

# Sichuan university

## User Guidance

### Genetic-Driven Rapid and Precise Mimicry of Cardiovascular Fluctuations

Minghao Liao<sup>1,2,3</sup>, Zhongyou Li<sup>1,2,3\*</sup>, Wentao Jiang<sup>1,2,3,4\*</sup>, Taoping Bai<sup>1,2,3</sup>, Lingjun Liu<sup>5</sup>,  
Fei Yan<sup>6</sup>

1. Sichuan Province Biomechanical Engineering Laboratory, Sichuan University, Chengdu, China
2. Key Laboratory of Deep Earth Science and Engineering, Ministry of Education, College of Architecture and Environment, Sichuan University, Chengdu, China
3. Department of Mechanical Science and Engineering, College of Architecture and Environment, Sichuan University, Chengdu, China
4. College of Computer Science, Sichuan Normal University, Chengdu, China
5. Key Laboratory of Obstetric and Gynecologic and Pediatric Diseases and Birth Defects of Ministry of Education, West China Second University Hospital, Sichuan University, Chengdu, China
6. Chongqing University Three Gorges Hospital, Chongqing University, Chongqing, China

**\*Corresponding Author 1:** Dr. Zhongyou Li ([zhongyou\\_li@outlook.com](mailto:zhongyou_li@outlook.com))

**\*Corresponding Author 2:** Prof. Wentao Jiang ([scubme\\_jwt@outlook.com](mailto:scubme_jwt@outlook.com))

## Contents

1.	General .....	I
1.1	Development background .....	I
1.2	Download and Installation .....	I
1.3	Code Introduction.....	I
1.4	Operational prerequisites.....	II
2.	Parameter file .....	III
2.1	Prerequisites for use .....	III
2.2	File formats .....	III
2.3	Document parameters.....	IV
2.3.1	Line 1 .....	IV
2.3.2	Line 2.....	IV
2.3.3	Line 3.....	IV
2.3.4	Line 4.....	IV
2.4	TargetData.Info array .....	IV
2.4.1	TargetData.Info(1) .....	V
2.4.2	TargetData.Info(2) .....	V
2.4.3	TargetData.Info(3) .....	V
2.4.4	TargetData.Info(4) .....	V
2.4.5	TargetData.Info(5) .....	V
2.4.6	TargetData.Info(6) .....	V
2.4.7	TargetData.Info(7) .....	V
2.4.8	TargetData.Info(8) .....	V
2.4.9	TargetData.Info(9) .....	VI
2.4.10	TargetData.Info(10) .....	VI
2.4.11	TargetData.Info(11) .....	VI
2.4.12	TargetData.Info(12) .....	VI
2.4.13	TargetData.Info(13) .....	VI
2.4.14	TargetData.Info(14) .....	VI
3.	<b>Model settings</b> .....	VII
3.1	Prerequisites for use .....	VII
3.2	Signal Line Setting.....	VII
3.2.1	Module Connections.....	VII
3.2.2	Recording signals .....	VII
4.	<b>Target data</b> .....	IX
4.1	Prerequisites for use .....	IX
4.2	Data files .....	IX
5.	<b>Documentation of functions</b> .....	X
5.1	Prerequisites for use .....	X
5.2	Main function files .....	X
5.2.1	Mainbody.....	X

5.2.2	LoopIterates .....	XI
5.3	Function files.....	XI
5.3.1	createFit .....	XI
5.3.2	DDTW .....	XI
5.3.3	ed .....	XI
5.3.4	Filter .....	XI
5.3.5	Fitness.....	XI
5.3.6	GeneticArithmetic .....	XI
5.3.7	InitializationParameters .....	XI
5.3.8	Normalization .....	XII
5.3.9	Plot.....	XII
5.3.10	Read.....	XII
5.3.11	Savfil.....	XIII
5.3.12	Simulink.....	XIII
5.3.13	Target.....	XIII
6.	<b>TargetData structure .....</b>	<b>XIV</b>
6.1	Introduction to Structures.....	XIV
6.2	Structure details.....	XIV
6.2.1	TargetData.mdlName .....	XIV
6.2.2	TargetData.Info .....	XIV
6.2.3	TargetData.System.....	XIV
6.2.4	TargetData.lineName .....	XV
6.2.5	TargetData.Targetcell.....	XV
6.2.6	TargetData.pop.....	XV
6.2.7	TargetData.parpoolNum .....	XV
6.2.8	TargetData.parpool .....	XVI
6.2.9	TargetData.IterationData.....	XVI
6.2.10	TargetData.fitness .....	XVI
6.2.11	TargetData.popOther.....	XVI
7.	<b>Output files.....</b>	<b>XVII</b>

### ***1. General***

#### ***1.1 Development background***

This code is based on the Lumped parameter model (LPM) problem, mainly for parameter optimization. The MATLAB code is the main body of the code program, which is based on the genetic algorithm (GA) and the derivative dynamic time warping (DDTW) algorithm, which can quickly determine the model parameters based only on the target physiological curves, such as blood flow and blood pressure curves, and the output curve of the model can be quickly determined with the target curve. Warping (DDTW) as the basis, the development and implementation of a fully automated parameter optimization code program, can be based only on the target physiological curves, such as blood flow and blood pressure curves, and can quickly determine the model parameters, and the model output curves and the target curve similarity is good, for the future centralized parameter model to improve the efficiency of the establishment and use of the threshold of the user to reduce the use of the user.

