Pharao

- Version 3.0.2 -

Last update: September 29, 2010

Table of contents

1. Introduction

Pharao is an acronym for **Phar**macophore **A**lignment and **O**ptimization. A pharmacophore is an abstract concept based on the specific interactions that have been observed in drug-receptor interactions: hydrogen bonding, charge transfer, electrostatic and hydrophobic interactions. Molecular modeling and/or screening using pharmacophores have proven to be an important and useful method in drug discovery.

The functionality of *Pharao* consists mainly of two parts. The first functionality consists of the generation of pharmacophores from molecules (see section 2.2). Second, pairs of pharmacophores can be aligned (see section 2.4) and the score are calculated from the volume overlap resulting from the alignments.

For the representation of pharmacophore points, a 3-dimensional Gaussian volume is used in which the volume is defined by its center and spread or sigma.

Since alignment methods are dependent on the molecular orientation and position they tend to reflect a combinatorial problem that sometimes results in high computation times. Several approaches are introduced within *Pharao* to handle this problem and this makes *Pharao* a screening tool that is sufficiently fast. The alignment as implement in *Pharao* is called a 'rigid alignment', meaning that no flexibility of the input structures is assumed and the program always works with one, fixed, conformation. To obtain additional conformations of a molecule external software can be used as a preprocessing step.

In section 2 the implementation details are explained to give some insight in the working of *Pharao*, and in section 3 a detailed explanation of the command-line parameters and functions is given.

2. Implementation details

2.1. Pharmacophores

2.1.1. Concept

A pharmacophore is described as an ensemble of functional groups, or structural features, with a defined geometry. In *Pharao* a pharmacophore is represented as a set of pharmacophore points, whereby each pharmacophore point has the following properties:

- the type of functional group;
- the center of the point;
- the spread (α);
- the normal, if applicable.

Each pharmacophore point is modeled as a 3-dimensional spherical Gaussian volume represented by its center (coordinate) and spread (α). The definition of the Gaussian volume is given as follows:

$$V_p = \int p \exp\left(-\alpha |m - r|^2 dr\right)$$

with V_p being the Gaussian volume, p being normalization constant to scale the total volume to a level that is in relation to atomic volumes, m being the center of the Gaussian, and r being the distance variable that is integrated.

The coordinate m of a pharmacophore point defines the position in space. All pharmacophore points must have a position in space. Each pharmacophore point is also characterized by α that defines the volume of the point in space. α is chosen inverse proportional to the square root of the radius.

Each pharmacophore point is characterized by a functional type. These functional types are considered to be important in the selective binding of molecules. Each functional group is labeled with a four-lettered code and the possibilities as implemented within *Pharao* are given in Table 1.

Some pharmacophore points also have a 'direction' as defined by its normal. The normal is a vector originating from the center of the pharmacophore point. It is optional to include this information during alignment and scoring. The rationale for the use of a normal in the alignment is that, for instance, a hydrogen bond acceptor works to the outside of the molecule, and an aromatic ring is a planar structure that has an orientation in space. This spatial orientation is not modeled as such by the Gaussian volume, hence the use of the normal to take this orientation into account.

Table 1. The set of functional groups defined in Pharao.

Code	Description	α	Normal	Hybrid
AROM	Aromatic ring	0.7	Yes	No
HDON	Hydrogen bond donor	1.0	Yes	No

Code	Description	α	Normal	Hybrid
HACC	Hydrogen bond acceptor	1.0	Yes	No
LIPO	Lipophilic region	0.7	No	No
POSC	Positive charge center	1.0	No	No
NEGC	Negative charge center	1.0	No	No
HYBH	Hydrogen bond donor and acceptor	1.0	Yes	Yes
HYBL	Aromatic and lipophilic	0.7	No	Yes
EXCL	Exclusion sphere	1.7	No	No

2.1.2. Format

Once generated, pharmacophores can be written to a file using a special tab-delimited format. This way, pharmacophores of molecules can be stored and used for screening or mapping without generating this information each time again. It is recommended to use the '`.phar`' extension for pharmacophore files.

The following format is defined for writing pharmacophores:

```
name
CODE Cx Cy Cz α norm Nx Ny Nz
...
CODE Cx Cy Cz α norm Nx Ny Nz
$$$$
```

Every pharmacophore starts with a variable *name*, which is used to identify the pharmacophore. In principle, the name of the pharmacophore is set identical to the title of the molecule of which the pharmacophore is calculated. Then for each pharmacophore point a new line is used, containing the following information:

- *CODE* is one of the nine codes listed in Table 1;
- *Cx*, *Cy* and *Cz* are the coordinates of the pharmacophore point;
- α is the spread of the Gaussian;
- *norm* is a Boolean parameter (1 or 0) indicating whether this particular point contains normal information;
- *Nx*, *Ny* and *Nz* are the coordinates of the normal. With pharmacophore points that have no normal information, these three data points are set to 0.

The end of the pharmacophore is indicated with a `$$$$` token. This way, a file can contain multiple pharmacophores.

