
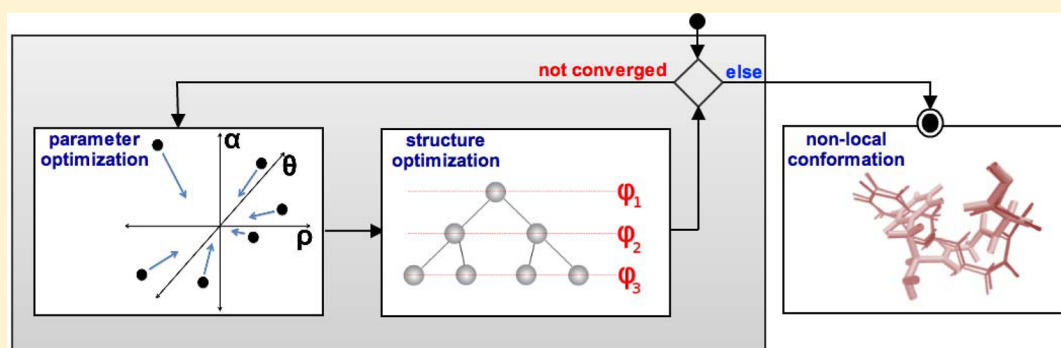


Hybrid Metaheuristic Approach for Nonlocal Optimization of Molecular Systems

Thomas Dresselhaus, Jack Yang, Sadhana Kumbhar, and Mark P. Waller*

Organisch-Chemisches Institut, Westfälische Wilhelms-Universität Münster, Corrensstraße 40, 48149 Münster, Germany

 Supporting Information



ABSTRACT: Accurate modeling of molecular systems requires a good knowledge of the structure; therefore, conformation searching/optimization is a routine necessity in computational chemistry. Here we present a hybrid metaheuristic optimization (HMO) algorithm, which combines ant colony optimization (ACO) and particle swarm optimization (PSO) for the optimization of molecular systems. The HMO implementation meta-optimizes the parameters of the ACO algorithm on-the-fly by the coupled PSO algorithm. The ACO parameters were optimized on a set of small difluorinated polyenes where the parameters exhibited small variance as the size of the molecule increased. The HMO algorithm was validated by searching for the closed form of around 100 molecular balances. Compared to the gradient-based optimized molecular balance structures, the HMO algorithm was able to find low-energy conformations with a 87% success rate. Finally, the computational effort for generating low-energy conformation(s) for the phenylalanyl-glycyl-glycine tripeptide was approximately 60 CPU hours with the ACO algorithm, in comparison to 4 CPU years required for an exhaustive brute-force calculation.

INTRODUCTION

Optimization is one of the challenging areas in the field of computational chemistry. In principle, finding the minima of a high dimensional function is a purely mathematical problem. A number of efficient methods have been developed,¹ and in order to quickly reach the extrema, most of them are based on the gradient or on higher derivatives. Examples include the steepest descent algorithm and the quasi-Newton method when the analytical forms of the underlying functions are accessible. Typically these methods can only find local minima, meaning that they are not designed to overcome barriers on the hyperspace. A carefully selected starting point is therefore required for reaching a sensible outcome. In contrast, nonlocal optimization techniques can find optimum solutions independent from the starting point or are at least not hindered completely by barriers.²

Indeed, for every task in computational chemistry which cannot be treated using a very simplified model (e.g., Ising model), a starting structure for a given molecule is needed on which calculations can be performed.^{2–7} In the case of structural optimization, starting geometries might be obtained from experimentally determined crystal structures or built upon chemical intuitions for simple molecular systems. However, this

becomes overwhelmingly difficult when the structures cannot be predetermined experimentally, and the system scale/flexibility are too large to be comprehended based on intuitions.

Calculating properties from a single structure is an approximation, and one would ideally survey the molecular conformational space in order to reach a thorough understanding of the molecular system. The Basin-Hopping Monte Carlo approach^{8,9} has proven very successful in many areas, such as protein folding and structure predictions for molecular clusters.¹⁰ Molecular dynamics (MD) simulation is another way to achieve this by generating ensembles of structures, and such simulations are typically run for long periods of time in order to sample many possible conformations. For example, a set of low-energy molecular structures for the FG (Phenylalanyl-Glycyl-Glycyl) tripeptide used in quantum chemical benchmark studies were generated with this approach.¹¹ MD simulations are often performed using parametrized force-fields; however, as novel materials emerge, the transferability of existing force-fields are questionable. Therefore, optimization of force-field

Received: December 10, 2012

Published: February 20, 2013



parameters is often required before performing structural optimization/prediction.¹²

An alternative method to MD is to treat such problems with metaheuristic algorithms. These algorithms are used to find answers where there is heuristic information to guide the search, and a brute-force approach is out of the questions due to the enormous size of the search space.¹³ A number of problems in computational chemistry have already been treated with metaheuristic algorithms, such as simulated annealing,^{14,15} genetic algorithms,^{2,12,16–21} discrete lattice search,²² and Tabu search.^{23–25} Examples for the use of other metaheuristic algorithms can be found more rarely, such as the Ant Colony Optimization (ACO)^{26,27} and Particle Swarm Optimization (PSO),²⁸ are two feasible methods for solving the nonlocal optimization problems.

A typical character of metaheuristic algorithms is that they also depend on several parameters. These parameters are often coupled with each other and can influence the efficiency as well as the quality of the results obtained from the algorithms. A set of parameters which is good for one metaheuristic optimization might not be suitable for other problems; this problem of parameter transferability is similar to the problem with force-fields. This makes it almost impossible to quickly find the ideal parameters in a deterministic manner for a given problem/algorithm combination.²⁹ This problem description is one that can be treated with metaheuristic algorithms itself. To optimize parameters of an algorithm with other algorithms has shown to work well in many cases^{30,31} and is the recommended way to obtain good parameters.³² Herein, we explored the performance of a *hybrid metaheuristic* approach for on-the-fly parameter meta-optimization for nonlocal optimization of molecular systems. By using structural optimization as a prototypical example, we showed that the ACO algorithm could be optimized on-the-fly by performing PSO optimizations to fine-tune the ACO parameters. We would like to point out that metaheuristic algorithms are typically not guaranteed to find the global minimum.⁵ In the following discussions, the word 'global best' only refers to the best solution found by a particular metaheuristic algorithm at a given point of the optimization cycle, which may or may not be the true global minimum conformation.

METHOD

Ant Colony Optimization. The ACO algorithm was invented by Dorigo in 1992.³³ This algorithm has been frequently applied to various mathematical problems in many different implementations.^{34,35} In computational chemistry, however, it had been rarely applied. A few possible examples include docking studies³⁶ and conformational searching.⁵

The ACO algorithm is inspired by the foraging behavior of ants in nature. To achieve this, ants do not walk around randomly, but they tend to follow the trails of pheromones which have been laid by prior ants. The later ants will also deploy pheromones while exploring new paths themselves. At the same time, the deposited pheromones will evaporate at a certain rate. The shortest route will have more pheromones built up over time, since the evaporated pheromones will quickly be redeposited by ants traveling (back) on the same route. As a result, at the junction points, more ants will choose the shorter one of the two possible routes, emphasizing this route for the following ants. Long routes that have been traveled some time ago, but are not any more, will lose the pheromones over time. Eventually pheromones can only be

found on the shortest route, and all ants will predominantly be traveling via the shortest route. This is the positive feedback learning mechanism on how the algorithm is able to find optima in a diverse search space.

There are some differences between what a basic ant colony algorithm provides and what is actually needed to solve the problem of nonlocal structural optimization, we now discuss specific implementation details for our implementation of the ACO algorithm.

Initially, ants generate a set of random solutions (i.e., routes), among which pheromone levels are evenly distributed. From here, the probability to choose a certain value j at a decision point i , i.e. the branching point at two joining routes, could be determined as

$$p_{ij} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_j \tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}} \quad (1)$$

where τ is the pheromone level, and η is a heuristic information, called visibility. The parameters α and β were adjustable to dictate whether a specific search is to be more *explorative* or *exploitive*. Determining the visibility would require a calculation of the local information about each junction point, for example, the energy value for a set of atoms that defines a particular dihedral angle in a given molecular system, which cannot be easily obtained in our current implementation. Therefore, the visibility term was not included, and consequently the β parameter was not needed. After all ants have tested a possible solution, the pheromone levels are then updated by scaling down the old pheromone level using a resistance parameter [In the literature this parameter is often expressed as the evaporation rate: $\rho_{\text{evaporation rate}} = 1 - \rho_{\text{pheromone resistance}}$] ρ and the information ants gathered within the last step of the algorithm, expressed as the evaluation function g for each ant s :

$$\tau_{ij}^{\text{new}} = \rho \cdot \tau_{ij}^{\text{old}} + \sum_s g_s(i, j) \quad (2)$$

The evaluation function is based on a fitness f , which is a measurement of the quality of a solution. Often the evaluation function is chosen simply as

$$g_s(i, j) = \begin{cases} f(s) & \text{if the ant } s \text{ has chosen value } j \text{ at decision point } i \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