#### ***1.2 Download and Installation***

The code for this program is uploaded to GitHub and the CSDN community and is only guaranteed to work on computers with Windows 11, installed to MATLAB R2023b and above, and 16GB of RAM and above.

**Note: no installation is required, just run the code.**

#### ***1.3 Code Introduction***

These program codes are MATLAB codes, and the files are ".m" as a suffix, which is to "Mainbody", and "LoopIterates" code. "Mainbody" and "LoopIterates" codes are the main code files, and the rest are function files. All codes are shown in **Fig 1-1**.

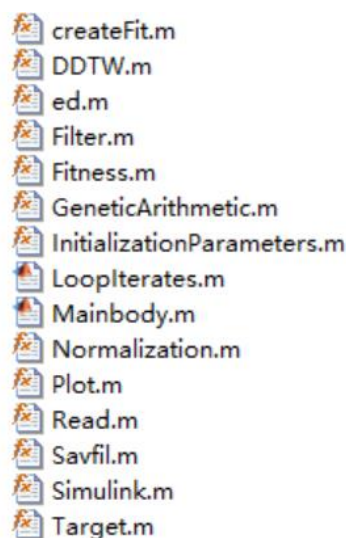


Fig 1-1 All code files for this program

### 1.4 Operational prerequisites

The code file, parameter file, target data file, and Simulink model for optimization are required to run this program, it is recommended to put them in the same folder and the folder is in the MATLAB path (**MATLAB path added**).

The current MATLAB folder must be the folder when the code is run.

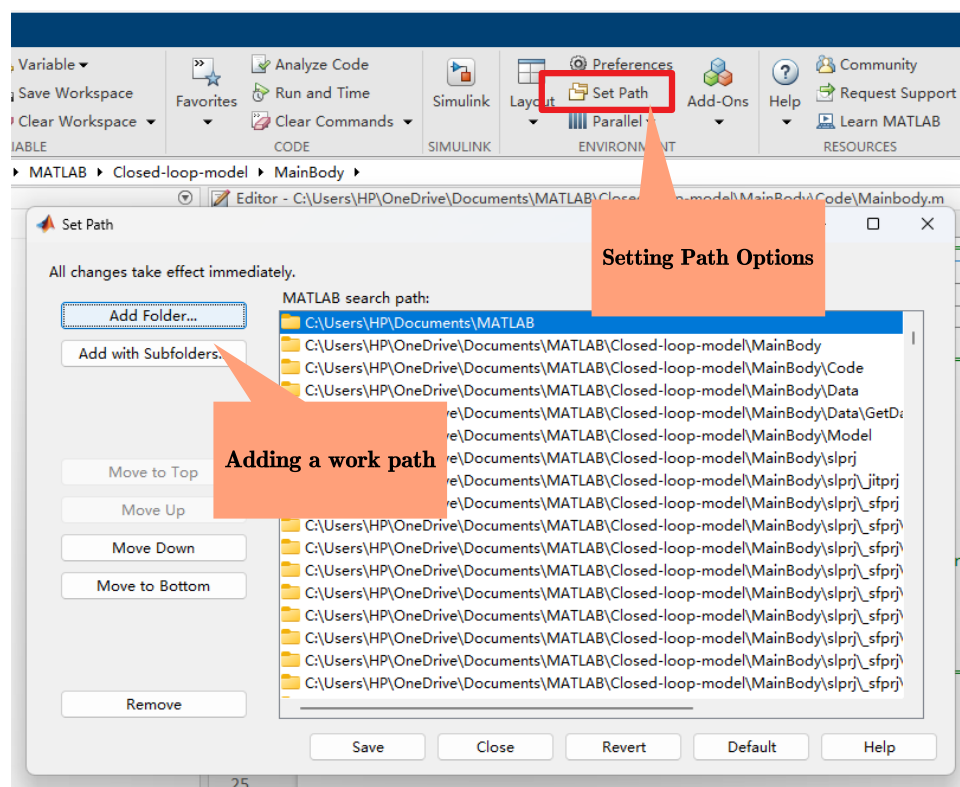


Fig 1-2 Setting up the work path in the main MATLAB interface

2. Parameter file

2.1 Prerequisites for use

The parameter file must be in the MATLAB work path.

To facilitate the user to carry out the parameter optimization of multiple centralized parametric models at the same time, the parameter file of this program is in txt text format, which is easy to be modified for the user. The name of the parameter file can be decided by yourself, but it should be consistent with the name input area of the parameter file read in the "Mainbody" or "LoopIterates" code.

