Tai-Ji MPC 5.0

Versions 5.0.0.2

User's Guide

Hangzhou Tai-Ji Yucai Software Co., Ltd.
Room 590, 5th Floor, Building 6, No. 262 Wantang Road, Xihu District, Hangzhou, Zhejiang Province
Contact: Zhu Yujie
Phone: +86 18069891080
Email: info@tjcontrol.com
Website: http://www.tjcontrol.com/

Contents

1 Introduction

1.1 Software Introduction

Tai-Ji MPC is a model predictive controller software package developed by Hangzhou Taiji Yucai Software Co., Ltd. Tai-Ji MPC is used for online control and economic optimization of multivariable industrial production units to increase product revenue, improve product quality, enhance production safety and stability, as well as save energy, reduce consumption, decrease raw material usage, and reduce pollution.

Once the MPC controller design is specified, Tai-Ji MPC can automatically perform controller deployment and maintenance. The MPC controller design includes tables for manipulated variables (MV), disturbance variables (DV), and controlled variables (CV), as well as their control strategies, which include MV limits, CV limits or setpoints, economic optimization parameters, etc. Based on the MPC controller design, Tai-Ji MPC can automatically perform online model identification and MPC control parameter settings. If the MPC controller performance degrades due to model degradation, Tai-Ji MPC can automatically maintain the controller, including re-identification and replacement of the model and re-tuning of control parameters.

Users can provide two formats of test data: 1. Matlab 6.5 version .MAT files 2. Excel CSV files, import the above files into Tai -Ji Online and execute model identification and MPC controller.

Tai-Ji MPC is designed for system control engineers, process engineers, and operators involved in model-based process control. Tai-Ji MPC users do not need to master advanced theories of system identification and MPC control. Tai-Ji MPC can also serve as a useful tool for scholars and students in the field of automatic control to learn MPC control strategies.

Tai-Ji MPC can run on Windows 10 Professional operating system. The communication method between Windows and DCS is OPC (OLE for Process Control).

1.2 Control Engineering Introduction

1.2.1 Structure

Tai-Ji MPC consists of three components: MPC Control Component (see Control Component), Online Identification Component (see Identification Component), and Control Performance Monitoring Component (see Monitoring Component). Figure 1 is the control system block diagram of the Tai-Ji MPC controller. Note: The Control Performance Monitoring Component is under development, stay tuned.

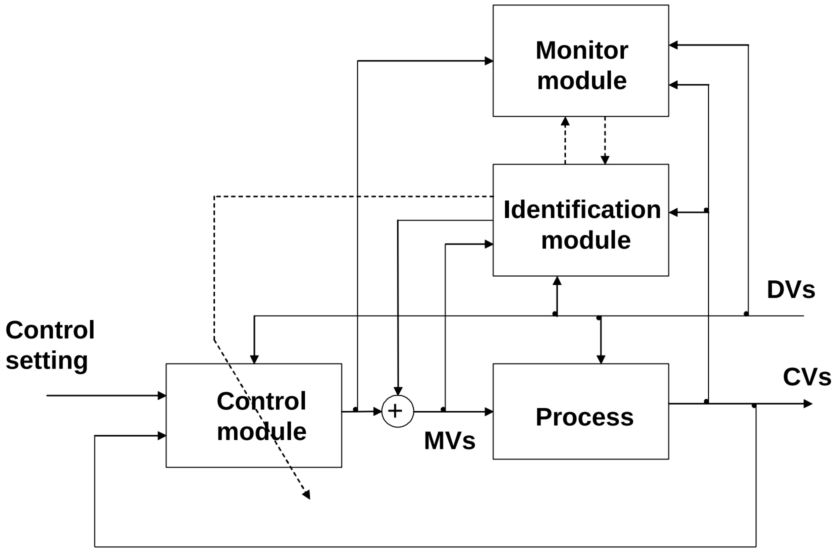


Figure 1: Tai-Ji MPC controller control system block diagram

These three components semi-automatically complete their respective tasks and coordinate with each other to achieve MPC control. Assuming the design of an MPC controller is given, during the commissioning of the MPC controller, the online identification component automatically performs identification experiments and automatic model identification. If some of the identified models are of good quality and the control system simulation results are ideal, the MPC controller automatically uses these models during the identification experiment to automatically control the corresponding manipulated variables (MV), disturbance variables (DV), and controlled variables (CV). As the experiment progresses, more and more models are fed into the MPC controller, and more and more MVs, DVs, and CVs are brought under automatic control. When all the desired models are of good quality and are used by the MPC controller, the online identification component stops working, and the commissioning of the MPC controller is completed.

When the MPC controller is working online, the control monitoring component continuously monitors the performance of the MPC. When poor control performance and poor model quality are detected, the control monitoring component activates the online identification component. While the MPC controller continues to work, identification experiments and model identification begin. During the experiment and identification process, poor quality models are gradually replaced by new high-quality models. Once all poor-quality models are replaced, the online identification component stops working, and the maintenance of the MPC controller is completed. Note: The control performance monitoring component is currently under development and has not yet been completed.

Tai-Ji MPC can perform identification experiments, model identification, controller simulation, and controller deployment in parallel, significantly reducing the deployment cost of MPC. For most of the time, identification experiments are conducted in a closed-loop state, keeping the time for open-loop identification experiments to a minimum (typically 1 to 3 days for petrochemical processes), thereby greatly reducing interference with the production process. Meanwhile, Tai-Ji MPC is a highly integrated comprehensive software package that can automatically execute steps, thus significantly saving manpower.

1.2.2 Three Components and Their Integration

The following will briefly introduce the three components and their integration.

A) Identification Module

The identification component uses the Asymptotic Method (ASYM). The following will introduce how to use this method for automatic online identification of industrial processes.

1) Experimental Signal Design and Identification Experiment

The optimal power spectrum of the experimental signal can be obtained through asymptotic theory. Here, optimality means that the identified model is optimal for MPC control. The power spectrum of the experimental signal is achieved by improving Generalised Binary Noise (Generalised Binary Noise). The characteristics of the GBN signal depend on its average switching time and amplitude. The amplitude of the GBN signal is usually determined based on prior knowledge of the process, and studies have shown that the optimal average switching time of the GBN signal is related to the time it takes for the production process to reach a steady state, i.e., the steady-state time.

The experiment module generates experimental signals, writes them into experimental variables, and automatically executes the experiment, collecting and saving experimental data to be used in model identification. The identification experiment is a multivariable identification experiment, meaning it simultaneously excites multiple MVs or all MVs. For each MV, the identification experiment can be conducted in either open-loop or closed-loop states. When the MV is in an open-loop state, the experiment module operates on the full value of the MV. When the MV is in a closed-loop state, the experiment module only operates on the experimental signal, and the MPC controller modifies the mean value of the MV. During the identification experiment, you can adjust the MV's step size and evaluation switch time, and also switch the MV between open-loop and closed-loop states.

2) Parameter Estimation

Parameter estimation is divided into two steps: 1) Estimating a high-order ARX (Equation Error) model, 2) Performing model order reduction with frequency weighting. It can be proven that this parameter estimation method can achieve maximum likelihood estimation, which is the most accurate model obtainable from experimental data. Additionally, this method can obtain an unbiased model in closed-loop identification experiments.

3) Order Selection

The optimal order of the reduced-order model is determined by a frequency domain criterion. The basic idea of this criterion is to balance the bias and variance of the transfer function in frequency segments that are important for control.

4) Error Upper Bound Matrix for Model Validation

Based on the theory of gradual progression, the 3σ error upper bound of each transfer function in the identification model can be obtained. We can propose the following engineering method for model verification based on this error upper bound:

Classify the model by grade, that is, classify the model into four grades A (excellent), B (good), C (average), D (poor or no model) based on the error upper bound and the relative value of the frequency response at low and medium frequencies. Extensive simulation and engineering practice experience show that models of grades A, B, and C can be used in the controller. For models of grade D, the following treatment can be done:

1) If there is no model between MV and CV, set the grade D model to zero.
2) If there is a model between MV and CV, and the controller requires this model, adjust the ongoing identification experiment to improve the quality of the model.

To improve the quality of the model, the following adjustments can be made to the identification experiment:

- Increasing the step size of the experimental signal will reduce model error: When other experimental conditions remain unchanged, doubling the step size will halve the error across the entire frequency band.
- Increasing the experiment duration will reduce model error: When other experimental conditions remain unchanged, doubling the experiment duration will reduce the error across the entire frequency band by 1.4 times.
- Adjusting the average switching time of the GBN signal will change the frequency domain distribution of the error: Doubling the average switching time will halve the low-frequency error of the model, while halving the average switching time will halve the high-frequency error of the model.

Model identification and model validation can be performed as needed or at fixed intervals of every 200 sample points. The identification experiment can be adjusted based on the identified model. When the quality of most desired models is A and B, the identification experiment can be concluded.

Automatic Model Selection

A large industrial MPC controller often contains many MVs and CVs, and not all MVs and CVs are related. This means that the model transfer function matrix will have many zero transfer functions. Model selection determines whether a particular model is used in the MPC control module. This can be done automatically based on the results of model validation in the identification model and the process prior knowledge contained in the desired matrix. The rows and columns of the desired matrix correspond to CVs and MVs, respectively. The elements of the matrix have four values: '+',

'-', '?', ' No ': '+': There is a model between the corresponding MV and CV with positive gain; '-': There is a model between the corresponding MV and CV with negative gain; '?': It is uncertain whether there is a model between the corresponding MV (DV) and CV; ' No ': There is no model between the corresponding MV (DV) and CV.

The principle of automatic model selection is: **If the quality of a certain model is A, B, or C, and the model's symbols are consistent with the corresponding elements of the expected matrix, then this model will be deployed in the MPC controller.**

B) MPC Control Module

The MPC control component performs automatic tuning of MPC control parameters, MPC simulation, and online control. The MPC control algorithm uses a multi-objective layer optimization method to control each CV so that it matches the set value or stays within a certain range. When there are not enough degrees of freedom to control all CVs, the CVs to be controlled can be selected based on priority and weight factors. In steady-state economic optimization, linear programming (linear programming) and quadratic programming techniques are used, and each MV and CV is assigned a corresponding ideal resting value.

In each control cycle, the MPC control algorithm consists of the following three steps: 1) Prediction, 2) Steady-state optimization, 3) Dynamic control. The prediction step uses the identified model and the current measurements of related MVs, DVs, and CVs to predict the future values of CVs. These predicted values are then used in steady-state optimization and dynamic control.

In steady-state optimization, feasibility analysis is performed first, followed by economic optimization. Feasibility analysis checks whether there are enough degrees of freedom to control all CVs. If there are not enough degrees of freedom, CVs to be controlled can be selected based on priority and weight factors. If there are surplus degrees of freedom, economic optimization is performed. Economic optimization uses a combination of linear programming and quadratic programming:

$$\begin{aligned} &\min_{u} J = \|u - IRV_u\|_{w_u}^2 + \|y - IRV_y\|_{w_y}^2 + b_1^T u + b_2^T y \\ &s.t. \quad y = Gu + d(t) \\ &\qquad y_{\min} \le y \le y_{\max} \\ &\qquad u_{\min} \le u \le u_{\max} \end{aligned} \tag{1}$$

Among them, u is the MV vector, y is the CV vector, IRVu is the ideal reset value vector for MV, IRVy is the ideal reset value vector for CV, wu is the diagonal matrix of quadratic programming weights for MV, wy is the diagonal matrix of quadratic programming weights for CV, b1 is the linear programming weight vector for MV, b2 is the linear programming weight vector for CV, G is the model gain matrix, d (t) is the deviation at sampling time t, ymin and ymax are the upper and lower limit vectors for CV, respectively, and umin and umax are the upper and lower limit vectors for MV.

Assuming all parameters in the steady-state optimization have been determined in the MPC control design, the result of the steady-state optimization is the steady-state values of MV and CV, denoted as vector y^* and vector u^* respectively.

The dynamic control part of the MPC control algorithm calculates the MV control actions based on the predicted values of CV and the identified model, so that the production process reaches the steady-state values obtained in the steady-state optimization. The dynamic control calculation is also a quadratic programming problem:

$$\begin{aligned} \min_u J &= \sum_{j=1}^{P} \left\| y(t+j|t) - y^* \right\|_Q^2 + \sum_{j=1}^{P} \left\| \Delta y(t+j|t) \right\|_S^2 \\ &\quad + \sum_{j=0}^{M-1} \left\| \Delta u(t+j) \right\|_R^2 + \sum_{j=0}^{M-1} \left\| u(t+j) - u^* \right\|_{Ru}^2 \quad (2) \\ s.t. &\quad u_{\min} \le u(t+j) \le u_{\max} \\ &\quad \Delta u_{\min} \le \Delta u(t+j) \le \Delta u_{\max} \end{aligned}$$

In (2), the vectors y^* and u^* are the steady-state values of MV and CV determined in the steady-state optimization, P is the prediction horizon, M is the control horizon, Q is the diagonal matrix of CV weights, S is the diagonal matrix of CV increment weights, Ru is the diagonal matrix of MV weights, R is the diagonal matrix of MV increment weights.

The parameters for dynamic control include P, M, the weight matrix for each CV. Q, S, R, Ru, these parameters can be selected autonomously.

To achieve MPC control, the MPC control module needs to 1) automatically select the identified model and use it in control, 2) automatically tune the MPC control parameters.

Self-tuning of control parameters

The parameter self-tuning function can help users tune three weighting coefficients $\boldsymbol{Q}$, $\boldsymbol{S}$, $\boldsymbol{R}$. First, users need to normalize MV and CV according to historical data or process requirements as follows:

$$q_i = \frac{k_y}{\left(y_{i,\max} - y_{i,\min}\right)^2}, r_j = \frac{k_u}{\left(u_{j,\max} - u_{j,\min}\right)^2} \quad (3)$$

Among them, $y_{i,\max}$ and $y_{i,\min}$ respectively represent the maximum and minimum values in the historical data of the i-th CV; $u_{j,\max}$ and $u_{j,\min}$ respectively represent the maximum and minimum values in the historical data of the j-th MV. k_y and k_u are used to adjust the bandwidth of the entire closed-loop system, which can be set to $k_y = 1$, and k_u adjusted according to actual needs.

The output increment weighting coefficient s_i is given by the following formula:

$$s_i = \frac{q_i}{16}\left(T_i^{cl}\right)^2 \tag{4}$$

where T_i^{cl} represents the desired closed-loop response time of the i-th output (given by the user).

C) Monitoring Module

Temporarily unavailable.

1.2.3 Tai-Ji MPC Commissioning Process

The following describes how Tai-Ji MPC automatically completes the commissioning of MPC control.

Assume that the design of an industrial process MPC controller is given, which means that the MV, DV, CV, and the limits of MV and CV have been determined, as well as the parameters for steady-state optimization, i.e., the weights in the linear and quadratic programming of formula (1). The MPC user estimates the main steady-state time of the process based on operational experience and determines the appropriate amplitude of the experimental signals for all MVs in the identification experiment. Based on pre-experiments and operational experience, the user also constructs the desired matrix. All this information is input into the Tai-Ji MPC software package.

Users can now initiate identification experiments via mouse click or a single key press. During the experiment, the identification module and control module perform the following tasks:

1) The identification module excites all MVs based on the pattern and amplitude of the experimental signals and collects MV, DV, and CV experimental data.

2) Users monitor the identification experiment and adjust it as necessary to ensure stable industrial process operation. The steps are as follows: If all CVs remain within the normal operating range, continue the experiment without adjustment; if an open-loop CV drifts slowly, adjust the mean of the relevant MV according to the desired matrix; if an open-loop or closed-loop CV oscillates repeatedly at the upper and lower limits, reduce the amplitude of the relevant MV.

3) Model Identification: When the experiment reaches one-quarter of the planned time, the model identification component starts and uses the existing data to build the model. This process is repeated at regular intervals, for example, every time 100 new data samples are obtained. The user can also press the **Identify** button to start model identification.

4) Model Verification, and if necessary, adjust the experiment to improve model quality or reduce production interference. The steps are as follows: Each time it starts, the model identification module classifies the model into levels A (Excellent), B (Good), C (Average), and D (Poor) based on the upper bound of the model error. If certain MVs generate enough A and B models, and these models remain consistent with the expected matrix, then reduce the amplitude of these MVs to minimize interference with normal production operations. At the same time, the model identification module also calculates the upper bound of the future model error and model level

at the end of the planned experiment. If the future model level cannot reach A or B, then increase the amplitude of the relevant MVs to improve the signal-to-noise ratio.

5) After identifying the model, under the premise of ensuring that the model gain's sign matches the expected matrix, models with quality grades A, B, or C are sent to the control module.

6) Users can press the **Auto-tuning** button to perform self-tuning of control parameters and use the current model to simulate part of the control system. If the simulation results show good control performance, the control module will put the corresponding MV, DV, and CV into automatic control. As identification experiments and model identification continue, more and more models are loaded into the control module, and more and more MVs, DVs, and CVs are put into automatic control by the control module.

7) When most models reach grade A or B, stop the identification experiment. The actual experiment time may be slightly shorter or longer than planned. Users can use all obtained models to tune parameters and simulate the control system. If the simulation results indicate good control performance, all MVs, DVs, and CVs are put into automatic control, completing the commissioning of the MPC controller. The MPC controller parameters can be fine-tuned by control experts if necessary.

Note: During the above commissioning process, the identification experiment initially proceeds in an open-loop state, meaning CVs are not under automatic control. When certain MVs, DVs, and CVs are put into automatic control, the identification experiment enters a partial closed-loop state. Finally, when most models are loaded into the controller, all MVs and CVs are put into automatic control. This function of putting MVs and CVs into automatic control during the identification experiment reduces the interference of the identification experiment with the operation of the production unit.

2 Getting Started

2.1 System Requirements

Hardware: IBM-compatible computer with a CPU frequency above 1 GHz and 1 GB or more of memory
Operating System: Windows 2000/XP /Vista /07/08

2.2 Installing Tai-Ji MPC

Prepare the CD with the Tai-Ji MPC software and follow these steps:
1） Start the computer and log in as the system administrator
2） Insert the CD into the CD drive (assuming the drive letter is D:)
3） Run D: TaiJiMPCSetup.exe and follow the installation instructions. Please select to install all components.

The installation program will automatically create the following directories:
C:\Taiji Main directory of the program

C:\Taiji\Common\lib DLL files, and the program to convert Tai-Ji ID data files into Tai-Ji MP

project files

C:\Taiji\Tools\bin Driver and upgrade programs for software lock, and the program to convert DMC files into

Tai-Ji MPC project files

C:\Taiji\TaiJiMPC\bin Tai-Ji MPC files

C:\Taiji\TaiJiMPC\Demo Tai-Ji MPC demo project files

C:\Taiji\TaiJiOPCSim\bin OPC server simulation program

C:\Taiji\TaiJiOPCSim\model Simulation model

C:\Taiji\Uninstall Uninstall program

2.3 Install simulation test environment

2.3.1 Install old version simulation test environment

Tai-Ji MPC uses OPC as the standard method to communicate with DCS/PLC systems. Users need to ensure that the OPC DA components are installed. If not, please run Tai-Ji MPC CD's OPC_DA20_Components.exe to install. If using PHD, please ensure that the PHD components are installed.

2.3.2 Install New Version Simulated Test Environment

Usually OPC versions are older, if it does not run properly, you can refer to the following methods to install the new version.

The new version of OPC is compatible with older versions OPC1.0 , 2.0, and 3.0 , and can add new groups and new items . The installation process is as follows:

1) Download and install .NET Framework 2.0 or newer versions, the download link is as follows:
   http://msdn.microsoft.com/en-us/netframework/cc378097.aspx
2) Run `OPCComponents\OPCCoreComponentsRedistributable(x86).msi` to install OPC core components
3) Run `OPCComponents\TaijiOpcDonet.msi` to install OPC COM components for .NET
4) Run `OPCComponents\TaijiOpc.msi` . This will create a new Tai-Ji MPC project file `TaijiOPC.dll` used by the software. Open the folder `TaiJiMPC\bin`, back up this `TaijiOPC.dll` , and then copy the new `TaijiOPC.dll` to this folder to complete the OPC update.

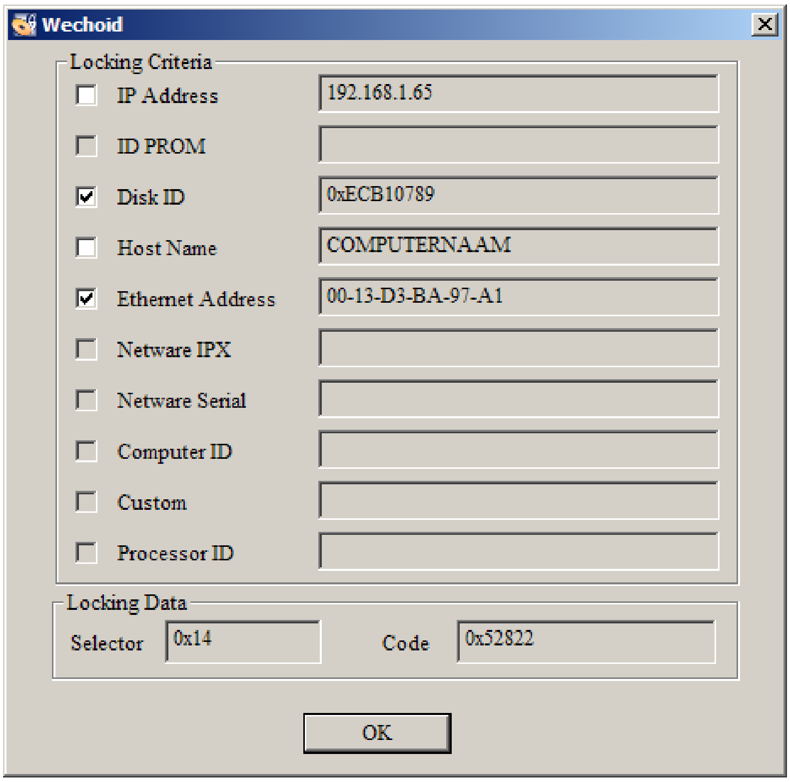
2.4 Tai-Ji MPC Software Encryption Protection System and Update Process

Tai-Ji MPC is protected by the following two methods: 1) Sentinel RMS Software key (license); 2) HASP hardware USB key . When users receive Tai-Ji MPC software, they will be informed which protection method to use.

Sentinel RMS Software key (license)

Users can obtain this key by binding the license to the computer. After completing the Tai-Ji MPC installation, follow the steps below to obtain the software key

1. Run `C:\Taiji\TaiJiMPC\LicenseTool\wechoid.exe` , the following window will be displayed



2. Select “Disk ID” and “Ethernet Address” , uncheck other options as shown in the figure above .
3. Send the above screenshot to Tai-Ji Control , you will receive a file named *lservrc* license file .
4. Copy this file to C:\Taiji\TaiJiMPC\Bin ; you can then run Tai-Ji MPC .

Software License Notes :

- The license is bound to the computer running wechoid.exe and cannot be used on other computers.
- For licenses with a validity period, do not change the computer's date . When the time is not correct , the license will pop up a warning message, please change it to the correct date to ensure the license works properly.

HASP USB key

Tai-Ji MPC software is protected by a HASP key. Before using this software, you need authorization from Tai-Ji Light Control Software Company. When you purchase our product, we will provide you with a USB dongle. You need to insert the USB dongle into the USB port of the computer running the Tai-Ji MPC software to use the software normally. During the installation of Tai-Ji MPC, the software driver for the dongle will also be automatically installed.

Manual installation of HASP driver software precautions:
If the automatic installation fails, you can run C:\ Taiji\ Tools\bin\ HASPUserSetup.exe to reinstall.

When a new version of Tai-Ji MPC is released, the security key related to the old version of Tai-Ji MPC may no longer be applicable, please update to the new version of Tai-Ji MPC.

How to update the dongle?

A. Retrieve dongle information
   1. Ensure the dongle is connected to your computer.
   2. Run C:\ Taiji\HaspDrivers\ hasprus.exe
   3. **In the Collect Key Status Information interface, select the button Collect information , choose a file name, and the information will be saved to that file.**
   4. Send the file via email to Tai-Ji YuCai Software Company, and we will send you an RUS password file via email.

B. Update your dongle
   1. Ensure the dongle is connected to your computer; copy the RUS password file we sent you to the installation directory of Tai-Ji MPC.
   2. Run C:\Tai j i \HaspDrivers\ hasprus.exe .
   3. In the **Apply License Update** interface, select the **RUS** password file name, and choose **Apply Update** .
   4. The dongle has now been updated.

Note: An **RUS** password can only update one dongle.

2.5 Multi-language Version Settings

The program automatically detects the operating system. If it is a Chinese operating system, the Chinese interface is selected. If it is an English operating system, the English interface is selected. If you need to change the default language of the interface, you need to edit the configuration file C:\Taiji\TaiJi MPC\Bin\TaiJiMPC.ini

1. After opening the file, find the following line
[Multi-Language]
# multi language ver : English(default), Chinese, Auto
# Multi-language version definition
Language = Auto

2. Change Language = Auto to:
    Language = Chinese to force the use of the Chinese interface
    Language = English to force the use of the English interface

Note: If the program is already running, you need to close and restart the program for the changes to take effect.

2.6 Start Tai-Ji MPC

After installing the Tai-Ji MPC program and inserting the dongle, select
    Start --→ All Programs --→ Taiji--TaijiMPC--TaijiMPC→→
You can start Tai-Ji MPC. Double-clicking C:\Taiji\TaiJiMPC\bin\TaijiMPC.exe can also start Tai-Ji MPC.

If an OPC simulation test environment is needed, run:
    Start --→ All Programs --→ Taiji--TaijiSimulation--Taiji OPC Server→→
Start the OPC simulation test environment and select the appropriate model. Double-clicking C:\Taiji\TaiJiOPCSim\bin\TaiJiOPCSim.exe
can also start the OPC simulation test environment.

3 Tai-Ji MPC Menus and Modules

3.1 Tai-Ji MPC Menus

Tai-Ji MPC Menus are

File	View	Toolbar	Help

File Menu

— **New:** Create a new project
— **Open:** Open an existing project
— **Save:** Save the current project. When model identification is in progress, save the model as a .sim file for use with the OPC simulator TaiJiOPCSim
— **Save As:** Save the current project under a different name
— **Recent Projects:** List of recently opened projects (up to 4)
— **Print:** Print the current window
— **Exit:** Exit Tai-Ji MPC

View Menu

— **Toolbar:** Show/Hide the toolbar
— **Status Bar:** Show/Hide the status bar
— **Log Window:** Show/Hide the log window

Toolbar Menu

—**Check Configuration:** Check communication connection, obtain current values of MV, DV, CV

—**Track MV Average Value** If selected, track the average value of the open-loop MV during identification experiments

—**Transpose MV, CV Matrix** If selected, transpose rows and columns when displaying MV, CV matrix

—**X-Axis as Time** If selected, the X-axis represents time; otherwise, it represents the number of sample points

—**View Internal State** View the maximum and minimum execution time of the MPC controller

Help Menu

—**About TaiJi MPC:** Display the version number and copyright information of Tai-Ji MPC

3.2 Tai-Ji MPC Module

Tai-Ji MPC includes the following 5 modules:

Configure	ID Test	Model ID	Controller Simulation	Controller

All functions of Tai-Ji MPC are included in these modules, which are arranged from left to right in the order of execution of the MPC project: 1) Project Configuration, 2) Identification Experiment, 3) Model Identification, 4) Simulation Controller, 5) Online Controller.

The corresponding windows for each module are as follows:

Configuration Module

General	MVs	DVs	CVs	Expectation

Identification Experiment Module

MVs	DVs	CVs	Test signal	Covariance

Model Identification Module

MVs&DVs	CVs	Model Response	Delay	Gain

Simulation Controller Module

MVs	DVs	CVs	Model	Gain	Tuning

Online Controller Module

MVs	DVs	CVs	Model	Gain	Tuning

Each module has 5 to 6 windows. Since the design of the MPC controller is determined by MV, DV, CV, and their upper and lower limits, all modules contain MV windows, DV windows, and CV windows. Some windows only have tables for MV, DV, and CV, while others have both tables and graphs.

If corresponding to the three components of Tai-Ji MPC in Section 1.2, the **ID Test** module and the **Model ID** module together form the identification component, while the **Simulation Controller** module and the **Online Controller** module together form the control component.

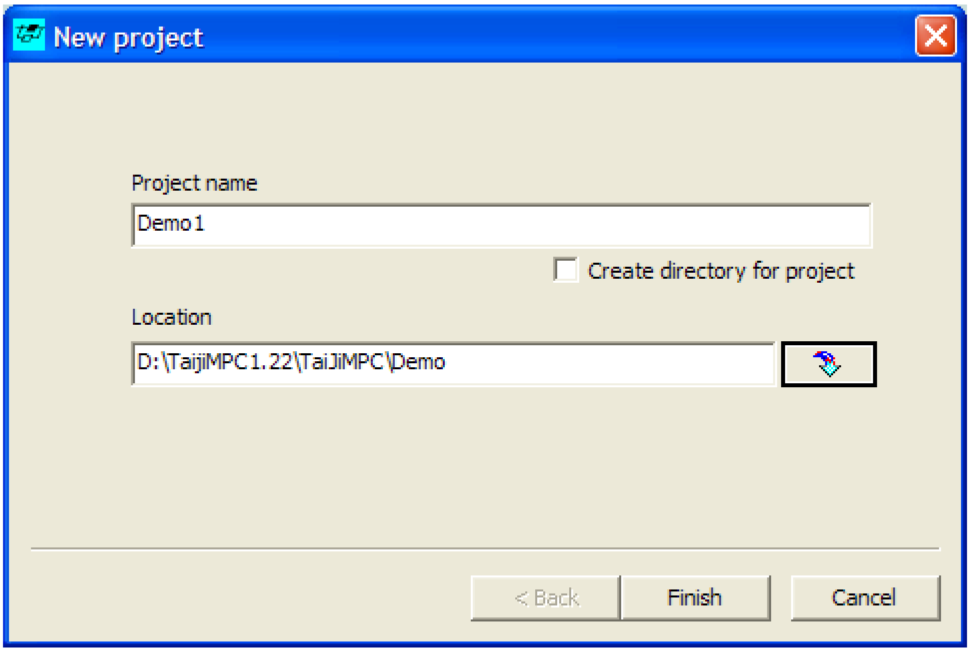
Users can quickly learn how to use these windows by following the examples in Chapter 4. For more detailed information, see Chapter 5.

4 Typical Examples

The purpose of this chapter is to help users quickly learn Tai-Ji MPC through a simple example.

4.1 Creating a New Project

Click **File → New** in the main menu to open the following window. You need to enter the project name and specify the location of the project folder:

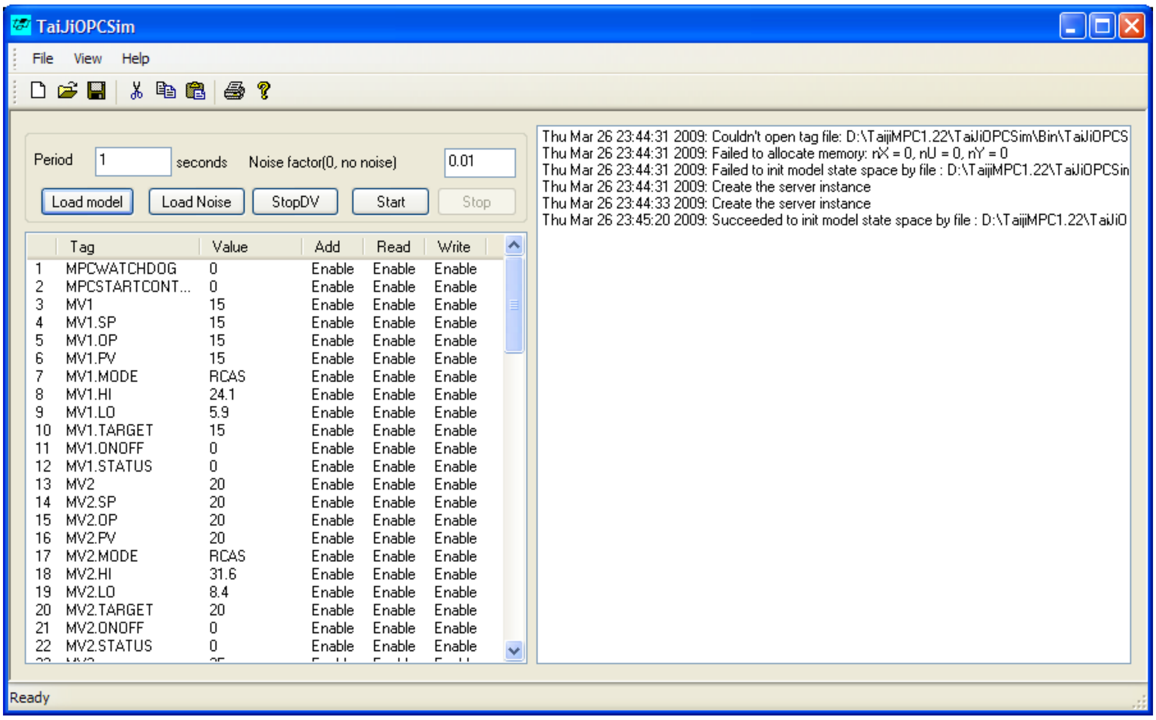


Click Finish button to create a new project Demo1.ojp in the directory D:\ TaiJiMPC5\ TaiJiMPC_Projects

OPC Simulation Server: TaiJiOPCSim

In this example, the production process is the distillation column model in the OPC simulation server TaiJiOPCSim. Double-click C:\Taiji\ TaiJiOPCSim\ bin\ TaijiOPCSim.exe to start the OPC simulation server and open the TaiJiOPCSim window. Click the **Load model** button to load the model file C:\Taiji\TaiJiOPCSim\Model\Distillation.ini. This model file is a text format file that defines the distillation column model parameters using S-domain transfer functions. The simulation server can convert the S-domain transfer function model into a discrete-time model based on the sampling time in the model file and simulate it.

Enter 1 in the **Period** field, which will set the model simulation time to 1 sample point/second. Note that the simulation sampling time is usually much shorter than the actual sampling time of the production process, mainly to speed up the required simulation time, i.e., time compression. Enter 0.01 in the **Noise factor** field, which defines the amplitude of the unmeasurable disturbance superimposed on the process output. Clicking the **Start** button will start the simulation.



Configure Basic Parameters (Configuration Module)

Click **Configure → General** to open the **General** window, where you can configure basic parameters such as the OPC server name, controller sampling time, process steady-state time, DCS data extension, DCS MPC user interface data extension, auto-identification, and model selection options.

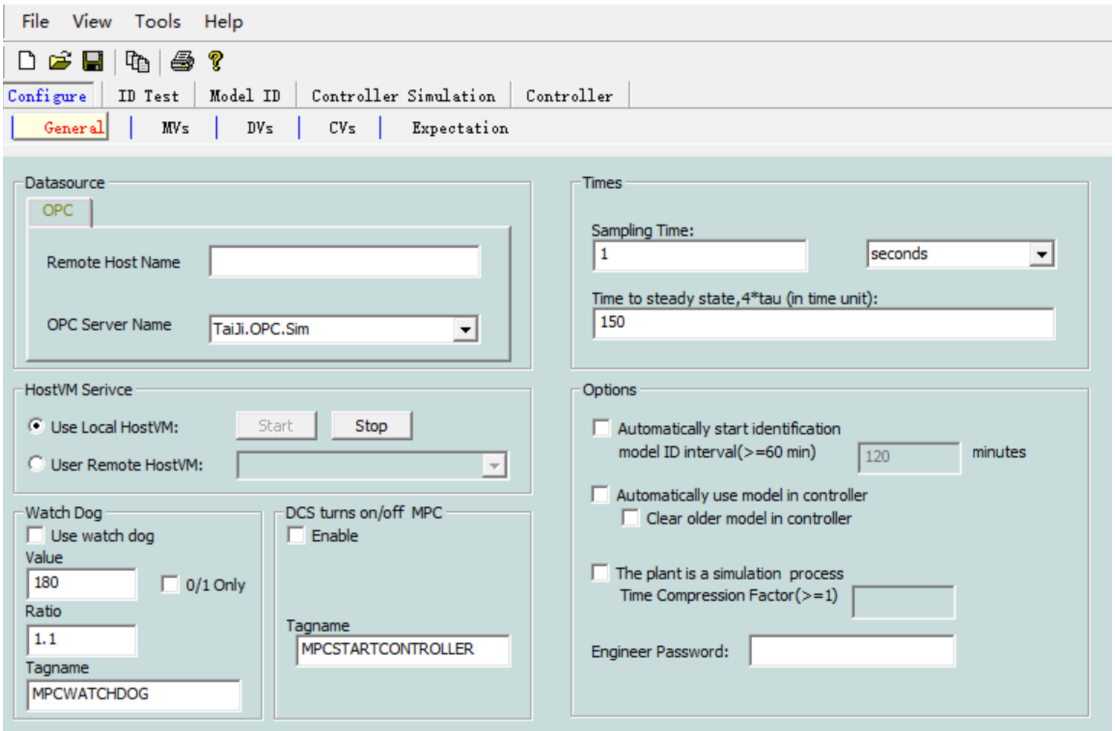
In the **Datasource** section, you can enter the OPC server name. The OPC server name for TaiJiOPCSim is “TaiJi.OPC.Sim” . Since the OPC server and Tai-Ji MPC are installed on the same computer, the **Remote host name** field can be left blank.

In the **Times** section, select the controller sampling time as 1 minute and the process steady-state time as 150 minutes. This parameter is used to generate the identification experiment signal and determine the duration of the identification experiment.

In the **HostVM** Service section, select 'Use Local HostVM', click 'Start', enable the local virtual machine service, and then you can call the control algorithm and identification algorithm of TaiJi MPC.

