**机器人系统**

|  |  |  |  |
| --- | --- | --- | --- |
| **创建人名** | **修改人名** | **时间** | **修改内容** |
| 杨靖 |  | 2022-02-15 | 创建文档 |
|  |  |  |  |
|  |  |  |  |

# 局外逻辑相关

## 机器人的账号创建

### 根据任务需求即时创建

* 没有可执行任务的机器人时，创建一个对应编号的机器人
  + 先确定需执行任务的房间编号
  + 根据RoomConfig【房间配置】表的，读取对应的机器人类型
  + 再根据机器人类型，在RobotsConfig【机器人配置】表里，根据【序号】，读取下一个未用的机器人
    - 这里【序号】当指针用
    - 如果全用尽，则序号回到1
      1. 一般来说不会用到，只是防止意外情况
* 创建流程同创建一个真实玩家账号
  + 给该账号打上【机器人】标签，以方便后台进行数据过滤

## 机器人的数据初始化

* 在创建机器人时，直接根据RobotsConfig【机器人配置】，为机器人进行赋值

### 游戏资源初始化

* 初始化资源
  + 金币
* 初始化Vip
  + Vip经验

### 社交信息初始化

* 初始化以下信息
  + 名称——使用与玩家相同的生成逻辑
  + 性别
  + 头像——使用与玩家相同的生成逻辑
  + 头像框，给予道具并赋值
  + 骰盅，给予道具并赋值

### 对局属性赋值

* 技巧
  + 技巧越高，计算能力越强，越不容易犯错
  + 用来【让AI犯错】，从而像个真人
* 梭哈
  + 梭哈倾向越高，跳开的倾向越高，越倾向于快速逼近50%胜率线
  + 风格属性

## 机器人的状态

### 待机

* 可以响应任务需求
* 被创建后，机器人默认处于待机状态

### 任务中

* 同一机器人，同一时间只能执行一个任务

### 休眠

* 无法响应任务需求
* 进入休眠状态后，会有CD，CD结束后，机器人将恢复至待机状态

## 机器人的任务

### 执行匹配任务

* 根据玩家所匹配的房间类型
  + 当玩家进行匹配达到一定时长时，投放一个机器人与其匹配
    - 读RoomConfig【房间配置】的【机器人匹配等待时长】字段
  + 投放的机器人类型
    - 读RoomConfig【房间配置】的【机器人类型】字段
* 没有达到理想房间人数时，继续投放机器人
  + 读RoomConfig【房间配置】的【理想房间人数概率数组】字段
  + 随一个人数
  + 房间内人数小于人数时，继续投放机器人
* 匹配任务所对应的目标为房间，而非具体玩家

### ~~执行放水任务~~

* ~~在对局结束后【游戏间隔期】，检测是否有需要被【放水】的玩家~~
  + ~~【放水】任务检测玩家范围~~
    - ~~最长注册时间~~
      1. ~~SystemConfig【系统常量配置】的【MaxRegisterTime】字段~~
    - ~~最高房间等级~~
      1. ~~SystemConfig【系统常量配置】的【MaxRoomLevel】字段~~
  + ~~【放水】任务触发条件~~
    - ~~最近X局中，败场大于等于Y局，胜场小于等于Z局~~
* ~~若该玩家所处房间内，有空位，无机器人~~
  + ~~投放一个机器人进入其所在房间，执行【放水】任务~~
* ~~若该玩家所处房间内，无空位，无机器人~~
  + ~~跳过~~
* ~~若该玩家所处房间内，有机器人~~
  + ~~若机器人有其他的【放水】任务，跳过~~
  + ~~否则由该机器人执行【放水】任务~~
* ~~放水任务执行完成后，机器人的任务会被转为【对当前房间的匹配任务】~~

### 解除匹配任务

* 根据机器人所处的房间类型
  + 当房间人数达到一定数量后，机器人在本局结束后概率解除匹配任务
    - 读RoomConfig【房间配置】的【机器人退出房间所需人数&概率数组】字段
      1. 每次最多只退出1个人
    - 概率为百分之
* 若执行任务达到时长，则本局结束后立即解除匹配任务
  + 读RobotsConfig【机器人配置】的【执行任务最大时长】字段
* 若金币不足导致被迫站起，则本局结束后立即解除匹配任务

