

# 面向多平台的井下作业仿真培训系统<sup>①</sup>

沈旭霞, 刘贤梅, 岳岩岩

(东北石油大学 计算机与信息技术学院, 大庆 163318)

**摘要:** 针对井下作业培训平台无法满足员工随时随地进行培训、培训内容缺少交互性等问题, 设计并开发了一套面向多平台的支持三维交互的井下作业仿真培训系统. 本系统应用 WEB、导航网格、数据库、Android 等技术, 融合移动培训、数字化培训和交互式培训等多种培训方式, 改进传统试题模式, 解决多平台下的数据源支持问题, 利用导航网格技术重写自动寻路. 井下作业培训不再受网络、设备等的限制, 培训内容交互性强, 对于提高培训质量、降低作业现场安全事故的发生率具有较大的价值.

**关键词:** 井下作业; 虚拟现实; 仿真; 导航网格寻路; 多平台

## Multi-Platform Oriented Downhole Operation Training Simulator

SHEN Xu-Xia, LIU Xian-Mei, YUE Yan-Yan

(College of Computer and Information Technology, Northeast Petroleum University, Daqing 163318, China)

**Abstract:** Since downhole operation training system can't meet the staff's requirements of learning whenever and wherever and training content lack of interactive, a set of multi-platform which supports 3D interaction of the downhole operation simulation training system is established. This system applies many technologies such as WEB, Navigation Mesh, Database, Android, fused mobility training, digital training, interactive training, optimizes the mode of tradition training, solves the problem of data source supporting in multi-platform, overwrites target homing road with the help of Navigation Mesh. Downhole operation training isn't restricted by network and equipment, interactive training content, and has the great value of improving training quality and reducing the accident rates of job location.

**Key words:** downhole operation; virtual reality; simulation; navigation mesh; multi-platform

井下作业是油田生产最前沿的岗位, 也是事故总量较高, 危险性较大的岗位. 通过对事故类型的分析, 事故多发的主要原因有: (1)违反操作规程, 不按工艺规程施工; (2)安全意识不足, 对灾情的应急响应和逃生技巧不够. 因此, 针对操作规程和逃生技巧的培训非常必要, 这不仅能降低作业现场安全事故的发生率, 而且能维持钻井作业安全有效的运转.

随着技术的进步, 井下作业培训的方式不断更新. 首先出现了数字化培训, 弥补传统培训受时间、地点限制, 建立实际培训基地费用高的缺陷. 但传统数字化培训的培训内容有限, 智能化不足, 二维的网页无法实现实际环境与参与者的动态交互, 培训效果不尽

如人意. 为此, 在数字化培训的基础上引入了虚拟现实技术. 虚拟现实技术与数字化培训相结合产生了三维页面, 将人的行为作为浏览器的主体, 所有的表现都随着操作者行为而改变, 改变了原来网页交互性差的缺点. 然而这类培训方式大都基于局域网, 对访问的 IP 地址有很大限制, 不能保证员工随时随地访问系统进行培训.

针对上述问题, 本文将 ASP.NET 服务器端技术和虚拟现实技术相结合, 采用跨平台的 Unity3D 开发工具以及目前较流行的 Visual Studio 2010 工具, 使用 C# 语言, 开发支持手机、平板电脑、PC 机等多个平台, 在线、离线多种方式的井下作业仿真培训系统. 便于员

<sup>①</sup> 基金项目:国家自然科学基金(61502094);黑龙江省教育科学规划重点课题(GJB1215019)

收稿时间:2015-11-09;收到修改稿时间:2015-12-12 [doi:10.15888/j.cnki.csa.005225]

工根据自身条件有选择性的利用各种终端进行培训, 满足经常在野外的工作人员对移动培训的需求.

### 1 系统整体架构及设计流程

图 1 描述了本系统的整体架构, 井下作业仿真培训系统采用 B/S+C/S 的四层体系结构, 支持手机、平板电脑、PC 机多种终端.

