Graph Coloring with Neutral Atoms

Experience and Lessons Learned from the QOPS project

ANDREA SCARABOSIO, PAOLO VIVIANI, CHIARA VERCELLINO – LINKS FOUNDATION

18TH ADVANCED SCHOOL ON PARALLEL COMPUTING: QUANTUM COMPUTING
CINECA, FEB 2022

Context

QOPS Iscra project

- Collaboration with CINECA and PASQAL
- Project started in March 2021 and is ongoing
- Goal is to experiment with Neutral Atoms architectures for combinatorial optimization
- Initially involved experiments on emulators, we are currently getting ready to run on QPUs

FONDAZIONE links PASSION FOR INNOVATION

CINECA

PASQAL

Background: PCI assignment problem

Optimization problem

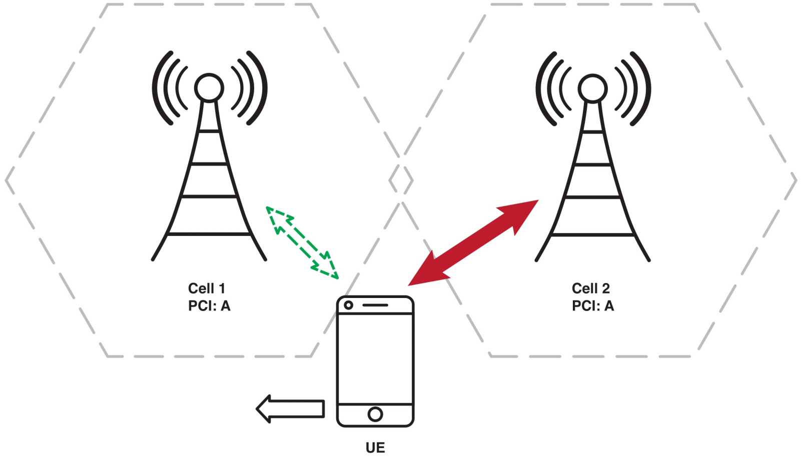
Physical Cell Identifiers (PCIs) problem

Assign PCIs “label” to the nodes of a cellular communication network

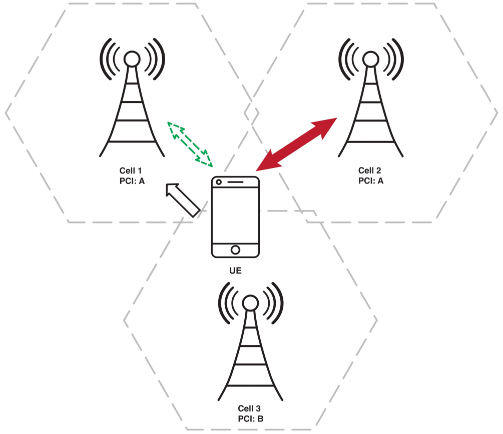
- **Goal:** maximise data throughput by avoiding conflicts and confusion between nodes and end terminal (cell phone)
- **Constraints:** PCIs are limited to 504 in 4G (1008 in 5G). Networks can be very dense (cities)!

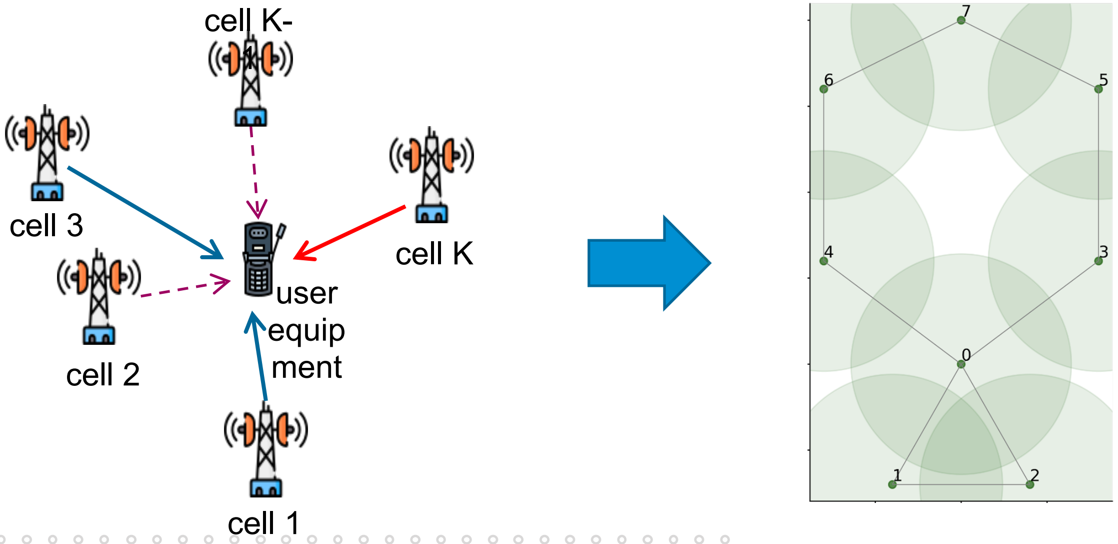
"Naive” PCI assignment translate easily to a graph format and to one or more instances of **Graph Coloring (GC) problem** an example of constraint satisfaction problem (CSP)

Nodes collision



Nodes confusion

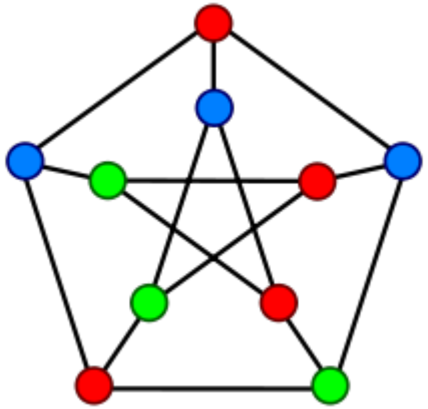




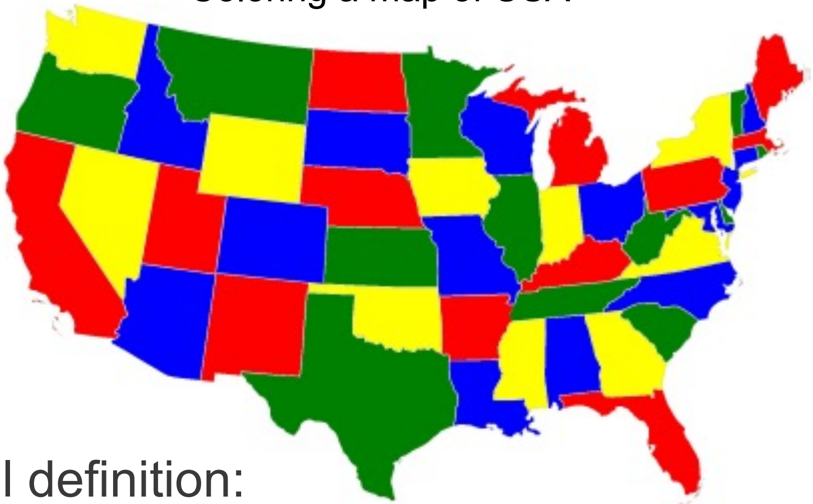
Graph colouring: definition

- GC definition: **coloring** of a graph is a labeling of the graph's vertices with (k) colors such that no two vertices sharing the same edge have the same color.
- The smallest number of colors needed to color a graph *G* is called its **chromatic number** and is often denoted χ(G).
- Strictly related to a MIS problem: a subset of vertices assigned to the same color is called a *color class*, every such class forms an independent set. Thus, a *k*-coloring is the same as a partition of the vertex set into *k* independent sets.
- k-colorable? ➔ NP-complete problem
- Find chromatic number ➔ NP-hard problem

