

其中,  $x$  为各输入属性值。

实验中,堆栈降噪自编码结构的设计如表 5 所列。

表 5 堆栈降噪自编码超参数表

Table 5 Hyper-parameters of stacked denoising auto-encoder

超参数	值
堆栈降噪自编码预训练学习率	1
输入层节点数	11
输出层节点数	4
第一层隐含层节点数	32
第二层隐含层节点数	5
第三层隐含层节点数	4
预训练批次	1
微调批次	1
激活函数	Sigmoid
稀疏系数	0.05

不同模型迁移预测均方误差及泛化误差的对比情况如表 6 所列。通过观察表 6 可以看出,在 wine Quality 数据集上所观察的结果与在模拟数据集上建模实验的结果基本是一致的。本文方法构建的迁移预测模型可以很好地利用相似任务间的大数据进行通用特征提取,利用小样本对通用特征下的支持向量机参数进行微调,获得较为精确的预测结果。

表 6 不同模型迁移预测均方误差及泛化误差对比

Table 6 Comparison of standard deviation and generalization ability of different models

模型	均方误差	泛化误差
支持向量机(目标域小样本训练)	0.448	$5.03e-2$
支持向量机(源域大样本训练)	0.487	$5.93e-2$
PCA+支持向量机(目标域训练)	0.447	$5.02e-2$
PCA+支持向量机(源域训练)	0.561	$7.87e-2$
SDA+SVR 混合模型	0.254	$1.61e-2$

在实验中,我们也发现堆栈降噪自编码存在过学习现象,对此通过在微调时引入 L1 范数约束来限制模型的表达。为了进一步了解堆栈降噪自编码结构对混合模型在迁移预测效果方面的影响,比较了 5 种不同隐含层数下自编码混合模型的预测效果。

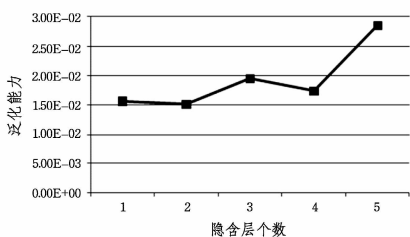


图 6 混合模型中隐含层数量与泛化能力的关系

Fig. 6 Relationship between the number of hidden layer and the generalization ability of our hybrid model

从图 6 中发现,当隐含层为 2 时,深层网络构成的预测模型的泛化能力较好。该实验表明,网络层数越多,特征抽象能力更强,能更细致地刻画源域训练样本,但可能会降低在测试样本上的迁移泛化能力。

**结束语** 为解决大数据时代大样本环境下的小样本预测问题,提出了基于堆栈降噪自编码与支持向量回归机的混合模型。利用堆栈降噪自编码器良好的特征抽取能力,发掘源领域与目标领域共有的特征知识,从而辅助支持向量回归机

在高维噪声小样本数据集上的预测。实验表明,该方法具有迁移学习能力,优于传统的 SVR 预测方法。对于本文提出的模型,仍需要大量不同领域的数据进行进一步验证,更严谨的理论证明也有待进一步研究。

## 参考文献

- [1] TAO J W, CHUNG F L, WANG S T. A kernel learning framework for domain adaptation learning[J]. Science China Information Sciences, 2012, 55(9): 1983-2007.
- [2] PAN J L, YANG Q. A survey on transfer learning [J]. IEEE Trans. on knowledge and Data Engineering, 2010, 22(10): 1345-1359.
- [3] PANS J L, KWOK J T, YANG Q. Transferring learning via dimensionality reduction[C]// Proc. of the 23st National Conference on Artificial Intelligence, Menlo Park; AAAI Press, 2008: 677-682.
- [4] WEI F M, ZHANG J P, CHU Y, et al. FSFP: Transfer learning from long texts to the short[J]. Applied Mathematics & Information Sciences, 2014, 8(4): 2033-2044.
- [5] OQUAB M M, BOTTOU L, LAPTEV I, et al. Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks [C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), USA; IEEE Computer Society, 2014: 1717-1724.
- [6] AREL I, ROSE D C, KARNOWSKI T P. Deep Machine Learning-A new Frontier in Artificial intelligence research[J]. IEEE Computational Intelligence Magazine, 2010, 5(4): 13-18.
- [7] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. Nature, 2015, 521(5): 436-444.
- [8] YU K, JIA L, CHEN Y Q, et al. Deep Learning. Yesterday, Today, and Tomorrow[J]. Journal of Computer Research and Development, 2013, 50(9): 1799-1804. (in Chinese)  
余凯, 贾磊, 陈雨强, 等. 深度学习的昨天、今天和明天[J]. 计算机研究与发展, 2013, 50(9): 1799-1804.
- [9] BENGIO Y. Learning deep architectures for AI[J]. Foundations and Trends in Machine Learning, 2009, 2(1): 1-127.
- [10] HINTON G, SALAKHUTDINOV R R. Reducing the dimensionality of data with neural networks[J]. Science, 2006, 313(5786): 504-507.
- [11] ZHANG L, LIU Z, ZHANG J Q, et al. Reduction method of weapon system-of-systems assessment index system based on autoencoder[J]. Journal of Central South University (Science and Technology), 2013, 44(10): 4130-4137. (in Chinese)  
张乐, 刘忠, 张建强, 等. 基于自编码神经网络的装备体系评估指标约简方法[J]. 中南大学学报(自然科学版), 2013, 44(10): 4130-4137.
- [12] HU Z, FU K, ZHANG C S. Audio Classical Composer Identification by Deep Neural Network[J]. Journal of Computer Research and Development, 2014, 51(9): 1945-1954. (in Chinese)  
胡振, 傅昆, 张长水. 基于深度学习的作曲家分类问题[J]. 计算机研究与发展, 2014, 51(9): 1945-1954.

