Eur. Phys. J. Special Topics **226**, 639–649 (2017)

DOI: 10.1140/epjst/e2016-60333-2

THE EUROPEAN PHYSICAL JOURNAL SPECIAL TOPICS

Regular Article

Structure optimisation by thermal cycling for the hydrophobic-polar lattice model of protein folding

Florian Günther[1,2,a], Arnulf Möbius[3,b], and Michael Schreiber[4,c]

[1] Helmholtz-Zentrum Dresden-Rossendorf, Institut für Ionenstrahlphysik und Material forschung, Center for Advancing Electronics Dresden (cfaed), 01328 Dresden, Germany
[2] Technische Universität Dresden, Institut für Physikalische Chemie und Elektrochemie, 01062 Dresden, Germany
[3] Leibniz-Institut für Festkörper- und Werkstoffforschung Dresden (IFW), Institut für Theoretische Festkörperphysik, 01069 Dresden, Germany
[4] Technische Universität Chemnitz, Institut für Physik, 09107 Chemnitz, Germany

Received 17 October 2016 / Received in final form 19 November 2016
Published online 5 April 2017

Abstract. The function of a protein depends strongly on its spatial structure. Therefore the transition from an unfolded stage to the functional fold is one of the most important problems in computational molecular biology. Since the corresponding free energy landscapes exhibit huge numbers of local minima, the search for the lowest-energy configurations is very demanding. Because of that, efficient heuristic algorithms are of high value. In the present work, we investigate whether and how the thermal cycling (TC) approach can be applied to the hydrophobic-polar (HP) lattice model of protein folding. Evaluating the efficiency of TC for a set of two- and three-dimensional examples, we compare the performance of this strategy with that of multi-start local search (MSLS) procedures and that of simulated annealing (SA). For this aim, we incorporated several simple but rather efficient modifications into the standard procedures: in particular, a strong improvement was achieved by also allowing energy conserving state modifications. Furthermore, the consideration of ensembles instead of single samples was found to greatly improve the efficiency of TC. In the framework of different benchmarks, for all considered HP sequences, we found TC to be far superior to SA, and to be faster than Wang-Landau sampling.

1 Introduction

Proteins are basic to all life forms on earth since they are involved in quite diverse biological processes [1]. They are chain-like macromolecules consisting of amino acids,

[a] e-mail: f.guenther@hzdr.de
[b] e-mail: a.moebius@ifw-dresden.de
[c] e-mail: schreiber@physik.tu-chemnitz.de

where the specific sequence of the amino acids determines all protein properties. For correct functioning, a protein has to acquire its specific three-dimensional (3D) structure. This structure is referred to as the native, functional, or biological fold. Misfolded proteins do not function or even function in the wrong way, which can cause serious diseases as Alzheimer, Parkinson [2], bovine spongiform encephalopathy (BSE, also known as "mad cow disease"), and Creutzfeldt-Jakob disease [3].

Therefore, the problem of protein folding, that is the prediction of the 3D structure of a protein based on the knowledge of its amino acid sequence, is of long standing interest to biologists, chemists, and physicists. The folding is governed by attraction or repulsion of single atoms or groups of atoms according to their chemical properties. Among all these non-covalent intra-molecular interactions [4], the hydrophobic effect has the strongest influence [5,6].

The hydrophobic interaction arises from a collective phenomenon, the effective interaction of non-polar molecules in an environment consisting of polar components like H_2O molecules. Driven by an entropic effect, the water molecules at the boundary of the polar solvent and the non-polar phase form extra fluctuating hydrogen bonds with their nearest neighbours [7,8]. In protein folding, this effect causes the formation of a core consisting of hydrophobic amino acids which is surrounded by polar amino acids.

The computation of the functional fold is very difficult not only because of poorly understood contributions to the free energy [4], but also because of the huge and complex space of possible configurations. The first of these two problems requires the investigation of detailed models and an adjustment by comparing to experimental results [9]. The second problem can be approached considering simplified models such as the hydrophobic-polar (HP) model [10] by means of sophisticated optimisation algorithms.

In this work, we investigate the performances of three heuristic optimisation algorithms when applied to the HP model: we focus on the thermal cycling (TC) algorithm [11] and compare its performance to that of multi-start local search (MSLS) procedures and to that of the well know simulated annealing (SA) algorithm [12]. For all these methods, we study here how the performance depends on the respective algorithm parameters; related more detailed, although preliminary, information is given in [13].

2 The hydrophobic-polar model of protein folding

The HP model is the simplest model used in protein folding simulations. Nevertheless, it is one of the most frequently studied ones. This model was introduced by Lau and Dill in order to explore the energy landscape of model proteins in both conformational space and sequence space [10]. Although it is very simple, the HP model exhibits the important features of real protein folding: a huge configuration space with a very large number of local minima, funnels in the energy landscape, and a dependence on the sequence of amino acids [9]. The first aspect was analysed in some detail by the group of Wolfhard Janke; they established an algorithm for the exact enumeration of HP chains, see reference [14].

