

FINDING MINIMA IN COMPLEX LANDSCAPES: ANNEALED, GREEDY AND RELUCTANT ALGORITHMS

PIERLUIGI CONTUCCI* and CRISTIAN GIARDINÀ†

*Dipartimento di Matematica, Università di Bologna,
Piazza di Porta S. Donato 5, 40127 Bologna, Italy*

**contucci@dm.unibo.it*

†giardina@dm.unibo.it

CLAUDIO GIBERTI

*Dipartimento di Informatica e Comunicazione,
Università dell'Insubria,*

via Mazzini 5, 21100 Varese, Italy

claudio.giberti@uninsubria.it

CECILIA VERNIA

Dipartimento di Matematica Pura ed Applicata,

Università di Modena e Reggio Emilia,

via Campi 213/B, 41100 Modena, Italy

vernia@unimore.it

Received 23 August 2004

Revised 15 December 2004

Communicated by N. Bellomo

We consider optimization problems for complex systems in which the cost function has a multivalleyed landscape. We introduce a new class of dynamical algorithms which, using a suitable annealing procedure coupled with a balanced greedy-reluctant strategy drive the systems towards the deepest minimum of the cost function. Results are presented for the Sherrington–Kirkpatrick model of spin-glasses.

Keywords: Combinatorial optimization; statistical mechanics; spin glass.

AMS Subject Classification: 82B44, 82B80, 90C27

1. Introduction

There is a standard barrier in applied science: the computational complexity of hard (non-polynomial) problems. The modelling of competing interactions among the components of a large system often leads to consider the solution of a problem as the minimum of a functional with a complex landscape. The extensive search for the optimal configurations has a cost that grows too quickly (usually exponentially) and

become practically intractable when the number of composing units is of the order of a few hundreds as in the interesting cases. The study of optimizing algorithms is then a basic step toward the solution of specific practical problems emerging in different fields of applied science.

In this paper we build a strategy to efficiently explore the landscape of complex functionals in combinatorial optimization in order to find its minima both local and global. To allow the reader to better focus on our method, let us describe the functional to be minimized as the mathematical representation of a quickly changing mountain profile (in large dimensions), with a high multiplicity of local minima separated by high barriers. The *a priori* knowledge of the landscape geometry is very poor and our strategy to explore the territory in order to find good quality minima (close to the global one) is to send signals in random directions (initial configurations), follow their evolution according to a specified dynamics (algorithm) and collect the observed results. Our investigation procedure is not dissimilar from an optical instrument in which we may tune a few parameters to better observe the landscape and find the sites which we are interested in.

The algorithm is preliminary set by choosing the elementary dynamical moves: this choice reflects the topology that we are associating to our landscape and comes with a notion of vicinity and nearest neighboring sites. The successive step is to decide the criteria after which to select among a large multiplicity of moves. This is done by keeping into account what we search for and what we most fear: we want to reach the best possible minima as quickly as possible and the worse to happen is to get stuck in a local minimum which is still far from the optimal or near optimal ones. It appears rather intuitive that an algorithm with a too steep descent (greedy) has a very high risk to get stuck in poor local minima, but at the same time a too slow descent (reluctant) would cost a very high price in terms of computer time. It is natural to expect, and indeed it is what we find, an optimal speed of descent that compromises at best among having a wide exploration basin in a reasonable amount of time. Yet the danger of remaining caught in wrong local minima remains. To avoid it we also allow moves which locally and momentarily deviate from the descending directions. In other terms: to reach a good minimum it is often necessary to overcome a high barrier. Physically, the introduction of a similar possibility works like the availability of thermal energy where the probability of its happening is related to the temperature of the system: the higher the temperature the more likely are moves upwards and vice versa.

To introduce such a useful strategy we initially allow upward and downward moves; with the time passing, the probability to go up is progressively decreased at a rate which we may optimize (this simulates the annealing of a physical system) and the algorithm will continue evolving according to its downward moves. Our work and the implementation of the algorithm is built and tested toward a standard model in combinatorial optimization with origins in condensed matter physics: the Sherrington–Kirkpatrick (SK) model for the mean field spin glass phase.^{1,2} Among the advantages of our approach, there is the flexibility of our algorithms and their

wide applicability to practical problems like protein folding in biology,² portfolio optimization in financial mathematics,³ error correcting codes for digital signal transmissions.⁴

2. Results

In the following sections we will present details of the Model and Algorithms we used in our simulations. Here we summarize the main ideas and results of our analysis.

In the SK model the cost function is identified with the energy of the system, the domain of the cost function is the discrete spin configuration space and the optimization problem amounts to finding the spin configuration with the lowest energy (ground state). Given a proper definition of distance in the configuration space (we can think of two spin configurations to be close if they differ only for a single spin-flip), the energy of the system is a real-valued function forming a complex and corrugated energy landscape, with valleys (local minima) and peaks (local maxima). Our optimization algorithms are described as dynamical evolution rules in this energy landscape which, starting from a random initial condition, drive the system towards local minima of the energy. The random transition from a point of the trajectory to the successive, which is a nearest neighboring one, is ruled by a probability with exponential density. We consider four different algorithms: starting from the simplest one (Algorithm 0) which allows only energy-decreasing trajectories, we implement a sequence of refinements (Algorithms 1–3) leading to more efficient strategies, which exploit also increases in the cost function.

With Algorithm 0 the cost-decreasing trajectory ends up as soon as it reaches a configuration which, according to our notion of vicinity (see Sec. 3) is a local minimum. The parameter controlling the transition probability function tunes the steepness of descents, generating a continuum of behaviors ranging from a reluctant-type dynamics (very small jumps and slow convergence) to a greedy-type one (very large jumps deep into a valley).

A first improvement of this strategy, implemented in Algorithms 1 and 2, is obtained by introducing a “temperature” in the system, which enables random positive fluctuations of the cost function. This is obtained through the choice of a transition probability which gives a nonzero weight to upwards moves. With these choices we have the following scenario for Algorithms 1 and 2: the dynamics starts with a given initial temperature and equal probability of positive and negative moves. As the time goes on, the system is gradually cooled until it reaches a state in which positive fluctuations are forbidden and the dynamics continues as either greedy or reluctant, depending on the initial temperature. With a high initial temperature the long term behavior of the dynamics will be greedy-like, while a low initial temperature will lead to reluctant-type motion. The difference between Algorithms 1 and 2 lies in the convergence criterium: while the former stops when the first local minimum is attained (likewise Algorithm 0), the latter

allows the trajectory to escape from it in view of the possibility to reach deeper minima (supplementary stopping conditions are required in this case).

A further improvement of the algorithm efficiency is obtained with Algorithm 3. In this case, the transition probability is designed to model an initially hot system with high probability of positive moves, which is gradually quenched; when the system is cool, positive fluctuations are absent and the decreasing trajectories are forced to follow greedy-like paths. In Fig. 1 typical trajectories for the four different algorithms are reported.

