

# MEM6810 工程系统建模与仿真

案例 软件

## 第四讲: FlexSim 软件应用基础

沈海辉

中美物流研究院  
上海交通大学

🏠 [shenhaihui.github.io/teaching/mem6810p](https://shenhaihui.github.io/teaching/mem6810p)  
✉ [shenhaihui@sjtu.edu.cn](mailto:shenhaihui@sjtu.edu.cn)

2024年春 (MEM非全日制)



上海交通大学  
SHANGHAI JIAO TONG UNIVERSITY

董浩云智能制造与服务管理研究院  
CY TUNG Institute of Intelligent Manufacturing and Service Management  
(中美物流研究院)  
(Sino-US Global Logistics Institute)



## 1 FlexSim 概述

- ▶ FlexSim 特点
- ▶ FlexSim 仿真环境
- ▶ FlexSim 基本操作

## 2 FlexSim 基础教程

- ▶ 建立第一个三维模型
- ▶ 显示仿真结果
- ▶ 建立第一个过程流模型
- ▶ 过程流和三维模型结合

## 3 FlexSim 进阶教程

- ▶ 任务逻辑教程
- ▶ 自动导引运输车 (AGV)
- ▶ 传送带
- ▶ 分布拟合工具 (ExpertFit)
- ▶ 实验工具 (Experimenter)
- ▶ 优化工具 (Optimizer)



## 1 FlexSim 概述

- ▶ FlexSim 特点
- ▶ FlexSim 仿真环境
- ▶ FlexSim 基本操作

## 2 FlexSim 基础教程

- ▶ 建立第一个三维模型
- ▶ 显示仿真结果
- ▶ 建立第一个过程流模型
- ▶ 过程流和三维模型结合

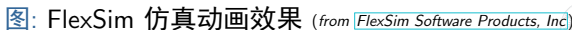
## 3 FlexSim 进阶教程

- ▶ 任务逻辑教程
- ▶ 自动导引运输车 (AGV)
- ▶ 传送带
- ▶ 分布拟合工具 (ExpertFit)
- ▶ 实验工具 (Experimenter)
- ▶ 优化工具 (Optimizer)



# FlexSim 概述

- FlexSim 是一款离散事件仿真软件，它由 FlexSim Software Products, Inc. 开发。  <https://www.flexsim.com>
- 它目前包括了一个通用的建模环境和一个医疗系统建模环境。

 图: FlexSim 仿真动画效果 (from [FlexSim Software Products, Inc](https://www.flexsim.com))



上海交通大学  
SHANGHAI JIAO TONG UNIVERSITY

- 基于面向对象技术的建模
  - FlexSim 使用的是面向对象的建模逻辑
  - 对象可以重复利用, 减少重复劳动
- 突出的 3D 图形显示功能
  - FlexSim 使用的三维图形引擎是游戏中广泛使用的 OpenGL
  - 支持 3ds、wrl、dxf 和 stl 等文件格式
- 建模和调试方便
  - 基础的建模不需要编写程序, 通过拖拽的方式构建出模型
  - 进阶功能可通过 C++ 代码编辑实现
- 扩展性强
  - 支持用户自定义对象, 构建自己的对象库
  - 可直接通过 C++ 代码编辑, 还可以调用 DLL 文件, 拥有 Module SDK 和 API 接口, 二次开发功能强大
- 开放性好
  - 可通过 ODBC 与外部数据库相连
  - 通过 socket 接口与外部硬件设备相连, 与 Excel、Visio 等软件配合使用

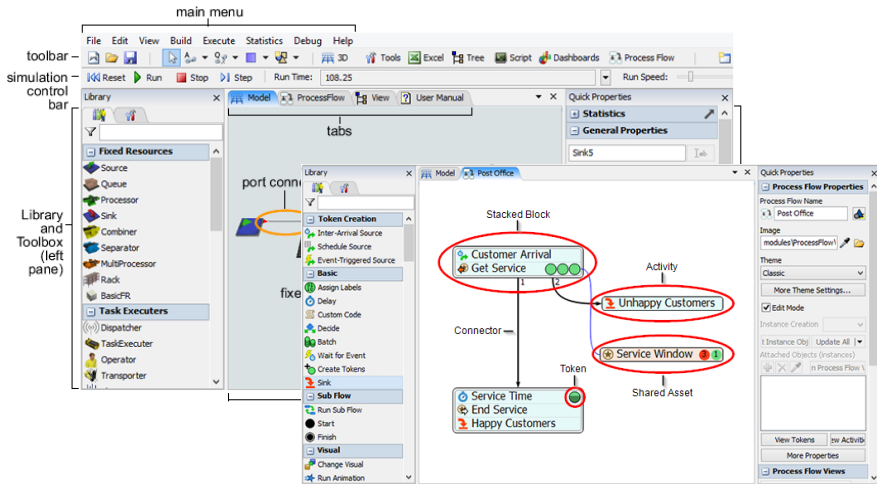


- FlexSim 目前仅支持 Windows 系统, 能够完美地支持 Windows 10.
- Mac 系统需安装 Windows 虚拟机 (推荐 Parallels <https://www.parallels.cn>).
- 推荐配置

