Ref. Ares(2024)8337575 - 22/11/2024

Software Suite for Benchmarking of Quantum Algorithms Applied to Two Typical Smart-Charging Optimization Problems

Rodolphe Griset[1] and Joseph Mikael[1]
[1]EDF R&D, 7, boulevard Gaspard Monge, 91120 Palaiseau, France

November, 2024

Abstract

This report constitutes the delivery n° 5.10 of the NEASQ project. It details the implementation of a software suite for benchmarking quantum algorithms applied to two typical smart-charging optimization problems. This deliverable builds upon the methods developed during the Phd of Margarita Veshchezerova [6] and already described in the reports of deliverable n° 5.3 : "Specification and implementation of QAOA algorithm for the minimizing of electric vehicles charging time" and integrates them into a comprehensive open-source library which can be use as an initial step for future studies.

Contents

1 Graph Coloring and Maximum Independent Set (MIS)

The library implements methods for solving the Graph Coloring and MIS problems, these problems are classical in the operations research field :

- **Graph Coloring:** Assigns colors to vertices of a graph such that no two adjacent vertices share the same color, minimizing the number of colors used.
- **Maximum Independent Set:** Finds the largest set of vertices in a graph that are not adjacent to each other.

The next section describes why those two problems are useful for smart-charging problems. For more detailed explanations of this problems methods, please refer to the report D5.3.

1.1 Graph Coloring

Graph Coloring involves assigning colors to the vertices of a graph such that no two adjacent vertices share the same color. The goal is to minimize the number of colors used. In the context of smart charging, this can be applied as follows:

- **Vertices:** Represent charging tasks or intervals for different vehicles.
- **Edges:** Represent conflicts between tasks, such as overlapping time intervals or tasks belonging to the same group (e.g., vehicles from the same fleet).
- **Colors:** Represent different charging points or time slots.

By finding a minimal coloring of the graph, we can determine the minimum number of charging points required to ensure that no two conflicting tasks are assigned to the same point. This helps in optimizing the allocation of charging resources and minimizing the total completion time for all charging tasks.

1.2 Maximum Independent Set (MIS)

Maximum Independent Set (MIS) involves finding the largest set of vertices in a graph that are not adjacent to each other. In the context of smart charging, this can be applied as follows:

- **Vertices:** Represent charging tasks or intervals for different vehicles.
- **Edges:** Represent conflicts between tasks, such as overlapping time intervals or tasks belonging to the same group.
- **Independent Set:** Represents a set of non-conflicting tasks that can be scheduled simultaneously.

By finding the MIS, we can determine the largest set of non-overlapping charging tasks that can be scheduled at the same time. This helps in maximizing the utilization of available charging points and ensuring that the charging process is as efficient as possible.

2 Implemented Methods

This section provides an overview of the various methods implemented in the software suite for benchmarking quantum algorithms applied to smart-charging optimization problems. These methods are detailed in the report D5.3, which offers a comprehensive explanation of their specifications.

2.1 Quantum Approximate Optimization Algorithm (QAOA) and Recursive QAOA (RQAOA)

QAOA is a variational quantum-classical heuristic designed to find low-energy states of Ising Hamiltonians. It is particularly useful for combinatorial optimization problems that can be mapped to such Hamiltonians. The algorithm involves preparing and measuring a parameterized quantum state, with the goal of minimizing the expectation value of the Hamiltonian. The optimization of parameters is made by a classical routine before calling the quantum computer to stochastically evaluate the expectation. As shown in [3] it is possible for Max-Cut and $p = 1$ to use the analytical expression that allows evaluating in polynomial time this expectation. This method is implemented in the library. It means that the library does not currently call a quantum emulator to simulate the quantum behaviors but directly solves it using the analytical formula. However, the pricing problems can be easily replaced by a call to a NISQ device.

RQAOA [2] extends QAOA by performing recursive variable elimination. At each recursion step, RQAOA evaluates the expectation values of variables and their correlations, fixing the variable or pair with the highest absolute mean. This process continues until the problem size is reduced to a manageable level, at which point a brute-force algorithm is used to find the optimum. The library implements the entire RQAOA approach with parameterized parameters.

2.2 Classical Heuristics

The library includes several classical heuristics for the MWIS problem for comparison with quantum methods:

- **Greedy :** A solution build iteratively by making sequential decisions [4] .
- **Local Search:** An optimization technique that iteratively improves the solution by exploring its neighborhood [1].
- **Branch and Bound:** An exact algorithm that systematically explores the solution space, pruning branches that cannot yield better solutions.

2.3 Hybrid Classical/Quantum Methods

The software suite integrates quantum algorithms into classical decomposition schemes, specifically the Branch and Price framework. This hybrid approach leverages the strengths of both classical and quantum methods to tackle large-sized integer or mixed integer linear programs.

2.3.1 Column Generation and Dantzig-Wolfe Decomposition

Column generation is used when the number of variables in the Integer Linear Program (ILP) is too large. This situation often arises in various contexts, such as when the optimization problem is formulated in a way that results in a large number of variables. Dantzig-Wolfe decomposition is a well-known and widely used decomposition scheme for solving large linear programs. It is particularly useful for linear problems with a specific structure, where subproblems are coupled by a set of binding constraints and become independent when these constraints are relaxed.

The matrix of such problems takes a "block diagonal" structure, where each block corresponds to a subproblem. The Master Problem (MP) in this decomposition framework is formulated by replacing the variables of the original problem with variables representing the convex combination of extreme points of the subproblems' feasible regions.