Graph minimum colors = 3



Coloring a map of USA



- CSP mathematical definition:

  $X = \{X_1, \dots, X_n\}$ is a set of variables
  $D = \{D_1, \dots, D_n\}$ is a set domain values
  $C = \{C_1, \dots, C_n\}$ is a set of constraints

- A solution includes al variables (complete) and satisfy all constraints (consistent) ↔ minimise cost C

$$C = -\sum_{i=1}^{n}\sum_{j=1}^{n} A_{i,j}\left(1-\delta_{i,j}\right) = 0$$

Graph colouring: one-hot encoding

- One hot-encoding (reminder, see [1]):
  1. For all *n* nodes and *k* colors we have *n*k* binary variable $x_{i,j}$, i=1,...,*n, j=1,...,k* (1 if node (cell) is assigned color j, 0 otherwise)
  2. For each node an assignement constraint equation $\sum_{j=1,..k} x_{i,j} = 1$ (one and only one color must be assigned to each node)
  3. For each couple of connected vertices, a set of adjacency constraints $x_{ij} + x_{mj} \leq 1$, j=1,...,k_{colors} (no conflicts)
- "Unrolling" unknown matrix **x** and transforming constraints into quadratic penalties using known transformation rules [1] leads to a QUBO (unconstrained) format min. x^tQx:
  1. Ax = b ➔ $P_1(Ax = b)^t * (Ax = b)$ for assignment constraint
  2. $(x_i + x_j \leq 1)$ ➔ $P_2(x_i\ x_j)$ for adjacency constraints
- One-hot encoding is very simple but requires many variables ↔ qubits

[1] F. Glover, G. Kochenberger, Y. Du, 'A Tutorial on Formulating and Using QUBO Models', 2019, https://arxiv.org/ftp/arxiv/papers/1811/1811.11538.pdf

Background: PCI assignment problem in real

Optimization problem

- **Physical Cell Identifiers (PCIs) problem in reality is somewhat more complicated:**
  - Two types of identifier, a primary y=[0-2] and secondary x=[0,167] numbers ➔ z = x + 3*y [0-503]
  - Adjacency constraints can be "hard" or "soft" and violation cost can be different for "x" and "y" violation ➔ total cost C = $\mathbf{x}^t \cdot \mathbf{C}_x \cdot \mathbf{x} + \mathbf{y}^t \cdot \mathbf{C}_y \cdot \mathbf{y}$
  - Nodes (antenna) are clustered into sites with variable number of nodes (often 3)
- Specific PCIs assignment problem changes with provider but all can be formulated ad GC problem eventually with additional (linear) constraints
- **QUBO formulation of PCI problem is very effective, network of thousands of nodes can be successfully optimised with purely classical and hybrid quantum-classical algorithm**

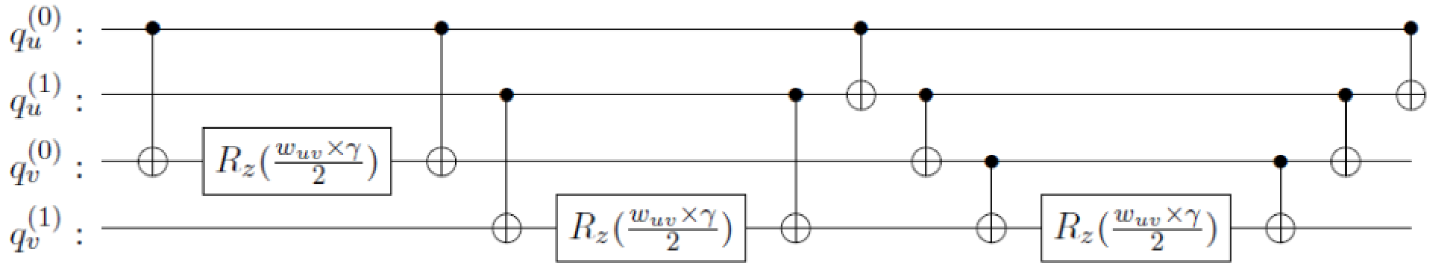
Graph colouring: alternative encoding

hot-encoding vs binary

- One-hot (or unary) encoding uses many qubits, nowaday a sparse resource. In a gate model QC other options are available for encoding GC (and other optimisation problems).
- For ex. binary encoding for integer $k = 2^l$ where at each node are associated only *l* qbits. Note that we need *nl* ~ *n* ln(*k)* qubits to encode the entire coloring of the graph..
- Cost operator is non quadratic but up to 2*l*-body interaction is required[1]:

$$C = -\sum_{u=1}^{n}\sum_{v=1}^{n} A_{u,v}(1-\delta_{u,v}) \xRightarrow{\mathit{classical\ to\ quantum}} -\sum_{1\leq u\leq v\leq n} A_{u,v}\left(1-\frac{1+\sigma_u^0\sigma_v^0}{2}\dots\frac{1+\sigma_u^{l-1}\sigma_v^{l-1}}{2}\right)$$

- Interaction terms involving more than 2-body operators are not directly implementable on most quantum computers. Instead, they can be decomposed as sums of two-body terms, which can be realized with CNOT gates

