VSDMIP: virtual screening data management on an integrated platform

Rubén Gil-Redondo · Jorge Estrada ·
Antonio Morreale · Fernando Herranz ·
Javier Sancho · Ángel R. Ortiz

Abstract A novel software (VSDMIP) for the virtual screening (VS) of chemical libraries integrated within a MySQL relational database is presented. Two main features make VSDMIP clearly distinguishable from other existing computational tools: (i) its database, which stores not only ligand information but also the results from every step in the VS process, and (ii) its modular and pluggable architecture, which allows customization of the VS stages (such as the programs used for conformer generation or docking), through the definition of a detailed workflow employing user-configurable XML files. VSDMIP, therefore, facilitates the storage and retrieval of VS results, easily adapts to the specific requirements of each method and tool used in the experiments, and allows the comparison of different VS methodologies. To validate the usefulness of VSDMIP as an automated tool for carrying out VS several experiments were run on six protein targets (acetylcholinesterase, cyclin-dependent kinase 2, coagulation factor Xa, estrogen receptor alpha, p38 MAP kinase, and neuraminidase) using nine binary (actives/inactive) test sets. The performance of several VS configurations was evaluated by means of enrichment factors and receiver operating characteristic plots.

Keywords Docking · Virtual screening · Drug design · Database · Platform

Abbreviations AChE Acetylcholinesterase fXa Coagulation factor Xa CDK2 Cyclic dependant kinase 2 Era Estrogen receptor α p38MAP MAP Kinase P38 VS Virtual Screening EF Enrichment Factor ROC Receiver Operating Characteristic

Introduction

Launching a new molecule to the market requires tremendous effort in research, development and money investment. Recent studies estimate in 15 years and around 800 million dollars the average time and cost per approved molecule [1]. After more than 30 years of using combinatorial chemistry and high-throughput screening (the two techniques that researchers thought would be the solution to the drug discovery bottleneck), the ratio between the number of new drugs obtained and the funds invested in their generation is well below the initial expectations [2, 3]. In some sense this has fuelled the development of theoretical techniques that attempt to accelerate the initial steps in the drug design cycle. Theoretical methods, if correctly derived, allow pinpointing the more promising candidates (hits) out of pools of thousands or even millions of molecules (chemical libraries). This reduced set can then be subjected to experimental analysis, and any promising compound can be subsequently optimized to attain the desired pharmacological profile and become a lead.
From a theoretical perspective different scenarios can be envisaged depending on the structural data available \[4, 5\]. The most favourable one is when the structures of both the target and the ligand(s) are known. In this case docking and VS techniques are the methods of choice \[6\]. More elaborate approaches based on molecular dynamics \[7\] or free energy perturbation coupled to thermodynamic integration can also be used \[8\]. The problem here is that the amount of time required to perform the calculations becomes prohibitive when a large collection of molecules is used.

The goal of docking is to identify, among the large number of possible orientations of a ligand within the binding site of the target, the one closest to the experimental structure of the complex \[4\]. This is done by using a mathematical function that accounts for the goodness of the coupling between ligand and target. Consequently, two key elements of the docking problem are: (a) a good sampling method, and (b) an accurate scoring function. VS is the extrapolation of docking to the case in which a large database of molecules is going to be processed \[9\]. Here the primary goal is somewhat different from that in docking because promising hit candidates are picked out from a database of mostly non-binders and no attempt is made to correctly classify all the molecules in the library or to identify all the actives.

Generating the experimental pose for a ligand is feasible with today’s sampling techniques. However, positioning this pose at the top of a prioritized list of candidates is more problematic \[10\]. The main reason for this limitation is that the physical effects that describe the binding process are incompletely represented in the scoring function despite the fact that the underlying principles are reasonably well understood. In particular, the influence of the solvent, entropic effects and target flexibility are difficult to implement without compromising speed. Although many advances are being made to overcome these deficiencies, we are still at an early developmental stage.

Besides the problems outlined above, filtering millions of molecules using molecular descriptors and properties, docking them with one or more programs endowed with different accuracies, dealing with complex effects such as desolvation and/or protein flexibility at post-docking stages, etc. generates huge amounts of data that cannot be easily stored or utilized. It would then seem that the introduction of relational databases and potent database managing tools could be of aid in this regard.

Here we propose a flexible, fully automated computational platform to perform VS experiments that combines all the necessary steps to generate a short list of candidates starting from a database of 2D molecular structures. VSDMIP is intended to fill an existing gap in the docking and VS fields in relation to the storage and handling of the data. We are committed to making this platform available to interested parties so that the scientific community at large can benefit from it.

**Methods**

VSDMIP software and architecture

The VSDMIP architecture (Fig. 1) consists of (1) a database (in a multithreaded multi-user Structured Query Language [MySQL] DataBase Manager System), (2) a library of service interfaces and plugins, and (3) a set of workflows and its implementing commands. All small molecule data and VS results are stored in the VSDMIP database. The user controls the platform through different command line utilities and configures it using XML files. VSDMIP currently runs on Linux/x86 platforms (in our...
case, a cluster of ten 2.4 and thirty-two 3.0 GHz Xeon Biprocessor CPUs with 2 GB RAM each).

Relational database

The database (Fig. 2) contains tables for the compound libraries (ORIGINDB, MOLECULE, CONFIGURATION, CONFORMER), results from filtering experiments (DOCK_FILTER, RESULT_DF), and results from VS experiments (VIRTUAL_SCREENING, RESULT_VS). Each compound library has its origin entered in an ORIGINDB entry. Each molecule has a MOLECULE entry with its global charge and a SMILES string representing its topology. The different stereoisomers of each molecule in combination with the different ring conformations are stored, together with a 3D structure, in a CONFIGURATION entry; discrete conformational changes on each of these entries are stored in the CONFORMER table, with a score resembling its internal energy (VDW_ENERGY). Filtering experiments are identified by a DOCK_FILTER entry. The best docking pose for each of the selected conformers included in the experiment is stored in a RESULT_DF entry, together with its score. These poses are stored as a translation and rotation of the conformer with respect to the original ligand coordinates stored in the database. More accurate docking experiments are identified by a VIRTUAL_SCREENING entry, and each individual conformer solution has an entry in RESULT_VS, with the same translation and rotation information as explained for RESULT_DF. Besides, the docking score can be stored with the details of its interaction energy terms.

