二维布朗运动仿真可行性探究

——仿真代码实现

余柯润 19307110540

赵玉雪 20300190010

组号：B17

目录

介绍

- 我们想要解决什么问题？
  ——二维小球碰撞模型是否能模拟布朗运动(颗粒在流体中做无规则运动)
- 模拟算法有何亮点？
  ——Vue.js框架+完整的输入输出交互
- 如何得出结论？
  ——将模拟的输出和布朗运动的理论计算进行对比（队友）

难点

为什么“二维小球碰撞模型是否能模拟布朗运动”是个问题？

布朗运动：颗粒在流体中做无规则运动

1. 颗粒和液体分子的半径比：100～1000
2. 液体分子密度高→小球数量多
3. 所有小球之间的碰撞都需要模拟 ～O(n^2)

难点： 所有小球之间的碰撞都需要模拟

ChatGPT的建议：

1.高级算法

1. **空间哈希**：这种技术将空间划分为固定大小的格子，并为每个格子分配一个哈希值。每个物体都根据其位置被分配到一个或多个格子。通过检查每个格子内的物体，你可以只比较那些在相同格子或邻近格子内的物体。这种方法的优点是实现简单，灵活性高，可以应用于动态和非均匀的环境。缺点是需要正确选择格子的大小，以便在减少碰撞检测的次数和避免过多的格子之间找到平衡。
2. **树结构，例如四叉树或八叉树**：四叉树（2D）和八叉树（3D）是分层数据结构，用于将空间划分为更小的区域。这些区域可以进一步划分，直到每个区域内只有一个或没有物体。这些数据结构的优点是它们可以高效地处理动态物体，而且适用于非均匀分布的物体。缺点是它们需要更复杂的代码，可能需要更多的内存，并且需要处理物体跨越区域边界的情况。
3. **扫描线算法**：这种方法基于将物体排序，并在某个方向上（例如，x轴）扫描这些物体。每个物体都有一个开始点和结束点。当扫描线遇到开始点时，它会将物体添加到当前活动的物体集合中；当遇到结束点时，它会将物体从集合中移除。这样，你只需要检查当前活动集合中的物体，而不是所有物体。扫描线算法的优点是它可以在动态环境中高效地工作，但是它需要物体在扫描方向上不重叠，并且需要频繁地排序物体。

2. Web Workers并行计算

1. **创建一个worker文件** (worker.js):

```javascript
self.onmessage = function(event) {
    var balls = event.data;
    for (var i = 0; i < balls.length; i++) {
        for (var j = i + 1; j < balls.length; j++) {
            // 碰撞检测和处理的代码
        }
    }
    self.postMessage(balls);
}
```

这个worker文件接收主线程发送的小球数组，然后执行碰撞检测和处理，最后将处理后的小球数组发送回主线程。

2. **在主线程中创建并使用worker**:

```javascript
var worker = new Worker('worker.js');
var balls = [...]; // 初始化小球数组
worker.postMessage(balls);
worker.onmessage = function(event) {
    balls = event.data;
    // 更新小球的位置
}
```

代码实现

runMotion()

在每一帧都被调用：清空canvas，绘制球的新位置

HandleCollisionAndSpeed()

遍历所有的球:

- 更新每个球的位置(当前位置和速度向量决定)
- 检查球是否越界，将越界球的速度分量反转
- 只检查大球是否与其他小球发生碰撞

collideBalls():