# 改进的鸡群优化算法及其在 DTI-FA 图像配准中的应用

郑伟 蒋晨娇 刘帅奇 赵杰

(河北大学电子信息工程学院 河北保定 071002) (河北省数字医疗工程重点实验室 河北保定 071002)

**摘要** 鸡群优化算法(Chicken Swarm Optimization,CSO)是一个全新的群智能优化算法,简单且具有良好的扩展性。针对鸡群优化算法中因为母鸡的寻优能力差而使算法容易陷入局部极值的问题,提出了一种结合混沌思想的改进鸡群优化算法(Chaotic Improved Chicken Swarm Optimization Algorithm,CICSO)。该算法结合混沌思想的遍历性初始化鸡群位置,将母鸡的位置更新公式改为仅向全局适应度值最好的公鸡学习,并引入学习系数来避免陷入局部最优。最后将改进的鸡群优化算法(CICSO)应用于 DTI-FA 图像配准。仿真实验结果表明,在解决高维问题时,改进的鸡群优化算法避免了陷入局部极值,提高了收敛精度,在 DTI-FA 图像配准中提高了图像的配准精确度。

**关键词** 鸡群优化算法,搜索范围,改进的鸡群优化算法,测试函数,图像配准

中图分类号 TP391.41 文献标识码 A DOI 10.11896/j.issn.1002-137X.2018.01.050

## Improved Chicken Swarm Optimization Algorithm and Its Application in DTI-FA Image Registration

ZHENG Wei JIANG Chen-jiao LIU Shuai-qi ZHAO Jie

(College of Electronic and Information Engineering, Hebei University, Baoding, Hebei 071002, China)

(Key Laboratory of Digital Medical Engineering of Hebei Province, Baoding, Hebei 071002, China)

**Abstract** Chicken swarm optimization algorithm(CSO) is a new swarm intelligence optimization algorithm, which has the characteristics of simple and good scalability. In view of the problem that the chicken swarm optimization algorithm is easy to fall into the local extremum for a hen has poor optimization ability, a chaotic improved chicken swarm optimization algorithm (CICSO) was proposed. In this algorithm, the position of chicken is initialized by the ergodicity of the chaotic idea, and the position updating formula of the hen is changed to the rooster with the best global fitness value. In addition, the learning factor is introduced to avoid the local optimum. At last, the improved algorithm (CICSO) was applied to DTI-FA image registration. Simulation results show that the improved algorithm can avoid falling into local extremum, improves the convergence precision and the registration accuracy of the image in the DTI-FA image registration.

**Keywords** Chicken swarm optimization, Search scope, Chaotic improved chicken swarm optimization, Test functions, Image registration

## 1 引言

许多工程技术领域中的问题最终都被转换为求解优化模型的全局最优值,传统的方法在解决这些问题时往往存在着计算复杂度高、计算结果精度低等问题。因此,许多学者会模拟生物的一些习性来创造仿生元启发式算法,以解决全局最优问题,如 1995 年由 Kennedy 等提出的粒子群优化算法(Particle Swarm Optimization, PSO)<sup>[1]</sup>、1996 年由 Dorigo 等人提出的蚁群优化算法(Ant Colony Optimization, ACO)<sup>[2]</sup>、2002 年由李晓磊等人提出的人工鱼群算法(Artificial Fish

