**工业流水线智能规划排队问题**

**仿真软件使用说明**

1. **文件目录**

Simulator：软件运行根目录

Alg.py：控制算法

config\_\*.yml：配置文件

Configuration.py：配置类

gui.py：GUI程序

info.py：仿真软件运行状态信息类

InfoLock.py：数据同步锁

JudgeOverlap.py：包裹碰撞检测模块

JudgeRunOver.py：包裹穿越检测模块

main.py：主函数，仿真软件启动入口

Parcel.py：包裹类

RandomParcelsGenerator：随机包裹生成模块

requirements.txt：软件依赖环境

RotatedRectangle.py：旋转矩形类

Sim.py：仿真软件主类

Singleton.py：单例函数

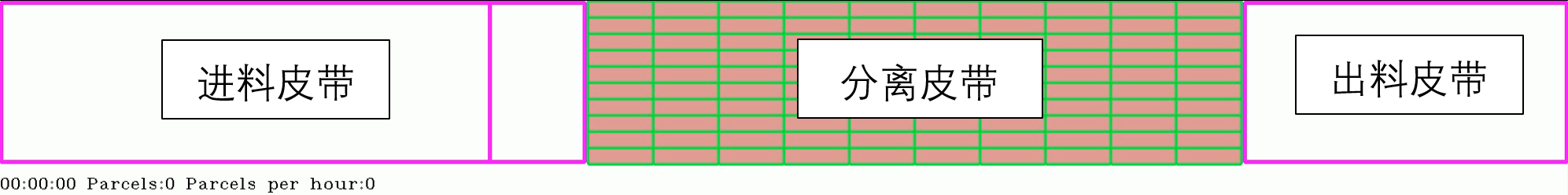
gui.bat：win环境启动gui脚本

main.bat：win环境启动仿真软件脚本

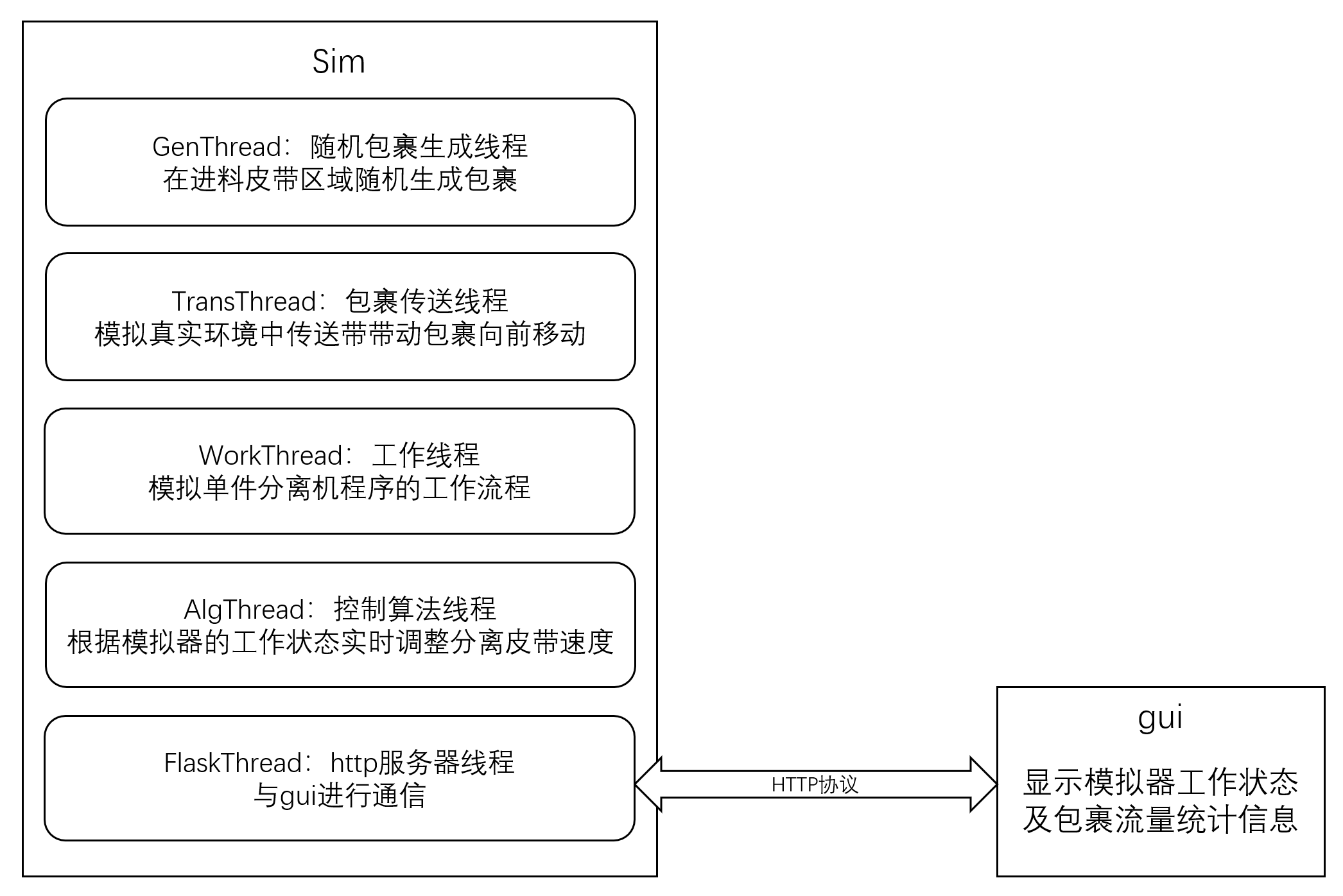
start\_scripts\_win.bat：win环境启动所有程序脚本

start\_scripts\_linux.sh：linux环境启动所有程序脚本

1. **仿真软件**
2. 软件框架

****

皮带布局示意图



软件框架示意图

1. 软件环境
2. 安装Python 3.8 + pip3
3. 控制台进入仿真软件代码根目录，运行如下指令安装python依赖库：

pip3 install -r requirements.txt

1. 启动程序

* Windows环境：

控制台进入仿真软件代码根目录，执行命令：

start\_scripts\_win.bat [CONFIG\_PATH]

例如：start\_scripts\_win.bat config\_default.yml

* Linux环境：

终端进入仿真软件代码根目录，执行命令：

chmod a+x ./start\_scripts\_linux.sh

start\_scripts\_linxu.sh [CONFIG\_PATH]

