Supplementary Material

EASYMIFs and SITEHOUND: a toolkit for the identification of ligand-binding sites in protein structures

Dario Ghersi and Roberto Sanchez

1 Running EASYMIFs or SITEHOUND separately

In addition to the combined used of EASYMIFs and SITEHOUND through `auto.py` it is possible to use each of the tools separately as described here (and in more detail in the manual).

EASYMIFs requires a PDB file as input and produces a MIF file as output using simple commands:

```
prepare_pdb.py 1kna.pdb
easymifs -f=1kna.easymifs -p=PROBE
```

where the first command pre-processes the PDB file of the structure of interest (e.g. 1kna.pdb). The pre-processed file (`1kna.easymifs`) is then used as input to EASYMIFs and a probe is specified with the `-p` option. EASYMIFs automatically determines the dimensions of a box large enough to enclose the whole protein, with a clearance of 5Å in each direction and a resolution of 1Å. Alternatively, command line options (described in the manual) can be used to specify the center, dimensions, and resolution of the grid.

SITEHOUND uses the output .dx file from EASYMIFs (or other MIF calculation programs such as `AutoGrid`) and produces several output files using the following command:

```
sitehound -f=1kna_CMET.dx -t=easymifs -e=-8.9 -l=average -s=7.8
```

where `-f` specifies the MIF file and `-t` the format of the file (for example, affinity maps from `AutoGrid` can be used with the `-t=autogrid` option). `-e` is the energy threshold above which MIF points are removed, and `-l` specifies the linkage for the clustering algorithm. `-s` is the spatial cutoff, i.e. the level at which the hierarchical tree obtained during the clustering step will be cut.

2 Methods

2.1 Calculation of MIFs in EASYMIFS

EASYMIFS computes the potential energy between a chemical probe (represented by a particular atom type) and the protein on a regularly spaced grid, using the following equation:

$$V_i = \sum (V_{LJ}(r_{ij}) + V_E(r_{ij})) \tag{1}$$

where the potential energy calculated for a probe at a point i in the grid is equal to the sum of a Lennard-Jones and an electrostatics term over all the atoms of the protein. r_{ij} represents the distance between the probe at point i in the grid and an atom j of the protein. The Lennard-Jones and the electrostatics term are expressed by the following two equations:

$$V_{LJ}(r_{ij}) = \frac{C_{ij}^{(12)}}{r_{ij}^{12}} - \frac{C_{ij}^{(6)}}{r_{ij}^{6}} \tag{2}$$

$$V_E(r_{ij}) = \frac{1}{4\pi\epsilon_0} \frac{q_i q_j}{\epsilon(r_{ij}) r_{ij}} \tag{3}$$

The $C^{(12)}$ and $C^{(6)}$ parameters in the Lennard-Jones term depend on the chosen probe and the particular atom type and are taken from a matrix of LJ-parameters distributed with the GROMACS package[1]. The dielectric constant $\frac{1}{4\pi\epsilon_0}$ has been set to 138.935485. The distance-dependent dielectric sigmoidal function has been taken from Solmajer and Mehler[2] and has the following form:

$$\epsilon(r_{ij}) = A + \frac{B}{1 + \kappa e^{-\lambda B r_{ij}}} \tag{4}$$

where $A = 6.02944$; $B = e0A$; $e0 = 78.4$; $\lambda = 0.018733345$; $k = 213.5782$. When the distance between the probe and an atom becomes less than 1.32Å, a dielectric constant of 8 is used. The parameters reported above for the distance-dependent dielectric have been taken from Cui et al.[3]

2.2 Brief overview of clustering in SITEHOUND

The main idea implemented in SITEHOUND is to group the points of the interaction energy map that have passed the energy filter into clusters and to rank them by TIE. It is important to understand the options related to the clustering step in order to effectively use the program. The principles of clustering algorithms and the relevant parameters used by SITEHOUND are discussed here.