Swarm Algorithm, AFSA)<sup>[3]</sup>、2003 年由李威武提出的细菌趋药性算法(Bacterial Chemotaxis Algorithm, BCC)<sup>[4]</sup>、2010 年由 Yang 提出的蝙蝠算法(Bat Algorithm, BA)<sup>[5]</sup>。这些算法有效地解决了全局最优问题,但是在精度和鲁棒性方面还有待提高。

鸡群优化算法<sup>[6]</sup>是由 Meng 等人于 2014 年 10 月在第五次国际群体智能会议(ICSI)上提出的一种新的仿生算法。该算法是通过模仿鸡群(公鸡、母鸡、小鸡)的等级制度以及它们的行为等来解决优化问题。该算法在求解优化算法时具有精度高、算法鲁棒性好等特点。但是这些算法,包括鸡群优化算

到稿日期:2016-12-08 返修日期:2017-02-24 本文受国家自然科学基金(61572063, 61401308),河北省自然科学基金(F2016201187, F2016201142),河北省高等学校科学技术研究项目(QN2016085, ZC2016040),河北大学引进人才科研启动经费(2014-303),中西部综合实力提升项目资助。

郑伟(1972—),女,博士,教授,主要研究方向为图像处理与分析、图像安全通信、图像加密和隐藏;蒋晨娇(1992—),女,硕士生,主要研究方向为图像处理和计算机视觉;刘帅奇(1986—),男,博士,副教授,主要研究方向为计算视觉与多维信号处理;赵杰(1969—),男,博士,教授,主要研究方向为智能数据处理、图像处理与分析、信号检测与模式识别, E-mail: zhaojie\_hbu@126.com(通信作者)。

法,在解决优化问题时不能同时兼顾全局搜索和局部最优的问题,尤其是在复杂的优化问题及高维空间搜索时效果不是很理想,会出现收敛精度变低、容易陷入局部极值、早熟收敛的情况。因此,有必要探索新的改进方法和策略来改善这些问题。文中提出了一种结合混沌思想的改进鸡群优化算法,该算法结合混沌思想对鸡群进行初始化,并对母鸡的位置更新公式进行了改进,母鸡仅向全局适应度值最好的公鸡学习,并且引入学习系数,从而提高解决高维问题时的收敛精度,避免早熟收敛情况。仿真实验验证了所提方法的有效性和可行性。

目前,老年病发病率较高且不易治愈。比如,阿尔茨海默病(Alzheimer Disease, AD),又叫老年性痴呆,是一种中枢神经系统变性病,是老年期痴呆最常见的一种类型。阿尔茨海默病成为世界各地家庭和经济的一个不断扩大的危机。由于AD确诊困难,因此需加强早期诊断技术,早期进行治疗。目前的检查技术有CT, MRI, SPECT, PET成像技术,这些多为发病之后的诊断,无法进行有效预防。而弥散张量成像(Diffusion Tensor Imaging, DTI)<sup>[13]</sup>是一种新的描述大脑结构的方法,是核磁共振成像(Magnetic Resonance Imaging, MRI)的特殊形式。它由一组不同方向信息的扩散加权成像(DWD)构成,包含了脑白质组织的标量信息,还包括脑白质纤维束的矢量信息。Basser等人于1994年<sup>[13]</sup>首次系统地描述了DTI成像的原理,于2000年又提出了一种可靠地使用DTI数据进行纤维束追踪的方法<sup>[14]</sup>。因此,在脑成像研究中,DTI不仅能提供脑白质形态结构的解剖信息,还能提供脑白质纤维束的走行方向等信息。神经原纤维缠结是AD的最典型改变之一。而在目前的医学影像技术中,DTI成像技术是当前唯一能有效观察和追踪脑白质纤维束的非侵入性检查方法,水分子在大脑内的各向异性运动是用一个二阶张量来表征的。根据DTI获取的可直接反映水分子各向异性度的部分各向异性数(Fractional Anisotropy, FA)图像有着较好的灰白质对比,有便于观察大脑白质中神经纤维的走向,而FA也是AD早期诊断的敏感指标,因此需要DTI-FA图像(即弥散张量成像FA图像)精确配准,该配准对精度及鲁棒性要求较高,因此用鸡群及改进的鸡群来满足这些要求。仿真实验验证了本文算法在DTI-FA配准中的有效性和可行性。

## 2 基本鸡群优化算法

由文献<sup>[6]</sup>可知,基本鸡群优化算法是模拟鸡群的等级制度来寻找食物位置的过程,将自然界中鸡群觅食的生物特性抽象为数学模型,从而设计出基本鸡群优化算法。简单起见,对鸡群作以下理想化处理:

(1)在鸡群中存在若干个组,每一组都由一只公鸡(领头)和几只母鸡与小鸡组成。

(2)通过鸡的适应度值来确定身份。适应度值最好的作为公鸡,适应度值最差的几只作为小鸡,其他的作为母鸡。母鸡随机选择属于哪一组,母子关系也随机建立。

(3)等级制度、支配关系以及母子关系在一组中保持不变,但是由于小鸡会长大,因此在每 $G$ ( $G$ 为更新间隔步数,是一个常量,其值在后文确定)步之后进行状态更新。

鸡群一共有3类个体:公鸡、母鸡和小鸡。假设 $RN$ ,  $HN$ ,  $CN$ ,  $MN$ 分别代表公鸡、母鸡、小鸡、母鸡妈妈的数目, $N$ 为鸡群个体总数, $t$ 代表迭代次数, $D$ 代表维度。每个个体的位置 $x_{i,j}^t$ ( $i \in [1, 2, \dots, N]$ ,  $j \in [1, 2, \dots, D]$ )表示第 $i$ 个个体 $j$ 维在第 $t$ 次迭代的值。其中公鸡的适应度值 $f$ 最好,因此它可以在更广泛的范围内寻找食物。它的位置更新公式如下:

$$x_{i,j}^{t+1} = x_{i,j}^t \times (1 + \text{Randn}(0, \sigma^2)) \quad (1)$$

$$\sigma^2 = \begin{cases} 1, & \text{if } f_i \leq f_k \\ \exp\left(\frac{f_k - f_i}{|f_i| + \epsilon}\right), & \text{otherwise} \end{cases} \quad (2)$$

其中, $\text{Randn}(0, \sigma^2)$ 代表均值为0、标准差为 $\sigma^2$ 的高斯分布; $\epsilon$ 是一个很小的常数,用来保证分母有意义; $k \in [1, N]$ ( $k \neq i$ ),代表异于 $i$ 的任一公鸡; $f$ 是对应 $x$ 的适应度值, $f_i$ 代表第 $i$ 只公鸡的适应度值(即当前公鸡的个体适应值), $f_k$ 代表第 $k$ 只公鸡的适应度值(即鸡群中除第 $i$ 只公鸡外任一公鸡个体的适应度值)。

母鸡可以从其他鸡那里抢夺食物,也可以在自己的领土内寻找食物。它的位置更新公式如下:

$$x_{i,j}^{t+1} = x_{i,j}^t + S1 \times \text{Rand} \times (x_{r1,j}^t - x_{i,j}^t) + S2 \times \text{Rand} \times (x_{r2,j}^t - x_{i,j}^t) \quad (3)$$

$$S1 = \exp\left(\frac{f_i - f_{r1}}{\text{abs}(f_i) + \epsilon}\right) \quad (4)$$

$$S2 = \exp((f_{r2} - f_i)) \quad (5)$$

其中, $\text{Rand}$ 是 $[0, 1]$ 之间的随机数; $r1 \in [1, 2, \dots, N]$ 代表第 $i$ 只母鸡所在组的公鸡; $r2 \in [1, 2, \dots, N]$ 代表整个鸡群中的任一公鸡或母鸡;并且 $r1 \neq r2$ 。当 $S1 = 0$ 时,母鸡从其他鸡那里抢夺食物,因为存在竞争,所以当两只鸡的适应度值差距越小,两只鸡的位置差越大时,母鸡越不易取得食物。当 $S2 = 0$ 时,母鸡在自己的领土寻找食物,若 $S1$ 的值越接近1,则第 $i$ 只母鸡与它所在组的公鸡的位置越接近。

小鸡是跟随母鸡寻找食物的,它的位置更新公式如下:

$$x_{i,j}^{t+1} = x_{i,j}^t + FL \times (x_{m,j}^t - x_{i,j}^t) \quad (6)$$

其中, $x_{m,j}^t$ 表示第 $m$ 只母鸡妈妈的 $j$ 维在第 $t$ 次迭代的值; $FL$ ( $FL \in (0, 2)$ )表示小鸡跟随母亲寻找食物的参数。

