**实验五 特殊比赛模式的设计**

**一、实验目的**

1.掌握 Robocup 仿真机器人足球比赛中特殊比赛模式发生的条件；

2.掌握 Robocup 仿真机器人足球比赛特殊比赛模式的规则要求；

3.了解 Robocup 仿真机器人足球比赛特殊比赛模式的战术设计思想；进一步熟悉 WorldModel 类。

**二、实验设备**

硬件环境：PC机；

软件环境：虚拟机

**三、实验内容**

**1.角球。最近的球员开球，次近的球员跑到接应点。**

角球是要求向场内踢。编写pointKickTo()返回在不同角时应该踢向的位置。

当球在自己半场时，将位置的X坐标调大，尽量向对方半场踢；而在对方半场时，X设置得比较小，为夺球攻门创造机会。

在basicplayer.cpp里添加：

VecPosition BasicPlayer::pointKickTo()//计算角球应该踢向的点

{

VecPosition posBall, posKickTo;

double x, y;

posBall = WM->getBallPos();

x = posBall.getX();

y = posBall.getY();

posKickTo.setX(- x / fabs(x) \* 5 + x );

posKickTo.setY(- y / fabs(y) \* 12 + y );

return posKickTo;

}

在PlayTeams.cpp里添加：

else if( WM->isCornerKickUs())//我方角球

{

if(WM->isBallKickable() )

{

soc = kickTo(WM->pointKickTo(), 2.0);

}

else if(WM->getFastestInSetTo( OBJECT\_SET\_TEAMMATES, OBJECT\_BALL, &iTmp)

== WM->getAgentObjectType())

{

soc = moveToPos(posBall, PS->getPlayerWhenToTurnAngle());

}

else if(WM->getSecondClosestInSetTo(OBJECT\_SET\_TEAMMATES, OBJECT\_BALL)

== WM->getAgentObjectType())

{

soc = moveToPos(pointKickTo(), PS->getPlayerWhenToTurnAngle());

}

else

{

soc=moveToPos(WM->getStrategicPosition(),PS->getPlayerWhenToTurnAngle());

}

ACT->putCommandInQueue( soc );

ACT->putCommandInQueue( turnNeckToObject( OBJECT\_BALL, soc ) );

}

**2.界外球。实验基本思路和内容和角球相同。编写程序，类似角球实现接应。**

界外球和角球类似，区别在于比赛中很少有机会在对方底线发界外球。所以大部分情况都将接应点设置得更加靠近敌方半场，增加进攻机会。

在PlayTeams.cpp里添加：

else if( WM->isKickInUs() )//我方界外球

{

if( WM->isBallKickable() )

{

soc = kickTo(pointKickToK(), 2.0);

}

else if(WM->getFastestInSetTo( OBJECT\_SET\_TEAMMATES, OBJECT\_BALL, &iTmp)

== WM->getAgentObjectType())

{

soc = moveToPos(posBall, PS->getPlayerWhenToTurnAngle());

}

else if(WM->getSecondClosestInSetTo(OBJECT\_SET\_TEAMMATES, OBJECT\_BALL)

== WM->getAgentObjectType())

{

soc = moveToPos(pointKickToK(), PS->getPlayerWhenToTurnAngle());

}

else

{

soc=moveToPos(WM->getStrategicPosition(),PS->getPlayerWhenToTurnAngle());

}

ACT->putCommandInQueue( soc ); // 放入命令队列

ACT->putCommandInQueue( turnNeckToObject( OBJECT\_BALL, soc ) );

}

在Basicplayer.cpp里添加：

//此函数返回界外球应该踢向的点，目的是把球尽可能往敌方半场踢

VecPosition BasicPlayer::pointKickToK()//计算界外球应该踢向的点

{

VecPosition posBall, posKickTo;

double x, y;

posBall = WM->getBallPos();

x = posBall.getX();

y = posBall.getY();

posKickTo.setX( x + 12 );

posKickTo.setY(- y / fabs(y) \* 12 + y );

return posKickTo;

}

**3.定位球/任意球，设计思路和实验内容也类似角球。区别在于情况合适时可以选择直接射门。**

基本思路同界外球，都是尽可能向对方半场踢。但当条件合适（比如X比较大或者在禁区内），我们选择向较大的夹角直接射门。

else if( WM->isFreeKickUs() )//我方任意球

{

if( WM->isBallKickable() )

{

if(WM->isInTheirPenaltyArea())//在禁区内直接射门

{

soc = maxAngShoot( posAgent );

}

else

{

soc = kickTo(pointKickToK(), 2.0);

}

}

else if(WM->getFastestInSetTo( OBJECT\_SET\_TEAMMATES, OBJECT\_BALL, &iTmp)

== WM->getAgentObjectType())

{

soc = moveToPos(posBall, PS->getPlayerWhenToTurnAngle());

}

else if(WM->getSecondClosestInSetTo(OBJECT\_SET\_TEAMMATES, OBJECT\_BALL)

== WM->getAgentObjectType())

{

soc = moveToPos(pointKickToK(), PS->getPlayerWhenToTurnAngle());

}

else

{

soc=moveToPos(WM->getStrategicPosition(),PS->getPlayerWhenToTurnAngle());

}

ACT->putCommandInQueue( soc ); // 放入命令队列

ACT->putCommandInQueue( turnNeckToObject( OBJECT\_BALL, soc ) );

}

1. **实验截图**

****

1. **实验心得**

通过本次实验我们了解了仿真机器人足球比赛中特殊比赛模式发生的条件，掌握了 Robocup 仿真机器人足球比赛特殊比赛模式的规则要求，学习了很多关于足球的知识；了解了 Robocup 仿真机器人足球比赛特殊比赛模式的战术设计思想，并进一步尝试设计并进行试验；进一步熟悉 WorldModel 类。