The fundamental goal of a clustering algorithm can be considered as finding a partition of a set of points, defined in a multidimensional space, according to some **optimality criterion** (usually, one seeks to minimize intra-clusters distances and maximize inter-clusters distances). It is worth pointing out that the problem is NP-complete, because one should calculate all the possible partitions of the points, a combinatorial problem that scales with the factorial of the number of points. In practice, one can resort to heuristics that make the problem amenable to computation and yield satisfactory results.

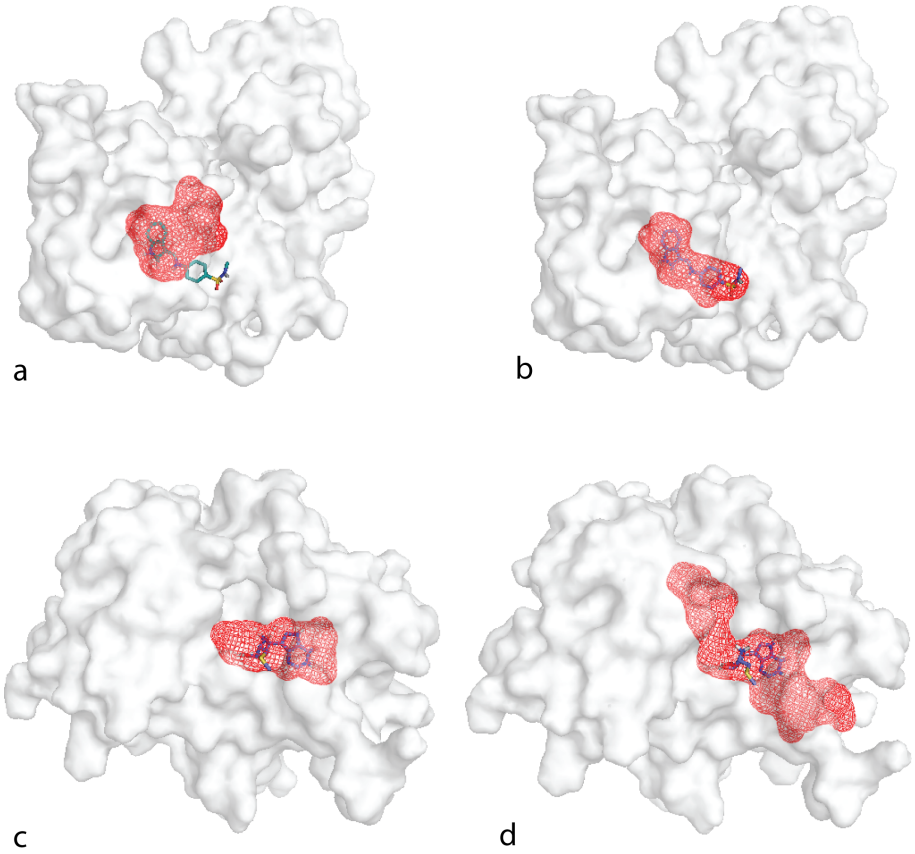

Figure 1: **Effects of linkage on clustering results** - a) and b) show the results of average and single linkage on cyclin-dependent kinase 2 (PDB code 1ke5). Single linkage yields a better coverage of the binding pocket, which is quite elongated. On the other hand, for human pregnenolone sulfotransferase (PDB code 1q1q) average linkage is the best choice, since it corresponds more closely to the ligand contour.

More formally, given:

$$\mathbf{x_1} = \{x_{11}, x_{12}, \ldots, x_{1n}\}, \ \ldots, \ \mathbf{x_m} = \{x_{m1}, x_{m2}, \ldots, x_{mn}\} \tag{5}$$

as a set of m points belonging to an n dimensional space, we can define the following two quantities:

1. $D_p(\mathbf{x_1}, \mathbf{x_2})$
2. $D_c(\mathbf{R}, \mathbf{S})$

that represent the distance between two points $\mathbf{x_1}$ and $\mathbf{x_2}$ and the distance between two clusters $\mathbf{R}$ and $\mathbf{S}$, respectively. A natural choice for D_p in our problem is the simple euclidean distance between the points.

