Restricting the search space to boost Quantum Annealing performance

Pierre Bergé and Baptiste Cavarec
CentraleSupélec, Université Paris-Saclay
3 rue Joliot-Curie, 91192 Gif-sur-Yvette, France

Arpad Rimmel and Joanna Tomasik
LRI, CentraleSupélec, Université Paris-Saclay
Bat 650, Rue Noetzlin, 91405 Orsay, France

***Abstract*—We are interested in Quantum Annealing (QA), an algorithm inspired by quantum theory and Simulated Annealing (SA). It is based on quantum replicas, which explore an energy surface, and are less prone to be trapped in local minima. Moreover, kinetic energy helps replicas to find a global minimum. This method has proved its efficiency for several optimization problems. We start this study by presenting the application of QA to a new problem: the Multidimensional Knapsack Problem (MKP).**

We then present a new idea to speed up the quantum annealing process by detecting the resemblance between replicas. If many of the replicas exhibit the same properties, our assumption is that these properties will also be present with a high probability in a global solution. Consequently, the QA may restrict certain mutations in order to preserve those similarities. We call this algorithm Restrictive Quantum Annealing (RQA).

We establish that RQA has better performances than QA and SA by carrying out an adequate analysis of the RQA performance, taking the Traveling Salesman Problem (TSP) and the above-mentioned MKP as references. We also advance guidelines indicating types of NP-hard problems for which our algorithm is particularly well adapted.

I. Motivation

Simulated Annealing (SA) is commonly recognized as an efficient metaheuristic algorithm to solve combinatorial hard problems by local search. Its principle is typically illustrated for the NP-hard Traveling Salesman Problem (TSP) [1] because it performs very well for problems with a distinct global minimum [2].

The Quantum Annealing (QA) metaheuristic algorithm tries to incorporate the annealing principle into a quantum representation of discrete optimization problems. As reported in the literature, QA outperforms SA for the TSP [3], [4] and for the Vertex Coloring Problem (VCP) [5].

QA is still a new approach. As it looks very promising, we use it for another hard problem. We are interested in a classical NP-hard problem [6], the Multidimensional Zero-One Knapsack Problem (MKP), which was already treated with SA in [7]. By doing this we show a high expressiveness of an Ising model (see [8]) applied to the combinatorial optimization and an excellent performance of QA which is obtained thanks to the tunneling effect.

The second contribution of this article is an improvement of the performance of QA. In a nutshell, we attempt to speed up the quantum descent by restraining an area of local search. We do this by imposing a restriction upon the set of possible mutations at a given step of the annealing schedule. If the mutation is about to modify the components of an intermediate solution considered essential for the quality of a final result, this mutation is simply abandoned. In other words, we introduce "a blockade" of elements of states composing a search space in order to avoid changing them. As the behavior of the annealed particle is modeled by a set of replicas it encourages us to consider as crucial these state components which appear in "sufficiently many" replicas.

As the reader might already notice, our improved QA with mutation restriction, which we call Restrictive Quantum Annealing (RQA), is dedicated to hard optimization problems where a decision can be taken about what part of an intermediate solution is liable to appear in a final solution. Therefore, our next contribution is the identification of a class of hard optimization problems whose solutions may be obtained with RQA more quickly than with the regular QA. The above mentioned TSP and MKP belong to this class.

Dealing with random algorithms, we take care to evaluate their performance in a thorough manner. As the annealing metaheuristics require parametrization, we provide guidelines to find appropriate parameter values. We believe that these indications could promote the use of QA and our RQA for a broad scope of combinatorial hard problems.

We start this article with the outline of QA taking the TSP as an example. That section also allows us to present works related to QA. Next (Section III), we present QA applied to the MKP. Our goal is twofold: we show that Ising modeling may be comfortably adapted to any discrete optimization problem and we provide a use-case to analyze the performance of SA, QA, and RQA.

The section defining RQA follows where we explain the method of identifying problems liable to be treated more efficiently with RQA than with QA. We show formally that both the TSP and the MKP are good candidates to be solved by our RQA. This section also contains a detailed description of RQA.

Section V includes a performance evaluation of SA/QA/RQA applied to TSP and MKP instances. The objective is to eventually affirm that RQA has a better performance than SA and QA if the running time is large enough.

This article then ends with a conclusion and a discussion of possible avenues for further research.

II. QA STATE OF THE ART

QA is an optimization algorithm inspired by SA and quantum mechanics. The idea is to use a quantum particle instead of a classical one as in SA. It is impossible to simulate a quantum particle with an infinite amount of replicas on deterministic computers. QA deals with its approximation based on several classical particles, called replicas, interacting in response to a kinetic effect. The quantum particle is noted ω. Each replica corresponds to a configuration ω_k (also called "state") of the considered instance. As a consequence, QA may be seen as several SAs launched in parallel with interactions between annealed particles. Unlike SA, QA does not use a thermal decrease to find a solution. As the temperature remains constant, the probability for a particle jumping over a potential barrier does not change throughout the process. Furthermore, contrary to parallel SA, the replicas' states are dependent as an additional kinetic term is introduced in the function to be minimized. This kinetic term depends on the structure of each replica and helps to explore the convex hull shaped by replicas. A possible way to find a kinetic energy formulation for the replicas is to represent them as an Ising model [8] and couple them with the Suzuki-Trotter transform [9].

