

1-2-2019

## Collision Detection Algorithm of Deformable Object Based on Snake Model Optimization

Li Zhao

*School of Computer Science and Control Engineering, North University of China, Taiyuan 030051, China;*

Yanxia Jin

*School of Computer Science and Control Engineering, North University of China, Taiyuan 030051, China;*

Zhipeng Qin

*School of Computer Science and Control Engineering, North University of China, Taiyuan 030051, China;*

Ren Chao

*School of Computer Science and Control Engineering, North University of China, Taiyuan 030051, China;*

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research, Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

---

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

---

## Collision Detection Algorithm of Deformable Object Based on Snake Model Optimization

### Abstract

**Abstract:** In view of the existing collision detection algorithm which is difficult to solve the collision of real-time and reality, an improved algorithm of collision detection was proposed based on deformable objects. The algorithm based on K-DOPs bounding box applied to update process *used the improved PSO algorithm to optimize the Snake model boundary*. The concave in collision region, the control point clusters and the center formed multi-linear swarm *with the improved selection method and the ulti PSO algorithm*. Experimental results show that the improved collision detection algorithm can accurately simulate object contour, simplify the bounding box to improve the efficiency of collision detection.

### Keywords

detection, (K-dops) bounding box, multi-linear swarm, particle swarm optimization algorithm, snake model

### Recommended Citation

Li Zhao, Jin Yanxia, Qin Zhipeng, Ren Chao. Collision Detection Algorithm of Deformable Object Based on Snake Model Optimization[J]. Journal of System Simulation, 2018, 30(1): 62-68.

# 基于优化 Snake 模型的变形物体碰撞检测算法研究

李照, 靳雁霞, 秦志鹏, 任超

(中北大学 计算机与控制工程学院, 山西 太原 030051)

**摘要:** 针对现有碰撞检测算法难以解决变形物体碰撞的实时性和逼真性问题, 提出一种基于可变形物体的改进碰撞检测算法。该算法在 K-dops(K-discrete orientation polytopes)包围盒基础上使用 Snake 模型应用于包围盒更新过程中, 使用改进的 Pso(Particle swarm optimization)算法对 Snake 模型进行边界优化。在优化过程中依据碰撞部位的内凹特性, 在 Snake 模型控制点簇与中心点构成的多直线上, 采用改进的粒子选取方式和多种群粒子优化算法思想构成多直线群体。实验结果表明, 改进后的碰撞检测算法可以快速准确的模拟物体轮廓, 简化了包围盒更新过程, 提高了碰撞检测效率。

**关键词:** 碰撞检测; K-dops 包围盒; 多直线群体; Pso 算法; Snake 模型

中图分类号: TP391.9

文献标识码: A

文章编号: 1004-731X (2018) 01-0062-07

DOI: 10.16182/j.issn1004731x.joss.201801008

## Collision Detection Algorithm of Deformable Object Based on Snake Model Optimization

Li Zhao, Jin Yanxia, Qin Zhipeng, Ren Chao

(School of Computer Science and Control Engineering, North University of China, Taiyuan 030051, China)

**Abstract:** In view of the existing collision detection algorithm which is difficult to solve the collision of real-time and reality, an improved algorithm of collision detection was proposed based on deformable objects. The algorithm based on K-DOPs bounding box applied to update process used the improved PSO algorithm to optimize the Snake model boundary. The concave in collision region, the control point clusters and the center formed multi-linear swarm with the improved selection method and the multi PSO algorithm. Experimental results show that the improved collision detection algorithm can accurately simulate object contour, simplify the bounding box to improve the efficiency of collision detection.