The HP model belongs to the backbone-only models and is based on three rough simplifications. First, the amino acids are grouped into only two types of nodes, hydrophobic (H) and polar (P) ones [10,15]. Second, the chain is placed on a regular lattice where the bond lengths equal the lattice constant and where each lattice site can be occupied by at most one node. Third, the energy of a conformation is assumed to be given by the negative number of neighbouring pairs of unconnected H nodes. With this, the formation of a hydrophobic core is reflected in a very simple way.

Table 1. Benchmark sequences [16] considered in this work. The names comprise dimension and length, L, of the chains. The E_0 values are the lowest energies reported [16]. Here we assume them to be the ground state energies although exact proofs are partly still missing.

Name	E_0	Sequence
2D64	−42	$H_{12}PHPHP_2H_2P_2H_2P_2HP_2H_2P_2H_2P_2HP_2H_2P_2H_2P_2$ $HPHPH_{12}$
2D85	−53	$H_4P_4H_{12}P_6H_{12}P_3H_{12}P_3H_{12}P_3HP_2H_2P_2H_2P_2HPH$
2D100a	−48	$P_6HPH_2P_5H_3PH_4PH_2P_4H_2P_2H_2PH_5PH_{10}PH_2PH_7$ $P_{11}H_7P_2HPH_3P_6HPH_2$
2D100b	−50	$P_3H_2P_2H_4P_2H_3PH_2PH_2PH_4P_8H_6P_2H_6P_9HPH_2PH_{11}$ $P_2H_3PH_2PHP_2HPH_3P_6H_3$
3D48	−32	$HPH_2P_2H_4PH_3P_2H_2P_2HPH_3PHPH_2P_2H_2P_3HP_8H_2$
3D58	−44	$PHPH_3PH_3P_2H_2PHPH_2PH_3PHPHPH_2P_2H_3P_2HPH$ $P_4HP_2HP_2H_2P_2HP_2H$
3D64	−56	$PH_2PH_2PH_3P_2HPHP_2HPHP_2H_3PH_2PH_2P_2H_2PH_2P$ $H_3P_2HPHP_2HPHP_2H_3PH_2PH_2P$

The HP model has been extended in several directions. Beside square and simple cubic lattices, other common lattices structures were considered, for instance the triangular and face centred cubic lattices [17,18]. Furthermore, side chains were taken into account [19]. In the present study, we focus on the square and simple cubic lattices and consider the single stranded sequences given in Table 1.

For all the heuristic optimisation methods considered in our work, finding an appropriate set of small but to some extent also complex state modifications is basic. Here, we use the pull move set suggested by Lesh et al. [20], which was designed for self-avoiding chains on regular lattices. Such state modifications consist in choosing a node, shifting it to a lattice site being a next-nearest neighbour of its current position, and, if necessary, pulling nodes either out of the previous or out of the subsequent parts of the chain to neighbouring sites of the last node moved until a valid sequence of node positions, see above, is obtained again. However, not all pull moves are feasible: they may be forbidden by a target site being already occupied or by the impossibility to relax the chain considering either only previous or only subsequent nodes.

The set of pull moves provides a good balance between local and global configurational changes [16]. Moreover, each such state modification is reversible, and their set fulfils the ergodicity demand [20]. Finally, pull moves can be implemented in such a way that the effort to calculate the total energy change by any such state modification scales with the number of nodes moved rather than with the chain length L.

3 Numerical methods

3.1 Local search by iterative improvements

For discrete optimisation tasks, such as the search for the ground state of an HP sequence, the answer to the question whether or not a given state is a local minimum depends on the set of considered state modifications. This so-called move class defines which states are neighbours to each other.

Although the idea of corresponding local minimisation procedures is very simple, they can be rather efficient, especially in the case of a small number of local minima. In this context, we point out that the number of local minima is usually lowered when moves of higher complexity are additionally taken into account.

The choice of random initial configurations may be non-trivial. Treating combinatorial problems such as the HP model requires first to select a valid configuration.

Then, the move class considered in the local search can be used to randomise this configuration in an iterative process.

Based on any local search procedure, a simple composed heuristic optimisation algorithm can be easily constructed: repeatedly, a random initial configuration is created and afterwards quenched by this local search. The best of the quenched states is considered as final result. This composed algorithm is referred to as multi-start local search (MSLS) in the following.

3.2 Simulated annealing

SA is one of the most famous heuristic algorithms since it is easy to implement and exhibits reasonably good computing performance [12]. In such a calculation, the temporal development of a system is simulated, often by means of a Metropolis algorithm [21]. In doing so, the energetic change is controlled via a parameter Θ, which can been interpreted as the simulation temperature.

