J Mol Model (2011) 17:201–208
DOI 10.1007/s00894-010-0684-x

SOFTWARE REPORT

Open3DQSAR: a new open-source software aimed at high-throughput chemometric analysis of molecular interaction fields

Paolo Tosco · Thomas Balle

Received: 25 December 2009 / Accepted: 1 February 2010 / Published online: 11 April 2010

Abstract Open3DQSAR is a freely available open-source program aimed at chemometric analysis of molecular interaction fields. MIFs can be imported from different sources (GRID, CoMFA/CoMSIA, quantum-mechanical electrostatic potential or electron density grids) or generated by Open3DQSAR itself. Much focus has been put on automation through the implementation of a scriptable interface, as well as on high computational performance achieved by algorithm parallelization. Flexibility and interoperability with existing molecular modeling software make Open3DQSAR a powerful tool in pharmacophore assessment and ligand-based drug design.

Keywords Chemometrics · Molecular interaction fields · PLS · 3D-QSAR · Variable selection

Part of this work was presented at the "Model(l)ing´09" meeting in Erlangen, Germany (7-11 September 2009). This paper is dedicated to the memory of Warren Lyford DeLano.

Electronic supplementary material The online version of this article (doi:10.1007/s00894-010-0684-x) contains supplementary material, which is available to authorized users.

P. Tosco (✉)
Department of Drug Science, University of Torino,
Via Pietro Giuria 9,
10125 Torino, Italy
e-mail: paolo.tosco@unito.it

T. Balle
Department of Medicinal Chemistry,
The Faculty of Pharmaceutical Sciences,
University of Copenhagen,
2 Universitetsparken,
2100 Copenhagen, Denmark

Introduction

From a methodological point of view computer-aided drug design is usually split into two branches, namely structure-based and ligand-based techniques. The latter are the only option when direct knowledge about the structure of the target cannot be attained, but a pool of compounds displaying different degrees of binding affinity, functional potency or enzyme inhibitory activity at the aforementioned target is available. This information can be conveyed into quantitative models which can guide further development of ligands with improved pharmacological profile and also provide indirect insight into the complementary structural features of the target. Three-dimensional quantitative structure-activity relationships (3D-QSAR) were introduced more than 25 years ago with the aim of finding statistical correlation between three-dimensional descriptors, whose arrangement in space gives rise to molecular interaction fields (MIFs), and biological activity. 3D-QSAR constitutes a *trait d'union* between classic QSAR and pharmacophore modeling, since, at least in its most popular implementations such as GRID/GOLPE [1, 2] and CoMFA/CoMSIA [3], it requires a consistent 3D alignment of all molecules, which basically corresponds to a pharmacophoric hypothesis. While a number of open-source tools aimed at structure-based design have been developed throughout the years by the academic community (AMBER, CHARMM, GROMACS, AutoDock, DOCK to cite but a few), commercial closed-source packages have largely dominated the scene of 3D-QSAR modeling so far [2, 4, 5]. Herein we describe Open3DQSAR, a free, open-source tool aimed at pharmacophore exploration by high-throughput chemometric analysis of MIFs [6]. Recently we proposed a GRID/GOLPE [1, 2] "consensus 3D-QSAR" approach complementing binding affinity and functional potency to gain insight into the differences

between a good binder and a good agonist across a series of ligands with affinity to the $\alpha_4\beta_2$ nicotinic receptor [7]. The choice of the most favorable superposition pattern was based on the identification of a model yielding good and balanced fit of the GRID MIFs to both pIC_{50} and pEC_{50} data. This model provided valuable hints on the putative pharmacophoric motifs which influence the degree of agonism, and thus also on the activation mechanism of $\alpha_4\beta_2$ receptors. The need for fast, automated exploration of a large number of different superposition schemes prompted us to develop a scriptable PLS engine implementing parallelized algorithms for variable selection and model validation. Open3DQSAR was written in C to achieve maximum performance; along with pre-built binaries for mainstream operating systems (native 32/64-bit Windows, Linux, Solaris, Mac OS X), fully commented source code is released under a no-fee license. Much effort has been put in developing Open3DQSAR's code in a modular fashion: basically, an input-parsing main function calls a number of computational subroutines. This arrangement is intended to facilitate the implementation of new features onto the original core, so that users are enabled to customize Open3DQSAR to their specific needs. Moreover, while Open3DQSAR has the capacity to work as a standalone application, it has been structured as a versatile application programming interface (API) whose functions may be called by external programs. This means that Open3DQSAR's computational core may constitute an ideal starting point to code a full-featured pharmacophore explorer relying on 3D-QSAR modeling for scoring and evaluation of the results.