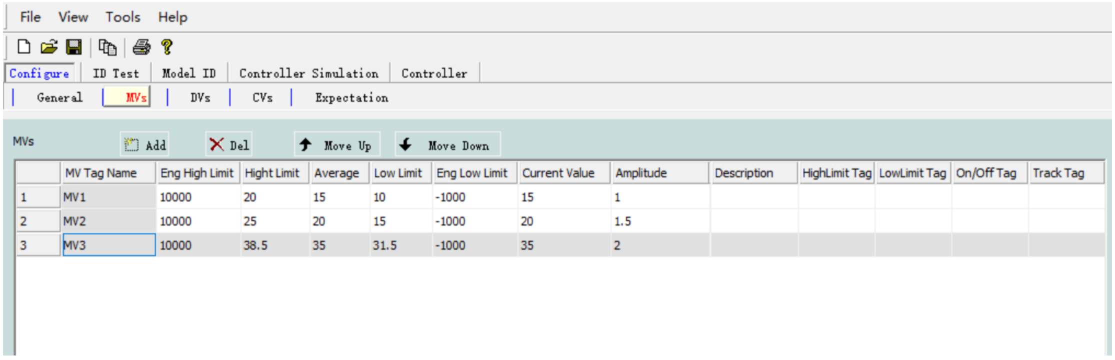
In the **Watch dog** section, select "Use watch dog" and enter the value 180 to represent 180 sample points of the DCS system. Enter "MPCWATCHDOG" in the "Tag name" field as the Watch Dog Tag name for the DCS.

In the **Options** section, do not select "Automatically start identification" or "Automatically use model in controller" . Select "The plant is a simulation process" and enter the value 60 in the "Time Compression factor" field to indicate that 1 second in the simulation means 180 sample points in the DCS system. Enter "MPCWATCHDOG" in the "Tag name" field as the Watch Dog Tag name for the DCS.



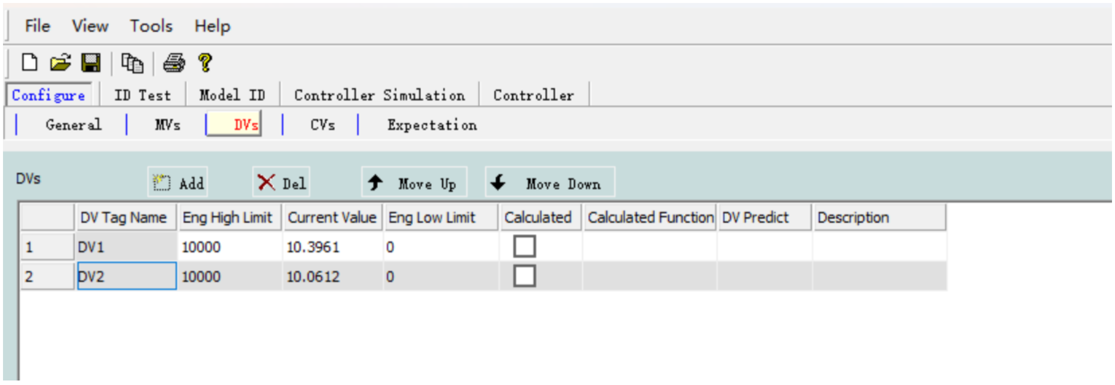
Configure MVs, DVs and CVs (Configuration Module)

Select Configure → Control Variables (MV), open the Control Variables (MV) window.

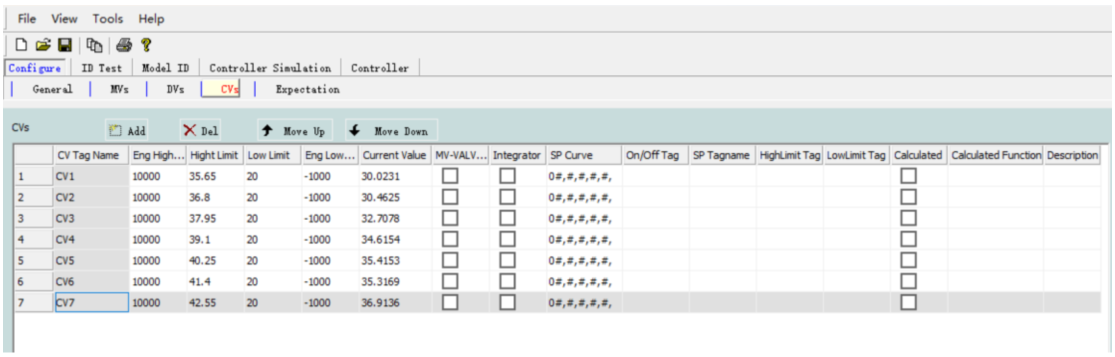


Click the **Add** button three times to add 3 control variables (MV), with control variable labels as MV1, MV2, and MV3 respectively. If the configuration in **General Configuration** is correct, the relevant values of the control variables will be automatically fetched from the OPC server when entering names in the control variable (MV) labels. If the control variable (MV) does not exist in the OPC server, the control variable (MV) label will be displayed in red, providing an OPC error alarm message.

Select **Configure →Disturbance Variables (DV)** , click **Add** button twice to add 2 DVs, their labels are DV1 and DV2.



Select **Configure → Controlled Variables (CV)** , click **Add** button seven times to add 7 CVs, their labels are CV1, CV2, …, CV7.



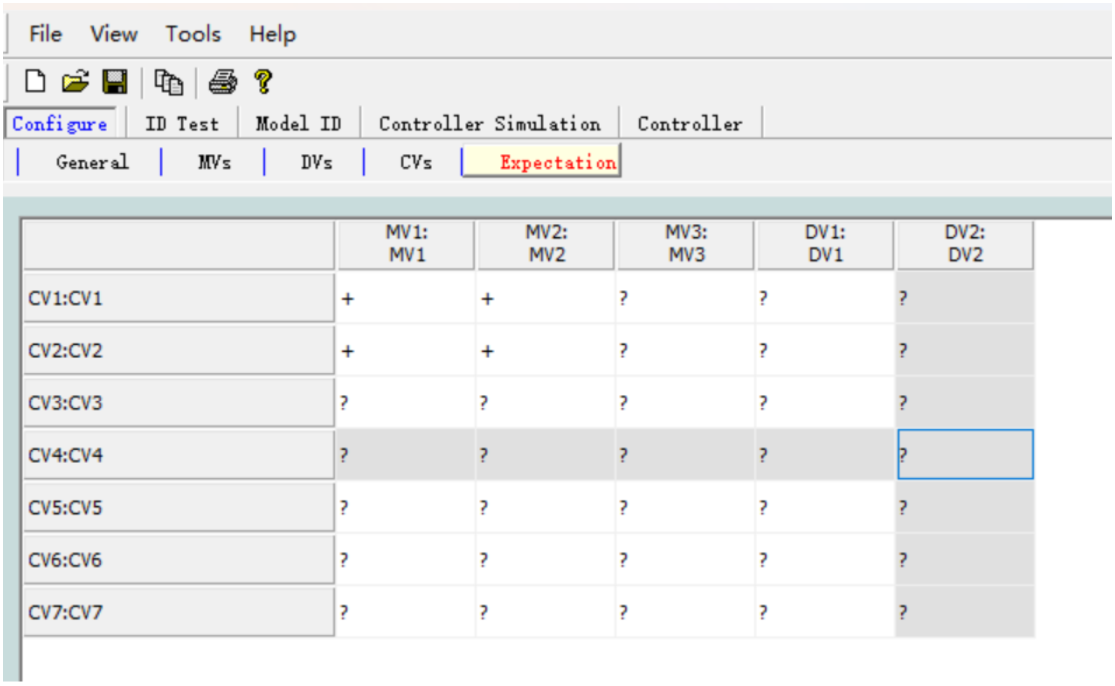
In the above configuration steps, users can specify the upper and lower limits of MV and CV, as well as the step size of MV in the identification experiment. Of course, these values can also be specified during the identification experiment (in the **Identification Experiment** module).

Important Note: All labels in Tai-Ji MPC must use uppercase letters, and the Tai-Ji MPC label names used in the OPC server must be named using uppercase letters.

Specify Desired Matrix (Configuration Module)

Operational experience and a priori knowledge of the production process typically include the relationships between MVs/DV and CV, which are reflected in a so-called 'desired matrix'. The rows and columns of the desired matrix correspond to CV and MV, respectively. The elements of the matrix have four values: '+', '-', '?', ' No '. '+': There is a model between the corresponding MV and CV with positive gain; '-': There is a model between the corresponding MV and CV with negative gain; '?': It is uncertain whether there is a model between the corresponding MV (DV) and CV; ' No ': There is no model between the corresponding MV (DV) and CV. In the example provided here, all elements of the desired matrix are '+', indicating that all models have positive gain.

The expectation matrix is used in model identification and model selection. When used in model identification, if the expectation matrix indicates that there is no model between certain MVs and CVs (i.e., the corresponding elements of the expectation matrix are 'No '), then that model is excluded from the identification process, significantly reducing computation time and increasing model accuracy. When used in model selection, if the model's quality grade is A, B, or C and the model gain's sign matches the expectation matrix, then that model is selected and sent to the control module.

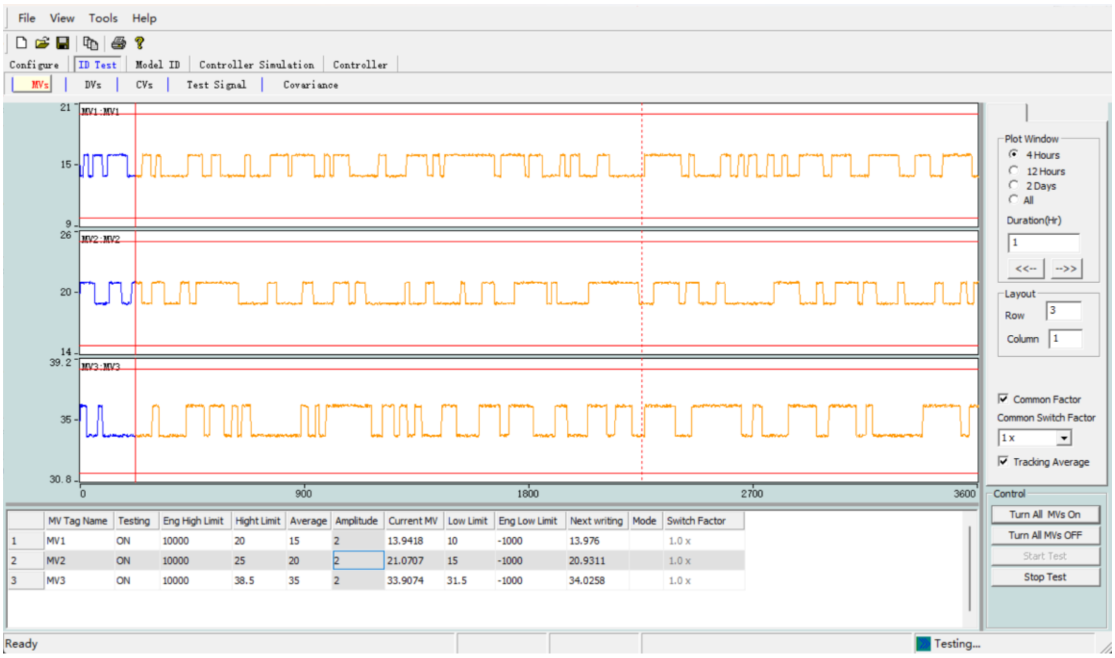


After the controller is configured, other modules of Tai-Ji MPC will automatically configure the MV, DV, and CV list windows and related trend chart windows. Click **Identification Experiment**, **Model Identification**, **Simulation Controller**, and **Online Controller** to see these configurations.

A lot of information has already been entered. It is recommended to save the project file (**File Save→**) after configuring all MVs, DVs, CVs, and the desired matrix.

4.2 Identification Experiment (Identification Experiment Module)

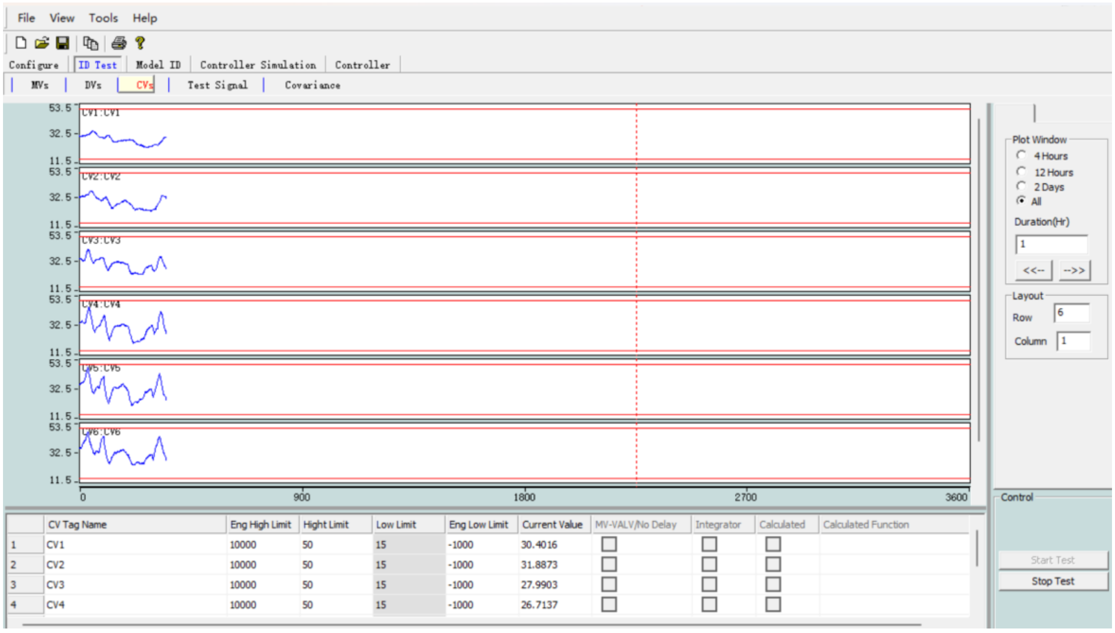
Click **ID Test→ Control Variables (MV)** to open the Control Variables (MV) window as shown below. Before starting the identification experiment, you need to specify the upper and lower limits of the MV, the engineering upper and lower limits of the MV, and the step size of the MV experiment signal. Please use the values shown in the MV window below. The step size of the experiment signal should be chosen appropriately, so that the CV can respond without disrupting production operations. A suitable choice is to use the step size that operators use in daily production processes.



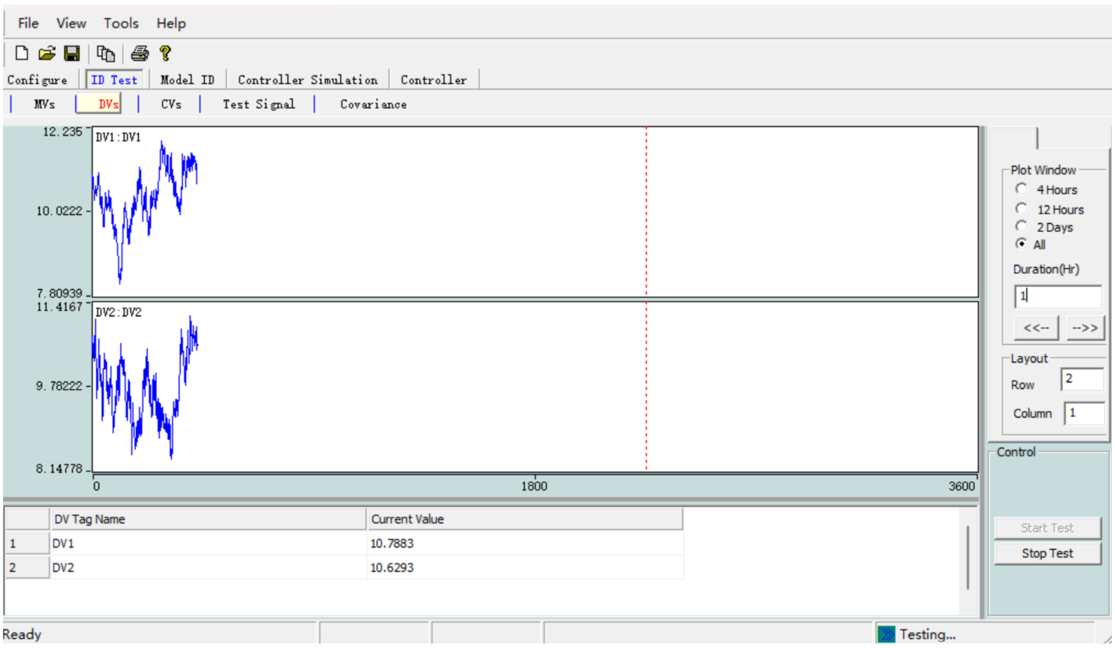
Click **Identify Experiment → Controlled Variable (CV)** to open the Controlled Variable (CV) window as shown below. The user needs to specify the upper and lower limits of the CV, as well as the engineering upper and lower limits of the CV. The values can be entered as shown in the CV window below.

Now the user is ready to start the identification experiment. In the **ID Test →Manipulated Variable (MV)** window that opens, click the Start Test button on the right side of the MV table. This will start data collection but will not write the experiment signal into the production process. To write the experiment signal into the production process and truly start the identification experiment, the user needs to switch the Testing status of the corresponding MV to ON. It is recommended to switch the MV to ON one by one when conducting identification experiments on the actual production process. To simplify the ON /OFF operation, the Turn All On and Turn All Off buttons can be used.

When the identification experiment is in progress, the MV curve is displayed in the Control Variable (MV) window. The blue line represents past signals, the orange line represents future actions based on experimental signals, the vertical red line indicates the current time, and the vertical dashed line indicates the planned end time of the identification experiment. The actual experiment time may be slightly shorter or longer than planned.

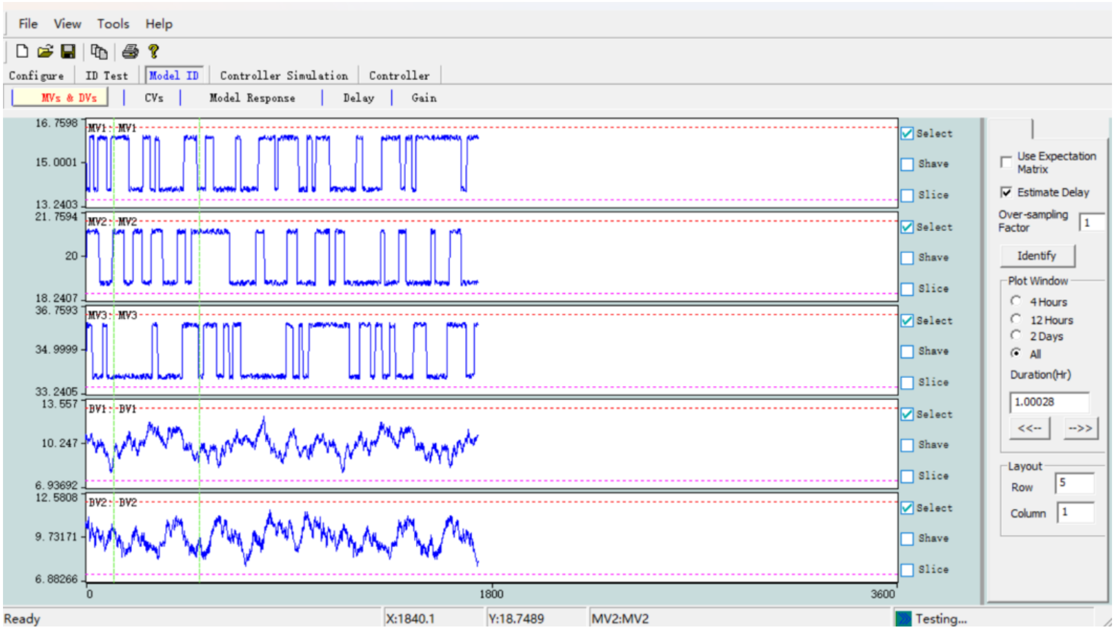


In the OPC simulation program, the two DVs shown in the figure below are generated by filtered white noise.

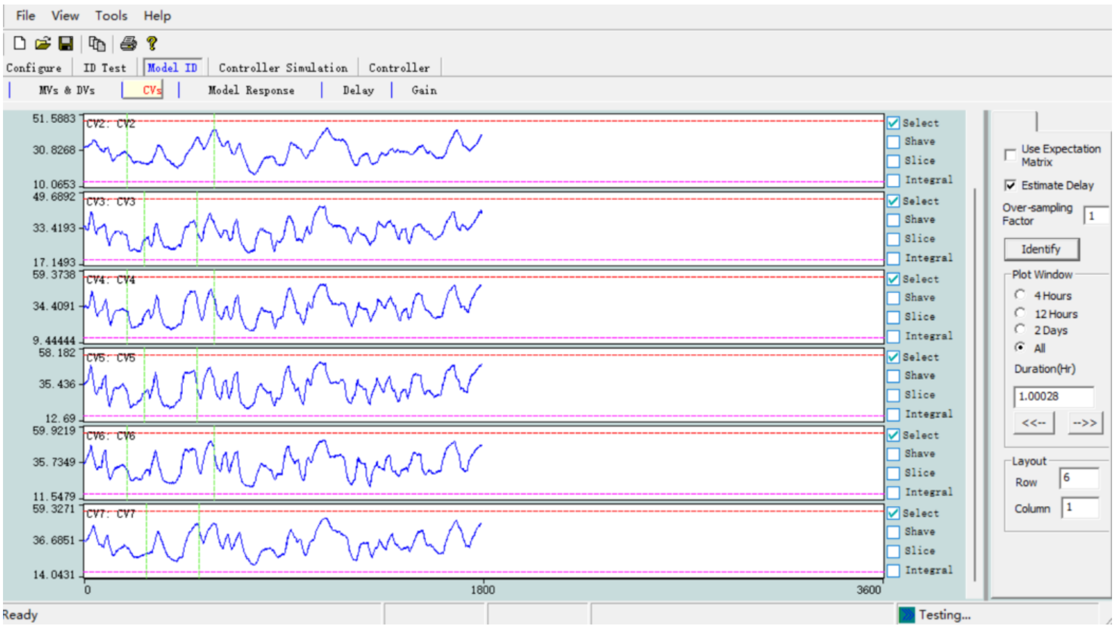


4.3 Model Identification (Model Identification Module)

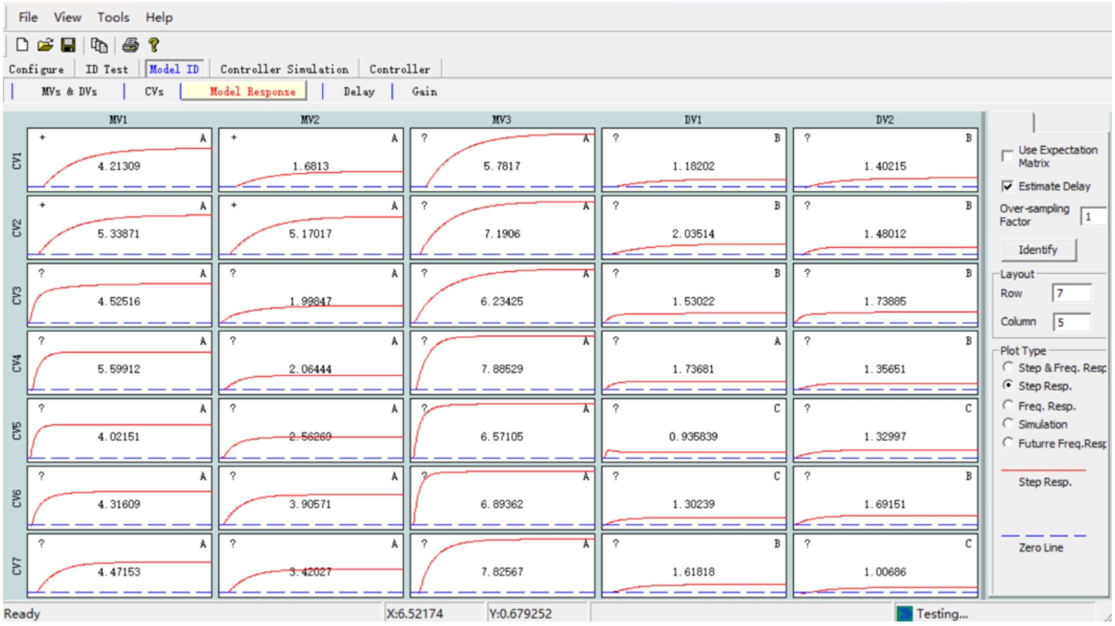
Click Model Identification → Control Variable (MV) & Controlled Variable (CV) to open the MV &DV window as shown below, displaying the MV/DV trend chart. In the model identification module, MV and DV are placed in the same window because DVs are considered to have the same role as MVs in model identification, as they are both inputs to the production process (CVs are the outputs of the production process).



Click **Model Identification → Controlled Variable (CV)** to open the window as shown below **Controlled Variable (CV)** window, displaying the CV trend chart.



Click **Model ID →Model** to open the window as shown below, which includes several windows such as the model step response window, the frequency response window (with error bounds), the model simulation window, and the future frequency response window. Step response and frequency response can also be displayed in the same window, and these windows can be selected for display through the buttons in the **Plot type** area.



Starting model identification is very simple. Click the **Identify** button to proceed automatically. If "Auto estimate delay" is selected, the user will then see a progress bar displaying "Estimate delay…" , followed by another progress bar displaying "Identify model…" . When the model identification calculation is complete, the identified model will be displayed.

Each step response or frequency response plot also displays additional information to assist in model validation and selection. The top left corner of the plot shows the corresponding value from the expected matrix (+, -, ? or No) , the center of the plot shows the model gain, and the top right corner shows the model quality grade A (Excellent), B (Good), C (Average), and D (Poor or No Model). Generally, if the model quality grade is A, B, or C and the sign of the model gain matches the expected matrix, then the model is selected and used in the controller.

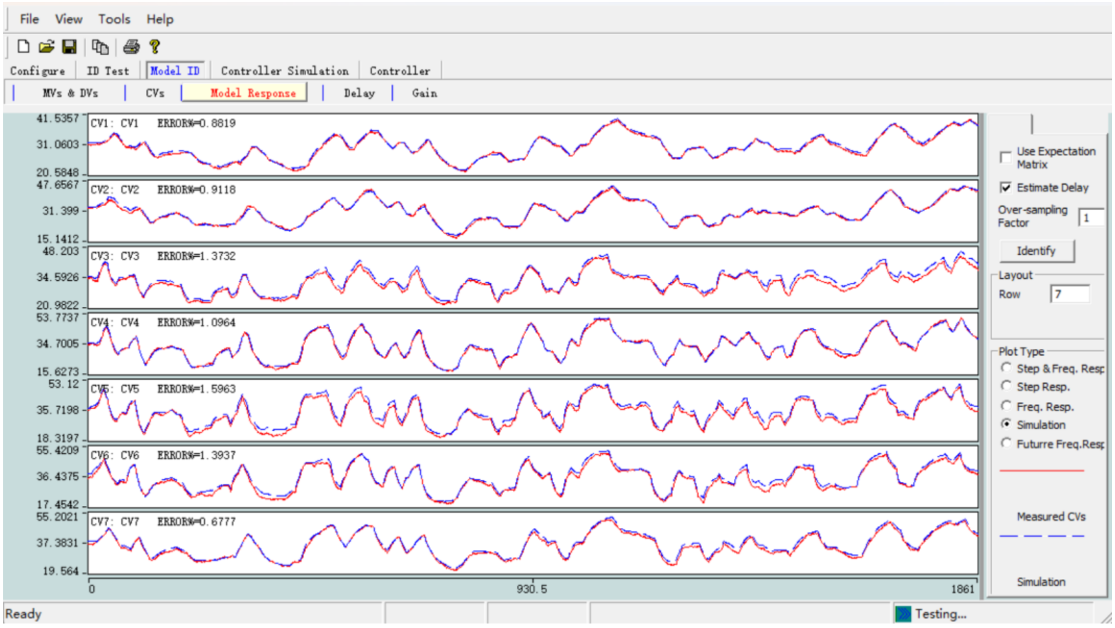
The method for selecting a model is as follows:

- To select a model for the controller, right-click on the model plot and choose "**Use in controller**". The model will then be loaded into the **Online Controller** module and the **Simulation Controller** module.
- If you want to select all suitable models for the controller, right-click on the model diagram and choose "Use all valid models in controller" , then all models with quality A, B, or C and whose gain sign matches the desired matrix will be loaded into the Online Controller module and the Simulation Controller module.
- If you want to select all models including those with quality D for the controller without considering the desired matrix, right-click on the model diagram and choose " Use all models in controller " , then all models will be loaded into the Online Controller module and the Simulation Controller module.

Based on the identification results, users can modify the expected matrix: right-click on the model diagram and then select "Change expect value", then you can modify the expected value of the corresponding model (+, -, ? or No) .

Right-click on the model diagram and then select “Zoom out”, then the window will display the response curves of the entire model.

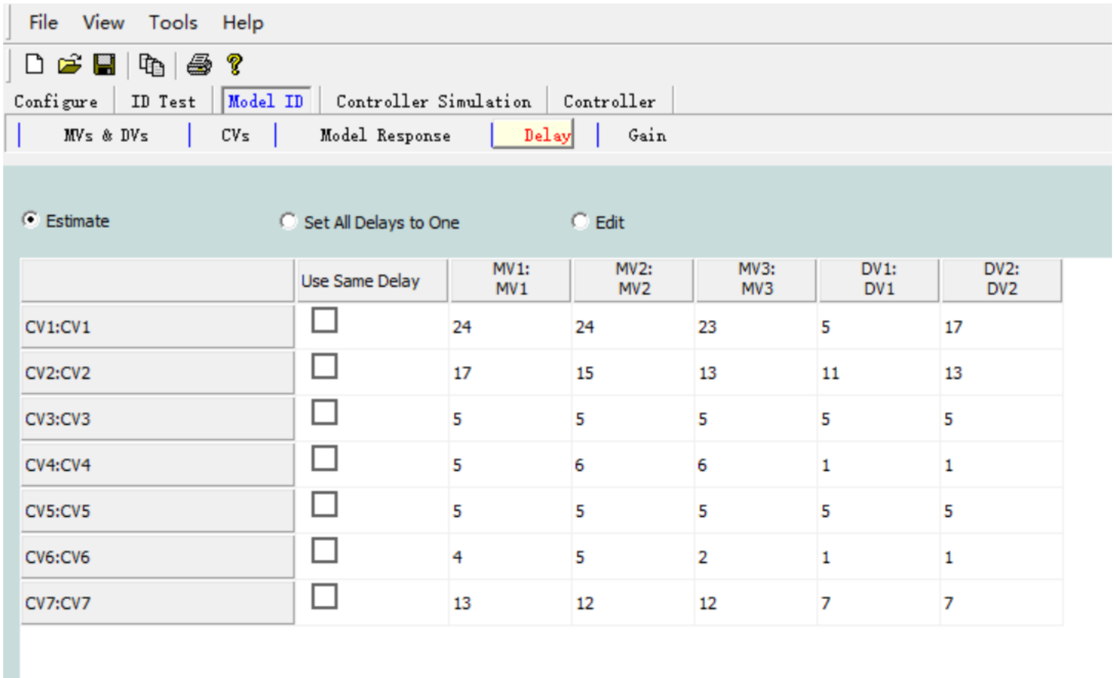
Clicking the **Simulation** button will display the measured and estimated values of the CV, as shown in the figure below. The ERROR% of the CV is the ratio of the standard deviation of the estimation error to the standard deviation of the CV. Empirical evidence suggests that good identification results correspond to an ERROR% typically between 1% and 40%.



Above the **Identify** button, there are two selection boxes related to model identification, as detailed below:

- When **Use expectation matrix** is selected, model identification will exclude empty models corresponding to the No elements in the expectation matrix. This reduces computation time and improves model accuracy. When **Use expectation matrix** is not selected, all models will be identified.
- When **Auto estimate delay** is selected, model identification will automatically identify and use the model's delay. When not selected, model identification will use the delay in the **Delay Window**.

Clicking **Model Identification→ Delay** will open the Delay window as shown below. The delay can be automatically estimated or provided by the user, and the user can also edit the delay matrix. The default delay for most CVs is 1, while the delay for CVs marked as “MV-VALVE/No delay” in the Configure Controlled Variables (CV) window is 0. →



	Use Same Delay	MV1: MV1	MV2: MV2	MV3: MV3	DV1: DV1	DV2: DV2
CV1:CV1	☐	24	24	23	5	17
CV2:CV2	☐	17	15	13	11	13
CV3:CV3	☐	5	5	5	5	5
CV4:CV4	☐	5	6	6	1	1
CV5:CV5	☐	5	5	5	5	5
CV6:CV6	☐	4	5	2	1	1
CV7:CV7	☐	13	12	12	7	7

Click **Model Identification → Gain** to open the **Gain** window (not shown here). These model gains cannot be edited in the **Model Identification** module but can be modified in the **Online Controller** module and the **Simulation Controller** module.

If you select **Automatically start identification** in the **General Configuration** window and enter 120 minutes in the Model ID interval, then model identification will automatically start every 120 minutes based on the latest experimental data during the identification experiment. Note: Model identification can only start when the number of data sample points collected in the identification experiment exceeds 250. Additionally, when using a simulated production process, if the time compression ratio is 60, model identification will occur every 120 seconds (2 minutes).

If **Automatically use model in controller** is selected in the **General Configuration window,** Tai-Ji MPC will automatically load the appropriate model into the **Online Controller** module and the **Simulation Controller** module. The appropriate model refers to a model with a quality grade of A, B, or C, and the model gain's sign matches the expected matrix.

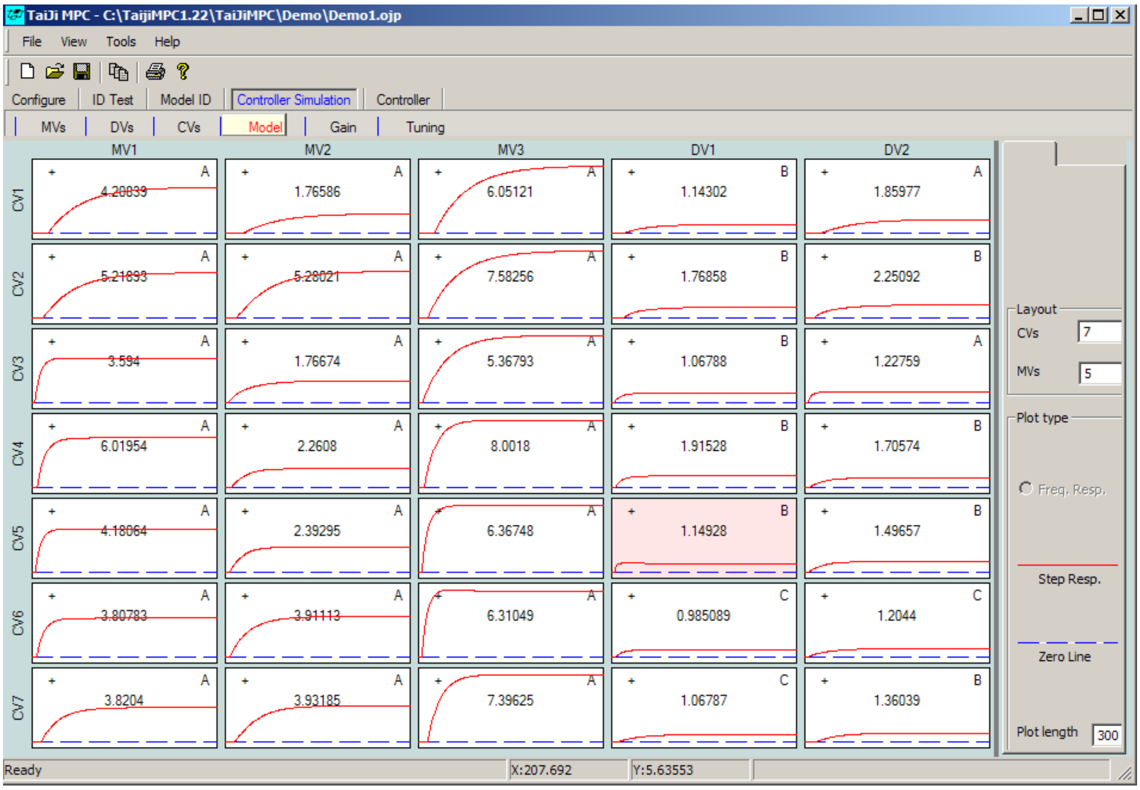
If you are a new user of Tai-Ji MPC, it is recommended to manually start model identification and manually load the model into the controller; if you are familiar with Tai-Ji MPC, you can automate these steps.

4.4 Simulation Controller

When some or all of the identified models are of good quality (i.e., the model's quality grade is A, B, or C, and the model gain's sign matches the expected matrix), the user can use some or all of the models for the control process simulation. Note: Simulation can begin simultaneously with the identification experiment.

Click **Simulation Controller Model→** , the identified model will be displayed as follows. The step response of the model here, the corresponding values in the desired matrix, the gain of the model, and the quality grade of the model will be displayed simultaneously. Right-clicking on the model diagram will present a menu bar, allowing several operations such as copying/pasting the model, modifying the model, manually adding the model, etc.

Note: **Online Controller** and **Simulation Controller** use exactly the same model.

