

无人系统设计

课 程：软件工程专业-专业实践类课程

学 分：3

总课时：48

课程参考教材：

《认识飞行（第二版）》 / 《Understanding Flight, 2nd》

作者：David F. Anderson, Scott Eberhardt

译者：周尧明（2019年） / 韩莲（2011年）

北京联合出版公司2019.07 / 航空工业出版社2011.01

授课教师：王赓

课程助教：李旭辉、蒋李康、方俊杰、张源娣、范文婷、曹恺洋、杨道

课程主要内容

(1) 认识飞行

- ▣ 牛顿力学（作用力与反作用力）
- ▣ 刚体转动（转矩、陀螺、进动） / 大学物理基础

(2) 认识多种多样的无人飞行系统

- ▣ 飞行原理
- ▣ 动力技术（螺旋桨、喷气式）

(3) 控制技术

- ▣ 飞行操纵原理（机翼、襟翼、旋翼、尾桨、自动倾斜器）
- ▣ 作动器（电动机、舵机（PWM调制））
- ▣ 传感器（电子指南针、加速度计、陀螺仪、GPS、高度计、高速相机、全景相机、……）
- ▣ 电子控制器（PID算法、飞行控制原理与算法）

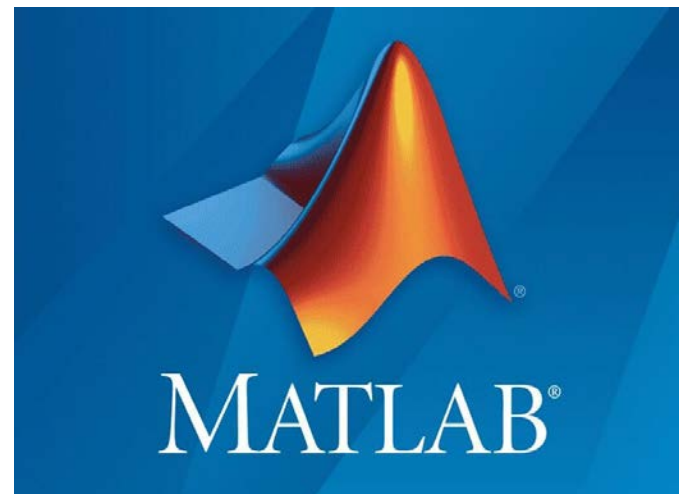
(4) 飞行性能（飞行性能指标体系、稳定性、可靠性、易操作性）

课程主要内容

- (5) 基于4旋翼、固定翼模型机的认知验证实验
(含早期自由组合发现学习过程)
- (6) 仿真技术
 - ▣ 飞行器建模 (动力学、运动学) / 大学物理基础、高等数学
 - ▣ 软件技术 (Unity3D、MATLAB/Simulink)
- (7) 仿真技术实践 (半实物)
 - ▣ **Dog Fight** (狗斗) 战机模拟格斗对抗系统
 - ▣ 软件技术 (Unity3D、MATLAB/Simulink、图像处理技术、人工智能AI技术、计算加速技术
- (8) 发挥想象力和所学的自由拓展设计 (理论设计/尽量据情实验验证)
- (9) 课程综合设计与答辩

MATLAB/Simulink软件应用技术

■ MATLAB/Simulink



Simulink/MATLAB

- Simulink是一个基于模块图结构的设计和仿真环境，用于多个领域的系统仿真以及基于模型的设计验证。
- 它支持系统级设计、仿真、自动代码生成以及嵌入式系统的连续测试和验证。
- Simulink 提供图形编辑器、可自定义的模块库以及求解器，能够进行动态系统建模和仿真
- Simulink 与 MATLAB相集成，这样不仅能够在 Simulink 中将 MATLAB 算法融入模型，还能将仿真结果导出至 MATLAB 做进一步分析。

参考：MATLAB/Simulink自身帮助文档

MATLAB/Simulink

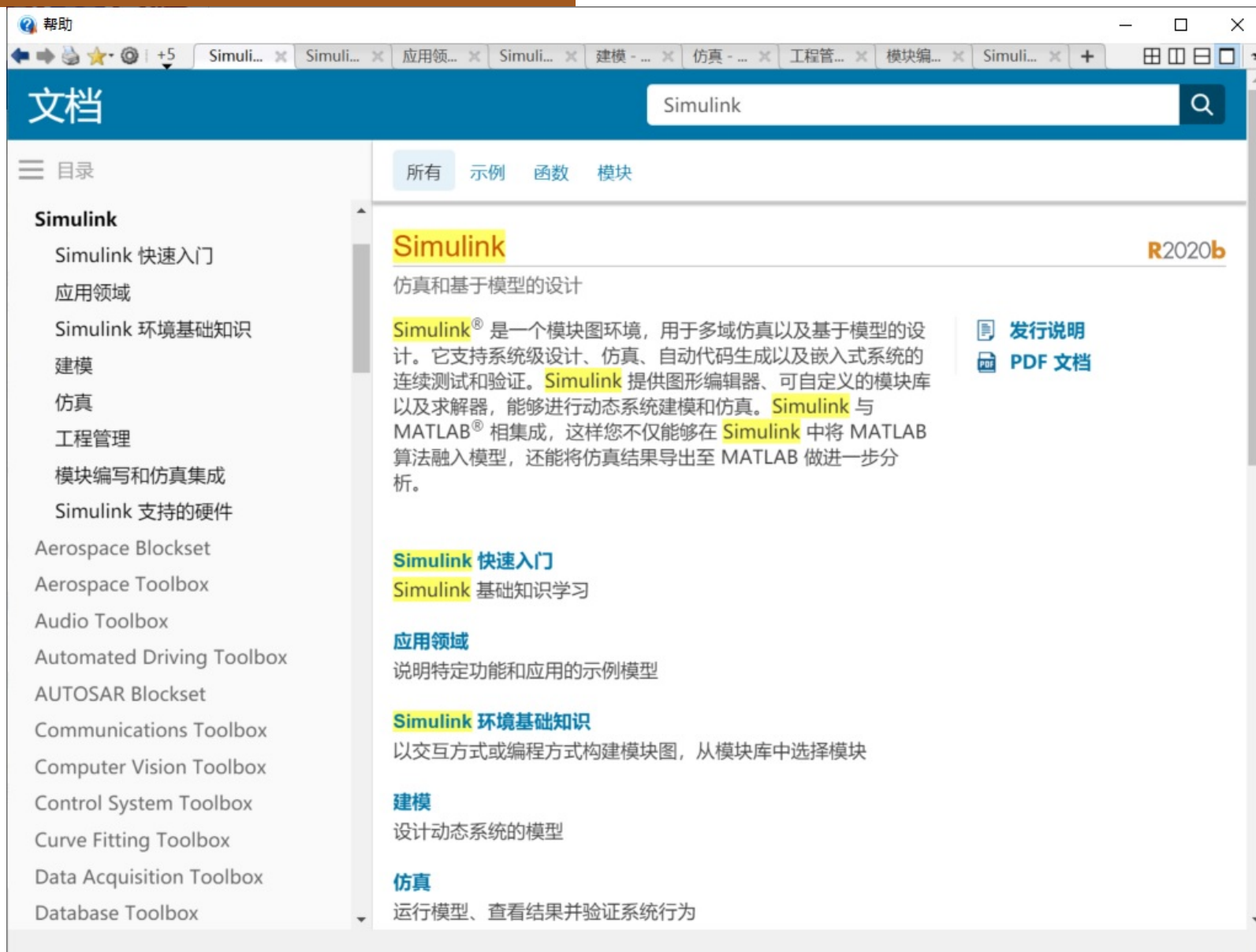
- 20世纪70年代，美国新墨西哥大学计算机科学系主任Cleve Moler为了减轻学生编程的负担，用FORTRAN编写了最早的MATLAB。
- 1984年由Little、Moler、Steve Bangert合作成立的MathWorks公司正式把MATLAB推向市场。到20世纪90年代，MATLAB已成为国际控制界的标准计算软件。
- MATLAB统一了用于一维、二维与三维数值积分的函数并提升了基本数学和内插函数的性能。
- 不断推出基于数学计算解算的行业应用算法工具箱。