VSDMIP software library

This C/C++ library manages the database, interconnects the different applications used, and offers biochemical and geometrical utilities. For each external application it is possible to add functionalities to the platform by creating an interface class and a storage class, under a common framework. The interface class provides methods for managing the application configuration attributes, preparing the execution of the application, performing the execution, managing errors and storing the results in the database. Six service interfaces currently exist: 3D structure generation, conformational analysis, atomic charge calculation, filtering, VS, and electrostatic calculations. Several plugins have been developed to interface with CORINA 3.0.5 [11] (3D generation), ALFA [12] (conformational analysis), MOPAC 7 [13] (atomic charges), DOCK 3.5 [14] and FRED 2.2 [15] (filter interface), CDOCKER [16] and Autodock 3.0.5 [17] (VS interface), and for DelPhi 4 [18] and ISM [19] (electrostatic calculations).

Process workflows

The currently included commands allow inserting molecules into the database, docking a set of molecules
within a protein binding site (for filtering or VS), rescoring the results of a docking experiment, and retrieving the results from a filtering or docking step. Each command uses one or more of the services plugged into VSDMIP and task execution is allowed to take place in a computer cluster.

Validation tests

Eleven different VS protocols (VSP) were designed and applied to nine different ligand datasets involving six protein targets. The protocols differ in several aspects: (i) the use of a filter and the number of molecules and conformers allowed to pass the filter, (ii) the docking engine used, and (iii) the scoring function(s) employed for ranking (Table 1). The active compounds for the validation tests (Table 2) were obtained from the binary (active/inactive) datasets available from CHEMINFORMATICS.ORG. Inactive compounds were randomly selected from a 9862 subset of the Maybridge Hit-Finder collection. Most of these molecules follow Lipinski’s rule of 5 [20]. Table 3 shows the general properties of the database. The results were evaluated using receiver operating characteristic (ROC) plots [21], which represent the sensitivity (y-axis, true positives rate, see Eq. 1) versus (1—specificity) (x-axis, false positives rate, see Eq. 2), as well as areas under the ROC curves (AUC), enrichment factors (EF), and computing time.

Table 2 Relevant information for each of the datasets used in the virtual screening experiments

<table>
<thead>
<tr>
<th>VS experiment</th>
<th># of actives</th>
<th># of inactives</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>fXa (Fontaine)</td>
<td>432</td>
<td>500</td>
<td>932</td>
</tr>
<tr>
<td>fXa (Jacobsson)</td>
<td>127</td>
<td>500</td>
<td>627</td>
</tr>
<tr>
<td>fXa (Jorissen-Gilson)</td>
<td>50</td>
<td>500</td>
<td>550</td>
</tr>
<tr>
<td>AChE (Jacobsson)</td>
<td>54</td>
<td>1000</td>
<td>1054</td>
</tr>
<tr>
<td>CDK2 (Jorissen-Gilson)</td>
<td>50</td>
<td>1000</td>
<td>1050</td>
</tr>
<tr>
<td>ERa (Jacobsson)</td>
<td>142</td>
<td>1000</td>
<td>1142</td>
</tr>
<tr>
<td>ERa (Stahl)</td>
<td>50</td>
<td>1000</td>
<td>1050</td>
</tr>
<tr>
<td>Neuraminidase (Stahl)</td>
<td>17</td>
<td>1000</td>
<td>1017</td>
</tr>
<tr>
<td>p38MAP (Stahl)</td>
<td>22</td>
<td>1000</td>
<td>1022</td>
</tr>
</tbody>
</table>

Sensitivity = \frac{TruePositives}{True Positives + False Negatives} \quad (1)

Specificity = \frac{TrueNegatives}{True Negatives + False Positives} \quad (2)

EF represents the ratio of active compounds detected in a fixed percentage of the scanned database to the total number of actives in the whole database (Eq. 3).

Table 1 Virtual screening protocols used in the validation of VSDMIP

<table>
<thead>
<tr>
<th>VSP id.</th>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DOCK_100_1_CS</td>
<td>DOCK is performed on the 100 lower-energy conformers per molecule. The best conformer is selected using contact scoring.</td>
</tr>
<tr>
<td>2</td>
<td>DOCK_100_1_FF</td>
<td>DOCK is performed on the 100 lower-energy conformers per molecule. The best conformer per molecule is selected using force-field scoring.</td>
</tr>
<tr>
<td>3</td>
<td>DOCK_100_1_CS-XS</td>
<td>The same as in #1 but a final re-scoring is performed with XSCORE.</td>
</tr>
<tr>
<td>4</td>
<td>CDOCK_50_VDW_COUl</td>
<td>CDOCK is performed on the 50 lower-energy conformers per molecule. The best conformer per molecule is selected using the sum of van der Waals and coulombic interaction energies.</td>
</tr>
<tr>
<td>5</td>
<td>CDOCK_50_VDW</td>
<td>The same as in #1 but the final rank is done only with van der Waals interaction energies</td>
</tr>
<tr>
<td>6</td>
<td>CDOCK_50_ALL500</td>
<td>#4 is followed, and then for the 500 top-ranking molecules the solvation correction term is calculated using DelPhi. The final score is obtained by summing up the van der Waals and desolvation energies.</td>
</tr>
<tr>
<td>7</td>
<td>DOCK_100_1_CS + CDOCK_ZS3.0</td>
<td>#1 is followed, and then those compounds with ZScore ≥ 3.0 are submitted to CDOCK.</td>
</tr>
<tr>
<td>8</td>
<td>DOCK_100_10_CS + CDOCK_ZS3.0</td>
<td>The same as in #7 but now the best 10 conformer for each molecule are passed on to CDOCK.</td>
</tr>
<tr>
<td>9</td>
<td>DOCK_100_1_CS + CDOCK_ZS1.5</td>
<td>The same as in #7 but with ZScore ≥ 1.5.</td>
</tr>
<tr>
<td>10</td>
<td>DOCK_100_10_CS + CDOCK_ZS1.5</td>
<td>The same as in #8 but with ZScore ≥ 1.5.</td>
</tr>
<tr>
<td>11</td>
<td>DOCK_100_1_CS + CDOCK_ZS3.0_ALL500</td>
<td>#7 is followed, and then solvation correction using DelPhi is calculated for the first 500 molecules. The final score is obtained by summing up the van der Waals and desolvation energies.</td>
</tr>
</tbody>
</table>
Table 3 Mean and standard deviation of Lipinsky’s properties for the Maybridge database ligands extended with the number of rotatable bonds