**Keywords:** detection; (K-dops) bounding box; multi-linear swarm; particle swarm optimization algorithm; snake model

## 引言

碰撞检测是计算机图形学、计算机虚拟现实等诸多领域中研究的一项重要内容。随着计算机技术的不断发展, 对虚拟环境中的物体模型要求越来越

高。因此, 对物体碰撞的过程的实时性和逼真性要求也越来越严格。这使得如何快速提高碰撞检测效率和物体真实性变的尤为重要。

当今大规模复杂场景越来越多, 因此对变形物体碰撞研究变的越来越重要。碰撞检测在空间域上主要有两大经典算法: 层次包围盒法<sup>[1-2]</sup>和空间分割法<sup>[3]</sup>。针对变形物体碰撞检测算法改进提出了很多观点, 例如李文辉等<sup>[4]</sup>提出了将粒子群算法应用于变形体自适应随机碰撞检测中。李艳波等<sup>[5]</sup>利用传统 Snake 模型的特点, 将能量函数应用到 AABB



收稿日期: 2015-12-08 修回日期: 2016-04-25;  
基金项目: 国家自然科学基金项目(61379080), 山西省自然科学基金项目(2013011017-7);  
作者简介: 李照(1993-), 男, 山西运城, 硕士生, 研究方向为虚拟现实技术。

<http://www.china-simulation.com>

包围树的更新过程中, 提高了更新的效率。但传统 Snake 模型比较依赖初始轮廓, 若初始轮廓不接近目标无法获取正确目标轮廓并且在目标凹陷处无法收敛, 因此无法准确模拟碰撞部位轮廓。

根据以上存在问题, 本文提出一种优化 Snake 模型碰撞检测算法, 该算法利用多直线群体 PSO 优化后的改进 Snake 模型实现对包围盒的更新, 提高碰撞检测更新效率。

## 1 K-DOPs 包围盒碰撞检测

K-DOPs 包围盒<sup>[6]</sup>顾名思义, 它有  $K$  个固定方向向量并且由  $K$  个超平面包围而成。每个方向向量即为对应超平面的法向量。由定义可知<sup>[7]</sup>,  $K$  个方向向量为  $K/2$  个互为方向相反的向量对。

层次包围盒算法是碰撞检测经典算法之一, 该算法关键在于构造层次包围盒树。本文采用自上而下的方式进行构造, 由于使用不同的遍历策略<sup>[8]</sup>, 构造  $d$  叉树的消耗比不同。本文遍历采用“ $A$ 、 $B$  同时下降”策略, 当  $d=2$  时消耗比为  $f(d)=d^2 \log_d(n)$  最小, 因此选择 2 叉树。遍历时,  $A$  树和  $B$  树同时遍历, 此遍历策略可以快速遍历至叶节点。若节点发生碰撞, 节点包围盒进行更新并采用自下而上更新层次包围盒树的每个节点。

## 2 传统 Snake 模型

Snake 模型<sup>[9]</sup>定义一条封闭曲线, 这条曲线有  $N$  个控制点集构成, 每个控制点在  $E_{int}$  和  $E_{ext}$  的作用下向目标轮廓靠近并不断迭代更新, 最终得到能量最小的曲线。Snake 模型目标能量函数  $E_{snake}$  定义如公式(1)所示, 其中,  $v(s)=(x(s), y(s))$ ,  $s \in [0,1]$  表示曲线上的控制点,  $x, y$  为控制点坐标。 $E_{int}$  是对 Snake 曲线平滑性和连续性的约束,  $E_{ext}$  是对图像的外在约束。

$$E_{Snake} = \int_0^1 E_{Snake}(v(s))ds = \int_0^1 [E_{int}(v(s)) + E_{ext}(v(s))]ds \quad (1)$$

文献[10]提出一个向心能量  $E_{center}$ , 它可以使

轮廓凹陷处收敛并可以加快收敛速度, 离散化如公式(2)所示, 其中  $s$  为当前运动点  $i$  周围指定区域;  $P$  为目标物体选定一点;  $k$  为迭代次数;  $\|v_i - p\|$  为  $s$  区域内一点到  $P$  点的距离;  $\max(k)$  为  $k$  次迭代中,  $s$  内一点到  $P$  的最大距离;  $\min(k)$  为最小距离;  $\varepsilon(i)$  为向心能量系数。

$$E_{center} = \varepsilon(i)E_{center}(i, k) = \frac{(\|v_i - p\| - \min(k))}{\max(k) - \min(k)} \quad (2)$$