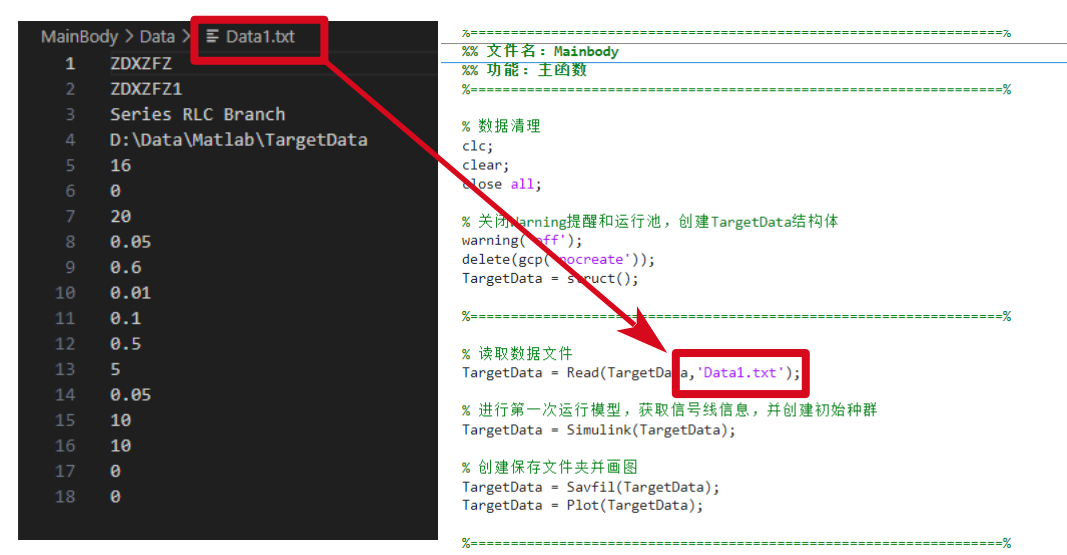


Fig 2-1 Parameter file (left) with corresponding parameter name input fields in the main Mainbody code (right)

2.2 File formats

The parameter file format needs to strictly follow the format given on the right-hand side of Fig 2-2 (MATLAB reads the corresponding parameters line by line).

1	ZDXZFZ	1	模型文件名称（不需要后缀名，且放在Matlab目标路径内）
2	ZDXZFZ1	2	运行模型文件名称（不要和模型文件名称相同）
3	Series RLC Branch	3	读取的RLC名称前缀
4	D:\Data\Matlab\TargetData	4	保存文件的文件夹地址
5	16	5	TargetData.Info(1) 记录仿真结束时间
6	0	6	TargetData.Info(2) 决定是否修正输入目标曲线时间系数
7	20	7	TargetData.Info(3) 记录种群个数popSize
8	0.05	8	TargetData.Info(4) 记录交叉率crossRate起始值
9	0.6	9	TargetData.Info(5) 记录交叉率crossRate阈值
10	0.01	10	TargetData.Info(6) 记录变异率mutationRate起始值
11	0.1	11	TargetData.Info(7) 记录变异率mutationRate阈值
12	0.5	12	TargetData.Info(8) 记录参数选取范围Range
13	5	13	TargetData.Info(9) 记录规定迭代终止次数
14	0.05	14	TargetData.Info(10) 记录遗传算法中最低可接受适宜度差值百分比fitnessdiff
15	10	15	TargetData.Info(11) 记录遗传算法中更新选取数据的适宜值相同次数。
16	10	16	TargetData.Info(12) 记录Parpool的个数（视电脑性能而定）
17	0	17	TargetData.Info(13) 决定计算适宜值方式（0为EU，1为DTW，2为DDTW）
18	0	18	TargetData.Info(14) 决定计算适宜值时是否选取点个数

Fig 2-2 Parameter file (left) with corresponding parameter file format (right)

## 2.3 Document parameters

### 2.3.1 Line 1

That is, the model file name (no suffix required) must be entered correctly (by which MATLAB looks for Simulink model files).

### 2.3.2 Line 2

That is the file name of the copy of the model file, which can be changed by yourself (no suffix is needed because the program needs to be run for the model file to be modified, so to retain the original model file, this program modifies the model file and then saves it as a copy).

### 2.3.3 Line 3

That is, the name of the RLC module that needs to be optimized, the default is Series RLC Branch, if you want to optimize for a specific RLC module, then modify the name of the corresponding module to "specific name + number (the number starts from 1, and interruptions are not recommended), and the third line only enter the specific name (the program determines the RLC module that needs to be optimized by its name). The third line contains only the specific name (the program uses the name to identify the RLC module to be optimized).

### 2.3.4 Line 4

In the folder where the output data is saved, it is recommended to copy and paste from Explorer.

## 2.4 TargetData.Info array

Starting at line 5, from the bottom are the contents of the TargetData.Info array, an array of parameters used in connection with the direct program-oriented input to the genetic algorithm.

### 2.4.1 *TargetData.Info(1)*

Record the simulation stop time.

### 2.4.2 *TargetData.Info(2)*

Determines whether to correct the input target curve time coefficient. This parameter is for the time parameter of the target data file, if the time axis of the target curve data in the target data file is in the start-termination time interval of the first row then enter 0, if not and starting from 0 then enter 1, refer to **Fig 4-1**.

### 2.4.3 *TargetData.Info(3)*

Record the number of individuals in the population in the genetic algorithm.