<b>CPU</b>	A CPU scoring <b>15,000 or better</b> in <u>PassMark - CPU Mark multi-thread</u> benchmark, and <b>2,000 or better</b> in the <u>single-thread</u> benchmark, such as: <ul style="list-style-type: none"><li>• <u>Intel Core i7-8700</u> or higher</li><li>• <u>AMD Ryzen 7 Pro 2700</u> or higher</li></ul>
<b>RAM</b>	<b>16 GB RAM or more</b>
<b>Graphics</b>	A GPU scoring about <b>10,000 or better</b> in <u>PassMark - G3D Mark</u> benchmark, such as: <ul style="list-style-type: none"><li>• <u>NVIDIA GeForce GTX 1660</u> or higher</li><li>• <u>AMD Radeon RX 590</u> or higher</li></ul>
<b>OS</b>	One of the <b>last two</b> versions of Microsoft Windows 10, 64-bit. (See the <u>Windows 10 version history</u> table).

- 试用版 (FlexSim Express) 的功能限制:
  - 模型的大小受限;
  - 统计的工具不可用: 如,
    - 分布拟合工具 (ExpertFit)
    - 实验工具 (Experimenter tool)
    - 优化工具 (Optimizer tool, OptQuest)
  - 随机数发生器是固定的
  - 其他一些菜单选项和工具不可用
- 购买序列号激活之后可使用全部功能.

- 基本界面：三维模型界面、过程流 (Process Flow) 界面





- 新建模型时设定单位

- 视图控制

- 平移视角: 鼠标左键拖动
- 旋转视角: 鼠标右键拖动
- 缩放视角: 鼠标滚轮, 或同时按住左右键

- 实体操控

- 黄选 (highlight): 单击选中一个; 单击空白处取消
- 红选 (select): Shift 或 Ctrl 加单击, 可选中多个; Shift 加单击空白处取消
- 沿 X 轴和 Y 轴平移: 选中后拖动
- 沿 Z 轴平移: 选中后滚动滚轮

- 实体连接

- A 连接
- S 连接

- 将实体对象从对象库中拖拽到三维模型中
  
  
  
  
  
  
  
  
  
  
- 连接实体对象

## 1 FlexSim 概述

- ▶ FlexSim 特点
- ▶ FlexSim 仿真环境
- ▶ FlexSim 基本操作

## 2 FlexSim 基础教程

- ▶ 建立第一个三维模型
- ▶ 显示仿真结果
- ▶ 建立第一个过程流模型
- ▶ 过程流和三维模型结合

## 3 FlexSim 进阶教程

- ▶ 任务逻辑教程
- ▶ 自动导引运输车 (AGV)
- ▶ 传送带
- ▶ 分布拟合工具 (ExpertFit)
- ▶ 实验工具 (Experimenter)
- ▶ 优化工具 (Optimizer)



- 单服务台排队系统, 当排队时间达到一定值时顾客会离开.

- 教程: User Manual / Tutorials / FlexSim Basics Tutorials / 1.1 - Build a 3D Model
- 模型: First3DModel.fsm

- 动态显示队伍长度、平均等待时间等结果.

- 教程: User Manual / Tutorials / FlexSim Basics Tutorials / 1.2 - Get Data from the 3D Model
- 模型: GetDataFrom3DModel.fsm



- 同一个问题, 采用过程流工具 (而非三维建模工具) 来实现, 更抽象但是更灵活.
  
- 教程: User Manual / Tutorials / FlexSim Basics Tutorials / 1.3 - Build a Process Flow Model
- 模型: `FirstProcessFlowModel.fsm`





## 1 FlexSim 概述

- ▶ FlexSim 特点
- ▶ FlexSim 仿真环境
- ▶ FlexSim 基本操作

## 2 FlexSim 基础教程

- ▶ 建立第一个三维模型
- ▶ 显示仿真结果
- ▶ 建立第一个过程流模型
- ▶ 过程流和三维模型结合

## 3 FlexSim 进阶教程

- ▶ 任务逻辑教程
- ▶ 自动导引运输车 (AGV)
- ▶ 传送带
- ▶ 分布拟合工具 (ExpertFit)
- ▶ 实验工具 (Experimenter)
- ▶ 优化工具 (Optimizer)