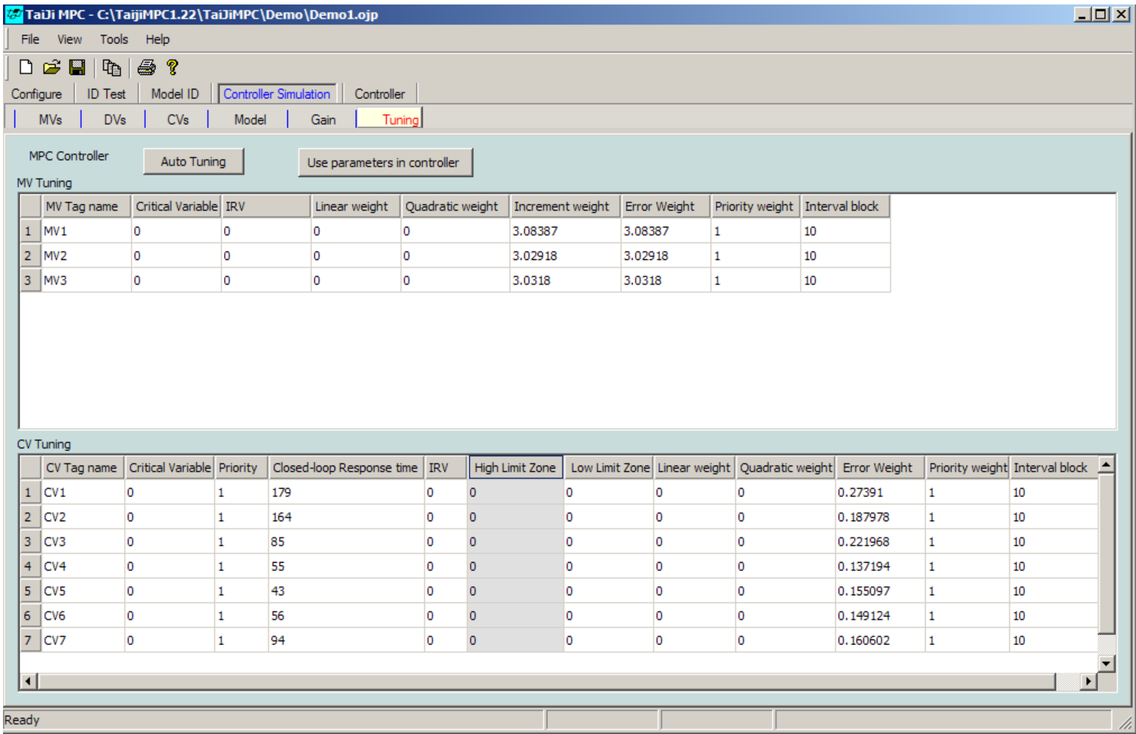


We temporarily ignore economic optimization and only consider dynamic control issues. Now the user can adjust the control parameters, click **Simulation Controller→ Tuning** to open the **Tuning** window as shown below. Section 1.2.2 introduced how to automatically tune all dynamic control parameters. Clicking the **Auto Tuning** button will automatically tune all dynamic control parameters based on MV/CV data and the identified model. In Tai-Ji MPC, the performance of dynamic control depends on the following tuning parameters:

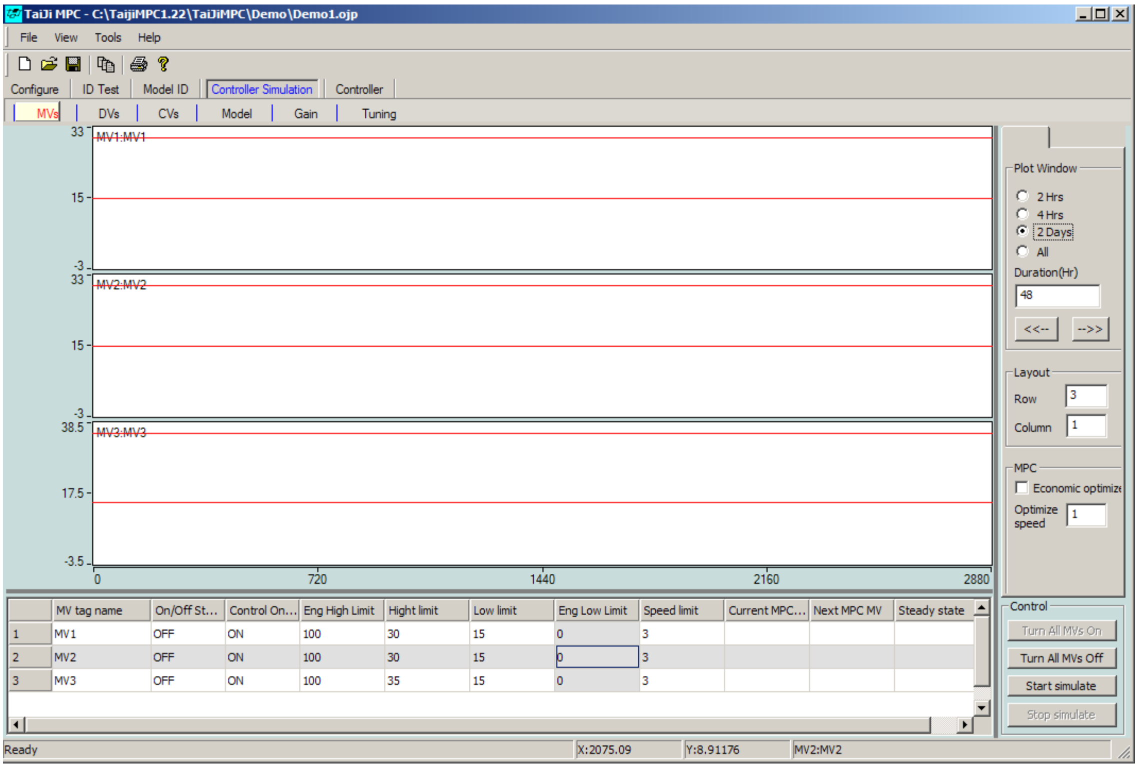
- **CV Closed-loop Response Time:** Tai-Ji MPC will assign a first-order model response reference trajectory for each CV. The control algorithm will make the CV follow this reference trajectory. Reducing the CV closed-loop response time will increase the control speed, while increasing it will slow down the control speed but improve robustness to model errors.

- **CV Error Weight:** Increasing the CV error weight will cause the control algorithm to apply more control actions to that CV, reducing the controller's robustness. Decreasing the CV error weight will make the control of that CV more lenient but will improve the controller's robustness.
- **MV Increment Weight:** Increasing the MV increment weight will slow down the control action of the algorithm on this MV, enhancing the robustness of the controller; decreasing the MV increment weight will speed up the control of this MV but reduce the robustness of the controller.

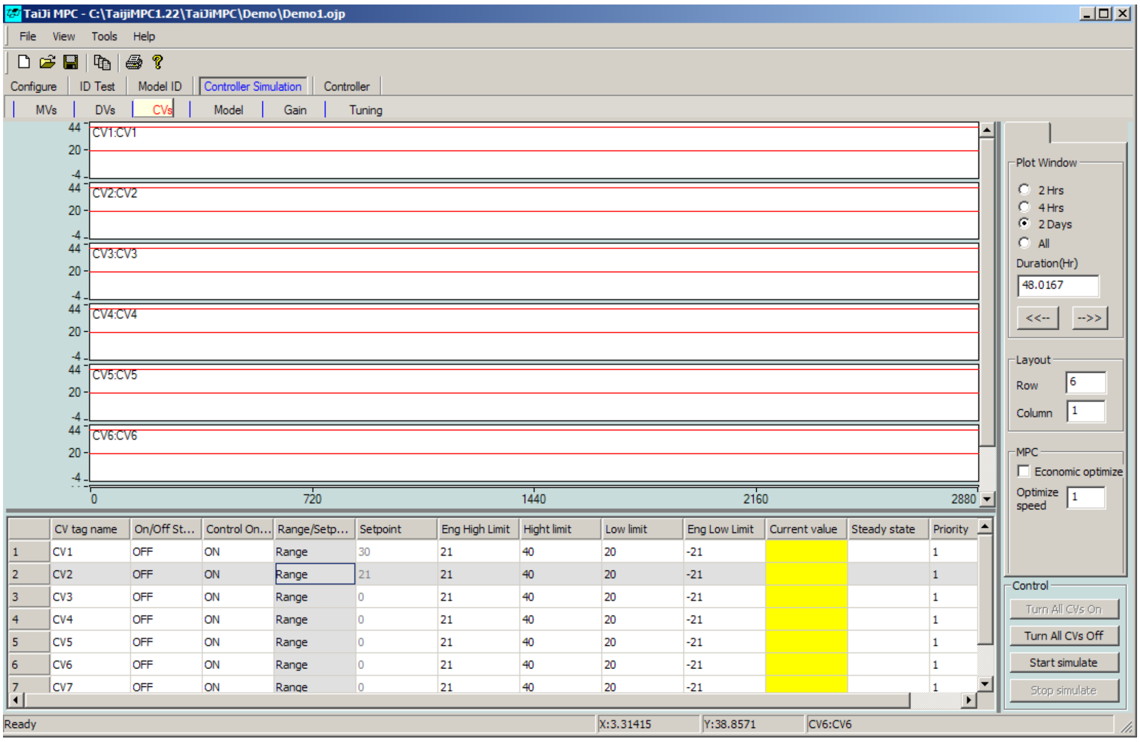
The tuning parameters in the Simulation Controller and the Online Controller are usually different. When good controller parameters are obtained in the simulation, the user can click the Use parameters in controller button to transfer the tuning parameters to the Online Controller .



→ Click Simulation Controller Control Variable (MV) to open the Control Variable (MV) window as shown below. Since we are temporarily only studying dynamic control, do not select Economic optimize for now, and set the upper and lower limits and the rate of change limits for the MV according to the values in the figure. The initial value of the MV in the simulation is the average of the upper and lower limits of the MV.



→Click Simulation Controller Controlled Variable (CV) to open the window as shown below. Then set the upper and lower limits of the CV according to the values in the figure. The initial value of the CV in the simulation is the average of the upper and lower limits of the CV.

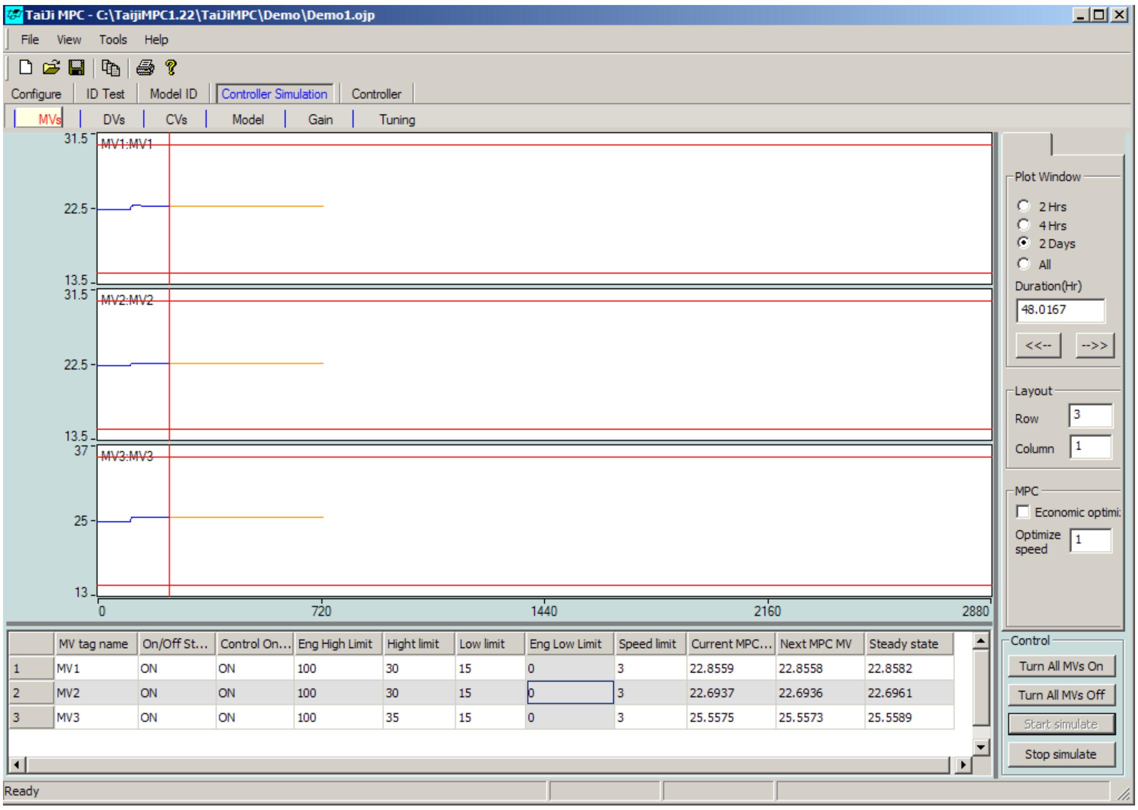


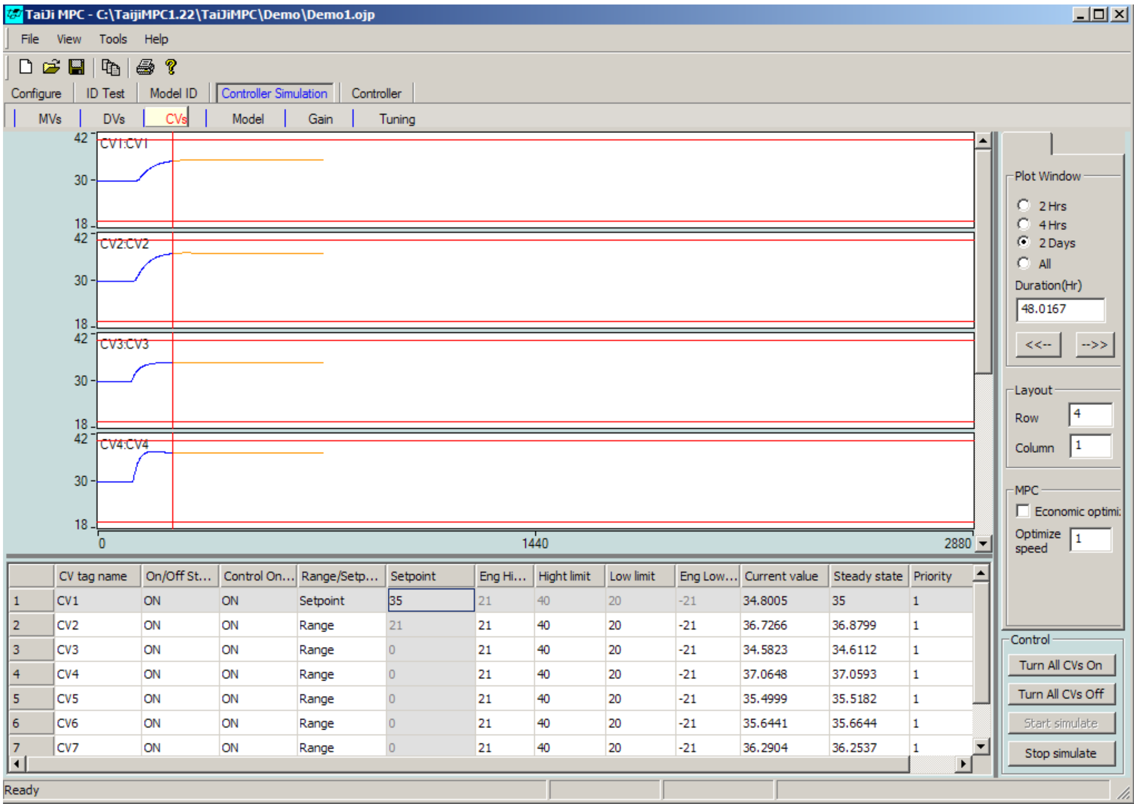
In the **Simulation Controller** module, DV remains zero and does not take effect in the simulation.

→ Now ready to start the simulation. In the Simulation Controller Manipulated Variable (MV) window that opens, click the Start Simulate button. A small window will appear where you can enter the simulation sampling time as 1 second. Click the Okay button to start the simulation. Click the Turn All MVs On button or use the Control On/Off button to turn each MV to the On state.

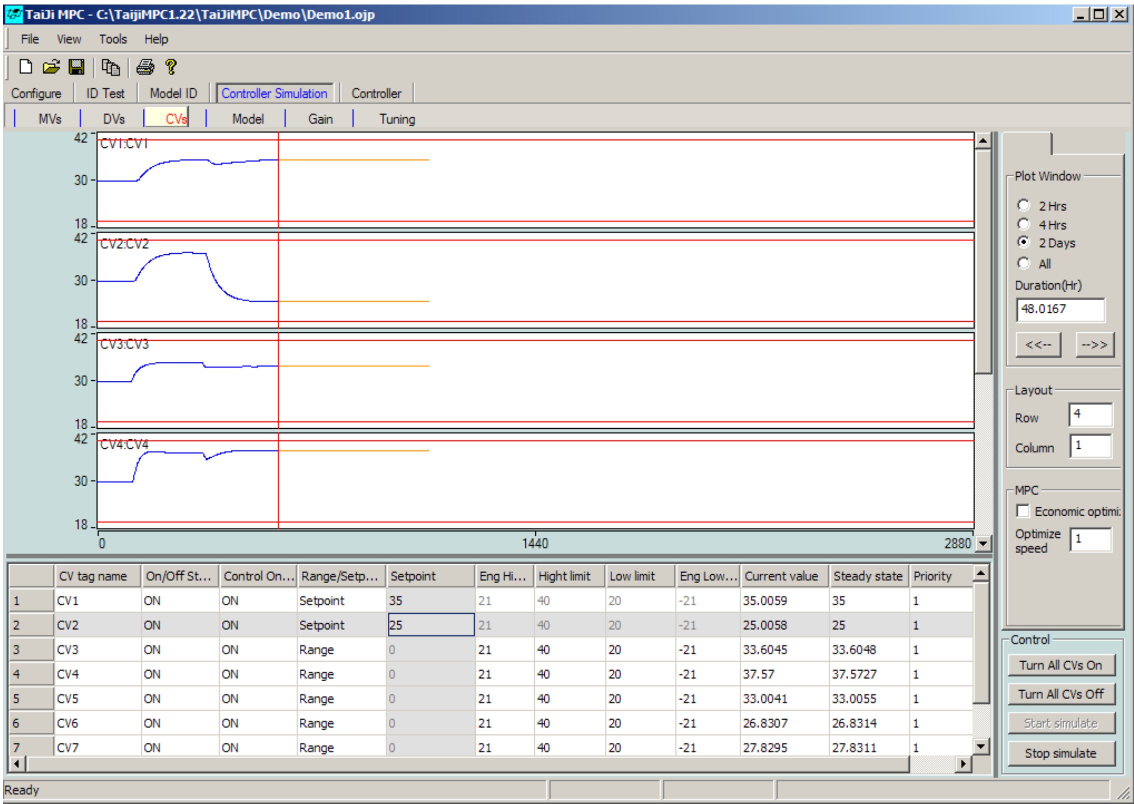
→In the simulation controller, click the Turn All CVs On button or use the Control On/Off button to turn each CV to the On state in the Controlled Variables (CV) window.

At the beginning of the simulation, there is no change because the CVs are in zone control mode, and currently, all CVs are between their upper and lower limits. To test the control performance, change the control mode of CV1 to Setpoint mode and set the setpoint to 35. After confirming this setpoint in the information window, the MV will start to act to try to control CV1 to 35; CV1 will smoothly change to 35 while other CVs will increase, but since they are in zone control mode and between their upper and lower limits, the controller will not adjust these CVs; as shown in the figure below.

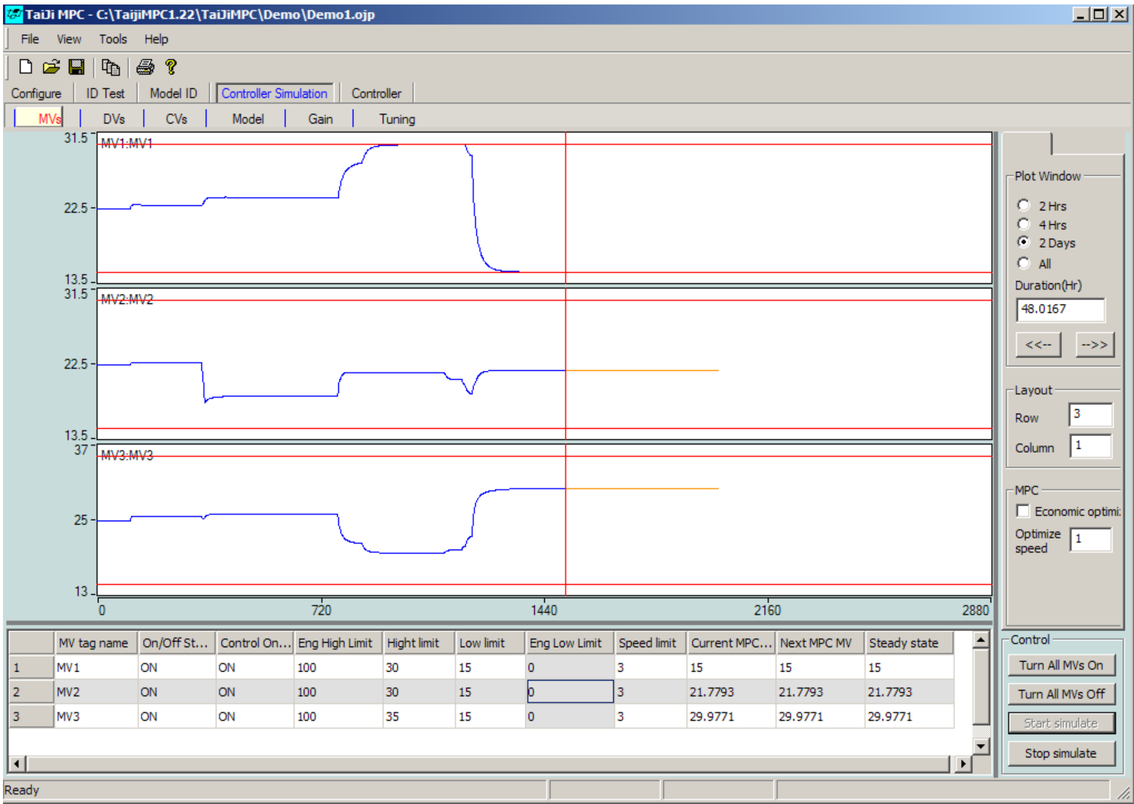


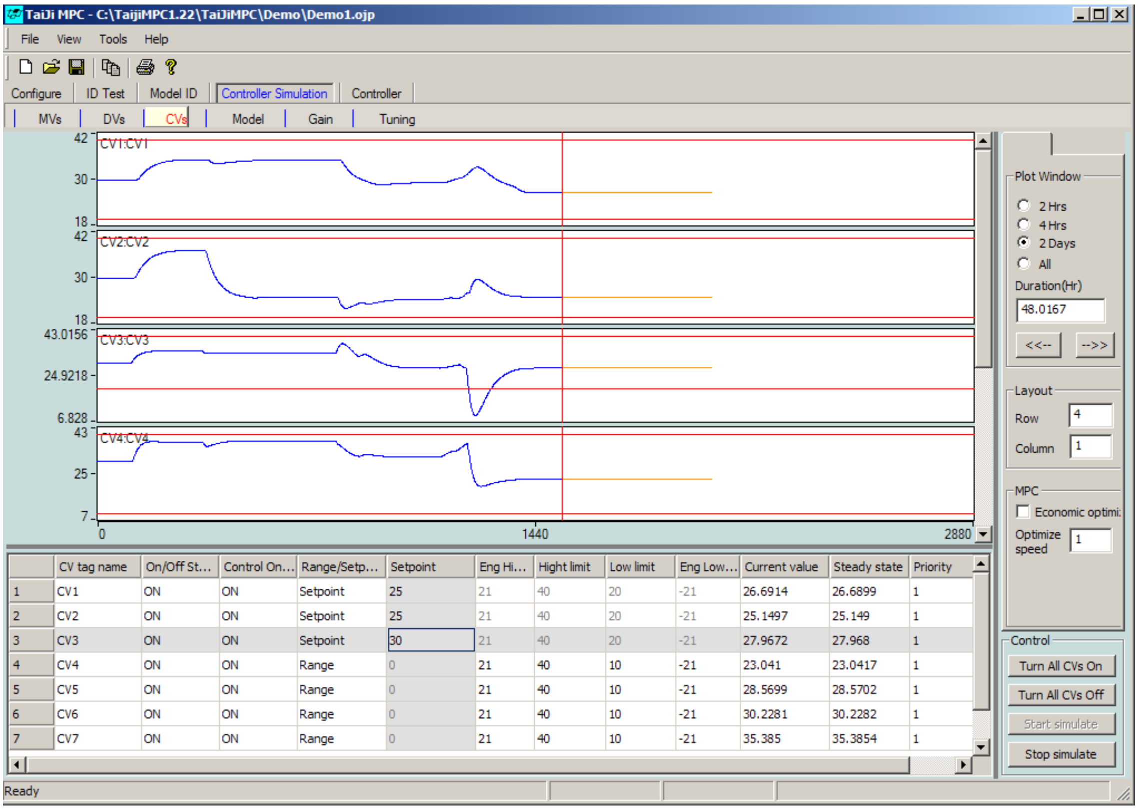


Change the control mode of CV2 to **Setpoint** mode and set the setpoint to 25. After confirming this setpoint in the information window, the MV will start to act in an attempt to control CV2 to 25. Note that CV1 is currently in setpoint control mode, so there will only be slight changes.



Change the control mode of CV3 to **Setpoint** mode and set the setpoint to 25. After waiting for a while, change the setpoint to 30. The user will find that the controller cannot control all CVs near their respective setpoints. Clicking the **Control Variable (MV)** window will show that MV1 first reaches the upper limit and then the lower limit; as shown in the figure below.

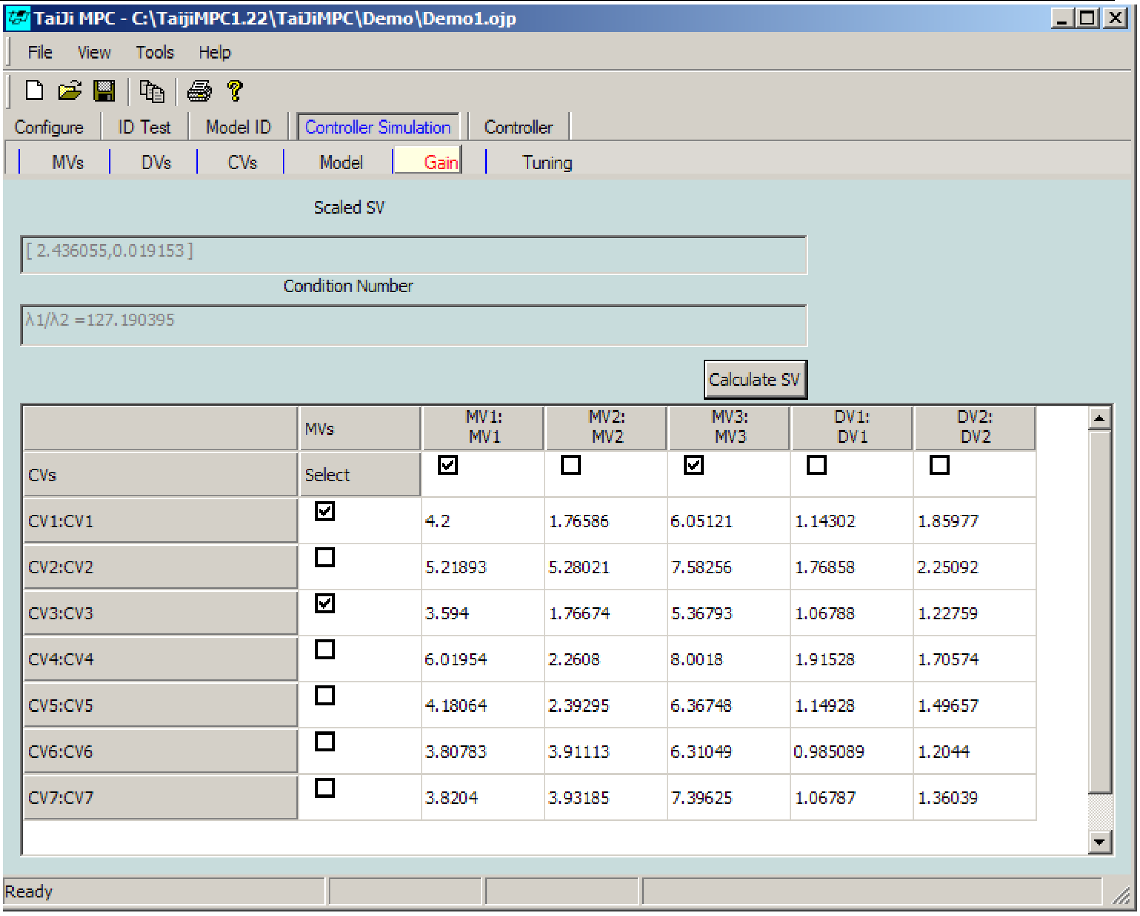




The user may have noticed that when CV3 is in setpoint control mode, the step change of MV is very large, meaning the controller requires significant action. The reason for this phenomenon is the strong interaction between CV1 and CV3 (also known as strong coupling, ill-conditioning, or linear dependency), making it difficult to control them independently. This controllability issue can be analyzed through the singular values of the model gain matrix. Click **Simulation Controller Gain →** to open the **Gain** window as shown below.

In the **Gain** window, select CV1, CV2, MV1 and MV2 , then click the **Calculate SV** button (SV stands for Singular Value). The singular values of the selected 2 x 2 gain matrix are [1.7975, 0.3577] , and the condition number (the ratio of the first singular value to the second singular value) is 5.025 . Generally, if the condition number is between 1 and 10, the corresponding 2 x 2 process is easy to control.

After selecting CV1, CV3, MV1 and MV2 , click Calculate SV , the condition number is 35.5; After selecting CV1, CV3, MV1 and MV3 , click Calculate SV , the condition number is 127.2 . These larger condition numbers indicate that it is difficult to control CV1 and CV3 individually.



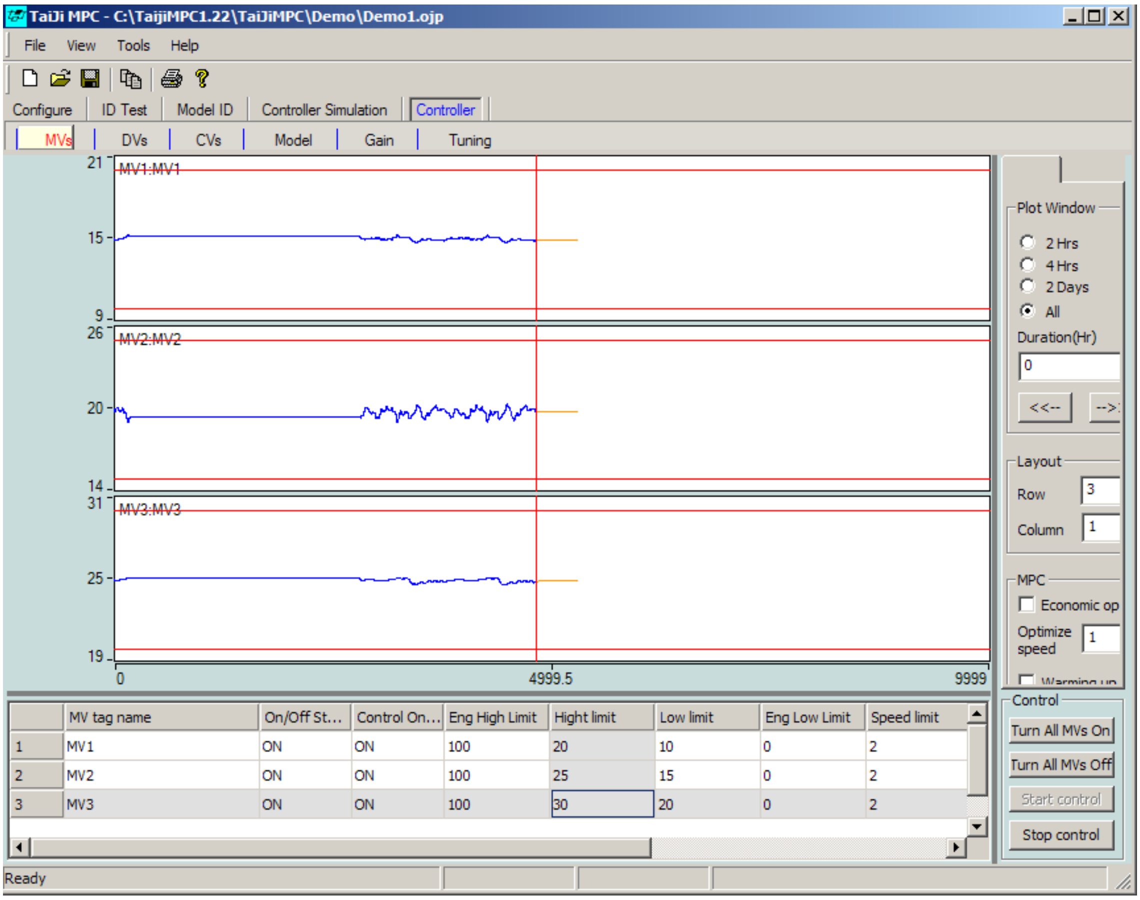
When the interaction between two CVs is strong, they should not be controlled independently. When setpoint control or small region control is achieved for one of the CVs, the control requirement for the other CV needs to be reduced, i.e., adopting a larger region control. When both CVs have high control requirements, it will result in very large MV actions, which may interfere with other CVs and potentially cause instability due to model errors.

4.5 Running the Online Controller

When the identified partial or full model quality is good (i.e., the model quality grade is A, B, or C, and the model gain sign matches the expected matrix), and the control process simulation using the partial or full model has achieved good control effects, the user can start real-time control in the **Online Controller** module. Note: Real-time control can be started simultaneously with the identification experiment.

At this point, we will stop the identification experiment and run the real-time controller. Note that in this example, the production process is a simulated distillation column model. Please use the same method as in the identification experiment to run the OPC simulator TaiJiOPCSim .

➔Click Online Controller Control Variable (MV) to open the Control Variable (MV) window as shown below.



When running the real-time controller to control the actual process for the first time, it is recommended to follow these steps:

- Do not select the **Economic optimize** button, as it is necessary to first check the performance of dynamic control.
- Select **Warming up** , let the real-time control run for 50 sample points, and then deselect **Warming up** . This allows the predictor to initialize its parameters.
- Gradually bring MVs, CVs and DVs into control state.

The window and functions of the Online Controller module are basically the same as those of the Simulation Controller module, so they will not be explained one by one here.

The Simulation Controller module only considers ideal conditions, meaning there are no model errors and disturbances. In the Online Controller module, if it is a real process, there will be model errors, DV disturbances, and unmeasurable disturbances, so the response of the CV will be affected by noise and disturbances.

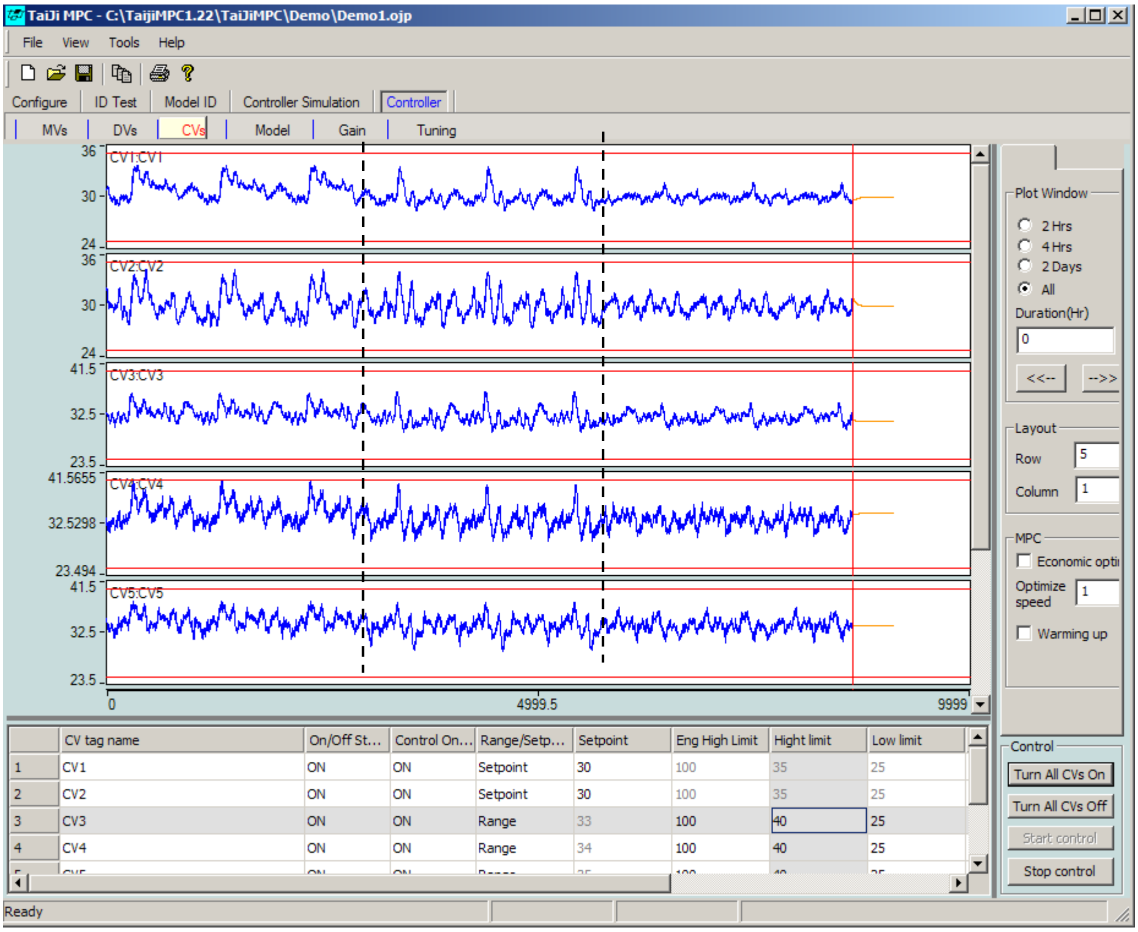
Now you can use the tuning parameters of the **Simulation Controller** module in the **Online Controller** module to run real-time control. We will first examine the control performance in noise suppression in the following three stages:

1) Click the **Start Control** button and keep all MVs, DVs and CVs in the OFF state, allowing the controller to run continuously for 2800 samples (two days) or a slightly shorter period (e.g., 1000 samples). During this time, the controller is in an open-loop state, and the collected data is uncontrolled.

