

Matlab

+ Yalmip + Gurobi

助教 董九舟

清华大学电机系

djz23@mails.tsinghua.edu.cn

MATLAB 优化工具箱

MATLAB 优化工具箱

➤ Matlab 的安装

学校提供Matlab正版软件：

1. 登录清华大学信息化用户服务平台
(<https://its.tsinghua.edu.cn/>)
2. → 公共软件 → 计算类 → Matlab
3. 下载“Matlab安装手册及所有相关文件”，按照引导进行安装

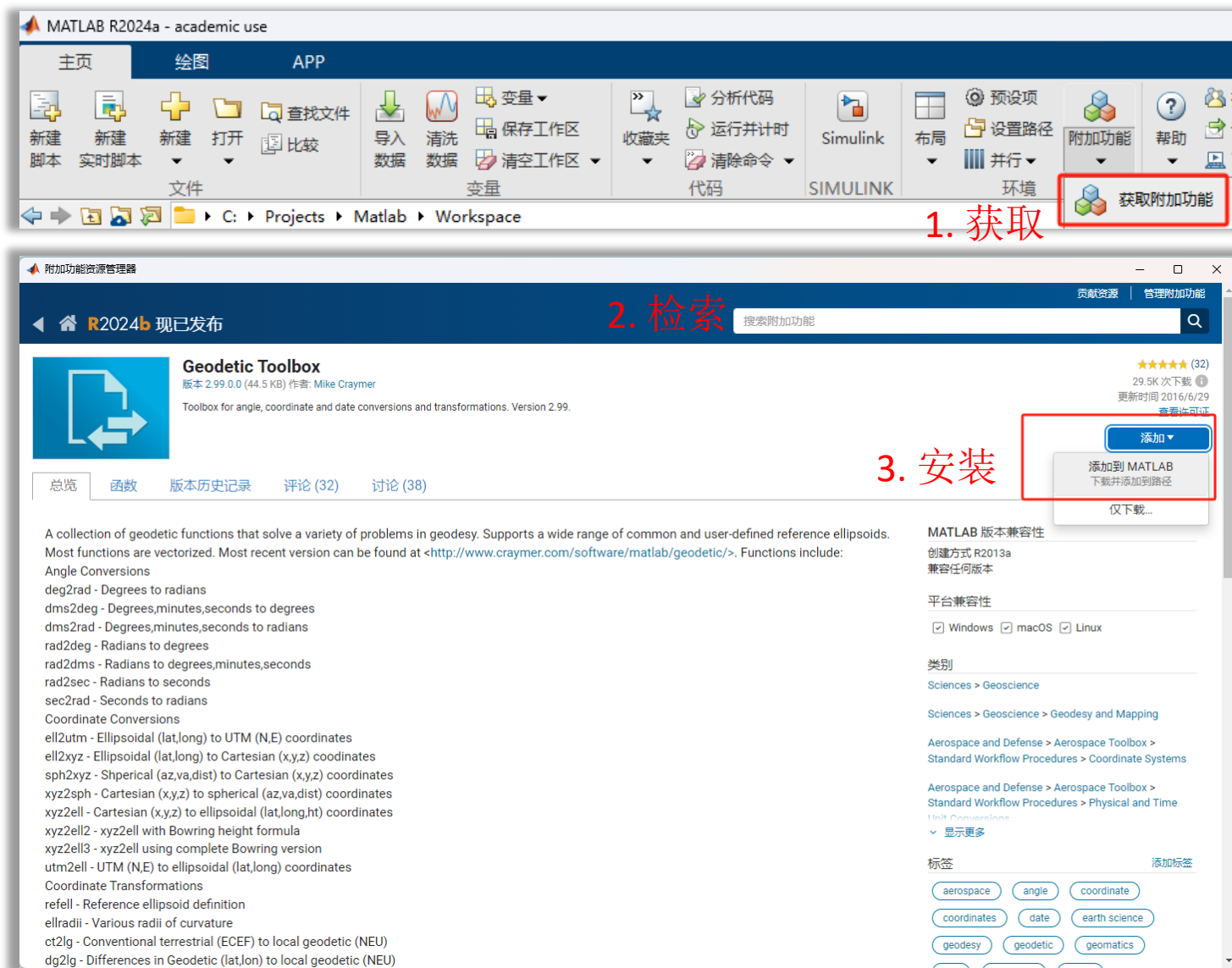


MATLAB 优化工具箱

➤ 优化工具箱简介

Matlab自身可以求解优化问题，需要使用Optimization Toolbox，可在安装Matlab时勾选此工具箱或自行添加

1. → 附加功能 → 获取附加功能
2. 检索“Optimization Toolbox”，点击“添加到MATLAB”（如右图）



MATLAB 优化工具箱

➤ 优化工具箱简介

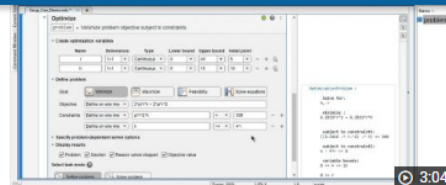
优化工具箱的使用详见官方说明

<https://cn.mathworks.com/products/optimization/features.html>

具体函数的参数含义、功能说明、使用示例等可在官方帮助文档中学习

<https://ww2.mathworks.cn/help/index.html>

Optimization Toolbox

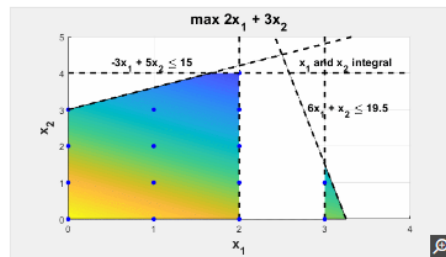


定义优化问题

将设计或决策问题建模为优化问题。将设计参数或决策方案设置为优化变量。使用变量来定义要优化的目标函数，并使用约束来限制可能的变量值。

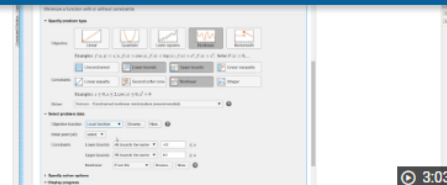
🕒 数学建模优化，第 2a 部分：基于问题的线性规划 (6:04)

文档 | 示例



线性和混合整数线性规划

求解具有连续和/或整数变量线性约束的线性目标的优化问题。

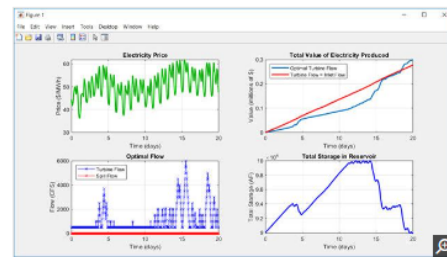


求解优化问题

应用求解器处理优化问题，确定最优解，即一组优化变量值，可实现目标函数最优值（如有）且满足约束（如有）。

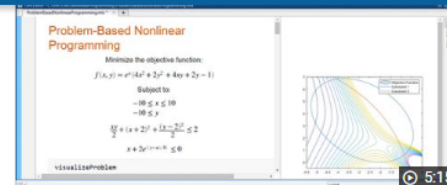
🕒 数学建模优化，第 2 部分：基于求解器的线性规划 (10:46)