<table>
<thead>
<tr>
<th></th>
<th>MW</th>
<th>HBA</th>
<th>HBD</th>
<th>RB</th>
<th>logP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>311</td>
<td>4.82</td>
<td>1.12</td>
<td>4.58</td>
<td>2.65</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>75.7</td>
<td>1.87</td>
<td>1.06</td>
<td>2.39</td>
<td>1.65</td>
</tr>
</tbody>
</table>

As calculated with OpenEye’s Filter 2.0. The rules are (a) MOL_WT [0, 500], ROT_BONDS [0, 20], LIPINSKI_DONORS [0, 5], LIPINSK_ACCEPTORS [0, 10] and XLOGP [-5.0, 5.0]. 743 molecules failed Lipinski’s test.

\[ EF = \frac{\{N_{\text{Actotal}}/NT_{\text{subset}}\}}{\{N_{\text{Actotall}}/NT\}} \]

where subset is the fixed percentage of the database, \( N_{\text{Actbo}} \) is the number of active molecules found in the subset, \( N\text{T}_{\text{subset}} \) is the total number of molecules in the subset, \( N\text{T}_{\text{total}} \) is the total number of active molecules in the entire database, and NT is the total number of molecules in the database. For comparative purposes we have computed the best EF found (EFmax), the maximum EF possible for each experiment, and the percentage of the database at which EFmax is obtained. To calculate EFmax, only subsets with an NTsubset multiple of b are considered, where b is max{10, 0.01 × NT}. This avoids artefacts in EFmax at the start of database scanning.

Setup of small molecule databases

All molecules were first converted to isomeric Simplified Molecular Input Line Entry Specification (SMILES) [22] format, using the OpenEye OEChem 1.4 library. A 10,000 subset of the Maybridge HitFinder database was obtained in 2D Structure Data Format (SDF) [23]. The stereo information was not considered, and only those molecules bearing 3 or less stereogenic centres were selected. An additional set of 12 molecules had improbable connectivities and were also discarded by the VSDMIP insertion application, leaving a total of 9,862 molecules. The active datasets were prepared from SDF or SMILES input files. Duplicates (most likely arising from different stereoisomers devoid of the stereo information) were removed. When known, stereo and protonation information was preserved. Otherwise, protonation was assigned following OpenEye’s Filter 2.0 rules for pH 7.4. Changes in bond order and number of hydrogens were done in some molecules to obtain standard valences. Using the final SMILES strings, the totally automatic process, insertVS DB, inserted the molecules into the database after carrying out the following steps: (i) conversion from SMILES to 3D MOL2 using CORINA: up to 6 stereogenic centres were considered, ring conformations were generated, hydrogen atoms were added, and salt ions were removed; (ii) atomic charge calculations with MOPAC: single point calculations were performed with the MNDO semiempirical method [24] obtaining atom centred charges via electrostatic potential fitting techniques on each single structure provided from CORINA (one or more for each SMILE string depending on the number of stereogenic centres and ring conformations); (iii) atom type assignment and conformational analysis using ALFA.

Protein set up

All protein structures were obtained from the PDB and correspond to X-ray experiments with resolutions of 2.6 Å or better. The structures were chosen based on previous published VS and docking experiments: 1f0r (chain A) for coagulation factor Xa (fXa) [25, 26]; Ieve for acetylcholinesterase (ACHE) [27, 28], 1e1x for cyclin-dependent kinase 2 (CDK2) [29, 30], 3ert for estrogen receptor alpha (Era) [31, 32], 1nsc (chain B) for neuraminidase [33, 34], and 1p38 for p38 MAP kinase [35, 36]. All HETATM records, including water molecules and ions, were removed from the PDB files. Side chains with missing atoms were rebuilt using SCWRL3 software [37]. In 1e1x, MODEL-LE r v6.2 [38] was employed to reconstruct a missing loop. Then, the AMBER 8 [39] ff99 force field [40] was used to assign atom types and partial charges to each atom in the proteins, and hydrogen atoms were added assuming standard protonation states of titratable groups.

The H++ web server [41] was employed to study and assign protonation states for key interacting residues in the binding site (see below). The Poisson-Boltzmann (PB) method [42, 43] was used, at pH 7.4, 0.15 M salt concentration, and using internal and external dielectric constants of 4 and 80, respectively. Based on the information provided by H++, the following residues were protonated: HIS57 and ASP189 in 1f0r; HIS440, GLU278, and GLU443 in Ieve; HIS125 in CDK2; and ASH323 in 1nsc. The modified proteins were subjected to 10,000 steps of energy minimization (500 initial steps of steepest descent, followed by conjugate gradient), in vacuum, and only the hydrogen atoms were allowed to move. A further 10,000 steps of energy minimization were done with a Generalized-Born (GB) implicit solvent model [44–46] during which all atoms were allowed to move but heavy atoms were positionally restrained with a harmonic potential (100 kcal/mol Å²). To check for consistency, the optimized structures were submitted again to the H++ web server. No significant changes were observed. The final
minimized structures differed less than 0.05 Å (for Cz atoms) compared to the initial X-ray structure.

