Charles University in Prague
Faculty of Science
Department of Physical and Macromolecular chemistry

Doctoral Thesis

Quantum-chemical and Molecular-dynamical Study of Noncovalent Interactions

Jan Řezáč

Advisor: Prof. Ing. Pavel Hobza, DrSc.

Institute of Organic Chemistry and Biochemistry AS CR
Center for Biomolecules and Complex Molecular Systems

Univerzita Karlova v Praze
Přírodovědecká Fakulta
Katedra fyzikální a makromolekulární chemie

Disertační práce

Kvantově-chemické a molekulárně-dynamické studium nekovalentních interakcí

Jan Řezáč

Školitel: Prof. Ing. Pavel Hobza, DrSc.

Ústav organické chemie and biochemie AV ČR
Centrum biomolekul a komplexních molekulárních systémů

Declaration of the author

I declare that I have worked out this thesis by myself using the cited references. Neither the thesis nor its parts were used previously for obtaining any academic degree.

Prague, 20th July 2008

Jan Řezáč

Acknowledgements

In the first place, I would like to thank to Prof. Pavel Hobza; this work would not be possible without his support and advice. Special thanks should be rendered to Tomáš Kubař and Vojtěch Spiwok who participated in development of the Cuby code. Many studies presented here are a result of broad collaboration and I would like to thank to all the colleagues involved in it. Finally, I should apreciate the friendly and stimulating environment in our laboratory.

Preface

Modern computational methods and powerful computers made quantum mechanical calculations applicable to large sets of data. In this work, we exploit the efficiency of density functional theory calculations, and apply it to problems requiring calculations of thousands points. We can run molecular dynamics with potential calculated *ab initio* on the fly or use statistics to analyze large sets of results.

The first part of this work summarizes outcomes of our research in past four years. Only the results relevant to the topic of the thesis are described in detail, more information could be found in original papers, which are listed in List of publications and attached as Appendices.

Some calculations presented here would be impossible to achieve with readily available software. To adopt and combine new methods, we are developing our own code. Description and documentation of this code is integral part of this work.

List of Abbreviations

AFM	Atomic force microscopy
CBS	Complete basis set
CCSD(T)	Coupled clusters with single, double and triple (using perturbation theory) excitations
C-PCM	Conductor-like polarizable continuum model
DFT	Density functional theory
DFTB-D	Short notation of SCC-DFTB-D
DFT-D	Density functional theory augmented with empirical dispersion correction
FES	Free energy surface
GPW	Gaussian atomic orbitals with plane wave auxiliary basis set
HF	Hartree-Fock method
MAXE	Maximum unsigned error
MD	Molecular dynamics
MM	Molecular mechanics
MP2	Møller-Plesset perturbation theory (second order)
MP3	Møller-Plesset perturbation theory (third order)
MUE	Mean unsigned error
OVOS	Optimized virtual orbital space
PD	Parallel displaced (structure of benzene dimer)
PES	Potential energy surface
QM	Quantum mechanics
RESP	Restricted electrostatic potential (fit)
RMSE	Root mean square error
SCC-DFTB-D	Self-consistent charges density functional tight binding, with dispersion correction
SCF	Self-consistent field
SCS-MP2	Spin component scaling MP2
TS	T-shaped (of benzene dimer)
WC	Watson-Crick (base pair)
WFT	Wavefunction theory

Part One:

Quantum-chemical and Molecular-dynamical Study of Noncovalent Interactions

Table of Contents

1 Introduction

1.1 Noncovalent interactions

Noncovalent interactions are weaker than covalent bonds, but it does not mean they are less important. It is especially true in biomolecules – their primary structure is simple, they are composed from limited number of building blocks. What determines their structure and biological function are noncovalent interactions.

Strongest noncovalent interactions are of electrostatic nature and involve charged species. Weaker electrostatic interactions act between dipoles and higher multipoles. Electrostatic interactions are easy to describe even with simple computational methods, beginning with Coulomb's law in point-charges model.

Specific type of interaction is the hydrogen bond, formed between hydrogen bonded to electronegative atom and another electronegative atom. Although major part of this hydrogen bonding can be described by simple electrostatics, quantum mechanical effects come into play at such a short distance, resulting in partial covalent character of hydrogen bonds. Quantum-chemical calculation is thus needed for realistic description of hydrogen bond, although it could be simple calculation, beginning with semiempirical methods, density functional theory and Hartree-Fock calculation.

Last between the most important noncovalent interactions is the van der Waals interaction, also known as London dispersion. It is attractive interaction of instantaneous and induced dipoles, acting even in neutral and nonpolar atoms or molecules. Although it is weaker then interactions mentioned above and acts over relatively short range, its power in real world is in its abundance. Acting between every pair of atoms, substantial effect can be achieved when all small contributions are summed up. For proper description of dispersion, it is necessary to describe electron correlation – a method beyond single electron approximation is needed, what makes the calculation expensive. Fortunately, it can be also described empirically using simple formalism, such as the Lennard-Jones potential.

1.2 Overview of studied systems

Our main interest is the role of noncovalent interactions in biomolecules. In order to adopt and develop new methods for description of noncovalent interactions we also work with model systems, on which new approaches are tested.

Major part of this work is devoted to DNA. Structure and function of the DNA molecule is determined by interaction of nucleic acid bases, which can be studied using quantum-mechanical (QM) methods. Knowledge of stability of the DNA double helix, measured as free energy of its unwinding (dissociation, denaturation), is important not only for understanding of the function of DNA, but also for working with DNA in laboratory. Stability of DNA oligomers is known do be dependent on sequence of nucleotides and can be estimated using empirical statistical models. In two papers presented here[1, 2], we investigated the relationship between interaction between DNA bases, which can be readily calculated, and experimentally evaluated stability of DNA oligomers.

Long DNA molecule is structure large enough to be mechanically manipulated using recent experimental techniques. Atomic force microscopy (AFM) can study stretching of single DNA double helix. We simulated this experiment in molecular dynamics simulations and studied resulting structures with QM methods. Different behavior was observed in poly-AT and poly-CG, which can be explained by different properties of the bases and their interaction.

Another part of this work focuses on small peptides. These peptides can be studied by accurate experiments in gas phase, which, in conjunction with theory, help to understand their properties and properties of peptides in general. Gas-phase experiments can be directly compared to calculations on isolated molecule – theory can help to interpret experimental data and experiment helps to refine theoretical methods. In one paper[3], we thoroughly investigated free energy surface of glycyl-phenylalanyl-alanine tripeptide (GFA) to assign structure to measured infrared spectra of different conformers. In second paper on peptides[4], we evaluated the performance of wide array of computational methods applied to several small peptides. Although a tripeptide is relatively small molecule, its structure is determined by intramolecular noncovalent interactions.

Development of efficient density functional theory methods (DFT) with dispersion correction (DFT-D) (For details, see Methods section at page 7) in our laboratory allowed us to move from static description of selected points to on-the-fly molecular dynamics (MD) simulations with accurate potential calculated using QM in each step. We have developed new parametrization[5] (Appendix I) of DFT-D for benzene dimer, a prototype system for studying π...π interactions in aromatic systems. This accurate potential was then used in MD simulations investigating structure and thermodynamics of benzene dimer[6] (Appendix J).

Another part of my work was implementation of the DFT-D method within combined quantum-mechanical / molecular mechanical (QM/MM) calculations. Resulting code is used to study carborane inhibitors of HIV protease[7] (Appendix H). Specialized implementation of DFT-D was also used in study of adsorption of aromatic molecules on water surface[8].

1.3 The Cuby code

Some of these studies are based on new methods not implemented in available software. This is especially true for calculations combining more methods, such as QM/MM or molecular dynamics with DFT-D. To be able to use these methods, we have developed our own code, named Cuby, what stands for Chemistry in Ruby. However, we do not want to compete with established software packages. On the contrary, we make use of them. Our code calls external programs to perform the QM calculation and performs only the following manipulation with the results, such as adding the dispersion correction, combining more calculations in the QM/MM procedure and manipulating the geometry in optimization or molecular dynamics.

It is written in Ruby[9], a high level object-oriented language, what makes the development faster and easier. Although it started as one-purpose tool for QM/MM calculations, it has grown (including a major rewrite) to universal package that can handle all the most frequent types of calculations. It was designed to be user-friendly from the very beginning, and now it offers unified interface to calculations in several software packages.

At present, Cuby is widely used in our laboratory. In future, we plan to make it available to the public, but it will take some more time, because some parts of it are still in development. The documentation should also be improved; this work is part of the attempt to consolidate it.

More details, including documentation of all its features, are provided in the second part of this work.

2 Methods

Many widely used computational methods were used thorough this study. They are well described in available literature and textbooks. In the following sections, we present only several recent methods that are of key importance for this work.

2.1 DFT-D

As it was said before, rigorous treatment of the London dispersion requires description of electron correlation. Wavefunction theory based methods (WFT) can naturally describe it, when we pass from single-electron approximation to post Hartree-Fock methods. These methods could be very accurate, but also very expensive. For systems where the dispersion is not important, good results can be achieved efficiently with DFT methods. Unfortunately, dispersion is missing in common DFT functionals.

There are attempts to include the dispersion into DFT. Some approaches are based on proper theoretical description of electron correlation; these methods tend to be expensive as their WFT counterparts. Another approach is to reparametrize existing functionals, mainly in the exchange term, to simulate the dispersion correction. Functionals recently introduced by Zhao and Truhlar[10] are the most successful, but even these are limited by the form of the functional and different distance dependence of exchange and dispersion. A third approach is to add empirical correction, similar to MM forcefield, after the DFT calculation is done.

The last approach is very efficient and good results can be achieved. Its origins lie in similar scheme for HF calculations, devised by Scoles et al.[11, 12] Simple formalism using the C_6/R^6 term is damped to correct the interaction at short distances, where the QM method itself provides good description. Similar HF-based method was also used by Hobza[13-16] for larger molecular complexes, including base pairs. Later, the dispersion correction was applied to DFT as well[17-21]. Grimme[22] studied effect of basis set used and introduced scaling of the damping function individually for each functional used.

Recently, Jurečka[23] introduced fitting parameters in the damping function for each functional/basis set combination to benchmark data, using set of molecular complexes ranging from hydrogen bonds to dispersion-bonded complexes. Compared to Grimme's approach, there is no global scaling parameter in the dispersion term, only the damping function is adjusted to the given method. This formalism is more accurate at larger distances, where the dispersion is not damped. We use the formalism and parameters presented in the original Jurečka's work thorough this study, unless it is noted otherwise.

2.2 On-the-fly molecular dynamics

Recent development of efficient methods as well as new powerful computers made the usage of quantum chemical methods in molecular dynamics possible. On-the-fly *ab initio* molecular dynamics is also known as Born-Oppenheimer molecular dynamics, because it obeys the Born-Oppenheimer approximation. Nuclear motion is treated classically, but the potential is calculated using *ab initio* method in each step of the calculation.

The main advantage, compared to molecular mechanics (MM), is that we are not limited by given form of forcefield. There are no parameters needed to be calculated before the actual simulations. Chemical nature of the system can change during the simulation – we could study chemical reactions. Looking at molecular vibrations, on-the-fly molecular dynamics brings true nonharmonic potential without any restrictions on it's form.

Compared to molecular mechanics, there is one obvious limitation: QM calculations are expensive. Even with very efficient methods, only relatively small systems can be studied and length of the trajectories is limited.

When analyzing the results, we must be also aware that it is not exact description of quantum reality, we are using classical mechanics. On the other hand, full quantum mechanical description of such a systems is impossible.

Many QM software packages have the option to perform molecular dynamics, but their capabilities are often limited. Moreover, to use the DFT-D method, the dispersion correction must be added to each QM calculation. For these reasons, we decided to implement molecular dynamics at our own in the Cuby code (For details, see chapter 1.3 and Part two of this work, devoted to Cuby). It uses external programs to do the QM calculation. Every supported QM code can be coupled with dispersion correction in this implementation. On top of this layer providing the potential, molecular dynamics with wide variety of options is implemented. It might not be the fastest MD implementation, but in the case of on-the-fly MD, the overall performance is determined by the QM calculation. In our implementation, we use the widely adopted Verlet algorithm for integration of the equations of motion.

To perform simulations at constant temperature, there exist broad range of thermostat algorithms. The choice of proper thermostat is very important in simulations of isolated molecules or molecular complexes. One class of thermostats, such as the popular Nosé-Hoover algorithm[24, 25], uses one factor for rescaling all atomic velocities in the system. In simulation using this algorithm, redistribution of the energy in the system is limited to intramolecular energy flow, which is relatively slow. This leads to poor sampling of the configuration space in the simulations. This problem avoided simulations of condensed phase, where the energy transfer is facilitated by the solvent.

For simulations of isolated molecule or cluster, which serves as a model for experiments in gas phase, usage of stochastic thermostats is more appropriate. We use the Andersen[26] thermostat, which simulates collisions with other particles with the desired temperature. With an average collision frequency, new velocity from the Maxwell distribution is generated for randomly selected atom. In contrast to the global scaling algorithms, this thermostat does not conserve direction of the total momentum in the system, what significantly improves the sampling.

Another issue in any MD simulation is the conservation of energy. For good conservation of total energy, we need accurate integration of the equations of motion. The key factor is the timestep. In simulations at constant temperature (NVT, what stands for constant number of particles N, volume V and temperature T), step size of 1 fs is generally accepted value. In accurate simulations at constant energy (NVE, what abbreviates constant number of particles N, volume V and energy E), shorter step is recommended.

There is one more factor affecting the energy conservation in on-the-fly molecular dynamics not found in MM simulations. The *ab initio* calculation itself, based on iterative self-consistent procedure(SCF). is converged just to some finite limit, what affects not only the electronic energy, but introduces some residual gradient. In common QM calculations, result of previous step is used as the initial guess for construction of molecular orbitals. As it was pointed out by Pulay and Fogarasi[27], this makes the error introduced by imperfect SCF convergence systematic, what results into leaking of the total energy. One possible solution is to minimize this problem by setting of substantially tighter convergence limit, but this is impractical because it makes the calculation longer. Another possibility, adopted in our MD simulations at DFT-D level, is to start each QM calculation from scratch. Although it makes the calculation longer, it completely eliminates the problem. Another possible solution was suggested by Pulay and Fogarasi: molecular orbitals, or the Fock matrix itself, can be constructed by extrapolation from multiple previous steps. This not only improves the performance, but it also makes the error random, what leads to its cancellation during the simulations. Unfortunately, we can not adopt this method because we use closed source package for the calculations.

2.3 Metadynamics

Metadynamics[28, 29] is a recent method of calculation of free energy based on molecular dynamics. It allows calculation of free energy surface (FES) in limited (usually two) internal coordinates, also called collective variables. In our implementation, these can be distances, angles, dihedral angles and coordination (coordinate describing closeness of groups of atoms). From the simulation, we obtain profile of free energy in these coordinates, while the remaining degrees of freedom are thermodynamically averaged.

The method is based on MD simulation, during which the internal coordinates are evaluated. In these coordinates, bias potential (in form of Gaussian functions) is periodically added. This bias potential then affects the simulations, because its gradient in the internal coordinates is converted to Cartesian space and added to the potential in the simulation. Since the system is likely to be close to minimum on FES, the bias potential fills these minima. In addition to this "direct"[30] metadynamics, it is also possible to introduce virtual particles propagated in the selected internal coordinates, which are coupled to the simulated system. Moment of inertia of these particles could enhance the sampling of the FES. In our simulations, however, this approach led to problems. The system was forced to leave a minimum in the direction of gradients of the internal coordinate only, while there was another more energetically favorable way. To avoid this problem, which impairs convergence of the simulations, we used only the direct metadynamics.

The simulation is converged when all minima in the surface determined by the internal coordinates are filled. The bias potential is then an inverse of the FES itself.

The metadynamics method is implemented in Gromacs[31] MM simulation package. Because we wanted to run not only MM simulations, but also use on-the-fly MD with self-consistent charges density functional tight binding potential aurmented with empirical dispersion[32] (SCC-DFTB-D, DFTB-D for short), we implemented the algorithm in the Cuby code. Here, it is possible to combine the method with any level of calculation, and we used it for both MM and DFTB-D simulations for consistency.

2.4 DFT-D in QM/MM study of carborane inhibitors of HIV protease

Metallocarboranes were recently found[33] to be inhibitors of HIV-1 protease, an important target enzyme in AIDS therapy. Carboranes have unique properties due to high electropositivity of boron. Although the molecule is hydrophobic, the hydrogen on its surface have negative partial charge and can interact with biomolecules via formation of dihydrogen bonds[34, 35].

Our study was focused on computational refinement of a X-ray structure of the enzyme in complex with metallocarborane inhibitor. Two main questions, which can not be explained by the experiment, were addressed: Boron and carbon can not be distinguished by the X-ray crystallography, what makes impossible to identify orientation of the carborane cages containing two asymmetric carbons. The same applies for sodium ions and water, there were two uncertain positions to be resolved.

Description of the system by molecular mechanics would be difficult. The system contains transition metal, cobalt, and carborane cages with specific properties. On the other hand, quantum mechanical description of the whole system would be impossible. We decided to apply combined QM/MM method. However, it was necessary to use method that is able to describe London dispersion, which plays an important role in interaction of the inhibitor and hydrophobic residues in the active site of the enzyme. Considering the size of the QM region in

QM/MM scheme, DFT-D was the only possibility.

To be able to perform these calculation, QM/MM was implemented in our Cuby code. In this study, we couple MM calculations in AMBER[36, 37] with DFT in Turbomole[38]. Subtractive QM/MM scheme similar to the ONIOM method[39, 40] is used to compose these calculations. The whole system is calculated at MM level, then, QM region (called cluster) calculated using the same MM method is subtracted and the QM of the cluster calculation is added. It leads to following formula for energy:

$$E_{system}^{QM/MM} = E_{system}^{MM} - E_{cluster}^{MM} + E_{cluster}^{QM} . \qquad (1)$$

If the boundary of the cluster cuts covalent bond, it is replaced by hydrogen link atom in the cluster. Link atoms are placed along the bond in a distance given by:

$$d(XH) = d(XY)\frac{d_{eq}(XH)}{d_{eq}(XY)}, \qquad (2)$$

where d stands for actual distance and d_{eq} for equilibrium distance. X and Y are the original bonded atoms, H is the hydrogen link atom. Energy derivative is calculated according to the original work of Morokuma.

Scheme described above calculates the interaction of the QM region with its environment only at MM level. To improve it, we use "electrostatic embedding": Point charges of atoms in the environment (from MM forcefield) are added into the QM calculation (and removed from MM). As a result, the QM region is polarized by its environment. This approach can bring problems with point charges in proximity of link atoms. We use one of suggested possibilities and remove charge on selected closest atoms. To conserve total charge of the system, this charge is distributed evenly in the rest of the residue. Cuby code also implements distance cutoff for selection of the point charge for calculations of very large systems.

Geometry optimization of large systems takes many steps to converge. When the QM calculation is expensive, optimization of the whole QM/MM system could be impossible. This issue is addressed by two trick used in our calculations.

Firstly, the outermost layer of the system is not optimized at all, what reduces the number of degrees of freedom in the calculation. It also helps to stabilize the experimental geometry of the system, which might not be stable when we work with isolated molecule taken from condensed phase.

Secondly, we employ microiterations[41] to reduce the number of necessary QM calculations to minimum. In this approach, the optimization is done in two nested cycles. In each step of optimization of the whole system, the MM environment is fully optimized prior the QM calculation. This reduces the dimensionality of the main optimization, which requires QM gradients, to the size of the QM region.

3 Performance of used methods

When performing the calculations, we must be aware of the performance and limits of the used methods. For that reason, evaluation of used methods in an important part of every theoretical work.

3.1 Efficient methods applied to interactions in DNA

In our calculations on stretched DNA, as well as in the study of DNA stability, it was necessary to evaluate large number of interactions in base pairs. The most accurate method applicable to this problem is, in our opinion, the DFT-D. It was shown[23, 42] that it yields results close to the benchmark high-level calculations, and it describes well the balance between various types of interactions, namely hydrogen bonding and stacking in DNA.

For some applications, even the DFT-D could be too expensive. For this reason, we also tested DFTB-D, a semiempirical DFT-based method with dispersion correction, and AMBER forcefield for calculation of interaction energies.

Firstly, we can compare these methods with benchmark data calculated using accurate CCSD(T)/CBS method (coupled clusters, extrapolation to complete basis set limit). We did this comparison for AT base pair in hydrogen-bonded (Watson-Crick) and stacked structure from the S22 data set[43], results are summarized in Table 1.

Table 1: Interaction energy (in kcal/mol) in the adenine – thymine base pair in Watson-Crick (WC) and stacked geometry. For comparison, DFT energies are presented without and with dispersion correction (labeled + D.)