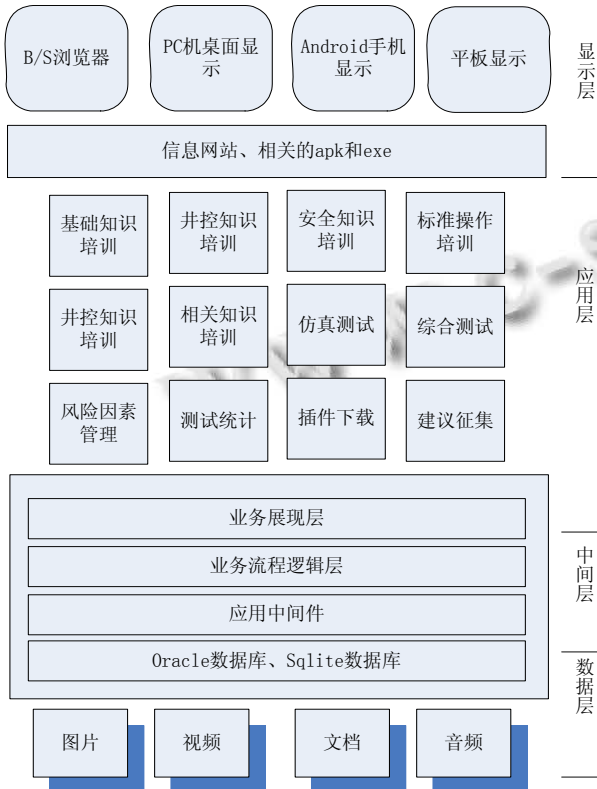


图 1 系统整体架构图

本系统包含三个版本: 安卓版、PC 版、网页版. 安卓版和 PC 版是在 Unity3D 中编程后直接发布成安装包. 网页版是三维的网页, 它主要分为网页树形结构和 Unity3D 虚拟仿真培训两部分. 网页树形结构是负责用户在浏览器上的各种操作和管理; Unity3D 虚拟仿真培训则利用 Unity3D 引擎开发具有统一接口的展示框架, 实现各类培训的场景及交互式虚拟仿真培训. 井下作业仿真培训系统的设计流程如图 2 所示.

### 2 关键技术

#### 2.1 题型设计

结合现有的井下作业培训视频和文档, 在传统题型的基础上, 将题型分为三类, 分别是: 单项选择题、

工具选择题以及自动播放题.

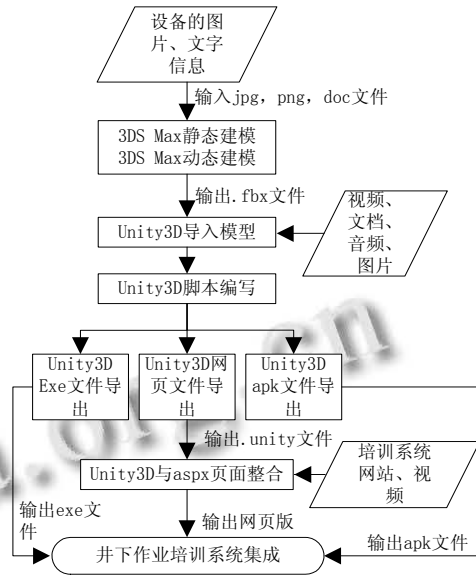


图 2 系统设计流程图

(1)单项选择题: 传统题型具有题型灵活、评分简单客观、考试结果有一定的诊断功能等优点. 在脚本设计时, 添加了一类单项选择题, 共有三个可选项, 只有一个正确答案. 为了弥补传统培训中各个岗位之间没有明确的分工导致石油井下作业的安全管理不能有效进行的缺陷, 脚本中的选择题在传统选择题的基础上进行改进, 用户不仅需要进行单项选择, 而且要选择由在场的哪个岗位执行选项中的操作, 以此来明确各岗位的分工.

(2)工具选择题: 井下作业是利用一系列的、工具保证作业正常进行的工作流程, 工具操作比较复杂, 所以在设计时加入了工具选择题. 在工具选择题中, 用户需要在菜单栏中选择正确的工具, 再点击工具应该操作的设备, 以便于用户熟悉如何使用各类工具.

(3)自动播放题: 在交互式培训中, 完整的过程反馈子体系是不可或缺的, 用户的一切行动都要及时反馈到虚拟现实环境中, 而用户也必须及时看到虚拟空间的反馈以便做出反应. 因此, 在设计题型时添加了一项自动播放题, 将用户的行动导致的结果以仿真的方式在三维空间中形象地显示出来, 以便提高学员汲取知识和掌握技能的效率、从而改善教育培训的效果.

#### 2.2 多数据源支持

本文所开发的系统应用于各种不同的平台, 而不同平台所支持的数据源是不同的, 因此系统提供了多

数据源支持. 为便于新增和修改试题信息, 将试题信息存储于 XML 文件中; 手机版和 EXE 版采用 Sqlite 数据库; 网页版采用 oracle 数据库. 并且使用上传和下载按钮保障客户端和服务端的数据一致.