When this temperature is slowly reduced, the system is driven towards a minimum of the energy landscape. According to the exact proof in reference [22], for exponentially slow cooling, the energy converges to the ground state energy. In practice, however, the computing time is limited, so that the simulations are mostly trapped in merely local minima. Typically, the energies reached are the lower the slower the cooling.

A simple approach to improve SA, as well as other heuristic optimisation algorithms, is the best-of-N procedure. Its idea consists in distributing the computing effort to several independent runs with lower accuracy and considering the best of the individual results as final one. This procedure is the most straightforward way to treat an optimisation task in parallel. Furthermore, it may offer some performance benefits [23].

3.3 Thermal cycling

The TC procedure combines the features of the algorithms discussed above. In it, a multiply repeated cyclic process is substituted for the slow cooling down in SA [11].

First, starting from the best configuration obtained so far, the system is disturbed whereby its energy increases. This step is referred to as heating, but this notion has to be understood in a qualified sense: it is basic to TC that the distortion is limited to only a small part of the degrees of freedom. Thus, most of the knowledge gained in the previous optimisation cycles is retained. One can understand this step as a short heat pulse, where length and height together determine the amplitude of distortion.

Second, the system is quenched by means of a local search procedure, see Section 3.1. In principle, it is a big advantage of TC that, in case of combinatorial optimisation, branch-and-bound strategies can be used to reach stability concerning certain classes of complex moves which are inappropriate for SA.

At the third and last step, the quenched state is compared to the initial one and the best of both of them is selected for the ongoing optimisation. Concerning this point, TC differs from so-called basin hopping methods, in which transitions to higher local minima can be performed also [24]. In basin hopping, in analogy to SA, the acceptance rate of those transitions is controlled by an appropriate schedule.

These three steps are cyclically repeated many times while the amplitude of the distortion decreases slowly. On average over the entire optimisation process, substitutions of quenched for initial states occur seldom. Therefore, it is tempting to distribute the performing of complete cycles, or even of groups of complete cycles, to different

Table 2. Comparison of parameter sets characterising the distributions of final energies obtained in 10^5 local search runs without ($m = 0$) and with ($m = 5$) performing energy conserving moves found; for the definition of m see text. Here, values of the average, $\langle E \rangle$, the standard deviation, σ, the median, E_{mdn}, and the lowest energy found, E_{b}, are presented together with the ground state energies, E_0, from Table 1. Additionally, averages of best values out of 1000 minimisations, $\langle E_{1000} \rangle$, are given. The random initial states were constructed by means of the TM method with $k_{\mathrm{tm}} = 10\,L$.

	$m = 0$				$m = 5$					
sequence	$\langle E \rangle$	σ	E_{mdn}	E_b	$\langle E \rangle$	σ	E_{mdn}	E_{b}	$\langle E_{1000} \rangle$	E_0
2D64	−18.8	3.8	−19	−35	−26.6	2.8	−27	−37	−35.4	−42
2D85	−26.5	4.8	−27	−46	−37.2	3.5	−37	−52	−48.6	−53
2D100a	−21.3	4.5	−22	−36	−33.1	3.0	−33	−44	−41.9	−48
2D100b	−21.7	4.5	−22	−38	−34.1	3.0	−34	−47	−42.4	−50
3D48	−18.0	3.3	−18	−30	−22.5	2.9	−22	−32	−29.7	−32
3D58	−22.6	3.3	−23	−37	−27.6	3.1	−28	−41	−37.4	−44
3D64	−25.8	4.0	−26	−45	−30.6	3.5	−30	−46	−42.9	−56

CPUs. In this, the substitution rule may be slightly modified. Thus TC should be well suited for parallelisation; but this aspect is behind the scope of the present work.

Applying the TC algorithm to the travelling salesman problem, it was shown that the performance can be strongly improved by generalisation to the consideration of an ensemble of n_{ens} states [11,25]. In doing so, the third step is modified in the following way: if and only if the quenched state has a lower energy than the initial state of the cycle, it is substituted for the ensemble state with the highest energy. In this way, the diversity of the ensemble is maintained as far as possible [26].

4 Results and discussion

4.1 Local search procedures

In our study of the properties of local search codes, we implemented two specific features which turned out to be very useful: (i) while sweeping through the whole move class, we perform not only the energy reducing moves, $\Delta E < 0$, but, under the condition that an energy reduction happened within the last m sweeps, also the energy conserving moves, $\Delta E = 0$. (ii) To reduce the computing effort, we skip trials of those moves which were found to be not feasible in a previous trial and which have not been released again since then in consequence of another move. For this aim, we establish a dynamically ordered list in which these forbidden moves are grouped on the bottom. Through the remaining moves, we run in a pseudo-random order.

