Tracking with QUBO

Quadratic unconstrained binary optimization

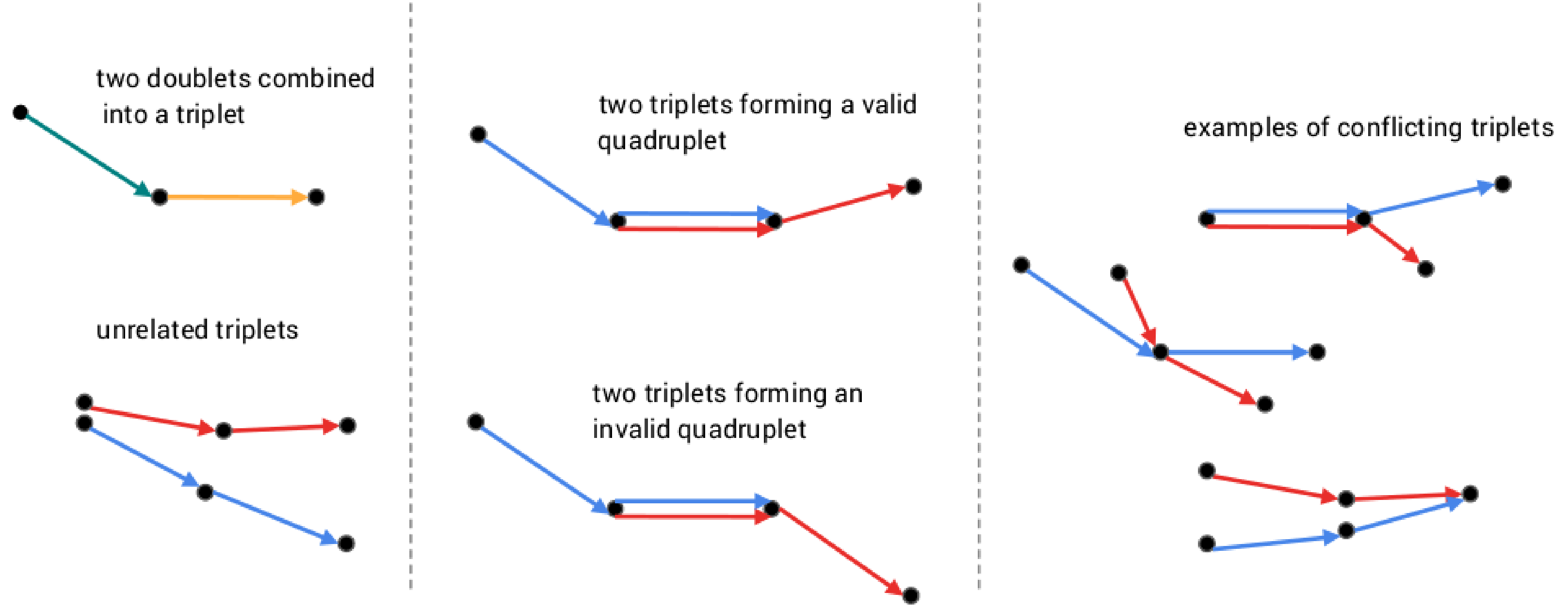
Paolo, Lucy, Miha, Sean
18 December 2018