Lines starting with the # symbol are considered documentation and are skipped during parsing of a pharmacophore file.

This human-readable format enables the manual modification of a pharmacophore. To remove a pharmacophore point from a pharmacophore, it is sufficient to remove the corresponding line in the file.

2.2. Generating pharmacophore points

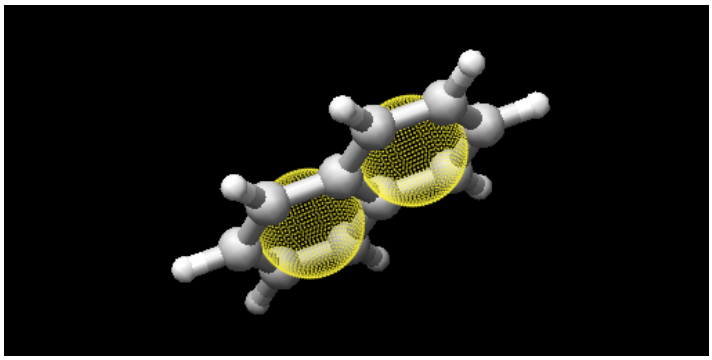
2.2.1. Aromatic rings

The generation of aromatic ring pharmacophore points, or AROM points, includes ring detection and aromaticity detection.

Ring systems containing multiple aromatic rings will be converted into multiple AROM points. Figure 1 illustrates this for the molecule naphthalene, consisting of a ring system with two benzene rings.

The position of the AROM point is the center of the ring it represents. AROM points also contain a normal as extra information. This normal indicates the orientation of the aromatic ring and is placed perpendicular on the plane formed by the ring. Because its sole purpose is to indicate the orientation of the plane, the normal is always a unit vector with length 1 Å.

Figure 1. Visualization of the two generated AROM points for naphthalene. Both points are shown as yellow spheres. The normals are not shown.



If the angle between two normal vectors is zero, then two corresponding ring planes are parallel to each other. The value of this angle can act as a penalty when comparing two AROM points to each other.

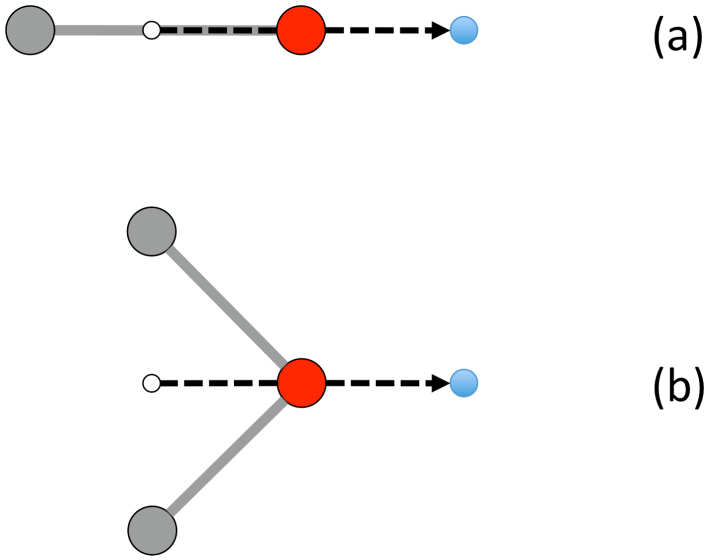
2.2.2. Hydrogen bond donors

The generation of hydrogen bond donor pharmacophore points, or HDON points, is based on topological information according a simple procedure. To be labeled as a hydrogen bond donor, every atom is checked for the following conditions:

- Only nitrogen or oxygen atoms;
- Formal charge is not negative;
- At least 1 attached hydrogen atom.

The center of the HDON point is the position of the heavy atom that is labeled as a valid hydrogen bond donor. Hydrogen bond donor pharmacophore points are also characterized by normal information. The direction of this normal is calculated from the average position of all the non-hydrogen atoms that are bound to the hydrogen bond donor atom, shifted to a length of 1 Å and projected along this vector to the other side of the hydrogen bond donor atom (Figure 2). The position of the hydrogen atom is not taken into account for the calculation of the normal.

Figure 2. Illustration of the procedure to position the normal on a hydrogen bond donor pharmacophore point as shown for a hydrogen bond donor atom connected to a single heavy atom (a) or to two heavy atoms (b). The hydrogen bond donor atom is colored red, the associated normal point light blue, and the attached atoms gray. A similar procedure is used to calculate the normals of the hydrogen bond acceptor pharmacophore points.



2.2.3. *Hydrogen bond acceptors*

The generation of hydrogen bond acceptor points, or `HACC` points, is less straightforward than the generation of `HDON` points. A hydrogen bond acceptor needs to fulfill four conditions:

- Only nitrogen or oxygen atoms;
- Formal charge not positive;
- At least one available lone pair;
- Atom is 'accessible'.

These conditions, which will be described in more detail below, where previously described by Greene and coworkers[1].