A. Quantum Annealing Algorithm

QA minimizes an Hamiltonian $H(\omega)$, which is the sum of the potential $H_{\text{pot}}(\omega)$ and kinetic $H_{\text{kin}}(\omega)$ energies of a particle, $H(\omega) = H_{\text{pot}}(\omega) + H_{\text{kin}}(\omega)$, where:

$$H_{\text{pot}}(\omega) = \frac{1}{P}\sum_{k=1}^{P} H_{\text{pot},k}(\omega) = \frac{1}{P}\sum_{k=1}^{P} H_{\text{pot}}(\omega_k) \qquad (1)$$

where P defines the number of replicas. One should note that each replica has its own potential energy $H_{\text{pot},k}$ that is the function to be minimized (for example, the Hamiltonian cycle length for the TSP in Def. 1 or the utility function for the MKP in Def. 2). We need then to define H_{kin}. The kinetic term of the Hamiltonian can be represented through the Ising model [8] of the simulated quantum system. As mentioned in [5], a state is modeled by an Ising matrix S of size $N_1 \times N_2$ where N_1 and N_2 are parameters depending on the problem in question. By noting $S(k)$ the Ising matrix of the k^{th} replica, the overall kinetic energy is given by:

$$H_{\text{kin}}(\omega) = -J_\Gamma \sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\sum_{k=1}^{P-1} S_{ij}(k)S_{ij}(k+1) + S_{ij}(P)S_{ij}(1), \qquad (2)$$

where J_Γ corresponds to a first order interaction between replicas. Based on [5], we define J_Γ as: $J_\Gamma = -\frac{T}{2}\ln\left(\tanh\frac{\Gamma}{PT}\right)$. Parameter Γ decreases linearly throughout the algorithm from Γ_{start} to Γ_{end}. Consequently, as revealed by Eq. (2), each replica is coupled with two other neighboring replicas. In order to understand the interaction principle, one can visualize these replicas placed on an imaginary circle.

Once these principles posed, we focus on the QA algorithm (Algorithm 1). We need to select replica i for each iteration step (line 9). For this replica, we choose a mutation then calculate the $\Delta H_{\text{pot},i}$ brought by this mutation (line 10). If $\Delta H_{\text{pot},i}$ is negative, we accept the mutation otherwise we compute ΔH_{kin} (with its neighbors) and we accept the mutation if ΔH is negative or with the Metropolis probability $e^{-\frac{\Delta H}{T}}$ where $\Delta H = \Delta H_{\text{pot}} + \Delta H_{\text{kin}}$ and T is a fixed temperature in QA. Once each replica has been iterated, we increase J_Γ.

1: **Input**: Problem p, Γ_{start}, Γ_{end}, fixed temperature T, number of replicas P
2: **Output**: best known solution ω_{best}
3: Initialize each replica $\omega = \{\omega_k\}$
4: Initialize best solution: $\omega_{\text{best}} = \omega_1$
5: Initialize Γ: $\Gamma = \Gamma_{\text{start}}$
6: **repeat**
7: Randomly order the replicas
8: **for** $k \leftarrow 1$ **to** P **do**
9: Choose the k^{th} replica ω_k
10: Randomly mutate ω_k
11: The mutation transforms ω_k into ω'_k
12: $\Delta H_{\text{pot}} \leftarrow H_{\text{pot}}(\omega'_k)$ - $H_{\text{pot}}(\omega_k)$
13: $\Delta H \leftarrow H(\omega') - H(\omega)$
14: **if** $\Delta H_{pot} < 0$ **or** $\Delta H \leq 0$ **or** $e^{\frac{-\Delta H}{T}} \geq$ random() **then**
15: $\omega \leftarrow \omega'$
16: **if** $H_{pot}(\omega'_k) < H_{pot}(\omega_{best})$ **then**
17: $\omega_{\text{best}} \leftarrow \omega'_k$
18: **endif**
19: **endif**
20: **endfor**
21: Decrease Γ
22: **until** $\Gamma = \Gamma_{end}$
23: **return** Best solution: ω_{best}

Algorithm 1: Quantum annealing implementation inspired by [5]

B. Application to the TSP

The traveling salesman problem could be described as follows: finding the Hamiltonian cycle of the minimal length in a weighted graph. Put formally:

Definition 1: **Traveling Salesman Problem (TSP)**
Input: A weighted graph G of n nodes defined by the symmetric adjacency matrix $D = [d_{ij}]$, where weight d_{ij} is a distance between nodes i and j.
Objective: Minimize $\sum_{(i,j)\in r} d_{ij}$ when r is a Hamiltonian cycle in graph G.

In this context, QA works on a TSP particle for which replicas ω_k are Hamiltonian cycles. If we consider the potential aspect only (we will handle the kinetic energy explicitly below), replicas are treated independently, one after another, and QA behaves like P simulated annealings launched in parallel. As a consequence, we will define the potential energy of a replica as the length of the Hamiltonian cycle: $H_{\text{pot}}(\omega_k) = \sum_{(i,j)\in\omega_k} d_{ij}$. The global potential energy of our

particle of P replicas is the mean of potential energies of all the replicas given by Eq. (1).

The mutation used to explore the potential surface is 2-*opt move*, introduced in [10]. We explain it by considering two edges chosen randomly: $a \leftrightarrow b$ and $c \leftrightarrow d$. The pairwise exchange is performed by replacing these edges by $a \leftrightarrow c$ and $b \leftrightarrow d$. Note that this action preserves the Hamiltonian cycle property. According to [11] and confirmed by our experiments, this mutation is more efficient than 3-*opt move* or a swap of two nodes because its execution only changes two edges of the initial Hamiltonian cycle while the others modify three or four edges, respectively.