The efficiency of the algorithms are quantified on one hand by measuring the average time needed to reach a local minimum, on the other hand by the quality of the found minima (i.e. how deep they are). The optimization is done by tuning the parameters which control the transition probabilities; in particular, for Algorithms 1 and 2 this parameter is mainly the initial temperature, while for Algorithm 3 it is the rate of the quench, i.e. the speed of convergence to zero of the temperature of the system. As one would expect, for low initial temperatures (very low possibility of energy increase), Algorithms 1 and 2 behave very similar to Algorithm 0. However, their differences become effective for sufficiently high initial

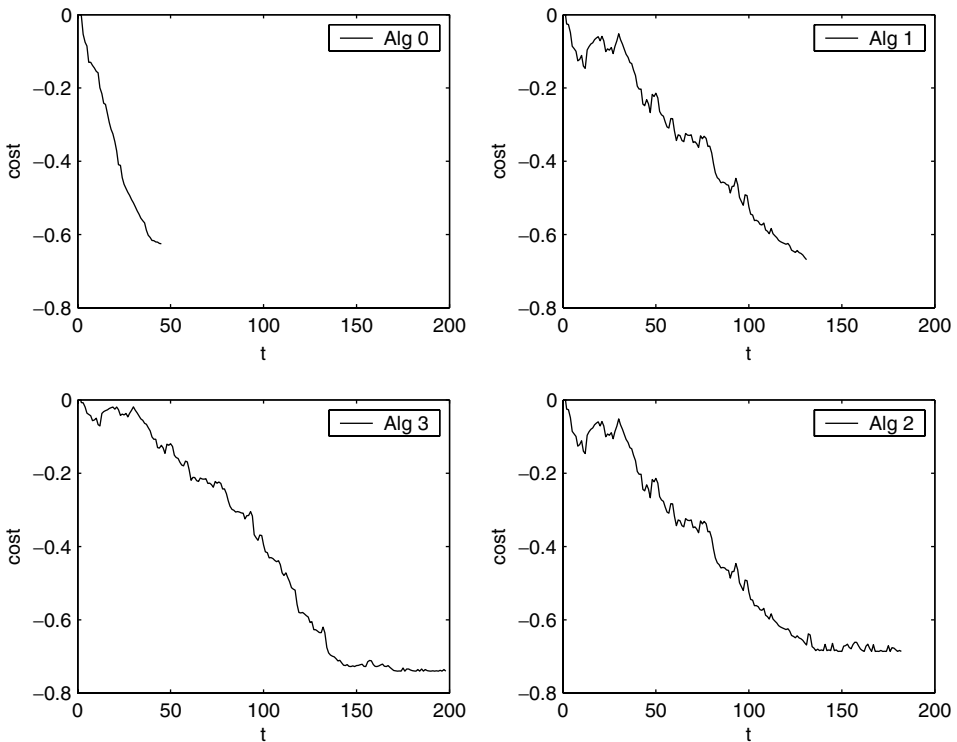


Fig. 1. Typical trajectories to reach a local minimum configuration for Algorithms 0, 1, 2 and 3. Note that the trajectories generated by Algorithms 1 and 2 coincide until the first minimum is reached.

temperatures. Obviously, allowing positive jumps and escapes from local minima, the relaxation times increase passing from Algorithm 0 to Algorithm 2; less trivially, numerical results show that the scaling of the execution times with respect to the system size is greatly enhanced. This is an important fact, because it suggests that a crossover between computation times is to be expected for systems with larger sizes. As regard to the lowest values found, similar conclusions can be drawn: going from Algorithm 0 to Algorithm 2 deeper minima are attained.

Algorithm 3 can be consistently compared with Algorithm 2, which is the best performing among the first three. The computation times and their scaling with the size are similar for the two algorithms when the initial temperature (for Algorithm 2) is high, but a clear enhancement is obtained by Algorithm 3 when it is low. Also the minimal values of the cost functional are similar for high temperatures, while they are lower for Algorithm 2 with low initial temperatures. The previous remarks refer to an experimental protocol in which the search for low cost configurations is performed testing a fixed number of trajectories. The minimization of cost at fixed elapsed computer time is another relevant criterium for the comparison of the algorithms. In this case the best result is obtained with Algorithm 3, even though Algorithm 2 gives comparable results.

3. The Model and the Algorithms

3.1. The Sherrington–Kirkpatrick model

The system we study is the SK model of spin-glasses.¹ It is defined by the Hamiltonian

$$H(J, \sigma) = -\frac{1}{\sqrt{N}} \sum_{1 \leq i < j \leq N} J_{ij} \sigma_i \sigma_j, \quad (3.1)$$

where $\sigma_i = \pm 1$ for $i = 1, \dots, N$ are Ising spin variables which interact through couplings J_{ij} . These are Gaussian random variables, independent and identically distributed with zero mean and variance 1. The random sign (and strength) of the interaction generates frustration in the system, i.e. the fact that in low energy configurations some of the couples will have unsatisfied interaction. In particular, the ground state of the system is far from the standard ground state of ferromagnetic models, where all spins point in the same direction. The model has been solved through the replica symmetry breaking ansatz by Parisi,² while the rigorous solution is still a debated issue in the mathematical physics community. From the numerical point of view, the model poses amazing difficulties and indeed it is often presented as the standard example of NP-problems. Several numerical studies have tried different algorithms in the search of ground-state energies, for example gradient descent,^{5,6} simulated annealing,^{7,8} genetic algorithms,^{9,10} branch-and-bound algorithm¹¹ and extremal optimization.^{12–14} The study that we present here does incorporate in itself the possibility to recover as special cases some of the strategies used in the above-mentioned works. For example (see Sec. 3.2) our algorithm has

a gradient descent regime when we subpress the positive energy jumps. On the other hand, our annealing procedure is a generalization of the standard simulated annealing whose only optimization parameter is the temperature. We want to stress that the spirit of this work is to build a flexible algorithm for the investigation of complex landscapes emerging in NP complete problems. Our purpose is to provide a useful tool for applications at finite volumes and not the study of the physical properties of the model, especially the limiting (infinite volume) thermodynamic properties. On the other hand, once the performance of the algorithm is known for different volumes the study of the statistical mechanics of the model become more accessible. We plan to return to these important topics in a future work.

3.2. Dynamical algorithms

We focus our attention on stochastic dynamics that generates a sequence of spin configurations ending up on a local energy minimum. The smooth interpolation between greedy and reluctant dynamics studied in previous works^{15–17} follows an energy-decreasing trajectory and terminates in the first local minimum it encounters: only transitions corresponding to a decrease in the cost (energy) function are allowed by the algorithm. In the same spirit of Simulated Annealing strategies,⁷ where a slow decrease of the temperature leads the system through successive metastable states with lower and lower energy, we think of a class of algorithms which also accept, in some limited way, transitions corresponding to an increase in the cost function. In fact, these algorithms are based on the statistical properties of metastable states: they are organized with some structure so that the evolution dynamics can be considered as the overlapping of a “fast” motion in the basin of attraction of a local minimum and of a “slow” motion with jumps between minima (the time of the dynamics is determined by the energy barriers between these metastable states).

In the algorithms that we are going to introduce, the transition between the spin configuration at time t , $\sigma(t) = (\sigma_1(t), \dots, \sigma_N(t))$, and the successive configurations at time $t+1$, $\sigma(t+1) = (\sigma_1(t+1), \dots, \sigma_N(t+1))$ depend on the spectrum of energy changes of $\sigma(t)$, obtained by flipping the spin in position i , for $i = 1, \dots, N$:

$$\Delta E_i = \frac{2}{\sqrt{N}} \sigma_i(t) \sum_{j \neq i} J_{ij} \sigma_j(t). \quad (3.2)$$

Let us also define $\Delta E_{\bar{i}} = \min_{1 \leq i \leq N} \Delta E_i$ that will be used in what follows.

As a first step, let us briefly recall the algorithm studied in Ref. 17, where only energy decreasing trajectory are considered. It is described by the following procedure:

Algorithm 0

- (1) Initialization: choose an initial spin configuration $\sigma(0)$ and a parameter value for $\lambda > 0$.

(2) Generate a random number D with probability density

$$f(x) = \begin{cases} \lambda e^{\lambda x} & \text{if } x \leq 0, \\ 0 & \text{if } x > 0. \end{cases} \quad (3.3)$$

(3) Select the site i^* associated with the closest energy change to the value D , i.e.

$$i^* : |\Delta E_{i^*} - D| = \min_{i \in \{1, \dots, N\}} \{|\Delta E_i - D| : \Delta E_i < 0\}. \quad (3.4)$$

(4) Flip the spin on site i^* :

$$\sigma_i(t+1) = \begin{cases} -\sigma_i(t) & \text{if } i = i^*, \\ \sigma_i(t) & \text{if } i \neq i^*. \end{cases} \quad (3.5)$$

(5) If $\Delta E_i > 0, \forall i = 1, \dots, N$, then the algorithm stops ($\sigma(t)$ is a local minimum); otherwise repeat from step 2.