Methods

Open3DQSAR's workflow is controlled by a single input file (see Fig. 1 for an example); the main output is constituted by a human-readable plain ASCII text, from which relevant numeric information can be easily extracted with shell tools such as grep, sed, awk. In addition to text output, a number of additional files are generated to store data and to export the results of computations for further analysis and visualization with third party software. A sample input file (Online Resource 1) and its respective text output (Online Resource 2) can be found in the Electronic Supplementary Material; moreover, full documentation and examples are available at the URL http://www.open3dqsar.org.

The first step to generate a 3D-QSAR model out of a dataset is gathering MIF information for all compounds. MIFs can be imported from a number of different sources, namely:

- GRIDKONT binary files generated by GRID [1];
- CoMFA/CoMSIA fields [3] (exported from SYBYL [4] with the aid of a small SPL script);
- Quantum-mechanical (QM) electron density/electrostatic potential fields generated with a variety of QM programs such as GAMESS [8], GAUSSIAN [9], JAGUAR [10], MOLDEN [11], TURBOMOLE [12];
- Plain text files in free format generated with any custom application.

There is also the opportunity to load 3D coordinates of a dataset and compute basic force-field-based MIFs inside Open3DQSAR, namely a steric field based on AMBER FF99 Van der Waals parameters [13] and an electrostatic field based on a point charge model. The steric field is computed according to Lennard-Jones' 6-12 potential between the molecule's n atoms and an sp^3 carbon probe (Eq. 1),

$$E_{VdW} = \sum_{i=1}^{n} \left[\frac{A_i}{r_i^{12}} - \frac{B_i}{r_i^{6}} \right], \quad (1)$$

while the electrostatic field is computed summing up Coulomb interactions between a positively charged probe and the molecule's n atoms (Eq. 2).

$$E_{ele} = k \sum_{i=1}^{n} \left[\frac{q_i}{r_i^{m}} \right]. \quad (2)$$

Once all MIFs of interest have been gathered, the dataset can be split into a training set and a test set, so that external validation may be carried out at a later stage. Then, standard data pre-treatment operations are usually accomplished in order to exclude at an early stage poorly informative variables, which only add noise or may even unfavorably condition the model. The following pre-treatment operations have been implemented in Open3DQSAR:

- Zeroing (sets to zero grid values which are close to zero);
- Max/min cut-off (sets to user-defined maximum/minimum threshold values the grid points lying above or below these boundaries, respectively);
- Exclusion of grid points which exceed the cutoff in a specific MIF (*e.g.*, allows to exclude from the chemometric analysis the grid points which are very close to atom nuclei and therefore assume high steric energy values);
- Standard deviation cut-off (removes variables having a standard deviation among different objects lower than a user-defined threshold, in order to improve the signal-to-noise ratio);
- *N*-level variable elimination (removes variables assuming only a few different values across the different objects to prevent them from biasing the model).

Different transformations are available for both X and Y blocks of variables (logarithms, multiplication by a

```
# Sample Open3DQSAR input file

# GRID fields and biological activities are imported
import type=gridkont kont=grid.kont lont=grid.lont
import type=dependent file=affinity.txt

# Variable pre-treatment
zero field_list=all type=all level=0.05

# Splitting of the dataset into training and test set
set object_list=5,7,10,16,21,23,25,29,35 attribute=testset

# Further pre-treatment operations
sdcut field_list=all level=0.1
nlevel level=2-4
remove_x_vars type=nlevel
scale_x_vars type=buw

# Generation, internal and external validation of a PLS model
pls pc=5
cv pc=5 type=lmo runs=100 groups=5
predict pc=5

# Variable grouping and selection
srd pc=5 seeds=3000 collapse=yes critical_distance=1.0 \
    collapse_distance=2.0 type=weights
ffdsel pc=5 type=lmo runs=100 groups=5 percent_dummies=20 \
    use_srd_groups=yes combination_variable_ratio=1.0 fold_over=no
remove_x_vars type=ffdsel

# Generation, internal and external validation of a new PLS model after variable selection
pls pc=5
cv pc=5 type=lmo runs=100 groups=5
predict pc=5

# PLS pseudo-coefficient isocontours are exported for visualization in PyMOL
format=insight type=coefficients file=3d_contours field_list=all

# A plot of experimental vs recalculated values is exported for visualization in Gnuplot
plot type=recalc_vs_exp file=recalc_vs_exp_plot label=name
```

