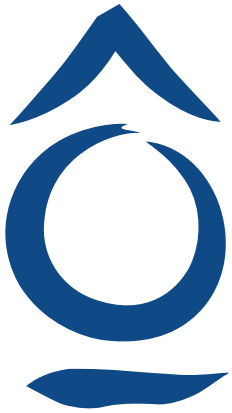

OpenEye
SCIENTIFIC

EON

Release 2.3.0.3

OpenEye Scientific Software, Inc.

December 12, 2018

CONTENTS

CHAPTER

ONE

INTRODUCTION

1.1 Overview

EON is a tool for scoring a database of molecules aligned to a query or template molecule based on similarity in shape and molecular electrostatic potential. The scores are used to rank molecules based on the probability that they share relevant (biological) properties with the query molecule.

EON requires pre-aligned molecules that may be obtained from a variety of sources: overlays from **ROCS**, overlays based on shared 2D features or common substructures, or docking into a protein binding site containing an active ligand, for example.

The minimal input into **EON** is a database of molecules in at least one 3D conformation, where each conformation is aligned to a query molecule from **ROCS** (**ROCS** outputs the query molecule and the alignment of each database molecule to that query). If other alignment methods are used, then the minimal input is a query molecule in at least one 3D conformation and a database of molecules, each in at least one 3D conformation, aligned to that query. The minimal output is a file of the best alignment and scores (shape and electrostatic similarity) for each of the database molecules to the query.

1.2 Applications

The **EON** distribution comprises 1 application:

EON

- Scores pre-aligned molecules in a database file to a query molecule.

CHAPTER

TWO

EON

2.1 EON

2.1.1 Overview

EON calculates the electrostatic similarity between two small molecules in the form of an Electrostatic Tanimoto (ET) score. Given a query molecule and a set of interesting molecules (ROCS overlay hits, for example), **EON** will calculate the Electrostatic Tanimoto between each database molecule and the query. Note that **EON** does not perform any overlay or alter the input orientation of the structures. They must be pre-aligned to the query on input. Also, since electrostatics calculations require high quality partial charges, **EON** will calculate new partial charges for the input structures using MMFF94. If the user provides an input file that contains structures with higher-quality partial charges, EON can use them as well.

EON is also dependent on pKa state and formal charges as these have a significant impact on electrostatics. EON now has the ability to adjust both the query and database molecule to a neutral pH model. This feature is on by default, but can be turned off by using appropriate command line flags.

Since electrostatics overlays are very dependent on alignment and require a good quality alignment between query and database molecule, **ROCS** provides the best input to **EON**. However, electrostatic complementarity is more dependent on subtle conformational changes than shape is, so there are several steps that can be taken to ensure the best possible success with **EON**.

Firstly, one can ensure that **ROCS** outputs multiple interesting conformers per molecule. **ROCS** includes a flag `-eon_input` that allows generation of a multi-conformer set of **ROCS**-aligned output specifically for input into EON. This file can be generated in parallel with a **ROCS** hit list so that in a single **ROCS** run you can find **ROCS** hits and prepare EON input. Please see the **ROCS** documentation for more detail on these flags.

Secondly, **EON** reads one or more conformers from the input file and uses technology from **OMEGA** to expand terminal torsions to search for subtle changes in conformation that might increase the score without changing the overall shape overlap with the query. To score just the input conformers and not search for alternate terminal conformations, the -scoreonly flag is provided.

Part of understanding **EON** results is visualization of the electrostatic grids used in the calculation. Although off by default, when writing **EON** results to a binary (OEB) file, ET grids can be attached to each molecule and visualized using the **EON** View mode in **VIDA**.

Since **EON** calculations can be time-consuming (approximately 1 molecule per second per CPU), EON can use the same distributed computing technology, Open MPI, that **ROCS** uses to help distribute the workload across a cluster of machines.