The dynamics generated by this algorithm follows a 1-spin flip decreasing energy trajectory and arrives at a configuration whose energy cannot be decreased by a single spin-flip. The control parameter λ in the probability distribution function for the move acceptance, tunes the speed of convergence to local energy minima: the larger λ , is the bigger is the probability of doing small energy-decreasing steps, so that the trajectory will follow an evolution path close to level curves (reluctant) while, small values of λ enrich the probability of large negative energy steps (greedy), which will quickly drive the dynamics to the endpoint.

As a modification of Algorithm 0 we consider two new algorithms (Algorithms 1 and 2). They generate a dynamics that follows a 1-spin flip trajectory that, in addition to energy-decreasing transitions, accepts also energy-increasing transitions with probability exponentially decreasing in time. The difference between the two is that while the trajectory of Algorithm 1 ends up in the first local minimum it encounters, in Algorithm 2 it may continue to explore the space of configurations through the visit of subsequent local minima.

Algorithm 1

(1) Initialization: choose an initial spin configuration $\sigma(0)$ and parameter values $0 < c_1(0) < \lambda_1, 0 < c_2 < \lambda_2(0)$, with the obvious constraint

$$\frac{c_1(0)}{\lambda_1} + \frac{c_2}{\lambda_2(0)} = 1. \quad (3.6)$$

In our simulation we choose λ_1 as the only free parameter, by taking $\lambda_2(0) = \lambda_1, c_1(0) = \lambda_1/2, c_2 = \lambda_1/2$. This amounts to start with an equal probability of energy decreasing and energy increasing transitions ($c_1(0)/\lambda_1 = c_2/\lambda_2(0) = 1/2$).

(2) Generate a random number D with probability function

$$f_t(x) = \begin{cases} c_1(t)e^{\lambda_1 x} & \text{if } x \leq 0, \\ c_2 e^{-\lambda_2(t)x} & \text{if } x > 0. \end{cases} \quad (3.7)$$

- (3) Select the site i^* associated with the closest energy change to the value D and with the same sign, i.e.

$$i^* : |\Delta E_{i^*} - D| = \min_{i \in \{1, \dots, N\}} \{|\Delta E_i - D| : \Delta E_i \cdot D > 0\}. \tag{3.8}$$

- (4) Flip the spin on site i^* :

$$\sigma_i(t+1) = \begin{cases} -\sigma_i(t) & \text{if } i = i^*, \\ \sigma_i(t) & \text{if } i \neq i^*. \end{cases} \tag{3.9}$$

- (5) If $\Delta E_i > 0, \forall i = 1, \dots, N$, then the algorithm stops ($\sigma(t)$ is a local minimum). Otherwise, change the parameter $\lambda_2(t)$ of the probability distribution in step 2 with a suitable scheduling, for example

$$\lambda_2(t) = \frac{\lambda_2(0)}{k^t}, \quad 0 < k < 1 \tag{3.10}$$

and return to step 2.

The trajectory generated by Algorithm 1 moves in the energy landscape (by a succession of moves which decrease and increase energy) till it arrives at a local minimum. Starting from a symmetric probability distribution for the spin-flip selection, as time goes on the probability of energy-increasing moves is decreased by the update rule (3.10).

Next, we want to consider an algorithm as the previous one but with the possibility of exploring subsequent minima. The problem one has to solve is to give an efficient criterium to stop the dynamics. We considered the following implementation:

Algorithm 2

- (1) Initialization: as in Algorithm 1. Set also $m = 1000$ and $\varepsilon = 10^{-4}$.
 (2) Generate a random number D as follows:
 with probability function

$$f_t(x) = \begin{cases} c_1(t)e^{\lambda_1 x} & \text{if } x \leq 0 \\ c_2 e^{-\lambda_2(t)x} & \text{if } x > 0 \end{cases} \quad \text{if } \frac{c_1(t)}{\lambda_1} \leq m \frac{c_2}{\lambda_2(t)} \tag{3.11}$$

and with probability function

$$f(x) = \begin{cases} \lambda_1 e^{\lambda_1 x} & \text{if } x \leq 0 \\ 0 & \text{if } x > 0 \end{cases} \quad \text{if } \frac{c_1(t)}{\lambda_1} > m \frac{c_2}{\lambda_2(t)}. \tag{3.12}$$

- (3) Select the site i^* associated with the closest energy change to the value D and with the same sign, i.e.

$$i^* : |\Delta E_{i^*} - D| = \min_{i \in \{1, \dots, N\}} \{|\Delta E_i - D| : \Delta E_i \cdot D > 0\}. \tag{3.13}$$

- (4) Flip the spin on site i^* :

$$\sigma_i(t+1) = \begin{cases} -\sigma_i(t) & \text{if } i = i^*, \\ \sigma_i(t) & \text{if } i \neq i^*. \end{cases} \tag{3.14}$$

(5) If $\Delta E_i > 0, \forall i = 1, \dots, N$, and $P_t(D \geq \Delta E_i) < \epsilon$ then Stop.

D is a random number, P_t is the cumulative function of the probability described in step 2 and ϵ is a small parameter. In other words, if we arrive at a minimum and the probability of a significant energy increasing transition from this local minimum is too small (or even zero when the energy increases are forbidden, see step 2), then the algorithm stops.

(6) Change the probability distribution (3.11) with the scheduling (3.10) for $\lambda_2(t)$ (the same scheduling used in Algorithm 1) and return to step 2.

As in Algorithm 1, the dynamics generated by this algorithm follows a 1-spin flip trajectory making a combination of upwards and downwards moves. However, in this case, the trajectory does not end up in the first 1-spin flip stable configuration it encounters, at least as long as the probability of positive moves ($c_2/\lambda_2(t)$) remains greater than a certain threshold ($1/m$ times the probability of negative moves $c_1(t)/\lambda_1$ — in our experiments $m = 1000$). With this strategy it is possible to escape from the local minima to explore the neighboring space in view of (possible) lower energy minima. When the probability of energy increases exceed this fixed threshold, from this point on, only decreases in energy are accepted and so the process terminates when the subsequent local minimum is reached. In fact, when the process starts at time $t = 0$ we choose equal probabilities $c_1(0)/\lambda_1$ and $c_2/\lambda_2(0)$ of cost-decreasing or cost-increasing moves, respectively, by settling $c_2 = \lambda_2(0)/2$. As the algorithm continues its execution, we decrease $c_2/\lambda_2(t)$ towards zero, varying the control parameter $\lambda_2(t)$ in accordance with the above-mentioned law (3.10):

$$\lambda_2(t) = \frac{\lambda_2(0)}{k^t}, \quad \lambda_2(0) = \lambda_1, \quad 0 < k < 1$$

(and keeping fixed λ_1) until $\frac{c_1(t)}{\lambda_1} \leq m \frac{c_2}{\lambda_2(t)}$; as a consequence, the probability of energy-decreasing move acceptance $c_1(t)/\lambda_1$ tends to one ($c_1(t) = \lambda_1(1 - c_2/\lambda_2(t))$). Therefore, while the speed of convergence to the local energy minima is mainly tuned by λ_1 , the vanishing velocity of the probability of energy-increasing steps is governed by the parameter k . Of course, large λ_1 (and $\lambda_2(t)$) lead to evolution paths generated by small (in absolute value) energy changes (*annealed reluctant dynamics*) and the closer k is to 1, the slower $\lambda_2(t)$ grows and then the more energy increases are enabled. When $\frac{c_1(t)}{\lambda_1} > m \frac{c_2}{\lambda_2(t)}$ the dynamics continues governed only by the parameter λ_1 , not depending on t .

We see that for Algorithm 2 the possibility to escape from the minima is effective only when λ_1 is sufficiently small (say $\lambda_1 \simeq 1$, and then $\lambda_2(0) \simeq 1$, see (3.10)). For larger values of λ_1 the possibility to explore successive minima is not exploited and both the dynamics 1 and 2 can be expected to give similar results in terms of achieved minimum energy level. In these cases, the dynamics generated by Algorithm 2 ends up naturally, after t' steps, in the first minimum it encounters, because the (step-dependent) probability $P_{t'}$ to escape from this configuration is too small;

therefore, we expect that for large values of λ_1 Algorithms 1 and 2 should be equivalent.