- 任务逻辑工具

- 使用三维模型逻辑

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 1 - Task Logic Tools

- Tutorial / 1.1 - Tasks Using Standard 3D Logic

- 模型: `TasksUsing3DModel.fsm`

- 使用过程流

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 1 - Task Logic Tools

- Tutorial / 1.2 - Tasks Using Process Flow

- 模型: `TasksUsingProcessFlow.fsm`

- 使用列表 (List)

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 1 - Task Logic Tools

- Tutorial / 1.3 - Tasks Using Lists

- 模型: `TasksUsingLists.fsm`



- 协调任务

- 标准装载任务

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 2 - Coordinated Tasks Tutorial / 2.1 - Create Standard Loading Tasks

- 模型: `StandardLoadingTasks.fsm`

- 协调装载任务

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 2 - Coordinated Tasks Tutorial / 2.2 - Create Coordinated Loading Tasks

- 模型: `CoordinatedLoadingTasks-incomplete.fsm`



- 条件任务

- 使用子过程流 (Sub Flows) 和数组 (Arrays)

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 3 - Conditional Tasks  
Tutorial / 3.1 - Use Subflows and Arrays

- 模型: UseSubFlowsAndArrays.fsm

- 添加条件任务

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 3 - Conditional Tasks  
Tutorial / 3.2 - Add Conditional Tasks

- 模型: AddConditionalTasks.fsm



- 自动导引运输车 (AGV)

- 使用三维模型逻辑

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 4 - Automatic Guided Vehicles (AGVs) / 4.1 - AGVs Using Standard 3D Logic

- 模型: AGVsUsing3DModel.fsm

- 使用过程流

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 4 - Automatic Guided Vehicles (AGVs) / 4.2 - AGVs Using Process Flow

- 使用电梯和 AGVs

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 4 - Automatic Guided Vehicles (AGVs) / 4.3 - Using Elevators With AGVs

- 自定义 AGV 设置

- 教程: User Manual / Tutorials / Task Logic Tutorials / Tutorial 4 - Automatic Guided Vehicles (AGVs) / 4.4 - Custom AGV Settings

- 传送带

- 拣选系统

- 教程: User Manual / Tutorials / Additional Tools Tutorials / Tutorial 1 - Conveyors / 1.1 - Sorting Systems

- 模型: `SortingSystems.fsm`, `SortingSystems2.fsm`, `SortingSystems3.fsm`

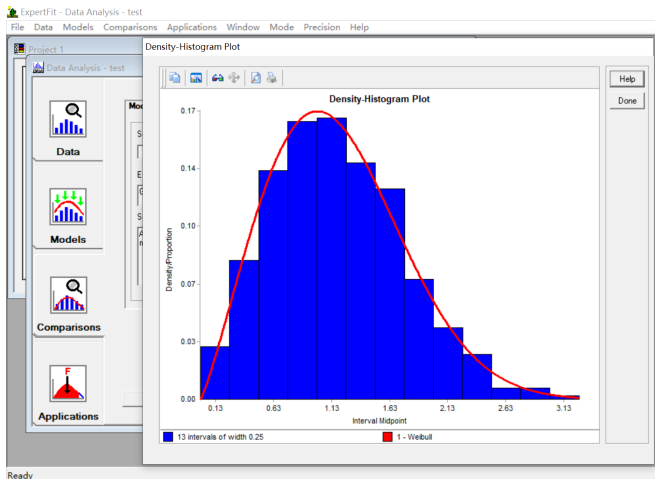
- 合并和区域限制

- 教程: User Manual / Tutorials / Additional Tools Tutorials / Tutorial 1 - Conveyors / 1.2 - Merging and Gapping Systems

- 增加和移除间隔

- 教程: User Manual / Tutorials / Additional Tools Tutorials / Tutorial 1 - Conveyors / 1.3 - Adding and Removing Gaps

- 分布拟合工具 (ExpertFit Version 8)

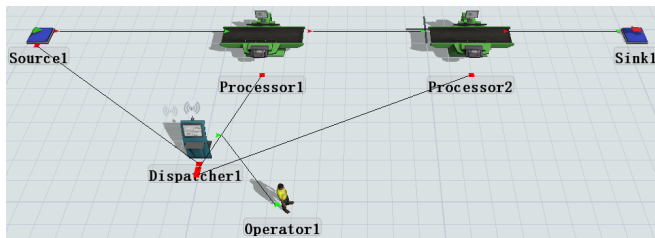


- 实验工具 (Experimenter)

- 使用户可以尝试同一个模型的不同参数设定, 并且评估在每一组参数下的表现.

