Theor Chem Acc (2003) 109:71–84
DOI 10.1007/s00214-002-0427-x

Theoretical
Chemistry Accounts

Regular article

Continuum solvation of large molecules described by QM/MM: a semi-iterative implementation of the PCM/EFP interface

Hui Li[1], **Christian S. Pomelli**[2], **Jan H. Jensen**[1]

[1] Department of Chemistry, University of Iowa, Iowa City, IA 52242, USA
[2] Dipartmento di Chimica Industriale, Universita di Pisa, Via Risorgimento 35, 56126 Pisa, Italy

Received: 18 October 2002 / Accepted: 13 November 2002 / Published online: 21 March 2003

Abstract. The direct inversion of the iterative subspace (DIIS) solution to the iterative integral equation formalism polarized continuum model (IEF–PCM, 2001 *Theor. Chem. Acc.* 105:1186) is applied to the effective fragment potential IEF–PCM interface (2002 J Chem Phys 116:5023). Compared to a direct matrix-inversion solution, the DIIS–PCM is up to an order of magnitude more efficient both in computing time and memory requirements for large systems. Multipole treatments of long-range electrostatic interactions further reduce the computing time by up to 50%. All the CPU intensive computations are parallelized. The data presented in this paper demonstrate that use of the iterative IEF–PCM is an efficient way to model bulk solvation of large biomolecules described by QM/MM.

Keywords: Continuum solvation – Quantum mechanics/molecular mechanics – Polarized continuum model – Protein

Introduction

In hybrid "quantum mechanics (QM)/molecular mechanics (MM)" methods the chemically active part of a molecular system is treated with QM and the rest with more approximate methods, such as MM [1, 2]. This approach has proven especially useful in the study of enzymatic reactions [3], where the enzyme active site and substrate is treated with a QM method and the rest of the protein with a MM force field. In these studies the description of the bulk solvent varies. In some cases [4, 5, 6] the solvent molecules are treated explicitly (e.g. by a TIP3P potential [7]), and their positions are averaged by Monte Carlo or molecular dynamics. The large number of energy and force evaluations necessary for this treatment necessitates a relatively simple QM method such as AM1 [8] or the empirical valence bond method [9]. Alternatively, bulk solvation can be represented by a continuum through the solution of the linearized Poisson–Boltzmann equation (LPBE). The QM/MM/LPBE interface can take the form of scaled MM charges [10] in a partially solvated QM/MM calculation [6] or a simultaneous solution of the QM/MM and LPBE equations [11, 12].

The interface between the effective fragment potential (EFP) method [13, 14] and the integrated equation formalism polarized continuum model (IEF–PCM) method [15], developed by Bandyopadhyay et al. [12], is an example of the latter approach. Similarly, the GAMESS/CHARMM QM/MM approach has recently been interfaced with the PCM method by Cui [16]. The EFP/PCM interface is similar to an all-ab-initio PCM calculation except that the electrostatic potential of the EFP region is due to its multipole representation of the electrostatic potential. The approach has been used to reproduce the pK_a of a residue in the 56-residue protease inhibitor turkey ovomucoid third domain by Li et al. [17]. However, the current implementation solves the PCM equation through a matrix inversion, which was found to be very central processing unit (CPU) intensive for protein-sized systems [17]. Tomasi, Pomelli, Barone and co-workers have addressed this general problem by the development of an iterative solution to the IEF–PCM equations, and demonstrated significant CPU time savings for both QM and MM solutes [18–23]. This paper discusses the implementation and testing of the iterative solution of the EFP/IEF–PCM equations, where the QM-self consistent-field (SCF) and IEF–PCM equations are simultaneously iterated to self-consistency. The paper presents the first use of the PCM method to calculate the solvation energy of a protein described by a QM/MM method.

Correspondence to: J. H. Jensen
e-mail: jan-jensen@uiowa.edu

The paper is organized as follows. After a review of the PCM, EFP/IEF–PCM, and iterative PCM, the theory behind the EFP/iterative-IEF–PCM is discussed, as are various approximations aimed at increasing the computational efficiency of the method. Second, the general computational methodology behind the calculations is discussed. Third, appropriate convergence criteria are determined, and the iterative and matrix-inversion solutions to the PCM equation are compared for a variety of solutes. Fourth, the rate of convergence is discussed. Fifth, the approximations discussed previously are tested numerically. Finally, our findings are summarized and future directions are discussed.

Theory

The PCM

In the PCM the solute molecule is placed in the bulk solvent described as a polarizable continuum with a dielectric constant ε. The cavity the solute molecule occupies in the bulk solvent can be defined in a variety of ways, the most popular of which is to use interlocking spheres centered at atoms or atomic groups. The surface of the cavity is the boundary between the solute and solvent. In the PCM the apparent surface charge (ASC) method is used to describe the electrostatic interaction between the solute and the bulk solvent. To numerically solve the electrostatic boundary equation, the continuous charge distribution on the boundary surface is divided into a set of point charges at a finite number of boundary surface elements, called tesserae. The resulting vector of ASCs, $\mathbf{q}$, is obtained by solving the matrix equation,

$$\mathbf{Cq} = \mathbf{g} \ , \tag{1}$$

where the vector $\mathbf{g}$ is a function of the molecular electrostatic potential of the solute $\mathbf{V}$, and $\mathbf{C}$ is a geometric matrix. Both $\mathbf{g}$ and $\mathbf{C}$ have different forms for different tessellation methods and different PCM formalisms. On the basis of the GEPOL–GB tessellation procedure [24], the $\mathbf{C}$ and $\mathbf{g}$ elements for the D-PCM are

$$\begin{aligned} C_{ii} &= \frac{1}{2}\frac{\varepsilon+1}{\varepsilon-1} + \frac{1.07\sqrt{4\pi a_i}}{8\pi R_i}, \quad C_{ij} = \frac{a_i(\vec{r}_i - \vec{r}_j)\hat{n}_i}{4\pi\left|\vec{r}_i - \vec{r}_j\right|^3}, \\ g_i &= -\frac{a_i}{4\pi}\frac{\partial V_i}{\partial \hat{n}_i} \ , \end{aligned} \tag{2}$$

where r_i, a_i, $\hat{n}_i$, and R_i are, respectively, the center, area, orthogonal unitary vector, and the sphere radius for tessera i.

For the C-PCM

$$\begin{aligned} C_{ii} &= 1.07\sqrt{\frac{4\pi}{a_i}}, \quad C_{ij} = \frac{1}{\left|\vec{r}_i - \vec{r}_j\right|} \\ \mathbf{g} &= -\mathbf{V} \ . \end{aligned} \tag{3}$$

For the isotropic IEF–PCM

$$\begin{aligned} \mathbf{C} &= \left(\frac{\mathbf{A}}{2} - \mathbf{D}\right)^{-1}\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{A}^{-1}\mathbf{S}\mathbf{A}^{-1}, \\ \mathbf{g} &= -\mathbf{V} \ , \end{aligned} \tag{4}$$

where

$$A_{ii} = a_i, \ A_{ij} = 0 \ , \tag{5}$$

$$S_{ii} = 1.07a_i\sqrt{4\pi a_i}, \quad S_{ij} = \frac{a_i a_j}{\left|\vec{r}_i - \vec{r}_j\right|} \ , \tag{6}$$

$$D_{ii} = -\frac{1.07a_i\sqrt{4\pi a_i}}{8\pi R_i}, \quad D_{ij} = \frac{a_i a_j(\vec{r}_i - \vec{r}_j)\hat{n}_j}{4\pi\left|\vec{r}_i - \vec{r}_j\right|^3} \ . \tag{7}$$

In all cases, $\mathbf{q}$ is used to represent the electrostatic potential of the solvent,

$$\Phi_{\mathrm{ASC}}(\vec{r}) = \sum_i^{N_{\mathrm{TS}}} \frac{q(i)}{|\vec{r} - \vec{r}_i|} \ , \tag{8}$$

which, in turn, polarizes the solute (with Hamiltonian H_0),

$$(H_0 + \Phi_{\mathrm{ASC}})|\Psi'\rangle = E'|\Psi'\rangle \ . \tag{9}$$

The total free energy of the solute–solvent system is

$$E_{\mathrm{total}} = \langle\Psi'|H_0|\Psi'\rangle + E_{\mathrm{NN}} + G_{\mathrm{elec}} \ , \tag{10}$$

where E_{NN} is the nuclear–nuclear repulsion energy and G_{elec} is the electrostatic interaction between the solute and the bulk solvent,

$$G_{\mathrm{elec}} = \frac{1}{2}\mathbf{q}\cdot\mathbf{V} \ , \tag{11}$$

and Ψ' is the solute wavefunction that minimizes E_{total}.

The solute nuclear potential, $\mathbf{V}_{\mathrm{N}}$, and electronic potential, $\mathbf{V}_{\mathrm{e}}$, can be separated and used to determine the corresponding $\mathbf{q}_{\mathrm{N}}$ and $\mathbf{q}_{\mathrm{e}}$ independently,

$$\begin{aligned} G_{\mathrm{elec}} &= \frac{1}{2}\mathbf{q}\cdot\mathbf{V} = \frac{1}{2}(\mathbf{q}_{\mathrm{N}} + \mathbf{q}_{\mathrm{e}})\cdot(\mathbf{V}_{\mathrm{N}} + \mathbf{V}_{\mathrm{e}}) \\ &= \frac{1}{2}(\mathbf{q}_{\mathrm{N}}\cdot\mathbf{V}_{\mathrm{N}} + \mathbf{q}_{\mathrm{N}}\cdot\mathbf{V}_{\mathrm{e}} + \mathbf{q}_{\mathrm{e}}\cdot\mathbf{V}_{\mathrm{N}} + \mathbf{q}_{\mathrm{e}}\cdot\mathbf{V}_{\mathrm{e}}) \ . \end{aligned} \tag{12}$$

This separation is useful since $\mathbf{q}_{\mathrm{e}}$ is affected by the penetration of Ψ' into the continuum, and can be corrected by various means [25, 26].