此外, 为了使控制点能均匀分布于指定区域, 文献[11]增加了一个  $E_{equal}$  控制能量, 它可以均匀动态分配控制点的位置, 可以弥补曲线在曲率较大的地方控制点分布不均匀的情况。离散化后如公式(3)所示, 其中  $i$  一般取偶数个控制点,  $v_i$  表示轮廓曲线上第  $i+1$  个控制点,  $\eta(i)$  为控制系数,  $d=(|v_i - v_{i-1}| + |v_{i+1} - v_i|)/2$  表示 3 个控制点的距离平均值。当  $v_{i-1}$  到  $v_i$  的距离比  $d$  小, 则  $v_i$  向  $v_{i+1}$  动态移动。

$$E_{equal} = \eta(i)E_{equal}(i, k) = \eta(i) \left| \frac{|v_i - v_{i-1}| + |v_{i+1} - v_i|}{2} - |v_i - v_{i-1}| \right| \quad (3)$$

令

$$\mu = \frac{(v_{i-1} - v_i)(v_{i+1} - v_i)}{|(v_{i-1} - v_i)| |(v_{i+1} - v_i)|} \quad (4)$$

式(4)中  $\mu$  的变化与 Snake 模型曲率成反比, 根据  $\mu$  判断曲率大小, 当曲率较大时, 若 3 点之间的最大距离  $|v_{i+1} - v_i|$  大于规定的阈值, 则在中间添加新的控制点, 若小于规定阈值, 则删除若干控制点。

在碰撞检测算法中, 为了更好的在初始位置实现目标轮廓的粗收敛, 本文综合采用  $E_{equal}$  和  $E_{center}$  实现均匀取点和凹处收敛, 如公式(5)所示, 经离散化如公式(6)所示。

$$E_{Snake} = \int_0^1 [E_{int}(v(s)) + E_{ext}(v(s)) + E_{equal}(v(s)) + E_{center}(v(s))]ds \quad (5)$$

$$E_{Snake} = \sum_{i=0}^{n-1} [\alpha(i)E_{conn}(i, k) + \beta(i)E_{cur}(i, k) + \gamma(i)E_{ext}(i, k) + \varepsilon(i)E_{center}(i, k) + \eta(i)E_{equal}(i, k)] \quad (6)$$

式(6)中  $E_{conn}(i,k)$ 和  $E_{cur}(i,k)$ 分别表示连接能量和弯曲能量;  $i$ 表示曲线上的第  $i$ 个点,如  $E_{ext}(i,k)$ 表示第  $i$ 个点在第  $k$ 次迭代后的外部能量;  $n$ 表示曲线上控制点个数;图 1 为控制点运动受力示意图。

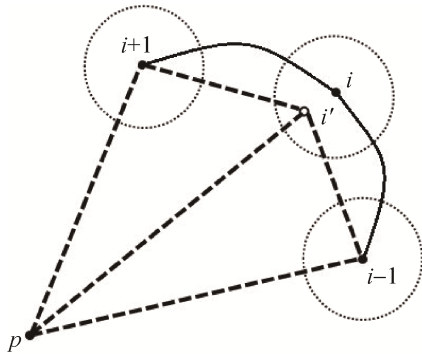


图 1 Snake 模型控制点受力图  
Fig. 1 Snake model control point force diagram

### 3 融合多直线群体 PSO 的 Snake 模型算法

#### 3.1 粒子群优化算法

粒子群优化(Particle Swarm Optimization, PSO)算法基本思想是受鸟类群体行为启发。Y.Shi 等<sup>[12]</sup>在原始方程中引入动态调整惯性权重  $\omega$  的概念来提高收敛速度,如公式(7)(8)所示。

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 r_{1j}(t)(p_{ij}(t) - x_{ij}(t)) + c_2 r_{2j}(t)(p_{gj}(t) - x_{ij}(t)) \quad (7)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (8)$$

#### 3.2 多直线群体 PSO 算法

由于在碰撞检测的过程中,碰撞点相邻的点都有可能发生碰撞,因此提出多直线群体 PSO 算法,该算法用控制点  $v_i(0 \leq i \leq n-1)$ 与中心点  $P$ 所构成直线上的  $m$ 个包括  $v_i$ 的点构成一个群体,群体内互相协作并在迭代过程中,每个群体可以和邻近群体共享信息搜寻该群体局部最优值,从而找到每个群体中最佳粒子为新的控制点带入公式(1)得到群体最佳适应值,不断迭代直到  $E_{snake}$ 趋于稳定。采用此算法,群体不会陷入局部僵局,从而得到一组更加准确的控制点。

