

## A NEURAL-NETWORK CONTROLLED DYNAMIC EVOLUTIONARY SCHEME FOR GLOBAL MOLECULAR GEOMETRY OPTIMIZATION

ANNA STYRCZ, JANUSZ MROZEK, GRZEGORZ MAZUR

Department of Computational Methods in Chemistry  
Jagiellonian University, ul. R. Ingardena 3, 30-060 Cracow, Poland  
e-mail: styrcz@chemia.uj.edu.pl

A novel, neural network controlled, dynamic evolutionary algorithm is proposed for the purposes of molecular geometry optimization. The approach is tested for selected model molecules and some molecular systems of importance in biochemistry. The new algorithm is shown to compare favorably with the standard, statically parametrized memetic algorithm.

**Keywords:** global optimization, memetic algorithm, molecular geometry.

### 1. Introduction

Studying the molecular properties and reactivity of molecular systems requires, in a majority of cases, finding the geometric structure of a molecule corresponding to the (global) energy minimum. The issue is especially difficult in studies on nano- and biosystems. The difficulty arises from the fact that the number of local minima on the potential energy hypersurface is growing exponentially with the system size (Unger and Moul, 1993; Hendrickson, 1995; Wales, 1999). Thus, as one could suspect, the search for the optimum geometry of a molecular system belongs to the class of NP-hard problems (Unger and Moul, 1993; Hendrickson, 1995).

In recent years, several attempts have been made to address the issue (Floudas and Pardalos, 2000; Pintér, 2006; Sierka *et al.*, 2007). Still, the problem of geometry optimization cannot be considered to be satisfactorily solved, and the computational chemistry community is looking forward to a robust, reliable method for finding the global minimum of molecular potential energy.

In this work we propose a new evolutionary scheme for molecular geometry optimization. The main feature of the algorithm is that a neural network is used to dynamically tune parameters of the evolutionary process. Additionally, the approach efficiently exploits domain specific features of the optimization problem.

The paper is structured as follows. In Section 2 the existing approaches to global geometry optimization are concisely presented and their drawbacks are briefly discussed. A new algorithm, addressing the shortcomings

of the preexisting ones, is introduced in Section 3, with the implementation details provided in Section 4. Results of calculations for selected molecular systems are shown and shortly discussed in Section 5. Conclusions are briefly drawn in Section 6.

### 2. Motivation

To keep the paper self-contained, we start with a short introduction to the notion of memetic algorithms (Moscato, 1999). In this approach, the Genetic Algorithm (GA) (Holland, 1975) global searcher is supplemented with a local optimization (e.g., gradient based) procedure. The local search is started in each step of the evolutionary process from the points in the solution space generated by the GA, providing the global searcher with information about the achieved local minimum. Formally, this approach can be treated as the Lamarckian type of evolution (Moscato and Cotta, 2004), with the passing of learned information (obtained in the local optimization process) to the future generations. Alternatively, the memetic approach can be understood as a way of reducing the space that has to be explored by the genetic algorithm to the subspace spanned by the local minima, leading to improved overall efficiency.

Among the variety of existing genetic algorithm families, memetic type algorithms seem to be the natural choice for the molecular geometry optimization problem. Selecting this option is substantiated by generally better efficiency of such an approach. They often require substantially fewer evaluations of the function be-

ing optimized than pure evolutionary algorithms, at the same time providing higher quality solutions (Moscato and Cotta, 2004). Moreover, the approach allows to leverage the existing high quality local optimization machinery present in a majority of quantum-chemical and molecular mechanics packages (Schmidt *et al.*, 1993; Frisch *et al.*, n.d.; te Velde *et al.*, 2001; Spoel *et al.*, 2005; Phillips *et al.*, 2005; Harrison, 1993).

Now we are in a position to analyze the performance of the (memetic) GA<sup>1</sup> in greater detail. The basic theoretical foundation follows from Holland's theorem, which shows that above-average fitness increases exponentially in successive generations (Holland, 1975). This finding seems to be contradicted by the famous No-Free-Lunch (NFL) theorem of Wolpert and Macready (1997), which states that on an average set of problems no specific optimization algorithm can be better. The contradiction, however, is only seeming and disappears upon closer scrutiny.