2.1.2 Example Commands

The simplest way to run **EON** is to use the *-eon_input* flag in **ROCS** to create a set of **ROCS**-aligned structures with the **ROCS** query at the top of the file. By default, if **EON** is only provided a dbase file, it will assume the first molecule is the query.

```
prompt> eon -dbase rocs_eon_input.oeb.gz
```

Or you can provide a single molecule in a query file and a set of molecules in a dbase file. A common example would be to use a **ROCS** query as the EON query and the **ROCS** hits file as the dbase file for **EON**.

So for example:

```
prompt> eon -dbase rocs_hits_1.oeb -query rocsquery.sdf
```

will score all the structures in `rocs_hits_1.oeb` against the molecule in `rocsquery.sdf` and place the structures in `EON_hits.oeb` (with ET scores in SD tag data) and a table of results in `EON.rpt`.

To keep only the best 100 hits, use:

```
prompt> eon -dbase rocs_hits_1.oeb -query rocsquery.sdf -besthits 100
```

Note that the use of an output structure file is mostly for generation of a single file containing structures and tag data such that loading this single file into **VIDA** provides easy analysis of the results. If however, the only real desire is for the numerical scores, use *-nostructs* to suppress the creation of an output structure file.

```
prompt> eon -dbase rocs_hits_1.oeb -query rocsquery.sdf -nostructs
```

By default, **EON** calculates partial charges using MMFF94. However, if you have structure files that already contain good partial charges in both your dbase file and query file, you can tell **EON** to use them instead:

```
prompt> eon -dbase rocs_hits_chgs.mol2
-query rocsquery_chg.mol2 -charges existing
```

To prevent continually over-writing output files, the *-prefix* flag allows you to give unique names to these files.

```
prompt> eon -dbase rocs_hits_1.oeb -query rocsquery.sdf -prefix FOO
```

will write the hit structures into a file named `FOO_hits.oeb` and the numerical values will be in `FOO.rpt`. The parameter file for this run will likewise be named `FOO.parm`.

To prevent EON for searching alternate terminal torsions, use the *-scoreonly* flag.

```
prompt> eon -dbase rocs_eon_input.oeb.gz -scoreonly
```

Finally, to create ET grids as used in the calculation and attach them to each output molecule. Note, this only works for OEB output and can be very memory intensive.

```
prompt> eon -dbase rocs_eon_input.oeb.gz -oformat oeb.gz -writegrid
```

2.1.3 Report File

The **EON** report file format appears as a tab-delimited file with the following fields. Since the names of the query and the hits are of indeterminate length, fixed size fields for these names could result in loss of information. Unfortunately this gives a file that is hard to read in a terminal session, but it can easily be read into a spreadsheet program or into the data manager in **VIDA**.

Name This is the name of the database molecule.

EONQuery This is the name of the query molecule.

Rank The numerical ranking in the hitlist, based on the chosen score to rank by. Using the defaults, this is **ET_combo**. Can be altered by using *-rankby* command line switches. If no hitlist was used in the calculation, this field will be 0 (zero).

ET_pb This is the value of electrostatic Tanimoto, using full Poisson-Boltzmann (PB) electrostatics.

ET_coul This is the value of electrostatic Tanimoto using only the coulombic part of PB electrostatics.

ET_combo Sum of **ET_pb** and **EON_shape_tani**. This is a useful score that takes into account both shape match and ET match.

EON_shape_tani This the shape Tanimoto between the given molecule and the query. For calculations that use *-shapeonly*, this will be the same as the output Tanimoto from **ROCS**. When EON is allowed to alter terminal torsion, this will give the final shape Tanimoto.

2.1.4 Command Line Help

A description of the commandline interface can be obtained by executing **EON** with the *--help* option.

```
prompt> eon --help
```

will generate the following output:

```
Help functions:
  eon --help simple      : Get a list of simple parameters
  eon --help all         : Get a complete list of parameters
  eon --help <parameter> : Get detailed help on a parameter
  eon --help html        : Create an html help file for this program
```

2.1.5 Required Parameters

-dbase <filename>
File containing one or more 3D molecules to score against a reference or query molecule. If only this flag is given, EON will use the first molecule in the dbase file as the query and score all the remaining molecules against it. This is most useful when scoring ROCS results when ROCS was run with *-eon_input* equal to **true** since the ROCS query and therefore EON query will be the first in the input file. The query or reference molecule can also be specified separately using the *-query* flag below.