文档 | 示例



二次和锥规划

求解具有二次目标和线性约束的优化问题或具有二阶锥约束的问题。

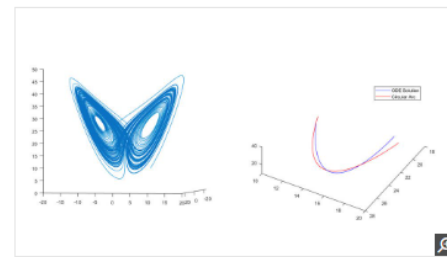


非线性规划

求解具有非线性目标或非线性约束的优化问题。

📖 使用基于问题的方法求解离散化最佳轨迹

文档 | 示例



最小二乘

求解具有边界、线性和非线性约束的线性和非线性最小二乘问题。



YALMIP + GUROBI

YALMIP + GUROBI

➤ 什么是 Yalmip、Gurobi?

- Yalmip 是一种优化模型**建模语言**
 - 是一个开源的 Matlab 库，提供了调用其他求解器的 API
- Gurobi是商业化的**优化求解器**
 - 商业求解器有多种，不同求解器“擅长”不同类型的优化问题
- Yalmip 与 Gurobi 的关系:



注: Yalmip本身不具有求解优化问题的能力!

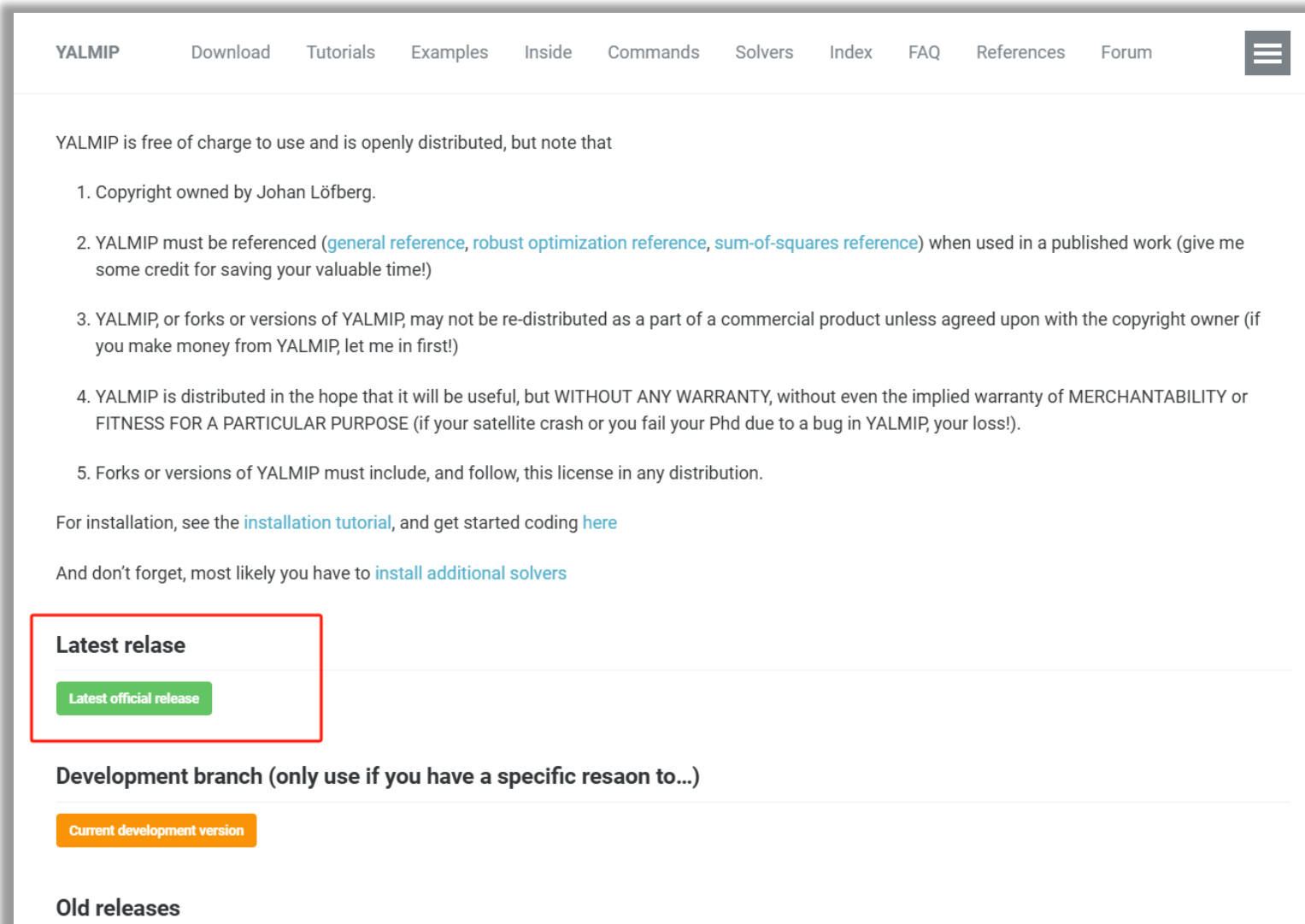
➤ 安装教程-Yalmip

1. Yalmip官网下载

<https://yalmip.github.io/download/>

（下载得到YALMIP-master.zip）

2. 将YALMIP-master.zip解压到合适的 路径下（例如MATLAB安装路径下 的toolbox文件夹，此步非必要）



The screenshot shows the YALMIP website's download page. The navigation bar at the top includes links for YALMIP, Download, Tutorials, Examples, Inside, Commands, Solvers, Index, FAQ, References, and Forum. The main content area states that YALMIP is free of charge and openly distributed, followed by five numbered points regarding copyright, referencing, redistribution, warranty, and license. Below this, there are links for installation and additional solvers. A red box highlights the 'Latest release' section, which contains a green button labeled 'Latest official release'. Below that is the 'Development branch' section with an orange button labeled 'Current development version'. The 'Old releases' section is also visible at the bottom.

