

8-13-2020

Particle Swarm Optimization Method Based on Weighted Dynamic Constraints for Interactive Intelligent Fish Swarm

Xingquan Cai

1. *School of Computer Science, North China University of Technology, Beijing 100144, China;;*

Buni Honghao

1. *School of Computer Science, North China University of Technology, Beijing 100144, China;;*

Mengxuan Li

1. *School of Computer Science, North China University of Technology, Beijing 100144, China;;*

Fengxia Li

2. *Beijing Laboratory of Intelligent Information Technology, Beijing Institute of Technology, Beijing 100081, 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.

Particle Swarm Optimization Method Based on Weighted Dynamic Constraints for Interactive Intelligent Fish Swarm

Abstract

Abstract: For particle swarm algorithm not being applied to the rapid evolution of virtual biological cluster in short range, a particle swarm optimization method was provided that oriented to interactive intelligent fish with weight dynamic constrained. *This method let particle swarm through the state of the particle separation, and dynamic constraint particle swarm. This method used the concept of "convergence coefficient manager" to retain the differential movement between the particles. On this basis, setting the evaluation function and using the dynamic constraint weights, the fast particle swarm was completed, applying to virtual biological cluster.* The experiments results show the best weights dynamic constraint effect in large-scale virtual biological cluster and intelligent fish pattern movement, and this method is more effect than common particle swarm optimization algorithm, and accelerates significantly. And this method has been used in the development of virtual aquarium system with stable and reliable.

Keywords

artificial intelligence fish, dynamic constraint weights, particle swarm, evaluation function

Recommended Citation

Cai Xingquan, Buni Honghao, Li Mengxuan, Li Fengxia. Particle Swarm Optimization Method Based on Weighted Dynamic Constraints for Interactive Intelligent Fish Swarm[J]. Journal of System Simulation, 2016, 28(10): 2490-2496.

面向可交互式智慧鱼群的权重动态约束的粒子群方法

蔡兴泉¹, 布尼泓灏¹, 李梦璇¹, 李凤霞²

(1. 北方工业大学计算机学院, 北京 100144; 2. 北京理工大学智能信息技术北京市重点实验室, 北京 100081)

摘要: 针对粒子群算法在短程迭代的状况下搜索精度差、波动大、粒子状态考证不足的问题, 提出了一种面向可交互式智慧鱼群的权重动态约束的粒子群算法。根据粒子状态将粒子群进行分离, 对粒子群进行动态约束管理, 并使用“系数收敛管理器”的概念保留了粒子间的差异化运动。设定估价函数, 采用权重动态约束, 完成粒子群的快速求解, 并使之应用于智慧鱼群模拟。结果表明, 在大规模虚拟生物集群移动中, 权重动态约束效果最好; 完成智慧鱼群运动时, 明显优于普通粒子群算法, 且速度明显加快。该方法已经很好的用在了自主开发的虚拟水族馆系统中, 运行稳定可靠。

关键词: 智慧鱼群; 权重动态约束; 粒子群; 估价函数

中图分类号: TP391

文献标识码: A

文章编号: 1004-731X (2016) 10-2490-07

Particle Swarm Optimization Method Based on Weighted Dynamic Constraints for Interactive Intelligent Fish Swarm

Cai Xingquan¹, Buni Honghao¹, Li Mengxuan¹, Li Fengxia²

(1. School of Computer Science, North China University of Technology, Beijing 100144, China;

2. Beijing Laboratory of Intelligent Information Technology, Beijing Institute of Technology, Beijing 100081, China)

Abstract: For particle swarm algorithm not being applied to the rapid evolution of virtual biological cluster in short range, a particle swarm optimization method was provided that oriented to interactive intelligent fish with weight dynamic constrained. *This method let particle swarm through the state of the particle separation, and dynamic constraint particle swarm. This method used the concept of "convergence coefficient manager" to retain the differential movement between the particles. On this basis, setting the evaluation function and using the dynamic constraint weights, the fast particle swarm was completed, applying to virtual biological cluster.* The experiments results show the best weights dynamic constraint effect in large-scale virtual biological cluster and intelligent fish pattern movement, and this method is more effect than common particle swarm optimization algorithm, and accelerates significantly. And this method has been used in the development of virtual aquarium system with stable and reliable.