The kinetic energy, which furthers similarities between replicas [3], [4], is defined by means of the Ising model. A Hamiltonian cycle is represented as an $n \times n$ symmetric matrix S whose elements are spins ± 1. We fill S_{ij} with $+1$ when the edge $i \leftrightarrow j$ is present inside the tour, -1 otherwise. For example, the Hamiltonian cycle $1 \leftrightarrow 4 \leftrightarrow 3 \leftrightarrow 5 \leftrightarrow 2$ ($n = 5$) is described by:

$$S = \begin{pmatrix} -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 \\ -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 & -1 \end{pmatrix}$$

Computing the kinetic energy of the TSP quantum particle, i.e. the set of P Hamiltonian cycles, means multiplying the spins of each edge of the coupled replicas. Focusing only on two Hamiltonian cycles, their coupling contributes to the global kinetic energy with a quantity $-\sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}S'_{ij}$. Ultimately, taking into account all the replicas, we apply Eq. (2).

The kinetic energy may stand for a measure of the similarity of the Hamiltonian cycles. If all the replicas are identical, the kinetic energy is minimal. In contrast, if no replicas share an edge in their Hamiltonian cycles, the kinetic energy reaches its maximal value: $-J_\Gamma \frac{Pn(n-1)}{2} \leq H_{kin} \leq J_\Gamma(-\frac{Pn(n-1)}{2} + 4n)$. As the algorithm minimizes the Hamiltonian, the replicas gradually become similar. Thanks to this property, a random walk over the potential surface can jump over energy barriers.

III. Application of QA to the MKP

The objective of this section is to introduce the necessary means to solve the MKP with QA. It is worth being mentioned because QA can be difficult to apply due to the definition of the kinetic energy. Furthermore, the use of the Ising model with the MKP needs to be defined. The Multidimensional Zero-One Knapsack Problem (MKP) consists in packing n items into the c-constraint bag with the maximization of the gain. Put formally:

Definition 2: **Multidimensional Zero-One Knapsack Problem (MKP)**

Input:

1) n items to be packed,
2) vector U_i of positive utilities of these items,
3) vector M_j of c positive constraints,
4) matrix $[p_{ij}]$ of size $n \times c$ of positive weights; p_{ij} indicates a cost of inserting item i with reference to constraint j.

Objective: Find an assignment $x = (x_i)_{1 \leq i \leq n} \in \{0,1\}^n$ that maximizes: $\sum_{i=1}^{n} U_i x_i$ under the constraints $\sum_{i=1}^{n} p_{ij} x_i \leq M_j, \forall j \in [\![1, c]\!]$.

Problem modeling: An MKP replica ω_k corresponds to the list of items which are packed in the bag. It is represented by an Ising vector v of length n. Technically speaking, it is coded on spins ± 1, not on bits $0/1$ as announced in the classical definition above.

For example, for $n = 5$, vector $v(k) = (-1, 1, 1, -1, 1)$ represents a configuration of an MKP replica k in which the items numbered by 2, 3 and 5 are packed in the bag. We use both of the notations, $x(k)$ and $v(k)$, in the formulæ below. The potential energy of a replica is opposite to the utility function from Def. 2 because H_{pot} is to be minimized. For all P replicas we get:

$$H_{\text{pot}} = -\frac{1}{P}\sum_{k=1}^{P}\sum_{i=1}^{n} U_i x_i(k) = -\frac{1}{P}\sum_{i=1}^{n} U_i \sum_{k=1}^{P} x_i(k) \quad (3)$$

To obtain the global kinetic energy, we multiply the spin vectors of MKP replicas in a way similar to the one we used for the matrices in the TSP case, Eq. (2):

$$H_{\text{kin}} = -J_\Gamma \sum_{k=1}^{P-1}\sum_{i=1}^{n} v_i(k)v_i(k+1) - \sum_{i=1}^{n} v_i(P)v_i(1). \quad (4)$$

The H_{kin} role is to strengthen similarities existing between MKP replicas.

Mutation used: According to our prior experience, mutations which modify a configuration as little as possible lead to efficient annealing processes. For this reason we want our mutation to take account of one item at a time. When an insertion of an item is not possible, we opt to exchange it with another one which has been packed already. We try to avoid removing an item from the bag because such an action could significantly decrease the utility value. The mutation is comprised of three steps:

1) Choose uniformly any item a which is not in the bag. Insert it into the bag if the constraints are respected.
2) If this insertion violate any of the constraints, choose uniformly an item b which is inside the bag and try to exchange a and b.
3) If this exchange is not possible, remove item b from the bag.

Our mutation is similar to the one proposed in [7] for the SA algorithm called `PROEXC` with a slight difference: in specific situations, `PROEXC` makes several evaluations of the Metropolis probability $e^{-\frac{\Delta H}{T}}$ per iteration.

IV. A QA improvement: Restrictive QA

In Section II, we explained the principle of QA is to decrease a Hamiltonian with potential and kinetic parts. The kinetic energy allows the different replicas to cross the region bounded by the quantum particle and to explore possible new minimizers. Nevertheless, the kinetic energy is not based on the quality of all the configurations of the particle: it only focuses on the similarities between these replicas. Given that

$0 \leftrightarrow 7 \leftrightarrow 9 \leftrightarrow 8 \leftrightarrow 10 \leftrightarrow 12 \leftrightarrow 6 \leftrightarrow 11 \leftrightarrow 5 \leftrightarrow 4 \leftrightarrow 3 \leftrightarrow 2 \leftrightarrow 13 \leftrightarrow 1$ len: 3345
$0 \leftrightarrow 1 \leftrightarrow 2 \leftrightarrow 3 \leftrightarrow 4 \leftrightarrow 5 \leftrightarrow 11 \leftrightarrow 13 \leftrightarrow 6 \leftrightarrow 12 \leftrightarrow 7 \leftrightarrow 10 \leftrightarrow 8 \leftrightarrow 9$ len: 3457
$0 \leftrightarrow 1 \leftrightarrow 13 \leftrightarrow 2 \leftrightarrow 3 \leftrightarrow 4 \leftrightarrow 5 \leftrightarrow 11 \leftrightarrow 6 \leftrightarrow 12 \leftrightarrow 7 \leftrightarrow 10 \leftrightarrow 8 \leftrightarrow 9$ len: 3330

Figure 1: Three solutions for `burma14`

the replicas converges on areas in which the energy is low, they naturally share more and more likenesses. For example, we noticed that, for the TSP, Hamiltonian cycles are composed of the same "key edges". In Fig. 1 are three configurations close to the optimal one (obtained through SA) on the small graph `burma14` taken from [12].