After the pheromone update, a new metacycle starts, and all ants create and test new solutions. The algorithm is converged if pheromones are only left on one certain decision value for each decision point within a defined threshold. A known problem of the general ACO algorithm is that the ants tend to trap themselves in local extrema very quickly within the first cycles. Because the potential energy surface of a molecular system can be expected to have a high number of local minima, this is a problem that needs to be avoided. Here, we adopted the solutions by Stützle and Hoos³⁷ to solve such a problem. In their max-min ant system the pheromone level has a lower (and upper) boundary, so that even if a very good solution was frequently found by the ants, there would still be a finite chance that other solutions might be tested. The maximum pheromone level can be defined as

$$\tau_{\text{max}} = \frac{1}{\rho \cdot f(s^{\text{gb}})} \quad (4)$$

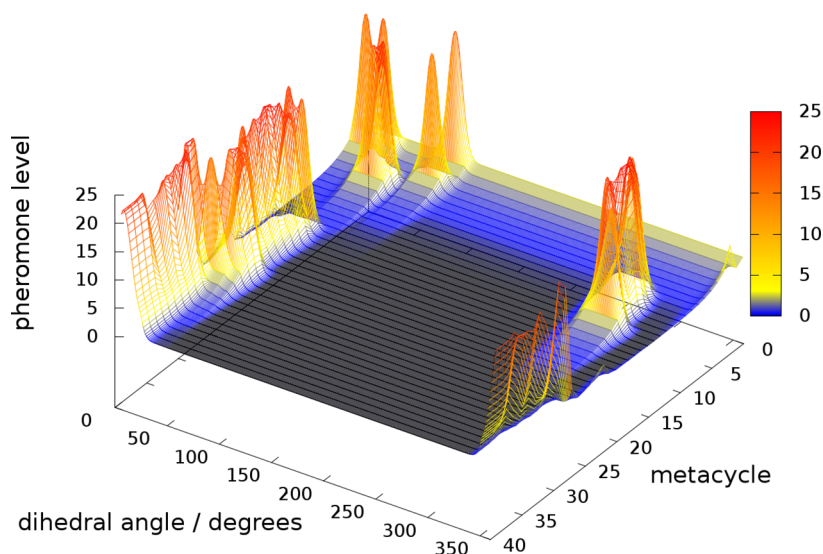


Figure 1. A 3D plot of a typical pheromone distribution on a certain dihedral angle from a calculation on difluorooctatetraene. Several features can be seen, for example, an initial equal pheromone distribution (yellow line at metacycle 0), the Gaussian shape of newly added pheromones, and periodicity of the search space. In early stages several different solutions were explored (new peaks at different angles), but pheromones also disappeared again quickly. The Gaussian peaks also move slowly toward better angles. Convergence is reached when only one peak around the optimal value (0 degree) remains. The color bar corresponds to the pheromone level.

which depends on the fitness f of the global best solution found so far. A parameter p_{best} can be set to determine the highest allowed probability for an ant to choose exactly the global best solution. Hence, the minimal allowed pheromone level can be evaluated as

$$\tau_{\min} = \frac{\tau_{\max} \cdot (1 - \sqrt[p_{\text{best}}]{p_{\text{best}}})}{(\bar{n} - 1) \cdot p_{\text{best}}} \quad (5)$$

where \bar{n} is the averaged number of allowed values per decision point. The $\tau_{\min/\text{mix}}$ system ensures there would always be a finite chance that nonglobal best solutions would be tested during the optimization. As shown below, the convergence criteria must be defined with respect to these implementation details.

As suggested by Stützle and Hoos when using the max-min ant system, an elitist strategy is applied for the pheromone update.³⁷ In this implementation, only the best ant within one metacycle deposits pheromones to avoid further exploration of regions around unfavorable solutions. Also, according to their suggestions, a global best update rule was mixed in. Pheromones were deposited not only purely according to the solution of the best ant within one metacycle but also with respect to the overall best solution found by the algorithm so far. A smoothly adjustable parameter (global best percentage, θ) was implemented to achieve the mixing of purely iteration best and purely global best solutions.

Typically the ACO algorithm is set up on discrete search spaces, but the potential energy surface of a molecular system is continuous. This problem was treated analogously to the paper from Daeyaert et al.⁵ by creating continuous solutions based on the probability distribution using the roulette wheel selection algorithm. Furthermore, generating solutions in a continuous search space should be reflected in the pheromone levels. Therefore, instead of depositing pheromones *exactly* on the value j at decision point i , Daeyaert et al. used a Gaussian distribution for the pheromone update, which is centered on the exact solution value. The evaluation function is then

$$g_s(i, j) = \frac{f(s)}{\sqrt{2\pi}\sigma} \cdot \exp\left\{-0.5 \left[\frac{d(j, s(i))}{\sigma}\right]^2\right\} \quad (6)$$

with the distance d as the center of the field of the possible value j to the value that an ant s has chosen at decision point i . The standard deviation σ , called trail width, is an adjustable parameter. Precautions must be taken when choosing a Gaussian evaluation function, that one must take into account possible periodicity in the search space. The advantage of such an approach is that existing pheromone trails can be slowly shifted and thus optimized, without the need to explore a completely new route that improved only slightly. A disadvantage is that the algorithm might converge much slower, because fluctuations become very likely with the Gaussian pheromone distribution. In a δ -shaped pheromone update rule [see eq 3] such fluctuations are not possible. In Figure 1 the pheromone distribution is visualized for one search space dimension over a complete calculation.

The fitness function was also implemented in an analogous way to Daeyaert et al.⁵ with some adjustments. In the implementation of Daeyaert et al., the energy of a conformation is determined via a force field coupled to the algorithm. This reduces the transferability of the fitness function, where the energy differences determined by one force field could differ drastically from those determined by another method. Moreover, with methods that produce energy values containing atomic energies, the difference in energies for different conformations can be expected to be very small in contrast to the energy value itself. Certainly, these effects should not influence the overall behavior of the algorithm. A normalized exponential fitness function, similar to the one suggested by Daeyaert et al., was implemented in our algorithm to solve the problem. The fitness function has the following form

$$f(s) = \exp\left[-\gamma \frac{E_s - E_{\text{best}}}{\bar{E} - E_{\text{best}}}\right] \quad (7)$$

which takes into account the average (\bar{E}) and best so far encountered (E_{best}) energy to determine the fitness of ant s . The learning rate decay constant γ is an adjustable parameter.

Taking into account the minimum pheromone level and the non- δ -shaped pheromone update rule, fluctuations must be expected even if the algorithm runs infinitely long or has already found the best solution. Thus one cannot set the convergence criteria to be too strict. Considering the Gaussian profile of pheromone distribution, the convergence criteria were set in a similar fashion. In a Gaussian distribution, 95.45% of the area can be found within 2σ of the center. If one allows an extra σ fluctuation in the pheromone distribution, then a plausible convergence can be met if the integrated pheromone level in the interval of $[-3\sigma, 3\sigma]$ is greater than 95.45% or mathematically expressed as

$$\frac{\sum_{j=s_{ib}(i)-3\sigma}^{s_{ib}(i)+3\sigma} \tau_{ij}^{\alpha} - \tau_{\min}^{\alpha}}{\sum_j \tau_{ij}^{\alpha} - \tau_{\min}^{\alpha}} \geq 95.45\% \quad (8)$$

The convergence criterion is visualized in Figure 2. Additionally, the limits of the sum in the numerator of eq 8 needs to be

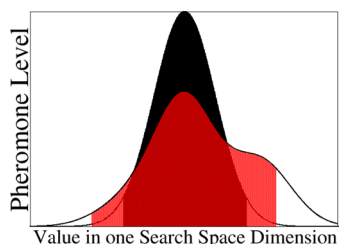


Figure 2. A scheme illustrating the convergence criterion of the ant colony optimization. The curve containing the black area is a normal Gaussian distribution. Within the shown area (2σ) of the Gaussian distribution 95.45% of the area below it can be found (black area). The algorithm is converged in a certain dimension if the red area (3σ) of the pheromone distribution is larger than the black area.

set with respect to the limits or periodicity of the search space dimension i . In our implementation, the center of the assumed Gaussian distribution is set to the solution value for dimension i of the best ant of the last step of the algorithm $s_{ib}(i)$. Satisfaction of the convergence criteria means that the algorithm has not tested any other good conformations that are different from the current best one found for a large number of metacycles, so one can assume that the algorithm will most likely not find any other better solutions within a reasonable number of additional metacycles.

Particle Swarm Optimization. The PSO algorithm, introduced by Kennedy and Eberhart,³⁸ has seen applications in computational chemistry combined with MD simulations.²⁸ At initialization, a certain number of particles are placed at random positions in the search space. The solutions representing these positions are then tested and evaluated. The algorithm then modifies the tested solutions, as opposed to other metaheuristic algorithms that explore the search space by designing completely new solutions each time, for example, the ACO above. Each particle is then accelerated with the acceleration rate determined by the results from the evaluated solutions and a random component. After moving the particles, the new resulting solutions are tested. The exact acceleration schemes are implementation specific. Some common features include the following:³⁹ First, the direction along which the

particles are accelerated is purely determined by the positions of other particles and the amount of acceleration is chosen randomly or scaled by a random number. Typically, particles are accelerated toward the positions of the overall best and a neighborhood best particle. The neighborhood can also be defined in different ways, for example, by partitioning the search space into different areas or with a social scheme, meaning that particles have fixed neighbors independent from their actual position. Second, search spaces covered by pulling particles toward either locally or globally good particles are explored well, while regions which have shown to be unattractive are not thoroughly explored. Such a problem might be overcome by running the algorithm for a few times and comparing the results from independent runs. Finally, PSOs are typically defined to be converged when all particles are at approximately the same position within the search space.

Similar to the ACO, N number of initial particles are positioned randomly in the search space. The fitness function $f(s)$ for each particle will then be evaluated, using the same fitness function as in the ant colony algorithm, followed by accelerating the particles toward the positions of global best and iteration best particles, so that the velocity results as

$$v_s(i) = c \cdot v_{s,\text{old}}(i) + a_s(i) \cdot dt \quad (9)$$

with an arbitrary time step dt (for simplicity it is set to 1) and the acceleration defined by

$$a_s(i) = a_{\text{neighbor}}(s, i) + a_{\text{best}}(s, i) \quad (10)$$

in which

1. The first term in the velocity equation can be seen as a friction term. It slows down the particles by a constant factor, which is an adjustable parameter. Additionally, if a particle gains a velocity higher than 20% of the search space size per metacycle in a certain dimension, then it is slowed down to 10% of the search space size per metacycle. Particles which move out of a search space border in a nonperiodic search space dimension are set to the limiting value. The algorithm is taken to be converged if the standard deviation of the particles' positions is lower than 1% of the search space size in each dimension. Without slowing down the particles their velocity increased too quickly so that they tended to hit the borders of the search space, or in case of periodic search space, the particles crossed the whole search space within a few steps of the algorithm. Without keeping the old velocity ($c = 0$) they tended to approach only the best solution found at the very beginning of the algorithm without searching the area around it, thus the algorithm quickly stops optimizing and converges to bad solutions.