Keywords: artificial intelligence fish; dynamic constraint weights; particle swarm; evaluation function

引言

近年来计算机产业和游戏产业飞速发展, 人



收稿日期: 2016-05-30 修回日期: 2016-07-14;
基金项目: 国家自然科学基金(61503005), 北京市自然科学基金(4162022), 北方工业大学长城学者(CC08);
作者简介: 蔡兴泉(1980-), 男, 山东, 博士, 副教授, 研究方向为虚拟现实。

们对于游戏娱乐的逼真度要求越来越高。尤其是, 近年来随着虚拟现实(VR, Virtual Reality)的流行, 虚实互动成为人们对于游戏娱乐需求的重要部分。在大规模集群角色互动时, 使用传统粒子群算法会导致角色交互性差、逼真度差、运动时角色相对独立。因此, 有必要研究面向可交互式智慧鱼群的权重动态约束的粒子群算法, 以提高可交互式智慧鱼群的交互性和逼真度。

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

• 2490 •

粒子群算法(Particle Swarm Optimization, PSO)又称微粒群算法, 该算法是一种基于种群行为的智能优化算法。是由 J. Kennedy 于 1995 年开发的一种演化计算技术^[1], 该技术来源于对一个简化社会模型的模拟。它可以模拟鸟群在觅食过程中迁徙和聚群行为, 利用群体智能搜索出较好的解。由于该算法基础可移植性强, 能够有效地应用于对虚拟角色的行为描述, 快速适应于虚拟角色行为方式的挖掘, 所以对该算法改进有一定的可行性和必要性。

本文针对粒子群算法在短程迭代的情况下搜索精度差、波动大、粒子状态考证不足, 无法应用于虚拟生物集群的快速演化的情况, 提出了一种面向可交互式智慧鱼群的权重动态约束的粒子群算法。该算法根据粒子状态将粒子群进行分离, 对粒子群进行动态约束管理, 并使用“系数收敛管理器”的概念保留了粒子间的差异化运动, 确保了粒子群整体快速找到准确的近优点, 改进了原搜索精度差和无状态判断的缺点, 使得该算法可以更加有效地应用于智慧鱼群的虚拟集群生物的快速模拟。实验证明, 该算法能够有效提升在短程迭代状况下, 粒子群算法的效率和搜索精度, 并能大幅度提高粒子群收敛效率和收敛准确率。

1 相关工作

粒子群算法收敛速度快、设置参数少且算法简单易实现, 最初是为了图形化的模拟鸟群运动, 通过观察动物的社会行为, 发现群体行为信息的一种演化模式。所以该算法一经提出就受到了重视, 并广泛应用于函数优化、模式分类等工程之中^[1]。但是由于粒子群算法存在搜索精度不足、易波动、对参数依赖高等问题, 许多学者都研究改进粒子群算法并取得了一定成果。其中, 莫思敏^[2-3]提出一种扩展的粒子群算法, 调节粒子群算法中的社会系数, 强化粒子间相对移动, 但该算法过于复杂, 应用难度高。姚玉成^[4]提出的混合离子交互微粒群算法优化了粒子群算法搜寻最优解的能力, 但该算法

参数过多, 在实际操作中难以产生最优效果。黄太安^[5]提出的一种改进的简化粒子群算法收敛速度较快、精度较高, 但对于输入数据有严格要求, 实际操作效果差。通常改进粒子群算法中采用相同的公式进行粒子间的进化迭代, 算法容错率、搜索精度都不高^[6-7], 且在粒子产生波动时无法有效地对粒子产生判断。只有在多代进化之后粒子间的移动序列才趋于稳定^[8]。

本文主要研究面向可交互式智慧鱼群的权重动态约束的粒子群算法, 在前人研究的基础上, 有效组织以粒子群算法为基础的鱼群运动, 提高可交互式智慧鱼群的运动拟真度, 为今后在虚拟水族馆中出现的大规模智慧鱼群应用起到了铺垫作用。