2.3.2 Embedding Quantum Algorithms

Quantum algorithms, such as QAOA, are embedded into the Branch and Price framework to solve the pricing subproblems. These subproblems often take the form of pure combinatorial problems, such as Max-Cut, Knapsack, Maximum Independent Set, and Shortest-Path, which are well-suited for quantum algorithms. By focusing quantum techniques on these well-defined, purely combinatorial subproblems, we can avoid many of the issues associated with directly applying quantum algorithms to large-scale problems.

The integration of quantum heuristics in the pricing step of column generation allows for the efficient solving of these subproblems. If the classical heuristic fails to find an improving variable, the quantum heuristic is launched. If both heuristics fail, an exact algorithm is executed to provide strong guarantees of the optimality of the relaxed solution.

2.3.3 Implementation in SCIP

This decomposition scheme has been implemented in the SCIP optimization suite, which provides a C++ extensible framework for Mixed Integer Optimization, including a Branch and Price module [5]. The implementation is designed to be generic, allowing any combinatorial pricing subproblem to be tackled according to the described scheme, provided it has been modeled as a QUBO/Ising Hamiltonian. In the library, the decomposition scheme is instantiated for coloring problem using an heuristic pricing problem and the RQAOA approach with the analytical formula when the heuristic classical pricing fails.

This hybrid approach represents a significant advancement in leveraging quantum computing for practical optimization problems, combining the strengths of classical and quantum methods to achieve better performance and scalability.

For detailed explanations of these methods and their implementations, please refer to the report D5.3.

3 Software Suite Structure and Components

The software suite is designed to benchmark quantum algorithms for smart-charging optimization problems. It is implemented in C++ exploiting CMake tool to link the different dependencies.

3.1 Library components

The software suite includes the following components:

- Graph Algorithms : This module contains fundamental graph operations and data structures essential for various optimization tasks, including algorithms for graph coloring and handling Hamiltonian paths and cycles.
- Optimization Methods : This module implements various optimization techniques, including local search heuristics, greedy algorithms for initial solution generation, and integration with solvers like CPLEX for solving complex optimization problems.
- Quantum Methods : This module focuses on quantum-inspired optimization techniques, particularly those involving Hamiltonian paths and cycles, which are crucial for quantum optimization routines.
- test_data : the folder contains a benchmark of graph instances of increasing size from 20 to 400 nodes with different structure which facilitate future work exploiting the library.

3.2 Library dependencies

Depending on the method the library needs to install dependencies :

- **NLopt (version 2.7.1)**: A library for nonlinear optimization. This library is mandatory but a zip file of the library is stored inside the depot and installation procedure is detailed in the ReadMe of the library.
- **Optional: SCIP (version 8.0.2)**: Used for the hybrid Branch & Price approach.
- **Optional: CPLEX (version 12.7)**: Optional solver for the MWIS problem

Instruction to provide paths to those dependencies and to activate them in the library can be found in the ReadMe.md file of the library.

4 Conclusions and Perspectives

This software suite provides a robust toolset for benchmarking quantum algorithms in smart-charging optimization. It integrates various quantum and classical methods, offering a comprehensive environment for testing and evaluation. The genericity of the hybrid quantum/classical Branch & Price library developed allows for benchmarking quantum algorithms across a wide range of Integer Linear Programs.

The theoretical methods developed during the PhD of Margarita Veshchezerova [6] have been integrated into this open-source library. This integration provides a solid foundation for future studies in the field of quantum operations research, particularly as applied to smart charging problems. The library's modular structure and extensive documentation make it accessible for researchers and practitioners looking to explore and extend quantum optimization techniques.

The software suite's implementation in the SCIP optimization suite, along with its support for key libraries such as SCIP, NLopt, and optionally CPLEX, ensures that it is both versatile and powerful. The inclusion of both classical and quantum methods allows users to leverage the strengths of each approach, providing a balanced and effective solution for complex optimization problems.

This work constitutes the delivery n° 5.10 of the NEASQ project: "Software suite for benchmarking of quantum algorithms applied to two typical smart-charging optimization problems." The successful completion of this deliverable marks a significant milestone in the project, demonstrating the practical applicability and potential of quantum algorithms in real-world optimization scenarios.

Future work will focus on enhancing the suite's capabilities, expanding its application to other optimization problems, and further integrating advanced quantum algorithms. The open-source nature of the library encourages collaboration and continuous improvement, paving the way for innovative solutions in the rapidly evolving field of quantum computing.

References

[1] Diogo Andrade, Mauricio Resende, and Renato Werneck. Fast local search for the maximum independent set problem. In *International Workshop on Experimental and Efficient Algorithms*, pages 220–234. Springer, 2008.

[2] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to variational quantum optimization from symmetry protection. *Physical Review Letters*, 125(26):260505, 2020.

[3] Constantin Dalyac, Loïc Henriet, Emmanuel Jeandel, Wolfgang Lechner, Simon Perdrix, Marc Porcheron, and Margarita Veshchezerova. Qualifying quantum approaches for hard industrial optimization problems: A case study in the field of smart-charging of electric vehicles. *EPJ Quantum Technology*, 8(1):12, 2021.

[4] Stephan Held, William Cook, and Edward Sewell. Maximum-weight stable sets and safe lower bounds for graph coloring. *Mathematical Programming Computation*, 4, 2012.

[5] The scip optimization suite. `https://www.scipopt.org/`. Accessed: 2024-11-21.

[6] Margarita Veshchezerova. Quantum algorithms for energy management optimization problems. *Theses, Université de Lorraine, December*, 2022.