

Jul 19, 2019

# A multi-scale model of cardiac electrophysiology

 In 1 collection

DOI

[dx.doi.org/10.17504/protocols.io.5nkg5cw](https://dx.doi.org/10.17504/protocols.io.5nkg5cw)

Colleen clancy<sup>1</sup>, Pei-Chi<sup>1</sup>, Parya Aghasafari<sup>1</sup>

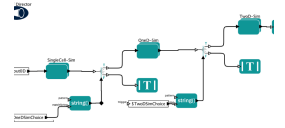
<sup>1</sup>University of California, Davis

SPARC

Tech. support email: [info@neuinfo.org](mailto:info@neuinfo.org)



Parya Aghasafari



OPEN  ACCESS



**DOI:** [dx.doi.org/10.17504/protocols.io.5nkg5cw](https://dx.doi.org/10.17504/protocols.io.5nkg5cw)

**External link:** [https://github.com/ClancyLabUCD/Workflow\\_Kepler](https://github.com/ClancyLabUCD/Workflow_Kepler)

**Protocol Citation:** Colleen clancy, Pei-Chi, Parya Aghasafari 2019. A multi-scale model of cardiac electrophysiology. [protocols.io https://dx.doi.org/10.17504/protocols.io.5nkg5cw](https://dx.doi.org/10.17504/protocols.io.5nkg5cw)

**Manuscript citation:**

**License:** This is an open access protocol distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

**Protocol status:** Working

**We use this protocol and it's working**

**Created:** July 19, 2019

**Last Modified:** July 19, 2019

**Protocol Integer ID:** 26028

## Abstract

Multi-scale computational modeling is a major branch of computational biology as evidenced by the US federal interagency Multi-Scale Modeling Consortium and major international projects. It invariably involves specific and detailed sequences of data analysis and simulation, often with multiple tools and datasets, and the community recognizes improved modularity, reuse, reproducibility, portability and scalability as critical unmet needs in this area. Scientific workflows are a well-recognized strategy for addressing these needs in scientific computing. While there are good examples of the use of scientific workflows in bioinformatics, medical informatics, biomedical imaging and data analysis, there are fewer examples in multi-scale computational modeling in general and cardiac electrophysiology in particular. Cardiac electrophysiology simulation is a mature area of multi-scale computational biology that serves as an excellent use case for developing and testing new scientific workflows. In this dataset, we develop, describe and test a computational workflow that serves as a proof of concept of a platform for the robust integration and implementation of a reusable and reproducible multi-scale cardiac cell and tissue model that is expandable, modular and portable. The workflow described leverages Python and Kepler-Python actor for plotting and pre/post-processing. During all stages of the workflow design, we rely on freely available open-source tools, to make our workflow freely usable by scientists.

## Attachments



[A demonstration of m...](#)

2.4MB



[UserManual.pdf](#)

662KB

## 1 **Creating multi-scale model of cardiac electrophysiology**

We developed a workflow containing differential equation models of cardiac physiology that automate the execution of simulations with user defined options of outputs from a single cell, 1 or 2D tissue, and a pseudo-ECG output, which can be compared to experimental or clinical data.

## 2 **Compiling Human\_OD1D2DWF program**

- **In order to compile 1D2D Human\_OD1D2DWF program model, located in Code\Human\_OD1D2DWF\1D2D folder in our data set, follow these instructions:**

To compile the program:

---

(1) Using Intel Compiler:

```
icc masterCompute.cpp -openmp -o oneD
```

```
./oneD initial_WTstates.txt stim_param.txt testOutputFolderTWoD
```

---

OR

(2) Using GCC compiler:

```
g++ masterCompute.cpp -fopenmp -o oneD
```

```
./oneD initial_WTstates.txt stim_param.txt testOutputFolderONEd
```

---

Input settings in stim\_param.txt

1 Set Na blocker drug concentration

2 Set Kr blocker ratio

3 Set basic cycle length (BCL)

4 Set how many beats for 1D (e.g 200).

For 2D, **set beat = 1**

5 Set Ligand concentrations

6 Adjust cAMKII activity levels (expression = 'WT', 'OE', or 'KO')

7. 1D or 2D

8. tissue size =  $tw * tl$ . Enter value for  $tw$ .

9. tissue size =  $tw * tl$ . Enter value for  $tl$ .

