# GUIDED OPTIMIZATION METHOD FOR FAST AND ACCURATE ATOMIC CHARGES COMPUTATION

Jana Pazúriková
Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno, Czech Republic
email: pazurikova@ics.muni.cz

Aleš Křenek
Luděk Matyska
Institute of Computer Science
Masaryk University
Botanická 68a
602 00 Brno, Czech Republic

**KEYWORDS**
global and local optimization, model analysis, stochastic model, computational chemistry

**ABSTRACT**

Current advances in hardware and algorithm development allow the life science researchers to replace the experiment with a computer simulation. A key object of these computations is a molecule - a group of atoms interconnected via a cloud of electrons. For its computational processing, electrons around the atom are often represented by one number: partial atomic charge. It can be calculated by quantum mechanics (QM), which offers high accuracy at the cost of long computation time, or markedly faster by empirical methods such as Electronegativity Equalization Method (EEM). Empirical methods calibrate their parameters to the particular QM charge calculation approach by multi-dimensional optimization procedure. This work systematically summarizes and compares the accuracy and computational performance of available EEM parameterization approaches with local, global or combined optimization (least squares, evolutionary and genetic algorithms). Moreover, we propose a new methodology called guided minimization. We found that local optimization plays a crucial role in the parametrization, and only methodologies combining a global and a local optimization provide high-quality EEM parameters. Furthermore, we observed that global iterations of evolutionary of genetic algorithm often do not contribute to the result. Therefore, we reduced the global search method to guided minimization that achieves same or better accuracy than state-of-the-art methods and surpasses them with simplicity and speed.

## INTRODUCTION

Advances in computer science together with availability of high performance computational resources enable the researchers to effectively replace an experiment with a computer simulation. Especially life sciences have intensified their computational approaches: chemoinformatics, bioinformatics and computational chemistry have emerged and are taking more and more attention. A core object of research in these disciplines is a molecule, simplified as a group of atoms interconnected via a cloud of electrons. The location of atoms and distribution of electrons are therefore essential information for modelling and simulating the key life science processes such as chemical reactions, interactions between molecules, solvation, biodegradation and so on.

The electrons do not occupy a stable spot in a space - we can obtain only a probability of their occurrence at some point (Jensen 2007). We can represent the cloud of electrons as electron densities in dedicated parts of the space (so called molecular orbitals). However, this approach causes many inconveniences due to its complexity. Instead, we can represent electron distribution around the atom as one real number, partial atomic charge. This concept found its use in many applications in chemoinformatics (Tervo et al. 2005), bioinformatics (Vařeková et al. 2013) and computational chemistry (Park et al. 2006).

The charges can be calculated via quantum mechanical (QM) methods, but these demand high computational resources, so cheaper and faster methods based on empirical parameters were developed. Empirical methods open a possibility to use charges in common modelling and simulation tasks thanks to their speed and low cost. The challenging part is the calculation of empirical parameters, values that are calibrated to provide close fit to QM charges from the training set through an optimization in multi-dimensional space.

Several Electronegativity Equalization Method (EEM) parametrization approaches were developed, while they are based on least squares parametrization method (Mortier et al. 1986), global optimizations (Ouyang et al. 2009) or a combination of global and local optimizations (Menegon et al. 2002; Bultinck et al. 2004; Chaves et al. 2006; Raček et al. 2016). Each of these methodologies brings its strong and weak points. Least squares parametrization methods can not handle heterogeneous datasets (Raček et al. 2016), genetic algorithms have high computational demands and require execution with proper settings. A goal of our work is to discover

real methodological demands of EEM parameterization, e.g.: Can we use only local optimization or just global optimization? Or it is necessary to combine them? How much does the global optimization contribute to the result? Can sole differential evolution (Ouyang et al. 2009) successfully parametrize heterogeneous datasets?
For this purpose, we describe, analyze and compare various methodologically different EEM parameterization approaches. They include published EEM parameterization methods (e.g. least squares optimization (Mortier et al. 1986), differential evolution (Ouyang et al. 2009), genetic algorithm combined with local optimization method (Menegon et al. 2002; Bultinck et al. 2004; Chaves et al. 2006)), well-known optimization methods never used in EEM (sole genetic algorithms, NEWUOA (Powell 2004a)) and also our innovative approach GDMIN (guided minimization). GDMIN development was based on our rich experience with EEM parameterization and it connects the top class local minimization approach NEWUOA with effective global search.