## 机器人的休眠

### 执行任务达到时长，休眠

* 读RobotsConfig【机器人配置】的【执行任务最大时长】字段

### 金币不足，休眠

* 休眠结束后，重置金币数量

### 金币超出，休眠

* 在待机状态下方触发
  + 读RobotsConfig【机器人配置】的【最大金币】字段
* 休眠结束后，重置金币数量

### 休眠CD【时长】

* 读RobotsConfig【机器人配置】的【休眠CD】字段
* 休眠结束后，转为待机状态

## 机器人的社交

### 好友

* 机器人可以被好友搜索到，可以被发送好友申请，但会在一定延时后拒绝
  + 到机器人下次上线时拒绝

### 聊天/语音

* 机器人可以被发送聊天消息
* 本版本，机器人不会回应聊天消息，也不会主动向他人发送聊天消息
* 语音频道里，机器人的麦默认关闭

# 局内逻辑相关【吹牛模式】

## 新增玩家数据

### 机器人对战盈亏

* 玩家在吹牛模式中，与机器人对战时，从机器人身上盈利/损失的累计值
  + 盈利只计算扣税后玩家获得的纯利
  + 损失计算全值
* 该统计包含对局结果的盈亏，以及竞猜的盈亏
  + 竞猜部分，按比例计算玩家获得/损失的利润
  + 如玩家在竞猜胜利方，占比为A；失败方机器人占比为B，总金额为C
    - 则玩家的实际机器人盈亏为A\*B\*C，之后扣税
    - 反之同理，亏损不扣税

## 胜率查询

* 【上回合叫点者模拟数量】
  + 若1未叫
    - 读SystemConfig【系统常量配置】表的上回合叫点者模拟数量【1未叫】字段
  + 若1已叫
    - 读SystemConfig【系统常量配置】表的上回合叫点者模拟数量【1已叫】字段
* 以【对局中其他玩家人数】为key\_1
  + 【对局中其他玩家人数】=对局总人数-2
    - 减去【自身】&【上回合叫点者】
  + 如果上回合为自己叫点，该回合机器人用不到胜率查询
* 【减去自身骰子后的叫点数量】=当前叫点数量-自身拥有对应点数的数量（包括1点的效果）-【上回合叫点者模拟数量】
  + 当【对局中其他玩家人数】=0时，【减去自身骰子后的叫点数量】=当前叫点数量-自身拥有对应点数的数量（包括1点的效果）
  + 若该值<=0，设为0
  + 若该值>=【对局中其他玩家人数】\*6，设为【对局中其他玩家人数】\*6
  + 为key\_2
* 双key，在BoastModeWinProbConfig【吹牛模式胜率配置】表，查询对应的胜率
  + 此值为叫开胜率
* 查询时，key\_2使用机器人的技巧值进行扰乱
  + 先根据技巧值，判断是否扰乱
    - 扰乱概率=（100-技巧值）/100
  + 扰乱值 =向上取整（（100-技巧值）/100\*（其他玩家人数+1））
  + 实际扰乱值从[-扰乱值,-1]&[1,扰乱值]，中随机取一个整数
  + 扰乱后的key2 <=0，设为0；
  + 扰乱后的key2 >=【对局中其他玩家人数】\*6，设为【对局中其他玩家人数】\*6

## 本局开始时，计算针对各玩家的【胜利倾向】

* 【胜利倾向】=该玩家的【机器人对战盈亏】/【房间底注】\*【胜利倾向系数】
  + 【胜利倾向系数】读SystemConfig【系统常量配置】表
  + 【胜利倾向】取值区间为[-8000,8000]
    - SystemConfig【系统常量配置】表的WinAddRange【胜利倾向区间】
* 【初始机器人对战盈亏】读SystemConfig【系统常量配置】表
* 针对机器人的【胜利倾向】永远为0
* 备注：胜利倾向>0意味着该玩家赢了机器人钱

## 回合对战策略【表现】

### 如果是自己叫点后一回合，跳过

### 先执行【开/抢开】判断

* 根据当前叫点，获取当前的【查询所得叫开胜率】
  + 【查询所得叫开胜率】>=【最小叫开胜率基数】
    - 叫开
    - 否则跳过
    - 【最小叫开胜率基数】读SystemConfig【系统常量配置】表