参考：<https://baike.baidu.com/item/MATLAB/263035>

Simulink/MATLAB

- Simulink是**Mathworks**公司推出的MATLAB中的一种可视化仿真工具。
- Simulink提供图形编辑器、可自定义的模块库以及求解器，能够进行动态系统建模和仿真。
- Simulink与MATLAB相集成，能够在Simulink 中将MATLAB算法融入模型，还能将仿真结果导出至MATLAB 做进一步分析。
- Simulink应用领域包括汽车、航空、工业自动化、大型建模、复杂逻辑、物理逻辑，信号处理、图像分析等多方面。

参考：<https://baike.baidu.com/item/SIMULINK/10306793?fr=aladdin>
(建议同学们直接阅读MATLAB/Simulink的说明和帮助文档)



注：建议同学们在学习中直接阅读MATLAB/Simulink的说明和帮助文档

帮助

Simuli... x Simuli... x 应用领... x Simuli... x 建模 - ... x 仿真 - ... x 工程管... x 模块编... x Simuli... x +

文档

搜索 R2020b 文档

目录

« 文档主页
« Simulink

类别

Simulink 快速入门

应用领域

Simulink 环境基础知识

建模

仿真

工程管理

模块编写和仿真集成

Simulink 支持的硬件

所有 示例 函数 模块

Simulink 快速入门

R2020b

仿真和基于模型的设计

Simulink® 是一个模块图环境，用于多域仿真以及基于模型的设计。它支持系统级设计、仿真、自动代码生成以及嵌入式系统的连续测试和验证。Simulink 提供图形编辑器、可自定义的模块库以及求解器，能够进行动态系统建模和仿真。Simulink 与 MATLAB® 相集成，这样您不仅能够在 Simulink 中将 MATLAB 算法融入模型，还能将仿真结果导出至 MATLAB 做进一步分析。

教程

Simulink 模块图

了解 Simulink 的基础知识。

创建简单模型

在 Simulink 中对一个简单系统建模。

导航模型

导航一个系统中的层次结构和连接。

使用 Simulink 进行基于模型的设计

使用 Simulink 进行基于模型的设计。

步骤 1: 系统定义和布局

步骤 2: 建模并验证系统

步骤 3: 在 Simulink 中设计系统

在线学习

Simulink 入门之旅

Simulink 许可证附带三小时交互式培训课程

视频

Simulink 控制快速入门

注：建议同学们在学习中直接阅读MATLAB/Simulink的说明和帮助文档

文档

搜索 R2020b 文档

目录

- « 文档主页
- « Simulink

类别

- Simulink 快速入门
- 应用领域**
 - 一般应用领域
 - 汽车应用领域
 - 航空应用领域
 - 工业自动化应用领域
 - 信号处理
 - 物理建模
 - 复杂逻辑
 - 离散事件仿真
 - 系统工程
 - 大型建模
- Simulink 环境基础知识
- 建模
- 仿真
- 工程管理
- 模块编写和仿真集成
- Simulink 支持的硬件

所有 示例 函数 模块

应用领域 R2020b

说明特定功能和应用的示例模型

查看关于特定行业和应用的动态系统建模和仿真的示例。

一般应用领域

说明一般应用的示例模型

汽车应用领域

使用 Simulink® 和其他 MathWorks® 产品对汽车系统进行建模和仿真

航空应用领域

使用 Simulink 和 Aerospace Blockset™ 软件进行航空航天系统建模

工业自动化应用领域

说明工业自动化应用的示例模型

信号处理

使用 DSP System Toolbox™ 软件对信号处理和通信系统建模

物理建模

使用 Simscape™ 软件进行物理系统建模

复杂逻辑

使用 Stateflow® 图为复杂逻辑建模

离散事件仿真

进行离散事件系统建模和仿真

系统工程

使用 System Composer™ 软件设计和分析系统体系结构

大型建模

适用于大型模型和多用户开发团队的模型架构

注：建议同学们在学习中直接阅读MATLAB/Simulink的说明和帮助文档

文档

搜索 R2020b 文档

目录

« 文档主页

« Simulink

类别

Simulink 快速入门

应用领域

Simulink 环境基础知识

Simulink 概念

交互式模型编辑

编程式模型编辑

Simulink 环境自定义

模型升级

模块库

建模

仿真

工程管理

模块编写和仿真集成

Simulink 支持的硬件

所有 示例 函数 模块

Simulink 环境基础知识

R2020b

以交互方式或编程方式构建模块图，从模块库中选择模块

Simulink 提供可自定义的模块库、用于交互式模型编辑的图形编辑器和用于编程式模型编辑的 API。

一个全面的预定义模块库可以帮助您构建模型来表示动态系统。使用 Simulink Editor 将库中的模块添加到模型中。然后，使用信号线连接模块，以确立各系统组件之间的数学关系。您可以细化模型的外观，并控制用户如何与模型交互。