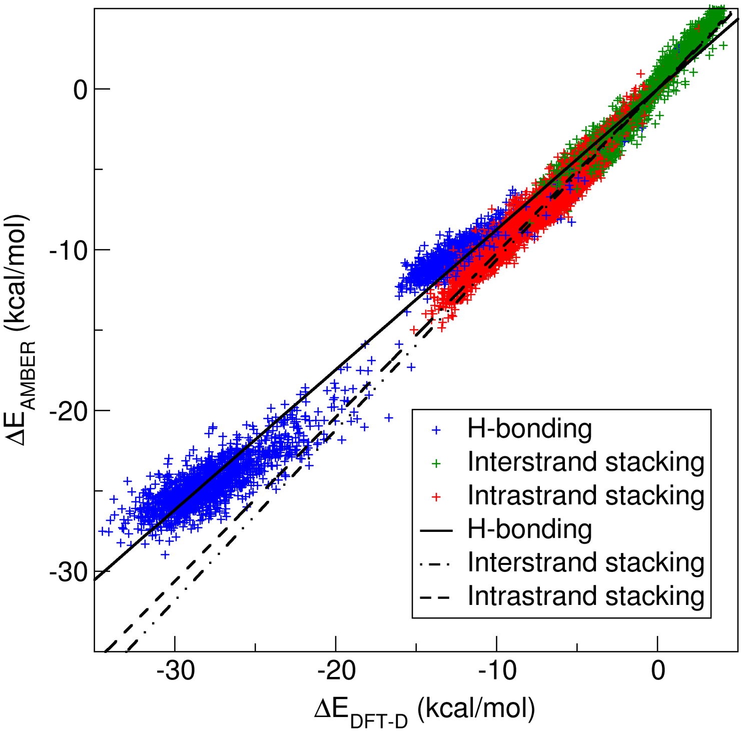
Method	ΔE^{int} (WC)	ΔE^{int} (Stack)
CCSD(T)/CBS	-16.37	-12.23
TPSS/TZVP	-14.15	-0.35
TPSS/TZVP + D	-17.08	-11.17
TPSS/6-311++G(3df,3pd)	-13.56	-0.64
TPSS/6-311++G(3df,3pd) + D	-16.71	-11.89
AMBER ff99	-12.75	-10.76
SCC-DFTB-D	-10.57	-10.13

DFT-D interaction energies are within 1 kcal/mol from the benchmark CCSD(T) results. Selected combination of TZVP[44] basis set and TPSS[45] functional was tested to give acceptable results while being very efficient, more accurate results can be achieved using larger basis set (TPSS/6-311++G(3df,3pd)[46] combination in Table 1). DFT calculations without dispersion are included for illustration; they fail to describe stacking completely. Even interaction energy in hydrogen bonded complexes is improved by the dispersion correction, although it is then slightly overestimated. AMBER ff99[47] forcefield underestimates both values; DFTB-D is even worse, yielding only negligible difference between the two interaction energies.

Calculation of several structures allows the comparison against accurate data, but it does not show the nature of the discrepancies. Analysis of large sets of data in our works allowed us to analyze the trends in differences between forcefield and DFTB-D compared to more reliable DFT-D data.

In the case of MD simulations of stretched DNA, it was crucial to check how the forcefield describes unusual structures far from commonly studied equilibrium geometry. Since DFT-D calculations of interaction energies were performed on selected snapshots from these simulations, we used them as a benchmark and calculated the same structures with AMBER ff99 forcefield. RESP charges were derived for isolated bases using the recommended procedure[47].

Figure 1: Interaction energies of base pairs calculated using DFT-D and AMBER ff99 forcefield on snapshots from MD simulations of stretched DNA



The interaction energies between base pairs calculated using the AMBER forcefield for both polyGC and polyAT are plotted against those calculated using DFT-D in Figure 1. These energies have been partitioned into the contribution from hydrogen bonding (blue) and intra- and inter-strand stacking (shown in green and red respectively). There is an excellent agreement between the MM and QM methods for intra and inter-strand stacking interactions (AMBER covers 104 % of the DFT-D value). However, the hydrogen bond interactions are systematically underestimated by AMBER, giving an average difference of -1.9 kcal/mol for an AT base pair and 3.3 kcal/mol for a GC base pair. We attribute these differences to effects that cannot be included by MM approaches; in particular to polarization, but also to the partial covalent character of the hydrogen bonds between complementary base pairs. This hypothesis is supported by the observation that the effect is more pronounced for higher base pair interaction energies (Only 87% of the interaction calculated at DFT-D level).

Good results achieved by the AMBER forcefield justify its use in MD simulations, including stretched DNA. The forcefield had also shown good performance in our study of DNA stability, where the systematic underestimation of hydrogen bonding was eliminate in the fitting procedure used.

Very similar results apply for DFTB-D and its comparison to DFT-D. Following discussion is based on our study of stability of 140 DNA octamers. The correlation between the sum of pairwise interaction energies of a particular type of interaction obtained from both computational methods is very good (R^2(H-bonding) = 0.9999, R^2(interstrand stacking) = 0.9903 and R^2(intrastrand stacking) = 0,9949). For the stacking interactions, DFTB-D interaction energy is 107% of the DFT-D value. For H-bonding, the DFTB-D value is only 80% of DFT-D, what is even worse than AMBER. Again, this systematic error was not important in our calculation of DNA stability, because each interaction was weighted separately in the fitting procedure.

3.2 Potential energy surface of peptides

Small peptides became recently an important topic in our laboratory. They are model systems proteins, yet they are small enough to be studied in gas phase or using QM calculations. These peptides have vast number of possible conformations, but only some of them are energetically favorable. It turned out that very accurate computational methods must be applied in reproduce the experimental data and sort the conformers according to their energy.

Our interest in this field, as well as development of new promising methods led to a systematic study[4] (Appendix E) of selected peptide structures using wide range of computational methods ranging from accurate benchmark calculations to evaluation of MM forcefields.

Here, we would like to present part of this study focused on performance of low cost methods.

3.2.1 Structures

Five peptides were considered in this study: WG, WGG, FGG, GGF and GFA. These letters abbreviate following amino acids: alanine (A), glycine (G), phenylalanine (F) and tryptophan (W). Two of them, phenylalanine and tryptophan, have aromatic side chain, which takes part in intramolecular dispersion interactions. Presented structures are the lowest energy conformers from conformational search employing MD/quench and subsequent refining of both energy and geometries [48]. In total, 76 structures, incuding geometries from recent works of Valdes et al.[48-50] was used in this study.

Different types of conformations are represented in this set. In some, multiple intramolecular hydrogen bond are formed, while other structures are stabilized by dispersion. For reliable calculations on this variable dataset, accurate method properly covering different types of interactions is necessary.

26 of these peptides had been selected for a more detailed study. They have been selected to represent different families of peptide conformations. In analogy to the S22 database[43], this collection, named P26, is a balanced set that could be used in development and testing of new methods.

3.2.2 Geometries

Final step of the geometry refinement procedure used in search for conformers with lowest energy is a MP2/cc-pVTZ optimization. On the P26 set, optimizations using some of the studied methods were performed to compared obtained geometries with the benchmark MP2/cc-pVTZ ones.

Following methods were tested: DFT-D is represented by the well performing combination of TPSS functional and 6-311++G(3df,3pd) basis set (abbreviated as LP). Other approach to dispersion in DFT represents MO6-2X functional of Truhlar and Zhao[10] in combination with recommended basis set. The DFTB-D method is included because it has proven itself to yield good results for peptides with an unparalleled efficiency. Two calculations were performed using AMBER ff99 forcefield, one with charges derived from HF/6-31G* calculation, the other at B3LYP/cc-pVTZ level. Finally, B3LYP/6-31G* level is included, although it lacks the dispersion, because it is widely used in literature on similar systems. RMSE of these structures from MP2 geometry, averaged over the P26 set, is listed in Table 2.

Table 2: RMSE of geometries, obtained by studied methods, compared to benchmark MP2/cc-pVTZ structure, averaged over the P26 set.

Method	avg. RMSE (Å)	max. RMSE (Å)
TPSS/LP + D	0.16	0.64
MO6-2X/MIDI	0.19	0.46
DFTB-D	0.09	0.28
ff99 w. HF charges	0.16	0.24
ff99 w. DFT charges	0.17	0.32
B3LYP/6-31G*	0.55	1.47

DFT-D shows average performance, but the maximum RMSE is surprisingly large. MO6-2X is found to be worst between the methods that should cover the dispersion. Poor B3LYP results are not surprising; this method was expected to fail. This evidence is, however, very important, because this method has become standard among many users of computational software.

The best of the tested methods was the DFTB-D, although it is only semiempirical method and is very cheap. Moreover, it was not parametrized on peptides in particular (unlike the forcefield, derived to describe proteins).

The forcefield have shown average performance without extremes, regardless on the charges used. It remains good solution where cost of the calculation must be kept low.

In this evaluation, one question remains open – the quality of the reference MP2 geometries. It is known to overestimate dispersion, while the DFT-D and MO6-2X were parametrized on coupled clusters data, and structures obtained by these methods could be more accurate than the MP2 ones. This hypothesis is also supported by the fact that the outlying points differ mainly in position of the nonpolar side chains, which is affected mostly by the dispersion. These differences, large in geometric measures, may also lead only to small change of the energy, because the dispersion has, due to its nonspecific nature, rather flat potential.

This issue could be resolved by comparison of CCSD(T) energies on these optimized structure and on the original one. It would not be surprising if the DFT-D geometries were, at least in the described cases, better. These calculations are not trivial, but open possibilities to expand this study.

3.2.3 Relative energies of conformers

In the comparison of energies, all 76 structures was included. Structures are numbered in the plots; the set contains 16 GFA, 15 WG, 15 WGG, 15 GGF and 15 FGG structures in this order. In each peptide, energies relative to the lowest energy conformer (ΔE) are calculated. These energies are then compared to CCSD(T)/CBS values in terms of mean unsigned error (MUE) and maximum unsigned error (MAXE).

Before we start to look at the results, it must be mentioned that there was more methods included in the original study. Here, we compare only the cheaper methods up to DFT. Even here, we pick only some functionals most relevant to the rest of this work. Wavefunction methods with explicit calculation of correlation (MP2, MP3, SCS-MP2) all performed better than the cheap method listed here.

In Table 3, selected methods are listed in order of growing MUE. Let us start from the end and discuss the failure of the forcefield (see Figure 2) first. Maximum errors of 9 and 12 kcal/mol (for DFT and HF charges) are far higher than the range of 4 kcal/mol where all CCSD(T) conformer energies lie. MM calculation allows decomposition of the energy into separate terms of the forcefield. This analysis reveals that the source of these large errors is in dihedral angles. This deficiency of the forcefield was observed before[51] and several modifications were proposed to fix this problem. In addition to ff99 presented in the paper, we have also tested ff99SB[52] and FF03[53], where, according to the literature, dihedral angles are described better. Nevertheless, our results have shown only slight improvement; these forcefield can not be compared to *ab initio* methods. Systematic rebuild of the dihedral parameters could still bring substantial improvements to these forcefield.

Poor performance of BLYP/6-31G* method (Figure 3) was already discussed on the geometries, the same applies here. Methods that do not cover dispersion should not be applied to this problem, where interaction of nonpolar side chains plays an important role.

DFTB-D performs well and at a fraction of the cost of full DFT calculation. Good and balanced performance proved our previous experience with application of this method to peptides. We can recommend it as an efficient tool in evaluation of large number of structures or MD simulations.

Finally, let us compare the two DFT approaches covering dispersion. Here, the MO6-2X functional yields better results, although the difference from DFT-D (represented here by the TPSS/LP +D method) is not large and both approaches exhibit similar behavior. On the other hand, DFT-D is about one order of magnitude more efficient, because the resolution of identity approximation (RI) can be applied.

Table 3: Mean unsigned error (MUE) and maximum error (MAXE) in relative conformer energies of 76 peptide structures calculated using listed methods compared to CCSD(T) benchmark calculations.

Method	MUE (kcal/mol)	MAXE (kcal/mol)
MO6-2X/MIDI	0.68	2.11
DFTB-D	0.79	2.16
TPSS/LP + D	1.00	3.02
B3LYP/6-31G*	2.03	6.67
AMBER ff99/DFT charges	2.43	8.79
AMBER ff99/HF charges	3.45	11.72

Figure 2: Relative energies of peptide conformers in set of 76 structures. Empirical and semiempirical methods.

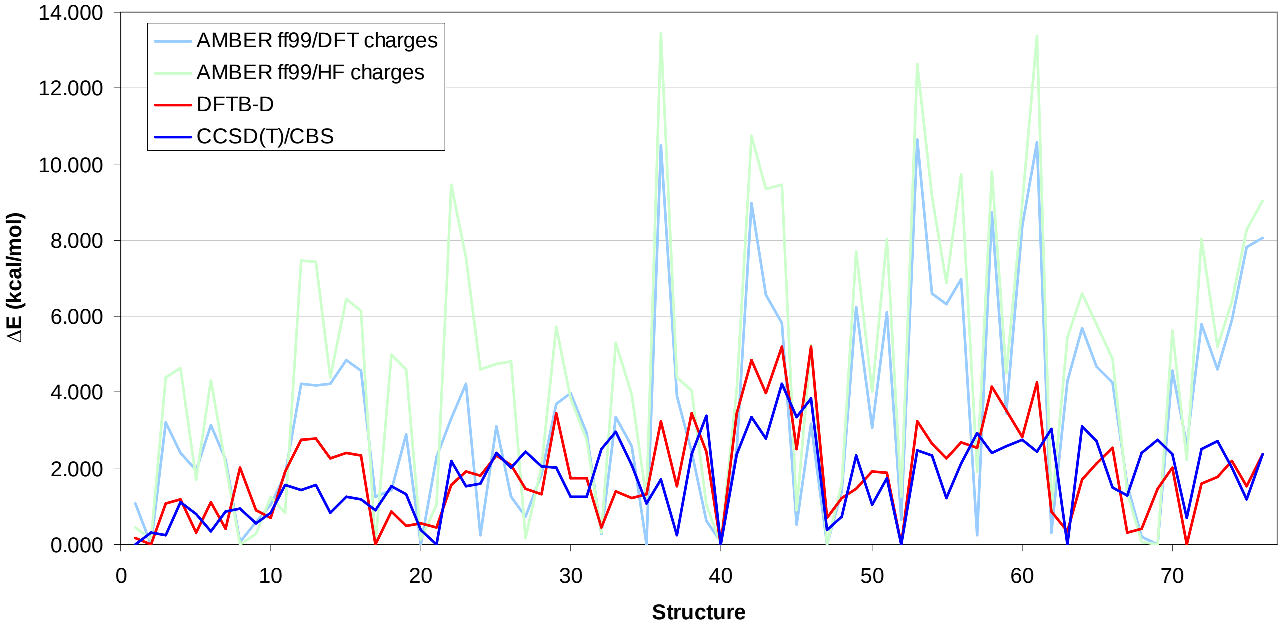
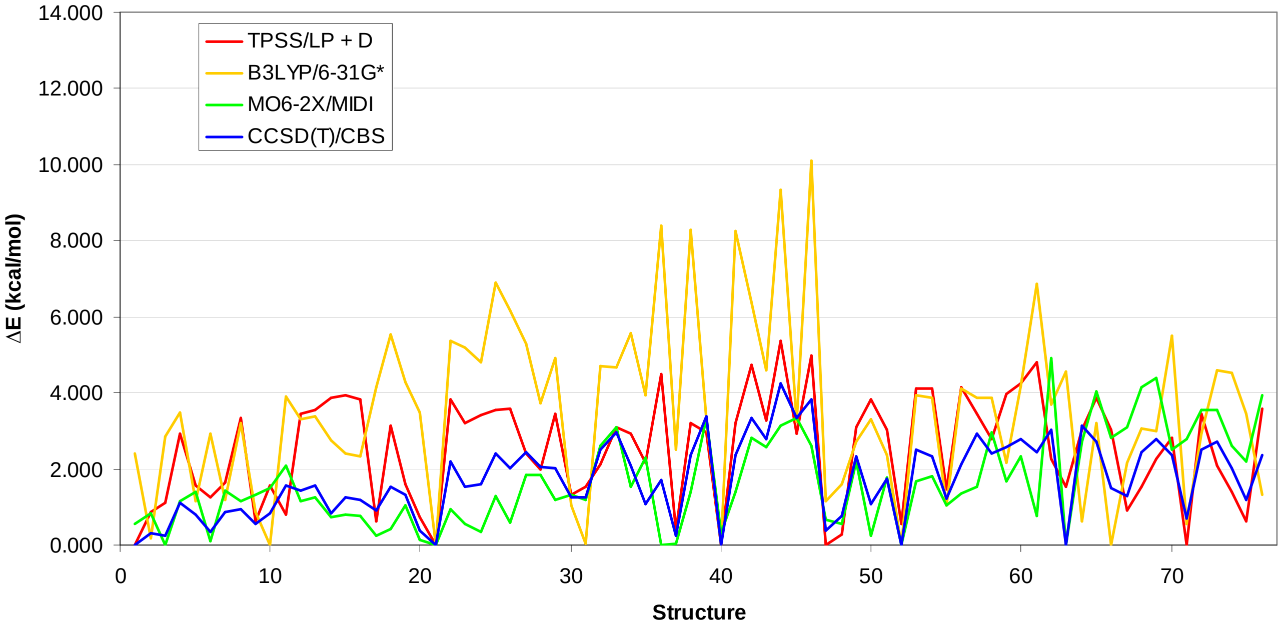


Figure 3: Relative energies of peptide conformers in set of 76 structures. DFT based methods.



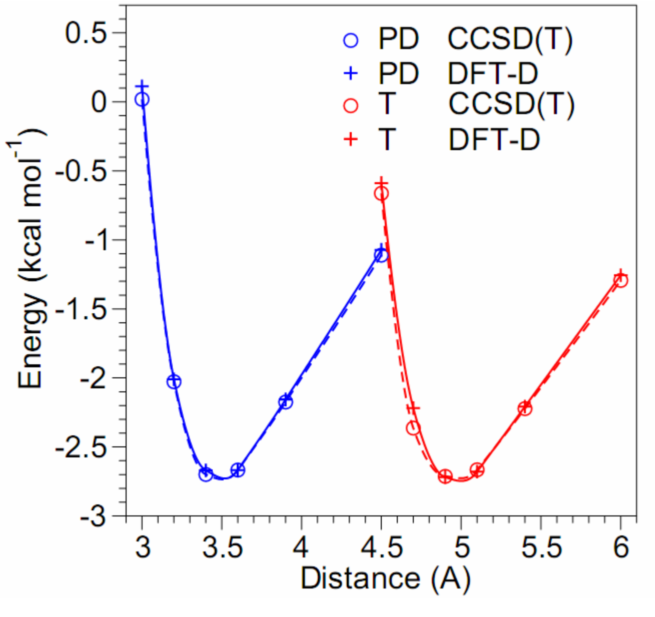
3.3 DFT-D for benzene dimer

Another study where efficient but accurate method was needed was the application of on-the-fly MD to benzene dimer. DFT-D method is a promising candidate here, because it is efficient enough for molecular dynamics and covers the dispersion. Commonly used parameters, designed to be transferable, were developed as a compromise between accurate description of dispersion on one side and short-ranged interactions, namely hydrogen bonds, on the other. As a result, reasonable accuracy was achieved, but it is not enough in benzene dimer. Using the standard

DFT-D parameters for B-LYP/TZVP level, the parallel displaced (PD) structure is stabilized by 0.24 kcal/mol compared to the tilted T-shaped (TS) structure, while in reality, this difference should be negligible (The best calculation available[5], CCSD(T)/CBS extrapolated from large basis sets, predicts the TS structure to be 0.08 kcal/mol more stable than PD). Using the original DFT-D parametrization by Grimme[22], opposite is true, and the TS structure is stabilized by about 0.6 kcal/mol. This method was used by Pavone et al.[54] to perform calculations and short simulations of benzene dimer, but this bias in potential makes their findings questionable.

As a basis for the DFT-D calculation, we have selected B-LYP functional and TZVP triple-zeta basis set. Where efficiency is crucial, we can not use hybrid or meta-GGA functional. The basis set is a compromise between accuracy and efficiency, use of smaller basis sets is not recommended in DFT-D.
For our study, we reparametrized the dispersion term to fit exactly dissociation curves of both TS and PD structures, calculated at CCSD(T)/CBS level. The CCSD(T)/CBS interaction energy was composed from HF calculation in aug-cc-pVQZ basis, MP2/CBS term extrapolated[55] from aug-cc-pVTZ and aug-cc-pVQZ basis, and CCSD(T) correction (difference between CCSD(T) and MP2) calculated in aug-cc-pVTZ using truncated optimized virtual orbital space (OVOS) in the CCSD(T) procedure. All three parameters in the dispersion correction formula (scaling of van der Waals radii s_R, global scaling of the dispersion term s_6 and exponent in the damping function α) were optimized. In the optimization procedure, the signed difference between DFT-D and CCSD(T) interaction energy, weighted by Boltzmann factor at 50K (to ensure more accurate fitting in the minimum) was minimized. Some arbitrary adjustments were introduced to correct the relative energy of TS and PD minima. The resulting parameters ($s_R = 0.88$, $s_6 = 1.503$, $\alpha = 6$) are significantly different from the original, transferable ones, but they yield perfect agreement with the CCSD(T) data (Figure 4).

Figure 4: Fitting of the dispersion term for the parallel displaced (PD, blue) and T-shaped (T, red) benzene dimer; circles – reference CCSD(T)/CBS values, lines and crosses – fitted DFT-D interaction energy.



These parameters were not used only in the on-the-fly MD, but they were also used to refine geometries of the benzene dimer for further coupled clusters calculations in our paper.

4 Studied systems

4.1 Stability of DNA double helix

4.1.1 Introduction

Structure of the DNA, the double helix, is closely related to its function. The structure is a result of interplay of many contributions, especially noncovalent interactions. Since the double helical structure is very regular, it is simple to characterize and study these interactions. For a long time, it was believed that it is the hydrogen bonding in Watson-Crick base pairs what forms the structure of DNA. Interaction of the stacked bases, the dispersion energy, was considered less important, probably it was difficult to quantify.

This was changed by development of accurate computational methods, which allowed to accurately calculate strength of these interactions (For a review of the topic, see Ref. [56]). In addition, different nature of these interactions should be taken into account. In solution, formation of hydrogen bonds can be compensated by favorable interaction with water. Stacking, in contrast, do not take advantage of dipole moment of the NA bases, which lies in their plane, and is thus less affected by the environment.

Stability of the double helix is defined by the free energy of its dissociation into two separate strands (denaturation). While structure of the double helix is well defined, there is no simple picture of the single strand. For short oligomers we studied, the strand stays extended and stacking of the bases is at least partially conserved. Hydrogen bonds, as well as the interstrand stacking, are of course lost.

Although we can accurately calculate energies of all the interactions in DNA molecule, we are not able to calculate the free energy. There is no method that can be used for this calculation on such a large system. Stability of DNA oligomers can be measured experimentally using calorimetry. From these experiments on oligomers with different sequence, empirical models for prediction of DNA stability were derived.

The crudest approach is to correlate stability of the duplex with contents of CG pairs, in which the hydrogen bonding is stronger than in AT. More advanced models, which include the effect of all bases next to the selected pair, are called nearest neighbor models (For a review, see Ref. [57]). These models are able to predict the stability from oligomers sequence with good results.