Fig. 1 A sample Open3DQSAR input file showing a typical workflow consisting of model building, evaluation and refinement

coefficient, *etc*.), as well as scaling paradigms, such as autoscaling, custom scaling, and Block Unweighted Scaling [14], which has proven to be probably the most appropriate way to normalize multiple sets of field variables having different magnitudes. Once pre-treatment, transformations and scaling operations have been accomplished, a PLS model can be built extracting a user-defined number of principal components with the NIPALS algorithm [15]. The predictivity of the obtained model can be challenged by cross-validation (leave-one-out, leave-two-out, leave-many-out) yielding q^2/standard deviation of error of predictions (SDEP) statistics, as well as against an external test set. The robustness of the correlation can be further assessed by progressive scrambling as described by Clark and Fox [16]. It has been verified that the predictivity of a 3D-QSAR model can be significantly improved by appropriate variable clustering and selection procedures [17]. Some of them have been implemented in Open3DQSAR, namely:

- D-optimal design variable selection [17, 18];
- Smart region definition (SRD), as previously described by Pastor and co-workers [19]. SRD groups variables on the basis of their original localization in three-dimensional space. This procedure reduces redundancy arising from the existence of multiple nearby descriptors which basically encode the same kind of information;
- Fractional factorial design (FFD) variable selection, as originally described by Baroni *et al*. and implemented in GOLPE [17, 20]. FFD selection aims at selecting the variables which have the largest effect on predictivity, and can operate on both single variables or on groups identified by a previous SRD run;
- UVE-PLS variable selection, as originally described by Centner and co-workers [21], as well as the modified iterative IVE-PLS methodology developed by Polanski and colleagues [22]. These procedures remove the least informative variables, *i.e*., those characterized by small PLS pseudo-coefficients. Open3DQSAR's implementation of UVE/IVE-PLS has been further augmented including the possibility to use other cross-validation paradigms in addition to the leave-one-out scheme originally proposed by Centner, as recently suggested by Grohmann and Schindler [23]. Additionally, the algorithms can operate on both single variables and SRD groups, just as for FFD selection.

Apart from potentially improving the predictivity of a model, a very important consequence of variable selection is a diminished complexity of the resulting PLS pseudo-coefficient contour maps which significantly aids visual interpretation of 3D-QSAR models.

All matrix calculations rely on double precision BLAS and LAPACK libraries, either their publicly available implementations [24, 25] or their vendor-specific versions. Parallelization has been achieved using POSIX threads, an implementation of which is available for all mainstream

operating systems. Benchmarks were performed on two models of different size, obtained from the same 36-object dataset [7] using a coarse (1.0000 Å step size, 11088 variables) or fine (0.3333 Å step size, 272285 variables) grid mesh. The FFD selection was carried out using LMO CV (5 groups, 100 runs) extracting 5 principal components, using SRD groups rather than single variables, including 20% dummy variables, with a 1.0 combination/variable ratio. The version of Open3DQSAR used for benchmarks was built for a 64-bit Core2 architecture with Sun Studio Compiler for Linux version 12 Update 1 and linked against the Sun Performance Library. The benchmarks were obtained on a 2.40 GHz Dual Quad Core Intel Xeon E5530 workstation equipped with 24 GB 1066 MHz ECC DDR3 RAM. Program testing was carried out taking advantage of the ShareGrid distributed platform [26]. Some algorithms described by other authors have been implemented in Open3DQSAR to accomplish tricubic interpolation [27], to compute Student's *t* distribution [28], to generate high quality random numbers [29] and to perform an efficient quicksort [30].

Calling Open3DQSAR C functions from an external program is very straightforward; basically, input data from any source is loaded into a ModelData structure (see the open3dqsar.h header included in the source code for details), a pointer to which is passed to the computational routines together with other relevant parameters, as exemplified in Fig. 2.

