

当物体达到完全压缩状态时,

$$v'_A = v'_B = v' \quad (6)$$

此时 A 对 B 的完整压缩冲量设为 I_1 , 仿照式 (5), 有

$$I_1 \left(\frac{1}{m_A} + \frac{1}{m_B} \right) = (v'_B - v'_A) + (v_{A0} - v_{B0}) = v_{A0} - v_{B0} \quad (7)$$

由式 (5) 和式 (7), 可得

$$(I - I_1)/I_1 = (v_B - v_A)/(v_{A0} - v_{B0}) \quad (8)$$

令 $\Delta I = I - I_1$, 则由式 (1) 可知

$$e = \Delta I/I_1 \quad (9)$$

式中 ΔI 表示剩余冲量, 由于式 (9) 是从任意碰撞中推导得到的, 故适用于任何碰撞过程, 即恢复因数等于碰撞的剩余冲量与完整压缩冲量之比. 根据式 (9) 可将恢复因数的取值范围分为如下几段:

(1) $e = 0$, 即 $\Delta I = 0, I = I_1$. 碰撞总冲量等于完整压缩冲量, 表明碰撞过程刚好进行到完整压缩状态就结束了, 无恢复过程, 碰撞为完全非弹性碰撞;

(2) $0 < e < 1$, 即 $0 < \Delta I < I_1, I_1 < I < 2I_1$. 碰撞总冲量大于完整压缩冲量, 但小于两倍完整压缩冲量, 表明碰撞进行到完整压缩状态后, 还有一个不完整的恢复过程, 碰撞为弹性碰撞;

(3) $e = 1$, 即 $\Delta I = I_1, I = 2I_1$. 碰撞总冲量等于两倍完整压缩冲量, 表明碰撞不但有一个完整的压缩过程, 还有一个完整的恢复过程, 碰撞为完全弹性碰撞;

在以上 3 种取值范围中, 碰撞过程都有一个完整的压缩过程, 即 $I \geq I_1$, 剩余冲量就是恢复冲量, $\Delta I = I_2$, 在此时式 (9) 可转化为式 (2).

(4) 当 $-1 < e < 0$, 即 $-I_1 < \Delta I < 0, 0 < I < I_1$. 碰撞总冲量小于完整压缩冲量, 表明碰撞过程尚未进行到完全压缩状态就已结束, 整个碰撞过程是一个不完整的压缩过程, 穿透就是这类碰撞.

(5) 当 $e > 1$, 即 $\Delta I > I_1, I > 2I_1$. 碰撞总冲量大于两倍完整压缩冲量, 表明在碰撞过程有新能量补充进来, 爆炸就对应这类碰撞.

从以上的分析可见, 将恢复因数的定义由式 (2) 改为式 (9) 之后, 可由式 (9) 反映出各种碰撞, 扩大了恢复因数的取值范围. 故笔者认为在现行教材中, 应按式 (9) 定义恢复因数, 这更有助于学生对碰撞问题的全面理解, 取得更好的教学效果.

参 考 文 献

- 1 哈尔滨工业大学理论力学教研室. 理论力学 (第 6 版). 北京: 高等教育出版社, 2002
- 2 刘延柱, 杨海兴, 朱木华. 理论力学 (第 2 版). 北京: 高等教育出版社, 2001
- 3 洪嘉振, 杨长俊. 理论力学 (第 2 版). 北京: 高等教育出版社, 2002

利用 Flash 技术开发理论力学网络作业

胡 丰 董正筑 陈俊国

(中国矿业大学理学院, 徐州 221008)

摘要 针对开发理论力学网络作业中的特殊要求, 选择 Flash 软件及其编程技术作为开发工具. 将功能模块化, 每一模块给出相应程序, 使动画具有作图、判断和传递信息等交互功能.

关键词 Flash, 网络作业, ActionScript, 交互性, 模块化

近年来, 网络作业已成为网络教学中一个重要发展方面, 网络作业丰富了网络教学的形式, 它能及时反馈学生的学习情况, 使单向的网络教学变为双向, 真正实现教学的交

互性. 将理论力学习题制作成网络作业, 并通过互联网来考察学生对理论力学知识的理解和运用, 是理论力学教学创新的有益尝试. 较通常形式的理论力学作业, 电子形式的作业让学生省去了抄写原题、绘制题图等步骤, 教师也不必批改作业, 自然可提高教学效率; 在与网络结合之后, 也使理论力学远程教育成为可能.