One may observe that the edges $8 \leftrightarrow 9$, $3 \leftrightarrow 4$ and $6 \leftrightarrow 12$ appear in all these solutions. This means that there is a natural convergence due to potential energy only. The problem of an independent kinetic term is that it forces replicas to be similar independently of the potential energy value. We would like to strengthen resemblances due to the "potential convergence" in order to boost the QA performance.

The main idea of our improvement is to block the elements that are frequent in the particle by avoiding mutations that will remove these particular elements. For the TSP, the elements blocked are edges. If we see the three solutions of `burma14` as a quantum particle of three replicas, the edges $8 \leftrightarrow 9$, $3 \leftrightarrow 4$ and $6 \leftrightarrow 12$ will be considered as blocked: any mutation that tries to remove these edges will be forbidden. For instance, the following *2-opt move* (cf. Section II-B) on the first replica will not be treated because it would remove the edge $8 \leftrightarrow 9$: a = 8, b = 9, c = 13, d = 2. For the MKP, the elements blocked are the items that are packed: a mutation which removes an item that is packed in a great number of replicas will be forbidden by RQA.

Such an approach allows us to contract the space of available mutations. Using RQA requires the definition of a type of element which we propose as follows:

Definition 3: **of an element of an instance**
e is an element of an instance if it is a subset of at least one configuration ω_0 for this instance.

To benefit from RQA performance improvement, the important issue is to determine numerous types of elements (i.e. elements defined by a certain property) on which a blockade should be set. We formulate two criteria which allow us to identify the elements which may be efficient. With this definition, for the TSP, we can imagine several kinds of elements: edges ($e \in E$ with a graph $G = (V, E)$), couples of edges ($e \in E \times E$) or nodes ($e \in V$), for example. Taking nodes as elements for TSP instances would be pointless as all nodes form an Hamiltonian cycle: they are always present. Edges, being subject to mutations, are naturally good candidates to be instance elements. That is why the first criterion we introduce is the ability for the type of element chosen to affect as directly as possible the energy of the configurations ω_0 for a given instance. The second aspect, which can be quantified (see Section IV-A), is the frequency of occurences of a given element in the configuration: the observed similarities cannot be due to chance but have to be caused by the minimization of H_{pot}. To quantify this aspect, we define the characteristic probability for an element to be naturally inside any state.

A. Definition and properties of the characteristic probability

We propose a characteristic probability $\alpha(e)$ which allows us to check if the second criterion is verified. It measures the probability for e to be in a random configuration for the given instance. The goal is to determine which value of α may make the restriction of RQA efficient.

Definition 4: **Characteristic probability of the elements**
Let us consider an element e and Ω_0 the uniform random variable giving all the configurations of the instance, i.e. the possible valid solutions to the problem. The characteristic probability of e, noted $\alpha(e)$, is defined as: $\alpha(e) = P(e \in \Omega_0)$

If $\alpha(e)$ is small, it means that e appears rarely in any configuration. If an element with a weak α is frequent in the quantum particle, this element may be essential to obtain a good solution. On the contrary, if $\alpha \geq \frac{1}{2}$, it is probably observed in several replicas and it would be pointless to block it in an attempt to improve the annealing.

For any problem, there exists a characteristic probability provided that the elements concerned are defined. In some problems, like the TSP, $\alpha(e)$ only depends on the nature of the problem itself: each edge has the same probability to be, or not, in any Hamiltonian cycle: $\alpha_{\text{TSP}}(e) = \alpha_{\text{TSP}}$. In other cases, $\alpha(e)$ depends on one or several parameters that are contained by the element e. With the MKP, the elements are the items which are packed. The probability for an item to be in the bag or not depends on the weights of this item and the total number of items. If the weights are considerably large compared to the others, it will be more difficult to fill a bag with it. On the other hand, it will be easier to push it in the bag if the weights are small. We determine $\alpha(e)$ in both problems:

Proposition 1: The characteristic probability of the TSP: $\alpha_{\text{TSP}} = \frac{2}{n-1}$

Proof. We compute the probability for an edge $x \leftrightarrow y$ to be in any Hamiltonian cycle r. Any node of cycle r has obviously two neighbors. The number of possible pair of neighbors is $\binom{n-1}{2}$, n is the number of nodes in the graph. Among these pairs of nodes, some contain y: $[1, y]$, $[2, y]$,... in total $n - 2$ pairs in which x can not be included. As a consequence, we have:

$$\alpha_{\text{TSP}} = \frac{n-2}{\binom{n-1}{2}} = \frac{n-2}{\frac{(n-1)\cdot(n-2)}{2}} = \frac{2}{n-1}$$

■

Concerning the MKP, it is difficult to get a formula that gives α depending on the weights of the item p_j, all the constraints M_j, all the weights of the instance p_{ij} and the number of items n. We thus limit ourselves to a bound and an approximation. We start by treating the case of the single

constraint knapsack problem (KP). Let us note B as the set of all acceptable bags for the instance considered, B_e is the subset of B composed by bags containing the item e. $\overline{B_e}$ is the complement of B_e. We express $\alpha(e)$ as:

$$\alpha_{\text{KP}}(e) = \frac{|B_e|}{|B|} = \frac{|B_e|}{|B_e| + |\overline{B_e}|} \quad (5)$$

Proposition 2: The characteristic probability of the KP satisfies: $\alpha_{\text{KP}} \leq \frac{1}{2}$.