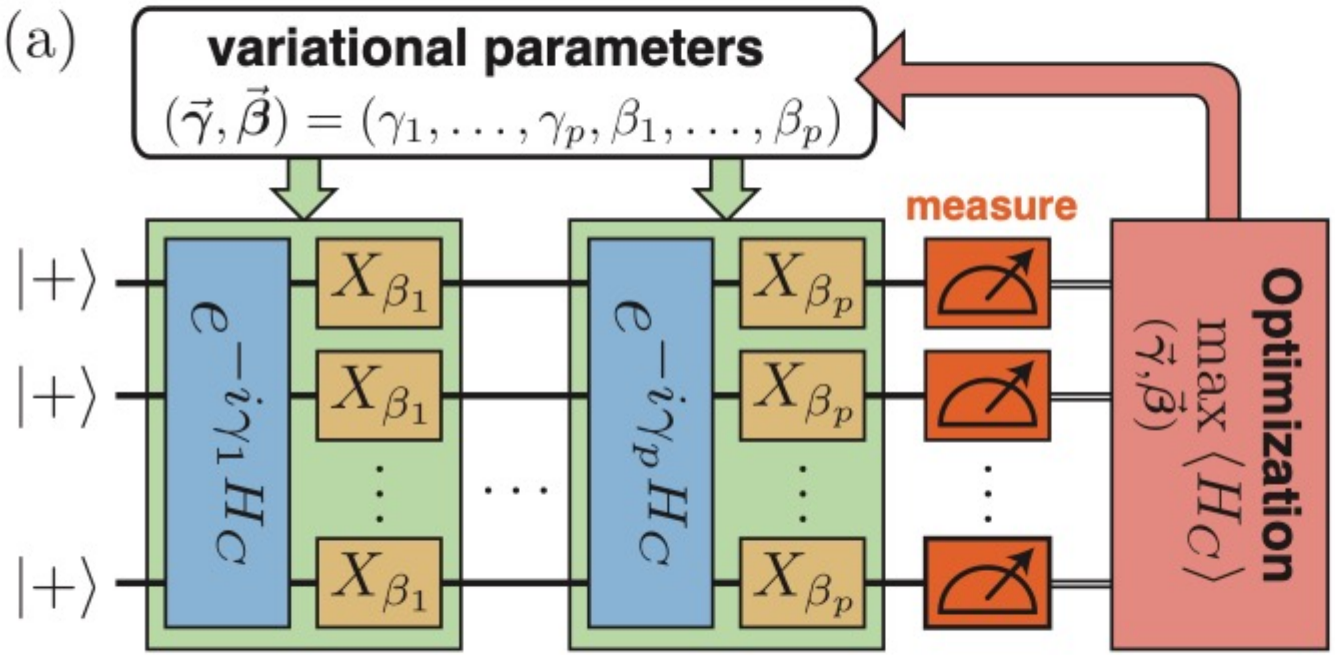

$q_u^{(0)}$:
$q_u^{(1)}$:
$q_v^{(0)}$:
$q_v^{(1)}$:
$R_z(\frac{w_{uv}\times\gamma}{2})$
$R_z(\frac{w_{uv}\times\gamma}{2})$
$R_z(\frac{w_{uv}\times\gamma}{2})$

[1] Dalyac et al. EPJ Quantum Technology (2021) 8:12, https://doi.org/10.1140/epjqt/s40507-021-00100-3

Optimization beyond annealers

QAOA

- Beside from quanutm annealers, other quantum paradigms can be used for optimization by means of **Quantum Approximate Optimization Algorithm (QAOA)**
- QAOA is an **Hybrid quantum-classical** algorithm
- QAOA requires the **cost Hamiltonian** to be applied on the qubit register
- Applying an arbitrary QUBO cost hamiltonian requires an **arbitrary connectivity among qubits** to encode off-diagonal values of the QUBO matrix



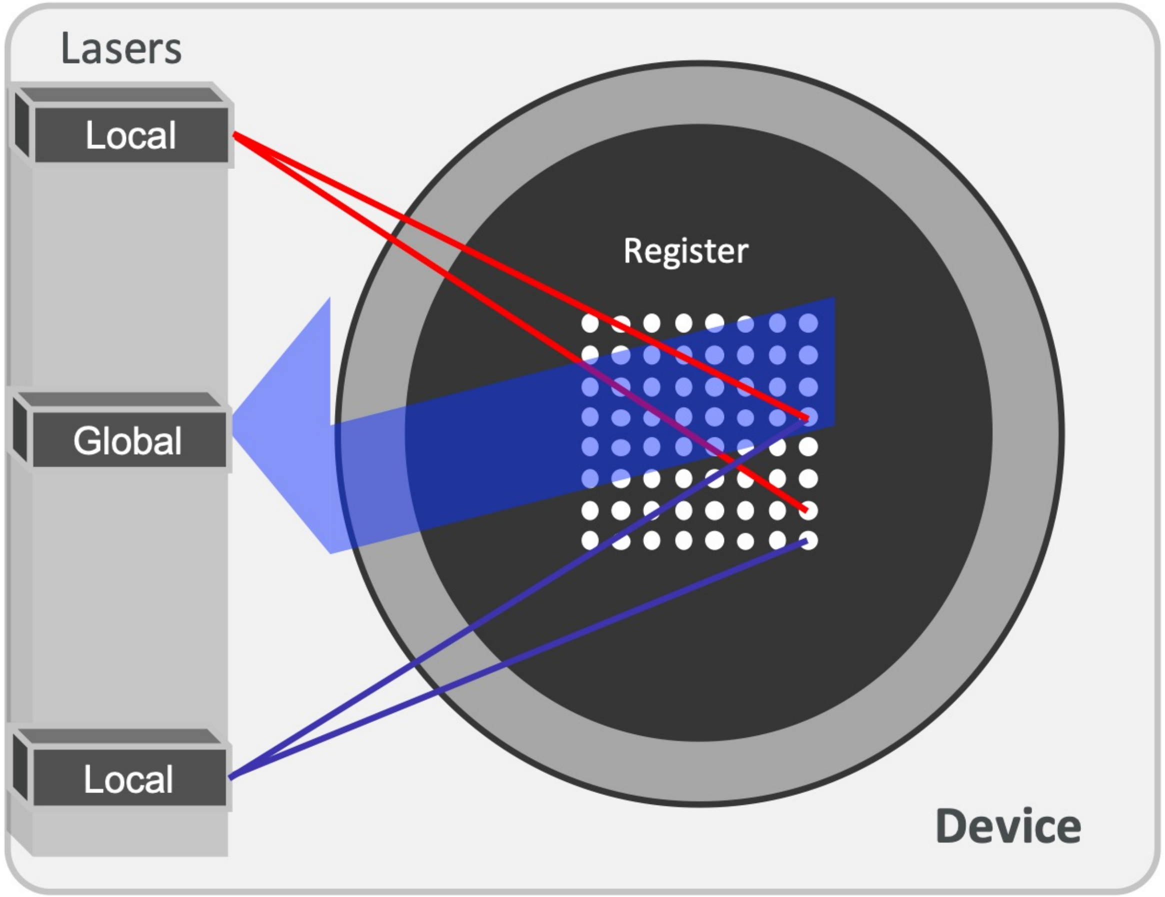
source: arxiv.org/abs/1812.01041

Optimization beyond annealers

Neutral Atoms

- Current set-up of the Pasqal machine allow for:
  - **Analog operations**
  - **Global Rydberg pulses**
  - **2D qubit arrays**

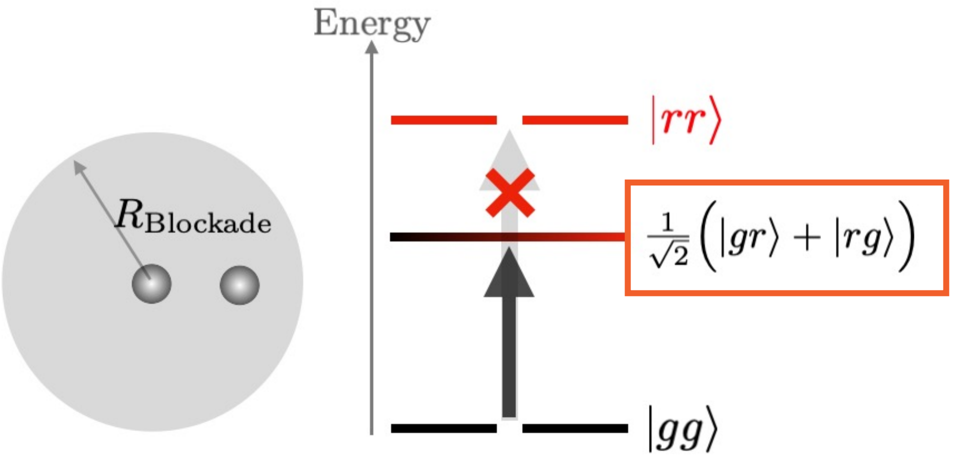
- This are the **tools we use to implement our cost Hamiltonian on a qubit register**



Rydberg blockade

Main tool for analog operation

- Pair-wise interactions between the atoms are regulated by the **Rydberg Blockade effect**
- Rydberg blockade prevents two atoms within radius R (correlated to laser frequency) to be both in the excited state
- This creates quantum **superposition and entanglement between atoms that are "connected"** (i.e. closer than R to each other).