## 3 基本鸡群算法的参数分析

基本鸡群算法中有7个参数: $N$ ,  $HN$ ,  $RN$ ,  $MN$ ,  $CN$ ,  $G$ 以及 $FL$ ,它们分别代表鸡群中个体数目、母鸡个体数目、公鸡个体数目、鸡妈妈个体数目、小鸡个体数目、更新间隔步数和小鸡在母鸡周围寻找食物的参数。因为母鸡可以下蛋,对人类更有用,所以 $HN > RN$ ;又因为并不是所有的母鸡都会同时孵化鸡蛋,所以 $HN > MN$ ;  $G$ 应该是一个适当的值,如果过大会影响收敛速度,如果过小可能会陷入局部极值,因此通过实验具体确定 $FL$ 与 $G$ 的值。实验中均采用CSO算法,测试函数如表1所列; $F1, F2, F3$ 测试函数实验中用2维; $F4, F5, F6, F7$ 测试函数实验中取100维;其中 $F3, F4, F5$ 是单模态函数,在定义域内只有一个极值点,主要用来测试算法的寻优精度和收敛速度; $F1, F2, F6, F7$ 是非线性多模态函数,存在多个局部极值点,用来测试算法的全局寻优能力和避免

早熟的能力。在实验过程中鸡群数目  $N$  取值为 100, 迭代次数取值为 1000, 低维函数选择  $F1$ , 高维函数选择  $F4$ , 且  $RN=0.15 \times N$ ;  $HN=0.7 \times N$ ;  $CN=N-RN-HN$ ;  $MN=0.5 \times HN$ 。根据文献[6],  $G$  的范围为  $G \in [2, 20]$ ,  $FL$  的范围为  $FL \in [0.2, 1.4]$ , 则在  $G$  分别取值为 5, 10, 15, 20, 25 时  $FL$  在 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4 对应的  $F1$  各函数值的关系如图 1 所示,  $F4$  各函数值的关系如图 2 所示。

表 1 7 个测试函数

Table 1 Seven test functions

测试函数	代号	取值范围	最优值
$f_1(x, y) = (x^2 + y^2)^{0.25} \cdot (\sin^2(50(x^2 + y^2)^{0.1}) + 1.0)$	F1	$[-20, 20]$	0
$f_2(x, y) = 20 + (x^2 - 10\cos(2\pi x) + y^2 - 10\cos(2\pi y))$	F2	$[-600, 600]$	0
$f_3(x, y) = \frac{(\sin \sqrt{x^2 + y^2})^2 - 0.5}{(1 + 0.001(x^2 + y^2))^2} - 0.5$	F3	$[-100, 100]$	-1
$f_4(x) = \sum_{i=1}^n x_i^2$	F4	$[-50, 50]$	0
$f_5(x) = \sum_{i=1}^n (\lfloor x_i + 0.5 \rfloor)^2$	F5	$[-100, 100]$	0
$f_6(x) = \frac{1}{4000} \sum_{i=1}^n (x_i)^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	F6	$[-600, 600]$	0
$f_7(x) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10]$	F7	$[-5.12, 5.12]$	0

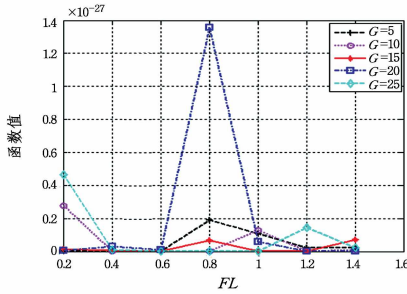
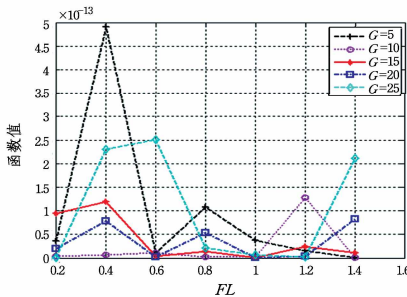
图 1  $F1$  中  $FL$  与函数值的关系图Fig. 1 Relations between  $FL$  and function values in  $F1$ 图 2  $F4$  中  $FL$  与函数值的关系图Fig. 2 Relations between  $FL$  and function values in  $F4$ 

图 1、图 2 中横轴代表  $FL$  的取值, 纵轴代表不同  $G$  值下  $FL$  值对应的函数值。从图 1 中可以看出, 当  $FL$  取值为 0.2 时,  $G$  取值为 5, 15, 20 时都能达到函数理论最优值; 当  $FL$  取值为 0.4 时,  $G$  在 5, 10, 15, 25 时都能达到最优; 当  $FL$  取值为 0.6 时,  $G$  在 5, 10, 15, 20, 25 时均能达到最优; 当  $FL$  取值为 0.8 时,  $G$  在 10, 25 时能达到最优; 当  $FL$  取值为 1.0 时,  $G$  在 15, 25 时达到最优; 当  $FL$  取值为 1.2 时,  $G$  在 10, 15, 20 时都