The integrated EFP/PCM method

Bandyopayhay et al. [12] have interfaced the IEF–PCM and the EFP method in the GAMESS program [27]. The EFP method [14] is a QM/MM method in which the

MM region is described by static multipoles (up to octupoles) and dipole polarizability tensors. The integrated EFP/PCM method differs from the full ab initio PCM only in the solute potential **V**. In the full ab initio PCM, **V** consists of two terms, the electronic term $\mathbf{V}_e$ and the nuclear term $\mathbf{V}_N$:

$$\mathbf{V} = \mathbf{V}_{QM} = \mathbf{V}_e + \mathbf{V}_N \ , \tag{13}$$

while in the EFP/PCM method, **V** consists of four terms, the electronic term $\mathbf{V}_e$, the nuclear term $\mathbf{V}_N$, the EFP multipole term $\mathbf{V}_{MUL}$ and the EFP induced-dipole term $\mathbf{V}_{POL}$:

$$\begin{aligned}\mathbf{V} &= \mathbf{V}_{QM} + \mathbf{V}_{EFP} \\ &= \mathbf{V}_e + \mathbf{V}_N + \mathbf{V}_{MUL} + \mathbf{V}_{POL} \ .\end{aligned} \tag{14}$$

The total free energy of the solute–solvent system is

$$E_{total} = \langle \Psi' | H_0 + \Phi_{EFP} | \Psi' \rangle + E_{NN} + E_{N-EFP} + G_{elec} \ , \tag{15}$$

where Φ_{EFP} and $E_{N\text{-}EFP}$ describe the interaction of the EFP with the electrons and nuclei of the QM region, respectively. G_{elec} is given by (cf. Eq. 12)

$$\begin{aligned}G_{elec} = &\frac{1}{2}(\mathbf{q}_N + \mathbf{q}_e + \mathbf{q}_{MUL} + \mathbf{q}_{POL}) \\ &\times (\mathbf{V}_N + \mathbf{V}_e + \mathbf{V}_{MUL} + \mathbf{V}_{POL}) \ .\end{aligned} \tag{16}$$

Similar to before, Ψ' is the wavefunction that minimizes E_{total} of Eq. (15), so that the QM charge density is polarized by the EFP and continuum solvent regions simultaneously.

In both the all-QM and QM/EFP implementation in GAMESS, Eq. (1) is solved by the direct matrix-inversion method,

$$\mathbf{q} = \mathbf{C}^{-1}\mathbf{g} \ , \tag{17}$$

which assumes the coefficient matrix **C** is nonsingular. For a small solute this method is very efficient since $\mathbf{C}^{-1}$, once computed and stored, can be used to compute a new **q** vector for each SCF iteration. However, it is less efficient for larger solutes because the CPU time and memory requirements scale as N_{TS}^3 and N_{TS}^2, respectively. For example, in an application of the EFP/PCM interface by Li et al. [17] to the calculation of the pK_a in the 56-residue protein turkey ovomucoid third domain, the calculation of the solvation energy took several days, while the gas-phase calculation on the same computer took only 25 min.

Iterative implementation of the IEF–PCM

The difficulty of applying the matrix-inversion implementation to large solutes, known for some time, has been addressed by implementing iterative solutions to Eq. (1). For IEF–PCM the iterative solution is done in two steps (see Appendix 1),

$$\mathbf{S}'\mathbf{q}_1 = -\mathbf{V} \ , \tag{18}$$

$$\mathbf{D}'\mathbf{q}_2 = -\frac{\mathbf{I}}{\varepsilon - 1}\mathbf{q}_1 \ , \tag{19}$$

with

$$\mathbf{q} = \mathbf{q}_1 + \mathbf{q}_2 \ , \tag{20}$$

$$S'_{ii} = 1.07\sqrt{\frac{4\pi}{a_i}} \quad , S'_{ij} = \frac{1}{|\vec{r}_i - \vec{r}_j|} \ , \tag{21}$$

$$D'_{ii} = \frac{1}{2}\frac{\varepsilon + 1}{\varepsilon - 1} + \frac{1.07\sqrt{4\pi a_i}}{8\pi R_i}, \quad D'_{ij} = \frac{a_i(\vec{r}_j - \vec{r}_i)\hat{\mathbf{n}}_i}{4\pi|\vec{r}_i - \vec{r}_j|^3} \ . \tag{22}$$

Here we note that $\mathbf{S}'$ in Eq. (21) is the C-PCM coefficient matrix in Eq. (3), so Eq. (18) is the C-PCM matrix equation. $\mathbf{D}'$ in Eq. (22) is similar to the D-PCM coefficient matrix in Eq. (2) and Eq. (19) is a D-PCM-like equation based on the scaled C-PCM polarization charges. In other words, the isotropic IEF–PCM can be decomposed into two phases, the C-PCM phase and the D-PCM-like phase. We refer to them as "phase 1" and "phase 2" hereafter.

The simplest iterative solution to the linear matrix equation,

$$\mathbf{M}\mathbf{X} = \mathbf{Y} \ , \tag{23}$$

is the point Jacobi iterative (PJI) method, where the coefficient matrix **M** is partitioned into a diagonal matrix $\mathbf{M}_0$ and an off-diagonal matrix $\mathbf{M}_1$:

$$\mathbf{M} = \mathbf{M}_0 + \mathbf{M}_1. \tag{24}$$

The PJI solution is

$$\mathbf{X}^{(n)} = \mathbf{M}_0^{-1}\left(\mathbf{Y} - \mathbf{M}_1\mathbf{X}^{(n-1)}\right) \ . \tag{25}$$

According to the Stein–Rosenberg theorem, convergence is guaranteed if **M** is a diagonal dominant matrix [28]. However, $\mathbf{S}'$ of Eq. (18) is not diagonally dominant using the current tessellation scheme GEPOL-GB (Appendix 2) and it is found empirically that the associated PJI approach is divergent. $\mathbf{D}'$ of Eq. (19) is diagonally dominant for well-tessellated cavities but not for general cases (Appendix 2) and the associated PJI may be either convergent or divergent. Other methods, such as a semi-iterative method, are necessary to solve the IEF–PCM equations. An earlier study [22] has tried various methods, such as conjugate gradients and damping functions, and found direct inversion of the iterative subspace (DIIS [29]) most effective.

The DIIS method is a semi-iterative method that predicts the next guess $\mathbf{X}^{(n)*}$ by linearly combining all $(n-1)$ or part of $(n-m)$ the previous iterates:

$$\mathbf{X}^{(n)*} = \sum_{k=m}^{n-1} \lambda_k \mathbf{X}^{(k)} \quad . \tag{26}$$

The weight factors, λ_k, are determined by a least-squares procedure that minimizes the quantity

$$S(\lambda) = \left| \sum_{k=m}^{n-1} \lambda_k \mathbf{e}^{(k)} \right|^2 \tag{27}$$

under the constraint

$$\sum_{k=m}^{n-1} \lambda_k = 1 \quad , \tag{28}$$

where the error vector for iteration k, $\mathbf{e}^{(k)}$, is defined as (cf. Eq. 26)

$$\begin{aligned} \mathbf{e}^{(k)} &= \mathbf{X}^{(k)} - \mathbf{X}^{(k)*}, \\ \mathbf{X}^{(k)} &= \mathbf{M}_0^{-1}\left(\mathbf{Y} - \mathbf{M}_1 \mathbf{X}^{(k)*}\right) \quad . \end{aligned} \tag{29}$$

Minimizing the quantity $S(\lambda)$ under the constraint in Eq. (28) is equivalent to minimizing the quantity

$$S(\lambda) + \mu\left(\sum_{k=m}^{n-1} \lambda_k - 1\right) \tag{30}$$

where μ is a Lagrange multiplier, or solving the linear equation

$$\begin{pmatrix} 0 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \mathbf{e}^{(m)} \cdot \mathbf{e}^{(m)} & \mathbf{e}^{(m+1)} \cdot \mathbf{e}^{(m)} & \mathbf{e}^{(m+2)} \cdot \mathbf{e}^{(m)} & \cdots & \mathbf{e}^{(n-1)} \cdot \mathbf{e}^{(m)} \\ 1 & \mathbf{e}^{(m)} \cdot \mathbf{e}^{(m+1)} & \mathbf{e}^{(m+1)} \cdot \mathbf{e}^{(m+1)} & \mathbf{e}^{(m+2)} \cdot \mathbf{e}^{(m+1)} & \cdots & \mathbf{e}^{(n-1)} \cdot \mathbf{e}^{(m+1)} \\ 1 & \mathbf{e}^{(m)} \cdot \mathbf{e}^{(m+2)} & \mathbf{e}^{(m+1)} \cdot \mathbf{e}^{(m+2)} & \mathbf{e}^{(m+2)} \cdot \mathbf{e}^{(m+2)} & \cdots & \mathbf{e}^{(n-1)} \cdot \mathbf{e}^{(m+2)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \mathbf{e}^{(m)} \cdot \mathbf{e}^{(n-1)} & \mathbf{e}^{(m+1)} \cdot \mathbf{e}^{(n-1)} & \mathbf{e}^{(m+2)} \cdot \mathbf{e}^{(n-1)} & \cdots & \mathbf{e}^{(n-1)} \cdot \mathbf{e}^{(n-1)} \end{pmatrix} \times \begin{pmatrix} \mu \\ \lambda_m \\ \lambda_{m+1} \\ \lambda_{m+2} \\ \vdots \\ \lambda_{n-1} \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{31}$$

by direct inversion of the error matrix.