## RELATED WORK

As a purely theoretical concept, partial atomic charges cannot be experimentally measured. Moreover, as there are many ways how to reduce electron distribution into one value, there are many methods how to compute atomic charges. The most accurate methodology for their calculation is an application of QM, which works with QM theory level, basis set and charge calculation scheme (Gupta 2005). These three components can be variously combined, e.g. B3LYP/6-311G/NPA denotes the method based on B3LYP theory (Becke 1993) using 6-311G basis set and Natural Population Analysis (Reed et al. 1985). Different combinations produce different charge values and are suitable for different applications. For example, charges computed by B3LYP/6-311G/NPA were used in a simulation of heme-containing complexes (Rong et al. 2007) or in prediction of $pK_a$ values (Vařeková et al. 2013). The validity of computed charges can be verified indirectly, as they determine e.g. dissociation constant for acid in the solution that can be compared to experimental measurements.
Quantum mechanics methods provide precise results, but have high computational demands, e.g. 20 CPU days for protein ubiquitin charge calculation (Ionescu et al. 2013). To overcome this, many empirical methods have been developed to approximate QM using concepts based on common physico-chemical laws and including some simplifications. The empirical methods often work with parameters: empirical constants calculated to mimic QM charges from the training set. With known parameters, charges can be computed quickly for atoms in any molecule with the same atom types (usually based on atom elements and their bonds, e.g. atom type C2 represents a carbon with two bonds).

Electronegativity equalization method (Mortier et al. 1986), the most popular and reliable empirical approach, stands on the electronegativity equalization principle. (Sanderson 1951) states that when the molecule is formed, the electron distribution spreads around atoms and their electronegativities (the ability to keep electrons near) get equalized, see Equation (1).

$$\overline{\chi} = \chi_1 = \chi_2 = \cdots = \chi_n \tag{1}$$

Equalized electronegativity of the given atom depends on its charge, charges of surrounding atoms and their distance and also on the empirical parameters, see Equation (2a). Equations (1) and (2a) together with the charge conservation principle in Equation (2b) give us the system of linear equations (LS). During the parametrization phase, we know $q_i$ (QM charges from the training set), and we compute $\mathbf{A}$, $\mathbf{B}$, $\kappa$. During the charge calculation phase, we use precomputed $\mathbf{A}$, $\mathbf{B}$, $\kappa$ to find $q_i$ for atoms in an arbitrary molecule.

$$\chi_i = A_i + B_i q_i + \kappa \sum_{j \neq i} \frac{q_j}{R_{ij}} \tag{2a}$$

$$Q = \sum_{i}^{N} q_i \tag{2b}$$

where $i, j$ denote the given atom, $q$ charge, $R$ the interatomic distance, $\kappa$ parameter shared by all atom types, $A_i$, $B_i$ parameters shared by all atoms of the same atom type, $N$ number of atoms, $Q$ the total charge of the molecule.
The parametrization represents an optimization problem in many dimensions: 2× number of atom types ($\mathbf{A}$ and $\mathbf{B}$) +1 ($\kappa$). We look for the parameters that produce charges as close to QM charges as possible. Common fitness functions include (squared) Pearson correlation coefficient, root-mean square deviation (RMSD) or Euclidean distance.
Least squares parametrization method (LR) optimizes only through one dimension: $\kappa$. Parameters $\mathbf{A}$ and $\mathbf{B}$ are directly computed via least squares minimization. The method is extremely fast, but it fails in case of training sets that are heterogeneous in terms of molecules' types variability. Optimization methods searching through all dimensions deal with them better. Ouyang et al. applied differential evolution (DE) (Ouyang et al. 2009). In their experiments, they used 141 small organic molecules of thirteen atom types (from five elements) to find EEM parameters. They achieved $R^2$ above 0.98 when validating on polypeptides. No local optimization method for polishing the results is mentioned. (Menegon et al. 2002; Chaves et al. 2006) applied genetic algorithm and minimized the result with simplex local optimization method (Nelder and Mead 1965). Bultinck et al. applied simplex also at each member of a new population at every step (Bultinck et al. 2004). The population of 10-30 items undergone a genetic algorithm for 500 iterations. They all achieved

high correlation on small homogeneous datasets (< 200 molecules) of five atom elements.

## OPTIMIZATION METHODS

We compared several global optimization methods for EEM parametrization:

- least squares method (LR)
- local optimization method applied on one random vector (NEWUOA)
- differential evolution (DE)
- differential evolution combined with local minimization (DEMIN)
- genetic algorithm (GA)
- genetic algorithm combined with local minimization (GAMIN)
- guided minimization (GDMIN)

The methods LR (Mortier et al. 1986), NEWUOA (Powell 2004a), DE (Ouyang et al. 2009), GAMIN (Menegon et al. 2002; Bultinck et al. 2004; Chaves et al. 2006), DEMIN (Raček et al. 2016) were taken from literature. The approach GDMIN was newly developed by us and to our best knowledge, it has not been applied to EEM parametrization before.

