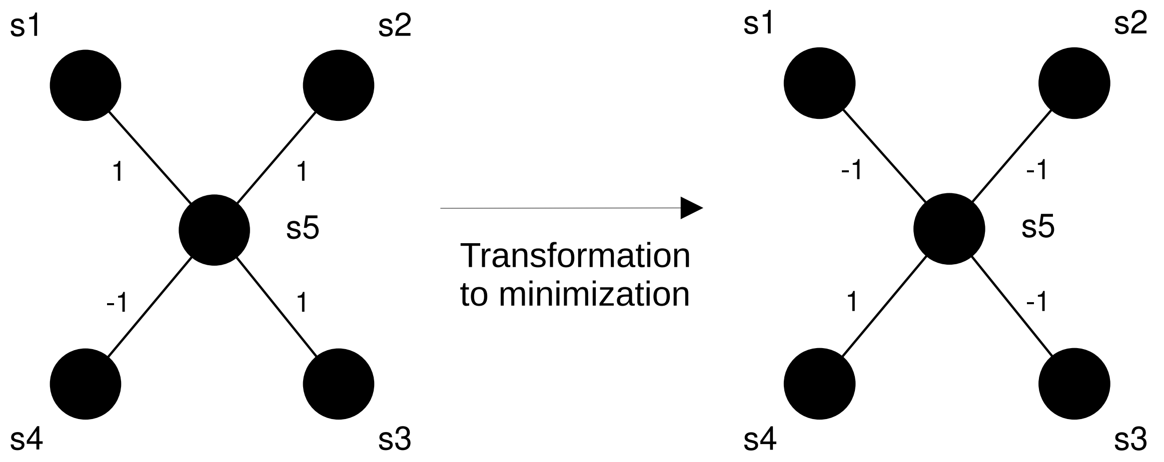

2. Transformation to quadratic formula
Input problem
6. Build quantum circuit
QA
Quadratic cost function
3. Find chip embedding
Cost function embedded on chip
4. Run Annealing Quantum process
Result samples
Quantum circuit
Quantum circuit
7. Circuit transpilation on chip
Embedded Quantum circuit
8. Execute QC
Result samples
11. Extraction of the solution
5. Extraction of the solution (unembed solution)
Solution to the problem
Process run with classical computer
Process run with quantum computer

- **Ising cost function: Minimization of a quadratic cost function**

$$\text{Minimize} \ -\sum_{i=0}^{n} h_i s_i - \sum_{i<j} J_{ij} s_i s_j$$

$$s_i, s_j \in \{-1, +1\} \text{ and } h_i, J_{ij} \in \mathbb{R}$$

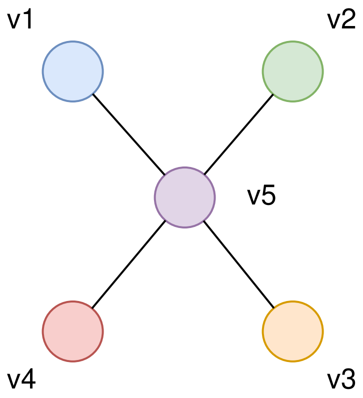
- **Ising problem formulation to solve Max-Cut problem**



$$\text{Minimize} \ -(-s_1 s_5 - s_2 s_5 - s_3 s_5 + s_4 s_5)$$

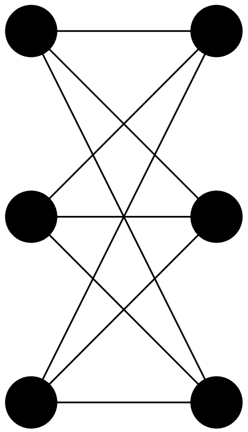
II- Graph Minor Theory

- **Graph minor N. Robertson et al. [1]**



$G_s = (V_s, E_s)$

$\phi : V_s \to V_t \times V_t$



$G_t = (V_t, E_t)$

- **Rules for the graph minor:**

1. Each vertex $v \in V_s$ is mapped onto a connected subgraph $\phi(v)$ of V_t
2. Each connected subgraph must be vertex disjoint: $\phi(v) \cap \phi(v') = \emptyset$ for $v \neq v'$
3. $\forall (u, v) \in E_s, \exists u' \in \phi(u), \exists v' \in \phi(v)$ such that $(u', v') \in E_t$