### 2.4.4 *TargetData.Info(4)*

Record the starting value of crossRate crossRate with an input interval of (0,1).

### 2.4.5 *TargetData.Info(5)*

Record the crossover rate crossRate threshold with an input interval of (0,1).

### 2.4.6 *TargetData.Info(6)*

Record the mutation rate mutationRate starting value with an input interval of (0,1).

### 2.4.7 *TargetData.Info(7)*

Record the mutation rate mutationRate threshold with an input interval of (0,1).

### 2.4.8 *TargetData.Info(8)*

Record the parameter selection range, the input interval is (0,1) This parameter is more important, involving population generation, parameter mutation, and other links. The number indicates that the parameter random selection process in the parameter around the field of multiples of the interval of random selection, such as the parameter  $Q$  size of  $A$ , TargetData.Info (8) is set to  $x$ , the mutation link parameter  $Q$  size mutation for  $((1 - x) * A, (1 + x) * A)$  in the interval of the random value. Generally the larger  $x$  is, the larger the solution space is determined.



### **2.4.9 TargetData.Info(9)**

Record the number of prescribed iteration terminations.

### **2.4.10 TargetData.Info(10)**

Record the minimum acceptable fitness difference percentage fitnessdiff in the genetic algorithm with an input interval of (0,1).

### **2.4.11 TargetData.Info(11)**

Record the same number of times in the genetic algorithm to update the suitable values of the selected data. This parameter is related to TargetData.Info(10) measures the convergence speed of the genetic algorithm, which is related to the crossover rate and mutation rate. When the genetic algorithm converges faster, the crossover rate and mutation rate will be lower; when the genetic algorithm converges slower, the program will record the percentage difference between the fitness of the output of the iteration and the previous one. If TargetData.Info(10) is  $x$  and TargetData.Info(11) is  $n$ , when the percentage of difference is lower than  $x$  redundant  $n$  times, the genetic algorithm iteration is considered to be in stagnation, and at this time, the crossover rate and mutation rate will dynamically rise to the threshold.

### **2.4.12 TargetData.Info(12)**

Record the number of Parpool, this parameter depends on the performance of the computer, in general, 2G memory can expand a Matlab parallel pool.

### **2.4.13 TargetData.Info(13)**

Decide how to calculate the fitness value (0 for Euclidean distance EU, 1 for DTW, 2 for DDTW)

### **2.4.14 TargetData.Info(14)**

Decide whether to select the number of points when calculating the fitness value, if the output curve itself in the selected period of the number of points is small, then enter 0; if the output curve in the selected period of the number of points is large, then enter 1, more applied to the function **Fitness**.

### 3. Model settings

#### 3.1 Prerequisites for use

The Simulink model file must be in the MATLAB work path!

Industry please make sure that the Simulink model can run properly, (the model comes with initial parameters).

#### 3.2 Signal Line Setting

##### 3.2.1 Module Connections

As shown in Fig 3-1, the signal line is connected between the oscilloscope module (Scope) and the measurement module (Voltage Measurement, Current Measurement). Before selecting the signal line, you need to build the oscilloscope module, i.e. connect the circuit, the measurement module, and the oscilloscope module.

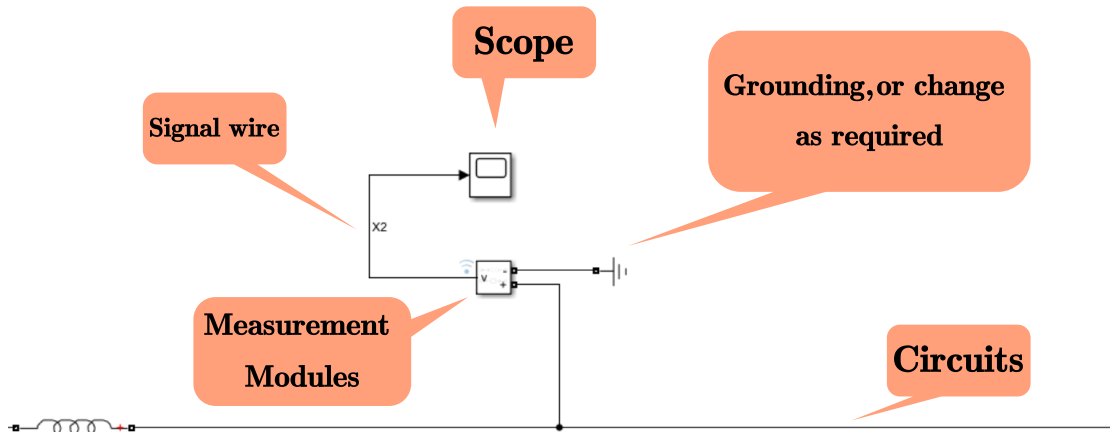


Fig 3-1 Module Connection Diagram

##### 3.2.2 Recording signals

For the Simulink model, the signal line corresponding to the oscilloscope module needs to be recorded to record the output curve information in the MATLAB workspace.

First, you need to change the name of the signal line, which should be the same as the corresponding target data file name; second, you need to record the signal on the signal line. This can be done in the Simulink main interface or the signal table (Fig 3-2).