Beside the local search itself, the construction of the starting state has substantial impact on the final result. To study this effect, we considered the following three options: (i) performing a self-avoiding random walk which ends when the length of the considered sequence is reached; if the walk terminates in a dead end before, it is restarted. (ii) Starting from the linear chain, we apply a number of modifications out of the whole pull move class. (iii) Also starting from a linear chain, we consider only those moves pulling at one end of the chain and perform k_{tm} of such tail moves (TM). In our experiments, the TM method was found to offer the best balance of final minimal energies and required computing effort. We observed that $k_{\mathrm{tm}} = 10\,L$ results in appropriately randomised starting configurations; for details, see [13].

To evaluate the effectiveness of our local search algorithm, we applied it 10^5 times to each of the sequences given in Table 1. For all these tasks, we obtained almost Gaussian-like distributions of the final energies. Our Table 2 presents the

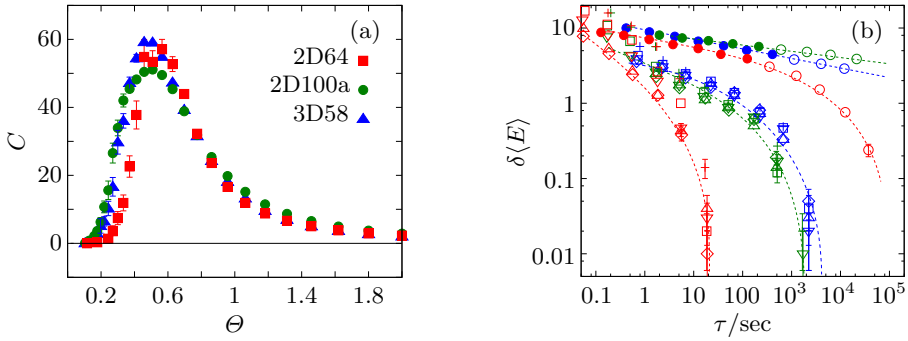

Fig. 1. (a) Specific heat $C(\Theta)$ versus the temperature Θ for 2D64, 2D100a, and 3D58. (b) Logarithmic performance plot, presenting the mean deviations of the final energy from the ground state energy, $\delta\langle E\rangle$, versus computing time, τ, for SA applied to 2D64 (red), 2D100a (green), and 3D58 (blue) using the best-of-N approach: $N=1$ ($\diamond$), $N=3$ ($\triangle$), $N=10$ ($\triangledown$), $N=30$ ($\square$), and $N=100$ ($+$). For comparison, performance curves for MSLS ($\bullet$, $\circ$) are given; full symbols mark data obtained from overall 10^5 local searches starting from random configurations, as the values in Table 2, empty symbols refer to additional MSLS runs with larger numbers of such trials. In all our performance plots, the dashed lines serve as guide to the eye, and averaging is performed over the results of 100 independent runs.

corresponding characteristic parameter values. Moreover, in its column $\langle E_{1000}\rangle$, it contains mean values of the best final energies out of 1000 minimisations. The data in this column provide a first impression of what can be reached by means of MSLS.

Table 2 shows that, for none but one of the investigated sequences, any of the minimisations could find the ground state energy. Only for the 3D48 sequence, the ground state energy was obtained in a few runs.

This finding testifies how challenging these minimisation tasks are. Simultaneously, however, Table 2 demonstrates that performing pull moves with $\Delta E = 0$ leads to a surprisingly strong improvement of the search quality. This effect, however, saturates at about $m = 5$. We remark that the computing effort which is required to perform the additional sweeps through the move class is rather low: due to the use of the list of forbidden moves, the computing time increases only by up to 15%.

4.2 Simulated annealing

In our SA studies, we used a pseudo-exponential cooling schedule: the simulation starts at an initial temperature, $\Theta = \Theta_\mathrm{i}$. For each value of Θ, n_Ms Metropolis steps are performed; after that, Θ is diminished by a factor of 0.9. The simulation is terminated as soon as Θ has fallen below the final temperature, Θ_f. Then, a local search step as described above is applied to the configuration with the lowest energy found in the Metropolis part; this local search yields the final result of our SA run.

For choosing appropriate values of Θ_i and Θ_f, we make use of the temperature dependence of the specific heat, $C(\Theta)$, shown in Figure 1a. For the here considered cases, $C(\Theta)$ is only weakly dependent on dimension, length, and HP sequence. It has a broad peak in the region of $0.2 < \Theta < 0.8$, in agreement with reference [16]; in this temperature interval, folded structures are formed [16]. Therefore, we chose $\Theta_\mathrm{i} = 1$ and $\Theta_\mathrm{f} = 0.1$ in all our simulations.