2 面向可交互式智慧鱼群的权重动态约束的粒子群算法

在应用于虚拟互动中的虚拟角色时, 使用传统的粒子群算法是并不能够快速的根据粒子状态、环境将群体中的个体移动到最佳区域内。本文所提粒子群算法是一种基于粒子群算法权重参数优化的高速算法, 该算法可以快速根据粒子状态实行差异化计算, 使粒子可以迅速进入最优解空间。具体粒子群权重系数动态约束步骤的流程图如图 1 所示。

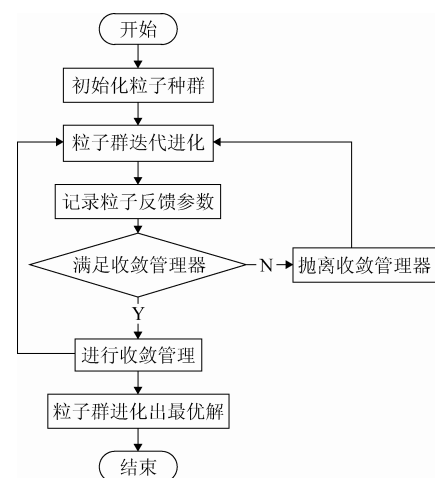


图 1 粒子群权重系数动态约束步骤

粒子群算法基础公式如公式(1)和公式(2)所

示, 在 D 维空间中有诸多内有体积的粒子, 第 i 个粒子表示为 $X_i = (X_{i1}, X_{i2}, \dots, X_{iD})$, 将每一个粒子所经历过的最好位置记为 $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$, 也称 p_{best} 。在微粒群中每个微粒经历的最好位置时, 使用索引号 g 表示, 即 P_g , 也称 g_{best} 。粒子 i 的速度使用 $V_i = (V_{i1}, V_{i2}, \dots, V_{iD})$ 表示。粒子根据如下方程不断迭代进化自己的最优速度和位置:

$$V_{iD} = W V_{iD} + C_1 r_1 (P_{best} - X_i) + C_2 r_2 (P_{best} - X_i) \quad (1)$$

$$X_{iD} = X_{iD} + V_{iD} \quad (2)$$

其中: W 为惯性权重(inertia weight); C_1 和 C_2 为加速常数(acceleration constants); r_1 和 r_2 为两个在 $[0,1]$ 范围里变化的随机值。为了防止粒子跳出解空间的搜索范围, 一般令 $V_i \in [-V_{max}, V_{max}]$ 。

由于普通粒子群算法在惯性飞行的途中受到恒定的认知系数和社会系数的影响, 粒子在寻找最优解位置时的收敛几率恒定, 这使得粒子在运动初期经历群组最优解点时会产生粒子波动效应和其伴生效应, 即粒子本身无法确定该解是否为最优解, 也无法通过认知系数感知自己的位置, 这样的问题会导致粒子再次寻找错误的位置。

针对这一点, 本文提出的权重动态约束粒子群算法, 通过收敛管理器的概念实现, 有效地解决了这一问题。在权重动态约束粒子群算法流程中, 收敛管理器的管理方式如何确定, 与收敛管理器的收敛条件如何确定是该算法核心问题。

2.1 粒子群中的权重系数动态约束步骤

在虚拟鱼群互动应用中, 以权重动态约束的粒子群算法为核心的智慧鱼群运动, 需要以动态约束粒子群算法的思想为基础, 粒子群系数收敛管理器为核心, 将粒子群中的权重系数作为动态收敛管理器, 粒子群中权重系数以动态约束方式调节, 基本步骤如下:

Step 1: 初始化一定数量的粒子 $X_{i1}, X_{i2}, \dots, X_{iD}$ 。

Step 2: 将当前粒子的反馈参数记录, 每一组粒子的相对位置记为 α , 每一个粒子和自己的相对位置记为 β 。

Step 3: 根据式(4)初始化认知系数收敛管理器。

$$C_1 = \begin{cases} 2 & \alpha > \rho \\ \frac{C_1}{\mu} & \alpha \leq \rho \end{cases} \quad (4)$$