Convergence is reached when the root-mean-square (RMS) of the error,

$$\mathrm{RMS} = \sqrt{\frac{\mathbf{e}^{(n)} \cdot \mathbf{e}^{(n)}}{N_{\mathrm{TS}}}} \quad , \tag{32}$$

is below a user-defined threshold.

This approach has been shown to result in good convergence for solutes described by point charges [22]. Next we consider the implementation of the DIIS iterative solution to the IEF–PCM/EFP equations, and its application to systems as large as proteins.

Iterative implementation of the IEF–PCM/EFP interface

The implementations of the iterative DIIS solution of the IEF–PCM equation for all-QM and QM/EFP in GAMESS are straightforward. The main difference between the two implementations is that the computational bottleneck shifts from the computation of the two-electron integrals to the iterative solution of the PCM equation on going from QM/PCM to QM/EFP/PCM calculations. The computational steps, and their associated costs, for a given SCF iteration are

1. The computation of $\mathbf{V}_e$ (cf. Eq. 14), which scales as $N_{\mathrm{AO}}^2 N_{\mathrm{TS}}$, where N_{AO} is the number of basis functions in the QM region.
2. The solution of Eqs. (18) and (19), which both scale as $N_{\mathrm{TS}}^2 N_{\mathrm{ITER}}$, where N_{ITER} is the total number of iterations needed for convergence.
3. The computation of $\langle \Psi' | \Phi_{\mathrm{ASC}} | \Psi' \rangle$ (cf. Eq. 9), which scales as $N_{\mathrm{AO}}^2 N_{\mathrm{TS}}$.

For our biochemical applications of the QM/EFP/PCM method the second step is the computational bottleneck, since $N_{\mathrm{AO}} \approx 350$, $N_{\mathrm{ITER}} \approx 600$, $N_{\mathrm{TS}} \approx 10{,}000$. However, for a larger QM region, steps 1 and 3 may dominate. The computational costs of all steps can in principle be reduced as discussed next.

The solute electrostatic potential

The solute electrostatic potential $\mathbf{V}$ is calculated at each tessera according to Eq. (14). The most CPU intensive part of this step is the calculation of $\mathbf{V}$ due to the electrons in the QM region at each tesserae representative point r_i,

$$V_e(i) = \mathrm{tr}\mathbf{P}\mathbf{U}(i) \quad , \tag{33}$$

where $\mathbf{P}$ is the density matrix in the atomic orbital basis $\{\chi\}$ and $\mathbf{U}(i)$ has the elements

$$U(i)_{\mu\nu} = \left\langle \chi_\mu \left| \frac{1}{|\vec{r} - \vec{r}_i|} \right| \chi_\nu \right\rangle \quad . \tag{34}$$

Hence the computation of $\mathbf{V}$ scales as $N_{\mathrm{AO}}^2 N_{\mathrm{TS}}$.

For a QM/EFP solute, typically only a small part of the solute is described ab initio and, hence, many tesserae are situated outside the density of the ab initio region (Fig. 1). For a distant tessera, i, the potential of the electrons in the ab initio region can be approximated by a multipole expansion without significant loss of accuracy,

$$V_e(i) = V_e^{\mathrm{Mult}}(i). \tag{35}$$

Here Stone's distributed multipole analysis [30] is used to generate $V_e^{\mathrm{Mult}}(i)$ and the expansion points are placed at each atom and bond midpoint, as in the EFP method.

Tessera i (at position x_i, y_i, z_i) is considered "distant" if, $x_i > x_{\max} + d_{\mathrm{ABI}}$ or $x_i < x_{\min} - d_{\mathrm{ABI}}$, where $x_{\max}$ and $x_{\min}$ are the largest and smallest x coordinates of the atoms in the QM region (and similarly for y and z) and d_{ABI} is 4.0 Å. Compared to a simple distance-dependent criterion (e.g. between r_i and the center of mass of the QM) region, this "box approach" allows for more aggressive screening for nonspherical QM regions. A more elaborate distance-dependent criterion, such as the minimum distance between r_i and a QM atom will significantly decrease the CPU time-saving since it must be performed N_{TS} times. Though the "box approach" is not strictly rotationally invariant, the default value of 4.0 Å for d_{ABI} is large enough that this does not present a practical problem, and in general the approximation is found to affect the final SCF energy by less than 10^{-5} hartrees.

The computing of the $U(i)$ elements in Eq. (34) is parallelized by using the distributed data interface [31] (DDI) in GAMESS. The conventional way to deal with the $U(i)$ integrals is to compute them for all the tesserae and store them on the hard disk, then read them from the disk for every tessera. This approach can be very time consuming and necessitate a large disk storage space. A direct method (similar to direct SCF) is implemented and is found to be more efficient, especially for parallel computing.

Iterative solution of the PCM equation

Equation (18) is solved iteratively by partitioning $\mathbf{S}'$ into two parts (cf. Eq. 24):

$$\mathbf{S}' = \mathbf{S}'_0 + \mathbf{S}'_1 \ . \tag{36}$$

The PJI equation is

$$\mathbf{q}_1^{(n)} = -(\mathbf{S}'_0)^{-1}\left(\mathbf{V} + \mathbf{S}'_1\,\mathbf{q}_1^{(\mathrm{n}-1)}\right) \ , \tag{37}$$

i.e.

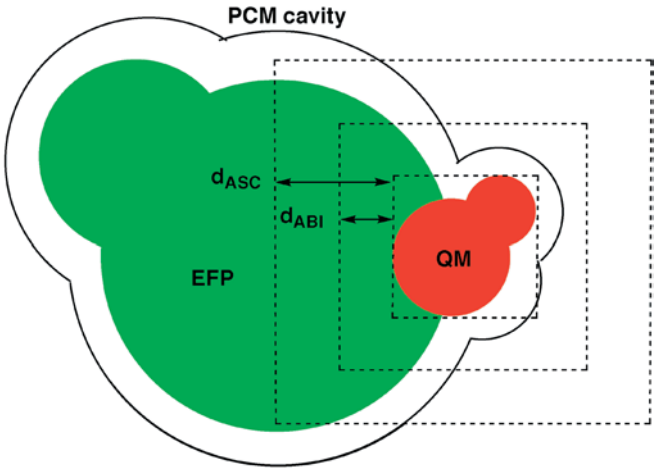

Fig. 1. The QM/EFP/PCM system. The tesserae outside of the d_{ABI} boundary see the QM region as Stone's multipole expansion points. The electrons in the QM region see the apparent surface charges (*ASCs*) outside d_{ASC} as regionwise multipole expansion points

$$q_1^{(n)}(i) = -\frac{1}{1.07\sqrt{4\pi/a_i}}\left(V(i) + \sum_{j\neq i}^{N_{\mathrm{TS}}} \frac{q_1^{(n-1)}(j)}{|\vec{r}_i - \vec{r}_j|}\right) \ . \tag{38}$$

Similarly, Eq. (19) is solved iteratively by partitioning $\mathbf{D}'$ into two parts,

$$\mathbf{D}' = \mathbf{D}'_0 + \mathbf{D}'_1. \tag{39}$$

The PJI equation is

$$\mathbf{q}_2^{(n)} = -(\mathbf{D}'_0)^{-1}\left(\frac{\mathbf{q}_1}{\varepsilon - 1} + \mathbf{D}'_1\,\mathbf{q}_2^{(\mathrm{n}-1)}\right) \ , \tag{40}$$

i.e.

$$\begin{aligned} q_2^{(n)}(i) = &-\left(\frac{1}{2}\frac{\varepsilon+1}{\varepsilon-1} + \frac{1.07\sqrt{4\pi a_i}}{8\pi R_i}\right)^{-1} \\ &\times \left(\frac{q_1(i)}{\varepsilon - 1} + \sum_{j\neq i} \frac{a_i(\vec{r}_j - \vec{r})_i \hat{n}_i}{4\pi|\vec{r}_i - \vec{r}_j|^3} q_2^{(n-1)}(j)\right) \ . \end{aligned} \tag{41}$$

Clearly, the computations of the last terms in Eqs. (38) and (41) both scale as N_{TS}, and must be repeated for N_{TS} charges until convergence is reached (in N_{ITER} total steps). Thus the entire procedure scales roughly as $N_{\mathrm{TS}}^2(N_{\mathrm{ITER},1} + N_{\mathrm{ITER},2})$. Empirically, we find that $N_{\mathrm{ITER},1} > N_{\mathrm{ITER},2}$ and that the magnitude of both can vary greatly from one SCF cycle to another.

To decrease the CPU time required for this step we have implemented the linear-scaling scheme due to Pomelli and Tomasi [23]. We briefly summarize the principle behind this method using Eq. (38) and note that similar considerations apply to Eq. (41). The calculation of the solvent electrostatic potential at each tessera is approximated by

$$\sum_{j\neq i}^{N_{\mathrm{TS}}}\frac{q^{(n-1)}(j)}{|r_i-r_j|}=\sum_{k}^{K}\sum_{j\neq i}^{N_{TS}(k)}\frac{q^{(n-1)}(j)}{|r_i-r_j|}$$
$$=\sum_{k}^{K_1}\sum_{j\neq i}^{N_{TS}(k)}\frac{q^{(n-1)}(j)}{|r_i-r_j|}+\sum_{k}^{K_2}\Lambda(i)_k+\sum_{k}^{K_3}Q(i)_k. \quad (42)$$

Here, contributions from each sphere (k, centered at $\mathbf{R}_k$) on the cavity surface, are identified and approximated for large $|\mathbf{R}_i-\mathbf{R}_k|$ (where $\mathbf{r}_i$ is on a sphere with center $\mathbf{R}_i$). For the K_2 spheres for which $R_2>|\mathbf{R}_i-\mathbf{R}_k|>R_1$ the electrostatic potential of the charges belonging to sphere k is approximated by a multipole expansion centered at $\mathbf{R}_k$ and truncated at quadrupoles [$\Lambda(i)_k$]. For the remaining K_3 spheres for which $|\mathbf{R}_i-\mathbf{R}_k|>R_2$, the multipole expansion is truncated at the monopole term [$Q(i)_k$]. For molecules containing only neutral groups the last term in Eq. (42) can be neglected and in this case the CPU time is found to vary linearly with N_{TS}. The computations of Eqs. (38), (41) and (42) are parallelized with DDI.