### 3.3 改进 Snake 模型的粗收敛

碰撞检测中进行相交检测时,若两个节点包围盒检测到相交,使用本文所提出的改进 Snake 模型对包围盒所包目标物体轮廓进行粗收敛。在进行收敛的过程中会产生凹收敛(曲线位置位于目标边界内)和凸收敛(曲线位置位于目标边界外)的情况。但通过改进的 Snake 模型进行目标物体轮廓的粗收敛,不仅可以均匀地动态取得控制点的初始位置,而且可以减少迭代的计算量。

### 3.4 Snake 模型函数的选取

传统 Snake 模型和改进的 Snake 模型分别作为适应度函数。当对目标轮廓进行粗收敛时,使用改进的 Snake 模型作为适应度函数。在此后的迭代过程中,使用传统 Snake 模型为适应度函数。因为经过改进 Snake 模型粗收敛后,控制点已经均匀分布并完成粒子的选取,控制点已经基本确定,曲线对目标凹陷位置已经大致收敛,所以不需要使用改进的 Snake 模型,会最大程度减小凸收敛和凹收敛。

### 3.5 Snake 模型轮廓初始粒子选取

(1) 经过 3.3 粗收敛过程后,曲线上有  $n$ 个均匀分布的控制点  $\{v_0, v_1, \dots, v_{n-1}\}$ ,取曲线所包围的区域中心点  $P$ ,将点  $v_i$ 和  $P$ 点进行连接并且在直线  $L_d(0 \leq d \leq n-1)$ 取  $m$ 个点  $\{\alpha_{i,0}, \alpha_{i,1}, \dots, \alpha_{i,m-1}\}$ ;

(2) 中心点  $P$ 的选取:通过该时刻曲线上所有控制点的坐标平均值确定,通过粗收敛后,  $n$ 个控制点坐标已经确定为  $(x_i, y_i)$ ,则  $P$ 点坐标为

$$\left( \frac{\sum_{i=0}^{n-1} x_i}{n}, \frac{\sum_{i=0}^{n-1} y_i}{n} \right);$$

(3) 在取值过程中,由于两物体的碰撞过程一般都是碰撞部位内凹的过程,因此,为了更快找到最佳粒子,规定:

将碰撞部位和对应包围盒投影到与对应固定向量平行并垂直与碰撞包围盒超平面的平面上,此平面可以显示碰撞的全部信息。根据  $K$ -DOPs 包围盒固定方向向量的定义,现选择碰撞部位包围盒超

平面所对应的向量为固定向量;

当包围盒与碰撞部位投影有交点时, 此交点作为  $L_d$  上最远点;

当包围盒与碰撞投影没有交点时, 取包围盒与  $L_d$  相交点为  $L_d$  上最远点, 并将该  $L_d$  上的控制点分布在粗收敛曲线两侧;

(4) 选取  $a_{ij}(0 \leq i \leq n-1, 0 \leq j \leq m-1)$  作为一个粒子, 共得到  $n \times m$  个粒子, 如图 2 为粒子选取示意图。

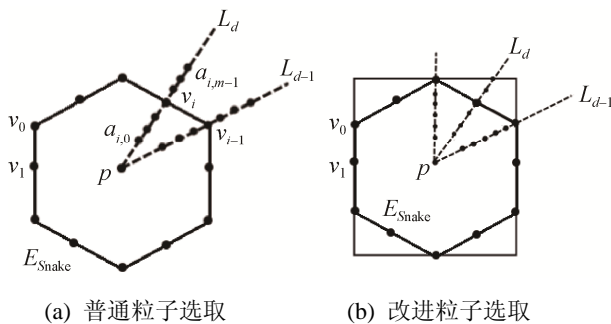


图 2 粒子选取示意图

Fig. 2 Schematic diagram of particle selection

### 3.6 融合算法的实现

Step 1: 使用改进 Snake 模型对初始轮廓进行粗收敛, 根据控制点集  $\{v_0, v_1, \dots, v_{n-1}\}$ , 进行粒子采集得到  $n \times m$  个粒子并初始化粒子位置  $x_{ij}$  和初始速度  $v_{ij}$ ;