In Figure 1b, the deviation, $\delta\langle E\rangle = \langle E\rangle - E_0$, of the mean value of the final energies of SA runs, $\langle E\rangle$, from the ground state energy is plotted versus the computing time τ in a double logarithmic presentation for three HP sequences. We compare here

the results of individual SA runs with those of an SA based best-of-N procedure. The latter calculations were performed sequentially, still without using parallelisation tools; the τ values are total times of individual optimisation runs. All performance data points presented in this and the other diagrams of our study were obtained by means of 100 independent runs on 2.6 GHz AMD Opteron(tm) processors 6238.

The main message of Figure 1b is that the best-of-N approach applied to SA works very nicely: for the two more complicated tasks, 2D100a and 3D58, if $\tau > 3$ sec, the performances are roughly the same from $N = 1$ up to $N = 100$; for the simplest task, 2D64, this holds up to $N = 10$. Thus, distributing such SA based optimisations to many CPUs should be easily possible without significant loss of efficiency.

Furthermore, one feature of the SA performance curves in Figure 1b is particularly noteworthy: there is always a threshold τ_t separating a low-τ region of slow power law decrease of $\delta\langle E\rangle(\tau)$ with increasing τ from a high-τ region in which $\delta\langle E\rangle(\tau)$ very rapidly drops. Already, if τ is only moderately larger than τ_t, almost all SA runs yield the ground state energy. The transition between the two regimes seems to occur when $\delta\langle E\rangle(\tau) \sim 1$. Thus this effect likely arises from only very few, in the final stage even only two, energy levels remaining relevant. This feature seems to occur also in global minimum searches by means of TC, see Subsection 4.3.

For comparison, Figure 1b includes performance data for MSLS applied to the same sequences. It shows that, not surprisingly, this approach is by several orders of magnitude slower than SA. Simultaneously, for 2D64, it demonstrates that, when the number of trials rises to values far above 10^5, the MSLS result, too, tends to the ground state energy. Finally, we emphasise that, similarly as for SA, also the MSLS performance plot for 2D64 rapidly bends down above a certain τ threshold.

Further SA results and a comparison to TC are given in Subsection 4.4.

4.3 Thermal cycling

As in the case of SA, a few parameters have to be set before the TC simulation can start. In the heating steps, we modify the initial state of the cycle by a given number of randomly chosen pull moves, n_h, where, in contrast to [11], all proposed moves are performed. The simulation starts with $n_h = L/2$. After n_{cyc} cycles, n_h is diminished by substituting the integer part of 0.9 n_h for it. The simulation proceeds as long as $n_h > 0$.

In Subsection 4.1, we found that executing also energy conserving moves improves the performance of the local search starting from random states. Thus, we investigated the impact of this idea when applied within the quenching step of TC. Again, we observed that this modification of the iterative improvement leads to considerably better results, where $m = 5$ seems to be an appropriate choice again.

In our first attempts to utilise TC, we struggled with missing convergence for 2D64. Figure 2a demonstrates this failure by presenting the distributions (red) of final energies which we obtained with $n_{cyc} = 500$ and $n_{cyc} = 1000$. The origin of the failure becomes clear when the shapes of the configurations are compared. While the ground state configurations with $E_0 = -42$ have the shape of a C, the metastable configurations with $E = -36$ resemble an S. Therefore, the finding that the height of the peak at $E = -36$ does not decrease with increasing n_{cyc} while the low-energy part of the distribution changes considerably can be understood as follows: the number of pull moves which is required to reach and overcome the barrier between the regions of S and C structures in the energy landscape seems to exceed the maximum number of modifications performed in the heating step.

To approach this problem, we modified the starting stage of TC. We now use MSLS to obtain an appropriate initial configuration of the HP sequence instead of

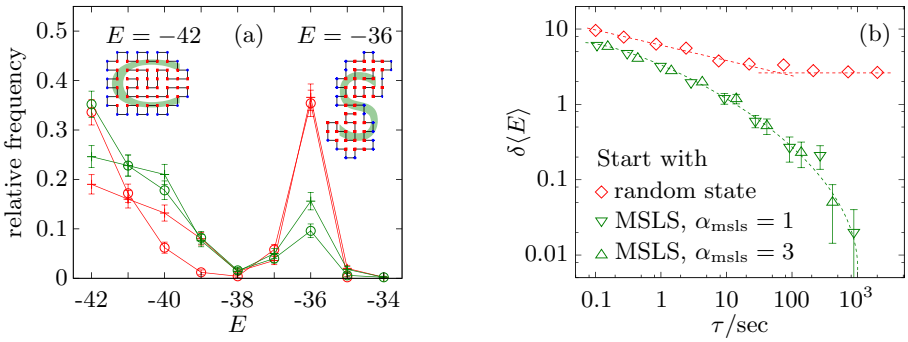

Fig. 2. Comparison of the TC approaches applied to the 2D64 sequence utilising the standard initialisation (red) and the construction of the initial configurations by means of MSLS (green). (a) Distribution of the final energies obtained from 500 independent initialisations with $n_{\mathrm{cyc}} = 500$ (+) and $n_{\mathrm{cyc}} = 1000$ (∘). (b) Corresponding performance plot.