2) In the second phase, put the MVs and CVs under control, keeping the DVs in the OFF state. Set CV1 and CV2 to setpoint control mode, and the other CVs to zone control mode. Allow the controller to run continuously for 2800 samples, and the user will observe that the variation range of CV1 and CV2 will slightly decrease.

3) In the third phase, introduce two DVs into control and let the controller run continuously for 2800 sample points. At this point, both feedback and feedforward control are present. Users will notice that the range of variation for CV1 and CV2 is significantly reduced, and since other CVs are related to CV1 and CV2, their range of variation will also decrease.

The trend chart of CV is shown below. We have conducted long-term tests to verify the control performance in terms of noise suppression. In practice, since open-loop control data can be obtained from the real-time database, the first phase is often not necessary, and the second phase does not need to be very long.



Next, we will test the impact of process controllability on control performance. This issue has been discussed in the controller simulation section, and here we will observe the impact of process controllability on disturbance rejection.

In the **Online Controller→** Gain **open the** Gain window, select CV1, CV2, MV1 and MV2, **click** Calculate SV, and the condition number obtained is 5.025, which indicates that CV1 and CV2 can be easily controlled through MV1 and MV2.

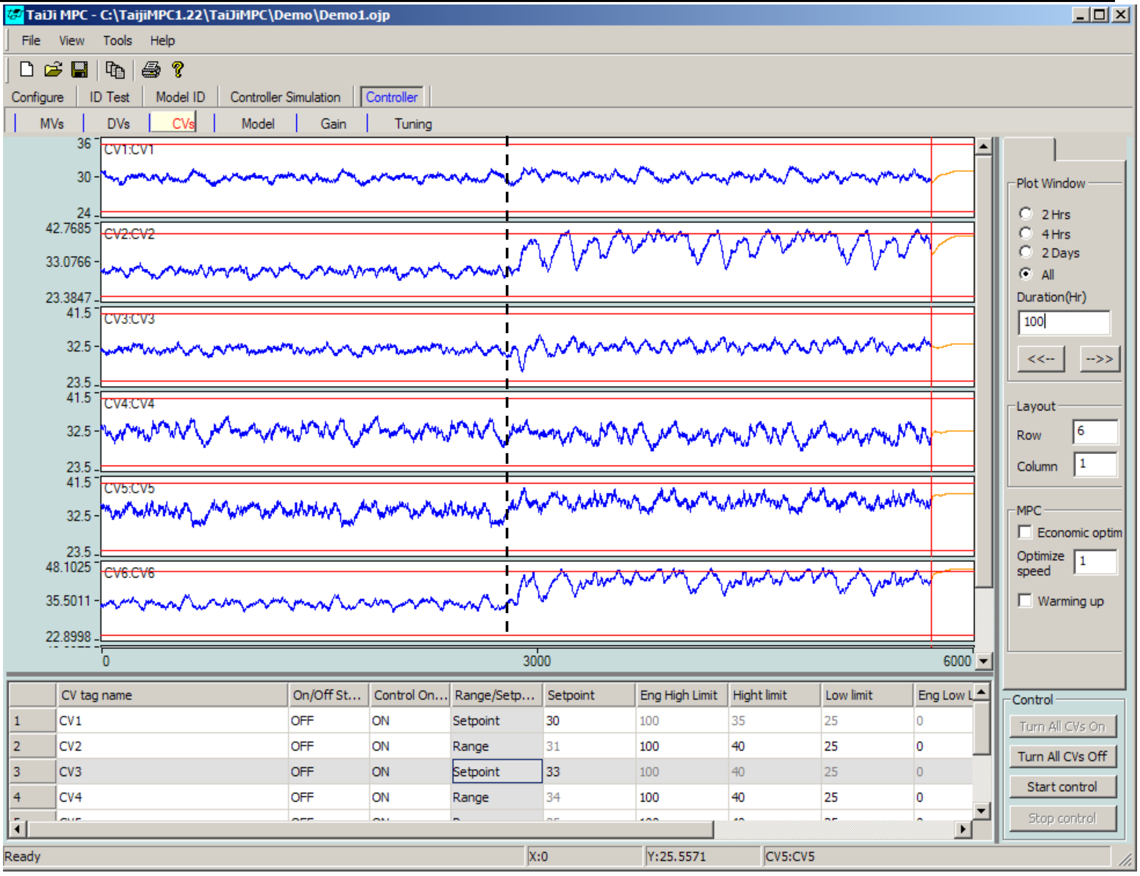
Select CV1, CV 3, MV1 and MV2, click Calculate SV, and the condition number obtained is 35.5, which indicates that it is relatively difficult to achieve independent control of CV1 and CV3 through MV1 and MV2.

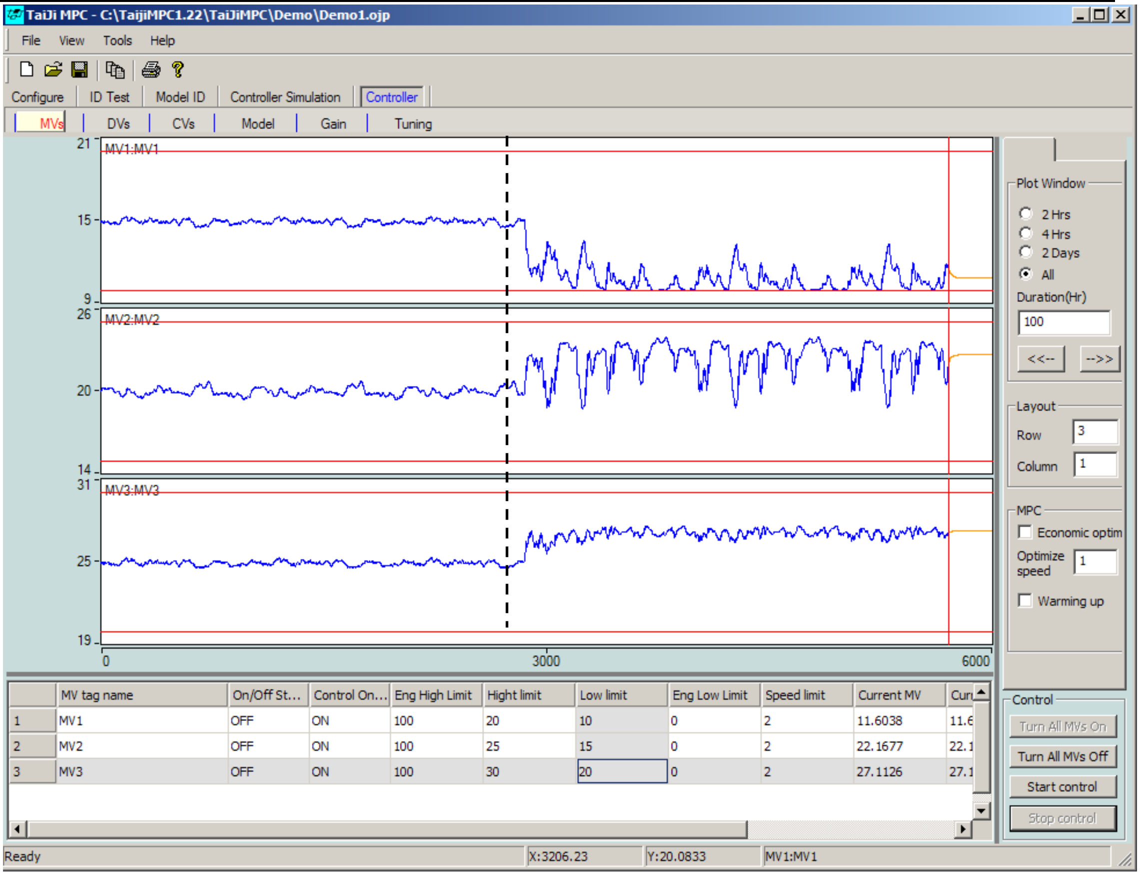
Select CV1, CV 3 , MV1 and MV 3 , click **Calculate SV** , and the condition number is **127.2** , which indicates that achieving independent control of CV1 and CV3 through MV1 and MV3 is very difficult.

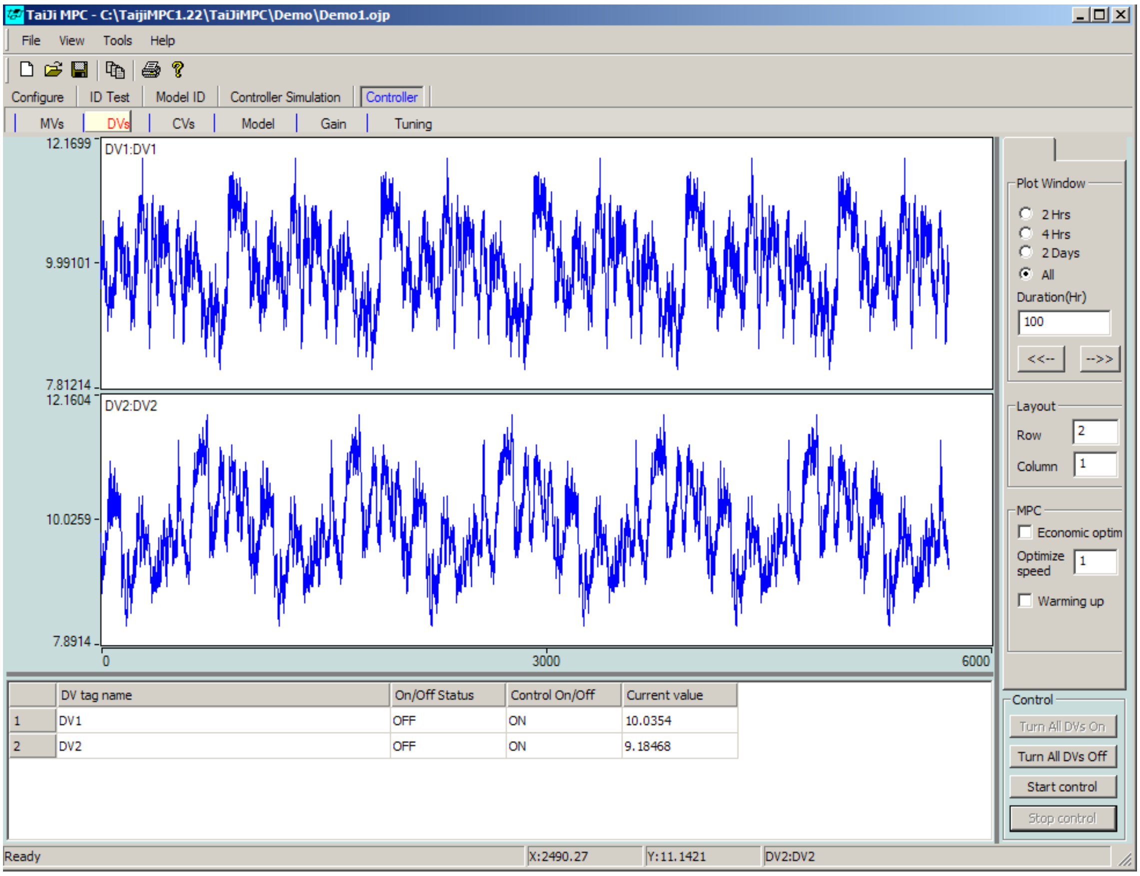
Now we will conduct the following test:

1) First, control CV1 and CV2 near the set values of 30 and 31 respectively, and apply zone control to other CVs. Click the **Start Control** button to put MV, DV and CV into control mode. Let the controller run continuously for 2800 samples, during which time the controller will perform both feedback and feedforward control, effectively overcoming disturbances.
2) Set all MVs, DVs, and CVs to ON state, and control CV1 and CV3 near the set values of 30 and 33 respectively, while applying range control to other CVs. Let the controller run continuously for 2800 sample points. During this period, fluctuations in all CVs will increase, especially CV2 and CV6. This is because the interaction between CV1 and CV3 is very strong, and controlling them individually at set values will cause significant MV actions, leading to severe fluctuations in other CVs.

The trend chart of CVs, MVs, and DVs is shown below:


TaiJi MPC - C:\TaijiMPC1.22\TaiJiMPC\Demo\Demo1.ojp
File View Tools Help
Configure
ID Test
Model ID
Controller Simulation
Controller
MVs
DVs
CVs
Model
Gain
Tuning
CV1:CV1
CV2:CV2
CV3:CV3
CV4:CV4
CV5:CV5
CV6:CV6
Plot Window
2 Hrs
4 Hrs
2 Days
All
Duration(Hr)
100
Layout
Row
Column
MPC
Economic optim
Optimize speed
Warming up
Control
Turn All CVs On
Turn All CVs Off
Start control
Stop control
CV tag name
On/Off St...
Control On...
Range/Setp...
Setpoint
Eng High Limit
Hight limit
Low limit
Eng Low L
Ready
X:0
Y:25.5571
CV5:CV5


TaiJi MPC - C:\TaijiMPC1.22\TaiJiMPC\Demo\Demo1.ojp
File View Tools Help
Configure ID Test Model ID Controller Simulation Controller
MVs DVs CVs Model Gain Tuning
MV1:MV1
MV2:MV2
MV3:MV3
0 3000 6000
Plot Window
2 Hrs
4 Hrs
2 Days
All
Duration(Hr)
100
Layout
Row 3
Column 1
MPC
Economic optim
Optimize speed 1
Warming up
Control
Turn All MVs On
Turn All MVs Off
Start control
Stop control
MV tag name On/Off St... Control On... Eng High Limit Hight limit Low limit Eng Low Limit Speed limit Current MV
1 MV1 OFF ON 100 20 10 0 2 11.6038
2 MV2 OFF ON 100 25 15 0 2 22.1677
3 MV3 OFF ON 100 30 20 0 2 27.1126
Ready X:3206.23 Y:20.0833 MV1:MV1


TaiJi MPC - C:\TaijiMPC1.22\TaiJiMPC\Demo\Demo1.ojp
File View Tools Help
Configure ID Test Model ID Controller Simulation Controller
MVs DVs CVs Model Gain Tuning
DV1:DV1
DV2:DV2
0 3000 6000
Plot Window
2 Hrs
4 Hrs
2 Days
All
Duration(Hr)
100
Layout
Row 2
Column 1
MPC
Economic optim
Optimize speed 1
Warming up
Control
Turn All DVs On
Turn All DVs Off
Start control
Stop control
DV tag name On/Off Status Control On/Off Current value
1 DV1 OFF ON 10.0354
2 DV2 OFF ON 9.18468
Ready X:2490.27 Y:11.1421 DV2:DV2

The above experiment indicates that, in general, we should not control two strongly interacting CVs individually at their respective set values or within small ranges. However, sometimes it is necessary to control two strongly interacting CVs individually, and we are willing to see significant MV actions. A typical example is a high-purity distillation column, where the two CVs are the purity of the top product and the purity of the bottom product, and the two MVs are the reflux rate and the steam rate. In such cases, users need to pay extra attention to identification experiments to obtain accurate models for controlling strongly interacting CVs.

4.6 Real-time Simulation Using Online Controller Module

The Online Controller module can, like the previous section, simulate MPC control based on any simulator of the production process in the OPC server. The advantage of simulating with the **Online Controller** module is that the simulation can be closer to real conditions, for example, the simulator can use some nonlinear elements, DV signals, and unmeasurable disturbances. Users can also check whether the OPC link in a certain OPC server is working properly.

We will briefly explain the steps to use the **Online Controller** module for simulation in a real MPC project, which is aimed at a real production process rather than a simulator.

Assume that the design of the MPC controller has been given and configured in Tai-Ji MPC, with the project file named Preject1.ojp. Assume the controller sampling time is 1 minute. Since this is a real production process and time compression cannot be used, please ensure that in the Configuration General opened General window, the Options area does not have The plant is a simulation process selected.→

Please follow these steps:

1) Start the identification experiment for the real production process and perform model identification to obtain the models between MVs/DVs and CVs; see Sections 4.2 and 4.3 for details.

2) After identifying the model, save the project as Preject1sim.ojp. This new project will be used in simulation, while the original project Preject1.ojp will be used in real-time control. Whenever the project file is saved, the model and data of the production process will also be saved in a simulation file with the suffix .sim.ojp , which can be used in the OPC simulator **TaiJiOPCSim** . If you need to use the identified model in **TaiJiOPCSim** , the delay between each CV and all MVs and DVs should be the same, but the delay corresponding to different CVs may be different. In the **Model Identification Delay→** window, select **Use Same Delay** to ensure that the identified model meets this requirement.

3) Run OPC Simulator **TaiJiOPCSim** , click the **Load model** button, and load the model file Preject1sim.sim.ojp in the **Open** window. At this point, **TaiJiOPCSim** will use the identified model and DV data to simulate the production process.

4) In the **TaiJiOPCSim** window, enter 2 at the **Period** field to represent a simulation sampling time of 2 seconds. Correspondingly, in the Tai-Ji MPC **configuration General→** window,

select **Options** area **The plant is a simulation process** and enter 30 in the **Time Compression Factor** field.

5) Simulate MPC control without considering DV signals. In the **TaiJiOPCSim** window, click the **StopDV** button (after clicking, this button will become the **StartDV** button), which will keep the DV signal unchanged. Click the **Start** button to start the simulation. Open the **Control** module of Tai-Ji MPC, deploy the controller, and verify the performance of setpoint tracking and constraint control, which is the same as in the **simulation controller** module.

6) Use DV signals in MPC-controlled simulations. In the **TaiJiOPCSim** window, click the **StartDV** button (after clicking, the button will change to **StopDV** ; if the button is already **StopDV** , do not click it). Now the simulation starts using the DV signals measured in the identification experiment. Click the **Start** button to start the simulation. Open the **Online Controller** module of Tai-Ji MPC, deploy the controller, and check the controller's performance in overcoming disturbances.

7) Use DV signals and unmeasurable disturbances in MPC-controlled simulations. Click the **StartDV** button (after clicking, the button will change to **StopDV** ; if the button is already **StopDV** , do not click it). Now the simulation starts using the DV signals measured in the identification experiment; additionally, unmeasurable disturbance signals can be superimposed on the CV in the simulation. The unmeasurable disturbance signals are stored in an Excel CSV file, where the 1st column corresponds to the disturbance signal for CV1, the 2nd column corresponds to the disturbance signal for CV2, and so on. Click the **Load Noise** button and load the Excel CSV file. Click the **Start** button to start the simulation. Open the **Control** module of Tai-Ji MPC, deploy the controller, and check the controller's performance in overcoming disturbances.

Congratulations! If you have successfully completed all the steps above and obtained the same or similar graphics as shown in this section, you have initially mastered how to use Tai-Ji MPC. Do you find using Tai-Ji MPC complex or simple? We welcome your feedback.

In the next chapter, we will describe the windows and functions of Tai-Ji MPC in more detail.

Start Project and perform model identification with local data

Test data may not be obtained from the identification experiment module but from other testing methods. In such cases, the data set can be loaded into Tai-Ji Online for model identification. Tai-Ji Online can import the following data types: 1) Matlab 6.5 MAT format files 2) Excel CSV format files.

4.7.1 Import Matlab 6.5 MAT File

The Matlab 6.5 MAT file to be imported should include the following variables:

```
MVdata : Data matrix including MVs symbols  (arranged by columns)
```

MVname : Character matrix containing MVs label names (arranged by rows)

CVdata : Data matrix containing CVs symbols (arranged by columns)

CVname : Character matrix containing CVs label names (arranged by rows)

Tident : Data sampling time

Tunit : Character matrix of time units, should be 'sec' (seconds) or 'min' (minutes).

Note Matlab's variable name is case-sensitive , users should confirm the names are consistent.

Example : a Distillation case . Assume the signal is loaded into Matlab workspace as a column vector.

MVs (inputs)	**Matlab variables**
Reflux	`Reflux`
Steam	`Steam`
Pressure	`Press`
Feed flow (DV)	`Feed`

CVs (outputs)	**Matlab variables**
Top composition	`Topcom`
Bottom composition	`Botcom`
Flooding	`Flood`

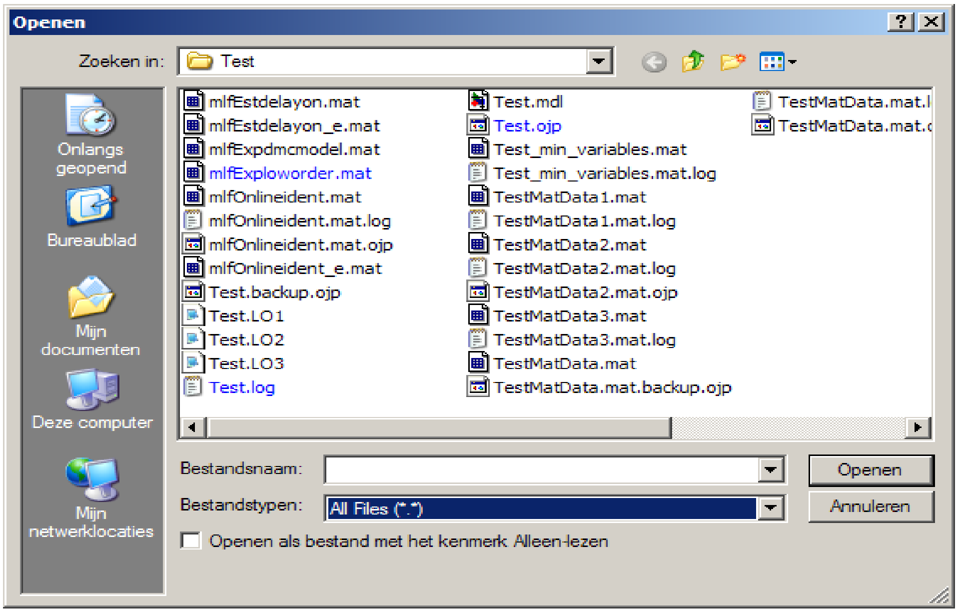
When preparing the .MAT file , you need to execute the following Matlab commands .

```
>> MVdata = [reflux steam press feed];
>> MVname = str2mat('reflux','steam','press','feed');
>> CVdata = [topcom botcom press];
>> CVname = str2mat('topcom','botcom','press');
>> Tident = 1;
>> Tunit = 'min';
>> MVsize = ones(size(MVdata));
>> save dist.mat MVdata MVname CVdata CVname Tident Tunit MVsize
```

If the Matlab version is higher than 6.5, you need to execute the following statements to create a Matlab 6.5 MAT file:

```
>> save dist.mat MVdata MVname CVdata CVname Tident Tunit MVsize
-v6
```

Load the MAT file, click **File Open,→ the following interface will be displayed:**



In this window, select the file type (File Type) as All Files (*.*) , find the MAT file , and click the Open button. The dataset will be imported into Tai-Ji Online , and you can see this data in the Identification Test Module and Model Identification Module.

The user can modify the parameters mentioned in Chapter 4.1 and execute the model identification module in Chapter 4.3 to obtain an appropriate model. The following parameters need to be set:

1） Sampling time of the data , used to generate a correct low-order model with delay

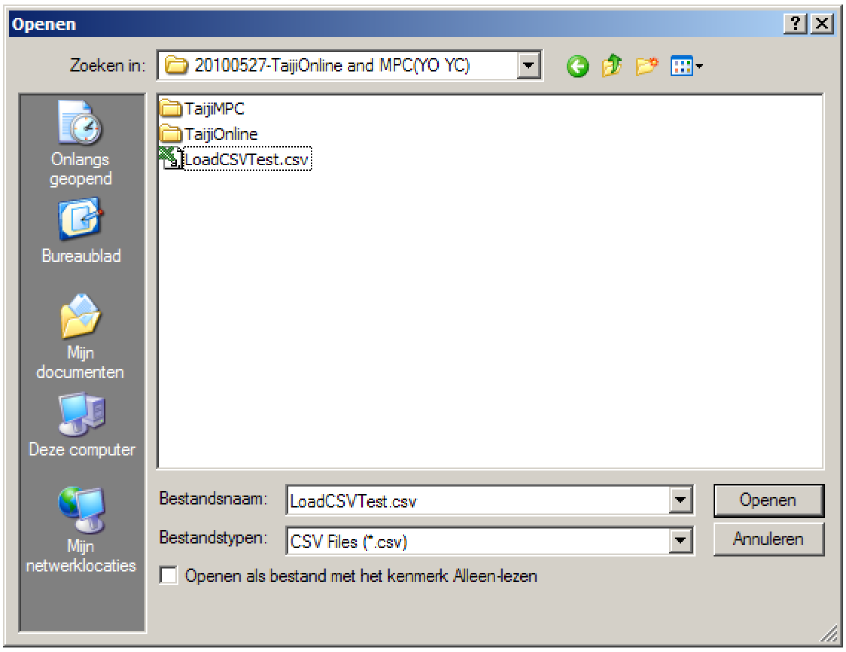
2） Time unit , same purpose as 1)

3） Time to steady state , used to generate an FIR model. Ensure this time is sufficiently long.

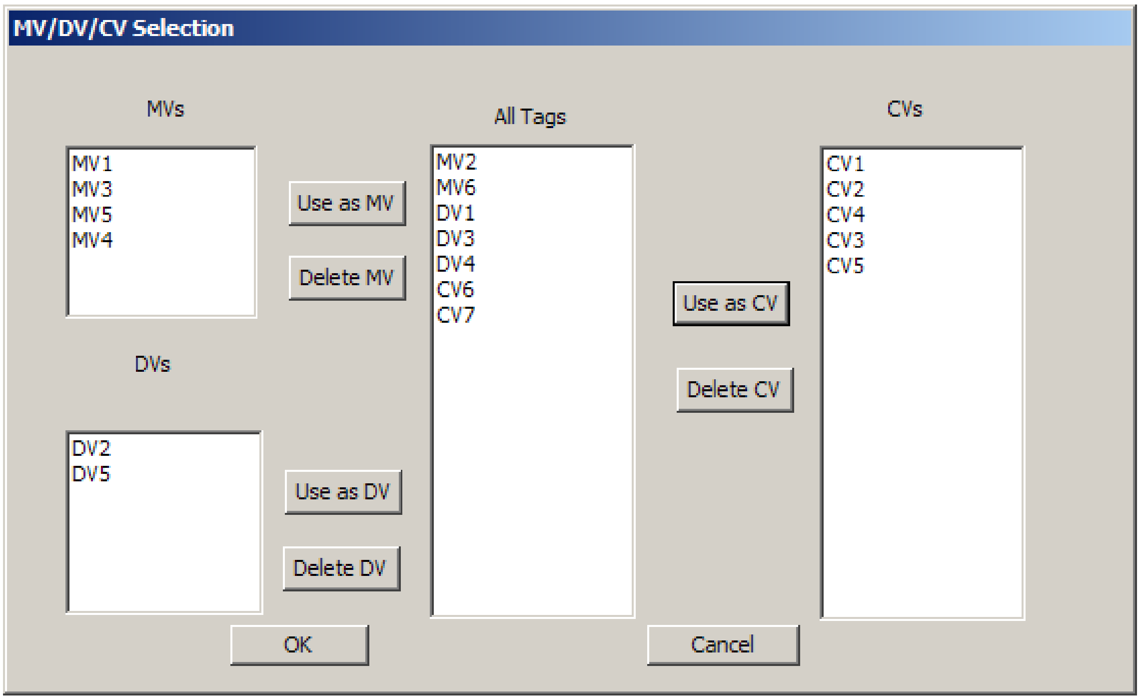
The settings of these parameters are in: General Configuration→

4.7.2 Import Excel CSV File

When the user uses an Excel CSV file to store test data , each column in the CSV file represents the signal of MV/CV/DV , and the first row includes the label names of MV/CV/DV . Clicking File Import CSV Data File will open the following window. →



Select and open the CSV file , the following window will be displayed.



The middle column identifies all the labels in the CSV file . The user can select a label and define it as MV, CV, or DV. After making the selection, click the OK button .

5 Tai-Ji MPC Window and Functions

Tai-Ji MPC includes the following 5 modules listed in the table below. All functions of Tai-Ji MPC are included in these modules, arranged from left to right in the order of executing an MPC project: 1) Project Configuration, 2) Identification Experiment, 3) Model Identification, 4) Simulation Controller, 5) Online Controller.

Configure	**ID Test**	**Model ID**	**Controller Simulation**	**Controller**

Each module has 5 to 6 windows. Since the design of the MPC controller is determined by MV, DV, CV, and their upper and lower limits, all modules contain MV windows, DV windows, and CV windows. Here, MV represents Manipulated Variables, DVs represent Disturbance (feedforward) Variables, and CV represents Controlled Variables.

The windows of each module are as follows:

Configuration Module

General	MVs	DVs	CVs	Expectation

Identification Experiment Module

MVs	DVs	CVs	Test signal	Covariance

Model Identification Module

MVs&DVs	CVs	Model Response	Delay	Gain

Controller Simulation Module

MVs	DVs	CVs	Model	Gain	Tuning

Controller Module

MVs	DVs	CVs	Model	Gain	Tuning

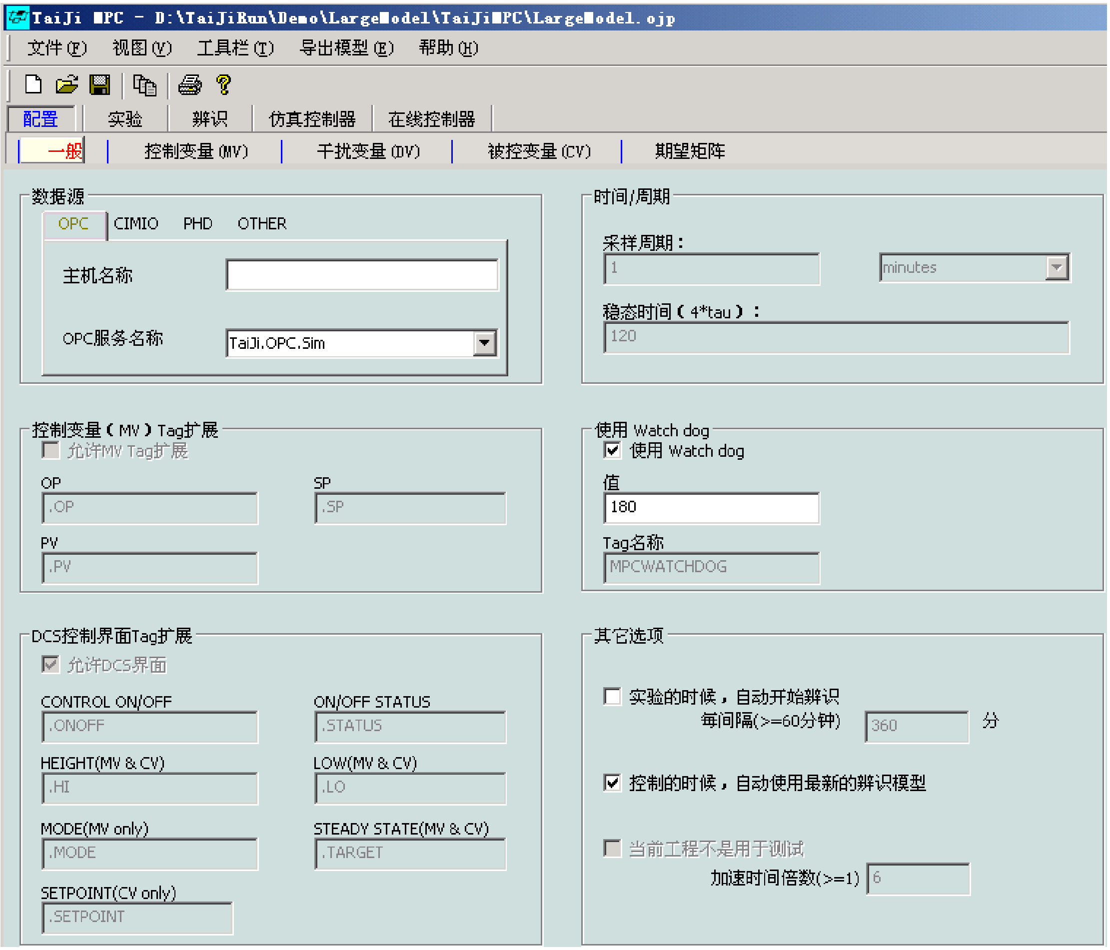
Each module will be introduced in detail in the following sections.

5. 1 Configuration Module

The Configuration Module is used to configure the MPC project and contains the following 5 windows:

5 .1.1 General Window

Click with the mouse **Configuration General→** to open the **General** window as shown below:



The General window includes 7 areas where users can configure general parameters such as the MPC controller's OPC server, controller sampling time, process steady-state time, and DCS/PLC related parameters.

Data Source Area:

OPC: Host Name: Machine IP address or name; for example: 192.168.1.1

OPC Service Name: OPC Server name; for example: TaiJi.OPC.Sim

CIMO: Host Name: Reserved

CIMO Service Name: Reserved

PHD: PHD Host Name: If not specified, the default is PHD_HOST

PHD Component Name: If not specified, the default is VisualPHD.data

OTHER: Host Name: Reserved

Server Name: Reserved

Note: When using the OPC method, if OPC enum is disabled, the 'OPC Server Name' cannot be enumerated. You can ask the network administrator to run dcomcnfg to configure OPC enum permissions, and then run this program again; or, you can directly fill in a valid OPC Server name in the 'OPC Name'. If you cannot access the OPC Server, please run dcomcnfg on both the client and server to confirm whether you have access permissions to the OPC Server.

Time/Period Area:

Sampling Period: Set the sampling period and control period for the experiment and MPC controller operation

Steady-State Time: The open-loop response time (estimated value) of the controlled system, which determines the length of the experiment time, **note that it is used here Time unit in seconds or minutes, not the number of samples**

Control Variable (MV) Extension Area:

Allow MVTag Extension: If selected, the values of OP, SP, and PV are read from the Server; otherwise, they are MV values

OP: OP Tag Suffix

SP: SP Tag Suffix

PV: PV Tag Suffix

Watchdog Area:

Use Watchdog: If selected, the Watchdog is used

Value: The value of the Watchdog

Tag Name: The Tag used by the Watchdog

Note: The Watchdog only takes effect during experiments or (and) when the MPC controller is working, and if 'Use Watchdog' is selected. In this case, the specified 'Value' will be written into the 'Tag name' of the Watchdog at each sampling period.

DCS Control Variable Tag Extension:

Allow DCS Interface: Only when selected, the control parameters of the controller will be read from the Server;

CONTROL ON/OFF: Allows the controller (using MV, DV, CV) Tag suffix

ON/OFF STATUS: Whether the controller is in use (MV, DV, CV) Tag suffix

HEIGHT (MV & CV): Upper limit Tag suffix for MV, CV

LOW (MV & CV): Lower limit Tag suffix for MV, CV

MODE (MV only): Mode Tag suffix for MV

STEADY STATE (MV & CV): Steady state value Tag suffix for MV, CV

SETPOINT (CV Only): Setpoint Tag suffix for CV

Other Options:

Automatically start identification during experiments: When selected, if in an experiment, identification will be initiated every x minutes.

During control, automatically use the latest identification model: If selected, after each identification, the model that meets the requirements (model quality A, B, C, and the gain symbol is the same as the expected matrix symbol) will be used as the MPC controller model. If the MPC is controlling and using this model, it will immediately switch to the latest model.

The current project is not for testing: If selected, it will be used as an actual industrial controller (without time doubling); otherwise, the sampling frequency will be accelerated under the condition that other parameters remain unchanged, which is convenient for system testing.

5 .1.2 Control Variable (MV) Window

➔Use the mouse to click Configure Control Variables (MV) to open the Control Variables (MV) window as shown below. This window contains a table of MVs.

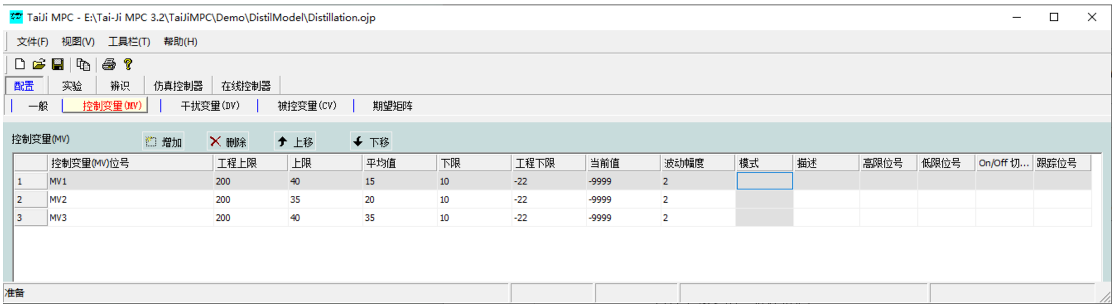


Table Operation Commands:

Add : Add an MV , a new row will be added at the end of the table

Delete : Delete the selected table row, and simultaneously delete the corresponding MV

Move Up : Move the selected MV one position forward

Move Down : Move the selected MV one position backward

Meaning of Table Columns:

Control Variable (MV) Tag : Tag of the control variable

Test Variable Tag : Tag of the test variable, which can be different from the MV tag

DCS Interface Tag Name: Tag name of the MV in the DCS interface

Engineering Upper Limit: Engineering upper limit of MV

Upper Limit : Upper limit of MV

Average Value : Initial value of MV during the experiment; if 'Track MV Average Value' is set, it tracks the change of MV average value. If the MV average value changes, the changed MV is set as the average value.