File format for *-dbase* can be one of:

File type	Extension
OEBinary	.oeb .oeb.gz
SDF	.sdf .mol .sdf.gz .mol.gz
MOL2	.mol2 .mol2.gz
PDB	.pdb .ent .pdb.gz .ent.gz
MacroModel	.mmod .mmod.gz

2.1.6 Optional Parameters

Execute Options

-param
The argument for this flag is the name of a file containing control parameters. The control parameter file acts to either replace or augment the command line interface. All parameters necessary for program execution may be provided in the control parameter file, although any command given explicitly on the command line will

supersede options found in the parameter file. The application generates a new parameter file containing the full set of execution parameters upon every execution. The name of the parameter file is created by combining the prefix base name with the *'.param'* extension.

-mpi_np <n>
Specifies the number of processors `n` when the application is run in MPI mode.

-mpi_hostfile <filename>
Specifies the name of the file containing processors configuration. For every host this file should contain a line `host_name slots=n` where `n` is the number of processors on the host.

Input Options

-query <filename>
File containing one 3D molecule to use as a query. File format can be any of the formats given in the table for *-dbase* above.

-charges
Specifies charges to be used on the query and the database. Default is to calculate them internally with `mmff`. The option `-charges existing` will use precalculated charges that must be set in the input files. If `existing` is selected and any molecule doesn't have charges then EON will default to `mmff` for those molecules.

[default = mmff]

-scdbase
Since EON reads multi-conformer input molecules, when reading from non-binary files, EON will compare consecutive molecules and if they are determined to be the same structure, they will be concatenated into a single, multi-conformer molecule. If the user desires to score each input conformation independently, then using this flag will turn off the conformer comparison step.

[default = false]

Output Options

-oformat <extension>
Format of output structure file(s). The default is **oeb** so that ET score can be included as tag data. The format for the file is determined by giving the extension. Valid values include all formats listed above for *-query*

[default = oeb]

-prefix <prefix>
Defines a prefix used to name output files. Using `-prefix FOO` will create a hits structure file named like `FOO_hits.oeb` and a report file, `FOO.rpt`

[default = EON]

-besthits <N>
Process entire dbase file but only keep **N** best scores sorted by property specified by *-rankby*. Using a value of **0** implies no hitlist will be maintained and structures will all be scored and output in input order.

[default = 500]

-maxhits <N>
Stop after finding first **N** hits. This option overrides any setting for *-besthits*

[default=0]

-rankby <score>
Property to use to sort hitlist. Values include **ET_combo**, **ET_pb** and **ET_coul**.

[default = ET_combo]

-cutoff <score>
Minimum score to keep as a hit.

[default = -1.0]

-outputquery
Write the query to the top of the hits file. This make visualization of results much easier inside **VIDA**.

-scoreonly
Turn off the terminal torsion conformer search and just score each input conformation as-is.

-hitsfile <filename>
Explicit filename for writing hits. Overrides the default filename created from *-prefix*.

-reportfile <filename>
Explicit filename for writing hits. Overrides the default filename created from *-prefix*.

-sdTags
This parameter controls whether to attach score information to output molecules as SD data.

Log Output Options

-logfile <filename>
Filename for log file. Overrides log filename created from *-prefix*.

-progress
Method for showing job progress on the command line. Choices include:

- percent - show a percent complete progress bar (DEFAULT)
- log - echo the log message for each molecule
- dots - show dots as in EON 1.1
- none - print nothing to console

[default = percent]

-statusfile
Write status info to this file. Use "none" for no status file.

-verbose
Give verbose output to console instead of simple progress.