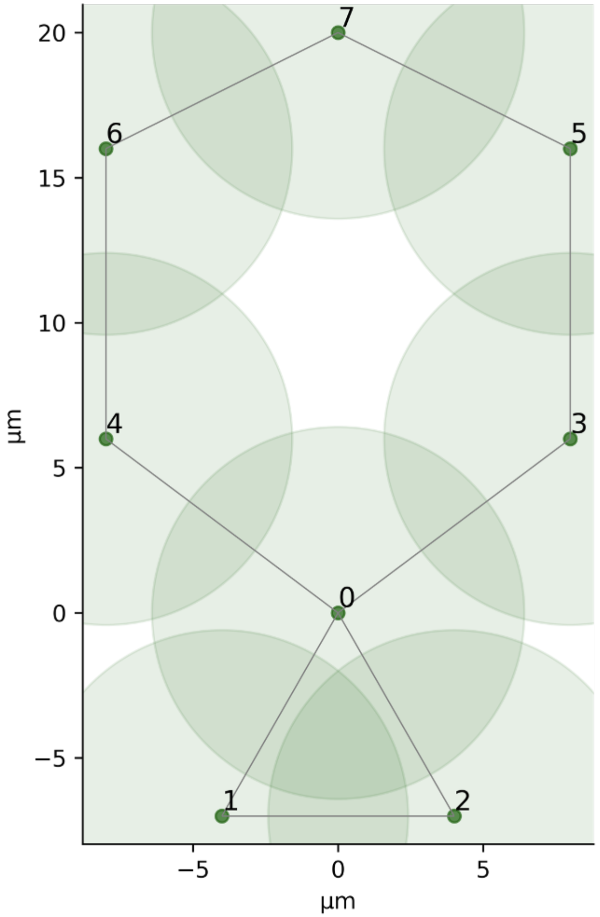
Neutral Atoms Hamiltonian

- The tools we introduced, allow us to create a **quantum register behaving as the Hamiltonian**

$$\mathcal{H} = \underbrace{\sum_{i=1}^{N} \frac{\hbar\Omega}{2}\sigma_i^x - \sum_{i=1}^{N} \frac{\hbar\delta}{2}\sigma_i^z}_{\text{diagonal terms}} + \underbrace{\sum_{j<i} \frac{C_6}{|r_i - r_j|^6} n_i n_j}_{\text{off-diagonal terms}}$$

- We know that we can map any arbitrary QUBO problem as an Ising Hamiltonian, but **we have constraints**
  - Global pulses force a **constant value for the QUBO diagonal terms**
    - **No problem for graph coloring**
  - **QUBO off-diagonal terms are limited** by the Rydberg-blockade interactions on a 2D plane

Unit-Disk Graphs

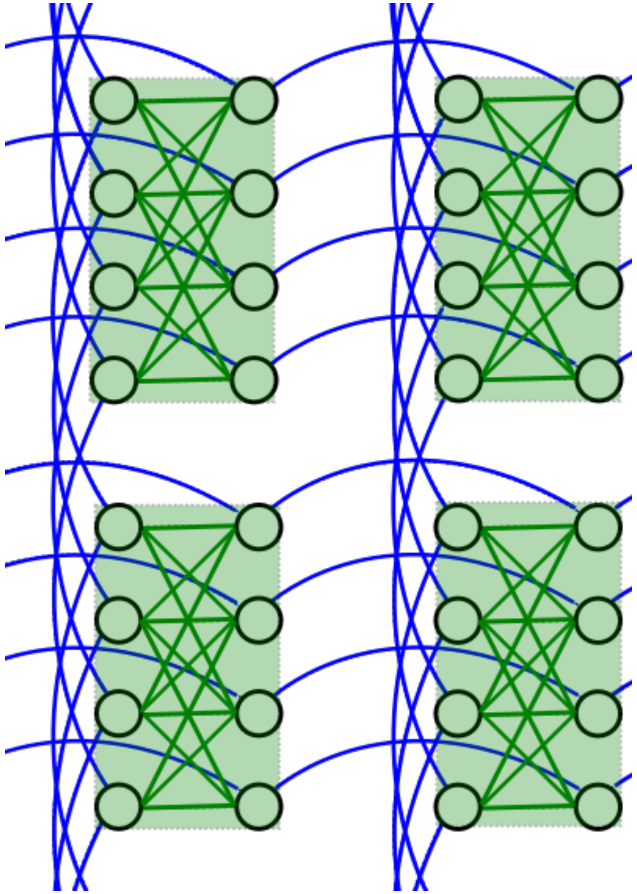


- Interactions between qubits of a register can be represented as a **Unit-Disk graph (UD)**
- A QUBO matrix can be seen as an adjacency matrix of a graph
- Mapping a QUBO problem to a Unit-Disk graph Hamiltonian involves the **embedding** of the graph induced by the QUBO matrix into a Unit-Disk graph
- Not all graphs can be represented as UD graphs
  - Checking if this embedding is feasible is **NP-hard**
  - Finding the explicit coordinates of the nodes is even harder
- In general this problem is **harder than the original problem**, and rarely feasible - https://arxiv.org/abs/2012.14859

Embedding problem in quantum computing

A quick detour

- Embedding into limited connectivity graphs is not new in quantum computing
- D-Wave have the same problem to map QUBO matrices onto chimera graphs
- Contraints are less stringent, so they found an efficient heuristic to achieve that
- We are also investigating that, a starting point can be looking to the algorithms used to plot graphs (i.e. Force-directed drawing)
- Also, auxiliary qubits can be used to build *chains* to connect qubits that are far away
  - D-Wave already does that, we are investigating also the possiblity to do this on Pasqal machines



So what?

How do we use Rydberg atoms to solve graph coloring?