Since for these algorithms the speed of convergence to the finale state is governed by the probability function $f_i(x)$, we can consider a third algorithm in which the time dependence is present only in the control parameters $\lambda_i(t)$, $i = 1, 2$; in this case, starting from a (in general) nonsymmetric probability function, the dynamics evolves gradually towards a final scenario in which the system is cooled by tuning the control parameter $\lambda_1(t)$.

Algorithm 3

- (1) Initialization: choose an initial spin configuration $\sigma(0)$ and parameter values $\lambda_1(0)$, $\lambda_2(0)$ such that $1/\lambda_1(0) + 1/\lambda_2(0) = 1$. Set also $m = 1000$ and $\varepsilon = 10^{-4}$.
- (2) Generate a random number D as follows:
with probability function

$$f_i(x) = \begin{cases} e^{\lambda_1(t)x} & \text{if } x \leq 0 \\ e^{-\lambda_2(t)x} & \text{if } x > 0 \end{cases} \quad \text{if } \frac{1}{\lambda_1(t)} \leq m \frac{1}{\lambda_2(t)} \tag{3.15}$$

and with probability function

$$f(x) = \begin{cases} \lambda_1 e^{\lambda_1 x} & \text{if } x \leq 0 \\ 0 & \text{if } x > 0 \end{cases} \quad \text{if } \frac{1}{\lambda_1(t)} > m \frac{1}{\lambda_2(t)}. \tag{3.16}$$

- (3) Select the site i^* associated with the closest energy change to the value D and with the same sign, i.e.

$$i^* : |\Delta E_{i^*} - D| = \min_{i \in \{1, \dots, N\}} \{|\Delta E_i - D| : \Delta E_i \cdot D > 0\}. \tag{3.17}$$

- (4) Flip the spin on site i^* :

$$\sigma_i(t+1) = \begin{cases} -\sigma_i(t) & \text{if } i = i^*, \\ \sigma_i(t) & \text{if } i \neq i^*. \end{cases} \tag{3.18}$$

- (5) If $\Delta E_i > 0$, $\forall i = 1, \dots, N$, and $P_t(D \geq \Delta E_i) < \varepsilon$ then Stop (as in Algorithm 2).
- (6) Change the probability distribution defined in (3.15) with the same scheduling for $\lambda_2(t)$ used in Algorithm 2 and return to Step 2.

The main difference between Algorithms 2 and 3 is that in the latter, when the process starts at time $t = 0$ we have (if $\lambda_1(0) \neq 2$) different probabilities of energy-decreasing moves ($1/\lambda_1(0)$) and of energy-increasing moves ($1/\lambda_2(0)$). As Algorithm 3 continues its execution, we decrease $1/\lambda_2(t)$ towards zero, varying the control parameter $\lambda_2(t)$ in accordance with the scheduling:

$$\lambda_2(t) = \frac{\lambda_2(0)}{k^t}, \quad \lambda_2(0) = \frac{\lambda_1(0)}{\lambda_1(0) - 1}, \quad 0 < k < 1 \tag{3.19}$$

until $\frac{1}{\lambda_1(t)} \leq m \frac{1}{\lambda_2(t)}$; as a consequence, the probability of energy-decreasing move acceptance $1/\lambda_1(t)$ tends to one ($\lambda_1(t) = \frac{\lambda_2(t)}{\lambda_2(t)-1}$). Therefore, while the speed of

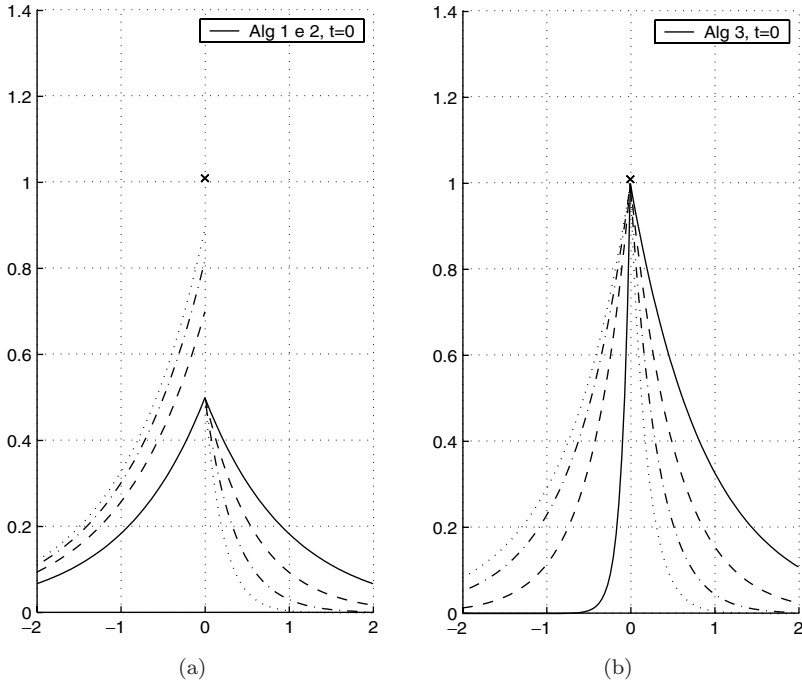


Fig. 2. Probability density functions for Algorithms 1 and 2 (part (a)) and for Algorithm 3 (part (b)) for different values of time t . The continuous lines refer to $t = 0$; the time goes on passing from broken lines to dotted ones.

convergence to the final state is mainly tuned by the initial value $\lambda_1(0)$ of the time-dependent parameter $\lambda_1(t)$ (which tends to 1, as time t increases), the vanishing velocity of the probability of energy-increasing steps is governed by the parameter k . When $\frac{1}{\lambda_1(t^*)} > m \frac{1}{\lambda_2(t^*)}$ the dynamics continues, for $t > t^*$, governed only by the parameter $\lambda_1 = \lambda_1(t^*)$ (close to 1) not depending on t . The dynamic evolution of the probability density functions for Algorithms 1 and 2 compared with Algorithm 3 is reported in Fig. 2.

Summarizing: the control parameters are λ for Algorithm 0, λ_1 and k for Algorithms 1 and 2, and $\lambda_1(0)$ and k for Algorithm 3. Varying them we study the efficiency of the algorithms by measuring the average time to reach a metastable configuration and the lowest energy value found for different system sizes.

4. Data Analysis

To compare these annealed algorithms with those carried out in previous works^{15–17} and in particular with Algorithm 0, we performed a set of trials for different values of N , starting from N initial conditions (for a system of size N) and averaging the data on $n_{\text{real}} = 1000$ disorder realizations. We measured two quantities to test the

performance of the algorithms:

- the average time (i.e. the number of spin flips) to reach a minimum energy level

$$\tau = \frac{1}{M} \sum_{i=1}^M t_i, \quad (4.1)$$

with $M = N \cdot n_{\text{real}}$ and $t_i, i = 1, \dots, M$ the time for each initial condition;

- the lowest energy found (averaged over disorder)

$$H_N = \left\langle \frac{\min_{\sigma} H_N(J, \sigma)}{N} \right\rangle_{n_{\text{real}}}, \quad (4.2)$$

where $\min_{\sigma} H_N(J, \sigma)$ is the minimum value of the energy of the metastable states attained starting from the set of the N initial conditions.

Our numerical experiments follow two different protocols:

- (i) with a fixed number of initial conditions;
- (ii) with a fixed elapsed computer time.

The results are described in the following subsections.

4.1. Fixed number of initial conditions

The dynamics of Algorithm 0 has been shown¹⁷ to behave as a smooth interpolation between greedy and reluctant dynamics¹⁵ depending on the parameter λ : small λ (say $\lambda \simeq 1$) plays the role of the greedy algorithm, while large λ (say $\lambda \simeq 100$) that of reluctant. In fact, the relaxation time $\tau(N)$ grows linearly with the system size when $\lambda \simeq 1$ and quadratically when $\lambda \simeq 100$ (see Table 1), as it was previously observed in Ref. 15 for deterministic greedy and reluctant regimes.