Results and discussion

In the development of Open3DQSAR much focus has been put on achieving high computational efficiency. For this purpose, the most demanding computational tasks, namely cross-validation and variable selection procedures, have been parallelized to take advantage of multi-processor machines. In Fig. 3 two benchmarks are reported showing Open-3DQSAR's performance in carrying out FFD variable selection on two models of distinctly different size (11088 and 272285 variables, respectively) using an increasing number of CPU cores. While in the case of the smaller model the increase of performance is almost linear with the increasing number of CPU cores, as expected moving to the larger model the competition among CPU cores to access RAM reduces the performance gain when attempting to distribute the workload over more than six threads. Compared to the performance of Open3DQSAR on a single CPU, the Linux version of GOLPE 4.5 [2] was about 25-30% slower for the same variable selection procedure; we are perfectly aware that such a comparison is unfair, since GOLPE has a 32-bit architecture and probably has not been optimized for the instruction sets of modern CPUs. Rather, what we wish to stress is the possibility to introduce a change of perspective in ligand-based drug design using Open3DQSAR as a high-throughput tool to evaluate the quality of a number of pharmacophoric hypotheses by challenging their performance in fitting experimental

Fig. 2 C code snippet showing how to call Open3DQSAR's API functions from an external program

```
#include <open3dqsar.h>

[...]

result = import_gridkont (modeldata, replace_object_name, have_multi_mol2);
switch (result) {
    /*    check for errors    */
}

/*    calculate which variable are active    */
result = calc_active_vars (modeldata, FULL_MODEL);

/*    allocate structures for PLS analysis    */
result = alloc_pls (modeldata, modeldata->overall_active_x_vars, pc, FULL_MODEL);

/*    fill the X PLS matrix    */
result = fill_x_matrix (modeldata, FULL_MODEL, 0);

/*    fill the Y PLS matrix    */
result = fill_y_matrix (modeldata);

/*    mean-center X values    */
trim_mean_center_x_matrix (modeldata, FULL_MODEL, active_object_num);

/*    mean-center Y values    */
trim_mean_center_y_matrix (modeldata, FULL_MODEL, active_object_num);

/*    do PLS analysis with the NIPALS algorithm    */
pls (modeldata, pc, FULL_MODEL);

[...]
```

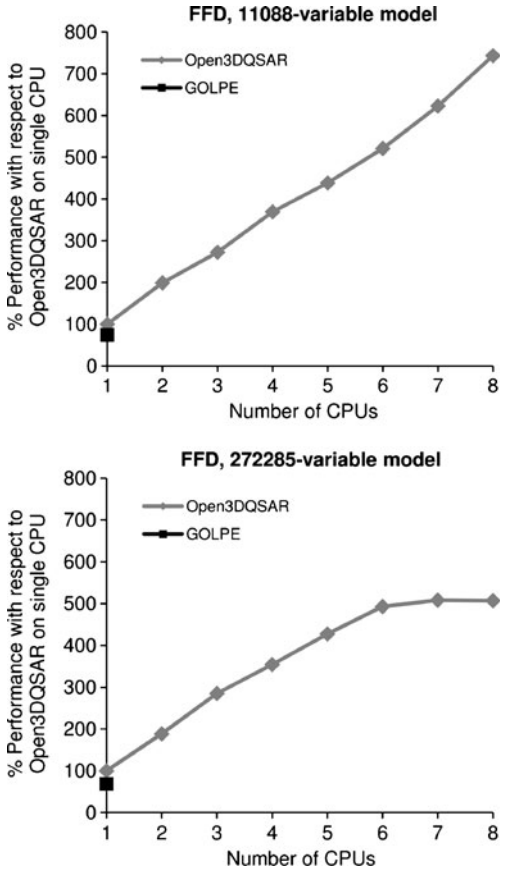

Fig. 3 Performance of Open3DQSAR as a function of the number of CPUs in carrying out a FFD variable selection on two models of different size (11088 and 272285 variables, respectively). Performance was evaluated as the inverse of computing time, and expressed as % gain with respect to single-CPU Open3DQSAR. The performance of GOLPE 4.5 for Linux [2] in the same benchmarks is reported as a reference