At the first SCF cycle, we use the following initial guesses:

$$\mathbf{q}_1^{(0)*}=-\frac{\mathbf{I}}{20}\cdot(\mathbf{S}'_0)^{-1}\cdot\mathbf{V}\ , \quad (43a)$$

$$\mathbf{q}_2^{(0)*}=-\frac{\mathbf{I}}{\varepsilon}\cdot\mathbf{q}_1\ . \quad (43b)$$

The initial guess for the phase 1 charges is obtained from Eq. (37) by setting $\mathbf{q}_1{}^{(n-1)}=0$. Empirically, we found that scaling by 1/20 reduces $N_{\mathrm{ITER},1}$, though modestly. The initial guess for the phase 2 charges is obtained by solving the common approximation $\mathbf{q}\approx\frac{\varepsilon-1}{\varepsilon}\mathbf{q}_1$, for $\mathbf{q}_2$ (recall that $\mathbf{q}_1$ corresponds to C-PCM charges and $\mathbf{q}=\mathbf{q}_1+\mathbf{q}_2$). For subsequent SCF cycles, $\mathbf{q}_1$ and $\mathbf{q}_2$ from the last SCF cycle are used instead of Eq. (43a).

Electrostatic interactions between solute electrons and the ASCs

The electrostatic potential due to the ASCs is included in the Fock matrix of the QM region at each SCF iteration, so that Ψ' of Eq. (15) minimizes E_{total}.

$$F_{\mu\nu}^{\mathrm{QM}'}=F_{\mu\nu}^{\mathrm{QM}}+T_{\mu\nu}\ , \quad (44)$$

where

$$T_{\mu\nu}=\sum_{i}^{N_{\mathrm{TS}}}U(i)_{\mu\nu}q(i)\ , \quad (45)$$

where $U(i)_{\mu\nu}$ is defined in Eq. (34). The cost of computing $\mathbf{T}$ is $N_{\mathrm{AO}}^2N_{\mathrm{TS}}$.

Since many tesserae are far away from the ab initio region it is possible to apply the multipole expansion approximation discussed in the previous subsection:

$$T_{\mu\nu}=\sum_{k}^{K_1}\sum_{i}^{N_{\mathrm{TS}}(k)}U(i)_{\mu\nu}q(i)+\sum_{k}^{K_2}\sum_{i}^{N_{\mathrm{TS}}(k)}U(i)_{\mu\nu}q(i)$$
$$=\sum_{k}^{K_1}\sum_{i}^{N_{\mathrm{TS}}(k)}U(i)_{\mu\nu}q(i)+\sum_{k}^{K_2}\langle\chi_\mu|\Lambda(r)_k|\chi_\nu\rangle\ . \quad (46)$$

The K_2 spheres, for which this approximation is used, are determined by the "box approach" described already (Fig. 1). Tessera i (on a sphere centered at position X_i, Y_i, Z_i) is considered "distant" if $X_i>x_{\max}+d_{\mathrm{ASC}}$ or $X_i<x_{\min}-d_{\mathrm{ASC}}$, where $x_{\max}$ and $x_{\min}$ are the largest and smallest x coordinates of the atoms in the QM region (and similarly for y and z). The evaluation of the $\mathbf{T}$ elements in Eqs. (45) and (46) is also parallelized with DDI and a direct method is implemented to avoid hard disk work.

Computational methodology

For protein QM/EFP/PCM calculations, the polarizability tensors are not included in the EFP description. The reason is somewhat complicated. Within the EFP method intrafragment energies (including interactions between induced dipoles) are not evaluated since the internal EFP geometry is frozen and the internal energy change due to polarization of the EFP is treated by the linear response approximation (giving rise to the factor of 1/2 in Eq. 16). However, the ACSs, which have contributions from the induced dipoles, are interacting, leading to an error in the EFP/PCM energy. For small neutral EFPs (such as the water EFPs used with the original implementation [12]) this error is probably small, and is out-weighed by a more accurate description of EFP/EFP and EFP/QM interaction energies since they include polarization terms. However, for a system described by a single large EFP with many charged functional groups, the inclusion of polarizable points leads to large errors and in some cases to a divergence of the energy. A previous study by some of us [17] has shown that sufficiently accurate solvation energies can be computed for large systems without the inclusion of dipole-polarizability tensors in the EFP region.

The details of constructing and using EFPs in systems with covalent QM/EFP boundaries can be found elsewhere [14, 17, 34, 35].

All calculations were performed with a locally modified version of the GAMESS program [27] on four-CPU IBM RS/6000 44P 270 workstations.

Results and discussion

Iterative convergence criterion

The iterative solution to the PCM equation at each SCF cycle is considered converged when the RMS of the iterative error (Eq. 32) falls below a convergence criterion. Because the average absolute charge on the tesserae varies from 0.002 au for neutral and less polar molecules to 0.005 au for charged and more polar molecules, the convergence criterion must be considerably smaller. The final SCF energies and CPU time usage obtained for four molecules (Fig. 2) are compared in Table 1 using four

convergence criteria: 10^{-5}, 10^{-6}, 10^{-7} and 10^{-8} au. Comparisons of the final SCF energies to the matrix-inversion solutions show that the convergence criterion needed to reproduce the SCF to within 10^{-5} hartrees is size-dependent. A criterion of 10^{-7} au is adequate for systems with around 1,000 tesserae or fewer, while 10^{-8} is needed for a system with 10,000 tesserae. Use of the 10^{-8} convergence criterion does not significantly increase the required CPU time for smaller molecules, so this criterion is used for all molecules.

Since the electron density changes in the SCF course, looser criteria could be used in the early SCF cycles when the density is far from convergence. The final SCF energy and the CPU time obtained by using looser iterative convergence criteria in the early SCF cycles where the density change is above 0.01 au are listed in Table 2. The data indicate that the accuracy of the SCF final energy is determined only by the final criterion (i.e. 10^{-8} au), indicating that obtaining very accurate ASCs is unnecessary in the early cycles.

The energies and density changes for each SCF iteration for the molecule OMTKY3-Lys55-a are shown in Fig. 3. Since the PCM is not a variational method, the energies may be artificially below the final SCF energy, owing to the inaccuracy in the determination of the ASC with a loose convergence criterion; however, subsequent use of the tighter convergence criterion recovers this error. The convergence of the density is insensitive to the quality of the ASCs.

In the current implementation both the loose criterion and the SCF density change, which defines the early SCF cycles, can be defined by the user, with the default values 5×10^{-4} and 0.01 au. For a system with around 10,000 tesserae, the use of 5×10^{-4} au convergence criteria for such early SCF cycles can reduce the computing time by up to 28%.

Comparison of the matrix-inversion and iterative PCM solutions

The final SCF energies and requisite CPU times obtained for a variety of molecules (cf. Fig. 2) using both the matrix inversion and iterative IEF–PCM (using the convergence criteria described previously) are compared in Table 3. The data show that the matrix-inversion and iterative implementations consistently produce SCF energies that agree to within 10^{-5} hartrees.†

For small molecules described by around 1,000 tesserae or fewer, the matrix-inversion method tends to be faster than the iterative version. The exact relative timings depend on the description of the solute, since the evaluation of the solute potential is significantly faster for a QM/EFP solute. For example, for GKG-H the iterative method is slower than the matrix-inversion method for the QM solute, but not for the QM/EFP solute. For larger QM/EFP solutes, the iterative method is more than 20 times faster than the matrix-inversion method, thus shortening the run time from days to hours on our computers. In addition, the memory requirements are significantly smaller for the iterative method as discussed in the next subsection.

Performance of the DIIS method

In Sect. 2 it was argued that the PJI method diverges for IEF–PCM, and that semi-iterative methods such as DIIS are necessary. The RMS error in the PCM iteration phases 1 and 2 of the first SCF cycle for GKG-H, with PJI and DIIS, is shown in Fig. 4. A "tight" criterion of 10^{-8} is used for illustrative purposes. Both PJI iterations clearly diverge, while DIIS starts to force the convergence from the third step when interpolation is available. The first step is the initial guess and the second step has only one previous step; thus no interpolation can be done. So the first two steps of phase 1 are identical for both PJI and DIIS iterations.

A plot similar to that in Fig. 4 is shown in Fig. 5a, for OMTKY3-Lys55-a. Clearly, more iteration steps are needed to converge a larger number of ASCs, and empirically we find that approximately $\mathbf{N}_{\mathrm{TS}}/20$ iterations are required for convergence. The DIIS needs $2 \times N_{\mathrm{DIIS}} \times N_{\mathrm{TS}}$ words of core memory to store the previous iterates and their error vectors, where N_{DIIS} is the maximum number of iterations. For OMTKY3-Lys55-a the memory requirement is thus around 10 megawords, still an order of magnitude less than that required for the matrix-inversion method. However, for very large molecules, both N_{TS} and N_{DIIS} are very large and the core memory (around ${N_{\mathrm{TS}}}^2/10$ words) becomes significant.

