MM-ISMSA: An Ultrafast and Accurate Scoring Function for Protein–Protein Docking

Javier Klett,[†] Alfonso Núñez-Salgado,[†] Helena G. Dos Santos,[†] Álvaro Cortés-Cabrera,[†,‡] Almudena Perona,[†,§] Rubén Gil-Redondo,[†,§] David Abia,[†] Federico Gago,[‡] and Antonio Morreale*,[†]

[†]Unidad de Bioinformática, Centro de Biología Molecular Severo Ochoa (CSIC-UAM), Campus de Cantoblanco UAM, E-28049 Madrid, Spain

[‡]Departamento de Farmacología, Universidad de Alcalá, Alcalá de Henares, E-28871 Madrid, Spain

[§]SmartLigs Bioinformática S.L., Fundación Parque Científico de Madrid, c/Faraday, 7. Campus de Cantoblanco UAM, E-28049 Madrid, Spain

ABSTRACT: An ultrafast and accurate scoring function for protein–protein docking is presented. It includes (1) a molecular mechanics (MM) part based on a 12–6 Lennard-Jones potential; (2) an electrostatic component based on an implicit solvent model (ISM) with individual desolvation penalties for each partner in the protein–protein complex plus a hydrogen bonding term; and (3) a surface area (SA) contribution to account for the loss of water contacts upon protein–protein complex formation. The accuracy and performance of the scoring function, termed MM-ISMSA, have been assessed by (1) comparing the total binding energies, the electrostatic term, and its components (charge–charge and individual desolvation energies), as well as the per residue contributions, to results obtained with well-established methods such as APBSA or MM-PB(GB)SA for a set of 1242 decoy protein–protein complexes and (2) testing its ability to recognize the docking solution closest to the experimental structure as that providing the most favorable total binding energy. For this purpose, a test set consisting of 15 protein–protein complexes with known 3D structure mixed with 10 decoys for each complex was used. The correlation between the values afforded by MM-ISMSA and those from the other methods is quite remarkable ($r^2 \sim 0.9$), and only 0.2–5.0 s (depending on the number of residues) are spent on a single calculation including an *all vs all* pairwise energy decomposition. On the other hand, MM-ISMSA correctly identifies the best docking solution as that closest to the experimental structure in 80% of the cases. Finally, MM-ISMSA can process molecular dynamics trajectories and reports the results as averaged values with their standard deviations. MM-ISMSA has been implemented as a plugin to the widely used molecular graphics program PyMOL, although it can also be executed in command-line mode. MM-ISMSA is distributed free of charge to nonprofit organizations.

1. INTRODUCTION

Molecular association (binding) plays a key role in cellular function and communication, and many illnesses can be directly linked to an improper balance of interactions among distinct molecular species. Of special importance are those established between different proteins or between small molecules and proteins. In the latter case, we usually talk about ligands and receptors, but in the following, we will use this terminology to refer to the two binding partners. Understanding how binding takes place and how this event can be theoretically modeled is of paramount importance in today's drug discovery campaigns,[1] especially in fields like protein engineering,[2] ligand and fragment docking,[3] virtual screening (VS),[4] and computational mutagenesis,[5] among others.

A large number of methodological advances have been introduced since the first theoretical simulation of a biologically relevant system[6] that, in favorable cases, allow one to reproduce experimental binding affinities with an error comparable to that of the experimental measurements.[7] On top of that, modern computers, sometimes including tailor-made architectures (e.g., the Anton machine[8]), and supercomputers[9] allow researchers to undertake calculations that were unimaginable just a few years ago to address difficult problems such as simulating protein folding,[10] long MD of very large systems,[11] or unbiased drug binding patterns.[12]

Nevertheless, although theoretical methods and computer technologies are continuously improving, there are still some bottlenecks. Representative examples are intrinsically massive calculations, as undertaken in VS, where the number of molecules to be simulated can reach the order of several millions, and the analysis and interpretation of long MD trajectories especially when solvent effects must be properly accounted for, per-residue analyses must be performed, or one wishes to estimate elusive entropic effects.

Here, we are interested in the electrostatic contribution to the free energy of binding, and more specifically in the effect played by the solvent. There are two opposite, although complementary, ways to account for this effect:[13] either representing the solvent explicitly by means of a set of discrete water molecules surrounding the solute or *via* a mathematical function able to describe the behavior of the bulk solvent. There are also examples in which both methods have been combined in a sort of hybrid solvent model.[14] However, both have advantages and caveats, and selecting the most appropriate description for a particular study greatly depends on the computational power available, as explicit models require the calculation of a large amount of interactions due to the presence of numerous water molecules.

Received: June 15, 2012
Published: August 2, 2012

For docking-related tools, implicit models are usually preferred, as they are faster than their explicit counterpart while performing quite similarly. Implicit models start by solving the classical Poisson equation (PE).[15] But, in many cases it is too computationally expensive, and other alternatives such as the generalized Born (GB) model are employed instead.[16] In addition, solvation models based on group contributions (effective energy function, EFF1) have also been proposed and successfully employed in proteins[17] and protein−ligand force fields.[18]

In this paper, we extend our previously developed GB-like solvent model, called ISM (Implicit Solvent Model),[19] to the protein−protein docking problem and compare its performance to well-established methods like APBS (PE solver) and MM-PB(GB)SA (as implemented in AmberTools[20]) using two different test sets. For the procedures to be strictly comparable, we first incorporated MM (Molecular Mechanics) and SA (Surface Area) terms to the APBS (MM-APBSSA) and ISM (MM-ISMSA) methods. Comparisons between the three different techniques were performed in order to assess their relative speeds and to validate the different values provided by our method for total binding free energies, individual components, and per residue energy decompositions. Finally, to extend the usability of MM-ISMSA within the scientific community, we have developed a graphical user interface (GUI) that allows its use *via* the popular PyMOL program.[21] This plugin, which can be used to analyze single structures or complete MD trajectories, can be downloaded following free registration from the CBM Bioinformatics Unit's web page (http://ub.cbm.uam.es).

2. THEORETICAL BACKGROUND

2.1. Statistical Thermodynamics of Binding: Interaction Terms. According to classical thermodynamics,[22] molecular association can be described as an equilibrium ([R] + [L] ⇌ [RL], where R and L represent receptor and ligand, respectively, and RL, the complex formed between them) governed by the association and dissociation rate constants. The ratio between them is the equilibrium binding constant, K, which is related to the free energy change taking place in the process ($\Delta G_{\text{binding}}$) by the well-known equation:

$$\Delta G_{\text{binding}} = \Delta G_{\text{RL}} - \Delta G_{\text{R}} - \Delta G_{\text{L}} = -R_{\text{gas}}T \ln K \qquad (1)$$

where R_{gas} is the universal gas constant, T is the temperature in Kelvin, and ΔG_{X} represents the free energy corresponding to the complex (X = RL), receptor (X = R), and ligand (X = L).

The calculation of $\Delta G_{\text{binding}}$ would entail extremely lengthy simulations in which the ligand diffuses into the receptor's binding site, but this is hardly ever done. As useful alternatives, the ligand can be "grown" slowly both in the bulk solvent and inside the binding site to calculate free energy differences,[7] or $\Delta G_{\text{binding}}$ can be estimated as the difference between the free energies of bound and unbound states, as in linear interaction energy (LIE)[23] and MM-PB(GB)SA[24] "end-point" methods.

It has been customary to describe the binding process through the thermodynamic cycle in Figure 1, where ΔG_{int} refers to the binding process in the gas phase, and $\Delta G_{\text{solv}}^{\text{RL}}$, $\Delta G_{\text{solv}}^{\text{R}}$, and $\Delta G_{\text{solv}}^{\text{L}}$ are the free energies of solvation for the complex, receptor, and ligand, respectively. Next, because $\Delta G_{\text{binding}}$ is a state variable, the cycle can be solved to yield:

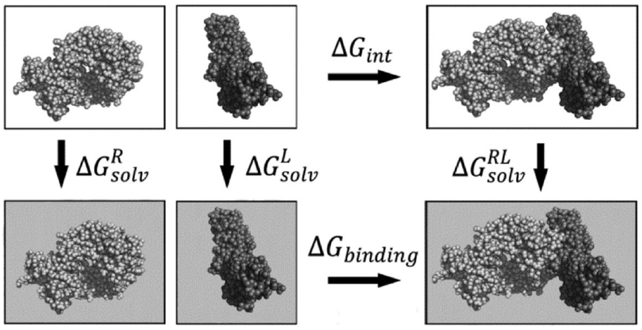


Figure 1. Graphical representation of the commonly used thermodynamic cycle to estimate $\Delta G_{\text{binding}}$. The shadowed boxes represent the systems (receptor, ligand, and complex) immersed in the solvent.

$$\Delta G_{\text{binding}} = \Delta G_{\text{int}} + \Delta G_{\text{solv}}^{\text{RL}} - \Delta G_{\text{solv}}^{\text{R}} - \Delta G_{\text{solv}}^{\text{L}}$$
$$= \Delta G_{\text{int}} + \Delta G_{\text{solv}} \qquad (2)$$

Usually, from Figure 1 and eq 2 the solvation contribution (ΔG_{solv}) is obtained as a single term (e.g., in GB) and, as a consequence, individual solvation energies for the ligand and receptor are not available. As an alternative, a different description of the binding process[25] can be considered (Figure 2) that consists of first desolvating the apposing surfaces of both ligand and receptor and then letting the charges of the two molecules interact.

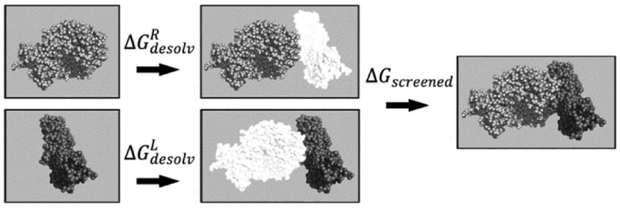


Figure 2. Alternative description of the binding process to estimate $\Delta G_{\text{binding}}$. This type of equilibrium is commonly used in combination with PE solvers. White molecules are uncharged, and they are used to replace the high-dielectric solvent with a low-dielectric medium in the desolvation calculations.

Thermodynamically, $\Delta G_{\text{binding}}$ has an enthalpic ($\Delta H_{\text{binding}}$) and an entropic ($\Delta S_{\text{binding}}$) component:

$$\Delta G_{\text{binding}} = \Delta H_{\text{binding}} - T\Delta S_{\text{binding}} \qquad (3)$$

$\Delta H_{\text{binding}}$ contains van der Waals ($\Delta G_{\text{binding}}^{\text{vdW}}$), hydrogen bonding ($\Delta G_{\text{binding}}^{\text{hb}}$), and solvent-related contributions that can be further subdivided into polar ($\Delta G_{\text{binding}}^{\text{p}}$) and apolar ($\Delta G_{\text{binding}}^{\text{np}}$) components. The former contains the Coulombic interactions ($\Delta G_{\text{binding}}^{\text{elec,coul}}$) together with ligand ($\Delta G_{\text{binding}}^{\text{elec,desolv}_{\text{R}}}$) and receptor ($\Delta G_{\text{binding}}^{\text{elec,desolv}_{\text{L}}}$) desolvation terms, whereas the latter includes the cavitation term (the work required to create a cavity within the solvent to introduce the solute) and the van der Waals solute−solvent interactions. A linear relationship is usually assumed between the composite of the latter two components and the change in solvent-accessible surface area (SASA) of the ligand and receptor upon binding. On the other hand, the entropic contribution arises from the loss of some protein degrees of freedom that become frozen when the complex is formed and also from solvent reorganization, as some water molecules present within the binding site will be released to the bulk solvent as a consequence of the binding event. This entropic contribution is rarely taken into account when $\Delta G_{\text{binding}}$ is computed due to its complexity, high computational demand, and slow convergence.[26] The electrostatic component is the most challenging term, and this will be the focus of the present work.

