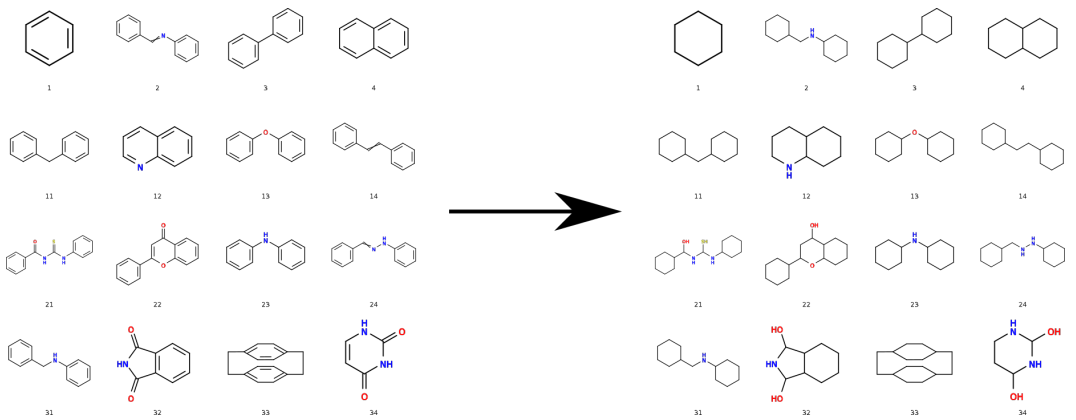
1 Background

The Bemis-Murcko scaffold[1] provided by `DataWarrior`[2] retains information about bond order and chirality. Sometimes, however, it suffices to retain only atom connectivity, like an assumption «there are only single bonds». Note `DataWarrior` equally offers the export of Bemis-Murcko skeleton, however this simplifies e.g. the scaffold about an imidazole into one of cyclopentane.



2 Typical use

The script runs from Python's CLI with a file listing SMILES to process as parameter. File `test_input.smi` (from sub-folder `test_data`) is an example:

```
python saturate_murcko_scaffolds.py [test_input.smi]
```

This generates `test_input_sat.smi` as permanent record; the addition of `_sat` is only a reminder of the performed saturation. The input file is preserved.

The file extension `.smi` of the input file is a suggestion, because it is frequently seen (e.g., around `OpenBabel`[3]). Internally, the script considers any character prior to the first period as part of the name of the input file. The name of the report file is a mere concatenation of this and the string `_sat.smi`.

While written for current branch of Python 3.6+ (e.g., Python 3.9.1), the script equally works well with the legacy branch of Python 2 like Python 2.7.16.

[1] Bemis GW, Murcko MA *J. Med. Chem.* 1996, **39**, 2887-2893, doi 10.1021/jm9602928.

[2] Sander T, Freyss J, von Korff M, Rufener C, *J. Chem. Inf. Model.* 2015, **55**, 460-473, doi 10.1021/ci500588j. The program, (c) 2002–2021 by Idorsia Pharmaceuticals Ltd., is freely available under `http://www.openmolecules.org`. For the source code (GPLv3), see `https://github.com/thsa/datawarrior`.

[3] www.openbabel.org. The script initially was developed for and tested with OpenBabel (release 2.4.1; Nov 12, 2018) and Python 2.7.17 provided by Linux Xubuntu 18.04.2 LTS. It equally works with Python 3.9.1+ (released January 20, 2021) and OpenBabel (release 3.1.1 by January 6, 2021) as provided in Debian 10.

3 Example

For a collection of organic materials, the Bemis-Murcko scaffolds were extracted with `DataWarrior` (then release 5.0.0 for Linux, January 2019) as listing `test_input.smi` including higher bond orders (see folder `test_data`). The effect of the «artificial saturation» is easy to recognize while comparing the scaffold lists (fig. 1) in a difference view of the two `.smi` files.