[default = false]

ZAP/PB Options

-fixpka_query
Apply a neutral pH model to the query molecule.

[default = true]

-fixpka_dbase
Apply a neutral pH model to the database molecules.

[default = true]

-salt
Add salt to the Zap calculation. To aid in moderating large, local charges, salt is added into the calculation. Legal values are between 0.0 and 0.1 (mM).

[default = 0.04]

-spacing
Sets the grid spacing for the internal Zap calculations.

[default 0.5]

-writegrid
Write ET grid to output attached to each molecule. Useful for visualization in VIDA but this only works when writing hits to an OEB (.oeb or .oeb.gz) file. Note that while this feature is quite useful, grids do take a large amount of memory so care should be taken when using this feature for hit lists of more than 500 molecules.

[default = false]

Omega Options

-ewindow
Omega energy window used for conformer selection.

[default = 10.0]

-rms
Omega RMS threshold used to determine duplicate conformations

[default = 0.3]

-sampleHydrogens
Sets whether hydrogens will be sampled. This option enables sampling of hydrogen locations for -OH, -SH, and amines.

[default = false]

CHAPTER

THREE

THEORY

3.1 Theory

EON uses a field-based measure of Tanimoto to compare the electrostatic potential of two small molecules. This electrostatic potential is calculated internally using Zap, OpenEye's Poisson-Boltzmann (PB) electrostatics toolkit.

The basic equation for a field Tanimoto is:

$$Tanimoto_{A,B} = \frac{\int A(\vec{r}) * B(\vec{r})}{\int A(\vec{r}) * A(\vec{r}) + \int B(\vec{r}) * B(\vec{r}) - \int A(\vec{r}) * B(\vec{r})}$$

The two boundary cases for Electrostatic Tanimoto occur when B = A:

$$\begin{aligned} Tanimoto &= \frac{\int A(\vec{r}) * A(\vec{r})}{\int A(\vec{r}) * A(\vec{r}) + \int A(\vec{r}) * A(\vec{r}) - \int A(\vec{r}) * A(\vec{r})} \\ &= 1 \end{aligned}$$

and the opposite case, when B = -A:

$$\begin{aligned} Tanimoto &= \frac{\int A(\vec{r}) * -A(\vec{r})}{\int A(\vec{r}) * A(\vec{r}) + \int -A(\vec{r}) * -A(\vec{r}) - \int A(\vec{r}) * -A(\vec{r})} \\ Tanimoto &= \frac{-\int A(\vec{r}) * A(\vec{r})}{\int A(\vec{r}) * A(\vec{r}) + \int A(\vec{r}) * A(\vec{r}) + \int A(\vec{r}) * A(\vec{r})} \\ &= -\frac{1}{3} \end{aligned}$$

In EON, we report two different Electrostatic Tanimoto(ET) measures, based on the outer dielectric used in the PB calculation. ET_pb uses an outer dielectric of 80, while ET_coul uses a value of 2.0. The rationale for using a PB electrostatic field is that the external potential is dampened by orientation of the aqueous solvent. It is a common observation that proteins essentially act to reproduce the aqueous desolvation of well-bound ligands. As a result a PB electrostatic field is more likely to correctly capture the essential elements of binding than that from the Coulombic field. However, this would still seem to be a point to be proven. As such we provide both Tanimotos. They typically track each other very closely.

For hit list ranking, we also report a score (ET_combo) that is the sum of the Shape Tanimoto (ST) and the PB Electrostatic Tanimoto (ET_pb).

CHAPTER

FOUR

RELEASE NOTES

4.1 Release Notes

4.1.1 EON 2.3.0

November 2018

- This version of **EON** has been built using **OEToolkits 2018.Oct**. The previous version was built using **OEToolkits 2012.Oct**.