YALMIP is free of charge to use and is openly distributed, but note that

1. Copyright owned by Johan Löfberg.
2. YALMIP must be referenced ([general reference](#), [robust optimization reference](#), [sum-of-squares reference](#)) when used in a published work (give me some credit for saving your valuable time!)
3. YALMIP, or forks or versions of YALMIP, may not be re-distributed as a part of a commercial product unless agreed upon with the copyright owner (if you make money from YALMIP, let me in first!)
4. YALMIP is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY, without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE (if your satellite crash or you fail your Phd due to a bug in YALMIP, your loss!).
5. Forks or versions of YALMIP must include, and follow, this license in any distribution.

For installation, see the [installation tutorial](#), and get started coding [here](#)

And don't forget, most likely you have to [install additional solvers](#)

Latest release

[Latest official release](#)

Development branch (only use if you have a specific reason to...)

[Current development version](#)

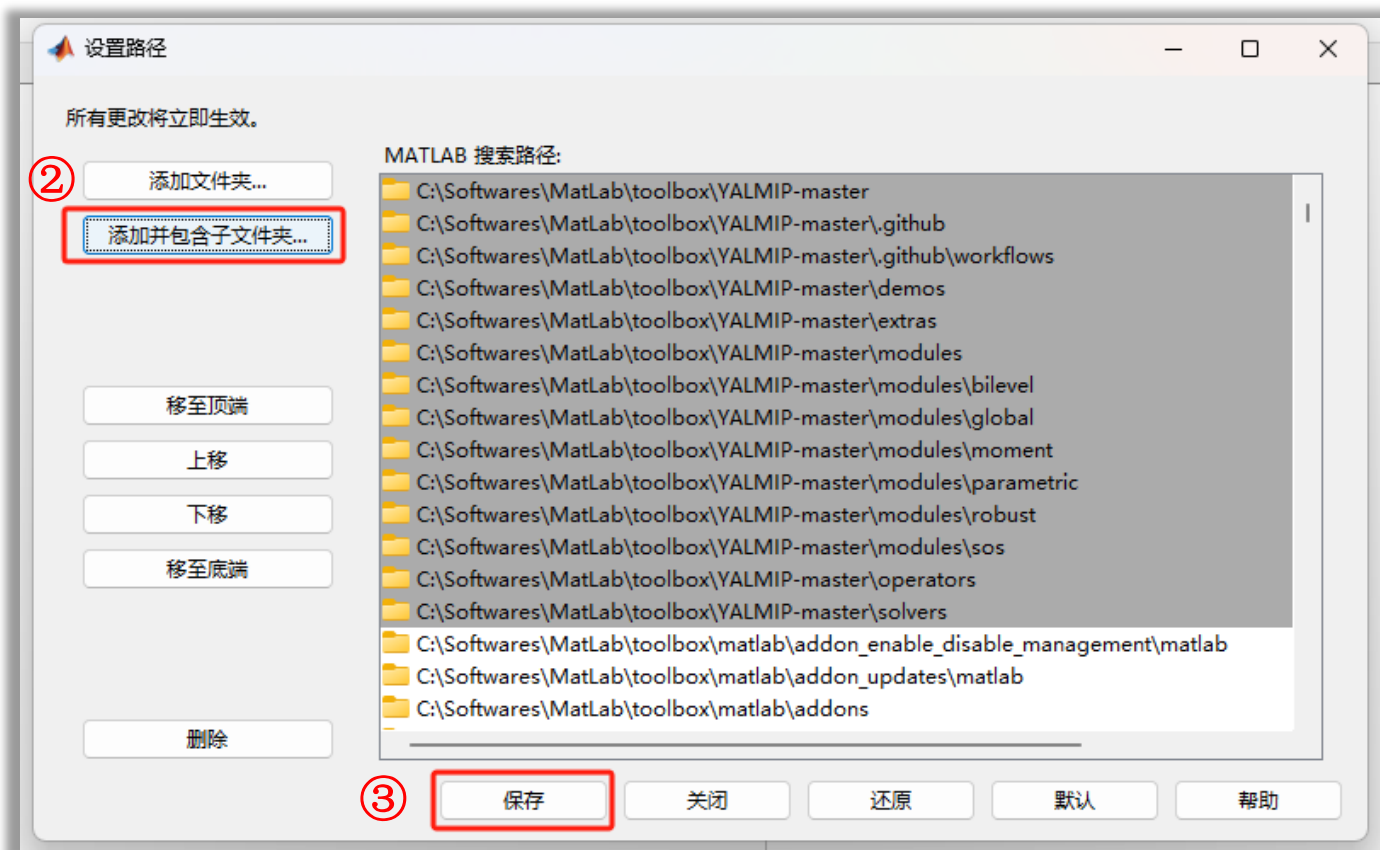
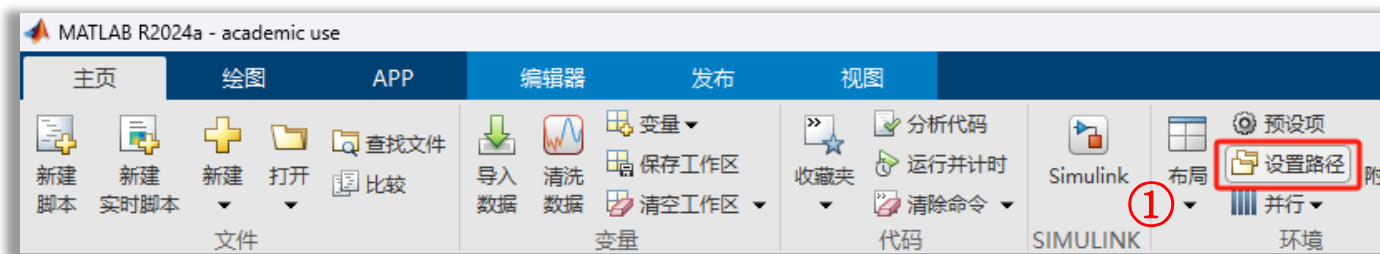
Old releases

YALMIP + GUROBI

➤ 安装教程-Yalmip

3. Matlab-主页 → 设置路径

4. “添加并包含子文件夹” → 选择解压的YALMIP-master → “保存”

