Bringing order to the QUBO-zoo

QUBO parameters and optimization algortithms

Spataro David
Hamburg, 25.9.2022

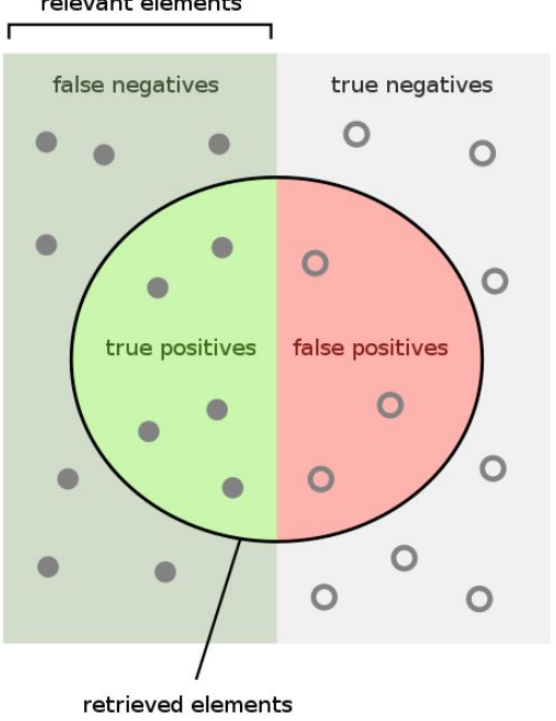
Initial Configuration

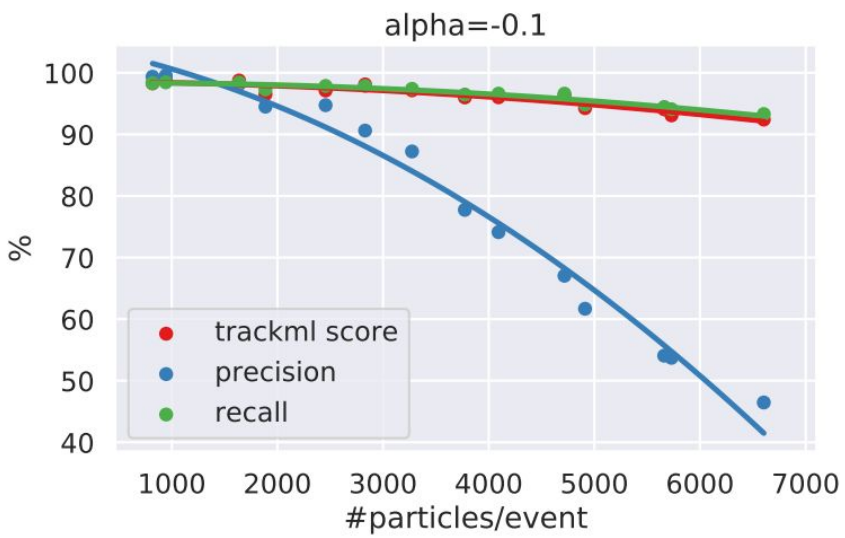
QUBO parameters

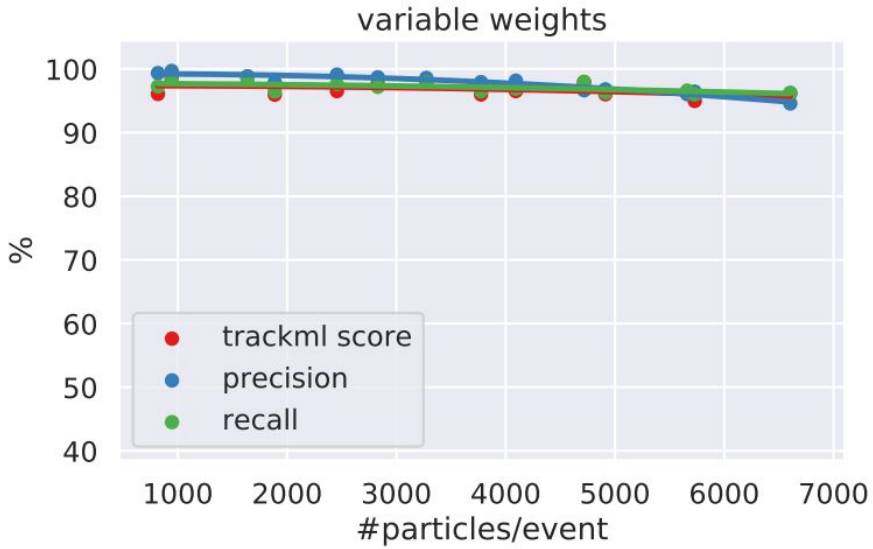
What they did in :