10. homogeneous or heterogeneous

---

For 1D simulations: Read initial\_WTstates.txt which is the initial variables file corresponding to SS\_rabbit\_varNames.txt (total variables are 206, although some of them are not used in the model)

(1) outputs generated in ecgs.txt (1D case)

1st column: time

2nd column: ecg

(2) Time course of AP surface (1D)

---

For 2D simulations: Read in fiber.cell which is generated from 1D simulations

(3) Time snapshots for 2D

- **In order to compile Single\_Cell Human\_OD1D2DWF program model, located in Code\Human\_OD1D2DWF\ Single\_Cell folder in our data set, follow these instructions:**

To compile the program:

Using Intel Compiler ICC `icc masterCompute.cpp -o singlecell`

```
./singlecell initial_WTstates.txt stim_param.txt testOutputFolder
```

---

Using GCC

```
g++ masterCompute.cpp -o singlecell
```

```
./singlecell initial_WTstates.txt stim_param.txt testOutputFolder
```

---

plotall.m collects the results from masterCompute.cpp and plots the time course of voltage or currents

---

Input settings in stim\_param.txt

1. Set Na blocker drug concentration
  2. Set Kr blocker ratio
  3. Set basic cycle length (BCL)
  4. Set how many beats.
  5. Set Ligand concentration
  6. Adjust cAMKII activity levels (expression = 'WT', 'OE', or 'KO')
- 

initial\_WTstates.txt is the initial variables file corresponding to SS\_rabbit\_varNames.txt (total variables are 206, although some of them are not used in the model)

---

(1) outputs generated in vm\_1Hz.txt 29 Ca bound to Casqn

Ca SR  
Na dyad  
Na sl 33 Na cytosol  
K  
Ca dyad  
Ca sl  
Ca cytosol  
Vm

(2) outputs generated in allresult\_1Hz.txt

I\_Ca\_store,  
I\_to\_store[0],

I\_Na\_store,  
I\_K1\_store,  
Jserca,  
IKs\_store,  
IKr\_store,  
Jleak[0],  
Jleak[1],  
ICFTR,  
pars1.Incx

(3) outputs generated in apds\_1Hz.txt

1st column: beat  
2nd column: APD90

### 3 **Compiling Mouse\_OD1D2DWF program**

- **In order to compile 1D2D Mouse\_OD1D2DWF program model, located in Code\Mouse\_OD1D2DWF\1D2D folder in our data set, follow these instructions:**

To compile the program:

---

(1) Using Intel Compiler:

```
icc masterCompute.cpp -openmp -o oneD
```

```
./oneD initial_WTstates.txt stim_param.txt testOutputFolderTWoD
```

---

OR

(2) Using GCC compiler:

```
g++ masterCompute.cpp -fopenmp -o oneD
```

```
./oneD initial_WTstates.txt stim_param.txt testOutputFolderONEd
```

---

Input settings in stim\_param.txt

1 Set Na blocker drug concentration

2 Set Kr blocker ratio

3 Set basic cycle length (BCL)

4 Set how many beats for 1D (e.g 200).

For 2D, **set beat = 1**

5 Set Ligand concentrations

6 Adjust cAMKII activity levels (expression = 'WT', 'OE', or 'KO')

7. 1D or 2D

8. tissue size =  $tw * tl$ . Enter value for  $tw$ .

9. tissue size =  $tw * tl$ . Enter value for  $tl$ .

10. homogeneous or heterogeneous

---

For 1D simulations: Read initial\_WTstates.txt which is the initial variables file corresponding to SS\_rabbit\_varNames.txt (total variables are 206, although some of them are not used in the model)

(1) outputs generated in ecgs.txt (1D case)

1st column: time

2nd column: ecg

(2) Time course of AP surface (1D)

---

For 2D simulations: Read in fiber.cell which is generated from 1D simulations

(3) Time snapshots for 2D

- **In order to compile Single\_Cell Mouse\_OD1D2DWF program model, located in Code\Mouse\_OD1D2DWF\Single\_Cell folder in our data set, follow these instructions:**

To compile the program:

Using Intel Compiler ICC `icc masterCompute.cpp -o singlecell`

`./singlecell initial_WTstates.txt stim_param.txt testOutputFolder`

---

Using GCC

`g++ masterCompute.cpp -o singlecell`

`./singlecell initial_WTstates.txt stim_param.txt testOutputFolder`

---

plotall.m collects the results from masterCompute.cpp and plots the time course of voltage or currents