Proof. Let us consider the following function φ defined over the entire B_e:

$$\varphi : B_e \mapsto \overline{B_e} \text{ such that } \varphi(X) = X - \{e\}$$

B_e and $\overline{B_e}$ are finite, φ is injective: $|B_e| \leq |\overline{B_e}|$

$$\alpha_{\text{KP}}(e) = \frac{|B_e|}{|B_e| + |\overline{B_e}|} \leq \frac{|B_e|}{|B_e| + |B_e|} \leq \frac{1}{2}$$

■

The most straightforward approximation is obtained by assuming the distribution of weights as concentrated around the mean. In one dimension, if we note μ the mean of the weights p_i, we can approximate the maximal number of items in the bag by $z = \min(n, \lfloor \frac{M}{\mu} \rfloor)$ where n is the total number of items and M the constraint. Using Eq.(5):

$$\alpha_{\text{KP}} \approx \begin{cases} \frac{\sum_{i=0}^{z-1} \binom{n-1}{i}}{\sum_{i=0}^{z} \binom{n}{i}} & \text{if } z \geq 2 \\ \frac{1}{n} & \text{if } z = 1 \end{cases}$$

Then, looking at all dimensional α_{KP} allows us to interpret a general tendancy for α_{MKP}. If every α_{KP} is approximately equal to $\frac{1}{2}$, the implementation of RQA is useless.

Input: Problem p, Γ_{start}, Γ_{end}, fixed temperature T, number of replicas P, blockade frequency f
Output: best known solution ω_{best}
Initialize each replica $\omega = \{\omega_k\}$
Initialize best solution: $\omega_{\text{best}} = \omega_1$
Initialize Γ: $\Gamma = \Gamma_{\text{start}}$
repeat
  Randomly order the replicas
  for *k = 1..P* **do**
    Choose the k^{th} replica ω_k
    Choose elements ϵ_j in ω_k which appears less than $\lceil f \cdot P \rceil$ times in the particle
    Pick elements ϵ'_j that are not in ω_k
    The mutation transforms ω_k into ω'_k by turning ϵ_j in ϵ'_j
    ΔH_{pot} = $H_{\text{pot}}(\omega'_k)$ - $H_{\text{pot}}(\omega_k)$
    ΔH = $H(\omega')$ - $H(\omega)$
    if $\Delta H_{pot} < 0$ **or** $\Delta H \leq 0$ **or** $e^{\frac{-\Delta H}{T}} \geq random()$ **then**
      $\omega := \omega'$
      if $H_{pot}(\omega'_k) < H_{pot}(\omega_{best})$ **then**
        $\omega_{\text{best}} := \omega'_k$
      endif
    endif
  endfor
  Decrease Γ
until $\Gamma = \Gamma_{end}$
return Best solution: ω_{best}

Algorithm 2: Restrictive Quantum Annealing algorithm

B. RQA algorithm

RQA starts as the "classical" QA. After a certain number of iterations, some replicas could have certain number elements in common. For instance, if we consider the TSP, particular edges appear in replicas. When the elements are present in a certain percentage of replicas, we block them. The blockade will last until the end of the annealing. RQA is described in Algorithm 2.

In the MKP case, the mutation described in Section III only replaces one element in the current replica. For the TSP, we are forced to replace two elements to preserve the Hamiltonian cycle property. To execute these mutations, we choose the elements that will be replaced in the replica. They must not exceed the limit of occurrences in the particle. We define the blocking frequency f: if an element appears more than $\lceil f \cdot P \rceil$ times in the particle, we block it.

V. Experimentation and results

We now compare RQA and QA with SA for two TSP instances, taken from TSPLIB [12] (`bier127` and `pr1002`) and for two MKP instances (`OR500x30_0.75` from [13] and `gk11` from Glover and Kochenbergen which is also used in [14]).

From the experiments, we extract the potential energy of the best replica met in the exploration as a function of the iteration number. Each curve represents the mean of 100 experiments. Confidence intervals computed with a confidence level 0.05 are given in Table I. As the confidence intervals are small compared to the gaps between the curves, we do not represent them on graphics to keep the figures neat and comprehensible.

For each algorithm (SA, QA and RQA), we estimate parameters by launching a preliminary series of experiments. According to these preparatory runs, we choose the following values. The thermal function on SA is linearly decreasing from T_{start} to 0. The temperature of QA and RQA is chosen as $T_{\text{QA}} = \frac{T_{\text{SA,start}}}{5}$ in order to get good performances of QA. Initial configurations for the TSP (i.e. Hamiltonian cycles) are selected through an uniform random selection on all possible tours. For the MKP, initial bags are empties.

