Large, Polarizable QM/MM/Continuum Computations
Ancient wishes and recent advances

Filippo Lipparini
Horizon Maths 2014
The ddCOSMO wonder team!
Outline

1. Why multiscale modeling?
2. Polarizable embedding: continuum, discrete, both
3. ddCOSMO: continuum solvation made fast
4. Achieving linear scaling
5. But...
Introduction

Outline

1. Why multiscale modeling?
2. Polarizable embedding: continuum, discrete, both
3. ddCOSMO: continuum solvation made fast
4. Achieving linear scaling
5. But...

Key assumption

We will assume that the QM part of the computation, including computing the QM/environment interaction, is manageable: the focus will be the description of the environment.
QM methods: what can we do?

- **Wavefunction-based methods**
  - Very accurate
  - Horrible scaling (i.e., $O(N^7)$!)
  - Limit: 10-20 atoms (with a lot of patience)

- **DFT methods**
  - Better scaling ($O(N^3)$ or even better)
  - Various approximation (difficult to control)
  - Limit: 1000 atoms (with a bit of patience)

- **Semiempirical methods**
  - Cheap!
  - Very approximate (difficult to control)
  - Limit: 5000 atoms (in reasonable time)
What we would like to do
What we would like to do
What we would like to do
What we would like to do
What we would like to do
Hybrid strategies:

- We definitely can’t afford a full quantum approach
- However, environmental effects are fundamental
- The environment is not important *in se*, but only for its effect on the ”chemical center”

Focused models

- We focus the description on the core
- We introduce the effects of the environment on the core by adding a proper term into its Hamiltonian
- We don’t care (too much) for the environment: we use a cheap model!

Two Families of focused models...

QM/MM models
QM/Continuum models
Everything together: QM/MM/Continuum models
What are we computing?

- Main interest: molecular properties (spectroscopy)
- Response properties depend on the changes of the electronic density due to some external perturbation
- Such properties are influenced by the environment and by how the environment responds to such changes!

We need to be able to capture both effects!

We need a model for the environment that can respond to changes in the electronic density of the core: we need a polarizable model.
Polarizable Continuum Solvation

- The solute is accommodated in a hollow cavity (Ω)

Cramer, Truhlar, Chem. Rev. 99, 2161
Mennucci, Cammi, Tomasi, Chem. Rev. 105, 2999
Polarizable Continuum Solvation

- The solute is accommodated in a hollow cavity ($\Omega$)

Cramer, Truhlar, Chem. Rev. 99, 2161
Mennucci, Cammi, Tomasi, Chem. Rev. 105, 2999
The solute is accommodated in a hollow cavity ($\Omega$)
Polarizable Continuum Solvation

- The solute is accommodated in a hollow cavity ($\Omega$)
- An infinite, structureless, dielectric continuum occupies the rest of the space ($\mathbb{R}^3 \setminus \Omega$)
- Solute-solvent interactions are modeled as the electrostatic interaction between the density of charge of the solute and the dielectric
- Various models for the interaction (Poisson, Poisson-Boltzmann, approximated models...)
- Other interactions (dispersion, repulsion, cavitation) can be treated with empirical expressions

Cramer, Truhlar, Chem. Rev. 99, 2161
Mennucci, Cammi, Tomasi, Chem. Rev. 105, 2999
Apparent Surface Charge

Idea:

- Polarization can be represented via an induced density of charge at the interface
  \[ \sigma = \mathbf{P} \cdot \mathbf{n} \]

- More rigorously, one can use the Green function formalism to recast the problem (3D linear PDE) into an integral equation at the boundary (2D linear IE)

- After some trivial (but tedious) computations, one gets
  \[ Q\sigma = -R\Phi \]

We can get \( \sigma \) (and hence \( \mathbf{P} \))...

... by solving an integral equations on \( \Gamma = \partial\Omega \): a 2D problem!

