Requirements for ISIM graphical user interface
version 0.10

Contents

1 Purpose

This document describes the function of a proposed graphical user interface to Mark Englehardt's modified version of the ISIM program.

2 Introduction

ISIM, for Ion SIMulator, is a computer program for computation of the ionic environment in the vicinity of macromolecules. ISIM was created in Nathan Baker's laboratory at Washington University in St. Louis. Mark Engelhardt at Stanford has created a modified version of ISIM. This document describes a proposed graphical user interface to this modified version of ISIM.

3 Functional Requirements

3.1 Outstanding issues that need to be resolved for the SimTK version 1.0 version of this program:

3.1.1 Application name

3.1.2 License issues

SimTK.org software should in general be distributed under a rather unrestrictive BSD-style license.

MALOC, APBS, and their associated libraries are distributed under the General Public License (GPL, or GNU Public License). ISIM, which links to these libraries might, therefore, also be subject to the GPL. Future derivatives of ISIM might also be subject to GPL terms.

It may be possible to distribute an interface *wrapper* around ISIM under a BSD-style license, while distributing ISIM itself under a GPL license.

In any case, the exact license situation needs to be clarified with Nathan Baker, author of ISIM, and possibly with Holst and McCammon.

In any case, the various licenses must be viewable in the user interface.

3.2 User interface

The user must be able to view parameters that will be used in the simulation, including temperature, dielectric constant of the solvent, and dielectric constant of the macromolecule.

3.2.1 Progress

The user must have available a progress monitor, including the following:

- progress bar, showing approximate fraction completed
    - update this a frequently as is reasonable
- estimated time remaining
- elapsed time since start
- Display of the steps completed, current step, and future steps not completed, including
    - calculate field of macromolecule alone (APBS)
    - calculate field in the presence of ions

3.2.2 Job control

The interface will permit the user to adjust the CPU resources used by the program. The options will include: raise priority, lower priority, max priority, min priority.

The user will also be able to save the current state, for continuation at a later time ("stop for now"). There should also be an autosave process, for restarting after a crash.

When there is pending activity, a "continue" function must also be available. It should be possible to modify the total number of steps at this point. Changing the total number of steps will immediately change the estimated total and remaining running time.

3.2.3 Input

3.2.3.1 Macromolecular structure(s)

3.2.3.2 Ion types and concentrations

The user must be able to select from a predefined list of ion sets, for example "5mM Mg++, 10mM Na+, 20 mM Cl-" might constitute one set of ions.

The user should be able to activate documentation about how each ion type was defined. For example, "hard sphere model, 3 Angstroms radius, -1 charge".

A sister application should be provided to create parameters for new ion sets.

3.2.3.3 Options

All option must have reasonable defaults, and explanation of why those defaults were chosen in the documentation.

3.2.3.3.1 Number of steps

3.2.3.3.2 How often to save (stride size)

3.2.3.3.3 Grid spacing

3.2.4 Output

3.2.4.1 Number of ions

Compare the number of ions in the solvent volume to the expected number in the absence of the macromolecule.

3.2.4.2 Electrostatic potential

Permit the user to use output grid spacings other than those that were actually used in the calculation.

3.2.4.3 Coordinate files of ions ensembles

This is a large number of files. The user must be warned about how many files she will be dealing with. The user will have the option to save these files to another directory.

4 Documentation required

4.1 User manual

4.2 Installation guide

4.3 Tutorial example

4.4 How to create new ion sets

5 References

APBS (Adaptive Poisson Boltzman Solver)

ISIM: Software for the Simulation of Ions in the Grand Canonical Ensemble

6 Acknowledgments

This work was funded by the National Institutes of Health through the NIH Roadmap for Medical Research Grant U54 GM072970.
Information on the National Centers for Biomedical Computing can be obtained from http://nihroadmap.nih.gov/bioinformatics.