但理论力学网络作业实现起来较其它科目有其特殊性, 主要原因在于理论力学习题通常不是单纯的文字求解题, 几

乎每一道题目都要与作图求解相结合. 因此, 在理论力学网络作业中必须具备作图功能, 方便学生在网页中自由绘制受力图、速度图、加速度图等, 同时计算机要能够判断学生作图结果的对错, 并将作图信息向外传递.

为解决以上作图、判断和传递信息的问题, 本文选用 Flash 软件作为开发的主要工具. Flash 软件是美国 Macromedia 公司开发研制的一种矢量动画制作软件, 矢量动画的优点是: 文件体积小, 图像清晰, 任意放大或缩小图像不失真, 便于网络传输, Flash 集成的 ActionScript (动作脚本语言) 使动画具有很强的交互性. Flash 是专门的多媒体开发软件, 用它来开发多媒体程序相比于大型编程软件来说, 开发速度要快很多. 同时, 以本文为例, 还应当认识到 Flash 软件对图形具有很好的控制能力, 在交互动画中图形可根据鼠标的操作和程序的设定做出相应的变化, 如移动、拉伸、旋转等, 这就可实现作图功能; 图形变化的信息, 如坐标值、长度、旋转的角度等, 也可以实时被计算机提取, 这就可实现判断作图结果对错的函数; Flash 动画兼容性好, 与其它多媒体软件有变量传递接口, 这就可实现传递信息功能.

1 开发设计思想

有很多 Flash 制作的动画被利用在多媒体教学中, 但这些动画大多只具有演示功能, 缺乏交互能力, 无法由用户的操作产生相应的响应. 一个具有交互性的 Flash 动画需要使用到编程语言 ActionScript.

考虑到理论力学学习题众多, 每一道题目又不尽相同, 作图的结果也千变万化, 若开发每一道题目时, 都要对程序代码进行大量的修改, 必然耗时耗力. 为此程序代码应当具备可重复使用性, 一道题目的代码在稍作修改后就可直接放在其它题目中.

欲实现代码的可重复使用性, 需要将功能模块化. 一个完善的 Flash 作图程序必须实现以下 4 种功能:

- (1) 可在任意两点间自由生成辅助线, 并能将其删除, 同时实时显示辅助线角度变化;
- (2) 可在某一点上任意添加矢量, 并对大小、方向进行调节, 可删除多余矢量, 实时显示角度变化, 能显示下标, 下标显示方式可分为两种, 自动生成和用户输入. 作用力和反作用力情况时还要能显示上标, 例如用 F_A 来代表作用力, 用 F'_A 代表反作用力;
- (3) 具有简单的工具栏, 点击其中的按钮可切换到相应的作图功能.
- (4) 作图完成时, 计算机能判断作图结果的对错. 作图信息要能向外传递, 以判断文字求解部分的对错.

2 程序实现步骤

2.1 辅助线

辅助线是作图的重要工具, 在矢量分解、汇交力系等习题中都需要借助辅助线来完成作图. 本文设计绘制辅助线方式, 如图 1 所示, 当点击鼠标左键时, 开始画直线, 直线以此时光标的位置作为起点, 当鼠标移动时, 直线随着光标的移动一起伸长和旋转, 再次点击左键时, 结束绘制. 用 Flash 实现这一动画特效, 可先将一小段直线作为影片剪辑, 再利用 onClipEvent() 函数来根据鼠标的移动改变这一影片剪辑的属性. 程序为:

```
onClipEvent (mouseMove) {
    if (line.move) {
        // 调整直线倾斜角度
        _rotation = Math.atan(_parent.fixpt._ymouse/
            _parent.fixpt._xmouse)*180/Math.PI;
        // 显示角度
        _parent.line_angle.angle = Math.floor(-_rotation);
        // 调整直线长度
        _xscale = Math.distance(_parent.fixpt._xmouse,
            _parent.fixpt._ymouse, 0, 0);
        if (_parent.fixpt._xmouse<0) {
            _xscale = -_xscale;
        }
    }
}
```

为提高作图的精度, 辅助线还需要显示直线倾斜的角度, 如图 1 中的数字 16. 这可在直线旁添加一输出文本框 (Dynamic Text), 输出值为以上程序中的变量 line_angle.angle.

