User Manual

Multiscale Cardiac cell Workflow

Table of Contents

Chapter 1: Introduction and Tool Description

We present an automated computational workflow (Figure 1) that can perform simulations to generate user defined instances and configurations of a single-cell cardiac action potential, conduction of a cardiac action potential in a 1-dimensional (1D) or 2-dimensional (2D) tissue representation and generation of a signal average of electrical activity in time and space to represent a pseudo-ECG.

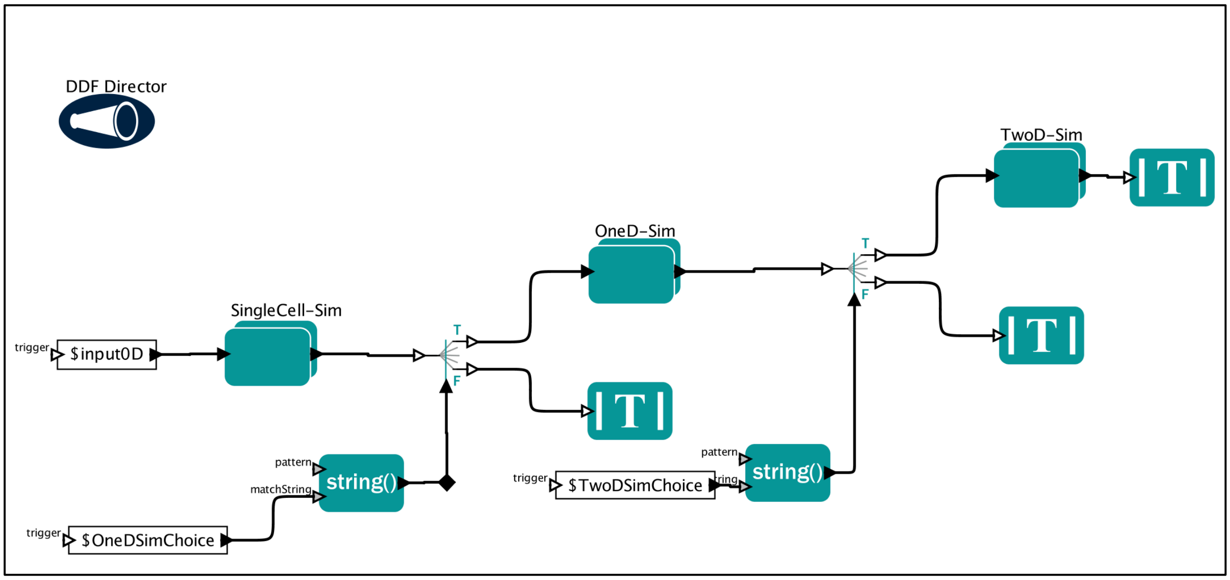

Figure 1: Multiscale Cardiac Cell Workflow

Chapter 2: Installation and Program Requirement

The MD workflow requires users to have Kepler, bioKepler 1.2, a compiler and all necessary input scripts installed on their local machines (2-4). Users can find more information about Kepler (https://kepler-project.org/) and bioKepler 1.2 at (https://www.biokepler.org/install-biokepler-1.2). Intel C++ compiler is used when developing this workflow and the conversion of source codes. For more information, please check out Intel developer website (https://software.intel.com/en-us/intel-compilers).

Installing Kepler:

Users can download the latest Kepler software from the following link:
https://kepler-project.org/users/downloads

To install on **Windows**, double click on the executable Kepler-2.5-win.exe.
To install on **Mac OS X**, mount the Kepler-2.5.dmg by double-clicking on it and dragging-and-dropping the Kepler-2.5 folder into the /Applications folder.
To install on **Linux**, simply untar Kepler-2.5-linux.tar.gz at a location of your choice, which will generate a folder named Kepler-2.5. This is the command line command to untar a tar.gz file:

```
tar -C /[myfolder] -zxvf Kepler-2.5-linux.tar.gz
```

For the following tutorial, we will work on a Linux platform as an example. Windows and Mac OS X installations follow similar steps.

Chapter 3: Downloading the Multiscale Cardiac Cell workflow

The Multiscale Cardiac workflow is prepared to run on the Kepler platform. The workflow itself is a product developed by NBCR (http://nbcr.ucsd.edu) and WorDS (http://words.sdsc.edu/). The Multiscale Cardiac Cell workflow, this manual, and sample inputs can be downloaded from github or from the following link: https://github.com/ClancyLabUCD/Workflow_Kepler

Chapter 4: Step-by-Step Tutorial

1. Setting up input folders:

Before running the workflow, you need to set up and organize input folders in a specific required format. Inside the input directory, there are two required sub-directories: “Single_Cell” and “1D2D” that contain source code, dependent code files and input file. Users can also find the sample input folder here (LINK HERE).