2. The first component of the acceleration points into the direction of the best particle in its neighborhood. The neighborhood was set up in a social ring style, i.e. it is defined by the particle indexes: the m th particle is in direct neighbor with the $m \pm 1$ th particles (hence the first and last particles are also direct neighbors). Therefore each particle is influenced by two direct neighbors. The acceleration is scaled by the distance d of the particle to the better of its neighbors, by the granularity ρ of the search space, and by a random number φ_0 between zero and two. The distance d takes into account the possibility of a periodic search space.

$$a_{\text{neighbor}}(s, i) = \varphi_0 \cdot \rho \cdot d(s_{\text{neighbor}}(i), s(i)) \quad (11)$$

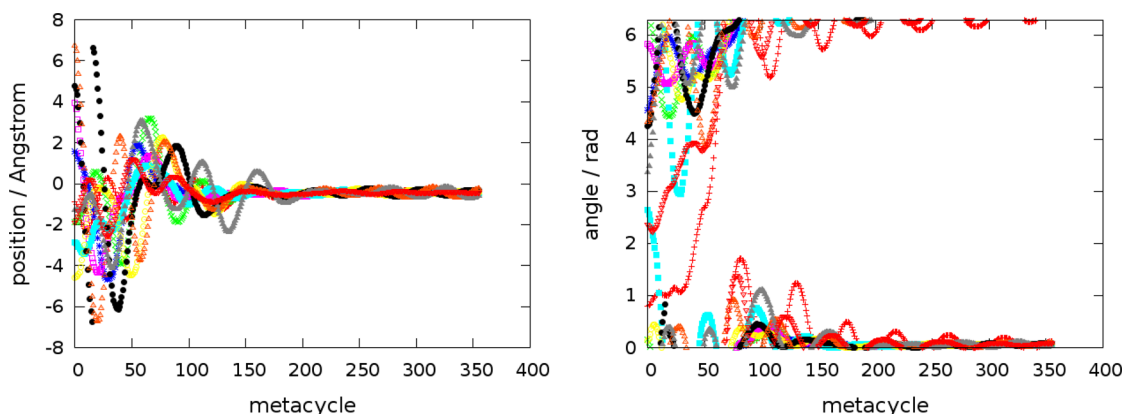
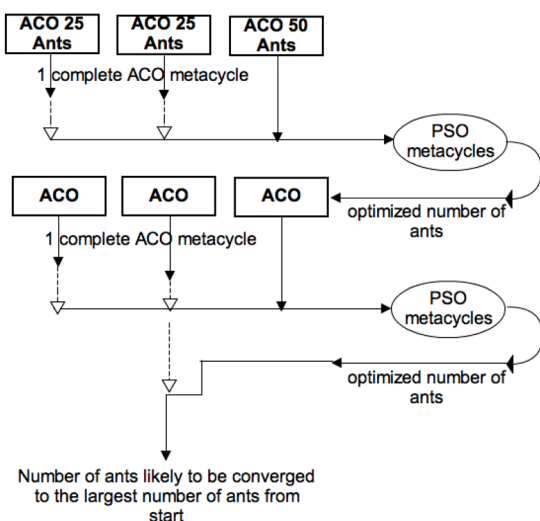


Figure 3. Illustration of the typical behavior of particles in a nonperiodic (left) and periodic (right) search space. The PSO was performed in the search for the optimum conformation of a borane-ammonia dimer, in which the borane was kept fixed in space. The search was performed over a six dimensional search space [three for Cartesian coordinates (nonperiodic search space) and three rotational angles (periodic search space)]. In both plots, the equilibrium Cartesian and rotation coordinates for ammonia were set to zero, and the deviation of a Cartesian coordinate and a rotation coordinate from zero is plotted for different particles as a function of the metacycle number. The random starting points (metacycle 0) as well as an oscillating behavior, which is typical in a particle swarm optimization, can be seen while approaching the ideal conformation.

Synchronous Hybrid Metaheuristic Optimization



Asynchronous Hybrid Metaheuristic Optimization

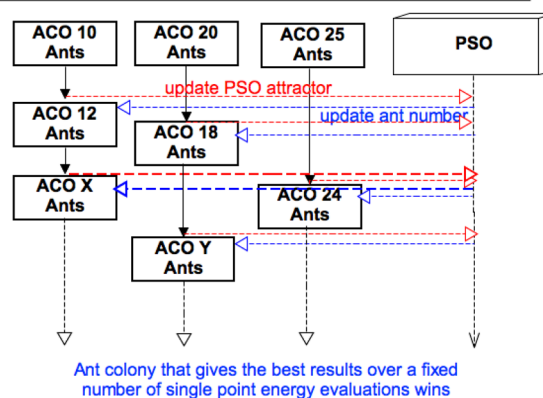


Figure 4. Algorithm flowchart for a synchronous and asynchronous HMO with PSO optimizing the parameters for the underlying ACO. The length of the vertical solid arrow indicates the number of single point energy evaluation each ACO has to perform per metacycle, which is equal to the number of ants for a given ant colony. In a synchronous implementation, a PSO cycle is only allowed to start if each underlying ACO has completed a full metacycle, thus at a given point of algorithm run, there will be an equal number of metacycles completed by each ant colony. In an asynchronous scheme, a PSO is running in conjunction with several underlying ACOs, and the parameters were updated once when all ACOs have completed one single point energy evaluation. In this case, each ant colony would have completed the same number of single point energy evaluations.

3. The second component of the acceleration is similar, but using the distance to the best solution of the whole swarm within one step of the algorithm (iteration best, ib) as well as to the overall best solution found so far (global best, gb), controlled by the global best percentage θ like in the ant colony optimization algorithm.

$$a_{\text{best}}(s, i) = \varphi_1 \cdot \rho \cdot [\theta \cdot d(s_{ib}(i), s(i)) + (1 - \theta) \cdot d(s_{gb}(i), s(i))] \quad (12)$$

Typical behaviors of particles in both nonperiodic and periodic search spaces are illustrated in Figure 3.

Hybrid Metaheuristic Optimization (HMO). For optimal performance of the ACO algorithm 6 parameters are needed to be optimized in the parameter space. In comparison, the PSO

algorithm is less demanding and requires only three adjustable parameters ($\{c, \theta, \gamma\}$). In principle, one could run several independent ACO and/or PSO optimizations with different parameters and then select the most appropriate ones. However, using a meta-optimization based approach, one may apply the metaheuristic algorithm with less adjustable parameters (in this case, PSO) to optimize on-the-fly the parameter space for another metaheuristic algorithm with more parameters (such as ACO) to reach the optimum solutions.

Several independent ACO runs, the parameters of which are to be optimized, were started simultaneously at the beginning of the HMO. This gave a collection of random starting parameter sets $\{\mathcal{P}_i^{\text{ACO}}\}$ to be optimized by the coupled PSO, where \mathcal{P}_i is an analogy to real-space particle position in the parameter space. The algorithms were implemented in a way,

that no difference in optimizing a set of spatial coordinates (structural optimization) and algorithm parameters (parameter optimization) exists. We note that it is not ideal to optimize the parameters on-the-fly with the ACO algorithm, since a new set of randomly generated parameters will most likely interfere with the running algorithm and prevent it from running properly. Furthermore, as the PSO can adjust parameters more smoothly, it allows the parameters for ACO to be optimized in a more systematic manner.

HMO requires an algorithm that allows its parameters to be changed while running. One of the parameters for ACO, i.e. the number of ants (N_{ants}), can only be optimized on-the-fly with precautions. In general, HMO can be implemented in two different fashions, which will be denoted here as synchronous and asynchronous HMO, see Figure 4. In a synchronous approach, a PSO is only allowed to perform a meta-optimization of the ACO parameters after all independent ACOs have performed one metacycle. It is termed synchronous as all ACO would have performed the same number of metacycles at the point when a PSO performs the parameter optimization. This introduces an advantage to the colonies with higher number of ants, since they would have performed more energy evaluations than the other colonies with fewer ants, and hence are more likely to generate lower energy conformations. In this case, the colony with the largest N_{ants} value would tend to have a better fitness, resulting in the underlying PSO to have a tendency of converging N_{ants} toward N_{max} . What is missing in such an implementation is that the efficiency (in terms of number of single point energy evaluations required for an ant colony to reach the best result) cannot be properly accounted for, when optimizing the N_{ants} parameter. In this regard, the synchronous version of HMO should not be used to optimize N_{ants} .

An alternative solution to overcome such a problem is to adopt an asynchronous implementation of HMO. In this implementation, the PSO will perform *one* parameter optimization step for *one* single point energy evaluation per ant colony, i.e. the PSO will perform a complete metacycle when all the ant colonies have performed exactly one single point energy evaluation. Let us denote N_{PSO} as the number of PSO optimization metacycles performed and $[N_1^{(i)}, N_2^{(i)}, \dots, N_j^{(i)}]$ be the set collecting the number of ants for the i th colony updated until the current (j th) metacycle, then the i th ant colony is only allowed to perform the next metacycle if $N_{\text{PSO}} = \sum_j N_j^{(i)}$. Furthermore, the i th ant colony will start a new metacycle with the optimized $N_{j+1}^{(i)}$ returned from the current PSO optimized value. It is then apparent that the fitness of N_{ants} can be easily judged based on the N_{PSO} requested so far. Also note that when optimizing the parameters with PSO, the i th attractor for particles is updated once the i th ACO has finished the j th metacycle and will remain fixed for the subsequent $N_{j+1}^{(i)}$ metacycles of PSO optimizations. This means that the attractors for certain particles might stay the same for a large number of PSO cycles, which will lead to an undesired behavior of the particles that should be rectified.