biological activities. As already mentioned, we successfully pursued this strategy in a recent work of ours [7], and we chose the same dataset and experimental design as a validation suite for Open3DQSAR. The software perfectly reproduced the results previously obtained with GOLPE [2] for calculations not involving random numbers. For procedures such as leave-many-out cross-validation (LMO CV) and related variable selection procedures, which rely on random number generation, minor differences were observed. In particular, we rebuilt with Open3DQSAR the models based on the 29 possible different alignments of a series of ligands with affinity for the $\alpha_4\beta_2$ nicotinic receptor [7], and for each of them we calculated SDEP for both pIC_{50} and pEC_{50} using an external test set. In addition to serving as a testbed for Open3DQSAR, the attempt to reproduce our previous results confirmed the robustness of our "consensus" scoring approach: in fact, in spite of the randomness involved in the LMO CV which is at the basis of the FFD variable selection, the correlation is very satisfactory (r^2= 0.88, Fig. 4). The model showing the best predictive performance with respect to both pIC_{50} and pEC_{50}, which is consequently more likely to represent a good guess of the binding mode in the $\alpha_4\beta_2$ nicotinic receptor, is the same as the one formerly identified using GOLPE (Fig. 4). The same consideration holds true also for the least predictive models, while as expected the models scoring half-way, which are characterized by SDEP values very close to each other, show a slightly higher variability. Increasing the number of LMO CV runs as well as the number of models evaluated during the FFD variable selection would obviously further improve the correlation, since the impact of the random group composition in building the LMO CV groups would then be minimized. The whole test suite was run overnight in a completely automated fashion by means of a trivial shell script on the same dual quad-core workstation used for the benchmarks reported in Fig. 3. On the contrary, in the original work the 58 models (29 for binding affinity and 29 for functional potency) had to be manually loaded, pretreated, refined through variable selection and re-evaluated one by one in the graphical user interface of GOLPE; it is evident that such a procedure is error-prone and would have been impossible with larger datasets. In Fig. 5 the coefficient plots referring to the highest ranked alignment have been visualized in a number of different molecular visualization/modeling applications (*i.e.*, PyMOL [31], MOE [32], SYBYL [4], Maestro [33]) to demonstrate the ease of

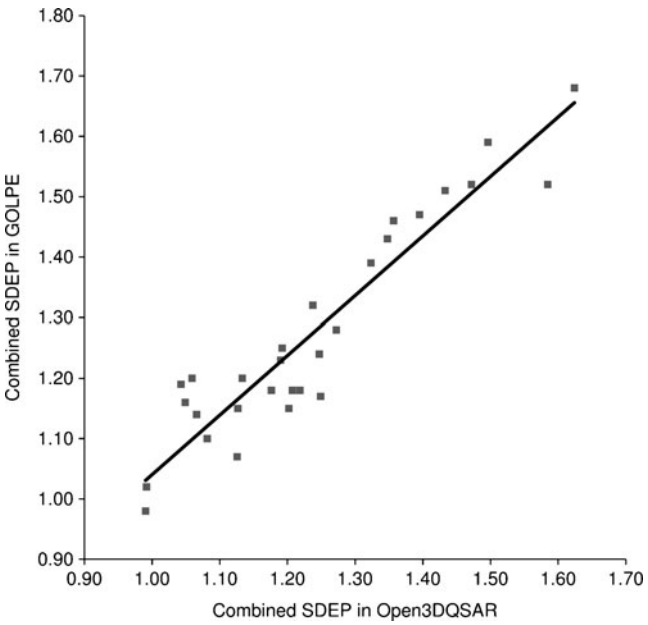

Fig. 4 Predictive performance of the 29 models previously built in GOLPE [7] compared to those newly built in Open3DQSAR. Scoring was accomplished on the basis of the combined SDEP statistics obtained in predicting pIC_{50} and pEC_{50} for an external test set, after performing a FFD selection on the training set