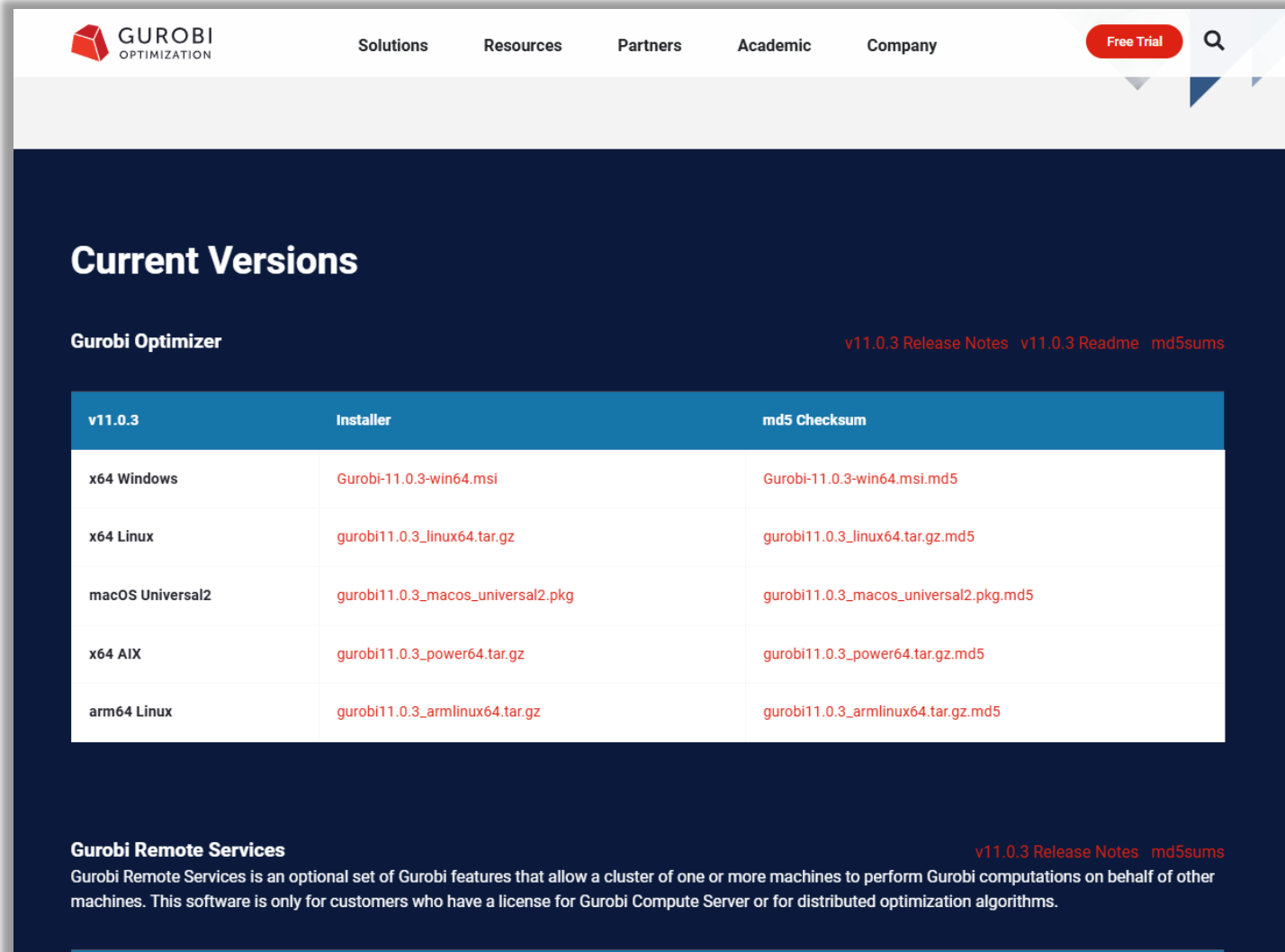


YALMIP + GUROBI

➤ 安装教程-Gurobi

进入官网选择合适的版本下载安装

<https://www.gurobi.com/downloads/gurobi-optimizer/>



The screenshot shows the Gurobi Optimization website. The top navigation bar includes links for Solutions, Resources, Partners, Academic, and Company, along with a 'Free Trial' button and a search icon. The main content area is titled 'Current Versions' and features a section for 'Gurobi Optimizer' v11.0.3. Below this, there is a table listing download links for various operating systems and architectures. The table has three columns: 'v11.0.3', 'Installer', and 'md5 Checksum'. The rows include x64 Windows, x64 Linux, macOS Universal2, x64 AIX, and arm64 Linux. Each row provides a specific installer file name and its corresponding md5 checksum.

v11.0.3	Installer	md5 Checksum
x64 Windows	Gurobi-11.0.3-win64.msi	Gurobi-11.0.3-win64.msi.md5
x64 Linux	gurobi11.0.3_linux64.tar.gz	gurobi11.0.3_linux64.tar.gz.md5
macOS Universal2	gurobi11.0.3_macos_universal2.pkg	gurobi11.0.3_macos_universal2.pkg.md5
x64 AIX	gurobi11.0.3_power64.tar.gz	gurobi11.0.3_power64.tar.gz.md5
arm64 Linux	gurobi11.0.3_armlinux64.tar.gz	gurobi11.0.3_armlinux64.tar.gz.md5

Below the table, there is a section for 'Gurobi Remote Services' with a brief description and links to release notes and md5sums.