In the asynchronous PSO run, it can be seen that the number of PSO cycles are significantly larger than the case of a synchronous run. If the granularity (ρ) for the underlying PSO was kept fixed as a normal PSO (which will be termed as “non-finegrained”), an asynchronous PSO run would allow the particles to traverse the entire search space multiple times, while the attraction points stayed the same. This would result in a low probability of exploring the regions where it was

supposed to, namely the region in the search space which contains the global, iteration, and neighboring bests, see Figure 5. A solution to overcome such a problem is to finegrain the

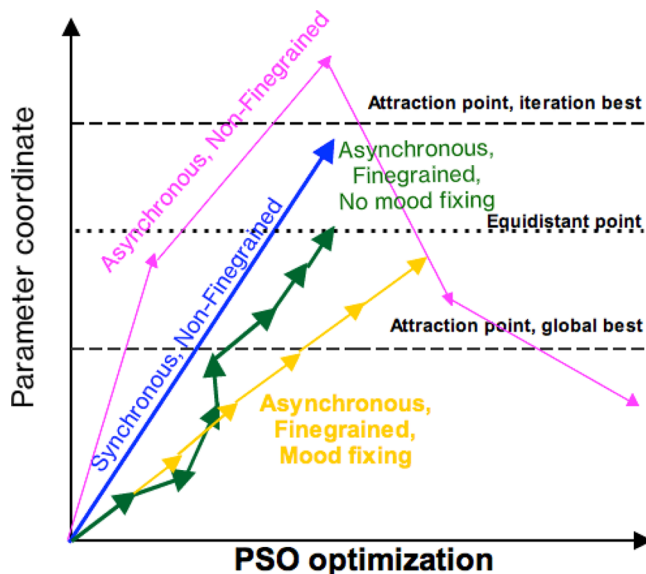


Figure 5. Schematic illustration of particle behaviors in different implementation of PSO for parameter optimizations.

PSO search by reducing the search space granularity. In our implementation, the search space granularity is reduced to $\rho / \langle N_{\text{ants}} \rangle$, where $\langle N_{\text{ants}} \rangle$ is the expectation value of N_{ants} across multiple colonies. The expectation value can be predetermined from the search space limit $[N_{\text{min}}, N_{\text{max}}]$ as

$$\langle N_{\text{ants}} \rangle = \frac{N_{\text{min}} + N_{\text{max}}}{2} \quad (13)$$

As shown in Figure 5, the particles could adopt finer exploration steps in the search space for a finegrained asynchronous search, which could better reproduce the particle behavior in a synchronous, nonfinegrained search.

Nevertheless, when the particle attractors were kept fixed for a large number of PSO steps, the random numbers determining the particle accelerations would tend to average out, making the particles accelerate primarily toward the equidistant point between the global best and iteration best attractors (Figure 5). This behavior can be easily illustrated in a random walk. Figure 6 demonstrated a numerical simulation of a one-dimensional random walk experiment, with a particle started at position $y_0 = 3$, and with consecutive position updates as $y_i = y_{i-1} + c_i \rho$, with ρ to be a vanishing small constant and c_i to be a random number from a normal distribution centered around 0, updated at *each* step. It can be seen that in over 50,000 random walk steps, the particle deviated insignificantly from y_0 (red line). In terms of PSO, this means that the search space defined between two attractors cannot be well explored. To circumvent this problem, we introduced a “mood parameter” for HMO algorithm, which defines the number of PSO steps during which the random numbers generated for determining the particle accelerations will be kept fixed. This enables the particles to better explore the search space defined by multiple attractors (Figure 5). This can be clearly demonstrated from Figure 6 (blue curve). By keeping the c_i value fixed for 300 random walk steps (mood=300) before it was allowed to pick a new random value, the particle could traverse a large region

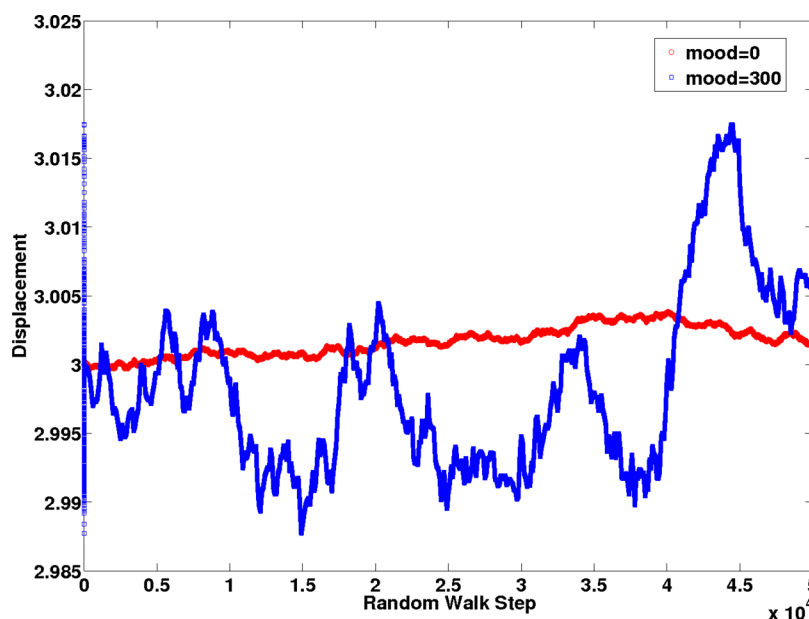


Figure 6. Numerical simulation of a one-dimensional random walk between two attractors. Without a mood parameter, the particle tends to move along a fixed direction for a large number of steps, whereas introducing the mood parameters brings in more fluctuations in the displacement, which could lead to better exploration of the space between the two attractors.

across the search space, allowing the search space to be better explored for optimum solutions. For testing purposes, the mood parameter can be changed based on user specification. Nevertheless, it is recommended to set the mood parameter to $\langle N_{\text{ants}} \rangle$ for closely reproducing the results from synchronous, nonfinegrained PSO results.

Computational Details. The HMO searches were performed using JACOB,⁴⁰ which is our own framework for computational chemistry, based on enterprise software design practices. The PM6-DH+⁴¹ semiempirical gas-phase single-point energies were computed using MOPAC2012.⁴² The default optimizer in MOPAC2012 is Baker's EigenFollowing (EF) procedure.^{43–45} All conformations obtained from HMO were further reoptimized with the PM6-DH+ method within MOPAC2012. Starting structures for all molecular balances were constructed using GaussView.⁴⁶ The Genconf (C++) code⁴⁷ was used to perform the brute-force calculations of the FGG tripeptide. The algorithm traverses a flexible molecule to determine the rotatable dihedral angles and generates new conformations by rotating each rotatable bond sequentially by a fixed increment until all possible conformations have been generated. Generated structures are checked for steric clashes, and if present the conformation is disregarded.

RESULTS

In this section, we present the results for the metaheuristic nonlocal conformation searching algorithms. The ACO algorithm was used to optimize molecules with flexible conformations by rotating a set of dihedral angles. Therefore, the primary goal of the ACO is to optimize these sets of dihedral angles (i.e., solutions) in searching for low-energy conformations. We begin by discussing the optimized ACO parameters with the HMO approach using a set of small difluorinated polyenes. The ACO algorithm will be further validated by optimizing a collection of molecular balances. These balances possess well-defined *closed* conformation, thus the ACO algorithm can be validated by comparing the number

of positive hits against these known minima. Finally, an exhaustive brute-force optimization on the FGG tripeptide was performed to identify the global best conformation, and nonlocal minima on the FGG energy landscape were found using the ACO algorithm.

Training Set for HMO. Conjugated polyenes with one fluorine atom attached to the first and one attached to the last carbon atom of the chain were used for testing and training the ACO algorithm. Conformations with all double bonds in a *trans* conformation were chosen to be the reference conformations, since those minimize the steric repulsion between the two fluorine atoms, and would most likely be the global best conformation. Local minima can be expected to exist for conformations which have a small number of *cis*-configured double bonds. With increasing system size these solutions are expected to be found more frequently. The molecules tested are shown in Figure 7. The simplicity of these molecules means that parameter optimizations could be performed with relatively low computational costs required for each single-point calculation.

Twenty-five independent HMO runs were performed for each polyene to obtain averaged values for the optimized ACO parameters. The parameters for the PSO algorithms were fixed at the values shown in Table 1. Since the HMO algorithm performs parameter optimizations alongside the structure optimization, the optimized parameters would be sensible only if the algorithm is able to find good structures. This is demonstrated in Figure 8, where all the conformations produced from HMO (without further optimization) were compared to the all-*trans* conformation from the HMO runs. It is immediately apparent that more than half of the optimized conformations exhibit all-*trans* conformation, indicated by RMSD (root-mean-squared-deviation) values lower than 0.5 Å, calculated with quaternion based algorithm.⁴⁸ This validates the underlying ACO algorithm. The appearance of plateaus (e.g., at RMSD around 1 Å) is attributed to local minima which possess one double bond in the *cis* configuration.

