

# 计算机应用数学实验报告

王子豪

May 21, 2016

## 1. 引言

基于物理的视觉效果模拟技术一直是计算机图形学领域热点研究内容。本文基于 Unified particle physics for real-time applications, 展示了一个统一化的实时物理模拟系统框架。本文以由约束互相连接的粒子作为统一的构建物体的基本元素, 这使得我们可以用统一化的方法处理物体的接触与碰撞。该框架的核心是一个以基于位置的动力学为基础的统一化求解器。求解器将并行的求解基于位置的动力学的二次规划问题, 且所有针对粒子的操作都在位置层级上, 这样虽然牺牲了一部分的真实性, 却可以利用 GPU 并行计算加速模拟流程, 从而达到实时模拟的目的。本文实现了 Unified particle physics for real-time applications 中固体模拟的部分, 通过形状匹配、基于有向距离场的碰撞约束、位置层级的摩擦力约束描述了物体的物理行为, 实现了刚体和变形体的实时模拟, 并解决了许多物理模拟中的常见问题。

## 2. 方法概述

统一化的求解器可以用一段简单的程序模拟多种物体。现在统一化求解器已经成为视觉效果模拟的流行工具, 但这些求解器通常只能进行离线的模拟。本文以基于位置的动力学模型为基础, 构建了一个统一化物理模拟实时系统, 在保持视觉可信性的前提下, 借助基于 GPGPU (General Purpose GPU, 通用计算图形处理器) 的大规模并行计算技术对系统进行加速, 从而达到实时渲染的目的。

本文给出的物理模拟系统以粒子为描述系统的基本单位, 每个粒子记录其位置、速度、质量的倒数、相标示等数据。系统根据物理定律和物体特性给出相应约束, 并通过这些约束组织粒子, 构建一个带约束的粒子系统。由于本系统以基于位置的动力学模型为基础, 所有的约束都是在位置层级施加的, 即所有约束函数的自变量都是模拟空间内的位置。模拟的关键在于求出每一时刻每个粒子  $i$  的位置  $x_i$ , 并根据此绘制粒子或重新构建三维模型 (三角形网格)。本文采用增量的思想进行计算粒子位置  $x$ 。已知上一时刻粒子位置, 只需求解粒子在当前时间步 (timestep) 内的位移增量  $\Delta x$ , 并在下一时间步开始时将该增量叠加到粒子位置上即可。此外还需根据  $\Delta x$  和  $\Delta t$  更新粒子的速度  $v$ 。

根据高斯最小约束准则 (Gauss' s principle of least constraint), 本文把基于位置的动力学模型看成一个带一系列约束的二次规划问题

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \Delta x^T M \Delta x \\ & \text{subject to} && C_i(x + \Delta x) = 0, \quad i = 1, \dots, n \end{aligned} \quad (1)$$

其中  $x = [x_1, x_2, \dots, x_n]^T$  为记录所有粒子位置的向量,  $\Delta x$  为向量  $x$  在当前时间步内的增量。 $M = \text{diag}(m_1, m_2, \dots, m_n)$  为记录所有粒子质量的对角矩阵。由于粒子的数量众多, 该规划问题的优化变量数目也就非常多, 如果采用传统的方法求解将十分复杂。各粒子间以及各约束

间的相互关系也不利于算法的并行化。故本文经过一系列数学变换，最终将该规划问题转化为如下公式：

$$\Delta x_i = \frac{\omega}{n} \sum_n \nabla C_j \lambda_j \quad (2)$$

该公式的推导过程和含义将在第三章详细阐述。可以看出，各约束对各粒子造成的位置增量都可以独立求解，可以通过 GPU 的并行计算对程序进行加速。

本文给出了刚体和变形体的模拟算法。为了实现刚体和变形体的模拟，本文给出了如下约束：

- 碰撞约束：检测粒子碰撞的情况，保证粒子间互不穿透。
- 形状匹配约束：保持刚体粒子的相对位置不变，让变形体粒子在平衡位置附近变换。
- 摩擦力：通过操作粒子的相对切向位移实现动摩擦力和静摩擦力。