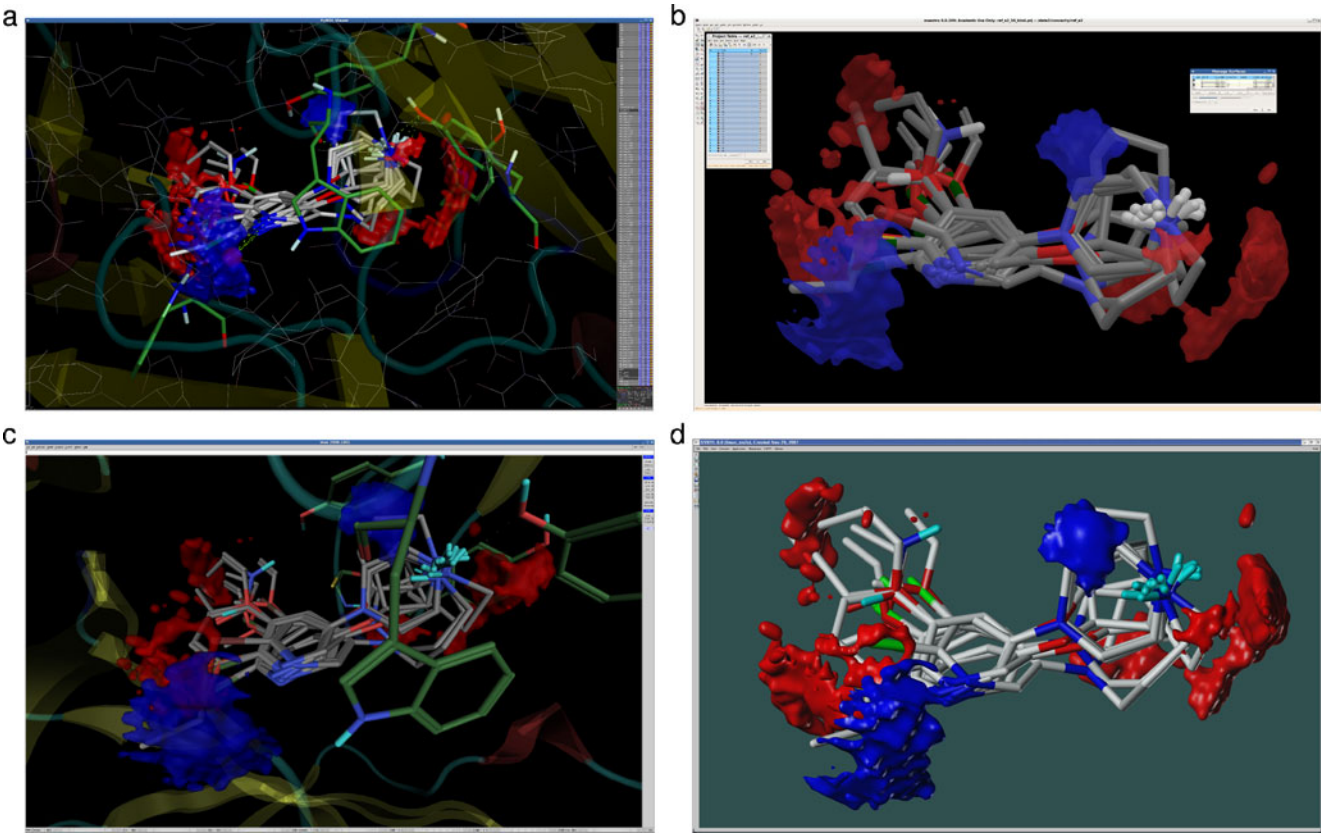

Fig. 5 PLS pseudo-coefficients visualized in different molecular visualization/modeling packages: **(a)** PyMOL [31]; **(b)** Maestro [33]; **(c)** MOE [32]; **(d)** SYBYL [4]

interfacing Open3DQSAR with existing software. In Fig. 6 the recalculated *vs.* experimental pIC_{50} chart for the top scoring consensus 3D-QSAR model is reported as an example of the numerous plot files which Open-3DQSAR can generate for subsequent visualization in Gnuplot [34]; this tool was chosen as the external plotting engine due to its popularity in the scientific community and to its free availability on all platforms under which Open3DQSAR currently runs.

Summary

The availability of a free, open-source 3D-QSAR engine fills a gap in the ligand-based design field, allowing medicinal chemists to have access to a well-established methodology which, after more than 20 years from its first introduction, still plays a key role in computer-aided drug design [35]. Open3DQSAR represents a versatile and customizable standalone platform where computational chemists can conveniently implement new procedures, algorithms and variable selection paradigms; at the same time, it can be regarded as a high-level PLS/variable selection API whose functions may be easily incorporated

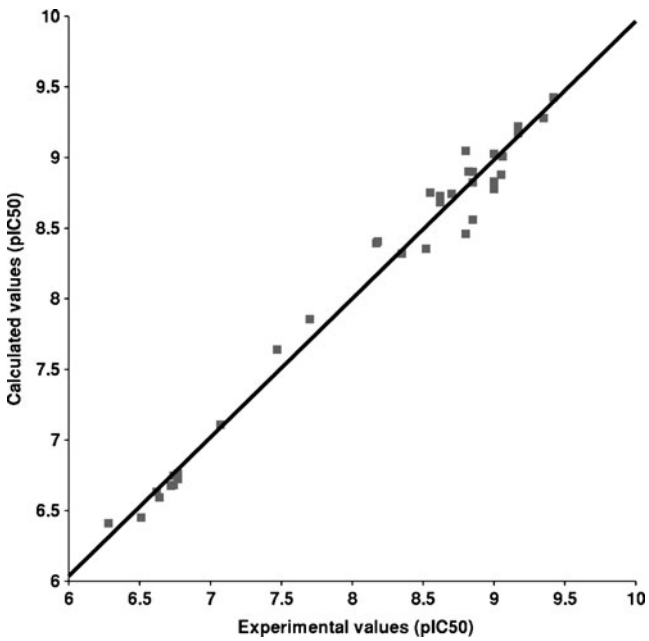

Fig. 6 Calculated *vs* experimental binding affinity values visualized through Gnuplot [34]

in external programs. The interoperability with the majority of existing molecular mechanics and quantum mechanics software packages should greatly facilitate the integration of 3D-QSAR modeling with other methodologies and open the possibility to combine MIFs from different sources (*e.g.*, quantum-mechanically derived fields in combination with more traditional molecular mechanics-derived ones). Documentation and information on how to obtain Open3DQSAR can be found at the URL http://www.open3dqsar.org.