In Fig. 3, which refers to Algorithm 2, we represent τ as a function of N ($N \in [25, 300]$). We performed the analysis for different values of the control parameters. Due to space constraint, we show only the values $\lambda_1 = 1, 10, 100$ and three values of k ($k = 0.98, 0.99, 0.995$) for each λ_1 , together with the best numerical fits. Figure 3 shows the progressive increase of the slope in log-log scale from a sub-linear law in N for $\lambda_1 = 1$ and $k = 0.98$ (\diamond) to a super-linear one for $\lambda_1 = 100$ and $k = 0.98$ (\times). More in detail, the numerical fits of $\tau_{\lambda_1, k}(N) \sim N^a$ in Fig. 3 are reported in Table 1.

With the same protocol (fixed number of initial conditions), we measured the lowest energy H_N found by the algorithms. As a general remark we recall that from a theoretical point of view that the monotonicity in N of the ground state energy (this follows from sub-additivity¹⁸) is proved. For the largest size we have studied, some values of the simulation parameters give a non-monotone behavior in N , suggesting that we are not actually finding the true lowest energy state. A larger number of trials (i.e. initial conditions) would be needed to achieve the global minimum. However, our principal aim here is not to have a perfect measure of ground state energies. In Fig. 4 we represent, for Algorithm 2, H_N as a function of N for different values of λ_1 and k . The best results for large N are obtained for

Table 1. Numerical fits of $\tau_\lambda(N) \sim N^a$ for Algorithm 0 (with the symbols of Fig. 6) and of $\tau_{\lambda_1,k}(N) \sim N^a$ for Algorithm 1 and Algorithm 2 (with the symbols of Fig. 3).

λ	Alg. 0		Alg. 1			Alg. 2			
	a	Symbol	λ_1	k	a	λ_1	k	a	Symbol
1	1.027	*	1	0.98	0.687	1	0.98	0.549	\diamond
				0.99	0.630		0.99	0.475	+
				0.995	0.592		0.995	0.299	\square
10	1.263		10	0.98	1.041	10	0.98	1.030	\diamond
				0.99	0.948		0.99	0.891	+
				0.995	0.858		0.995	0.687	\square
100	1.932	\diamond	100	0.98	1.724	100	0.98	1.771	\times
				0.99	1.591		0.99	1.691	\triangle
				0.995	1.499		0.995	1.567	*

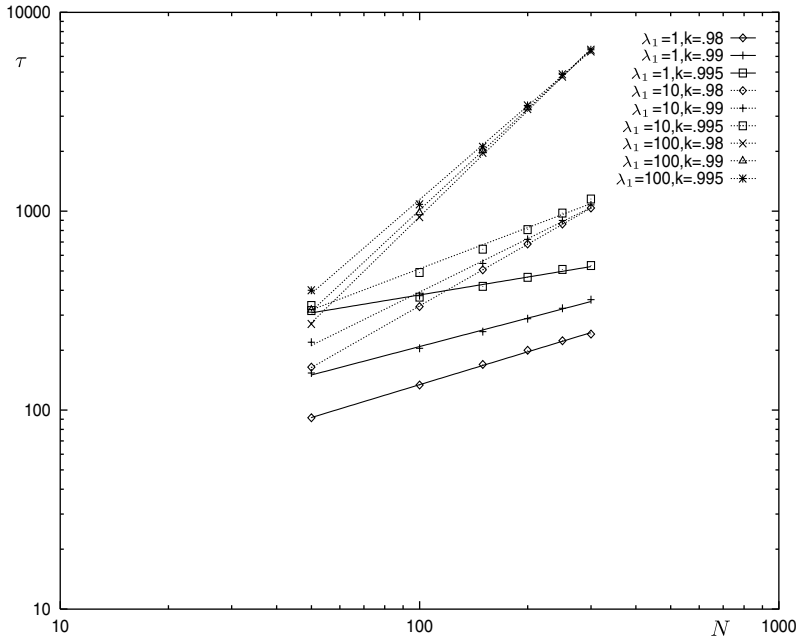


Fig. 3. Average time τ to reach a metastable configuration as a function of N for different values of λ_1 and k for Algorithm 2 and for a fixed number of initial spin configurations.

$\lambda_1 = 100$ and $k = 0.98$ which corresponds to annealed reluctant dynamics (as found for Algorithm 0, see Fig. 5). Therefore, this confirms^{16,17} that, for a fixed number of initial spin configurations, the algorithm that makes moves corresponding to the “smallest” possible energy change keeping the possibility of energy increase only for the first steps of the algorithm is the most efficient in reaching low-energy states. Note that, for $\lambda_1 = 1$ and $k = 0.995$ the attained energy values are sufficiently low: even if these results are not better than those for $\lambda_1 = 100$ (with $k = 0.98$

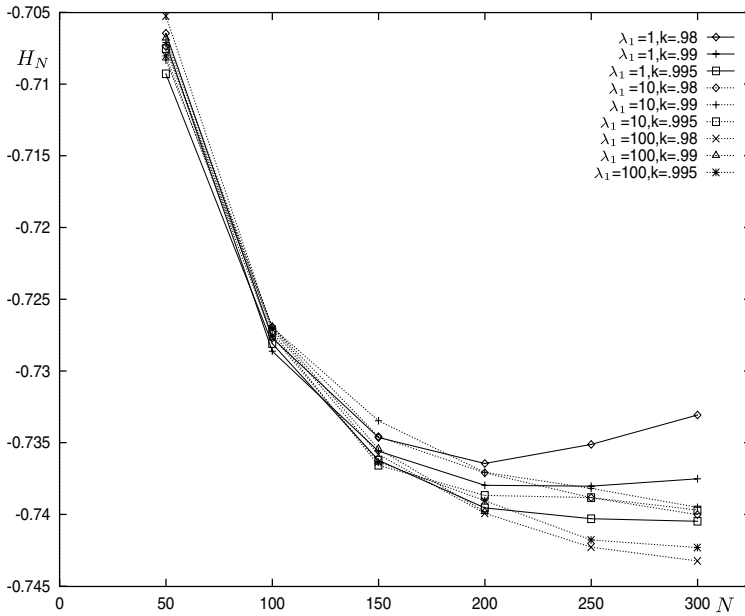


Fig. 4. Lowest energy value H_N as a function of N for different values of λ_1 and k for Algorithm 2 and for a fixed number of initial conditions.

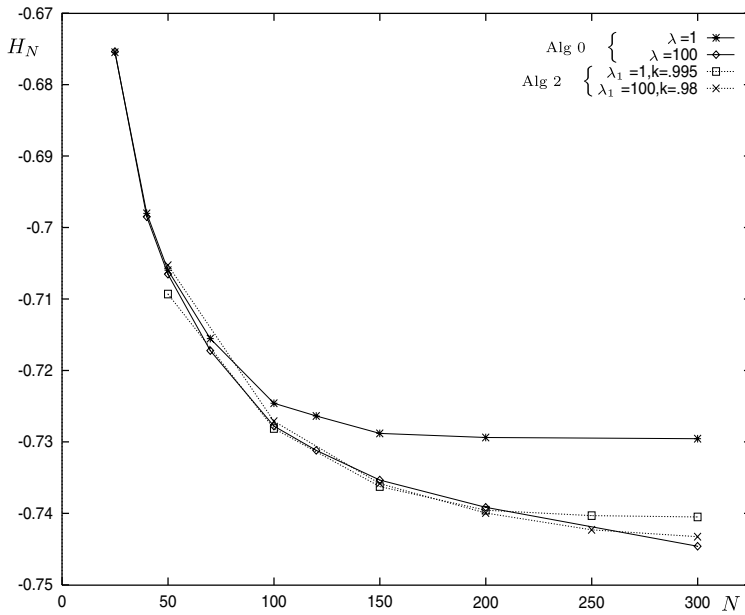


Fig. 5. Lowest energy value H_N as a function of N obtained using a protocol with a fixed number of initial conditions for $\lambda = 1$ (*) and $\lambda = 100$ (◇) for Algorithm 0 and for $\lambda_1 = 1$ and $k = 0.995$ (□) and for $\lambda_1 = 100$ and $k = 0.98$ (×) for Algorithm 2.