基于前文的描述，可以得出一个时间步内的模拟的流程：

1. 时间积分（time integration）
2. 预稳定迭代（pre-stabilization）
3. 主迭代
4. 更新粒子数据

时间积分主要分为两步：将外力的作用施加到粒子速度上（冲量公式），然后将速度的作用施加到粒子的位置上；预稳定迭代是为了处理每个时间步开始时粒子的非法状态；主迭代根据约束并行的求解出每个粒子在当前时间步内的位移 $\Delta x_i$ ，是模拟算法中最为核心的一步；最后根据主迭代求出的位移更新粒子的位置和速度，供渲染系统进行渲染并作为下一时间步的输入进行下一循环的运算。

本文描述的模拟框架的伪代码如下：

---

**算法 1** 模拟算法

---

```

1: for all 粒子  $i$  do
2:   应用外力  $\mathbf{v}_i \leftarrow \mathbf{v}_i + \Delta t \mathbf{f}_{ext}(\mathbf{x}_i)$ 
3:   预测位置  $\mathbf{x}_i^* \leftarrow \mathbf{x}_i + \Delta t \mathbf{v}_i$ 
4:   质量缩放  $m_i^* \leftarrow m_i e^{-kh(\mathbf{x}_i^*)}$ 
5: end for
6: for all 粒子  $i$  do
7:   找到相邻的粒子  $\mathbf{N}_i(\mathbf{x}_i^*)$ 
8:   检测碰撞
9: end for
10: while iter < stabilizationIterations do
11:    $\Delta \mathbf{x} \leftarrow \mathbf{0}, n \leftarrow \mathbf{0}$ 
12:   求解碰撞约束，得到 $\Delta \mathbf{x}$ 和 $n$ 
13:   更新原始位置  $\mathbf{x}_i \leftarrow \mathbf{x}_i + \Delta \mathbf{x} / n$ 
14:   更新预期位置  $\mathbf{x}^* \leftarrow \mathbf{x}^* + \Delta \mathbf{x} / n$ 
15: end while

```

---

---

```

16: while iter < solverIterations do
17:   for each 约束组 G do
18:      $\Delta x \leftarrow 0, n \leftarrow 0$ 
19:     求解所有约束, 得到 $\Delta x$ 和 $n$ 
20:     更新预期位置 $x^* \leftarrow x^* + \Delta x/n$ 
21:   end for
22: end while
23: for all 粒子  $i$  do
24:   更新速度  $v_i \leftarrow \frac{1}{\Delta t}(x_i^* - x_i)$ 
25:   更新位置  $x_i \leftarrow x_i^*$ 或冻结粒子
26: end for

```

---

### 3. 实验结果

本文在多个场景进行了物理模拟并渲染。这些场景包含了不同类型的物体, 规模也各不相同, 用户可以通过控制碰撞球与场景中的物体碰撞来进行交互, 从而可以观察不同类型物体的模拟效果。在模拟时, 可以对时间步长、重力加速度、质量缩放系数  $k$ , 静摩擦系数  $\mu_s$ , 动摩擦系数  $\mu_k$ , 形变系数  $\beta$ , 逐次超松弛系数  $\omega$  等模拟参数进行修改, 以便实时的观察不同模拟参数下的模拟结果。

表 1 各场景模拟结果

场景	粒子数	面片数	顶点数	FPS
刚体碰撞	1412	6144	3076	402
线性变形体	1412	6144	3076	399
二次变形体	1412	6144	3076	385
刚体堆叠	760	15360	7690	395
颗粒物质摩擦	16000	0	0	117
刚体 Bunny	1170	80000	40004	381
Bunny 堆叠	25800	1156000	578200	66

本文主要对以下几个场景进行了模拟:

1. 刚体碰撞: 2 个刚体立方体发生碰撞, 每个立方体包含 706 个粒子, 其三角形网格模型包含 3072 个面片, 1538 个顶点。

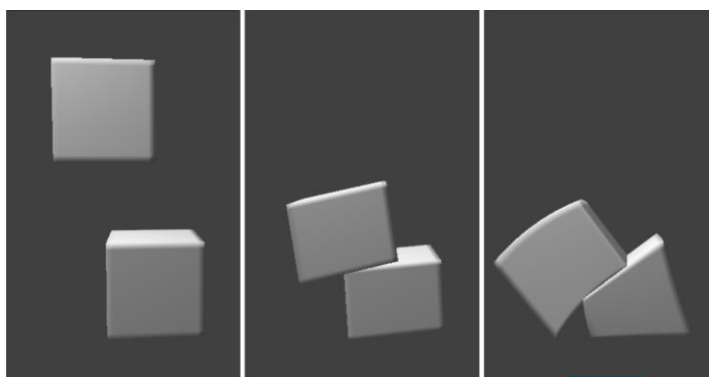


图 1 物体坠落后（左），线性形变（中）和二次形变（右）的结果

2. 线性变形体：2 个线性变形体立方体发生碰撞，其余参数同上。形变效果见图 1（中）。
3. 二次变形体：2 个二次变形体立方体发生碰撞，其余参数同上。形变效果见图 1（右）。
4. 刚体堆叠：5 个刚体立方体依次堆叠，每个立方体包含 156 个粒子，其三角形网格模型包含 3072 个面片，1538 个顶点，如图 2 所示。