1D2D:	Single_Cell:
ECG_1D.py	stim_param.txt
README.docx	README.md
README.md	README.txt
snapshot2D.py	SS_rabbit_varNames.txt
initial_WTstates.txt	initial_WTstates.txt
integrate_rk2.h	integrate_rk2.h
masterCompute.cpp	masterCompute.cpp
masterCompute_1D.h	plotall.py
masterCompute_2D.h	soltis_biophysJ2010_BARsignalling_odefile.h
soltis_biophysJ2010_BARsignalling_odefile.h	soltis_biophysJ2010_CaMKII_signaling_ODEfile.h
soltis_biophysJ2010_CaMKII_signaling_ODEfile.h	soltis_biophysJ2010_camODEfile.h
soltis_biophysJ2010_camODEfile.h	soltis_biophysJ2010_eccODEfile.h
soltis_biophysJ2010_eccODEfile.h	soltis_biophysJ2010_eccODEfile_old.h
soltis_biophysJ2010_masterODEfile.h	soltis_biophysJ2010_masterODEfile.h
spiralwave_mov_2D.m	MSWF-KeyValueParser.py
stim_param.txt	

Download the files and create a new directory.

```
mkdir Workflow_Sample
```

Then, move the zip file into the new directory.

```
mv ~/Downloads/Input_Files.zip Workflow_Sample
```

Change directory into the MD_TEST folder, unzip the Input_Files.zip file and then remove it from the directory.

```
cd Workflow_Sample
unzip Input_Files.zip
rm Input_Files.zip
```

Two folders named DBP-Clancy-rabbit-1D2D and DBP-Clancy-Rabbit-Single-Cell will now appear in the empty directory.

Now, you are ready to run the workflow. There are three modes of execution and we will go over each of them in the next section: **local execution**, execution on the **XSEDE Comet GPU cluster,** and execution on a **private GPU cluster**.

2. Running the workflow using the Local Execution mode:

In your local computer, go to the directory where Kepler was installed and where kepler.sh is located.

```
cd /PATH/kepler-2.5
```

And type the following command line in that directory to start the MD workflow. Users will need to modify the command based on their needs. The definition for each parameter is listed below. An **ABSOLUTE** path is need when running Kepler. If an absolute path is not specifically typed out in command prompt, Kepler won't be able to find the files or folders of interest and it will throw an error message.

The general command to run Kepler through command prompt is

```
./kepler.sh –runwf -nogui -parameters values /PATH/workflow.xml
```

Here is an example of how Kepler can be run for our test case:

```
./kepler.sh -runwf -nogui -ExecutionChoice LocalExecution -input0D
/PATH/Single_Cell -input1D2D /PATH/1D2D /PATH/Workflow.xml
```

For more information about each parameter, please check the parameters (see section below).

Parameters:
ExecutionChoice: This parameter gives option to run same workflow on a local machine or a remote Cluster. For local execution: LocalExecution
input0D: The inputFolder path for singlecell on your local machine. For example: /Path/to/DBP-Clancy-Rabbit-Single-Cell
input1D2D: The inputFolder path for 1D2D on your local machine. For example: /Path/to/DBP-Clancy-rabbit-1D2D

3. Running the workflow using the Remote Cluster Submission mode:

Step 1.

Creating id_rsa.pub files:
(Skip to step 3 if the user has already configured the SSH key.)

Most remote clusters require login authentications and passwords. In order to connect to the remote cluster through the command-line interface without having to type in password for each submission step, the easiest way is to generate an SSH key. To run Kepler from the command prompt smoothly, we will create SSH keys for logging into a remote cluster without password.

First, create an .ssh directory in your home directory in your local computer. If users already have an .ssh directory in their home directory, they can skip this step.

```
mkdir ~/.ssh
```

Then, in the .ssh directory, generate the public/private RSA key pairs with the below command line:

```
cd ~/.ssh
ssh-keygen -t rsa
```

The system will ask you the following questions:

```
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/you/.ssh/id_rsa): [Press enter]
Enter passphrase (empty for no passphrase): [Press enter]
Enter same passphrase again: [Press enter]
Your identification has been saved in /Users/you/.ssh/id_rsa.
Your public key has been saved in /Users/you/.ssh/id_rsa.pub.
The key fingerprint is: # 01:0f:f4:3b:ca:85:d6:17:a1:7d:f0:68:9d:f0:a2:db
```

Step 2.