2.2. The PE Model.

The classical way to deal with the electrostatic contribution to the binding energy (ΔG_{elec}) is by solving the PE, which relates the electrostatic potential $\phi(r)$ and the charge distribution $\rho(r)$:

$$\nabla[\varepsilon(r)\cdot\nabla\phi(r)] = -4\pi\rho(r) \tag{4}$$

where $\varepsilon(r)$ is a distance-dependent dielectric function. For a given $\rho(r)$, $\phi(r)$ can be calculated *via* the PE so that

$$\Delta G_{\text{elec}} = \frac{1}{2}\int \rho(r)\,\phi(r)\,\mathrm{d}v \tag{5}$$

Because the analytical solution of PE is possible only for very simple geometries, for biological molecules we have to rely on numerical methods such as finite differences,[27] finite elements,[28] or boundary elements.[29] Nonetheless, solving the PE is still a computationally demanding task in many molecular modeling areas, despite constant improvements over the years.[30]

Equation 5 can be used in different ways to obtain an estimation of the binding free energy, either by computing the desolvation terms of the thermodynamic cycle depicted in Figure 1 (as implemented in the MM-PBSA method) or by applying the cycle shown in Figure 2 as described in section 3.2.1, where the Coulombic contribution is obtained by computing the product of ligand charges times the electrostatic potential generated by the protein on the ligand charge centers. On the other hand, receptor and ligand electrostatic desolvation energies are calculated in two successive steps (Figure 2): a first one, where a calculation is performed for the receptor and ligand alone, and a second one, for the ligand in the complex, with uncharged receptor, and for the receptor in the complex, with uncharged ligand.

2.3. The GB Model.

The GB model is based on the Born approximation and can be easily derived from the PE, assuming a spherical solute that has the whole charge located at its center, according to

$$\Delta G_{\text{solv}}^{\text{elec}}(\text{Born}) = -166\left(1 - \frac{1}{\varepsilon}\right)\frac{q^2}{r} \tag{6}$$

where ε is the dielectric constant of the solvent, and q and r are the charge and the radius of the sphere, respectively. Taking into account that molecules can be represented as a set of interacting spheres, eq 6 can be generalized to the expression commonly used in the GB models:

$$\begin{aligned}\Delta G_{\text{solv}}^{\text{elec}}(\text{GB}) &= \Delta G_{\text{vac}} + \Delta G_{\text{pol}}\\ &= 332\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\frac{q_iq_j}{r_{ij}} - 166\left(1 - \frac{1}{\varepsilon}\right)\\ &\quad \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{q_iq_j}{f_{\text{GB}}}\end{aligned} \tag{7}$$

where N is the total number of atoms, r_{ij} is the distance between atoms i and j, q_i and q_j are the atomic charges of atoms i and j, and f_{GB} is the GB function defined as

$$f_{\text{GB}}(r_{ij}, i, j) = [r_{ij}^{\;2} + \alpha_i\alpha_j\,\mathrm{e}^{(-r_{ij}^{\;2}/4\alpha_i\alpha_j)}]^{1/2} \tag{8}$$

α is the so-called effective Born radius, which is the distance from an atom to the molecular surface. Note that f_{GB} is different depending on the system, that is, the receptor, the ligand, or the complex. In eq 7, the first term (ΔG_{vac}) represents the electrostatic interaction in vacuo, while the second (ΔG_{pol}) accounts for the *polarization* effects due to the solvent. In fact, it is this second term that is calculated as the electrostatic component of the free energy of solvation in GB-based methods, and as such it has been implemented in many different programs and, in particular, in the AmberTools package.

Applying eq 7 to the thermodynamic equilibrium depicted in Figure 1, it is possible to separate the electrostatic interaction between charges (in a vacuum and in the solvent) from the pure desolvation terms for the receptor and ligand:

$$\begin{aligned}\Delta\Delta G_{\text{solv}}^{\text{elec}}(\text{GB}) &= 332\sum_{i=1}^{N_{\text{R}}}\sum_{j=1}^{N_{\text{L}}}\frac{q_i^{\text{R}}q_j^{\text{L}}}{r_{ij}} - 332\left(1 - \frac{1}{\varepsilon}\right)\\ &\quad \sum_{i=1}^{N_{\text{R}}}\sum_{j=1}^{N_{\text{L}}}\frac{q_i^{\text{R}}q_j^{\text{L}}}{f_{\text{GB}}^{\text{RL}}} - 166\left(1 - \frac{1}{\varepsilon}\right)\\ &\quad \sum_{i=1}^{N_{\text{R}}}\sum_{j=1}^{N_{\text{L}}}\left[\frac{q_i^{\text{R}}q_j^{\text{R}}}{f_{\text{GB}}^{\text{RL}}} - \frac{q_i^{\text{R}}q_j^{\text{R}}}{f_{\text{GB}}^{\text{R}}}\right]\\ &\quad - 166\left(1 - \frac{1}{\varepsilon}\right)\sum_{i=1}^{N_{\text{R}}}\sum_{j=1}^{N_{\text{L}}}\left[\frac{q_i^{\text{L}}q_j^{\text{L}}}{f_{\text{GB}}^{\text{RL}}} - \frac{q_i^{\text{L}}q_j^{\text{L}}}{f_{\text{GB}}^{\text{L}}}\right]\end{aligned} \tag{9}$$

Unfortunately, as commented upon above, the GB method as implemented in AmberTools does not allow for such decomposition, and only the total polarization energy is obtained (ΔG_{pol} in eq 7). Nevertheless, setting the atomic charges in the receptor/ligand to zero and subtracting the resulting GB term from the GB term of a standard calculation would afford the ligand/receptor desolvation terms independently. Accordingly, zeroing the ligand's charges:

$$\begin{aligned}\Delta\Delta G_{\text{desolv}}^{\text{R}} &= \Delta G_{\text{elec}}^{\text{MMGBSA}}(\text{GB}) - \Delta G_{\text{elec}}^{\text{MMGBSA}}(\text{GB}, q_i^{\text{L}} = 0)\\ &= -166\left(1 - \frac{1}{\varepsilon}\right)\sum_{i=1}^{N_{\text{R}}}\sum_{j=1}^{N_{\text{R}}}(q_i^{\text{R}})^2\left[\frac{1}{f_{\text{GB}}^{\text{RL}}} - \frac{1}{f_{\text{GB}}^{\text{R}}}\right]\end{aligned} \tag{10}$$

The same can be done if the receptor's charges are zeroed:

$$\begin{aligned}\Delta\Delta G_{\text{desolv}}^{\text{L}} &= \Delta G_{\text{elec}}^{\text{MMGBSA}}(\text{GB}) - \Delta G_{\text{elec}}^{\text{MMGBSA}}(\text{GB}, q_i^{\text{R}} = 0)\\ &= -166\left(1 - \frac{1}{\varepsilon}\right)\sum_{i=1}^{N_{\text{L}}}\sum_{j=1}^{N_{\text{L}}}(q_i^{\text{L}})^2\left[\frac{1}{f_{\text{GB}}^{\text{RL}}} - \frac{1}{f_{\text{GB}}^{\text{L}}}\right]\end{aligned} \tag{11}$$

Choosing this alternative would require (a) manipulating the topology (*top*) files used by AmberTools to set the charges of the ligand (or the receptor) to zero and (b) an additional GB calculation (ΔG_{pol} in eq 7) to obtain the Coulombic interaction screened by the solvent ($\Delta G_{\text{screened}}$, the first right-hand term in eq 9). That is, taking into account

$$\Delta G_{\text{pol}} = \Delta G_{\text{screened}} + \Delta G_{\text{desolv}}^{\text{R}} + \Delta G_{\text{desolv}}^{\text{L}}$$

and the fact that we can calculate both desolvation energies as shown above:

$$\Delta G_{\text{screened}} = \Delta G_{\text{solv}}^{\text{elec}}(\text{GB}) - \Delta G_{\text{desolv}}^{\text{R}} - \Delta G_{\text{desolv}}^{\text{L}}$$

three calculations are needed to obtain the total free energy of solvation and its components.

2.4. The ISM Model.

The model starts from the Lorentz−Debye−Sack theory of polar liquids,[31] which establishes that the screening effect due to the solvent shows a sigmoidal distance-dependent dielectric function of the form:

$$D(r) = \frac{\varepsilon + 1}{1 + k\,\mathrm{e}^{-\lambda(\varepsilon+1)r}} - 1 \tag{12}$$

where ε is the solvent dielectric constant, $k = (\varepsilon - 1)/2$, λ is a parameter controlling the rate of change of $D(r)$, and r is the distance. ISM considers that the main contribution to the electrostatic desolvation of an atom originates from the displacement of the first shell of water molecules that surrounds that atom. Taking these two facts into account, ISM's starting equation, as proposed by Hassan et al.,[32] is the following:

$$\begin{aligned}\Delta G_{\text{solv}}^{\text{elec}}(\text{ISM}) = {} & \sum_{i<j}^{N} \frac{q_i q_j}{r_{ij}}\left[\frac{1}{D_{\text{s}}(r_{ij})} - \frac{1}{D_{\text{v}}(r_{ij})}\right] \\ & + \frac{1}{2}\sum_{i}^{N} q_i^2 \left\{\frac{1}{R_{i,\text{Bs}}}\left[\frac{1}{D_{\text{s}}(R_{i,\text{Bs}})} - 1\right] \right. \\ & \left. - \frac{1}{R_{i,\text{Bv}}}\left[\frac{1}{D_{\text{s}}(R_{i,\text{Bv}})} - 1\right]\right\}\end{aligned} \tag{13}$$

where $R_{i,\text{Bs}}$ and $R_{i,\text{Bv}}$ are the effective Born radii for the processes of transferring an atom from a vacuum into a protein interior, surrounded by either solvent or a vacuum, respectively. The model has proven to be useful to study the structure and dynamics of proteins[33] and has been implemented as a solvation method within the molecular dynamics code of the program CHARMm.[34]

This model has been extended by us to deal with ligand−receptor[19] and protein−protein interactions (this work). Considering the thermodynamic cycle of binding depicted in Figure 1 and the expression for $\Delta G_{\text{solv}}^{\text{elec}}(\text{ISM})$:

$$\begin{aligned}\Delta G_{\text{elec}}^{\text{ISM}} = {} & \frac{1}{2}\sum_{i=1}^{N_{\text{R}}}\sum_{j=1}^{N_{\text{L}}} \frac{q_i^{\text{R}} q_j^{\text{L}}}{r_{ij}}\left[\frac{1}{D_{\text{s}}(r_{ij})}\right] \\ & + \frac{1}{2}\sum_{i}^{N_{\text{RL}}} (q_i^{\text{RL}})^2 \left[\left(\frac{1}{D_{\text{s}}(R_{i,B_{\text{s}}}^{\text{c}})R_{i,B_{\text{s}}}^{\text{c}}} - \frac{1}{D_{\text{s}}(R_{i,B_{\text{s}}}^{\text{u}})R_{i,B_{\text{s}}}^{\text{u}}}\right) \right. \\ & \left. + \left(\frac{1}{R_{i,B_{\text{s}}}^{\text{u}}} - \frac{1}{R_{i,B_{\text{s}}}^{\text{c}}}\right)\right]\end{aligned} \tag{14}$$

where the superscripts c and u stand for complexed and uncomplexed forms of both the ligand and receptor. There is an evident resemblance between ISM (eq 14) and GB (eq 9) formulations: the first term describes the interaction established between the receptor and ligand screened by the dielectric function, and the second is the desolvation penalty, accounted for by the difference between complexed and uncomplexed partners in terms of their Born radii and dielectric function in the solvent.