quenching only one random configuration. To do so, we first perform $n_{\mathrm{msls}} = \alpha_{\mathrm{msls}} n_{\mathrm{cyc}}$ local searches starting from different random configurations. The corresponding final energy distributions (green) in Figure 2a demonstrate the success of this idea: when utilising it, the risk to end up in an S shaped configuration decreases with increasing n_{cyc}. Furthermore, the low-energy part of the distribution is amplified by utilising MSLS as initialisation of TC. These features are related to the improved efficiency obvious from the performance curves in Figure 2b. In the following, we set $\alpha_{\mathrm{msls}} = 1$.

We showed above that performing also energy conserving modifications strongly improves the effectiveness of the local search. The same idea can also be incorporated into the selection step of TC; in the original version of TC, the quenched state with energy E_{q} is only substituted for the initial state with energy E_{i} if $E_{\mathrm{q}} < E_{\mathrm{i}}$. Figure 3a shows that performing the substitution whenever $E_{\mathrm{q}} \leq E_{\mathrm{i}}$ leads to a remarkable improvement of the performance.

So far, we have focused only on TC versions employing a single sample. Now we turn to the ensemble approach and consider n_{ens} states simultaneously; in doing so, we perform $n_{\mathrm{cyc}} n_{\mathrm{ens}}$ cycles for each value of the heating amplitude. For this aim, the substitution rule has to be extended. Three cases have to be treated separately: first, if $E_{\mathrm{q}} > E_{\mathrm{i}}$, no substitution is done. Second, if $E_{\mathrm{q}} < E_{\mathrm{i}}$, the new state is substituted for the worst state in the ensemble. Third, if $E_{\mathrm{q}} = E_{\mathrm{i}}$, the quenched state is substituted for the initial state of the cycle. This way, we utilise the advantage of accepting modifications without energy change and, simultaneously, maintain the diversity of ensemble as far as possible, compare [26]. As start, we again perform an MSLS initialisation, where the best n_{ens} final states obtained in $n_{\mathrm{msls}} = \alpha_{\mathrm{msls}} n_{\mathrm{cyc}} n_{\mathrm{ens}}$ local searches starting from random states are selected to initialise the ensemble.

The influence of n_{ens} on the performance of the ensemble TC approach is demonstrated in Figure 3b for the sequence 2D100b. For small τ, the performance slightly declines with increasing n_{ens}. However, for large τ, with increasing n_{ens}, the performance improves enormously. For ensemble size $n_{\mathrm{ens}} = 300$ and $\tau > 300$ sec, all our 100 runs ended with the global minimum energy. We remark that the ensemble sizes considered here are much larger than the ensemble sizes used in the previously performed TC investigation reference [25].

However, since treating larger ensembles within the same computing time means to perform less cycles for each of the individual samples, the improvement of the performance by increasing n_{ens} is limited and an optimum compromise must exist; it certainly depends to some extent on the sequence considered. Therefore, finding an

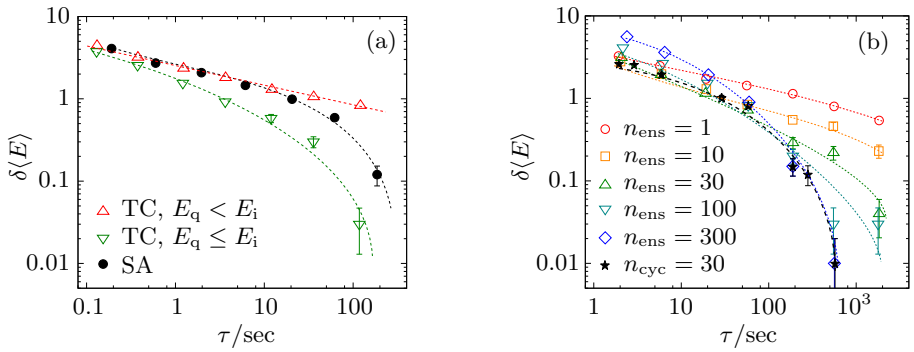

Fig. 3. (a) Performance plot of two TC versions using the different selection criteria explained in the text for the sequence 2D85. For comparison, corresponding SA results are included. (b) Performance plot of TC applied to ensembles of fixed size, n_{ens}, for the sequence 2D100b. Additionally, such a relation for fixed n_{cyc} and variable n_{ens} is included.