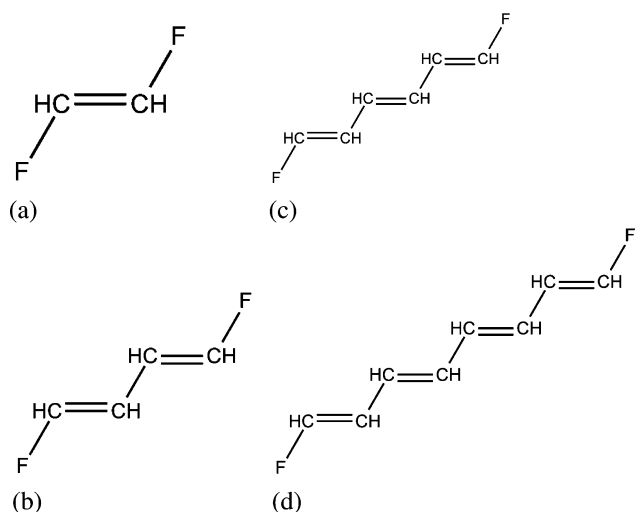


Figure 7. Structures of test molecules for intramolecular nonlocal optimization: (a) difluoroethene, (b) difluorobutadiene, (c) difluorohexatriene, and (d) difluorooctatetraene.

Table 1. Parameters Used in the PSO Algorithm for HMO

parameter (symbol)	value
number of particles	25
acceleration ratio	0.95
global best percentage (θ)	0.6
learning rate decay constant (γ)	5
granularity	0.01
mood	30

The optimized ACO parameters for each set are shown in Table 2. As the length of the polyene increases, most of the ACO parameters did not show sizable variations across the test set. This may be because the parameter optimization depends more on the nature of the underlying ACO to be optimized, and the molecules chosen for the training sets were very similar. The number of ants required for optimizing difluoroethene showed a 15% increase compared with the other three cases;

however, this test system with only one degree-of-freedom is most probably too small for a reliable treatment with the HMO algorithm. The essential message here is that rather than the user providing an educated guess for an appropriate set of ACO parameters, it can be determined by the algorithm itself based on the problem that it was required to solve.

Validation Set. Molecular balances are foldable molecules, which are good choices for validating conformation search algorithms because they tend to adopt (by design) either an *opened* or *closed* form, i.e. well-defined energetic minima, where the *closed* form maximizes the number of possible intramolecular interactions within a given molecule. Noncovalent interactions themselves are interesting because they play an important role in biology,⁴⁹ supramolecular chemistry,^{50,51} and material science,⁵² etc. The individual interactions are not particularly strong, but they can be combined to form a sizable interaction. The molecular balances chosen here have been previously studied experimentally, so their thermodynamic properties are fairly well understood.⁵¹ In particular, the *closed-form* gives a well-defined “global” minimum for the molecule. Therefore, we optimized a set of 102 molecular balances to be used as a validation set for the ACO algorithm. Table 3 provides an overview for the validation set.

The heat of formation and the geometry for the PM6-DH+ optimized *closed* form for each balance, using the EF based approach, were used as the reference points (denoted as $H_{f,EF/PM6-DH+}$), to which the ACO optimized structures would be subsequently compared. In total, 20 metaheuristic ACO runs for each molecular balance were performed, and the global best structure from each run was further optimized with the EF algorithm using the PM6-DH+ method (denoted as $H_{f,ACO/PM6-DH+}$). Many of the molecular balances studied involve π - π interactions between aromatic rings, which are absent from the difluorinated polyene training sets. Therefore, it may be necessary to obtain parameters that are more specific to the set molecular balances. We chose one molecular balance to perform 20 runs of HMO parameter optimizations, and the optimized parameters are given in Table 4. Despite the large differences between the molecular balance and the fluorinated polyenes, only slight differences in the optimized parameters

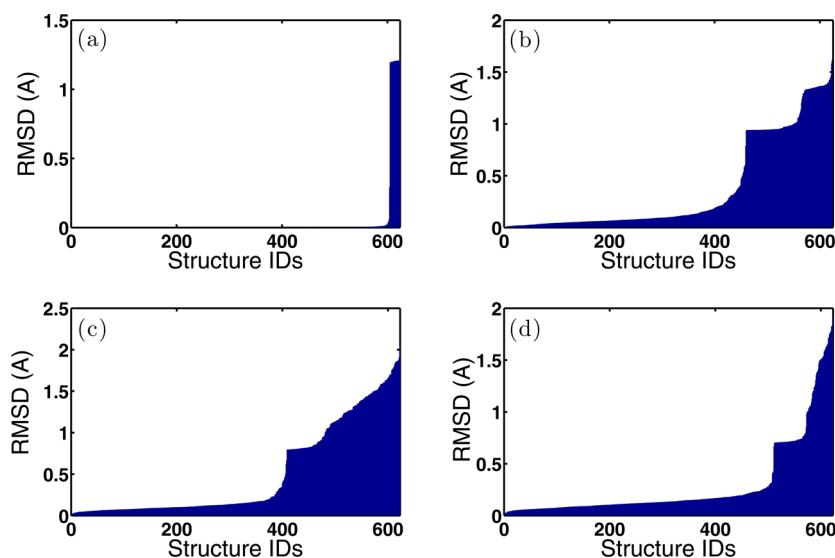
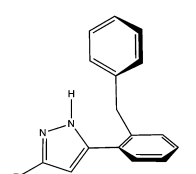
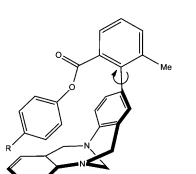
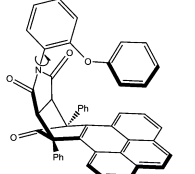
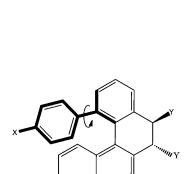
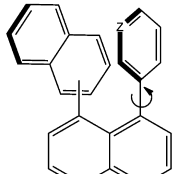
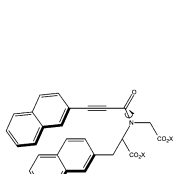
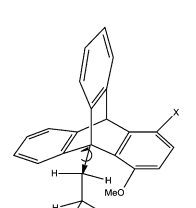
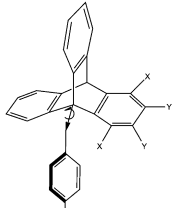
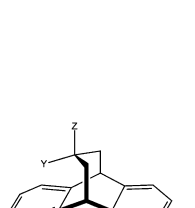


Figure 8. Analysis for the global best polyene conformations from the HMO search: (a) difluoroethene, (b) difluorobutadiene, (c) difluorohexatriene, and (d) difluorooctatetraene. All RMSDs were compared to the all-*trans* conformations of each polyene.

Table 2. Parameters Optimized for Each Test system

parameter (symbol)	ethene	diene	trien	tetraene
number of ants N	65 ± 0.0	49 ± 3	49 ± 2	49 ± 2
pheromone resistance ρ	$0.33 \pm 8 \times 10^{-4}$	0.31 ± 0.02	0.32 ± 0.02	0.32 ± 0.02
probability to choose global best p_{best}	$0.12 \pm 2 \times 10^{-4}$	0.11 ± 0.01	0.09 ± 0.003	0.09 ± 0.004
global best percentage θ	$0.28 \pm 7 \times 10^{-4}$	0.25 ± 0.01	0.25 ± 0.01	0.25 ± 0.01
explorative/exploitive search controller α	$0.85 \pm 4 \times 10^{-4}$	0.85 ± 0.01	0.85 ± 0.01	0.85 ± 0.01
trail width σ	$0.02 \pm 8 \times 10^{-5}$	0.02 ± 0.002	0.02 ± 0.001	0.02 ± 0.001

Table 3. Diverse Set of Molecular Balances^a

(i) Carnago <i>et al.</i> ⁵³ (3)	(ii) Diederich <i>et al.</i> ⁵⁴ (4)	(iii) Shimizu <i>et al.</i> ⁵⁵ (3)
 R=Me, CF ₃ and Ph	 R=CF, CH, NH, OH	 pyrene (as shown), or phenanthrene, or benzene
(iv) Cozzi <i>et al.</i> ^{56–58} (18)	(v) Zoltewicz <i>et al.</i> ^{59–64} (4)	(vi) Gellman <i>et al.</i> ^{65–67} (5)
 X=NO ₂ , F, OMe, Me, H; Y=OH, O	 Z=C=Me, C-NMe ₂ , C-OMe, C-SMe	 X=Na, CH ₃
(ix) 1,9-Disubstituted triptycenes <i>et al.</i> ^{68–72} (12)	(x) 9-Benzyl triptycene derivatives ⁷³ (43)	(xii) Motherwell <i>et al.</i> ^{74,75} (10)
 X=OMe; R=COOH, COOCH ₃ , COOPh, COOCH ₂ Cl, COOCHCl ₂ , COOCCl ₃ , COOCF ₃ , H. X=NO ₂ ; R=H. X=H; R=H. R=p-N(CH ₃) ₂ , p-OMe, H, p-NO ₂ , R=o-NO ₂ , p-NO ₂ .	 X=CH ₃ ; Y=H; Z=NO ₂ , H, N(CH ₃) ₂ . X=H, COOCH ₃ , CN; Y=H; Z=H. X=CH ₃ O; Y=H; Z=H, Cl, CH ₃ O. X=Cl, Br; Y=Cl, Br; Z=H. X=Me, COMe, COEt, COi-Pr, COH, COCN, COCF ₃ ; Y=H; Z=MeO, Me, H, Cl, F, CN, CF ₃ . X=O(C=O)- <i>t</i> -bu; Y=H; Z=p-NO ₂ , p-NO ₂ , p-CN, p-CF ₃ , p-Br, p-Cl, H, p-Me, p-OMe, p-NMe ₂ , pentafluoro	 Z=OH; Y=H, Me, <i>n</i> -Bu, CN, CaL ₄ CH. Z=OMe; Y=Me, <i>n</i> -Bu. Z=O; Y=CH ₂ , (CH ₂) ₂ . Z=OCH ₂ , Y=SCH ₂ . Z=F, Y=Me. Z=NH ₂ ; Y=Me, CN.

^aThe number in the parentheses denotes the total number of structures in the set.