In order to determine the third condition, only nitrogen atoms need to be validated whether their lone pair electrons are delocalized or not. Some simple heuristic rules are implemented to validate this. A nitrogen has no available lone pair electrons if the atom obeys one of the following patterns:

- N is in aromatic ring and has three bonds (e.g. pyrrole);
- N-S=O (e.g. sulfonamide);
- N-C=X with X=N,O,S (e.g. peptide bond);
- N is adjacent to aromatic ring and has three bonds (e.g. aniline).

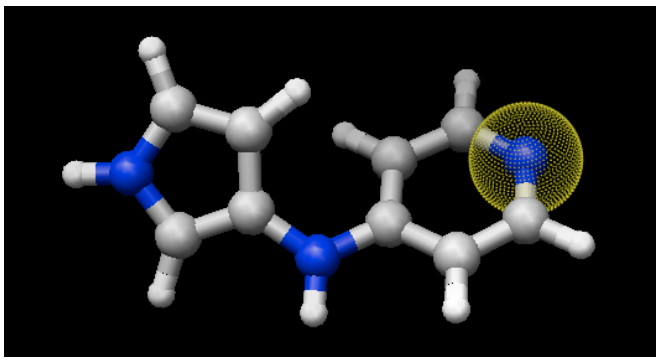
[1] Green J.; Kahn S.; Savoi H.; Sprague P.; Teig S. Chemical function queries for 3D database search. *J. Chem. Inf. Comput. Sci*. **1994**, *34*, 1297-1308

Condition four, the accessibility of the hydrogen bond acceptor atoms, is somewhat more difficult to calculate. Accessibility means that there is enough space for a putative hydrogen atom to form a hydrogen bond without forming a steric clash with any of the other atoms of the molecule.

This accessibility is calculated by placing a sphere around the putative hydrogen bond acceptor atom with a radius of 1.8 Å, thereby mimicking the possible locations where a hydrogen atom can be localized in theory. Subsequently a number of points are sampled on this sphere and for every point it is verified whether there is a collision with any of the neighboring atoms. If at least 2% of the points are labeled as 'free', the hydrogen bond acceptor atom is labeled as 'accessible'.

By imposing conditions three and four as additional criteria for the determination of a hydrogen bond acceptor pharmacophore point, the number of hydrogen bond acceptors are significantly reduced by removing a large number of unrealistic pharmacophore points (Figure 3).

Figure 3. Illustration of hydrogen bond acceptor pharmacophore points. Only one HACC point was generated and is shown as a yellow sphere. The molecule contains three nitrogen atoms that could serve as hydrogen bond acceptor pharmacophore centers, but only the right nitrogen satisfies all four constraints and therefore gets labeled as a hydrogen bond acceptor. The normal of the point is not shown.



The normal of the hydrogen acceptor pharmacophore point is calculated in an identical manner as the normals of the hydrogen bond donor pharmacophore points (Figure 2).

2.2.4. *Lipophilic spots*

To generate lipophilic pharmacophore points, or LIPO points, the following procedure is used. First, every atom is assigned a lipophilic contribution value. This value is the product of a topology-dependent term t and the accessible surface fraction s. Term t is obtained using some simple heuristic rules, which are listed in Table 2, and fraction s, representing the accessibility of an atom, is calculated using a method similar to the one described in section 2.2.3. For example, a carbon atom with an accessibility of 80% and located three bonds away from double bonded oxygen will have a lipophilic contribution of 0.48 (s = 0.8, t = 0.6)

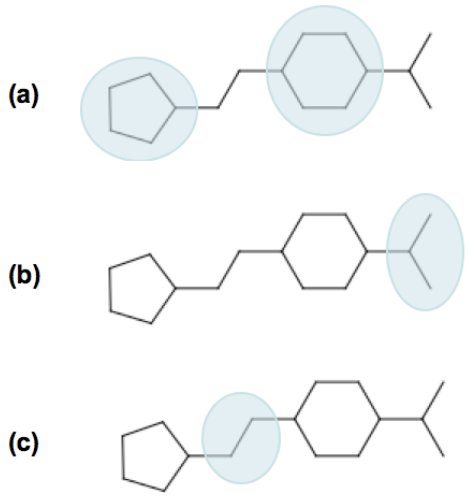
Table 2. Topology-dependent lipophilic factors. The total term t of an atom is the product of all appropriate f values for all the categories into which the atom falls.

Category	*f*	Description
1	0	N, O or H
2	0	S in SH
3	0	≤ 2 bonds away from charged atom
4	0	≤ 2 bonds away from OH or NH with no delocalized electrons
5	0	≤ 1 bond away from SH with no delocalized electrons
6	0	≤ 2 bonds away from O with double bond
7	0	≤ 1 bond away from S with valence > 2
8	0	S with double bond
9	0.6	3 bonds away from O with double bond
10	0.6	2 bonds away from S with valence > 2
11	0.6	1 bond away from S with double bond
12	0	Two or more instances of any of the previous three conditions (cat 9-11)
13	0.25	1 neighboring O or N with no delocalized electrons
14	0	> 1 neighboring O or N with no delocalized electrons
15	1	Not belonging to any of the previous conditions (cat 1-14)

After that a lipophilic contribution is assigned to every atom, the next step is to group atoms into regions or spots. The procedure to group atoms into spots is illustrated in Figure 4.