Table 3. Comparison of the CPU times required by TC with $n_{ens} = 1, 10, 100$, and by SA, respectively, to reach a ground state in at least 50% of 1000 runs. The times are given in seconds per run; the success percentage is added in brackets. The value of q_{acc} denotes the maximum acceleration which is reached here by substituting the best ensemble TC for SA.

Name	TC, $n_{ens} = 1$	TC, $n_{ens} = 10$	TC, $n_{ens} = 100$	SA	q_{acc}
2D64	0.9 (50%)	0.5 (50%)	1.4 (51%)	3.2 (53%)	6
2D85	10.7 (52%)	5.5 (51%)	6.6 (56%)	88.5 (55%)	16
2D100a	297 (54%)	122 (50%)	52.7 (54%)	226 (52%)	4
2D100b	>1200	169 (52%)	79.5 (50%)	1046 (58%)	13
3D48	4.8 (53%)	4.7 (52%)	7.9 (51%)	14.3 (55%)	3
3D58	561 (58%)	128 (58%)	64.9 (53%)	467 (50%)	7
3D64	>1500	354 (50%)	80.2 (56%)	1255 (61%)	16

appropriate rule of thumb would be very helpful. As a first attempt in this direction, Figure 3b includes the performance relation for n_{cyc} fixed to 30 and n_{ens} being varied.

4.4 Comparison of the algorithms

So far, we have studied the behaviour of the SA and TC procedures mainly separately. Now, we compare the individual computing efforts of these algorithms in more detail. The results for MSLS are not taken into account here, since the global minimum was found only in extremely rare cases this way.

Table 3 presents the CPU times required so that the median of 1000 runs reaches the ground state energy. For the shortest chains considered here, 2D64 and 3D48, single-state TC is by roughly a factor of 3 faster than SA. For the sequence 2D85, which has a particularly high portion of H nodes, the acceleration factor amounts even to 8. For the other four sequences, however, SA is more efficient than single-state TC.

Using the ensemble approach totally changes the situation since TC is considerably accelerated by this modification. In particular, for the sequences 2D85, 2D100b, and 3D64, the global minimum search can be accelerated by factors of 16, 13, and 16, respectively. Table 3 contains only data for three fixed ensemble sizes. Thus, finding the optimal ensemble size might yield even substantially greater acceleration factors.

It is instructive to compare the data in our Table 3 to the CPU times given in Table 2 of reference [16]: concerning finding a ground state, the optimised ensemble

TC seems to be superior even to the Wang-Landau approach. At the current stage, however, it is unclear to which extent this holds also for other, in particular for longer HP sequences, and to which extent implementation details cause this effect.

5 Conclusions

This study has been devoted to the investigation of the performances of the MSLS and TC algorithms measured against that of SA. Studying the folding of chains of 48 up to 100 amino acids out of the 2D and 3D versions of the HP model, we utilised pull moves for the modification of single states.

We found that the efficiency of the conventional iterative local search can be considerably improved by performing also the energy conserving moves found; on average, we obtained substantially lower metastable states within only slightly longer computing time in this way. In a qualified sense, our approach may be interpreted as incorporation of short Metropolis simulations with infinitely small temperature. Furthermore, we observed that the method of the preparation of the initial state has a strong influence on the final result; careful randomisation is a must. Nevertheless, for all but two of the considered examples, our MSLS code could not find the ground state within reasonable computing time in contrast to SA.

In applying the best-of-N approach to SA for three of the sequences considered here, we observed that the performance of this combined algorithm is almost independent of N for $N \leq 100$. Thus, this extended SA should be well suited for parallelisation.

After incorporating our optimised local-search method in the TC procedure, we obtained ground states of all considered sequences with reasonable computing effort. In TC, performing also energy conserving modifications leads to a substantial improvement in two ways: within the local search and in the comparison of the quenched state to the initial state of the cycle. In contrast to the basin hopping approach, the here modified selection decision of TC is still deterministic. It enables, however, the sample to move through flat basins. Thus, it should also accelerate the treatment of other combinatorial optimisation tasks with high degrees of degeneracy.

In studying the sequence 2D64, we noted missing convergence of TC caused by the distortions reached in heating being too small to leave a basin of attraction separated by a high wall from the global minimum. This problem can be avoided by initialising TC by means of MSLS where the number of trials is chosen the larger the slower the amplitude decrease in TC. Hence, such a start of TC is highly recommended.

In our numerical experiments, TC proved to be particularly efficient when it was applied to an ensemble of states instead of to a single state. For the sequences 2D100b and 3D64, the implementation of this idea led to an acceleration of the TC global minimum search by more than one order of magnitude.