As an additional advantage, ISM directly yields individual desolvation terms without the need to perform any extra calculations, as commented upon before for GB. In fact, to isolate the desolvation term for the receptor, only the summation of the atoms concerning the receptor must be considered in the GB-like term (second term on the right-hand side of eq 14), which is the same as setting to zero the ligand's charges (eq 15) or, for the ligand desolvation term, setting to zero the receptor's charges (eq 16):

$$\begin{aligned}\Delta G_{\text{desolv}}^{\text{R}} = {} & \frac{1}{2}\sum_{i}^{N_{\text{R}}} (q_i^{\text{R}})^2 \left[\frac{1}{R_{i,B_{\text{s}}}^{\text{c}}}\left(\frac{1}{D_{\text{s}}(R_{i,B_{\text{s}}}^{\text{c}})} - 1\right) \right. \\ & \left. - \frac{1}{R_{i,B_{\text{s}}}^{\text{u}}}\left(\frac{1}{D_{\text{s}}(R_{i,B_{\text{s}}}^{\text{u}})} - 1\right)\right]\end{aligned} \tag{15}$$

$$\begin{aligned}\Delta G_{\text{desolv}}^{\text{L}} = {} & \frac{1}{2}\sum_{i}^{N_{\text{L}}} (q_i^{\text{L}})^2 \left[\frac{1}{R_{i,B_{\text{s}}}^{\text{c}}}\left(\frac{1}{D_{\text{s}}(R_{i,B_{\text{s}}}^{\text{c}})} - 1\right) \right. \\ & \left. - \frac{1}{R_{i,B_{\text{s}}}^{\text{u}}}\left(\frac{1}{D_{\text{s}}(R_{i,B_{\text{s}}}^{\text{u}})} - 1\right)\right]\end{aligned} \tag{16}$$

2.5. Pairwise Decomposition of the Binding Energy.

The value of the free energy of binding as a whole is useful when comparing the binding strength of a set of ligands toward a target of interest. However, it does not provide any information on the relative contributions of individual residues. Knowing which residues are the most important in the interaction would allow the design of specific mutations to increase or even disrupt the association in protein−protein complexes. In protein−ligand docking, this knowledge is essential to suggesting chemical modifications on ligand structures guided by the binding site residues. Accordingly, several approaches have been developed.[35,36]

The MM-PB(GB)SA method, as implemented in AmberTools, has all the terms already pairwise decomposed, as a consequence of the double summations in eq 7. On the other hand, in MM-ISMSA the solvation term entails a single summation (eq 14), so the individual solvation for each residue is obtained but not the corresponding contribution of a given interacting pair. The simple addition of the individual solvation values would overestimate this contribution, as not only the atoms involved in the interaction take part in its calculation but also those from their environments. Therefore, we have devised a weighting scheme (see Methods) by means of which the solvation contribution of a given pair is balanced by the sum of the van der Waals and Coulombic interaction in which any of the two residues in that pair is involved.

3. METHODS

3.1. Nonelectrostatics Calculations.

We refer here to the calculations involving the van der Waals and the solvent accessible-related terms to account for shape complementarity and the loss in surface area produced upon complex formation (the nonpolar part of the desolvation), respectively. van der Waals interactions (ΔG_{vdW}) are calculated through the well-known 12−6 Lennard-Jones potential:

$$\Delta G_{\text{vdW}} = \sum_{ij}\left[\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}}\right]$$

where A_{ij} and B_{ij} are the van der Waals parameters of the atom types to which atoms i and j belong, and r_{ij} is the distance between the ith atom in the protein and the jth atom from the ligand. A_{ij} and B_{ij} parameters were taken directly from the

AMBER ff03 force field.[37,38] The nonpolar part of the desolvation (ΔG_{np}) was modeled as a linear relationship to the change in SASA:

$$\Delta G_{np} = a + b \cdot \Delta \text{SASA}$$

where a is 0.092 kcal·mol^{-1}, b is 0.00542 kcal·mol^{-1} Å^{-2}, and the change in SASA refers to the complex SASA minus the sum of that of the protein and the ligand alone. SASA values were obtained with our own implementation of the LCPO approximation.[39]

Both terms were added to the ISM electrostatic scoring function to configure a more complete tool termed MM-ISMSA, in clear allusion to MM-PB(GB)SA and related approaches.

3.2. Electrostatic Calculations. In all of the cases where PE was employed, we refer to the linearized Poisson–Boltzmann equation; that is, the Boltzmann part of the equation (related to ions in solution) was not taken into account. Atomic radii were automatically assigned with the *tleap* module in AMBER 10 so that they correspond to the "modified Bondi" set.[40]

3.2.1. APBS. APBS uses the adaptive finite element method to solve the Poisson–Boltzmann equation numerically.[30] First, grid size, grid center, and the number of grid points were computed with the *psize.py* module provided in the APBS package, which properly fits the input complexes into their respective grid boxes. Then, the following parameters were chosen: (a) dielectric constants of 4 and 80 for solute and solvent, respectively, (b) a dielectric boundary calculated using a solvent probe radius of 1.4 Å, (c) potentials at the grid points delimiting the box calculated using the multiple Debye–Hückel method, and (d) multigrid PB calculations configured to run in automatic mode. All of these calculations were performed with the APBS program (see section 2.2).

3.2.2. PB/GB. For PB calculations, each complex was immersed in a cubic box with a grid spacing of 0.5 Å. The solute dielectric constant was set to 4, while that of the solvent was set to 80, and the dielectric boundary was calculated using a solvent probe radius of 1.4 Å. The potentials at the grid points delimiting the box were calculated analytically by treating each charge atom as a Debye–Hückel sphere. Similar parameters were employed for GB calculations, namely, the internal and external dielectric constants and the solvent probe radius. These calculations were performed with *mm_pbsa.pl* and *MMPBSA.py* scripts as provided in the AMBER package.

3.2.3. ISM. ISM models the screening effect due to the solvent by means of a sigmoidal distance-dependent dielectric function (eq 12). Solvent-related parameters are the slope of the sigmoidal dielectric function (λ), which has two values, 0.013 for all of the atoms except for those with a formal positive charge and 0.007 for the latter ones; ε, the dielectric constant of the bulk solvent (80); and the solvent probe radius (1.4 Å) to calculate the SASA. For additional parameters, the reader is referred to the original publications.[19,32] All of these calculations were performed with our in-house version of the ISM program (eq 14).

3.3. Hydrogen Bonding Term. The existence of a hydrogen bond was characterized by defining the three atoms involved in the interaction (donor, D; acceptor, A; and the proper hydrogen atom, H) plus the atom bonded to A (X) and three geometrical parameters describing the relative disposition of these atoms: (1) the A···H distance (r), (2) the D···H···A angle (α), and (3) the H···A···X angle (β).

As a training set, we used the Astex Diverse Set (ADS) of protein–ligand complexes. On the other hand, the test sets included the 23 protein–ligand complexes used in the original ISM paper and the 15 protein–protein complexes previously described in section 3.4. Each complex in the training set was visually inspected in PyMOL to determine the number and geometrical parameters of all of the possible hydrogen bonds, whereas for the first test set the program LIGPLOT,[41] as implemented in the PDBsum web server,[42] was used. The first test allowed us to check to what extent the parameters derived for hydrogen bonds in protein–ligand complexes could be extended to protein–protein assemblies. As a default, r was set to 3.5 Å and α to 90°, and no restrictions were imposed on the β angle. Finally, for the second test set, the hydrogen bonds obtained with the HBPLUS program,[43] employing default parameters, were used for comparison.

The distributions of r, α, and β values were analyzed by means of the nonparametrical BOX plot statistical technique. A set of ideal values and upper and lower limits were defined for each variable. These values and the shape of the data distribution were incorporated into *ad hoc* block functions to determine the contribution of each parameter (eq 17) to the total score for each hydrogen bond (HBScore(i), eq 18):

$$\text{score}(x) = \begin{cases} 1 & \text{if} \quad x_{\text{min -ideal}} \le x \le x_{\text{max -ideal}} \\ 1 - \dfrac{x_{\text{min -ideal}} - x}{x_{\text{min -ideal}} - x_{\text{min}}} & \text{if} \quad x_{\text{min}} \le x \le x_{\text{min -ideal}} \\ 1 - \dfrac{x - x_{\text{max -ideal}}}{x_{\text{max}} - x_{\text{max -ideal}}} & \text{if} \quad x_{\text{max -ideal}} \le x \le x_{\text{max}} \\ 0 & \text{if} \quad x_{\text{max}} < x \\ 0 & \text{if} \quad x > x_{\text{min}} \end{cases} \tag{17}$$

where x refers to r, α, and β, and ideal and max values are those obtained from the BOX plot analysis.

$$\text{HBScore}(i) = \prod_{x} \text{score}(x) \tag{18}$$

Hydrogen bonds were further classified by the type of interactions in charged–charged (cc), neutral–charged and charged–neutral (nc), and neutral–neutral (nn) and assigned a numerical value of −3, −2, and −1 kcal/mol, respectively. Finally, HBScore(i) was used to weigh the interaction energy for each hydrogen bond to configure ΔG_{HB}:

$$\Delta G_{\text{HB}} = \sum_{i=1}^{\text{NHB}} \text{HBScore}(i)\, E_{\text{HB}}(i) \tag{19}$$

where i stands for each hydrogen bond; NHB is the total number of hydrogen bonds; and $E_{\text{HB}}(i)$ is equal to −3, −2, or −1 depending on the type of hydrogen bond.

3.4. MM-ISMSA Scoring Function. According to the terms described in the above sections (3.1, 3.2.3, and 3.3), the starting equation for MM-ISMSA (eq 20) reads as follows:

$$\Delta G_{\text{binding}} = \Delta G_{\text{vdW}} + \Delta G_{\text{elec}} + \Delta G_{\text{desolv}}^{\text{R}} + \Delta G_{\text{desolv}}^{\text{L}} + \Delta G_{\text{apo}} + \Delta G_{\text{HB}} \tag{20}$$

3.5. Comparison between Methods. A validation test set consisting of 15 antigen–antibody complexes with available 3D structures (PDB ID codes: 1AHW, 1BGX, 1BJ1, 1BVK, 1DQJ, 1FSK, 1I9R, 1IQD, 1JPS, 1KXQ, 1MLC, 1NCA, 1NSN, 1VFB, and 2E6J) was used to compare the performance between MM-ISMSA and the other methods in terms of interaction energies (total, individual terms, and pairwise decomposed). Up to 100 docking poses for each complex were obtained using program

FRODOCK[44] for a total of 1242 structures. Then, the following protocol was employed for each single complex: (a) The AMBER ff03 force field was used to assign atom types and partial charges to each atom in the complexes. (b) Hydrogen atoms were added using the *tleap* module from the AMBER suite assuming standard protonation states for titratable groups. (c) The structures were subjected to an energy refinement process using the GB implicit solvent model as implemented in *sander* (500 cycles of steepest descent followed by 1000 cycles of conjugate gradient until the root-mean-square value of the potential energy gradient was below 0.1 kcal·mol^{-1}·Å^{-1}) to remove possibly existing steric clashes. (d) APBS, MM-ISMSA, and MM-PB(GB)SA calculations were performed on the refined complexes. Finally, we compared the numerical values for the total free energy of binding and its vdW and electrostatic components (Coulombic and desolvation terms) obtained with the three methods.