Binding site definition and characterization

For each protein structure, the initial binding site was defined as the space delimited by the axis-parallel box containing the co-crystallized ligand, augmented by 5 Å in each axis direction. Structure 1p38 was structurally aligned to PDB 1ywtr [47], and the co-crystallized ligand found in 1ywtr was used to define the binding site in the p38MAP studies. Protein interaction grids covering the binding site (1.0 Å spacing in all directions) were calculated for atom probes C, N, O, S, P, H, F, Cl, Br, and I. Each grid point represents the interaction between the protein and the probe atom as the sum of a van der Waals Lennard-Jones 12–6 potential and an electrostatic term modelled with a sigmoidal dielectric screening function [48]. For each binding site, a set of about 20 interaction points were defined to guide docking studies. These points were selected from the GAGA [49] centers of the gaussian sphere functions that best captured the interaction maps between the protein and benzene, water, and methanol molecules, as calculated by docking experiments using CDOCK. The set of points thus selected represent hydrophobic, hydrophilic and hydrogen bonding interactions. For docking using DOCK, the binding site was defined as the space delimited by the axis-parallel box containing the selected interaction points, augmented by 7.5 Å in each axis direction. The interaction grids used by DOCK and covering the binding site had a spacing of 0.3 Å. DOCK grids (see DOCK documentation) represent electrostatic, van der Waals and contact scores. For docking studies using CDOCK, the binding site was the same as that defined for DOCK, adding an extra 2.5 Å (for a total of 10 Å) in each axis direction. CDOCK grids had a spacing of 0.5 Å, and considered both protein interactions with different atom probes and electrostatic interactions, as explained above.

Filter plugin

For the tests shown in this article, DOCK 3.5 was used as a fast initial filter of the chemical libraries to be screened. DOCK uses a sphere-matching algorithm to fit ligand atoms to spheres in the binding pocket. We defined such points using GAGA as explained before. We chose to evaluate docking poses using the DOCK contact score in most cases and the DOCK force-field score in one case (VSP 2). For each conformer the single best DOCK solution obtained was retained. For VSP 1, 2, 3, 7, 9, and 11 DOCK scores for the best conformer of each molecule were stored; for VSP 8 and 10 the scores for the 10 best conformers of each molecule were kept. For VSP 3 the best solution per molecule was rescored using XSCORE [50] v. 1.2.1. but no improved performance was found and therefore, in the following, no further reference to this program will be made. Stored scores were normalized to ZScore (Eq. 4),

\[
ZScore_i = \frac{\text{score}_i - \text{score}}{\sigma}
\]

where \(\text{score}\) is the mean and \(\sigma\) is the standard deviation values for the scores.

The VSDMIP application runDOCKFilter extracts information for all selected molecules from the database, performs the docking and stores the results in the database. Then getResultsFromDOCK extracts the docking results in a single coordinate file in MOL2 format for all the conformers with a ZScore higher than that provided by the user.

Docking plugin

We have used our in-house program CDOCK for the detailed docking phase of our protocols. Using the interaction energy grids calculated with CGRID, CDOCK exhaustively docks each molecule within the binding site. The centres of mass of the molecules are positioned on grid points equally spaced 1 Å, and discrete rotations of 27° on each axis are performed. A molecular mechanics force-field scoring function is used to score each pose (Eq. 5),

\[
E_{\text{MM}} = \sum_i \sum_j \left[ \frac{A_{ij}}{r_{ij}^6} - \frac{B_{ij}}{r_{ij}^6} + q_i q_j \frac{D(r_{ij})}{r_{ij}} \right]
\]

where \(A_{ij}\) and \(B_{ij}\) are the van der Waals parameters of the atom types to which atom \(i\) and \(j\) belong, \(r_{ij}\) is the distance between the \(i\)th atom in the protein and the \(j\)th from the ligand, \(q_i\) and \(q_j\) are the partial charges of atom \(i\) and \(j\), respectively. \(D(r_{ij})\) is a sigmoidal dielectric function that accounts for solvent screening (Eq. 6),

\[
D(r_{ij}) = \frac{\varepsilon + 1}{1 + k e^{-\lambda (\varepsilon + 1)r_{ij}}}
\]

where \(\varepsilon\) is the dielectric constant for water (78.39), \(k\) is \((\varepsilon - 1)/2\), \(\lambda\) is \(\alpha/(\varepsilon + 1)\) and \(\alpha = 1.0367\). The docking score for each pose (van der Waals plus electrostatic) is then calculated using a trilinear interpolation method.

The VSDMIP application runCDOCK extracts information for all selected molecules from the database, performs the docking and stores the results in the database. Then getResultsFromVS extracts the docking results in a single coordinate file in MOL2 format [51] containing the number of molecules defined by the user, and creates a text file with the score for each result.
CDOCK scores were corrected with solvation energies (Eq. 7) obtained by solving the Poisson equation (electrostatic part, hereafter PB).

$$
\Delta G_{\text{desolv}} = \Delta G_{\text{ele}} + \Delta G_{\text{np}}
$$

(Eq. 7)

The electrostatic part of the energy (Eq. 8) is the sum of the electrostatic interactions between the ligand and protein in the complex ($E^L_{\text{ele}}$), the change in solvation energy of the ligand upon binding ($\Delta G^L_{\text{desolv}}$) and the change in solvation energy of the receptor upon binding ($\Delta G^R_{\text{desolv}}$).

$$
\Delta G_{\text{ele}} = E^L_{\text{ele}} + (\Delta G^L_{\text{desolv}} + \Delta G^R_{\text{desolv}})
$$

(Eq. 8)

The first term in Eq. 8 was computed as the product of ligand charges times the electrostatic potential created by the protein at each charge. The ligand desolvation energy was computed as the difference in energy between the solvated ligand and the energy of the ligand complexed to the uncharged receptor. The receptor desolvation term was computed analogously. All calculations were performed by numerically solving the linear Poisson equation using the finite difference method as implemented in DelPhi, PARSE atomic radii [52], AMBER ff99 partial charges for protein atoms, and MOPAC-calculated charges for ligand atoms, as described above. Each complex was immersed in a cubic box occupying 65% of the total volume 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. The dielectric boundary was calculated using a solvent probe radius of 1.4 Å and a minimum separation of 11 Å was allowed between any solute atom and the box walls. The potentials at the grid points delimiting the box were calculated analytically by treating each charge atom as a Debye-Hückel sphere. The non-polar part of the desolvation (Eq. 9) was modelled as a linear relationship to the change of solvent-accessible surface area (SASA),

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

(Eq. 9)

where $a$ is 0.092 kcal/mol, $b$ is 0.00542 kcal/molÅ$^2$, and the change in SASA refers to the complex SASA minus the sum of that of the protein and the ligand alone. SASAs were calculated using the analytical method implemented in TINKER [53].

The VSDMIP application runDelPhiAndApolair extracts information for all selected molecules from the database, performs the calculation and stores the results as new entries into the database.