Tracking Workshop for HEP @ LBL

The algorithm

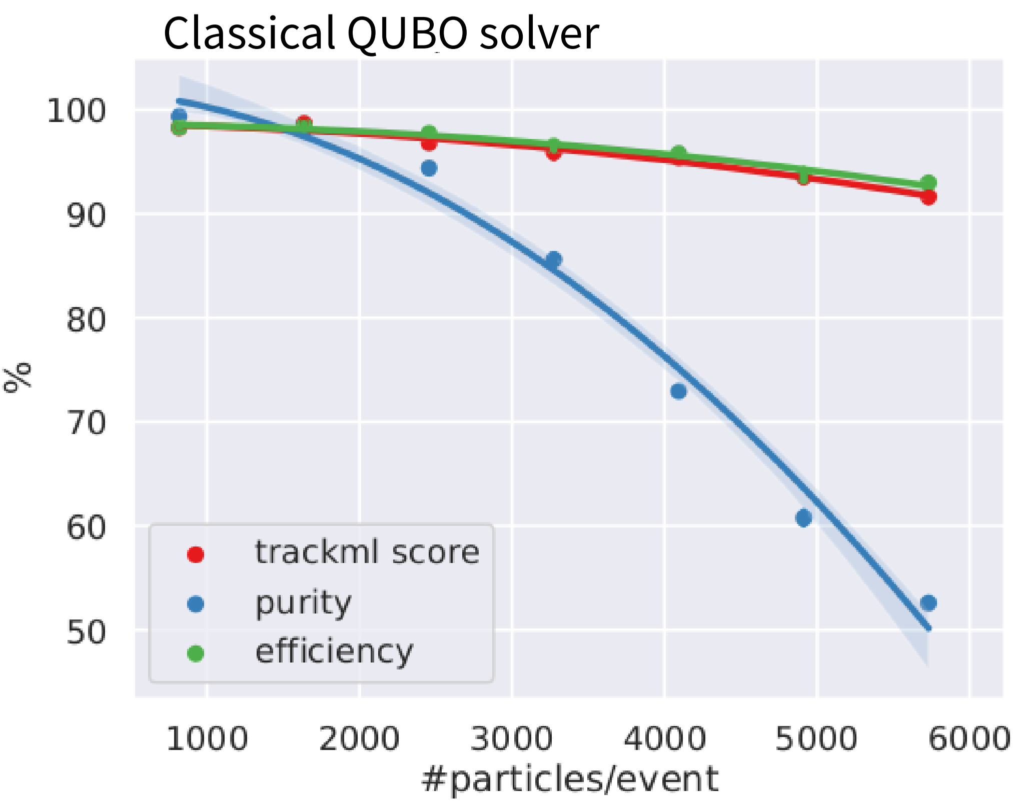
1) Construct triplets T_i from three hits in consecutive layers,
2) Construct the energy function (QUBO) from triplets,
3) Minimize the energy as a function of used (accepted) triplets.

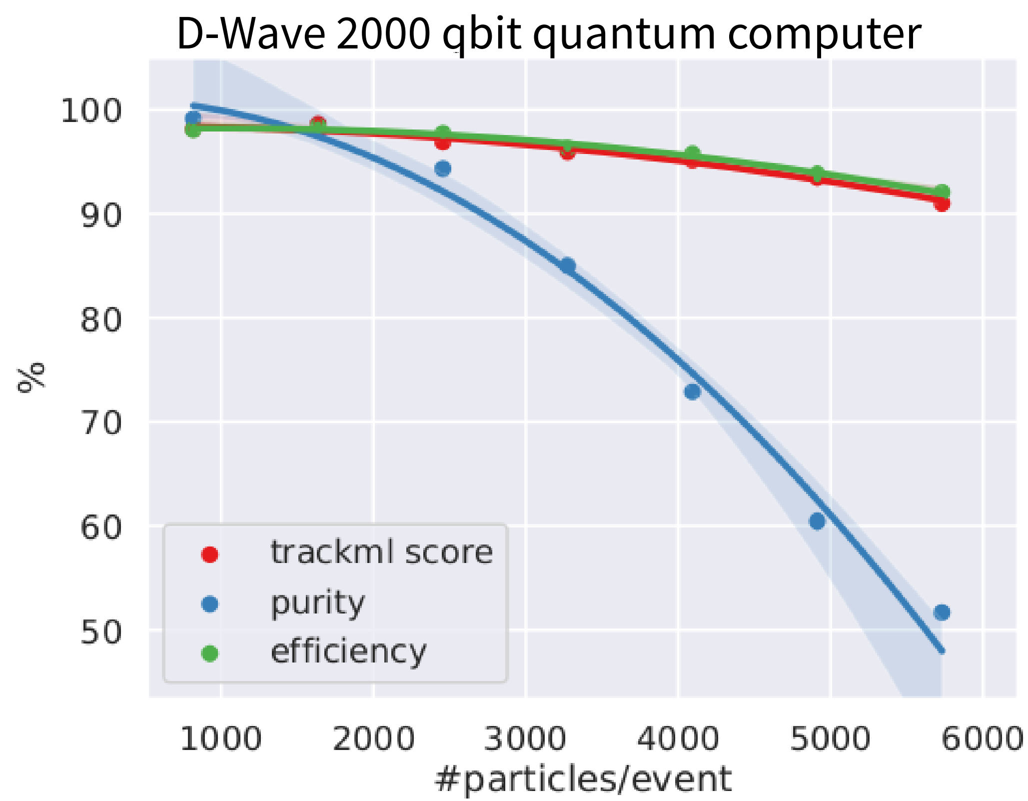
$$O(a,b,T) = \sum_{i=1}^{N} a_i T_i + \sum_{i}^{N} \sum_{j<i}^{N} b_{ij} T_i T_j \qquad b_{ij} = \begin{cases} -S(Ti,Tj), & \text{if } (T_i,T_j) \text{ form a quadruplet,} \\ \zeta & \text{if } (T_i,T_j) \text{ are in conflict,} \\ 0 & \text{otherwise.} \end{cases}$$