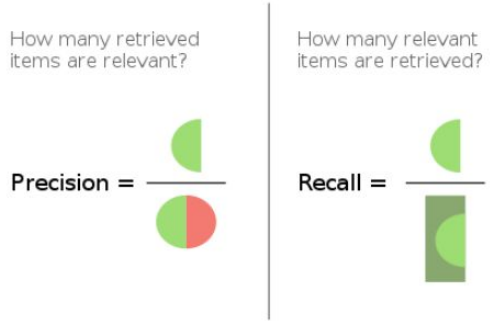
"**A pattern recognition algorithm for quantum annealers**":

- a_i : not set at the start, but later used in the Thesis of Lucy Linder
- b_ij: based on angular information









Solving algorithm

Optimization algorithm + Initial solution

What they did in :

"**Partitioning Optimization Problems for Hybrid Classical/Quantum Execution**":

- Random initial guess [0, 1, 1, 0, 0, 1, ….]
- Impact list from lowest to highest impact

"A Multilevel Algorithm for Large Unconstrained Binary Quadratic Optimization"

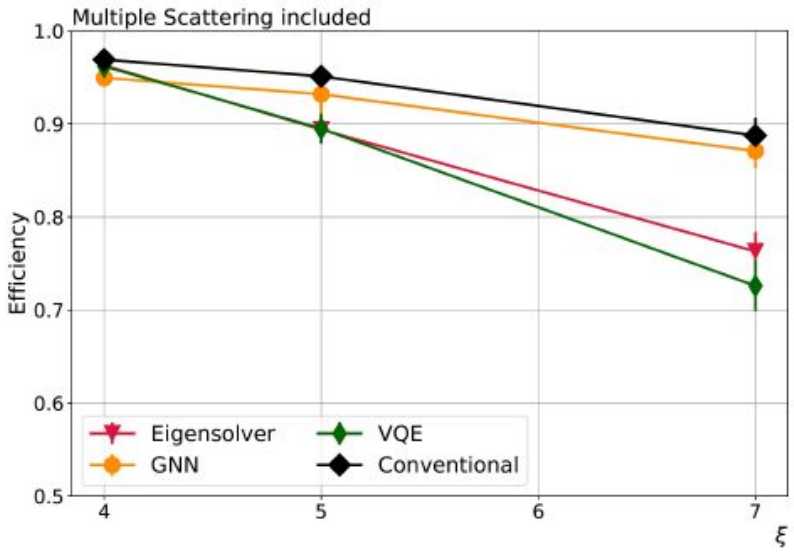
→ "*qbsolv uses a backbone-based method inspired by Glover et al. , which may result in getting stuck at a local minima despite the use of large-neighborhood moves. Other partitioning strategies may yield better results*."

Algorithm summary: Choose a random starting point and try to mess up the QUBO as little as possible

What we did so far (example for CTD)

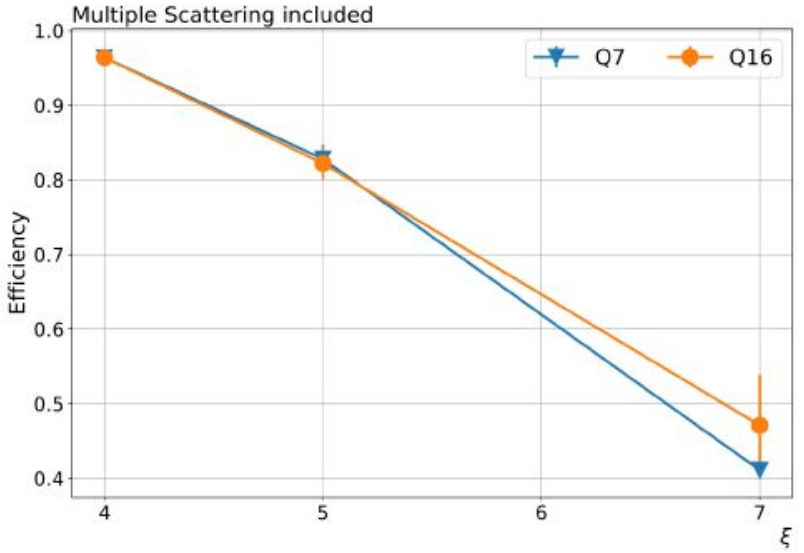
QUBO parameters / initial guess / impact list ordering