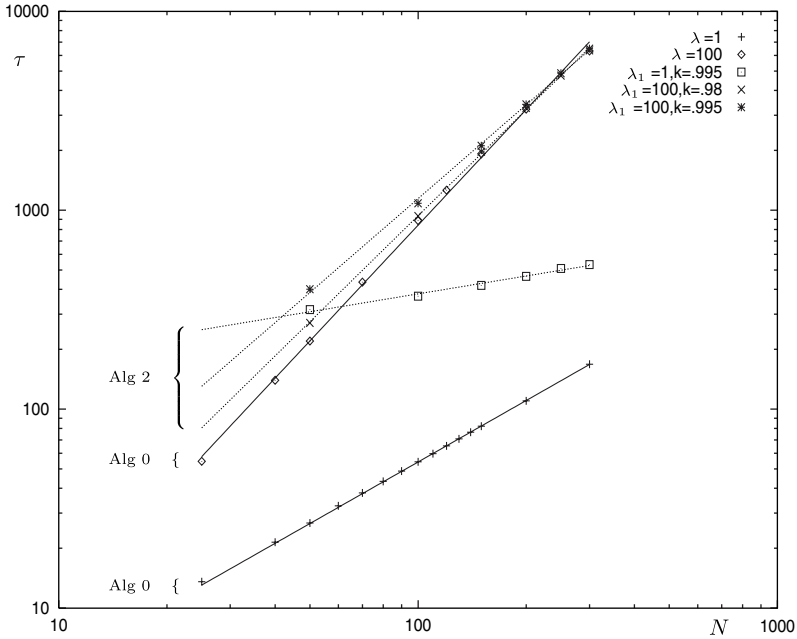


Fig. 6. Average time τ to reach a metastable configuration as a function of N for $\lambda = 1$ (+) and for $\lambda = 100$ (o) for Algorithm 0, and for $\lambda_1 = 1$ and $k = 0.995$ (□), for $\lambda_1 = 100$ and $k = 0.98$ (x), and for $\lambda_1 = 100$ and $k = 0.995$ (*) for Algorithm 2.

and $k = 0.995$), they should not be discarded since the average time scales better ($\tau_{1,0.995}^{(2)}(N) \sim N^{0.299}$ instead of $\tau_{100,0.98}^{(2)}(N) \sim N^{1.771}$ or $\tau_{100,0.995}^{(2)}(N) \sim N^{1.567}$).^a

Comparing these results with those obtained with the interpolating greedy and reluctant algorithm (Algorithm 0)¹⁷ we note (Figs. 5 and 6 and Table 1) that for small λ and λ_1 , Algorithm 2 is better performing than Algorithm 0 both with respect to average time and energy levels, while for greater λ and λ_1 we find comparable energy values but with lower cost for the computational time for Algorithm 2 ($\tau_{100,0.98}^{(2)}(N) \sim N^{1.771}$ instead of $\tau_{100}^{(0)}(N) \sim N^{1.932}$).

The same analysis is considered also for Algorithm 1. The comparison between Algorithms 1 and 2 shows that the possibility of exceed the energy barriers between minima is useful only for small values of λ_1 (for λ_1 close to 1 Algorithm 2 is more efficient than Algorithm 1 in reaching lower energy states) while for $\lambda_1 \geq 5$ the performances of Algorithms 1 and 2 are practically indistinguishable (see Figs. 7 and 8). Moreover, we note that the best scaling of the average time $\tau_{\lambda_1,k}$ with respect to N is obtained with Algorithm 2 (see Table 1), though for fixed N , λ_1 and k , we have $\tau_{\lambda_1,k}^{(1)} < \tau_{\lambda_1,k}^{(2)}$.

^aFrom now on, the superscript (x) in the notation of the average time $\tau^{(x)}$ will refer to the number of the corresponding algorithm.

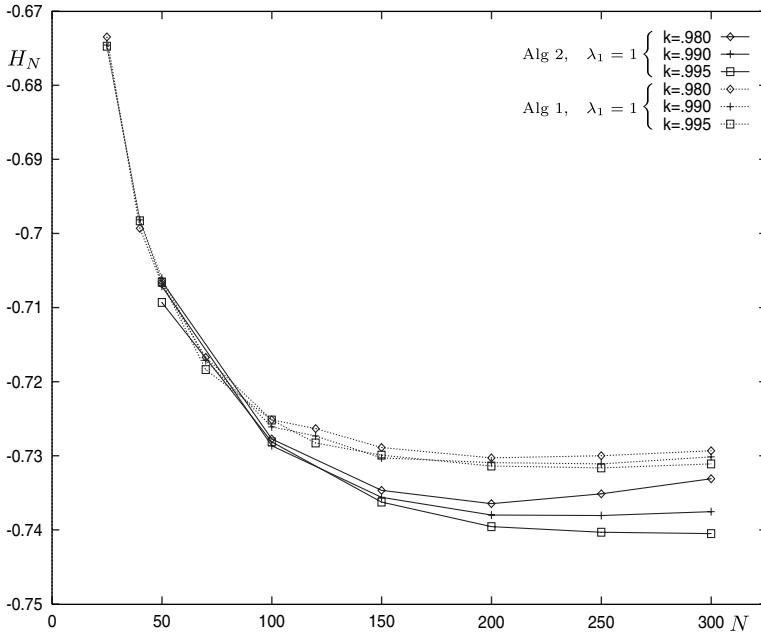


Fig. 7. Lowest energy value H_N as a function of N for $\lambda_1 = 1$ and different values of k , for Algorithms 1 and 2.

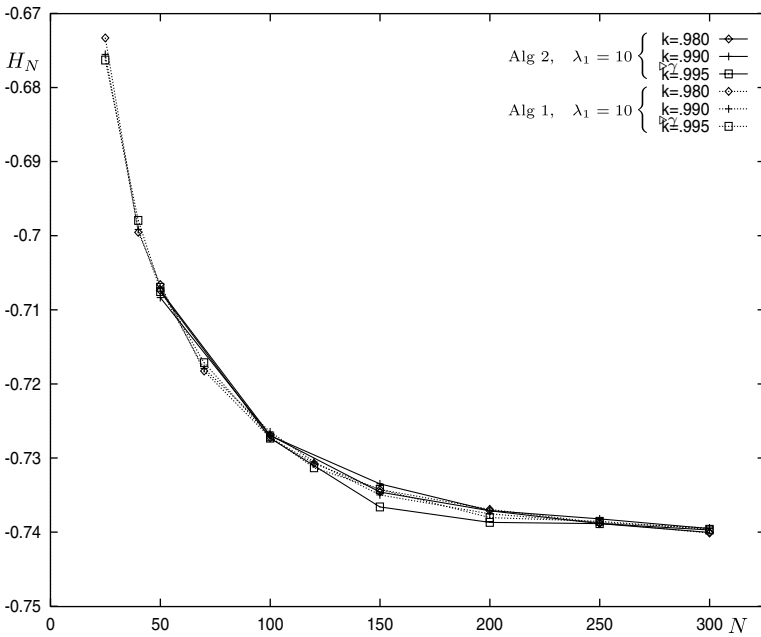


Fig. 8. Lowest energy value H_N as a function of N for $\lambda_1 = 10$ and for different values of k obtained with Algorithms 1 and 2.