The memory requirement can be addressed by storing data on a hard disk. However, the inversion of the $N_{\mathrm{DIIS}} \times N_{\mathrm{DIIS}}$ error matrix may become a computational bottleneck for large N_{TS}. Another solution is to only use the last N'_{DIIS} error vectors (cf. Eq. 27) as is commonly done for SCF convergence. The convergence rate for several values of N'_{DIIS} for OMTKY3-Lys55-a is shown in Fig. 5b. Clearly, the performance of the IEF–PCM DIIS method is very sensitive to the size of the iterative subspace. DIIS with N'_{DIIS} smaller than 200 shows no sign of convergence, while the convergence rate is very low for $N'_{\mathrm{DIIS}} = 200$ and 400, so neither meets the convergence criterion in 1,000 steps. Thus, limiting the size of the iterative subspace to reduce the memory requirements for large systems does not seem like a promising approach.

† The original EFP/PCM-by-matrix-inversion interface by Bandyopayay et al. used the following approximations (cf. Eq. 16: $\frac{1}{2}(\mathbf{q}_e \cdot \mathbf{V}_{\mathrm{MUL}} + \mathbf{q}_{\mathrm{MUL}} \cdot \mathbf{V}_e) \approx \mathbf{q}_{\mathrm{MUL}} \cdot \mathbf{V}_e$ and $\frac{1}{2}(\mathbf{q}_e \cdot \mathbf{V}_{\mathrm{POL}} + \mathbf{q}_{\mathrm{POL}} \cdot \mathbf{V}_e) \approx \mathbf{q}_{\mathrm{POL}} \cdot \mathbf{V}_e$. These approximations are adequate for small and neutral molecules, but not for large and charged molecules. In this implementation, all terms are computed explicitly so that the final SCF energies agree numerically with the iterative results.

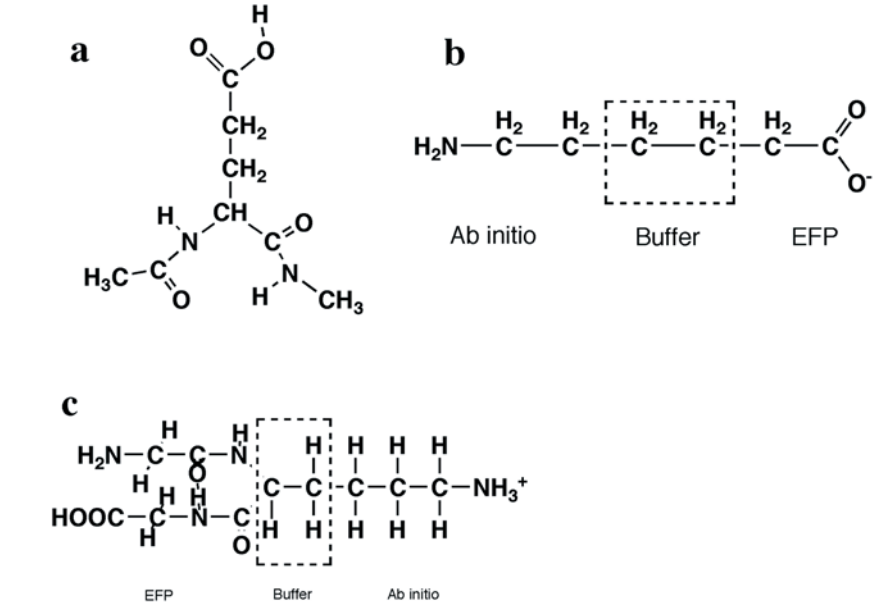

Fig. 2a–e. Schematic representations of the molecules used in this work. **a** Glu-model-a: a small model of a glutamic acid residue in a protein; Glu-model-b corresponds to the deprotonated form. **b** 6-Aminocaproate and **c** the tripeptide glycyl–lysyl–glycine, GKG-H, both divided into a QM, buffer, and EFP region. **d** OMTKY3-Lys55-a or OMTKY3-Lys55-b: ab initio/buffer/EFP regions (*red*/*blue*/*green*) division of turkey ovomucoid third domain, with Lys55 and Tyr20 are described by QM. a and b refer to protonated and deprotonated Lys55, respectively. **e** OMTKY3-glu43-a or OMTKY3-glu43-b: same as for **d** but with Glu43 described by QM

Table 1. The final SCF energy (hartrees) and CPU time (minutes) of the iterative IEF –PCM computation for four molecules by using various convergence criteria. The digits in the energy that differ from matrix-inversion values (in Table 3) are *underlined*. N_{TS} is the number of tesserae on the solute cavity surface, reflecting the size of the molecule. No polarizable points are used in the QM/EFP method. The solvent is water with $\varepsilon = 78.39$. The multipole expansion approximation methods are turned off

Molecules	Method	N_{TS}	Iterative convergence criterion			
			10^{-5}	10^{-6}	10^{-7}	10^{-8}
Acetic acid	QM	163	−227.8237304421 0.8	−227.8237325730 0.8	−227.8237330147 0.8	−227.8237330138 0.8
GKG-H	QM/EFP	1,121	−222.7315447940 2.6	−222.7315036907 2.4	−222.7314949153 2.7	−222.7314946143 2.7
OMTKY3-Lys55-a	QM/EFP	10,060	−691.6263856511 127.0	−691.6109597206 167.2	−691.6088688428 165.5	−691.6088567959 225.9
OMTKY3-Lys55-b	QM/EFP	10,064	−691.1603773378 133.2	−691.1456263586 174.1	−691.1432956617 179.5	−691.1432639944 239.5

Increasing computational efficiency

The use of multipole expansion methods to further reduce the computing time for three steps in the PCM calculation was described previously. This subsection discusses the resulting CPU time savings.

The solute electrostatic potential

The final SCF energy and CPU time of the iterative IEF–PCM calculation of OMTKY3-Lys55-a and OMTKY3-Lys55-b, using different cutoff values (d_{ABI}) for "distant" tesserae for which the solute electrostatic potential is

Table 2. The final SCF energy (hartrees) and CPU time (minutes) of the iterative isotropic IEF–PCM computation for four molecules by using various "loose" iterative convergence criteria at the early SCF cycles where the density change is above 0.01 au. After the density change falls below 0.01 au, the criterion switches to 10^{-8}

Molecules	Method	N_{TS}	Loose criteria for early SCF cycles			Tight criterion
			5×10^{-4}	5×10^{-5}	5×10^{-6}	10^{-8}
Acetic acid	QM	163	−227.8237330214 0.8	−227.8237330144 0.8	−227.8237330138 0.8	−227.8237330138 0.8
GKG-H	QM/EFP	1,121	−222.7314945843 2.5	−222.7314945821 2.7	−222.7314945690 2.7	−222.7314946143 2.7
OMTKY3-Lys55-a	QM/EFP	10,060	−691.6088564955 175.9	−691.6088567447 186.9	−691.6088558327 209.3	−691.6088567959 225.9
OMTKY3-Lys55-b	QM/EFP	10,064	−691.1432704623 172.7	−691.1432704964 183.5	−691.1432702452 208.6	−691.1432639944 239.5

represented by multipoles, are listed in Table 4. The total SCF energy can be reproduced to within 10^{-5} hartrees, even when using a very aggressive d_{ABI} value of 2.0 Å. This cutoff classifies 88% of all tesserae as distant from the ab initio region, and reduces the total CPU time by 18%. We use a default value for d_{ABI} of 4.0 Å in this implementation since the accuracy of the SCF energy is somewhat improved at little CPU time expense.

Iterative solution of the PCM equations

The final SCF energy and CPU time of the iterative IEF–PCM calculation of OMTKY3-Lys55-a and OMTKY3-Lys55-b for different values of R_1 and R_2 used to determine whether pairs of spheres are near (K_1), mid-range (K_2) or distant [(K_3) cf. Eq. (42)] are listed in Table 5. The total SCF energy can be reproduced to within 2×10^{-4} hartrees by using R_1/R_2 values of 15/30 Å. These cutoffs classify 39% and 2% of all pairs of tesserae as mid-range and distant, respectively, and reduce the total CPU time by 16%. Using smaller R_1/R_2 values leads to errors in the SCF energy of up to 2×10^{-3} hartrees, which is unacceptable for most applications.

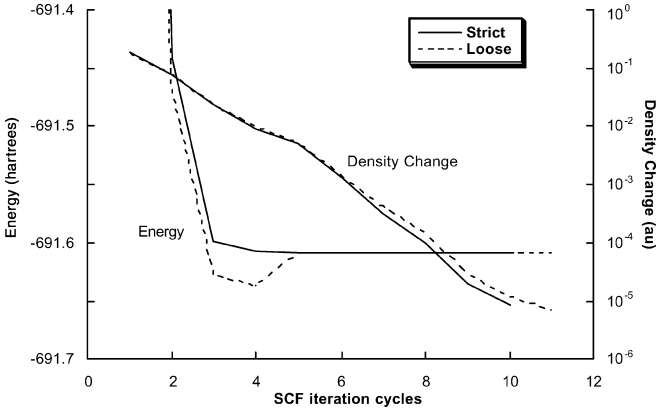

Fig. 3. The energies and density changes along the SCF course in the iterative isotropic IEF–PCM calculation for the molecule OMTKY3-Lys55-a by using a strict criterion (10^{-8} au) and a loose criterion (5×10^{-4} au) in the early SCF cycles where the density changes are above 0.01 au. After the density change falls below 0.01 au, the strict criterion is used till the SCF converges. The intermediate energies, which include the PCM electrostatic interaction, may be artificially lower than the final energy owing to use of the loose criterion. Though the energy is sensitive to the quality of the ASCs, the SCF convergence is not; thus similar density changes are found for the strict and loose criteria

Electrostatic interactions between solute electrons and the ASCs

The final SCF energy and CPU time of the iterative IEF–PCM calculation of OMTKY3-Lys55-a and OMTKY3-Lys55-b for different values of d_{ASC} in the classification of distant ASC regions in the multipole expansion approximation of the solvent potential (cf. Eq. 46) are listed in Table 6. Even a d_{ASC} value of 15 Å can result in SCF energy errors of greater than 10^{-3}, while the CPU time savings are very modest because the step time is a small part of the total time. For larger molecules with more tesserae and/or more basis functions, the saving of the total CPU time may be significant; however, d_{ASC} must be larger than 15 Å to ensure the 10^{-4} hartree accuracy for the SCF energy. We use a default value of 20 Å, which does not result in any CPU timesaving for the systems considered here.