The crucial observation is that the performance of a GA strongly depends on its parameters. The dependence is so drastic that in many cases a small variation in the parameter set may change the convergence rate by orders of magnitude. To make things worse, (quasi)optimal parameter sets seem to differ substantially for even quite similar problems. In principle this means that using a genetic algorithm we trade the complexity of finding the solution for the complexity necessary to find the (at least quasi) optimal set of the algorithm parameters. This hypothesis is substantiated by the fact that there seems to be no generally applicable universal choice of those parameters (Eiben *et al.*, 1999).

One may see it as a manifestation of the NFL theorem. However, as pointed out by Culberson (1998), using some insight information about the optimized function softens assumptions of NFL. This is probably the main reason behind the success of memetic algorithms, where information obtained from the local search influences future generations. However, again, there is no universal set of optimum parameters for the GA based global searchers because of substantial differences in the topology of the potential energy surface between molecular systems.

The typical workaround for this problem is to find an optimal parameter set for a representative molecule and subsequently use the parametrization for a series of somewhat similar systems. Many techniques may be used for that purpose, leading to the so-called meta-GA family of methods. Common for these methods is the fact that there are two optimization algorithms, the inner, performing the requested optimization, and the outer, finding the optimal parameter set for the inner one. Most often, in the spirit of the evolutionary approach, the outer algorithm is just another GA.

<sup>1</sup>From now on all references to the GA should be understood as concerning memetic GAs.

This technique, however, is far from ideal. Firstly, the two-level procedure is much more time consuming than a simple optimization task.<sup>2</sup> Secondly, the similarity of the studied systems, which is relied upon in the meta-GA approach, is not a well defined notion. This may lead to reasonable optimization performance for some of the molecules, and extremely bad results for others, even though they are all very similar according to chemical intuition. Eventually, the procedure is based on the assumption that there exists a parameter set for a genetic algorithm which is optimal for the whole process of evolutionary optimization. Unfortunately, the assumption does not hold (Eiben *et al.*, 1999), making it impossible to obtain the optimal performance this way.

### 3. Algorithm

To avoid the drawbacks of the static meta-GA described above, a completely different approach is called for. To account for the fact that the optimal values of the parameters depend on the current state of the evolution, the optimal algorithm has to be dynamic. In this context, this means the ability to control (change) the parameter values during the evolutionary optimization (Eiben *et al.*, 1999). The observation, however, leaves us with the crucial issue of deciding which parameter values are optimal at the current stage of the evolution.

This problem has been extensively studied in the context of dynamic meta-GAs (Eiben *et al.*, 1999; de Landgraaf *et al.*, 2007; Cicirello and Smith, 2000). However, to our knowledge, all the existing solutions either use simple heuristics (Eiben *et al.*, 1999; Bäck, 1993; Wu *et al.*, 1997) or, in the most advanced cases, the parameter set is embedded into the optimization problem and optimized as a part of the solution (Spears, 1995; Angeline, 1995).

In this work we propose a Neural Network (NN) as the agent controlling the evolutionary process. The input of the NN comprises the statistical information about the current population and the performance of the current parameter set, and the output is the parameter set for the next stage of the evolutionary process. The choice of the neural network as the controlling agent is substantiated by the ability of properly trained neural networks to reproduce any mapping and to generalize it (Hertz *et al.*, 1991). The computational procedure is presented in more detail in Algorithm 1.

### 4. Implementation details

A program implementing the proposed algorithm was written in the C language. The main part of it is implemented within the framework of the Genetic Algorithm

<sup>2</sup>One may note that, in principle, the problem is recursive, and finding the optimal parameters for the outer algorithm is not a trivial task either.