其中: ρ 为认知收敛边界, 为该认知收敛系数管理器的边界控制系数, 其值越小, 对于粒子移动的边界限制就越大, 与粒子移动速度成反比。通常在 1.00~1.08 之间取值。 μ 为认知收敛系数, 用于控制粒子进入控制器后的移动速度, 其值越大, 对于粒子移动的速度限制就越大, 与粒子的移动速度成反比。

Step 4: 根据式(5)初始化社会系数收敛管理器。

$$C_2 = \begin{cases} 2 & \beta > \rho \\ \frac{C_2}{\gamma} & \beta \leq \rho \end{cases} \quad (5)$$

其中: ρ 为社会收敛边界, 为该社会系数收敛管理器的边界控制系数, 与粒子的相对移动速度成反比。通常在 1.05~1.09 之间取值。 γ 为社会收敛系数, 用于控制粒子之间的相对运动速度。其值与粒子相对运动速度成反比。

Step 5: 若粒子相对位置满足管理器要求, 则管理器内置函数将对粒子群系数进行操作, 减弱系数对于粒子群的影响, 将粒子稳定带入最优解范围, 直至粒子群进化若干迭代后完全处于管理器中。

Step 6: 若粒子相对位置不满足收敛管理器要求, 则抛离收敛管理器, 立即进入快速寻找最优解的进化。并在下一次迭代之中返回 Step 5。

2.2 粒子群中的收敛管理器管理方式

在应用于虚拟鱼群, 使用权重系数动态优化算法为核心时, 收敛管理器作为该算法的重点, 对于改进鱼群的运动轨迹质量及速度有着极为重要的作用。这也是本文中改进粒子群算法的重要内容。

传统的粒子群算法采用“先位移、再计算”的核心流程, 即粒子群在进入传统算法中时, 在不充分考虑当前粒子位置和速度的情况下, 优先对粒子

进行公式计算, 产生新的位置信息。一旦位移生效, 所积累的位移信息并不会被承接, 而是直接被简单迭代后清除, 该过程是独立的、简易的。

本文中改进的动态约束粒子群算法, 则采用“先计算, 再位移”的核心策略。在粒子群进入改进算法中时, 充分考虑到粒子的相对位置和相对速度, 检测当前粒子是否存在于收敛管理器的管理范围, 优先对粒子进行个性化规划, 大幅度减少了粒子因简单迭代错误而产生的冗余时间成本。且在以往被减弱的空间信息中采用了承接方式, 对粒子群的相对信息做出记录。时间与空间成本大幅减少。

动态约束粒子群算法的思想是: 为解空间中每一个粒子设置一个动态收敛管理器, 当多数粒子处于如下两种状态时, 将创建收敛管理器对粒子进行管理。

1) 接近其上一次移动的位置时, 将粒子群认知系数迅速缩小, 使粒子相对于自身处于凝固状态;

2) 接近于其他粒子时, 将粒子群的社会系数迅速缩小, 使粒子相对于其他粒子处于凝固状态。

当系统检测到粒子不处于这两种状态时, 释放粒子的动态收敛管理器, 使粒子恢复到之前的系数状态。具体如公式(3)所示。

$$C(k) = \begin{cases} 2 & f(i_x + 1 - i_x) > f(\Delta i_x) \\ \frac{C_1}{P(\Delta i_x)} & f(i_x + 1 - i_x) \leq f(\Delta i_x) \end{cases} \quad (3)$$

其中: $f(\Delta i_x)$ 为收敛条件, 在通过收敛规则定义处理后, 判断粒子相对位置是否满足收敛控制器; $P(\Delta i_x)$ 作为收敛管理器的收敛系数, 对于粒子位置的收敛速率和收敛时效性产生影响, 在与 $C(k)$ 系数的共同作用之下, 以任意合力大小对任意空间位置粒子通过的速度进行控制。

该管理方式可以将粒子群中的粒子有效地进行状态存储、划分, 同时进行差异化管理。

2.3 粒子群中的系数收敛管理器收敛条件

根据动态约束粒子群算法的收敛思想, 如果要通过系数收敛管理器对粒子群中的粒子管理,

必须先获取粒子间相对位置或粒子自身相对位置的值劣解与值优解, 在适应粒子的对比状态下产生系数收敛管理器。由此, 动态约束粒子群算法中的收敛规则定义为:

粒子 i_1 自身趋向于自身粒子 i_1 :
 $P_{\text{best}}(i_1) - P_{\text{best}}(i_1) \approx 0, \exists i_1 \in X(i)$, 则产生边界系数 μ 的同时, 使认知系数收敛管理器处于激发状态, 对管理器内的粒子的边界系数进行初始化。产生认知收敛系数 μ , 控制认知系数 C_1 。

粒子 i_1 自身趋向于其他社会粒子 i_2 :
 $P_{\text{best}}(i_1) - P_{\text{best}}(i_2) \approx 0, \exists i_1 \in X(i)$, 则产生边界系数 μ 的同时, 使社会系数收敛管理器处于激发状态, 对管理器内的粒子的边界系数进行初始化。产生认知收敛系数 μ , 控制认知系数 C_1 。

经过上述收敛规则的处理, 粒子群自身迭代的过程中, 在相对于自身收敛速率和相对于社会粒子的收敛速率上有了额外认知, 可以快速的对粒子相对位置差值进行反应处理。使得粒子迭代进化时拥有稳定的性能和较短的运行时间。

3 实验结果与分析

为了验证本文的面向可交互式智慧鱼群的权重动态约束的粒子群算法, 本文完成了传统算法与改进算法的综合数据对比实验、粒子行为数据对比实验、虚拟鱼群应用对比实验, 设计并实现了一款虚拟场景互动的面向少年儿童的游戏系统, 很好地应用了该算法。该系统的硬件环境为 Intel(R) Xeon(R) E5-2670 v3 2.30GHz 的 CPU、64GB 内存、NVIDIA GeForce740 的显卡, 软件开发环境为 Windows7、Unity3D、Eclipse 等。

3.1 算法综合数据对比实验

为了直观的从数据上验证传统算法与改良算法的区别, 实验采用多组对比数据, 分别使用传统粒子群算法与本文所提的改进粒子群算法。通过对粒子相对位移堆积方差、粒子自身位移标准偏差、优质进化代数、聚群效果表现、实际应用表现等方面的实验对比, 减少了原算法中时间、

数据的冗余, 改进了算法的时效性与精确性, 使之能够更佳的体现在实际应用上。同时进行算法间的比较, 拆解粒子群算法中粒子解空间最大值、最小值与范围, 体现在弱迭代次数的情况下两者之间的区别。

由表 1 中数据可见传统粒子群算法在优质进化代数上有显著劣势, 在初始数据不理想的情况下需要多代进化才能产生找到近优点的优质进化粒子。同时在粒子解空间范围上也过大, 这将导致粒子大范围反复波动效应, 由此产生的结果是

粒子位移标准偏差与相对位移堆积方差过大, 即粒子无法快速确认自身位置是否为最优解。

本文所提的改进粒子群算法, 在优质进化代数上有显著优势, 在初始数据不理想的前提下也只需要很少的进化代数就能得到近优点粒子。由于收敛管理器的控制, 粒子解空间范围缩减, 使得粒子位移标准差与相对位移堆积方差大幅度收敛。粒子获得较小的位移标准偏差和相对位移堆积方差。这意味着粒子将在较小的范围内和较短的时间内获得近优点。

表 1 传统算法与改进算法测试数据对比

算法	种群数量	最大迭代次数	优质进化代数	粒子解空间范围	粒子位移标准偏差	相对位移堆积方差
传统算法	15	100	25~45	[-200,200]	50.37	2 538.11
改进算法	15	100	6~12	[0,100]	28.31	801.24

3.2 粒子行为数据对比实验

为验证改进粒子群算法的性能, 在取得两种算法综合数据对比实验结果后, 需要验证两种粒子群算法中的微观粒子的行为是否符合改进算法描述, 同时也需要验证宏观的粒子行为是否满足能够快速凝聚、收敛的标准。粒子群中的宏观、微观粒子行为, 都将作为改进算法是否有效地重要依据。