- Atoms in a ring of size 7 or less form a group (Figure 4a).
- Atoms with three or more bonds, together with their neighbors not bonded to any other atom, form a group (Figure 4b).
- The remaining of the atoms (the chains) also form groups (Figure 4c).

Figure 4. Schematic representation of procedure to group atoms into spots. This example molecule contains four spots.



During the third and final step, the total lipophilic contribution is calculated for each of the spots as the summation of the contributions of every atom belonging to that spot. If this value exceeds a pre-defined threshold, a LIPO pharmacophore point is created with the center of it being set to the center of the spot. The threshold value is set to 9.87, which is half of the lipophilic contribution of an exposed methyl carbon terminating a carbon chain[1].

2.2.5. Charge centers

For the generation of charge center pharmacophore points, the formal charges on the atoms of the molecule are used. Atoms with a positive formal charge will correspond with a positive charge center pharmacophore point, or POSC point, and atoms with a negative formal charge will define the position of a negative charge center pharmacophore point or NEGC point.

The position of the POSC or NEGC point coincides with the position of the atom with the formal charge.

2.2.6. Hybrid lipophilic centers

Hybrid lipophilic pharmacophores HYBL are generated by merging closely positioned the LIPO and AROM points together. In order for these to be merged, the distance between the two respective centers should be less than 1.0 Å. The center coordinates of the new point are calculated by taking the average of the two original centers. When hybrid lipophilic centers are requested, all LIPO and all AROM points are also renamed into HYBL. After merging or renaming, the normal information of the original aromatic centers is disguarded.

To summarize, generation of HYBL points leads to the following:

- Isolated AROM points are renamed to HYBL and their normal information is disguarded;
- Isolated LIPO points are renamed to HYBL;
- Closely located AROM and LIPO points are merged into a HYBL single point and the normal information of the original AROM point is removed. The new coordinates are calculated from the average of the original coordinates

2.2.7. Hybrid hydrogen donors and acceptor centers

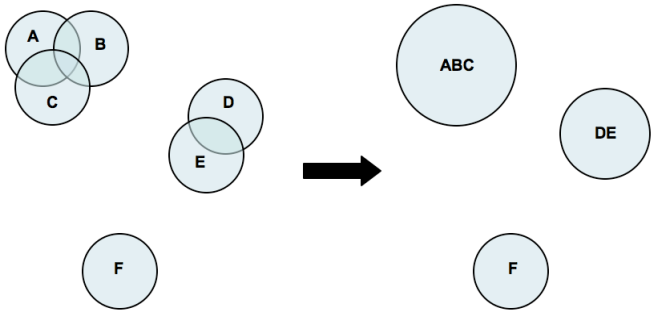
Hybrid hydrogen acceptor/donor pharmacophores HYBH are generated by merging together HACC and HDON points that are located on the same atom. In order for these to be merged, the distance between the two respective centers should be less than 0.00001 Å. After merging, the normal of the new center is calculated by taking the average location of the two original normals.

2.3. Merging pharmacophore points

Because of the combinatorial nature of the feature mapping (see section 2.4.2), extended sets of pharmacophore points can result in long computational time. A possible solution to circumvent this problem is to merge

neighboring pharmacophore points of the same category, as is illustrated in Figure 5.

Figure 5. Schematic representation of the merging process. A pharmacophore consisting of 6 points is reduced to a new pharmacophore consisting of only 3 points.



Pharmacophore points are considered neighboring if their overlap volume exceeds a threshold value of 0.075. The α of the resulting pharmacophore point is set to 70% of the combination of all the alpha values of the individual pharmacophore points. A merged pharmacophore point does not have normal information.

2.4. Aligning pharmacophores

2.4.1. Problem situation

The quantification of the similarity between two pharmacophores can be computed from the overlap volume of the Gaussian volumes of the respective pharmacophores. The idea is to identify the subset of matching functional groups in each pharmacophore that gives the largest overlap. The procedure finds its roots in the work of Grant and Pickup[2], where the volume overlap between two molecules is computed from a Gaussian description of the atomic volumes. In *Pharao* this approach is translated into the overlap of pharmacophore points.

The procedure to compute the volume overlap between two pharmacophores is implemented in two steps. In the first step, a list of all feasible combinations of overlapping pharmacophore points is generated. In the second step, the corresponding features are aligned with each other using an optimization algorithm. The combination of features that gives the maximal volume overlap is retained to give the matching score.

