8 January 2025 · CC-BY 4.0

Peer Review

Review of: "An Approximated QUBO Formulation for Solving Practical SAT Problems"

Georg Hahn[1]

1. University of Cambridge, Cambridge, United Kingdom

The author proposes a conversion from k-SAT to QUBO formulation. Solving an input SAT instance can then be attempted by solving the generated QUBO with a QUBO solver. The advantages of such a construction should be highlighted more. In particular, there should be more detail on the tradeoff of the approximation, since the proposed conversion method cannot guarantee an exact solution even if the QUBO were to be solved exactly. Can the optimality gap of the QUBO be translated to the quality of the SAT solution? Can the author please include solution results for an advanced SAT solver (e.g., miniSAT) as well? Further comments are as follows:

page 1, first paragraph: One could be more specific here and say what recent advances, aka in quantum annealing, gate quantum computing, neuromorphic computing.

page 2, first paragraph: "More formally, a SAT instance" -- One should introduce k-SAT already here, since k is a parameter needed to precisely define the SAT problem.

page 2, first paragraph: "As a variant of kSAT" -- typically, the problem is written as k-SAT

page 2, first paragraph: As general info for the reader, the difficulty classes for MAX-k-SAT and NAE-k-SAT should maybe also be stated here. E.g., MAX-k-SAT is also NP-hard as a decision problem.

page 2, second paragraph: "We simply apply MAX2SAT formalization to binary clauses and NAE3SAT formalization to ternary clauses, resulting in an approximated QUBO." -- I feel the problem is insufficiently described; one can say much more here: For instance, what largest size can SAT solvers solve (typically several hundred), and QUBO solvers (to the best of my knowledge, fewer variables). Why is it thus a good idea to convert SAT to QUBO? What does approximate mean here, why not convert exactly? What do you gain with the approximate formulation, is it fewer variables, etc., compared to the known exact conversion to QUBO? Why consider an approximation in the first place,

 doi.org/10.32388/LE65KM

is it because the QUBO can likely be only solved approximately afterwards? If the QUBO is solved up to x% of the optimal solution (optimality bound), then how does it translate to your approximate solution of the SAT instance? These questions should be answered here.

page 3, equation (3): Why is the formula stated as Hamiltonian H? One should be more rigorous here and say we are given an input SAT problem clause c_i as defined earlier. Then one should state all options for what c_i could be and how they are translated, without yet introducing H. And then at the end, one can give the formula for how H is assembled. Of course, one can kind of guess what the author means, but this should be defined more rigorously.

page 3, first paragraph of "Experiments": The author uses dwave-neal to solve the QUBOs. One should at least state here that this is not an exact solver and discuss what this means for the comparison and how it might be impacted, as e.g., some conversion method might result in QUBOs that are easier for dwave-neal. A fair comparison is only achieved, in my opinion, when solving the QUBOs exactly. This could be done as well for up to several hundred variables with e.g., CPLEX or GUROBI. Can the author please add such a comparison?

page 5, Table 1: I suggest giving the method a name instead of referring to it as "Our".

page 5, Table 1: Can the author add results for solving the k-SAT instances directly with an advanced SAT solver (e.g., miniSAT), please? A reader familiar with the topic will surely be interested in seeing what happens when attempting to solve the SAT instances directly.

Declarations

Potential competing interests: No potential competing interests to declare.

doi.org/10.32388/LE65KM