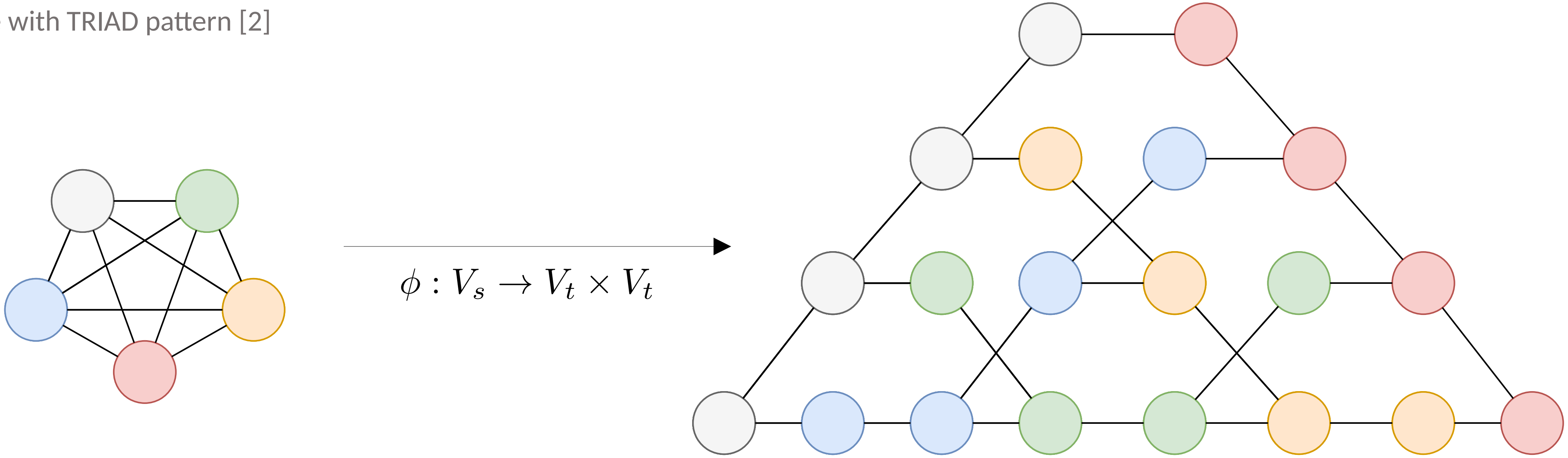
- **Limitations & Complexity [1]**
  - For fixed G_s finding the minor embedding in G_t has a polynomial complexity: $O(|V_s|^3)$

    BUT
  - The **algorithm runtime** is **exponential** in the **size of G_s**

- **Heuristics are required to solve this problem efficiently**
  - Mapping of cliques on regular graphs
  - Mapping of random source graphs on random target graphs
  - Other methods

Existing Approaches to Minor-embed Graphs

- **Mapping of cliques with near-optimal patterns [2, 3]**
  - Example with TRIAD pattern [2]