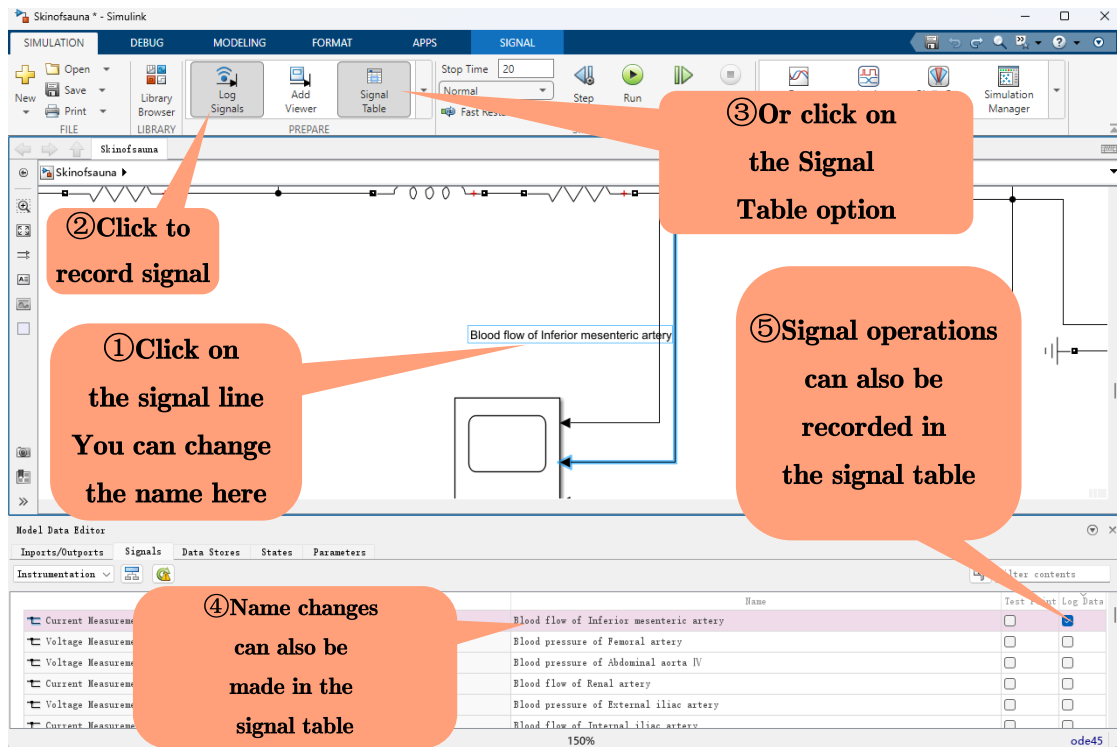


Fig 3-2 Signal Line Record Setting

## 4. Target data

### 4.1 Prerequisites for use

Ensure that the target data file is in the current MATLAB workspace or the MATLAB work path and is in txt format. Ensure that there are no data for two points with the same time in the target data.

### 4.2 Data files

The target data file is mainly used to input the target curve, so it is necessary to confirm the time point and curve information of the output curve corresponding to the target curve.

The first line of the target data file corresponds to the start time and final time of the output curve (in general, the time length of the output curve is equal to TargetData.Info(1), but the target data only needs to correspond to a period or a cycle, so it is necessary to determine the start time and final time of the corresponding output curve). The final time should be less than or equal to TargetData.Info(1).

Since the first line below the target curve data, in general, there will be two cases, as shown in **Fig 4-1** one is the target curve data in the target data file time axis in the first line of the start and end time interval, the other is not and from 0, both can be, but to correspond to the modification of TargetData.Info(2).

Blood pressure of Abdominal aorta 1.txt			X1.txt		
1	15.2	16	1	15.2	16
2	0.0129832	70	2	15.2010057495799	33.9235899713131
3	0.0387897	68.2769	3	15.2010057495801	33.9235899713103
4	0.0645161	66.8	4	15.5671657571107	43.7606352253840
5	0.0903226	66.0615	5	15.5693289839440	43.7044262790625
6	0.109677	73.6923	6	15.5693289839441	43.7044262790595
7	0.141935	83.7846	7	15.5754629766012	43.5494683688715
8	0.16129	104.708	8	15.5815969692582	43.3991192103559
9	0.187097	111.108	9	15.5900916863988	43.1940834332388
10	0.212903	111.108	10	15.5905864035393	42.9890242752736
11	0.225806	108.154	11	15.6070811206798	42.7818633962950
12	0.258065	102.246	12	15.6186554944770	42.4960409765643
13	0.270968	97.3231	13	15.6331640850677	42.1351688150486
14	0.296774	93.3846	14	15.6476726756584	41.7755791559951
15	0.322581	93.6308	15	15.6655858631073	41.3364975561110
16	0.348387	93.6308	16	15.6834990505563	40.9026854493624
17	0.412903	90.6769	17	15.7071142038109	40.3372302961891
18	0.451613	87.7231	18	15.7360060100097	39.6550747665050
19	0.490323	85.5077	19	15.7648978162084	38.9843806796612
20	0.56129	82.5538	20	15.8017396967316	38.1460047008203
21	0.658065	76.8923	21	15.8519372563609	37.0334298732315
22	0.735484	71.9692	22	15.9307653279528	35.3536988819205
23	0.787097	69.7538	23	15.9909999999999	33.9439329513789
			24	16	33.9439329513764

**Fig 4-1** Two cases of target curve data, the left is not in the first row of the time interval, and the right opposite

## 5. Documentation of functions

### 5.1 Prerequisites for use

Make sure the function file is in the current MATLAB workspace or the MATLAB working path.