We knew how to solve GC with a QUBO formulation:

- ✓ Binary variables through one-hot-encoding → qubits
- ✓ QUBO constant diagonal terms → ok with global pulses
- X QUBO off diagonal terms → arbitrary graph embedding into a UD graph
- X Number of optimization variables → Nodes * Colors, many qubits required

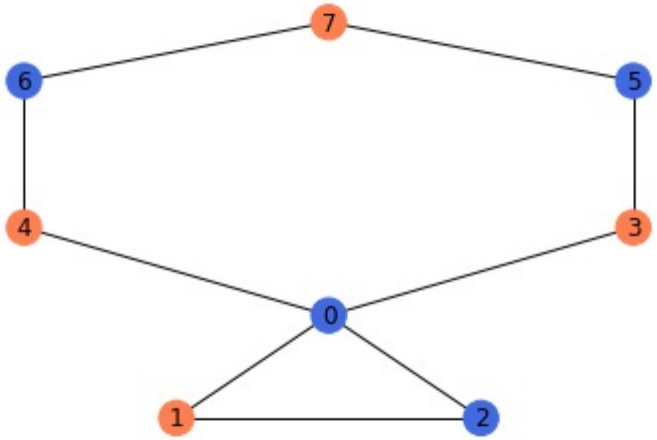
However, the **PCI problem is not an arbitrary QUBO, it has its own a geometric structure**:

- ✓ Cellular antennas can be drawn as a planar graph and their **interference pattern can be represented as a UD graph**.
- ✓ While we can't directly map graph coloring to a UD graph, we can try to start from a related problem
  - ✓ **Maximal Independent Set (MIS)**

Building Block: MIS

Definition

- An **independent set** is a subset of nodes in a graph such as no node is connected to another node in the subset
- **Maximal vs. Maximum**
  - Maximal means that you cannot add a node to the set without violating the independence constraint
  - Maximum means refers to the globally largest maximal independent set that can be identified on a given graph
- It can be described as an optimization problem with the following hamiltonian



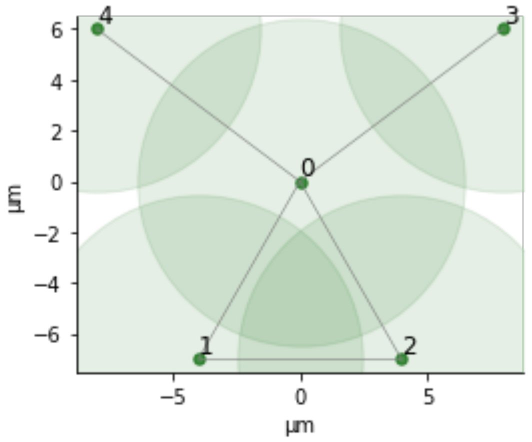
$$H_{mis} = -\sum_{i} n_i + U \sum_{\langle i,j \rangle} n_i n_j$$

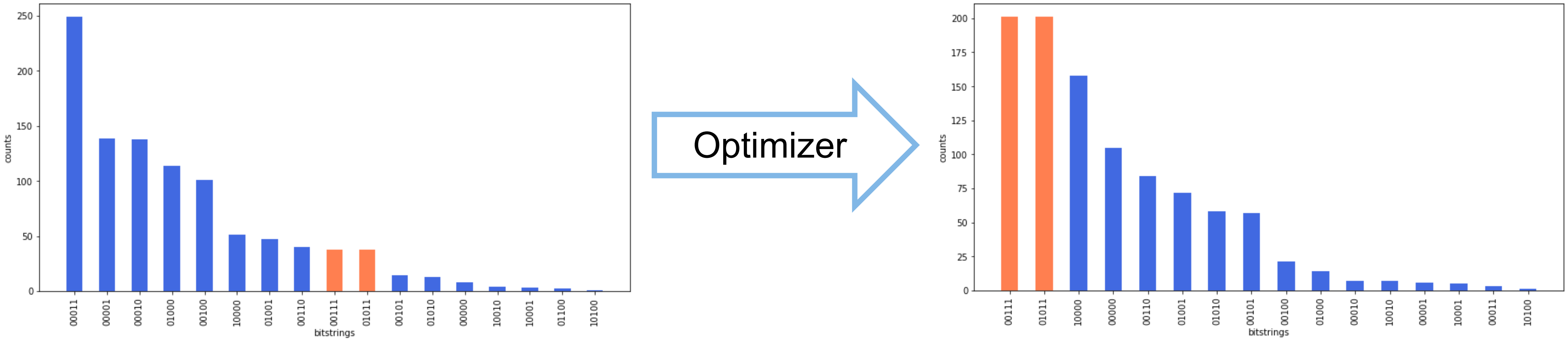
Building Block: MIS

MIS on Unit-Disk graphs

For 2D planar graphs, the **Hamiltonian of the MIS can be directly mapped to a Rydberg atoms array** respecting the constraint we mentioned before

- We can use QAOA to tune laser pulses, so that MIS solutions appear more frequently
- We leverage rydberg blockade to obtain a configuration of the register where the |1⟩ qubits represent the members of a MIS





MIS DEMO

https://github.com/LINKS-Foundation-CPE/cineca_aspc

PCI topological solution: Iterative MIS

From MIS to graph coloring

✓ MIS implementation through Neutral Atoms is straightforward given the antennas' positions (qubits positions) and the radius which determines conflicts (Rydberg radius):

- Scale antennas' positions to the dimensions of the physical device
- Compute Rydberg blockade radius corresponding to the desired antennas' conflicts distance
- Set pulses' parameters according to the Rydberg blockade radius

✓ An MIS solution provides a feasible color in a graph coloring problem

Iterative MIS ⇨ Graph coloring

✓ MIS approach is helpful:

- It requires only one binary variable (i.e., one qubit) for each vertex in the graph G → the number of variables (and qubits) needed is reduced at least by a factor of $\chi(G)$, i.e, chromatic number of G
- It solves the issue of finding a feasible embedding for the QUBO formulation of the graph coloring problems

PCI topological solution: Iterative MIS

GREEDY APPROACH (Greedy-itMIS)

Graph coloring problem is solved by iteratively solving MIS problems on induced subgraphs

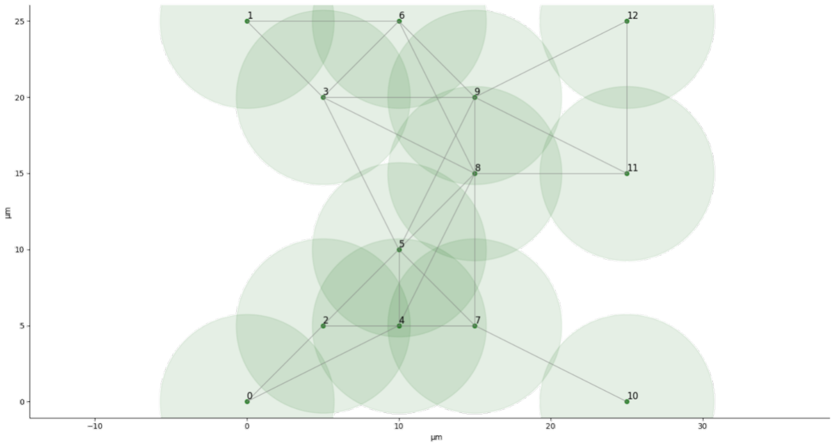
Greedy-itMIS algorithm:

- Find an MIS solution (with analog QAOA)
- Assign a color to the subset of graph’s vertexes and remove them ⇨ induced graph
- Iterate on the induced graph until all conflicts are solved, i.e., no edges remain in the induced subgraph