图 2 和图 3 对比了两种算法中, 微观粒子行为在初始数据优劣两种情况之下的位移堆积图对比。由图 2 可知, 在初始数据劣势的情况下, 应用改良算法粒子的位移进度整体平稳收敛, 并在第 7 次迭代进化时已经接近最优解点, 而使用传统粒子群算法的粒子经过 15 次进化迭代, 仍然处于波动期, 未找到最优解。图 3 体现了在初始数据优势时, 改良算法粒子仍能比传统算法粒子更快找到最优解。

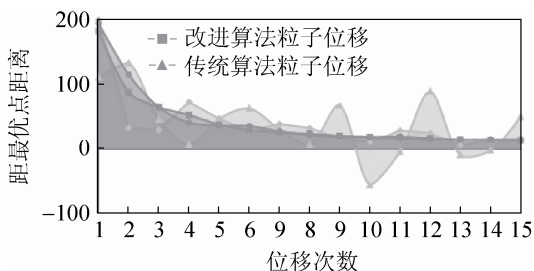


图 2 劣势初始数据两种粒子堆积对比

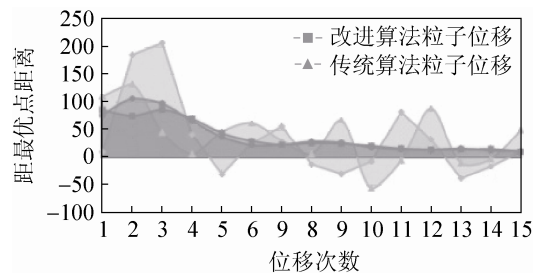


图 3 优势初始数据两种粒子堆积对比

图 4 和图 5 给出了两种算法的微观粒子在位移时的运动轨迹。由图 4 和图 5 对比可知, 使用本文提出的改良动态约束粒子群算法后, 粒子群具有了平稳朝向最中心点移动的特点, 同时相比于传统算法, 改良算法使粒子群快速收敛至群组最优解处, 不再产生明显波动。

图 6 给出了传统粒子群算法和改进粒子群算法在宏观下的粒子集群演化效果。图 6(a)、(b)、(c)所示为传统粒子群算法演化效果, 其中图 6(c)为传统算法迭代进化第 35 代。传统算法不适应应用于短程迭代的计算, 其仿真效果由于迭代次数上的缩减, 对于粒子群整体靠拢感不佳, 单一粒子无法快速找到近优点。通过对算法添加收敛管理器, 改进管理粒子间的状态之后, 粒子迭代进化过程如图 6(d)、(e)、(f)所示为改进后的粒子群算法演化效果, 其中图 6(f)为改进算法迭代进化第 16 代。粒子

群在中期迭代时已经明显收敛, 末期迭代明显聚拢, 聚拢效果优于传统粒子群算法。通过与传统算法比较可以发现, 改进粒子群算法在时效性与精准性上明显优于传统粒子群算法。