The change, if the MV average value changes, the changed MV is set as the average value.

Lower Limit : Lower limit of MV

Engineering Lower Limit: Engineering lower limit of MV

Current Value : Current value of MV

Fluctuation Range : Increment range of MV in identification experiment, the actual value of MV is the average value ± 0.5* fluctuation range

Description : Description of the control variable

High Limit Position Number : MV upper limit position number

Low Limit Position Number : MV lower limit position number

On n /Off Toggle Tag : MV Control Switch Tag

Tracking Tag : MV Tracking Tag, set for MPC and PID disturbance switching function

5 .1.3 Disturbance Variable (DV) Window

Click with the mouse **Configure Disturbance Variable (DV)→** , switch to the window shown below.

Table Operation Commands:

Add : Add a DV, add a row at the end of the table

Delete : Delete the selected table row, and delete the corresponding DV

Move Up : Move the selected DV up one position

Move Down : Move the selected DV down one position

Meaning of DV Table Columns:

Disturbance Variable (DV) Tag: Tag of the disturbance variable.

DCS interface tag name: DV tag name on the DCS interface

Current value: Current value of DV.

Description: Description of the disturbance variable DV

5 .1.4 Controlled Variable (CV) Window

Click with the mouse **Configure Controlled Variable (CV)→** to switch to the window shown below:

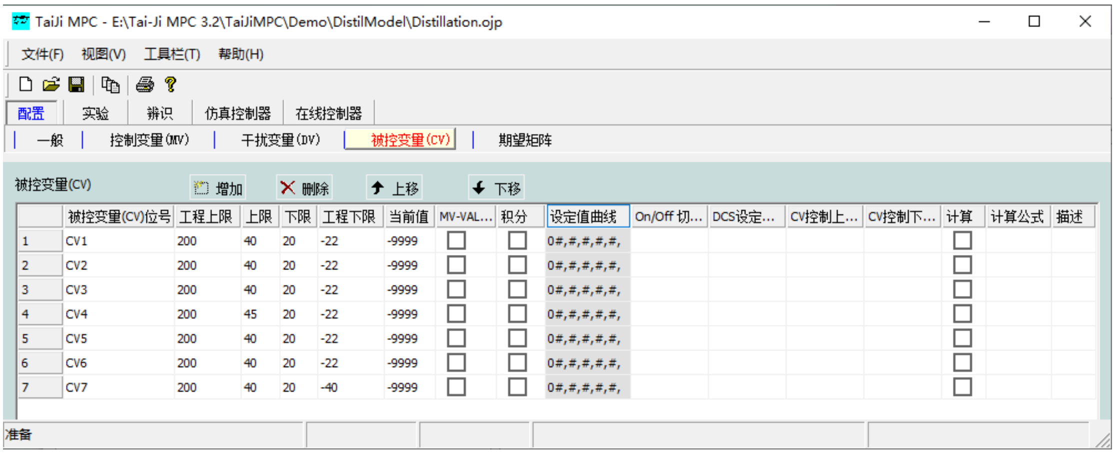


Table Operation Commands:

Add : Add a CV, adding a row at the end of the table.

Delete : Delete the selected table row, and simultaneously delete the corresponding CV.

Move Up : Move the selected CV up one position.

Move Down : Move the selected CV down one position.

Meaning of the CV **table columns:**

Controlled Variable (CV) Tag : Tag of the controlled variable.

DCS Interface Tag Name: CV Tag Name on DCS Interface

Engineering Upper Limit: CV Engineering Upper Limit

Upper Limit : Expected Upper Limit of CV.

Lower Limit : Expected Lower Limit of CV.

Engineering Lower Limit: CV Engineering Lower Limit

Current Value : Current Value of CV.

MV-VALV/No Delay : If selected, the model corresponding to CV has no delay.

Integral : If selected, CV is an integral variable.

Setpoint Curve: When using the setpoint curve function, it needs to be set. See Section 5.1.5.

On n /Off Toggle Tag : MV Control Switch Tag

DCS Setpoint Tag: If the setpoint of CV is provided by the DCS end, use this function.

CV Control Upper Limit Tag : CV Upper Limit Tag

CV Control Lower Limit Tag : CV Lower Limit Tag

Calculation : If selected, the value of the controlled variable will be calculated using the formula.

Calculation Formula : The calculation formula is only effective when the 'Calculation' column is selected. After selecting the 'Calculation' column, double-click the column with the mouse to pop up a dialog box that allows you to edit and verify the calculation formula.

Description : Description of the controlled variable (CV).

Note: The calculation formula is written using VB Script. Use the ReadValue function to get the Tag value and use SetCalculatedResult to pass the result to this program. For example: the following script calculates the sum of the square root of the tag “FIC-2001.PV” and the tag “FIC-2001.PV”. These two tags should be retrievable from the OPC server.

dbl1 = ReadValue ("FIC-2001.PV")

dbl2 = ReadValue ("FIC-2002.PV")

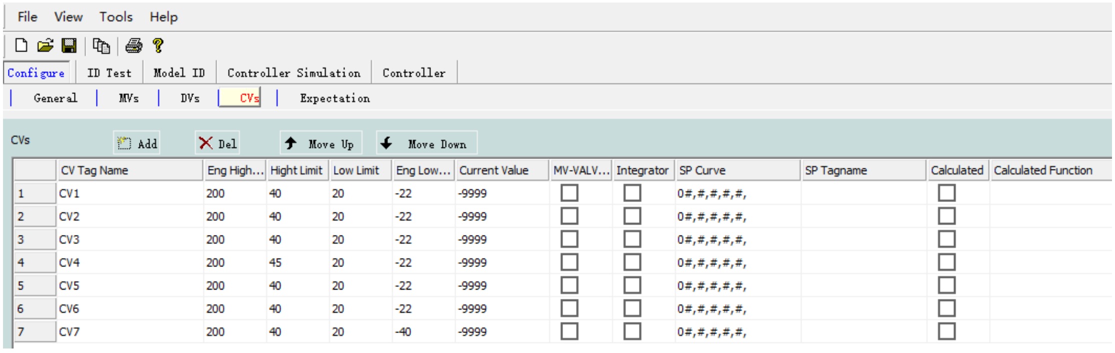
dblResult = Sqr (dbl1) + dbl2

SetCalcaulatedResult (dblResult)

5 .1.5 Configure CV Setpoint Curve

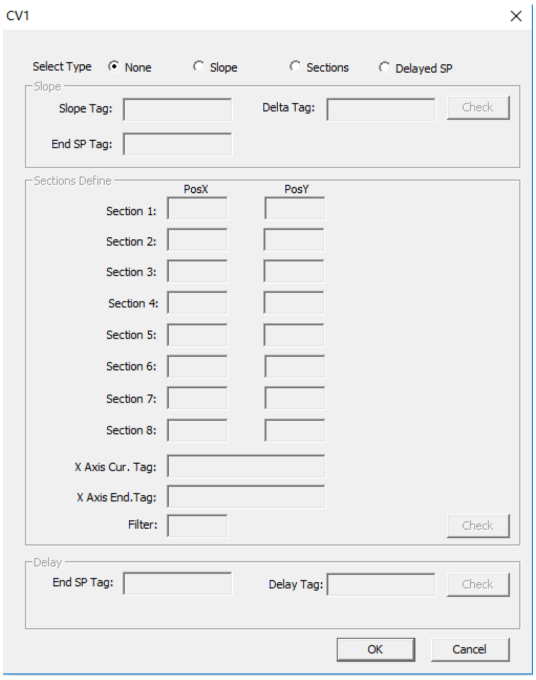
T ai-Ji MPC v2.7 and later versions have added the function of tracking the CV setpoint curve. This feature is significant for AGC coordination control of generator sets and control of intermittent processes. Here is the method to configure the CV setpoint curve.

Select Configuration Controlled Variable (CV s), open the Controlled Variable (CV s) window.



	CV Tag Name	Eng High...	Hight Limit	Low Limit	Eng Low...	Current Value	MV-VALV...	Integrator	SP Curve	SP Tagname	Calculated	Calculated Function
1	CV1	200	40	20	-22	-9999	☐	☐	0#,#,#,#,#,		☐	
2	CV2	200	40	20	-22	-9999	☐	☐	0#,#,#,#,#,		☐	
3	CV3	200	40	20	-22	-9999	☐	☐	0#,#,#,#,#,		☐	
4	CV4	200	45	20	-22	-9999	☐	☐	0#,#,#,#,#,		☐	
5	CV5	200	40	20	-22	-9999	☐	☐	0#,#,#,#,#,		☐	
6	CV6	200	40	20	-22	-9999	☐	☐	0#,#,#,#,#,		☐	
7	CV7	200	40	20	-40	-9999	☐	☐	0#,#,#,#,#,		☐	

Double-click the corresponding row of the CV in **0# ,#,#,#,#,** , a window will pop up as shown below



Select Type contains four options

None: Do not set the setpoint curve (default)

Slope: Ramp mode

Selections: Polyline Mode

Delayed SP: Delay Mode (Testing phase, temporarily unavailable)

Configure Slope Mode

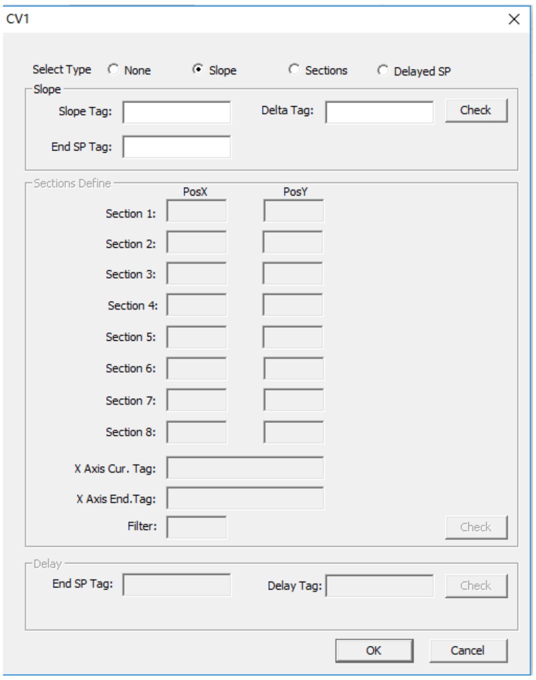
Open the setpoint curve configuration window and select Slope.

Slope Tag: Slope rate tag (change per minute, required)

End SP Tag: Target setpoint tag (required)

Delta Tag: Offset tag (optional). After filling in the offset tag, the target setpoint = target setpoint + offset

After configuring the required tags, click Check, then click OK to complete the slope mode configuration



Configure Polyline Mode

Open the setpoint curve configuration window and select Selections.

Slope Tag: X-axis rate of change tag (required)

Delta Tag: Offset amount tag (optional)

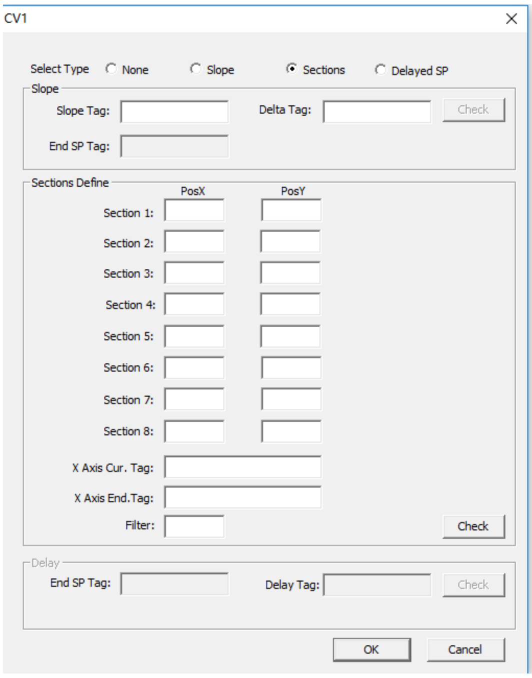
Section1~8: PosX fill in the corresponding point's horizontal coordinate, PosY fill in the corresponding point's vertical coordinate (Section1~8 must be fully filled)

X Axis Cur .Tag: X-axis current value tag (required)

X Axis End .Tag: X-axis target value tag (required)

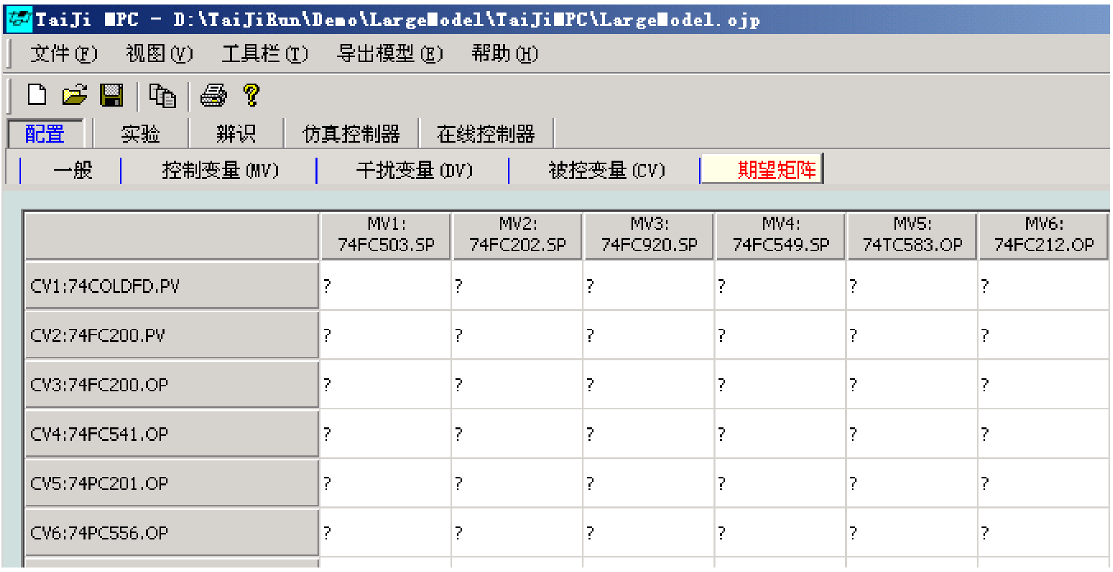
Filter: Filter time constant (optional, unit: minutes)

After configuring the required tags, click Check, then click OK button to complete the polyline mode configuration



5.1.6 Expectation Matrix

→Click with the mouse Configure Expectation Matrix to switch to the Expectation Matrix window as shown below:



This shows an expectation matrix that can be edited. Based on experience and relevant knowledge, users can determine the gain relationship between a certain MV (DV) and CV, and configure this table to improve identification accuracy.

The elements of the expectation matrix can be:

+: Corresponding MV (DV) , CV has a model, and the gain is positive
no: Corresponding MV (DV) , CV has no model
-: Corresponding MV (DV) , CV has a model, and the gain is negative
?: Unclear whether the corresponding MV (DV) , CV has a model

The expectation matrix is used in model identification and model selection. When used in model identification, if the expectation matrix indicates that there is no model between certain MVs and CVs (i.e., the corresponding elements of the expectation matrix are 'No '), then that model is excluded from the identification process, significantly reducing computation time and increasing model accuracy. When used in model selection, if the model's quality grade is A, B, or C and the model gain's sign matches the expectation matrix, then that model is selected and sent to the control module.

Note: If the menu **Toolbar Transpose MV, CV Model→** is selected, the table rows and columns will be transposed, allowing the user to choose their preferred way to view the MV, CV matrix

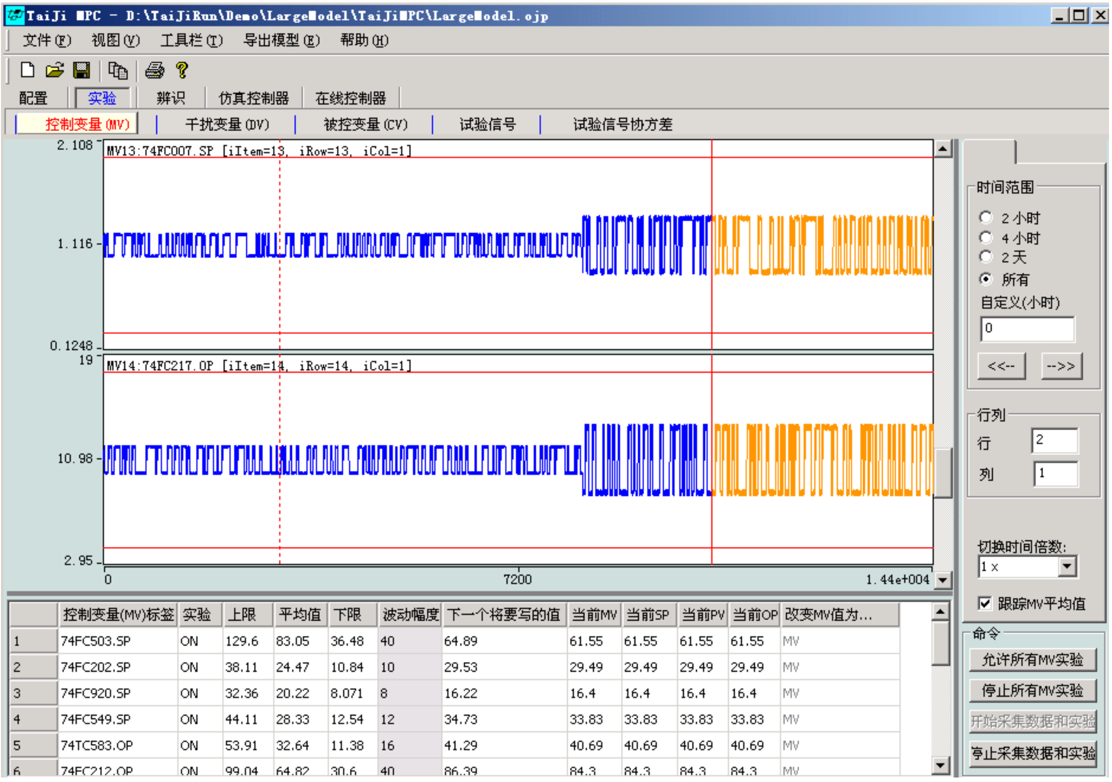
5. 2 Identification Experiment Module

The identification experiment module is used for conducting identification experiments, monitoring experiments, and data collection. It includes 5 windows:

MVs	DVs	CVs	Test signal	Covariance

5.2.1 Control Variable (MV) Window

→Click with the mouse on Experimental Control Variable (MV) to switch to the Control Variable (MV) window as shown below:



MV Trend Chart

In the MV trend chart, the blue line represents past signals, the orange line represents future actions based on experimental signals, and the two red horizontal lines represent the upper and lower limits of the CV. The vertical red line indicates the current time, and the vertical dashed line indicates the planned end time of the identification experiment. The actual experiment time may be slightly shorter or longer than planned.

Meaning of Table Columns:

Control Variable (MV) Tag : The tag of the control variable.

Test Variable Tag : The tag of the test variable, which can be different from the MV tag.

DCS Interface Tag Name: Tag name of the MV in the DCS interface

Upper Limit : The upper limit of the MV.

Average Value : Initial value of MV during the experiment; if 'Track MV Average Value' is set, it tracks the change of MV average value. If the MV average value changes, the changed MV is set as the average value.

The change, if the MV average value changes, the changed MV is set as the average value.

Lower Limit : The lower limit of the MV.

Current Value : The current value of the MV.

Fluctuation Range : Increment range of MV in identification experiment, the actual value of MV is the average value ± 0.5* fluctuation range

Description : Description of the control variable.

Meaning of MV Table Columns:

Control Variable (MV) Tag : The tag of the control variable.

Test Variable Tag : The tag of the test variable, which can be different from the MV tag.

Experiment : ON, allows fluctuation of MV value (i.e., adding experiment signal), otherwise only data collection.

Engineering Upper Limit: Engineering upper limit of MV

Upper Limit : The upper limit of the MV.

Average Value : Initial value of MV during the experiment; if 'Track MV Average Value' is set, it tracks the change of MV average value. If the MV average value changes, the changed MV is set as the average value.

Change, if the MV mean value changes, then set the changed MV as the mean value. If the system is closed-loop

control controlling the MV, it always tracks the mean value and cannot be edited.

Lower Limit : The lower limit of the MV.

Engineering Lower Limit: The engineering lower limit of the MV.

Current Value : The current value of the MV.

Fluctuation Range : The increment range of MV in the identification experiment, the actual value of MV is the mean value ± 0.5* fluctuation range *Next value to be written* : The MV value expected to be written in the next sampling period.

Current MV : The value of the current MV.

Current SP : The value of the current SP.

Current PV : The value of the current PV.

Current OP : The value of the current OP.

Mode: MV mode in DCS: Manual, Automatic, Cascade, Remote Cascade

MV Switch : Switch between MV's SP and OP (currently unavailable)

Graph Display Commands (on the right side of the window):

2 Hours : Display 2 hours of data in the graph

4 Hours : Display 4 hours of data in the graph

2 Days : Display 2 days of data in the graph

All : Display all data in the graph

Custom (hours) : Display data within a user-specified range in the graph

<<--: Display range moves forward "Current display range"

-->>: Display range moves backward "Current display range"

Row and Column Area:

Row : Number of rows in the graph

Column : Number of columns in the graph

Note: In the graph, hold down the left mouse button and move the mouse to select the display area.

Switch Time Multiplier : During the experiment, the number of times each experimental signal is reused = 2 * *Switch Time Multiplier* .

Indirectly achieve the change of switching frequency.

Track MV Average Value:

Same function as **Toolbar Track MV Average Value→** in the same menu. In open-loop experiments, if selected, the value modified by the operator will be taken as the new average value of MV.

Experiment Control Commands (at the bottom right of the window)

Allow all MV experiments : During the experiment, allow writing all MV values.

Stop all MV experiments : During the experiment, prohibit writing all MV values.

Start data collection and experiment : Start the experiment and collect data simultaneously. If the 'experiment' flag of MV is ON,

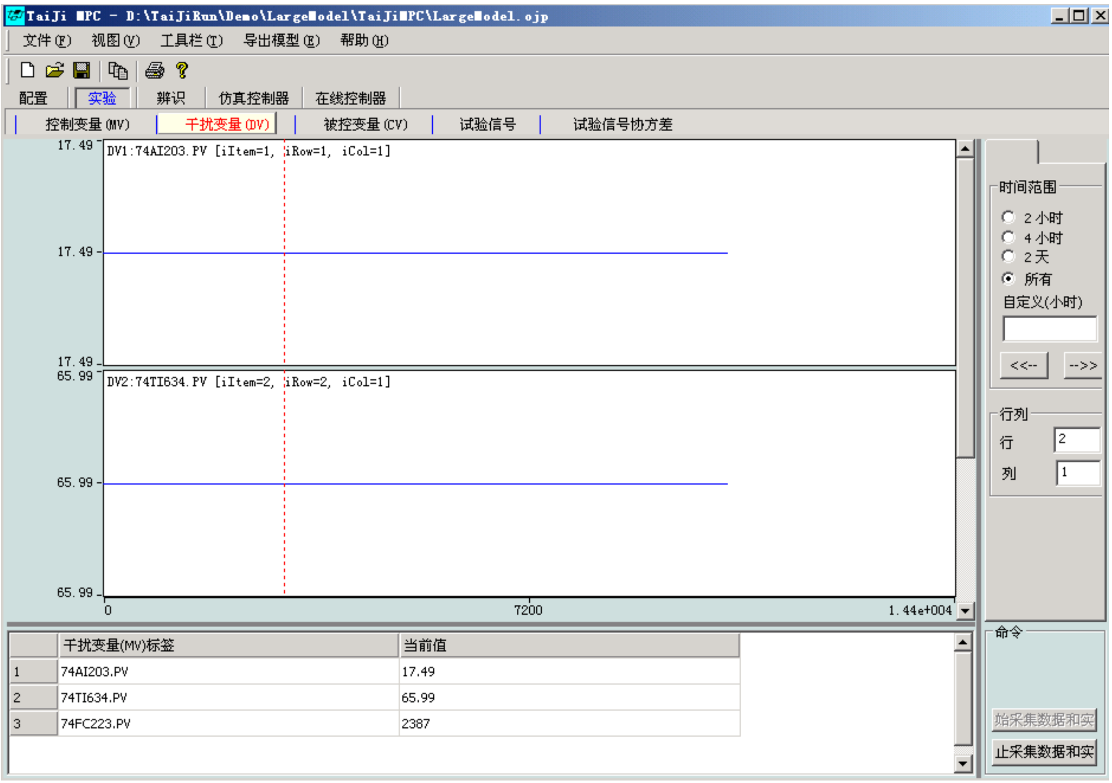
then experiment; otherwise, only collect data.

Note: When this button is pressed, the 'experiment' flag of all MVs will be automatically set to OFF, meaning only data collection.

Stop data collection and experiment : Stop the experiment and data collection simultaneously.

5.2.2 Disturbance Variable (DV) Window

Click with the mouse **Experimental Disturbance Variable (DV)→** to switch to the following **Disturbance Variable (DV)** window:



Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

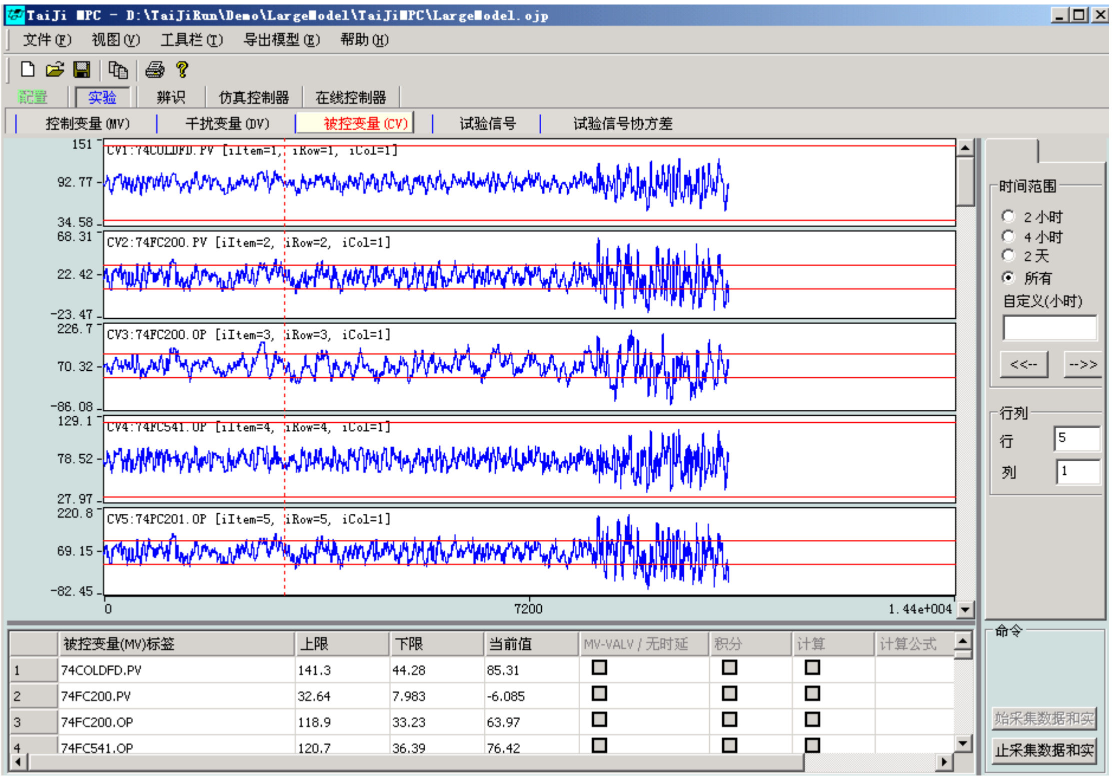
→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

Experimental Control Area

The Start Experiment button **and Stop Experiment** button **are the same as the →** Experimental Control Area **in the MV window of** Experimental Control Variable (MV)

5.2.3 Controlled Variable (CV) Window

Click with the mouse **Experimental Controlled Variable (CV)→** to switch to the following **Controlled Variable (CV)** window:



CV Trend Chart

In the CV trend chart, the blue line represents past signals, the orange line represents future actions based on experimental signals, and the two red horizontal lines represent the upper and lower limits of the CV. The vertical red line indicates the current time, and the vertical dashed line indicates the planned end time of the identification experiment. The actual experiment time may be slightly shorter or longer than planned.

Meaning of CV table columns:

Controlled Variable (CV) Tag : Tag of the controlled variable.

Engineering Upper Limit: Engineering upper limit of the CV

Upper Limit : Expected Upper Limit of CV.

Lower Limit : Expected Lower Limit of CV.

Engineering Lower Limit: Engineering lower limit of the CV

Current Value : Current Value of CV.

MV-VALV/No Delay : If selected, the model corresponding to CV has no delay.

Integral : If selected, CV is an integral variable.

Calculation : If selected, the value of the controlled variable will be calculated using the formula.

Calculation Formula : Calculation formula, effective only when the ' *Calculate* ' column is selected. After selecting the ' *Calculate* ' column, double-click with the mouse

This column pops up a dialog box that allows input and validation of calculation formulas.

Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

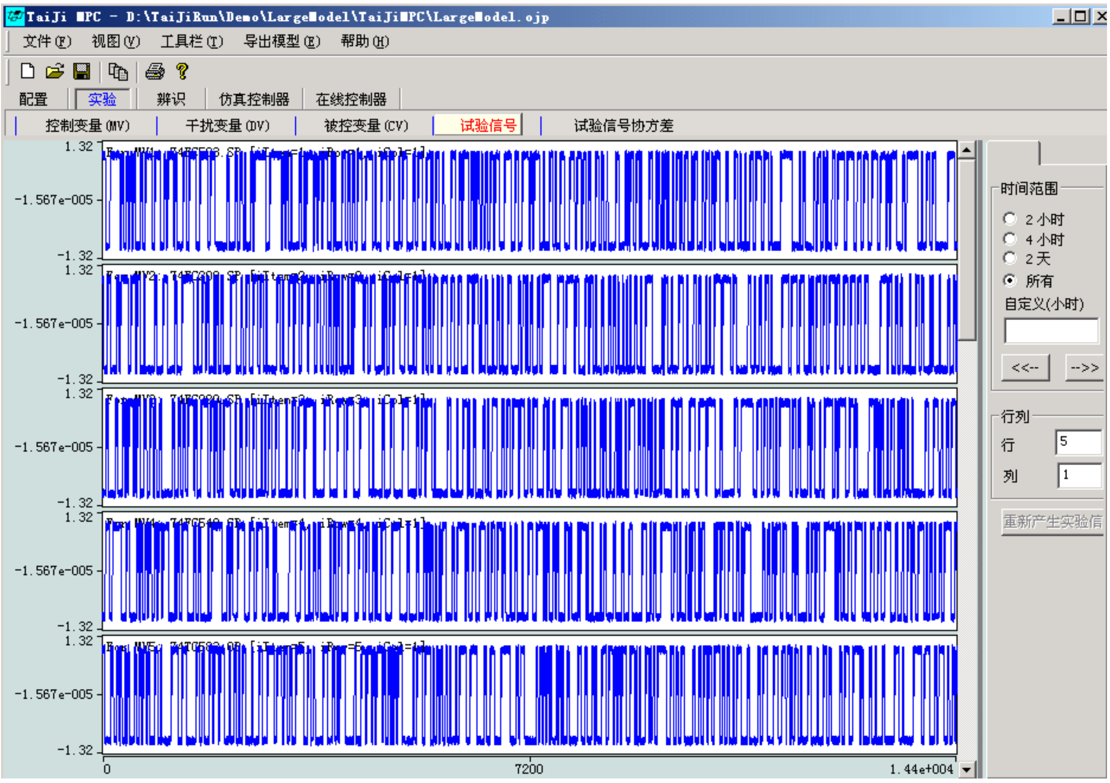
→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

Experimental Control Area

The Start Experiment button **and Stop Experiment** button **are the same as the** → Experimental Control Area **in the MV window of** Experimental Control Variable (MV)

5.2.4 Experimental Signal Window

Click with the mouse **Experimental Signal→** to switch to the **Experimental Signal** window as shown below:



Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

ReGenerate Button

Pressing the button will regenerate the experimental signal, which is randomly generated, so each regenerated experimental signal will be different, but the signal evaluation switch time remains unchanged. During the experiment, if there is no experimental signal, Tai-Ji MPC will automatically generate an experimental signal, so the user can completely ignore this button.

5 .2.5 Experimental Signal Covariance

Click with the mouse **Experimental Signal Covariance→** to switch to the **Experimental Signal Covariance** window as shown below, where the covariance matrix of the experimental signal is displayed:

	MV1: 74FC503.SP	MV2: 74FC202.SP	MV3: 74FC920.SP	MV4: 74FC549.SP	MV5: 74TC583.OP	MV6: 74FC212.OP	MV7: 74TC006.SP	MV8: 74PC012.SP	MV9: 74TC005.SP
MV1:74FC503.SP	1	6.16e-002	3.843e-002	-8.518e-002	5.799e-003	2.034e-002	-5.78e-002	2.722e-002	9.657e-003
MV2:74FC202.SP	6.16e-002	1	-5.983e-002	2.802e-002	5.789e-002	-3.515e-002	-4.788e-002	1.951e-002	8.606e-003
MV3:74FC920.SP	3.843e-002	-5.983e-002	1	4.771e-003	-7.381e-003	-2.597e-002	7.393e-003	-7.586e-002	-9.435e-002
MV4:74FC549.SP	-8.518e-002	2.802e-002	4.771e-003	1	-1.078e-002	-4.526e-002	-4.595e-003	1.009e-002	-2.051e-003
MV5:74TC583.OP	5.799e-003	5.789e-002	-7.381e-003	-1.078e-002	1	6.856e-002	-7.415e-002	-4.56e-002	1.111e-002
MV6:74FC212.OP	2.034e-002	-3.515e-002	-2.597e-002	-4.526e-002	6.856e-002	1	2.993e-002	-1.87e-002	-1.024e-002
MV7:74TC006.SP	-5.78e-002	-4.788e-002	7.393e-003	-4.595e-003	-7.415e-002	2.993e-002	1	9.312e-003	-2.906e-002
MV8:74PC012.SP	2.722e-002	1.951e-002	-7.586e-002	1.009e-002	-4.56e-002	-1.87e-002	9.312e-003	1	3.558e-002
MV9:74TC005.SP	9.657e-003	8.606e-003	-9.435e-002	-2.051e-003	1.111e-002	-1.024e-002	-2.906e-002	3.558e-002	1
MV10:74FC010.SP	4.577e-002	-1.604e-002	3.267e-002	-1.321e-002	2.711e-002	-3.447e-002	1.398e-002	4.727e-002	4.86e-002
MV11:74TC003.SP	5.021e-004	-1.522e-002	2.575e-002	2.706e-002	5.804e-002	-7.232e-003	-1.274e-002	-4.054e-002	4.636e-002
MV12:74FC008.SP	-2.095e-002	1.217e-002	-7.994e-003	9.108e-003	5.901e-002	2.626e-002	-3.683e-002	3.279e-002	-7.377e-003
MV13:74FC007.SP	-2.928e-002	-2.602e-002	-3.863e-002	3.23e-002	-5.422e-002	-9.654e-003	1.945e-002	-8.64e-003	-2.459e-003
MV14:74FC217.OP	-9.248e-003	-3.237e-003	5.964e-002	2.083e-002	-7.545e-002	2.982e-002	4.018e-002	7.383e-002	-3.313e-002
MV15:74FC504.SP	4.222e-003	-1.225e-002	6.646e-002	-1.763e-002	-2.716e-002	-2.165e-002	8.008e-002	1.244e-002	-2.242e-002
MV16:74FC522.SP	3.203e-002	8.324e-003	1.481e-002	3.431e-002	-1.486e-002	-2.231e-002	4.257e-003	6.887e-002	9.903e-003

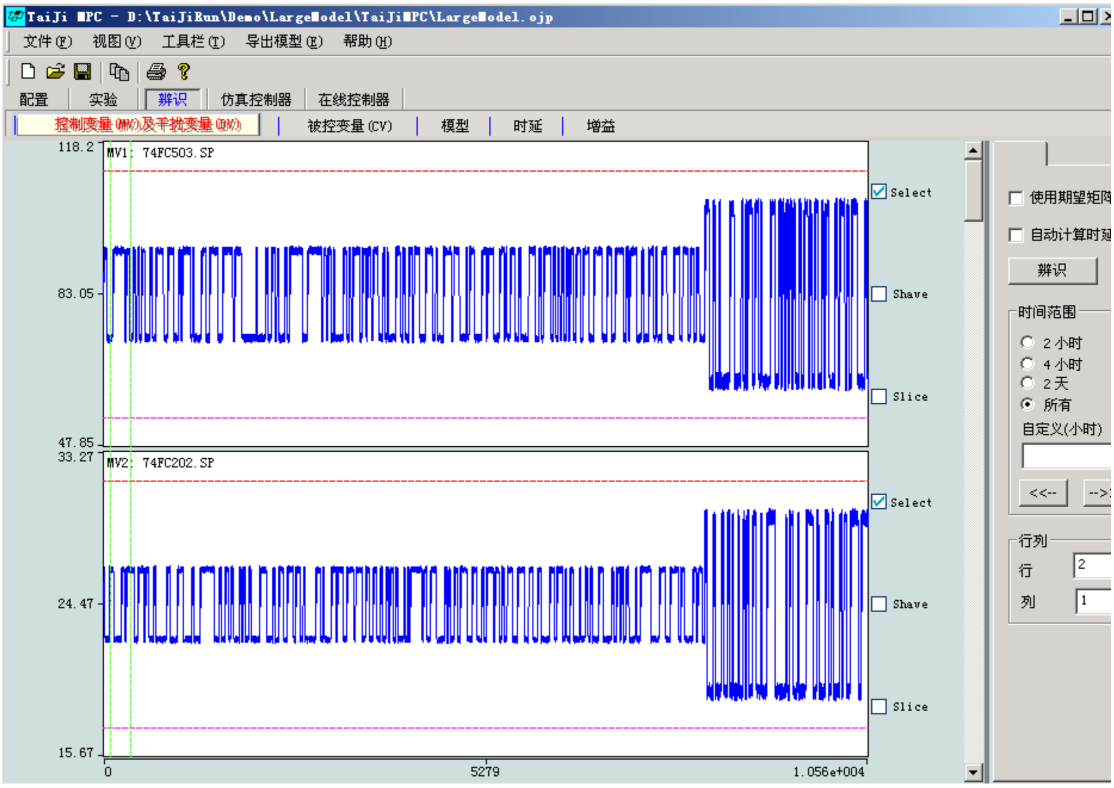
5.3 Model Identification Module

The model identification module performs model identification, displays the identified model, and loads the model into the controller simulation module and controller module. The model identification module includes the following five windows:

MVs & DVs	**CVs**	**Model Response**	**Delay**	**Gain**

5 .3.1 Control Variables (MV) and Disturbance Variables (DV) Window

→Click with the mouse Identify Control Variables (MV) and Disturbance Variables (DV) to switch to the Control Variables (MV) and Disturbance Variables (DV) window as shown below. In the model identification module, MV and DV are placed in the same window because DV and MV are considered to have the same function, both being inputs to the production process.