Preparing for job submission in a remote cluster:

Log in to the remote cluster of your choice, for example, the NBCR Kivid Cluster.

```
ssh username@kivid.ucsd.edu
```

Type in the password. Go to the home directory and create an .ssh folder in the remote cluster. (The .ssh directory may already be there, which is fine; ignore this step).

```
cd
mkdir -p ~/.ssh
logout
```

Then, in your local computer, append the public key to your remote cluster.

```
cat ~/.ssh/id_rsa.pub | ssh username@kivid.ucsd.edu 'cat >> ~/.ssh/authorized_keys'
```

Now you should be able to log into the remote cluster from your local computer without needing to type your password.

Log in to the remote cluster and create a folder. This folder is the working directory in the remote cluster where users will run their simulation jobs

```
ssh username@kivid.ucsd.edu

cd
mkdir Workflow_Sample
```

You are ready to run the MD workflow.

Step 3.

Running MD Workflow on a GPU cluster

1. *Running the workflow using NBCR Cluster:*

In your local computer, go to the directory where Kepler was installed and where kepler.sh is located.

```
cd /PATH/WHERE/KEPLER/IS/INSTALLED/kepler-2.5
```

And type the following command line in that directory to start the MD workflow. Users will need to modify the command based on their needs. The definition for each parameter is listed below. An **ABSOLUTE** path for each file is need when running Kepler. If an absolute path is not typed out, Kepler won't be able to find the files or folders of interest and throw an error message.

The general command to run Kepler through command prompt is

```
./kepler.sh -runwf -nogui -parameters values /PATH/workflow.
```

Here is an example of how Kepler can be run for our test case.

```
./kepler.sh -runwf -nogui -ExecutionChoice kividCluster -TargetHost
username@kivid.ucsd.edu -IdentityFile /Users/username/.ssh/id_rsa -
input0D /Path/to/DBP-Clancy-Rabbit-Single-Cell -input1D2D /Path/to/DBP-
Clancy-rabbit-1D2D -remoteDir /data/username/Workflow_Sample
/path/to/DBP-1D2D-WF.xml
```

Please find brief description of the parameters below. Please customize below parameters for your execution:

Parameters:

ExecutionChoice – This parameter gives option to run same workflow on a local machine or a remote Cluster. For remote execution: kividCluster
TargetHost: Username@host. For example: username@kivid.ucsd.edu
IdentityFile: The identity File to make SSH and SCP password less. For example: /Users/username/.ssh/id_rsa
input0D: The inputFolder path for singlecell on your local machine. For example: /Path/to/DBP-Clancy-Rabbit-Single-Cell
input1D2D: The inputFolder path for 1D2D on your local machine. For example: /Path/to/DBP-Clancy-rabbit-1D2D
remoteDir: Work Directory on the remote cluster. For example: /home/username/Workflow_Sample

4. Checking the MD simulation output data:

Initial check

The outputs contain an output directory for each set of simulations (DBP-Clancy-Rabbit-Single-Cell/outputFolder, DBP-Clancy-rabbit-1D2D/output1DFolder, and DBP-Clancy-rabbit-1D2D/output2DFolder). Inside each output folder, the following files can be found:

DBP-Clancy-Rabbit-Single-Cell/outputFolder	**DBP-Clancy-rabbit-1D2D/output1DFolder**
apds_1Hz.txt	ECG.png
vm_1Hz.txt	y_1D.txt
allresult_1Hz.txt	ECGs.txt
RyR.png	
I_NCX.png	
Na.png	**DBP-Clancy-rabbit-1D2D/output2DFolder**
IKr_IK1.png	movie_FBL.avi
IKs_ICFTR.png	ap*.dat files
INa.png	
Ito.png	
Cai.png	
CaSRT_Caj.png	
ICa.png	
vm.png	

Chapter 5: For Advanced Users

1. Configuring Kepler for multiple workflow executions simultaneously through command line

Step 1.

Close the Kepler GUI, if open

Go to KeplerData directory. It is usually located in the user's home directory

```
cd /home/username/KeplerData

or

cd ~/KeplerData
```

Step 2.

Edit DB Host parameter in configuration. file

```
vi modules/provenance/resources/configurations(latest)/configuration.
```

In configuration file: provide DB Host value as "localhost" and save changes.

```
<!-- host name running the database -->
<pair>
    <name>DB Host</name>
    <value>localhost</value>
</pair>
```

Step 3.

Start HSQL

Go to the directory where Kepler is installed. Start HSQL.

```
cd /PATH/WHERE/KEPLER/IS/INSTALLED/kepler-2.5

kepler.sh -hsql start
```

Step 4.

Run multiple workflows through the command line

```
./kepler.sh -runwf -nogui -parameters values /PATH/workflow1.
./kepler.sh -runwf -nogui -parameters values /PATH/workflow2.
./kepler.sh -runwf -nogui -parameters values /PATH/workflow3.
./kepler.sh -runwf -nogui -parameters values /PATH/workflow4.
```

Step 5.

Stop HSQL when you are done

```
kepler.sh -hsql stop
```