LR iteratively calculates **A**, **B** for fixed $\kappa$, slightly increasing $\kappa$ each time within empirical interval.

NEWUOA locally optimizes unconstrained problems and does not need derivatives, which makes computing more efficient even for systems with hundreds of variables (Powell 2004a).

All other compared algorithms, DE(MIN), GA(MIN) and GDMIN can be described with the following terms. The *population* consists of vectors, each *vector* consists of $\kappa$ and $A_i$, $B_i$ for all atom types. The *evaluation* of the vector means computing the EEM charges with parameters included in the vector and comparing them to QM charges with the *fitness function*, e.g. $avg(\mathrm{RMSD}_a)$, the average of RMSD through atom types.

The pseudoalgorithm 1 of global methods based on population describes the basic flow.

All methods create a random population, *MIN methods locally minimize part of it. The best vector is stored. Differential evolution then iteratively creates a new trial vector by adding the difference between parents' values to the original vector, as Equation (3) shows (Storn and Price 1997). Trial might be also minimized locally in DEMIN. If trial outperforms the so-far-best vector, we save it.

$$trial \leftarrow original + F \times (a - b) \tag{3}$$

where $F \in \{0, 1\}^n$, $a, b$ are randomly selected from population, *original* is (randomly or according to fitness) selected from population.

Genetic algorithm iteratively creates a new population by crossover of parents (better half of population) and mutation. Parents are minimized in GAMIN. If the best vector in population outperforms the so-far-best vector, we save it.

In the end, *MIN methods locally minimize the best vector and then return it as the result.

Guided minimization basically skips the iterative creation of new trial vectors/generations, and performs only population generation, minimization of its part and minimization of the best vector.

**Algorithm 1** Algorithm for EEM Parametrization with DE(MIN), GA(MIN) or GDMIN

```
1: function FIND_PARAMETERS(method)
2:     population ← generate_random_population()
3:     ∀ x ∈ population: R(x) ← EVALUATE(x)
4:     if method is *MIN then
5:         ∀ x ∈ (subset ⊂ population): x ← NEWUOA(x)
6:     best ← find_the_best(population)
7:     switch method do
8:         case de: DE(population)
9:         case ga: GA(population)
10:    best ← NEWUOA(best)
11:    return best

1: function EVALUATE(x)
2:     ∀ molecule ∈ training_set: eem_charges ← eem(x)
3:     R(x) ← compare(eem_charges, qm_charges)

1: function DE(population)
2:     loop
3:         trial ← combine(select_random(population), select_random(population))
4:         R(trial) ← EVALUATE(trial)
5:         if method is *MIN then
6:             trial ← NEWUOA(trial)
7:         if R(trial) < R(best) then
8:             best ← trial

1: function GA(population)
2:     loop
3:         parents ⊂ population
4:         parents ← NEWUOA(parents)
5:         population ← generate_new_population(parents)
6:         ∀ x ∈ population: R(x) ← EVALUATE(x)
7:         this_generation_best ← find_the_best(population)
8:         if R(this_generation_best) < R(best) then
9:             best ← this_generation_best
```

## IMPLEMENTATION

We implemented DE(MIN), GA(MIN) and GDMIN as a part of NEEMP (Raček et al. 2016), a tool for calculation, parametrization and validation of EEM parameters. Local minimization in *MIN methods is done by Powell's NEWUOA algorithm (Powell 2004a), implemented in FORTRAN (Powell 2004b). Random generation of population is done by Latin Hypercube Sampling, implemented by (Burkardt 2004). For solution of linear equation system, LAPACK implementations are applied and parallelized by OpenMP.

## EVALUATION METHODS

To compare methods, we analysed their accuracy, validity of produced parameters and computational performance.

To evaluate the accuracy of methods, we ran parametrizations on three datasets complementing each other in variability of molecule and atom types. Dataset *set1* consists of 1956 small organic molecules compounded from 6-176 atoms of eight atom types of five atom elements. Dataset *set2* consists of 4475 small organic molecules compounded from 5-124 atoms of seventeen atom types of ten atom elements. Dataset *set3* consists of 4443 small organic and inorganic molecules, organometals and peptides compounded from 5-305 atoms of fifteen atom types of nine atom elements. For further details about datasets, see (Raček et al. 2016). We compared calculated charges to QM charges from the training set by applying average of root mean square deviations per atom type ($avg(\mathrm{RMSD}_a)$) as a fitness function for all runs. For each method, we ran all combinations of settings described in Table 1 for all datasets. For each configuration, we ran four experiments with different random seeds. We call a specific combination of settings a *configuration*.