MV and DV Trend Chart

Displays the trend chart of MV and DV signals generated in the identification module

MV Operation Command

Select : If selected, the identified MV will be used; otherwise, it will not be used

Shave : If Shave operation has been performed, it will be automatically selected; if not selected, Shave will be canceled

Slice : If Slice operation has been performed, it will be automatically selected; if not selected, Slice will be canceled

How to Shave (limit the maximum and minimum values of the signal): Drag the upper red horizontal dashed line on the trend chart to change the maximum value, and drag the lower red horizontal dashed line to change the minimum value.

How to Slice (cut off certain segments of the signal): 1) Double-click the vertical green dashed line in the trend chart with the mouse to add signal segments; move these dashed lines to the position you want to cut. 2) Double-click the mouse in the trend chart (the mouse position is not on any dashed line), and if the color of the corresponding signal segment changes, it indicates that the cut is successful, and the signal segment will not be used. 3) The slicing of all MV signals is the same and applies to all MV signals. 4) The slicing of MV signals will be automatically mapped to all CV signals, performing the same slicing. Further slicing of CV signals can also be done in the **Identified Controlled Variables (CV)**→ window.

Use expectation matrix

When selected, model identification will exclude empty models corresponding to No elements in the expectation matrix. This reduces computation time and improves model accuracy. When **Use expectation matrix** is not selected, all models will be identified.

Auto estimate delay

When selected, model identification will automatically identify and use the model's delay. When not selected, model identification will use the delay in the **Delay Window** .

Identify button

Starting model identification is very simple. Click the Identify button to automatically proceed. If “Auto estimate delay” is selected, the user will then see a progress bar displaying “Estimate delay…” , followed by another progress bar displaying “Identify model…” . When the model identification computation is complete, the identified model will be displayed in the Identified Model window.→

Note: The identification results will not immediately affect the MPC controller unless the user selects 'Automatically use the latest identification model during control' in the Configuration General window.→

Graph Display Commands (on the right side of the window):

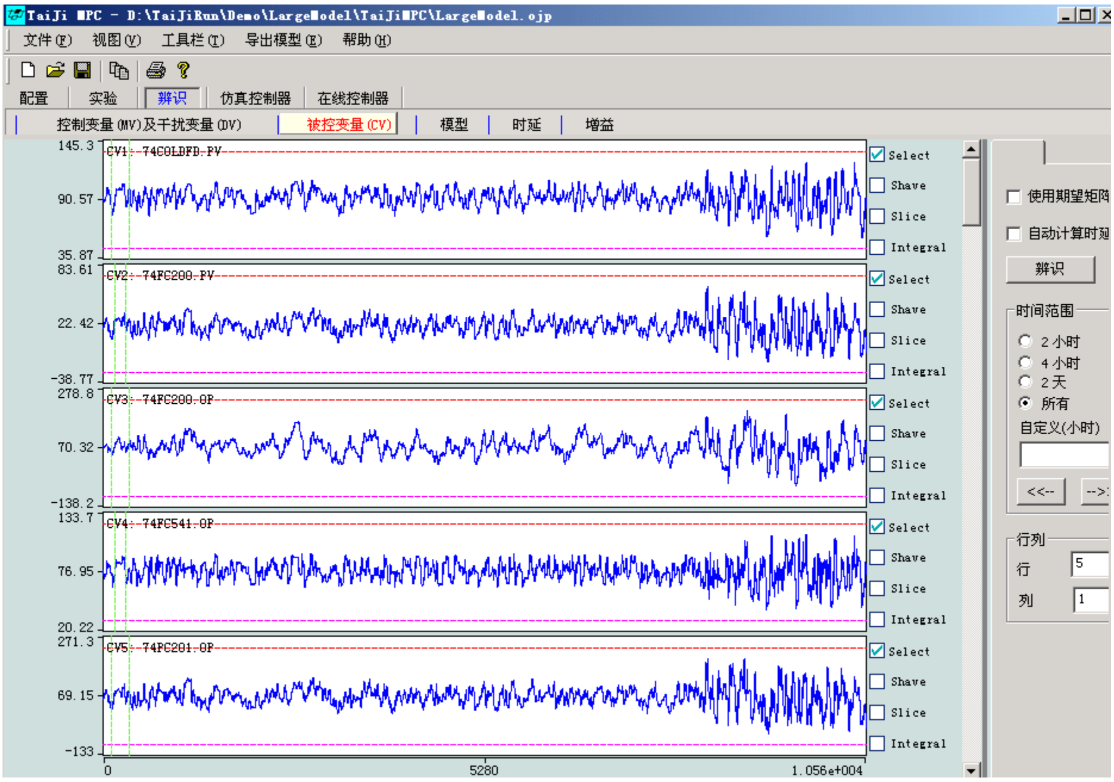
→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

5 .3.2 Controlled Variable (CV) Window

→Click with the mouse on Identify Controlled Variable (CV) to switch to the Controlled Variable (CV) window as shown below:



CV Trend Chart

Displays the trend graph of the CV signal generated in the identification module

CV Operation Commands:

Select : If selected, the identified MV will be used; otherwise, it will not be used

Shave : If Shave operation has been performed, it will be automatically selected; if not selected, Shave will be canceled

Slice : If Slice operation has been performed, it will be automatically selected; if not selected, Slice will be canceled

Integral : Check for integral, otherwise not integral

How to Shave (limit the maximum and minimum values of the signal): Drag the upper red horizontal dashed line on the trend chart to change the maximum value, and drag the lower red horizontal dashed line to change the minimum value.

How to Slice (cut off certain segments of the signal): 1) Double-click the vertical green dashed line in the trend graph with the mouse to add signal segments; move these dashed lines to the position where you want to cut. 2) Double-click the mouse in the trend graph (the mouse position is not on any dashed line), the color of the corresponding signal segment changes, indicating that the cut is successful, and the signal segment will not be used. 3) Each CV signal is cut individually and can be different.

Use expectation matrix

Same as **Identification Control Variables (MV) and Disturbance Variables (DV)→** window

Auto estimate delay

Same as **Identification Control Variables (MV) and Disturbance Variables (DV)→** window

Identify button

Same as **Identification Control Variables (MV) and Disturbance Variables (DV)→** window

Graph Display Commands (on the right side of the window):

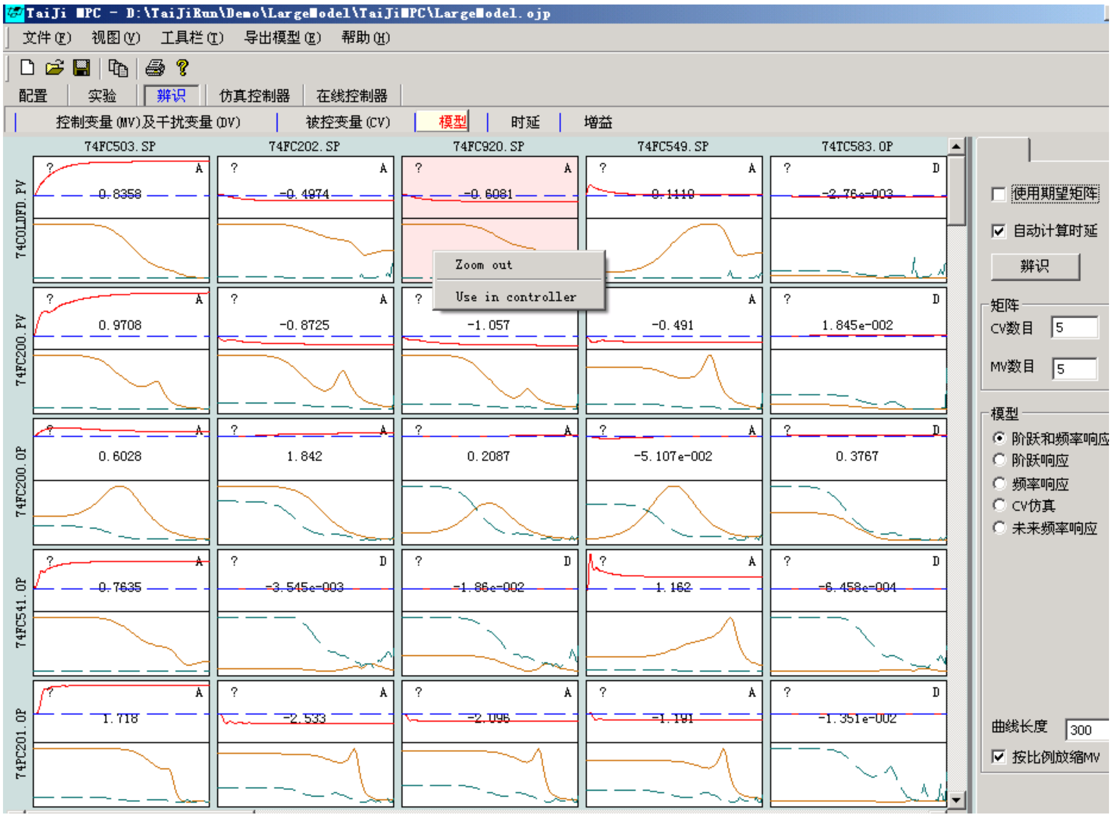
→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

5 .3.3 Model Window

Click **Identification Model →** to open the window shown below, which includes several windows such as the model step response window, the frequency response window (with error bounds), the model simulation window, and the future frequency response window. Step response and frequency response can also be displayed in the same window. These windows can be selected for display using the buttons in the **Plot type** area. Click **Identification Model→** with the mouse to switch to the view shown below:



Each step response or frequency response plot also displays additional information to assist in model validation and selection. The top left corner of the plot shows the corresponding value from the expected matrix (+, -, ? or No) , the center of the plot shows the model gain, and the top right corner shows the model quality grade A (Excellent), B (Good), C (Average), and D (Poor or No Model). Generally, if the model quality grade is A, B, or C and the sign of the model gain matches the expected matrix, then the model is selected and used in the controller.

The method for selecting a model is as follows:

- To select a model for the controller, right-click on the model plot and choose " **Use in controller** " . The model will then be loaded into the **Online Controller** module and the **Simulation Controller** module.
- To select all suitable models for the controller, right-click on the model diagram and choose "Use all valid models in controller" . All models with quality A, B, or C and whose gain signs match the desired matrix will be loaded into the **Online Controller** module and the **Simulation Controller** module.
- To select all models, including those with quality D, without considering the desired matrix, right-click on the model diagram and choose "Use all models in controller" . All models will be loaded into the **Online Controller** module and the **Simulation Controller** module.

Based on the identification results, users can modify the expected matrix: right-click on the model diagram and then select "Change expect value" , then you can modify the expected value of the corresponding model (+, -, ? or No) .

Right-click on the model diagram and then select “Zoom out” , then the window will display the response curves of the entire model.

Clicking the **Simulation** button will display the measured and estimated values of the CV, as shown in the figure below. The ERROR% of the CV is the ratio of the standard deviation of the estimation error to the standard deviation of the CV. Empirical evidence suggests that good identification results correspond to an ERROR% typically between 1% and 40%.

Use expectation matrix

Same as **Identification Control Variables (MV) and Disturbance Variables (DV)→** window

Auto estimate delay

Same as **Identification Control Variables (MV) and Disturbance Variables (DV)→** window

Identify button

Same as **Identification Control Variables (MV) and Disturbance Variables (DV)→** window

Row and Column Area:

CVs: Number of CVs displayed in the model response curve

MVs: Number of MVs displayed in the model response curve

Different model display modes (on the right side of the window):

Step and Frequency Response **:** Displays both the model step and frequency response.

Step Response **:** Displays only the model step response.

Frequency Response **:** Model step frequency response.

CV Simulation **:** Simulation curve of the CV.

Future Frequency Response **:** During experiments, the model frequency response after a period of time in the future, used to improve the experiment.

Curve Length : Length of the model step and frequency response

Scale MV proportionally: If selected, the step and frequency response of the normalized model will be displayed.

Zoom in : In the graph, hold down the left mouse button and move the mouse to select the model area to be zoomed in.

Note: If the model quality is A, B, or C, but the sign of the gain does not match the sign of the expected matrix, you must be cautious and re-examine the relationship between the corresponding MV (DV) and CV.

The following is a brief description of how this program manages models:

The system currently uses 3 model matrices:

Initial model : If you already have a model before using this program, you can import it according to the specified format.

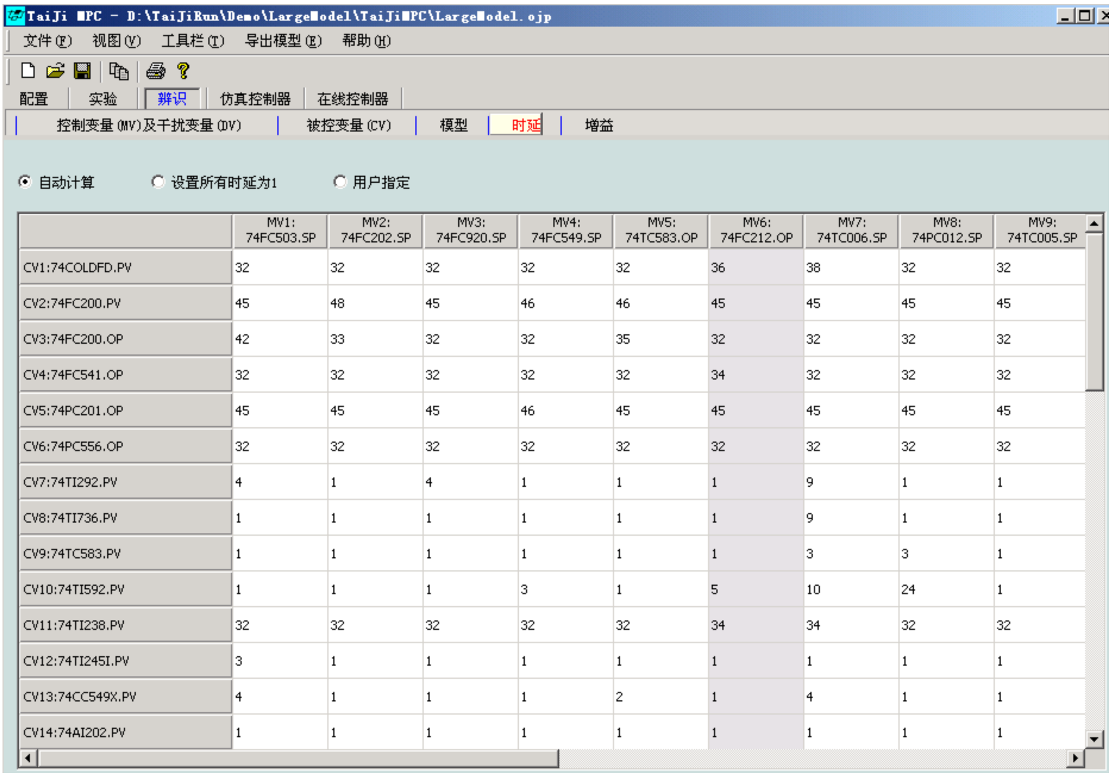
Identification Model : The model obtained from the last identification run of the program

Control Model : The model used by the controller (both the simulation controller and the online controller use the same model)

→Generally, the result of the identification does not affect the Control Model , unless the user has set 'Automatically use the latest identification model during control'. In the Identification Model window (excluding the CV Simulation window), the user can right-click to bring up a menu and use the menu commands to place the identification model into the control model; the user can also operate on the control model in the Simulation Controller Model window or the Online Controller Model window.→→

5 .3.4 Delay Window

→Click with the mouse Identify Delay to switch to the delay window as shown below:



	MV1: 74FC503.SP	MV2: 74FC202.SP	MV3: 74FC920.SP	MV4: 74FC549.SP	MV5: 74TC583.OP	MV6: 74FC212.OP	MV7: 74TC006.SP	MV8: 74PC012.SP	MV9: 74TC005.SP
CV1:74COLDFD.PV	32	32	32	32	32	36	38	32	32
CV2:74FC200.PV	45	48	45	46	46	45	45	45	45
CV3:74FC200.OP	42	33	32	32	35	32	32	32	32
CV4:74FC541.OP	32	32	32	32	32	34	32	32	32
CV5:74PC201.OP	45	45	45	46	45	45	45	45	45
CV6:74PC556.OP	32	32	32	32	32	32	32	32	32
CV7:74TI292.PV	4	1	4	1	1	1	9	1	1
CV8:74TI736.PV	1	1	1	1	1	1	9	1	1
CV9:74TC583.PV	1	1	1	1	1	1	3	3	1
CV10:74TI592.PV	1	1	1	3	1	5	10	24	1
CV11:74TI238.PV	32	32	32	32	32	34	34	32	32
CV12:74TI245I.PV	3	1	1	1	1	1	1	1	1
CV13:74CC549X.PV	4	1	1	1	2	1	4	1	1
CV14:74AI202.PV	1	1	1	1	1	1	1	1	1

Auto Calculate :

If selected, the delay will be calculated immediately, and before each identification, the delay will be automatically estimated.

Set All Delays to 1 :

If selected, all delays will be set to 1 immediately.

User Specified :

If selected, the user-defined delay will be used during identification.

Use Same Delay Box

If the user selects a CV and defines its delay, all delays related to that CV will use the user-defined delay as the general delay.

The user can also edit the delay matrix. The default delay for most CVs is 1, while the delay for CVs marked as **"MV-VALVE/No delay"→ in the** ConfigureCV Window is 0.

'Transpose MV, CV Model' is selected, the rows and columns of the table will be swapped. Users can choose their preferred way to view the MV, CV matrix.

5 .3.5 Gain Window

Click with the mouse on **Identify Gain→** to switch to the **Gain** window as shown below:

	MV1: 74FC503.SP	MV2: 74FC202.SP	MV3: 74FC920.SP	MV4: 74FC549.SP	MV5: 74TC583.OP	MV6: 74FC212.OP	MV7: 74TC006.SP	MV8: 74PC012.SP	MV9: 74TC005.SP
CV1:74COLDFD.PV	0.8358	-0.4974	-0.6081	0.1119	-2.76e-003	-0.4909	1.423e-005	-2.117e-003	-8.758e-004
CV2:74FC200.PV	0.9708	-0.8725	-1.057	-0.491	1.845e-002	0.1331	-4.476e-004	2.049e-003	4.874e-003
CV3:74FC200.OP	0.6028	1.842	0.2087	-5.107e-002	0.3767	2.293	-1.167	-0.1325	5.576e-002
CV4:74FC541.OP	0.7635	-3.545e-003	-1.86e-002	1.162	-6.458e-004	-0.2269	-2.031e-003	1.353e-002	-1.356e-003
CV5:74PC201.OP	1.718	-2.533	-2.096	-1.191	-1.351e-002	-1.177	-6.255e-004	-1.672e-002	6.611e-004
CV6:74PC556.OP	-2.354e-005	2.518e-004	2.985e-004	3.822	1.371e-004	-1.933e-004	2.214e-005	1.449e-004	9.391e-006
CV7:74TI292.PV	-0.8659	0.294	2.507	0.2101	0.5084	-1.229	0.4481	-0.1682	0.1197
CV8:74TI736.PV	0.2824	5.908	-1.09	0.5921	-0.2435	0.2549	0.553	0.1706	7.965e-002
CV9:74TC583.PV	7.657e-003	5.027e-002	1.98e-002	-4.055	-0.8954	-1.628	-0.9192	2.893	6.161e-003
CV10:74TI592.PV	-2.422e-003	0.3152	-0.6369	-1.137	-3.424e-002	-1.186	-0.3666	2.005	7.952e-003
CV11:74TI238.PV	-1.387	1.131e-003	6.976e-004	-1.375e-003	-3.139e-003	-1.903e-002	3.792e-004	1.507e-003	-1.249e-003
CV12:74TI245I.PV	-1.661	5.675e-003	3.691e-003	-0.9108	5.452e-003	1.085	-1.816e-004	1.393e-003	2.387e-003
CV13:74CC549X.PV	-4.917e-003	3.933e-004	1.073e-003	-5.013e-003	-1.295e-003	4.015e-003	-1.218e-004	-4.089e-004	-9.357e-005
CV14:74AI202.PV	-0.4104	6.831e-003	2.232e-003	-0.3798	7.901e-004	0.2349	-0.1295	-3.933e-003	8.351e-004
CV15:74TI247.PV	3.892e-004	5.945e-004	-4.362e-003	2.547	-5.025e-003	-1.126	1.615	2.935e-003	-1.031e-003
CV16:74AHMXSK.PV	4.195e-003	0.1133	-6.483e-002	-4.393e-003	4.657e-003	-2.361e-002	6.534	4.111e-003	-1.654e-002

The matrix displays the gain information between CV and MV in the identification model. Users cannot edit or modify it, but the gain of the controller model can be modified. See the next section for details.

'Transpose MV, CV Model' is selected, the rows and columns of the table will be transposed. Users can choose their preferred way to view the MV, CV matrix.

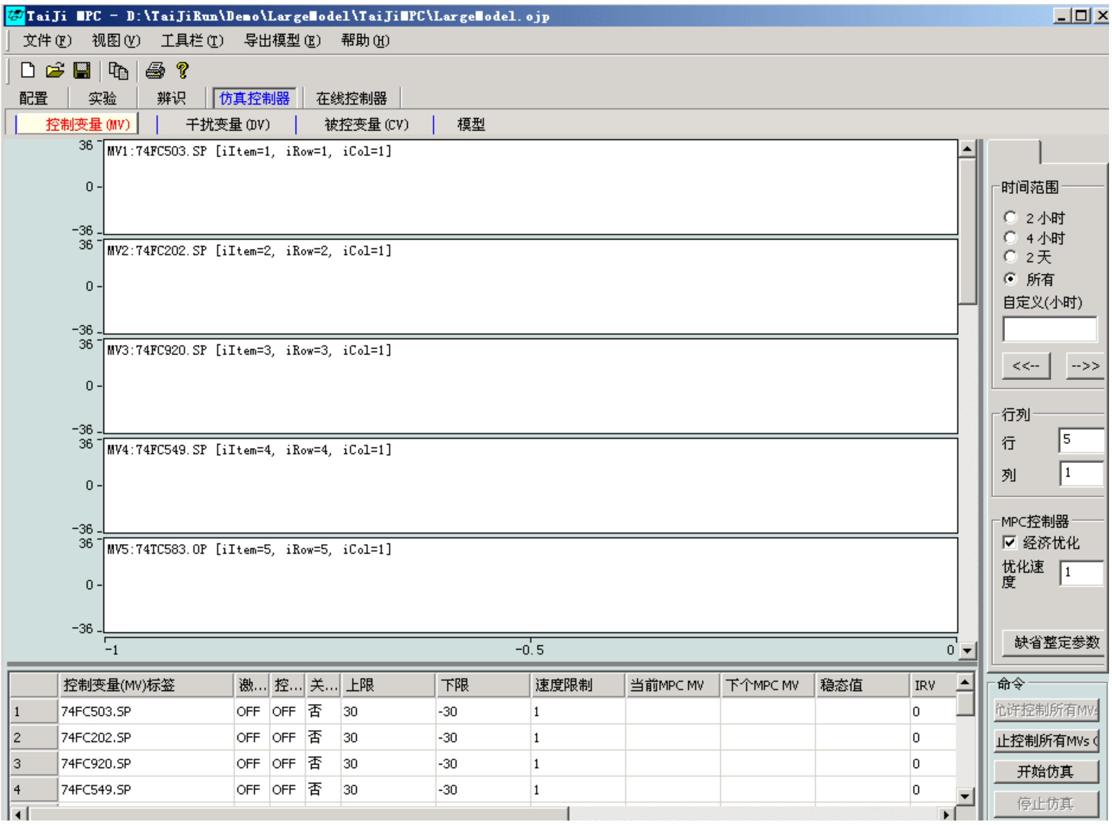
5. 4 Simulation Controller Module

The simulation controller module uses the controller model obtained from the model identification module (or imported from other platforms) for MPC simulation. The simulation controller module includes the following 6 windows:

MVs	DVs	CVs	Model	Gain	Tuning

5 .4.1 Control Variable (MV) Window

→Click with the mouse on Simulation Controller Control Variable (MV) to switch to the Control Variable (MV) window as shown below:



Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control

Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

MPC Controller:

Economic Optimization : When selected, the controller starts the economic optimization program; otherwise, only dynamic control is performed.

Optimization Speed : The speed of economic optimization. A larger value indicates slower optimization speed. If set to 1, the optimization speed is the same as the dynamic control speed. The recommended optimization speed is between 3 and 6.

Control Commands:

Allow control of all MVs : Press to allow the simulation controller to use the MVs.

Stop control of all MVs : Press to prohibit the simulation controller from using any MVs.

Start simulation : Press to automatically prohibit the use of any MVs, then create a simulation controller and start the model simulation.

Stop simulation : Press to stop the simulation.

Meaning of MV Table Columns:

Control Variable (MV) Tag : The tag of the control variable.

Activation flag : OFF, the simulation controller is not using this MV or there is a fault;
ON, the simulation controller is using this MV.

Control command : OFF, the simulation controller does not use this MV;
ON, the simulation controller uses this MV.
Under normal circumstances, it is the same as the *activation flag* value (may differ by one sampling period).

Key Variable : If an MV is a key variable, the entire MPC controller stops when that MV is uncontrollable.

Engineering Upper Limit: Engineering upper limit of MV

Upper *Limit* : The upper limit of the MV.

Lower Limit : The lower limit of the MV.

Engineering Lower Limit: The engineering lower limit of the MV.

Speed Limit : The upper limit of the MV increment per sampling period.

Current Value MPC MV : The MV value currently used by the simulation controller for calculation.

Next MPC MV : The MV value predicted by the simulation controller.

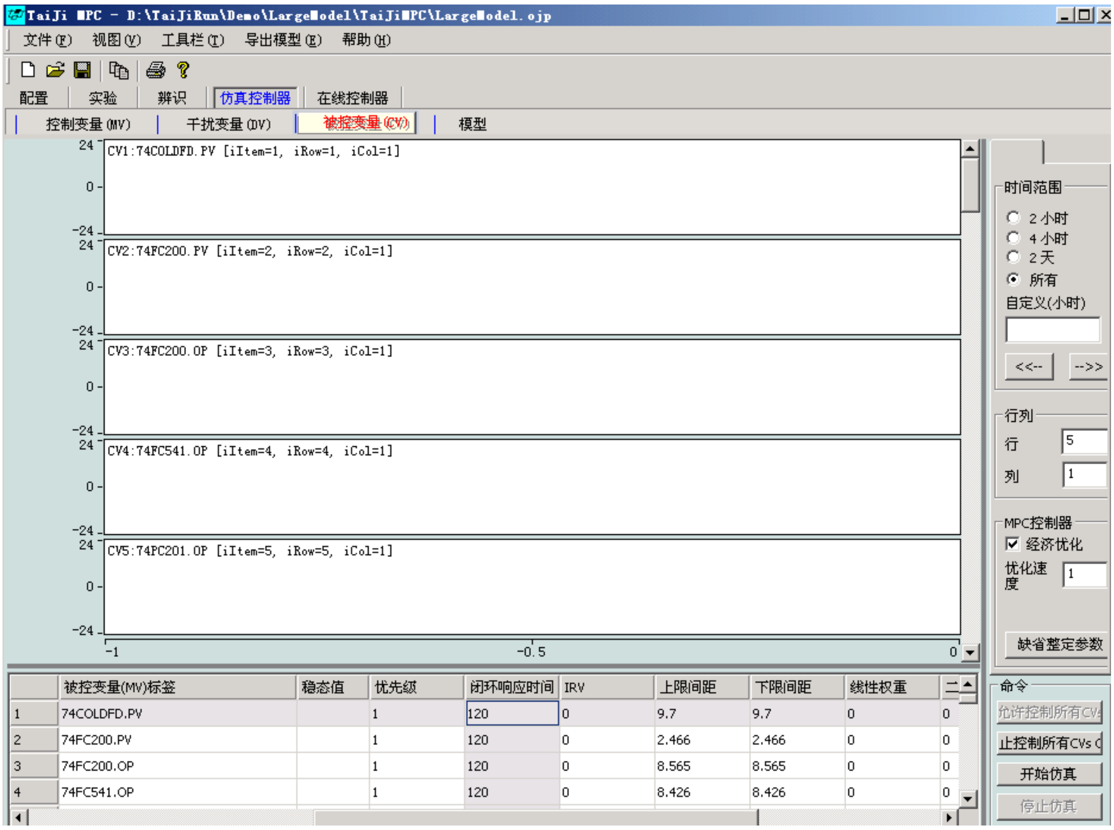
Steady State Value : The MV value when the system is in steady state, calculated by the MPC controller.

5 .4.2 Disturbance Variable (D V) Window

In the **simulation controller** module, DV is always kept at zero and does not play a role in the simulation. The **disturbance variable (DV) window** will be introduced later in the **controller** module.

5 .4.3 Controlled Variable (CV) Window

Click with the mouse on the **Simulated Controller Controlled Variable (CV)→** to switch to the Controlled Variable (CV) window as shown below. Please set the upper and lower limits of the CV according to the values in the figure. The initial value of the CV in the simulation is the average of the upper and lower limits of the CV.



Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

MPC Controller:

→Same as the Simulated Controller Manipulated Variable (MV) in the MV window of the MPC Controller

Control Commands:

Allow control of all CVs : After pressing, the CVs used by the simulated controller are allowed.

Stop control of all CVs : After pressing, the simulated controller is prohibited from using any CV.

Start Simulation : Press to start the model simulation.

Stop simulation : Press to stop the simulation.

Meaning of CV table columns:

Controlled Variable (CV) Tag : The tag of the controlled variable.

Activation Flag : OFF, the simulation controller is not using this CV or it has a fault;
ON, the simulation controller is using this CV.

Control Command : OFF, the simulation controller does not use this CV;
ON, the simulation controller uses this CV.
Under normal circumstances, it is the same as the *activation flag* value (may differ by one sampling period).

Critical Variable : If a CV is a critical variable, the entire MPC controller stops when this CV is uncontrollable.

Range/Setpoint : The control mode of the CV. Select 'Setpoint' to control the CV to a specific value.
Select 'Range' to control the CV within a range.

Setpoint : The setpoint of this CV

Engineering Upper Limit: Engineering upper limit of the CV

Upper Limit : The upper limit of this CV.

Lower Limit : The lower limit of this CV.

Engineering Lower Limit : The engineering lower limit of this CV

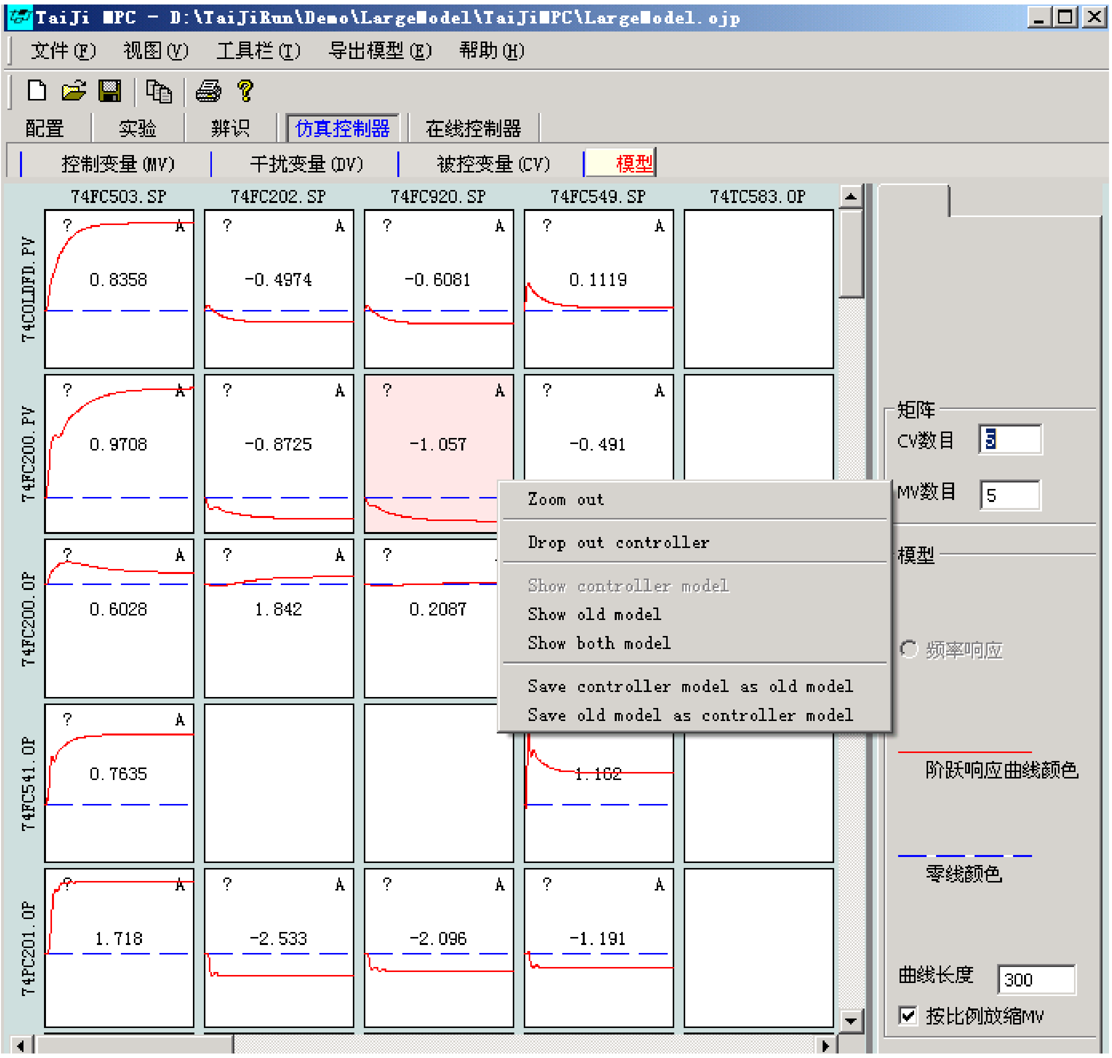
Current Value : The current value of this CV

Steady State Value : The steady state value of this CV calculated by the controller

Priority : The priority of this CV. It takes a positive integer, the smaller the value, the higher the priority. When the control of several CVs conflicts, the controller will abandon the CV with lower priority and only control the CV with higher priority.

5 .4.4 Model Window

Click with the mouse **Simulation Controller Model→** to switch to the **Model** window as shown below:



Each step response plot also shows the desired matrix, the model's gain, and the model's quality grade. The corresponding values in the desired matrix are displayed in the upper left corner of the plot (+, -, ? or No) , the central value of the plot is the model's gain, and the upper right corner of the plot shows the model's quality grade A (Excellent), B (Good), C (Average), and D (Poor or No Model). Right-clicking on the model plot will present a menu bar that allows several operations such as copy/paste model, modify model, manually add model, etc.

Note: The control models under the Simulation Controller and Online Controller are the same, and the above operations on the model will also affect the model in the Online Controller module.

Row and Column Area:

CVs: Number of CVs displayed in the model response curve

MVs: Number of MVs displayed in the model response curve

Different model display methods:

Curve Length : Length of the model step and frequency response

Scale MV proportionally: If selected, the step and frequency response of the normalized model will be displayed.

Graphic element right-click menu:

Zoom out : Display all models; if there are too many models, they may not be displayed, and only the background color will be shown in the graphic,

At this time, you need to set the number of CVs and MVs to appropriate values

Copy: Copy model

Paste: Paste model at the current position

Drop out controller : The MPC controller (simulation and online) does not use this model

Show controller model : Display controller model

Show old model : Display initial model (model management)

Show both model : Display both the controller model and the initial model simultaneously

Save controller model as old model : Save the controller model as the initial model, overwriting the initial model,

The operation is irreversible.