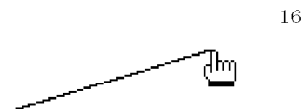


图 1 辅助线效果图

辅助线的生成区域由一足够大的按钮 (Button) 来定义, 这个按钮不需要添加任何图形, 仅仅用来形成一个热区, 以起到接收鼠标键盘指令的作用, 利用 attachMovie() 命令可将库中的影片剪辑复制出来, 程序为

```
on (release) {
    if (dot_count == 0) {
        // 左键第一次点击
        attachMovie("lines", "line"+count, count);
        setProperty("line"+count, _y, _ymouse);
        setProperty("line"+count, _x, _xmouse);
    }
}
```

```

this["line"+count].line.line_move = true;
dot_count++;
} else if (dot_count == 1) {
// 左键第二次点击
this["line"+count].line.line_move = false;
count++;
dot_count = 0;
}
}

```

2.2 矢 量

矢量的效果图如图 2 所示, 设计思想与辅助线相同, 但要注意起始点不再由光标位置决定, 而是固定在某一点上, 如力的作用点. 在表示该矢量的符号周围, 还需要一些输出文本框来显示矢量信息, 如角度、下标、上标等. 上标和下标也可以改为输入文本框 (Input Text) 由用户自行输入. 矢量生成与辅助线生成的方法相同, 在作用点上添加按钮以形成热区, 程序为:

```

on (release) {
attachMovie("red_force", "forceB_"+countB,
force_depth);
setProperty("forceB_"+countB, _y, pointB._y);
setProperty("forceB_"+countB, _x, pointB._x);
// 下标为 B
this["forceB_"+countB].F_text.sub = "B";
// 不显示上标
this["forceB_"+countB].F_text.ubsub = "";
countB++;
force_depth++;
}

```

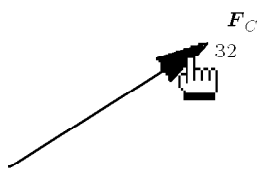


图 2 矢量效果图

2.3 工具栏

工具栏中的按钮用于切换不同的作图功能, 一个简单的工具栏可如图 3 所示, 图 3 中包括辅助线和力矢量两种作图功能. 为达到切换作图功能的目的, 可利用 setProperty() 命令调节相应功能热区的 _visible 属性, 设为 true 时, 热区有效, 可实现该热区功能; 反之 _visible 属性设为 false, 相应功能被屏蔽.

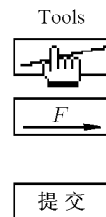


图 3 工具栏效果图

点击工具栏中的“提交”按钮, 表示向计算机发送作图结束指令, 计算机将判断作图结果的对错. 利用 fscommand() 命令可使作图信息向外传递, 传递的信息包括作图结果的对错、每一矢量旋转的角度和长度、输入的下标等. 本文的外部程序是 Authorware, 在 Authorware 中添加 ActiveX 控件便可接收到从 Flash 传递的信息. 该按钮的程序为:

```

on (release) {
if (判断条件) {
// 结果发送至 Authorware
fscommand("variable", "正确");
// 发送作图信息
.....
} else {
// 结果发送至 Authorware
fscommand("variable", "错误");
}
}
}

```

3 举 例

静力学中的三力汇交习题需要使用辅助线来完成作图. 如图 4 所示, 利用本文设计的程序, 可以先做出两个已知方向的力矢量, 再用辅助线延长这两个力矢量, 得到一个交点, 最后将交点和未知方向力矢量作用点连接, 便确定了最后一个力矢量的方向.

在运动学、动力学部分中, 会遇到其它一些矢量, 如速度、加速度等. 为明确各矢量意义, 不同类矢量可选择不同颜色表示, 而程序代码无需再做改动. 图 5 是一道利用基点法求速度的运动学题, 在这类题中需要对速度矢量分解, 学生可利用辅助线画平行四边形的方法来完成, 而矢量的下标、角度、长度等信息也将被传递以供文字求解部分使用. 为满足其它需要, 工具栏中有时还需添加新的按钮, 图 6 是用瞬心法求解同一道题, 程序还要判断瞬心的位置是否正确, 标出瞬心的程序设计方法也与前类同.