Table 1: Method's Settings Applied in Experiments
(Following shortcuts are used: *pop*ulation, *iter*ations, *min*imized part of *pop*ulation, *iter*ations applied at the *beg*inning, *iter*ations applied in the *end*.)

| | | |
|---|---|---|
| LR | interval and step | $< 0, 1 >$ by 0.05 |
| DE | pop size | 50 100 250 500 1000 |
| | iters | 500 1000 2500 5000 10000 |
| DEMIN | pop size | 50 100 250 500 1000 |
| | iters | 100 5000 |
| GA | pop size | 50 100 250 500 1000 |
| | iters | 500 1000 2500 |
| GAMIN | pop size | 5 10 20 50 |
| | iters | 5 50 100 500 |
| GDMIN | pop size | 50 100 250 500 1000 |
| | min pop size | 1 10 20 50 100 |
| | iters beg | 100 250 500 1000 |
| | iters end | 500 1000 2000 3000 |

To validate the parameters on a non-training dataset, we computed EEM charges for validation datasets and compared them with QM charges. We had two validation datasets: we call the first one *test set* (Geidl et al. 2015) consisting of 657 small organic molecules of seven atom types of four atom elements. We used it for set1 and set2 validation. The second validation dataset is the whole *ligandexpo* database, details in (Raček et al. 2016). It is a highly diverse set of 17769 small organic and inorganic molecules, organometals and peptides of fifteen atom types of eleven atom elements. Training set3 presents a random quarter of ligandexpo.

To show the computational performance of different optimization methods, we also analysed computation time and evaluated the number of linear system's solutions depending on the method's settings.

## RESULTS AND DISCUSSION

To evaluate the accuracy and efficiency of optimization methods, we investigated minimal configurations necessary for the method to satisfy following requirements: total $R^2 > 0.9$, total RMSD $< 0.1$, per atom type $R_a > 0$ for all atom types and per atom type $\mathrm{RMSD}_a < 0.2$ for all atom types. We denote such parameters of high-quality. If at least three out of four runs of same configuration with different random seeds satisfied these requirements, we call such configuration reliable. For methods that did not reach the high-quality requirements, we inspected if they satisfy benevolent requirements: $R^2 > 0.85$, $R_a > 0$, RMSD $< 0.15$, $\mathrm{RMSD}_a < 0.25$.

### Local Optimization Without Global

First, to show the role of global optimization in EEM parametrization, we ran just local optimization, i.e. minimize one random vector with NEWUOA. We found no high-quality parameters, regardless of number of local iterations. Results were basically random, $R^2 \approx 0$. Clearly, sufficiently high number of experiments might succeed in finding high-quality parameters. However, some sort of global search would make this more reliable and independent of random seed. Therefore, global method should be present in the parametrization.

### Global Optimization Without Local

Second, to show the role of local optimization method in EEM parametrization, we ran just global methods, without any local minimization. We found no high-quality parameters. GA did not get over random results, but LR and DE managed.

*Least Squares Method*

LR, apart from other methods we compare in this work, produces only one set of parameters for one training

dataset due to its non-stochastic nature. Found parameters were not of high-quality, but many satisfied benevolent requirements, see Table 2. With increasing heterogeneity in atom and molecule types (in set2 and set3), the results deteriorated.

Table 2: Quality of Parameters Calculated by Least Squares Parametrization Method (Overline denotes average, subscript $a$ per atom statistics.)

| | $R^2$ | $\overline{R^2_a}$ | RMSD | $\overline{\text{RMSD}}_a$ |
|---|---|---|---|---|
| set1 | 0.931 | $0.806 \pm 0.085$ | 0.092 | $0.110 \pm 0.080$ |
| set2 | 0.921 | $0.628 \pm 0.256$ | 0.102 | $0.152 \pm 0.123$ |
| set3 | 0.864 | $0.502 \pm 0.303$ | 0.145 | $0.280 \pm 0.563$ |

*Genetic Algorithm*
We found no high-quality parameters. Moreover, the results were often random, $R^2 \approx 0$, although sometimes $R^2 \approx 0.5$ occurred. As the randomly generated vectors in initial population only rarely get over $R^2 > 0.2$, the population does not have quality necessary for genetics to gradually improve.

*Differential Evolution*
We found no high-quality parameters, but the differential evolution performed better than genetic algorithm. For set1, some runs even reached benevolent requirements. However, DE performed worse than LR.