Table 4. Optimized Parameters for the Molecular Balance Set

parameter (symbol)	
number of ants N	46
pheromone resistance ρ	0.35
probability to choose global best p_{best}	0.11
global best percentage θ	0.26
explorative/exploitive search controller α	0.85
trail width σ	0.02

occurred. Therefore one can expect these parameters are suitably transferable across the set of balances.

The success rate of the ACO algorithm was measured with a combined energy/RMSD merit, based on the number of ACO runs which successfully generated conformations that were either better than or similar to the EF/PM6-DH+ conformations. Such a merit was formulated and presented in Table 5, and a more detailed analysis for each molecular balance studied is presented in Figure 9. Caution was taken when comparing conformations, to ensure that, for example, conformations that were generated only by a 180° flip of

Table 5. ACO Validation Criteria^a

merit	criteria	number of runs
perfect hits	$\Delta H_f = 0$, RMSD < 0.5	60
good hits	$ \Delta H_f < 0.5$, RMSD < 0.5	978
adjacent minima found (by ACO)	$ \Delta H_f < 0.5$, RMSD > 0.5	317
out-performers	$\Delta H_f \leq -0.5$	444
failure	$\Delta H_f \geq 0.5$	231

^a $\Delta H_f = H_{f, \text{ACO/PM6-DH+}} - H_{f, \text{EF/PM6-DH+}}^{\text{closed}}$ (in kcal/mol) is the difference in heat of formation between the PM6-DH+ optimized minima from a given ACO run and the corresponding *closed* form from EF/PM6-DH+ optimization, whilst RMSD measures the structural similarity between these two conformations. The last column gives the number of runs that falls into the criteria.

benzene ring around the rotation axis of the phenyl group were filtered out. In this case, an atom-index based overlay and RMSD calculation strategy is not viable⁴⁸ and should be performed based on the atomic identities. This was achieved by using the obfit routine from Open Babel.^{76,77}

Out of 2030 successful ACO runs for 102 molecular balances, 1779 runs were able to identify conformations that were similar or better than the EF/PM6-DH+ optimized *closed* forms. This demonstrates the ACO algorithm is able to find the *closed* form of molecular balances. More specifically, 1038 runs were able to identify conformations that were within 0.5 kcal/mol energy difference to the respective *closed* forms, with a high structural similarity indicated by RMSD lower than 0.5 Å. It is important to point out that the metaheuristic algorithm is able to identify more conformations than the gradient-based optimization. This can be seen from Table 5 and Figure 9 that over 700 runs were able to identify conformations that are either lower in H_f by more than 0.5 kcal/mol or structurally distinct (RMSD > 0.5 Å) but energetically similar to the *closed*

form of the respective molecular balances. Nevertheless, for a few cases, ACO failed completely to predict any conformations similar or better than the EF/PM6-DH+ optimized *closed* form. Figure 10 demonstrates examples of optimized molecular

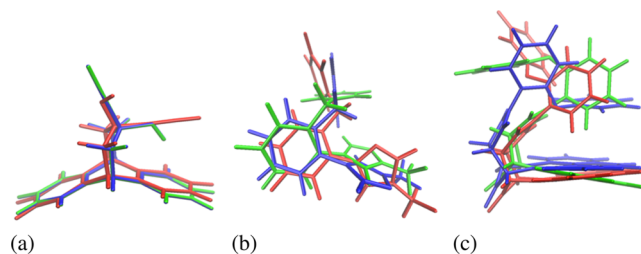


Figure 10. Examples of molecular balance conformations in *closed* (blue), *opened* (red), and one found by the ACO algorithm (green), showing the cases when ACO found conformation that (a) matched perfectly with the *closed* form (Motherwell set, Z=OH, Y=C≡CH), (b) had lower PM6-DH+ energy than the *closed* form (Cornago set, R=Me), and (c) had higher PM6-DH+ energy than the *closed* form (Schimizu set with pyrene).

balances in *opened* and *closed* forms in comparison to one global best conformation found by the ACO algorithm. It can be seen that when $|\Delta H_f| > 0.5$ kcal/mol, ACO had targeted conformations that were distinctively different from both *opened* and *closed* forms [Figure 10(b) and (c)]. However, to answer why certain conformations were more often found other than the *closed/opened* forms might require more detailed investigation on the favorable interactions in those particular systems, and it is beyond the scope of the present work.

Performance. Another aspect of the HMO algorithm worth considering is the performance. In principle, a brute-force approach guarantees the global minimum, while the metaheuristic based algorithms cannot. Therefore, for a metaheur-

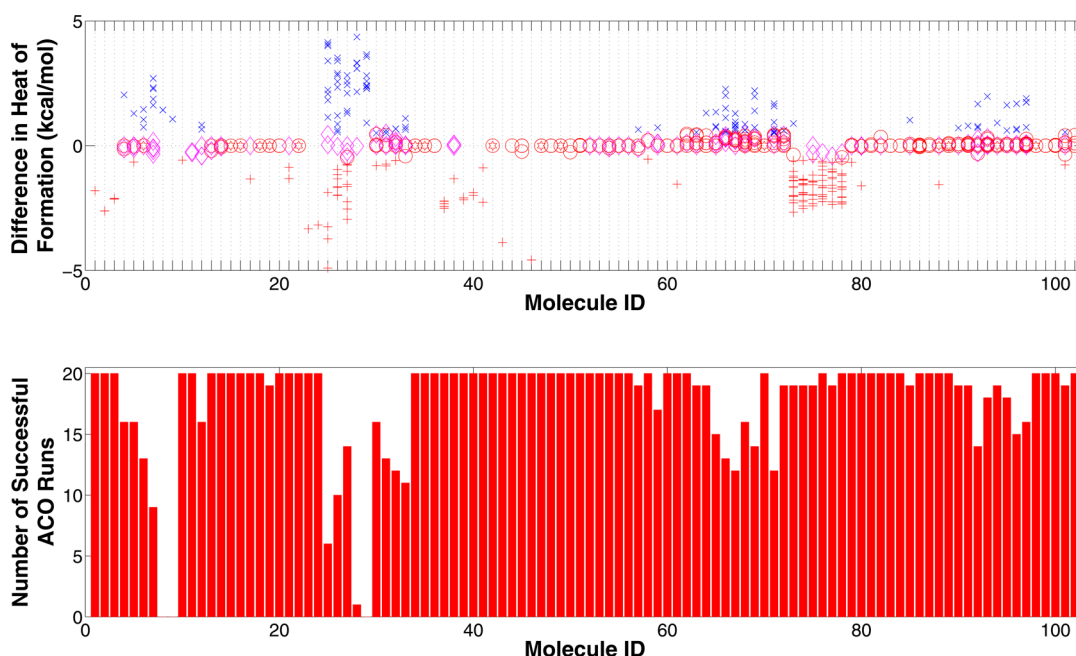


Figure 9. Hit statistics for ACO runs over 102 molecular balances, with 20 ACO meta optimizations for each system. (Top) ΔH_f for the global best from each ACO runs for each individual molecules. The markers are assigned according to the merits in Table 5 as perfect hit (☆), good hit (○), adjacent minimum (◇), out performer (+), and failure (×). (Bottom) Histogram showing the total number of runs that were not considered to be failure for each molecular balance tested.

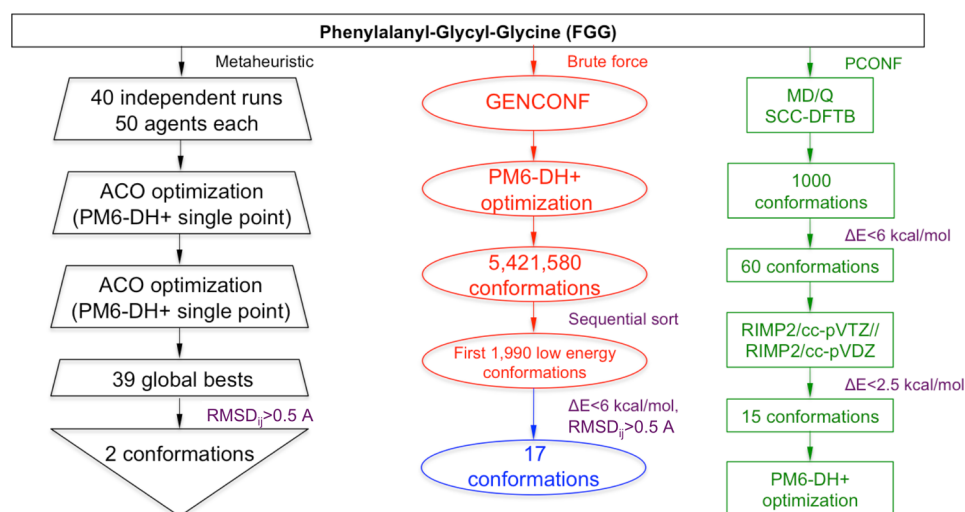


Figure 11. The workflows used to determine the performance of the HMO compared to a brute-force approach. Both sets of results are compared to the MD/Q data from Hobza et al.

istic algorithm to compete in generating low-energy conformations, the computational effort should be significantly less. In order to compare the computational effort, we investigated the relatively well-understood FGG tripeptide, with 10 rotatable dihedral angles, reported by Hobza et al.¹¹ as a test-case.

For the brute-force calculations, a pool of starting conformations was generated by rotating the dihedral angles of all the rotatable bonds with 60 degree increments, using the GENCONF program.⁴⁷ A total of 5,421,580 geometry optimizations were performed from the possible 60 million conformations (among which 55 million conformations were filtered out by GENCONF due to the presence of steric clashes and/or identical conformations being generated), with the CPU time estimated to be around 4 years/CPU. The 1,990 lowest energy conformations were then extracted from the optimized conformations. Figure 12 is a cluster plot of the energy difference versus the RMSD of each conformation to the one of the lowest energy in the ensemble. This cluster plot is useful for analyzing the ensemble of low-energy conformations (red dots) that are clustered together, and these clusters indicate a set of similar conformations which should be filtered out. By choosing cutoffs for energy difference of 6 kcal/mol (as by Hobza et al.) and pairwise RMSD of 0.5 Å, 17 distinct low-energy structures can be selected from brute-force calculations as indicated by blue circles in Figure 12.