相关主题

以交互方式构建和编辑模型

保存模型

Simulink 建模的键盘快捷方式和鼠标操作

编程建模基础知识

以交互方式构建和编辑模型

格式化模型

Simulink 概念

Simulink[®] 中有关动态系统建模和仿真的概念

交互式模型编辑

使用模块和信号线创建模块图模型

编程式模型编辑

以编程方式创建模型、在模型中添加和连接模块、设置参数和属性以及执行其他基本建模操作。

Simulink 环境自定义

在菜单和对话框中添加或删除项目、更改显示方式和自动保存设置

模型升级

将模型升级到当前 Simulink 版本

模块库

代表方程和建模组件的模块

注：建议同学们在学习中直接阅读MATLAB/Simulink的说明和帮助文档

三 目录

« 文档主页

« Simulink

类别

Simulink 快速入门

应用领域

Simulink 环境基础知识

建模

设计模型架构

管理设计数据

设计模型行为

配置信号、状态和参数

配置输入和可视化

分析和重新设计模型

测试模型组件

建模规范

仿真

工程管理

模块编写和仿真集成

Simulink 支持的硬件

所有

示例

函数

模块

建模

R2020b

设计动态系统的模型

通过 Simulink® 利用模块图为算法和物理系统建模。您可以为线性及非线性系统建模，以便模拟真实的现象，如摩擦、齿轮滑动和强迫停止。

通过将各组模块组织成子系统，可以将您的模型设计成层次结构。利用这种方法，可以构建反映真实系统的离散组件，并对这些组件之间的交互进行仿真。

尽早考虑系统设计需求可以减少设计过程后期发现的错误数量。有关迭代设计的示例，请参阅[Basic Modeling Workflow](#)。

使用 Simulink 进行基于模型的设计，其中系统模型是整个开发过程（从需求开发到设计、实现和测试）的核心。

相关主题

[基于组件的建模规范](#)

[创建子系统](#)

[设置模块参数值](#)

[关于查找表模块](#)

[决定如何可视化仿真数据](#)

[使用模型顾问检查您的模型](#)

设计模型架构

创建独立组件以便在模型或团队之间共享，通过模型层次结构降低视觉上的复杂度

管理设计数据

使用模型工作区、符号、数据对象和数据类来指定变量值

设计模型行为

特定功能和应用的建模方法

配置信号、状态和参数

配置模块参数、信号范围、初始状态、数据类型、采样时间

配置输入和可视化

提供信号数据并决定如何可视化其仿真



仿真

R2020b

运行模型、查看结果并验证系统行为

使用 Simulink[®]，您可以通过交互方式对系统模型进行仿真，并通过波形和图形显示查看结果。对于连续、离散和混合信号系统的仿真，您可以从一系列固定步长和可变步长求解器中进行选择。求解器是能够计算系统随时间动态变化的积分算法。

由于 Simulink 已经与 MATLAB[®] 集成，所以您可以使用 MATLAB 命令对 Simulink 模型运行无人值守的批量仿真。

相关主题

[选择求解器](#)[决定如何可视化仿真数据](#)[Inspect Simulation Data](#)[Prototype and Debug Models with Scopes](#)[导出仿真数据](#)[以编程方式运行仿真](#)

准备模型输入和输出

定义外部接口用于输入数据和控制信号，定义输出信号用于查看和记录仿真结果

配置仿真条件

选择求解器、设置初始条件、选取输入数据集、设置步长大小

运行仿真

以交互方式或批量模式进行模型仿真、使用 SimState 创建可重复的仿真、运行蒙特卡罗模拟。

查看和分析仿真结果

查看仿真结果以便为模型建立原型并进行调试，检查和比较多次仿真的数据以验证设计

测试和调试仿真

提高仿真准确性、管理和诊断仿真错误

优化性能

针对特定目的优化性能、加快仿真速度以及设计高效模型

特色示例



工程管理

R2020b

创建工程、管理共享的模型组件、与源代码管理进行交互

通过工程，您可以查找所需的文件、管理并共享文件和设置以及与源代码管理进行交互，从而有助于您组织大型建模工程。请参阅[什么是工程？](#)。

相关主题

[什么是工程？](#)[从模型创建工程](#)[Automate Startup Tasks](#)[Automate Project Tasks Using Scripts](#)[Perform an Impact Analysis](#)[About Source Control with Projects](#)[通过一个弹体工程来了解工程工具](#)

工程设置

将工作组织为工程、自动启动和关闭、找出常见任务、重用标准模板设置

工程文件管理

搜索、标记、批处理、共享和升级工程文件以及创建脚本。

依存关系分析

查找所需文件、执行影响分析

工程中的源代码管理

配置源代码管理、检索文件、审核更改内容、提交已修改文件

大型工程的组件化

使用参考工程来组织大型工程

模型比较

比较模型以识别更改或将模型之间的差异合并

需求可追溯性

查看工程、模型和模块中的关联需求

特色示例

文档

搜索 R2020b 文档

目录

« 文档主页

« Simulink

类别

Simulink 快速入门

应用领域

Simulink 环境基础知识

建模

仿真

工程管理

模块编写和仿真集成

模块创建基础知识

使用自定义模块扩展建模功能

与现有仿真或环境集成

创建、组织和发布自定义模块集

创建模块封装

Simulink 支持的硬件

所有 示例 函数 模块

试用版 产品更新

模块编写和仿真集成

R2020b

使用 MATLAB® 和 C/C++ 代码扩展现有 Simulink® 建模功能

您可以通过创建新模块并将它们添加到 Simulink 库浏览器来扩展 Simulink 的现有建模功能。使用自定义模块可以：

- 对 Simulink 内置解决方案未提供的行为进行建模。
- 使用现有仿真构建更高级的模型
- 将多个模型组件封装为一个库模块，可以将该库模块复制到多个模型中。
- 提供自定义用户界面或分析例程。