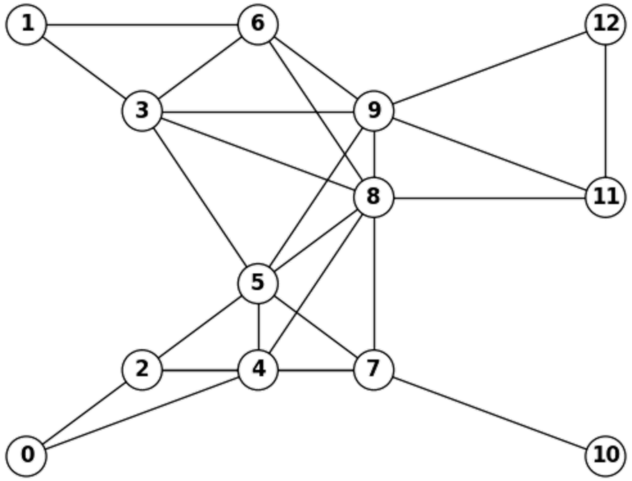
The solution, in this case, is feasible but typically suboptimal

PCI topological solution: Iterative MIS

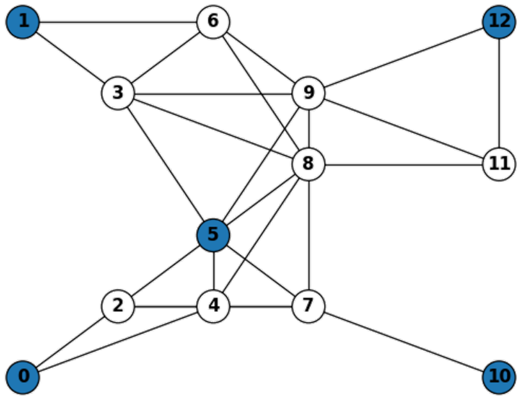
GREEDY APPROACH (Greedy-itMIS)



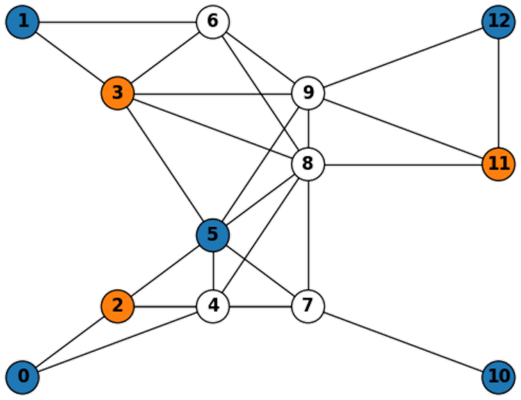
Iteration 0



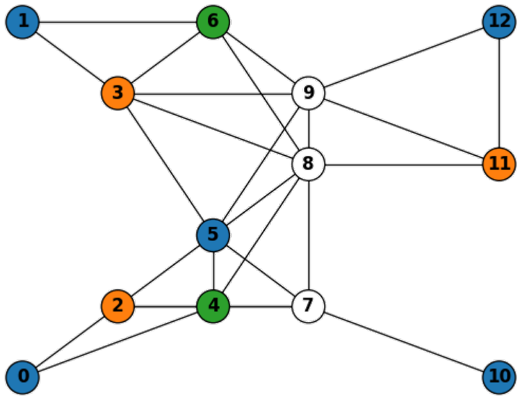
Iteration 1



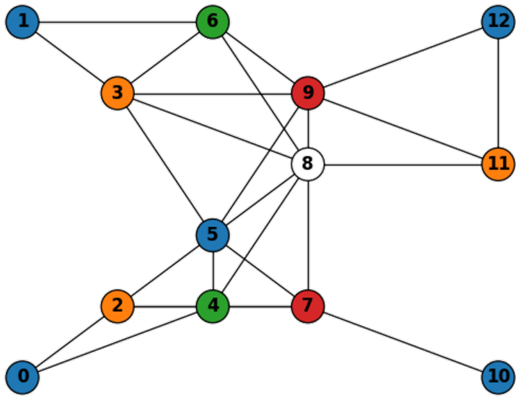
Iteration 2



Iteration 3



Iteration 4



PCI topological solution: Iterative MIS

BRANCH&BOUND APPROACH (BB-itMIS)

Theorem: Every graph G has an optimal coloring in which (at least) one of the colors is an MIS.

Greedy-itMIS issue: consider just one MIS solution at each iteration

⇨

BB-itMIS improvement: outer optimization loop that considers all MIS solutions at each iteration

Notation:

- Vertex = vertex of the graph G
- Node = node of the BB process
- MIS = Maximal Independent Set

PCI topological solution: Iterative MIS

BRANCH&BOUND APPROACH (BB-itMIS)

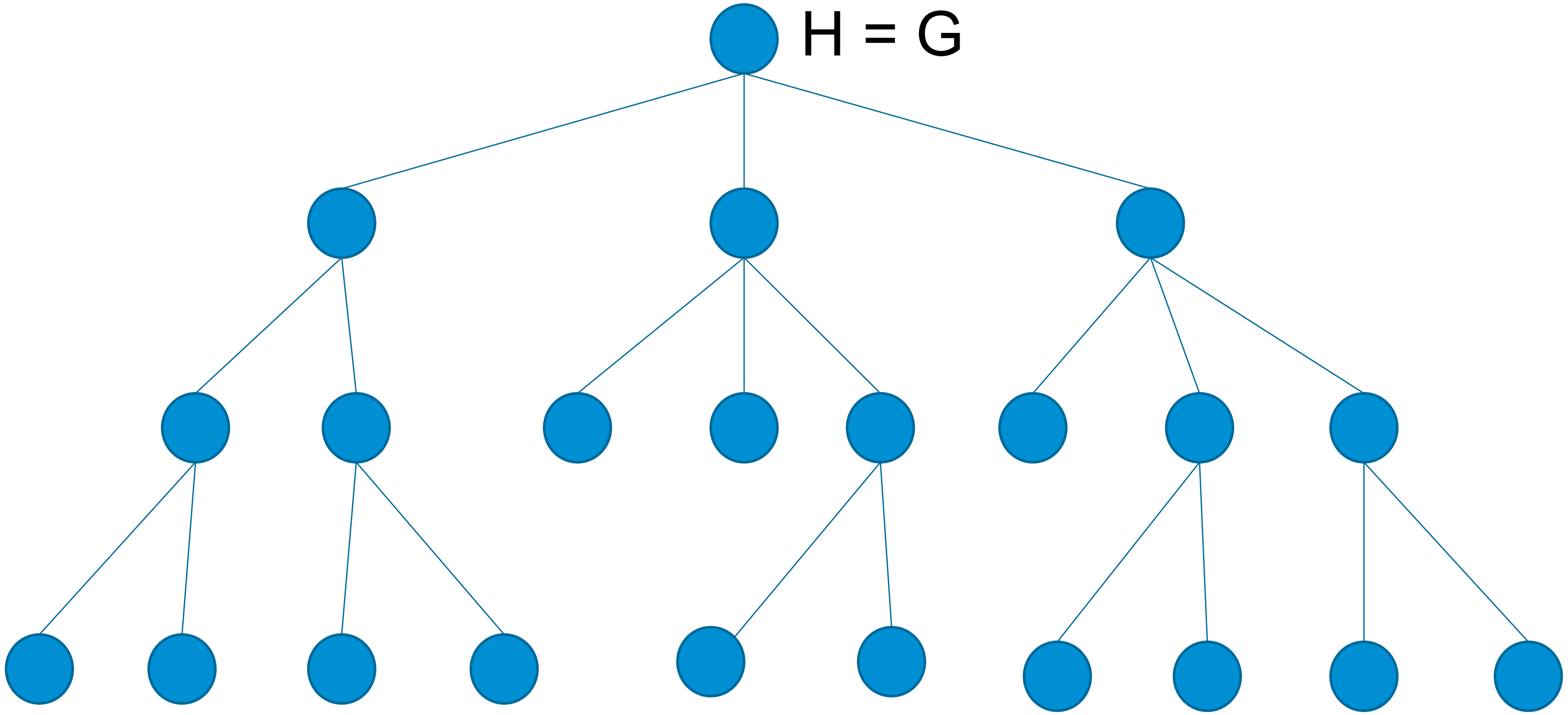
The objective of the optimization is still the number of colors used, but the BB approach concerns the MIS solutions found through QAOA, instead of the binary variables as it typically would happen in classical optimization.