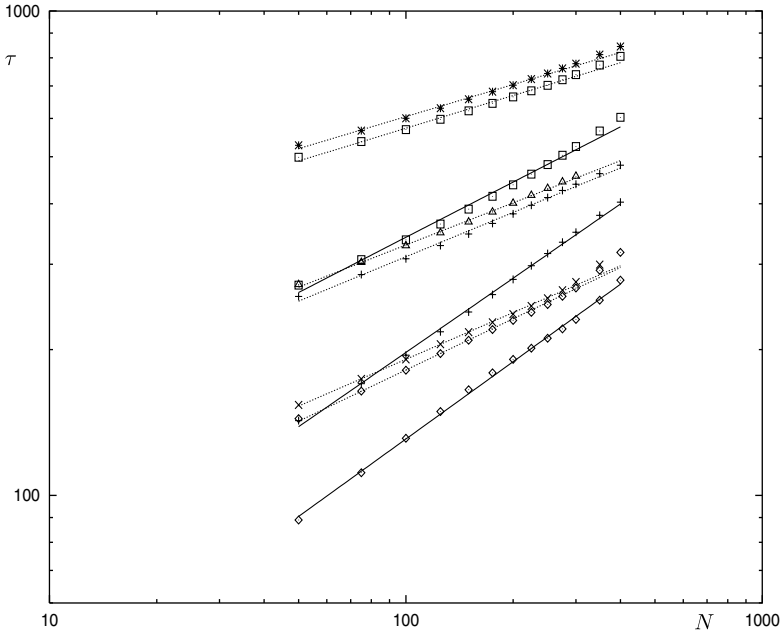


Fig. 9. Average time τ to reach a metastable configuration as a function of N for different values of $\lambda_1(0)$ and k for Algorithm 3, together with the best numerical fits for a fixed number of initial conditions. We represent $\lambda_1(0) = 2$ ($k = 0.98$ (\diamond), $k = 0.99$ ($+$) and $k = 0.995$ (\square)), $\lambda_1(0) = 10$ ($k = 0.98$ (\diamond), $k = 0.99$ ($+$) and $k = 0.995$ (\square)) and $\lambda_1(0) = 100$ ($k = 0.98$ (\times), $k = 0.99$ (\triangle) and $k = 0.995$ ($\cdot*$))

Figures 9 and 10 show the results of the analysis of Algorithm 3 with a fixed number of initial conditions: $N \in [25, 400]$ for three distinct values of $\lambda_1(0)$ ($\lambda_1(0) = 2, 10, 100$) and for four values of k ($k = 0.98, 0.99, 0.995, 0.997$) for each $\lambda_1(0)$. Because of high computational costs (which increase with $\lambda_1(0)$ and k), the cases $N = 350$ and $N = 400$ for $\lambda_1(0) = 100$ are only partially studied. For the same reason also the case $k = 0.997$ is considered only for $\lambda_1(0) = 2$.

Figure 10 shows that Algorithm 3 seems to depend weakly on the parameter $\lambda_1(0)$, its behavior being mainly ruled by k . In fact, the lines of the H_N values corresponding to the same choices of k are grouped into narrow bands well separated from one another. Moreover, a closer look to Fig. 10 shows that the best result for H_N is obtained for $\lambda_1(0) = 2$ and $k = 0.997$. Note that for any $\lambda_1(0)$, the closer the values of k to one, the lower the values of energy: slow growths of the parameter $\lambda_2(t)$ enable energy increases and then the possibility to exceed the energy barriers. Even though Algorithm 2 is slightly better performing ($\lambda_1 = 100, k = 0.98$ see Fig. 4) in terms of minimum energy level reached, the best scaling of $\tau_{\lambda_1(0),k}(N)$ is obtained by Algorithm 3. In fact, for Algorithm 3 we note (Fig. 9 and Table 2) the progressive increase of the slope in log-log scale from a scaling law $\tau_{\lambda_1(0),k}^{(3)}(N) \sim N^{0.22}$ for $\lambda_1(0) = 100$ and $k = 0.995$ ($\cdot*$) to $\tau_{\lambda_1(0),k}^{(3)}(N) \sim N^{0.53}$ for $\lambda_1(0) = 2$ and

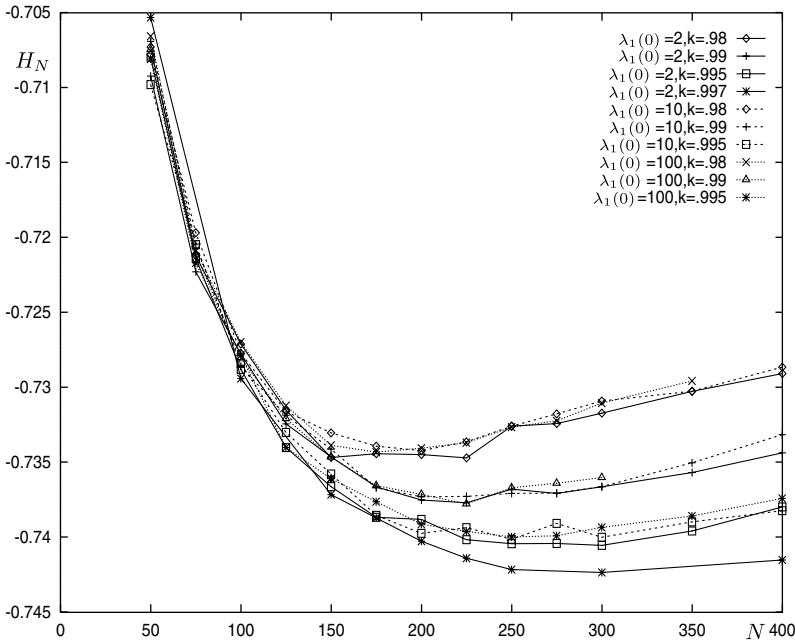


Fig. 10. Lowest energy value H_N as a function of N for different values of $\lambda_1(0)$ and k for Algorithm 3 and for a fixed number of initial conditions.

Table 2. Numerical fits of $\tau_{\lambda_1(0),k}(N) \sim N^a$ for Algorithm 3.

$\lambda_1(0)$	k	a	Symbol
2	0.98	0.531	\diamond
	0.99	0.509	+
	0.995	0.379	\square
	0.997	0.272	*
10	0.98	0.352	$\diamond \cdot$
	0.99	0.304	$\cdot + \cdot$
	0.995	0.225	$\cdot \square \cdot$
100	0.98	0.321	$\cdot \times \cdot$
	0.99	0.289	$\cdot \triangle \cdot$
	0.995	0.220	$\cdot * \cdot$

$k = 0.98$ (\diamond). More in detail, the numerical fits of $\tau_{\lambda_1(0),k}(N) \sim N^a$ for Algorithm 3 are reported in Table 2.

To conclude the analysis of the protocol with a fixed number of initial conditions we can say that taking into account also the average time τ , the best performing algorithm in reaching minimum energy level is Algorithm 3 (Fig. 11). In fact, Algorithm 3 with $\lambda_1(0) = 2$ e $k = 0.997$ attains minimum energy levels comparable with those obtained by the other algorithms with λ and λ_1 equal to 100 but with lower

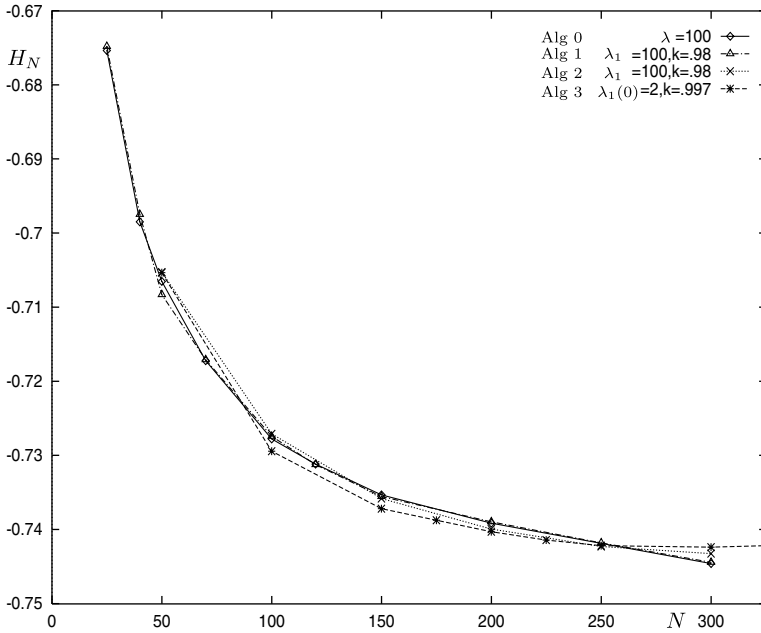


Fig. 11. Lowest energy value H_N as a function of N for $\lambda = 100$ (\diamond) for Algorithm 0, for $\lambda_1 = 100$ and $k = 0.98$ (\triangle) for Algorithm 1 and (\times) for Algorithm 2 and for $\lambda_1(0) = 2$ and $k = 0.997$ ($*$) for Algorithm 3.

computational costs ($\tau_{2,0.997}^{(3)} \sim N^{.272}$ while $\tau_{100}^{(0)} \sim N^{1.932}$, $\tau_{100,0.98}^{(1)} \sim N^{1.724}$ and $\tau_{100,0.98}^{(2)} \sim N^{1.771}$, see Tables 1 and 2).