## Global Optimization Combined With Local

As results above suggest, neither local nor global optimization are able to produce high-quality EEM parameters. Chaves et al. combined GA with local simplex optimization (Chaves et al. 2006), we developed DEMIN recently (Raček et al. 2016). We present GDMIN as a novel contribution here for the first time. In all methods, we apply NEWUOA as the local minimization method.

*Genetic Algorithm With NEWUOA*
We found high-quality parameters with genetic algorithm combined with NEWUOA, but no reliable configuration. We experimented with local minimization applied at parents in each iteration and at the best vector after genetic algorithm finished.
When applying NEWUOA at both places (GAMIN2), we achieved high-quality parameters in one or two runs with population of size 10 (set1, set2) or 20 (set3). In all cases, the success was independent on the numbers of iterations, i.e. the same results were obtained by 50 and 500 iterations (for set1 and set2, even five iterations sufficed).
When applying NEWUOA only at the genetic result vector (GAMIN1), the results were random most of the time ($R^2 \approx 0$). Only rarely one out of four runs managed to satisfy the requirements for set1.

*Differential Evolution With NEWUOA*
We found high-quality parameters with differential evolution combined with NEWUOA and many reliable configurations. We experimented with local minimization placed at three points: at the part of initial population (on vectors with $R^2 > 0.2$ and $R > 0$), at trials (if $R^2 > 0.6$ and $R > 0$) and at the vector resulting from the evolution process.

When applying NEWUOA at all three places (DEMIN3), almost all runs reliably produced high-quality parameters for set1 and set3. For set2, the best results were found with population of 250, more vectors caused increased dependency on random seed. In general, no trials were better than the best vector of minimized initial population, thus making evolution iterations superfluous.

When applying NEWUOA on trials and the evolution result vector (DEMIN2), all configurations reliably resulted in high-quality parameters for set1 (regardless of population size), set2 (population 50 and 100) and set3 (for population 250 and more). For larger populations in set2, atom type S2 (sulphur with two bonds) exceeded RMSD little above the limit 0.2. In those cases, 5000 evolution iterations managed to overcome this. Trial vectors gradually improved, reaching $R^2 \approx 0.5$ at the end of evolution process. Local minimization at the resulting vectors managed the rest.

When applying NEWUOA only on the evolution result vector (DEMIN1), the situation mirrored the previous case, thus again making the evolution iterations expendable.

*Guided Minimization With NEWUOA*
We found high-quality parameters at almost all configurations with our novel method. At least three out of four runs were of high-quality at following settings for set1/set2/set3: population 100/100/250 or more, selecting a single one best vector (or more), minimizing it (them) for 100 coarse local iterations (or more), selecting the best one, and minimizing for 3000/3000/2000 iterations. Some high-quality parameters were found also with smaller populations or lower number of iterations, but only in one or two runs out of four, thus more dependant on a "lucky" random seed.

We observed some overtraining, when more iterations applied at the minimal selection or at the best vector at the end resulted in worse parameters. For example, set3 runs with 1000 population, selecting the best one, running 100 coarse and 3000 fine local iterations found high-quality parameters in all four runs. With 4000 iterations at the end, three of them fell bellow the requirements. Fitness function not considering $R^2$ causes fall into low RMSD sacrificing the correlation.

Table 3 shows the quality measurements of the best parameters produced by smallest reliable configuration.

Table 3: Quality of Parameters Calculated by Guided Minimization with the Smallest Reliable Configuration Satisfying $R^2 > 0.9$ and RMSD $< 0.1$ and $\forall a: \ R_a > 0$ and $\text{RMSD}_a < 0.2$

| | configuration | random seed | $R^2$ | $\overline{R_a^2}$ | RMSD | $\overline{\text{RMSD}_a}$ |
|---|---|---|---|---|---|---|
| set1 | 100 pop, 1 selected, 100 coarse, 500 fine local iters | 202 | 0.970 | $0.789 \pm 0.140$ | 0.058 | $0.074 \pm 0.026$ |
| set2 | 100 pop, 1 selected, 100 coarse, 3000 fine local iters | 200 | 0.959 | $0.674 \pm 0.257$ | 0.070 | $0.078 \pm 0.033$ |
| set3 | 250 pop, 1 selected, 100 coarse, 2000 fine local iters | 202 | 0.970 | $0.741 \pm 0.227$ | 0.0653 | $0.0632 \pm 0.035$ |

### To Sum Up

Sole local (NEWUOA) and sole global (LR, DE, GA) optimization does not suffice for EEM parametrization. Their combinations (DEMIN, GAMIN, GDMIN) manage multi-dimensional optimization well even for heterogeneous datasets and often reliably, not depending on lucky random seed. In our experiments, evolution or genetic iterations were repeatedly redundant, high-quality results occurred regardless of their number. Our method, simple guided minimization (GDMIN), which was developed to reflect this, has succeeded to produce high-quality parameters reliably with population of same size than DEMIN. Moreover, GDMIN excelled also in accuracy, for all datasets the best results from all our experiments were obtained by guided minimization. However, DEMIN did not fall behind much, the differences were tiny. GAMIN produced the worst parameters out of these three methods although still satisfying the requirements.