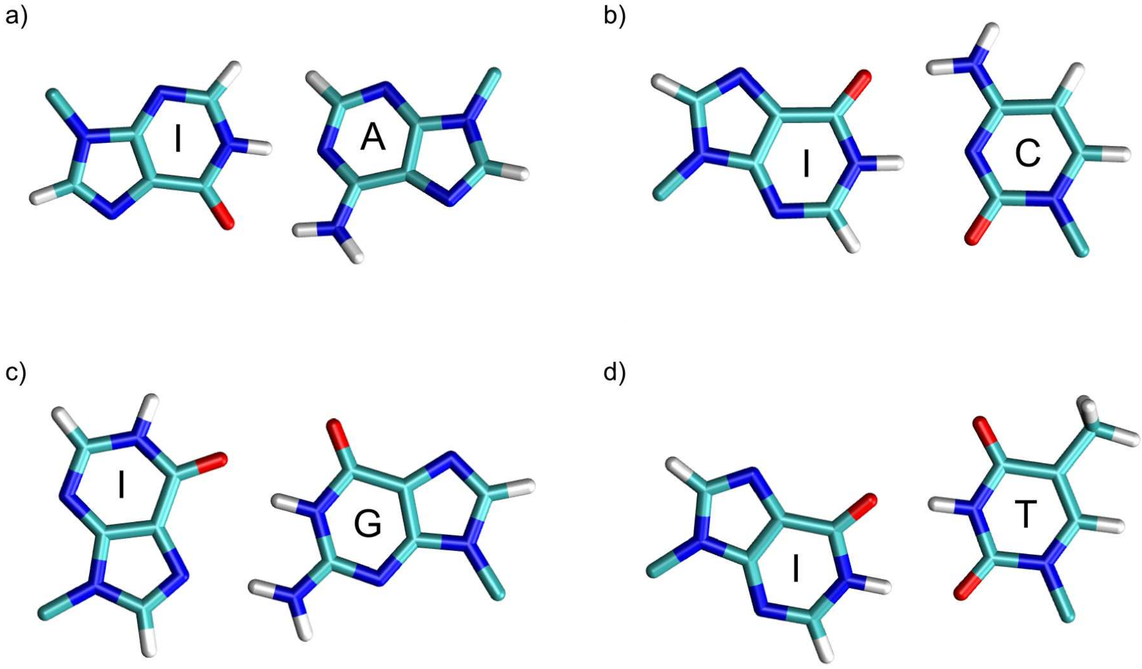
Since we are able to calculate the interaction energies in DNA, we were interested in their relationship with the overall stability of the duplex. With available experimental data, we can study this relationship by creating a model based on the calculated properties and fitted to reproduce experiment. We can also assume that the contributions that are not calculated are either very similar because of the regular nature of the structure, or proportional to the calculated variables. These variables will be thus covered by the coefficients weighting the known contributions in the fitting procedure. Such a model would give us an insight in relative importance of the particular types of interactions, and it could be used to predict stability of DNA purely from calculations.

To start with, we selected a set of 140 DNA octamers published by Doctycz et al.[58] In the original paper, measured melting temperatures and dissociation ΔG are provided, and a nearest-neighbor model is derived from these data. (All experimental free energies used here were measured at the same conditions, in 1M NaCl, and are reported as ΔG°_{310}.) In set of oligomers of the same size, the model does not need to account for the length. Paper resulting from this study[1] is attached as Appendix A. Some data from this study were also used in evaluation of performance of the DFTB-D method[59] (Appendix B).

Our attempt was successful and we decided to extend our model (Ref. [2], Appendix C) to oligomers of different length. This study was based on experimental data compiled by Sugimoto et al.[60]. In addition, we also tried to apply this model on DNA containing unnatural bases. In this case, the empirical models must be parametrized on new set of experimental data. Model based purely on calculation would be useful tool for prediction of stability of the modified DNA.

We have chosen inosine as a prototype for unnatural bases. It has one unique property: it pairs with all the natural bases. The four bonding patterns found in DNA are pictured in Figure 5. Experimental data for fitting our model were taken from the work of Watkins et al.[61].

Figure 5: Base pairs of inosine with a) adenine, b) cytosine, d) guanine, e) thymine



4.1.2 Strategy of calculation

Structure preparation

Starting from the sequence only, it is necessary to build the structure of the oligomer first. Nucgen program from AMBER[37] package was used to generate molecular geometry of the oligomers. Since this program handles only natural bases, inosine-containing sequences were created by replacement of original base. Orientation of inosine had to be adjusted in some pairs to form the experimentally determined bonding pattern.

All structures were then optimized in continuous solvent using generalized Born model[62, 63] (GBM) implemented in AMBER , using the ff99 forcefield. This procedure was previously checked to yield good B-DNA geometry of DNA oligomers. Forcefield parameters were generated using the antechamber tool according to the procedure described in Ref. [64].

Interaction energies

Only interaction energies between bases are considered in our model. All pairwise interactions of adjacent bases, the H-bonding, inter- and intrastrand stacking, were calculated using DFT-D method. Well tested combination of TPSS functional and TZVP basis set was used. Inclusion of dispersion interaction is necessary for obtaining realistic balance between hydrogen bonding and stacking. For such a large set of structures, however, methods more expensive than DFT-D would make the calculation impossible. Performance of two cheaper methods was also evaluated: on part of the structures, we calculated the interaction energies using DFTB-D method, in the second paper we tested MM calculations in AMBER, using parameters generated for isolated bases.

Interaction energies calculated in one structure are then summed to create total interaction energies of the given type: hydrogen bonding (E_h), interstrand stacking (E_i) and intrastrand stacking (E_s).

Solvation

As it was mentioned above, a role of solvation free energy is also important in DNA stability. It affects mostly strength of hydrogen bonding, due to the polar nature of the bases. Because of their dipole moments, AT pair is affected less than CG and the difference between them is reduced. To reflect this in our model, we included a free energy term in description of hydrogen bonding. Since it is not possible to calculate the solvation free energy for such a large molecule, we again use summation over contributions of base pairs. Even this calculation is not trivial, because it must reflect the environment of the base in DNA structure. For this reason, it was calculated only once for each base pair in model system, and this value was used in the real structures.

It is only possible to calculate a relative value, the difference of ΔΔGsolv of dissociation between two base pairs, but it is all what is needed in our model. To simulate access of water to the base pair, it must be embedded in DNA. Smallest possible oligomer, a trimer, was used as a model. Terminal base pairs exposed more to the solvent are treated separately, using dimer as a model.

Solvation free energy was calculated using the C-PCM method[65] (Conductor-like Polarizable Continuum Model) at HF/6-31G(d) level, implemented in Gaussian 03[66]. The C-PCM calculation is very efficient and covers not only the electrostatic term, but also other contributions such as cavitation energy. Parameters optimized for the computational level were used.

Firstly, free energy change upon dissociation was calculated for each pair. Relative value compared to AT pair is then calculated. From these contributions, relative solvation free energy upon duplex dissociation (compared to sequence containing only AT pairs) is constructed. For more details on this procedure, refer to the original papers (Appendices A and C).

Backbone deformation energy

Since introduction of the unnatural base pairs can disrupt regular structure of the double helix, we attempted to add a parameter describing irregularities in the DNA backbone. It was calculated using MM on the backbone extracted from the DNA structure. Energy of backbone in optimized structure is made relative to the unperturbed one in the ideal model constructed from the sequence. To validate the MM results, we compared them to DFT-D calculation on a model trimer, finding good agreement.

Statistical model

In our model, each type of interaction (sum of pairwise contributions) is individually weighted by corresponding coefficient c. Contribution of H-bonding is corrected for solvation prior the weighting. To account for sequence-independent effects, constant K is introduced. For oligomers of variable length, constant K from our work on octamers was expanded to length-independent part K and term KN * N, which scales with the length. The final equation, describing oligomers of variable length, is:

$$\Delta G = K + K_N * N + c_h * (Eh + DG^{solv}(\text{sequence}) + DG^{solv}(\text{ends})) + c_i * E_i + c_s * E_s\,, \qquad (3)$$

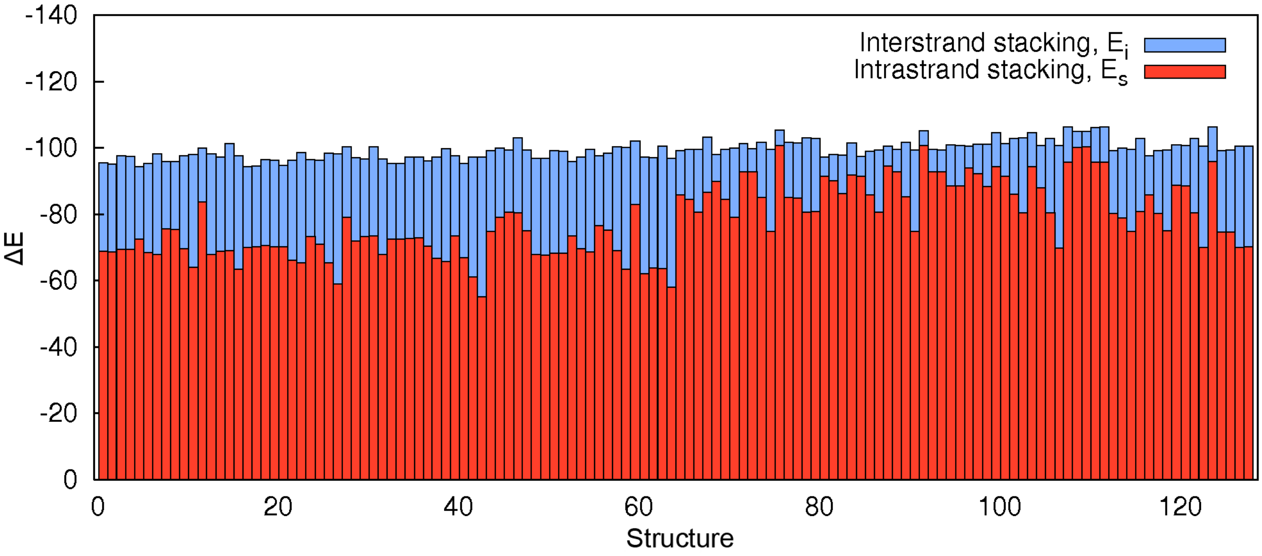
where c_h, c_i and c_s are weighting coefficients for the sums of interaction energies E_h (hydrogen bonding), E_i (interstrand stacking) and E_s (intrastrand stacking). The best fit to experimental ΔG values was obtained using the least squares method. For the fitting procedure, larger part of the experimental data was used as a training set, while the remaining sequences were used for validation of predictive capabilities of the model.

4.1.3 Results and discussion

Complementarity of stacking

Working with data calculated for 128 octamers, we found surprising relationship between interstrand and intrastrand stacking. If these two interactions are summed in the octamers, results show only very little variance (the value is 99 ± 2.8 kcal/mol), although the components varied substantially more (± 11 kcal/mol). The complementarity is clearly illustrated in Figure 6, where sum and it's components is plotted for all 128 octamers. Similar observation was noted before[67], but we have proven it in statistically unquestionable number of structures.

Figure 6: Complementarity of intrastrand (red) and interstrand (blue) stacking energy (kcal/mol) in 128 DNA octamers

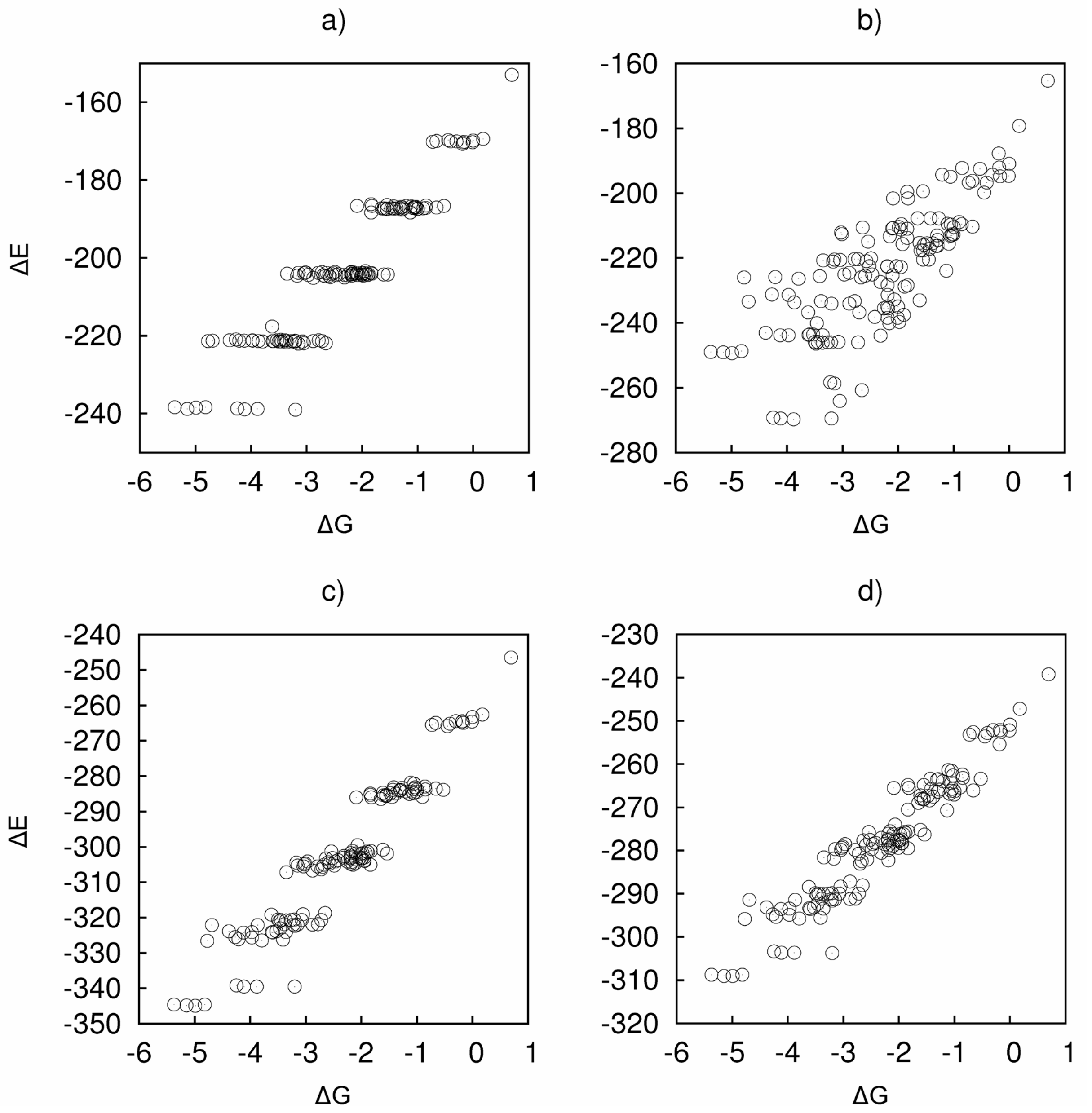


We attribute this to the nonspecific nature of the dispersion correction. Interaction of the stacked bases is determined basically by their geometric overlap, which varies between bases in one strand, but is very similar when bases from both strands are considered.

Partial model results for octamers

We can get an important insight into importance of studied contributions looking at correlation of DNA stability with some of the calculated contributions. Firstly, we plotted various combinations of the interaction energies against experimental ΔG without any fitting (Figure 7). Although the scales of the axis are different, the trends are clearly visible. The best correlation is achieved only when all the contributions are included. The most important finding here is the effect of solvation. Without the solvation corrections, results are clustered according to CG contents, because the AT/CG difference is dominant (17.2 kcal/mol). When the solvation is included, this difference is reduced to 3.0 kcal/mol.

Figure 7: Sums of total interaction energies a) E_h, b) E_h + Ei, c) E_h + Ei + E_s, d) E_h + E_i + E_s + DG^{solv} plotted against the ΔG of DNA duplex dissociation,in kcal/mol



Next step is to fit our model (Equation 3) to the experimental data. Several partial models, where some contributions were neglected, and constrained models, where more terms shared the weighting coefficient, were tested. Results of these incomplete models (levels 1-3) were summarized in Table 3 in the original paper (Appendix A). The error function optimized in the fit, root mean square error (RMSE) ranges from 0.36 to 0.94 kcal/mol.

Stability of octamers

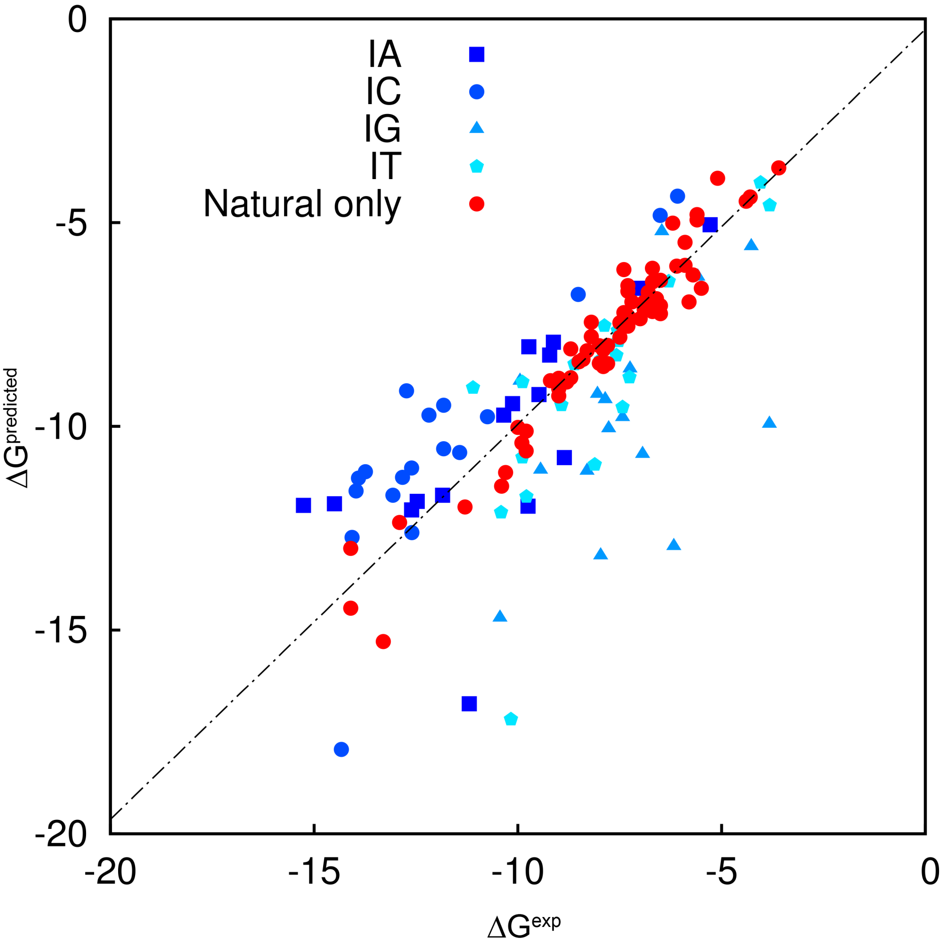
The best correlation with experimental data was achieved with the complete model (labelled Octamers in table 4) where all the contributions are weighted separately. RMSE in training set is 0.33 kcal/mol and the model is able to predict stability of octamers in validation set with RMSE 0.38 kcal/mol. It compares well with the empirical nearest neighbour model presented in the original paper[58], which achieves RMSE 0.35 kcal/mol.

In addition, we also tried to optimize the solvation term independently on the hydrogen bonding term (Level 5 in the original paper). Although the fit was slightly better, results for the validation set were worse.

Stability of oligomers of variable length

Natural extension of the work of octamers was including oligomers of various length. It required only introducing one length-dependent term into the model. Oligomers in the set range from 6 to 16 nucleotides. In the set of 65 structures, 50 randomly selected sequences were used as a training set for fit, whereas the remaining 15 structures were used for validation.

Figure 8: Predicted stability of oligomers (ΔG, in kcal/mol) plotted against experimental values. Results for oligomers containing inosine were obtained using parametrization on natural bases.



In this study, we used only the complete model with all coefficients optimized (Table 4, column Variable length). The results are plotted in Figure 8. The RMSE over the training set was 0.65 kcal/mol, which is about twice the value which we have achieved with octamers only. This is caused by the broader range of Δ*G* values covered by the training set, which contains longer oligomers. For the validation set of structures, we got a RMSE of 0.49 kcal/mol.

Table 4: Optimized fitting coefficients and error measures in models estimating ΔG od DNA duplex dissociation from RI-DFT-D calculations.

	Octamers	**Variable length**	**IX**	**All**
K	27.36 ±1.50	2.37 ±0.74	-2.11 ±1.88	2.38 ±0.77
K_n		2.76 ±0.38	0.95 ±0.45	1.31 ±0.30
c_h	0.06 ±0.00	0.06 ±0.01	0.12 ±0.01	0.09 ±0.01
c_i	0.19 ±0.02	0.23 ±0.04	-0.05 ±0.04	0.03 ±0.03
c_s	0.19 ±0.02	0.22 ±0.03	-0.07 ±0.03	0.05 ±0.02
RMSE (all)		1.77	1.90	1.45 *
RMSE (training)	0.33 *	0.65 *	2.47	1.18
RMSE (validation)	0.38	0.49	1.38	0.68
RMSE (IA)		2.01	1.92	1.73
RMSE (IC)		2.08	1.26	1.31
RMSE (IG)		3.30	1.43	2.19
RMSE (IT)		2.03	1.32	1.62
RMSE (IX)		2.40	1.33 *	1.66
$\langle R^2 \rangle$		0.39	0.53	0.55
	IA	**IC**	**IG**	**IT**
K	3.60 ±5.26	6.74 ±3.58	-6.00 ±3.22	-0.13 ±2.37
K_n	-0.25 ±1.31	-0.22 ±0.64	0.92 ±0.91	0.74 ±0.58
c_h	0.10 ±0.05	0.02 ±0.04	0.07 ±0.03	0.10 ±0.02
c_i	-0.11 ±0.14	0.10 ±0.09	0.05 ±0.09	-0.04 ±0.06
c_s	-0.08 ±0.14	0.09 ±0.08	-0.06 ±0.07	-0.05 ±0.04
RMSE (all)	1.71	2.58	2.72	2.12
RMSE (training)	1.48	1.96	3.56	2.54
RMSE (validation)	0.93	1.82	1.54	1.32
RMSE (IA)	1.52 *	2.16	2.48	2.61
RMSE (IC)	1.16	0.79 *	3.11	2.22
RMSE (IG)	2.29	4.69	1.05 *	1.61
RMSE (IT)	2.57	3.32	1.24	0.78 *
RMSE (IX)	1.97	3.09	1.99	1.71
$\langle R^2 \rangle$	0.51	0.31	0.52	0.53

RMSE values are listed for following sets of sequences: the whole set (all), training set of natural oligomers (training), validation set of natural oligomers (training), sequences containing specific inosine pairs (IA, IC, IG, IT) and all inosine containing sequences (IX).