3.6. The MM-ISMSA Scoring Function in Protein–Protein Docking. A diverse set of 15 protein–protein complexes with experimentally determined 3D structures was taken from the PDB (PDB ID codes: 2FUE, 2JK6, 2LYN, 2O3B, 2ONE, 2Y43, 3AAB, 3AIK, 3DH9, 3F1R, 3G3G, 3MIO, 3PC6, 3PY2, and 3KF3) and used to test the ability of the MM-ISMSA scoring function to select near-native docking poses from a pool of incorrect solutions (decoys). Each complex was separated into two individual structure files containing the receptor and the ligand. The addition of hydrogen atoms and computation of the protonation state of ionizable groups at pH 6.5 were carried out using the H++ server,[45] which relies on AMBER force-field parameters and finite difference solutions to the Poisson–Boltzmann equation. Then, for every pair, the ClusPro server[46] was used to generate different docking poses, and the 10 best-ranked solutions were selected. These structures (plus the native ones) were energy minimized using *sander*. Finally, MM-ISMSA calculations were performed on these refined structures.

The quality of the ranking provided by the MM-ISMSA scoring function was compared to the quality of the docking poses in terms of the set of common contacts found in the docking pose and the native structure, that is, the contact overlap (C_{overlap}; eq 21). Two residues were considered to be in contact if any of their respective atoms were closer than a given cutoff distance, 4 Å in our case:

$$C_{\text{overlap}} = \frac{\sum_{ij} C_{ij}^{a} C_{ij}^{b}}{\sqrt{(\sum_{ij} C_{ij}^{a})(\sum_{ij} C_{ij}^{b})}} \in [0,\ 1] \subset \mathbb{R} \tag{21}$$

where *i* and *j* stand for receptor and ligand residues, respectively, while *a* and *b* represent the native and decoy structures, respectively. Then, C_{ij}^{a} refers to the contacts in the native structure (1 if the contact between *i* and *j* exists, 0 otherwise) and C_{ij}^{b}, to those in the decoy.

As it is more common to assess the structural goodness of a scoring function in terms of the root-mean-square deviation (RMSD) found between a scored solution and the corresponding native structure, we have also employed it for comparative purposes although it has been demonstrated that in some cases this measure is devoid of accuracy. We have termed this parameter RMSD$_{\text{L}}$, and its calculation encompasses two steps: first, the alignment of the receptor structure using the McLachlan algorithm[47] and, second, the evaluation of the RMSD for backbone atoms of the whole superimposed structure. These calculations were performed with the ProFit software.[48]

3.7. Pairwise Decomposition. To calculate the pairwise decomposition of the interaction energy, we have devised a weighting scheme by means of which the solvation contribution of a given pair (polar [eq 22] and nonpolar [eq 23]) is balanced by the sum of other interactions (vdW and Coulombic) involving any of the two residues in that pair:

$$\Delta G_{\text{solv}}^{i\leftrightarrow j} = w_{i\leftrightarrow j}^{\text{R}} \Delta G_{\text{solv}}^{i} + w_{i\leftrightarrow j}^{\text{L}} \Delta G_{\text{solv}}^{j} \tag{22}$$

$$\Delta G_{\text{binding}}^{\text{np}} = w_{i\leftrightarrow j}^{\text{R}} \Delta G_{\text{binding}}^{\text{np},i} + w_{i\leftrightarrow j}^{\text{L}} \Delta G_{\text{binding}}^{\text{np},j} \tag{23}$$

where *i* and *j* are the interacting residues belonging to the receptor and ligand, respectively, and

$$w_{i\leftrightarrow j}^{\text{R(L)}} = \frac{\Delta G_{\text{binding}}^{\text{vdW},i\leftrightarrow j} + \Delta G_{\text{binding}}^{\text{elec,coul},i\leftrightarrow j}}{\sum_{k=1}^{N_{\text{L}}(N_{\text{R}})} (\Delta G_{\text{binding}}^{\text{vdW},i\leftrightarrow (j)k} + \Delta G_{\text{binding}}^{\text{elec,coul},i\leftrightarrow (j)k})}$$

Based on this scheme, we have classified receptor–ligand interaction pairs into three types: (a) hydrogen bonding; (b) hydrophobic (which includes ΔG_{vdW} and nonpolar contributions from $\Delta G_{\text{desolv}}^{\text{R}}$, $\Delta G_{\text{desolv}}^{\text{L}}$, and ΔG_{apo}); and (c) hydrophilic (which includes ΔG_{elec} and the polar contribution from $\Delta G_{\text{desolv}}^{\text{R}}$, $\Delta G_{\text{desolv}}^{\text{L}}$, and ΔG_{apo}). Polar and apolar contributions are calculated according to atom types. Namely, N and O atoms are considered polar and the rest apolar. Therefore, any interacting pair is classified as (a) hydrogen bonding, whenever a hydrogen bonding interaction is detected, independently of the other interactions that may occur; (b) hydrophilic, if the relative weight of the hydrophilic term in the total interaction energy is above 60%; (c) hydrophobic, if the relative weight of the hydrophobic term in the total interaction energy is above 60%; and (d) mixed, if the relative weight of the hydrophilic term in the total interaction energy is found to be between 40% and 60%.

3.8. Computational Performance. In this section, we analyze our scoring function (MM-ISMSA) and the MM-PB(GB)SA method in relation to their implementation within the AMBER suite, including the old version (*mm_pbsa.pl*) as well as the new one (*MMPBSA.py*), and their efficiency. First, we analyze the implementation of the algorithms focusing on the programming language employed and how the calculated data are stored and handled. Then, we estimate the efficiency (or the complexity, *T*) of the algorithms employing the commonly used asymptotic approach (assuming a very large amount of input data) and the big *O* notation as the parameter to state the order of running time growth. As the elemental unit function, we consider a single energetic calculation which, depending on the method, is represented by the following functions:

$$T_{\text{MM-ISMSA}} = (N_{\text{R}} \cdot N_{\text{L}}) + N_{\text{R}} + N_{\text{L}} \tag{24}$$

$$T_{\text{MM-PB(GB)SA}} = (N_{\text{R}} + N_{\text{L}})^2 + N_{\text{R}}^2 + N_{\text{L}}^2 \tag{25}$$

N_{R} and N_{L} stand for the number of atoms in the receptor and ligand, respectively.

3.9. MM-ISMSA Graphical User Interface. All of the functionalities available from the MM-ISMSA code have been implemented in a graphical user interface (GUI), written in the Python programming language. This allows its facile use as a plugin to the popular molecular visualization program PyMOL. The GUI front end uses PyMOL software version 1.2 or higher and requires the NumPy (version C 1.3) module and the portable command-line driven graphing utility gnuplot (version 4.6). The GUI has been prepared to be executed on Linux operating systems. The minimum recommended amount of

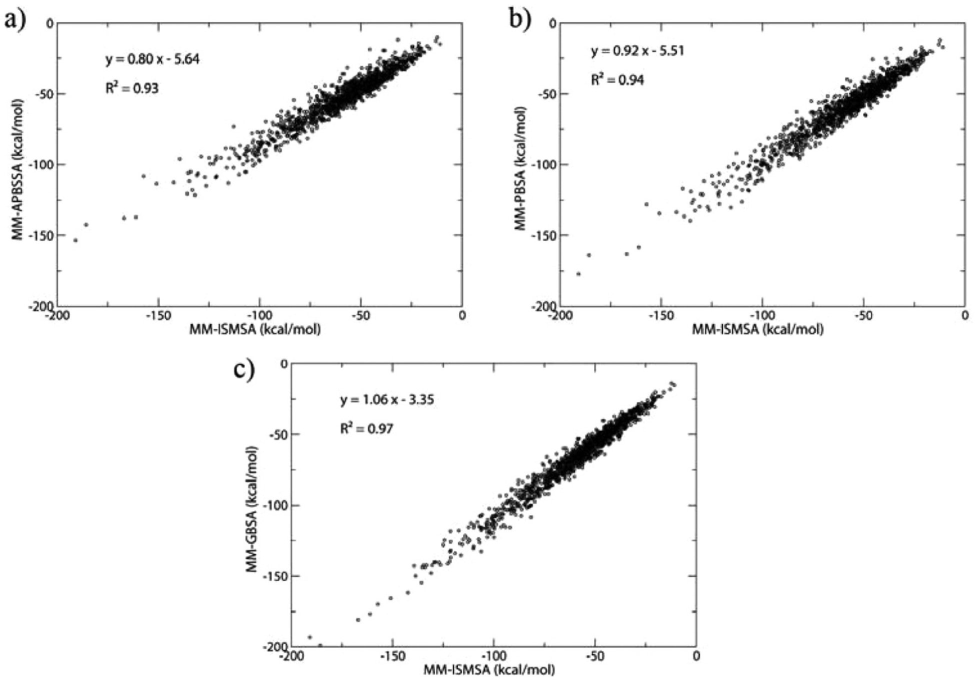


Figure 3. Correlation between total binding free energies ($\Delta G_{\text{binding}}$ in kcal/mol) as obtained by (a) MM-APBSSA, (b) MM-PBSA, and (c) MM-GBSA methods and by the MM–ISMSA method.

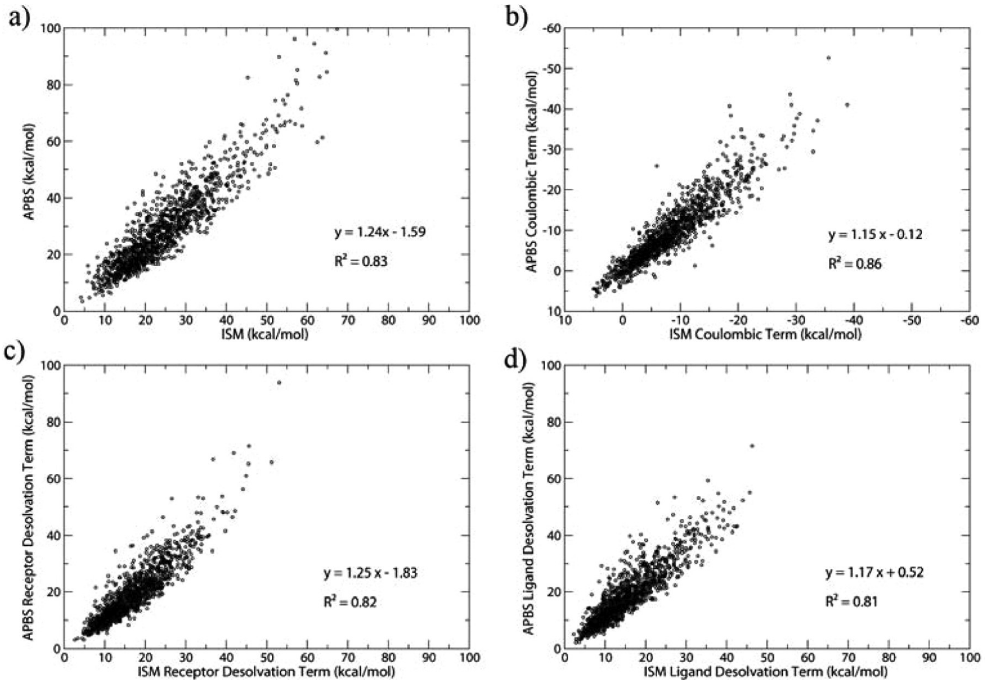


Figure 4. Comparison of total electrostatic binding free energies and their different contributions as obtained by APBS and ISM. (a) Total electrostatic binding free energy. (b) Coulombic contribution. (c) Receptor desolvation. (d) Ligand desolvation.

main memory is around 800 MB (for an average of 900 residues in total per system), and no significant storage capabilities are required to hold the outcomes of the calculations.