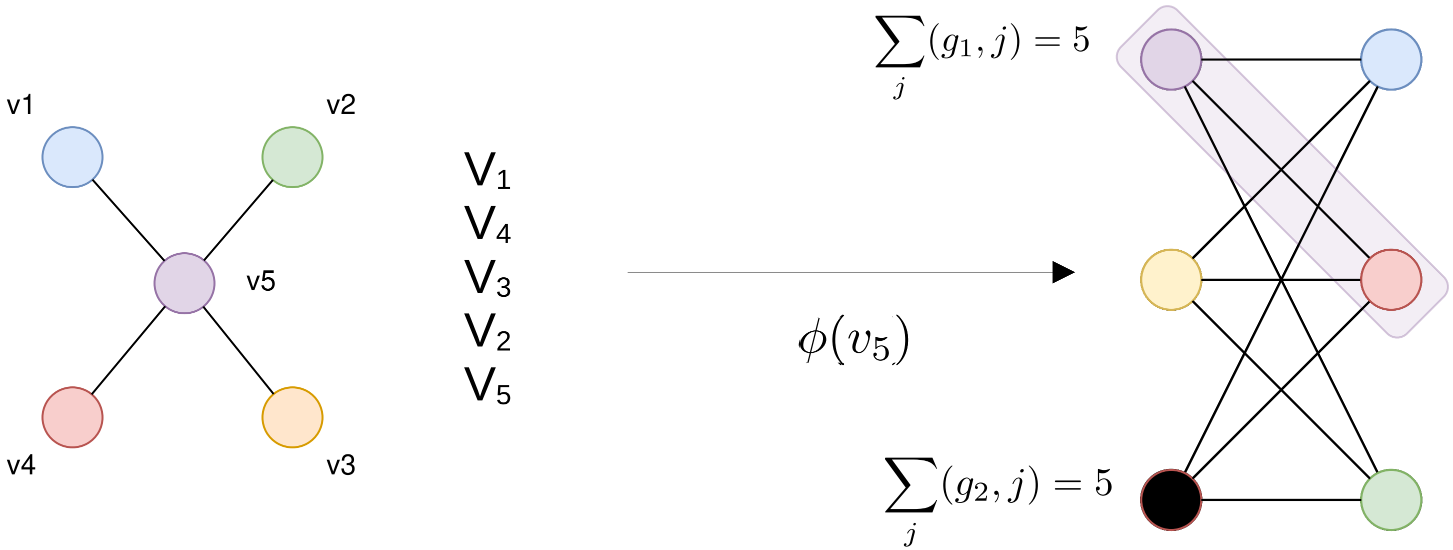


  - Strength:
    - Near optimal encoding for dense graphs
  - Weaknesses:
    - Does not always consider defective qubits
    - Not adapted for sparse graphs

- **Algorithm of J. Cai et al [4, 5]**

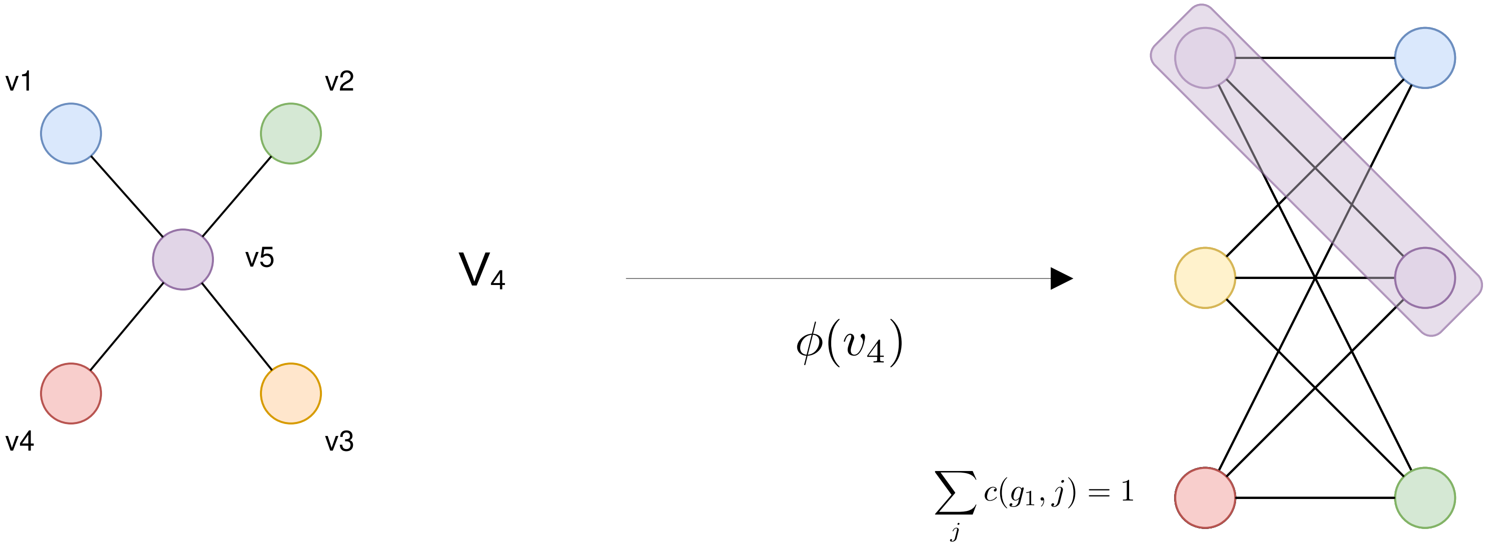
Stage 1: Initialization

- Set the vertices in random order.
- For each vertex, find the set of vertices in the target graph that minimize the weighted shortest path distance to its neighbours (using Dijkstra's algorithm).