**Algorithm 1** Neural Network + Genetic Algorithm + Local Search.

```
for  $i = 1$  to POPULATION_SIZE do
   $s_i \leftarrow$  local_search(random solution)
end for
generation  $\leftarrow$  0
while generation < MAX_GENERATIONS do
  get population statistics
  parameters  $\leftarrow$  neural_network(statistics)
  for  $i = 1$  to POPULATION_SIZE step 2 do
     $k =$ selection(parameters, POPULATION_SIZE)
     $j =$ selection(parameters, POPULATION_SIZE)
     $x_i, x_{i+1} \leftarrow$  crossover(parameters,  $s_j, s_k$ )
  end for
  for  $i = 1$  to POPULATION_SIZE do
     $x_i \leftarrow$  mutation(parameters,  $x_i$ )
     $s_i \leftarrow$  local_search( $x_i$ )
  end for
  generation  $\leftarrow$  generation + 1
end while
```

Utility Library (GAUL) (Adcock, n.d.). The neural network controlling the parameters of the genetic algorithm is based on the Fast Artificial Neural Network Library (FANN) (Nissen, 2003). The program includes a flexible interface to external quantum-chemical and molecular mechanics packages. The external programs are used to perform the local optimization and provide data for the fitness function evaluation.

The feed-forward (Hertz *et al.*, 1991) neural network controlling the parameters of the genetic algorithm consists of six inputs, four hidden layers comprising six neurons each and six outputs. The connection rate between the hidden layers was set to 0.8. The network is fed with floating point numbers normalized to the  $[0, 1]$  range. They represent minimum, average and maximum distances between the chromosomes, the maximum and average value of the fitness function and a fraction indicating how far in the past the best value of fitness function has been found, respectively. Two of the output neurons provide the mutation rate and crossover rate. The other output neurons indicate which type of crossover, mutation, crossover selection and mutation selection should be used in the next generation of the evolutionary process.

## 5. Results and discussion

To verify the efficiency and reliability of the presented algorithm, a series of calculations were performed for selected medium sized molecules. We chose decane ( $C_{10}H_{22}$ ), icosane ( $C_{20}H_{42}$ ), ergotamine and met-enkephalin. The linear alkanes were selected because the topology of the potential energy surface is well known for these systems. This allows us to check whether the results obtained by the proposed algorithms are physically sound.

Ergotamine and met-enkephaline were chosen because of their biological significance and because they constitute a challenging benchmark for global optimization methods. The structural formulas of the molecules under study are shown in Fig. 1.

Locally optimized structures with the corresponding energy were obtained using the AMMP (Harrison, 1993; Harrison *et al.*, 1995) molecular mechanics package. The neural network was trained according to the RPROP protocol (Riedmiller, 1994), with the learning rate set to 0.7. In the training process, the optimal parameters of the genetic algorithm were correlated with properties of the population at different stages of the evolution. The training set was constructed using data obtained from the optimization of several medium-sized molecules (alkanes, ergotamine and met-enkephaline).

The parameter sets for benchmark calculations were obtained in the meta-GA (Clune *et al.*, 2005) fashion, with the outer evolutionary algorithm performing optimization in the parameter space of the inner genetic algorithm. The fitness function of the outer algorithm was defined as

$$f(\varepsilon, \tau) = A(\varepsilon_{\max} - \varepsilon) + B(\tau_{\max} - \tau),$$

where  $\varepsilon$  represents the energy of the molecule obtained by the inner algorithm,  $\varepsilon_{\max}$  is the maximum energy recorded so far,  $\tau$  is the time spent by the inner algorithm to perform the optimization and  $\tau_{\max}$  is the maximum time accounted for. The values of the scaling parameters  $A$  and  $B$  were set to 1000 and 1, respectively. This way a strong preference for a parameter set leading to optimal solutions was established.

The procedure was performed for a series of molecules. From the quasi-optimal parameter sets obtained, three showing the best overall performance

are used as the benchmark in this work. Two of those parametrizations result from calculations for met-enkephalin, and the third set was generated from the optimization of the icosane molecule. The parameter sets, named respectively M1, M2 and I1, are reported in Table 1.