能达到最优; 当  $FL$  取值为 1.4 时,  $G$  在 10, 20 时能达到最优。从图 2 可以看出, 当  $FL$  取值为 0.2 时,  $G$  在 10, 25 时都能达到最优; 当  $FL$  取值为 0.4 时,  $G$  在 10 时能达到最优; 当  $FL$  取值为 0.6 时,  $G$  在 5, 10, 15, 20 时都能达到最优; 当  $FL$  取值为 0.8 时,  $G$  在 10 时都能达到最优; 当  $FL$  取值为 1.0 时,  $G$  在 10, 15, 20 时都能达到最优; 当  $FL$  取值为 1.2 时,  $G$  在 10, 20 时都能达到最优; 当  $FL$  取值为 1.4 时,  $G$  在 10 时能达到最优。综上所述, 当  $G$  取值为 10 时,  $FL \in [0.4, 1.0)$  时均能够取得最优值, 该文实验中取  $G=10, FL=0.8$ 。

## 4 改进的鸡群优化算法

文献[6]已经验证了低维时 CSO 算法在收敛速度以及收敛精度方面是优于 PSO 算法的, 基本上能够实现理论最优值; 但对高维情况没有说明。因此, 文中采用 CSO 算法对测试函数  $F4, F5, F6, F7$  在高维条件下进行了求解, 并与 PSO 算法进行了比较, 实验结果如表 2 所列。

表 2 2 种方法在高维测试函数下的比较

Table 2 Comparison of two methods under high dimensional

		test function			
测试函数	算法	最优值	达到最优迭代次数	运行时间/s	最差值
F4	PSO	2.65911	1000	0.909284	5.51724
	CSO	$3.84959 \times 10^{-8}$	1000	6.960864	7.52361
F5	PSO	2	1000	0.887009	10
	CSO	0	133	40.141713	182
F6	PSO	0.0527184	1000	4.293715	0.0847786
	CSO	$8.84355 \times 10^{-7}$	1000	48.873868	0.604961
F7	PSO	488.615	1000	1.482941	594.627
	CSO	$2.92179 \times 10^{-10}$	1000	6.290342	961.74

测试函数  $F4$  和  $F5$  用于测试算法的寻优精度和收敛速度, 测试函数  $F6$  和  $F7$  用于测试算法的全局寻优能力和避免早熟的能力。从表 2 中可以看出, 在  $F4$  函数上 CSO 最好值的收敛精度比 PSO 提高了 9 个数量级, 但是 CSO 算法的最差值比 PSO 算法的结果还差, 已经偏离了全局最优值; 在  $F5$  函数上 CSO 的最好值已经达到了理论最优值, 但最差值却比 PSO 还差, 依然偏离全局最优值; 在  $F6$  和  $F7$  函数上同理, CSO 的最好值收敛精度比 PSO 高出了 10 多个数量级, 但是最差值依旧偏离全局最优值。综上所述, 在解决高维问题时 CSO 算法偏离了全局最优值, 已经陷入局部最优, 出现早熟收敛的情况。因此, 该文对此算法进行了改进。首先, 原位置更新公式中母鸡可以从其他鸡那里抢夺食物, 也可以在自己的领土寻找食物, 但是母鸡自己寻找食物的能力差, 还会受到伙伴公鸡的影响; 与鸡群中其他个体抢夺食物又会受到其他公鸡和母鸡的影响, 并且其他鸡也未必会比自身的能力强, 因此会出现局部最优的情况。因此, 该文对母鸡的位置更新公式进行了改进, 母鸡仅向全局适应度值最好的公鸡学习, 并且学习系数为  $S$ 。改进后母鸡的位置更新公式如下:

$$x_{i,j}^{t+1} = x_{i,j}^t + S \times \text{Rand} \times (\text{best}X - x_{i,j}^t) \quad (7)$$

其中  $S = \exp(f_{\text{best}X} - f_i)$  是母鸡向公鸡学习的学习系数;  $f_{\text{best}X}$  代表全局最优值;  $\text{best}X$  是全局最优值对应的位置。

其次, 因为混沌具有随机性、遍历性、敏感性、规律性, 所

以基于混沌的探索方法比无目的、无顺序地随机探索更加出色。Logistic 方程式(8)是一个具有代表性的混沌系统:

$$S_{k+1} = \mu \times S_k \times (1 - S_k) \quad (k=0, 1, 2, \dots) \quad (8)$$

其中,  $\mu$  为控制参数, 一般取值为 4; 假设  $0 \leq S_0 \leq 1$ , 则整个混沌系统处于混沌状态, 若  $S_0$  在  $[0, 1]$  随机取值, 能够迭代出一个时间序列  $S_1, S_2, S_3, \dots$ 。该文利用混沌的思想, 对个体最初的位置实行混沌序列设置, 利用混沌改变了个体搜索的种群遍历性, 在出现许多最初群体的基础上择优选出初始群体。