Visualization

Visualization of results is accomplished using the well-documented, free, open-source program Pymol [54].

Results

VSDMIP architecture and relational database

The architecture of VSDMIP is depicted in Fig. 1, and has been commented on, to some extent, in the Methods section. The main role played by the VSDMIP database is to act as a common origin and destination for all stages of the VS process. The XML configuration files and the different plug-in interfaces allow the user to configure different custom-made VS protocols, as exemplified below.

The Entity-Relationship schema of the database used by VSDMIP is shown in Fig. 2. The table primary keys are depicted with a key symbol, and foreign keys are marked with FK in brackets. The schema shows a compact way of storing and organizing chemical libraries and VS results through the MOLECULE, CONFIGURATION, and CONFORMER tables, and their relationships with RESULT_DF and RESULT_VS. Many different VS protocols can be composed by using as many DOCK_ FILTERers (and their related RESULT_DF) and VIRTUAL_SCREENINGS (and their related RESULT_VS) as desired, and allow the results to be easily reused. Only the results from the last step are needed to run the following step in the workflow.

Performance of the individual docking tools (Tables 4-6)

DOCK performance is really poor for all but one system (ERa Jacobsson set), where it is marginally better than random selection. The results obtained using the force-field scoring function (VSP 2) are always better than those obtained with the contact scoring function (VSP 1), but the difference is not significant, the greater being around 0.3 U of AUC. This is also the range of variation in AUC values for different proteins with both scoring schemes. The EFs are very low and far from EF max in all the cases and no significant differences are found when the force-field scoring function is used instead of the contact one. The best EFs are obtained for the ERa Jacobsson set (EF$\text{best} = 3.35$). In most cases, the CDOCK AUCs are above 0.6, with the exception of the ERa receptor, for which results are below random (Jacobsson set) and slightly above random (Stahl set). In all the cases, except for the three fXa test sets, compound selection based only on van der Waals interaction energies (VSP 5) leads to smaller AUC values but this effect is not important. A noteworthy exception is the neuraminidase example where the electrostatic energy term improves the AUC by 0.47. Variations in AUC values across protein systems are significant (around 0.6 U) but independent from the scoring function (van der Waals plus electrostatic or van der Waals alone). For complete
CDOCKER scoring (van der Waals plus electrostatic, VSP 4) 
EF are always better than when using DOCK (with both 
scoring functions, VSP 1 and 2) and above 10% of EFmax, 
reaching this value in one case (fXa Fontaine set), 80% 
(AChE), and 55% (ERa, Sthal set). If only the van der 
Waals term is used (VSP 5), the EF is greatly reduced, and 
this decrease is most dramatic in the AChE (30%), ERa 
(Sthal set, 20%), and neuraminidase (16%) examples.

Performance of mixed protocols: DOCK as a filter 
for CDOCKER (Tables 4–6)

(a) Considering different ZScores. Filtering using DOCK, 
selecting molecules with a ZScore above 3.0, and then 
employing CDOCKER (VSP 7) yields AUC values that 
are very close to those obtained from a random screen. 
No appreciable differences are observed when a more 
restrictive ZScore is used (1.5 vs. 3.0, VSP 9). The two 
exceptions in this case are AChE and neuraminidase. 
In the former, from an almost random performance 
(0.49) to a respectable value of 0.71, a difference of 
0.22 units of AUC, while in the latter the difference is 
0.14, thus reaching an AUC value close to 0.70. The 
differences across target proteins are of the same order 
(0.2 for VSP 7 and 0.3 for VSP 9). In only two cases 
the EF is above 50% of EFmax (ERa Sthal set 64% and 
fXa Fontaine set 70%) while the rest range between 
4% and 35%. Small variations in EFs are observed for 
most of the sets when the lower ZScore value is used. 
These variations are more important in AChE (18% 
increase) and ERa (Sthal set, 28% decrease).

(b) Selecting 10 (instead of 1) conformations for each 
ligand from DOCK. When the ZScore is 3.0 (VSP 8) 
the AUC values are always above 0.5, the exception 
being the ERa example (Jacobsson set). Variations 
across targets are of the order of 0.3 AUC units. For a 
ZScore value of 1.5 (ID VSP 10) better results were 
obtained. In general, values for AUC above 0.6 are 
common, except for ERa (Jacobsson set) once again. 
Here, variations across the targets are as high as 0.5 
AUC units. Three cases present EFs values above 
50% of EFmax while the others range between 7% and 
27%, when ZScore is 3.0. With the lower value, i.e. 
1.5, the major differences are observed in AChE (18% 
increase), neuraminidase (14% increase), and ERa 
(Sthal set, 32% decrease).

Including solvent effect: PBSA as a correction term 
(Table 4–6)

Molecular mechanics interaction energies (CDOCKER scor-
ing function) are corrected for desolvation effects using the 
PBSA method [55] on results directly obtained from 
CDOCKER (VSP 6) or after a combined protocol encom-
passing DOCK and CDOCKER (VSP 11). In the first case, the 
AUCs obtained are somewhat above random in many of 
the tests. The best AUC are for AChE and neuraminidase 
targets, while ERa (Jacobsson test) is the only case with 
AUC below random. In this case, the introduction of sol-
vent effects via PBSA does not lead to an improvement 
over plain CDOCKER results. Five sets show EFs above 30%, 
with fXa (Fontaine set) achieving EFmax. Except for AChE, 
the inclusion of desolvation always reduces the EF values. 
The reduction range goes from negligible (ERa, Jacobsson 
set, 2% or p38, 5%) to notorious (fXa Fontaine set, 30% or 
ERa Sthal set, 28%). In the second case, the AUC values

Table 4 Area Under the Curve (AUC), EFbest, EFmax for the three fXa sets obtained for each VSP used