### Validation

We validated the best parameters acquired by our experiments on validation datasets test set (set1, set2) and ligandexpo (set3). All of them were produced by GDMIN.

Parameters trained on set1 by GDMIN (500-vector population, 1 selected, 250 coarse and 500 fine iterations, random seed 203) applied to compute charges for test set and compared with test set's QM charges achieved $R^2$ 0.9718 and RMSD 0.0597. Top figure 1 shows the correlation per atom.

Parameters trained on set2 by GDMIN (100-vector population, 5 selected, 500 coarse and 3000 fine iterations, random seed 202) used for test set's charge computation also showed excellent agreement: $R^2$ 0.9713, RMSD 0.0607. Bottom figure 1 shows the correlation per atom.

Parameters trained on set3 by GDMIN (250-vector population, 5 selected, 250 coarse and 3000 iterations, random seed 203) used for EEM charge computation of whole ligandexpo dataset show high quality in Figure 2. They achieved $R^2$ 0.9724 and RMSD 0.0622.

Therefore, the validation successfully showed a robustness of EEM parameters calculated via GDMIN.

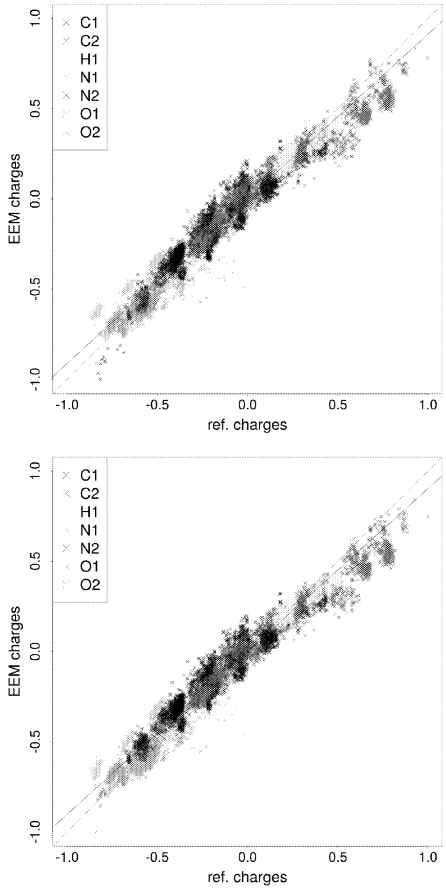



Figure 1: Correlation of Test Set's QM Charges and EEM Charges Trained on Set1 and Set2

### Computation Time Comparison

We analysed the computation time of different optimization methods in a theoretical and practical way.

The bottleneck of EEM parametrization is to find the solution of the system of linear equations (LS). Therefore, we took one solution for one molecule as a measure unit and computed how many are needed, see Table 4. Actual numbers would be multiplied by the number of molecules in the training set, we omit these as they do not influence the order. If we evaluate the expressions with settings from reliable configurations (or average, if