The computing time is the same for each algorithm. The number of iterations for each replica of QA and RQA is equal to the number of iterations used for SA divided by P, the number of replicas. We want to find a compromise between the annealing duration and the number of replicas used. There are 20 replicas for experiments on `bier127` (TSP), 10 replicas for all the other instances. It is justified by the fact that `bier127`'s size is only 127 vertices and within a fixed annealing time the simpler the instance is, the more we take replicas to catch the global minimizer. The blocking frequency chosen is 65% for `bier127`, 80% for `pr1002`

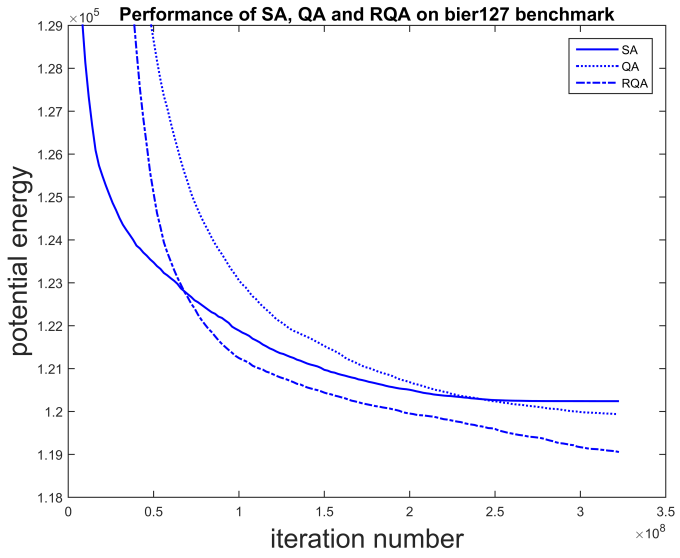

Figure 2: SA, QA and RQA runs on the TSP `bier127`

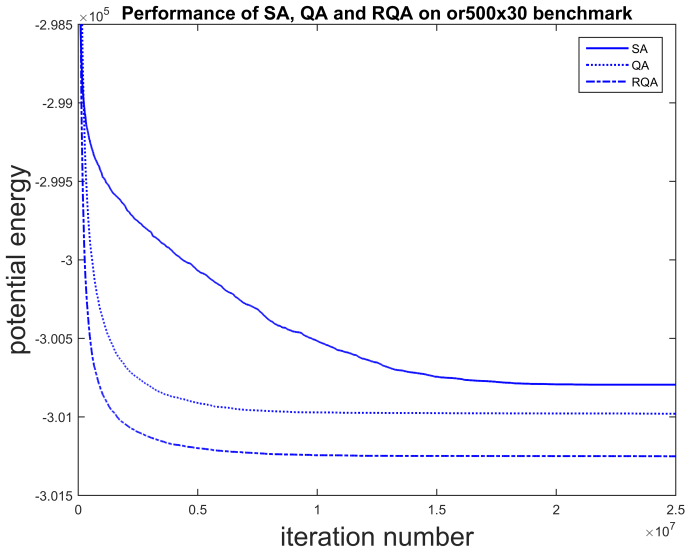

Figure 3: SA, QA and RQA runs on the MKP `OR500×30_0.75`

(listed in Table I) and 100% for all MKP instances because α is higher for this problem.

A. Performance of SA, QA and RQA

Figures 2 and 3 show that within a reasonable amount of iterations, the QA metaheuristic performs better than SA. This behavior for the TSP was already reported in [4], we confirm it for the MKP. By comparing these curves, we see that in the MKP case (Fig. 3), QA starts to have better performance than SA in the early phases of the iteration. This might be due to the specificity of the problem itself. As QA checks more valid bags with its different replicas, it has a better chance to find a more appropriate bag than SA. Moreover, by matching the similarities between replicas, QA refines them through the influence of the kinetic energy. On the other hand, SA's potential energy decreases quickly in the early iteration phases on the TSP instance while P parallel explorations slow RQA down (see Fig. 2). However, during the last iterations of the process, when an impact of H_{kin} becomes significant, QA explores the convex hull formed by the replicas and can find new minimizers while SA often does not succeed in finding better solutions from only one state. As a consequence, for the TSP and the MKP, if the number of iterations is high enough, QA will outperform SA thanks to the kinetic effect. But, if the number of iterations is too small, SA will be better than QA thanks to the fast descent observed at the beginning of the annealing. This means that, for each instance of the TSP and the MKP, there is a limit to the number of iterations (which in our experiments is greater for the TSP than for the MKP) from which we could assert that QA exceeds SA. The number of iterations we have at our disposal determines the algorithm to choose for the TSP and the MKP.

As this may be observed in Fig. 2, QA and RQA start in the same way, as in the beginning of the annealing, a chance to have similarities between replicas is very small (as we start from a random tour, replicas are far from each other). Then, as the number of iteration increases, certain edges start to be present in more than one replica, to finally overpass the limit $\lceil f \cdot P \rceil$ and get blocked. RQA's curve differs from QA's one because of the blocking action which speeds up the descent of the potential energy. The final part of RQA's curve is almost parallel to QA's one, that confirms that the action of the kinetic energy on the end of both RQA and QA is the same. As for TSP results, RQA behavior is the same as QA's in the initial iterations of the MKP run in Fig. 3. Thanks to the restriction, RQA overtakes QA to get better results. This is due to RQA's tendency to keep the best items untouched. Given that SA was worse than QA all along the algorithm execution, RQA is consequently better than SA on `OR500×30_0.75` MKP instance.

B. Overview on larger instances

The experiment concerning a very large instance of the TSP is depicted in Fig. 5. We hardly used 8 minutes of a standard PC processor to solve it. The QA results are much worse than those of SA and RQA as QA would need more time to overtake the SA potential energy. RQA, however, defeats SA after a small number of iterations. The results for a large instance of the MKP presented in Fig. 4 confirm that on the early annealing stage, useful items have not been selected yet and RQA behaves as QA. Then as the annealing process goes on, SA starts its efficient descent phase and tends to catch up QA, but then, as kinetic energy influence is stronger than that of the potential energy, QA explores more easily the energy surface and finds a better solution. As RQA enhances this part by retaining the best items, it overwhelms both SA and QA with an early descent more effective than those of QA.