Step 2: 初始化粒子群算法各个参数: 加速度因子  $c_1, c_2$ , 惯性因子  $\omega$ , 最大迭代次数  $k$ ;

Step 3: 根据 Step 1, 将  $L_d(0 \leq d \leq n-1)$  上  $m$  个点作为一个群体, 计算出每个粒子初始适应度值  $E_s(s=0, 1, \dots, m-1)$ , 比较得出种群最佳适应度值  $f_{sbest}$ , 种群全局最优值  $g_{best}$ , 个体最优值  $P_{best}$ ;

Step 4: 用公式(7)(8)更新粒子位置  $X_i=(x_{i1}, x_{i2}, \dots, x_{im-1})$  和速度  $V_i=(v_{i1}, v_{i2}, \dots, v_{im-1})$ , 并规定若粒子超出规定范围, 按边界值计算;

Step 5: 利用 Step 4 计算每个粒子的适应度值 (适应度函数为传统 Snake 模型函数)。比较当前粒子适应度值与个体最优适应值, 得到该粒子当前的局部最优值  $P_{best}$ , 并与相邻种群进行沟通可得种群

全局最优适应值, 再与  $f_{sbest}$  比较得到历史最佳适应值, 随之得到  $g_{best}$ ;

Step 6: 重复执行 Step 5, 可得到所有群体的  $g_{best_i}(0 \leq i \leq n-1)$ ;

Step 7: 将  $g_{best_i}(0 \leq i \leq n-1)$  带入式(1)得到轮廓曲线  $E_{Snake}$ , 经过不断迭代, 曲线已基本稳定;

为了更好地说明本文算法, 给出不同算法的效果图。图 3(a) 表示初始轮廓, 选择物体包围盒投影作为初始轮廓。图(b) 为传统 Snake 模型算法, 可以看出此算法在物体投影凹陷处收敛不明显。图(c) 为经典 GVF-Snake<sup>[13]</sup> 算法, 此算法虽然能实现凹陷处收敛, 但易产生凹收敛和凸收敛。图(d) 为本文算法, 由图可看出, 该算法相比于 GVF-Snake 在各角的位置进行轮廓提取时不会产生凹收敛和凸收敛现象, 有较好的抗干扰性显然好于其他算法。

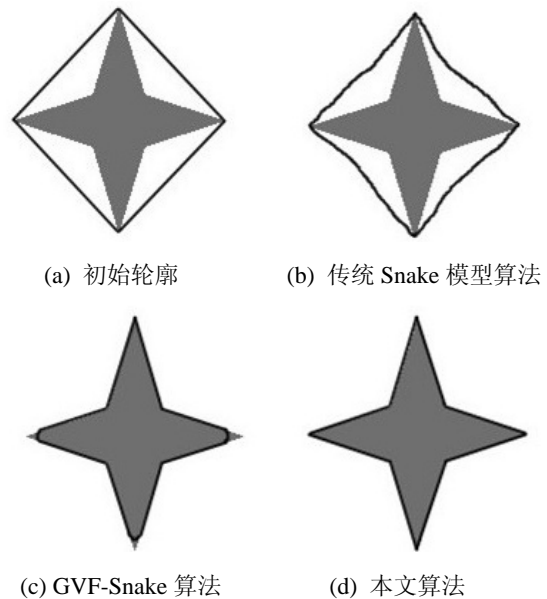


图 3 不同算法收敛结果

Fig. 3 Convergence results of different algorithms

## 4 融合 PSO 优化的 Snake 模型碰撞检测算法

首先构建层次包围盒树, 每个节点均采用 K-DOPs 包围盒, 当检测到发生碰撞时, 利用改进 Snake 模型对目标物体进行包围形成封闭曲线, 其次利用多直线群体 PSO 算法, 经过不断迭代, 得

到一个比较理想的曲线。根据此封闭曲线更新包围盒,之后采用自底向上和后序遍历的方式进行整棵树的更新。

改进后的算法首先将两目标物体用 K-DOPs 包围盒包围,采用二叉树构造层次包围盒树,通过“同时下降”遍历策略<sup>[14]</sup>遍历两棵二叉树,当检测到某节点发生碰撞时,物体形状发生改变,更新该节点包围盒。更新时,在物体碰撞部位发生形变的位置只更新形变部位对应的一个或多个包围盒面,然后将更新后的包围盒面与其余包围盒面构成新的包围盒,采用自下而上和后序遍历更新包围盒。