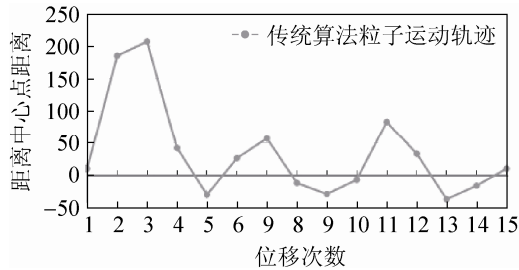


图 4 传统算法粒子群位移的运动轨迹

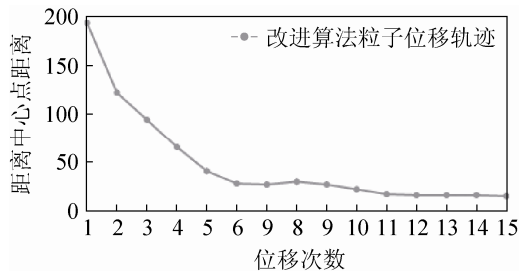


图 5 改进算法粒子群位移的运动轨迹

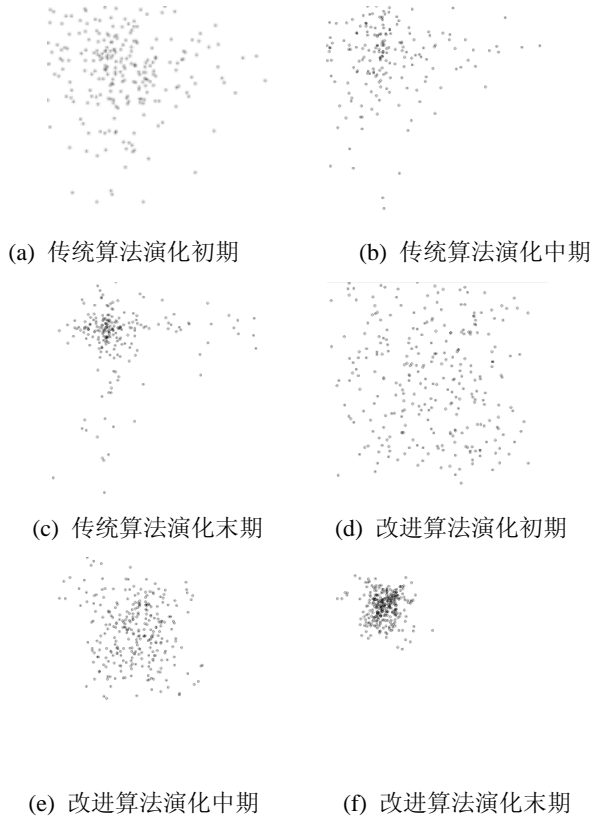


图 6 传统粒子群算法和改进粒子群算法演化效果对比

该实验演示很好地展现了改良算法在粒子群处于微观、宏观两种要求下, 对粒子行为的改良成果。从微观单一的粒子行为来看, 改良粒子群算法中的粒子能够快速、稳定的找到最优解趋于, 明显优于传统粒子群算法; 从宏观粒子群行为上来看, 改良粒子群算法中的粒子能够快速收敛、聚拢。可以较好地应用于虚拟角色互动系统。

3.3 虚拟鱼群应用对比实验

虚拟鱼群应用以虚拟水族馆系统为基础, 以大规模集群的方式演示了改良算法在智慧鱼群中的应用效果。为了区别与传统算法的不同之处, 系统产生虚拟鱼群时, 同时应用了两种算法作为内核, 用于演示改良算法在智慧鱼群快速聚群、收敛上的优势。

图 7 和图 8 为两种粒子群算法在虚拟水族馆互动应用中的表现, 其中图 7 为短程迭代(5~10 次)使用普通粒子群算法的人工智能海洋鱼, 与图 8 中使用改进粒子群算法的人工智能海洋鱼相比, 改进算法在规定时间内明显有效的完成了快速集群、聚拢的任务, 起到了很好的智能生物群仿真效果。

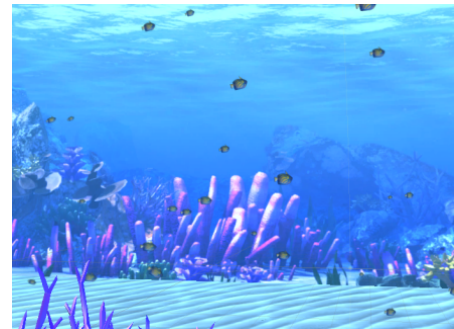


图 7 普通粒子群算法智能海洋鱼群

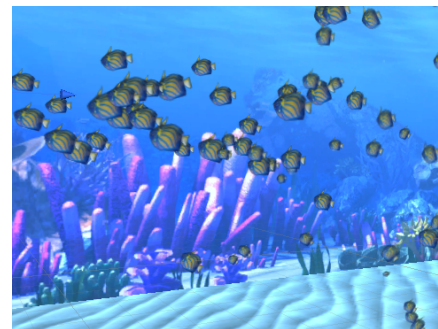


图 8 改进粒子群算法智能海洋鱼群

4 结论

本文针对粒子群算法在短程迭代的状况下搜索精度差、波动大、粒子状态考证不足,无法应用于虚拟生物集群的快速演化的情况,提出了一种面向可交互式智慧鱼群的权重动态约束的粒子群算法。该算法根据粒子状态将粒子群进行分离,对粒子群进行动态约束管理,并使用“系数收敛管理器”的概念保留了粒子间的差异化运动,确保了粒子群整体快速找到准确的近优点,改进了原搜索精度差和无状态判断的缺点,使得该算法可以更加有效地应用于智慧鱼群的虚拟集群生物的快速模拟。实验证明,该算法能够有效提升在短程迭代状况下,粒子群算法的效率和搜索精度,并能大幅度提高粒子群收敛效率和收敛准确率。该算法已经在实验室的虚拟水族馆互动系统中很好的应用。

