

主函数

function [ ] = main\_tca( ~ )

%---------------------------------------------------------------------------------------------------------------------------------------------------------------

%%

%

%

%

%

%% %

%-------------------------------------------------------------------------%

%基础变量声明~

%-------------------------------------------------------------------------%

%%

%% -------------------------------------------------------------------------%

%基本仿真信息（或以配置文件方式直接获得）

%-------------------------------------------------------------------------%

[Volume SimTime TimeStep LaneNum CellNum VMAX] = GetBasicInfo();

%% -------------------------------------------------------------------------%

%生成元胞空间

%-------------------------------------------------------------------------%

[CellSpace\_current CellSpace\_nextstep] = GenerateCellSpace(LaneNum,CellNum);

%% -------------------------------------------------------------------------%

%生成车辆空间

%-------------------------------------------------------------------------%

[VehicleSpace] = GenerateVehicleSpace(Volume);

%% -------------------------------------------------------------------------%

%开始仿真

%-------------------------------------------------------------------------%

TrafficSimulating(SimTime,TimeStep, CellSpace\_current, CellSpace\_nextstep, VehicleSpace,VMAX);

%% -------------------------------------------------------------------------%

%展示结果

ResultsPlotting();

%-------------------------------------------------------------------------%

end

function [ CellSpace\_nextstep VehicleSpace ] = TrafficSimulating( SimTime,TimeStep,CellSpace\_current, CellSpace\_nextstep, VehicleSpace,VMAX )

%TRAFFICSIMULATING Summary of this function goes here

% 仿真程序主体

CellSpace\_Init = CellSpace\_nextstep;

% 读取信号配时数据

SignalCycleMat = load('SignalCycleInfo.ini');

sCycle = SignalCycleMat(1);%周期长度

sGreenTime = SignalCycleMat(2);%绿灯时长

sRedTime = SignalCycleMat(3);%红灯时长

%% 是否加载换道模型

LaneChangingModelINIMat = load('LaneChangingModeInfo.ini');

UseLaneChangingModelFlag = LaneChangingModelINIMat(1);

%%

%时间步长和时长必须满足整除关系

for iStep = 1:TimeStep:SimTime

%判断信号配时

tCurrentMod = mod(iStep,sCycle);

if tCurrentMod <= sGreenTime

signalFlag = 0;

else

signalFlag = 1;

end;

%搜寻元胞上存在的车辆，获得每个元胞的车辆占用情况，以及占用车辆的编号

[TakenMat TakenCarHeadMat TakenCarIDMat] = SearchCellSpaceForTaken(CellSpace\_current);

%计算每个车辆的更新信息

[CellSpace\_nextstep VehicleSpace] = CountingNextStepVehicleState(CellSpace\_current,CellSpace\_nextstep,VehicleSpace,TakenMat, TakenCarIDMat,VMAX,signalFlag,UseLaneChangingModelFlag);

%判断是否发新车，并计算新车的更新信息

%更新元胞空间

[CellSpace\_nextstep VehicleSpace] = AddingNewCarsToEntry(CellSpace\_current,CellSpace\_nextstep,VehicleSpace);

%

%统计相关信息

%保持数据

SavingDataAsFiles(CellSpace\_current,VehicleSpace,TakenMat,TakenCarHeadMat,TakenCarIDMat);

%绘制图像

DrawingCellSpace(TakenMat);

%进入下一轮循环

CellSpace\_current = CellSpace\_nextstep;

CellSpace\_nextstep = CellSpace\_Init;

end;

end

function istaken = IsCellTaken(CellSpace,laneid,startTag,endTag)

%从startTag开始到endTag结束的这些元胞是否都是空闲，有一个及以上被占用返回1

for i = startTag:1:endTag

if GetCellTakenState(CellSpace,laneid,i) == 1

istaken = 1;

return;

end;

end;

istaken = 0;%这些元胞全部空闲

end

function ResultsPlotting()

%% 读取配置信息

SavingSettingsMat = load('OutputDataSettings.ini');

%输出数据子文件夹

subfoldername = SavingSettingsMat(1);

%%

lanenodatafile = [num2str(subfoldername) '\车辆占用车道数据.txt'];

positiondatafile = [num2str(subfoldername) '\车辆所处位置.txt'];

speeddatafile = [num2str(subfoldername) '\车辆每一时刻速度.txt'];

cartypeinfofile = [num2str(subfoldername) '\车辆基本信息.txt'];

%%

LaneTakenDMat = importdata(lanenodatafile);

VehPositionDMat = importdata(positiondatafile);

SpeedDMat = importdata(speeddatafile);

VehicleInfoMat = load(cartypeinfofile);

%%

[stepNum, carNum] = size(SpeedDMat);

[Volume SimTime TimeStep LaneNum CellNum VMAX] = GetBasicInfo();

%%

travelTime = zeros(1,carNum);%每辆车的旅行时间

for i = 1:carNum

travelTime(1,i) = length(find(SpeedDMat(:,i)>=0));

end;

%%

meanSpeed = zeros(stepNum,1);%平均速度

meanQueue = zeros(stepNum,1);%平均排队长度

volDensity = zeros(stepNum,1);%密度

for j = 1:stepNum

existedCarIDs = find(SpeedDMat(j,:)>=0);

volDensity(j,1) = sum(GetVehicleLength(existedCarIDs)) / (LaneNum \* CellNum);

meanQueue(j,1) = sum(GetVehicleLength(find(SpeedDMat(j,:)<=1)))/LaneNum;

meanSpeed(j,1) = mean(SpeedDMat(j,SpeedDMat(j,:)>=0));

end;

%%

%%

%%

%%

end