# Discrete optimization: A quantum revolution?

Stefan Creemers[1, 2], Luis Fernando Pérez[1]
[1] IÉSEG School of Management
{s.creemers,l.perez}@ieseg.fr
[2] KU Leuven (ORSTAT)
stefan.creemers@kuleuven.be

**Keywords** : *Quantum computing, discrete optimization*

## 1 Introduction

The idea of quantum computing was first launched in 1980, when Benioff defines a quantum mechanical model of a Turing machine. A few years later, Feynman (1982) introduces the idea of a universal quantum simulator. Building on the work of Benioff and Feynman, Deutsch (1985) describes the first universal quantum computer that is able to efficiently simulate any other quantum computer. In addition, Deutsch also develops the Deutsch algorithm, the first quantum algorithm that has a proven speedup when compared to a classical algorithm. Later, Deutsch and Josza (1992) extend this algorithm. Deutsch and Josza, however, consider a problem (the Deutsch-Josza problem) that has no practical use, and it takes until 1994 for quantum computing to really take off. In 1994, Shor presents a quantum algorithm to find the prime factors of large integers in polynomial time (whereas the best classical algorithm requires sub-exponential time). In theory, Shor's algorithm can be used to break many of the cryptography schemes in use today. Not surprisingly, the publication of Shor's algorithm sparked an enormous interest in quantum computing. A few year later, Grover (1996) published a quantum algorithm that builds on the algorithm of Deutsch and Josza (1992). The algorithm of Grover performs an unstructured search, and achieves a quadratic speedup when compared to a classical algorithm. Arguably, Grover's algorithm is one of the most important algorithm in quantum computing today. It is the corner stone of many other quantum algorithms (see e.g., the Quantum Algorithm Zoo) and can be used to solve a multitude of problems.

## 2 Problem description

We use Grover's algorithm as a subroutine in a binary search procedure that can be used to solve any discrete optimization problem. A discrete optimization problem may be defined as:

Maximize (or minimize) $g(\boldsymbol{x})$

Subject to:

$$x_i \in \Omega_i,\ \forall i: 0 \leq i \leq n$$

(any other constraint)

Where:

- $g(\boldsymbol{x})$ is the objective function that evaluates a solution $\boldsymbol{x} = \{x_1, x_2, \dots, x_n\}$.
- $n$ is the number of decision variables.
- $x_i$ is the $i^{\text{th}}$ decision variable.
- $\Omega_i$ is the set of discrete values that can be assigned to decision variable $x_i$.

Note that the objective function and/or constraints do not have to be linear. Examples of discrete optimization problems include: 3SAT, knapsack, TSP, non-linear integer programming problems…

## 3 Results and conclusion

We first propose a binary search procedure that requires $O(\eta L\sqrt{2^{nb}})$ operations (where $\eta$ is the number of operations required to evaluate the feasibility of a solution, $L$ is a logarithmic function of the bounds on the optimal solution value, and $2^b$ is the number of discrete values that can be assigned to a decision variable). For instance, for solving a binary knapsack problem that has n items, $b = 1$, $O(\eta)$ is equivalent to $O(n)$, and the binary search procedure requires $O(nL\sqrt{2^n})$ operations (whereas the best classical algorithms require $O(n\sqrt{2^n})$ operations).

We extend this work, and also present two new algorithms that can solve any discrete optimization problem using only $O(\eta\sqrt{2^{nb}})$ operations. Not only do these algorithms match the best classical algorithms for solving the binary knapsack problem, they can be used to solve any discrete optimization problem (regardless of whether constraints and/or objective function are linear; regardless of the structure of the problem). We conclude that these algorithms herald a revolution in the field of discrete optimization.

## Références


Benioff, P. 1980. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. J. Stat. Phys., 22(5), 563–591.

Creemers, S., Pérez, L. F. 2023a. Discrete optimization: A quantum revolution (Part I). Under review in Oper. Res.

Creemers, S., Pérez, L. F. 2023b. Discrete optimization: A quantum revolution (Part II). Under review in Oper. Res.

Deutsch, D. 1985. Quantum theory, the Church–Turing principle and the universal quantum computer. Proc. R. Soc. Lond. A, 400(1818), 97–117.

Deutsch, D., Jozsa, R. 1992. Rapid solution of problems by quantum computation. Proc. R. Soc. Lond. A, 439(1907), 553—558.

Feynman, R. P. 1982. Simulating physics with computers. Int. J. Theor. Phys., 21(6-7), 467–488 .

Grover, L. K. 1996. A fast quantum mechanical algorithm for database search. Proc. Annu. ACM Symp. Theory Comput., 212–219.

Shor, P.W. 1994. Algorithms for quantum computation: Discrete logarithms and factoring. Proc. Annu. Symp. FOCS, 124–134.