### 5.2 Main function files

#### 5.2.1 Mainbody

The main function file shown in **Fig 5-1**, when the parameter file, mode settings, and target data preparation are complete, you can start running the code in the MATLAB code interface, you can modify the parameter file selection and MATLAB parallel pool reset time (MATLAB parallel pool calculations, a single parallel program calculations can be slow leading to the parallel pool of the overall computational inefficiencies, so if the parallel pool running time is recommended to reset the parallel pool) Parallel pool reset time.

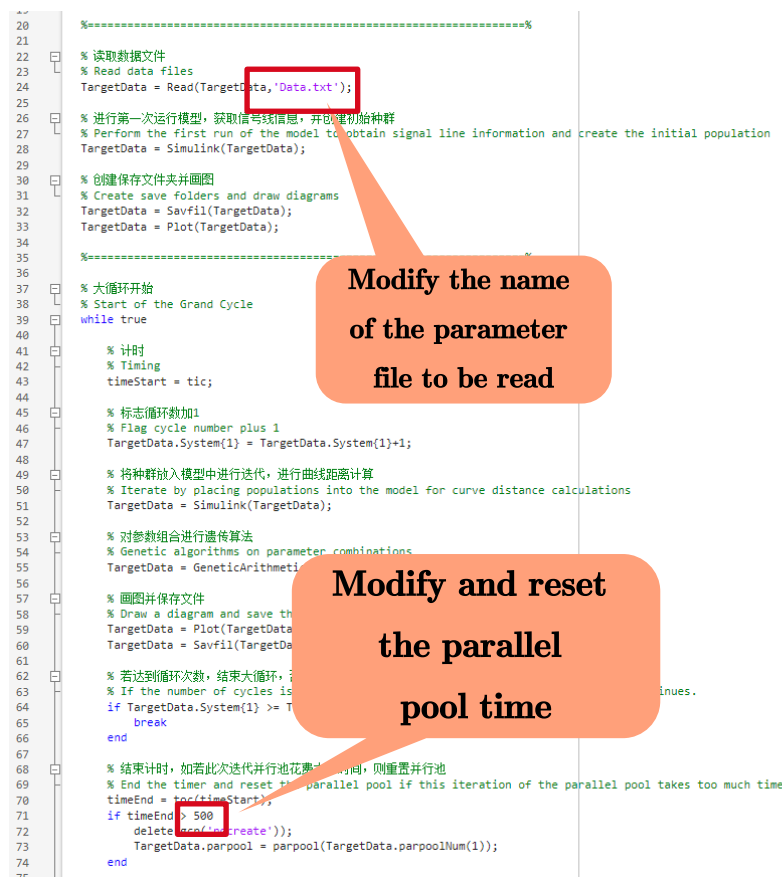


Fig 5-1 Mainbody Code Interface

### 5.2.2 *LoopIterates*

This main function code only applies to the optimization of a target curve in turn, if the model file records  $n$  signal lines at the same time, run this code will be optimized  $n$  times, each time only for a signal line optimization, save the data will be saved separately, you can modify the parameter file selection and MATLAB parallel pool reset time.

## 5.3 *Function files*

### 5.3.1 *createFit*

MATLAB native Fourier fit curve code file for fitting the input target curve points to a smooth curve for use in the **DDTW** function.

### 5.3.2 *DDTW*

The fitness function that calculates the distance from the curve can be categorized into three calculations, EU, DTW, and DDTW, depending on the input, for use in **Fitness**.

### 5.3.3 *ed*

Determine the estimated derivatives of each point in the computed DDTW to be used in the **DDTW** function.

### 5.3.4 *Filter*

Output curve because of precision problems caused by the selection of points in the array of points with the same time, this function is to filter these points for **Fitness**.

### 5.3.5 *Fitness*

Calculate the fitness, extract the points in the target and output curves to form a time series, and process them into **DDTW** to calculate the distance and finally output for **Simulink**. MATLAB parallel pooling is involved.

### 5.3.6 *GeneticArithmetic*

Imported parameter combinations for the genetic algorithm's cross-variate selection process for use in the master function.

### 5.3.7 *InitializationParameters*

Initialize the population, determine the number of MATLAB parallel pools, and turn on

MATLAB parallel pools for Simulink.

### 5.3.8 Normalization

Perform normalization of curve data for Fitness.

### 5.3.9 Plot

Responsible for popping up the curve schematic window (shown as **Fig 5-2**), schematizing the fitted target curve and with the output curve, and saving the image in a subfolder in the path of Line 4 of the parameter file for use in the main function.