4.2. Fixed elapsed computer time

Finally, we analyze the lowest energy states found by the dynamics varying the control parameters for a given elapsed running time for all algorithms. In Fig. 12 we consider the minimum energy values H_N , obtained by choosing different system sizes N and, for each of them, different parameter values ($\lambda = 1, 10, 100$ for Algorithm 0, $\lambda_1 = 1, 5, 10, 100$ for Algorithm 1, $\lambda_1 = 1, 10$ for Algorithm 2 and $\lambda_1(0) = 2, 10, 100$ for Algorithm 3) with different annealing scheduling each ($k = 0.98$ and $k = 0.995$ for Algorithms 1 and 2, $k = 0.995$ and $k = 0.997$ for Algorithm 3), for a fixed time of 50 h of CPU on a IBM SP4. For Algorithm 2 we consider in detail mainly the case ($\lambda_1 = 1$) in which the dynamics behaves differently from that generated by Algorithm 1. Each run (i.e. for fixed N and for fixed control parameter) consists of 1000 disorder realizations, with the same CPU time length (3 min.) assigned to each sample, in order to compare these results with Refs. 15–17. With all this dynamics, for $N \leq 150$, we can find the ground state of the system, since varying the control parameters and independently on the algorithm used, the values of H_N coincide, within our numerical accuracy (10^{-10}). The best result is obtained with

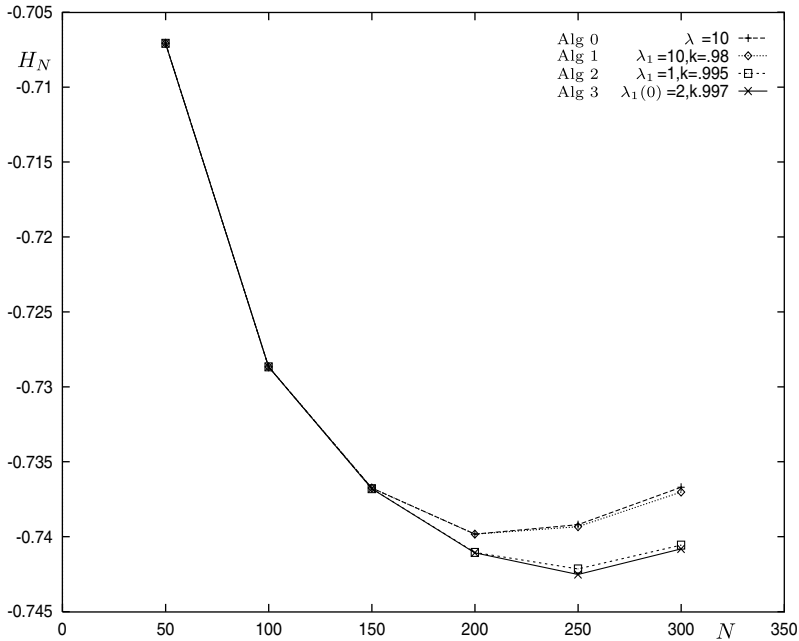


Fig. 12. Lowest energy value H_N as a function of N for different values of control parameters for Algorithms 0, 1, 2 and 3, for a fixed CPU time of 50 h on a IBM SP4. The symbol (+) refers to $\lambda = 10$ for Algorithm 0, (\diamond) to $\lambda_1 = 10$ and $k = 0.98$ for Algorithm 1, (\square) to $\lambda_1 = 1$ and $k = 0.995$ for Algorithm 2 and (\times) for $\lambda_1(0) = 2$ and $k = 0.997$ for Algorithm 3.

Algorithm 3 for the case $\lambda_1(0) = 2$ and $k = 0.997$ (even though the result provided by Algorithm 2 for $\lambda_1 = 1$ and $k = 0.995$ is comparable). Note that, for Algorithm 1 the best result is for $\lambda_1 = 10$ and $k = 0.98$ in good agreement with the best result of Algorithm 0 obtained for $\lambda = 10$ (Fig. 12). Moreover, it is worthnoting that the values H_N obtained with Algorithm 3 for the case $\lambda_1(0) = 2$ and $k = 0.997$ are the best (for fixed CPU time) with respect to all algorithms we consider in the present paper and in Refs. 15–17.

Acknowledgments

We thank Prof. S. Graffi and Prof. I. Galligani for their encouragement. The Cineca staff and in particular Dr. G. Erbacci and Dr. C. Calonaci are acknowledged for the technical support. The computation resources were provided by Cineca (High Performance Computing Grant) and by CICAIA (Università di Modena e Reggio Emilia).

References

1. D. Sherrington and S. Kirkpatrick, Solvable model of a spin-glass, *Phys. Rev. Lett.* **35** (1975) 1792–1796.

2. M. Mézard, G. Parisi and M. A. Virasoro, *Spin Glass Theory and Beyond* (World Scientific, 1987).
3. J.-P. Bouchaud and M. Potters, *Theory of Financial Risk* (Alea-Saclay, 1997).
4. H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing* (Oxford Univ. Press, 2001).
5. F. T. Bantilan and R. G. Palmer, Magnetic properties of a model spin glass and the failure of linear response theory, *J. Phys.* **F11** (1981) 261–266.
6. S. Cabasino, E. Marinari, P. Paolucci and G. Parisi, Eigenstates and limit cycles in the SK model, *J. Phys. A: Math. Gen.* **21** (1988) 4201–4210.
7. S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, Optimization by simulated annealing, *Science* **220** (1983) 671–680.
8. G. S. Grest, C. M. Soukoulis and K. Levin, Cooling-rate dependence for the spin-glass ground-state energy: Implications for optimization by simulated annealing, *Phys. Rev. Lett.* **56** (1986) 1148–1151.
9. J.-P. Bouchaud, F. Krzakala and O. C. Martin, Energy exponents and corrections to scaling in Ising spin glasses, *Phys. Rev.* **B68** (2003) 224404.
10. M. Palassini, Ground-state energy fluctuations in the Sherrington–Kirkpatrick model, cond-mat/0307713.
11. S. Kobe, Ground-state energy and frustration of the Sherrington–Kirkpatrick model and related models, cond-mat/0311657.
12. S. Boettcher and A. G. Percus, Optimization with extremal dynamics, *Phys. Rev. Lett.* **86** (2001) 5211–5214.
13. S. Boettcher and P. Sibani Comparing extremal and thermal explorations of energy landscapes, cond-mat/0406543.
14. S. Boettcher, Extremal Optimization for the Sherrington–Kirkpatrick spin glass, cond-mat/0407130.
15. L. Bussolari, P. Contucci, M. Degli Esposti and C. Giardinà, Energy-decreasing dynamics in mean-field spin models, *J. Phys. A: Math. Gen.* **36** (2003) 2413–2421.
16. L. Bussolari, P. Contucci, C. Giardinà, C. Giberti, F. Unguendoli and C. Vernia, Optimization strategies in complex systems, Science and Supercomputing at Cineca — 2003 Report, 386–390, <http://arxiv.org/abs/math.NA/0309058>.
17. P. Contucci, C. Giardinà, C. Giberti, F. Unguendoli and C. Vernia, Interpolating greedy and reluctant algorithms, to appear on *Optimization Methods and Software* (2004), <http://arxiv.org/abs/math-ph/0309063>.
18. F. Guerra and F. Toninelli, The thermodynamical limit in mean field spin glass model, *Comm. Math. Phys.* **230** (2002) 71–79.