4. RESULTS AND DISCUSSION

4.1. MM-ISMSA Compared to MM-APBSSA and MM-PB(GB)SA. In this section, we first compare the numerical values for the total binding free energies ($\Delta G_{\text{binding}}$) as obtained from MM-ISMSA, MM-APBSSA, and MM-PB(GB)SA methods. We find MM-ISMSA reproduces MM-APBSSA's $\Delta G_{\text{binding}}$ with great accuracy ($r^2 = 0.93$, Figure 3a) as it does when compared to MM-PBSA ($r^2 = 0.94$, Figure 3b) or MM-GBSA ($r^2 = 0.97$, Figure 3c). In addition, we obtain small deviations from MM-PBSA (slope = 0.92) and MM-GBSA (slope = 1.06) and slightly higher from APBS (slope = 0.80). Finally, when MM-PBSA is compared to MM-GBSA in terms of the total interaction energy, the correlation coefficient ($r^2 = 0.95$), slope (0.86), and intercept (−3.16) are similar to the values obtained when comparing MM-

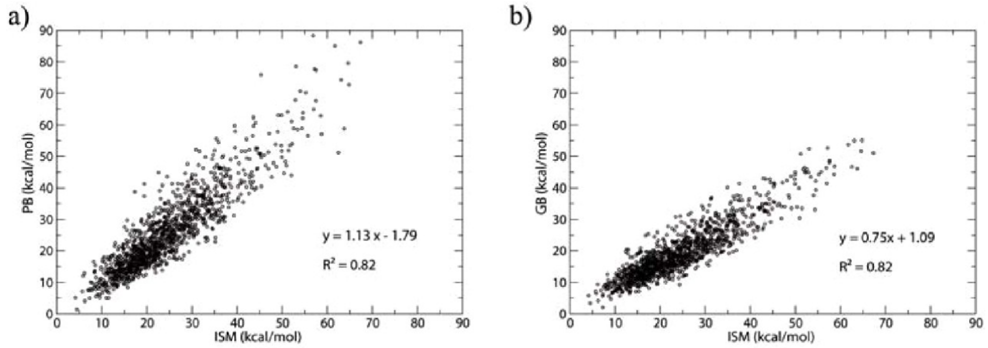


Figure 5. Comparison of total electrostatic binding free energies obtained by ISM with those calculated using (a) PB (as implemented in MM-PBSA) and (b) GB (as implemented in MM-PBSA).

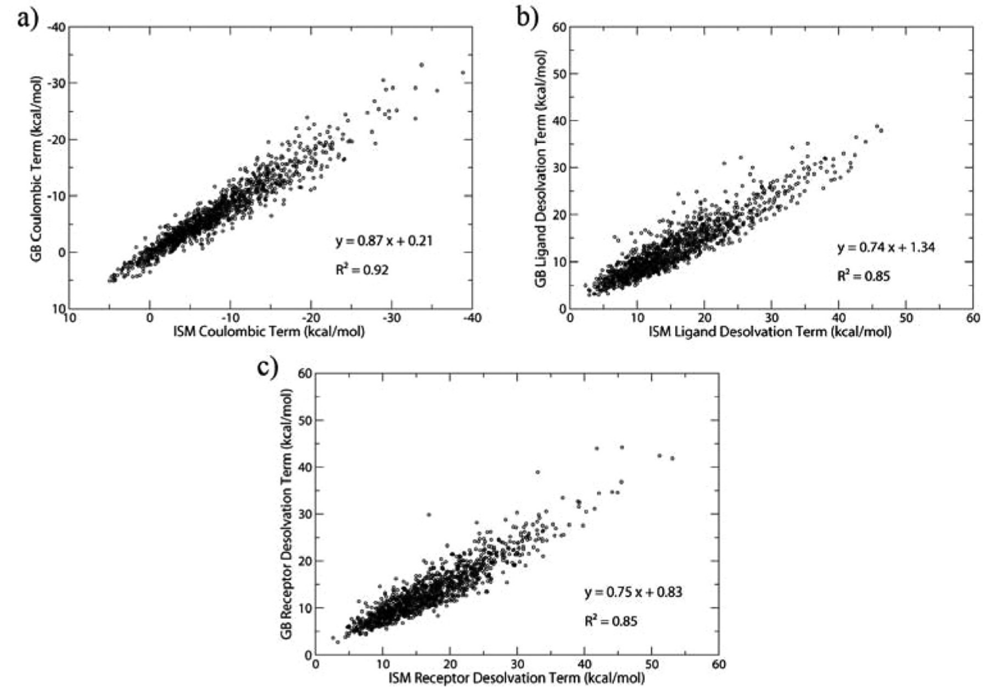


Figure 6. Comparison between GB and ISM in terms of the different contributions to the total electrostatic binding free energy as obtained by applying the three-calculation method. (a) Coulombic contribution. (b) Ligand desolvation. (c) Receptor desolvation.

ISMSA to either MM-GBSA or MM-PBSA. As the nonelectrostatic terms (ΔG_{vdW} and ΔG_{np}) are always computed in the same way, the rest of this section will focus on the electrostatic part ($\Delta G_{\mathrm{binding}}^{\mathrm{elec}}$) of $\Delta G_{\mathrm{binding}}$ and its decompositions into the Coulombic ($\Delta G_{\mathrm{binding}}^{\mathrm{elec,coul}}$) and receptor and ligand desolvation terms ($\Delta G_{\mathrm{binding}}^{\mathrm{elec,desolv_R}}$ and $\Delta G_{\mathrm{binding}}^{\mathrm{elec,desolv_L}}$, respectively). In the following, unless otherwise stated, we will refer to the methods simply as APBS, ISM, GB, or PB.

The correlation between APBS and ISM in terms of $\Delta G_{\mathrm{binding}}^{\mathrm{elec}}$ is very good (r^2 = 0.83, Figure 4a), and the same is true when individual components are considered: $\Delta G_{\mathrm{binding}}^{\mathrm{elec,coul}}$ (r^2 = 0.86, Figure 4b), $\Delta G_{\mathrm{binding}}^{\mathrm{elec,desolv_R}}$ (r^2 = 0.82, Figure 4c), and $\Delta G_{\mathrm{binding}}^{\mathrm{elec,desolv_L}}$ (r^2 = 0.81, Figure 4d). Slopes are close to unity (1.15 for $\Delta G_{\mathrm{binding}}^{\mathrm{elec,coul}}$ and 1.17 for $\Delta G_{\mathrm{binding}}^{\mathrm{elec,desolv_L}}$) or slightly higher (1.24 and 1.25 for $\Delta G_{\mathrm{binding}}^{\mathrm{elec}}$ and $\Delta G_{\mathrm{binding}}^{\mathrm{elec,desolv_R}}$, respectively), and the intercepts are very small in all cases.

Furthermore, when ISM is compared to either PB (Figure 5a) or GB (Figure 5b), the correlation coefficients (r^2 = 0.82 in both cases), slopes (1.13 and 0.75), and intercepts (1.79 and 1.09) are on the same order as before.

As stated above, the thermodynamic cycle on which ISM is based (Figure 2) directly dissects, by construction, the electrostatic contribution to binding into its components (eq 13) in just a single step. On the contrary, three calculations are needed to obtain the same partition scheme when PB or GB models (eqs 9−11) are employed. In particular, when the triple calculation is used, ISM compared to GB (Figure 6) affords a correlation coefficient close to 0.9 (r^2 = 0.85) for both desolvations (Figure 6b,c), while the slopes indicate that these values are overestimated by the GB model (0.75 and 0.74 for receptor and ligand desolvation, respectively). The agreement between Coulombic terms is even better, with both the correlation coefficient and the slope yielding a value of ~0.9 (Figure 6a).

4.2. The MM-ISMSA Scoring Function in Protein−Protein Docking.

We have challenged the MM-ISMSA scoring function by mixing the native structures for a set of 15 protein−protein complexes with docking decoys (10 for each complex obtained from the ClusPro program) and then evaluating the binding free energy for each complex (eq 14). Although this function provides more detailed information on the binding event than just the global value of the binding energy (hydrogen bond types and count, HBScore, van der Waals and electrostatic interactions, desolvation terms, etc.), we were not able, however, to discriminate between the native structure and the decoys on the basis of this detailed individual information. Rather, it was only the global binding energy parameter which really selects the native pose as the best scored pose among the decoys in 12 out of the 15 complexes studied here (80% of success, Figure 7).

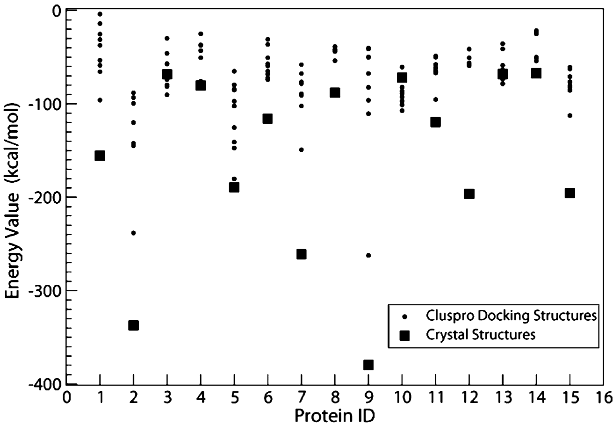


Figure 7. Total free energies of binding for each of the decoys and the X-ray structure in the test set. The small black dots represent the decoys, while the black squares symbolize the X-ray structure.

The finding that descriptors as important as the types and number of hydrogen bonds or the electrostatic complementarity between receptor and ligand, considered as essential in protein–protein recognition, are not able to discriminate a native structure from a pool of decoys might be due to the ability of ClusPro to provide challenging docking poses, although with a significant difference in contact patterns (Figure 8a). In fact, less than half of the complexes showed $C_{overlap}$ values above 0.7. On the contrary, ClusPro is able to supply in most of the cases (10 out of 15) a decoy closer than 10 Å to the native structure (Figure 8b).

Finally, we have a weak correlation ($r^2 \sim 0.5-0.6$) between the rankings provided by MM-ISMSA and the contact overlap (Figure 9a) or $RMSD_L$ (Figure 9b). Despite these modest figures it is worth noting that in many cases (10 out of 12 if complexes without very low $C_{overlap}$ values are excluded) good docking poses according to the MM-ISMSA scoring function usually correspond to an assembly with high and low $C_{overlap}$ and $RMSD_L$ values, respectively.

4.3. Pairwise Decomposition. Figure 10 shows the relationship between all individual inter-residue interaction energies calculated by MM-ISMSA and MM-GBSA for the 15 proteins in the second test set. A very good correlation was obtained ($r^2 = 0.95$), indicating the feasibility of employing MM-ISMSA as an alternative to MM-GBSA.