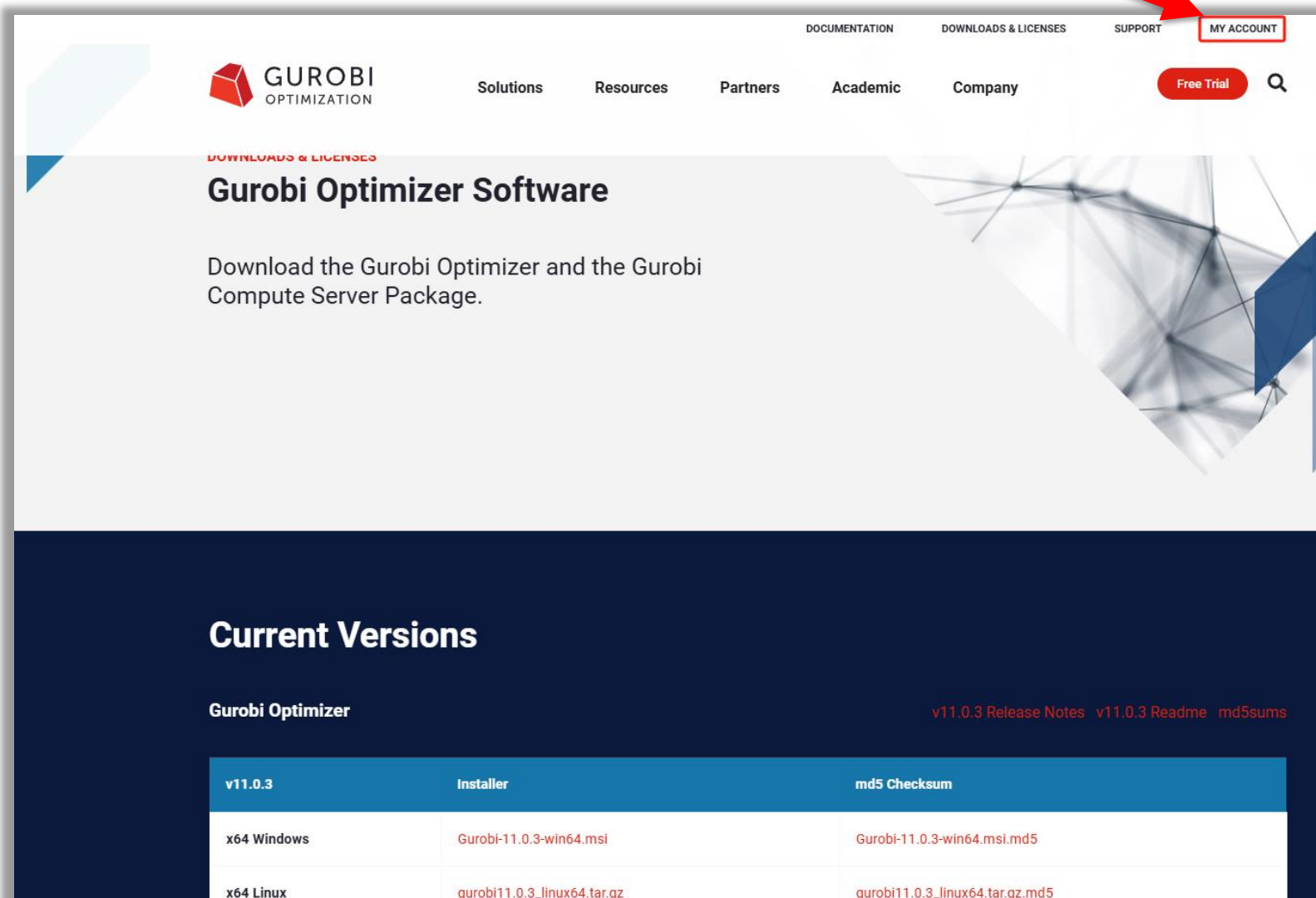
YALMIP + GUROBI

- 若未登录，则此处为REGISTER和LOGIN
- 登录后点击MY ACCOUNT进入个人页面

➤ 安装教程-Gurobi

申请license并激活

1. 登录或注册账号
2. 转到个人页面，点击 License → Request，申请“Named-User Academic”凭证



DOCUMENTATION DOWNLOADS & LICENSES SUPPORT **MY ACCOUNT**

Solutions Resources Partners Academic Company Free Trial

Gurobi Optimizer Software

Download the Gurobi Optimizer and the Gurobi Compute Server Package.

Current Versions

Gurobi Optimizer v11.0.3 Release Notes v11.0.3 Readme md5sums

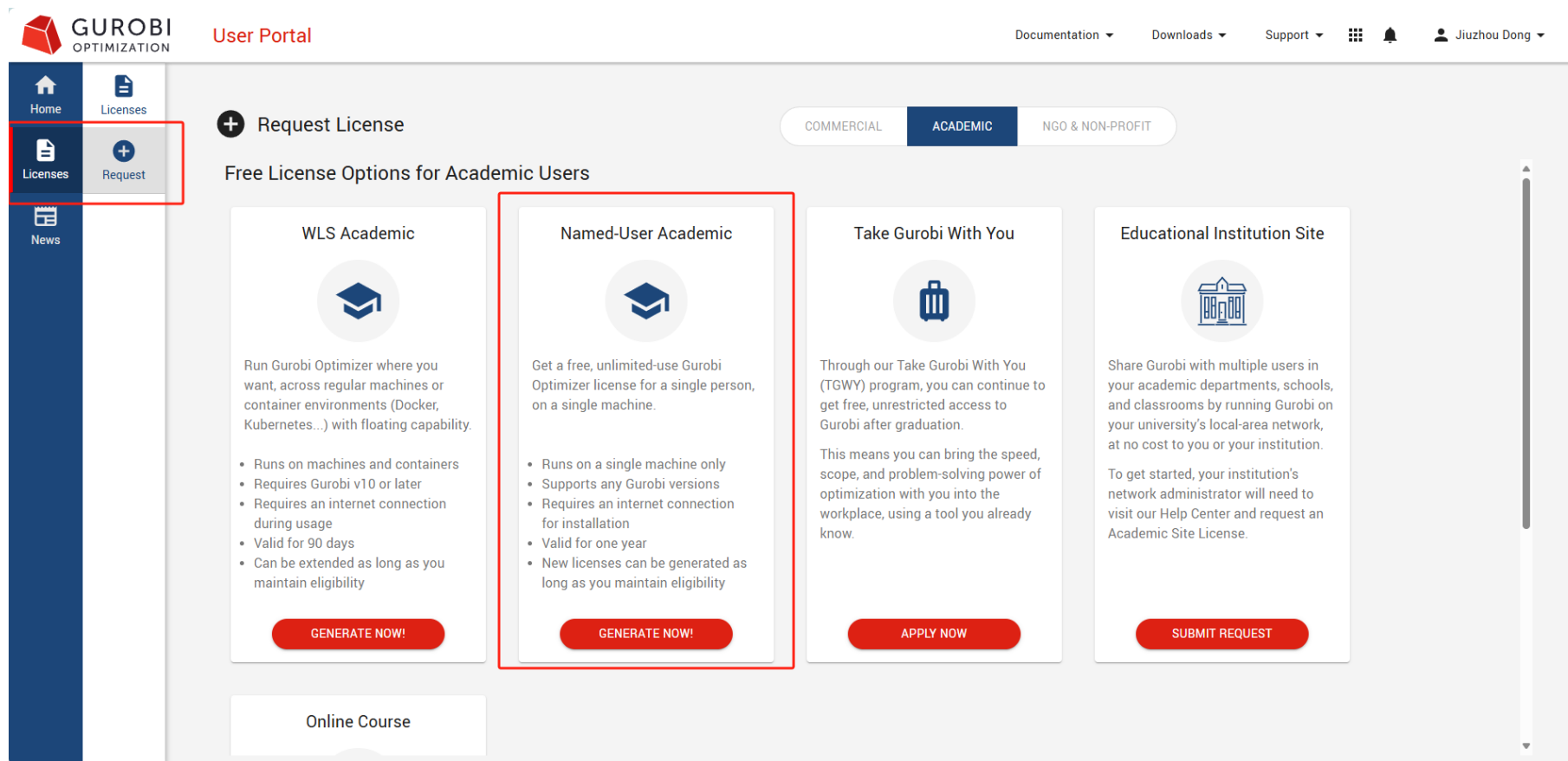
v11.0.3	Installer	md5 Checksum
x64 Windows	Gurobi-11.0.3-win64.msi	Gurobi-11.0.3-win64.msi.md5
x64 Linux	gurobi11.0.3_linux64.tar.gz	gurobi11.0.3_linux64.tar.gz.md5