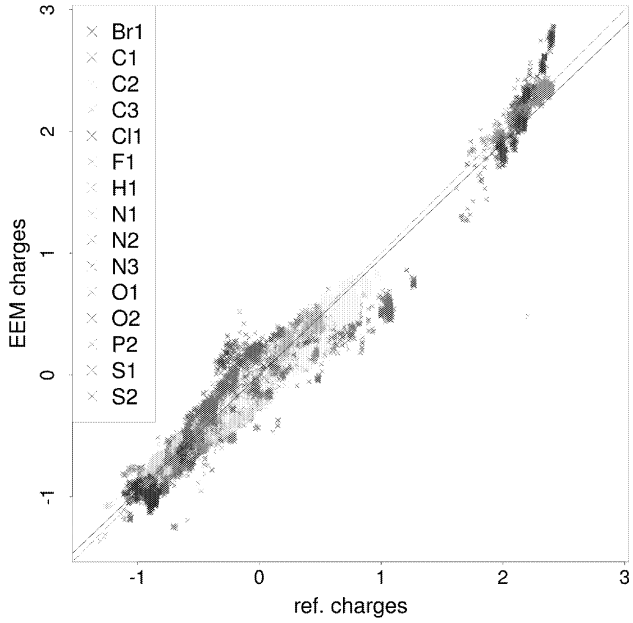



Figure 2: Correlation of Ligandexpo's QM Charges and EEM Charges Trained on Set3

they do not have reliable ones), we get following order of computational complexity: LR $<$ DE $\simeq$ NEWUOA $<$ DEMIN1 $\simeq$ GDMIN $<$ DEMIN2 $<$ DEMIN3 $<$ GA $<$ GAMIN1 $<$ GAMIN2.

The order of method's complexity stems from the order of settings. Generally, population size is smaller than the number of iterations, $P < G$, the number of global iterations tends to be smaller than the number of local iterations, but not necessarily. The number of local iterations applied at the result vector, during global iterations and on the initial population gradually decreases, $L_1 > L_2 \geq L_3$. Also, in GDMIN the number of vectors selected from initial population to be minimized is much lower then the population size, $M \ll P$.

The walltime of computations in our experiments corresponded with this order. It ranged from seconds (LR, DE) through minutes and tens of minutes (NEWUOA, DEMIN, GDMIN) to hours and days (GA, GAMIN). We ran all experiments in parallel with OpenMP on four cores of Intel E5-2670 2.6 GHz. They all required just a few tens of MB of RAM.

From the methods that successfully found high-quality parameters (i.e. DEMIN, GAMIN, GDMIN), GDMIN and DEMIN are comparable, GAMIN is much more computationally demanding, as Figure 3 shows.

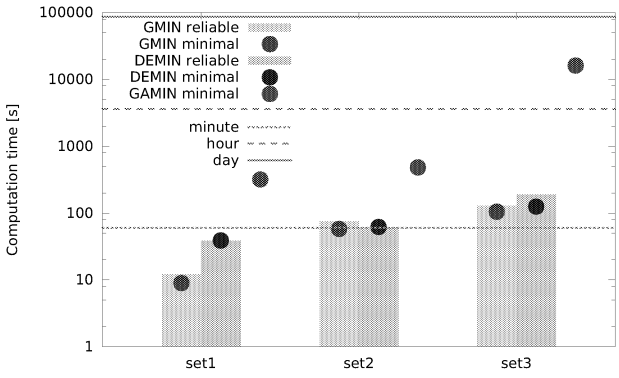
## CONCLUSION

In this work, we compared several optimization methods in parametrization of empirical atomic charges. We found that nor local neither global optimization alone can produce high-quality parameters, but their combination effectively does so. We observed that evolution or genetic iterations in combined methods often do not contribute to the result, but some sort of global search must be present. To reflect this, we simplified global search method and developed guided minimization method (GDMIN), which has not been applied to EEM parametrization before. It achieves accuracy better or comparable to the best methods available and surpasses them in simplicity, computational performance and ease of implementation. For future work, we aim to parametrize with datasets containing the protein fragments. These, because of the number of atom types and heterogeneity, still resist state-of-the-art EEM methods. More available and more accurate charges can afterwards contribute to modelling and simulations of protein docking, folding and interaction with drugs.

Table 4: How Many Solutions of Linear System (LS) Are Needed in Each Method? ($P$ is the size of population, $G$ number of global method iterations, $L$ number of local method iterations, $M$ number of vectors selected from population to be minimized. $k \in (0, 1)$ is percentage of population to be minimized (depending of the way of selection).)

| method | number of LS solutions |
|---|---|
| LR | interval size / step |
| NEWUOA | $L$ |
| DE | $P + G$ |
| GA | $P + PG$ |
| DEMIN1 | $P + G + L$ |
| DEMIN2 | $P + GL_2 + L_1$ |
| DEMIN3 | $P(1 + kL_3) + GL_2 + L_1$ |
| GAMIN1 | $P + PG + L$ |
| GAMIN2 | $P + G(P + (P/2)L_2 + L_1)$ |
| GDMIN | $P + ML_2 + L_1$ |

Figure 3: Computation Time for All *MIN Methods (The dots denote the run of minimal successful configuration, the bars minimal reliable successful configuration.)




## ACKNOWLEDGEMENT

Computational resources were provided by the CESNET LM2015042 and the CERIT Scientific Cloud LM2015085, provided under the programme "Projects

of Projects of Large Research, Development, and Innovations Infrastructures".

## REFERENCES

Becke A.D., 1993. *Density-functional thermochemistry. III. The role of exact exchange. The Journal of Chemical Physics*, 98, no. 7, 5648.