Cancès, Mennucci, Tomasi, J. Chem. Phys. 107, 3032
Numerical solutions for integral equations

Integral equations can be solved analytically...  
... only for very simple surfaces (sphere, ellipsoid...): for molecule-shaped cavities, a numerical procedure must be adopted

Boundary Elements Method:

- Create a mesh on $\Gamma$
- for each surface element $T_i$, choose a basis function $\eta_i(s)$ (piecewise constant, polynomial, Gaussian...)
- Expand $\sigma$ in the set of functions so obtained:
  \[
  \sigma(s) = \sum_{i=1}^{N_s} \frac{q_i}{a_i} \eta_i(s)
  \]
- project both the integral operators and the potential on the basis:
  \[
  Q_{ij} = \int_{T_i} ds \int_{T_j} ds' \eta_i(s) k_Q(s, s') \eta_j(s'), \quad V_i = \int_{T_i} ds \eta_i(s) \Phi(s), \ldots
  \]
What we *actually* need to do:

The discretization produces:

- A *dense*, symmetric, positive definite matrix $Q \in \mathbb{R}^{N_s \times N_s}$. $N_s$ is $\mathcal{O}(M)$.
- A *dense* matrix $R \in \mathbb{R}^{N_s \times N_s}$ such that $R^{-1}Q$ is symmetric and positive definite.
- Two *very ill conditioned* matrices!
- A linear system of equations $Qq = -RV$

How bad is this?

- Polarizable Continuum Solvation was first used in conjunction with QM methods for the solute: *small systems: use matrix inversion and just don’t care*
- With cheap methods (MM, Semiempirical Hamiltonians, QM/MM) the solution to the CS linear system can rapidly become the bottleneck of the computation
Again on the CS linear system:

- The size of the system grows linearly (with a big constant) with the size of the solute (∼ 100× number of atoms)
- Matrix inversion becomes infeasible very quickly
- Iterative techniques must be used, but the matrices are ill-conditioned....

Can we do better?

- Use the Fast Multipole Method to compute the Matrix/Vector products
- Unfortunately, very “difficult” problem for the FMM (very high density of points)
- Some accuracy issue
- Borderline, or not feasible, for very large systems
QM/MM

- Continuum solvation works well, but can’t model local, specific effects
- Solute-solvent specific interactions (H-Bond) play a fundamental role
- An atomistic description of the environment becomes mandatory

How does it work?

- The environment is introduced as a collection of point charges or higher multipoles at the environment’s atoms
- Polarization effects are recovered by endowing each atom of the environment with a linear electric polarizability
- Polarization equations (linear system) are solved to determine induced dipoles
- The interaction with the QM region is the classical, electrostatic interaction between the QM and MM densities of charge
Polarization equations

Variational principle

The induced dipoles are the ones that maximize the favorable interaction with the MM and QM electric fields and minimize the self-interaction

\[ E(\mu) = \frac{1}{2} \mu^\dagger T \mu - \mu^\dagger (E_{MM} + E_{QM}) \]

\[ T \mu = E \]

- \( T \) is symmetric and positive definite
- It is usually very well conditioned
- It is basically a dipole-dipole interaction matrix; the close neighbors interactions are scaled and damped to avoid overpolarization
Mutual polarization

Let us consider a polarizable force field based on (fixed) point charges and polarizable point dipoles. The charges will induce both atomic dipoles and the polarization of the continuum: these two fields will interact with each other.

Variational formulations can help sort things out:

\[ E(q, \mu, \sigma) = E_{qq} + E_{q\mu} + E_{\mu\mu} + E_{q\sigma} + E_{\mu\sigma} + E_{\sigma\sigma} \]
Mutual polarization

Let us consider a polarizable force field based on (fixed) point charges and polarizable point dipoles. The charges will induce both atomic dipoles and the polarization of the continuum: these two fields will interact with each other.

Variational formulations can help sort things out:

\[ E(q, \mu, \sigma) = \frac{1}{2} q^\dagger V_q - \mu^\dagger E_q + \sigma^\dagger \Phi_q + \frac{1}{2} \mu^\dagger T \mu + \mu^\dagger \Omega \sigma + \frac{1}{2} \sigma^\dagger S \sigma \]

We get the equations by minimizing the energy:

\[ \frac{\partial E}{\partial \mu} = T \mu - E_q + \Omega \sigma = 0, \quad \frac{\partial E}{\partial \sigma} = S \sigma + \Phi_q + \Omega^\dagger \mu = 0 \]
An even larger system!