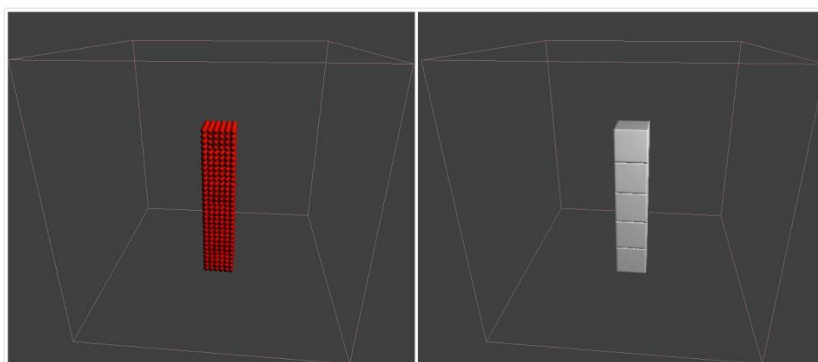


图 2 刚体堆叠，大约 250 帧后达到稳定状态

5. 颗粒物质摩擦：16000 个独立的粒子堆叠成长方体“沙塔”，碰撞球触碰“沙塔”顶端，令“沙塔”崩塌，如图 3 所示。

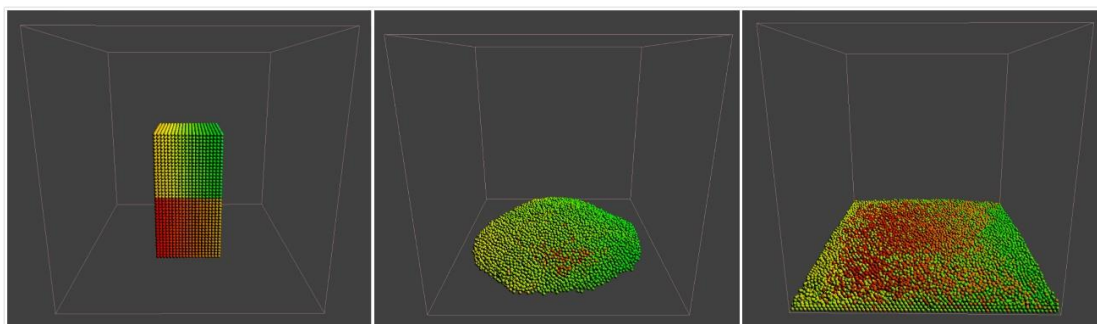


图 3 初始状态（左），带摩擦力（中），无摩擦力（右）

6. 刚体 Bunny：将两个 Stanford Bunny 的网格模型转化为刚体粒子模型，每个模型

包含 585 个粒子，其三角形网格模型包含 40000 个面片，20002 个顶点。该场景主要是为了测试不规则物体的模拟效果。

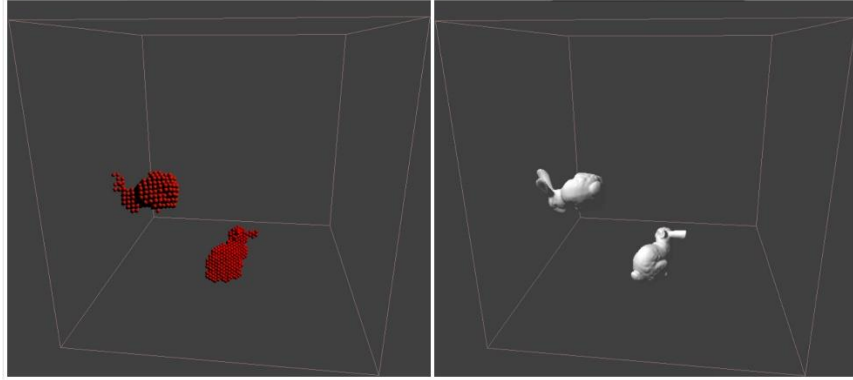


图 4 刚体 Bunny

7. Bunny 堆叠：100 只 Bunny 依次从空中落下，每个 Bunny 包含 256 个粒子，其三角形网格模型包含 11560 个面片，5782 个顶点。该场景主要是为了测试大量不规则物体的模拟性能，如图 5 所示。

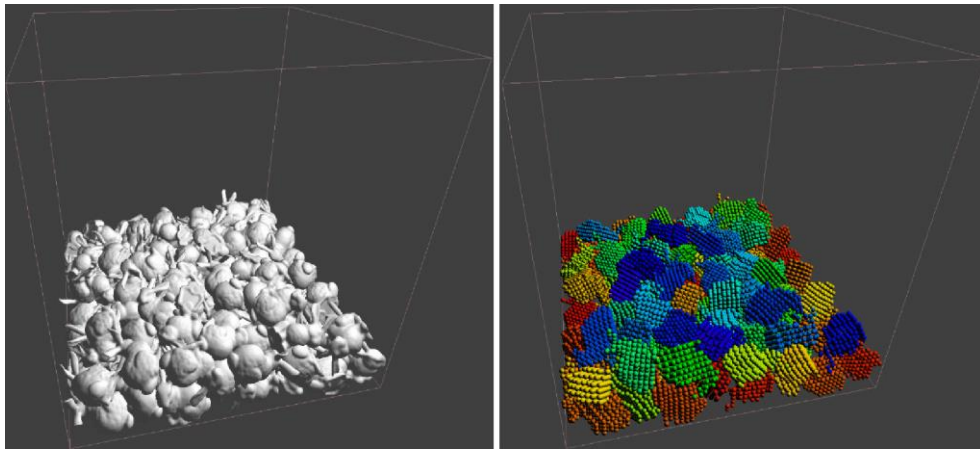


图 5 网格模型（左）与粒子模型（右）

## 4. 小结与分析

从表 1 中可以看出，本文描述的基于粒子的物体模拟系统在各种场景中均可以实现实时的模拟。由于本系统的绝大部分运算都是在 GPU 上并行的运算，所以模拟时每个时间步内的时间开销并不是随着场景规模线性增加的，即使是最为复杂的 Bunny 堆叠，其模拟时的 fps 仍然能达到 66，满足实时模拟（fps > 30）的要求。

由于物体是由粒子构成，粒子表面无法与网格模型完全贴合，当碰撞物体的速度过快时系统在时间步结束时也可能尚未收敛，所以在模拟的过程中会出现一些物体相互穿透的现象，这是该模型固有的缺陷，但大部分情况下不易察觉。通过增加每一帧时间步的个数并缩短时间步的长度可以缓解这一现象。