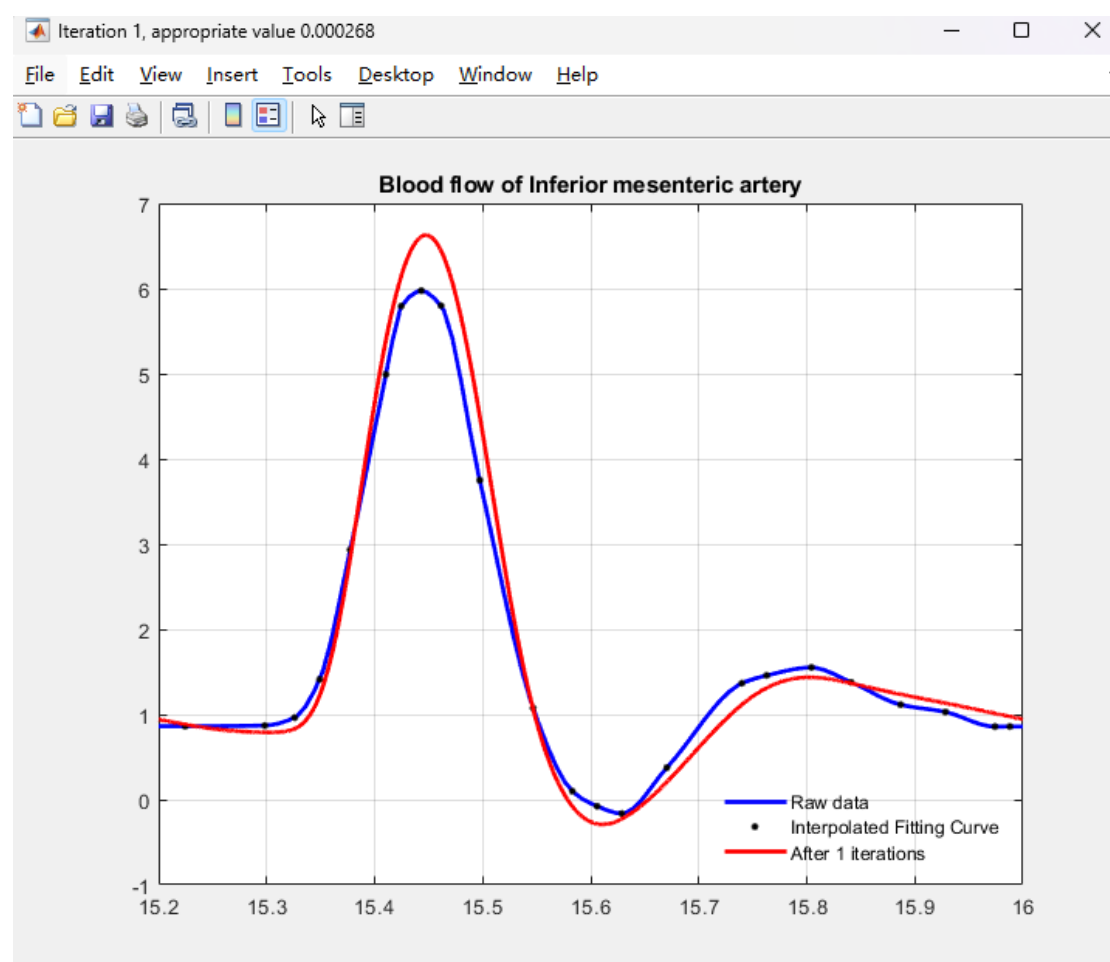


Fig 5-2 Plot function saves a picture of the output curve (red) and the target curve (blue)

### 5.3.10 Read

Read the parameter file information, its call function inputs in the main function that is, the name of the parameter file that the user wants to call, the function stores all the information in the parameter file into the TargetData structure and locates the RLC parameters of the optimization model, modifies the values of its parameters into matrix elements (for easy

modification of the iteration), determines the number of signal lines that need to be recorded in the model and obtains their information, creates and saves a copy for participating in the iteration for use in the main function.

### ***5.3.11 Savfil***

Determine the save file path, and file name, and create subfolders for saving the target curve information, parameter combinations of the best individual for each iteration, output curve information, and fitness for use in the main function.

### ***5.3.12 Simulink***

The model is called to obtain Simulink model information, Simulink model simulation is performed, and the output MATLAB parallel pooling is involved.

### ***5.3.13 Target***

Fit the target curve discrete points and Fourier fit the curve data and import it for **Simulink**.



## **6. TargetData structure**

### **6.1 Introduction to Structures**

TargetData structure (Matlab struct) is the important information carrier of this program to connect each function when running, this program is a fully automated parameter optimization code program, so the use of the level of the structure does not require the user to understand the adjustment, but as an open source code, if the user wants to optimize the modification of the code level, it is very important to understand the TargetData structure. However, as open-source code, it is important to understand the TargetData structure if you want to optimize your code.

The following can be viewed by opening the TargetData structure in the MATLAB workspace at the end of the program run.

### **6.2 Structure details**

#### **6.2.1 TargetData.mdlName**

Cell array storing the first 4 lines of the parameter file, i.e., the contents of 2.3.