<table>
<thead>
<tr>
<th>VSP id.</th>
<th>fXa Fontaine set</th>
<th></th>
<th></th>
<th></th>
<th>fXa Jacobsson set</th>
<th></th>
<th></th>
<th></th>
<th>fXa Jorissen-Gilson set</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC</td>
<td>EFbest (EFmax = 2,16)</td>
<td>log t</td>
<td>AUC</td>
<td>EFbest (EFmax = 4,94)</td>
<td>log t</td>
<td>AUC</td>
<td>EFbest (EFmax = 11)</td>
<td>log t</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.39</td>
<td>1.00 (0.99)</td>
<td>1.30</td>
<td>0.41</td>
<td>1.00 (0.99)</td>
<td>0.99</td>
<td>0.36</td>
<td>1.00 (1.00)</td>
<td>0.93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.37</td>
<td>1.00 (1.00)</td>
<td>1.32</td>
<td>0.41</td>
<td>1.01 (0.96)</td>
<td>0.91</td>
<td>0.39</td>
<td>1.02 (0.98)</td>
<td>0.93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.33</td>
<td>1.00 (1.00)</td>
<td>1.34</td>
<td>0.33</td>
<td>1.00 (1.00)</td>
<td>1.05</td>
<td>0.31</td>
<td>1.00 (1.00)</td>
<td>0.99</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.67</td>
<td>2.16 (0.01)</td>
<td>3.62</td>
<td>0.61</td>
<td>1.65 (0.05)</td>
<td>3.42</td>
<td>0.68</td>
<td>3.30 (0.04)</td>
<td>3.38</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.70</td>
<td>1.94 (0.01)</td>
<td>3.62</td>
<td>0.70</td>
<td>1.97 (0.05)</td>
<td>3.42</td>
<td>0.68</td>
<td>3.30 (0.02)</td>
<td>3.38</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.57</td>
<td>1.51 (0.14)</td>
<td>3.72</td>
<td>0.52</td>
<td>1.23 (0.22)</td>
<td>3.55</td>
<td>0.60</td>
<td>1.80 (0.20)</td>
<td>3.51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.54</td>
<td>1.51 (0.01)</td>
<td>1.43</td>
<td>0.55</td>
<td>1.73 (0.06)</td>
<td>1.28</td>
<td>0.58</td>
<td>3.30 (0.02)</td>
<td>1.26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.59</td>
<td>1.44 (0.03)</td>
<td>2.19</td>
<td>0.55</td>
<td>1.32 (0.05)</td>
<td>NA</td>
<td>0.65</td>
<td>2.57 (0.05)</td>
<td>1.99</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.58</td>
<td>1.58 (0.03)</td>
<td>1.76</td>
<td>0.56</td>
<td>1.38 (0.32)</td>
<td>1.53</td>
<td>0.61</td>
<td>2.93 (0.05)</td>
<td>1.46</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.64</td>
<td>1.43 (0.09)</td>
<td>2.58</td>
<td>0.57</td>
<td>1.29 (0.33)</td>
<td>2.37</td>
<td>0.65</td>
<td>2.48 (0.07)</td>
<td>2.31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.54</td>
<td>1.25 (0.16)</td>
<td>2.51</td>
<td>0.54</td>
<td>1.48 (0.02)</td>
<td>2.29</td>
<td>0.58</td>
<td>1.83 (0.05)</td>
<td>2.29</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Values in brackets refer to the percent (in decimal form) of the database scanned at which EFbest is found
are almost unaltered compared with the situation in which no solvent effects are introduced. Again, performance on the ERa target (Jacobsson test) was worse than random. In terms of variations across the targets, 0.46 AUC units are found when PBSA is applied to CDOCK without previous DOCK filter and 0.21 with the DOCK filter. The EFs here are similar to those commented on before, but only four sets are above 30% (instead of five) and none reaches EFmax. Again, a general reduction in these values is observed when desolvation is taken into account, the range being between 2% (p38) and 18% (AChE).

Finally, and related to the percentage of the database at which the best EF is found, almost the entire database has to be explored when DOCK is used. There are, however, some exceptions (6 out of 99): VSP 2 for ERa (Jacobsson and Stahl sets), CDK2, neuraminidase and p38MAP, and VSP 1 for ERa (Jacobsson test). For the rest of the protocols, the best EF is achieved very early for most of the cases, except in those for which over 25% of the database has to be screened: ERa (Jacobsson set, VSP 9 and 10), neuraminidase (VSP 5), and fXa (Jacobsson set, VSP 9 and 10).

**Discussion**

A plethora of methods are available to propose new drug candidates starting simply from 2D sketches of virtual chemical libraries. It should then be possible to integrate all

<table>
<thead>
<tr>
<th>VSP id.</th>
<th>AChE</th>
<th>ERa</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Jacobsson set</strong></td>
<td><strong>Stahl set</strong></td>
</tr>
<tr>
<td>AUC</td>
<td>EFbest(EFmax = 19.52)</td>
<td>log t</td>
</tr>
<tr>
<td>1</td>
<td>0.30</td>
<td>1.08 (0.86)</td>
</tr>
<tr>
<td>2</td>
<td>0.32</td>
<td>1.02 (0.98)</td>
</tr>
<tr>
<td>3</td>
<td>0.26</td>
<td>1.02 (0.98)</td>
</tr>
<tr>
<td>4</td>
<td>0.97</td>
<td>15.97 (0.01)</td>
</tr>
<tr>
<td>5</td>
<td>0.94</td>
<td>9.58 (0.05)</td>
</tr>
<tr>
<td>6</td>
<td>0.93</td>
<td>15.97 (0.01)</td>
</tr>
<tr>
<td>7</td>
<td>0.49</td>
<td>5.32 (0.01)</td>
</tr>
<tr>
<td>8</td>
<td>0.69</td>
<td>12.42 (0.01)</td>
</tr>
<tr>
<td>9</td>
<td>0.71</td>
<td>14.20 (0.01)</td>
</tr>
<tr>
<td>10</td>
<td>0.93</td>
<td>15.97 (0.01)</td>
</tr>
<tr>
<td>11</td>
<td>0.49</td>
<td>1.77 (0.01)</td>
</tr>
</tbody>
</table>

Values in brackets refer to the percent (in decimal form) of the database scanned at which EFbest is found.