基于物理的视觉模拟系统一直是计算机图形学中重要的研究课题，有着深刻的理论价值和广泛的实际应用。本文描述了一个基于粒子的并行物理模拟系统，该系统以粒子为构建物体的基本单位，以 position based dynamics 为物理基础，通过等一系列约束描述粒子的行为：

1. 通过形状匹配（shape matching）约束维持刚体的形状，并在此基础上实现了物体的线性形变和二次形变。
2. 通过统一化网格并行地进行局部搜索以实现碰撞检测。
3. 通过互不穿透约束实现的粒子的碰撞，并采用基于 SDF 刚体碰撞模型解决了刚体互相锁定的问题。
4. 通过位置层级的摩擦力模型实现的静摩擦力和动摩擦力的模拟
5. 通过质量缩放解决了刚体堆叠不稳定的问题。

本文所描述的物理模拟系统的大部分运算均利用 CUDA 技术在 GPU 上并行地执行，模拟过程可以高效的完成，以达到实时模拟的目标

## 参考文献

- [1] Macklin M, Müller M, Chentanez N, et al. Unified particle physics for real-time applications[J]. ACM Transactions on Graphics (TOG), 2014, 33(4): 153.
- [2] Müller M, Heidelberger B, Hennix M, et al. Position based dynamics[J]. Journal of Visual Communication and Image Representation, 2007, 18(2): 109-118.
- [3] Müller M, Heidelberger B, Teschner M, et al. Meshless deformations based on shape matching[C]//ACM Transactions on Graphics (TOG). ACM, 2005, 24(3): 471-478.
- [4] Green S. Cuda particles[J]. nVidia Whitepaper, 2008, 2(3.2): 1.