BB-itMIS algorithm:

- Start with the whole graph G in the root node of the BB tree
- For each BB node find a set of MIS solutions and generate one branch for each solution
- Each BB node is associated with a subgraph of G obtained by removing all the vertexes of one MIS

PCI topological solution: Iterative MIS

BRANCH&BOUND APPROACH (BB-itMIS)



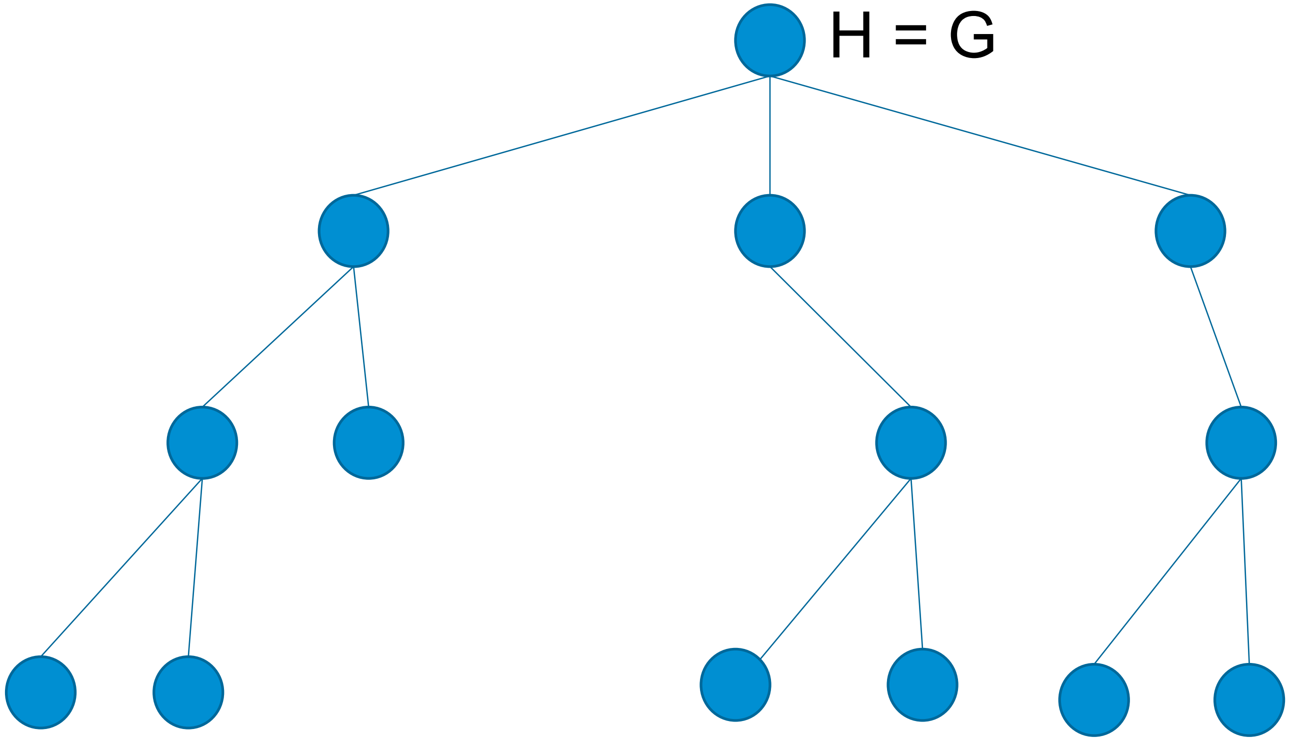
Each BB node has:

- **H**: induced subgraph of the original graph G
- **Feasible coloring**: MISs solutions of the parents nodes + greedy coloring
- **Obj. fun**: number of colors used in the feasible coloring
- **LB**: number of colors used + LB on chromatic number of H
- **UB**: upper bound on the chromatic number of H

greedy coloring: one color for each of the remaining vertexes in H
colors used: depth of the node, i.e. number of MISs sets yet removed

PCI topological solution: Iterative MIS

BRANCH&BOUND APPROACH (BB-itMIS)