---

Input settings in stim\_param.txt

1. Set Na blocker drug concentration
  2. Set Kr blocker ratio
  3. Set basic cycle length (BCL)
  4. Set how many beats.
  5. Set Ligand concentration
  6. Adjust cAMKII activity levels (expression = 'WT', 'OE', or 'KO')
- 

initial\_WTstates.txt is the initial variables file corresponding to SS\_rabbit\_varNames.txt (total variables are 206, although some of them are not used in the model)

---

(1) outputs generated in vm\_1Hz.txt

- Ca bound to Casqn
- Ca SR
- Na dyad





Na sl  
Na cytosol  
K  
Ca dyad  
Ca sl  
Ca cytosol  
Vm

(2) outputs generated in allresult\_1Hz.txt

I\_Ca\_store,  
I\_to\_store[0],  
I\_Na\_store,  
I\_K1\_store,  
Jserca,  
IKs\_store,  
IKr\_store,  
Jleak[0],  
Jleak[1],  
ICFTR, pars1.Incx

(3) outputs generated in apds\_1Hz.txt

1st column: beat  
2nd column: APD90

#### 4 **Compiling Rabbit\_OD1D2DWF program**

- **In order to compile 1D2D Rabbit\_OD1D2DWF program model, located in Code\Rabbit\_OD1D2DWF\1D2D folder in our data set, follow these instructions:**

To compile the program:

---

(1) Using Intel Compiler:

```
icc masterCompute.cpp -openmp -o oneD
```

```
./oneD initial_WTstates.txt stim_param.txt testOutputFolderTWOd
```

---

OR

(2) Using GCC compiler:

```
g++ masterCompute.cpp -fopenmp -o oneD
```

```
./oneD initial_WTstates.txt stim_param.txt testOutputFolderONEd
```

---

Input settings in stim\_param.txt

1 Set Na blocker drug concentration

2 Set Kr blocker ratio

3 Set basic cycle length (BCL)

4 Set how many beats for 1D (e.g 200).

For 2D, **set beat = 1**

5 Set Ligand concentrations

6 Adjust cAMKII activity levels (expression = 'WT', 'OE', or 'KO')

7. 1D or 2D

8. tissue size =  $tw * tl$ . Enter value for  $tw$ .

9. tissue size =  $tw * tl$ . Enter value for  $tl$ .

10. homogeneous or heterogeneous

---

For 1D simulations: Read initial\_WTstates.txt which is the initial variables file corresponding to SS\_rabbit\_varNames.txt (total variables are 206, although some of them are not used in the model)

(1) outputs generated in ecgs.txt (1D case)

1st column: time

2nd column: ecg

(2) Time course of ECG (1D)

---

For 2D simulations: Read in fiber.cell which is generated from 1D simulations

(3) Time snapshots for 2D

- **In order to compile Single\_Cell Rabbit\_OD1D2DWF program model, located in Code\Rabbit\_OD1D2DWF\Single\_Cell folder in our data set, follow these instructions:**

To compile the program:

Using Intel Compiler ICC `icc masterCompute.cpp -o singlecell`

`./singlecell initial_WTstates.txt stim_param.txt testOutputFolder`

---

Using GCC

`g++ masterCompute.cpp -o singlecell`

`./singlecell initial_WTstates.txt stim_param.txt testOutputFolder`

---

plotall.m collects the results from masterCompute.cpp and plots the time course of voltage or currents

---

Input settings in stim\_param.txt

1. Set Na blocker drug concentration
2. Set Kr blocker ratio
3. Set basic cycle length (BCL)

4. Set how many beats.
5. Set Ligand concentration
6. Adjust cAMKII activity levels (expression = 'WT', 'OE', or 'KO')

---

initial\_WTstates.txt is the initial variables file corresponding to SS\_rabbit\_varNames.txt  
(total variables are 206, although some of them are not used in the model)

---

(1) outputs generated in vm\_1Hz.txt

- Ca bound to Casqn
- Ca SR
- Na dyad
- Na sl
- Na cytosol
- K
- Ca dyad
- Ca sl
- Ca cytosol
- Vm

(2) outputs generated in allresult\_1Hz.txt

- I\_Ca\_store,
- I\_to\_store[0],
- I\_Na\_store,
- I\_K1\_store,
- Jserca,
- IKs\_store,
- IKr\_store,
- Jleak[0],
- Jleak[1],
- ICFTR,
- pars1.Incx

(3) outputs generated in apds\_1Hz.txt

1st column: beat  
2nd column: APD90