The coupled equations read

\[
\begin{pmatrix}
T & \Omega \\
\Omega^\dagger & S
\end{pmatrix}
\begin{pmatrix}
\mu \\
\sigma
\end{pmatrix}
= 
\begin{pmatrix}
E_q \\
-\Phi_q
\end{pmatrix}
\]

- Difficult problem! Usually, iterate on the dipoles and, for each iteration, fully solve for \(\sigma\)
- Four different, difficult quantities to compute:
  1. Solvent potential Solvent \(S\sigma\) \((N_s \times N_s)\)
  2. Solvent field Dipoles \(\Omega\sigma\) \((N_s \times M)\)
  3. Dipoles potential Solvent \(\Omega^\dagger\mu\) \((N_s \times M)\)
  4. Dipoles field dipoles \(T\mu\) \((M \times M)\)
- \(S\) makes the system globally ill-conditioned
- Convergence must be tight to compute forces (no cheating allowed!)
Conductor-like Screening Model

- We will focus on the simplest polarizable continuum solvation model
- COSMO treats the solvent as a conductor: simple boundary conditions!
- The results are scaled with an empirical factor to account for the dielectric nature of the solvent.
- The cavity is usually a Van der Waals cavity (union of spheres, one sphere per atom)
- Various parametrizations exist for the non-electrostatic terms
- The conductor approximation is perfectly reasonable for polar (\( \varepsilon > 20 \)) solvents
COSMO in pills:

- Goal: solving Poisson’s equation in the cavity with metallic boundary conditions

\[ \nabla^2 \varphi = -4\pi \rho \text{ in } \Omega, \quad \varphi = 0 \text{ on } \Gamma \]

- Separate the conductor contribution to the potential from the solute’s potential: \( \varphi = \Phi + W \)

- Recast as an integral equation:

\[ W(s) = \int_{\Gamma} \frac{\sigma(s')}{|s - s'|} =: (S\sigma)(s) \quad \Rightarrow \quad S\sigma = -\Phi \]

- Discretize somehow (usually, BEM).
Schwarz’s Domain Decomposition Method
\[ f''' = 0 \text{ in } [0, 1]; \quad f(0) = 0.5, \quad f(1) = 1.0 \]

\[ \Omega = ]0, 1[ \]
\[ f'' = 0 \text{ in } [0, 1]; \quad f(0) = 0.5, f(1) = 1.0 \]

\[ \Omega_1 = ]0, 0.75], \quad \Omega_2 = [0.25, 1[ \]
\[ f'' = 0 \text{ in } [0, 1]; \quad f(0) = 0.5, f(1) = 1.0 \]

\[ \Omega_1 = [0, 0.75], \quad \Omega_2 = [0.25, 1] \]
\[ f'' = 0 \text{ in } [0, 1]; \quad f(0) = 0.5, f(1) = 1.0 \]

\[ \Omega_1 = ]0, 0.75], \quad \Omega_2 = [0.25, 1[ \]
\[ f'' = 0 \text{ in } [0, 1]; \quad f(0) = 0.5, \ f(1) = 1.0 \]

\[ \Omega_1 = ]0, 0.75], \quad \Omega_2 = [0.25, 1[ \]
\[ f'' = 0 \text{ in } [0, 1]; \quad f(0) = 0.5, f(1) = 1.0 \]

\[ \Omega_1 = ]0, 0.75], \quad \Omega_2 = [0.25, 1[ \]
\[ f'' = 0 \text{ in } ]0, 1[; \quad f(0) = 0.5, f(1) = 1.0 \]

\[ \Omega_1 = ]0, 0.75], \quad \Omega_2 = [0.25, 1[ \]
Schwarz’s Domain Decomposition method

\[ \Omega = \bigcup_{i=1}^{M} \Omega_i \]

\[ S \sigma = -\Phi \]

\[ \downarrow ? \]

\[ \Gamma = \bigcup_{i=1}^{M} \Gamma_i \]

\[ S_i \sigma_i = g_i \]
Schwarz’s Domain Decomposition method

\[ \Omega = \bigcup_{i=1}^{M} \Omega_i \]

\[ S\sigma = -\Phi \]

\[ \Gamma = \bigcup_{i=1}^{M} \Gamma_i \]

\[ S_i\sigma_i = g_i \]

\[ \begin{align*}
\nabla^2 W &= 0 \quad \text{in } \Omega \\
W &= -\Phi \quad \text{on } \Gamma
\end{align*} \]
Schwarz’s Domain Decomposition method

\[ \Omega = \bigcup_{i=1}^{M} \Omega_i \]

\[ S\sigma = -\Phi \]

