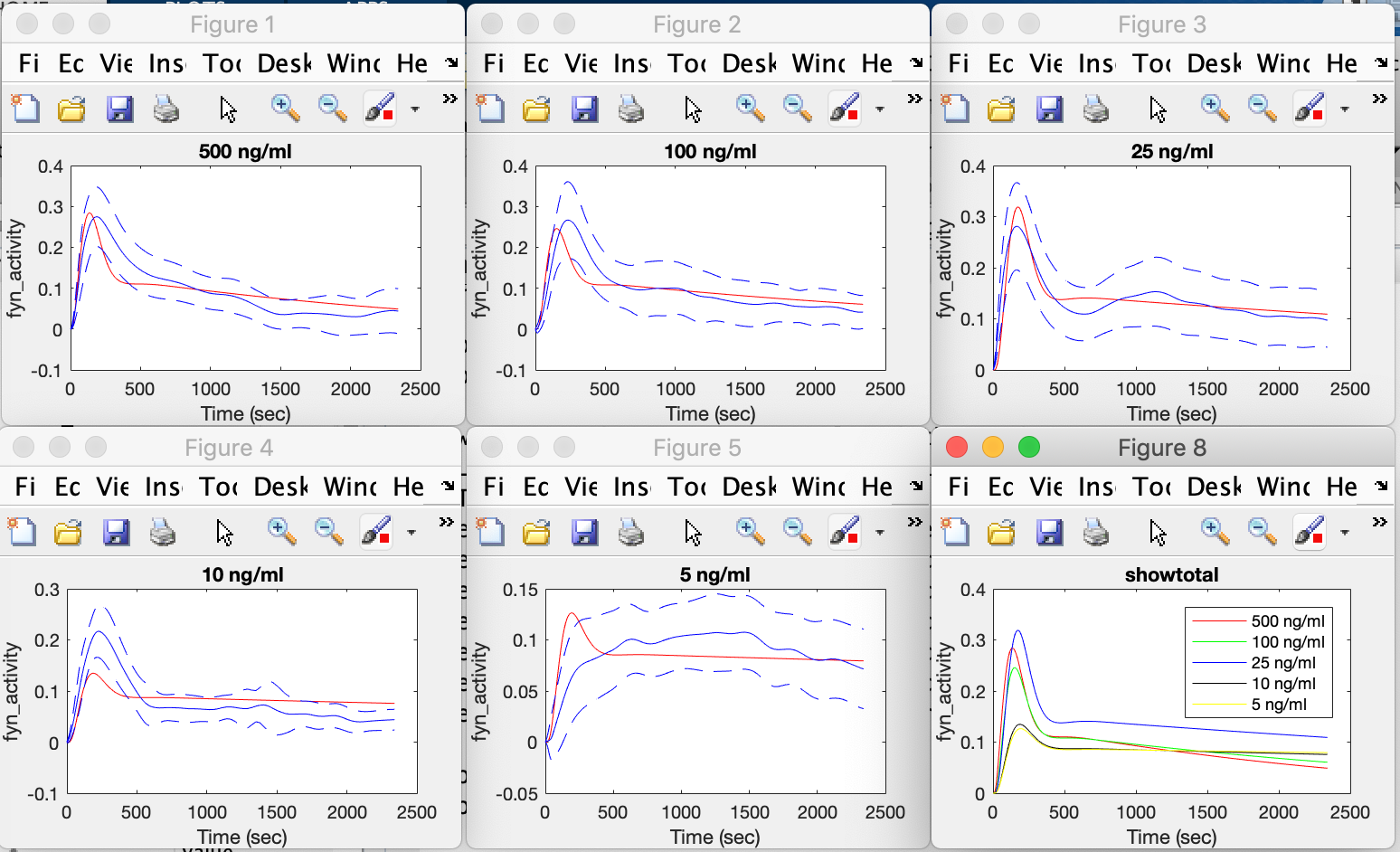
**Programmer’s Guide/Notes**

1. **Display the results by running the script watch.m**

>> cd /Users/kathylu/Documents/sof/odesys/app/egf

>> data = watch(‘egf1’);

****

1. **Run the optimization program for parameter fitting**

>> data = action(‘egf2’);

Prepare: Experiment type = egf2

Estimated latin Sampling time = 1 sec, 0.000 hr

Estimated total optimization time = 60 sec, 0.017 hr

Prepare: loading ../../data/1019\_2019/egf/exp.mat

Min function value for single concentration renewed is 30.975295.