4.4. Computational Performance. In routine calculations, when dealing with relatively large protein–protein complexes, we have found a tremendous bottleneck in the use of the mm_pbsa.pl module in AMBER to obtain the partition of the interaction energy into pairwise residue contributions. In fact, this decomposition scheme is actually impracticable in many cases due to the large execution times required, not to mention the huge demands on memory and disk space. After some careful examination, we concluded that the main culprit for this appeared to be the use of *hash tables* to store the data. On the other hand, we were aware that *perl*, being an interpreted language, is not adequate to handle highly computationally demanding mathematical calculations. To circumvent these problems, we decided to rewrite in C programming language the main part of the *mm_pbsa.pl* module where the statistical calculations are performed (*mm_pbsa_statistics.pm* module), replacing the *hash* tables with single *arrays*. We refer to this new code as optimized *mm_pbsa.pl*. By the time we were developing this optimized code, a new version of the *mm_pbsa.pl* module was released under the name *MMPBSA.py*. This new version avoids some problems of the early module by executing sander binaries on each snapshot and using a more adequate data structure and methods from *Python* programming language. Then, we compared the performance of the MM-ISMSA code to the old (*mm_pbsa.pl*) and new (*MMPBSA.py*) modules implemented in the AMBER package and to the C optimized module (Figure 11) in terms of execution times.

Three main observations can be derived from Figure 11: First, in the limit of high number of residues, the ISM code achieves execution times 4 orders of magnitude smaller than the old *mm_pbsa.pl* module. Second, at the same limit, the new implemented *MMPBSA.py* and our *mm_pbsa.pl* optimized module show very close execution times. Third, ISM outperforms all the other codes tested here, irrespective of the number of residues used in the calculation.

Continuing with the pairwise residue decomposition, and according to eqs 24 and 25 above, it is relatively straightforward to realize that the efficiency of both algorithms grows on the

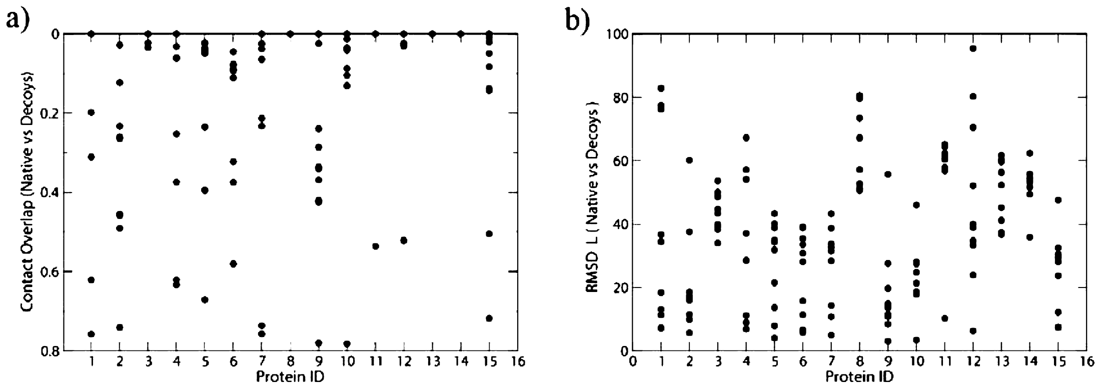


Figure 8. (a) Contact overlap (eq 21) and (b) $RMSD_L$ for the decoys generated with ClusPro for each of the targets in the test set.

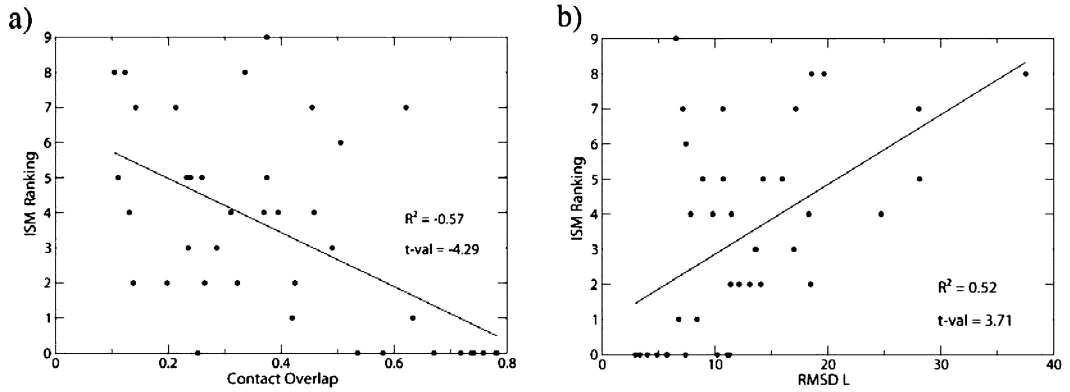


Figure 9. Relationships between (a) contact overlap (eq 21) and (b) $\mathrm{RMSD_L}$ and ISM ranking based on the total binding energy for the decoys generated with ClusPro for each of the targets in the test set.

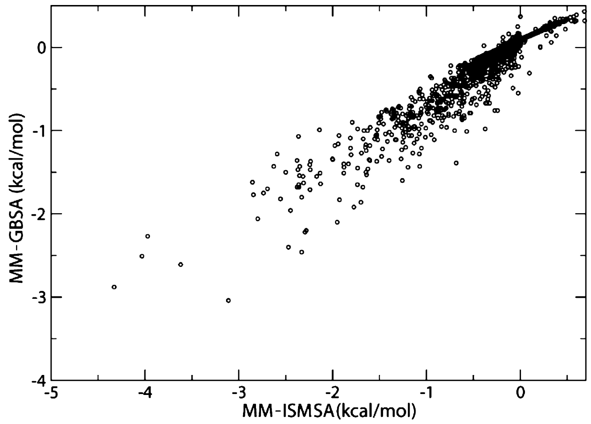


Figure 10. Relationship between the pairwise-decomposed binding energies obtained by MM-ISMSA and MM-GBSA methods for the 15 proteins in the second test set.

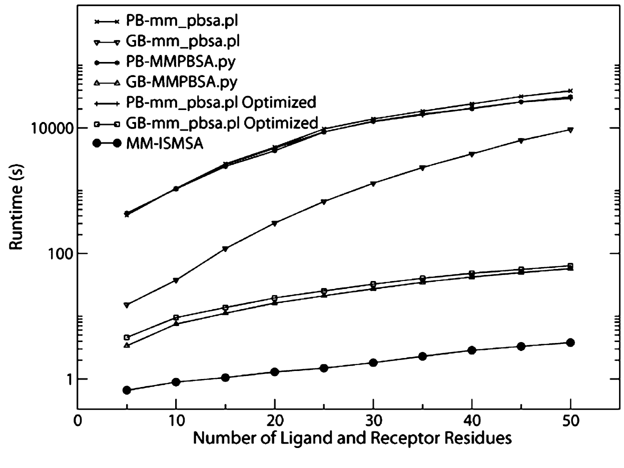


Figure 11. Run time (log scale) dependence on the number of residues for the different methods compared in this study.

order of $O(N^2)$, provided that N_R and N_L are of the same magnitude. However, the number of operations to be done in MM-ISMSA is smaller than in MM-PB(GB)SA, and this is due to the double counting of the cross-interaction terms between the receptor and the ligand in the latter method (double summation in eq 7, ΔG_{pol} term, with the indexes running from 1 to N for both receptor and ligand, respectively), while in the former they are calculated just once (single summation in the second term on the right-hand side of eq 14 with a single index running from 1 to the total number of atoms in the complex). To analyze in more detail the relative performance of both algorithms, we studied the two most representative cases: (a) protein−protein docking assuming an equal number of residues for both the receptor and ligand and (b) protein−ligand docking, where the number of ligand residues (one) can be neglected as compared to the number of residues in the receptor. Then, in the case of protein−protein docking:

$$\text{if } N_R \approx N_L$$
$$= N,\ \lim_{N\to\infty} \frac{T_{\mathrm{MM-PB(GB)SA}}(N)}{T_{\mathrm{MM-ISMSA}}(N)}$$
$$= \lim_{N\to\infty} \frac{6N^2}{N^2 + 2N} = 6 \tag{26}$$

and in the case of protein−ligand docking, we obtain:

$$\text{if } N_R \gg N_L,\ \lim_{N_R\to\infty} \frac{T_{\mathrm{MM-PB(GB)SA}}(N_R)}{T_{\mathrm{MM-ISMSA}}(N_R)}$$
$$= \lim_{N_R\to\infty} \frac{{N_R}^2}{N_R} = \infty \tag{27}$$

Therefore, even though both algorithms scale within the same complexity order for protein−protein interactions (eq 26), the differences are significant mainly due to the fact that MM-ISMSA performs six times less operations than MM-PB(GB)SA. To illustrate this point, consider a protein−protein complex consisting of a dimer with an average size of 150 residues per monomer. Assuming that MM-ISMSA requires the same amount of time to perform a unit calculation as MM-GBSA or MM-PBSA does (around 0.005 and 2 s, respectively), these differences would be translated into 83% or 98% savings in execution time, for protein−protein or protein−ligand complexes, compared to MM-GBSA or MM-PBSA, respectively.

Finally, it is worth commenting that under the assumption that *sander* energies are calculated following an optimized compiled code, and that the statistical optimized code accounts for less than 0.01% of the total execution time, we can conclude that *sander* has reached its optimization limit, and no further improvement can be performed unless a new implementation of the part of the code in charge of the energy calculations is undertaken.

4.5. Hydrogen Bonding.

Table 1 contains the geometrical parameters estimated from the statistical analysis of the hydrogen bonds.

Table 1. Geometrical Parameters Defining the Hydrogen Bonding Interactions[a]

	min	min−ideal	max−ideal	max
r	1.5[a]	1.8[a]	2.4	2.7
α	100[b]	130	165[b]	180
β	90	115	145	180

[a] r, in Å, is the distance between the hydrogen and acceptor atoms, and α and β, in degrees, are the angles between donor, hydrogen, and acceptor atoms, and the hydrogen, acceptor, and the atom bound to the acceptor atom, respectively. [b] The final values for these variables were finely tuned to avoid penalizing those hydrogen bonds that, although geometrically plausible, are either statistically underrepresented ($\alpha_{\min}$ = 90° and $\alpha_{\text{max-ideal}}$ = 180°) or already penalized by the van der Waals term ($r_{\min}$ = 0 and $r_{\text{min-ideal}}$ = 0).

Using these values, we have been able to correctly recover 92.5% of the hydrogen bonds in the training set and 83.7% in the first test set. In most of the cases, the reason why those hydrogen bonds were not identified was either because their defining values were close to the limiting ones or because their geometrical arrangement cast some doubts on their formation. But more importantly, their associated interaction energies were well below the average energies obtained for the recovered hydrogen bonds. Finally, for the second test set, all the hydrogen bonds were identified when compared to HBPLUS as the reference method.