基于混沌思想的改进鸡群优化算法 (Chaotic Improved Chicken Swarm Optimization Algorithm, CICS0) 的流程如下:

Step 1 初始化鸡群数量  $N$  及相关参数。

Step 2 计算  $N$  个鸡的适应度值, 对最初当前最好位置  $Pbest$  和全局最好位置  $Xbest$  (即前面公式中提到的  $bestX$ ) 进行混沌设置, 并设置迭代次数初始值  $t=1$ 。

Step 3 若  $t \% G=0$  即  $t$  对  $G$  求余为 0, 则对适应度值进行排序并且分组, 每组中有一只公鸡、若干母鸡和小鸡, 通过鸡的适应度值来确定身份, 然后将母鸡随机选择在哪一组 (在前面的基本鸡群优化算法中已介绍); 否则, 直接进入 Step 4。

Step 4 根据 3 种位置更新公式更新位置, 并计算适应度值。

Step 5 更新个体最优位置和全局最优位置,  $t=t+1$ , 当满足迭代停止条件时, 结束迭代并输出最优值; 否则继续迭代, 即返回 Step 3。

图 3 是基于混沌思想改进的鸡群优化算法 (CICS0) 的流程图。

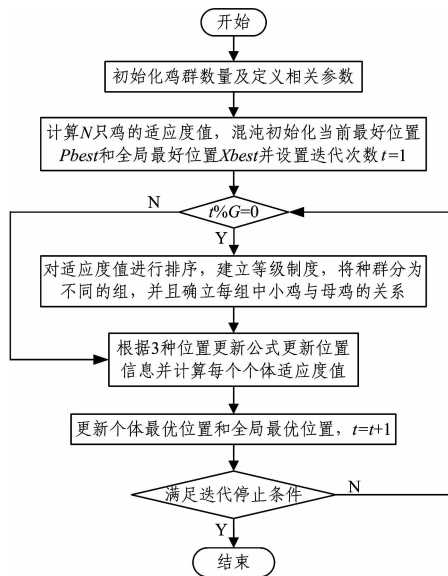


图 3 改进的鸡群优化算法的流程图

Fig. 3 Flow chart of the improved chicken swarm optimization algorithm

## 5 算法有效性的仿真实验与比较

通过与 BCC, PSO, CSO 等算法进行比较, 来验证所提算法的有效性。为了结果的公正性, 该文对 4 种算法均运行 30 次并取最好结果; 种群数目均为 100; 迭代次数为 1000; 低维选取 2 维, 高维选取 100 维; 其他相关参数如表 3 所列。

表 3 相关参数设置

Table 3 Setting of related parameter

算法	参数
BCC	$Nbl=2; ebegin=2.0; e=0.001; a=1.25$
PSO	$c1=1.4962; c2=1.4962; w=0.7298;$
CSO	$RN=0.15 * N; HN=0.7 * N; MN=0.5 * HN; CN=N-RN-HN$
CICS0	$RN=0.15 * N; HN=0.7 * N; MN=0.5 * HN; CN=N-RN-HN$

在 7 个测试函数中, 前 3 个函数实验时采用低维, 后 4 个函数实验时采用高维。该文算法与其他算法的比较结果如表 4、表 5 所列。

表 4 4 种方法在低维测试函数下的比较

Table 4 Comparison of four methods in low dimensional test function

测试函数	算法	最优值	达到最优迭代次数	迭代 1000 次运行时间/s	最差值
F1	BCC	$1.7641 * 10^{-6}$	33	3.309572	$1.1078 * 10^{-4}$
	PSO	$6.76845 * 10^{-25}$	1000	1.513335	$1.03354 * 10^{-23}$
	CSO	0	874	7.149610	0
	CICS0	$4.69076 * 10^{-72}$	1000	6.817151	$1.55323 * 10^{-62}$
F2	BCC	$5.5458 * 10^{-12}$	42	6.432145	0.9950
	PSO	0	169	1.482941	0
	CSO	0	47	6.690342	0
	CICS0	0	41	6.364251	0
F3	BCC	-1	129	6.087681	-0.9903
	PSO	-1	83	1.448382	-0.990284
	CSO	-1	32	7.082201	-1
	CICS0	-1	27	6.834761	-1