- no a_i values
- random initial guess
- impact list from least impact to highest impact



Observations:

- huge drop in efficiency at high multiplicity
- huge drop in efficiency at high density
- (sometimes) large error bars



Rethink the approach

QUBO parameters / initial guess / impact list ordering

- Set a_i values:

  a) P(real track | small angle) > P (real track | big angle) in general

- Start with [1, 1, 1, …, 1] means start at efficiency = 100%
- All triplets are now in state 1, so the impact list can be seen as ordering by number of conflicts. So the ones with the highest conflict potential should be cut away **first**

Probably the reason why the bit flip algorithm is so powerful

Bit flip is basically Eigensolver with SubQUBO size 1, just 100 times faster

Sneak Peak on results

Eigensolver subQUBO size 7,, xi = 4, whole BX

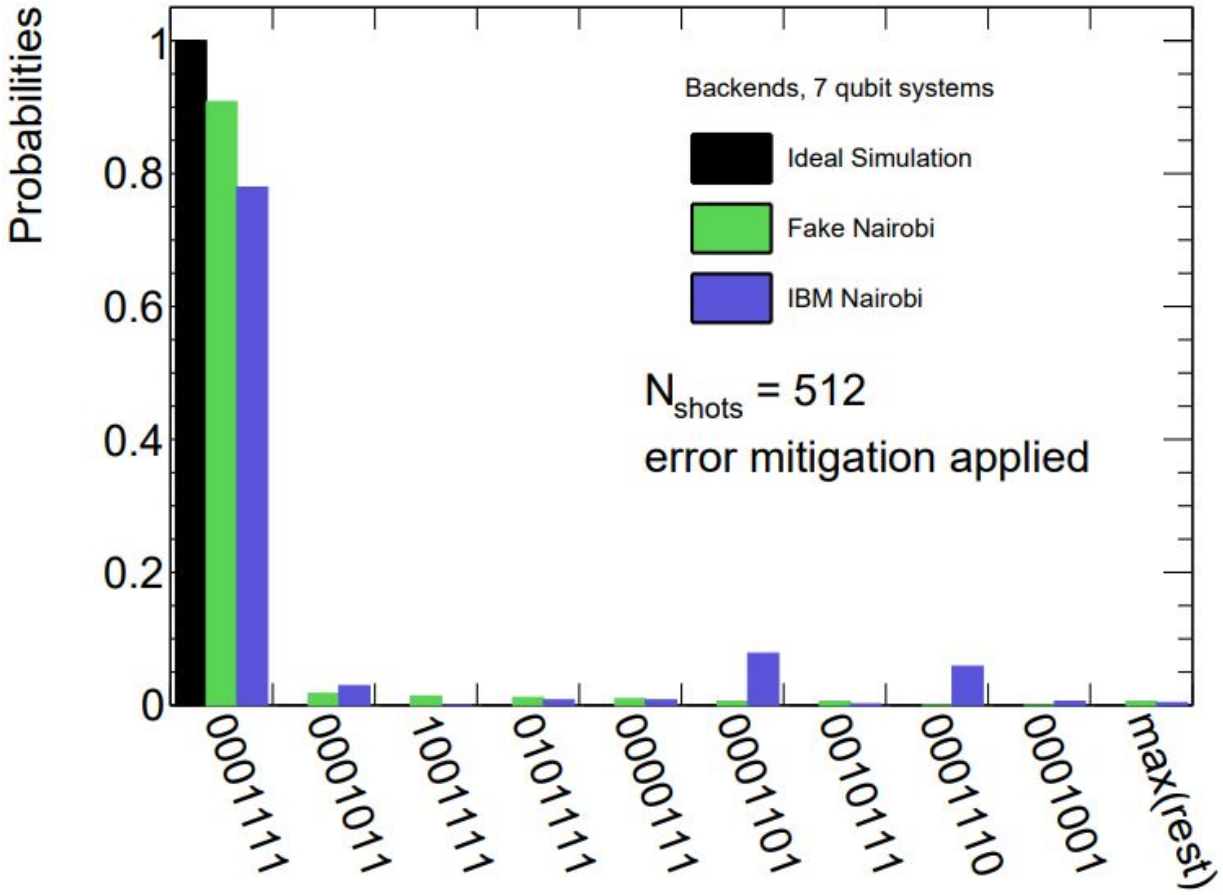
a_i = 0

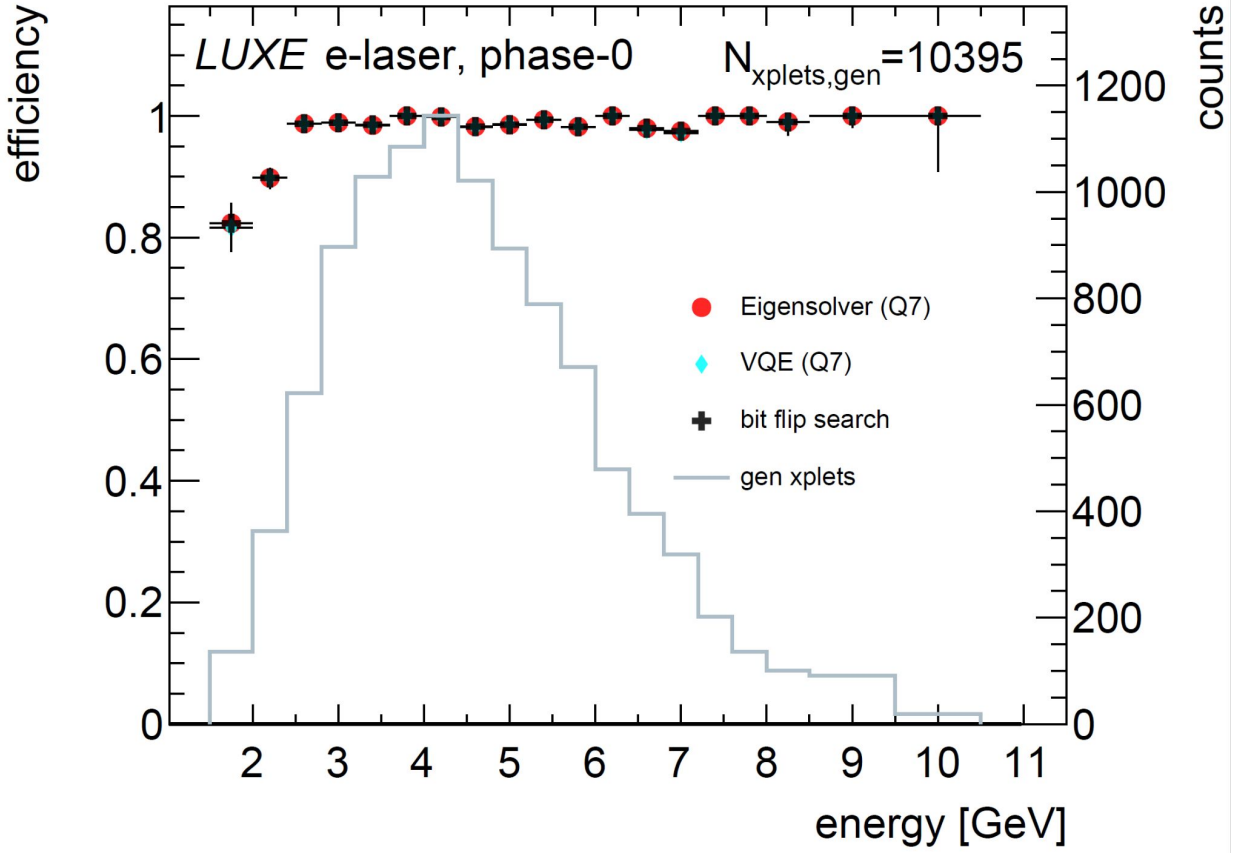
Configuration	initial guess: ONES impact list: REVERSED	initial guess: ZEROS impact list: REVERSED	initial guess: ONES impact list: NOT REVERSED	initial guess: ZEROS impact list: NOT REVERSED	initial guess: RANDOM impact list: REVERSED	initial guess: RANDOM impact list: NOT REVERSED
Efficiency	0.980 +/- 0.003	0.00 +/- 0.000	0.889 +/- 0.033	0.00 +/- 0.000	0.956 +/- 0.007	0.746 +/- 0.1

VQE ansatz circuit

TwoLocal ansatz

VQE with TwoLocal ansatz and Eigensolver for a 7 qubit system show same results !





What's next

Updated solving methods / Scaling / Alternative optimization algorithm

Updated solving methods (already in the making):

- impact list vs. impact list reversed
- initial guess for random, [0, 0, ...,0] and [1, 1, …,1]
- various subqubo sizes and QUBO parameter settings

Scaling:

- map number of tracks vs. subQUBO size (various track densities?)

Alternative optimization algorithms:

- impact list
- merged-cluster