使用 C Caller 模块集成 C 代码

Implement a MATLAB System Block

使用 MATLAB Function 模块创建自定义功能

Use a Bus Signal with S-Function Builder to Create an S-Function

模块创建基础知识

通过新模块开发扩展 Simulink 的模块功能的原则

使用自定义模块扩展建模功能

使用 MATLAB、C/C++ 和 Fortran 在 Simulink 中实现新算法

与现有仿真或环境集成

将第三方功能集成到 Simulink 中，包括功能模型单元 (FMU)、App、模型和工具箱

创建、组织和发布自定义模块集

创建模块集设计器工程来管理自定义模块及其工件

创建模块封装

为子系统和自定义模块创建自定义外观、创建用户定义的界面、封装逻辑以及隐藏数据

注：建议同学们在学习中直接阅读MATLAB/Simulink的说明和帮助文档



上海交通大学
SHANGHAI JIAO TONG UNIVERSITY

上海交通大学 软件学院

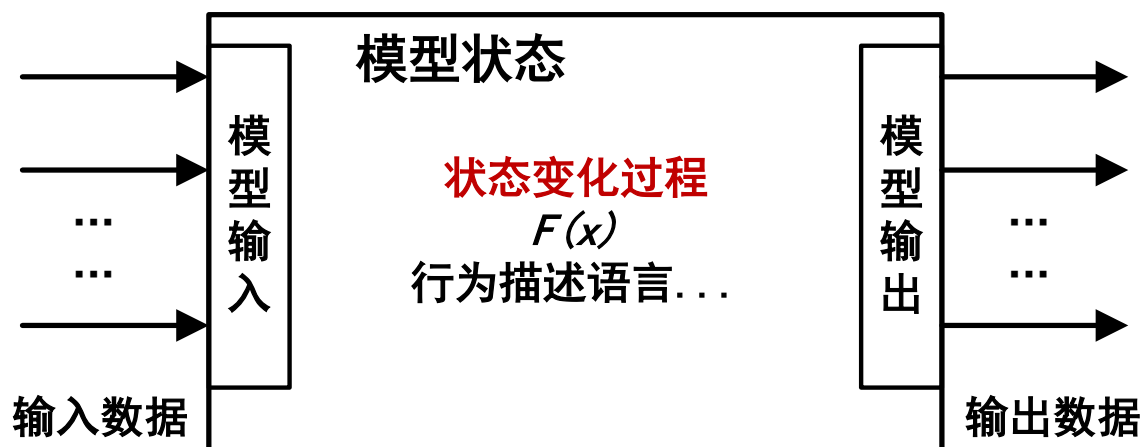
2021-03-01

15 / 31

Simulink工作原理

图形化模型与数学模型间的关系

- 现实中每个系统都有输入输出和状态3个基本要素，他们之间随时间变化的数学函数关系，即数学模型。



- 图形化模型也体现了输入、输出和状态随时间变化的某种关系。
- 只要这两种关系在数学上是等价的，就可以用图形化模型代替数学模型。

Simulink工作原理

图形化模型的仿真过程

- 包含：模型编译阶段、连接阶段、仿真阶段3个过程

1. 模型编译阶段

- simulink引擎调用模型编译器，将模型翻译成可执行文件。其中编译器主要完成以下任务：
 - 计算模块参数的表达式，以确定他们的值
 - 确定信号属性（如名称、数据类型等）
 - 传递信号属性，以确定未定义信号的属性
 - 综合优化模块
 - 展开模型的继承关系（如子系统）
 - 确定模块运行的优先级（前因后果）
 - 确定模块的采样时间