### 2.2.1 文件的存储和解析

试题信息都存储在根节点 list 下的 item 子节点中. 每一个试题信息 item 中包含以下内容: stid(试题编码)、xxa(选项 A 的内容)、xxb(选项 B 的内容)、xxc(选项 C 的内容)、ksgw(参加考试的人员, 主要包括: 班长编码为 0, 一岗编码为 1, 二岗编码为 2, 三岗编码为 3, 四岗编码为 4, 操作手编码为 5)、zqda(选择题的正确答案)、sfgj(是否需要工具, N 为否, Y 为是)、gjbm(若需要工具, 则填写工具编码)、stlx(试题类型, 1 为选择题, 2 是自动播放题, 3 是工具题)、djjjc(选择的工具的简称)、djmxjc(选择的模型的简称)、csdtsxx(测试点提示信息)、fz(分值).

使用类 XmlDocument 解析 XML 文件, 具体步骤如下:

(1)构造结构体 ST 来存放 XML 文件中的试题信息, 结构体 ST 中的属性与 XML 文件的 item 节点中的属性一一对应. 创建 ST 类型的数组存放所有试题.

(2)加载 st.xml 文件

```
string data = Resources .Load ("st" ).ToString ();
```

```
xmlDoc .LoadXml (data );
```

(3)获取根节点 list 下的所有子节点(获取的所有子节点即所有试题信息)

```
XmlNodeList nodeList=xmlDoc.SelectSingleNode("list").ChildNodes;
```

(4)遍历子节点, 取出子节点中的试题信息存于数组中.

### 2.2.2 多平台的 Sqlite 数据库访问

Sqlite 数据库主要是在 EXE 版本和手机版中使用, 所以在连接数据库的时候需要先判断系统所在平台. 默认是在电脑上运行该系统, 其连接数据库的代码为:

```
string path = Application.dataPath +
"/Plugins/Android/assets/" + "orcl1.db";
```

```
db = new DbAccess ("URI=file:" + path);
```

用 #if UNITY\_EDITOR 判断是否是在安卓系统上运行, 若是在安卓系统上则连接数据库的代码为:

```
string appDBPath = Application.dataPath +
"/Plugins/Android/assets/" + "orcl1.db";
```

```
db = new DbAccess ("URI=file:" + appDBPath);
```

### 2.2.3 Oracle 数据库访问

网页版的数据库访问使用的是 Oracle 数据库, 访问 Oracle 数据库时, 将数据库连接串放在 Web.config 中, 在程序中使用类 ConfigurationManager 中的 AppSettings 方法获得数据库连接串, 获得连接串后再创建 oracle 连接, 代码如下所示.

```
Connection = new OracleConnection(Con
nectionString);
```

```
Connection.Open();
```

### 2.2.4 数据更新

以 Sqlite 向 Oracle 数据库更新数据为例, 其实现分两部分:

(1)客户端发送数据

① 使用类 SqlDataReader 取出 Sqlite 中数据表的值.

② 将从 Sqlite 中取出的数据放入页面表单类 WWWForm 中, 生成表单数据.

③ 使用 WWW 类将表单数据传递到 web 服务器.

(2)服务器端接收数据, 更新 Oracle 数据库

① 使用类 Request 接收数据.

② 编写 SQL 语句, 将数据存入 Oracle 数据库中.

### 2.3 导航网格寻路

在仿真培训中, 各个岗位经常在井场中行走, 因此需要解决角色寻路问题. 由于 Unity3D 自带的寻路控件不能很好的融入到系统中, 所以本文借鉴博文 Fix Pathfinding Once and For All 的思路用 C#语言重写了导航网格寻路. 导航网格实现寻路包含两部分: (1)生成导航网格; (2)根据生成的导航网格进行寻路.

