HMSM V1.0

Zhiwei JI

Section 1: Background

HMSM model is a hybrid system modeling platform, which integrates an ODE system for intracellular signaling pathway simulation and an agent-based model (ABM) for tumor growth, immune response, and angiogenesis in CRPC development. The whole framework of ABM was built up with C++. The ODE system was developed with C and was optimized by using a Fortan package libf77. ODEs need to be compiled with the optimal parameters and then can be used in C++.
All the details can be found in the paper: Z Ji, *et al*., *Plos Computational Biology*, 2019. PMID: 31504033.

Source code locates at LS5 on TACC: /work/02480/jizhiwei/OLDCRPC.

Section 2: Functions

SolveMain.cpp: Main entrance of the model.

MicroEnv.cpp: Initialization of microenvironment and the parameters.

Par.cpp: the default values of some key parameters.

EC.cpp: the dynamic response of an endothelial cell to ME. It runs 1 time at each timestep.

PC.cpp: the dynamic response of a prostate cancer cell to ME. It runs 1 time at each timestep.

TAM.cpp: the dynamic response of a TAM cell to ME. It runs 1 time at each timestep.

TREG.cpp: the dynamic response of a TREG cell to ME. It runs 1 time at each timestep.

CTL.cpp: the dynamic response of a CTL cell to ME. It runs 1 time at each timestep.

DC.cpp. the dynamic response of a dendritic cell to ME. It runs 1 time at each timestep.

Location.cpp: the 3D position for each point in ME.

ODEmodel.c: ODE model of signaling pathway of PC (c code).

Helloworld.c: Parallel computing for multiple running of ABM.

*.h: the head files for the files with the same names.

run_main_N.sh was used to compile the whole C++ package as run_main.o

Section 3. How to use?

Before running, all the C++, C file need to be compiled and generate an execute file "run_main".

Bash run_main_N.sh run_main

Two ways to use:

1. Single run. Implement ABM model one time and generate one case of tumor growth.
   #main menu
   Run_main -h

```
login2.ls5(1002)$ run_main -h
Usage: run_main
        [-s Thread ID]
        [-t Time steps]
        [-P The switch of Prostate Cell]; 1 for on, 0 for off.
        [-G The switch of Macrophage]; 1 for on, 0 for off.
        [-C The switch of CTL]; 1 for on, 0 for off.
        [-T The switch of Treg]; 1 for on, 0 for off.
        [-A level of Drug 1(MDV), 0 to 10]; 0 means no drug.
        [-B level of Drug 2(PLX), 0 to 10]; 0 means no drug.
        [-I level of Drug 3(IL2), 0 to 10]; 0 means no drug.
        [-K level of Drug 4(SKP2), 0 to 10]; 0 means no drug.
        [-W level of Drug 5(WNT5), 0 to 10]; 0 means no drug.
        [-D The switch of Andrgen]; 1 for on, 0 for off.
```

   #run a case. 4 types of cells occurred and no treatment
   Run_main -S 999 -t 792 -P 1 -G 1 -C 1 -T 1

```
login2.ls5(1004)$ run_main -s 999 -t 792 -P 1 -G 1 -C 1 -T 1
thread: 999
time: 792
PC on or off? 1
TAM on or off? 1
CTL on or off? 1
Treg on or off? 1
Drug [MDV]: 0
Drug [PLX]: 0
Drug [IL2 Inhibitor]: 0
Drug [SKP2 Inhibitor]: 0
Drug [WNT5A Inhibitor]: 0
PCs:200, TAMs: 100, CTLs:80, TREGs:80, ECs:6

T:1, PC:189 (CRPC:0), TAM:100, CTL:80 (80,0), TREG:80 (80,0), EC:6
[VEGF]:0.00249003, [CSF]:0.00612156, [EGF]:0.00855981, [IL10]:0.00206953
[IL2]:2.85e-07, [WNT5A]:2.85e-07, [TRAIL]:0.000855952
Cell lysis:189  0

T:2, PC:185 (CRPC:0), TAM:100, CTL:80 (80,0), TREG:80 (80,0), EC:6
[VEGF]:0.00387324, [CSF]:0.00836467, [EGF]:0.00813182, [IL10]:0.00201522
[IL2]:2.7075e-07, [WNT5A]:2.7075e-07, [TRAIL]:0.000813155
Cell lysis:185  0

T:3, PC:179 (CRPC:0), TAM:100, CTL:80 (80,0), TREG:80 (80,0), EC:6
[VEGF]:0.00513937, [CSF]:0.010414, [EGF]:0.00772523, [IL10]:0.0019631
[IL2]:2.57213e-07, [WNT5A]:2.57213e-07, [TRAIL]:0.000772497
Cell lysis:179  0

T:4, PC:174 (CRPC:0), TAM:100, CTL:80 (80,0), TREG:80 (80,0), EC:6
[VEGF]:0.00630208, [CSF]:0.0122926, [EGF]:0.00733896, [IL10]:0.00191307
[IL2]:2.44352e-07, [WNT5A]:2.44352e-07, [TRAIL]:0.000733872
Cell lysis:174  0
```

2. Multiple runs. Parallelly run the ABM model and evaluate the average dynamics of tumor growth and ligand dose.

1) Helloworld.c will parallel to call run_main. For each parallel computing with ABM model, all the parameters were given in Helloworld.c, and it will repeat 200 times. You also can change the repeat times in Helloworld.c. Therefore, Helloworld.c should be complie each time when you update it:

   mpicc Helloworld.c Helloworld

2) myJob.sh was used to submit a job of parallel computing to server.

   Sbatch myJob.sh

```
------------------------------------------------------------
         Welcome to the Lonestar 5 Supercomputer
------------------------------------------------------------

No reservation for this job
--> Verifying valid submit host (login2)...OK
--> Verifying valid jobname...OK
--> Enforcing max jobs per user...OK
--> Verifying availability of your home dir (/home1/02480/jizhiwei)...OK
--> Verifying availability of your work dir (/work/02480/jizhiwei/lonestar)...OK
--> Verifying availability of your scratch dir (/scratch/02480/jizhiwei)...OK
--> Verifying valid ssh keys...OK
--> Verifying access to desired queue (normal)...OK
--> Verifying job request is within current queue limits...OK
--> Checking available allocation (Develop-Translationa)...OK
Submitted batch job 2920161
```

3) You also can use launcher on TACC to implement parallel computing.

```
#!/bin/bash   myJob.sh
#SBATCH -J MPIjob_Zhiwei
#SBATCH -o matlabjob_output.o%j
#SBATCH -N 51
#SBATCH -n 51
#SBATCH -p normal
#SBATCH -t 26:00:00

module load launcher

export LAUNCHER_PLUGIN_DIR=$LAUNCHER_DIR/plugins
export LAUNCHER_RMI=SLURM
export LAUNCHER_JOB_FILE=jobfile1
export LAINCHER_PPN=1

$LAUNCHER_DIR/paramrun
```

#Jobfile1 (assign different Thread ID to each replicate)

Run_main -S 001 -t 792 -P 1 -G 1 -C 1 -T 1
Run_main -S 002 -t 792 -P 1 -G 1 -C 1 -T 1
Run_main -S 003 -t 792 -P 1 -G 1 -C 1 -T 1
Run_main -S 004 -t 792 -P 1 -G 1 -C 1 -T 1
Run_main -S 005 -t 792 -P 1 -G 1 -C 1 -T 1