Simulink工作原理

图形化模型的仿真过程

- 包含：模型编译阶段、连接阶段、仿真阶段3个过程

2. 连接阶段

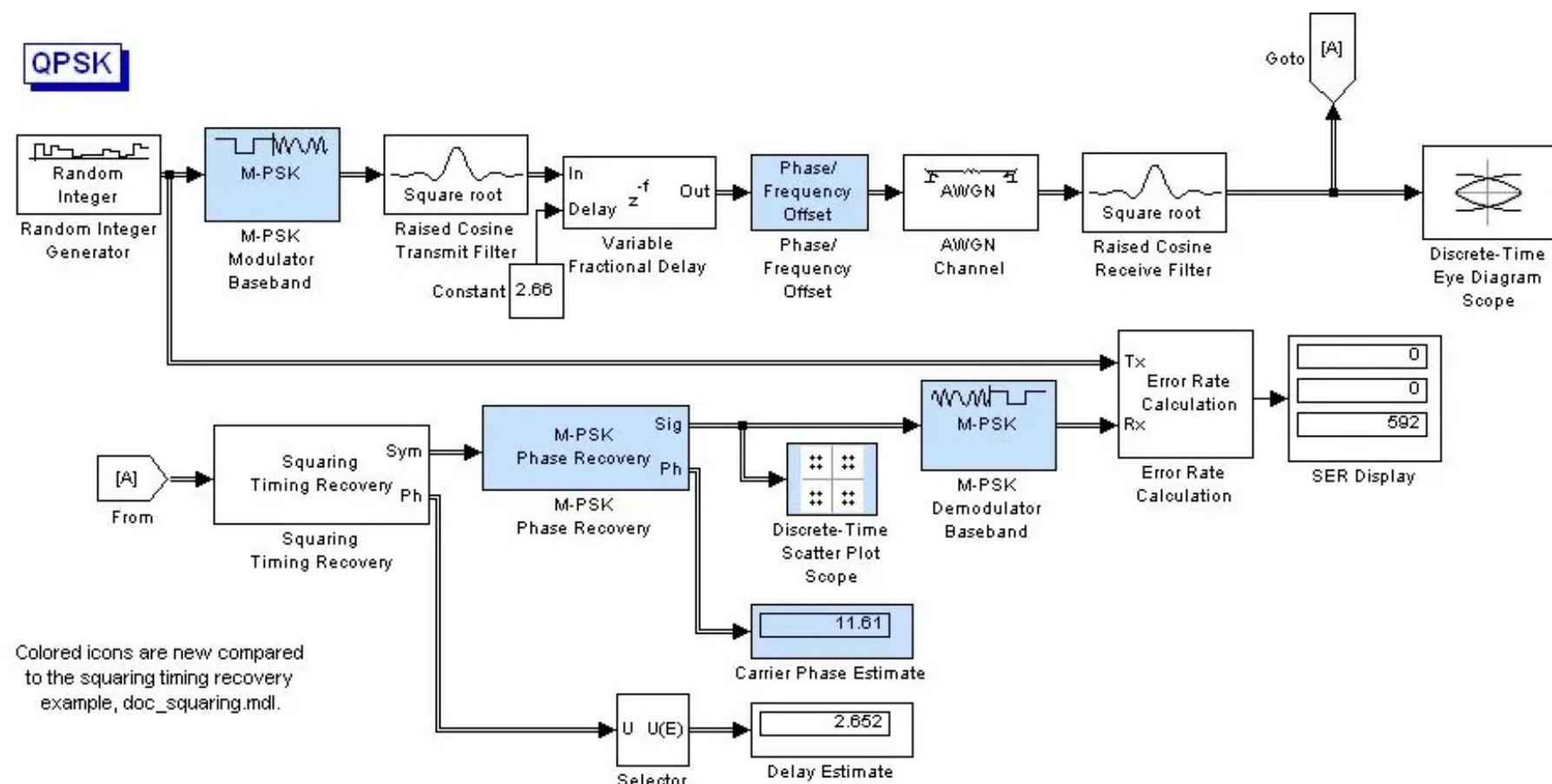
- simulink引擎按执行次序创建运行列表，初始化每个模块的运行信息。

3. 仿真阶段

- simulink引擎从仿真的开始到结束，在每一个采样点按运行列表计算各模块的状态和输出。该阶段又分成以下两个子阶段：
 - 初始化阶段：该阶段只运行一次，用于初始化系统的状态和输出。
 - 迭代阶段：该阶段在定义的时间段内按采样点之间的步长重复运行，并将每次的运算结果用于更新模型。在仿真结束时获得最终的输入、输出和状态值。

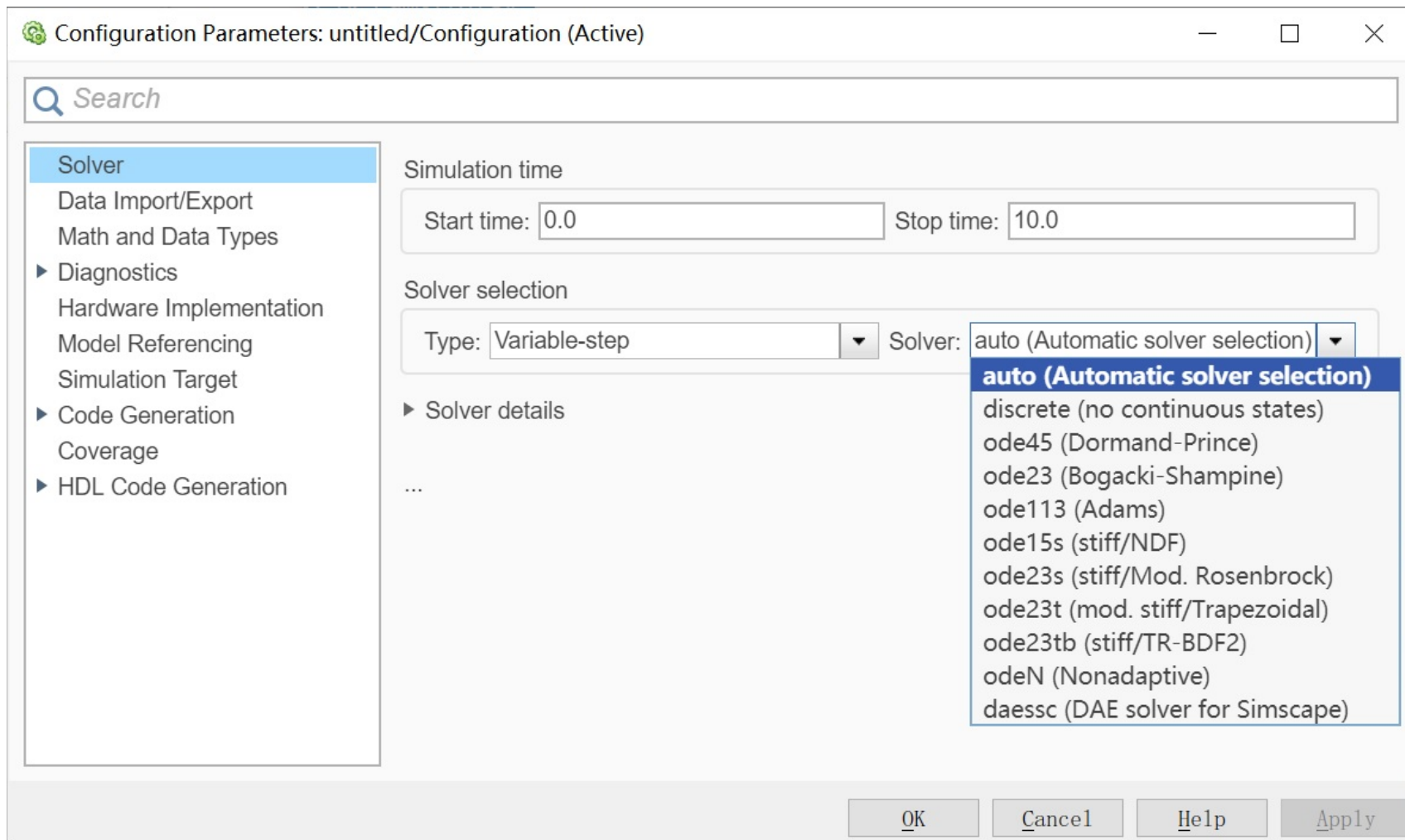
Simulink模型示例

建议同学们浏览系统自带提供的多种领域类型的实例示例



Simulink工作原理

■ 关于模型仿真中的求解器类型



选择求解器

R2020b

要仿真动态系统，可以计算其在指定时间跨度内连续时间步的状态。此计算使用系统模型提供的信息。*时间步*是发生计算的时间间隔。此时间间隔的大小称为*步长大小*。以这种方式计算模型状态的过程称为*解算*模型。没有任何一种模型解算方法能适用于所有系统。Simulink® 提供了一组程序，称为*求解器*。每个求解器代表一种特定的模型解算方法。