** RMSE values optimized by the fitting procedure*

Stability of inosine containing oligomers

Our models are able to predict stability of natural DNA. However, the same is possible with the empirical models without complicated calculations. The fact that our model is based on properties of the DNA calculated *ab initio* would be an advantage when we can apply it to new structures without reparametrization.

This hypothesis was tested on sequences containing inosine paired with all four natural bases. Results obtained by the model parametrized to natural DNA (Table 4, column Variable length) were poor (see blue points in Figure 8). We also attempted to include inosine-containing sequences in the training set, either all (column All), or fit the model just to one inosine pair type (columns IA, IC, IG and IT) but without success. Another attempt to improve the description of unnatural DNA was adding the backbone deformation energy, which should cover irregularities in double helix structure caused by the unnatural base pairs, especially by accommodation of purine-purine or pyrimidine-pyrimidine pairs. The results, however, show no improvement.

The final test showing that our model fails to describe unnatural DNA, although some correlation with experiments is still conserved, is based on following assumption: Majority of the inosine containing oligomers are still natural nucleosides, which are described well by our models, what might be the reason for the remaining correlation. We looked at sequences differing only in the inosine pairs and compared predicted stability with the experiment. Poor correlation (R^2 = 0.4) indicates that description of the unnatural bases is wrong.

Our model works well for natural DNA with regular structure, where all the contributions neglected in our calculations can be included in the fitted coefficients. This is not true when inosine is introduced, and it would require at least inclusion of more terms, which would not be easy to calculate, into the model.

Importance of the studied contributions

Accuracy of prediction of DNA stability for unknown sequences is similar in our calculation and in nearest-neighbour models, but predicting DNA stability was not our only goal. In nearest-neighbour models, many parameters (~ 40) with little relevance to interactions in DNA are used. In our approach, we use only few parameters with well defined physical meaning. They represent the importance of the particular interaction, or a term that correlates with it.

The constant terms K are positive, which means that they include destabilising effects not covered by the following terms related to each type of interaction. The coefficients *c* may seem to be negligibly small, but they weight sums of interaction energies that are an order of magnitude larger than the total Δ*G* of DNA dissociation. For example, the average (corresponding approximately to an octamer) interaction energies, weighted by the coefficients, are: $c_h*(E_h+\mathrm{D}G^{\mathrm{solv}})$ = -10.8 kcal/mol, c_i*E_i = -4.4 kcal/mol and c_s*E_s = -18.9kcal/mol.

From these numbers, it is clear that it is the interstrand stacking interaction what affects the stability of an oligomer most. Importance of stacking was recently accepted, after calculations shown the magnitude of these interactions, but our results link these calculations of interaction energy directly to the stability in terms of free energy.

Hydrogen bonding is the second strongest contribution in our model, even after substantial reduction from the solvation. However, we can not say that it is less important, it is hydrogen bonding what holds the strands together and what is responsible for molecular recognition, i.e. the selective pairing of the bases, which allows storage and reproduction of the genetic information.

4.2 Structure of stretched DNA

4.2.1 Introduction

Recent experiments allow mechanical manipulation with single molecules[68-70]. Using atomic force microscopy and similar techniques, it is possible to study mechanical properties of DNA. When the DNA double helix is stretches, it elongates. However, the force/extension profile is more complex than in simple spring obeying Hooke's law.

Mechanical properties of DNA are important in biology and biochemistry, because some cellular mechanism work in similar way, applying force on the DNA.

Stretching of DNA was also addressed by computer simulations. To mimic the experiment, the extension of the DNA must be slow to allow proper equilibration. In addition, studied DNA oligomers must be long enough to simulate the experiments performed on long DNA strands; short oligomers exhibit different behavior. Such a timescale is not directly accessible to atomistic molecular dynamics. There are studies employing coarse-grained models, which allowed for long simulations of large structures[68, 71]. Recently, Harris et al.[72] developed protocol that allows to overcome the problems with timescale: Selected snapshots from fast, nonequlibrium simulation are selected and equilibrated. Atomistic resolution allows us to study structural changes in the DNA.

In our following work[73] (manuscript in preparation), which is part of larger project on mechanical properties of DNA, recently reviewed in [74] (Appendix G), we studied relations between sequence of the DNA and its mechanical properties.

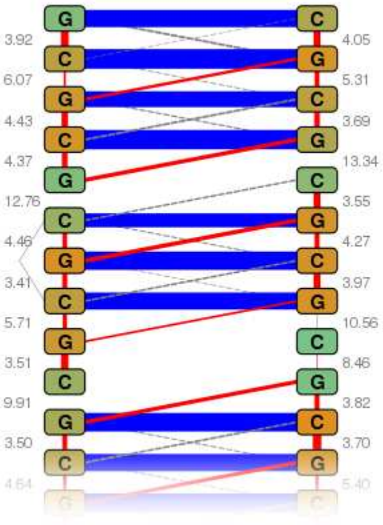
As a model, 30-mers of poly(AT) and poly(GC) were selected. Results from the MD simulation indicate that poly(AT) is softer than poly(GC) as expected. In atomistic simulations, we have observed different mechanism of the elongation.

Firstly, we evaluated performance of the used forcefield in interaction between bases in the stretched structures (see Chapter 3.1, page 11) to show that it can be used in simulations of stretched DNA.

Secondly, selected snapshots from the simulations were studied more accurately using DFT-D calculations followed by detailed analysis of the structure and interactions in it. Solvent (SASA) was calculated using Naccess program[75] to estimate solvation of bases in the structures.

To process such a large set of heterogeneous data, we have developed visualization tool that generates interactive scheme of the DNA, which includes interactions, their types and their energies, base-base distances and solvation of bases. It uses SVG (scalable vector graphics) format for output. Example is given in Figure 9.

Figure 9: Example of DNA visualization used in study of stretched DNA. Blue lines denote hydrogen bonds, red lines other attractive interactions, gray lines are close, but repulsive contacts. Line thickness is proportional to strength of the interaction. Color of bases indicates their exposition to solvent (from orange inside DNA to cyan fully exposed to water). Numbers denote distance between the bases in Å.



4.2.2 Sequence dependence of DNA extension mechanism

The DNA double helix behaves as a regular spring only at very low extension. At higher extensions, we observe regions where the original structure is corrupted (melted) and the backbone is elongated, and regions where the original B-DNA structure is conserved. This was confirmed by measurement of geometrical descriptors of the double helix using program

Curves[76]. This analysis has also shown that the double helix is untwisted in the melted regions.

Here, we should discuss the role of the sugar-phosphate backbone in the stretching of DNA. We calculated potential energy of the backbone using MM at various extensions. To be able to describe the localized melted regions, we split the backbone to dinucleotide steps. Energy was calculated relative to an average dinucleotide in B-DNA. Surprisingly, the energy differences are small even in the most elongated structures, the backbone is very flexible and in the observed region of extension the untwisting does not require much energy. The highest value calculated was about 3 kcal/mol is significantly less than the stacking energy between bases in the dinucleotide. This implies that the noncovalent interactions are determining behavior of the stretched DNA.

The experiment, as well as the force-extension profiles from simulations suggest that the poly(AT) melts at lower extension than poly(GC). Considering broken hydrogen bonds as a measure of this melting, it requires extensions of 18 Å to corrupt poly(AT), while poly(GC) starts to melt at 30 Å.

Nature of this process is also different. In poly(GC), the melting occurs in the middle of the oligomer. Firstly, only stacking is broken in some steps, what allows the DNA to unwind and extend. In poly(GC), the CG stack (in 3' to 5' direction) is always broken first, because it is weaker than the GC stack. Only after that, the hydrogen bonds are broken. At that time, water can get into the unwound structure and facilitate dissociation of the H-bonded pairs.

In poly(AT), the hydrogen bonds are weaker and their ratio to stacking interactions is lower. This oligomer melts almost always from the ends and it is hydrogen bonding what is broken first.

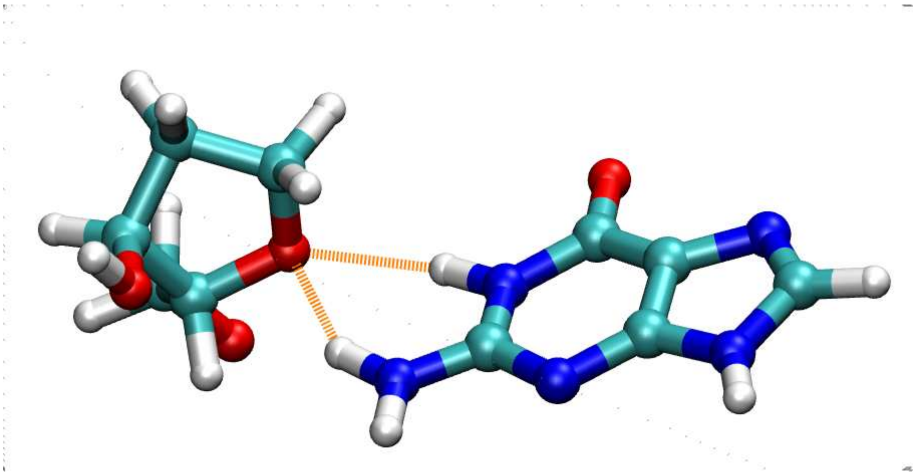
Once the H-bonds are broken, the interacting sites of the bases are exposed to water, what leads to stabilization of this structure because of the gain in solvation free energy (discussed also in chapter 4.1). This effect is stronger in GC pairs because of higher dipole moments in these bases. In both oligomers, but especially in poly(AT), we can observe extended structures where H-bonds are broken, but the bases are stacked – interleaving bases from both strands form a single column. This structure is similar to a structure of DNA containing artificial nonpolar base analogues[77].

4.2.3 Unusual structural patterns in stretched DNA

In analysis of the structures from the MD simulations, two unusual hydrogen-bonded patterns were found.

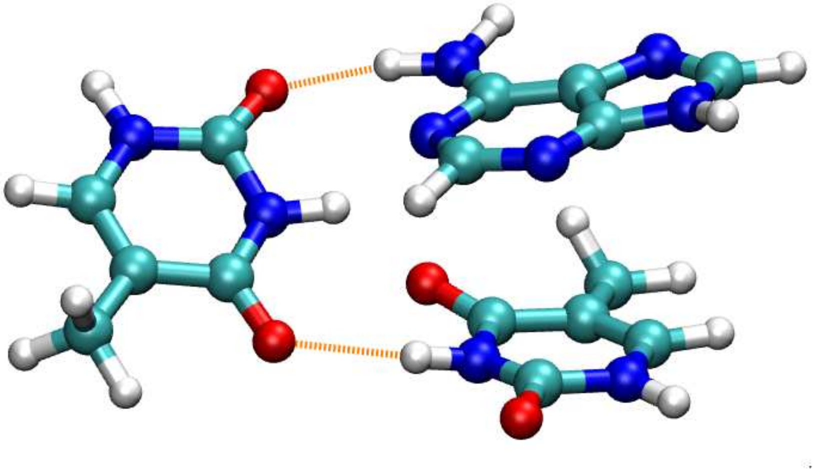
First is interaction of hydrogen in a base (mostly guanine, rarely cytosine) with the heterocyclic oxygen in the deoxyribose in the other strand (Figure 10). This pattern is found repeatedly in the melting regions of the poly(GC). It is surprisingly strong, we have found a structure where the base-sugar interaction amounts to -13 kcal/mol, an average (in structures selected by geometric criteria) is 7.5 kcal/mol. Such a structure could be an intermediate which makes the dissociation of strong hydrogen-bonded pair easier in terms of kinetics, because it divides the process in two energetically more feasible steps.

Figure 10: Deoxyribose-guanine hydrogen bonds found in stretched poly(GC)



Second is bridge-like hydrogen bonding, where one base (with high propeller twist value) is bonded to two bases from opposite strands (Figure 11). It is found at the ends of poly(AT). The interaction energy between base in one strand and two bases from the other can amount to -17 kcal/mol, what is more than an average interaction in Watson-Crick AT pair (~ -10 kcal/mol in the same simulation).

Figure 11: Interaction of a thymine with two bases from complementary strand in simulation of stretched poly(AT).



4.3 Free energy surface of GFA tripeptide

4.3.1 Introduction

The impulse to study free energy surface of isolated peptides came from the experiment. In a setup using multiple lasers, it is possible to measure IR spectra of different species in a mixture in gas phase. In peptides, it means different conformers. From combinatorial point of view, the number of possible conformers is enormous. In reality, only the conformers with lowest free energy exist. The infrared spectra, measured only in limited range (~ 3000 – 4000 cm^{-1}), give only indirect information on the structure. It is calculation what could assign structures to the spectra.

Simulation protocol[48] developed in our laboratory is based on MD/quenching technique, which produces structures for all minima populated at the conditions of the simulation. It was found that MM is not accurate enough, and DFTB-Ds selected as a relatively cheap method that yields very good results.

The structures with lowest energy are refined using high-level methods and smaller set of structures is finally selected. On these structures, harmonic frequencies are calculated. From this calculations, free energy of the conformers is estimated by rigid rotor / harmonic oscillator approximation. Vibrational spectra are then compared with experiment. This protocol was successfully used to resolve structures of phenylalanyl-glycyl-glycine (FGG) and tryptophyl-glycyl-glycine (WGG) tripeptides as well as tryptyophyl-glycine dipeptide (WG)[48, 49].

The same methodology was applied to the glycyl-phenylalanyl-alanine (GFA) tripeptide[50] (Appendix F). Structure of the species observed in the IR spectrum was successfully assigned on the basis of calculations.

In addition to the protocol mentioned above, we wanted to test another method of calculation that uses no preselection of the structures. Metadynamics based on the DFTB-D method should also cover anharmonicity of the low frequency vibrations, which contribute substantially to entropy of the molecule.

We know that the forcefield (AMBER ff99) is not able to predict stability of the conformers in agreement with experiment and higher level methods. To understand nature of this problem, we performed two metadynamics simulations using this forcefield, differing in atomic point charges. Both sets of charges were derived using the RESP procedure. First set was based on HF/6-31G* calculation (HF charges for short), a default level for this forcefield. However, this procedure was designed to derive parameters for simulations in condensed phase. For this reason, we also tested second set of charges based on B3LYP/cc-pVTZ calculation (DFT charges), which should be more appropriate in gas phase.

4.3.2 Simulation setup

The two internal coordinates defining the 2D FES were selected in order to distinguish between several families of conformations found in the previous analysis. One is Ramachandran angle φ of the alanine residue, the second coordinate *d* is a distance describing formation of hydrogen bond between hydrogen of the carboxyl terminal group and oxygen in the phenylalanine residue.

MM simulations were 5 ns long, with bias potential update every 0.5 ps. Parameters of the Gaussians composing the bias potential were: height 0.05 kcal/mol, widths in the internal coordinates were 0.3 rad and 0.6 Å.

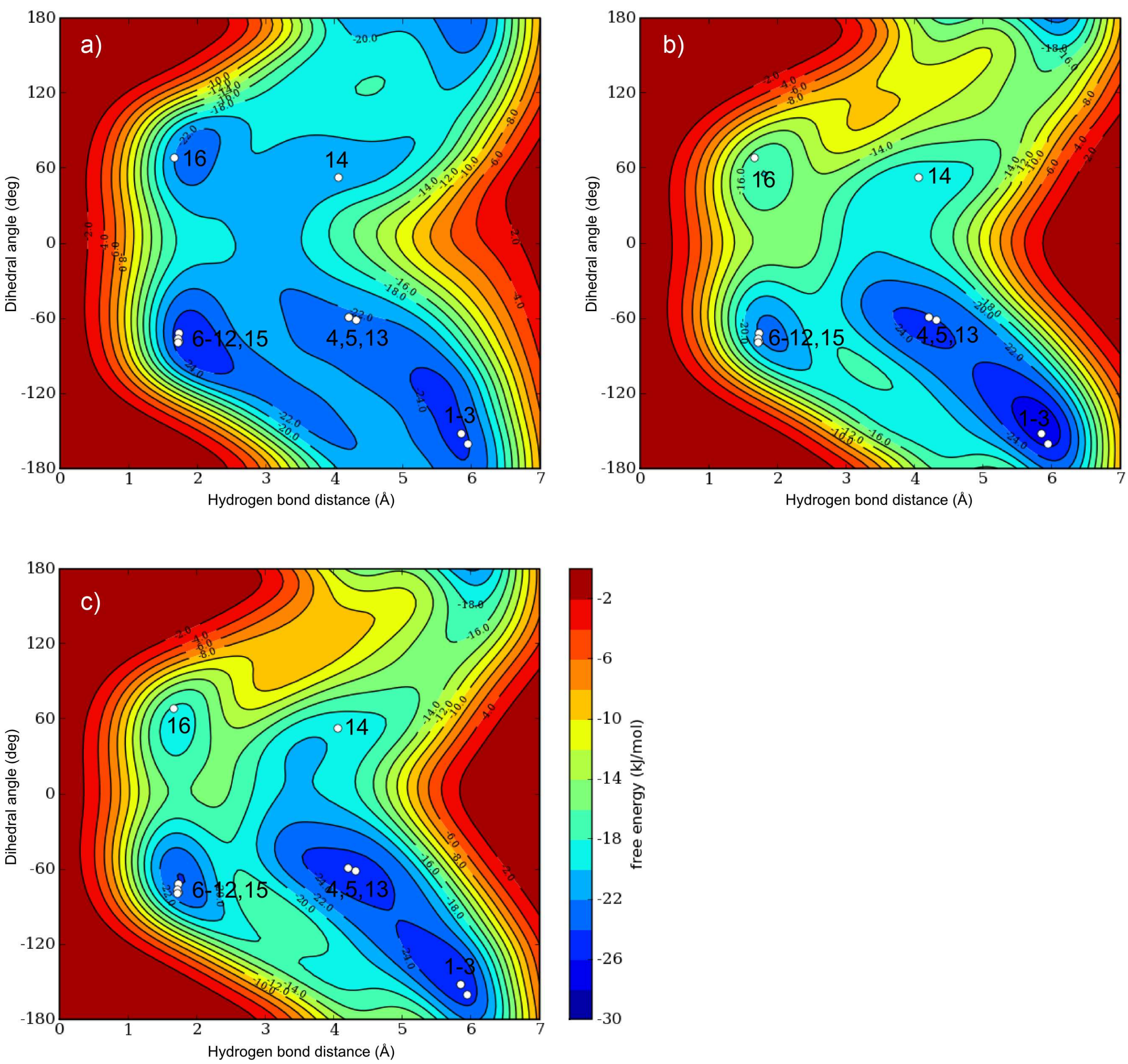
DFTB-D simulation was 600 ps long, bias potential was updated every 0.1 ps. Widths of the Gaussian were the same, its height was variable: 0.1 kcal/mol in first 500 ps of the simulation for faster convergence, and 0.05 kcal/mol in the end of the simulation for higher accuracy.

4.3.3 Results and Discussion

Free running simulations of the tripeptide would require very long time to be useable for thermodynamic analysis. Using the metadynamics, we can sample selected internal coordinates with much better efficiency.

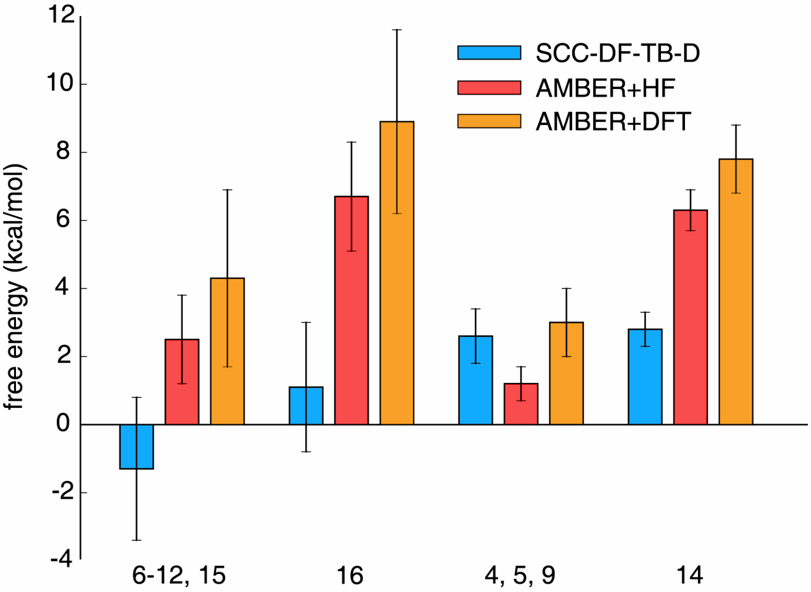
Results of the simulations were visualized as free energy maps with isoenergetic contours. Points corresponding to the structures from rigid rotor/harmonic oscillator calculations were included in the plots for comparison. Results from DFTB-D and AMBER simulations are presented in Figure 12

Figure 12: Free energy surface of GFA tripeptide calculated with DFTB-D (a), AMBER ff99 with DFT (b) and HF (c) charges. Structures from conformational search are marked by white points.



Free energy of the minima was recorded in converged part of the simulation to estimate the error (using 95 % confidence interval). Free energies of the minima with error bars are plotted in Figure 13.