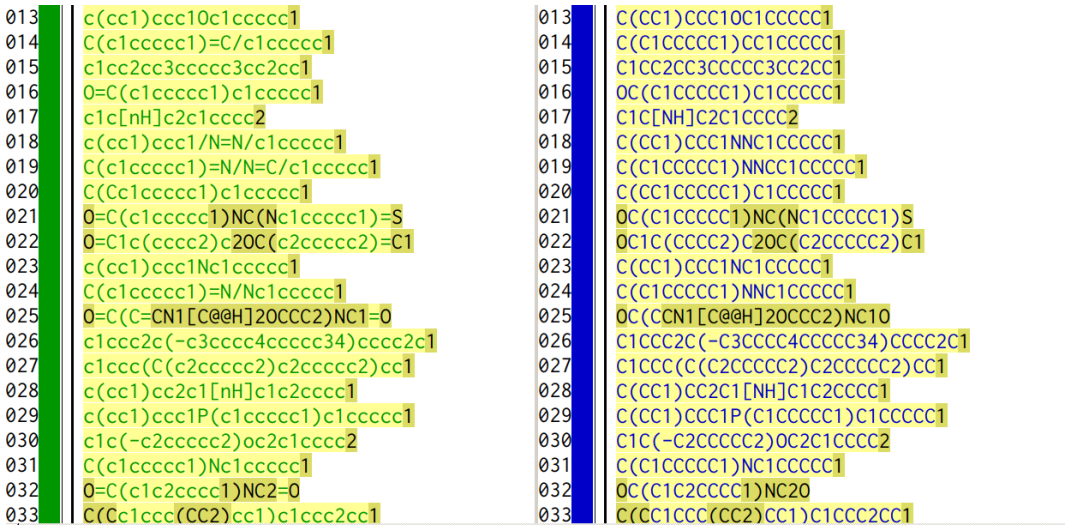


Figure 1: Difference view of the SMILES strings of a Murcko scaffold *prior* (left hand column) and *after* an «artificial saturation» (right hand column). The processing affects explicit bond order indicators, e.g. double bond (equality sign, e.g., line #14), triple bond bond (octohorpe, not shown); or about implicit aromatization (lower case → upper case) for atoms of carbon, nitrogen, oxygen (depicted); or phosphorus, sulfur (not depicted). Stereochemical indicators about double bonds will be removed (e.g., slashes in lines #18 and #19). Descriptors of stereogenic centers (@-signs, e.g., line #25) are copied verbatim.

Subsequently, `OpenBabel`[3] was used to illustrate the work performed. While eventually automated (cf. script `test_series.py`, deposit in folder `test_data`), instructions issued to `OpenBabel` on the command line followed the pattern of

```
obabel -ismi test_input.smi -O test_input_color.svg -xc10 -xr12 -xl --addinindex
```

to generate a `.svg` file (vector representation), or

```
obabel -ismi test_input_sat.smi -O test_input_sat_color.png -xc10 -xr12 -xl
↪  --addinindex -xp 3000
```

to generate a bitmap `.png` with structure formulae depicted in a grid of 10 columns by 12 rows.

It is remarkable how well `OpenBabel`'s displays the molecular structures with advanced motifs. In addition to those shown in the first illustration of this guide, see sub-folder `test_data` for a more extensive survey (e.g., the scaffold of cyclophane [entry #33], sparteine [#38], or adamantane [#50]).

4 Known peculiarities

The script neither removes, nor newly assigns SMILES descriptors about the absolute configuration of stereogenic centers (`@`). Thus, the «reduction» of double bonds e.g., ketones to secondary alcohols may yield new stereogenic centers with a description incomplete in this regard.

To resolve implicitly described aromatic systems, the script capitalizes the characters `c`, `n`, `o`, `p`, and `s` about the elements more frequently involved in ring formation. To avoid ambiguity processing data about tin – typically described by SMILES strings as `[Sn]`, which the algorithm would convert into `[SN]` – *any* SMILES string including either the pattern of `[Sn]` or `[sn]` is excluded from the reduction. This rule is applied applied to both tin analogues of benzene, e.g. `c1[sn]ccccc1`, and SMILES strings describing tin in a side chain (e.g., about a Stille reagent). Script `saturate_murcko_scaffolds.py` annotates these SMILES strings accordingly in the output file.

The script will not actively alter a charge assigned to an atom. If present (e.g., quaternary ammonium, carboxylate), this information will be carried over to the newly written SMILES string. Given the reduction of bond orders, depending on the substrate submitted, this approach may be sensible (e.g., about N in cetyltrimethylammonium bromide), or not (e.g., about N in pyridine *N*-oxide). Other libraries than the current script (e.g., RDKit[4]) might offer help to sanitize the processed SMILES strings.

If the input SMILES string describes more than exactly one molecule by the concatenating `"."` (period character), this special sign equally is the newly written SMILES string. This permits working with SMILES about e.g., co-crystals, like about 1,4-benzoquinone and hydroquinone, `C1=CC(=O)C=CC1=O.c1cc(ccc1O)O`.

5 License

Norwid Behrnd, 2019–21, GPLv3.

[4]For an overview about the freely available RDKit library, see www.rdkit.org. An introduction into the topic of «molecular sanitization» is provided in the section of this very title in the on-line RDKit Book.