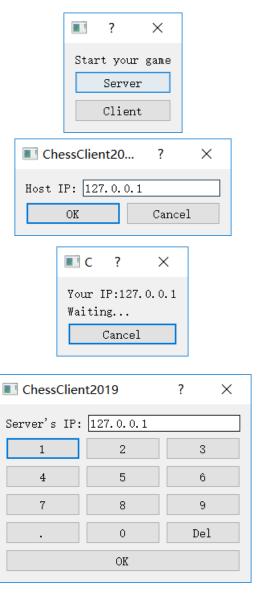
# 网络对战国际象棋设计文档

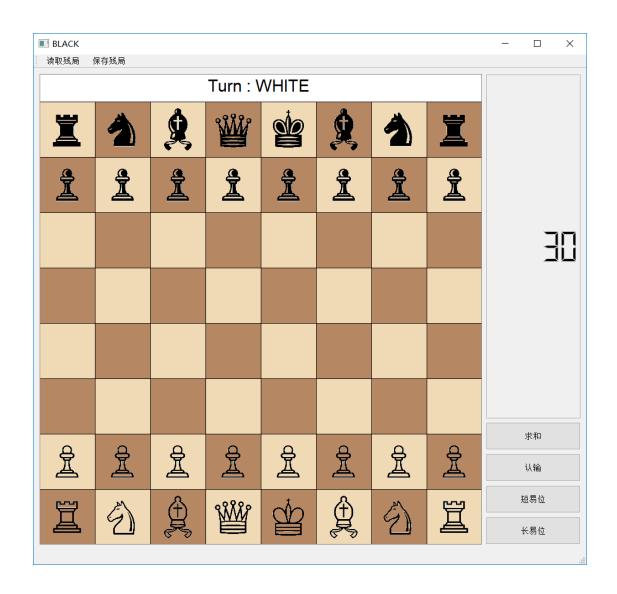
#### 励永辉

## 一. 软件介绍

本软件是利用 Qt 实现的双人网络对战国际象棋, 支持兵升变和王车易位这两个特殊规则。玩家可以用保存残局、恢复残局的方式来分别保存、加载游戏。

# 二. 界面展示





# 三. 客户端、服务器端的工作流程

- 1. 服务器端建立 QTcpServer, 设置监听 IP 和端口号, 进入监听状态, 等待客户端的连接
- 2. 客服端创建 QTcpSocket, 尝试利用 IP 地址和端口号对服务器进行连接
- 3. 服务器端受到用户的连接请求,新建 QTcpSocket 与用户端通信
- 4. 服务器端和用户端建立稳定的通信后,显示游戏界面。两端对棋盘的操作或者认输、求和等特殊操作都会通过通信进行实时同步。

5. 游戏结束后, 服务器端和用户端断开连接。

### 四. 通信协议

- 1. 用 QString 来保存、处理信息,再将其转换为 QByteArray 利用 socket 端口发送
- 2. (1) 最普通的移动位置操作利用利用 7 个字符位的字符串进行传输。第一位标识黑白方, 0 表示白方, 1 表示黑方。第二位为 m, 表示这是移动位置操作的信息。第三位到第六位表示移动前后的坐标。第七位表示移动棋子的属性, 用来校验之前的操作(其中 1 对应 Bishop,2 对应 King,3 对应 Knight,4 对应 Pawn,5 对应 Queen,6 对应 Rook)

例如: 0m12345 表示白方的皇后从(1,2)移动到了(3,4)

- (2) wl 表示白方的长易位, ws 表示白方的短易位; bl 表示黑方的长易位, bs 表示黑方的短易位。
- (3) 兵过升迁时,用 5 个字符位的字符串进行传输。第一位标识黑白方,0 表示白方,1 表示黑方。第二位为 t,表示这是兵过升迁的信息。第三位和第四位表示兵的坐标。第五位表示兵变成了什么(1 对应 Bishop,3 对应 Knight,5 对应 Queen,6 对应 Rook)

例如 0t056 表示白方位于(0,5)处的兵变成了车。

(4) w 表示赢得比赛, f 表示认输, h 表示求和, y 表示同意求和, n 表示拒绝求和

# 五. 网络通信编程框架

#### 1. 服务器端

QTcpServer \*listenSocket = new QTcpServer; //新建 QTcpServer
listenSocket->listen(QHostAddress::Any,55555); //开启监听
connect(listenSocket,SIGNAL(newConnection()),this,SLOT(acceptConnection()));
QTcpSocket \*readWriteSocket = listenSocket->nextPendingConnection();
connect(this->readWriteSocket,SIGNAL(readyRead()),this,SLOT(recvMessage()));

### 2. 用户端

QTcpSocket \*readWriteSocket = new QTcpSocket; //新建 QTcpSocket readWriteSocket->connectToHost(QHostAddress(address),55555); //连接服务器 connect(readWriteSocket,SIGNAL(connected()),this,SLOT(connected())); connect(readWriteSocket,SIGNAL(readyRead()),this,SLOT(recvMessage()));

### 六 . 设计思路

利用 QTcpServer、QTcpSocket 的信号和槽机制建立起服务器端和用户端的 网络连接。

整个游戏基于 Graphics View。

Piece 棋子类继承 QGraphicsPixmapItem, 可加载图片。Piece 重写了mousePressEvent 用来处理点击选取棋子的操作。Bishop、King、Knight、Queen、Pawn、Rook 继承 Piece,分别加载不同的图片来作为不同的棋子,同时重写了基类的 findways()函数以此依据规则来搜索能够到达的格子,禁止违规操作。

Grid 方格类继承 QGraphicsRectItem,重载了 mousePressEvent 来处理点击事件。Grid 内部储存了自己颜色,坐标,对应的棋子等信息。

Info 类继承 QGrahicsView, 内部储存了 8\*8 方格的数组, 且有一个 Info \*info

的全局变量可供不同类使用。Info 类作为整个程序的控制类, 负责了许多重要运算, 例如读取、储存文件。