Latin\_Sampling: saving ../../data/1019\_2019/egf/output/latin\_sample.mat

ng\_number = 0, bs\_number = 0, total\_number = 0, sample\_number = 6

Action.m : loading data from ../../data/1019\_2019/egf/output/latin\_sample.mat

Index Double-peak Value Volume-scaled Value Initial All All Value Time (s)

1 30.6721 72.1044 5.5963e+04 72.1044 86.94

2 34.5740 119.1248. 1.2117e+02 119.1248 75.83

3 8.2157 37.3783 3.8854e+02 37.3783 90.06

4 4.6283 70.4808 3.0026e+10 70.4808 91.88

5 4.4950 97.4204 8.1127e+10 97.4204 110.28

6 4.2993 98.5809 1.7254e+12 98.5809 82.28

>> data = action(‘pdgf2’); % running pdgf simulation.

Prepare: Experiment type = pdgf2

Estimated latin Sampling time = 1 sec, 0.000 hr

Estimated total optimization time = 60 sec, 0.017 hr

Prepare: loading ../../data/1019\_2019/pdgf/exp.mat

Min function value for single concentration renewed is 51.785079.

Min function value for single concentration renewed is 14.827342.

Latin\_Sampling: saving ../../data/1019\_2019/pdgf/output/latin\_sample.mat

ng\_number = 0, bs\_number = 0, total\_number = 0, sample\_number = 6

Index Double-peak Value Volume-scaled Value All Value Time (s)

1 14.827342 8898.989211 8898.989211 98.035058

2 51.785079 1343.017570 1343.017570 95.427276

3 113.909359 1661.206856 1661.206856 70.980720

4 350.039091 2920.829307 2920.829307 89.218153

5 824.835874 10022.577895 10022.577895 75.892299

6 2274.847678 10649.254163 10649.254163 112.930985

**Access the output**

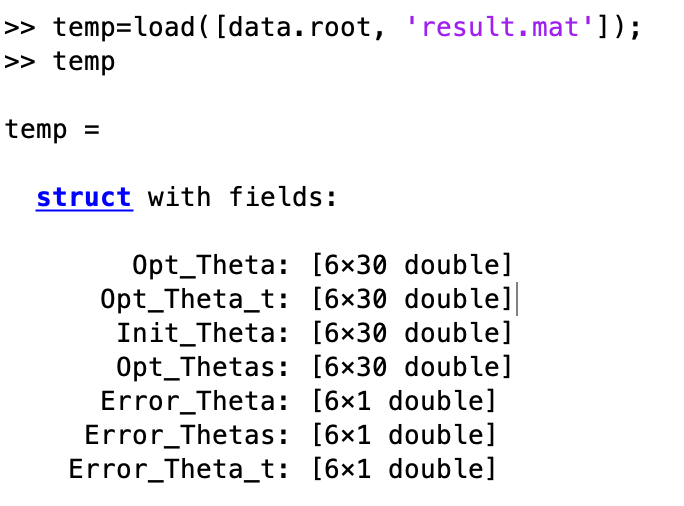
Load the results file, and copy the values of opt\_theta to an excel file

>> temp = load([data.root, 'result.mat']); % previously ‘results.mat’

>> opt\_theta = temp.Opt\_Theta';

>> para\_nametag' % print out names of parameters. % double-check collected\_results

**There is no column 7 in results**

****

1. **How not to change the scaling factor/ and the volume scale values?**
2. **Program structure**

The app/ folder contains work in progress with three subfolders

ode/ --- ode models and solvers

opt/ --- optimization models and solvers.

egf/ --- programs by Tongkai Li

The newest ode can be found in ode/ode\_model\_1118.m; the optimization method is defined in opt/optimize\_solve.m . The main test functions that can be run are ode/test\_odesys.m and opt/test\_ode\_optimize.m .

**Notes from Tongkai Li (10/18/2019)**

参数拟合程序简易操作指南：

以下简单任务默认初始文件夹为解压后的最高一级目录.

以egf为例, pdgf为完全类似.

任务一:

查看当前模型的一些已有拟合结果：

1. 进入\egf\collected\_results. (可以看到对应有文件夹data1-6是一些已经跑好的结果).

2. 进入\code，在matlab中打开 watch.m 文件.

3. 修改对应第6和第7行的数据文件夹地址来选取要查看的结果. (默认为data，可替换为data1-7等已经存在的文件夹).

4. 运行watch.m，会展示当前文件夹对应结果的图像. (默认每一组参数暂停，matlab命令行按任意键可继续展示).

5 如果想跳过后面结果，可以ctrl+c直接退出，之前展示的结果也会储存. (或修改倒数第二行的pause（））.