求解器运用一种数值方法来解算代表模型的一组常微分方程。通过这种计算，它确定下一个仿真步的时间。在解算此初始值问题的过程中，求解器还满足您指定的准确性要求。

数学家们开发出了多种数值积分方法来解算表示动态系统连续状态的常微分方程 (ODE)。提供了一套全面的固定步长和可变步长连续求解器，其中每个求解器实现一种特定的 ODE 解法（请参阅[比较求解器](#)）。在模型配置参数的 **Solver** 窗格中选择求解器。

MATLAB® 和 Simulink 提供的所有求解器都遵循类似的命名约定：ode 后跟两三个数字（表示求解器的阶）。有些求解器可以求解刚性微分方程，它们使用的方法由 s、t 或 tb 后缀表示。

求解器选择标准

仿真模型的合适求解器取决于以下特性：

- 系统动态特性
- 解稳定性
- 计算速度
- 求解器稳健性

因此，Simulink 提供的数值求解器可大致按两个属性分类。

计算步长大小类型

- 顾名思义，固定步长求解器使用相同的步长大小从仿真开始到仿真结束来解算模型。您可以指定步长大小，也可以让求解器选择步长大小。一般情况下，减小步长大小将提高结果的准确性，并增加系统仿真所需的时间。
- 可变步长求解器会在仿真过程中改变步长大小。这些求解器减小步长大小，以提高模型仿真过程中某些事件（如快速状态变化、过零事件等）的精度。另外，当模型状态变化缓慢时，求解器将增加步长大小以避免执行不必要的步长。计算步长大小将增加每个步长的计算开销，但是它可以减少步长总数，因此，对于具有许多快速变化的状态以及其他需要额外

比较求解器

R2020b

求解器通过使用模型提供的信息，计算动态系统在指定时间范围内连续时间步的状态，从而对动态系统进行仿真。根据系统模型计算系统连续状态的过程称为解算模型。没有任何一种模型解算方法可满足所有系统需要。因此，Simulink® 提供一组称为 **求解器** 的程序，每个程序都利用一种特定方法来解算模型。Configuration Parameters 对话框允许您选择最适合您的模型的求解器。

固定步长和可变步长求解器

您可以根据 Simulink 提供的求解器计算步长大小的方式来选择求解器：固定步长和可变步长。

固定步长求解器按固定时间间隔在从仿真开始到仿真结束的时间段内解算模型。间隔的大小称为步长大小。您可以指定步长大小，也可以让求解器选择步长大小。一般情况下，减小步长大小将提高结果的准确性，但会增加系统仿真所需的时间。

可变步长求解器会在仿真过程中改变步长大小。它们在模型状态快速变化时，会减小步长大小以提高准确性；当模型状态缓慢变化时，增加步长大小以避免执行不必要的时间步。计算步长大小会增加每个步长的计算开销，但可以减少对具有快速变化的状态或分段连续状态的模型维护指定级别的准确性所需的总时间步数，从而缩短仿真时间。

固定步长和可变步长求解器将下一个仿真时间计算为当前仿真时间与步长大小之和。**Solver** 配置窗格上的 **Type** 控件用于选择求解器类型。对于固定步长求解器，步长大小在仿真过程中保持不变。对于可变步长求解器，每一个时间步的步长大小可能各不相同，具体情况取决于模型的动态特性。具体来讲，就是可变步长求解器会增加或减少步长大小，以满足您指定的误差容限。

选择哪一种求解器取决于您计划如何部署模型以及模型的动态特性。如果您计划从模型中生成代码，然后在实时计算机系统上运行代码，请选择固定步长求解器来进行模型仿真。您不能将可变步长大小映射到实时时钟。

如果您不打算以生成代码的形式来部署模型，则选择可变步长求解器还是固定步长求解器取决于模型的动态特性。可变步长求解器也许能显著缩短模型的仿真时间。可变步长求解器之所以能够节省仿真时间，是因为求解器可以根据需要动态调整步长大小来达到给定级别的准确性。这种方法可以减少所需的步长数。固定步长求解器必须根据准确性要求在整个仿真过程中使用一种步长大小。为了在整个仿真过程中满足这些要求，固定步长求解器通常需要较小的步长。

ex_multirate 示例模型显示了可变步长求解器如何缩短多速率离散模型的仿真时间。



搜索



三 筛选

关闭

试用版

产品更新

« 所有产品

搜索的是 刚性微分方程

1 - 10 条搜索结果, 共 10 条

按类型分类

fx 函数	6
帮助主题	3
发行说明	1

按产品分类

MATLAB	8
Simulink	2

ode45 - 求解非刚性微分方程 - 中阶方法

fx

此 MATLAB 函数 (其中 tspan = [t0 tf]) 求微分方程组 $y' = f(t, y)$ 从 t0 到 tf 的积分, 初始条件为 y0。解数组 y 中的每一行都与列向量 t 中返回的值相对应。

[文档](#) > [MATLAB](#) > [数学](#) > [数值积分和微分方程](#) > [常微分方程](#)

ode23 - 求解非刚性微分方程 - 低阶方法

fx

此 MATLAB 函数 (其中 tspan = [t0 tf]) 求微分方程组 $y' = f(t, y)$ 从 t0 到 tf 的积分, 初始条件为 y0。解数组 y 中的每一行都与列向量 t 中返回的值相对应。

[文档](#) > [MATLAB](#) > [数学](#) > [数值积分和微分方程](#) > [常微分方程](#)

ode113 - 求解非刚性微分方程 - 变阶方法

fx

此 MATLAB 函数 (其中 tspan = [t0 tf]) 求微分方程组 $y' = f(t, y)$ 从 t0 到 tf 的积分, 初始条件为 y0。解数组 y 中的每一行都与列向量 t 中返回的值相对应。