<table>
<thead>
<tr>
<th>VSP id.</th>
<th>CDK2</th>
<th>Neuraminidase</th>
<th>p38MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Jorissen-Gilson set</strong></td>
<td><strong>Stahl set</strong></td>
<td><strong>Stahl set</strong></td>
</tr>
<tr>
<td>AUC</td>
<td>EFbest(EFmax = 21)</td>
<td>log t</td>
<td>AUC</td>
</tr>
<tr>
<td>1</td>
<td>0.36</td>
<td>1.00 (1.00)</td>
<td>1.20</td>
</tr>
<tr>
<td>2</td>
<td>0.41</td>
<td>1.91 (0.01)</td>
<td>1.18</td>
</tr>
<tr>
<td>3</td>
<td>0.41</td>
<td>1.02 (0.98)</td>
<td>1.27</td>
</tr>
<tr>
<td>4</td>
<td>0.67</td>
<td>2.45 (0.07)</td>
<td>3.02</td>
</tr>
<tr>
<td>5</td>
<td>0.63</td>
<td>2.12 (0.09)</td>
<td>3.02</td>
</tr>
<tr>
<td>6</td>
<td>0.57</td>
<td>2.06 (0.14)</td>
<td>3.33</td>
</tr>
<tr>
<td>7</td>
<td>0.55</td>
<td>1.91 (0.13)</td>
<td>1.28</td>
</tr>
<tr>
<td>8</td>
<td>0.63</td>
<td>3.82 (0.01)</td>
<td>1.67</td>
</tr>
<tr>
<td>9</td>
<td>0.59</td>
<td>2.67 (0.05)</td>
<td>1.40</td>
</tr>
<tr>
<td>10</td>
<td>0.61</td>
<td>2.10 (0.10)</td>
<td>1.95</td>
</tr>
<tr>
<td>11</td>
<td>0.55</td>
<td>1.91 (0.02)</td>
<td>2.54</td>
</tr>
</tbody>
</table>

Values in brackets refer to the percent (in decimal form) of the database scanned at which EFbest is found.
the needed software elements to create customized workflows for any desired project. But the data flow between the different steps (input/output connections) seems to be an important problem mainly due to the great variety of formats that can be used to describe molecular structures. Although some advances have been done (e.g., SMILES and InChI [IUPAC International Chemical Identifier]), a consensus format is far from having been adopted. On the other hand, life sciences, in general, and computer-aided drug design, in particular, witness a data deluge coming from the target side, i.e., 3D protein structures available from structural genomics projects, as well as from the ligand side, i.e., huge chemical libraries with millions of molecules to be screened. Finally, the amount of data to be processed, stored and managed requires potent database engines. These three aspects have motivated us to develop VSDMIP as an integrative platform for handling these data. The main advantages of VSDMIP are: (1) the possibility to perform automated VS experiments; (2) easy comparison of different protocols; (3) total flexibility to design VS protocols; (4) the implementation of an XML mechanism to plug in new software pieces to customize protocols at will; and (5) the generation of a coupled relational database to have all the data organized and ready to use. VSDMIP presently lacks a graphical user interface (GUI), but it may be added in the future. When this is done, it will be able to additionally provide information regarding both the receptor and the ligand-binding site interactions. Small changes in the database schema will make it possible to store the results from docking engines that generate new ligand conformations as solutions, thus extending the current capability that basically works with docking engines that rely on pre-computed sets of conformers.

As similar approaches have been published recently, a brief discussion of some of them in comparison with VSDMIP is in order. Probably the most similar approach is that reported by SciTegic, Inc. named Pipeline Pilot [56, 57]. This commercial software uses the technology known as data pipeline to construct and execute customizable workflows using components that encapsulate mainly cheminformatics-based algorithms (although docking can also be performed using programs GOLD and FLEXX). Hassan et al. have shown the usefulness of this platform in a recent review [57], where they also show results from virtual screening experiments using Bayesian learning technique on several targets. It has an underlying database in common with VSDMIP, while performing many different types of calculations and methods.

Along the the same line, Astex Therapeutics [58] reported a proprietary web-based platform that integrates an ORACLE relational database to store molecules (from the Astex Technology Library of Available Substances [ATLAS] database), properties, target data, binding interactions, and results. VS experiments can be performed using compounds obtained directly from the database or created virtually thorough stored chemical reactions. Molecules in SMILES 2D strings are converted into 3D structures using CORINA and are docked with GOLD. It also contains a GUI to set up the experiments and visualize the results. VSDMIP has some added advantages such as its modularity, the possibility for users to generate their own chemical libraries, and the fact that it will be available to academic users upon request. On the down side, to the present day VSDMIP works through a command line interface.

The idea of employing user-configurable XML files to perform customized drug design-related workflows (as we do in VSDMIP) has been implemented by Lehtovuori and Nyrönen [59]. In their SOMA approach, the user decides which protocol to use by selecting, from a web browser interface, the steps and adequate parameters (even though in the reported implementation only molecular structure-based properties calculation and/or docking experiments with GOLD can be performed). Then program Grape manages the workflow by joining the needed applications in the order established by the user through the execution nodes (to run and manage the applications). Finally, the same web browser interface is used to retrieve, visualize and analyze the results. The output produced by each executable node is encoded, labelled (to keep track of the process at any time), and updated with the output information generated by the subsequent steps (depending on the selected protocol). Another important component is the toolkit that contains utility programs to perform intermediate tasks (file format conversions and generation of execution files, among others). The entire SOMA protocol is encoded within a unique XML file, which means that, after the required input has been provided, the user does not need to interact with the system until the entire workflow has been completed. This is very convenient for already established protocols. On the contrary, VSDMIP is more focused on decision making after every step. This is so because if more than one docking program is going to be used, or different filters are available, it is necessary to stop at each step to check the results before carrying on. A fine tuning at this level is not easily tractable nor is it feasible to implement in an automatic protocol. Another aspect that deserves to be commented on in relation to VSDMIP is the lack of a database to manage the results. SOMA stores the results of the entire workflow in a single file that is displayed as a table within the interface. It is unclear whether the molecules and results already obtained could be used in another set of experiments. VSDMIP is totally flexible in this respect (i.e. once a molecule has been inserted into VSDMIP it can be reutilised as many times as desired). The availability of the SOMA XML schema and
More interesting, however, is the fact that when DOCK is used as a filter preceding CDOCKER (VSP7–VSP11) reasonable results are obtained, which attests to its suitability to remove undesirable ligand structures. On the other hand, as clearly shown in Fig. 4, the best results are obtained for VSP4, which is also the most time-consuming protocol. On the contrary, VSP10 displays similar results compared to VSP4 but the computer time increases between 1 and 2 orders of magnitude. In other words, the use of a filter saves computational time while retaining most of the AUC values. The representative ROC plots depicted in Fig. 5 illustrate the aforementioned effect.