2.4.2. Feature mapping

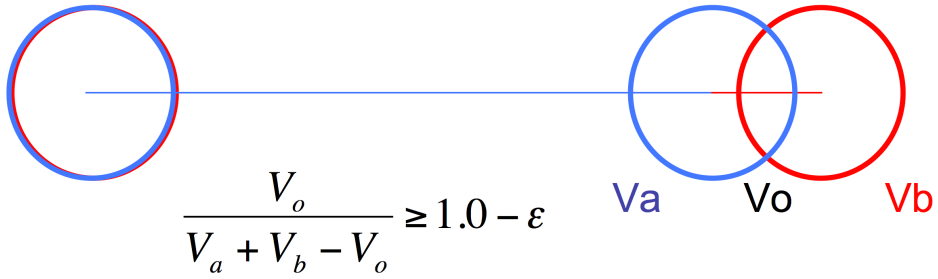
To compute the overlap between a pair of pharmacophores, the first step is to define the points from the database pharmacophore that can be mapped onto points from the reference pharmacophore. A mapping of two pharmacophores consists of a list of points from reference and database pharmacophores where corresponding points have a compatible functional group and the internal distance between the points is within a

[2] Grant, et al. 1996. J.Comp.Chem., 17, 1653-66.

given range requirement. This range, as defined by the parameter ε, controls the feasibility of a given combination of pharmacophore points.

The procedure starts by generating a list of all feasible pairs of features. First, two points from the reference pharmacophore are selected and the distance between these is calculated. Next, two points with matching features are selected from the database pharmacophore. Next, the first points of both couples are overlaid and then the relative overlap of the second points is computed. If the relative overlap is larger than (1.0 - ε), the combination of the two pairs is said to be feasible. This is illustrated in Figure 6. When ε is set to 1.0, this indicates that the two pairs do not have to show any overlap. Smaller values of ε indicate that more overlap is required and becomes as such a more stringent selection criterion.

Figure 6. Illustration of the ε parameter. Two subsets of corresponding pharmacophore points are selected (black and blue). The first points are placed on top of each other (left sphere). The relative overlap between the other spheres should be larger than 1.0 – ε. From this it implies that a smaller ε implements a more stringent feasibility criterion.



Once the list of feasible pairs is constructed, they can be combined into larger feasible combinations. This process is combinatorial in nature and the number of possible combinations grows very fast with the number of pharmacophore points. The ε parameter allows reducing the number of feasible combinations.

2.4.3. *Alignment phase*

Given the set of all feasible combinations, the combination that gives the largest volume overlap is searched for. For every combination, the procedure starts by translating the database pharmacophore subset such that its geometric center overlaps the geometric center of the reference pharmacophore subset. Next, using a combination of gradient-ascent and rigid-body rotation, the maximal volume overlap is determined. Details of the methodology are described in section **Error! Reference source not found.**.

The alignment procedure starts with the largest combinations and computes the volume overlap. Next the smaller combinations are processed until the best score so far is higher than the maximum score any smaller combination could achieve. The rationale is that the volume overlap has an upper boundary that depends on the number of features to align. If the current best overlap is larger than this upper bound then there is no need to compute the alignment of smaller subsets since the score will never be larger than the current best.

2.4.4. Alignment scores

Similarity between the reference and database molecules can be calculated using three different measures:

$$TANIMOTO = \frac{V_{overlap}}{V_{ref} + V_{db} - V_{overlap}}$$

$$TVERSKY\_REF = \frac{V_{overlap}}{V_{ref}}$$

$$TVERSKY\_DB = \frac{V_{overlap}}{V_{db}}$$

$V_{overlap}$ is the maximum volume overlap of the two pharmacophores; V_{ref} is the volume of the reference pharmacophores; and V_{db} the volume of the database pharmacophores.

The TANIMOTO measure is well known from bit vector comparison and is the default measure to score similarity between pharmacophores. The TVERSKY_REF measure is primarily intended to identify database compounds having a pharmacophore that is a superset of the reference pharmacophore, while the TVERSKY_DB measure has its use in identifying database compounds having a pharmacophore that is subset of the reference pharmacophore.

All three metrics return a score between 0 and 1.

3. Pharao usage

3.1. General

[OPTIONAL] -h, --help

Help on the use of *Pharao* is provided. This help message corresponds to the table defined in section 4.

[OPTIONAL] -q, --quiet

If this parameter is provided, no output, progress or warnings are written out by the program.

[OPTIONAL] --info <option>

With this option the user can get detailed information for each option listed below. For example to get some information about the **--dbase** option, type:

```
> Pharao --info dbase
```

or,

```
> Pharao --info d
```

3.2. Input

By default the format of input molecule files is determined from the extension of those files. *Pharao* supports all file types that are supported by Open Babel. A pharmacophore file is specified by the '`.phar`' extension.

[OPTIONAL] -r, --reference <file>

This command-line option defines the reference structure that will be used to screen and/or align the database. This option is not required, so when not given then the database will only be converted into pharmacophores without screening.

By default the format is deduced from the extension of the file but this format can be defined explicitly with the **--refType** option.

[OPTIONAL] --refType <MOL|PHAR>

With this option the format of the reference input file can be specified:

- '**MOL**': the reference file is a molecule file. The actual file type is specified by the extension of the molecular reference file. The reference molecule should contain coordinate information.
- '**PHAR**': the reference file is composed of a precalculated pharmacophore specified in a format as described in section 2.1.2.