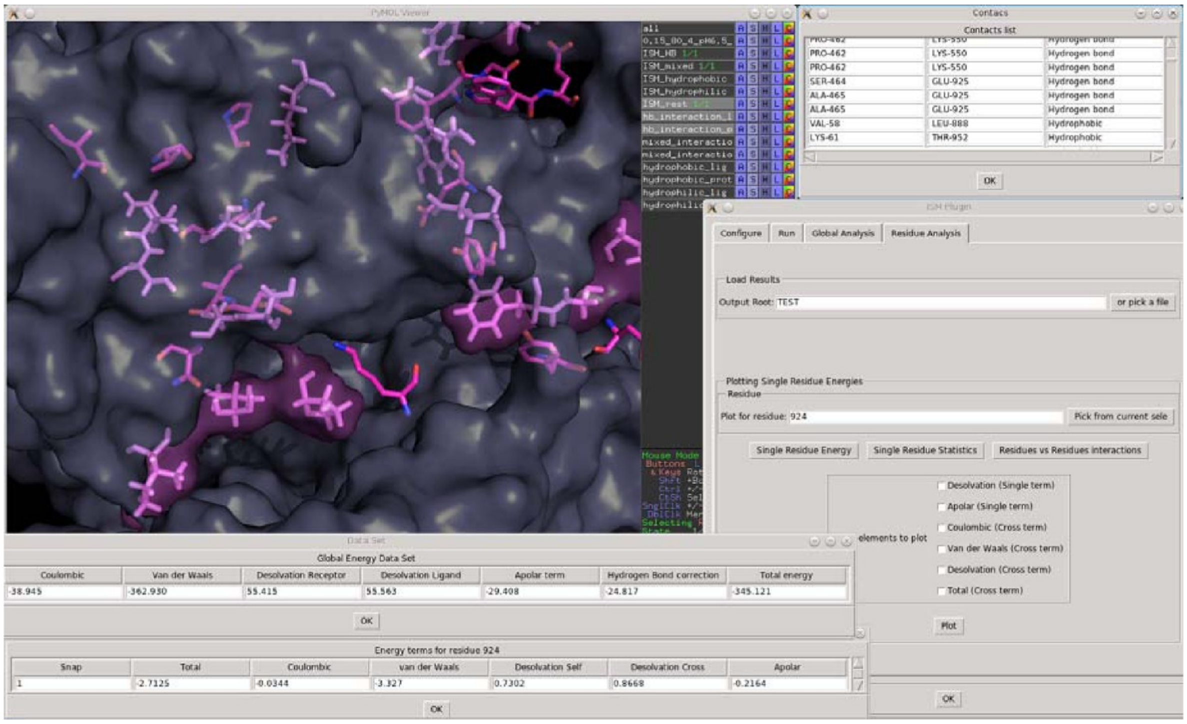
4.6. The PyMOL Plugin.

Due to the possible steep learning curve of the application, which requires significant knowledge of the underlying operations, we have implemented a GUI accessible within the popular molecular editor PyMOL. This plugin was designed to execute the application and process the results of the calculations. Figure 12 shows some of the more important graphic capabilities of the plugin, while a complete description is included in the user's guide available from the Web site.

Once the plugin is invoked from the PyMOL plugin interface, it displays a window with four main sections organized in different tabs: Configure, Run, Global Analysis, and Residue Analysis. At the bottom of this window, an OK button closes the plugin. The Configure tab, which is activated by default, comprises the following fields: (a) an area that contains two user-configurable variables (the path to the ISM executable file and the working directory) and a Save button that allows the user to save this configuration; (b) a second area that depicts the logos of the institutions involved in its development; and (c) a scrollable window with information about the plugin, its authors and institutions involved, contact details, the license, the disclaimer, and plugin update information.

The Run tab has two sections: (a) on the left, the user can (i) select to work with a single structure (either a unique PDB file or two top and crd AMBER-type files) or with an MD trajectory file, (ii) set the type of input (single structure or trajectory file), and (iii) set up the per residue analysis; (b) on the right, the user can (i) set up the root name for the output files, (ii) in the case of an MD trajectory file, select the initial, final, and step size of the snapshot to be analyzed, and (iii) explore the output of the calculation in a small window. A Run button on the right on this window starts the calculation. A Stop button is also provided to stop the calculation.

In the Global Analysis tab, the user can do the following: (a) Load the results from a previous calculation by typing the root of the file name or selecting the file. (b) Export the results of the current calculation. The options are (i) a PDB file with the desolvation energy values per residue loaded into the temperature factor field for an easy visualization in PyMOL, (ii) a file



Contacts list

PRO-462	LYS-550	Hydrogen bond
PRO-462	LYS-550	Hydrogen bond
PRO-462	LYS-550	Hydrogen bond
SER-464	GLU-925	Hydrogen bond
ALA-465	GLU-925	Hydrogen bond
ALA-465	GLU-925	Hydrogen bond
VAL-58	LEU-888	Hydrophobic
LYS-61	THR-952	Hydrophobic

Load Results

Output Root: TEST

Plotting Single Residue Energies

Residue

Plot for residue: 924

elements to plot

- ☐ Desolvation (Single term)
- ☐ Apolar (Single term)
- ☐ Coulombic (Cross term)
- ☐ Van der Waals (Cross term)
- ☐ Desolvation (Cross term)
- ☐ Total (Cross term)

Global Energy Data Set

Coulombic	Van der Waals	Desolvation Receptor	Desolvation Ligand	Apolar term	Hydrogen Bond correction	Total energy
-38.945	-362.930	55.415	55.563	-29.408	-24.817	-345.121

Energy terms for residue 924

Snap	Total	Coulombic	van der Waals	Desolvation Self	Desolvation Cross	Apolar
1	-2.7125	-0.0344	-3.327	0.7302	0.8668	-0.2164

Figure 12. Combined snapshots of some graphical capabilities provided by the PyMOL-implemented MM-ISMSA plugin.

with the Global Energy and its components, and (iii) a file with the Global Statistics in case an MD trajectory is processed. (c) Visualize the contacts between ligand and receptor in PyMOL at various levels: (1) all the contacts, (2) only hydrophobics, (3) only hydrophilics, (4) mixed contacts, and (5) hydrogen bonds. It is worth remembering that the contact classification is not based on the type of side chains found in the interacting residue but on the kind of interaction energy (see section 2.5). (d) Plot the evolution of the different energetic terms along the MD trajectory. These are Coulombic, van der Waals, receptor and ligand desolvation energies, apolar, hydrogen bond, and total.

The last tab, Residue Analysis, allows the user to perform single residue analysis and visualization. The main functionalities available are as follows: (a) Load the results of a previous calculation by typing the root of the file name or selecting the file. (b) Export the results of the current calculation. The options are (i) a file with the energies by residue with its components, (ii) a file with the Global Statistics in case an MD trajectory is processed, and (iii) a matrix with residue−residue interaction energies. (c) Plot the evolution of the different energy terms along the MD trajectory. These are Coulombic (cross term), van der Waals (cross term), receptor and ligand desolvation energies (single and cross terms), apolar (single term), and total (cross term).

Within PyMOL, there are currently some other plugins developed to run and analyze MD trajectories produced with the AMBER suite of programs,[49] to compute molecular electrostatic potentials that can be used as the basis for the estimation of binding and desolvation energies,[30] or to calculate protein−protein interactions.[50] However, they are not as complete as our MM-ISMSA plugin. On the other hand, a great variety of tools are included within the molecular visualization program VMD[51] to analyze MD trajectories calculated with the CHARMm force field. But again, they appear as separate plugins although they are completely integrated within the VMD working environment. In addition, there are some Web-based applications that can estimate contact and binding free energies in protein−protein complexes,[52] predict hot spot residues in protein interfaces,[53] and analyze and visualize contacts at the interface of biomolecular complexes,[42,54] just to mention a few. Finally, and as far as MD protocols are concerned, GUIs for the most commonly used MD codes have appeared recently.[55] Summarizing, we think that our MM-ISMSA PyMOL plugin condenses some of the advantages of the aforementioned tools while maintaining its integrity in a single, unified tool that is implemented in a widely used and powerful molecular graphics program.

CONCLUSIONS

A new scoring function for protein−protein docking, MM-ISMSA, which incorporates desolvation and hydrogen bonding terms explicitly, is presented. This function allows calculation in a given protein−protein complex of (i) the total binding free energy, (ii) the contributions from different components, (iii) individual residue desolvations, and (iv) all pairwise residue interactions.

The accuracy of MM-ISMSA was tested using two different protein−protein sets; in the first one, a total of 1242 structures (15 experimentally determined antigen−antibody complexes and 1227 decoys, with a maximum of 100 decoys per complex) were used to study whether or not MM-ISMSA was able to reproduce the interaction energies (i−iv above) as compared to other well-established methods in the field. The results showed that in all cases a good agreement was achieved. The second set (15 diverse experimentally determined complexes and 135 decoys, with a maximum of 10 per complex) was used to test the ability of the MM-ISMSA scoring function to select near-native docking poses from a pool of solutions (decoys). The outcome was an 80% success rate.

Besides its accuracy, an additional advantage of MM-ISMSA is its reduced computational cost, as it is able to analyze large systems (~1000 residues) in less than 5 s, yielding a complete report on the different energy terms and its decomposition. Compared to the commonly used MM-PB(GB)SA method (as implemented in AmberTools), MM-ISMSA performs 6 times fewer calculations than MM-PB(GB)SA. For this reason, it should be particularly preferable to process long MD trajectories.

Finally, MM-ISMSA has been implemented as a plugin for the popular molecular visualization program PyMOL, although it can also be used in command-line mode. The code is open source and is offered free of charge to noncommercial parties for download following registration at the CBM Bioinformatics Unit's web page (http://ub.cbm.uam.es/).

AUTHOR INFORMATION

Corresponding Author

*Phone: + 34 911 964 633. Fax: + 34 911 964 422. E-mail: amorreale@cbm.uam.es.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was supported by grants from CICYT (SAF2009-13914-C02-02 to F.G.) and Comunidad Autónoma de Madrid (S-BIO-0214-2006 [BIPEDD] and S2010-BMD-2457 [BIPEDD2] to A.M. and F.G.). A.M. acknowledges financial support from Fundación Severo Ochoa through the AMAROUTO program. J.K., H.G.D.S., and A.N.-S. were supported by grants from Ministerio de Economía y Competitividad (BFU2011-24595) and Ministerio de Educación y Ciencia (CSD2006-00023). R.G.-R. enjoyed a MICINN contract from "Programa de Personal Técnico y de Apoyo 2008", and Á.C.-C. is the recipient of FPU grant AP2009-0203 from Ministerio de Educación. We thank Dr. Pablo Chacón (IQFR-CSIC, Madrid) for providing us with the structures used in the validation test set and Dr. Ugo Bastolla for useful discussions and encouragement.

REFERENCES

(1) Jorgensen, W. L. The many roles of computation in drug discovery. *Science* **2004**, *303*, 1813−1818.

(2) Kortemme, T.; Joachimiak, L. A.; Bullock, A. N.; Schuler, A. D.; Stoddard, B. L.; Baker, D. Computational redesign of protein-protein interaction specificity. *Nat. Struct. Mol. Biol.* **2004**, *11*, 371−379.

(3) Halperin, I.; Ma, B.; Wolfson, H.; Nussinov, R. Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins* **2002**, *47*, 409−443.

(4) Ripphausen, P.; Nisius, B.; Peltason, L.; Bajorath, J. Quo vadis, virtual screening? A comprehensive survey of prospective applications. *J. Med. Chem.* **2010**, *53*, 8461−8467.

(5) Ben-Shimon, A.; Eisenstein, M. Computational mapping of anchoring spots on protein surfaces. *J. Mol. Biol.* **2010**, *402*, 259−277.