Save old model as controller model : Save the initial model as the controller model, overwriting the controller model,

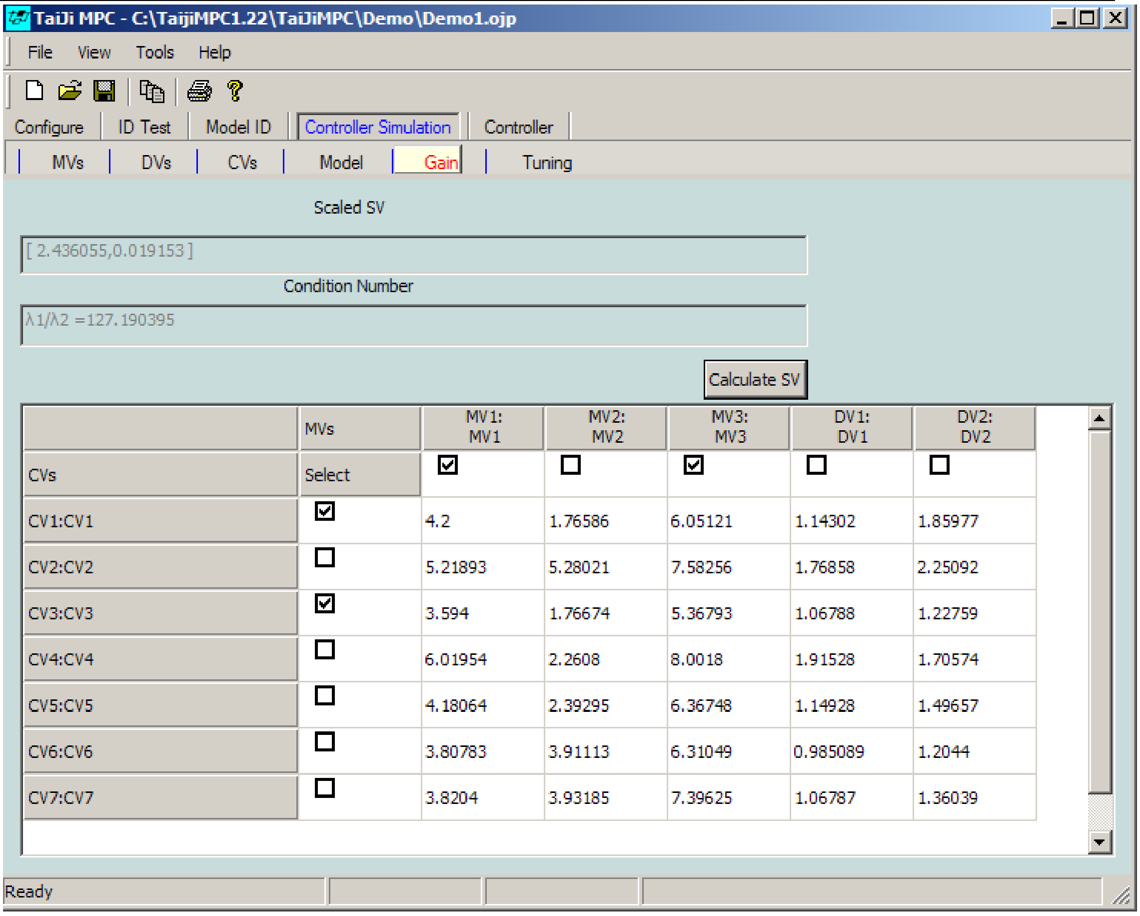
The operation is irreversible.

Modify controller model: Modify the current model. When selected, a new window will pop up displaying the model's step response. Users can specify the model's new gain and transition time and check the modified model.

Add/reset controller model: When selected, users can manually add a first-order or second-order model.

5 .4. 5 Gain Window

Click with the mouse **Simulation Controller Gain→** to switch to the **Gain** window as shown below:



The Gain window displays the gain matrix of the model. Users can select subsystems and use singular value decomposition techniques to analyze the controllability of the subsystem.

In the **Gain** window, select two CVs and two MVs, then click the **Calculate SV** button (SV stands for Singular Value). This will display the singular values and condition number (the ratio of the first singular value to the second singular value) of the selected 2 x 2 gain matrix.

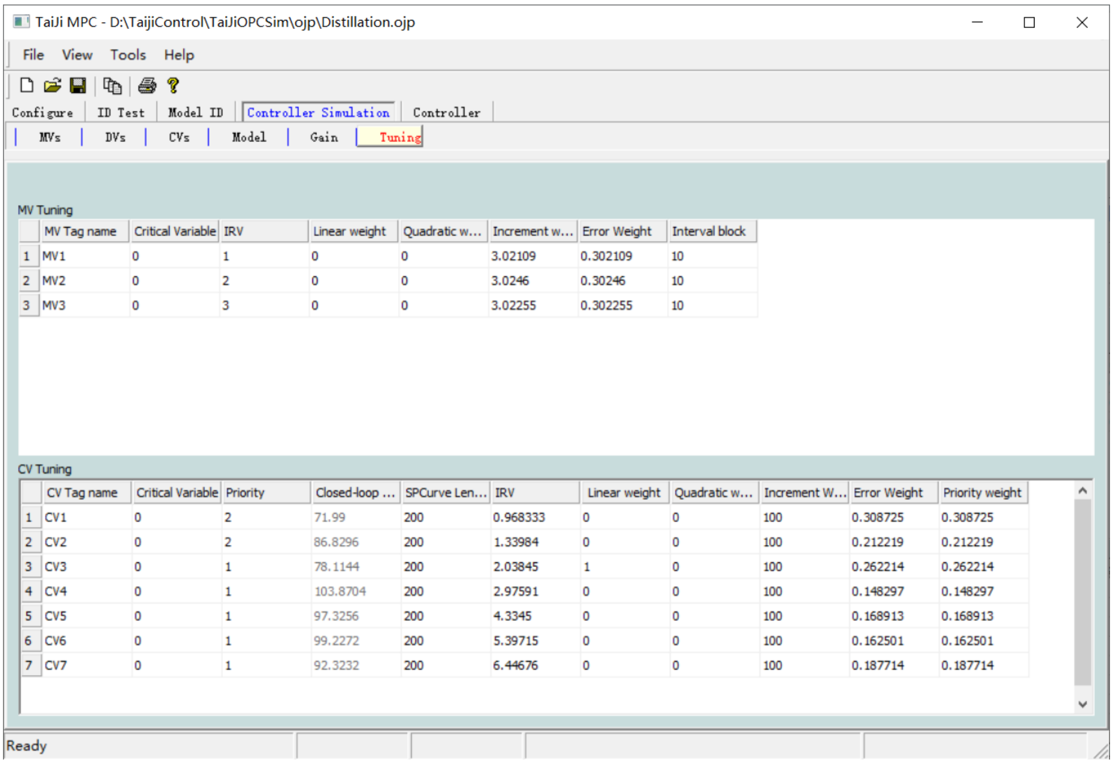
Generally, if the condition number is between 1 and 10, the corresponding 2 x 2 process is easy to control. If the condition number is around 100, the process is almost uncontrollable.

When the interaction between two CVs is strong, they should not be controlled independently. When setpoint control or small region control is achieved for one of the CVs, the control requirement for the other CV needs to be reduced, i.e., adopting a larger region control. When both CVs have high control requirements, it will result in very large MV actions, which may interfere with other CVs and potentially cause instability due to model errors.

Note: The gain can be modified in the model window opened by the simulation controller model. The singular values calculated here are for the normalized gain matrix (not shown here), not the original gain matrix.→

5 .4. 6 Tuning Window

→Click with the mouse on Simulation Controller Tuning to switch to the Tuning window as shown below:



Auto Tuning Button

If selected, the dynamic control parameters will be automatically tuned based on the identified model and MV/CV data, including MV increment weight, MV error weight, CV closed-loop response time, CV error weight, etc. Note: Auto-tuning cannot be used before model identification.

Use parameters in controller Button

If selected, all tuned parameters will be transferred to the **Controller** module.

MV Tuning Table

Control Variable (MV) Tag : The tag of the control variable.

Key Variable : If an MV is a key variable, the entire MPC controller stops when that MV is uncontrollable.

IRV : Ideal Value, economic optimization parameter, specified by the user.

Linear Weight : Linear programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Quadratic Weight : Quadratic programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Incremental Weight : Dynamic control parameter, can be set automatically, user can modify. The larger the number, the smaller the control effect.

Error Weight : Dynamic control parameter, can be set automatically, user can modify. The larger the number, the smaller the control effect.

Priority Weight : Economic optimization parameter, specified by the user. The larger the number, the more important it is.

Interval Blocks : Number of future MV sample points in quadratic programming for dynamic control, generally set to 10.

CV Tuning Table

Controlled Variable (CV) Tag : Tag of the controlled variable.

Critical Variable : If a CV is a critical variable, the entire MPC controller stops when this CV is uncontrollable.

Priority : The priority of this CV. Takes positive integers 1, 2, 3 ... , the smaller the value, the higher the priority. When the control of several CVs conflicts, the controller will abandon the lower priority CV and only control the higher priority CV.

Closed-loop response time : The closed-loop response time of this CV, **using the time unit as the number of samples (different from Tai-Ji MPC 3.x)** . Users cannot modify it. The shorter the closed-loop response time, the faster the CV control, and vice versa.

IRV : Ideal steady-state value, economic optimization parameter, specified by the user.

Upper limit margin : Dynamic control parameter, can be set automatically, and can be modified by the user.

Lower Bound Spacing : Dynamic control parameter, can be automatically set, user can modify.

Linear Weight : Linear programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Quadratic Weight : Quadratic programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Error Weight : Dynamic control parameter, can be set automatically, user can modify. The larger the number, the smaller the control effect.

Incremental Weight : Dynamic control parameter, can be automatically set, user can modify. The larger the number, the smoother the CV action.

Under the combined effect of the error weight and incremental weight of the CV, the equivalent closed-loop response time of the CV is approximately as follows:

$$\text{CV闭环响应时间} \approx 4\sqrt{\frac{\text{增量权重}}{\text{误差权重}}}$$

Priority Weight : Economic optimization parameter, specified by the user. The larger the number, the more important it is.

Interval Blocks : The number of future CV sample points in the quadratic programming of dynamic control, generally set to 10.

All dynamic control parameters can be automatically tuned, see Section 1.2.2 for details. Clicking the **Auto Tuning** button will automatically tune all dynamic control parameters based on the MV/CV data and the identified model.

In Tai-Ji MPC, the performance of dynamic control depends on the following tuning parameters:

- **CV Error Weight:** Increasing the CV error weight will cause the control algorithm to apply more control actions to that CV, reducing the controller's robustness. Decreasing the CV error weight will make the control of that CV more lenient but will improve the controller's robustness.
- **CV Increment Weight:** Increasing the CV increment weight will make the control algorithm apply a more smoothing effect to the CV, thereby increasing robustness to some extent; decreasing the CV increment weight will make the control of the CV more relaxed, potentially increasing the control speed of the CV but reducing robustness to some extent. Note: The setting of CV increment weight was introduced after Tai-Ji MPC 4.0, and the setting of CV closed-loop response time was canceled.
- **CV Closed-Loop Response Time:** Under the combined effect of CV error weight and CV increment weight, an equivalent CV closed-loop response time can be derived, meaning the CV approximately reaches the steady-state value along an equivalent first-order response trajectory.
- **MV Increment Weight:** Increasing the MV increment weight will slow down the control action of the algorithm on this MV, enhancing the robustness of the controller; decreasing the MV increment weight will speed up the control of this MV but reduce the robustness of the controller.

The tuning parameters of the simulation controller module and the **online controller** module are usually different. When good controller parameters are obtained in the simulation, the user can click the **Use parameters in controller** button to transfer the tuning parameters to the **online controller** module.

5.5 Online Controller Module

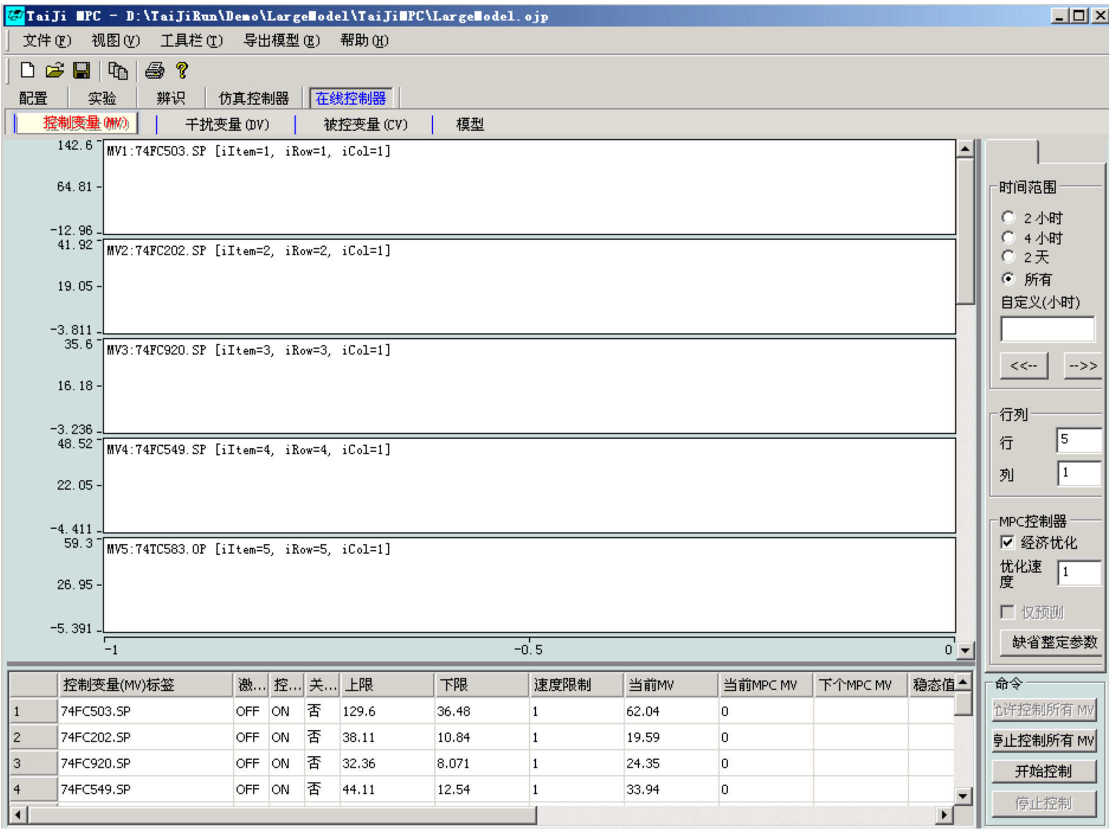
The online controller module uses the controller model identified in the model identification module for online control. It includes six windows as shown below, which are almost identical to the six windows of the simulation controller module.

MVs	**DVs**	**CVs**	**Model**	**Gain**	**Tuning**

Note: The online controller and the simulation controller use the same control model but different controller parameters.

5 .5.1 Control Variable (MV) Window

→Click with the mouse Online Controller Control Variable (MV) to switch to the Control Variable (MV) window as shown below:



Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

MPC Controller:

Economic Optimization : When selected, the controller starts the economic optimization program; otherwise, only dynamic control is performed.

Optimization Speed : The speed of economic optimization. A larger value indicates slower optimization speed. If set to 1, the optimization speed is the same as the dynamic control speed. The recommended optimization speed is between 3 and 6.

Warming up Box: When selected, the online controller will calculate the MV control action but will not write to the MV tag. This allows the predictor to initialize its parameters. If the controller is not warmed up, the online controller will experience a jump when put into operation.

Control Commands:

Allow control of all MVs : Press to allow the online controller to use the MVs.

Stop control of all MVs : Press to prohibit the online controller from using any MVs.

Start Control : Press to start online control.

Stop Control : Press to stop online control.

Meaning of MV Table Columns:

Control Variable (MV) Tag : The tag of the control variable.

Activation Flag : OFF, the online controller is not using this MV or there is a fault;
ON, the online controller is using this MV.

Control Command : OFF, the online controller does not use this MV;
ON, the online controller uses this MV.
Under normal circumstances, it is the same as the *activation flag* value (may differ by one sampling period).

Engineering Upper Limit: Engineering upper limit of MV

Upper *Limit* : The upper limit of the MV.

Lower Limit : The lower limit of the MV.

Engineering Lower Limit: The engineering lower limit of the MV.

Speed Limit : The upper limit of the MV increment per sampling period.

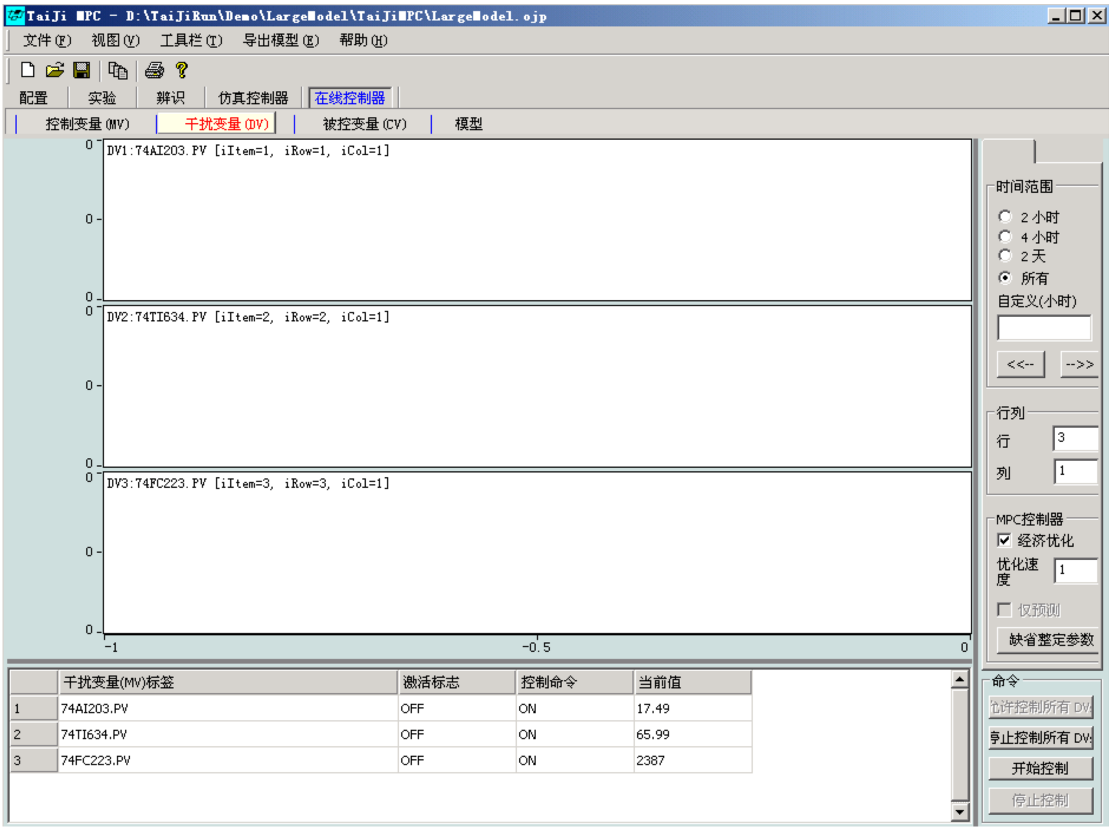
Current Value MPC MV : The MV value currently used by the simulation controller for calculation.

Next MPC MV : The MV value predicted by the simulation controller.

Steady State Value : The MV value when the system is in steady state, calculated by the MPC controller.

5 .5.2 Disturbance Variable (DV) Window

→Click with the mouse online controller disturbance variable (DV) to switch to the disturbance variable (DV) window as shown below:



Graph Display Commands (on the right side of the window):

→Same as the experimental disturbance variable (DV) MV window's graphical display command

Row and Column Area:

→Same as Experimental Disturbance Variable (DV) Row and Column Area in MV Window

MPC Controller:

→Same as Online Controller Manipulated Variable (MV) MPC Controller in MV Window

Control Commands:

Allow Control of All DVs : After pressing, the online controller is allowed to use DVs for feedforward control.

Stop Control of All DVs : After pressing, the online controller is prohibited from using DVs for feedforward control.

Start Control : Press to start online control.

Stop Control : Press to stop online control.

Meaning of DV Table Columns:

Disturbance Variable (DV) Tag : Tag of the disturbance variable.

Activation Flag : OFF, the online controller is not using this DV or it is faulty;

ON, the online controller is using this DV.

Control Command : OFF, the online controller does not use this DV;

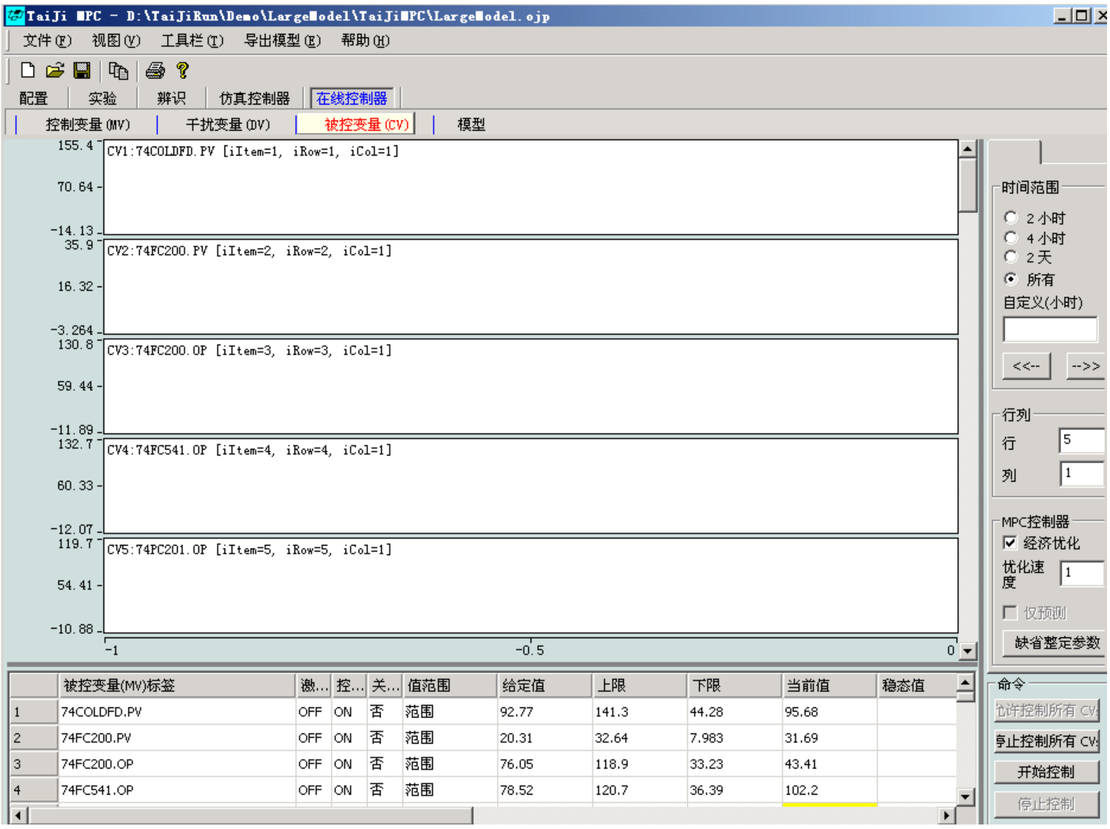
ON, the online controller uses this DV for feedforward control.

Under normal circumstances, it is the same as the *activation flag* value (may differ by one sampling period).

Current Value : The current value of the DV.

5 .5.3 Controlled Variable (CV) Window

Click with the mouse **online controller controlled variable (CV)→** to switch to the controlled variable (CV) window as shown below:



Graph Display Commands (on the right side of the window):

→Same as the Graphical Display Command in the MV window of Experimental Control Variable (MV)

Row and Column Area:

→Same as the Row and Column Area in the MV window of Experimental Control Variable (MV)

MPC Controller:

→Same as Online Controller Manipulated Variable (MV) MPC Controller in MV Window

Control Commands:

Allow control of all CVs : After pressing, the online controller is allowed to use the CVs.

Stop control of all CVs : After pressing, the online controller is prohibited from using any CV.

Start Control : Press to start online control.

Stop Control : Press to stop online control.

Meaning of CV table columns:

Controlled Variable (CV) Tag : Tag of the controlled variable.

Activation Flag : OFF, the online controller is not using this CV;
ON, the online controller is using this CV.

Control Command : OFF, the online controller does not use this CV;
ON, the online controller uses this CV.

Range/Setpoint : The control mode of the CV. Select 'Setpoint' to control the CV to a specific value.
Select 'Range' to control the CV within a range.

Setpoint : The setpoint of this CV

Engineering Upper Limit: Engineering upper limit of the CV

Upper Limit : The upper limit of this CV.

Lower Limit : The lower limit of this CV.

Engineering Lower Limit : The engineering lower limit of this CV

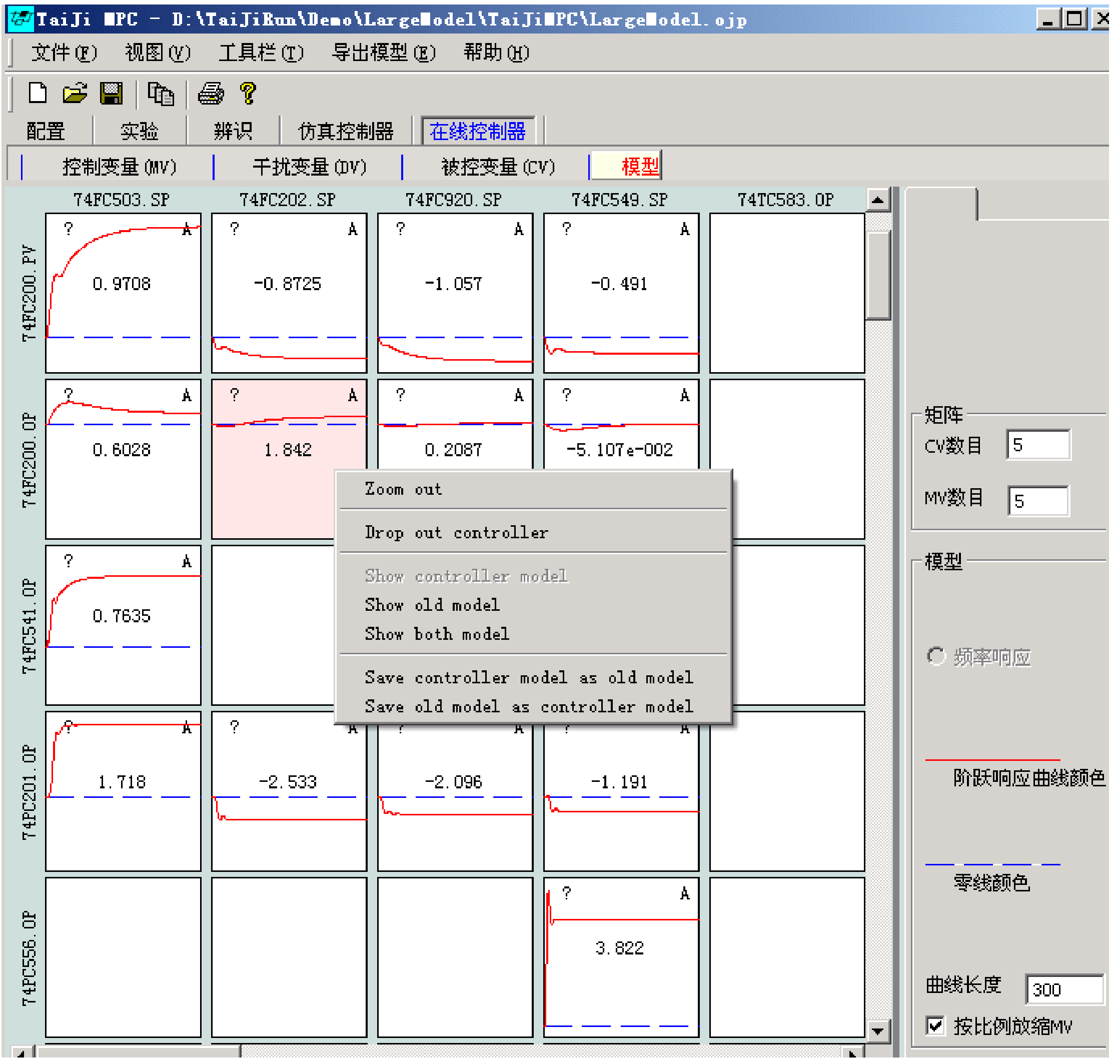
Current Value : The current value of this CV

Steady State Value : The steady state value of this CV calculated by the controller

Priority : The priority of this CV. It takes a positive integer, the smaller the value, the higher the priority. When the control of several CVs
conflicts, the controller will abandon the CV with lower priority and only control the CV with higher priority.

5 .5.4 Model Window

Click with the mouse **online controller model→** to switch to the **model** window as shown below:



Each step response plot also shows the desired matrix, the model's gain, and the model's quality grade. The corresponding values in the desired matrix are displayed in the upper left corner of the plot (+, -, ? or No) , the central value of the plot is the model's gain, and the upper right corner of the plot shows the model's quality grade A (Excellent), B (Good), C (Average), and D (Poor or No Model). Right-clicking on the model plot will present a menu bar that allows several operations such as copy/paste model, modify model, manually add model, etc.

Note: The control models under the Simulation Controller and Online Controller are the same, and the above operations on the model will also affect the model in the Online Controller module.

Row and Column Area:

CVs: Number of CVs displayed in the model response curve

MVs: Number of MVs displayed in the model response curve

Different model display methods:

Curve Length : Length of the model step and frequency response

Scale MV proportionally: If selected, the step and frequency response of the normalized model will be displayed.

Graphic element right-click menu:

Zoom out : Display all models; if there are too many models, they may not be displayed, and only the background color will be shown in the graphic,

At this point, you need to set the ***number of CVs*** and ***number of MVs*** to appropriate values.

Copy: Copy model

Paste: Paste model at the current position

Drop out controller : The MPC controller (simulation and online) does not use this model

Show controller model : Display controller model

Show old model : Display initial model (model management)

Show both model : Display both the controller model and the initial model simultaneously

Save controller model as old model : Save the controller model as the initial model, overwriting the initial model,

The operation is irreversible.

Save old model as controller model : Save the initial model as the controller model, overwriting the controller model,

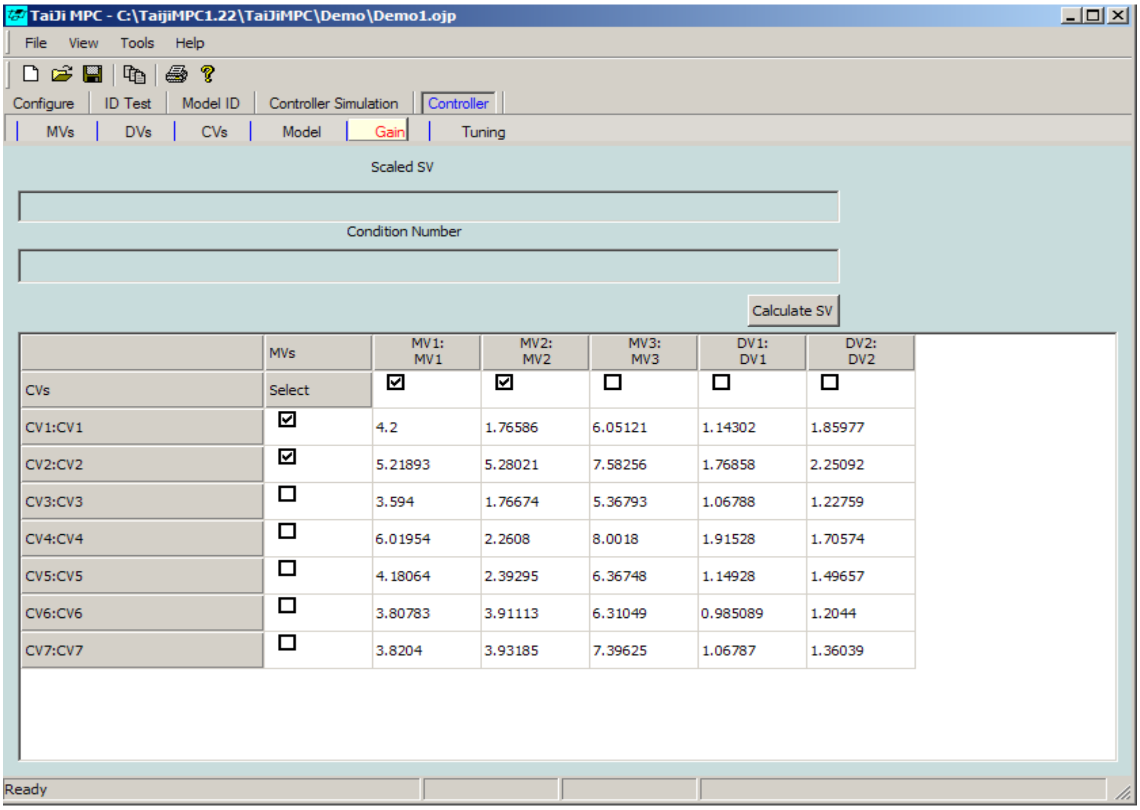
The operation is irreversible.

Modify controller model: Modify the current model. When selected, a new window will pop up displaying the model's step response. Users can specify the model's new gain and transition time and check the modified model.

Add/reset controller model: When selected, users can manually add a first-order or second-order model.

5 . 5 . 5 Gain Window

Click with the mouse **online controller gain→** to switch to the **gain** window as shown below:



The Gain window displays the gain matrix of the model. Users can select subsystems and use singular value decomposition techniques to analyze the controllability of the subsystem.

In the **Gain** window, select two CVs and two MVs, then click the **Calculate SV** button (SV stands for Singular Value). This will display the singular values and condition number (the ratio of the first singular value to the second singular value) of the selected 2 x 2 gain matrix.

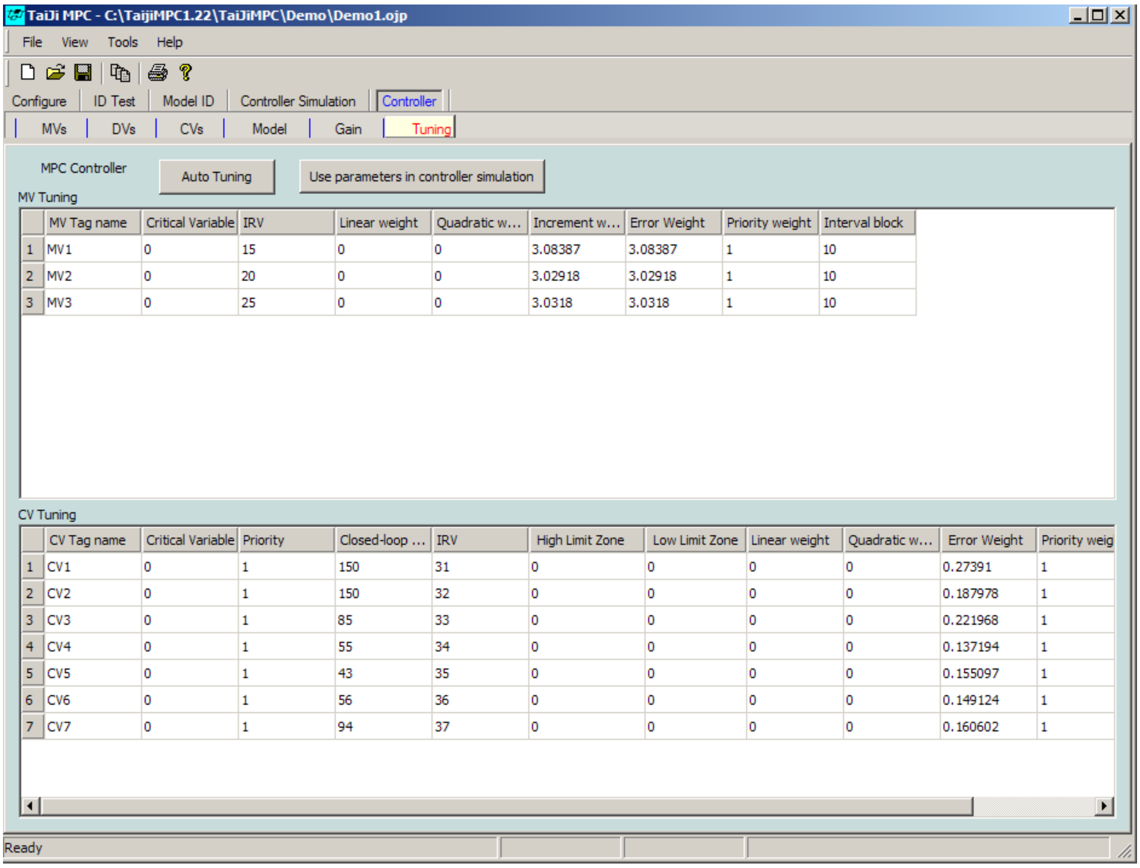
Generally, if the condition number is between 1 and 10, the corresponding 2 x 2 process is easy to control. If the condition number is around 100, the process is almost uncontrollable.

When the interaction between two CVs is strong, they should not be controlled independently. When setpoint control or small region control is achieved for one of the CVs, the control requirement for the other CV needs to be reduced, i.e., adopting a larger region control. When both CVs have high control requirements, it will result in very large MV actions, which may interfere with other CVs and potentially cause instability due to model errors.

Note: The gain can be modified in the model window opened by the **online controller model→** . Here, the singular values of the normalized gain matrix (not shown here) are calculated, not the singular values of the original gain matrix.

5 .5. 6 Tuning Window

→Click with the mouse on Online Controller Tuning to switch to the Tuning window as shown below:



Auto Tuning Button

If selected, the dynamic control parameters will be automatically tuned based on the identified model and MV/CV data, including MV increment weight, MV error weight, CV closed-loop response time, CV error weight, etc. Note: Auto-tuning cannot be used before model identification.