### 之后，如果是自己回合，执行叫点逻辑

* 选择点数策略，每次叫点时均随机
  + SystemConfig【系统常量配置】表的【点数策略概率数组】字段
  + 【随机的非1点数】：概率1
  + 【跟随上个叫点的点数】：概率2
  + 【最大数量的点数】：概率3；如果有多个点数数量相同，随机一个
* 选择数量策略
  + 【随机数量】：概率触发，未触发直接跳至【最小数量】策略
    - 概率读SystemConfig【系统常量配置】表的【叫点随机数量触发概率】字段
    - 倒查叫开胜率<=【叫点随机数量触发胜率】时，所对应的最大数量
      1. 倒查时，【对局中其他玩家人数】=对局总人数-1
    - 如果【该数量+自己拥有的数量】小于等于当前数量，执行【最小数量】策略
    - 否则，如果【点数】大于等于【当前点数】
      1. 从【当前数量】到【该数量+自己拥有的数量】之间，随一个值
      2. 否则，从【当前数量+1】到【该数量+自己拥有的数量】之间，随一个值
  + 【最小数量】：
    - 如果是第一回合，默认为对局人数
    - 否则，要叫的点数大于桌面点数，数量 = 桌面数量+0
    - 否则，数量 = 桌面数量+1

## 对局结算改牌策略

### 确认是否改牌

* 若结算双方是否为【机器人+玩家】，则下述两种情况，进行改牌随机判断
  + 玩家的【机器人对战盈亏】>0，且场面为【玩家胜利】
  + 玩家的【机器人对战盈亏】<0，且场面为【玩家失败】
* 改牌随机判断：
  + 根据【胜利倾向】随机，判断本次是否改牌
  + 改牌概率=Abs【胜利倾向】/10000

### 改牌策略

* 若点数要从小改大：
  + 获取要改动到的【最小点数】&【差值点数】
  + 每个机器人都进行骰子重摇
    - 若摇到比当前更大的点数，保留并记录，否则继续，直到重摇次数用尽
      1. ChangeMaxReRollCount【结算改牌最大重摇次数】
  + 若新牌型能够改变对局结果，则应用新牌型
    - 否则不改变
* 若点数要从大改小：
  + 获取要改动到的【最大点数】&【差值点数】
  + 每个机器人都进行骰子重摇
    - 若摇到比当前更小的点数，保留并记录，否则继续，直到重摇次数用尽
      1. ChangeMaxReRollCount【结算改牌最大重摇次数】
  + 若新牌型能够改变对局结果，则应用新牌型
    - 否则不改变

## 重摇策略【公共】

* 机器人有一定概率进行重摇
  + SystemConfig【系统配置】表的【机器人重摇概率数组】字段
  + 数组里为第1、2、3……次重摇的概率
* 如果首回合为该机器人行动，机器人若触发重摇，其叫点延时读另一个值

## 升场策略【公共】

* 机器人在金币满足的情况下，永远同意升场

## 竞猜策略【公共】

* 根据机器人的【竞猜倾向】和【当前叫开胜率】，随机判断本次是否参与竞猜
  + 【实际竞猜概率】=【竞猜倾向】/10000\*(Abs【当前叫开胜率-5000】/5000+竞猜概率基数)
    - 读RobotsConfig【机器人配置】表的VoteRatio【竞猜倾向】字段
    - 读SystemConfig【系统常量配置】表的VoteProbBase【竞猜概率基数】
  + 当前叫开胜率>=5000，押叫开方
    - 否则押被叫开方

## 竞猜结算换边策略

* 若胜利方有【胜利倾向】>0的玩家，则失败方机器人概率换边
  + 换边概率=【胜利倾向】/10000\*(1-Abs【当前叫开胜率-5000】/5000)
* 每个机器人只可判断一次

## 延时逻辑【公共】

* 重摇延时
  + 读SystemConfig表【重摇延时】字段
  + 每次延时独立随机
* 叫点延时
  + 读SystemConfig表【叫点延时】字段
  + 读SystemConfig表【重摇叫点延时】字段
* 叫开延时
  + 读SystemConfig表【叫开延时】字段
* 升场延时
  + 读SystemConfig表【升场延时】字段
* 竞猜延时
  + 读SystemConfig表【竞猜延时】字段

## 互动策略【暂不开发】

# 配置相关

## RobotsConfig【机器人配置】

## RoomConfig【房间配置】

## BoastModeWinProbConfig【吹牛模式胜率配置】

## SystemConfig【系统常量配置】