Figure 13: Free energy of minima on FES from tetadynamics simulations. Values are relative to free energy of minimum corresponding to structures 1-3.



The DFTB-D FES is in good agreement with the higher level calculation of selected conformers. An important feature is the description of the balance of structures with and without studied intramolecular hydrogen bond. Here the H-bonded structures are found about 3 kcal/mol more stable, what is in contrast to our experience with the DFTB-D method, which generally underestimates hydrogen bonds. This might be only an artifact of the simulation; this discrepancy is within the error bars (Figure 13). Also, the FES is averaged in all other dimensions than the selected internal coordinates, what makes it impossible to directly compare to the single point calculations. There is no minimum on the map for structured 04, 05 and 13, but this is probably only artifact of the low resolution of the FES map.

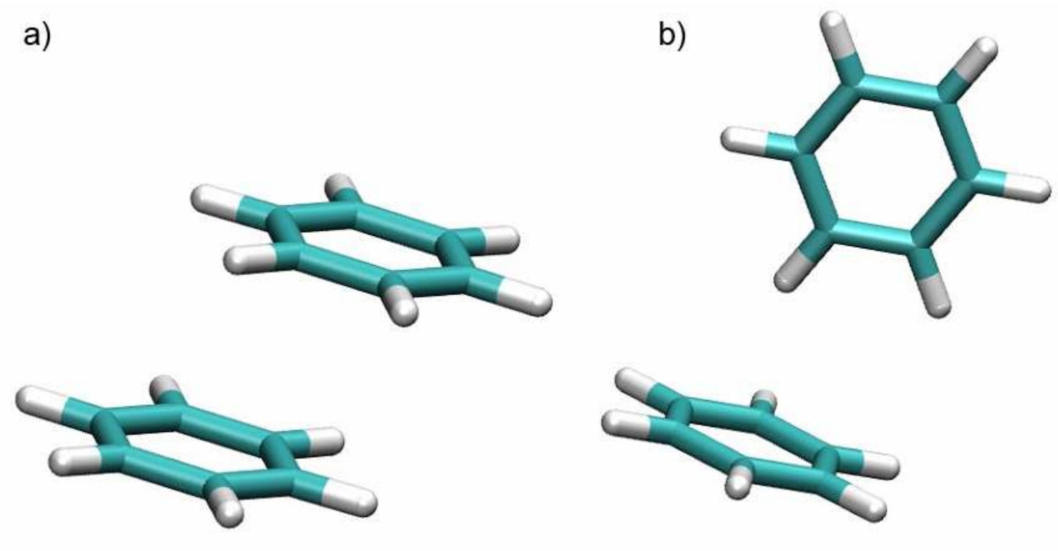
The AMBER ff99 free energy surfaces (Figure 12b, c) are similar. It is clear that DFT charges underestimate hydrogen bonding, where HF charges yield good results because they partially compensate for the missing polarization. Both surfaces are, however, biased towards negative values of the dihedral angle φ. This bias makes some of the studied structures energetically inaccessible.

4.4 Dynamics of benzene dimer

4.4.1 Introduction

Benzene dimer has been thoroughly studied for long time, but it remains to be interesting topic even now. It serves as a model system for interaction of aromatic molecules, which play important role in biochemistry, organic chemistry and nanomaterials science. The benzene dimer might look simple at first, but it is surprisingly complex. There exist two minima on potential energy surface, corresponding to different geometries: parallel displaced stacked structure (Figure 14a) and tilted T-shaped structure (Figure 14b). In recent theoretical studies[5, 78-82], it was shown that these two minima are practically isoenergetic. It must be also noted that only the highest levels of theory are accurate enough to describe the two different structures of different nature with required accuracy. Energetic barriers separating the minima are very low, what allows rapid interconversion even at low temperatures. This dynamic structure can hardly be described by a static method.

Figure 14: Parallel displaced (a) and tilted T-shaped (b) structure of benzene dimer



However, not only theoreticians face difficulties when dealing with benzene dimer. Infrared spectrum of this system is also complex. Due to high symmetry of the benzene molecule, only some vibrational modes are visible. Moreover, the C-H stretching modes are strongly coupled. Effect of dimerization on the C-H modes was subject of many experimental and theoretical studies. At the beginning, theory predicted[83] blue shift of the C-H mode involved in the improper hydrogen bond in T-shaped structure of the dimer. This effect was later experimentally confirmed in many similar systems[84], but in benzene dimer, red shift was observed[85]. The structure of the dimer can't be questioned, existence of the T-shaped structure was experimentally proven[85, 86]. Recent theoretical work of Wang[87] resolved this discrepancy: Blue shift would be found in fully symmetrical (c_{2v}) T-shaped structure, but it was found to be a transition state rather than global minimum. The energetic minimum corresponds to tilted T-shaped geometry with c_s symmetry, where theory predicts, in agreement with experiment, red shift.

In our work on benzene dimer[6] (Appendix J), we attempted to merge two important, but often contrasting, approaches. Accurate quantum chemical calculations are expensive and only small number of points can be calculated. Molecular dynamics, which would naturally describe this dynamic system, require calculation of very large number of points, and is thus limited to the most efficient methods, mostly molecular mechanics (MM). Recent development in methodology as well as in computer hardware led recently to wider applicability of on-the-fly *ab initio* molecular dynamic, a classical molecular dynamics based on potential calculated in each point using QM method. However, conventional methods, effective enough to be used in on-the-fly MD, do not have the accuracy needed for description of the benzene dimer. The only way to achieve required accuracy with DFT-D method was to derive specific set of parameters for benzene dimer (See above, chapter 3.3, page 16).

4.4.2 Strategy of calculations

The on-the-fly MD simulations based on DFT-D were run according to the setup described in chapter 3.3 (page 16). 1 fs timestep was used; Andersen thermostat algorithm with mean collision frequency 2.5 ps^{-1} was used to regulate the temperature.

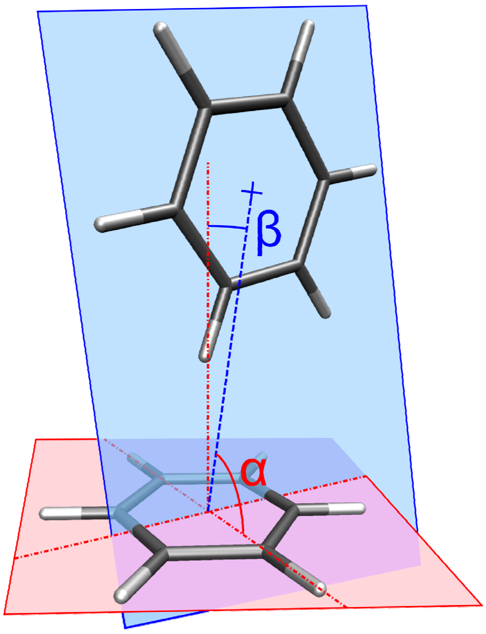
Using this method, we ran simulations covering temperatures from 10 to 100 K (with step of 10 K). Eight 20 ps simulations were performed at each temperature, half starting from TS and half from PD structure.

To eliminate the effect of initial conditions on the trajectories, we calculated autocorrelation function of displacement coordinates (difference from the average geometry) in each set of calculations. These functions were just crude approximation due to limited number of the simulations, but it was found that the correlation decays in about 5 ps. We decided to discard firs 10 ps of the trajectories, while the remaining 10 ps was used in the analysis.

The trajectories themselves present large amount of data. For analysis of the structure, they have to be reduced to meaningful variables. Firstly, we have to distinguish between PD and TS structure. This is described by angle between the ring planes α (0° for PD and 90° for TS structure). Secondly, we also examine the tilt angle β (0° in the symmetric transition state) in the T-shaped structure. Although this angle is not independent on α, it allows identification of the symmetrical c_{2v} transition state. Definition of these angles is pictured in Figure 15. Finally, we look at distance of the two monomers (their centers of mass), because this value is readily comparable with experiment.

These parameters from each set of trajectories with the same initial conditions were then processed into probability distribution functions (histograms). This form of visualization allows to assess occupation of the respective states during the simulations.

Figure 15: Definition of angle between benzene ring planes a and tilt angle in T-shaped structure β.



4.4.3 Results and Discussion

Benzene dimer at 0 K

Before we look at the MD results, our knowledge of benzene dimer based on static calculations should be reviewed. Recent studies[5, 78-82] employing high-level quantum-chemical methods show that the TS and PD structures are practically isoenergetic. Stabilization of the TS amounts to about 0.1 kcal/mol, what is at the edge of the accuracy of these methods. Our DFT-D method is parametrized to reproduce these data.

Unfortunately, this accuracy is lost when we move from electronic energy to enthalpy at 0 K. The additional term, the zero point vibration energy (ZPVE), is yet to be calculated at a reliable level. Harmonic calculations using MP2[88] or our customized DFT-D turn the balance in favor of parallel displaced structure (-0.15 kcal/mol using our potential). This finding is in contrast to the experiment, where T-shaped structure is observed. There are two possible explanations: Firstly, we are using crude approximation in our calculation of vibrational frequencies. Proper anharmonic calculation could yield different results. Such a calculation is yet to be done; neither previous attempt by Wang and Hobza[87], nor our calculation using DFT-D, was fully successful. We plan to address this issue in future by separate nonharmonic analysis of the intermolecular modes. Secondly, the experimental conditions are not completely known. In the jet cooling experiments, the gas at higher temperature (where TS is prevalent, as is shown later) is instantly cooled to temperatures close to zero. There is not enough time for relaxation of the structure, and the population may correspond to higher temperature than the actual temperature after the cooling is.

Balance between parallel displaced and T-shaped structure

Histograms of the plane angle α at the studied temperatures are plotted separately for simulations starting from T-shaped (Figure 16) and parallel-displaced (Figure 17) minima.

Figure 16: Histograms of the plane angle a insimulations strating from T-shaped structure.

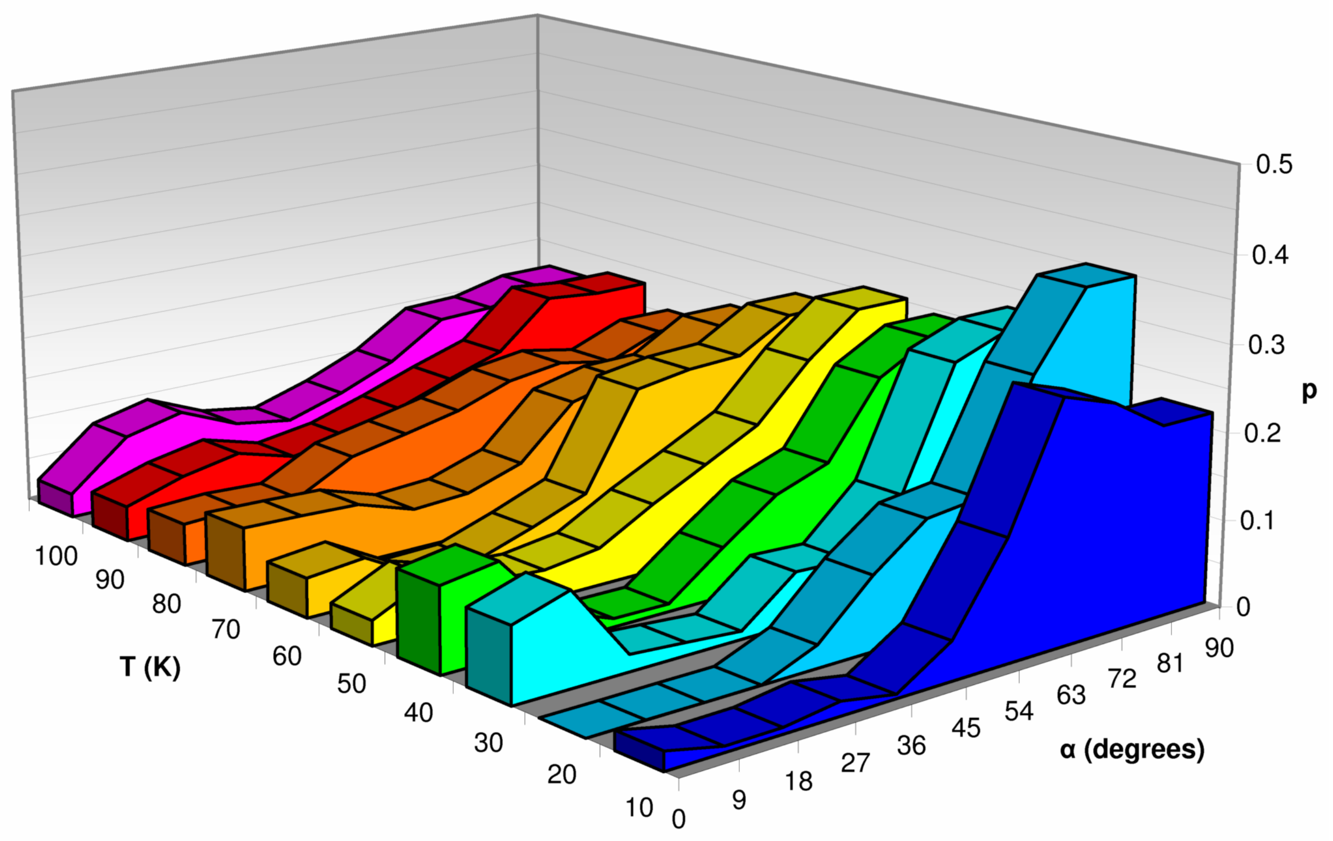
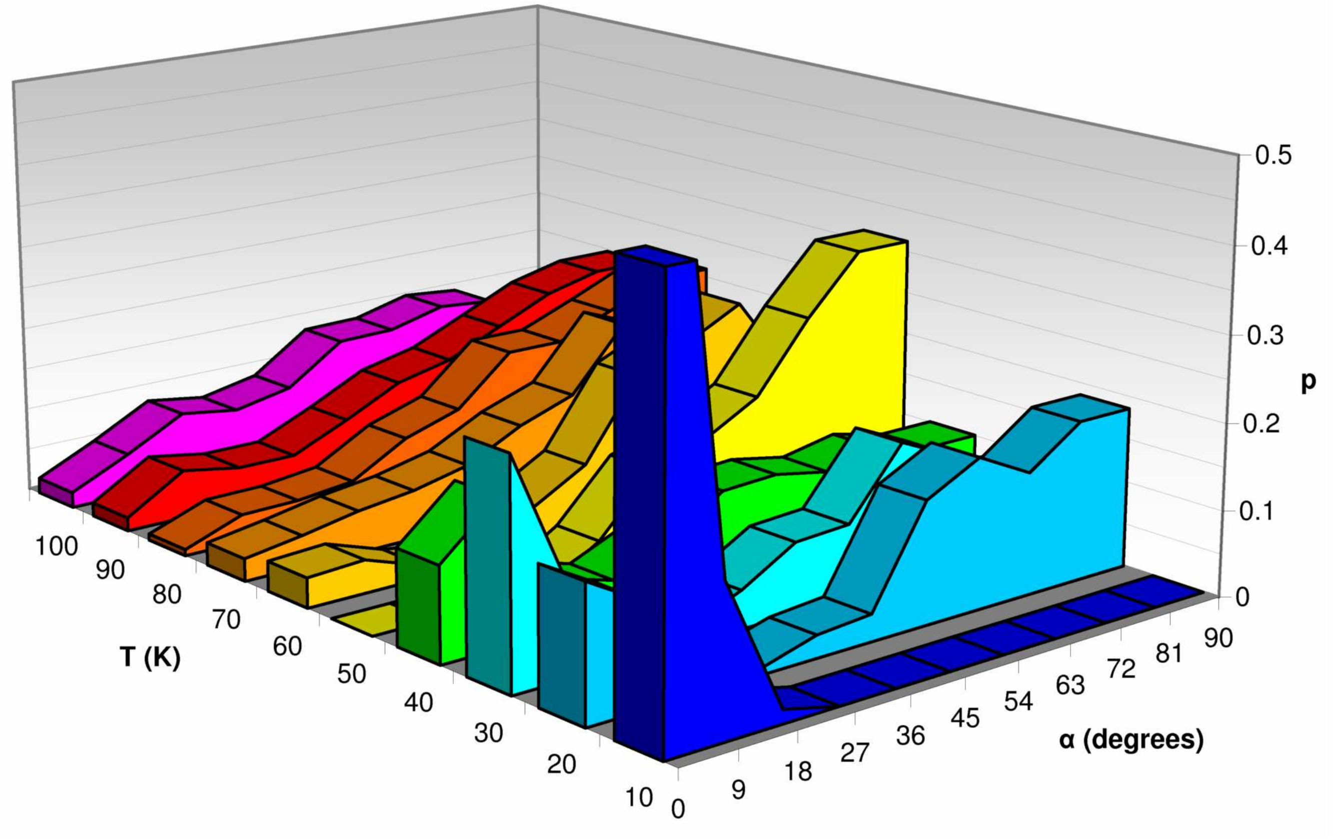


Figure 17: Histograms of the plane angle a insimulations strating from parallel displaced structure.



Significant difference between these two plots can be observed only at the lowest temperatures up to 20 K. There is not enough energy in the system to overcome the barrier separating TS and PD structures, and the system stays in the minima it started from.

At higher temperatures up to 40 K, there are well resolved peaks corresponding to the two minima. Interconversion occurs in the simulations, the distribution does not depend on the starting structure. At these temperatures, the two structures coexist, although TS minimum is more populated.

At higher temperature, the structure becomes very dynamic; the interconversion takes place very often. While there always is some fractions of PD structures, the TS becomes dominant.

This temperature dependence is a result of entropic effects. The PD structure is more rigid, while in the TS, the system is more flexible. Firstly, rotation of the horizontal (in Figure 15) benzene molecule is almost free. Secondly, the mode described by the α angle is very soft and has large amplitude. This is also visible from the plots – the peak of the TS structure is very broad.

The entropy can be estimated in a more rigorous way from equilibrium constant (K), which is defined as a ratio of the TS and PD structures. For this analysis, the border between the PD and TS structures should be defined. We divided the range into halves and set the threshold to 45°. From the equilibrium constant, we can readily obtain the free energy difference

$$\Delta G = -RT \ln K \,. \qquad (4)$$

From a linear regression of the temperature dependence of ΔG, we can extrapolate to 0 K to obtain ΔH^0. We found the TS structure disfavored by 0.035 ± 0.05 kcal/mol at 0 K. From the slope of the fitted line, we obtained an entropy difference ΔS = 0.004 ± 0.001 kcal/mol/K. At 100 K, the stabilization of the TS structure compared to PD one (ΔG^{100}) thus amounted to -0.4

kcal/mol. Although such an analysis based on limited number of simulations is not very accurate, as indicated by the error bars, it can definitely show the trends.

This result is extremely important for the interpretation of the experiments (PD structure was never observed), because it shows that the T-shaped conformation is more populated even at low temperatures and prevalent at high temperatures (which is important even in experiments at low temperature, as discussed above).

Tilt in T-shaped structure

The parameter studied so far does not distinguish between the tilted T-shaped structure (c_s symmetry) and the symmetrical transition state (c_{2v}). This is an important feature, because both structures have different signatures in IR spectrum. The symmetric structure would have blue-shifted CH stretching mode, while in the tilted minimum, red shift is predicted. For this reason, we also measured the overall tilt angle β in simulations at lower temperature, where the T-shape is conserved. In the optimized structure, this angle is 7.4°. In simulations at 10 K, the angle distribution ranges from 3° to 16° with maximum around 10°. At higher temperatures, there is some population of structures with tilt equal or close to zero, because the dimer passes the transition state, but their fraction is negligible. This is in perfect agreement with the experiment, where red shift is measured in structure with nonequivalent benzenes (what must be T-shaped, not PD structure).

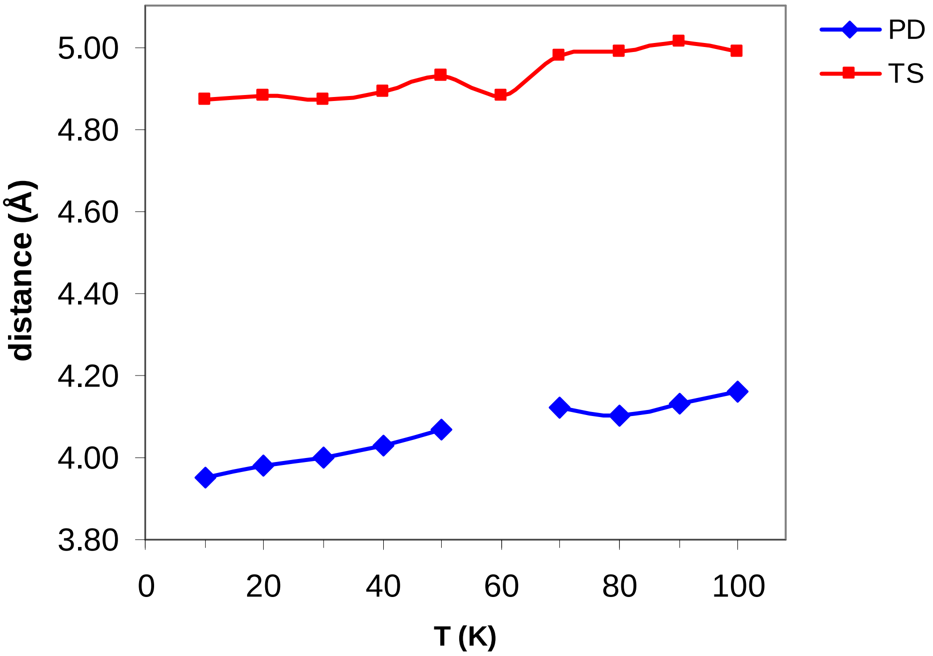
Center of mass distance

Another feature we looked at is the distance of the two benzene rings, represented by their centers of mass. This value is experimentally accessible, it can be derived from measured rotational constants. According to Ref. [86], the distance in TS structure is 4.90 ± 0.01 Å. Geometry optimization using our DFT-D potential yields distance 4.90 Å for TS and 3.90 for PD conformation.