YALMIP + GUROBI

➤ 安装教程-Gurobi

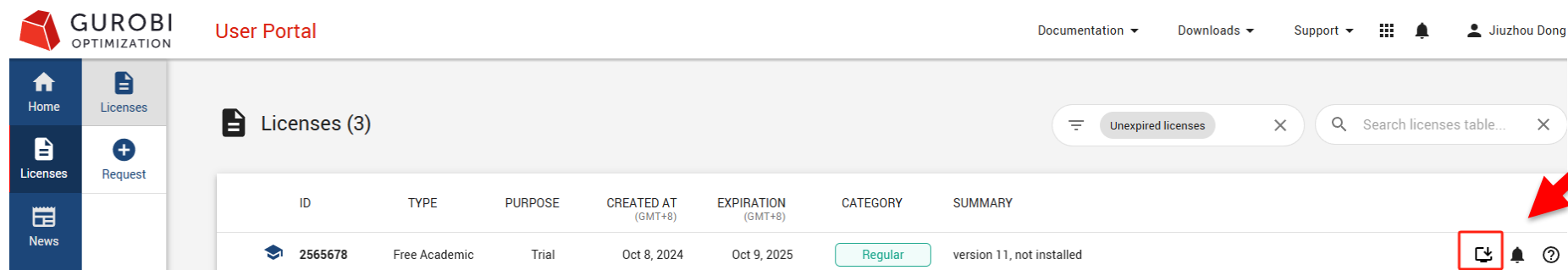
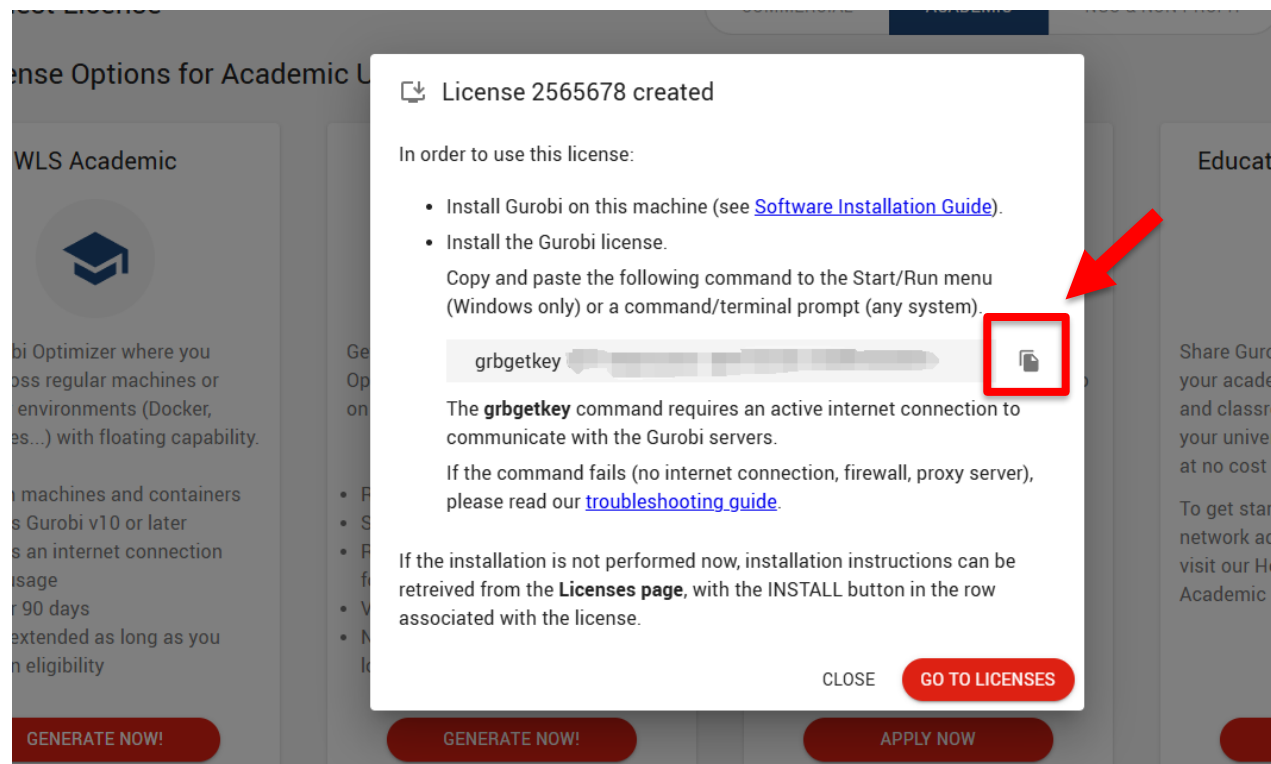
2. 转到个人页面，点击 License → Request，申请“Named-User Academic”凭证