If the pharmacophore format is not used, the program will generate a pharmacophore for the reference molecule using the procedure as described in section 2.2. The time needed for this generation is negligible compared to the time needed for alignment.

[REQUIRED] -d, --dbase <file>

Defines the database that will be used to screen. This option is required. By default the format is deduced from the extension of the file but can be defined explicitly with the **--dbType** option.

[OPTIONAL] --dbType <MOL|PHAR>

With this option the format of the database input file(s) can be specified:

- '**MOL**': the default format. The database file is a molecular file type. The actual file type is specified by the extension of the molecular database file. The database molecules should contain coordinate information.
- '**PHAR**': the database file is composed of precalculated pharmacophores specified in a format as described in section 2.1.2.

If the pharmacophore format is not used, the program will generate a pharmacophore for each molecule using the procedure described in section 2.2. The time needed for this generation is negligible compared to the time needed for alignment.

3.3. Output

[REQUIRED] -p, --pharmacophore <file>

The aligned pharmacophores of the structures in the input database are written to this file. The spatial position of these pharmacophores will not correspond to the original structures because they are aligned with respect to the reference input molecule and therefore can have a different orientation. Moreover, only the points that are used in the alignment are written out. If there is not a reference structure defined, or no alignment has taken place, then the complete pharmacophore is written out.

This file is written in the pharmacophore format as described in section 2.1.2.

[OPTIONAL] -o, --out <file>

The aligned database structures are written to this file. This file is written in a molecular format as specified by the file extension. Pharao also supports all formats supported by Open Babel.

[OPTIONAL] -s, --scores <file>

With this option a tab-delimited text file can be generated containing the following information:

- **column 1**: Id of the reference structure.

- **column 2**: Maximum volume of the reference structure.
- **column 3**: Id of the database structure.
- **column 4**: Maximum volume of the database structure.
- **column 5**: Maximum volume overlap of the two structures.
- **column 6**: Overlap between pharmacophore and exclusion spheres in the reference.
- **column 7**: Corrected volume overlap between database pharmacophore and reference.
- **column 8**: Number of pharmacophore points in the processed pharmacophore.
- **column 9**: TANIMOTO score.
- **column 10**: TVERSKY_REF score.
- **column 11**: TVERSKY_DB score.

More information about the scores in the last three columns can be found in 2.4.4.

[OPTIONAL] --cutOff <double>

This value should be between 0 and 1 and only structures with a score, as defined by the **--rankby** option, larger than this value are written to the files as defined by the **--out**, **--scores** and **--pharmacophore** options.

[OPTIONAL] --best <int>

With this option only a limited number of best scoring structures, as defined by the **--rankby** option, are reported in the three possible output files. If the **--cutOff** option is also provided, all best scoring structures are first passed through that filter. The user can specify the number of best scoring structures that should be reported.

[OPTIONAL] --rankBy <TANIMOTO|TVERSKY_REF|TVERSKY_DB>

This option defines the scoring used by the previous two options. More information about the three possible metrics can be found in 2.4.4. By default, the TANIMOTO measure is used.

3.4. Options

[OPTIONAL] -f, --funcGroup <AROM|HDON|HACC|LIPO|CHARGE>

By default the generated pharmacophores contain all functional groups and thus include all information that might be useful. With this option only a subset of the available functional groups can be used in the alignment. The user can define this subset by using the tags listed below with the ',' symbol as separator. See section 3.5 for an example.

- '**AROM**': Aromatic rings, see section 2.2.1.

- '**HDON**': Hydrogen bond donors, see section 2.2.2.
- '**HACC**': Hydrogen bond acceptors, see section 2.2.3.
- '**LIPO**': Lipophilic spots, see section 2.2.4.
- '**CHARGE**': Charge centers, see section 2.2.5.

If the reference and database structures are provided in the pharmacophore format then this option is discarded.

[OPTIONAL] -e, --epsilon <double>

This option can be used to change the tolerance for points to be matched in the alignment phase. This is an important parameter to control the feature-mapping phase as described in section 2.4.2.

The lower this value, the more strict the matching between two pharmacophores will be before they are aligned. Higher values imply a higher allowed level of initial mismatching and typically result in larger computing times.

The range of this parameter is between 0 and 1. The default value is 0.5.

[OPTIONAL] -m, --merge

Flag to indicate pharmacophore points will be merged as explained in section 2.2.6.

This flag also activates the **-n** or **--noNormal** flag because merged pharmacophore points cannot have a normal.

[OPTIONAL] -n, --noNormal

Flag to indicate that no normal information is included during the alignment. Using this flag makes the pharmacophore models less specific but also less conformation-dependent.

[OPTIONAL] --noHybrid

Flag to indicate that hybrid points should not be calculated. The list of hybrid pharmacophore points is given in Table 1 and is generated by default to reduce the number of pharmacophore points.