In the MD simulations, we have measured this distance and sorted the structures into TS and PD according to this distance. The threshold for distinguishing TS and PD structures was set to 4.4 Å. In resulting sets, average distance was calculated. It is plotted in Figure 18 as a function of temperature. The distance increases with temperature (more in PD, less in TS), because this intermolecular vibrational mode is anharmonic.

Again, distance 4.88 Å obtained for TS structure at low temperatures is in perfect agreement with experimentally measured value.

Figure 18: Center of mass distance in parallel displaced (PD) and tilted T-shaped (TS) structures of benzene dimer as a function of temperature in simulations (missing point corresponds to simulations where PD structure was not detected in the analyzed part of trajectory).



4.5 DFT-D in study of adsorption on water surface

DFT-D method was also employed in joint experimental and theoretical study of adsorption of aromatic compounds on water surface[8] (Appendix D). *Ab initio* MD simulation based on DFT-D was used to verify results obtained from MM simulations.

Ab initio MD simulation were performed using DFT implementation combining Gaussian atomic orbitals with plane wave auxiliary basis set (GPW)[89] in Quickstep program, which is a part of the CP2K package[90]. The BLYP functional with the double zeta valence polarized (DZVP) basis set was used. The energy cutoff for plane waves was set to 280 Ry and the Goedecker-Teter-Hutter pseudopotentials[91] were applied.

Since the Quickstep code is designed for simulations in periodic boundary condition and calculations of isolated molecules is thus difficult, the dispersion parameters originally derived for similar basis set 6-31G** were used. In our tests, BLYP/DZVP calculation with these dispersion parameters give better (closer to benchmark CCSD(T)/CBS data for benzene-water and pyridine-water clusters) than the original BLYP/6-31G**. The code was modified to call external program that calculates the dispersion correction. This correction was calculated in custom script using the libraries of the Cuby code.

In the simulations, angle between the plane of aromatic molecule and surface of the water slab was measured. Benzene prefers values around 0°, which correspond to orientation parallel to the water surface. Pyridine, due to the possibility of formation of hydrogen bond with water, prefers orientation perpendicular to the water slab.

These findings help to understand adsorption of aromatic compounds on water surface and shed light on strength and nature of these interactions. These systems are studied as a model for reactions of aromatic molecules in atmosphere, which could take place on surface of water droplets.

5 Conclusions

5.1 Noncovalent interactions and DNA stability

We have shown that it is possible to calculate large number of structures using DFT-D method with results within 1 kcal/mol from benchmark CCSD(T) data. Performance of the other two methods tested, semiempirical DFTB-D and AMBER ff99 forcefield, is substantially worse. The error is systematic, in both cases hydrogen bonds are underestimated. The correlation with DFT-D data is, however, good, and the methods can be used in application where the absolute values are not needed, such as our statistical model estimating DNA stability.

In the interaction energies, surprising and very strong complementarity was found between interstrand and intrastrand stacking when results for 128 octamers were studied. Sum of interaction energy of these two component was almost constant, with mean deviation 3 %.

We have shown that to get good correlation with experimentally measured stability of DNA, it is necessary to take into account all the studied contributions – hydrogen bonds as well as intra- and interstrand stacking. Hydrogen bonding term must be corrected for the effects of solvation; this reduces significantly the difference between AT and CG pairs.

Statistical model, weighting these contributions individually and fitted to experimental data can predict ΔG of DNA duplex dissociation with good accuracy, which is comparable to empirical nearest neighbor models. However, our model uses an order of magnitude less fitted parameters, and it is based on real properties calculated on the DNA structure. We developed this model for octamers and then extended it to oligomers of variable length.

Weighting coefficients obtained from this model can be used as a measure of importance of these contributions. Terms that correlates with intrastrand stacking was found to be most important, it is the interaction that is crucial for the double helical structure of DNA. Hydrogen bonding is second important contribution, it is, however, compensated by solvation of the interacting sites upon dissociation.

We attempted to extend our study to sequences containing unnatural base inosine, which pairs with all four natural bases in DNA. In this case, we were not able to create model able to predict stabily of these oligomers, although we added another term describing deformation energy of the DNA backbone. This failure could be attributed to the fact that our model depend on including omitted variables, such as entropy, into the fitted constants. This works in regular natural DNA, but not in structures perturbed by the unnatural base. Proper calculation of these additional terms is, however, impossible.

5.2 Structure of stretched DNA

Calculation of interaction energies on structures from MD simulations gave us insight into the mechanism of extension of DNA under mechanical stress. The extension of the DNA is not homogenous, there are regions where the extension is localized and the DNA is melted, and regions where the B-DNA structure is conserved.

Different mechanism was observed in poly(AT) and poly(GC) oligomers. It can be explained by the differences in noncovalent interactions between the bases, the potential energy of the backbone does not change significantly, because the double helix is locally unwound.

In poly(GC), the H-bonds are stronger and first step of the process is unstacking of CG steps. After that, water, as well as unusual H-bonding patterns discussed above, facilitate dissociation of the H-bonded pairs. In AT, the hydrogen bonds dissociate more easily and this process starts from the ends. These bases are also less hydrophilic and we observe more aggregation due to their stacking.

5.3 Free energy surface of GFA tripeptide

Metadynamics has proven itself to be efficient method for study free energy surfaces in systems, where we can define internal coordinates distinguishing between studied structures, what is true in the GFA peptide. The results are generally in good agreement with the rigid rotor/harmonic oscillator estimates of free energy previously used for the peptide.

The DFTB-D simulation was rather short, what might question its reliability in the balance between hydrogen bonded and free structures. On the other hand, it yields good description of the dihedral angle φ, qa coordinate where the forcefield fails.

There are new modifications to the forcefields, which should improve the description of backbone dihedral angles, but, as we have shown in our analysis of the database of peptide structures, none of them is reliable enough to properly describe energies of the conformers.

5.4 Dynamics of benzene dimer

For the purpose of this study, we have derived custom parameters for the dispersion correction in the DFT-D scheme, which the perfectly mimics CCSD(T)/CBS benchmark calculations.

This accurate and efficient potential allowed us to perform multiple on-the-fly am initio MD simulations in total length of 1.6 ns.

While the potential itself favors T-shaped structure by about 0.1 kcal/mol, when ZPVE is added, parallel displaced structure becomes more stable by 0.15 kcal/mol. This is in conflict in experiment, which detects T-shaped geometry at very low temperatures. This can be caused by the obvious limitations of the harmonic calculation of frequencies and ZPVE. Results of the experiment are not completely persuading too. Due to fast cooling of the complex, the structure could be conserved from equilibrium at higher temperature.

In the MD simulations at temperatures above 20 K, we observed mixture of both PD and TS species. With increasing temperature, the TS structure becomes dominant, because it is stabilized by entropy.

The T-shaped structure was confirmed to be tilted. This structure exhibits a red shift of one C-H stretch mode, what is in agreement with IR spectroscopy.

Center of mass distance of the benzene moieties is in perfect agreement wit experimental value. The distance increases with temperature, what shows significant anharmonicity in this intramolecular mode.

5.5 DFT-D in QM/MM study of carborane inhibitors of HIV protease

Using the DFT-D within QM/MM scheme, it was possible to get the information missing in experimental structure of HIV-1 protease in complex with metallocarborane inhibitor.

The size of the QM region is rather large – up to 250 atoms. The calculation was possible using resolution of identity approximation, B-LYP functional and small SVP basis set. 1100 atoms was optimized in the calculation of each examined rotamer.

These calculations were used to determine orientation of the inhibitor in active site of the enzyme. Several energetically accessible rotamers of the carborane cages were identified. Konwledge of the structure and energetics of binding of the inhibitor to the enzyme help to understand its activity and could be used to improve the inhibitor in future.

5.6 DFT-D in study of adsorption on water surface

DFT-D method was implemented in Quickstep GPW code used for simulations of condensed phase. The 12 ps DFT-D simulation of a slab of water built from 72 water molecules with adsorbed aromatic molecule (benzene and pyridine) confirmed results of longer MM simulations of larger model systems.

It was found that benzene prefers orientation parallel to the water surface, while pyridine molecule prefers perpendicular position stabilized by hydrogen bond with water.

Without the dispersion, the benzene detached from the surface in DFT simulation in first ten femtoseconds of the simulation.

6 Future plans

This work has shown possibilities of modern efficient quantum-mechanical methods in the study of noncovalent interaction. We of course continue to work on this topic and some of the projects are direct extension of the work presented here.

As it was mentioned above, the benzene dimer deserves closer look at its intermolecular vibration. Because conventional methods failed to describe the problem, we are going to apply proper quantum-mechanical treatment to these modes, what would require numerical construction of accurate intermolecular potential.

The on-the-fly *ab initio* dynamics is also very promising method. With growing power of computers, it can be applied to more chemical problems in near future. We are currently working on several projects using DFT-D based simulations to study isolated molecules and molecular complexes in gas phase. We would also like to continue with more accurate simulations based on the DFT-D method.

The metadynamics is a promising method for studying free energy surfaces. It is efficient enough to use semiempirical methods for potential calculation, and this combination is superior to molecular mechanical simulations. We plan to extend its use to molecular complexes e.g. base pairs in gas phase.

At present, we are also testing new semiempirical methods with empirical dispersion correction, such as the recently introduced OM3-D[92]. These methods could be valuable tool in applications where high efficiency is required.

7 List of publications

1. Jan Řezáč, Pavel Hobza
   On the nature of DNA-duplex stability
   Chemistry - A European Journal, 13 (10), 2983-2989, 2007
   Attached as Appendix A

2. Tomáš Kubař, Petr Jurečka, Jiří Černý, Jan Řezáč, Michal Otyepka, Haydee Valdes, Pavel Hobza
   Density-functional, density-functional tight-binding, and wave function calculations on biomolecular systems
   Journal of Physical Chemistry A, 111 (26), 5642-5647, 2007
   Attached as Appendix B

3. Jan Řezáč, Pavel Hobza
   Correlation Between the Thermodynamic Stability of DNA Duplexes and the Interaction and Solvation Energies of DNA Building Blocks
   Collection of Czechoslovak Chemical Communications 73 (2), 175-186, 2008
   Attached as Appendix C

4. Robert Vácha, Lukasz Cwiklik, Jan Řezáč, Pavel Hobza, Pavel Jungwirth, Kalliat Valsaraj, Stephan Bahr and Volker Kempter
   Adsorption of Aromatic Hydrocarbons and Ozone at Environmental Aqueous Surfaces
   Journal of Physical Chemistry A 112 (22), 4942-4950, 2008
   Attached as Appendix D

5. Haydee Valdes, Kristýna Pluháčková, Michal Pitonák, Jan Řezáč, Pavel Hobza
   Benchmark database on isolated small peptides containing an aromatic side chain: comparison between wave function and density functional theory methods and empirical force field
   Physical Chemistry Chemical Physics 10, 2747 - 2757, 2008
   Attached as Appendix E

6. Haydee Valdes, Vojtěch Spiwok, Jan Řezáč, David Řeha, Ali G. Abo-Riziqd, Mattanjah S. de Vries, Pavel Hobza
   Potential energy and free energy surfaces of glycyl-phenyalanyl-alanine (GFA) tripeptide: experiment and theory
   Chemistry - A European Journal 14, 4886-4898, 2008
   Attached as Appendix F

7. Jonathan Mitchell, Hlengisizwe Ndlovu, Jan Řezáč, Pavel Hobza, Sarah Harris
   Denaturing DNA in silico
   Journal of the Royal Society Interface (submitted)
   Attached as Appendix G

8. Jan Řezáč, Pavel Hobza, Sarah Harris
   Sequence dependence of the stretching behaviour of DNA investigated by molecular dynamics simulation
   (in preparation)

9. Jindřich Fanfrlík, Martin Lepšík, Jan řezáč, Jiří Brynda, Pavel Hobza
   QM/MM calculations refine the crystal structure of HIV-1 protease-metallocarborane complex
   Chemistry – Journal of physical chemistry (submitted)
   Attached as Appendix H

10. Pavlína Řezáčová, Jiří Brynda, Milan Kožíšek, Petr Cígler, Martin Lepšík, Jindřich Fanfrlík, Jan Řezáč, Jana Pokorná, Klára Grantz-Šašková, Irena Sieglová, Jaromír Plešek, Bohumír Grüner, Václav Šícha, Hans-Georg Kraeusslich, Vladimír Král, Jan Konvalinka
    Structure-based design of HIV protease inhibitors based on inorganic polyhedral metallacarboranes
    J. Am. Chem. Soc. (in preparation)

11. Michal Pitoňák, Pavol Neogrády, Jan Řezáč, Petr Jurečka, Miroslav Urban, Pavel Hobza
    Benzene dimer: High-level wavefunction and density functional theory calculations
    Journal of Chemical Theory and Computation (submitted)
    Attached as Appendix I

12. Jan Řezáč, Pavel Hobza
    Benzene dimer: dynamic structure and thermodynamics derived from on-the-fly ab initio DFT-D molecular dynamic
    Journal of Chemical Theory and Computation (submitted)
    Attached as Appendix J

8 Reference List

[1] J. Řezáč, P. Hobza. *Chem. Eur. J.* **2007**, *13*(10), 2983-2989.

[2] J. Řezáč, P. Hobza. *Collection of Czechoslovak Chemical Communications* **2008**, *73*(2), 161-174.

[3] H. Valdes, V. Spiwok, J. Řezáč, D. Řeha, A. G. bo-Riziq, M. S. de Vries, P. Hobza. *Chemistry.* **2008**, *14*(16), 4886-4898.

[4] H. Valdes, K. Pluháčková, M. Pitoňák, J. Řezáč, P. Hobza. *Phys. Chem. Chem. Phys.* **2008**, *10*(19), 2747-2757.

[5] Pitoňák, M., Neogrády, P., Řezáč, J., Jurečka, P., Urban, M., and Hobza, P. Benzene dimer: High-level wavefunction and density functional theory calculations. Journal of Chemical Theory and Computation (submitted).

[6] Řezáč, J. and Hobza, P. Benzene dimer: dynamic structure and thermodynamics derived from on-the-fly ab initio DFT-D molecular dynamic. Journal of Chemical Theory and Computation (submitted).

[7] Fanfrlík, J., Brynda, J., Řezáč, J., Hobza, P., and Lepšík, M. Interpretation of Protein/Ligand Crystal Structure using QM/MM Calculations: Case of HIV-1 Protease/Metallacarborane Complex. Journal of the American Chemical Society (submitted).

[8] R. Vácha, L. Cwiklik, J. Řezáč, P. Hobza, P. Jungwirth, K. Valsaraj, S. Bahr, V. Kempter. *J. Phys. Chem. A* **2008**, *112*(22), 4942-4950.

[9] Ruby programming language, www.ruby-lang.org.

[10] Y. Zhao, D. G. Truhlar. *Theoretical Chemistry Accounts* **2008**, *120*(1-3), 215-241.

[11] R. Ahlrichs, R. Penco, G. Scoles. *Chemical Physics* **1977**, *19*(2), 119-130.

[12] J. Hepburn, G. Scoles, R. Penco. *Chemical Physics Letters* **1975**, *36*(4), 451-456.

[13] P. Hobza, F. Mulder, C. Sandorfy. *Journal of the American Chemical Society* **1981**, *103*(6), 1360-1366.

[14] P. Hobza, F. Mulder, C. Sandorfy. *Journal of the American Chemical Society* **1982**, *104*(4), 925-928.

[15] P. Hobza, C. Sandorfy. *Canadian Journal of Chemistry-Revue Canadienne de Chimie* **1984**, *62*(3), 606-609.

[16] P. Hobza, C. Sandorfy. *Journal of the American Chemical Society* **1987**, *109*(5), 1302-1307.

[17] E. J. Meijer, M. Sprik. *Journal of Chemical Physics* **1996**, *105*(19), 8684-8689.

[18] W. T. M. Mooij, F. B. van Duijneveldt, van Duijneveldt-van de Rijdt, B. P. van Eijck. *Journal of Physical Chemistry A* **1999**, *103*(48), 9872-9882.

[19] Q. Wu, W. T. Yang. *Journal of Chemical Physics* **2002**, *116*(2), 515-524.

[20] X. Wu, M. C. Vargas, S. Nayak, V. Lotrich, G. Scoles. *Journal of Chemical Physics* **2001**, *115*(19), 8748-8757.

[21] U. Zimmerli, M. Parrinello, P. Koumoutsakos. *Journal of Chemical Physics* **2004**, *120*(6), 2693-2699.

[22] S. Grimme. *Journal of Computational Chemistry* **2004**, *25*(12), 1463-1473.

[23] P. Jurečka, J. Černý, P. Hobza, D. R. Salahub. *J. Comput. Chem.* **2007**, *28*(2), 555-569.

[24] W. G. Hoover. *Physical Review A* **1985**, *31*(3), 1695-1697.

[25] S. Nose. *Molecular Physics* **1984**, *52*(2), 255-268.

[26] H. C. Andersen. *Journal of Chemical Physics* **1980**, *72*(4), 2384-2393.

[27] P. Pulay, G. Fogarasi. *Chemical Physics Letters* **2004**, *386*(4-6), 272-278.

[28] A. Laio, M. Parrinello. *Proceedings of the National Academy of Sciences of the United States of America* **2002**, *99*(20), 12562-12566.

[29] M. Iannuzzi, A. Laio, M. Parrinello. *Physical Review Letters* **2003**, *90*(23).

[30] S. Piana, A. Laio. *Journal of Physical Chemistry B* **2007**, *111*(17), 4553-4559.

[31] H. J. C. Berendsen, D. Vanderspoel, R. Vandrunen. *Computer Physics Communications* **1995**, *91*(1-3), 43-56.

[32] M. Elstner, P. Hobza, T. Frauenheim, S. Suhai, E. Kaxiras. *Journal of Chemical Physics* **2001**, *114*(12), 5149-5155.

[33] P. Cigler, M. Kozisek, P. Rezacova, J. Brynda, Z. Otwinowski, J. Pokorna, J. Plesek, B. Gruner, L. Doleckova-Maresova, M. Masa, J. Sedlacek, J. Bodem, H. G. Krausslich, V. Kral, J. Konvalinka. *Proceedings of the National Academy of Sciences of the United States of America* **2005**, *102*(43), 15394-15399.

[34] J. Fanfrlik, M. Lepsik, D. Horinek, Z. Havlas, P. Hobza. *Chemphyschem* **2006**, *7*(5), 1100-1105.

[35] J. Fanfrlik, D. Hnyk, M. Lepsik, P. Hobza. *Physical Chemistry Chemical Physics* **2007**, *9*(17), 2085-2093.

[36] D. A. Pearlman, D. A. Case, J. W. Caldwell, W. S. Ross, T. E. Cheatham, S. Debolt, D. Ferguson, G. Seibel, P. Kollman. *Computer Physics Communications* **1995**, *91*(1-3), 1-41.

[37] Case, D. A, Darden, T. A, Cheatham, T. A, Simmerling, C. L., Wang, J., Duke, R. E., Luo, R., Merz, K. M., Pearlman, D. A., Crowley, M., Walker, R. C., Zhang, W., Wang, B., Hayik, S., Roitberg, A., Seabra, G., Wong, K. F., Paesani, F., Wu, X., Brozell, S., Tsui, V., Gohlke, H., Yang, L., Tan, C., Mongan, J., Hornak, V., Cui, G., Beroza, P., Mathews, H. D., Schafmeister, C., Ross, W. S., and Kollman, P. A. AMBER 9, University of California, San Francisco. **2006**.

[38] R. Ahlrichs. Turbomole. **1989**.

[39] S. Dapprich, I. Komaromi, K. S. Byun, K. Morokuma, M. J. Frisch. *Journal of Molecular Structure-Theochem* **1999**, *462*, 1-21.

[40] M. Svensson, S. Humbel, R. D. J. Froese, T. Matsubara, S. Sieber, K. Morokuma. *Journal of Physical Chemistry* **1996**, *100*(50), 19357-19363.

[41] T. Vreven, K. Morokuma, O. Farkas, H. B. Schlegel, M. J. Frisch. *Journal of Computational Chemistry* **2003**, *24*(6), 760-769.

[42] J. Cerny, P. Jurecka, P. Hobza, H. Valdes. *Journal of Physical Chemistry A* **2007**, *111*(6), 1146-1154.

[43] P. Jurecka, J. Sponer, J. Cerny, P. Hobza. *Physical Chemistry Chemical Physics* **2006**, *8*(17), 1985-1993.

[44] A. Schafer, C. Huber, R. Ahlrichs. *Journal of Chemical Physics* **1994**, *100*(8), 5829-5835.

[45] J. M. Tao, J. P. Perdew, V. N. Staroverov, G. E. Scuseria. *Physical Review Letters* **2003**, *91*(14).

[46] W. J. Hehre, L. Radom, P. R. Schleyer, J. A. Pople. *Ab initio molecular orbital theory,* New York, **1986**.

[47] J. M. Wang, P. Cieplak, P. A. Kollman. *J. Comput. Chem.* **2000**, *21*(12), 1049-1074.

[48] D. Reha, H. Valdes, J. Vondrasek, P. Hobza, A. bu-Riziq, B. Crews, M. S. de Vries. *Chemistry-A European Journal* **2005**, *11*(23), 6803-6817.

[49] H. Valdes, D. Reha, P. Hobza. *Journal of Physical Chemistry B* **2006**, *110*(12), 6385-6396.

[50] H. Valdes, V. Spiwok, J. Rezac, D. Reha, A. G. bo-Riziq, M. S. de Vries, P. Hobza. *Chemistry-A European Journal* **2008**, *14*(16), 4886-4898.

[51] T. Z. Lwin, R. Luo. *Protein Science* **2006**, *15*(11), 2642-2655.

[52] V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg, C. Simmerling. *Proteins-Structure Function and Bioinformatics* **2006**, *65*(3), 712-725.

[53] Y. Duan, S. Chowdhury, C. Wu, G. M. Xiong, W. Zhang, R. Yang, M. Lee, P. Cieplak, R. Luo, T. Lee, J. Caldwell, J. M. Wang, P. A. Kollman. *Abstracts of Papers of the American Chemical Society* **2003**, *225*, U755.