#### 2.3.1 导航网格的生成

早期 2D 游戏中常用栅格法<sup>[1,2]</sup>、四叉树法<sup>[3]</sup>等. 目前在 3D 游戏中手工设置路点法使用非常广泛, 但是手工添加导航点工作量大, 导航点只反映了位置和连通信息, 存在智能体移动的轨迹重合, 存在组合爆炸的问题<sup>[4]</sup>. 针对这种现状, 我们引入一种在三维环境下自动建立搜索空间的方法: 导航网格. 导航网格是一个凸多边形的集合, 凸多边形内部任意两点的连线都在其内部, 每个单元在路径搜索中高度独立. 导航网格的生成借鉴了曾薇等<sup>[5]</sup>在平面多边形域 Delaunay 三角化中的算法思想, 它采用了增量思想和均匀网格, 快速生成约束 Delaunay 三角形<sup>[6]</sup>. 其算法步骤如下:

- ① 取任意一条外边界边  $p_1p_2$ .
- ② 计算 DT 点  $p_3$ , 构成约束 Delaunay 三角形  $\Delta p_1p_2p_3$ .
- ③ 如果新生成的边  $p_1p_3$  不是约束边, 若已经在堆栈中, 则将其从中删除; 否则, 将其放入堆栈; 类似地, 可处理  $p_3p_2$ .
- ④ 若堆栈不空, 则从中取出一条边. 转 2; 否则, 算法停止.

其中计算 DT 点的步骤如下:

- ① (假设是找线段  $p_1p_2$  的 DT 点) 在线段堆栈中查找  $p_1p_2$  的所有的可见点存入堆栈 allVPoint 中, 先假设第一个找到的可见点 allVPoint[0] 为  $p_3$ .
- ② 构造  $\Delta p_1p_2p_3$  的外接圆  $C(p_1, p_2, p_3)$  及其包围盒  $B(C(p_1, p_2, p_3))$ .
- ③ 依次访问 allVPoint 中其他的可见点: 若存在可见点  $p$ , 并且  $\angle p_1pp_2 > \angle p_1p_3p_2$ , 则令  $p_3=p$ , 转 Step2; 否则, 转 Step4.
- ④ 若即  $C(p_1, p_2, p_3)$  内无可见点, 则  $p_3$  为的  $p_1p_2$  的 DT 点.

通过以上算法, 原场景图(如图3左)生成网格后的地图如图3(右)所示, 其中 A 点是寻路的起点, B 点是寻路的终点.

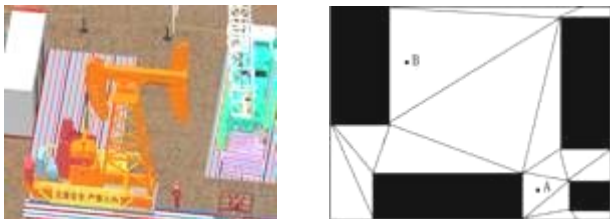


图3 场景地形图和网格化后的地图

### 2.3.2 网格寻路

路径搜索算法分为: 盲目搜索和启发式搜索<sup>[7]</sup>. 盲目搜索就是按照一定的规则下去, 不考虑结点带给的任何信息, 它只适用于求解比较简单的问题<sup>[8]</sup>. 而启发式搜索在状态空间搜索时对每一个搜索的位置进行评估, 得到最好的位置, 再从这个位置进行搜索直到目标. 这样可以省略大量无谓的搜索路径, 提高效率<sup>[9]</sup>. 因此本项目采用了启发式算法中非常典型的 A\* 算法来实现网格路径的搜索.

#### 2.3.2.1 A\* 算法寻找网格路径

A\* 的估价函数  $f(n)=g(n)+h(n)$ ,  $g(n)$ =穿入边和穿

出边中点的距离(如下图4 线段 DE),  $h(n)$ =三角形中心点到路径终点的  $x$  和  $y$  方向的距离(如下图4 线段 FG).

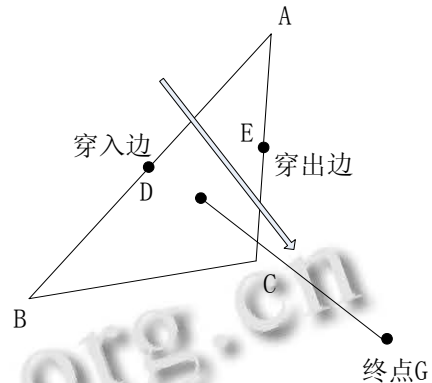


图4 系统设计流程图

A\* 算法寻找网格路径的步骤如下:

#### (1) 生成封闭列表

- ① 首先确定起点和终点所在的网格. 把终点所在的网格放入开放列表中.
- ② 把当前节点从开放列表删除, 加入到封闭列表.
- ③ 对当前网格相邻的每一个三角形网格依次执行以下步骤:

1) 如果该相邻网格不可通行或者该相邻网格已经在封闭列表中, 则什么操作也不执行, 继续检验下一个网格;

2) 如果该相邻节点不在开放列表中, 则将该节点添加到开放列表中, 并将它相邻节点的父节点设为当前节点, 计算并且保存该相邻节点的 G 和 F 值; 如果该相邻节点在开放列表中, 则判断若经由当前节点到达该相邻节点的 G 值是否小于原来保存的 G 值, 若小于, 则将该相邻节点的父节点设为当前节点, 并重新设置该相邻节点的 G 和 F 值.