[OPTIONAL] --scoreOnly

Flag to indicate that the poses will be used as provided in the input file. No translational or rotational optimization will be performed. The best score reported is the one from the feasible mapping with the highest volume overlap.

[OPTIONAL] --withExclusion

Flag to indicate if the exclusion spheres should be part of the optimization procedure. By default, the overlap between pharmacophore and exclusion spheres is only taken into account at the end of the alignment

procedure. When this flag is set, the exclusion spheres have also an impact on the optimization procedure.

3.5. Examples

In the first example the task is to simply generate pharmacophores for a number of structures and store them for later use:

```
> pharao --dbase db.sdf --pharmacophore output.phar
```

or shorter

```
> pharao -d db.sdf -p output.phar
```

In the next example a virtual screening is performed. After screening a database against a reference structure, only the ranking based on the TANIMOTO score is of interest:

```
> pharao --reference      ref.phar
         --refType        PHAR
         --dbase          db.phar
         --dbType         PHAR
         --pharmacophore  output.phar
         --scores         result.tab

> sort -k 9 -r result.tab > sortedResult.tab
```

Calculating pharmacophores as in the first example, but without using hydrogen bond donor and acceptor information, is done by typing:

```
> pharao -d db.sdf -p output.phar --funcGroup AROM,LIPO,CHARGE
```

Finally an example is presented whereby a small fragment is used as a reference pharmacophore and the only purpose is to find structures that include this pharmacophore. Only structures with a common overlap covering at least 80% of the reference volume are reported in `output.phar` and `result.tab`. Notice that a ranking is made based on column 10 instead of column 9.

```
> pharao --reference      ref.sdf
         --dbase          db.phar
         --dbType         PHAR
         --pharmacophore  output.phar
         --scores         result.tab
         --cutOff         0.8
         --rankBy         TVERSKY_REF

> sort -k 10 -r res.tab > sortedResult.tab
```

4. Command-line summary

Table 3. Summary of the command-line arguments to *Pharao*

Argument[3]	Default value[4]	Description
General		
[O] -h, --help	N/A	Provides a short description of usage.
[O] --info	N/A	Provides a detailed description for each option.
[O] -q, --quiet	N/A	If this flag is set, minimum output is given to the user during execution of the program.
Input		
[O] -r, --reference	-	Defines the reference molecule or pharmacophore that will be used to screen and/or align a database.
[O] --refType	'MOL'	Indicates the type of the reference data file.
[R] -d, --dbase	-	Defines the database that will be screened and/or aligned.
[O] --dbType	'MOL'	Indicates the type of the database data file.
Output		
[R] --, --pharmacophore	-	File with the computed pharmacophores of the input database.
[O] -o, --out	-	The transformed database molecules after aligning them to the reference pharmacophore.
[O] -s, --scores	-	Tab-delimited text file with for each molecule the number of corresponding pharmacophore points and the overlap scores.
[O] -l, --log	-	Log file of the current run.
[O] --cutOff	0.0	Minimum score for a structure to be reported.
[O] --best	0.0	Only best scoring molecules are reported.
[O] --rankBy	'TANIMOTO'	Define scoring used by `--cutOff` and `--best`.
Options		
[O] -f, --funcGroup	ALL	Flag to define functional groups used in the creation of pharmacophores.
[O] -e, --epsilon	'0.5'	Option to change the tolerance for points to be matched.
[O] -m, --merge	N/A	Flag to merge pharmacophore points.
[O] --noNormal	N/A	Flag to ignore normal information during alignment.
[O] --noHybrid	N/A	Flag to disable the use of hybrid pharmacophore points.
[o] --scoreOnly	N/A	Flag to indicate that the volume overlap should be computed from the given poses and that no translational or rotational optimization should be done.
[O] --withExclusion	N/A	Flag to add exclusion spheres into the optimization process instead of processing them afterwards.