Total CPU time savings and parallel efficiency

The combined use of the multipole expansions described previously leads to a 33% decrease in CPU time for OMTKY3-Lys55-a. When combined with the CPU time saving obtained by using a loose ASC-convergence criterion for early SCF cycles (cf. Sect. 4.1), an overall 52% reduction in CPU time is obtained for OMTKY3-Lys55-a. This CPU time-reduction is expected to increase for even larger systems.

Virtually all aspects of the PCM solution algorithm have been parallelized using the distributed data interface (DDI) in GAMESS. Timings for 1–4 nodes on a dedicated four-CPU RS/6000 44P 270 machine for OMTKY3-Lys55-a are listed in Table 7. Timings are listed for the SCF step where the ACS-convergence criterion decreases from 5×10^{-4} to 10^{-8}, because that step (in general) is the most time-consuming SCF step (i.e. N_{ITER} is largest). It is evident that all parts of the

Table 3. The comparison between the matrix inversion and iterative isotropic IEF–PCM in terms of the final SCF energy (hartrees) and CPU time (minutes). The CPU time needed for the gas-phase energy calculation is also listed for comparison. The initial loose iterative convergence criterion is 5×10^{-4}. EFP(p) denotes the use of polarizable points in the EFP region

Molecule	Method	N_{TS}	Final SCF energy		CPU time		
			Inversion	Iterative	Gas	Inversion	Iterative
Methanethiol	QM	99	−437.2392629373	−437.2392629361	0.1	0.1	0.7
Methanethiolium	QM	101	−437.7038865702	−437.7038865703	0.0	0.1	0.7
Methylammonium	QM	105	−95.6852579219	−95.6852579253	0.0	0.1	0.6
Methylamine	QM	105	−95.2169559843	−95.2169559836	0.0	0.1	0.6
Imidazolium	QM	160	−225.2959240179	−225.2959240169	0.4	0.5	1.1
Imidazole	QM	162	−224.8308136448	−224.8308136464	0.4	0.5	1.0
Acetate	QM	162	−227.3704781631	−227.3704781648	0.3	0.4	0.9
Aceticacid	QM	163	−227.8237330175	−227.8237330214	0.2	0.3	0.8
Phenol	QM	224	−305.5708382874	−305.5708382867	1.1	1.4	2.0
Phenolium	QM	247	−305.1036153418	−305.1036153424	1.5	2.7	3.4
6-Aminocaproate	QM/EFP(p)	314	−197.3163874626	−197.3163949099	0.6	1.3	1.9
6-Aminocaproate	QM/EFP	314	−197.3096895937	−197.3096895805	0.6	0.9	1.4
6-Aminocaproate	QM	320	−438.5400981377	−438.5400981347	6.4	6.9	8.4
Glu-model-a	QM	451	−719.5009997138	−719.5009997296	8.9	10.2	10.7
Glu-model-b	QM	459	−719.0468693271	−719.0468692934	12.0	14.5	15.9
GKG-H	QM	1,110	−908.1238693181	−908.1238693401	17.0	25.9	27.0
GKG-H	QM/EFP(p)	1,121	−222.7401532659	−222.7401472857	0.8	6.4	4.9
GKG-H	QM/EFP	1,121	−222.7314945397	−222.7314945843	0.8	4.5	2.5
OMTKY3-Lys55-a	QM/EFP	10,060	−691.6088566221	−691.6088564955	25.8	3,972.9	175.9
OMTKY3-Lys55-b	QM/EFP	10,064	−691.1432702564	−691.1432704623	25.3	5,391.5[a]	172.7
OMTKY3-Glu43-a	QM/EFP	10,066	−827.3567145031	−827.3567155047	22.5	5,448.6[a]	183.3
OMTKY3-Glu43-b	QM/EFP	10,075	−826.8990567991	−826.8990566155	28.6	3,974.8	210.1

[a]Two nodes instead of three

PCM solution scale well up to four nodes. While the ASC iterations dominate the CPU time for this SCF step, the other terms may dominate when N_{ITER} is small (e.g. in the last few SCF steps) or for systems with a larger QM region, and it is this important that all aspects of the iterative PCM method are parallelized.

Conclusions

The DIIS solution to the IEF–PCM [22] is applied to the EFP/IEF–PCM interface [12]. The iterative IEF–PCM equation is shown to be inherently divergent, so the use of a method like DIIS to force convergence is essential (Fig. 4). The entire iterative subspace must be used in the DIIS solution to achieve convergence for large systems (Fig. 5b). The accuracy of the iterative solutions is verified by reproducing SCF energies, obtained by a matrix-inversion solution of the IEF–PCM equation, to within 10^{-5} hartrees for a variety of molecules (Table 3, Fig. 2). For a large solute with 10,000 tesserae, the iterative solution is an order of magnitude faster than the matrix-inversion method.

For a typical application of the EFP/IEF–PCM method, the solute is described by a small ab initio region surrounded by a large EFP region. In this case the iterative solution of the IEF–PCM equation, performed at each SCF step, is the computational bottleneck. Several methodological innovations are introduced to reduce the CPU time:

1. A looser convergence criterion for the PCM equation solution is used at early SCF steps (Table 2).
2. Various multipole expansions are introduced to treat long-range electrostatic interactions (Tables 2, 5, 6).
3. Virtually all aspects of the solving the IEF–PCM equations are parallelized (Table 7).

The first two implementations reduce the CPU time by up to around 50%, and good scaling is found for up to four nodes (the highest considered).

The data presented in this paper demonstrate that use of the iterative IEF–PCM method is an efficient way to model bulk solvation of biomolecules described by QM/MM. However, for significantly larger solutes, storing

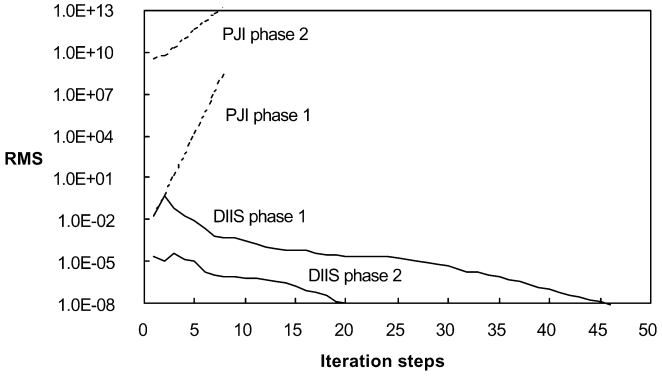

Fig. 4. Root mean square (*RMS*) of the errors of the iteration for phases 1 and 2 of the first SCF cycle in the iterative isotropic IEF–PCM. The molecule is GKG-H (method QM/EFP, no polarizable points) with $N_{TS}=1,121$. Both point Jacobi iterative (*PJI*) phases are divergent, while the direct inversion of the iterative subspace (*DIIS*) phases are convergent

and diagonalizing the DIIS matrix may prove prohibitive. Further improvements, in particular the use of the DEFPOL tessellation scheme [36], which gives rise to a smaller number of tesserae for large systems than the current GEPOL-GB scheme, are planned to address this concern.

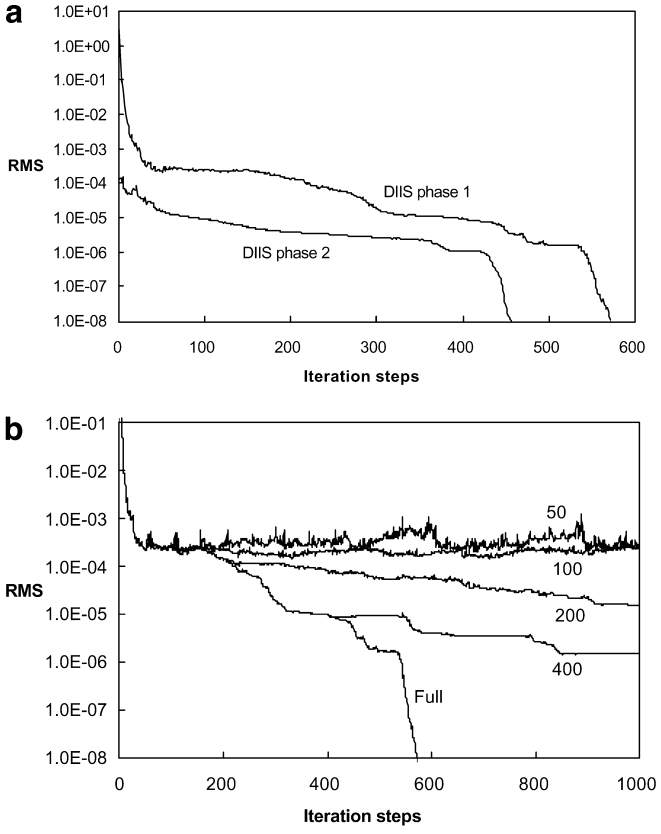

Fig. 5. **a** RMS of the errors of the iteration for phases 1 and 2 of the first SCF cycle in the iterative isotropic IEF–PCM. The molecule is OMTKY3-Lys55-a with $N_{TS} = 10{,}060$. The DIIS phases are convergent. **b** DIIS performance depends on its size. Five DIIS sizes are tested in phase 1 of the first SCF cycle in the iterative isotropic IEF–PCM calculation for the molecule OMTKY3-Lys55-a and the RMSs are plotted verses the iteration steps. The *numbers* beside the *lines* are the maximum size of the DIIS. *Full* means all the previous iterates are included. The forking points are where the maximum DIIS size is reached and the initial iterates start to be discarded from the DIIS

Acknowledgements. This work was supported by a Research Innovation Award from the Research Corporation, and a type G starter grant from the Petroleum Research Fund. H.L. gratefully acknowledges a predoctoral fellowship from the Center for Biocatalysis and Bioprocessing at the University of Iowa. The calculations were performed on IBM RS/6000 workstations obtained through a CRIF grant from the NSF (CHE-9974502). We thank Michael Schmidt for a careful reading of the manuscript, and Laurent Jay for useful discussions.