One of the most widely used heuristics to approach the clustering problem is to proceed from to the bottom to the top by iteratively merging clusters until one cluster containing all the points is obtained. This is where the D_c quantity plays a role, by defining the distance between clusters. The name **linkage**

is commonly used to indicate this quantity.

SITEHOUND incorporates two types of linkage, *single* and *average*, defined in the following way:

$$D_{c\_single}(\mathbf{R},\mathbf{S}) = \min_{\mathbf{x_1}\in\mathbf{R},\mathbf{x_2}\in\mathbf{S}} D_p(\mathbf{x_1},\mathbf{x_2})$$

$$D_{c\_average}(\mathbf{R},\mathbf{S}) = \frac{\sum_{\mathbf{x_1}\in\mathbf{R}}\sum_{\mathbf{x_2}\in\mathbf{S}} D_p(\mathbf{x_1},\mathbf{x_2})}{|\mathbf{R}||\mathbf{S}|}$$

where the | | notation indicates the cardinality of the set (i.e. the number of points of the cluster).

Two important properties shared by these two linkages are the fact that the distance between clusters increases monotonically at each step. Therefore, it is possible to cut the partition at a particular level obtaining the corresponding clusters. In SITEHOUND this level is called **spatial cutoff**. The type of linkage used affects (to some extent) the shape of the clusters obtained. In general, it can be shown that single linkage tends to yield more elongated clusters, whereas with average linkage the shape of the clusters is closer to a sphere. From a practical point of view, using single linkage can be more meaningful with peptide binding sites or elongated ligands, whereas average linkage performs better with small chemicals. These effects are illustrated in Figure 1. In general, it is desirable to run the calculations with both types of linkage, and compare the results. In some instances, with average linkage the binding site is split in two regions, whereas single linkage will tend to show one single site. This information could be valuable in the context of ligand design, since the two regions that show up with average linkage could both be exploited by connecting two fragments with a linker.

3 Benchmark

We performed a benchmark to estimate the time required to carry out the full binding site identification pipeline on proteins of different sizes. The dataset used was derived from the Astex Diverse Set[4]. The script `auto.py` (which is provided in the download package) automatically returns the actual time required by each individual step (protein preparation, interaction energy calculations and cluster analysis) and was used for the benchmark.
The hardware used was the following:

- Mac Pro, Intel Xeon 3GHz (Mac OS X 10.5)
- Dell Precision, Intel Core Duo 3GHz (Linux Ubuntu)
- Toshiba Satellite, Pentium IV 3GHz (Windows XP)

The machines used cover a broad spectrum (from a 5 years old laptop to much faster workstations) and therefore the results cannot be directly compared, but are meant to provide an approximate indication of the running time on a range of platforms. Figures 2 reports the total time required for each protein as a function of the number of residues and the individual time for EASYMIFS and SITEHOUND respectively.

References

[1] D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. Berendsen. Gromacs: fast, flexible, and free. *J Comput Chem*, 26(16):1701–18, 2005.

[2] T. Solmajer and E.L. Mehler. Electrostatic screening in molecular dynamics simulations. *Protein Eng*, 4(8):911–7, 1991.

[3] M. Cui, M. Mezei, and R. Osman. Prediction of protein loop structures using a local move monte carlo approach and a grid-based force field. *Protein Eng Des Sel*, 21(12):729–35, 2008.

[4] M.J. Hartshorn, M.L. Verdonk, G. Chessari, S.C. Brewerton, W.T. Mooij, P.N. Mortenson, and C.W. Murray. Diverse, high-quality test set for the validation of protein-ligand docking performance. *J Med Chem*, 50(4):726–41, 2007.