YALMIP + GUROBI

➤ 安装教程-Gurobi

3. 申请成功可获得一个key，点击复制

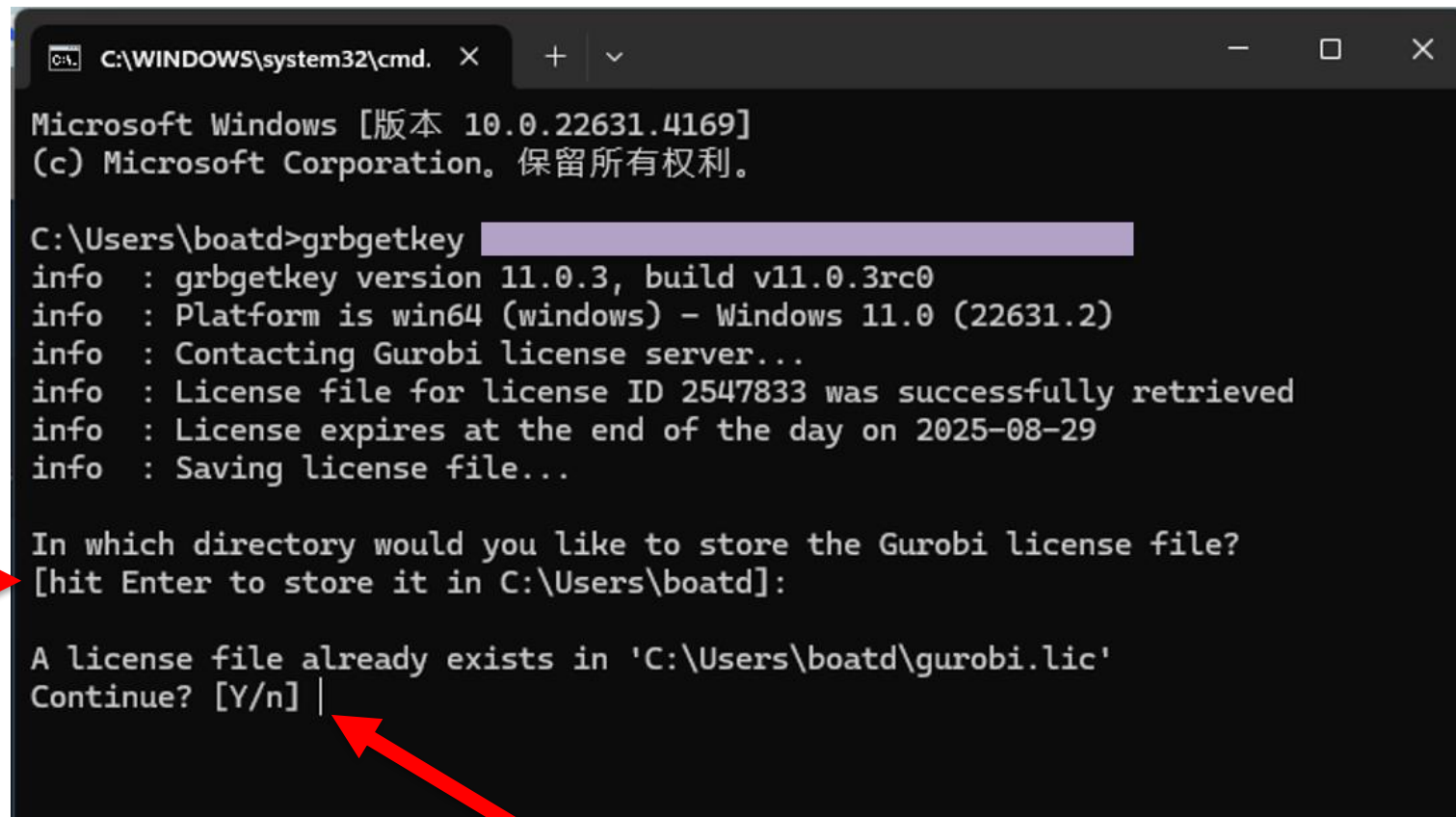


可在license中查看已申请的凭证

YALMIP + GUROBI

➤ 安装教程-Gurobi

4. 打开命令提示符（Win+R键入cmd而后回车），CTRL+V输入复制的key，回车激活
5. 完成安装和激活后，在Matlab中设置路径（Matlab-主页 → 设置路径 → “添加并包含子文件夹” → 选择<Gurobi根目录\win64\matlab> → “保存”



```
C:\WINDOWS\system32\cmd. X + v
Microsoft Windows [版本 10.0.22631.4169]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\boatd>grbgetkey
info : grbgetkey version 11.0.3, build v11.0.3rc0
info : Platform is win64 (windows) - Windows 11.0 (22631.2)
info : Contacting Gurobi license server...
info : License file for license ID 2547833 was successfully retrieved
info : License expires at the end of the day on 2025-08-29
info : Saving license file...

In which directory would you like to store the Gurobi license file?
[hit Enter to store it in C:\Users\boatd]:

A license file already exists in 'C:\Users\boatd\gurobi.lic'
Continue? [Y/n] |
```

按照提示操作即可，此处因设备已激活过而询问是否覆盖

YALMIP + GUROBI

➤ 安装教程-测试是否安装成功

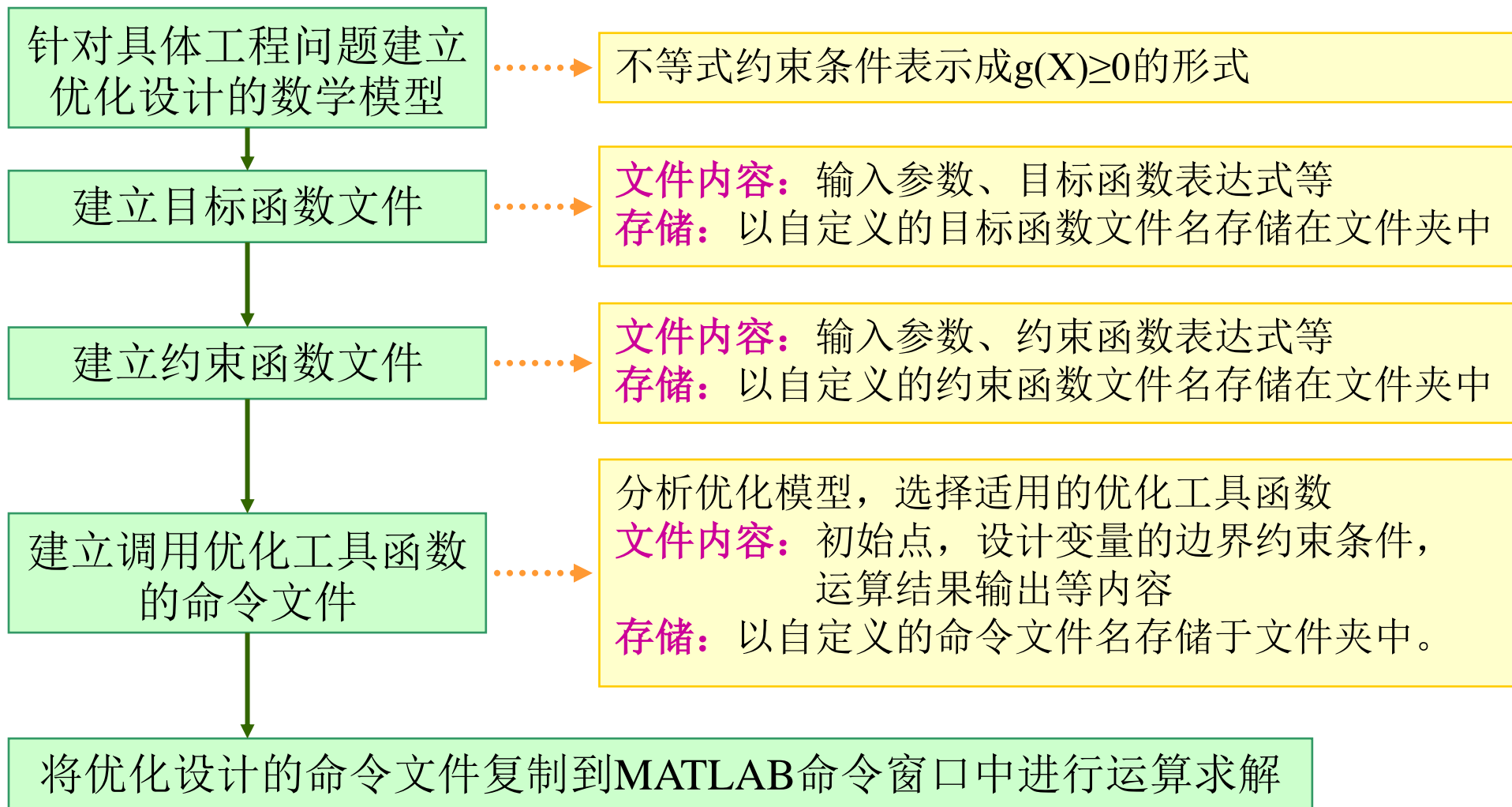
1. 在Matlab命令行键入“yalmiptest”，会显示已安装的求解器
2. 如右图所示，安装了Gurobi，但未安装Cplex
3. 可按任意键继续，完成测试

```
命令行窗口
①>> yalmiptest
+++++
| Searching for installed solvers |
+++++
| Solver| Version| Status|
+++++
| BISECTION| | internal|
| BMIBNB| | internal|
| BNB| | internal|
| CUTSDP| | internal|
| KKTQP| | internal|
| CONEPROG| | found|
| FMINCON| | found|
| FMINSEARCH| | found|
| GUROBI| | found|
| INTLINPROG| | found|
| LINPROG| | found|
| LMILAB| | found|
| CLP| | not found|
| COPT| | not found|
| CPLEX| | not found|
| CSDP| | not found|
| DAQP| | not found|
```