Established results

- Successfully used with the TrackML dataset (barrel only and no double hits).
- QUBOs can be efficiently solved both by classical and quantum computers using annealing algorithms.



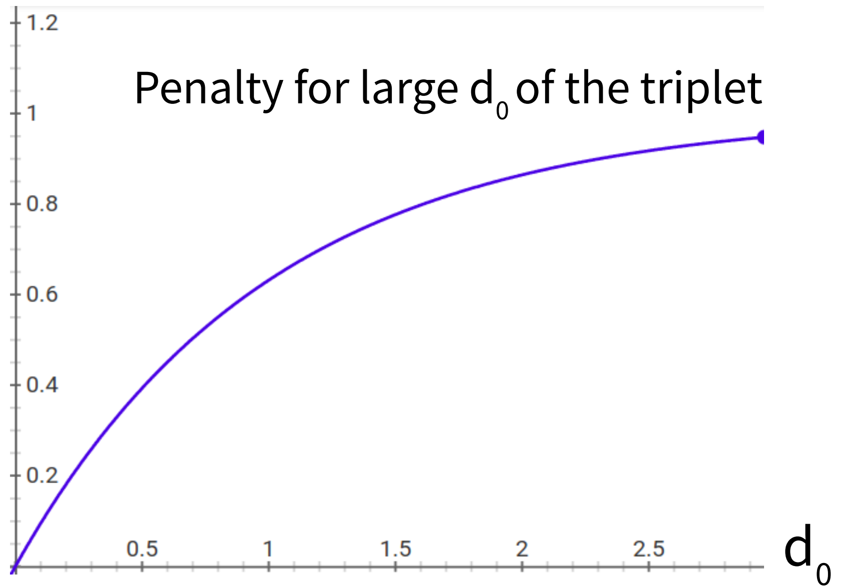


Progress during this week

- **Purity (1 – fake rate) drops rapidly for denser environments.**
- **So far, the bias weight a_i in QUBO was constant.**
  - improved by weighting each triplet T_i with the impact parameters:

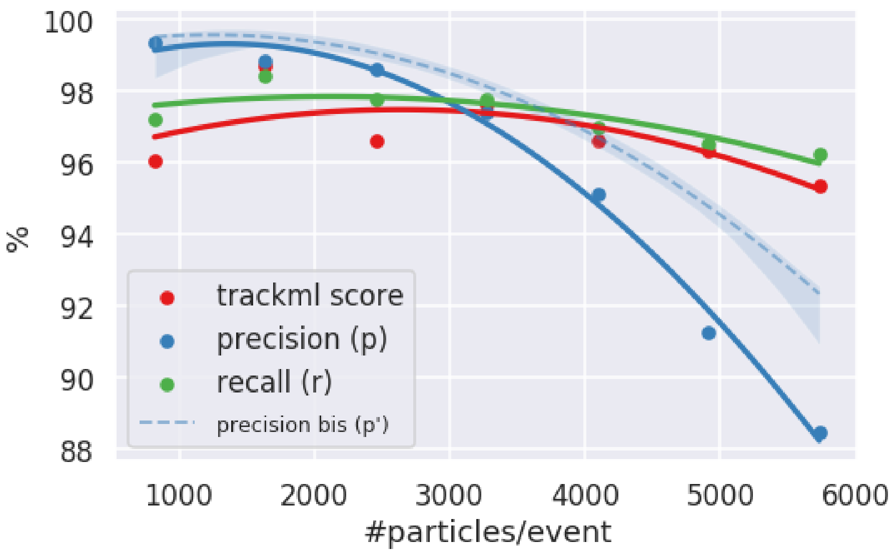
$$a_i = \alpha\left(1 - \exp\left(-d_0^i / \sigma_{d_0}\right)\right) + \beta\left(1 - \exp\left(-z_0^i \sin\theta^i / \sigma_{z_0}\right)\right)$$

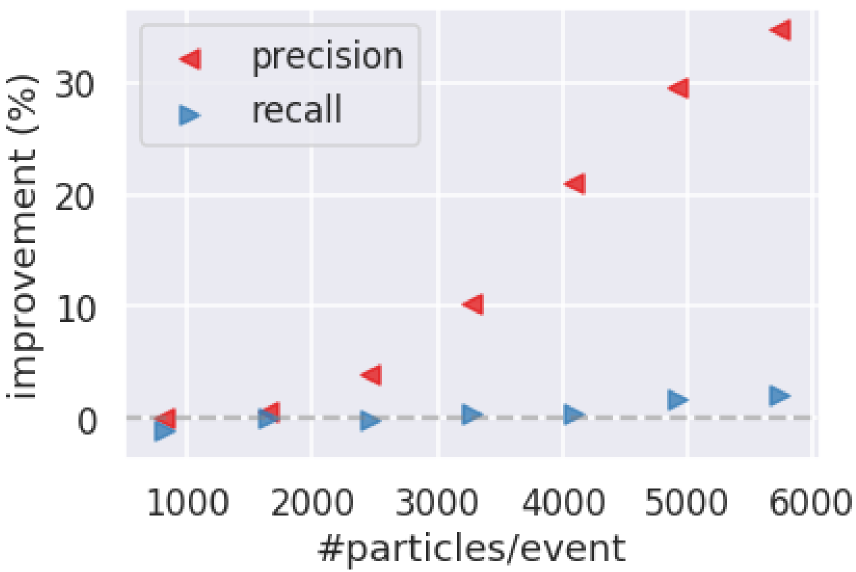
$$O(a, b, T) = \sum_{i=1}^{N} a_i T_i + \sum_{i}^{N} \sum_{j<i}^{N} b_{ij} T_i T_j$$



Results with the new weights

- Significant improvement in precision achieved by using impact parameters for the bias weight.
- Tracking with QUBO gives comparable results to conventional tracking algorithms.





Next steps

- **Using impact parameters may have unwanted side-effects (e.g. long lived particles, b-tagging, ...).**
  - Performance vs. the strength of the d_0 bias weight should be studied.
- **Currently the entire code is written in Python.**
  - Migration to C++ would be useful for performance improvements and compatibility with ACTS.
- **Investigate whether QUBOs can be efficiently solved by GPUs.**