We also performed geometry optimization with PM6-DH+ on the 15 FGG conformations provided by Hobza et al. (denoted as the PCONF set), obtained with the MD/Q (molecular dynamics with quenching) method (see Figure 11). The lowest energy conformation for this set of structures lies $\sim 0.03 \text{ eV}$ (or 0.7 kcal/mol) higher in energy when compared to the global minimum from the brute-force calculation, but interestingly it is identical to the second lowest minimum found by the brute-force approach (RMSD of 0.03 Å). Furthermore, no conformation was found close to the global minimum, and by inspecting the cluster plot, 4 conformations from PCONF were very similar to the conformations found with the brute-force approach below an energy threshold of 0.2 eV (or 6 kcal/mol).

In total, 40 ACO runs were performed for conformation searching of FGG (with the parameters as shown in Table 6). The lowest energy conformation from each run were further

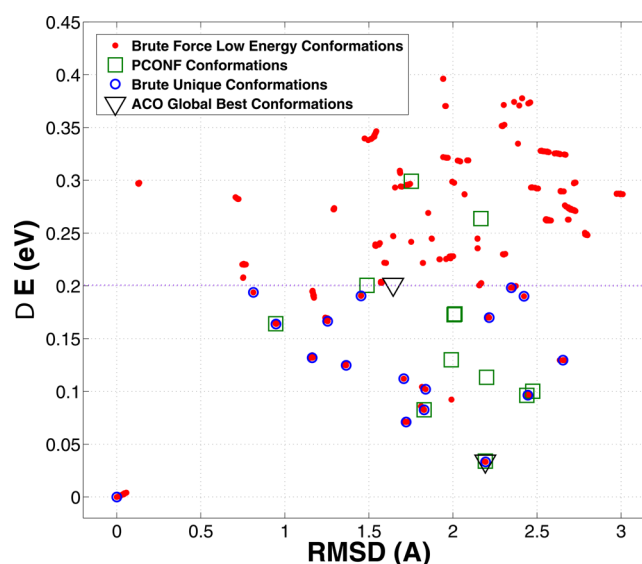


Figure 12. Cluster plot for conformation analysis and filtering. A combined energy/RMSD filtering was applied, with energy difference less than 0.2 eV (or $\sim 6 \text{ kcal/mol}$, as indicated by purple dashed line) and pairwise RMSD less than 0.5 Å, to identify 17 unique conformations from 1,990 low-energy conformations obtained from brute-force search. Also overlaid are results for all 16 optimized PCONF conformations from Hobza et al. (\square), each compared to the global minimum from brute-force calculation. The lowest energy PCONF conformation is identical to the second lowest minimum on the energy profile, which resides in a broad shallow basin. The ACO algorithm is able to identify two distinct conformations, with one being identical to the lowest energy conformation from PCONF.

Table 6. ACO Parameters Used for Optimizing the FGG Tripeptide

parameter (symbol)	
number of ants N	50
pheromone resistance ρ	0.25
probability to choose global best p_{best}	0.6
global best percentage θ	0.6
explorative/exploitive search controller α	0.8
trail width σ	0.02

optimized with the EF optimizer at the PM6-DH+ level of theory. Out of a total of 39 successful ACO runs, 27 runs were able to hit the lowest energy conformer from PCONF (or the second lowest energy conformation from the brute-force calculations). Thus, the ACO algorithm is capable of finding low-energy conformation(s). The run time with ACO was 1.5 h/CPU/run on average, which is significantly lower than a complete brute-force search. Figure 13 shows an overlay of the

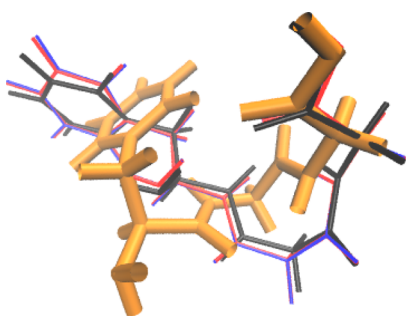


Figure 13. Overlay of the minimum energy conformations for FGG tripeptide: (red) second lowest energy conformation from brute-force calculations, (blue) global minimum from PCONF, (black) global minimum found from ACO calculations, and (orange thick line) global minimum from brute-force calculations.

minimum energy conformations obtained with three different approaches. The lowest energy conformation from the ACO overlays nicely with the best conformation from PCONF and the second best from brute-force. The global best conformation from the brute-force approach is also shown, which exhibits a more folded structure compared with the others. Interestingly, neither the ACO nor the MD/Q method was able to identify the global best conformation obtained from the brute-force approach.

CONCLUSIONS

In conclusion, we have presented a hybrid metaheuristic approach for nonlocal optimization of molecular systems and validated it on a set of relatively simple molecules. The metaheuristic ACO algorithm was implemented to optimize the conformations of molecules when their stable conformations are difficult to predict based solely on chemical intuition.

Our implementation of the ACO algorithm requires six adjustable input parameters, which may be system dependent. The hybrid metaheuristic optimization solved this problem by simultaneously optimizing the ACO parameters with the coupled metaheuristic PSO algorithm. We have shown that, with the use of a series of difluorinated polyenes as a training set, a consistent set of parameters can be systematically obtained for a group of molecules that fall into a particular structural category. In this sense, the algorithm is able to learn by itself to adjust the optimum parameter set for solving a particular problem. This significantly reduces the manual effort required by users who wish to apply the metaheuristic algorithms as a conformation search tool for diverse molecules.

Gradient-based optimizations are routinely applied by computational chemists for performing structural optimization. This requires a good knowledge of the (possible) low-energy conformation(s) as starting point(s). The validity of our metaheuristic ACO algorithm had been tested against the *closed* conformations for a set of molecular balances that were optimized with gradient-based methods. We were able to

demonstrate that, over multiple ACO runs, the algorithm is able to target those stable structures in most cases. Interestingly, apart from the high success rate of the ACO in finding the similar structures to the gradient-based optimized structures, the ACO algorithm also identified a number of even lower energy minima than the gradient-based optimized structures.

Furthermore, the computational effort of the ACO algorithm and the brute-force approach were studied by investigating the conformations of the FGG tripeptide. In principle, a brute-force search guarantees the identification of the global minimum, but it is computationally impractical for routine calculations as it required 4 CPU years of computational time for this relatively simple system. The second lowest energy conformation from the systematic search for FGG tripeptide can be quickly identified within 40 ACO runs; each requires only 1.5 CPU hours on average, which is far more feasible for routine calculations.

It is intrinsic to the metaheuristic algorithm that there is no guarantee for it to reveal the global best conformation, as demonstrated in the search case for the FGG conformations. In this sense, we propose that the HMO method could be a practical tool for exploring the conformational space of molecular systems before using traditional gradient-based optimization techniques. Naturally, the computational cost of the HMO algorithm does impose limitations on which Hamiltonian may be appropriate. Therefore, we advocate a two-step strategy where the HMO algorithm is used in combination with semiempirical methods to locate a collection of low-energy conformations and subsequently reoptimizing this collection with a gradient based algorithm using density functional theory or wave function based methods.

ASSOCIATED CONTENT

Supporting Information

Coordinates for training sets, *closed* form of molecular balances, and best conformations for FGG tripeptides. This material is available free of charge via the Internet at <http://pubs.acs.org>.

AUTHOR INFORMATION

Corresponding Author

*E-mail: m.waller@uni-muenster.de.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

Generous financial support by the Deutsche Forschungsgemeinschaft and SFB858 is gratefully acknowledged. Special thanks also go to the students Fabian Lied and Daniel Janßen-Müller who worked on this project during a lab course.

REFERENCES

- (1) Schlegel, H. B. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2011**, *1*, 790–809.
- (2) Hartke, B. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2011**, *1*, 879–887.
- (3) Catlow, C. R. A.; Bromley, S. T.; Hamad, S.; Mora-Fonz, M.; Sokol, A. A.; Woodley, S. M. *Phys. Chem. Chem. Phys.* **2010**, *12*, 786–811.
- (4) Woodley, S.; Catlow, R. *Nat. Mater.* **2008**, *7*, 937–946.
- (5) Daeyaert, F.; de Jonge, M.; Koymans, L.; Maarten, V. J. *Comput. Chem.* **2007**, *28*, 890–898.
- (6) Dieterich, J. M.; Hartke, B. *Mol. Phys.* **2010**, *108*, 279–291.