[文档](#) > [MATLAB](#) > [数学](#) > [数值积分和微分方程](#) > [常微分方程](#)

ode15s - 求解刚性微分方程和 DAE - 变阶方法

fx

此 MATLAB 函数 (其中 tspan = [t0 tf]) 求微分方程组 $y' = f(t, y)$ 从 t0 到 tf 的积分, 初始条件为 y0。解数组 y 中的每一行都与列向量 t 中返回的值相对应。

[文档](#) > [MATLAB](#) > [数学](#) > [数值积分和微分方程](#) > [常微分方程](#)

ode23s - 求解刚性微分方程 - 低阶方法

fx

此 MATLAB 函数 (其中 tspan = [t0 tf]) 求微分方程组 $y' = f(t, y)$ 从 t0 到 tf 的积分, 初始条件为 y0。解数组 y 中的每一行都与列向量 t 中返回的值相对应。

[文档](#) > [MATLAB](#) > [数学](#) > [数值积分和微分方程](#) > [常微分方程](#)

刚性方程

■ 摘自Wikipedia——刚性方程

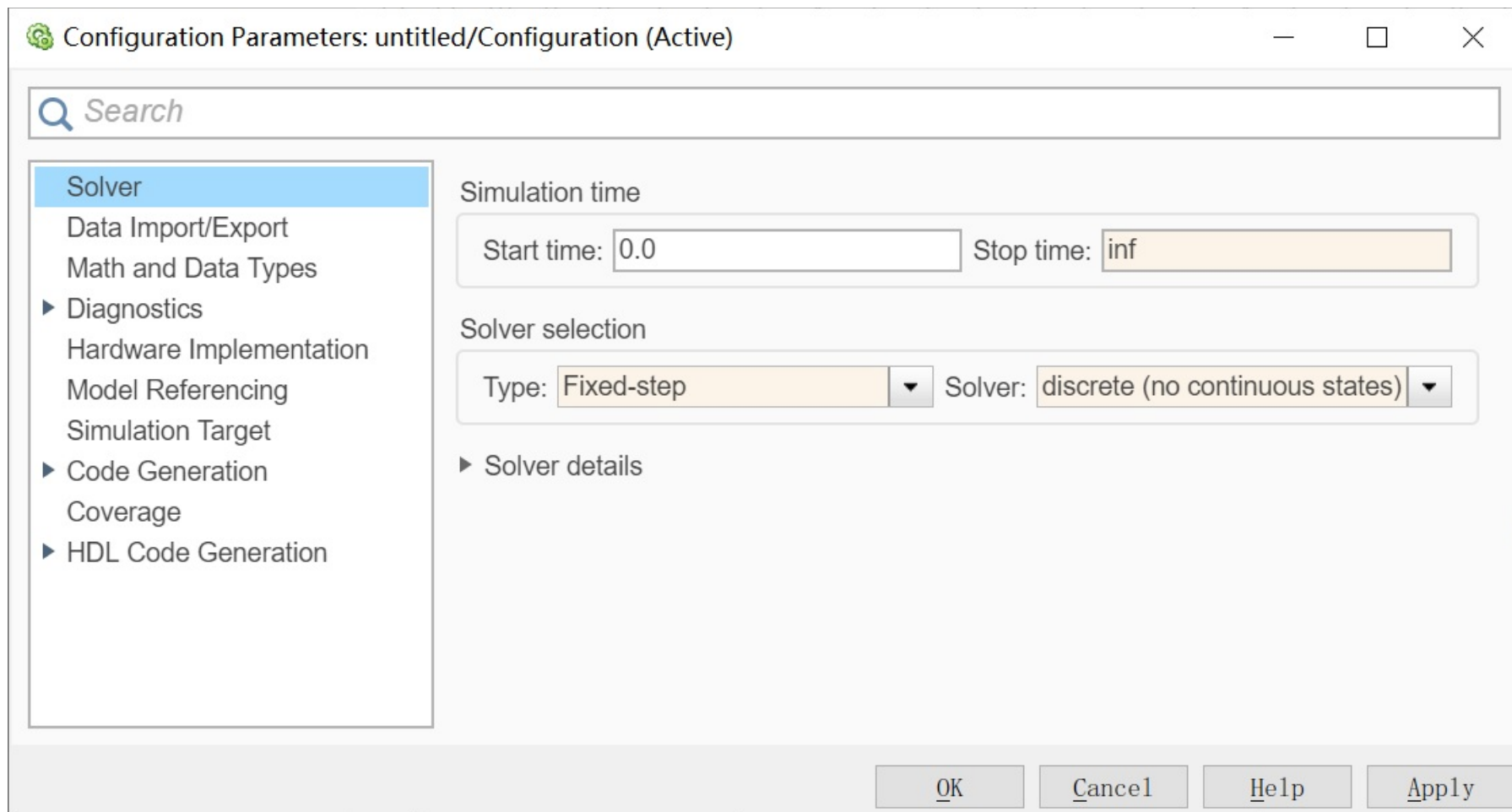
定义

- 在数学领域中，刚性方程（stiffness equation）是指一个微分方程，其数值分析的解只有在时间间隔很小时才会稳定，只要时间间隔略大，其解就会不稳定。
- 目前很难去精确地去定义哪些微分方程是刚性方程，然而粗略而言，若此方程式中包含使其快速变动的项，则其为刚性方程。
- 在积分微分方程时，若某一区域的解曲线的变化很大，会希望在这个区域的积分间隔密一些，若另一区域的曲线近似直线，且斜率接近零，会希望在这个区域的积分间隔松一些。不过针对一些问题，就算曲线近似直线，仍然需要用非常小的积分间隔来积分，这种现象称为“刚性”。有时可能会出现两个不同问题，一个有“刚性”，另一个没有，但两个问题却有同一个解的情形。因此“刚性”不是解本身的特性，而是微分方程的特性，也可以称为是刚性系统。