Table 1. Parameter sets for benchmark static GA calculations. See the text for details.

parameter	parameter set		
	I1	M1	M2
population size	100	50	10
crossover type	singlepoint	multiptoint	multiptoint
crossover rate	0.6	1.0	0.5
crossover selection	roulette	best of 3	roulette
mutation type	singlepoint	multiptoint	multiptoint
mutation rate	0.3	0.3	1.0
mutation selection	SUS	random	every

The first series of calculations were performed for decane (C<sub>10</sub>H<sub>22</sub>), icosane (C<sub>20</sub>H<sub>42</sub>), ergotamine and met-enkephalin. Each optimization process was repeated ten times and the energies obtained in these runs were averaged. The results of computations are reported in Table 2. Additionally, the optimized geometries were compared with the best geometry found for the molecule. The compliance of the structures was assessed by means of the Root Mean Square Deviation (RMSD) defined as

$$\text{RMSD}(A, B) = \min_R \sqrt{\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^3 w_i ((BR)_{ij} - A_{ij})^2},$$

where the matrices  $A$ ,  $B$  represent atomic coordinates,  $R$  stands for the rotation matrix and  $w_i$  denotes the weight associated with the  $i$ -th atom. We set the weights to 1 for heavy atoms and 0 for hydrogens. The results are shown in Table 3.

Another series of calculations was performed for selected derivatives of met-enkephalin. The substituent location is indicated in Fig. 2. The set of met-enkephaline derivatives was chosen to check the performance of the proposed approach for a series of similar molecular systems. The same averaging procedure was used. The results are presented in Tables 4 and 5.

As can be clearly seen, the dynamic evolutionary optimization procedure proposed in this work performs on a par with, and in some cases outperforms, the static genetic algorithm equipped with the parameter set specifically tuned for a given molecule. The superiority of the neural-network controlled dynamic approach is especially evident in the case of a series of chemically similar molecules, such as the derivatives of met-enkephalin.

It is instructive to see how the neural network control scheme influences the behaviour of the genetic evolution-

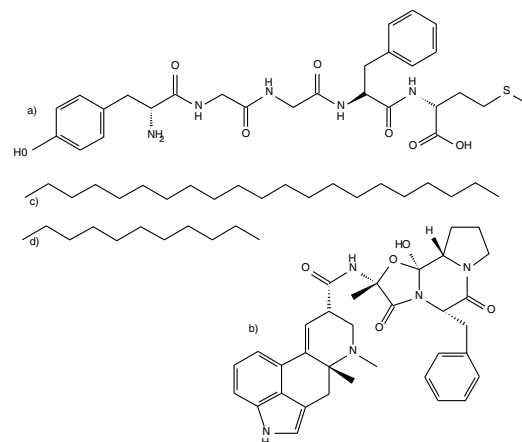


Fig. 1. Structural formulas of met-enkephalin (a), ergotamine (b), icosane (c) and decane (d).

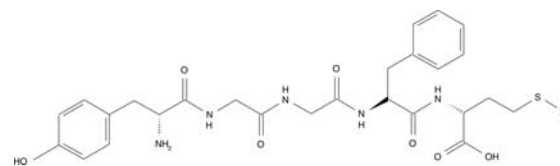


Fig. 2. Structure of a met-enkephalin derivative. X stands for a substituent.

ary process. To get some insight into the issue, we analyzed changes in the average energy and distance between the chromosomes in a population during optimization of the icosane molecule. Comparing Figs. 3 and 4, it is evident that the strongly randomized populations lead to the scattering of the average energy values while the overly concentrated ones lead to stagnation in the energy optimization process.

The neural network controlled process obtains the best solution in the shortest time. When average energy becomes stagnant, the neural-network supervisor increases the variability within the populations as indicated by the occurrence of a hump on a corresponding curve in Fig. 4. This, in turn, results in a significant decrease in average energy.

## 6. Conclusions

A new, neural network controlled, dynamic evolutionary algorithm was proposed for the purpose of molecular geometry optimization. For selected model molecular systems it was shown to perform on a par, and in several cases exceed, a statically parametrized genetic algorithm. The reliability of the algorithm, in concert with shorter calculation time, makes it a compelling alternative to the currently used global geometry optimizers.

