

# 如何运行当前版本 CSST-MCI 仿真代码

## 目录

- 代码获取 .....2
- 安装.....2
- 安装后的文件说明： .....3
- .config 配置文件说明.....4
- 仿真用到的数据文件 .....6
- MCI 仪器仿真软件使用样例 .....8

## 代码获取

从国台 gitlab 服务器上拉取 CSST 仿真代码：链接：<https://csst-tb.bao.ac.cn/code/shaosim/mci>

打包下载：`csst_mci_sim-develop.tar.gz`

程序运行需要用到的数据库获取链接（百度云盘）：通过网盘分享的文件：**MCI\_inputdata**

链接：<https://pan.baidu.com/s/1hj4YShnE0lASC3MHgzgwTA?pwd=ayjx> 提取码：ayjx

## 安装

软硬件环境要求：

内存：推荐 16GB 以上

存储：依赖数据文件 1GB 左右（可以和本体放在不同目录）

软件环境要求：Ubuntu 18.04 以上，Python 3.11

第三方包依赖参见 requirements.txt

安装说明：

推荐先安装 anaconda

拷贝安装包（包括本体和数据文件），进入目录：`cd ~/your_dir/`

检查软件依赖是否满足： `pip install -r requirements.txt`











`galsim` 的安装建议使用如下命令： `conda install -c conda-forge galsim`

（推荐）新建安装目录：`mkdir /the/dir/your/want/to/install/`

执行如下的安装命令： `pip install csst_mci_sim-develop.tar.gz`

## 安装后的文件说明：

安装包文件夹下的子文件夹如下图所示，

 CTI	update
 help	debug
 mci_data	Replace mci_C10.config
 mci_so	update
 support	update
 <code>_init_.py</code>	build
 <code>csst_mci_sim.py</code>	update
 <code>earthshine.py</code>	update
 <code>straylight.py</code>	update
 <code>zodiacal.py</code>	update

- ✧ **CTI**: CTI 效应模块
- ✧ **mci\_data**: 软件需要用到的数据及提供的默认.config 配置文件存放目录
- ✧ **mci\_so**: 仿真中需要用到的\*.so 文件
- ✧ **support**: 仿真中需要用到的 support 模块
- ✧ **csst\_mci\_sim.py**: 仿真主程序

## .config 配置文件说明

.config 文件中的参数列表及说明如下:

- **starcats**: 恒星星表输入的文件名, 需存放在 MCI\_inputData/star\_input/文件夹下
- **ver**: 内部迭代控制版本号
- **xsize**: 图像列大小, 默认值为 9216, 请勿修改。
- **ysize**: 图像行大小, 默认值为 9232, 请勿修改。
- **prescan**: prescan 大小, 请勿修改
- **overscan**: overscan 大小, 请勿修改
- **parallelTrapfile**: CTI 效应输入数据文件路径
- **serialTrapfile**: CTI 效应输入数据文件路径
- **cosmeticsFile\_g**: 紫外探测器热像素、坏像素输入数据文件路径
- **cosmeticsFile\_r**: 可见光探测器热像素、坏像素输入数据文件路径
- **cosmeticsFile\_i**: 近红外探测器热像素、坏像素输入数据文件路径
- **cosmicraylengths**: 宇宙线数据文件路径

- **cosmicraydistance**: 宇宙线数据文件路径
- **cosmicRays**: 宇宙线效应开关
- **darknoise**: 暗电流效应开关
- **cosmetics**: 热像素、坏像素开关
- **radiationDamage**: CTI 效应开关
- **bleeding**: 饱和溢出效应开关
- **overscans**: 虚拟读出效应开关
- **nonlinearity**: 非线性效应开关
- **readoutNoise**: 读出噪声效应开关
- **skyback**: 天光背景、杂散光效应开关
- **TianceEffect**: 天测效应开关
- **intscale**: 量化输出开关
- **ghosts**: 鬼像效应开关
- **shutterEffect**: 快门效应开关
- **flatfieldM**: 大尺度平场效应开关
- **PRNUeffect**: 像素非均匀响应效应开关
- **appFatt**: 亮胖效应开关
- **sky\_shift\_rot**: dither 模式开关
- **distortion**: 像场畸变开关
- **sim\_star**: 是否启动恒星观测模拟开关
- **sim\_galaxy**: 是否启动星系观测模拟开关
- **save\_starpsf**: 保存 psf 数据开关
- **save\_cosmicrays**: 保存宇宙线数据开关
- **lensing**: 引力透镜效应开关
- **fullwellcapacity**: 满阱电子数, 请勿修改。
- **dark**: [float]探测器暗电流, 单位:e/s/pixel
- **exptime**: 曝光时间, 单位 s
- **flatsigma**: 大尺度平场矩阵参数
- **coveringFraction**: 宇宙线覆盖率百分比