参考：<https://www.cnblogs.com/BlueHeart0621/p/12237942.html>

Simulink工作原理

■ 关于模型仿真中的求解器类型



Simulink工作原理

■ 一个模型过程演示（请参阅系统相应说明文档）

The screenshot shows the Simulink Documentation browser interface. At the top, there's a search bar with the text '搜索 R2020b 文档'. Below the search bar, there are tabs for 'All', 'Examples', 'Functions', 'Blocks', and 'Apps'. The 'Examples' tab is selected. On the left side, there's a sidebar with a '目录' (Table of Contents) section. It lists '« Documentation Home' and '« Instrument Control Toolbox'. Under '« Instrument Control Toolbox', there's '« Direct Interface' and 'Communication in Simulink'. Below that, 'Send and Receive Data over TCP/IP Network' is highlighted. Under 'ON THIS PAGE', there's a list of steps: 'Step 1: Create an Echo Server', 'Step 2: Create a New Model', 'Step 3: Open the Block Library', 'Step 4: Drag the Instrument Control Toolbox Blocks into the Model', 'Step 5: Drag the Sine Wave and Scope Blocks to Complete the Model', 'Step 6: Connect the Blocks', 'Step 7: Specify the Block Parameter Values', and 'Step 8: Specify the Block Priorities'. The main content area shows the title 'Send and Receive Data over TCP/IP Network' with the version 'R2020b'. Below the title, there's a paragraph: 'This example shows how to build a simple model using the Instrument Control Toolbox™ blocks in the block library in conjunction with other blocks in the Simulink® library. This example also illustrates how to send data to an echo server using TCP/IP and to read that data back into your model.' Below this, there's a 'Note' box with the text: 'Block names are not shown by default in the model. To display the hidden block names while working in the model, select **Display** and clear the **Hide Automatic Names** check box.' Below the note, there's a list of steps: 'Step 1: Create an Echo Server', 'Step 2: Create a New Model', 'Step 3: Open the Block Library', 'Step 4: Drag the Instrument Control Toolbox Blocks into the Model', 'Step 5: Drag the Sine Wave and Scope Blocks to Complete the Model', 'Step 6: Connect the Blocks', 'Step 7: Specify the Block Parameter Values', and 'Step 8: Specify the Block Priorities'.

Documentation

搜索 R2020b 文档

目录

- « Documentation Home
- « Instrument Control Toolbox
- « Direct Interface
- Communication in Simulink

Send and Receive Data over TCP/IP Network

ON THIS PAGE

- Step 1: Create an Echo Server
- Step 2: Create a New Model
- Step 3: Open the Block Library
- Step 4: Drag the Instrument Control Toolbox Blocks into the Model
- Step 5: Drag the Sine Wave and Scope Blocks to Complete the Model
- Step 6: Connect the Blocks
- Step 7: Specify the Block Parameter Values
- Step 8: Specify the Block Priorities

Send and Receive Data over TCP/IP Network R2020b

This example shows how to build a simple model using the Instrument Control Toolbox™ blocks in the block library in conjunction with other blocks in the Simulink® library. This example also illustrates how to send data to an echo server using TCP/IP and to read that data back into your model.

In this example, you create an echo server on your machine that simulates sending a signal to the TCP/IP Send block and echo the result back to the Send block to send data. You then use the TCP/IP Receive block to read that same data back into your model.

Note

Block names are not shown by default in the model. To display the hidden block names while working in the model, select **Display** and clear the **Hide Automatic Names** check box.

- Step 1: Create an Echo Server
- Step 2: Create a New Model
- Step 3: Open the Block Library
- Step 4: Drag the Instrument Control Toolbox Blocks into the Model
- Step 5: Drag the Sine Wave and Scope Blocks to Complete the Model
- Step 6: Connect the Blocks
- Step 7: Specify the Block Parameter Values
- Step 8: Specify the Block Priorities

MATLAB

- MATLAB交大授权正版下载地址：
- <http://lic.si.sjtu.edu.cn/Default/index>



MATLAB（在线授权版）安装授权操作说明

MATLAB是MathWorks公司出品的商业数学软件，用于数据分析、无线通信、深度学习、图像处理与计算机视觉、信号处理、量化金融与风险管理、机器人，控制系统等领域。MATLAB（在线授权版）适合校园网连线状态下获得授权使用，公网环境下也可通过开启交大VPN获得校内IP后授权使用。目前本平台可提供的最新版本为2020b，还可提供2020a，2019b和2019a，更早版本请至Mathworks官网自行下载。MATLAB支持Windows、Linux和MAC操作系统。本安装授权说明以Windows为例，适用于当前可提供所有版本。

☆4 👁5989

课程实践平台

- MATLAB/Simulink + Unity3D

- 几个示例及说明

- 实践平台接口构成

- 要求同学们尽早尽快熟悉Simulink/MATLAB应用

课程主要内容

(1) 认识飞行

- ▣ 牛顿力学（作用力与反作用力）
- ▣ 刚体转动（转矩、陀螺、进动） / 大学物理基础

(2) 认识多种多样的无人飞行系统

- ▣ 飞行原理
- ▣ 动力技术（螺旋桨、喷气式）

(3) 控制技术

- ▣ 飞行操纵原理（机翼、襟翼、旋翼、尾桨、自动倾斜器）
- ▣ 作动器（电动机、舵机（PWM调制））
- ▣ 传感器（电子指南针、加速度计、陀螺仪、GPS、高度计、高速相机、全景相机、……）
- ▣ 电子控制器（PID算法、飞行控制原理与算法）

(4) 飞行性能（飞行性能指标体系、稳定性、可靠性、易操作性）

课程主要内容

- (5) 基于4旋翼、固定翼模型机的认知验证实验
(含早期自由组合发现学习过程)
- (6) 仿真技术
 - ▣ 飞行器建模 (动力学、运动学) / 大学物理基础、高等数学
 - ▣ 软件技术 (Unity3D、MATLAB/Simulink)
- (7) 仿真技术实践 (半实物)
 - ▣ **Dog Fight** (狗斗) 战机模拟格斗对抗系统
 - ▣ 软件技术 (Unity3D、MATLAB/Simulink、图像处理技术、人工智能AI技术、计算加速技术
- (8) 发挥想象力和所学的自由拓展设计 (理论设计/尽量据情实验验证)
- (9) 课程综合设计与答辩

Question & Answer

任何疑问和建议，请不要犹豫！

王 赓: wgeng@sjtu.edu.cn