\[ \downarrow \]

\[ \Gamma = \bigcup_{i=1}^{M} \Gamma_i \]

\[ S_i \sigma_i = g_i \]

\[ \Omega = \bigcup_{i=1}^{M} \Omega_i \]

\[ \begin{cases} 
\nabla^2 W = 0 & \text{in } \Omega \\
W = -\Phi & \text{on } \Gamma 
\end{cases} \]

\[ \downarrow \]

\[ \begin{cases} 
\nabla^2 W_i = 0 & \text{in } \Omega_i \\
W_i = g_i & \text{on } \Gamma_i = \partial \Omega_i 
\end{cases} \]
Schwarz’s Domain Decomposition method

\[ \Omega = \bigcup_{i=1}^{M} \Omega_i \]

\[ S \sigma = -\Phi \]

\[ \downarrow \]

\[ \Gamma = \bigcup_{i=1}^{M} \Gamma_i \]

\[ S_i \sigma_i = g_i \]

\[ \bigg\{ \begin{array}{ll} \nabla^2 W = 0 & \text{in } \Omega \\ W = -\Phi & \text{on } \Gamma \end{array} \bigg\]
Local boundary conditions

We call $\Gamma_e^i$ the external portion of the local boundary $\Gamma_i$, $\Gamma_i^i$ the internal one.

We impose:

$$g_i = \begin{cases} -\Phi & \text{on } \Gamma_e^i \\ W_{i,N} & \text{on } \Gamma_i^i \end{cases}$$

where

$$W_{i,N}(s) = \frac{1}{|N_i(s)|} \sum_{j \in N_i(s)} W_j(s)$$

is the average potential computed from the solution on all the spheres that intersect $\Omega_j$ at $s$. 

Cancès, Maday, Stamm, JCP 139, 54111
A long story short: ddCOSMO equations

\[ \omega_{ij}(s) = \frac{\chi_{ij}(s)}{|N_i(s)|} \]

The ddCOSMO (coupled) integral equations become:

\[ (S\sigma_i)(s) = (1 - \sum_{j \in N_i(s)} \omega_{ij}(s))\Phi(s) + \sum_{j \in N_i(s)} \omega_{ij}(s)(\tilde{S}_{ij}\sigma_j)(s) \]
ddCOSMO facts:

😊 Discretization is easy: for each sphere, we use spherical harmonics and Lebedev grids (which are meant to integrate spherical harmonics!)

😊 The ddCOSMO matrix ($L$) is very block-sparse. A block corresponds to two spheres and is nonzero if and only if the two spheres intersect; furthermore, it is diagonally dominant and well conditioned.

😊 The approximation setting is variational (i.e., the more points/spherical harmonics, the better)

😊 Unfortunately, $L^* \neq L$: to compute the forces or other derivatives (including the ddCOSMO field) one needs to solve the adjoint equations

😊 The forces can be computed with arbitrary precision and always sum to zero

😊 The computations are fast, accurate and the cost and memory requirement scale linearly with respect to the solute’s size

Lipparini, Stamm, Cancès, Maday, Mennucci, JCTC 9, 3637
How fast are we?

Figure: Crambin (1EJG)
How fast are we?

**Figure: Crambin (1EJG)**

**Standard discretization**

$N_s = 30840$

- Matrix Inversion: 50’ 33”
- Iterative (PCG): 6’ 13”
- Iterative + FMM: 1’ 41”

*(Run on 2*E2650 Xeon, 16 cores)*

**Place your bets on ddCOSMO!**

$N_g = 146$ (exposed points: 27116)

$L_{max} = 6$

*(On my Laptop?)*
Some timings compared to other implementations:

<table>
<thead>
<tr>
<th>System</th>
<th>Atoms</th>
<th>CSC</th>
<th>CSC/FMM</th>
<th>ddCOSMO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vancomycin</td>
<td>377</td>
<td>20”</td>
<td>43”</td>
<td>&lt;1”</td>
</tr>
<tr>
<td>Hiv-1-GP41</td>
<td>530</td>
<td>1’26”</td>
<td>57”</td>
<td>&lt;1”</td>
</tr>
<tr>
<td>l-Plectasin</td>
<td>567</td>
<td>1’35”</td>
<td>1’17”</td>
<td>&lt;1”</td>
</tr>
<tr>
<td>Glutaredoxin</td>
<td>1277</td>
<td>8’54”</td>
<td>3’02”</td>
<td>&lt;1”</td>
</tr>
<tr>
<td>Glutaredoxin*</td>
<td></td>
<td>28’43”</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>UBCCH5B</td>
<td>2360</td>
<td>94’</td>
<td>6’18”</td>
<td>1”</td>
</tr>
<tr>
<td>Carboxylase</td>
<td>6605</td>
<td>777’</td>
<td>24’42”</td>
<td>3”</td>
</tr>
</tbody>
</table>