Table 2. Energies (kcal/mol) of selected molecules optimized by evolutionary algorithms. Boldface indicates the lowest energy obtained for a given molecule. See the text for details.

molecule	value	static GA			dynamic GA
		M1	M2	I1	
decane (C <sub>10</sub> H <sub>22</sub> )	average	-36.378	-34.005	<b>-36.914</b>	-35.597
	best	-36.914	-36.914	-36.914	-36.914
	standard deviation	1.192	2.372	0.000	1.052
icosane (C <sub>20</sub> H <sub>42</sub> )	average	-33.563	-30.295	-33.834	<b>-34.153</b>
	best	<b>-34.157</b>	-31.194	-34.152	-34.155
	standard deviation	1.250	1.436	0.933	0.0006
ergotamine	average	792.546	<b>792.540</b>	797.510	793.185
	best	<b>792.010</b>	792.140	792.122	792.160
	standard deviation	0.469	0.510	1.897	1.004
met-enkephalin	average	<b>175.759</b>	176.081	187.159	176.303
	best	173.690	174.390	176.430	<b>172.410</b>
	standard deviation	2.751	1.382	3.770	2.722

Table 3. RMSD (Å) of selected molecules optimized by evolutionary algorithms. The values were calculated with respect to the best geometry obtained. Bold script indicates the lowest value obtained for a given molecule. See the text for details.

molecule	value	static GA			dynamic GA
		M1	M2	I1	
decane (C <sub>10</sub> H <sub>22</sub> )	average	0.059	0.254	<b>0.000</b>	0.190
	best	0.000	0.000	0.000	0.000
	standard deviation	0.060	0.001	0.000	0.190
icosane (C <sub>20</sub> H <sub>42</sub> )	average	0.058	0.282	0.006	<b>0.006</b>
	best	<b>0.000</b>	0.243	0.001	0.005
	standard deviation	0.053	0.060	0.005	0.001
ergotamine	average	0.694	0.786	1.131	<b>0.595</b>
	best	<b>0.000</b>	0.171	0.819	0.070
	standard deviation	0.504	0.244	0.088	0.006
met-enkephalin	average	<b>1.167</b>	1.388	1.371	1.320
	best	0.602	0.882	1.090	<b>0.000</b>
	standard deviation	0.249	0.046	0.210	0.253

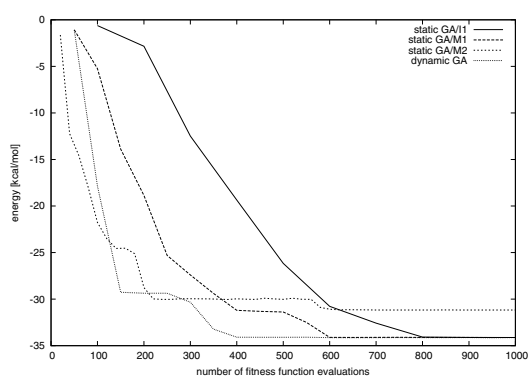


Fig. 3. Evolution of average population energy during the optimization process.

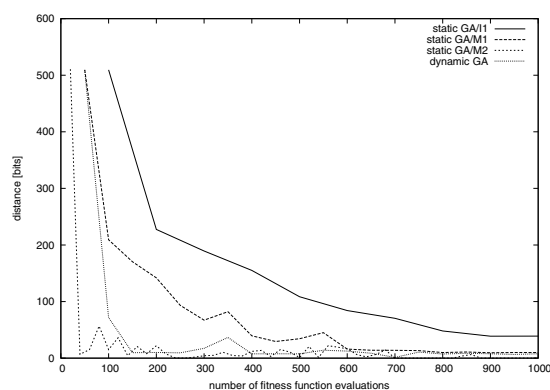


Fig. 4. Evolution of the average distance between chromosomes in a population during the optimization process.