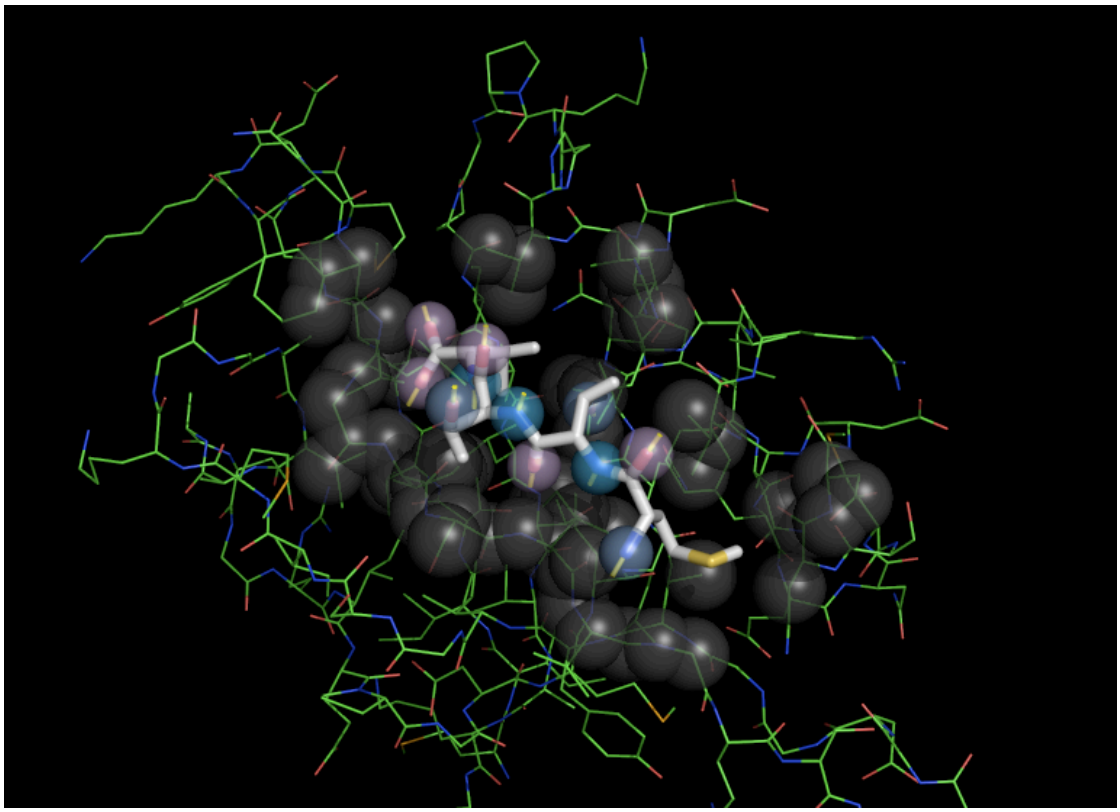
[3] [O] means that the argument is optional, and [R] indicates that the argument is required. Not specifying a required argument will cause the program to quit.

[4] N/A stands for 'not applicable'.

5. PyMOL integration

PyMOL[5] is an open source visualization program well suited to produce high quality images of small molecules and biological macromolecules such as proteins. One of the main advantages of PyMOL is its powerful scripting language. Using the Python script '`pharao.py`' , which can be installed as a plug-in, it is now possible to integrate the *Pharao* functionality into PyMOL.

Figure 7. Example pharmacophore visualization in PyMOL.



5.1. Installation on Mac OS X

In order to install the *Pharao* plug-in under Mac OS X, proceed according the following steps:

- Change the application name 'MacPyMOL.app' into 'PyMOLX11Hybrid.app'.
- Copy '`pharao.py`' to the following location:

  `/Applications/PyMOLX11Hybrid.app/pymol/modules/pmg_tk/startup/`
- Start PyMOL and activate the plug-in by choosing 'Pharao...' from the 'Plugin' menu.

If successful, a new window containing a simple menu will show up.

[5] DeLano Scientific LLC, USA

5.2. Usage

The *Pharao* PyMOL plug-in menu offers four options:

- **Create pharmacophore**

  To create a new pharmacophore the user first has to select a compound with the default selection name (sele). The *Pharao* application should be located at:

  `/usr/local/bin/pharao`

  After executing *Pharao*, all temporary files are removed and the pharmacophore is displayed and saved internally.

- **Read pharmacophore**

  Instead of calculating a pharmacophore it is also possible to read a pharmacophore file. Only files with `.phar` extension are recognized. The pharmacophore is also saved internally.

- **Write pharmacophore**

  Save the last calculated or read pharmacophore in the file format described in section 2.1.2.

- **Create exclusion spheres**

  Exclusion spheres are pharmacophore points and are generated based on the environment of the selected compound. All atoms of the target within a distance smaller than 4.5 Å of the ligand will correspond to a sphere with sigma 0.7 Å. The generated pharmacophore points are saved internally. If there was already a pharmacophore saved, they will be appended to it.

6. Revision history

6.1. Version 3.0

6.1.1. Version 3.0.0

Rewrote code for Open Babel API (OB version 2.3).

6.1.2. Version 3.0.1

Fixed some return values (no impact on results).

Improved output regarding the chosen command line parameters.

Removed the **--noAlign** option since this is automatically invoked when no reference file is provided.

6.1.3. Version 3.0.2

Improved input read coding.

6.2. Version 1.0

Initial release

6.2.1. Version 1.0.2

Added three additional options: **--cutOff**, **--best** and **--rankby**. Using these options it is now possible to reduce the reported information to only relevant structures. Both options can be combined and have an influence on the **-p**, **-o** and **-s** options.

Added option **--info** in order to provide for each command detailed information.

6.2.2. Version 1.0.3

Added **--combConf** option.

Added the PyMOL integration.

6.2.3. Version 1.0.5

Added **--merge** option.

6.2.4. Version 1.0.7

Added the **--noNormal** option.

Extended the information about the two different score schemes in section 2.4.4.

6.2.5. Version 1.0.8

Introduced three metrics: TANIMOTO (instead of SCORE), TVERSKY_REF (instead of VRATIO) and TVERSKY_DB.