- **ghostOffsetX**: 模拟鬼像效应时鬼像在 x 方向的偏移量
- **ghostOffsetY**: 模拟鬼像效应时鬼像在 y 方向的偏移量
- **ghostRatio**: 模拟鬼像效应时鬼像强度与原图像强度之比
- **treering**: 年轮效应参数
- **fatter**: 亮胖效应参数
- **difusion**: 电荷扩散效应参数
- **filter\_g**: 紫外通道滤光片名称
- **filter\_r**: 可见光通道滤光片名称
- **filter\_i**: 近红外通道滤光片名称
- **g\_gain1~16**: 紫外通道探测器第一~第十六个读出门的增益, 单位: e/ ADU
- **g\_rdnos1~16**: 紫外通道探测器第一~第十六个读出门的读出噪声, 单位:e/pixel
- **g\_detbia1~16**: 紫外通道探测器第一~第十六个读出门的 bias, 单位:ADU
- **r\_gain1~16**: 可见光通道探测器第一~第十六个读出门的增益, 单位: e/ ADU
- **r\_rdnos1~16**: 可见光通道探测器第一~第十六个读出门的读出噪声, 单位:e/pixel
- **r\_detbia1~16**: 可见光通道探测器第一~第十六个读出门的 bias, 单位:ADU
- **i\_gain1~16**: 近红外通道探测器第一~第十六个读出门的增益, 单位: e/ ADU
- **i\_rdnos1~16**: 近红外通道探测器第一~第十六个读出门的读出噪声, 单位:e/pixel
- **i\_detbia1~16**: 近红外通道探测器第一~第十六个读出门的 bias, 单位:ADU

## 仿真用到的数据文件:

仿真软件需要用到数据存放在 MCI\_inputdata 文件夹, 其下的文件如下图所示。

Name	Size (KB)
..	
configData	
star_input	
refs	
galaxy_Input	
PSF	
TianCe	
SED_Code	
MCI_filters	
tel_eff	
data	

- ✧ **configData:** 存放.config 配置文件
- ✧ **star\_input:** 存放恒星星表文件
- ✧ **refs:** 存放的模拟天光背景用到的数据
- ✧ **galaxy\_Input:** 存放星系星表图像及 SED 文件
- ✧ **PSF:** 存放各个滤光片七个波段的 PSF 数据文件
- ✧ **TianCe:** 存放的望远镜轨道参数模拟数据
- ✧ **SED\_Code:** SED 计算程序及数据文件
- ✧ **MCI\_filters:** 滤光片数据文件
- ✧ **tel\_eff:** 望远镜效率曲线文件
- ✧ **data:** 存放热像素、坏像素、宇宙线生成、CTI 效应等用到的数据文件

# MCI 仪器仿真软件使用样例

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  """
4  This is an example for mci simulation.
5
6  @author: yan
7  """
8  import os
9  from csst_mci_sim import csst_mci_sim
10
11  # Set the parent path where the MCI_inputdata folder is stored.
12  dir_path = 'home/yan/'
13
14  # Set the path to the called config file.
15  configfile = os.path.join(dir_path, 'MCI_inputData/configData/mci_C10.config')
16
17  # Set the path to the simulation data storage file.
18  result_path = os.path.join(dir_path, 'mci_sim_result')
19
20  sourcein = 'STAR' # or EXDF or FLAT or DARK or BIAS
21
22  # Specifies whether the debug mode is used. If debug=True,
23  # this mode is used and the program will only simulate a few wavelength slices
24  # to save time. If debug=False, the full simulation will be run.
25  debug = False
26
27  # set the simulaiton number. With modifications, parallel simulations
28  # can be executed.
29  iloop = 1
30
31  # run the main function
32  # When the last parameter is yes and sourcein is LAMP,
33  # the Hole simulation is executed.
34  csst_mci_sim.runMCIsim(sourcein, configfile, dir_path,
35  .....result_path, debug, iloop)
```