Table 4. Energies (kcal/mol) of selected met-enkephalin derivatives computed by evolutionary algorithms. Boldface indicates the lowest energy obtained for a given molecule. See the text for details.

substituent	value	static GA			dynamic GA
		M1	M2	I1	
unsubstituted	average	<b>175.759</b>	176.081	187.159	176.303
	best	173.690	174.390	176.430	<b>172.410</b>
	standard deviation	2.751	1.382	3.770	2.722
chlorine (Cl)	average	200.821	<b>200.680</b>	213.156	200.684
	best	197.900	<b>196.030</b>	205.610	196.440
	standard deviation	2.259	3.022	4.774	3.766
hydroxyl (OH)	average	178.001	179.440	189.010	<b>176.938</b>
	best	173.880	173.340	179.560	<b>173.120</b>
	standard deviation	3.044	4.550	6.893	2.633
methyl (CH <sub>3</sub> )	average	221.333	220.999	231.851	<b>220.222</b>
	best	217.200	218.260	221.960	<b>216.320</b>
	standard deviation	2.892	2.158	6.509	2.498
propyl (C <sub>3</sub> H <sub>7</sub> )	average	<b>237.951</b>	238.387	249.257	240.284
	best	<b>230.640</b>	233.780	240.070	232.760
	standard deviation	4.044	3.607	5.573	3.625
phenyl (C <sub>6</sub> H <sub>5</sub> )	average	316.853	316.377	356.772	<b>315.714</b>
	best	313.980	311.270	323.430	<b>310.030</b>
	standard deviation	2.918	4.401	8.445	3.327
toluene (C <sub>6</sub> H <sub>4</sub> -CH <sub>3</sub> )	average	317.121	314.838	325.346	<b>314.064</b>
	best	308.610	<b>307.940</b>	310.810	309.030
	standard deviation	5.302	5.306	7.662	3.049

Table 5. RMSD (Å) of selected met-enkephalin derivatives optimized by evolutionary algorithms. The values were calculated with respect to the best geometry obtained. Boldface indicates the lowest value obtained for a given molecule. See the text for details.

substituent	value	static GA			dynamic GA
		M1	M2	I1	
unsubstituted	average	<b>1.167</b>	1.388	1.371	1.320
	best	0.602	0.882	1.090	<b>0.000</b>
	standard deviation	0.249	0.046	0.210	0.253
chlorine (Cl)	average	1.362	<b>1.084</b>	1.679	1.203
	best	0.090	<b>0.000</b>	0.969	0.117
	standard deviation	0.509	0.530	0.364	0.507
hydroxyl (OH)	average	1.271	1.418	1.573	<b>0.957</b>
	best	0.454	0.260	1.042	<b>0.000</b>
	standard deviation	0.543	0.537	0.428	0.503
methyl (CH <sub>3</sub> )	average	1.149	1.365	1.347	<b>0.930</b>
	best	0.217	0.333	0.606	<b>0.000</b>
	standard deviation	0.546	0.460	0.408	0.572
propyl (C <sub>3</sub> H <sub>7</sub> )	average	<b>1.147</b>	1.160	1.569	1.445
	best	<b>0.000</b>	0.605	0.885	0.614
	standard deviation	0.536	0.389	0.354	0.414
phenyl (C <sub>6</sub> H <sub>5</sub> )	average	1.051	1.119	1.584	<b>1.022</b>
	best	0.723	0.598	1.228	<b>0.000</b>
	standard deviation	0.328	0.429	0.239	0.544
toluene (C <sub>6</sub> H <sub>4</sub> -CH <sub>3</sub> )	average	1.288	1.216	1.622	<b>1.046</b>
	best	0.736	<b>0.000</b>	0.633	0.541
	standard deviation	0.436	0.595	0.500	0.410