Use parameters in controller Button

If selected, all tuned parameters will be transferred to the **Controller** module.

MV Tuning Table

Control Variable (MV) Tag : The tag of the control variable.

Key Variable : If an MV is a key variable, the entire MPC controller stops when that MV is uncontrollable.

IRV : Ideal Value, economic optimization parameter, specified by the user.

Linear Weight : Linear programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Quadratic Weight : Quadratic programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Incremental Weight : Dynamic control parameter, can be set automatically, user can modify. The larger the number, the smaller the control effect.

Error Weight : Dynamic control parameter, can be set automatically, user can modify. The larger the number, the smaller the control effect.

Priority Weight : Economic optimization parameter, specified by the user. The larger the number, the more important it is.

Interval Blocks : Number of future MV sample points in quadratic programming for dynamic control, generally set to 10.

CV Tuning Table

Controlled Variable (CV) Tag : The tag of the controlled variable.

Critical Variable : If a CV is a critical variable, the entire MPC controller stops when this CV is uncontrollable.

Priority : The priority of this CV. Takes positive integers 1, 2, 3 ... , the smaller the value, the higher the priority. When the control of several CVs conflicts, the controller will abandon the lower priority CV and only control the higher priority CV.

Closed-loop response time : The closed-loop response time of this CV, **using the time unit as the number of samples (different from Tai-Ji MPC 3.x)** . Users cannot modify it. The shorter the closed-loop response time, the faster the CV control, and vice versa.

IRV : Ideal steady-state value, economic optimization parameter, specified by the user.

Upper limit margin : Dynamic control parameter, can be set automatically, and can be modified by the user.

Lower Bound Spacing : Dynamic control parameter, can be automatically set, user can modify.

Linear Weight : Linear programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Quadratic Weight : Quadratic programming weight, economic optimization parameter, specified by the user. The larger the number, the more important it is.

Error Weight : Dynamic control parameter, can be set automatically, user can modify. The larger the number, the smaller the control effect.

Incremental Weight : Dynamic control parameter, can be automatically set, user can modify. The larger the number, the smoother the CV action.

Under the combined effect of the error weight and incremental weight of the CV, the equivalent closed-loop response time of the CV is approximately as follows:

$$\text{CV闭环响应时间} \approx 4\sqrt{\frac{\text{增量权重}}{\text{误差权重}}}$$

Priority Weight : Economic optimization parameter, specified by the user. The larger the number, the more important it is.

Interval Blocks : The number of future CV sample points in the quadratic programming of dynamic control, generally set to 10.

All dynamic control parameters can be automatically tuned, see Section 1.2.2 for details. Clicking the **Auto Tuning** button will automatically tune all dynamic control parameters based on the MV/CV data and the identified model.

In Tai-Ji MPC, the performance of dynamic control depends on the following tuning parameters:

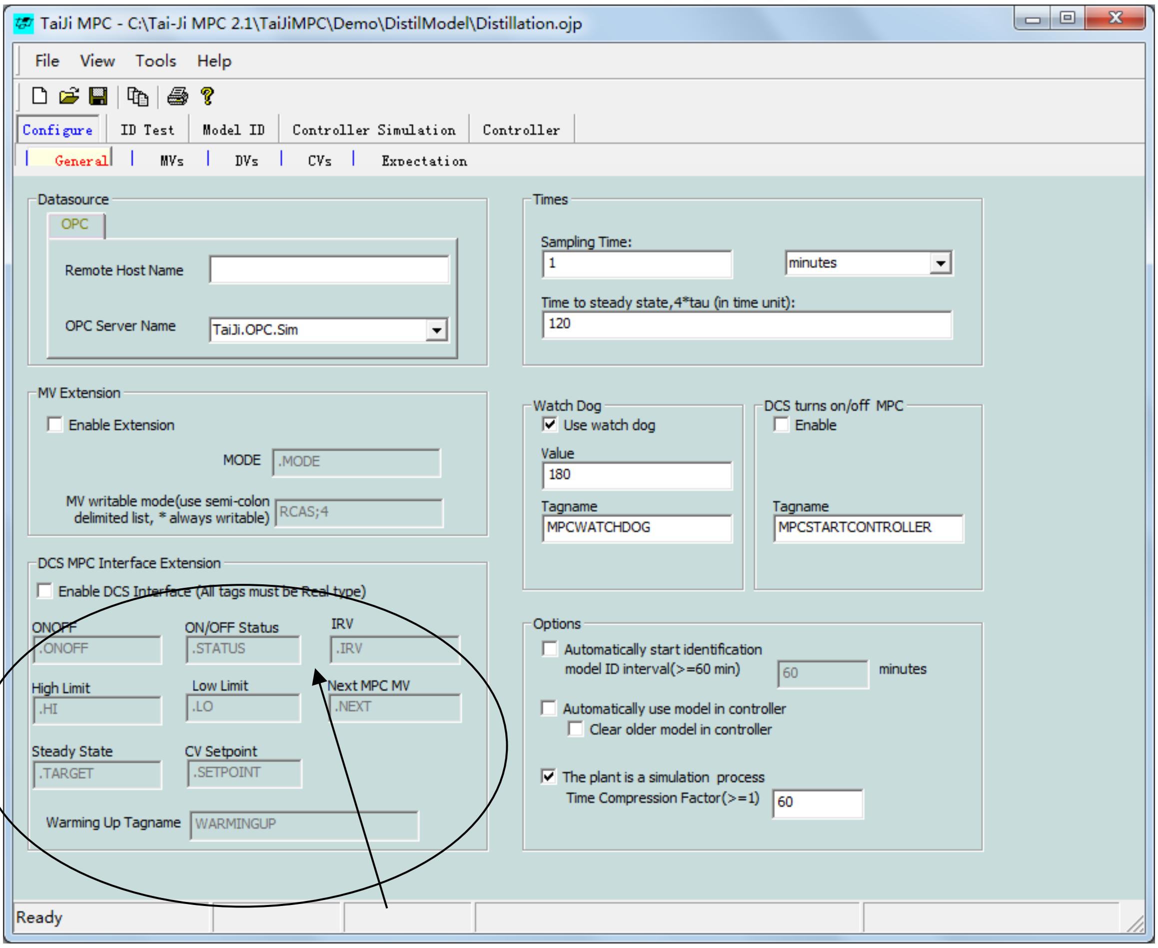
- **CV Error Weight:** Increasing the CV error weight will cause the control algorithm to apply more control actions to that CV, reducing the controller's robustness. Decreasing the CV error weight will make the control of that CV more lenient but will improve the controller's robustness.
- **CV Increment Weight:** Increasing the CV increment weight will make the control algorithm apply a more smoothing effect to the CV, thereby increasing robustness to some extent; decreasing the CV increment weight will make the control of the CV more relaxed, potentially increasing the control speed of the CV but reducing robustness to some extent. Note: The setting of CV increment weight was introduced after Tai-Ji MPC 4.0, and the setting of CV closed-loop response time was canceled.
- **CV Closed-Loop Response Time:** Under the combined effect of CV error weight and CV increment weight, an equivalent CV closed-loop response time can be derived, meaning the CV approximately reaches the steady-state value along an equivalent first-order response trajectory.
- **MV Increment Weight:** Increasing the MV increment weight will slow down the control action of the algorithm on this MV, enhancing the robustness of the controller; decreasing the MV increment weight will speed up the control of this MV but reduce the robustness of the controller.

The tuning parameters of the simulation controller module and the **online controller** module are usually different. When good controller parameters are obtained in the simulation, the user can click the **Use parameters in controller** button to transfer the tuning parameters to the **online controller** module.

6. DCS/PLC User Interface

The real-time controller can be operated using the DCS/PLC system. This is achieved through the DCS/PLC user interface. Below is the design of the interactive interface:

Tai-Ji MPC Configuration/General Window



In the lower left corner of the window, the user can define the DCS interface, with its settings given in the following three tables.

MV Screen

MPC Master Switch

DCS MV tag name	On/Off Status	ON/OFF	PID Mode	High Limit	Low Limit	IRV	Current Value	Steady State
MV1NAME	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET
MV2NAME	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET
MV3NAME	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET

MV4NAME	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET
MV5NAME	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET
„	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET
„	<MV>.STATUS	<MV>.ONOFF	<MV>.MODE	<MV>.HI	<MV>.LO	<MV>.IRV	<MV>	<MV>.TARGET

ON/OFF Status: If the MV is successfully enabled , the ON status will be displayed ; the data type is integer ; the data value is 0 (off) or 1 (on); PC writes to DCS; DCS screen displays ON (value is 1) or OFF (value is 0).

ON/OFF : MV ON/OFF switch; Data type is integer ; Data value is 0 (off) or 1 (on); PC read/write to DCS; DCS screen shows ON/OFF.

PID Mode : MV PID control mode; Data type is integer I4; Value is I4; PC reads DCS; DCS example: AUTO, MANUAL, LOCAL, REMOTE

High Limit : MV upper limit; Data type is R4; Real number; PC read/write to DCS; DCS displays value.

Low Limit : MV low limit; Data type is R4; Real; PC read/write to DCS; DCS displays value.

IRV : MV IRV value; Data type is R4; Real number; PC read/write to DCS; DCS displays value.

Current Value : MV current value; data type is R4; real number; PC read/write to DCS; DCS display value.

Steady State : MV steady state value; data type is R4; real number; PC writes to DCS; DCS display value.

DV Screen

DCS DV tagname	On/Off Status	On/Off	Current Value
DV1NAME	<DV>.STATUS	<DV>.ONOFF	<DV>
DV2NAME	<DV>.STATUS	<DV>.ONOFF	<DV>
„	<DV>.STATUS	<DV>.ONOFF	<DV>

ON/OFF Status: If DV is successfully on , the ON status will be displayed ; data type is integer; data value is 0 (off) or 1 (on); PC writes to DCS; DCS screen displays ON (value is 1) or OFF (value is 0).

ON/OFF : DV ON/OFF switch; data type is integer; data value is 0 (off) or 1 (on); PC read/write to DCS; DCS screen displays ON/OFF.

Current Value ： D V Current Value; Data type is R4; Real number; PC read/write to DCS; DCS display value.

CV Screen

MPC Master Switch

C V Tagname	On/Off Status	On/Off	High Limit	Low Limit	Current Value	IRV	Setpoint	Steady State
CV1NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
CV2NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET

CV3NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
CV4NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
CV5NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
CV6NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
CV7NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
CV8NAME	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
„	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET
„	<CV>.STATUS	<CV>.ONOFF	<CV>.HI	<CV>.LO	<CV>	<CV>.IRV	<CV>.SPT	<CV>.TARGET

ON/OFF Status： If CV is successfully ON , the ON status will be shown ; Data type is integer ; Data value is 0 (off) or 1 (on); PC writes to DCS; DCS screen shows ON (value is 1) or OFF (value is 0).

ON/OFF ： CV ON/OFF switch; Data type is integer ; Data value is 0 (off) or 1 (on); PC read/write to DCS; DCS screen shows ON/OFF.

High Limit ： CV upper limit; Data type is R4; Real number; PC read/write to DCS; DCS display value.

Low Limit : CV low limit; Data type is R4; Real; PC read/write to DCS; DCS display value.

IRV : CV IRV value; Data type is R4; Real; PC read/write to DCS; DCS display value.

Current Value : CV current value; Data type is R4; Real; PC read/write to DCS; DCS display value.

Steady State : CV steady state value; Data type is R4; Real; PC write to DCS; DCS display value.

7. Script Processing

Scripts are used to add actions or calculations that are not supported in conventional MPC control. The purpose of executing scripts is to increase the flexibility and intelligence of MPC control. There are three types of scripts as follows:

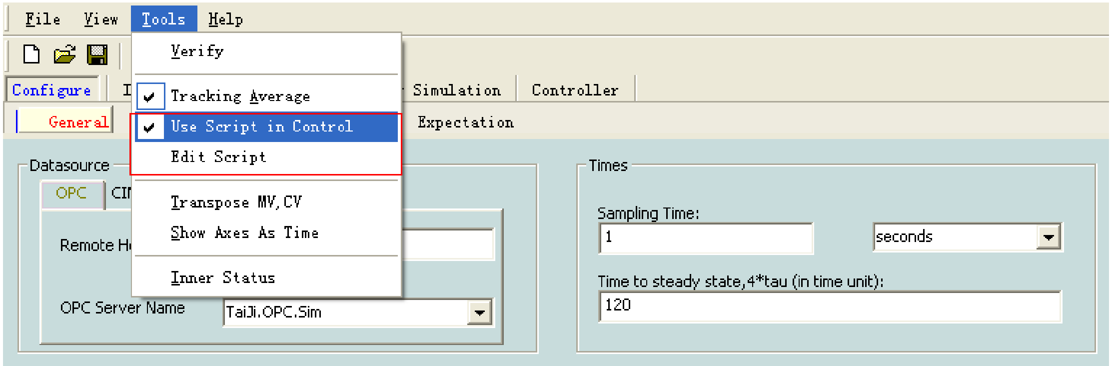
（1） **Initialization Script** , actions / calculations will start after the controller begins execution (when the “Start Control” button is pressed)

（2） Input Script , in each sampling period, actions / calculations will be implemented before the MPC control calculation , the input script will be executed in each sampling period.

（3） Output Script , in each sampling period, actions / calculations will be implemented after the MPC control calculation, the input script will be executed in each sampling period.

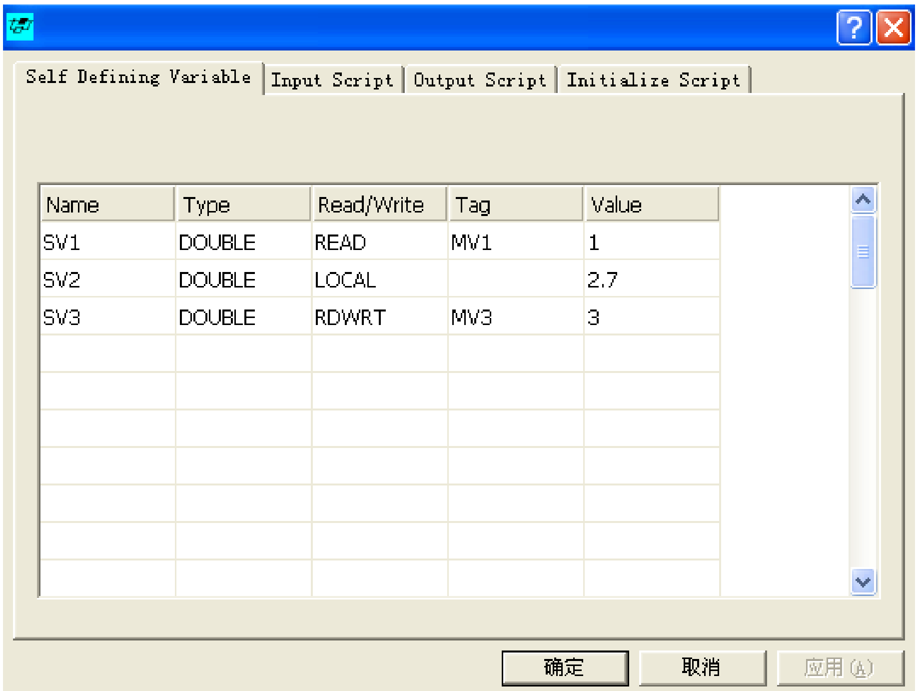
7.1 Enabling Script Functionality

In the Tools menu , click the Use Script in Control option , as shown below:

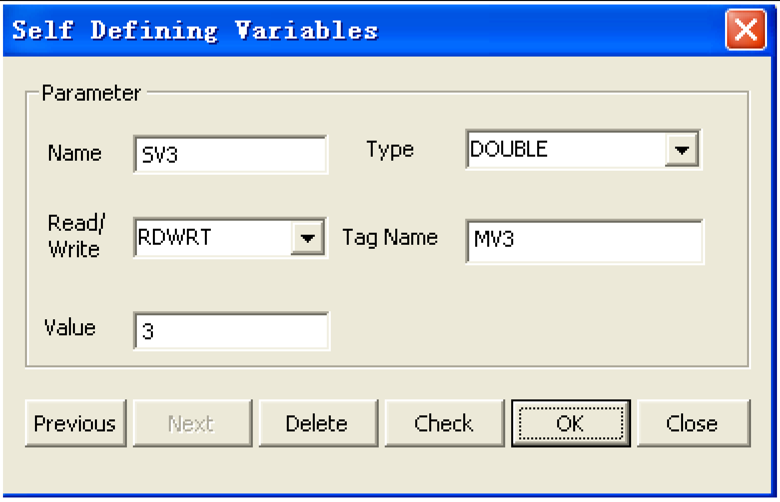


7.2 Edit Script

In the Tools menu , click the Edit Script button to open the script editing window, where users can define variables (click the Self Defining Variables tab), edit the input script (click the Input Script tab), edit the output script (click the Output Script tab), and edit the initialization script (click the Initialize Script tab).



In the **Self Defining Variable** tab, double-click the Name column of the module, and the following define variable window will appear, where users can enter the variable name, data type, read /write , DCS (OPC server) tag name , and optional value.



Name: Variable Name

Type: Type of the value DOUBLE or LONG

Reaf/Write: LOCAL, READ, WRITE, RDWRT.

LOCAL indicates local variable, does not communicate with DCS

READ indicates reading from DCS

WRITE indicates writing to DCS

RDWRT indicates reading/writing from/to DCS

Tag Name: Tag name in DCS (LOCAL variables cannot be used)

Value: Value of LOCAL variable (not used for READ, WRITE, or RDWRT).

The defined variable class is called the SV class.

7.3 Edit Input / Output Script

Users can use VB language to create and edit input scripts in the input script , and can use GetValue to obtain MV/DV/CV/SV variables , SetValue can set their values .

Example : The tag name of MV in DCS is `FIC1001.SP`

Use the following command to get the upper and lower limits of MV :

```
MVhi = GetValue(“MV:FIC1001.SP-SoftHighLimit”)
MVlo = GetValue(“MV:FIC1001.SP-SoftLowLimit”)
```

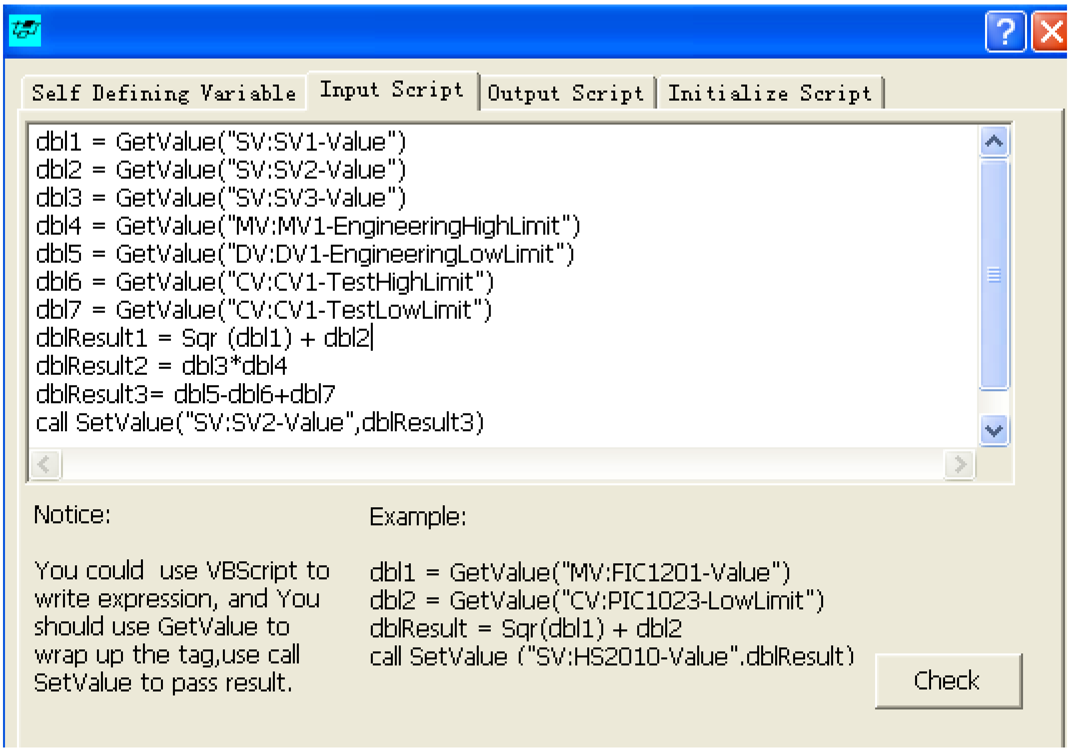
Use the following command to get the median value and store it in MV :

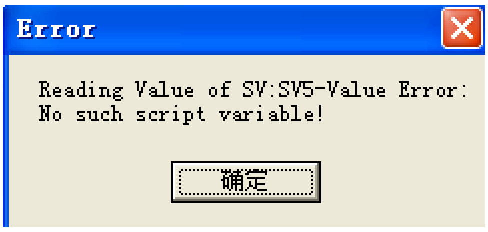
```
MVmed = 0.5*(MVhi + MVlo)
call SetValue(“MV:FIC1001.SP-MPCPredictionMV”,MVmed)
```

Note that when reading from or writing to MV/DV/CV/SV variables, you need to use their corresponding attribute values. The attribute values of MV/DV/CV/SV variables are marked by an extension symbol “-”. For example : The upper limit of MV FIC1001.SP is: FIC1001.SP-SoftHighLimit .

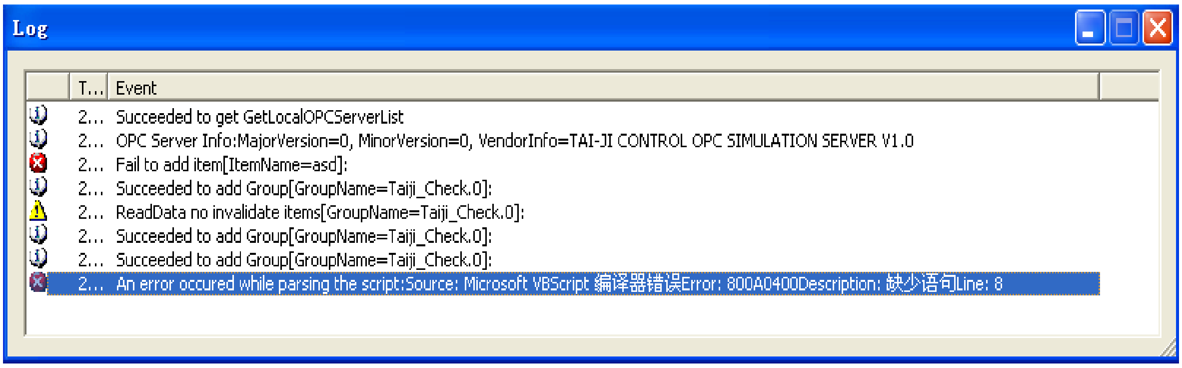
The attributes of MV/DV/CV/SV are given at the end of this chapter .

The figure below shows the input script. Click the button to create the script. If there are errors, an error window will pop up, and the error message will indicate which variable cannot be read with GetValue, and which variable cannot be written with call SetValue.





If there is a logical error, VBScript will detect it and provide error information in the log window as follows:



Output scripts and initialization scripts can be created in a similar manner.

Note when using scripts, the user cannot run the controller simulation and control at the same time.

7.4 Variable Attributes

CurrentProcessMV (MV measured in DCS)
EngineeringHighLimit
EngineeringLowLimit

MPCPredictionMV (MV control value to be written into DCS)
ControlStatus
CriticalVariable
SoftHighLimit (MV upper limit)
SoftLowLimit (MV lower limit)
SpeedLimit
IRVvalue
LinearWeight
QuadraticWeight
SpeedWeight
ErrorWeight
PriorityWeight
IntervalBlock
TestHighLimit
TestLowLimit
CurrentMPCMV
SteadyValue

SimulationMPCPredictionMV (same as above, but for control simulation)
SimulationControlStatus (same as above, but for control simulation)
SimulationCriticalVariable (same as above, but for control simulation)
SimulationSoftHighLimit (Same as above, but for control simulation)
SimulationSoftLowLimit (Same as above, but for control simulation)
SimulationSpeedLimit (Same as above, but for control simulation)

SimulationIRVvalue (Same as above, but for control simulation)
SimulationLinearWeight (Same as above, but for control simulation)
SimulationQuadraticWeight (Same as above, but for control simulation)
SimulationSpeedWeight (Same as above, but for control simulation)
SimulationErrorWeight (Same as above, but for control simulation)
SimulationPriorityWeight (Same as above, but for control simulation)
SimulationIntervalBlock (Same as above, but for control simulation)
SimulationTestHighLimit (Same as above, but for control simulation)
SimulationTestLowLimit (Same as above, but for control simulation)
SimulationCurrentMPCMV (Same as above, but used for control simulation)
SimulationSteadyValue (Same as above, but used for control simulation)

DV attributes

EngineeringHighLimit
EngineeringLowLimit
CurrentDV
ControlStatus
SimulationCurrentDV(Same as above, but used for control simulation)
SimulationControlStatus (same as above, but for control simulation)

CV attributes

TestHighLimit
TestLowLimit
Integral
MVVALV
CVLGTH
Calculated
EngineeringHighLimit
EngineeringLowLimit

CurrentCV (CV measured in DCS)
SteadyValue
ActiveIndicator
ControlStatus
CriticalVariable
SetpointFixed
SetpointValue
SetpointHighLimit
SetpointLowLimit
SoftHighLimit(CV upper limit)

SoftLowLimit(CV lower limit)
Priority
ClsLpRsponseTime
IRVvalue
LinearWeight
QuadraticWeight
ErrorWeight
PriorityWeight
IntervalBlock

SimulationCurrentCV (Same as above, but used for control simulation)
SimulationSteadyValue (Same as above, but used for control simulation)
SimulationActiveIndicator(Same as above, but used for control simulation)
SimulationControlStatus (same as above, but for control simulation)
SimulationCriticalVariable (same as above, but for control simulation)
SimulationSetpointFixed(Same as above, but used for control simulation)
SimulationSetpointValue(Same as above, but used for control simulation)
SimulationSetpointHighLimit(Same as above, but used for control simulation)
SimulationSetpointLowLimit(Same as above, but used for control simulation)
SimulationSoftHighLimit (Same as above, but for control simulation)
SimulationSoftLowLimit (Same as above, but for control simulation)
SimulationPriority (Same as above, but used for control simulation)
SimulationClsLpRsponseTime (Same as above, but used for control simulation)
SimulationIRVvalue (Same as above, but for control simulation)
SimulationLinearWeight (Same as above, but for control simulation)
SimulationQuadraticWeight (Same as above, but used for control simulation)
SimulationErrorWeight (Same as above, but for control simulation)
SimulationPriorityWeight (Same as above, but for control simulation)
SimulationIntervalBlock (Same as above, but for control simulation)

SV attribute

Value.

8. Troubleshooting

1. When starting the program, the prompt appears: Error 7: Hasp HL Key not found.

The encryption key is not inserted, or the inserted key is invalid. Refer to Section 2.4 Tai-Ji MPC Software Encryption Protection System and Update Process .

2. After starting the program, when opening or creating a new project, the program closes directly.

The environment Path is not set correctly. Copy all files from C:\Taiji\Common\lib to the executable file TaiJiMPC.exe located in the directory C:\Taiji\TaiJiMPC\bin .

3. In the **configuration general→** window, you cannot see the installed OPC servers

Check if OPC DA Components are installed. If not, please run Tai-Ji MPC from the CD under OPC_DA20_Components.exe to install.

Check if OPC enum is disabled. If so, you can ask the network administrator to run dcomcnfg to allow OPC enum permissions, and then run Tai-Ji MPC again;

Directly fill in a valid OPC Server name in the **OPC service name** field. If you still cannot access the OPC Server, please run dcomcnfg on both the client and server to confirm if you have access permissions to the OPC Server.

If it is a remote OPC server, check if the network connection is valid. You can use the PING command to check, ping *host name.* On some machines, the PING command is disabled. If so, please consult the network administrator.

4. In the **model identification model→** opened model window, the model step response or model frequency response only has background color

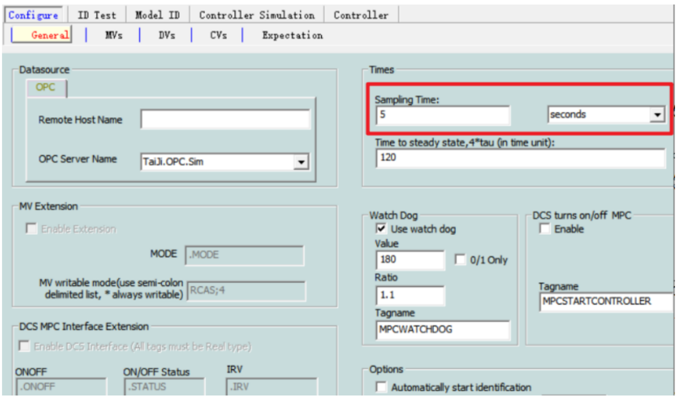
This is because there are too many graphs to display in the window, making it impossible to correctly display each model. You need to reduce the number of rows and columns or the number of CVs and MVs.

9. Frequently Asked Questions

9 . 1 About setting the sampling time in Tai-Ji MPC

Q1 What is the sampling time? Where is the sampling time set?

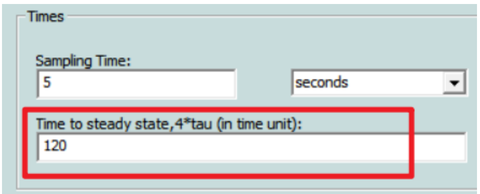
A1 The sampling time in Tai-Ji MPC refers to the execution period of the MPC controller. The sampling time is set in the configuration interface of Tai-Ji MPC, where you can choose 'seconds' or 'minutes' as the time unit. The minimum sampling time is 1 second.



Q2 How to fill in the steady-state time, and what is its relationship with the sampling time?

A2 The steady-state time refers to the time it takes for the controlled object to reach open-loop stability after the main input changes. Please fill in according to the time unit. For example: if the time for a certain heating furnace to stabilize after a change in coal feed is about 20 minutes, and the time unit is 'seconds', then the open-loop steady-state time in the Tai-Ji MPC software should be filled in as 1200.

The filling of the open-loop steady-state time has no relationship with the sampling time, but it is related to the time unit of the sampling time.



Q3 How to set the sampling time?

A3 Please set the sampling time according to the execution cycle of the MPC controller. The execution cycle of the MPC controller refers to how often the user wants the Tai-Ji MPC to calculate and output the MV. After setting the sampling time, the online controller calculates and outputs the MV according to the set sampling time, and data acquisition is also performed according to the set sampling time. For example: if the sampling time is set to 10 seconds in Tai-Ji MPC, then the online controller outputs the MV every 10 seconds, and the experimental interface collects data every 10 seconds after starting data collection (the collected data is at 10-second intervals).

Q4 I want to import CSV data for identification. The data in the CSV is sampled at 1-second intervals, but the sampling time set in Tai-Ji MPC is 10 seconds. How should I handle this?

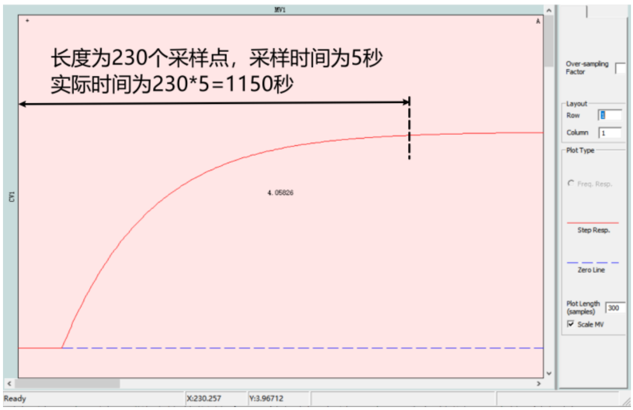
A4 Use external tools like Excel to open the 1-second CSV data and downsample it to 10 seconds, creating a 10-second CSV data file. Import the 10-second CSV data file into Tai-Ji MPC. **<u>The sampling time filled in the configuration interface must match the actual sampling time of the imported data.</u>**

Q5 The sampling time of Tai-Ji MPC is 10 seconds. How do I manually input a first-order model with a gain of 2.0, a delay of 30 seconds, and a response time of 240 seconds in the online controller? Are the manually input model parameters related to the sampling time? Are they related to the time units?

A5 According to engineering estimation, tao≈240/4=60. When the software's time unit is set to 'seconds', please enter 2.0/(60*s+1)*exp(-30*s). Since the manually entered model is a continuous-time model, the manually entered model parameters are only related to the time unit and not to the sampling time. Based on the given parameters, when the software's time unit is set to 'minutes', please enter 2.0/(1*s+1)*exp(-0.5*s).

Q6 Is the step response of the model drawn in MPC based on sampling points (Sample) or absolute time (Time)?

A6 Based on sampling points (Sample). For example, in an MPC project with a sampling time set to 5 seconds, the model is shown as in the figure below. Observing from the figure, the model reaches a steady state at approximately 230 sampling points, which means the steady-state time of the model is 230*5=1150 seconds. Additionally, the numbers displayed or entered in the 'Delay' interface under 'Identification' are also based on sampling points.

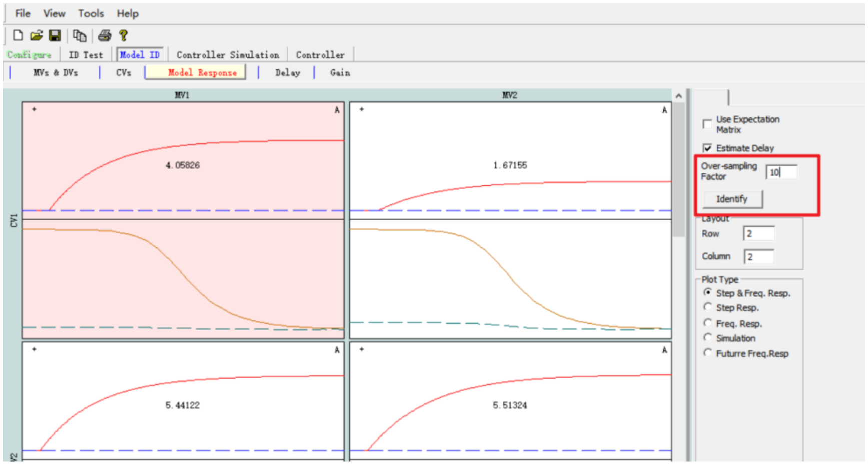


Q7 In the MPC controller table, does the closed-loop response time refer to the sampling point (Sample) or absolute time (Time)?

A7 It varies with different versions of Tai-Ji MPC. The interface will indicate this. If it is labeled as Time, it refers to absolute time; if labeled as Sample, it refers to sampling points. Tai-Ji MPC 4.x and Tai-Ji MPC 5.x use sampling points (Samples), while Tai-Ji MPC 3.x uses absolute time.

Q8 The sampling time is 1 second (the actual sampling time of the imported/tested data is also 1 second). I want to identify using a 10-second sampling interval. How should I proceed?

A8 Enter 10 (times) in the slow sampling multiplier in the 'Identification->Model' interface and then click identify to generate the model, as shown in the figure below.



Q9 The sampling time is 10 seconds (the actual sampling time of the imported/tested data is also 10 seconds). I want to identify a model suitable for a 1-second sampling time. How should I proceed?

A9 It is not possible. You can create a new project and set the sampling time to 1 second, then import data with an actual sampling time of 1 second.

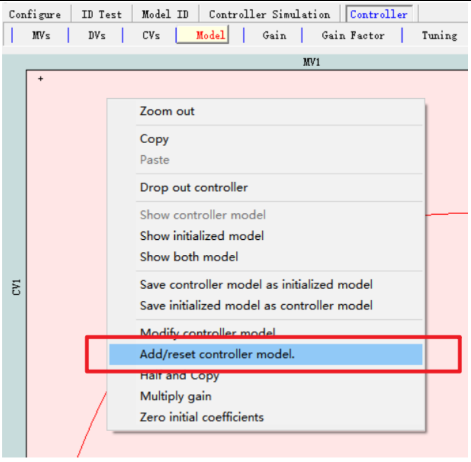
Q10 Can the sampling time be changed?

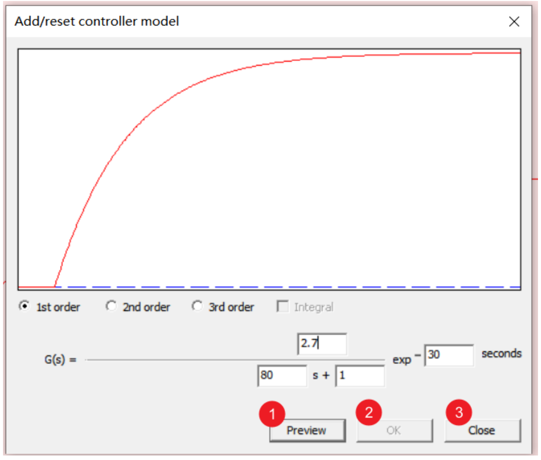
A10 Yes (the controller and testing need to be stopped). Be very careful when performing this operation! Please read the next two Q&As carefully.

Since the set sampling time must match the actual sampling time of the data, changing the sampling time may result in incorrect data and model. It is recommended to clear the data and identification model after changing the sampling time.

Q11 I only use the online controller function, and all models are manually input. I want to modify the sampling time. How should I proceed?

A11 To modify the sampling time, find the model of the online controller, right-click on each one -> add/reset controller model -> Preview -> OK -> Close, as shown in the figure below. The above operation is equivalent to re-entering the controller model.





Q12 I only use the online controller function. Some models are imported from the identification interface to the controller, and some models are manually input. I want to modify the sampling time. How should I proceed?

A12 This operation cannot be performed. Reason: The identification model cannot be reset.