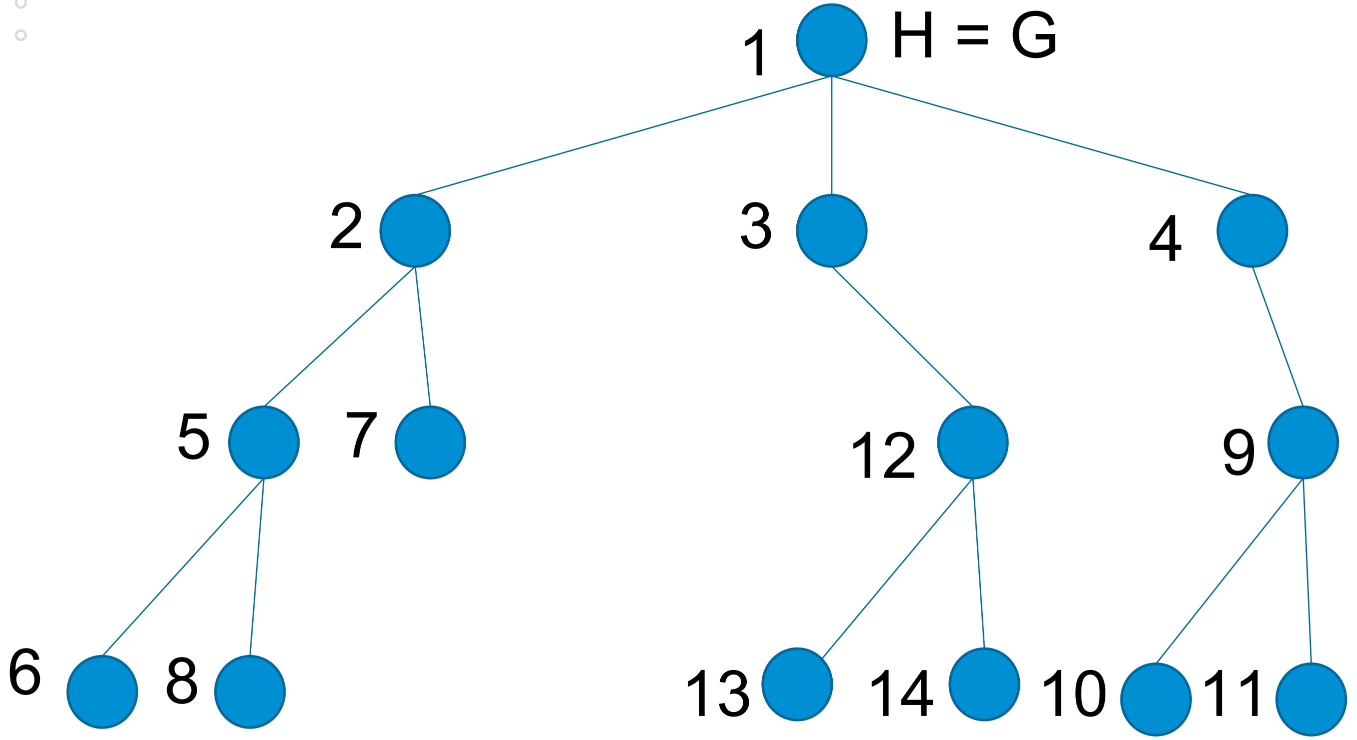


Pruning criteria:

- Non improving solution: LB ≥ best current objective
- Unfeasible solution:
  - Non-independent set solution
  - Non-maximal set solution
- Redundant solution: the same induced subgraph H has yet been explored previously

PCI topological solution: Iterative MIS

BRANCH&BOUND APPROACH (BB-itMIS)



BB tree exploration:

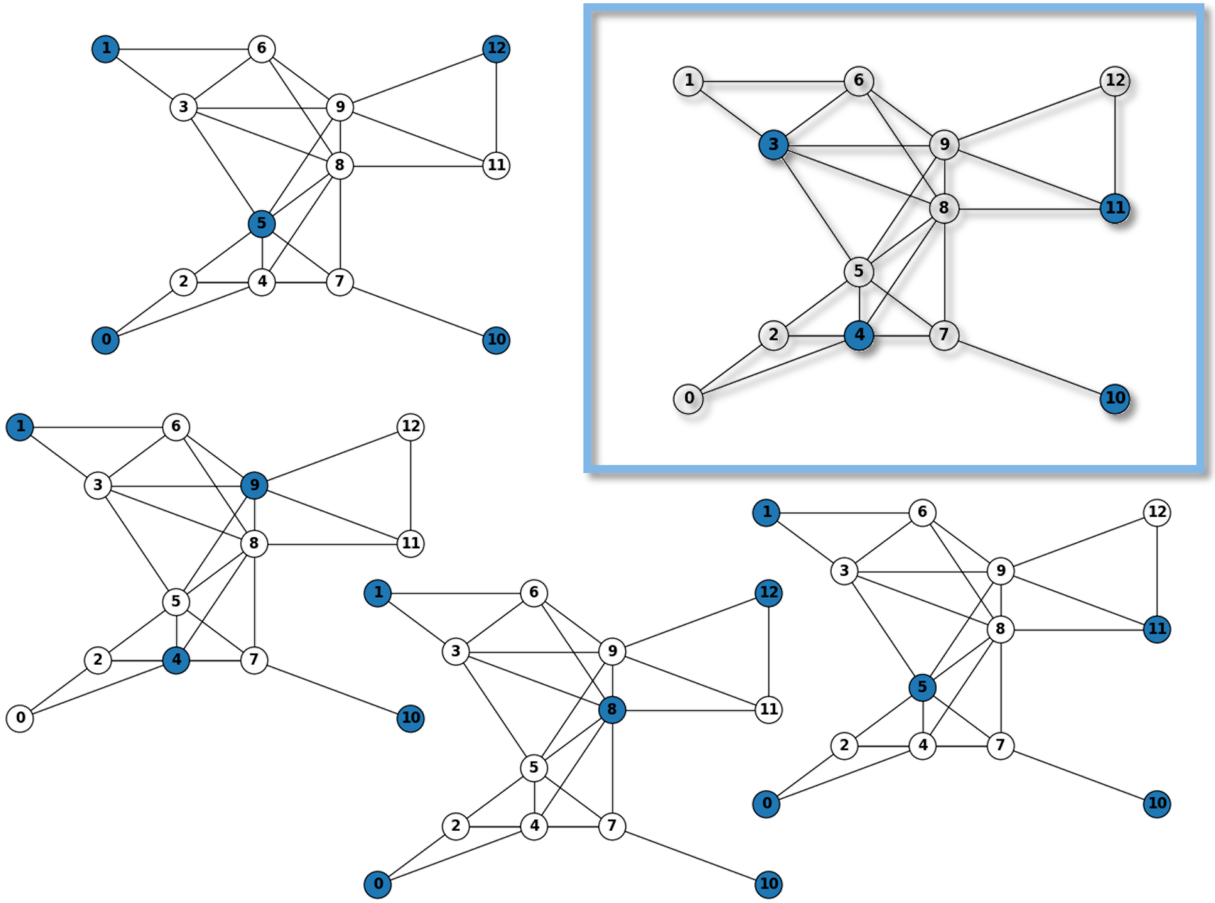
- Traditional strategies: FIFO, LIFO, gap-based, depth-first, etc...
- Custom exploration policy: each node has a priority defined as
  - UB × |edges(H)|

The tree exploration policy seems to be also dependent on the number of MPI processes that are used to parallelise the BB procedure.

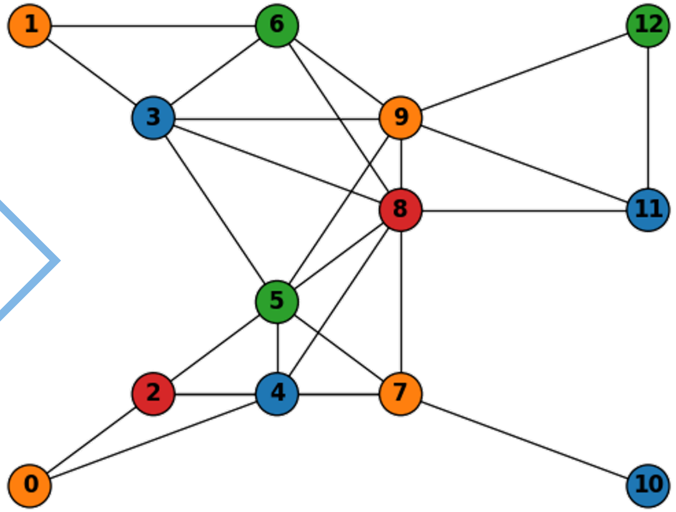
PCI topological solution: Iterative MIS

BRANCH&BOUND APPROACH (BB-itMIS)

MIS solutions at BB root node



BB-itMIS solution



4 colors: 1 less than the Greedy-itMIS solution

Graph Coloring DEMO

https://github.com/LINKS-Foundation-CPE/cineca_aspc

Questions?

ANDREA SCARABOSIO, PAOLO VIVIANI, CHIARA VERCELLINO – LINKS FOUNDATION

18TH ADVANCED SCHOOL ON PARALLEL COMPUTING: QUANTUM COMPUTING
CINECA, FEB 2022