Acknowledgments Open3DQSAR would never have seen the light of day without the invaluable pioneering work of Prof. Gabriele Cruciani and colleagues in the field of chemometrics applied to MIFs. We have referred to their detailed published methodologies [14, 17, 19, 20] to code Open3DQSAR's implementation of the Smart Region Definition and Fractional Factorial Design algorithms, a task which would have been extremely hard in the absence of such outstanding guidance. We are also indebted to the authors of the progressive scrambling, UVE-PLS and IVE-PLS methodologies, as well as to the authors of their later extensions [16, 21-23]. We gratefully acknowledge the ShareGrid management team for the computing power provided through the ShareGrid distributed platform. Finally, P.T. thanks Prof. Alberto Gasco and Prof. Roberta Fruttero (Università degli Studi di Torino) for their warm support and encouragement throughout the development. Part of the work was carried out by P. T. at the University of Copenhagen under a visiting scientist grant from the Drug Research Academy. T.B. was supported by grants from the Carlsberg Foundation and the Lundbeck Foundation.

References

1. GRID version 22C (2004) Molecular Discovery Ltd., Oxford, England; http://www.moldiscovery.com/. Accessed 24 December 2009
2. GOLPE 4.5 (1999) Multivariate Infometric Analysis S.r.l., Perugia, Italy; http://www.miasrl.com/golpe.htm. Accessed 24 December 2009
3. Cramer RD III, Wold S (1988) Comparative Molecular Field Analysis (CoMFA). Appl. No. 237,491, filed Aug. 26, 1988
4. SYBYL 7.3 (2009) Tripos International, St. Louis, MO, 63144, USA; http://www.tripos.com/. Accessed 24 December 2009
5. Phase version 3.1 (2009) Schrödinger, LLC, New York, NY; http://www.schrodinger.com. Accessed 24 December 2009
6. Tosco P, Balle T (2009) Open3DQSAR: a new open-source pharmacophore explorer based on chemometric analysis of molecular interaction fields. Proceedings of "Model(l)ing´09", 6-11 September 2009, Erlangen, Germany; http://www.chemie.uni-erlangen.de/modeling09/Abs_M09_Posters/Tosco.pdf. Accessed 24 December 2009
7. Tosco P, Ahring PK, Dyhring T, Peters D, Harpsøe K, Liljefors T, Balle T (2009) Complementary three-dimensional quantitative structure−activity relationship modeling of binding affinity and functional potency: a study on $\alpha_4\beta_2$ nicotinic ligands. J Med Chem 52:2311–2316. doi:10.1021/jm801060h
8. Schmidt MW, Baldridge KK, Boatz JA, Elbert ST, Gordon MS, Jensen JH, Koseki S, Matsunaga N, Nguyen KA, Su S, Windus TL, Dupuis M, Montgomery JA (1993) General atomic and molecular electronic structure system. J Comput Chem 14:1347–1363. doi:10.1002/jcc.540141112
9. Gaussian 03, revision C.02 (2004) Wallingford, CT, USA; http://www.gaussian.com. Accessed 24 December 2009
10. Jaguar version 7.6 (2009) Schrödinger, LLC, New York, NY, USA; http://www.schrodinger.com. Accessed 24 December 2009
11. Schaftenaar G, Noordik JH (2000) Molden: a pre- and post-processing program for molecular and electronic structures. J Comput-Aided Mol Des 14:123–134. doi:10.1023/A:1008193805436
12. TURBOMOLE V6.0 (2009) a development of University of Karlsruhe and Forschungszentrum Karlsruhe GmbH, 1989-2007, TURBOMOLE GmbH, since 2007; http://www.turbomole.com. Accessed 24 December 2009
13. Wang JM, Cieplak P, Kollman PA (2000) How well does a restrained electrostatic potential (RESP) model perform in calculating conformational energies of organic and biological molecules? J Comput Chem 21:1049–1074. doi:10.1002/1096-987X(200009)21:12<1049::AID-JCC3>3.0.CO;2-F
14. Kastenholz MA, Pastor M, Cruciani G, Haaksma EEJ, Fox T (2000) GRID/CPCA: a new computational tool to design selective ligands. J Med Chem 43:3033–3044. doi:10.1021/jm000934y