Table: Timings for the solution of the C-PCM/COSMO linear equations on 2*Xeon E2560 2GHz. *The computation was repeated without the $N^2$ storage for CSC.
Polarizable embedding models

ddCOSMO and polarizable force fields:

\[
\begin{pmatrix}
T & \frac{1}{2} f(\varepsilon)A & \frac{1}{2} f(\varepsilon)B \\
-B^* & L & 0 \\
-A^* & 0 & L^*
\end{pmatrix}
\begin{pmatrix}
\mu \\
X \\
S
\end{pmatrix}
=
\begin{pmatrix}
E_0 \\
g_0 \\
\Psi_0
\end{pmatrix}
\]

Algorithm:

- Compute the various Right-hand sides, make a guess for the dipoles
- Compute the dipoles potential \(-B^*\mu\) and \(\Psi\) function \(-A^*\mu\)
- Solve both the direct and adjoint ddCOSMO equations
- Compute the ddCOSMO field \(\frac{1}{2}f(\varepsilon)B X + \frac{1}{2}f(\varepsilon)A S\)
- Make a Jacobi step on the dipoles, use DIIS to accelerate convergence
Let’s look closer
Speeding up the other terms

- Thanks to ddCOSMO, continuum solvation *per se* is no longer the bottleneck nor the limiting factor in precision
- The JI/DIIS step on the dipoles scales quadratically
- The computation of the coupling terms scales quadratically - and with a big constant

Use the FMM!

All these terms can be seen as field/potentials, with some small modification: it is possible to use the FMM to achieve linear scaling in computational cost/memory!
Polarizable force fields and FMM

- The interaction between the solute and the solvent is purely electrostatic
- A FMM code able to manage point dipoles is however needed
- The dipole-dipole interactions are scaled and screened... (Near field!)

Some tuning is needed...

... to determine optimal FMM parameters and to properly account for non Coulombian interactions.
- Use Box = 12a.u. and $\ell = 8$ for the MM/MM interactions
- Use Box = 4a.u. and $\ell = 6$ for the MM/Solvent interactions

The terms involving the solvent...

... are problematic due to the high density of points and the relatively high precision required...
Introduction

Polarizable embedding models

ddCOSMO

Linear scaling

MM Only

![Graph showing the relationship between the number of atoms and elapsed time for MMPol solution. The graph is linear, indicating linear scaling.](image)
MM + ddCOSMO

![Graph showing the relationship between the number of atoms and elapsed time for various calculations including V(dip), ddCOSMO, E(cav), and Dipoles. The graph indicates a linear scaling with respect to the number of atoms.]
Introduction
Polarizable embedding models
ddCOSMO
Linear scaling

MM + ddCOSMO
**MM + ddCOSMO**

![Graph showing the linear scaling of MMPol/ddCOSMO solution](image-url)
In Conclusion...

- Available in Tinker for MD (including with PFF) and in Gaussian (including QM, QM/MM and QM/MMPol in a fully linear scaling implementation)
- Some molecular properties available, a lot yet to be done...
In Conclusion...

- Available in Tinker for MD (including with PFF) and in Gaussian (including QM, QM/MM and QM/MMPol in a fully linear scaling implementation)
- Some molecular properties available, a lot yet to be done...

What about the algorithms?

- ddCOSMO is what makes all of this possible!
- The FMM makes large computations feasible, however, the coupling terms are still very expensive
- Can we do better than that?
In Conclusion...

- Available in Tinker for MD (including with PFF) and in Gaussian (including QM, QM/MM and QM/MMPol in a fully linear scaling implementation)
- Some molecular properties available, a lot yet to be done...

What about the algorithms?

- ddCOSMO is what makes all of this possible!
- The FMM makes large computations feasible, however, the coupling terms are still very expensive
- Can we do better than that?

Thank you for your attention!