

SMOG-SERVER.ORG

**THE SMOG
USER GUIDE**

EXCERPT ON USING SCM.jar

Jeff Noel

info@smog-server.org

[HTTP://SMOG-SERVER.ORG/SHADOW.HTML](http://smog-server.org/shadow.html)

AUGUST 25, 2012

Contents

1	Shadow contact map analysis: <code>SCM.jar</code>	2
1.1	Introduction	2
1.2	Downloading <code>SCM.jar</code>	3
1.3	Citing <code>SCM.jar</code>	3
1.4	Running <code>SCM.jar</code>	3
1.4.1	Some examples	4
1.4.2	Full configuration parameter list	5
1.4.3	Running <code>SCM.jar</code> through the webtool	5

1 Shadow contact map analysis: `SCM.jar`

1.1 Introduction

This section describes a Java application `SCM.jar` that computes the “Shadow” map, a general contact definition for capturing the dynamics of biomolecular folding and function. It is described in the literature here [1]. A contact map is a binary symmetric matrix that encodes which atom pairs are given attractive interactions in the SBM potential. In the context of a SBM, the native contact map should approximate the distribution of stabilizing enthalpy in the native state that is provided by short range interactions like van der Waals forces, hydrogen bonding, and salt bridges. Any long range interactions or nonlocal effects are taken into account in a mean field way through the native bias.

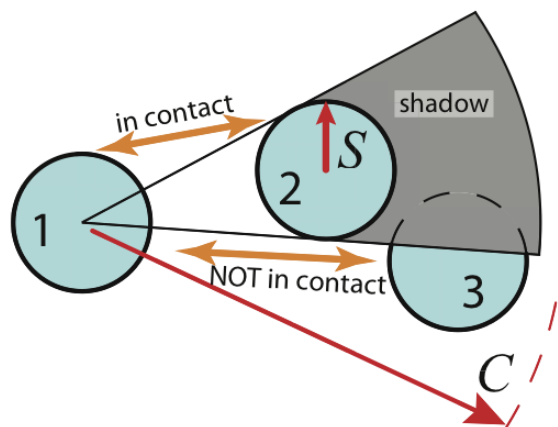


Figure 1: The Shadow contact map screening geometry. Only atoms within the cutoff distance C are considered. Atoms 1 and 2 are in contact because they are within C and have no intervening atom. To check if atoms 1 and 3 are in contact, one checks if atom 2 shadows atom 1 from atom 3. The three atoms are viewed in the plane, and all atoms are given the same shadowing radius S . Since a light shining from the center of atom 1 causes a shadow to be cast on atom 3, atoms 1 and 3 are not in contact.

1.2 Downloading `SCM.jar`

In the near future `SCM.jar` will be available at `smog-server.org`. There are no plans to make the source code publicly available. Any feature requests or source code requests should be directed to `info@smog-server.org`.

1.3 Citing `SCM.jar`

Most importantly cite [1], but since `SCM.jar` is part of the SMOG group of tools, one can also please cite [2].

1.4 Running `SCM.jar`

Like any java application, no compilation is necessary, but a virtual machine is required; `SCM.jar` requires JRE 1.6.0_29 or greater. `SCM.jar` reads SMOG formatted Gromacs input files. **Important! The all (heavy) atom geometry must be used**, even if the output will be a coarse-grained residue-based map for a C_α model. The atomic coordinates are read in `.gro` format and the bond connectivity is read via a `.top` obtained from the SMOG webtool (or source distribution). The topology is required since bonded atoms shadow each other differently and since contacts are automatically discarded between two atoms if they

share a bonded interaction. At the command line, the basic syntax is

```
user$ java [-Xmx1000m] -jar SCM.jar -g grofile -t topfile -o outputName [--chain chainFile] \  
          [--default | -m {shadow,cutoff}]
```

`-Xmx1000m` assigns 1000 MB of RAM to the Java virtual machine heap. With large complexes (>1e5 atoms) the default heap allocation can run out which gives the following error:

```
java.lang.OutOfMemoryError: Java heap space
```

The output all-atom contact file format is

```
chain_i atom_i chain_j atom_j [distance]
```

and similarly, the output residue contact file format is

```
chain_i residue_i chain_j residue_j [distance]
```

1.4.1 Some examples

- Shadow map, atomic contacts, shadowing radius 1 Å and cutoff 6 Å (default sizes). See Figure 1 for definition of radius and cutoff. Add `-chain` if you have multiple chains, since the `.gro` format does not allow for chain information. Specify the chains file you get from your SMOG output.

```
user$ java -jar SCM.jar -g protein.gro -t protein.top -o contactsOut --default [--chain chainsFile]
```

- Shadow map, atomic contacts, shadowing radius 2 Å and cutoff 4 Å.

```
user$ java -jar SCM.jar -g protein.gro -t protein.top -o contactsOut -m shadow -c 4 -s 2
```

- Cutoff map, atomic contacts, and cutoff 4 Å.

```
user$ java -jar SCM.jar -g protein.gro -t protein.top -o contactsOut -m cutoff -c 4
```

- OR -

```
user$ java -jar SCM.jar -g protein.gro -t protein.top -o contactsOut -s shadow -s 0 -c 4
```

- Shadow map, residue contacts, default, include contact distances

```
user$ java -jar SCM.jar -g protein.gro -t protein.top -o contactsOut \  
      --distances --coarse CA
```

- To calculate over a trajectory instead of a single structure, use `-multiple X`, where `X` is the number of frames in the trajectory `.gro` file. Assumes that the format of `proteinTraj.gro` is the same as the output of `trjconv`. This saves time relative to looping over many grofiles because the topology (and therefore the bond list) is only created once.

```
user$ $GROMACS/trjconv -f traj.xtc -o proteinTraj.gro
```

```
user$ java -jar SCM.jar -g proteinTraj.gro -t protein.top -o contactMapsOut --default --multiple 1000
```

1.4.2 Full configuration parameter list

The following will give a full list of configuration options:

```
user$ java -jar SCM.jar -help
```

1.4.3 Running SCM.jar through the webtool

On the webserver (<http://smog-server.org/Shadow.html>) one can build a shadow map from a SMOG formatted PDB file.

References

- [1] Noel, J. K, Whitford, P. C, & Onuchic, J. N. (2012) The shadow map: a general contact definition for capturing the dynamics of biomolecular folding and function. *J. Phys. Chem. B* **116**, 8692–8702.
- [2] Noel, J. K, Whitford, P. C, Sanbonmatsu, K. Y, & Onuchic, J. N. (2010) Smog@ctbp: simplified deployment of structure-based models in gromacs. *Nucleic Acids Res.* **38**, W657–61.