Bultinck P.; Vanholme R.; Popelier P.L.A.; De Proft F.; and Geerlings P., 2004. *High-speed calculation of AIM charges through the electronegativity equalization method. Journal of Physical Chemistry A*, 108, no. 46, 10359–10366.

Burkardt J., 2004. *LATIN_RANDOM.* URL `http://people.sc.fsu.edu/~jburkardt/cpp_src/latin_random/latin_random.html`.

Chaves J.; Barroso J.M.; Bultinck P.; and Carbó-Dorca R., 2006. *Toward an alternative hardness kernel matrix structure in the Electronegativity Equalization Method (EEM). Journal of Chemical Information and Modeling*, 46, no. 4, 1657–1665.

Geidl S.; Bouchal T.; Raček T.; Svobodová Vařeková R.; Hejret V.; Křenek A.; Abagyan R.; and Koča J., 2015. *High-quality and universal empirical atomic charges for chemoinformatics applications. Journal of Cheminformatics*, 7, no. 1, 1–10.

Gupta V.P., 2005. *Principles and applications of quantum chemistry.* Academic Press. ISBN 9780128034781.

Ionescu C.M.; Geidl S.; Svobodová Vařeková R.; and Koča J., 2013. *Rapid calculation of accurate atomic charges for proteins via the electronegativity equalization method. Journal of Chemical Information and Modeling*, 53, no. 10, 2548–2558.

Jensen F., 2007. *Introduction to computational chemistry.* John Wiley & Sons Ltd, Great Britain, 2nd ed. ISBN 9780470011867.

Menegon G.; Shimizu K.; Farah J.P.S.; Dias L.G.; and Chaimovich H., 2002. *Parameterization of the electronegativity equalization method based on the charge model 1. Physical Chemistry Chemical Physics*, 4, no. 24, 5933–5936.

Mortier W.J.; Ghosh S.K.; and Shankar S., 1986. *Electronegativity-equalization method for the calculation of atomic charges in molecules. Journal of the American Chemical Society*, 108, no. 15, 4315–4320.

Nelder J.A. and Mead R., 1965. *A Simplex Method for Function Minimization. The Computer Journal*, 7, no. 4, 308–313.

Ouyang Y.; Ye F.; and Liang Y., 2009. *A modified electronegativity equalization method for fast and accurate calculation of atomic charges in large biological molecules. Physical Chemistry Chemical Physics*, 11, no. 29, 6082.

Park H.; Lee J.; and Lee S., 2006. *Critical assessment of the automated AutoDock as a new docking tool for virtual screening. Proteins*, 65, no. 3, 549–54.

Powell M., 2004a. *Least Frobenius norm updating of quadratic models that satisfy interpolation conditions. Mathematical Programming*, Series B 1, no. 1, 183–215.

Powell M., 2004b. *NEWUOA software.* URL `http://mat.uc.pt/~zhang/software.html#powell_software`.

Raček T.; Pazúriková J.; Svobodová Vaeková R.; Geidl S.; Falginella F.; Horský V.; Hejret V.; and Koča J., 2016. *NEEMP - software for validation, accurate calculation and fast parametrization of EEM charges. Journal of Cheminformatics*, accepted with minor revisions. URL `http://www.fi.muni.cz/~xpazurik/submitted`.

Reed A.E.; Weinstock R.B.; and Weinhold F., 1985. *Natural population analysis. The Journal of Chemical Physics*, 83, no. 2, 735.

Rong C.; Lian S.; Yin D.; Zhong A.; Zhang R.; and Liu S., 2007. *Effective simulation of biological systems: Choice of density functional and basis set for heme-containing complexes. Chemical Physics Letters*, 434, no. 1, 149–154.

Sanderson R.T., 1951. *An Interpretation of Bond Lengths and a Classification of Bonds. Science (New York, NY)*, 114, no. 2973, 670–2.

Storn R. and Price K., 1997. *Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. Journal of Global Optimization*, 11, no. 4, 341–359.

Tervo A.; Rönkkö T.; Nyrönen T.; and Poso A., 2005. *BRUTUS: optimization of a grid-based similarity function for rigid-body molecular superposition. 1. Alignment and virtual screening applications. Journal of Medicinal Chemistry*, 48, no. 12, 4076–86.

Vařeková R.; Geidl S.; Ionescu C.M.; Skřehota O.; Bouchal T.; Sehnal D.; Abagyan R.; and Koča J., 2013. *Predicting p Ka values from EEM atomic charges. Journal of Cheminformatics*, 5, no. 1, 18.