Stage 2: Refinement

- Go through the vertices of G_s, remove its mapping and try to find another one, minimizing the weighted shortest path distance of the whole mapping to its neighbors.



$\sum_j c(g_2, j) = D^p$

D: diameter

p: number of vertex model passing by g_2

- Strength:
  - Work with any kind of target graph
- Weaknesses:
  - Costly algorithm due to the computation of the shortest distance path at each optimization step: $O(n^3 \log n)$
  - Does not work well for dense graphs

Existing Approaches to minor-embed graphs

- **Other algorithms:**
  - Extensions to the algorithm of J. Cai et al. :
    - Layout-aware minor-embedding [6]
    - Clique-based minor-embedding [7]
  - SA-based approaches [8]
    - Starts with a clique near-optimal encoding (like TRIAD)
    - Run a guided simulated annealing to reduce the number of vertices.

- **Objectives of current methods:**
  - Minimization of the number of qubits
  - Minimize the maximum chain length

- **Consequences of the qubit mapping**
  - Increases the number of qubits (qubit duplications)
  - Changes the required precision of couplings and auto-couplings (automated rescaling of weights).

    For D-Wave system Advantage6.1: $-1 \leq J_{ij} \leq 1$ $-2 \leq h_i \leq 2$

$$\text{Minimize } -\sum_{i=0}^{n}(h_i + \delta h_i)s_i - \sum_{i<j}(J_{ij} + \delta J_{ij})s_i s_j$$

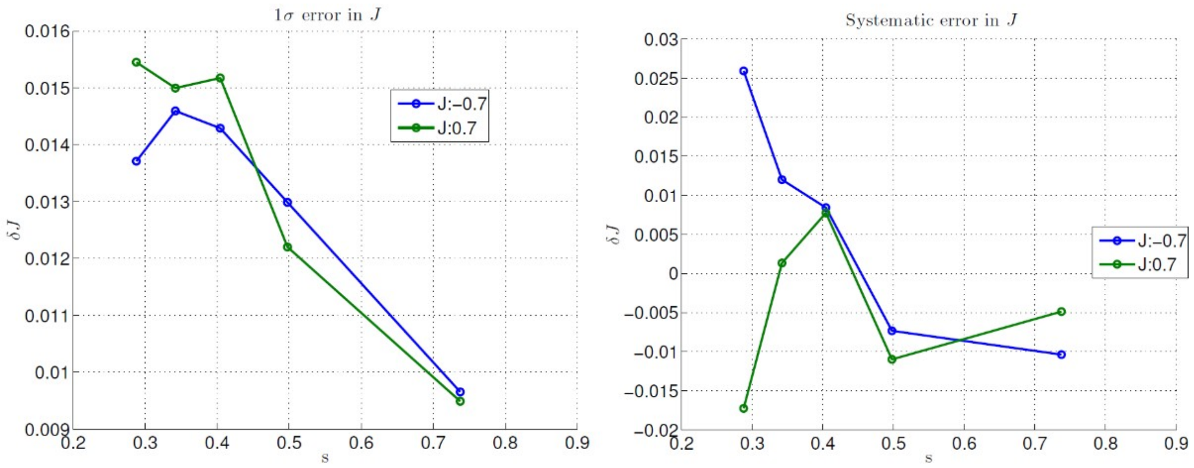


Figure taken from [9]. It Represents the standard deviation (left) and mean error rate (right) of J_ij coefficients with respect to arbitrary unit of annealing time.