(6) McCammon, J. A.; Gelin, B. R.; Karplus, M. Dynamics of folded proteins. *Nature* **1977**, *267*, 585−590.

(7) Jorgensen, W. L. Efficient drug lead discovery and optimization. *Acc. Chem. Res.* **2009**, *42*, 724−733.

(8) Shaw, D. E.; Deneroff, M. M.; Dror, R. O.; Kuskin, J. S.; Larson, R. H.; Salmon, J. K.; Young, C.; Batson, B.; Bowers, K. J.; Chao, J. C.;

Eastwood, M. P.; Gagliardo, J.; Grossman, J. P.; Ho, C. R.; Ierardi, D. J.; Kolossvry, I.; Klepeis, J. L.; Layman, T.; McLeavey, C.; Moraes, M. A.; Mueller, R.; Priest, E. C.; Shan, Y.; Spengler, J.; Theobald, M.; Towles, B.; Wang, S. C. Anton, a special-purpose machine for molecular dynamics simulation. *SIGARCH Comput. Archit. News* **2007**, *35*, 1−12.

(9) Top 500 Supercomputer sites. http://www.top500.org/ (accessed July 20, 2012).

(10) Lindorff-Larsen, K.; Piana, S.; Dror, R. O.; Shaw, D. E. How fast-folding proteins fold. *Science* **2011**, *334*, 517−520.

(11) Kruse, A. C.; Hu, J.; Pan, A. C.; Arlow, D. H.; Rosenbaum, D. M.; Rosemond, E.; Green, H. F.; Liu, T.; Chae, P. S.; Dror, R. O.; Shaw, D. E.; Weis, W. I.; Wess, J.; Kobilka, B. K. Structure and dynamics of the M3 muscarinic acetylcholine receptor. *Nature* **2012**, *482*, 552−556.

(12) Shan, Y.; Kim, E. T.; Eastwood, M. P.; Dror, R. O.; Seeliger, M. A.; Shaw, D. E. How does a drug molecule find its target binding site? *J. Am. Chem. Soc.* **2011**, *133*, 9181−9183.

(13) Orozco, M.; Luque, F. J. Theoretical Methods for the Description of the Solvent Effect in Biomolecular Systems. *Chem. Rev.* **2000**, *100*, 4187−4226.

(14) Roux, B.; Simonson, T. Implicit solvent models. *Biophys. Chem.* **1999**, *78*, 1−20.

(15) Honig, B.; Nicholls, A. Classical electrostatics in biology and chemistry. *Science* **1995**, *268*, 1144−1149.

(16) Still, W. C.; Tempczyk, A.; Hawley, R. C.; Hendrickson, T. Semianalytical treatment of solvation for molecular mechanics and dynamics. *J. Am. Chem. Soc.* **1990**, *112*, 6127−6129.

(17) Kuhlman, B.; Baker, D. Native protein sequences are close to optimal for their structures. *Proc. Natl. Acad. Sci. U. S. A.* **2000**, *97*, 10383−10388.

(18) Yin, S.; Biedermannova, L.; Vondrasek, J.; Dokholyan, N. V. MedusaScore: an accurate force field-based scoring function for virtual drug screening. *J. Chem. Inf. Model.* **2008**, *48*, 1656−1662.

(19) Morreale, A.; Gil-Redondo, R.; Ortiz, A. R. A new implicit solvent model for protein-ligand docking. *Proteins* **2007**, *67*, 606−616.

(20) Case, D. A.; Cheatham, T. E., 3rd; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M., Jr.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668−1688.

(21) *The PyMOL Molecular Graphics System*, version 1.1r2pre; Schrödinger, LLC: Portland, OR, 2008.

(22) Gilson, M. K.; Zhou, H.-X. Calculation of Protein-Ligand Binding Affinities. *Annu. Rev. Biophys. Biomol. Struct.* **2007**, *36*, 21−42.

(23) Aqvist, J.; Medina, C.; Samuelsson, J. E. A new method for predicting binding affinity in computer-aided drug design. *Protein Eng.* **1994**, *7*, 385−391.

(24) Kollman, P. A.; Massova, I.; Reyes, C.; Kuhn, B.; Huo, S.; Chong, L.; Lee, M.; Lee, T.; Duan, Y.; Wang, W.; Donini, O.; Cieplak, P.; Srinivasan, J.; Case, D. A.; Cheatham, T. E., 3rd. Calculating structures and free energies of complex molecules: combining molecular mechanics and continuum models. *Acc. Chem. Res.* **2000**, *33*, 889−897.

(25) Gilson, M. K.; Honig, B. Calculation of the total electrostatic energy of a macromolecular system: solvation energies, binding energies, and conformational analysis. *Proteins* **1988**, *4*, 7−18.

(26) Zhou, H. X.; Gilson, M. K. Theory of free energy and entropy in noncovalent binding. *Chem. Rev.* **2009**, *109*, 4092−4107.

(27) Warwicker, J.; Watson, H. C. Calculation of the electric potential in the active site cleft due to α-helix dipoles. *J. Mol. Biol.* **1982**, *157*, 671−679.

(28) Baker, N.; Holst, M.; Wang, F. Adaptive multilevel finite element solution of the Poisson−Boltzmann equation II. Refinement at solvent-accessible surfaces in biomolecular systems. *J. Comput. Chem.* **2000**, *21*, 1343−1352.

(29) Rashin, A. A.; Namboodiri, K. A simple method for the calculation of hydration enthalpies of polar molecules with arbitrary shapes. *J. Phys. Chem.* **1987**, *91*, 6003−6012.

(30) Baker, N. A.; Sept, D.; Joseph, S.; Holst, M. J.; McCammon, J. A. Electrostatics of nanosystems: Application to microtubules and the ribosome. *Proc. Natl. Acad. Sci. U.S.A.* **2001**, *98*, 10037−10041.

(31) Mehler, E. L., The Lorentz-Debye-Sack Theory and Dielectric Screening of Electrostatic Effects in Proteins and Nucleic Acids. In *Molecular Electrostatic Potential: Concepts and Applications*; Murray, J. S.; Sen, K., Eds.; Elsevier Science: Amsterdam, 1996; Vol. 3, pp 371−405.

(32) Hassan, S. A.; Guarnieri, F.; Mehler, E. L. A General Treatment of Solvent Effects Based on Screened Coulomb Potentials. *J. Phys. Chem. B* **2000**, *104*, 6478−6489.

(33) Hassan, S. A.; Mehler, E. L.; Zhang, D.; Weinstein, H. Molecular dynamics simulations of peptides and proteins with a continuum electrostatic model based on screened Coulomb potentials. *Proteins* **2003**, *51*, 109−125.

(34) Brooks, B. R.; Bruccoleri, R. E.; Olafson, B. D.; States, D. J.; Swaminathan, S.; Karplus, M. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem.* **1983**, *4*, 187−217.

(35) Luque, F. J.; Curutchet, C.; Munoz-Muriedas, J.; Bidon-Chanal, A.; Soteras, I.; Morreale, A.; Gelpi, J. L.; Orozco, M. Continuum solvation models: Dissecting the free energy of solvation. *Phys. Chem. Chem. Phys.* **2003**, *5*, 3827−3836.

(36) Morreale, A.; de la Cruz, X.; Meyer, T.; Gelpi, J. L.; Luque, F. J.; Orozco, M. Partition of protein solvation into group contributions from molecular dynamics simulations. *Proteins* **2005**, *58*, 101−109.

(37) Duan, Y.; Wu, C.; Chowdhury, S.; Lee, M. C.; Xiong, G.; Zhang, W.; Yang, R.; Cieplak, P.; Luo, R.; Lee, T.; Caldwell, J.; Wang, J.; Kollman, P. A point-charge force field for molecular mechanics simulations of proteins based on condensed-phase quantum mechanical calculations. *J. Comput. Chem.* **2003**, *24*, 1999−2012.

(38) Lee, M. C.; Duan, Y. Distinguish protein decoys by using a scoring function based on a new AMBER force field, short molecular dynamics simulations, and the generalized born solvent model. *Proteins* **2004**, *55*, 620−634.

(39) Weiser, J.; Shenkin, P. S.; Still, W. C. Approximate atomic surfaces from linear combinations of pairwise overlaps (LCPO). *J. Comput. Chem.* **1999**, *20*, 217−230.

(40) Tsui, V.; Case, D. A. Theory and applications of the generalized born solvation model in macromolecular simulations. *Biopolymers* **2000**, *56*, 275−291.

(41) Wallace, A. C.; Laskowski, R. A.; Thornton, J. M. LIGPLOT: a program to generate schematic diagrams of protein-ligand interactions. *Protein Eng.* **1995**, *8*, 127−134.

(42) Laskowski, R. A. PDBsum new things. *Nucleic Acids Res.* **2009**, *37*, D355−359.

(43) McDonald, I. K.; Thornton, J. M. Satisfying hydrogen bonding potential in proteins. *J. Mol. Biol.* **1994**, *238*, 777−793.

(44) Garzon, J. I.; Lopez-Blanco, J. R.; Pons, C.; Kovacs, J.; Abagyan, R.; Fernandez-Recio, J.; Chacon, P. FRODOCK: a new approach for fast rotational protein-protein docking. *Bioinformatics* **2009**, *25*, 2544−2551.

(45) Gordon, J. C.; Myers, J. B.; Folta, T.; Shoja, V.; Heath, L. S.; Onufriev, A. H++: a server for estimating pKas and adding missing hydrogens to macromolecules. *Nucleic Acids Res.* **2005**, *33*, W368−371.

(46) Comeau, S. R.; Gatchell, D. W.; Vajda, S.; Camacho, C. J. ClusPro: a fully automated algorithm for protein-protein docking. *Nucleic Acids Res.* **2004**, *32*, W96−99.

(47) McLachlan, A. Rapid comparison of protein structures. *Acta Crystallogr., Sect. A* **1982**, *38*, 871−873.

(48) Martin, A. C. R.; Porter, C. T. *ProFit*, version 3.1; University College London: London, 2009.

(49) Lill, M.; Danielson, M. Computer-aided drug design platform using PyMOL. *J. Comput.-Aided Mol. Des.* **2011**, *25*, 13−19.

(50) Steinkellner, G.; Rader, R.; Thallinger, G. G.; Kratky, C.; Gruber, K. VASCo: computation and visualization of annotated protein surface contacts. *BMC Bioinf.* **2009**, *10*, 32.

(51) Humphrey, W.; Dalke, A.; Schulten, K. VMD: visual molecular dynamics. *J. Mol. Graph.* **1996**, *14* (33−38), 27−38.

(52) Camacho, C. J.; Zhang, C. FastContact: rapid estimate of contact and binding free energies. *Bioinformatics* **2005**, *21*, 2534−2536.

(53) Assi, S. A.; Tanaka, T.; Rabbitts, T. H.; Fernandez-Fuentes, N. PCRPi: Presaging Critical Residues in Protein interfaces, a new

computational tool to chart hot spots in protein interfaces. *Nucleic Acids Res.* **2010**, *38*, e86.

(54) Vangone, A.; Spinelli, R.; Scarano, V.; Cavallo, L.; Oliva, R. COCOMAPS: a web application to analyze and visualize contacts at the interface of biomolecular complexes. *Bioinformatics* **2011**, *27*, 2915−2916.

(55) Knapp, B.; Schreiner, W. Graphical user interfaces for molecular dynamics-quo vadis? *Bioinform. Biol. Insights* **2009**, *3*, 103−107.