4.1.2 EON 2.2.0

Released June 2013

New features

- The default hitlist format has been changed from sdf to oeb for increased functionality and decreased filesize. The output format is adjustable with the *-oformat* parameter.
- The *-spacing* parameter has been added. This parameter can be used to adjust the grid spacing of the Zap/PB calculations.
- An option has been added to allow hydrogen atoms in -OH, -SH, and amines to take part in conformational sampling. This new option can be enabled via the *-sampleHydrogens* parameter. By default, hydrogen atoms are not sampled.
- Now SD tags are prefixed with *EON_*. The tags are optional with the *-sdTags* parameter. Additionally, any existing ROCS tags will not be removed because ROCS and EON tags no longer conflict.

Bug fixes

- A bug has been fixed where the multiconformer check was being applied to OEB format files. Additionally, the default for *-scdbase* has been changed from *false* to *true* to prevent inadvertent merging of molecules with other file formats.
- Warnings have been improved for charging failures.
- *-writegrid* will now write out a grid with the same spacing as the actual Zap/PB calculation. Previously the attached grid had tighter spacing which could lead to confusion.

Other changes

- PVM (parallel virtual machine) is no longer supported. OpenMPI version 1.6 is supported on all platforms. The *-mpi_np* and *-mpi_hostfile* flags are now used to run EON in MPI mode. These new flags replace the *oempirun* script.
- This will be the last release to support SuSe 10.

4.1.3 EON 2.1.0

Released June 2011

Enhancements

- The command line flags for choosing which charges to use have been simplified. There is a single flag `-charges`. The default is still to calculate MMFF charges. To use charges you have precalculated outside EON, first make sure you have charges for both the query and the dbase molecules, then use `-charges existing`.

  Note: To use `existing` charges you must provide input files in a format that can store charges. This means input should be in either OEB or MOL2.

- This release adds EON to the set of applications that now use a common script in `openeye/bin` to determine the appropriate architecture and run the appropriate binary.
- On Windows, there is now an installer that installs the documentation and sets up a command prompt to facilitate running the EON command line.

Bug Fixes

- Fixed a bug to ensure that SD data present in the database is passed on to the EON hitlist. Note that this is specific to SD data other than ROCS scores. Since EON can modify the conformation, any ROCS scores would no longer be valid so they are removed from the output. Note that the score reported as EON_ShapeTanimoto is the final Shape Tanimoto calculated from the best conformation found inside EON.
- Fixed a bug that caused a crash when trying to write an empty hitlist.
- Fixed a bug that prevented `-salt` from working.
- Adding better error checking and messages for input. All input must be molecules and cannot be shape queries.

4.1.4 EON 2.0.0

Release August 2007

Enhancements

This is a significant new release of EON. This release incorporates many internal changes in how Electrostatic Tanimoto is calculated compared to EON 1.1.

- Salt in the PB calculation is now on by default to help moderate large, local charges. This should make EON 2.0 perform much better than 1.1 in the case of molecules with formal charges.

- OpenEye's neutral pH model has been incorporated allowing the setting of a pH model on both the query and database molecules. Since a significant use of EON is similarity searching, having a consistent model between the query and database molecules is very important. However, for the case where the states are well defined, this feature can be turned off.
- Instead of the slow and buggy `-spin` model for searching terminal torsions, EON 2.0 uses technology from OMEGA to only search reasonable torsions and to prevent creation of bad structures while searching for better electrostatic overlap.
- The hitlist is now sorted by default by a new score, ET_combo, which is the sum of the Shape Tanimoto (ST) and the Electrostatic Tanimoto (ET_pb).
- To make visualization of results easier, EON can write out the actual ET grids to the hits file, attached to each hit. This file can be loaded into VIDA 3.0 or later and the grids can be visually compared.
- There are also several changes to the mechanics of running the calculation including a new *`-progress`* feature and a more detailed PVM log output for PVM jobs.

CHAPTER

FIVE

CITATION

5.1 Citation

Note: To cite EON please use the following:

EON 2.3.0.3: OpenEye Scientific Software, Santa Fe, NM. http://www.eyesopen.com.

Symbols

E