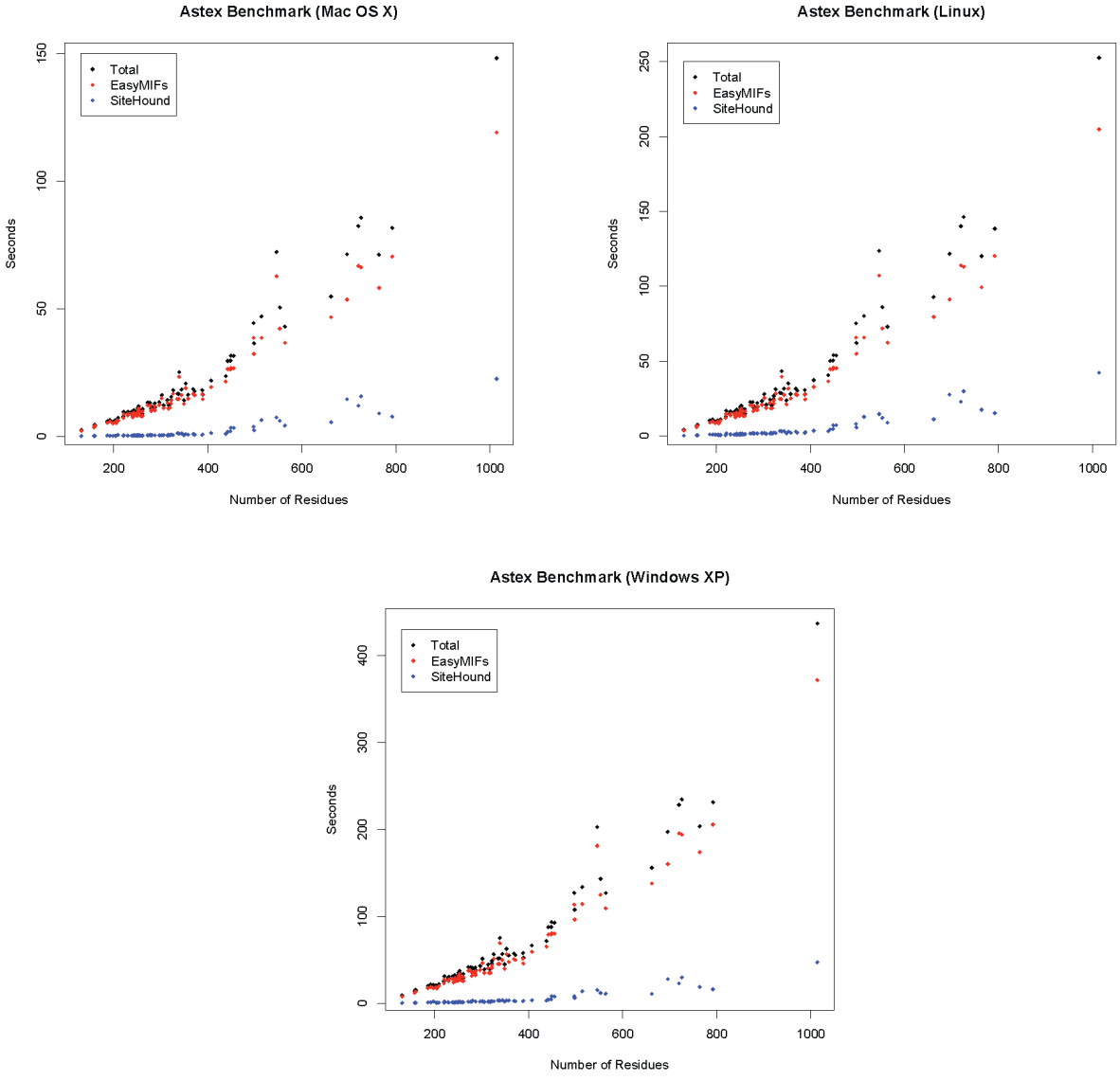

Figure 2: **Running time on different platforms as a function of number of residues**