#### (2) 从封闭列表中取出网格路径(关闭列表中的最后一个节点就是起点所在的网格 start, 然后再由 start.parent 获得下一个网格, 以此类推就能获得网格路径).

通过 A\* 算法可从网格化后的地图中寻找出的网格路径如图5(左)所示.

#### 2.3.2.2 拐角点法计算路径点

在使用 A\* 算法寻找到网格路径后, 可以使用光照射线法或者拐角点法获得网格内的路径点, 我们选择了实现起来相对简单的拐角点法<sup>[10]</sup>, 其步骤如下所示.

(1)首先找到起始点所在的多边形和穿出边的两个端点,由起点连接两个端点,形成两个线段  $lineLeft$  和  $lineRight$ .

(2)继续寻找路径网格中下一个网格的穿出边的两个端点,判断新的左点是否在  $lineLeft$  和  $lineRight$  之间,若在则更新  $lineLeft$ , 否则就不更新  $lineLeft$ ; 对穿出边的右点也进行同样的处理. 当穿出边的两个端点都在  $lineRight/lineLeft$  的右侧/左侧, 表示  $lineRight/lineLeft$  的终点即为路径的一个拐角点.

(3)循环执行第 2 步,直到到达路径网格的最后一个网格时,结束.

结束以上算法后可得出最终路径点如图 5(右)所示.

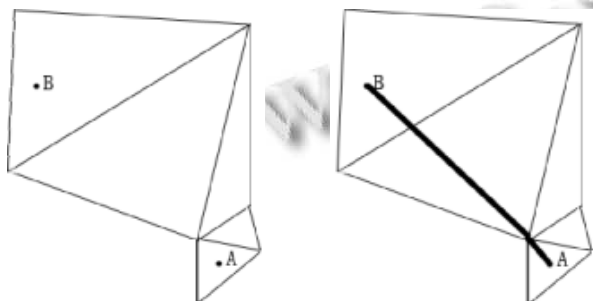


图 5 网格路径图和最终路径图

### 3 结语

井下作业安全问题是油田生产中亟待解决的问题,为帮助员工学习井下作业安全知识,出现了各种针对井下作业操作规程和应急措施的培训,例如数字化培训.但是,这些培训无法随时随地进行.因此构建数字化培训、交互式培训、移动培训相结合支持多平台的系统,便于培训者学习井下作业安全知识,提高培

训效率,有重要的应用价值.

本文下一步工作是在目前的基础上加入个性化推荐学习的新模式,分析学习者学习行为与知识掌握程度的数据,选择和定制合理的学习路径,不断提高井下作业培训质量.

### 参考文献

- 1 王卫红,顾国民,秦绪佳等.基于栅格法的矢量路径规划算法.计算机应用研究,2006,(3):57-59.
- 2 朱庆保,张玉兰.栅格法的机器人路径规划蚁群算法.机器人,2005,27(2):132-136.
- 3 许鹏,梁国柱.二叉树法非结构网格剖分技术研究.中国机械工程,2006,(17):312.
- 4 曹雷,饶真珍,贺毅辉.基于导航网格的三维空间表示.系统仿真学报,2008,(20):232.
- 5 曾薇,孟祥旭,杨承磊等.平面多边形域的快速约束 Delaunay 三角化.计算机辅助设计与图形学学报,2005,17(9):1933-1940.
- 6 Amanatides J, Woo A. A fast voxel traversal algorithm for raytracing. Proc. of Eurographic '87 Conference. Amsterdam. 1987. 3-10.
- 7 孟赞.三维场景中角色自动寻径算法研究.武汉理工大学学报,2011,33(12):125-130.
- 8 何国辉,陈家琪.游戏中智能路径搜索算法的研究.计算机工程与设计,2006,27(13):2334-2337.
- 9 蔡自兴,徐光祐.人工智能及其应用(研究生用书).北京:清华大学出版社,2004:150-160.
- 10 祁悦,赵洋,杨帆.一种基于 A\*算法的分层路径规划在 3D 游戏中的应用研究.电子设计工程,2014,22(14):37-42.