算法实现过程描述:

(1) 构造层次包围盒二叉树,利用“同时下降”遍历策略进行遍历,判断节点间是否发生碰撞;

(2) 若某节点发生碰撞,将此节点和对应包围盒投影到某一平面(由三维变为二维),投影后形成由包围盒和物体所组成的二维图像。若未发生碰撞,进行(6)操作;

(3) 将包围盒投影作为 Snake 模型的初始轮廓,并取若干个控制点,然后用改进 Snake 模型对初始轮廓进行粗收敛得到包围碰撞物体的一条封闭曲线;

(4) 利用多直线群体 PSO 算法和传统 Snake 模型,经过不断迭代,得到碰撞后的物体模型曲线进行包围盒更新;

(5) 采用自底向上和后序遍历对层次包围盒树进行更新;

(6) 进行下一次碰撞检测。

## 5 实验结果及分析

实验中进行粗收敛时,Snake 模型系数选择  $\alpha=1.2$ ,  $\beta=0.8$ ,  $\gamma=1$ ,  $\eta=0.5$ ,  $\varepsilon=1$ ; 精确收敛时,系数选择  $\alpha=1.2$ ,  $\beta=0.8$ ,  $\gamma=1$ 。实验中基本元素三角片数大约为 7000 片,Snake 模型控制点选择 30,每个控制点构成的点群为 4 个。实验效果如图 4 所示。

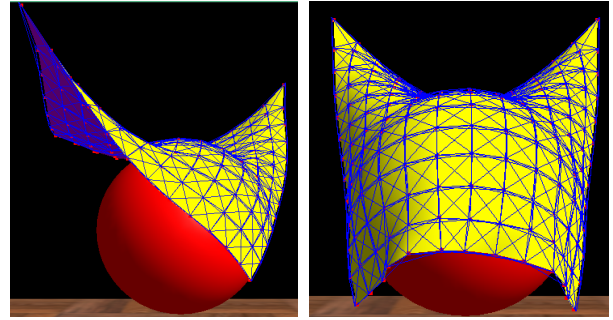


图 4 不同碰撞区域效果图

Fig. 4 Different collision area renderings

表 1 和图 5 表明了物体形变区域较小时不同算法的碰撞检测效率参数。可看出,本文算法和传统 Snake 模型算法在帧频、运行时间等方面都要显著高于普通 K-DOPs 算法。本文算法用多直线群体 PSO 对传统 Snake 模型进行优化并根据 PSO 算法快速收敛的特性,随着测试三角片数逐渐增加,图 5 中 3 条曲线开始平稳增长,3 种算法的运行时间相比较是本文算法所用时间最少,因此碰撞检测效率最高。在误差方面,3 种算法在 28 s 后基本趋于稳定。由于本文算法可以克服 Snake 模型的依赖初始位置和收敛不明显的缺点,因此误差得到提高。

表 1 实验数据比较

Tab. 1 Comparison of experimental data

碰撞检测算法	帧频/s	运行 1 K 步的时间/s	误差分析
K-DOPs 算法	7.12	1 834.0	1.24
传统 Snake 模型算法	16.25	803.7	0.54
本文算法	18.38	710.6	0.48

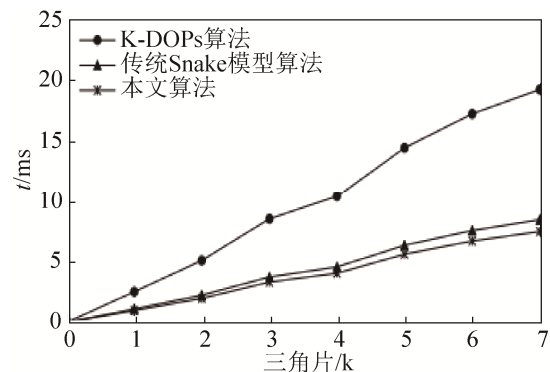


图 5 不同算法碰撞效率比较

Fig. 5 Comparison of collision efficiency between different algorithms

由表 2 和图 6 可以看出, 在物体变形区域较大时, 由于普通 K-DOPs 算法固定(每次碰撞进行  $k$  次比较运算), 所以在帧频、运行时间等方面基本不发生变化, 而传统 Snake 模型算法与本文算法效率都相应下滑, 误差也有所降低, 但本文算法从三角片达到 2 000 片开始相比其他 2 种算法, 碰撞效率提高较为明显。因此, 本文算法在整体上优于其他 2 种算法。