## References

- Adcock, S. (n.d.). Genetic algorithm utility library, <http://gaul.sourceforge.net/>.
- Angeline, P. J. (1995). Adaptive and self-adaptive evolutionary computations, in M. Palaniswami, Y. Attikiouzel, R. Marks, D. Fogel and T. Fukuda (Eds.) *Computational Intelligence: A Dynamic Systems Perspective*, IEEE Press, Ann Arbor, MN, p. 152.
- Bäck, T. (1993). Optimal mutation rates in genetic search, in S. Forrest (Ed.), *Proceedings of the 5th International Conference on Genetic Algorithms*, Morgan Kaufmann, San Francisco, CA, p. 2.
- Cicirello, V.A. and Smith, S.F. (2000). Modeling GA performance for control parameter optimization, in L.D. Whitley, D.E. Goldberg, E. Cantú-Paz, L. Spector, I.C. Parmee and H.-G. Beyer (Eds.), *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO*, Morgan Kaufmann, Las Vegas, NV, p. 235.
- Clune, J., Goings, S., Punch, B. and Goodman, E. (2005). Investigations in meta-GAs: Panaceas or pipe dreams?, *GECCO Workshops, Washington, DC, USA*, p. 235.
- Culberson, J.C. (1998). On the futility of blind search: An algorithmic view of “no free lunch”, *Evolutionary Computation* 6(2): 109.
- de Landgraaf, W.A., Eiben, A.E. and Nannen, V. (2007). Parameter calibration using meta-algorithms, *IEEE Congress on Evolutionary Computation, Singapore*, p. 71.
- Eiben, A.E., Hinterding, R. and Michalewicz, Z. (1999). Parameter control in evolutionary algorithms, *IEEE Transactions on Evolutionary Computation* 3(2): 124.
- Floudas, C.A. and Pardalos, P. (Eds.) (2000). *Optimization in Computational Chemistry and Molecular Biology*, Nonconvex Optimization and Its Applications, Vol. 40, Springer, New York, NY.
- Frisch, M.J., Trucks, G.W., Schlegel, H.B., Scuseria, G.E., Robb, M.A., Cheeseman, J.R., Montgomery, Jr., J.A., Vreven, T., Kudin, K.N., Burant, J.C., Millam, J.M., Iyengar, S.S., Tomasi, J., Barone, V., Mennucci, B., Cossi, M., Scalmani, G., Rega, N., Petersson, G.A., Nakatsuji, H., Hada, M., Ehara, M., Toyota, K., Fukuda, R., Hasegawa, J., Ishida, M., Nakajima, T., Honda, Y., Kitao, O., Nakai, H., Klene, M., Li, X., Knox, J. E., Hratchian, H.P., Cross, J.B., Bakken, V., Adamo, C., Jaramillo, J., Gomperts, R., Stratmann, R.E., Yazyev, O., Austin, A.J., Cammi, R., Pomelli, C., Ochterski, J.W., Ayala, P.Y., Morokuma, K., Voth, G.A., Salvador, P., Dannenberg, J.J., Zakrzewski, V.G., Dapprich, S., Daniels, A.D., Strain, M.C., Farkas, O., Malick, D.K., Rabuck, A.D., Raghavachari, K., Foresman, J.B., Ortiz, J.V., Cui, Q., Baboul, A.G., Clifford, S., Cioslowski, J., Stefanov, B.B., Liu, G., Liashenko, A., Piskorz, P., Komaromi, I., Martin, R.L., Fox, D.J., Keith, T., Al-Laham, M.A., Peng, C.Y., Nanayakkara, A., Challacombe, M., Gill, P. M.W., Johnson, B., Chen, W., Wong, M.W., Gonzalez, C. and Pople, J.A. (n.d.). *Gaussian 03, revision c.02*. Gaussian, Inc., Wallingford, CT.
- Harrison, R.W. (1993). Stiffness and energy conservation in molecular dynamics: An improved integrator, *Journal of Computational Chemistry* 14(9): 1112.
- Harrison, R.W., Chatterjee, D. and Weber, I.T. (1995). Analysis of six protein structures predicted by comparative modeling techniques, *Proteins: Structure Function and Genetics* 23(4): 463.
- Hendrickson, B. (1995). The molecule problem: Exploiting structure in global optimization, *SIAM Journal of Optimization* 5(4): 835.
- Hertz, J., Krogh, A. and Palmer, R.G. (1991). *Introduction to the Theory of Neural Computation*, Addison-Wesley, Redwood City, CA.
- Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI.
- Moscato, P. (1999). Memetic algorithms: A short introduction, in D. Corne, M. Dorigo and F. Glover (Eds.), *New Ideas in Optimization*, McGraw-Hill, London, p. 219.
- Moscato, P. and Cotta, C. (2004). Memetic algorithms, *Optimization Techniques in Engineering*, Springer-Verlag, New York, NY, p. 53.
- Nissen, S. (2003). Implementation of a fast artificial neural network library (FANN), <http://fann.sf.net>.
- Phillips, J.C., Braun, R., Wang, W., Gumbart, J., Tajkhorshid, E., Villa, E., Chipot, C., Skeel, R.D., Kale, L. and Schulten, K. (2005). Scalable molecular dynamics with NAMD, *Journal of Computational Chemistry* 26(16): 1781.
- Pintér, J.D. (Ed.) (2006). *Global Optimization, Nonconvex Optimization and Its Applications*, Vol. 85, Springer, New York, NY.
- Riedmiller, M. (1994). Rprop—Description and implementation details, *Technical report*, Institute for Logic, Complexity and Deduction Systems, University of Karlsruhe, Karlsruhe.
- Schmidt, M.W., Baldrige, K.K., Boatz, J.A., Elbert, S.T., Gordon, M.S., Jensen, J.H., Koseki, S., Matsunaga, N., Nguyen, K.A., Su, S., Windus, T.L., Dupuis, M. and Montgomery, J.A. (1993). General atomic and molecular electronic structure system, *Journal of Computational Chemistry* 14(11): 1347.
- Sierka, M., Döbler, J., Sauer, J., Santambrogio, G., Brümmer, M., Wöste, L., Janssens, E., Meijer, G. and Asmis, K.R. (2007). Unexpected structures of aluminum oxide clusters in the gas phase, *Angewandte Chemie International Edition* 46(18): 3372–5.
- Spears, W.M. (1995). Adapting crossover in evolutionary algorithms, *Proceedings of the 4th Annual Conference on Evolutionary Programming, San Diego, CA, USA*, p. 367.
- Spoel, D.V.D., Lindahl, E., Hess, B., Groenhof, G., Mark, A.E. and Berendsen, H.J.C. (2005). GROMACS: Fast, flexible, and free, *Journal of Computational Chemistry* 26(16): 1701–1718, <http://dx.doi.org/10.1002/jcc.20291>.