[54] M. Pavone, N. Rega, V. Barone. *Chemical Physics Letters* **2008**, *452*(4-6), 333-339.

[55] A. Halkier, T. Helgaker, P. Jorgensen, W. Klopper, H. Koch, J. Olsen, A. K. Wilson. *Chemical Physics Letters* **1998**, *286*(3-4), 243-252.

[56] P. Hobza, J. Sponer. *Chemical Reviews* **1999**, *99*(11), 3247-3276.

[57] J. SantaLucia, D. Hicks. *Annual Review of Biophysics and Biomolecular Structure* **2004**, *33*, 415-440.

[58] M. J. Doktycz, M. D. Morris, S. J. Dormady, K. L. Beattie. *Journal of Biological Chemistry* **1995**, *270*(15), 8439-8445.

[59] T. Kubar, P. Jurecka, J. Cerny, J. Rezac, M. Otyepka, H. Valdes, P. Hobza. *Journal of Physical Chemistry A* **2007**, *111*(26), 5642-5647.

[60] N. Sugimoto, S. I. Nakano, M. Yoneyama, K. I. Honda. *Nucleic Acids Res.* **1996**, *24*(22), 4501-4505.

[61] J. Watkins, J. SantaLucia. *Nucleic Acids Res.* **2005**, *33*(19), 6258-6267.

[62] G. D. Hawkins, C. J. Cramer, D. G. Truhlar. *Journal of Physical Chemistry* **1996**, *100*(51), 19824-19839.

[63] W. C. Still, A. Tempczyk, R. C. Hawley, T. Hendrickson. *Journal of the American Chemical Society* **1990**, *112*(16), 6127-6129.

[64] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, P. A. Kollman. *J. Am. Chem. Soc.* **1996**, *118*(9), 2309.

[65] V. Barone, M. Cossi. *Journal of Physical Chemistry A* **1998**, *102*(11), 1995-2001.

[66] Frisch, M. J., Trucks, G. W., Schlegel, H. B., Scuseria, G. E., Robb, M. A., Cheeseman, J. R., Montgomery, Jr, Vreven, T., Kudin, K. N., Burant, J. C., Millam, J. M., Iyengar, S. S., Tomasi, J., Barone, V., Mennucci, B., Cossi, M., Scalmani, G., Rega, N., Petersson, G. A., Nakatsuji, H., Hada, M., Ehara, M., Toyota, K., Fukuda, R., Hasegawa, J., Ishida, M., Nakajima, T., Honda, Y., Kitao, O., Nakai, H., Klene, M., Li, X., Knox, J. E., Hratchian, H. P., Cross, J. B., Bakken, V., Adamo, C., Jaramillo, J., Gomperts, R., Stratmann, R. E., Yazyev, O., AUstin, A. J., Cammi, R., Pomelli, C., Ochterski, J. W., Ayala, P. Y., Morokuma, K., Voth, G. A., Salvador, P., Dannenberg, J. J., Zakrzewski, V. G., Dapprich, S., Daniels, A. D., Strain, M. C., Farkas, O., Malick, D. K., Rabuck, A. D., Raghavachari, K., Foresman, J. B., Ortiz, J. V., Cui, Q., Baboul, A. G., Clifford, S., Cioslowski, J., Stefanov, B. B., Liu, G., Liashenko, A., Piskorz, P., Komaromi, I., Martin, R. L., Fox, D. J., Keith, T., Al-Laham, M. A., Peng, C. Y., Nanayakkara, A., Challacombe, M., Gill, P. M. W., Johnson, B., Chen, W., Wong, M. W., Gonzalez, C., and Pople, J. A. Gaussian 03, Revision D.02. **2004**.

[67] P. Hobza, J. Sponer. *Chemical Reviews* **1999**, *99*(11), 3247-3276.

[68] R. Lavery, A. Lebrun, J. F. Allemand, D. Bensimon, V. Croquette. *Journal of Physics-Condensed Matter* **2002**, *14*(14), R383-R414.

[69] T. R. Strick, M. N. Dessinges, G. Charvin, N. H. Dekker, J. F. Allemand, D. Bensimon, V. Croquette. *Reports on Progress in Physics* **2003**, *66*(1), 1-45.

[70] T. R. Strick, R. Sachidanandam, A. Revyakin, R. H. Ebright. *Biophysical Journal* **2003**, *84*(2), 309A.

[71] M. Balsera, S. Stepaniants, S. Izrailev, Y. Oono, K. Schulten. *Biophysical Journal* **1997**, *73*(3), 1281-1287.

[72] S. A. Harris, Z. A. Sands, C. A. Laughton. *Biophysical Journal* **2005**, *88*(3), 1684-1691.

[73] Rezac, J. and Harris, S. A. Sequence dependence of the stretching behaviour of DNA investigated by molecular dynamics simulation (manuscript in preparation).

[74] Mitchell, J., Ndlovu, H., Rezac, J., Hobza, P., and Harris, S. Denaturing DNA in silico. Jornal of the Royal Society Interface (submitted).

[75] Hubbard, S. and Thornton, J. NACCESS. **1996**.

[76] Lavery, R. and Sklenar, H. Curves, www.ibpc.fr/UPR9080/Curindex.html. **2004**.

[77] D. Reha, M. Hocek, P. Hobza. *Chemistry-A European Journal* **2006**, *12*(13), 3587-3595.

[78] R. A. Distasio, G. von Helden, R. P. Steele, M. Head-Gordon. *Chemical Physics Letters* **2007**, *437*(4-6), 277-283.

[79] T. Janowski, P. Pulay. *Chemical Physics Letters* **2007**, *447*(1-3), 27-32.

[80] C. T. Lee, W. T. Yang, R. G. Parr. *Physical Review B* **1988**, *37*(2), 785-789.

[81] R. Podeszwa, R. Bukowski, K. Szalewicz. *Journal of Physical Chemistry A* **2006**, *110*(34), 10345-10354.

[82] M. O. Sinnokrot, C. D. Sherrill. *Journal of the American Chemical Society* **2004**, *126*(24), 7690-7697.

[83] P. Hobza, V. Spirko, H. L. Selzle, E. W. Schlag. *Journal of Physical Chemistry A* **1998**, *102*(15), 2501-2504.

[84] P. Hobza, Z. Havlas. *Chemical Reviews* **2000**, *100*(11), 4253-4264.

[85] U. Erlekam, M. Frankowski, G. Meijer, G. von Helden. *Journal of Chemical Physics* **2006**, *124*(17).

[86] E. Arunan, H. S. Gutowsky. *Journal of Chemical Physics* **1993**, *98*(5), 4294-4296.

[87] W. Z. Wang, M. Pitonak, P. Hobza. *Chemphyschem* **2007**, *8*(14), 2107-2111.

[88] E. C. Lee, D. Kim, P. Jurecka, P. Tarakeshwar, P. Hobza, K. S. Kim. *Journal of Physical Chemistry A* **2007**, *111*(18), 3446-3457.

[89] J. VandeVondele, M. Krack, F. Mohamed, M. Parrinello, T. Chassaing, J. Hutter. *Computer Physics Communications* **2005**, *167*(2), 103-128.

[90] CP2K, cp2k.berlios.de. 2008.

[91] S. Goedecker, M. Teter, J. Hutter. *Physical Review B* **1996**, *54*(3), 1703-1710.

[92] T. Tuttle, W. Thiel. *Physical Chemistry Chemical Physics* **2008**, *10*(16), 2159-2166.

Part Two:

Cuby Manual

Cuby is an environment for chemical calculations. It relies on external software for QM and MM calculations and can manipulate and combine the results using various methods and simulation protocols. And it does so in user-friendly way.

cuby

Chemistry in Ruby

Table of contents

1 Introduction

Cuby (name is an abbreviation for Chemistry in Ruby) can be viewed from two angles: For most users, it is a set of tools that makes common calculations easy. For advanced user, it implements several specialized methods not widely available. For programmer, it is robust framework, on top of which it is easy to build new methods and calculation protocols. It is written in Ruby (www.ruby-lang.org), a high level object-oriented programming language. This allows rapid implementation of new functions when it is necessary and makes the development or modification of the code accessible to wider range of users

The development started because there was a need for QM/MM calculations based on the DFT-D method developed in our laboratory. Later, it has grown up to be more universal – once the basic framework was available, it was applied in many other projects to make the work easier. The code evolved during the process and most of the underlying libraries were rewritten to be more universal.

Cuby itself does not perform the QM or MM calculations, it calls external programs for the task. This approach allows combining methods implemented in different software packages within one calculation protocol, such as QM/MM. It also offers one user friendly interface to different programs, what significantly increases productivity.

Nowadays, it includes programs for calculating interaction energies, performing geometry optimizations using several algorithms, run molecular dynamics, calculate frequencies etc. Each of this algorithms can be combined with every external program, what open new possibilities.

This thesis was taken as an opportunity to consolidate documentation of Cuby's features and possibilities. There was some documentation available before, but it was not complete and did not cover all usages of the code. Nowadays, Cuby is getting more and more popular in our laboratory and new users start to use it, what makes better documentation necessary.

2 Installation

To install Cuby, ruby version 1.8 should be installed on your system. Installation and updates of Cuby are straightforward, because it uses packaging system designed to maintain Cuby on our clusters.

- Prepare a directory for installation. In following text, it is abbreviated as *install_dir*.
- Download installer.tar and cuby_update.tar from Cuby website
- Unpack installer.tar
- Run script unpackDistro. It extracts the cuby_update.tar package, taking care of previous version if it is installed.

To run Cuby for the first time, some configuration is required.

- Ruby interpreter should be able to find all necessary libraries. Add paths *install_dir*/classes and *install_dir*/lib to the RUBYLIB environment variable. It is useful to do this automatically on login, for example in .bashrc file if you are using bash shell.
- Add *install_dir* to your PATH variable so that you can run Cuby from anywhere.
- On some system, variable RUBYOPT must be set to empty string.
- Edit the configuration file *install_dir*/rubyqmmm.conf. It contains paths to all the external programs used by Cuby, and they should be set up according to your system.

Some parts of the code depend on GNU Scientific Library (www.gnu.org/software/gsl) which provides advanced math functions. This library, and ruby wrapper (rb-gsl.rubyforge.org) for accessing it from ruby, is necessary only for calculations where it is used, such as optimizations by optimize program or calculation of frequencies. On linux system, the library should be available via a package manager. The ruby wrapper can be downloaded from author's website. After installation and compilation, path to the library should be added to the RUBYLIB environment variable.

To update your installation to the latest version, download the update package and run the unpackDistro script. It updates the code, but it preserves the configuration.

3 Input

Rather than designing new input format, we decided to make use of an accepted standard – subset of PDB specification (www.wwpdb.org/docs.html). It has one major advantage – input (and output) files can be prepared and visualized in existing tools. Another advantage is that it can provide more information than plain coordinates, such as information on residues. It can be also easily extended to contain setup for the calculation, as described below.

PDB specification is broad and covers more than we need, so we use just a subset of the specification. We introduced several new elements that do not comply with the specification, but do not interfere with it, so that the resulting file can be opened in existing software.

Keywords specifying the calculation are provided in header of the file as REMARK records, and are thus ignored in other software according to the PDB standard. The format is:

```
REMARK    KEYWORD   Value
```

If value contains whitespace, it should be quoted using either single or double quotes. Note also that value is case sensitive. In most cases, upper case is used, but some values passed directly to external codes can have their own rules.

Molecular geometry is specified by ATOM records. HETATM record is not supported, but can be replaced by ATOM. Additional information on atoms can be provided in additional column(s) beyond coordinates. This information differs from PDB specification, but is ignored or cause no problems in other software. Following example shows input for geometry optimization usind DFTB-D method:

```
REMARK  METHOD          OPTIMIZE
REMARK  CODE            DFTB
REMARK  DFTB_DISP       YES
REMARK  CLUSTERCHARGE   0
REMARK  FREEZE          YES
REMARK
ATOM      1  O   WAT     1      -1.525  -0.066   1.000  F
ATOM      2  H1  WAT     1      -1.847   0.836   1.000  F
ATOM      3  H2  WAT     1      -0.565   0.035   1.000  F
TER
ATOM      4  O   WAT     2       1.445   0.060   0.000
ATOM      5  H1  WAT     2       1.838  -0.382  -0.756
ATOM      6  H2  WAT     2       1.838  -0.382   0.756
TER
END
```

Finally, we parse the PDB file as a free format, what allows more accurate specification of coordinates than the standard three decimal places. Such a file is readable by Cuby, but can not be opened in other pdb readers.

PDB does not provide direct information on element, since it uses atom types. There are two ways how to treat this in Cuby. Firstly, **element can be derived from the atom type**. Set of empirical rules and exceptions from these rules is used and works well for standard atom types found in biomolecules. **However, this procedure may fail for some atom types**. If the element can not be determined, the program is halted, but it might also assign the element incorrectly. Elements can be used in place of atom types and should be recognized, at least for the most common elements. Second option, forces direct reading of element without further processing, if the residue is named “UNK” (stands for “unknown”).

4 Modules

From the user's point of view, Cuby consist of several modules for performing different tasks. The main executable `cuby` is in fact a script that reads the input and decides which module to use. It is controlled by the `METHOD` keyword, which can have following values:

- **GEOMETRY** for geometry optimization and molecular dynamics, including QM/MM
- **OPTIMIZE** calls new code for geometry optimization
- **FREQ** invokes calculation of vibrational frequencies
- **INTE** specifies calculation of interaction energy
- **GAUSSIMOLE** calls the interface between Gaussian and Cuby – see chapter 7

5 Keywords

5.1 Common keywords

There is a set of keywords that is shared by all the programs. These are mainly keywords specifying method used for the calculation.

- **METHOD** GEOMETRY | INTE | FREQ
  Type of calculation. The "cuby" script calls specialized executable for each of these methods.
    - **GEOMETRY**: Geometry optimization, molecular dynamics and singlepoint calculations
    - **INTE**: Interaction energy calculations
    - **FREQ**: Calculation of harmonic vibrational frequencies
- **CODE** TURBOMOLE | GAUSSIAN | MOLPRO | DFTB | AMBER | VOID | EXTRAPOLATE | QMMM
  Specifies what external program will be used to do the actual calculation.
    - **TURBOMOLE**: Calculation using Turbomole program package
    - **GAUSSIAN**: Calculation using Gaussian 03 package
    - **MOLPRO**: Calculation using Molpro package (Only CCSD(T) calculations at the moment)
    - **DFTB**: Calculation using the DFTB+ code
    - **AMBER**: MM calculation using AMBER package
    - **VOID**: Virtual interface returning zeroes for testing purposes.
    - **EXTRAPOLATE**: Virtual interface for MP2/CBS extrapolation (Uses Turbomole) and CCSD(T)/CBS based on MP2/CBS + CCSD(T) correction calculated in Molpro.
    - **QMMM**: Virtual interface for QM/MM calculations, combines AMBER with selected QM code.
- **CHARGE** integer *(0)*
  Total charge of the system

5.2 Keywords specifying the calculation

Each method used for calculation (and interface to the program performing this calculation in Cuby) has its own set of keywords. In addition to interfaces to external programs, there are “virtual” interfaces for methods combining multiple calculations in one. Following interfaces are implemented in Cuby:

5.2.1 Turbomole

QM calculation using Turbomole software package.

- **LEVEL** SCF | DFT | RIDFT | RIMP2
  Level of calculation selected CODE can use

- **BASISSET** string *(TZVP)*
  Basis set for QM calculation

- **FUNCTIONAL** b-lyp | tpss | b3-lyp | pbe | pbe0 | slater-dirac-exchange | s-vwn | vwn | s-vwn_Gaussian | pwlda | becke-exchange | b-vwn | lyp | b-p | bh-lyp | b3-lyp_Gaussian | tpssh | lhf
  Functional for DFT calculation

- **DFT_GRID** string *(m3)*
  Grid quality in DFT calculation, se Turbomole manual for options

- **RIMEM** integer *(300)*
  Memory for RI calculations in turbomole, in MB

- **RIDISC** integer *(0)*
  Disc space limit for RI calculations in turbomole

- **RIDFT_DISP** YES | NO
  Dispersion correction calculation using the original tmdisp program. Outdated feature, use DISPERSION instead.

- **RIDFT_DISP_PARA** string *(radii_scaling,alpha)*
  Parameters for dispersion correction calculated by tmdisp. Outdated feature, see DISPERSION.

- **DEFINE** YES | NO
  Switch allowing to disable "define" step in turbomole calculation preparation.

- **TMOPTIONS** string *([LEVELSHIFT=float],[SCFITERS=integer],[MARIJ],[NOSCFDIIS],[SCFITERS=SCFCONV],[SCFITERS=DENCONV])*
  Options passed to turbomole calculation setup. For details, consult turbomole manual.

- **PREPARE_EACH** YES | NO
  By default, QM calculations are started from results of prevoius steps. This option can force restarting the QM calculation completely in each cycle of calculation.

- **PARALLEL** integer *(1)*
  Number of processors for parallel calculation.

- **COSMO** YES | NO
  COSMO calculation, using default radii and dielectric constant set by EPSILON keyword.

- **EPSILON** float *(78.5)*
  Dielectric constant in COSMO calculation.

5.2.2 Molpro

Basic interface to CCSD(T) calculations in Molpro package.

- **LEVEL** CCSD(T)
  Level of calculation selected CODE can use

- **BASISSET** string *(TZVP)*
  Basis set for QM calculation

- **MEM** integer *(300)*
  Memory for QM calculation, in MB

Gaussian

Interface to Gaussian 03 package.

- **MEM** integer *(300)*
  Memory for QM calculation, in MB

- **PARALLEL** integer *(1)*
  Number of processors for parallel calculation.

- **JOBTYPE** CUSTOM | PCM
  Selection of job type, allows use of predefined types of calculation.

  - **CUSTOM**: Custom calculation using setup provided in GAUSSKEYWORDS

  - **PCM**: Basic PCM calculation: HF/6-31G(d) SP SCRF=(CPCM,Read,Solvent=Water)

- **GAUSSKEYWORDS** string
  Calculation specification for Gaussian - is directly used as a job specification line, #P is prepended automatically to ensure detailed output.

- **MULTIPLICITY** integer *(1)*
  Multiplicity of system.

5.2.3 DFTB

Interface to DFTB+ code performing efficient SCC-DFTB(-D) calculations.

- **PARALLEL** integer *(1)*
  Number of processors for parallel calculation.

- **DFTB_DISP** YES | NO
  Dispersion correction built in DFTB+ code

- **DFTB_CONVLIMIT** float *(1.0e-05)*
  Convergence limit for DFTB procedure [Hartree]

- **DFTB_SLKO** string *(Read from config)*
  Custom path to Slater-Kostner parameter files

- **DFTB_USELAST** YES | NO
  Restart self-consistent charge calculation from values from previous step

5.2.4 Amber

MM calculations in AMBER software package. Requires modification to the sander module that adds output of gradient.

- **GBM** YES | NO
  Switch for Generalized Born Model implicit solvent in AMBER.

- **GBM_SALT** float *(0.0)*
  Salt concentration in GBM solvent, as implemented in AMBER.

- **LEAPRC** string *(read from .conf)*
  Optional specification of leaprc file used for AMBER input preparation.

5.2.5 QM/MM

Virtual interface combining QM methods with AMBER MM into QM/MM calculations.

- **CLUSTER_CODE** TURBOMOLE | DFTB
  Code used for calculation of the QM region (cluster) in QM/MM procedure

- **CLUSTER_LEVEL** SCF | DFT | RIDFT | RIMP2
  Level of calculation of QM region in QM/MM procedure

- **CLUSTER_CHARGE** integer *(0)*
  Charge of QM region (cluster) in QM/MM procedure

- **POLARIZED** YES | NO
  Polarization of QM region by point charges from MM region (electrostatic embedding).

- **CUTOFF** string *(NO / integer)*
  Cutoff distance (A) for point charges selection in polarized QM/MM calculation.

- **DISPENSE_CHRG** YES | NO
  Treatment of charges close to link atoms. If switched on, removed charge is evenly distributed in rest of the residue.

- **ESP_FIT** YES | NO
  Update of MM charges from QM calculation using ESP fit. Experimental feature.

- **ESP_LINK** FIT | MM
  Link atom treatment within ESP_FIT.

- **MICROITERATIONS** YES | NO
  Microiteration procedure optimizing MM part in each cycle of QM/MM calculation.

- **MICROIT_WRITE** YES | NO
  Save history of microiterations.

- **MICROIT_MAXCYC** integer *(5000)*
  Maximum number of steps in microiterations procedure

5.2.6 Extrapolate

Virtual interface for extrapolation to complete basis set limit from MP2 calculations with increasing basis set size. Also calculates CCSD(T) correction in smaller basis set.

- **EXT_SCHEME** string *(CCSDT/CBS:aD->aT)*
  Name of extrapolation scheme for CCSD(T)/CBS extrapolation. Schemes are defined in *install_dir*/input/extrapolation.yaml by default.

- **EXT_CONFIG** string *(extrapolation.yaml in installdir/input)*
  Specification of custom file with extrapolation schemes.

5.3 Dispersion

Standalone implementation of dispersion correction which can be added to any calculation.

- **DISPERSION** YES | NO
  Generaly applicable dispersion correction, can be used with any method.

- **DISPERSION_PARA** string *(radii_scaling,alpha[,global scaling])*
  Custom specification of dispersion parameters, default parameters for given method will be used if omitted.

- **DISPERSION_FILE** string
  Custom file with dispersion parameters, default is *install_dir* /input/tmdisp.yaml

- **DISPERSION_HYB** YES | NO
  Use different atomic parameters for different hybridizations

- **DISPERSION_MIX** PJ | GRIMME
  Mixing of C6 parameters and atomic radii

  - **PJ**: Petr Jurecka's setup

  - **GRIMME**: Stefan Grimme's setup

5.4 GEOMETRY methods

Module geometry performs geometry optimizations as well as molecular dynamics simulations. Keywords specific for MD simulations are listed separately in following section.

- **ALGORITHM** RELAX | SINGLEPOINT | SD | CG | DYNAMICS
  Algorithm for geometry update in GEOMETRY method.