15. Wold S, Sjöström M, Eriksson L (2001) PLS-regression: a basic tool of chemometrics. Chemometrics Intell Lab Syst 58:109–130. doi:10.1016/S0169-7439(01)00155-1
16. Clark RD, Fox PC (2004) Statistical variation in progressive scrambling. J Comput-Aided Mol Des 18:563–576. doi:10.1007/s10822-004-4077-z
17. Baroni M, Costantino G, Cruciani G, Riganelli D, Valigi R, Clementi S (1993) Generating Optimal Linear PLS Estimations (GOLPE): an advanced chemometric tool for handling 3D-QSAR problems. Quant Struct-Act Relat 12:9–20. doi:10.1002/qsar.19930120103
18. De Aguiar PF, Bourguignon B, Khots MS, Massart DL, Phan-Than-Luu R (1995) D-optimal designs. Chemometrics Intell Lab Syst 30:199–210. doi:10.1016/0169-7439(94)00076-X
19. Pastor M, Cruciani G, Clementi S (1997) Smart Region Definition: a new way to improve the predictive ability and interpretability of three-dimensional quantitative structure−activity relationships. J Med Chem 40:1455–1464. doi:10.1021/jm9608016
20. Baroni M, Clementi S, Cruciani G, Costantino G, Riganelli D (1992) Predictive ability of regression models. Part II: selection of the best predictive PLS model. J Chemometr 6:347–356. doi:10.1002/cem.1180060605
21. Centner V, Massart DL, de Noord OE, de Jong S, Vandeginste BM, Sterna C (1996) Elimination of uninformative variables for multivariate calibration. Anal Chem 68:3851–3858. doi:10.1021/ac960321m
22. Gieleciak R, Polanski J (2007) Modeling robust QSAR. 2. Iterative variable elimination schemes for CoMSA: application for modeling benzoic acid pK_a values. J Chem Inf Model 47:547–556. doi:10.1021/ci600295z
23. Grohmann R, Schindler T (2008) Toward robust QSPR models: synergistic utilization of robust regression and variable elimination. J Comput Chem 29:847–860. doi:10.1002/jcc.20831
24. Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J, Du Croz J, Greenbaum A, Hammarling S, McKenney A, Sorensen D (1999) LAPACK Users' Guide. Society for Industrial and Applied Mathematics, Philadelphia
25. Whaley RC, Petitet A (2005) Minimizing development and maintenance costs in supporting persistently optimized BLAS. Softw-Pract Exp 35:101–121. doi:10.1002/spe.626
26. Anglano C, Canonico M, Guazzone M, Botta M, Rabellino S, Arena S, Girardi G (2008). Peer-to-peer desktop grids in the real world: the ShareGrid project. Proceedings of the 8th IEEE International Symposium on Cluster Computing and the Grid (CCGRID'08), Lyon (France), May 2008, IEEE Press. doi:10.1109/CCGRID.2008.23

27. Lekien F, Marsden J (2005) Tricubic interpolation in three dimensions. Int J Numer Methods Eng 63:455–471. doi:10.1002/nme.1296
28. Brown B, Lovato J, Russell K (2006) DCDFLIB; http://people.sc.fsu.edu/~burkardt/f_src/dcdflib/dcdflib.html. Accessed 24 December 2009
29. Matsumoto M, Nishimura T (1998) Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Trans Model Comput Simul 8:3–30. doi:10.1145/272991.272995
30. Stewart DE, Leyk Z (1994) Meschach Library version 1.2b; http://www.math.uiowa.edu/~dstewart/meschach/. Accessed 24 December 2009
31. PyMOL (2009) DeLano Scientific LLC, Palo Alto, CA, USA; http://www.pymol.org. Accessed 24 December 2009
32. MOE version 2009.10 (2009) Chemical Computing Group Inc, Montreal, Quebec, Canada; http://www.chemcomp.com. Accessed 24 December 2009
33. Maestro version 9.0 (2009) Schrödinger LLC, New York, NY, USA; http://www.schrodinger.com. Accessed 24 December 2009
34. Gnuplot version 4.2 (2009); http://www.gnuplot.info/. Accessed 24 December 2009
35. Cross S, Cruciani G (2009) Molecular fields in drug discovery: getting old or reaching maturity? Drug Disc Today. doi:10.1016/j.drudis.2008.12.006