表 2 实验数据比较  
Tab. 2 Comparison of experimental data

碰撞检测算法	帧频/s	运行 1 K 步的时间/s	误差分析
K-DOPs 算法	7.12	1 834.0	1.24
传统 Snake 模型算法	8.67	1 506.1	1.01
本文算法	9.81	1 331.7	0.89

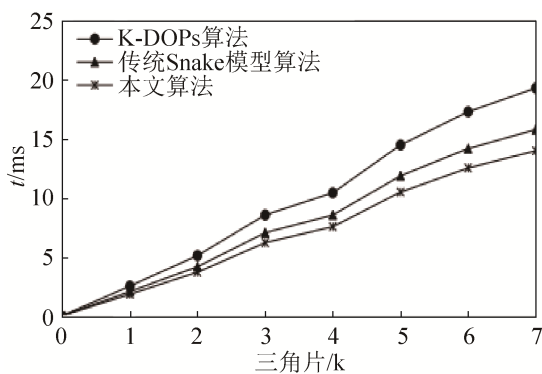


图 6 不同算法碰撞效率比较

Fig. 6 Comparison of collision efficiency between different algorithms

## 6 结论

本文针对现有碰撞检测算法的碰撞效率和逼真性研究, 提出一种基于改进 Snake 模型的碰撞检测算法。由于 Snake 模型收敛性不高, 特别在碰撞凹陷处不能准确收敛, 因此, 采用多直线群体 PSO 算法优化 Snake 模型, 以达到快速收敛的目的。实验结果证明, 本文提出的改进碰撞检测算法提高了碰撞检测效率, 并且在碰撞凹陷处可以很好收敛。由于碰撞场景越来越复杂, 特别在碰撞区域较大的环境中, 本文算法所涉及的计算量过大并且时间和效率也会受到一定影响, 所以还需要对算法进行进

一步的研究。

## 参考文献:

- [1] Du Peng, Tang Min, Tong RuoFeng. Fast Continuous Collision Culling with Deforming Non-collinear Filters [J]. Computer Animation&Virtual Worlds (S1546-427X), 2012, 23(3): 375-383.
- [2] 付跃文, 梁加红, 李猛, 等. 基于多智能体粒子群的快速碰撞检测算法研究[J]. 系统仿真学报, 2013, 25(8): 1876-1880.  
Fu Yuewen, Liang Jiahong, Li Meng, et al. Research on Fast Collision Detection Using Multi-Agent Particle Swarm Optimization[J]. Journal of System Simulation, 2013, 25(8): 1876-1880.
- [3] 黄欣, 杨文珍, 吴新丽, 等. 双触点虚拟织物触觉交互[J]. 系统仿真学报, 2014, 26(09): 1997-2008.  
Huang Xin, Yang Wenzhen, Wu Xinli, et al. Virtual Fabric Haptic Rendering with Double Haptic Interactive Points[J]. Journal of System Simulation, 2014, 26(9): 1997-2008.
- [4] 李文辉, 王天柱, 王祎, 等. 基于粒子群面向可变形物体的随机碰撞检测算法[J]. 系统仿真学报, 2006, 18(8): 2206-2209.  
Li Wenhui, Wang Tianzhu, Wang Wei, et al. Stochastic Collision Detection Between Deformable Models Using Particle Swarm Optimization Algorithm[J]. Journal of System Simulation, 2006, 18(8): 2206-2209.
- [5] 李艳波, 印桂生, 张菁. 基于 Snake 模型的碰撞检测算法研究[J]. 计算机应用研究, 2009, 26(3): 855-857.  
Li Yanbo, Yin Guisheng, Zhang Jing. Research on collision detection methods based on snake[J]. Application Research of Computers, 2009, 26 (3): 855-857.
- [6] Li Yunqi. Collision Detection Algorithm for Deformable Objects[J]. Journal of Jilin University (S1671-5489), 2012, 30(3): 285-293.
- [7] Christer Ericson. 实时碰撞检测算法技术[M]. 刘天慧译. 北京: 清华大学出版社, 2010: 81.  
Christer Ericson. Real-time collision detection algorithm [M]. translation by Liu TianHui. Beijing: Tsinghua university press, 2010: 81.
- [8] 陆睿, 刘卉. 基于完全二叉树 BVH 的自碰撞检测算法[J]. 计算机应用与软件, 2012, 29(12): 282-285.  
Lu Rui, Liu Hui. A Self-collision detection algorithm based on BVH of complete binary tree[J]. Computer Applications and software, 2012, 29(12): 282-285.
- [9] 周旻曦, 刘永学, 李满春, 等. 多目标珊瑚岛礁地貌