Appendix 1

By substituting Eq. (4) into Eq. (1), we obtain the isotropic IEF–PCM matrix equation:

$$\left(\frac{\mathbf{A}}{2} - \mathbf{D}\right)^{-1}\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{A}^{-1}\mathbf{S}\mathbf{A}^{-1}\mathbf{q} = -\mathbf{V} \ . \tag{47}$$

Equation (47) can be rearranged,

$$\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{A}^{-1}\mathbf{S}\mathbf{A}^{-1}\mathbf{q} = -\left(\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{V} \ , \tag{48}$$

and the right-hand side separated to give

$$\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{A}^{-1}\mathbf{S}\mathbf{A}^{-1}\mathbf{q} = -\left[\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right) - \frac{\mathbf{A}}{\varepsilon-1}\right]\mathbf{V} \ . \tag{49}$$

Equation (49) can be written as two independent equations,

$$\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{A}^{-1}\mathbf{S}\mathbf{A}^{-1}\mathbf{q}_1 = -\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{V} \tag{50}$$

and

$$\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2} - \mathbf{D}\right)\mathbf{A}^{-1}\mathbf{S}\mathbf{A}^{-1}\mathbf{q}_2 = \frac{\mathbf{A}}{\varepsilon-1}\mathbf{V} \ , \tag{51}$$

Table 4. The final SCF energy (E_{SCF} in hartrees) and CPU time of the iterative isotropic IEF–PCM computation for the molecules OMTKY3-Lys55-a and OMTKY3-Lys55-b by using various d_{ABI} to define the distant tesserae in the multipole expansion representation of the electronic part of the solute potential. N_{apx} is the number of distant tesserae that see the electronic potential as Stone's multipole expansion points. T_{total} is the total CPU time (minutes) to finish the job and T_{step} is the step time (seconds) to finish the electronic potential evaluations for all the tesserae (for one SCF cycle)

Molecules	N_{TS}		$d_{ABI} = 2.0$ Å	$d_{ABI} = 4.0$ Å	$d_{ABI} = 8.0$ Å	$d_{ABI} = \infty$
OMTKY3-Lys55-a	10,060	E_{SCF}	−691.6088471936	−691.6088555524	−691.6088555013	−691.6088564955
		T_{total} (m)	144.0	147.7	157.8	175.9
		T_{step} (s)	22.85	43.31	93.38	197.38
		N_{apx}	8,904	7,856	5,323	0
OMTKY3-Lys55-b	10,064	E_{SCF}	−691.1432605133	−691.1432697722	−691.1432694626	−691.1432704623
		T_{total} (m)	141.7	144.6	153.9	172.7
		T_{step} (s)	23.49	41.94	91.39	193.84
		N_{apx}	8,904	7,856	5,323	0

Table 5. The final SCF energy (E_{SCF} in hartrees) and CPU time of the iterative isotropic IEF–PCM computation for two molecules by using various criteria (R_1 and R_2) to classify the regionwise vicinities oftesserae in the multipole expansion representation of the solvent electrostatic potentials. N_{REG} is the number of regions on the solute cavity surface. N_{total} is the number of pair interactions between the N_{REG} regions. N_{near} is the number of pair interactions classified as near-ranged, N_{mid} is the number classified as mid-ranged and N_{long} is the number classified as long-ranged. T_{total} is the total CPU time to finish the job and $T_{phase\ 1}$ and $T_{phase\ 2}$ are the step times to finish the solvent potential evaluations for all the tesserae in phases 1and 2 (for one SCF cycle)

Molecules		$R_1=10,\ R_2=20$	$R_1=15,\ R_2=30$	$R_1=\infty,\ R_2=\infty$
OMTKY3-Lys55-a, $N_{REG}=417$, $N_{total}=173{,}889$	E_{SCF}	−691.6108164428	−691.6088439233	−691.6088564955
	T_{total} (min)	128.6	148.5	175.9
	$T_{phase\ 1}$ (s)	0.87	1.62	2.61
	$T_{phase\ 2}$ (s)	1.19	2.23	3.77
	N_{near}	49,013	105,827	173,889
	N_{mid}	101,174	67,642	0
	N_{long}	23,702	420	0
OMTKY3-Lys55-b, $N_{REG}=417$, $N_{total}=173{,}889$	E_{SCF}	−691.1418573043	−691.1431062559	−691.1432704623
	T_{total} (min)	134.3	149.9	172.7
	$T_{phase\ 1}$ (s)	0.87	1.60	2.62
	$T_{phase\ 2}$ (s)	1.17	2.27	3.77
	N_{near}	49,015	105,871	173,889
	N_{mid}	101,216	67,598	0
	N_{long}	23,658	420	0

Table 6. The final SCF energy (E_{SCF} in hartrees) and CPU time of the iterative isotropic IEF–PCM computation for the molecules OMTKY3-Lys55-a and OMTKY3-Lys55-b by using various d_{ASC} to define the distant regions in the multipole expansion representation of the solvent potential. N_{REG} is the number of regions on the solute cavity surface. N_{apx} is the number of distant regions seen as multipole expansion points by the QM region. T_{total} is the total CPU time to finish the job and T_{step} is the step time to finish the computation of the operator for the basis set (for one SCF cycle)

Molecules	N_{REG}		$d_{ASC}=10$ Å	$d_{ASC}=15$ Å	$d_{ASC}\geq 20$ Å
OMTKY3-Lys55-a	417	E_{SCF}	−691.6086616299	−691.6096829997	−691.6088564955
		T_{total} (min)	170.9	173.9	175.9
		T_{step} (s)	74.11	91.59	103.87
		N_{apx}	157	57	0
OMTKY3-Lys55-b	417	E_{SCF}	−691.1430719955	−691.1441049901	−691.1432704623
		T_{total} (min)	168.0	169.5	172.7
		T_{step} (s)	74.26	91.71	103.96
		N_{apx}	157	57	0

Table 7. The amounts of step time and total time (seconds) by parallel computing with 1–4 nodes, for the molecule OMTKY3-Lys55-a. The most time-consuming SCF cycle is the fifth one, in which the convergence criterion just shifts from 5×10^{-4} to 10^{-8} au. All the multipole expansion methods are turned on

	$n=1$	$n=2$	$n=3$	$n=4$
Tessellation by GEPOL-GB	329.46	166.39	113.48	86.21
Electronic part of the solute potential	130.73	65.48	43.48	33.43
DIIS phase 1 (566 iterations)	2,882.67	1,597.62	1195.70	984.35
DIIS phase 2 (459 iterations)	3,153.69	1,666.60	1193.59	937.11
Constructing ASC operator $\mathbf{T}_{\mu\nu}$	314.69	157.22	103.10	78.67
Total CPU time for the fifth SCF cycle	6,036.30	3,664.80	2,662.60	2,133.70
Total CPU time to finish the job	19,208.00	10,046.10	7,034.30	5,503.00
Total wall clock time to finish the job	19,212.00	10,243.00	7,191.00	5,901.00
Final SCF energy (hartrees)		−691.60884301 (11 iterations)		

with

$$\mathbf{q}=\mathbf{q}_1+\mathbf{q}_2. \tag{52}$$

Equation (50) can be simplified further,

$$\mathbf{A^{-1}SA^{-1}q}_1=-\mathbf{V}\ , \tag{53}$$

and substituted into Eq. (51),

$$\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{A}}{2}-\mathbf{D}\right)\mathbf{A^{-1}SA^{-1}q}_2=-\frac{\mathbf{A}}{\varepsilon-1}\mathbf{A^{-1}SA^{-1}q}_1\ . \tag{54}$$

Equation (54) can be simplified using the equality $\mathbf{DA^{-1}S}=\mathbf{SA^{-1}D^{T}}$ (where T indicates the transpose),

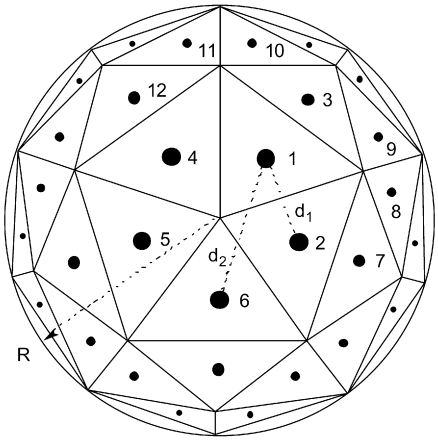

Fig. 6. The pentakisdodecahedron used by GEPOL-GB to generate tesserae on a spherical cavity. Half (30 in number) of the triangular faces are shown. The centers of the tesserae are the projections of the centers of the faces on the sphere, while the areas of the tesserae are the areas of the projection of the faces on the sphere

$$\left(\frac{\varepsilon+1}{\varepsilon-1}\frac{\mathbf{I}}{2}-\mathbf{D}^{\mathrm{T}}\mathbf{A}^{-1}\right)\mathbf{q}_2=-\frac{\mathbf{I}}{\varepsilon-1}\mathbf{q}_1 \ . \tag{55}$$

Equations (53) and (55) form the basis of the iterative method of the isotropic IEF–PCM, when rewritten as Eqs. (18) and (19) in Sect. 2.3.