The mean and best results of 100 experiments for each benchmark are reported in Tables I and II, respectively. Each

	TSP		MKP	
	bier127	pr1002	OR500x30_0.75	gk11
SA	$120'171 \pm 255$	$267'612 \pm 209$	$-300'762 \pm 53$	$-94'646 \pm 6.0$
QA	$119'956 \pm 128$	$285'728 \pm 1013$	$-300'973 \pm 24$	$-94'665 \pm 3.1$
RQA	$119'081 \pm 115$	$267'364 \pm 338$	$-301'248 \pm 17$	$-94'808 \pm 2.6$

Table I: Summary of results of H_{pot} for the benchmarks discussed

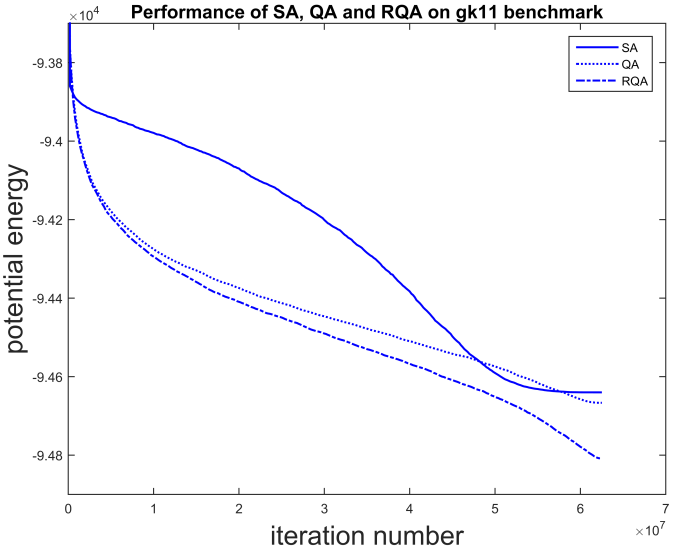

Figure 4: Comparing performances of SA and QA and RQA on MKP gk11

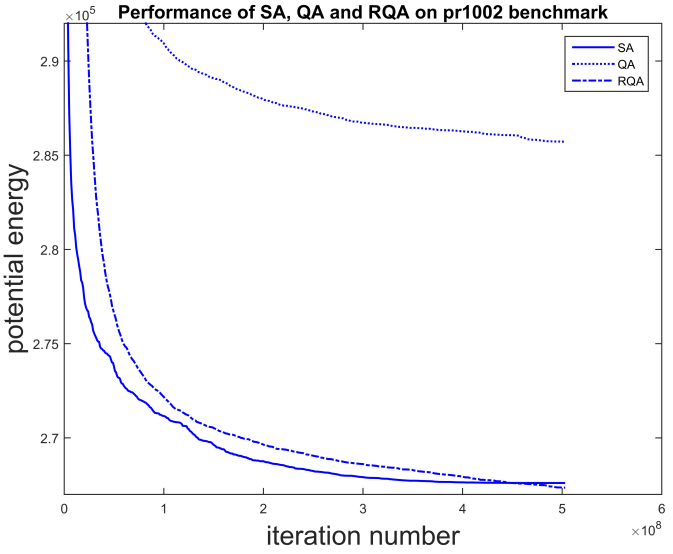

Figure 5: Comparing performances of SA and QA and RQA on TSP pr1002

	TSP		MKP	
	bier127	pr1002	OR500x30_0.75	gk11
SA	$118'336$	$265'885$	$-301'242$	$-94'695$
QA	$118'815$	$276'749$	$-301'217$	$-94'704$
RQA	$118'293$	$264'631$	$-301'422$	$-94'832$

Table II: Best results

Blck freq	Mean	±CI
none	$121'060$	$\pm$ 129
100%	$120'490$	$\pm$ 115
85%	$119'590$	$\pm$ 116
65%	$\mathbf{119'150}$	$\pm$ **88**
50%	$119'315$	$\pm$ 89
35%	$119'440$	$\pm$ 104

Blck freq	Mean	±CI
none	$288'700$	$\pm$ 1059
100%	$279'850$	$\pm$ 663
90%	$271'350$	$\pm$ 440
80%	$\mathbf{266'400}$	$\pm$ **285**
70%	$270'164$	$\pm$ 324
60%	$270'750$	$\pm$ 402

Table III: Performance of RQA on bier127 (left) and pr1002 (right) depending on blocking frequency for the TSP

experiment takes less than eight minutes (this maximal time is for pr1002). We compare it with the results found in the literature to be sure SA, QA and more particularly RQA are able to provide efficient results compared with other algorithms. The website [15] provides the best performance on well-known TSP benchmarks: $188'282$ for bier127 and $259'045$ for pr1002. For the MKP, we take results found in [16], [14]. Let us note that the potential energy obtained on series of experiments allows us to approach closely performances of the literature (OR500x30_0.75 the results obtained are even better than those from [16]). To compare with the hybrid approach from [14], the mean on 100 results provided by five-minute experiments on four double-thread cores at 2.4 GHz for gk11 is $94'808$ in Table I, the result is $94'832$ (Table II) while the hybrid algorithm found $95'237$ (0.45% gap) with an order of magnitude of a day.