② 按任意键继续



优化问题一般步骤



使用入门

以 Yalmip 例程1进行说明（更多例程详见
<https://yalmip.github.io/tutorial/basics/>）

1. 创建决策变量
2. 添加约束条件
3. 定义目标函数
4. 设置求解参数并运行

```
test.m x +
1 % Define variables
2 x = sdpvar(10,1);
3
4 % Define constraints
5 Constraints = [sum(x) <= 10, x(1) == 0, 0.5 <= x(2) <= 1.5];
6 for i = 1 : 7
7     Constraints = [Constraints, x(i) + x(i+1) <= x(i+2) + x(i+3)];
8 end
9
10 % Define an objective
11 Objective = x'*x+norm(x,1);
12
13 % Set some options for YALMIP and solver
14 options = sdpsettings('verbose',1,'solver','quadprog','quadprog.maxiter',100);
15
16 % Solve the problem
17 sol = optimize(Constraints,Objective,options);
18
19 % Analyze error flags
20 if sol.problem == 0
21     % Extract and display value
22     solution = value(x);
23 else
24     disp('Hmm, something went wrong!');
25     sol.info
26     yalmiperror(sol.problem)
27 end
```



使用入门-创建决策变量

主要使用三种决策变量:

- 实数型变量-sdpvar
- 整数型变量-intvar
- 0-1变量-binvar

创建变量可指定变量的维度:

- 标量: `a = sdpvar(1)`
- 向量: `b = sdpvar(1,3); c = sdpvar(2,1);`
- 矩阵: `d = sdpvar(3,4); e = sdpvar(2,2);`

可将Matlab内置的函数应用于决策变量, 例如:

1. 生成对角阵: `D = diag(b);`
2. 进行矩阵(向量运算): `f = c*b;` % 得到2x3矩阵
3. 元素求和: `S = sum(d, 'all');`
4.

1. 效果与“`e = sdpvar(2)`”相同
2. 当创建方阵时, 默认为对称矩阵, 若要创建非对称的矩阵则用“`e = sdpvar(2,2,'full')`”

使用入门

以 Yalmip 例程1进行说明（更多例程详见
<https://yalmip.github.io/tutorial/basics/>）

1. 创建决策变量
2. 添加约束条件
3. 定义目标函数
4. 设置求解参数并运行

注意：等式约束要用“==”符号

```
test.m x +
1 % Define variables
2 x = sdpvar(10,1);
3
4 % Define constraints
5 Constraints = [sum(x) <= 10, x(1) == 0, 0.5 <= x(2) <= 1.5];
6 for i = 1 : 7
7     Constraints = [Constraints, x(i) + x(i+1) <= x(i+2) + x(i+3)];
8 end
9
10 % Define an objective
11 Objective = x'*x+norm(x,1);
12
13 % Set some options for YALMIP and solver
14 options = sdpsettings('verbose',1,'solver','quadprog','quadprog.maxiter',100);
15
16 % Solve the problem
17 sol = optimize(Constraints,Objective,options);
18
19 % Analyze error flags
20 if sol.problem == 0
21     % Extract and display value
22     solution = value(x);
23 else
24     disp('Hmm, something went wrong!');
25     sol.info
26     yalmiperror(sol.problem)
27 end
```



使用入门

以 Yalmip 例程1进行说明（更多例程详见
<https://yalmip.github.io/tutorial/basics/>）

1. 创建决策变量
2. 添加约束条件
3. 定义目标函数
4. 设置求解参数并运行

注意：Yalmip默认问题为最小化问题

```
test.m x +
1 % Define variables
2 x = sdpvar(10,1);
3
4 % Define constraints
5 Constraints = [sum(x) <= 10, x(1) == 0, 0.5 <= x(2) <= 1.5];
6 for i = 1 : 7
7     Constraints = [Constraints, x(i) + x(i+1) <= x(i+2) + x(i+3)];
8 end
9
10 % Define an objective
11 Objective = x'*x+norm(x,1);
12
13 % Set some options for YALMIP and solver
14 options = sdpsettings('verbose',1,'solver','quadprog','quadprog.maxiter',100);
15
16 % Solve the problem
17 sol = optimize(Constraints,Objective,options);
18
19 % Analyze error flags
20 if sol.problem == 0
21     % Extract and display value
22     solution = value(x);
23 else
24     disp('Hmm, something went wrong!');
25     sol.info
26     yalmiperror(sol.problem)
27 end
```



使用入门

以 Yalmip 例程1进行说明（更多例程详见
<https://yalmip.github.io/tutorial/basics/>）

1. 创建决策变量
2. 添加约束条件
3. 定义目标函数
4. 设置求解参数并运行

注意：主要设置求解器域'solver'，
其他选项含义可在Matlab命令行中
输入“**doc sdpsettings**”查看

```
test.m x +
1 % Define variables
2 x = sdpvar(10,1);
3
4 % Define constraints
5 Constraints = [sum(x) <= 10, x(1) == 0, 0.5 <= x(2) <= 1.5];
6 for i = 1 : 7
7     Constraints = [Constraints, x(i) + x(i+1) <= x(i+2) + x(i+3)];
8 end
9
10 % Define an objective
11 Objective = x'*x+norm(x,1);
12
13 % Set some options for YALMIP and solver
14 options = sdpsettings('verbose',1,'solver','quadprog','quadprog.maxiter',100);
15
16 % Solve the problem
17 sol = optimize(Constraints,Objective,options);
18
19 % Analyze error flags
20 if sol.problem == 0
21     % Extract and display value
22     solution = value(x);
23 else
24     disp('Hmm, something went wrong!');
25     sol.info
26     yalmiperror(sol.problem)
27 end
```

求解函数的调用，三个输入参数
分别为约束、目标、求解参数



使用入门

其他常用函数：

- 检查约束条件是否起作用（余量大小）：`check`
- 查看变量或表达式的值：`value`
- 给变量赋值（一般用于调试）：`assign`

Yalmip 机组组合例程

- <https://yalmip.github.io/example/unitcommitment/>