进一步研究工作在于如何优化计算结构、如何更加精确的确定粒子群状态、如何调整收敛管理器的收敛系数等。

参考文献:

(上接第 2489 页)

参考文献:

- [1] Sohn K, Jüttler B, Kim M, et al. Computing Distances Between Surfaces Using Line Geometry [C]// Proceedings of the 10th Pacific Conference on Computer Graphics and Applications, Beijing, China. USA: IEEE, 2002: 236-245.
- [2] Chen X, Chen L, Wang Y, et al. Computing the minimum distance between two Bézier curves [J]. Journal of Computational and Applied Mathematics (S0377-0427), 2009, 229(1): 294-301.
- [3] 陈小雕, 雍俊海, 汪国昭. 平面代数曲线间最近距离的计算 [J]. 计算机辅助设计与图形学学报, 2008, 20(4): 459-463.
- [4] 徐金亭, 刘伟军, 卞宏友, 等. Bézier 曲线间最近距离的计算方法 [J]. 计算机辅助设计与图形学学报, 2009, 21(5): 595-599.
- [5] 寿华好, 黄永明, 闫欣雅, 等. 两条代数曲线间 Hausdorff 距离的计算 [J]. 浙江工业大学学报, 2013,

- [1] 李擎, 张超, 陈鹏, 等. 一种基于粒子群参数优化的改进蚁群算法 [J]. 控制与决策, 2013, 28(6): 873-878.
- [2] 单世民, 邓贵仕. 动态环境下一种改进的自适应微粒群算法 [J]. 系统工程理论与实践, 2006, 26(3): 39-44.
- [3] 莫思敏, 曾建潮, 谢丽萍. 扩展的微粒群算法 [J]. 控制理论与应用, 2012, 29(6): 811-816.
- [4] 姚成玉, 王斌, 陈东宁, 等. 混合粒子交互微粒群算法 [J]. 机械工程学报, 2015, 51(6): 198-207.
- [5] 黄太安, 生佳根, 徐红洋, 等. 一种改进的简化粒子群算法 [J]. 计算机仿真, 2013, 30(2): 327-330, 335.
- [6] Zhao Jiang, Zhou Rui. Particle swarm optimization applied to hypersonic reentry trajectories [J]. Chinese Journal of Aeronautics (S1000-9361), 2015, 28(3): 822-831.
- [7] Mingwei Li, Haigui Kang, Pengfei Zhou, et al. Hybrid optimization algorithm based on chaos, cloud and particle swarm optimization algorithm [J]. Journal of Systems Engineering and Electronics (S1004-4132), 2013, 24(2): 324-334.
- [8] R Jensi, G Wiselin Jiji. An enhanced particle swarm optimization with levy flight for global optimization [J]. Applied Soft Computing (S1568-4946), 2016, 28(3): 333-345.
- [9] 10(1): 574-577.
- [6] Krishnamurthy A, McMains S, Hanniel I. GPU-accelerated Hausdorff distance computation between dynamic deformable NURBS surfaces [J]. Computer-Aided Design (S0010-4485), 2011, 43(11): 1370-1379.
- [7] Hanniel I, Krishnamurthy A, McMains S. Computing the Hausdorff distance between NURBS surfaces using numerical iteration on the GPU [J]. Graphical Models (S1524-0703), 2012, 74(4): 255-264.
- [8] Bai Y, Yong J, Liu C, et al. Polyline approach for approximating Hausdorff distance between planar free-form curves [J]. Computer-Aided Design (S0010-4485), 2011, 43(6): 687-698.
- [9] Shou H, Lin H, Martin R, et al. Modified affine arithmetic is more accurate than centered interval arithmetic or affine arithmetic [J]. Lecture Notes in Computer Science (S0302-9743), 2003, 2768: 355-365.