Comparing the performances of TC and SA by determining the CPU time needed until at least 50% of the performed runs end up with the global minimum energy, we found the ensemble TC procedure to be far superior to SA for all sequences considered here. The advantage of TC is particularly great for the sequences 2D100b and 3D64, which were most demanding in our SA runs. In these cases, TC runs treating ensembles of 100 states were by factors 13 and 16, respectively, faster than the corresponding SA simulations. For all cases considered, we found optimised ensemble TC even to be superior to ground state search by means of Wang-Landau sampling.

In future TC and SA studies of the HP model, the efficiency of these algorithms may be improved to some extent by schedule optimisation. Moreover, finding a rule of thumb for estimating the optimum ensemble size would be highly desirable. A far larger gain might be reached by the incorporation of complex moves in TC making use of branch-and-bound strategies in their treatment. Furthermore, niching restrictions may be helpful to avoid the trapping in metastable states. For this, non-energetic

classifications of the states are required; the mean distance from the centre of the occupied space could be a useful characterisation. Such improvements may make it possible to solve also some current problems with very long chains, $L > 100$, as they were considered in reference [16], in particular the 3D136 problem [27].

Finally, we point out that TC is not only useful for the study of the discrete HP model. It can also be successfully applied to the continuous BLN model [13]. Further such investigations should be promising.

We are obliged to Johannes Zierenberg for his critical remarks. They were a substantial help in improving the presentation of our study. Furthermore, we are very thankful to Philipp Cain for his permanent and dedicated IT support.

References

1. N.A. Campbell, *Biologie*, 1st edn. (Spektrum Akademischer Verlag, Heidelberg, 1997)
2. F.E. Cohen, J.W. Kelly, Nature **426**, 905 (2003)
3. S.B. Prusiner, P. Natl. Acad. Sci. USA **95**, 13363 (1998)
4. K.A. Dill, Biochemistry **29**, 7133 (1990)
5. C.N. Pace, B.A. Shirley, M. McNutt, K. Gajiwala, FASEB J. **10**, 75 (1996)
6. G.D. Rose, P.J. Fleming, J.R. Banavar, A. Maritan, P. Natl. Acad. Sci. USA **103**, 16623 (2006)
7. T.P. Silverstein, J. Chem. Educ. **75**, 116 (1998)
8. E.M. Huque, J. Chem. Educ. **66**, 581 (1989)
9. R. Unger, J. Moult, J. Mol. Bio. **231**, 75 (1993)
10. K.F. Lau, K.A. Dill, Macromolecules **22**, 3986 (1989)
11. A. Möbius, A. Neklioudov, A. Díaz-Sánchez, K.H. Hoffmann, A. Fachat, M. Schreiber, Phys. Rev. Lett. **79**, 4297 (1997)
12. S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, Science **220**, 671 (1983)
13. F. Günther, *Structure optimisation of protein models by means of "local distortion"–quench cycles*, Master thesis, Technische Universität Chemnitz, 2013
14. R. Schiemann, M. Bachmann, W. Janke, J. Chem. Phys. **122**, 114705 (2005)
15. A.D. Ullah, L. Kapsokalivas, M. Mann, K. Steinhöfel, in *Computational Intelligence and Intelligent Systems – Proceedings of ISICA 2009*, edited by Z. Cai, Z. Li, Z. Kang, Y. Liu (Springer, Berlin, Heidelberg, New York, 2009), p. 138
16. T. Wüst, D.P. Landau, J. Chem. Phys. **137**, 064903 (2012)
17. H.-J. Böckenhauer, A.Z.M.D. Ullah, L. Kapsokalivas, K. Steinhöfel, in *Algorithms in Bioinformatics – Proceedings of WABI 2008*, edited by K.A. Crandall, J. Lagergren (Springer, Berlin, Heidelberg, New York, 2008), p. 369
18. M. Mann, S. Will, R. Backofen, BMC Bioinformatics **9**, 230 (2008)
19. S. Bromberg, K.A. Dill, Protein Sci. **3**, 997 (1994)
20. N. Lesh, M. Mitzenmacher, S. Whitesides, in *Proceedings of the seventh annual international conference on Research in computational molecular biology, Berlin, 2003*, edited by M. Vingron, S. Istrail, P. Pevzner, M. Waterman (ACM, New York, 2003), p. 188
21. N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, J. Chem. Phys. **21**, 1087 (1953)
22. S. Geman, D. Geman, IEEE Trans. Pattern Anal. Mach. Intell. **6**, 721 (1984)
23. B.A. Huberman, R.M. Lukose, T. Hogg, Science **275**, 51 (1997)
24. M.A. Miller, D.J. Wales, J. Chem. Phys. **111**, 6610 (1999)
25. A. Möbius, B. Freisleben, P. Merz, M. Schreiber, Phys. Rev. E **59**, 4667 (1999)
26. R.M. Brady, Nature **317**, 804 (1985)
27. E.E. Lattman, K.M. Fiebig, K.A. Dill, Biochemistry **33**, 6158 (1994)