  - Potentially changes the spectral gap of the problem (i.e., increases or decreases the time to solve the problem)

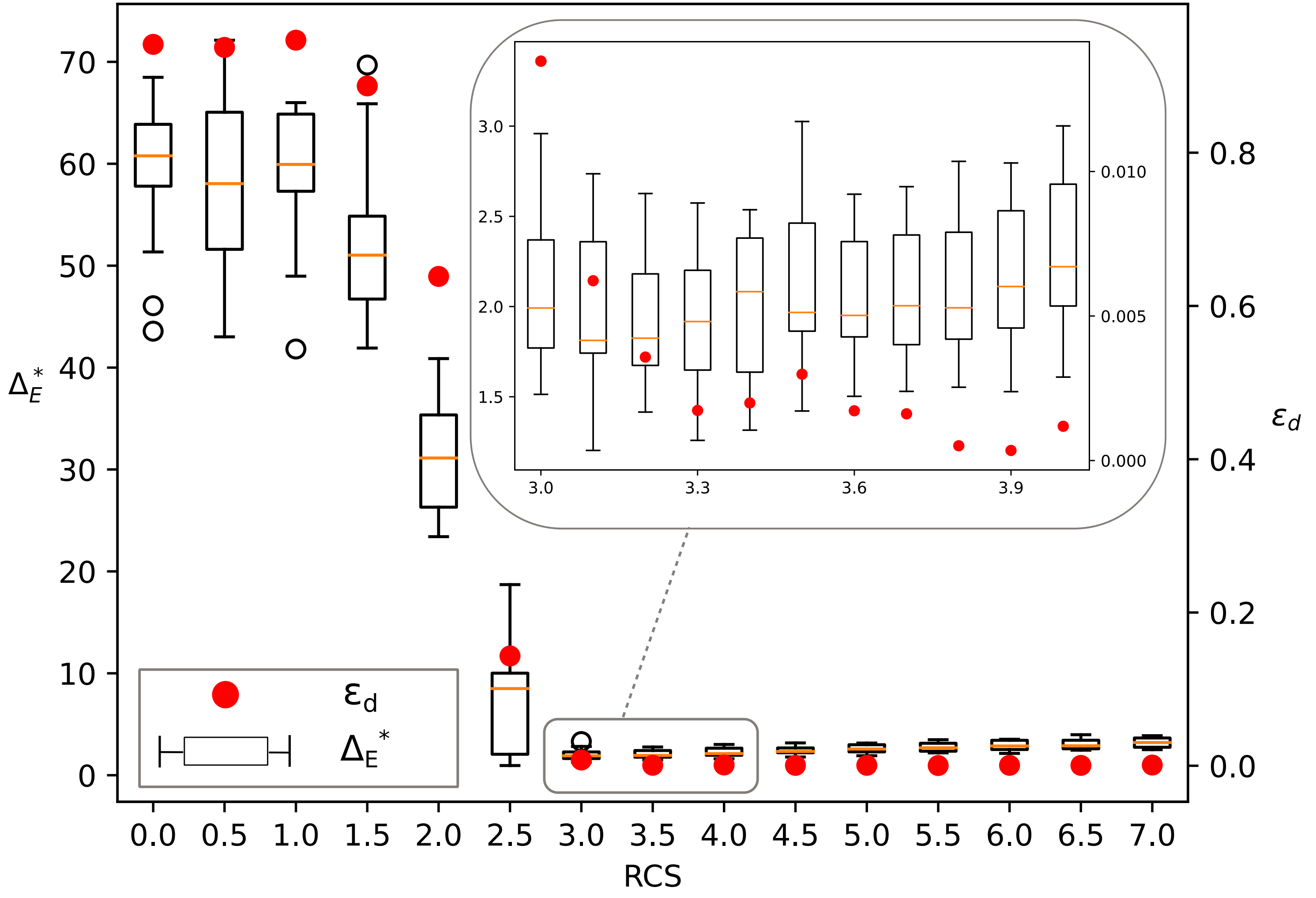
- **Phase transition of the energy gap considering the chain strength [2]**

Experimental determination of the optimal global chain strength (cs) and the Relative Chain Strength factor (RCS) [10]

$$cs = RCS \times max(\{h_i\} \cup \{J_{ij}\})$$

ϵ_d : qubit duplication error rate after the (less is better)