Given the amount of molecules present in standard databases, it is not yet computationally feasible to conduct VS experiments using extremely accurate docking programs. Instead, a common approach is to use some sort of concatenated filters to reduce the number of molecules to be docked. In other studies, after initial filtering and docking, different scoring functions are employed and then candidates are selected on the basis of a consensus criterion [62]. More promising alternatives make use of more than one docking program in increasing order of accuracy [63, 64]. We chose DOCK as an initial docking program because it is fast enough to screen a molecule in a few seconds, the total time depending on the number of conformers and the number of spheres used to describe the
binding site. In order to compare the relative performance of DOCK when used alone or in conjunction with another more exhaustive docking tool, an initial study was done considering DOCK alone. The performance of our configuration for DOCK is not better than random either with a contact- or a force-field-based scoring function, as can be seen from the AUC and EF\text{best} values. We subjected our in-house docking program CDOCK to the same test and the results show that CDOCK clearly outperforms this particular DOCK configuration both in terms of AUC and EF\text{best} values. As a trade-off for this increased accuracy, however, CDOCK employs more computer time. Another observation is that the elimination of the coulombic component does not really produce any variation in AUC or EF\text{best} values, a result that is presently being scrutinized.

As stated before, a good approach consists of using a sequential combination of docking programs in increasing order of accuracy. Here we tested different protocols using first DOCK and then CDOCK. Variations among them relate to the number of compounds and conformations per compound to be passed from DOCK to CDOCK. In all cases, 100 conformers are used in DOCK. First, two cut-off values for the numbers of molecules passed to CDOCK are set using ZScore (3.0 and 1.5), DOCK contact scores and only one conformer per molecule. The AUC values do not seem to be dependent on ZScore, whereas the EF\text{best} values show some degree of variation depending on the type of target. This means that although most of the molecules that survive after applying a low ZScore value are in fact inactive, they can be recognized and discarded by CDOCK. In general, the combination of both docking programs shows an intermediate degree of performance, as could be expected, between DOCK alone and CDOCK alone.

Secondly, better results are obtained when instead of a single conformation 10 are passed on to CDOCK. This is a commonly observed effect, provided that conformational sampling has been performed adequately, and simply states that with more conformations per molecule the docking algorithm stands a better chance of detecting the correct pose for a binder [65]. Finally, use of the same ZScore but a different number of conformers per molecule (1 or 10) does not lead to any appreciable changes in AUC or EF\text{best} values, the variation in the latter case depending on the type of target.

The third point addressed here is related to the introduction of solvent effects via PBSA as a post-scoring function. Together with flexibility issues [66], an adequate representation of the desolvation process that accompanies ligand-receptor binding is a major hurdle in VS studies. It is also a problem in traditional docking but, due to the fact that a small number of molecules are going to be studied, more elaborated solvation methods can be applied in this case, even when time consumption for such a calculation is often seen as a shortcoming. Methods based on PB or GB approximations are common but become impractical at the large scales a VS experiment requires although some promising approaches have already been published [19, 67, 68]. We have observed a small influence of solvent effects on AUC values using PB after CDOCK and no effect at all when used after the combination of DOCK and CDOCK. Again, the influence on EF\text{best} is target-dependent, but in all cases a decrease in these values has been found.

VSDMIP has already proved successful in some recent scientific applications, such as one devoted to the discovery of new inhibitors of the DNA repair protein O6-alkylguanine DNA alkyltransferase [69]. Four compounds selected out of 3.5 million molecules from the ZINC database [70] showed acceptable in vivo and in vitro activities. In another example using in vivo screenings of a chemical combinatorial library and VSDMIP, we were able to develop small molecules that compete with ubiquitin E2 variant (UEV) for its interactions with ubiquitin-conjugating enzyme UBC13 and inhibit its enzymatic activity. The UEV–UBC13 complex is also implicated in mechanisms of DNA repair (unpublished results).

### Conclusion

An integrated computational platform to perform VS experiments has been developed that includes an associated relational database which stores (i) molecules and molecular properties (energies, conformations, charges, etc.), (ii) results from docking filters, and (iii) final VS results. This procedure allows the inserted molecules to be reused in as many VS experiments as desired as well as the continued...
incorporation of new molecules. Also, it is easy and fast to create and allows a battery of analyses to be performed in order to test a particular VS protocol. The modular idea underlying its design is one of the stronger points, as the user is able to replace existing modules with new ones to create customizable protocols. Finally, and under development, is the idea to include protein set-up in an automatic fashion within the database, allowing the storage of geometrical and energetic characteristics of the binding site, which should serve as a classification tool for binding sites. The platform has been prepared as a bundled package to be distributed to the scientific community upon request from the authors [71]. In brief, all the programs implemented in the platform (except those that need to be purchased, by a modest prize, such as CORINA or DelPhi) are either free (MOPAC, DOCK, FRED, AutoDock) or will be released under a scientific/academic non-profit and non-commercial license as is the case for ALFA, CGRID, CDOCK, and ISM. The scripts to create the database structure as well as the XML configuration files will be also provided.

Acknowledgements Work at the CBM-SO was partially supported by a grant from “Comunidad de Madrid” thorough BIPEDD project (SBIO-0214-2006) and from “Ministerio de Educación y Ciencia” (BIO2005-0576). J.S. and J.E. were funded by grants BFU2007–61476/BMC (MEC, Spain) and PM076/2006 (DGA, Spain). J.E.’s research stage at CBM “Severo Ochoa” was funded by grants DGA (CONSI–D)/CAI (Spain) and FPU (MEC, Spain). J.E. is recepient of a scientific/academic non-profit and non-commercial license as is the case for ALFA, CGRID, CDOCK, and ISM.

The authors [71]. In brief, all the programs implemented in the platform (except those that need to be purchased, by a modest prize, such as CORINA or DelPhi) are either free (MOPAC, DOCK, FRED, AutoDock) or will be released under a scientific/academic non-profit and non-commercial license as is the case for ALFA, CGRID, CDOCK, and ISM. The scripts to create the database structure as well as the XML configuration files will be also provided.

References