- 简单例子: 调整机器的位置, 观察生产量的变化

- 教程: User Manual / Tutorials / Additional Tools Tutorials / Tutorial 4 - Experimenter / Optimizer / 4.1 - Experimenter
- 模型: `Experimenter.fsm`





- 创建变量 (variables) 和方案 (scenarios)

Simulation Experiment Control

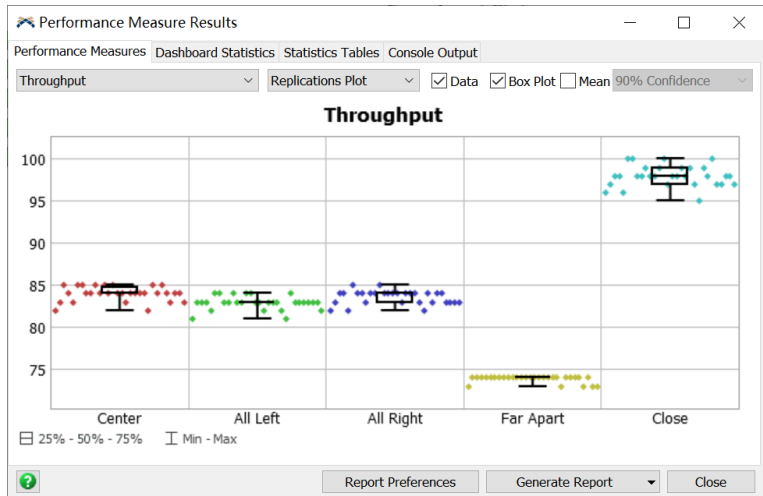
Simulation Experiment Control						
Scenarios Performance Measures Experiment Run Optimizer Design Optimizer Run Optimizer Results Advanced						
Variables		Scenarios			Choose default reset scenario:	
Variable	Center	All Left	All Right	Far Apart	Close	
Proc1X	MODEL:/Processor1>spatial/spatialx	-10	-13	-7	-13	-7
Proc2X	MODEL:/Processor2>spatial/spatialx	0	-3	3	3	-3

- 运行实验 (experiment)

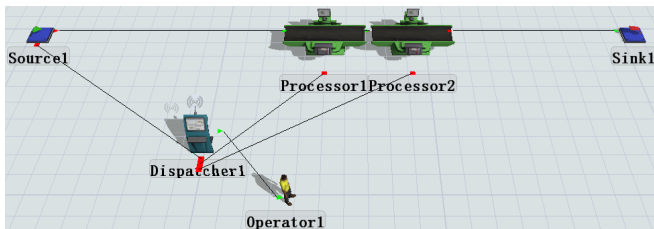
Simulation Experiment Control

Simulation Experiment Control						
Scenarios Performance Measures Experiment Run Optimizer Design Optimizer Run Optimizer Results Advanced						
<b>Reset Experiment</b>		End Time	9:00:00	2021/ 5/ 5	<input type="checkbox"/> Save statistics data for each replication	
Replications per Scenario		Run Time	3600.00	Seconds	<input type="checkbox"/> Save state after each replication	
		Warmup	0.00	Seconds	<input type="checkbox"/> Restore original state after each replication	
<b>Experiment Status</b>						
Center						
All Left						
All Right						
Far Apart						
Close						

## ● 实验结果



- 优化工具 (Optimizer): OptQuest
  - 自动生成一系列的方案进行评估, 并且找出最大化/最小化目标函数的方案.
- 简单例子: 寻找使生产量 (或产值) 最大化的机器的位置
  - 教程: User Manual / Tutorials / Additional Tools Tutorials / Tutorial 4 - Experimenter / Optimizer / 4.2 - Optimizer
  - 模型: `Optimizer.fsm`



## • 优化器设定

Simulation Experiment Control

Scenarios	Performance Measures	Experiment Run	Optimizer Design	Optimizer Run	Optimizer Results	Advanced
Variables			Constraints			
	Type	Lower Bound	Upper Bound	Step	Group	
Proc1X	Continuous	-13	-7	N/A	N/A	<input type="checkbox"/>
Proc2X	Continuous	-3	3	N/A	N/A	<input type="checkbox"/>

## • 运行优化

Simulation Experiment Control

Scenarios	Performance Measures	Experiment Run	Optimizer Design	Optimizer Run	Optimizer Results	Advanced
<b>Optimize</b>		End Time	10:46:40	2021/ 5/ 5	Wall Time	0.00
		Run Time	10000.00	Seconds	Max Solutions	50.00
		Warmup	0.00	Seconds		
<input type="checkbox"/> Show advanced options		<input type="checkbox"/> Run multiple replications per solution		<input type="checkbox"/> Manual stop only		



## ● 优化结果