- 两个球之间的距离如果小于或等于它们的半径之和，在碰撞方向使用一维弹性碰撞的公式来计算新的速度
- 防止小球在碰撞后的一帧中重叠

代码实现

输入

用户可以直接在页面上调节大小球半径比、质量比、小球数量和速度

输出

大球的速度和移动距离会以折线图实时反应在页面上

每间隔1s：

- 在页面输出大球的速度和移动的距离
- 将速度和移动的距离推送到chartOptions对象中，记录历史数据并作折线图

代码实现

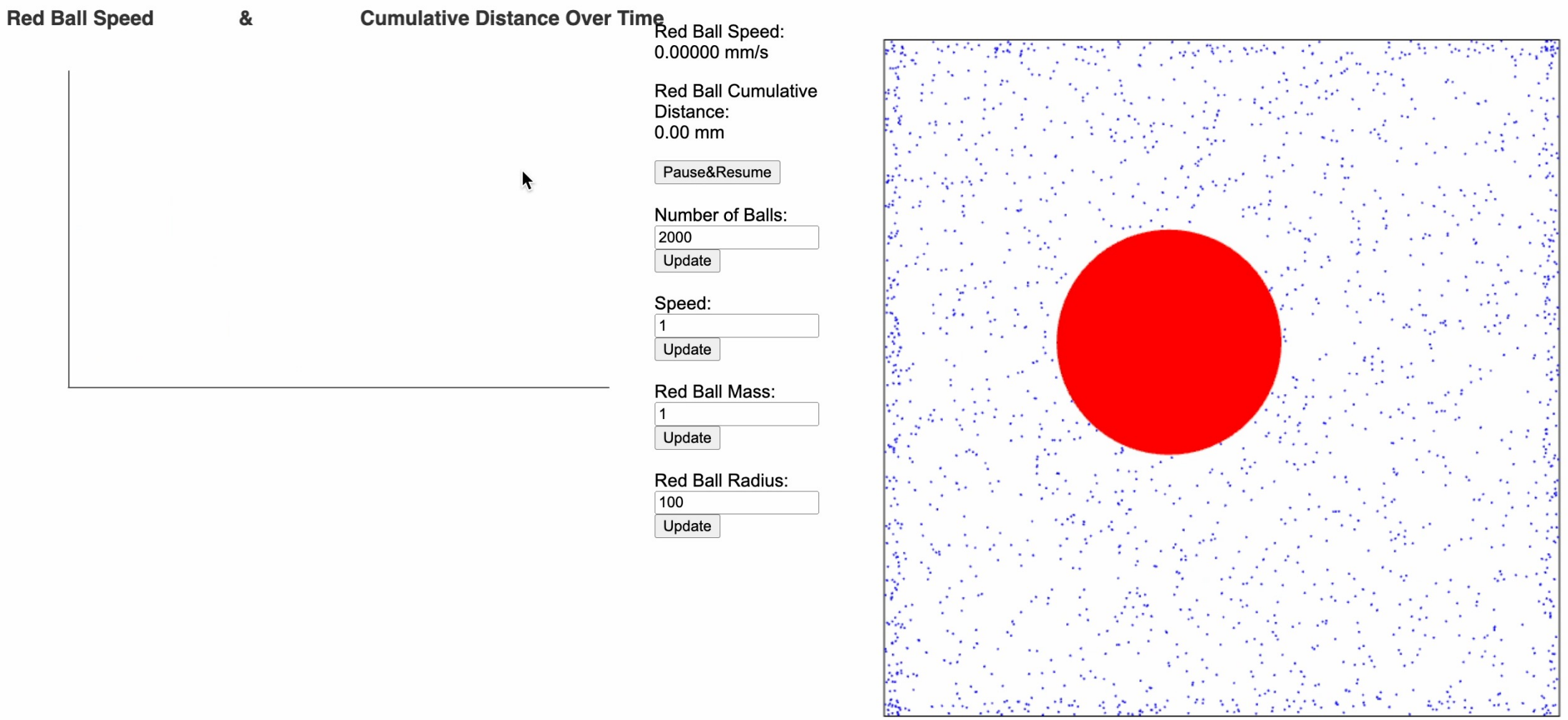
多个用户交互，全部”getElementById”?
折线图的点和线自己在canvas上手画？

Vue.js

- 响应式——改变数据时，视图会自动更新
  e.g., v-model双向绑定

- 方便的组件
  e.g., vue-echarts组件，不需要手动绘制点和线

结果展示


Red Ball Speed & Cumulative Distance Over Time
Red Ball Speed:
0.00000 mm/s
Red Ball Cumulative Distance:
0.00 mm
Pause&Resume
Number of Balls:
2000
Update
Speed:
1
Update
Red Ball Mass:
1
Update
Red Ball Radius:
100
Update

谢谢