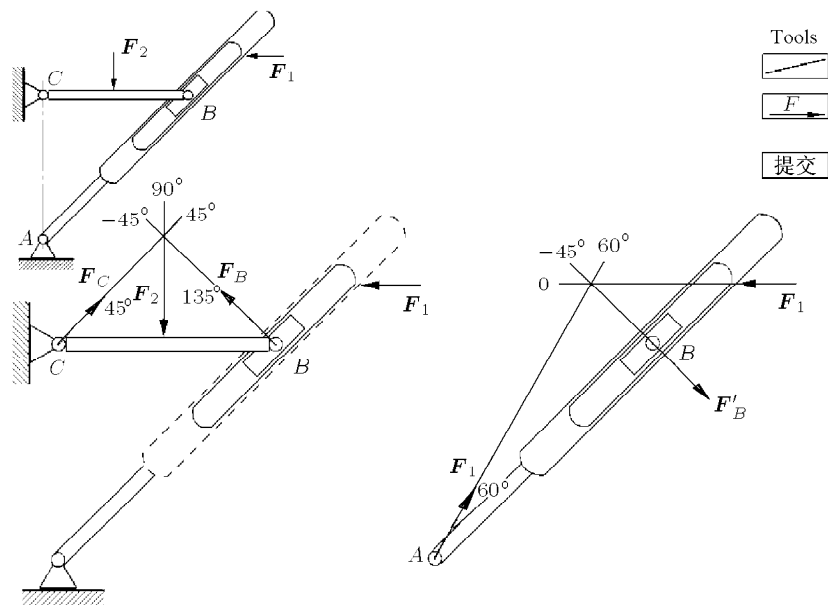


图 4 三力汇交

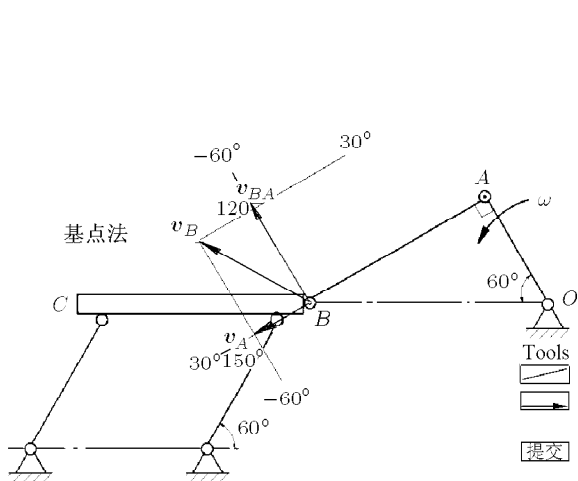


图 5 基点法求速度

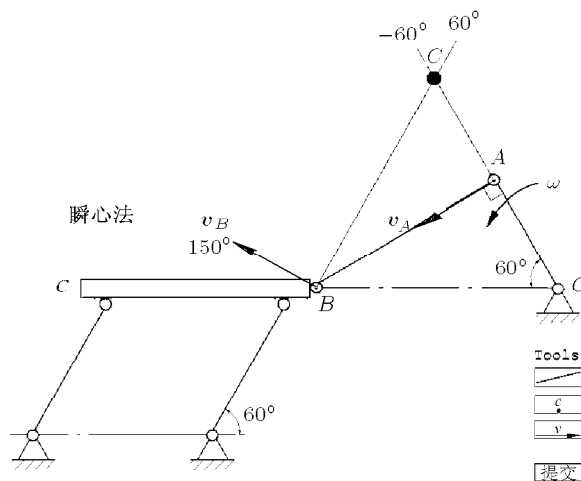


图 6 瞬心法求速度

4 结 语

本文详尽介绍了如何利用 Flash 软件及其编程技术开发理论力学网络作业，本文的方法在稍加训练后即可迅速掌握。Flash 软件同样也是一个优秀的绘图软件，习题原图也可由 Flash 来绘制。由于程序代码具备了可重复使用性，开发者在开发不同习题时，只需完成绘图工作及编写判断条件

即可，因此提高了开发效率。

参 考 文 献

- 1 颜金桄，KCLY 小土豆工作室. 闪客实战 ——Flash 高级编程. 北京：电子工业出版社，2003
- 2 哈尔滨工业大学理论力学教研室. 理论力学 (I). 北京：高等教育出版社，2002