- **CGSDSTEPS** integer *(20)*
  Number of steepest descent steps at the beginning of conjugated gradients optimization.

- **FROZENLAYER** YES | NO
  Atoms in frozen layer (marked X) are not included in set passed to geometry driver.

- **FREEZE_TO_X** YES | NO
  Converts atoms frozen by F flag to frozen layer (flag X)

- **MAXCYCLES** integer *(200)*
  Maximum number of steps in geometry optimization or lolecular dynamics.

- **OPTSTEP** float *(0.3)*
  Maximum step in optimization, in A.

- **CONVLIMIT** float *(1.2)*
  Geometry optimization convergence limit for max. gradient, in kcal/mol/A.

- **CONVLIMIT_E** float *(0.006)*
  Geometry optimization convergence limit for energy difference between subsequent steps, in kcal/mol.

- **FREEZE** YES | NO | ALL
  Freezing cartesian coordinates of atoms by setting gradients to zero. Note that algoritm used in Relax program can move these atoms.

- **THAW** string *(list of elements)*
  Removes freeze flag from atoms of given element.

- **FREEZE_AXES** string *([x],[y],[z])*
  Disables optimization in selected axis of cartesian coordiantes.

- **THAW_AXES** string *(list of elements)*
  Unfreezes all axes for selected elements.

- **IMPORTCOORD** string *(filename)*
  Import of new coordinates from .xyz for the molecule at the beginning of calculation.

- **WRITEHISTORY** integer *(0)*
  Optimization history / trajectory is logged each Nth step in file history.xyz. Set to zero to prevent logging.

- **HISTORYFORMAT** XYZ | TRAJ
  Format of optimization history/MD trajectory

  - **XYZ**: .xyz file

  - **TRAJ**: AMBER trajectory format (saves space, no information on atoms)

- **WRITERESTART** NO | STEP | END
  Controls writing of restart file (pdb with header) during geometry optimization / MD.

- **WRITEGRADIENT** YES | NO
  Optional output of cartesian gradient in each step.

- **LONGNUMBERS** YES | NO
  Switches on higher decimal precision in output .pdb files.

- **RUNSCRIPT** string *(filename)*
  Run external command at end of each step.

- **CONSTRAIN_BONDS** string *(list of constraints)*
  Adds harmonic potential to selected interatomic distances. Pairs are specifid in comma

separated list of following records: atom_index-atom_index:distance
Actual distance and added force is listed in output.

- **CONSTRAIN_K** float *(2000.0)*
  Force constant of bond constraints [kacal/mol/A]

5.5 Molecular dynamics

Keywords controlling molecular dynamics simulations:

- **TIMESTEP** float *(0.001)*
  Timestep in molecular dynamics simulations, in ps.
- **INIT_TEMP** float *(10.0)*
  Temperature used to generate initial velocities for molecular dynamics.
- **TCOUPLE** YES | NO
  Molecular dynamics at constant temperature.
- **THERMOSTAT** BERENDSEN | NOSE-HOOVER | ANDERSEN
  Thermostat algorithm for NVT simulations
- **TEMPERATURE** float *(300.0)*
  Thermal bath temperature fo NVT calculations.
- **TAU_TEMP** float *(0.5)*
  Thermostat relaxation time [ps]
- **ANDERSEN_TC** float *(0.4)*
  Mean period between collisions in Andersen thermostat [ps]
- **ANNEAL** YES | NO
  Simulated annealing: temperature is decreased to 0 K during the simulation.
- **VELOCITIES** RANDOM | READ
  Initial velocities for MD run
- **RANDOM_SEED** integer
  Random number generator initialization by non-random seed, for running identical MD trajectories.
- **REMOVECOM** YES | NO
  Center of mass translation/rotation removal in MD simulation.
- **REMOVECOM_STEPS** integer
  Center of mass motion removal limited to begnning of the simulation.
- **PRINT** string *([COMVELO],[KINETIC],[TOTAL])*
  Additional output for MD simulations: Center of mass velocity, kinetic energy, total energy.

5.6 Metadynamics

Metadynamics algorithm is applied on top of MD run.

- **FLOODING** YES | NO
  Switches flooding on/off

- **META_DIRECT** YES | NO
  Direct metadynamics alorithm does not use virtual particles to introduce moment of inertia into studied internal coordinates.

- **META_STARTHILLS** CLEAR | LOAD
  Allows to load hills from previous simulation, useful for restarts.

- **META_WRITE** CURRENT | AVERAGE
  Mode of writing hills: at current (original setup) and averaged (experimental feature) position.

- **META_CVNUM** 1 | 2 | 3
  Number of collective variables (studied internal coordinates).

- **META_HILLFILE** string *(hills.txt)*
  Name of file for storing the flooding potential ("hills").

- **META_PERSTEP** integer *(0)*
  Period of adding hills, in steps.

- **META_GHEIGHT** float *(0.23885)*
  Height of gaussian hill, in kcal/mol

- **META_MASS_1** float *(0.23885)*
  Mass of virtual particle in coordinate 1

- **META_MASS_2** float *(0.23885)*
  Mass of virtual particle in coordinate 2

- **META_MASS_3** float *(0.23885)*
  Mass of virtual particle in coordinate 3

- **META_TYPE_1** DISTANCE | ANGLE | DIHEDRAL | COORDINATION
  Type of coordinate 1

- **META_TYPE_2** DISTANCE | ANGLE | DIHEDRAL | COORDINATION
  Type of coordinate 2

- **META_TYPE_3** DISTANCE | ANGLE | DIHEDRAL | COORDINATION
  Type of coordinate 3

- **META_LIST_1** string
  Definition of coordinate 1, atom lists (coma separated) for coordinate centers (separated by "-")

- **META_LIST_2** string
  Definition of coordinate 2, atom lists (coma separated) for coordinate centers (separated by "-")

- **META_LIST_3** string
  Definition of coordinate 3, atom lists (coma separated) for coordinate centers (separated by "-")

- **META_LAMBDA_1** float *(95.54)*
  Strength to of coupling between virtual particle and the system

- **META_LAMBDA_2** float *(95.54)*
  Strength to of coupling between virtual particle and the system

- **META_LAMBDA_3** float *(95.54)*
  Strength to of coupling between virtual particle and the system

- **META_GWIDTH_1** float *(0.3)*
  Width of gaussian hill, in dimension of respective coordinate

- **META_GWIDTH_2** float *(0.3)*
  Width of gaussian hill, in dimension of respective coordinate

- **META_GWIDTH_3** float *(0.3)*
  Width of gaussian hill, in dimension of respective coordinate

5.7 Optimization using OPTIMIZE

Geometry optimizations using the BFGS algorithm implemented in module optimize.

- **CONVLIMIT** float *(1.2)*
  Geometry optimization convergence limit for max. gradient, in kcal/mol/A.

- **CONVLIMIT_E** float *(0.006)*
  Geometry optimization convergence limit for energy difference between subsequent steps, in kcal/mol.

- **FREEZE** YES | NO | ALL
  Freezing cartesian coordinates of atoms by setting gradients to zero. Note that algoritm used in Relax program can move these atoms.

- **THAW** string *(list of elements)*
  Removes freeze flag from atoms of given element.

- **CONVLIMIT_N** float *(0.6)*
  Geometry optimization convergence limit for gradient norm [kcal/mol/A].

- **OPT_STEP** float *(0.001)*
  Initial step size in optimization.

5.8 Vibrational frequencies

Vibrational frequencies can be calculated analytically with any method that has second derivatives. Numerical calculation is available for methods with gradients only.

- **NUMERICAL** YES | NO
  Numerical calculation of second derivatives.

5.9 Interaction energy calculation

Interaction energy calculation using module intEnergy.

- **RESIDUES_A** string *(1)*
  List of residues in subsystem A, comma separated list of integers.

- **RESIDUES_B** string *(2)*
  List of residues in subsystem B, comma separated list of integers.

- **CHARGE_A** integer *(0)*
  Charge of subsystem A

- **CHARGE_B** integer *(0)*
  Charge of subsystem B

- **BSSE_CORR** YES | NO
  Counterpoise correction of BSSE

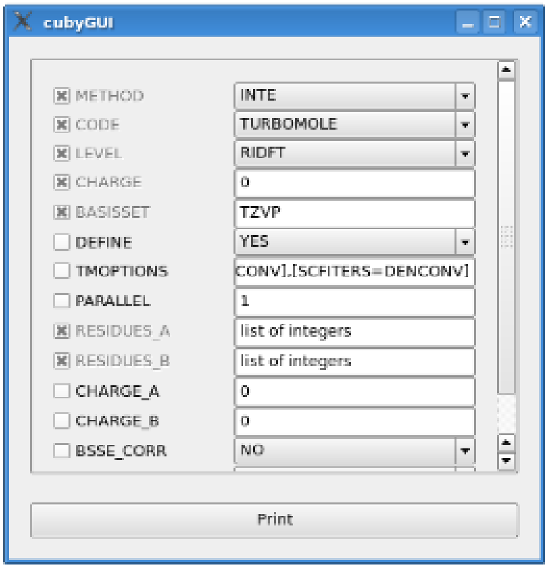
- **CHARGE_AUTO** YES | NO
  Charge of subsystems is built from charge of residues, specified by keywords CHARGE_RES_#.

- **DELETE_DIRS** YES | NO
  Deletes subrirectories with calculations to save disc space.

6 CubyGUI – graphical interface

To make preparation of input files easier, Cuby has graphical interface cubyGUI (Figure 1). It is based on a database of all keywords used in Cuby and their dependencies. Once a keyword is selected in cubyGUI, all possibilities depending on this selection are displayed.

Figure 1: cubyGUI graphical interface



☒	METHOD	INTE
☒	CODE	TURBOMOLE
☒	LEVEL	RIDFT
☒	CHARGE	0
☒	BASISSET	TZVP
□	DEFINE	YES
□	TMOPTIONS	CONV],[SCFITERS=DENCONV]
□	PARALLEL	1
☒	RESIDUES_A	list of integers
☒	RESIDUES_B	list of integers
□	CHARGE_A	0
□	CHARGE_B	0
□	BSSE_CORR	NO

The GUI uses the Qt4 libraries and ruby wrapper for them. The best way to install it is to use a package manager of your linux distribution, manual installation could be problematic.

7 Gaussimole, Gaussdisp

Gaussian software package can use external programs to perform the calculation of required points. We use this feature in Cuby module gaussimole. .Gaussian is used to provide the calculation protocol, such as geometry optimization or anharmonic vibrational frequencies calculation (what is the reason why gaussimole was developed) and Cuby provides an interface to the calculation in various programs. The communication between Gaussian and the external program is described in Gaussian manual (keyword external). This communication is different in older versions of Gaussian, version 03 D is required for proper operation.

To perform such a calculation, METHOD key is set to GAUSSIMOLE and GAUSKEYWORDS provide setup of the Gaussian calculation (calculation type must be set to “external”). Then, the calculation of the requested points is set up. Here is an example of a header for calculation of harmonic frequencies using RI-DFT-D from Turbomole:

```
REMARK  METHOD          GAUSSIMOLE
REMARK  GAUSSKEYWORDS   "external freq"
REMARK  MEM             300
REMARK
REMARK  CODE            TURBOMOLE
REMARK  LEVEL           RIDFT
REMARK  DISPERSION      YES
REMARK  CHARGE          0
REMARK  BASISSET        TZVP
REMARK  FUNCTIONAL      b-lyp
REMARK  RIMEM           300
REMARK  DISPERSION      YES
REMARK
```

Another tool coupling Gaussian and Cuby is the gaussdisp script. It is a standalone script, based on Cuby libraries, that provides dispersion correction to DFT calculations in Gaussian. It is designed to be extremely easy to use. To perform DFT-D calculation in Gaussian, place this script into the directory containing the Gaussian input file and use for the calculation via the “external” keyword. Following job specification line will invoke geometry optimization using the DFT-D method:

```
# external="gaussdisp 'BLYP/TZVP' " opt
```

Parameters for the dispersion correction for given combination of basis set and functional (Jurecka et al., J. Comp. Chem. 2006) are taken from a database contained in the script file. The gaussdip script is independent on the Cuby installation and can be used anywhere where ruby 1.8 is installed.

8 Tools

Cuby comes with a set of scripts and tools that could be used to prepare and analyze files used by Cuby. Here, the tools are divided into several categories:

QM/MM input preparation

- **getPdbCluster.rb** *filename.*
  Extracts QM region from QM/MM input file and save it in cluster_*filename*
- **markXByDistance.rb** *distance*|all|none *filename*
  Selects outer layer frozen in QM/MM optimizations by distance from QM region
- **linkZByBonds.rb** *distance filename*
  Selects atoms without point charges around link atoms. Distance is a number of bonds from the original atom.
- **linkZByDistance.rb** *distance filename*
  Selects atoms without point charges around link atoms. Atoms are selected according to their distance (in Å) from the link atom.
- **markXCleanup** *filename*
  residues partially included in the frozen layer are included fully if more than 50% of the residue was marked, otherwise the residue is deselected.

Batch calculation of interaction energies

- **interactionList.rb** *pairlist input_file*
  Script for calculation multiple interactions between residues in one PDB file. The input PDB file should have valid Cuby header for interaction energy calculation. Pairs of residues for calculation are read from the *pairlist* file, one interaction per line. Two list of residues, composing the monomers (as comma separated list of residue numbers) are separated by "-" character. For example pairlist:

  ```
  1 - 2
  1,2 - 3
  ```

  will calculate interaction energies between residues 1 and 2, and between 1+2 and 3.

Output log analysis

- **plotLOG** [-p] [variable]
  is a script for visualization of LOG files (output of Cuby calculation using GEOMETRY method). It uses xmgrace program for plotting the results, unless –p option is used to output the numbers directly. Variable option specifies which variable is printed, default is potential energy. Other possibilities, used on output of MD simulations, are:

T	temperature
kinetic	kinetic energy
COMa	Center of mass angular velocity
COMv	Center of mass translational velocity

Geometry file conversions

Several utilities are provided to convert between used file formats. Unless stated otherwise, they write the geometry in new format to standard output.

- `xyz2pdb.rb [-l]` *`filename`*
  xyz to PDB conversion. The -l switch can be used to produce PDB-like file with higher precision of the coordinates, which can be read by Cuby.
- `pdb2xyz.rb` *`filename`*
  PDB to xyz conversion
- `pdb2coord.rb` *`filename`*
  PDB to Turbomole coordinate file format conversion
- `pdb2rst.rb` *`filename`*
  PDB to AMBER coordinate (restart) format
- `orderWater` *`filename`*
  AMBER requires correct order of atoms in water molecules, which might be different ftom order found in some PDB files. This script sorts atoms in all WAT residues to fix the problem.
- `pdbRenumber` *`filename`*
  Script to fix atom and residue numbering in PDB files.
- `updatePdbFromXyz` *`pdbfile xyzfile`*
  Geometry from *xyzfile* is inserted into *pdbfile*, while all other information in the original PDB is conserved. The files must contain the same number of atoms. Output is written to file updated_*pdbfile*.
- `getxyzframe` *`frame_number filename`*
  Extracts one frame from xyz file with multiple geometries. Frame numbering starts with 1.

Structure comparison

- `rmse` *`file1 file2`*
  Reads two PDB files. Structure in file2 is fitted to file 1, minimizing RMSD. Fitted structure and RMSD values are printed.

9 Programming concepts

To understand how the code works, it is useful to know its internal structure. Although some information provided here may be too detailed, we hope it could be helpful to advanced user.

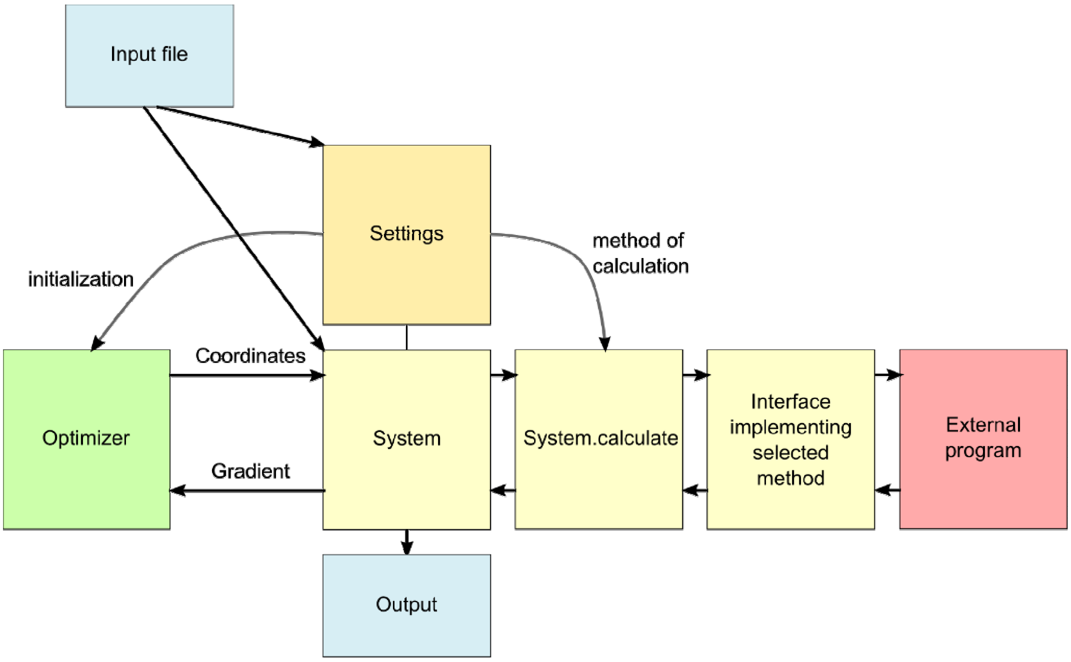
Ruby is strongly object-oriented language and this fact is thoroughly used in the Cuby framework. We found object oriented design very useful for description of chemical system. Library of classes is built from bottom to top. For example, the very basic class is cartesian coordinate (*Coordinate*). Atom then inherits all its properties and adds additional attributes and methods. Chemical system (class *System*) contains collection of atoms, settings (*System.settings*) for desired type of calculation, methods for performing the calculation as well as obtained results. (For details on internal structure of the source code, see following chapter)

Another important part of the code is modular set of interfaces to external programs, which perform the actual calculations. There is an abstraction layer (module StructureCalculation in file mod_StructureCalculation.rb) that decides which interface to call according to calculation setup. In addition to interfaces that works directly with external programs, there are also "virtual" interfaces for multiple-method calculations, such as complete basis set (CBS) extrapolation scheme or counterpoise-corrected calculations, which make use of the "real" interfaces.

Simulation protocols are either implemented directly in the particular program or, more recently, as independent and reusable objects. Ruby, as an interpreted language, allows loading parts of the source code in runtime. We use this feature to load only the modules that are needed for actual calculation, what improves performance.

The whole concept can be demonstrated on diagrammatic representation of setup for geometry optimization using program `optimize` (Figure 2).

Figure 2: Program structure in geometry optimization using Optimizer object.



Optimizer is instance of `Optimizer` class, which is general implementation of optimization algorithm operating on any vectors. It calls block of code, where calculation is performed on the *System* object (by calling external program via its interface – *System.calculate*) and passes the result back to the optimizer. Both optimizer and calculation read their setup from *settings* object, which was created from the input file, and do not need any additional information provided in the code.

This architecture allows easy creation of new simulation protocols for special tasks, as well as use of the framework within interactive shell.

10 Source code structure

Programs performing the calculations depend on the Cuby libraries providing all the functionality. These libraries are located in *install_dir*/classes and its subdirectories. The most important class System, which contains definition of the system as well as methods for the calculations, is contained in file CUBY. It is composed from modules found in files mod_StructureModuleName.rb Drivers performing the geometry manipulation are found in files driver_DRIVER_NAME.rb. Interfaces providing the calculations are located in interfaces subdirectory.

Each instance of System class has a variable *settings*, which contains information from the header of the input file (it can be later modified by the program as well). Some methods applied to the System object directly read options from here. It allows, for example, independent interfaces for calculations. The main program calls method *calculate*, which decides what interface to use. The interface also reads its options from *settings*.

Two another classes are very important. Firstly, class Atom (file atom.rb) is used to store all information on individual atoms. It is descendant of coordinate-based classes from file vectors.rb, where mathematical operations on vectors are defined. As a result, these operation can be performed on atoms as well. Secondly, class AtomArray, a descendant of Array, is a data structure used to store sets of atoms, and it provides methods for working with geometry of the set.

Directory *install_dir*/lib contains third-party classes and libraries used by Cuby.

Data files used by Cuby are loacated in directory *install_dir*/input. Preferred format to store structured data is YAML (www.yaml.org).

11 Extending the code, special uses

The code is easy to modify and extend. Ruby programming language is easy to comprehend and can be learned fast. Large library of existing classes and methods allows rapid implementation of new calculation protocols.

Interfaces to new programs can be also easily added. All what is needed is to write the interface module able to construct input, call the program and read its output, and to register this interface in module *StructureCaculation*. This task would be even easier with the currently developed “metainterface”. It is an interface, which uses instructions from a data file to construct the input and read the output of selected program. For simple programs, which require no special treatment, this will allow to connect them to Cuby in short time and without programming.

12 Cuby in interactive shell

In addition to the normal mode, ruby interpreter can be run in interactive mode using `irb` command. After loading Cuby libraries, we get powerful shell for working with molecular geometry and performing calculations. This usage, however, requires knowledge of the Cuby source code.

13 Future of Cuby

Cuby is still in rapid development. New features are added when they are needed in our projects. However, it reached a stage where the main parts of the code are consolidated. Nevertheless, it is still not mature enough to be released to public. The oldest parts of the code are not well documented and will require cleanup.

Major issue is the implementation of the QM/MM procedure, which is now integral part of the GEOMETRY program. In future, it should be rewritten as a virtual interface, what will make it more universal. We are working on this issues and improvements with the goal to make the code available to public. This might take some time, because development of the code is not the main topic of our work, it is just a tool we use to do the science and to do it more efficiently.

Finally, let us note that Cuby is an open project and anyone can participate in it.