#### **6.2.2 TargetData.Info**

Double array storing data from line 5 and onwards of the parameter file, i.e. 2.4 contents.

#### **6.2.3 TargetData.System**

Cell array that holds system information while the program is running.

Table 6-1 Detailed role of each parameter of TargetData.System array

TargetData.System	Functionality
1	Record the number of iterations
2	Number of recorded signal lines
3	Record the number of RLC parameters to be optimized
4	Record the number of repetitions of the appropriate value
5	Record the fitness value of the best individual for each iteration
6	The storage location of the drawing window
7	Record the curve image distribution matrix of the drawing window
8	Record output data and save path
9	Record the path and name of the subfile for each iteration under the path where the output data is saved

#### 6.2.4 TargetData.lineName

Cell array that holds the names of the signal lines in the model where signals need to be recorded.

#### 6.2.5 TargetData.Targetcell

Cell array, which records the target curve information, is divided into five columns, storing the signal line name, the corresponding target curve data, the corresponding start time of the output curve, the corresponding final time of the output curve, and the smoothed curve that is Fourier-fitted based on the curve data.

#### 6.2.6 TargetData.pop

Double array, storing the positions of RLC parameter combinations, is an  $n \times m$  matrix,  $n$  is TargetData.Info(3) and  $m$  is TargetData.System{3}, which mainly participates in the functions GeneticArithmetic, InitializationParameters, and Simulink.

#### 6.2.7 TargetData.parpoolNum

Double array, a matrix to record the number of parallel pools. Because the number of populations is not necessarily equivalent to the number of Matlab parallel pools and often may be more than the number of parallel pools, the parallel calculation needs to confirm the number of individuals in each calculation of the population, if the number of populations is greater than the number of specified parallel pool, the population is divided into several times and then brought into the parallel pool for calculation, if the number of populations is less than the number of specified parallel pool, then the number of populations is direct to the number of the

parallel pool.

### **6.2.8 *TargetData.parpool***

ProcessPool, which stores the MATLAB parallel pool.

### **6.2.9 *TargetData.IterationData***

SimulationOutput, storage of parameter combinations in the model after iteration output to the signal line information, including but not limited to the output curve of the point of the data, mainly involved in the function GeneticArithmetic, Plot.

### **6.2.10 *TargetData.fitness***

Double array storing the fitness values output by Fitness for each individual in a population.

### **6.2.11 *TargetData.popOther***

Cell array storing the suitable values and parameter combinations of the best individuals in the population after each iteration.

## 7. Output files

The output file save path is determined by **Line 4** of the parameter file.

The folder is automatically generated when the program is run. If you run the "**Mainbody**" code, the generated sub-file is called "Model file name Parameter selection range, fitness function calculation method Population size Specified number of iterations", and its sub-folder is shown in **Fig 7-1**.

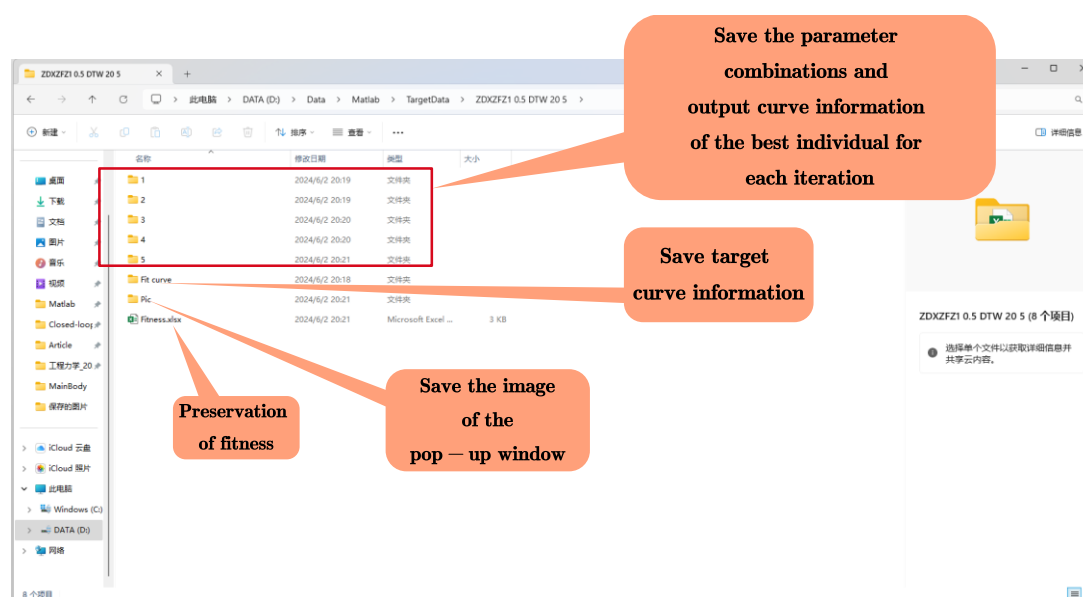


Fig 7-1 Subfile format

If you are running "**LoopIterates**", then the first name will be the name of the signal line you are targeting, and then you will generate the subfolders and data described above.