6. 展示的图像结果和参数值存储在\code上级目录的\results文件夹中, 可以直接调出.

7. 每个data\*对应一个results文件夹，重新选择data文件夹时如果需要注意将results文件改名防止被新存储的结果覆盖.

任务二：

简单运行程序进行数据预处理:

1. 进入\egf文件夹, egf\_unor是未经处理的数据.

2. 运行preprocess\_egf.m，最终会展示预处理之后的浓度图.

3. 在103行增设断点，可以查看被标记为outlier的曲线图. (默认为绝对L2距离较大的为离群数据).

pdgf对应断点为127行.

任务三:

简单运行程序进行参数拟合:

{ 0. 如果需要进行额外的数据预处理, 则将处理后得到的 egf\_exp.mat 存入\egf\data目录. }

1. 进入\egf\code文件夹.

2. 在matlab中运行Action.m即可.

一些简单可调的东西:

1. 程序的运行时间主要由拉丁方采样的设置时间决定. 服务器上默认12h

的采样，外加2天半的拟合时间. 6进程分别计算 (对应data1-6）.如需

快速得到一个拟合结果供参考, 可以通过大幅降低采样时间来实现：

I: 在matlab中打开Latin\_sample.m.

II: 修改第31行的最大采样时间. (如只采样10min以内可能会采集不到双峰初始值).

2. watch.m中21-32行提供了算法结果中的变量与其性质的解释.

可以通过修改接下的几行来查看不同的结果. (默认选取的带有

程序员的主观意识..)

3. 写程序的同学自己觉得的一些影响结果程序内变量：

Prepare:

special\_index：单浓度优化取得组号，egf一般取得是双峰.

Action.m:

M: 拉丁方格子数的参数, 不好取，经验上是别太大就行.

Coor\_descent.m:

pho & max\_armi: 搜索步长.

max\_iter: 最大终止步长, 高维的优化容易陷入下降很慢的区域，需要及时跳出。一般而言函数值下降很慢的时候，结果也比较接近了(或者说算法已经trap了).

虽然不太会写注, 但是文件中注释还是有一些的，希望能有帮助.

文件的汇总:

主文件夹为/code 和 /data, 分别存放代码和数据.

/data:

exp.mat：最早通用的储存实验数据的工作区.

egf\_exp.mat(egf\_data.mat): 名字不一样，其实大多时候是同一个东西.

latin\_sample.mat：拉丁方采样的结果. 高维优化采样很关键, 撒的点存储的地方.

results.mat：数学上没找到统一的度量误差的形式，所以这里面存了很多种不同标准的结果.

/code:

watch.m： 顾名思义，用于观看结果~~.

undimen.m: 由于已知系统的方程，我们可以做一些无量纲操作来挑选更好的初值，例如直接匹配实验和模拟结果的第一个峰值.

Scaling.m: 参数函数中有一个关于生物体内浓度和荧光读数的转化参数, 关于这一步的优化是可以解析求解的，故单独列出.

run\_Action.slurm：此文件略违和，是用于提交学院服务器多个进程伪并行优化的.

rhs\_2.m： 常微分方程的右端项函数，之所以是2大概是因为第二版后就在基础上就沿用了。。

Prepare.m: 初始化数据，初始化函数句柄，各种初始化.

Locate\_crit.m: 用来定位要处理的曲线的critical point，主要用作于施加双峰限制.

Latin\_Sampling.m：拉丁方采样。高维优化问题采样总是很关键但是却很玄学的，虽然做了一些经验的调试，但是并没有什么很大作用.

Is\_OK.m： 最早用来定位可以产生bell-shape行为的参数组合, 但没有太多找到有意思的行为. 后续也可以修改加其他限制。

forward\_solver.m: 开始觉得可能解法器要重新设计，但后来发现现成的很完善，所以就一句话。不过这是这个算法复杂度的最小单位，对于测试还是有一点儿作用的.

Error\_theta.m: 对于参数估计设计的误差函数，最简单的就是L2范数, 当然如果需要中间也可以加各种惩罚项。

Error\_ODE2EXP.m: 最早应该是为了把ode的误差和可能要加的关于参数的惩罚项分开。

Coor\_Descent.m: 坐标梯度下降法，不用全梯度，高维似乎都这么做，也没有找到更好的做法。由于此处用的是差分梯度，所以涉及到很多步长和参数的选取，但是似乎找不到统一标准，高维的就是很rough需要各种调试 .

Action.m: Action--还没cut. 码程序的时候希望自己开心一点取得主程序名字~~~