- 遥感信息提取方法——以西沙永乐环礁为例[J]. 地理研究, 2015, 34(4): 677-690.
- Zhou Minxi, Liu Yongxue, Li Manchun, et al. Geomorphologic information extraction for multi-objective coral islands from remotely sensed imagery: A case study for Yongle Atoll, South China Sea[J]. Geographical Research, 2015, 34(4): 677-690.
- [10] 王科俊, 郭庆昌. 基于粒子群优化算法和改进的 Snake 模型的图像分割算法[J]. 智能系统学报, 2007, 2(1): 53-58.
- Wang Kejun, Guo Qingchang. Image segmentation algorithm based on th PSO and improved Snake model [J]. CAAI Transactions on Intelligent Systems, 2007, 2(1): 53-58.
- [11] 钱建生, 厉丹. 基于 PSO 优化改进的 Snake 模型煤矿环境目标检测[J]. 煤炭学报, 2011, 36(11): 1949-1954.
- Qian Jiansheng, Li Dan. Snake model based on PSO optimization for coal mine enviroment object detection [J]. Journal of China Coal Cociety, 2011, 36(11): 1949-1954.
- [12] Shi Y, Eberhart R. Fuzzy Adaptive Particle Swarm optimizer[C]//Proceedings of IEEE International Conference on Evolutionary Computation. Korea: IEEE Service Center, 2001: 1045-1048.
- [13] 王根, 张建伟, 陈允杰, 等. 一种各向异性 GVF 模型的心脏左心室 MR 图像分割模型[J]. 计算机辅助设计与图形图像学报, 2010, 22(11): 1887-1891.
- Wang Gen, Zhang Jianwei, Chen Yunjie, et al. An Anisotropic GVF Model for the MR Image Segmentation of Left Ventricle[J]. Journal of Computer Aided Design and Computer Grapghics, 2010, 22(11): 1887-1891.
- [14] 孙劲光, 吴素红. 基于分类遍历的碰撞检测优化算法[J]. 计算机应用, 2015, 35(1): 194-197.
- Sun Jinguang, Wu Sugong. Collision detection optimization algorithm based on classified traversal [J]. Journal of Computer Applications, 2015, 35(1): 194-197.

### 《系统仿真学报》荣获“2017 中国国际影响力优秀学术期刊”证书

由中国学术期刊（光盘版）电子杂志社与清华大学图书馆联合成立的中国学术文献国际评价研究中心，发布了 2017 版《中国学术期刊国际引证年报》，《系统仿真学报》荣获“2017 中国国际影响力优秀学术期刊”。

《年报》（2017 版）采用的统计源期刊为 20192 种，涵盖 WoS 收录的 SCI 期刊 8874 种、SSCI 和 A&HCI 期刊 4645 种，ESCI 期刊 5578 种；增补期刊 1762 种。参照中外文学术期刊总被引频次、影响因子、半衰期等各项国际引证指标，计算期刊影响力指数(CI)，对国内 6210 种学术期刊排序，遴选了人文社科、自然科学与工程技术两个类别的 TOP10% 为国际影响力品牌学术期刊。TOP5% 以内的期刊为“最具国际影响力学术期刊”、TOP5-10% 之间的为“国际影响力优秀学术期刊”。