例如：start\_scripts\_linux.sh config\_default.yml

config\_default.yml为用于演示仿真软件功能的配置文件，参赛选手需要设计控制算法尽可能满足在config\_6x8.yml，config\_7x10.yml，config\_9x10.yml这三种不同配置环境下的包裹分离需求。

* GUI程序:

启动模拟器程序后运行gui.py即可查看模拟器的工作状态和包裹流量统计信息。在gui界面，按下’q’键可以保存模拟器运行画面并退出gui程序和模拟器程序。

1. 错误状态说明
2. 出双doubleout：任意时刻有超过一个以上的包裹的旋转矩形框横跨分离皮带与出料皮带的交界处。
3. 碰撞overlap、穿越runover：任意时刻任意两个包裹的旋转矩形框出现重叠。
4. 输出间距过小：在分离程序运行过程中，前一个包裹完全离开分离皮带（包裹完全进入出料皮带）的时刻为t1，后一个包裹开始离开分离皮带（包裹开始进入出料皮带）的时刻为t2，两者的时间差Δt=t2-t1，Δt必须大于配置文件设定好的阈值，也即先后离开分离皮带的包裹之间应在x方向有一定的间距。
5. **编写控制算法**

参赛选手自行设计控制算法，对Alg.py中的alg函数进行任意改写，注意不要修改函数的输入输出接口。alg函数接口说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 输入 | 参数 | 说明 | 类型 |
| parcels | 当前处于分离皮带上的包裹列表[parcel0,parcel1,parcel2, ... ]，parcel为Parcel类，包含的包裹状态信息参看Parcel类。 | List |
| cur\_speeds | 当前分离皮带的运行速度列表[s00, s01, s02,...]，顺序：先沿皮带前进方向从左到右，再沿垂直于皮带前进方向从上到下。 | List |
| config | 当前模拟器的配置信息 | SeperateConfig |
| 输出 | speeds\_cache | 经过控制算法计算后得到的各个分离皮带的预期运行速度列表[s00, s01, s02,...]，顺序：先沿皮带前进方向从左到右，再沿垂直于皮带前进方向从上到下 | List |