- (7) Strunk, T.; Wolf, M.; Brieg, M.; Klenin, K.; Biewer, A.; Tristram, F.; Ernst, M.; Kleine, P. J.; Heilmann, N.; Kondov, I.; Wenzel, W. J. *Comput. Chem.* **2012**, *33*, 2602–2613.
- (8) Li, Z.; Scheraga, H. *Proc. Natl. Acad. Sci. U.S.A.* **1987**, *84*, 6611–6615.
- (9) Wales, D. J.; Doye, J. P. K. *J. Phys. Chem. A* **1997**, *101*, 5111–5116.
- (10) Wales, D. *Energy Landscapes*; Cambridge University Press: Cambridge, UK, 2003; pp 434–563.
- (11) Řeha, D.; Valdés, H.; Vondrášek, J.; Hobza, P.; Abu-Riziq, A.; Crews, B.; de Vries, M. *Chem.—Eur. J.* **2005**, *11*, 6803–6817.
- (12) Al-Sunaidi, A. A.; Sokol, A. A.; Catlow, C. R. A.; Woodley, S. M. *J. Phys. Chem. C* **2008**, *112*, 18860–18875.
- (13) Luke, S. *Essentials of Metaheuristics*; Lulu, 2009. Available for free at <http://cs.gmu.edu/~sean/book/metaheuristics/>.
- (14) Ogata, K.; Umeyama, H. *J. Mol. Graphics Modell.* **2000**, *18*, 258–272.
- (15) Zheng, W.; Cho, S. J.; Waller, C. L.; Tropsha, A. Simulated annealing guided evaluation (SAGE) of diversity: A novel computational tool for diverse chemical library design and database mining. **1997**.
- (16) Gregurick, S. K.; Alexander, M. H.; Hartke, B. *J. Chem. Phys.* **1996**, *104*, 2684–2691.
- (17) Tekin, A.; Hartke, B. *Phys. Chem. Chem. Phys.* **2004**, *6*, 503–509.
- (18) Ole Carstensen, N.; Dieterich, J. M.; Hartke, B. *Phys. Chem. Chem. Phys.* **2011**, *13*, 2903–2910.
- (19) Carlotto, S.; Orian, L.; Polimeno, A. *Theor. Chem. Acc.* **2012**, *131*, 1–7.
- (20) Strassner, T.; Busold, M.; Herrmann, W. A. *J. Comput. Chem.* **2002**, *23*, 282–290.
- (21) Johnston, R. L. *Dalton Trans.* **2003**, 4193–4207.
- (22) Feng, Y.; Cheng, L.; Liu, H. *J. Phys. Chem. A* **2009**, *113*, 13651–13655, PMID: 19908881.
- (23) Stepanenko, S.; Engels, B. *J. Comput. Chem.* **2008**, *29*, 768–780.
- (24) Grebner, C.; Becker, J.; Stepanenko, S.; Engels, B. *J. Comput. Chem.* **2011**, *32*, 2245–2253.
- (25) Grebner, C.; Kästner, J.; Thiel, W.; Engels, B. *J. Chem. Theory Comput.* **2003**, *9*, 814–821.
- (26) Shmygelska, A.; Hoos, H. *BMC Bioinf.* **2005**, *6*, 30.
- (27) S. Fidanova, I. L. In *proceeding of: Computer Science and Information Technology, 2008. IMCSIT 2008. International Multi-conference on*; 2008.
- (28) Huber, T.; van Gunsteren, W. F. *J. Phys. Chem. A* **1998**, *102*, 5937–5943.
- (29) Eiben, A. E.; Hinterding, R.; Michalewicz, Z. *IEEE Trans. Evolutionary Computation* **1999**, *3*, 124–141.
- (30) Pedersen, M. E. H. *Tuning & Simplifying Heuristical Optimization*. Ph.D. thesis, University of Southampton, 2010.
- (31) Addicoat, M. A.; Page, A. J.; Brain, Z. E.; Flack, L.; Morokuma, K.; Irle, S. *J. Chem. Theory Comput.* **2012**, *8*, 1841–1851.
- (32) Smit, S. K.; Eiben, A. E. *Proceedings of IEEE International Conference on Evolutionary Computation* **2009**, 399–406.
- (33) Dorigo, M.; Di Caro, G. In *New Ideas In Optimization*; Corne, D., Dorigo, M., Glover, F., Eds.; McGraw Hill: London, UK, 1999; pp 11–32.
- (34) Dorigo, M., T., Stützle In *Handbook Of Metaheuristics*; Gendreau, M., Potvin, J.-Y., Eds.; Springer: 2010.
- (35) Dorigo, M., T., Stützle *Ant Colony Optimization*; MIT Press: Cambridge, MA, 2004.
- (36) Korb, O.; Stützle, T.; Exner, T. *Swarm Intell.* **2007**, *1*, 115–134.
- (37) Stützle, T.; Hoos, H. H. *Future Generation Computer Systems* **2000**, *16*, 889–914.
- (38) Kennedy, J.; Eberhart, R. *Particle Swarm Optimization*. 1995.
- (39) Poli, R. *J. Artif. Evolution Applications* **2008**, *1*, 1–10.
- (40) Waller, M. P.; Dresselhaus, T.; Yang, J. *J. Comp. Chem.* **2013**; DOI: 10.1002/jcc.23272.
- (41) Korth, M. *J. Chem. Theory Comput.* **2010**, *6*, 3808–3816.
- (42) Stewart, J. J. P. MOPAC2012. 2012. <http://OpenMOPAC.net>.
- (43) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. *J. Phys. Chem.* **1985**, *89*, 52–57.
- (44) Baker, J. *J. Comput. Chem.* **1986**, *7*, 385–395.
- (45) Culot, P.; Dive, G.; Nguyen, V.; Ghuysen, J. *Theor. Chim. Acta* **1992**, *82*, 189–205.
- (46) Dennington, R.; Keith, T.; Millam, J. GaussView Version 5. 2009; Semichem Inc.: Shawnee Mission, KS, 2009.
- (47) Budde, J. Effiziente Konformerensuche für große organische Moleküle. M.Sc. thesis, FernUniversität in Hagen: 2009.
- (48) Coutias, E. A.; Seok, C.; Dill, K. A. *J. Comput. Chem.* **2004**, *25*, 1849–1857.
- (49) Černý, J.; Hobza, P. *Phys. Chem. Chem. Phys.* **2007**, *9*, 5291–5303.
- (50) Schneider, H.-J. *Angew. Chem., Int. Ed.* **2009**, *48*, 3924–3977.
- (51) Mati, I. K.; Cockroft, S. L. *Chem. Soc. Rev.* **2010**, *39*, 4195–4205.
- (52) French, R. H.; et al. *Rev. Mod. Phys.* **2010**, *82*, 1887–1944.
- (53) Cornago, P.; Claramunt, R. M.; Bouissane, L.; Elguero, J. *Tetrahedron* **2008**, *64*, 3667–3673.
- (54) Fischer, F.; Wood, P.; Allen, F.; Diederich, F. *Proc. Natl. Acad. Sci.* **2008**, *3*, 17290–17294.
- (55) Carroll, W. R.; Pellechia, P.; Shimizu, K. D. *Org. Lett.* **2008**, *10*, 3547–3550, PMID: 18630926.
- (56) Cozzi, F.; Annunziata, R.; Benaglia, M.; Cinquini, M.; Raimondi, L.; Baldrige, K. K.; Siegel, J. S. *Org. Biomol. Chem.* **2003**, *1*, 157–162.
- (57) Cozzi, F.; Siegel, J. S. *Pure Appl. Chem.* **1995**, *67*, 683–689.
- (58) Cozzi, F.; Annunziata, R.; Benaglia, M.; Baldrige, K. K.; Aguirre, G.; Estrada, J.; Sritana-Anant, Y.; Siegel, J. S. *Phys. Chem. Chem. Phys.* **2008**, *10*, 2686–2694.
- (59) Zoltewicz, J. A.; Maier, N. M.; Fabian, W. M. F. *J. Org. Chem.* **1997**, *62*, 2763–2766.
- (60) Zoltewicz, J. A.; Maier, N. M.; Fabian, W. M. F. *J. Org. Chem.* **1996**, *61*, 7018–7021.
- (61) Zoltewicz, J. A.; Maier, N. M.; Fabian, W. M. F. *J. Org. Chem.* **1997**, *62*, 3215–3219.
- (62) Zoltewicz, J. A.; Maier, N. M.; Lavieri, S.; Ghiviriga, I.; Abboud, K. A.; Fabian, W. M. *Tetrahedron* **1997**, *53*, 5379–5388.
- (63) Zoltewicz, J. A.; Maier, N. M.; Fabian, W. M. F. *J. Org. Chem.* **1998**, *63*, 4985–4990.
- (64) Lavieri, S.; Zoltewicz, J. A. *J. Org. Chem.* **2001**, *66*, 7227–7230.
- (65) Gardner, R. R.; McKay, S. L.; Gellman, S. H. *Org. Lett.* **2000**, *2*, 2335–2338.
- (66) Gardner, R. R.; Christianson, L. A.; Gellman, S. H. *J. Am. Chem. Soc.* **1997**, *119*, 5041–5042.
- (67) McKay, S. L.; Haptonstall, B.; Gellman, S. H. *J. Am. Chem. Soc.* **2001**, *123*, 1244–1245.
- (68) Ōki, M.; Izumi, G.; Yamamoto, G.; Nakamura, N. *Chem. Lett.* **1980**, *9*, 213–216.
- (69) Izumi, G.; Yamamoto, G.; Ōki, M. *Chem. Lett.* **1980**, *9*, 969–972.
- (70) Ōki, M.; Izumi, G.; Yamamoto, G.; Nakamura, N. *Bull. Chem. Soc. Jpn.* **1982**, *55*, 159–166.
- (71) Yamamoto, Y. T. G.; Ōki, M. *Bull. Chem. Soc. Jpn.* **1987**, *60*, 1781–1788.
- (72) Tamura, Y.; Yamamoto, G.; Ōki, M. *Chem. Lett.* **1986**, *15*, 1619–1622.
- (73) Ōki, M. *Acc. Chem. Res.* **1990**, *23*, 351–356.
- (74) Motherwell, W.; Moise, J.; Aliev, A.; Nič, M.; Coles, S.; Horton, P.; Hursthouse, M.; Chessari, G.; Hunter, C.; Vinter, J. *Angew. Chem., Int. Ed.* **2007**, *46*, 7823–7826.
- (75) Aliev, A. E.; Moise, J.; Motherwell, W. B.; Nic, M.; Courtier-Murias, D.; Tocher, D. A. *Phys. Chem. Chem. Phys.* **2009**, *11*, 97–100.
- (76) O’Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. *J. Cheminf.* **2011**, *3*, 33.
- (77) The Open Babel Package. 2011. <http://openbabel.org>.