C. Results according to blocking frequency

The blocking frequency defines the number of replicas in which elements has to blocked (Table III). There is an optimal frequency for a given experiment which depends not only on the problem, but on a particular instance and the number of replicas as well. We tried to determine this optimal frequency on the two TSP instances bier127 and pr1002, by launching 100 RQAs with different frequencies. The 95% confidence interval is noted "CI".

Regarding pr1002, experiments were made with $P = 10$ replicas, that is why the values of blocking frequency are separated by a 10% step (as a consequence, $\lceil f \cdot P \rceil$ in Algorithm 2, line 10, take all natural numbers between 1 and P). 20 replicas were taken for bier127 experiments: the frequencies should

be separated by a 5% step but for clarity of presentation, we limit ourselves to a selection of frequencies in Table III. These results illustrate clearly the compromise which has to be made between a weak blockade which does not allow one to get the best performance and a strong blockade which does not block the proper elements and degrades the final potential energy. This also shows that choosing a high frequency ($> 90\%$) leads to the QA improvement in any case.

VI. Conclusion and further work

The QA metaheuristic is a new method which has not reached yet the popularity of the classical SA. We showed that it is worth of a wider interest as it may be easily adapted to solve any NP-hard problem. We implemented QA for the MKP and our experiments affirm that QA outperforms SA. Our experimental evaluation, done for the MKP and the TSP, leads to the conclusion that if the number of iterations is sufficiently large, the QA result is better than that given by SA.

In order to compensate relatively slow early stage descent of QA, we propose a method to boost this metaheuristic, RQA. By reducing the size of the search-space, we quantitatively improve the quality of the replicas in the early phases of RQA. The restriction is also advantageous since it brings better overall results. The mutation restriction was used in the QA context but it is not excluded that other evolutionary algorithms might benefit from it as well.

However, the RQA efficiency depends on the problem we treat. We proposed a means (the characteristic probability) which allows one to predict whether the mutation restriction is worth being implemented. Provided a sufficient number of iteration and with an appropriate value of the characteristic probability, RQA performs better than both SA and QA. Furthermore, the efficiency of RQA depends on the blocking frequency that should be tuned relatively to the number of replicas used. Anyway, choosing a high blocking frequency allows one to defeat SA in many cases even if it does not reach its best performance. The RQA performance was evaluated on the NP-hard problems MKP and TSP on two benchmarks each and was always better than both QA and SA. The RQA improvement is very significant for TSP and MKP and we assume it could bring better results than SA or QA to other combinatorial problems.

We think that SAT (Boolean Satisfiability Problem) is a problem for which RQA is not efficient given that the characteristic probability is too large and equal to $\frac{1}{2}$. But it would be particularly interesting to know if it can be efficient for small values of the characteristic probability (negligible compared to $\frac{1}{n}$). Concerning this case, the VCP (the reader is invited to refer to [5] which solved this problem with QA) could be, for example, a good candidate: elements could be classes of nodes colored in the same way but it seems to be, at a first sight, quite hard to implement.

Acknowledgements

The results presented in this paper were partially obtained within a CentraleSupelec students' long-term research project.

References

[1] M. Garey and D. Johnson, *Computers and Intractability : A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[2] S. KirkPatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 671–680, 1983.

[3] A. Das and B. Chakrabarti, "Quantum Annealing and Analog Quantum Computation," *Rev. Mod. Phys.*, vol. 80, p. 1061, 2008.

[4] R. Martonak, G. E. Santoro, and E. Tosatti, "Quantum annealing of the Traveling Salesman Problem," *Phys. Rev. E*, vol. 70, 2004.

[5] O. Titiloye and A. Crispin, "Quantum annealing of the graph coloring problem," *Discrete Optim.*, vol. 8, pp. 376–384, 2011.

[6] R. M. Karp, "Reductibility among combinatorial problems," *Complexity of Computer Computations*, pp. 85–103, 1972.

[7] A. Drexl, "A Simulated Annealing Approach to the Multiconstraint Zero-One Knapsack Problem," *Computing*, vol. 40, pp. 1–8, 1988.

[8] B. A. Cipral, "An Introduction to the Ising model," *Amer. Math. Monthly*, vol. 94, no. 10, pp. 937–959, 1987.

[9] M. Suzuki, "Relationship between d-Dimentional Quantal Spin Systems and (d+1)-Dimensional Ising System," *Prog. Theor. Phys.*, vol. 56, no. 5, pp. 1454–1469, 1976.

[10] G. A. Croes, "A method for solving Traveling-Salesman Problem," *Oper. Res.*, vol. 6, pp. 791–812, 1958.

[11] P. Tian, J. Ma, and D.-M. Zhang, "Application of the simulated annealing algorithm to the combinatorial optimisation problem with permutation property: An investigation of generation mechanism," *Eur. J. Oper. Res.*, vol. 118, pp. 81–94, 1999.

[12] G. Reinelt. (2013) TSP library. [Online]. Available: http://elib.zib.de/

[13] P. C. Chu and J. E. Beasley, "A genetic algorithm for the multidimensional knapsack problem," *J. Heuristics*, vol. 4, no. 1, pp. 63–86, 1998.

[14] M. Vasquez and J. Hao, "A hybrid approach for the 0–1 multidimensional knapsack problem," in *Proc. of IJCAI*, 2001, pp. 328–333.

[15] D. D. Universität Heidelberg and combinatorial optimization. (2007) Optimal solutions for symmetric tsps. [Online]. Available: http://comopt.ifi.uni-heidelberg.de/

[16] J. E. Gallardo, C. Cotta, and A. J. Fernández, "Solving the multidimensional knapsack problem using an evolutionary algorithm hybridized with branch and bound," in *Artificial Intelligence and Knowledge Eng. App.: A*, 2005, pp. 21–30.