Δ_E^* : energy gap between optimal and the solution obtained on D-Wave systems (less is better)



Premises to define High-Quality embeddings

- **What is considered High-Quality embedding ?**
  - Minimization of the number of qubits
  - Minimize the maximum chain length
- **What is the best structure for logical qubits ?**
  - Error propagation on logical chains starts at the boundaries of the chain.
  - Errors on logical qubits require less coupling strength when they form a clique [11]
  - What about other topologies as cycles, trees, etc. ?
- **Is there a maximum chain length that shouldn't be reached ?**
  - Definition of bounds on the maximal chain length (requiring extra costs) could be computed considering QA precision and weights distribution.
- **Does chains distribution impact solution finding ?**
  - Study of chains sparsity versus concentration over the chip.
  - Perform experiments over different chain length distributions.

Conclusion

- **Many pieces of algorithms exist for minor-embedding graphs**
  - Major contribution made by J. Cai et al. In 2014.
  - Combinations of heuristics start to exist.

- **Tips and advices owned during our experiments**
  - Chain strength value is crucial so it has to be chosen carefully.
  - When mapping dense problems, use clique embedding methods.
  - Experiments should be done using as many qubits as possible to maximize the bias induced by imperfect couplers.

- **Perspectives**
  - Realization of the listed experiments.
  - Define metrics and bounds to quantify the quality of the embedding.
  - Depending on the result, design of a new heuristic.
  - Benchmarking the heuristic with state of the art methods.

Bibliography

- **[1]** N. Robertson and P. Seymour. **“Graph minors .XIII. The disjoint paths problem”**. Journal of Combinatorial Theory, Series B, 63(1):65–110, Jan. 1995.
- **[2]** V. Choi. **“Minor-embedding in adiabatic quantum computation”** Ii. minor-universal graph design. Quantum Information Processing, 10(3):343–353, 2011.
- **[3]** C. Klymko, B. D. Sullivan, and T. S. Humble. **“Adiabatic quantum programming: minor embedding with hard faults.”** Quantum information processing, 13:709–729, 2014.
- **[4]** J. Cai, W. G. Macready, and A. Roy. **“A practical heuristic for finding graph minors.”** ArXiv preprint arXiv:1406.2741, 2014.
- **[5]** A. R. et al. **Minorminer**. https://github.com/dwavesystems/minorminer, 2023.
- **[6]** J. P. Pinilla and S. J. E. Wilton. **“Layout-aware embedding for quantum annealing processors”**. In: Lecture Notes in Computer Science, pages 121–139. Springer International Publishing, 2019.
- **[7]** S. Zbinden, A. B̈artschi, H. Djidjev, and S. Eidenbenz. **“Embedding algorithms for quantum annealers with chimera and pegasus connection topologies”.** In: Lecture Notes in Computer Science, pages 187–206. Springer International Publishing, 2020

Bibliography

- **[8]** Y. Sugie, et al. **“Minor-embedding heuristics for large-scale annealing processors with sparse hardware graphs of up to 102, 400 nodes”**. Soft Computing, 25(3):1731–1749, Jan. 2021.
- **[9]** D-Wave System Inc. **“Error Sources for Problem Representation”** https://docs.dwavesys.com/docs/latest/c_qpu_ice.html
- **[10]** Willsch, D., et al. **“Benchmarking advantage and d-wave 2000q quantum annealers with exact cover problems.”** Quantum Information Processing 21(4), 1–22 (2022)
- **[11]** E. Pelofske. **“4-clique network minor embedding for quantum annealers”**. arXiv preprint arXiv:2301.08807, 2023.

Thank you !

Introduction : AQC (Adiabatic Quantum Computing)

- **AQC, a continuous interpolation of 2 time independent Hamiltonians:**

$$H(t) = \left(1 - \frac{t}{T}\right) H_M + \frac{t}{T} H_C$$

- **The adiabatic theorem:**

  "If a quantum state is initialized in the ground state of the Hamiltonian H_M, and that t varies slowly enough between 0 and T, the quantum state will state close to the ground state of H(t)"

- **Quantum Annealing, a noisy version of AQC:**