Mixing Monte Carlo moves more efficiently with an evolutionary algorithm

Benoit Leblanc, Hervé Toulhoat,
Bertrand Braunschweig, Evelyne Lutton

When considering Markov Chain Monte Carlo sampling in the context of molecular simulations it is generally required to apply different types of Monte Carlo moves ([3], [1]). In addition of passing the acceptance criterion for each move, it is necessary to pick at random a move at each step, using the same distribution along the simulation. The relative frequencies for each type of move are usually empirically chosen from ranges that appears reasonable, but rather in an arbitrary manner. Here we propose an evolutionary algorithm that optimises these frequencies in order to improve the sampling efficiency.

Evolutionary algorithms are population-based stochastic optimisers [2], inspired by the Darwinian principles of evolution of species. They can be briefly described as follows: given a search space, the goal is find one (or more) points of this space that optimise a criterion:

1 Generate a set of points (called *individuals*) of the search space, that will be called a *population*.

2 Compute the criterion (positive real-valued function) for each individual, assigning them a *fitness* score.

3 Select individuals from the population, with random trial biased according to their fitness score: best individuals have more likely to be selected.

4 Selected individuals (called *parents*) are allowed to reproduce, i.e. *genetic operators* are applied: with a probability p_c each pair of parents is crossed (else duplicated), and with a probability p_m resulting *offsprings* are applied *mutation* (generally a small random perturbation of the individual). These genetic operators are specific to the type of the search space. Obtaining a new population of offsprings loop to step 2 until an end criterion (limited number of evaluations for example).

We design a specific evolutionary algorithm for the problem of relative frequencies discussed earlier. We consider NVT and NPT MC equilibrations of linear polyethylene chains in dense amorphous state (using the same model as described in [1]), a prototypical case for which sampling efficiency is critical. In that case, possible movements are: linear translation of a whole chain, rotation of an end monomer, reptation, flip (rotation of an internal monomer). The corresponding search space is then the relative frequencies μ of these four movements:

$$\mu \in S = \sum_{i=1}^{4} [0,1], \quad \text{with} \quad \sum_{i=1}^{4} \mu_i = 1 \tag{1}$$

The fitness function (quality criterion) must give a measure of how fast an MC simulation using a given μ produces uncorrelated configurations within a fixed computation budget. For this purpose it is defined using a measure of autocorrelation of end-to-end chain normalized vector C_v, and the square displacement d^2 of chains centers of mass:

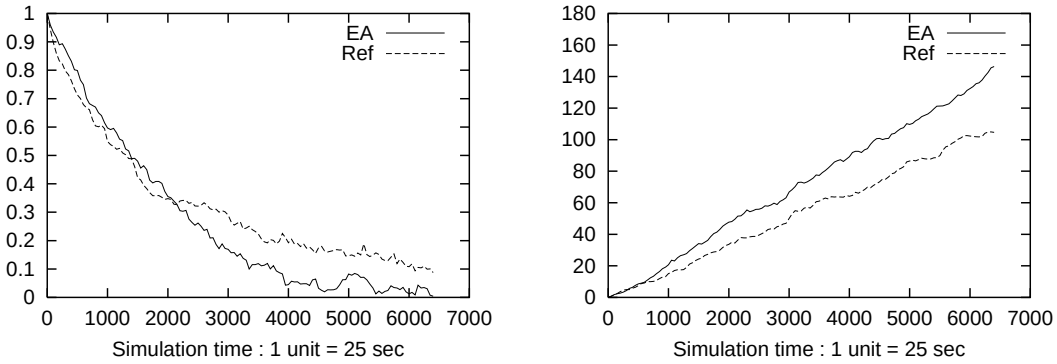
$$f(\mu, \omega) = [1 - C_v(\mu, \omega)] \times d^2(\mu, \omega) \tag{2}$$

The ω variable represents the random part that is inherent to an MC simulation. We see that a run where chains are largely displaced and loose memory of their initial orientations will be given a high fitness score.

Multiple instances of the same system (same thermodynamic conditions, but different initial configurations) are simulated in parallel using relative frequencies (individuals) produced by our evolutionary algorithm: after equally long subparts of the total simulation time fitness scores are measured and new frequencies are assigned (see [4] for more details). Applying new evolutionary methods [5] to handle noisy and time consuming fitness

evaluations we performed simulations on cluster of 200 PCs (Pentium III 750MHz, located at the INRIA Grenoble, France) with various configurations.

We report here one of them in the following conditions: $N = 10$ PE chains of 64 monomers, $P = 1atm$, $T = 450K$. There were 32 systems simulated in parallel for 160,000 seconds each, was divided in 100 equally long periods for fitness evaluations, making a total amount of 3200 fitness evaluation. In addition of this EA run, a reference run (Ref) was also performed, with the same initial states, the same simulation time, but with equal relative frequencies for each move. Finally, as the pressure is imposed for these simulations, a volume fluctuation is also added in the simulation, but with an equally prescribed frequency in both cases. The figure below displays the results (averaged on the 32 runs) in terms chain vector autocorrelation (left) and chain center of mass square displacement (right: ordinates are in reduces units, i.e. multiples of square Lennard Jones potential characteristic distance σ_{LJ}^2). They show that as the optimisation of frequencies progress, the chains are displaced faster are loose faster their initial orientations than for the reference runs, meaning that the EA simulation is more efficient regarding these criteria. For that particular case, the EA gave the highest frequencies to the reptation, and the lowest to the linear translation.



Finally we also apply the same algorithm to improve the Parallel Tempering technique, in order to optimize at the same time the relative frequencies of Monte Carlo moves and the relative frequencies of swapping between sub-systems simulated at different temperatures. Our results show that using this technique, the system can be equilibrated more efficiently at the (lowest) temperature of interest.

References

[1] V.G. Mavrantzas, T.D. Boone, E. Zervopoulou, and D.N. Theodorou. "End-Bridging Monte Carlo: A Fast Algorithm for Atomistic Simulation of Condensed Phases of Long Polymer Chains". Macromolecules, 32:5072-5096, 1999

[2] M. Schoenauer and Z. Michalewicz. "Evolutionary Computation: An Introduction". Control and Cybernetics, 26(3):307-338, 1997

[3] J.J. Siepmann and D. Frenkel: "Configurational-bias Monte Carlo: A new sampling scheme for flexible chains". Molecular Physics, 75:59-70, 1992.

[4] B. Leblanc, E. Lutton, B. Braunschweig, H. Toulhoat. "Improving molecular simulation: a meta optimisation of Monte Carlo parameters". In Proceedings of the *Congress on Evolutionary Computation*, CEC2001, Seoul, Korea, May 2001.

[5] B. Leblanc, E. Lutton, B. Braunschweig, H. Toulhoat. "History and Immortality in Evolutionary Computation". In Proceedings of the 5th International Conference of *Artificial Evolution*, EA01, Le Creusot, France, October 2001.