- te Velde, G., Bickelhaupt, F.M., Baerends, E.J., Fonseca Guerra, C., van Gisbergen, S.J.A., Snijders, J.G. and Ziegler, T. (2001). Chemistry with ADF, *Journal of Computational Chemistry* **22**(9): 931.
- Unger, R. and Moulton, J. (1993). Finding the lowest free energy conformation of a protein is an NP-hard problem: Proof and implications, *Bulletin of Mathematical Biology* **55**(6): 1183–1198, <http://dx.doi.org/10.1007/BF02460703>.
- Wales, D.J. (1999). Global optimization of clusters, crystals, and biomolecules, *Science* **285**(5432): 1368.
- Wolpert, D.H. and Macready, W.G. (1997). No free lunch theorems for optimization, *IEEE Transactions on Evolutionary Computation* **1**(1): 67.
- Wu, A.S., Lindsay, R.K. and Riolo, R.L. (1997). Empirical observations on the roles of crossover and mutation, in T. Bäck (Ed.), *International Conference on Genetic Algorithms, ICGA*, Morgan Kaufmann, San Francisco, CA, p. 362.
- Anna Styrz** is a Ph.D. student at the Faculty of Chemistry, Jagiellonian University in Cracow (Poland). She attained her M.Sc. in chemistry in 2006 at the same university. Her current research interest concentrates on molecular geometry optimization and hybrid optimization algorithms.
- Janusz Mrozek** D.Sc., Ph.D., associate professor in the Department of Computational Methods in Chemistry, Faculty of Chemistry, Jagiellonian University in Cracow, Poland. Research profile: theory of valency, electronic structure theory, relations between electronic structure and geometry of closed and open molecular systems, hybrid optimization algorithms.
- Grzegorz Mazur** is an assistant professor at the Department of Computational Methods in Chemistry of Jagiellonian University in Cracow (Poland). He attained his Ph.D. in chemistry in 2001 at the same university. His current research interests include theoretical description of the excited states properties and optimization of quantum chemistry algorithms.

Received: 14 August 2010

Revised: 31 January 2011