Appendix 2

The C-PCM coefficient matrix, $\mathbf{S}'$ (Eq. 18) is not diagonal-dominant and the PJI is divergent. This can be shown by the following analysis.

Consider the simplest cavity generated by GEPOL-GB with only one sphere (Fig. 6). The radius of the sphere is R. In GEPOL-GB, the cavity is generated by projecting an inscribed pentakisdodecahedron onto the sphere. The projections of the faces of the pentakisdodecahedron are the tesserae, of which there are 60. The projections of the centers of the triangles are the centers of the tesserae, i.e. the representative points of the tesserae. Note that these triangles are not equilateral. The area of each tessera is exactly 1/60 of the area of the sphere:

$$a_i=\frac{4\pi R^2}{60}=0.2094R^2 \ . \tag{56}$$

The side length of such a triangle, l, is approximately

$$l=\sqrt{\frac{4}{\sqrt{3}}\frac{4\pi R^2}{60}}=0.7R \ . \tag{57}$$

Now consider tessera 1 (Fig. 6). The distance between tessera 1 and tesserae 2, 3 and 4 is approximately

$$d_1=\frac{\sqrt{3}}{3}l=0.4R \ . \tag{58}$$

The distance between tessera 1 and tesserae 5, 6, 7, 8, 9, 10, 11and 12 is approximately

$$d_2=\sqrt{3}d_1=0.7R \ . \tag{59}$$

According to Eq. (22), the diagonal elements of matrix $\mathbf{S}'$ are

$$C_{ii}\left(=S'_{ii}\right)=1.07\sqrt{\frac{4\pi}{a_i}}=1.07\sqrt{\frac{4\pi}{4\pi R^2/60}}=8.29R^{-1} \ . \tag{60}$$

Three of the off-diagonal elements will be equal to

$$\frac{1}{d_1}=2.5R^{-1} \ , \tag{61}$$

and the sum is $7.5R^{-1}$. Eight of the off-diagonal elements will be equal to

$$\frac{1}{d_2}=1.4R^{-1} \ , \tag{62}$$

and the sum is $11.2R^{-1}$. Thus, the sum of the 11 off-diagonal elements corresponding to the closest 11 neighboring tesserae ($7.5R^{-1}+11.2R^{-1}=18.7R^{-1}$) is already greater than the diagonal element ($8.3R^{-1}$); hence, $\mathbf{S}'$ is not diagonal-dominant. It can be generally proved that the GEPOL-GB procedure cannot generate diagonal-dominant coefficient matrices, so the convergence of the PJI method is not guaranteed for C-PCM and isotropic IEF–PCM "phase 1". Empirically they are found to be divergent.

The situation for the isotropic IEF–PCM "phase 2" (and D-PCM) is more complicated than that in the C-PCM and "phase 1". It can be proved that for a well-tessellated cavity, like the single sphere discussed previously (Fig. 6), the diagonal elements in Eq. (22) are approximately

$$\begin{aligned} C_{ii}\left(D'_{ii}\right) &= \frac{1}{2}\cdot\frac{\varepsilon+1}{\varepsilon-1}+\frac{1.07\sqrt{4\pi a_i}}{8\pi R_i} \\ &> 0.5+0.069>0.56. \end{aligned} \tag{63}$$

The absolute values of the off-diagonal elements are

$$\begin{aligned} \left|C_{ij}\right|=\left|D'_{ij}\right| &= \left|\frac{a_i(\vec{r}_j-\vec{r}_i)\cdot\hat{n}_i}{4\pi\left|\vec{r}_i-\vec{r}_j\right|^3}\right|=\frac{a_i\left|\vec{r}_i-\vec{r}_j\right|\cdot\frac{\left|\vec{r}_i-\vec{r}_j\right|}{2R}}{4\pi\left|\vec{r}_i-\vec{r}_j\right|^3} \\ &= \frac{a_i}{8\pi R\left|\vec{r}_i-\vec{r}_j\right|}. \end{aligned} \tag{64}$$

The sum of the absolute values of the off-diagonal elements is

$$\begin{aligned} \sum_j^{59}\left|C_{ij}\right|=\sum_j^{59}\left|D'_{ij}\right| &= \sum_j^{59}\frac{4\pi R^2/60}{8\pi R\left|\vec{r}_i-\vec{r}_j\right|}=\frac{4\pi R^2/60}{8\pi R}\sum_j^{59}\frac{1}{\left|\vec{r}_i-\vec{r}_j\right|} \\ &= \frac{R}{120}\sum_j^{59}\frac{1}{\left|\vec{r}_i-\vec{r}_j\right|}<\frac{R}{120}\cdot 59\cdot\frac{1}{R}<0.50. \end{aligned} \tag{65}$$

So **C** and **D** are diagonal-dominant and the PJI convergence is guaranteed. However, for a cavity formed with overlapping spheres, the GEPOL-GB procedure generates many irregular-polygon-shaped tesserae and the distances among neighboring tesserae might be too small; thus, they may not be diagonal-dominant and the convergence is not guaranteed. Empirically both divergent and convergent cases are found for phase 2.

References

1. Warshel A, Levitt M (1976) J Mol Biol 103: 227
2. Bash PA, Field MJ, Davenport RC, Petsko GA, Ringe D, Karplus M (1991) Biochemistry 30: 5826–5832
3. Field MJ (2002) J Comput Chem 23: 48–58
4. Alhambra C, Corchado JC, Sanchez ML, Gao JL, Truhlar DG (2000) J Am. Chem Soc 122: 8197–8203
5. Villa J, Warshel A (2001) J Phys Chem B 105: 7887–7907
6. Cui Q, Elstner M, Karplus M (2002) J Phys Chem B 106: 2721–2740
7. Jorgensen WL, Chandrasekhar J, Madura JD, Impey RW, Klein ML (1983) J Chem Phys 79: 926
8. Dewar MJS, Zoebisch EG, Healy EF, Stewart JJP (1985) J Am Chem Soc 107: 3902–3909
9. Warshel A (1991) Computer modeling of chemical reactions in enzymes and solutions. Wiley, New York
10. Simonson T, Archontis G, Karplus M (1997) J Phys Chem B 101: 8349–8362
11. Murphy RB, Philipp DM, Friesner RA (2000) J Comput Chem 21: 1442–1457
12. Bandyopadhyay P, Gordon MS, Mennucci B, Tomasi J (2002) J Chem Phys 116: 5023–5032
13. Day PN, Jensen JH, Gordon MS, Webb SP, Stevens WJ, Krauss M, Garmer D, Basch H, Cohen D (1996) J Chem Phys 105: 1968–1986
14. Gordon MS, Freitag MA, Bandyopadhyay P, Jensen JH, Kairys V, Stevens WJ (2001) J Phys Chem A 105: 293–307
15. Cances E, Mennucci B, Tomasi J (1997) J Chem Phys 107: 3032–3041
16. Cui Q (2002) J Chem Phys 117: 4720–4728
17. Li H, Hains AW, Everts JE, Robertson AD, Jensen JH (2002) J Phys Chem B 106: 3486–3494
18. Camm R, Cossi M, Mennucci B, Pomelli CS, Tomasi J (1996) Int J Quantum Chem 60: 1165–1178
19. Rega N, Cossi M, Barone V (1998) Chem Phys Lett 293: 221–229
20. Rega N, Cossi M, Barone V, Pomelli CS, Tomasi J (1999) Int J Quantum Chem 73: 219–227
21. Rega N, Cossi M, Barone V (1999) J Comput Chem 20: 1186–1198
22. Pomelli CS, Tomasi J, Barone V (2001) Theor Chem Acc 105: 446–451
23. Pomelli CS, Tomasi J (2001) J Mol Struct (THEOCHEM) 537: 97–105
24. Cossi M, Mennucci B, Cammi R (1996) J Comput Chem 17: 57–73
25. Mennucci B, Tomasi J (1997) J Chem Phys 106: 5151–5158
26. Chipman DM (2002) J Chem Phys 116: 10129–10138
27. Schmidt MW, Baldridge KK, Boatz JA, Elbert ST, Gordon MS, Jensen JH, Koseki S, Matsunaga N, Nguyen KA et al (1993) J Comput Chem 14: 1347–1363
28. Varga RS (2000) Matrix iterative analysis, 2nd edn. Springer, Berlin Heidelberg New York
29. Pulay P (1980) Chem Phys Lett 73: 393–298
30. Stone AJ (1981) Chem Phys Lett 83: 233–239
31. Fletcher GD, Schmidt MW, Bode BM, Gordon MS (2000) Comput Phys Commun 128: 190–200
32. Hehre WJ, Ditchfield R, Pople JA (1972) J Chem Phys 56: 2257
33. Barone V, Cossi M, Tomasi J (1997) J Chem Phys 107: 3210–3221
34. Kairys V, Jensen JH (2000) J Phys Chem A 104: 6656–6665
35. Minikis RM, Kairys V, Jensen JH (2001) J Phys Chem A 105: 3829–3837
36. Pomelli CS, Tomasi J (1998) J Comput Chem 19: 1758–1776