表 5 3 种方法在高维测试函数下的比较

Table 5 Comparison of three methods in high dimensional test function

测试函数	算法	最优值	达到最优迭代次数	迭代 1000 次运行时间/s	最差值
F4	PSO	2.65911	1000	0.909284	5.51724
	CSO	$3.84959 * 10^{-8}$	1000	6.960864	7.52361
	CICS0	$1.75936 * 10^{-34}$	1000	5.387219	$1.71616 * 10^{-31}$
F5	PSO	2	1000	0.877009	10
	CSO	0	133	6.014124	182
	CICS0	0	135	5.441799	0
F6	PSO	0.0527184	1000	4.293715	0.0847786
	CSO	$8.84355 * 10^{-7}$	1000	9.873868	0.604961
	CICS0	0	510	9.399312	0
F7	PSO	488.615	1000	1.482941	594.627
	CSO	$2.92179 * 10^{-10}$	1000	6.590342	961.74
	CICS0	0	632	6.234251	0

F3 主要用来测试算法的寻优精度和收敛速度, F1 和 F2 用来测试算法的全局寻优性能和避免早熟的能力。从表 4 中可以看出, 在低维时, F1 函数上 CSO 达到了理论的最优值 0, 而 CICS0 在相同迭代次数下的收敛精度也比 PSO 提高了 47 个数量级, 比 BCC 算法提高了 56 个数量级; F2 函数上 PSO, CSO, CICS0 都达到了理论最优值 0, 从达到最优的迭代次数可看出 CICS0 的收敛速度更快, 只有 BCC 算法的收敛精度相差较多; F3 函数上 PSO, CSO, CICS0, BCC 均达到了理论最优值 -1, 从达到最优时的迭代次数也可看出 CICS0 的收敛速度更快。通过以上比较可以得出, 在低维时 PSO, CSO, CICS0 基本都可达到理论最优值, 在精确性和全局寻优能力等方面均优于 BCC 算法。

F4 和 F5 主要用来测试算法的寻优精度和收敛速度, F6 和 F7 用来测试算法的全局寻优性能和避免早熟的能力。从

表 5 可以看出,在高维时,  $F_4$  函数上 CICS0 的收敛精度比 CS0 提高了 25 个数量级,比 PSO 提高了 33 个数量级;在  $F_5$  函数上 CICS0 与 CS0 均达到了理论最优值 0,但是 CS0 在 30 次迭代中的最差值为 128,而 CICS0 的最差值依然能达到理论最优值 0;在  $F_6$  函数上只有 CICS0 达到了理论最优值,CS0 的收敛精度比 PSO 高 5 个数量级;在  $F_7$  函数上只有 CICS0 达到了理论最优值,CS0 的收敛精度比 PSO 提高了 12 个数量级。通过以上比较可以得出,在高维时粒子群优化算法(PSO)的效果最差,而结合混沌思想改进的鸡群优化算法(CICS0)也明显优于原始鸡群优化算法(CS0),最差值也基本上能实现最优化,因此该文提出的算法在收敛精度和收敛速度以及避免早熟方面有了明显的提高。

## 6 该算法在 DTI-FA 图像配准中的应用

DTI-FA 图像具有旋转不变性,可作为配准插值等处理的参考标量图像;具有较好的对比灰白质及较高的信噪比,反映的是标准化后组织的物理特性,具有可对比性。FA 值的改变反映了认知障碍的严重程度,FA 值也是 AD 早期诊断的敏感指标,无论是不同年龄组 AD 患者的 FA 值对比还是正常组和 AD 患者组的 FA 值对比,都需要 DTI-FA 图像的精确配准。

DTI-FA 图像配准对精度及鲁棒性要求较高,而鸡群及改进鸡群优化算法的精度及鲁棒性较好,因此应用本文提出的算法进行 DTI-FA 图像配准并与前面所提 PSO、CS0 算法及应用 Powell 算法(脑影像处理软件 SPM 中使用的配准优化算法)进行的 DTI-FA 图像配准结果进行比较,来验证本文算法的有效性。为了结果的公正性,该文对 4 种算法均运行 30 次并取最好结果;并且 PSO、CS0、CICS0 算法的种群数目均为 100;迭代次数为 1000 次。

图像配准是为了使两幅或多幅图像中的对应点在空间和解剖位置上对齐,在搜索空间中寻找使相似性测度达到最优的空间变换参数,配准原理框图如图 4 所示。该文空间变换采用仿射变换,并使用归一化互信息作为相似性测度评价(见式(9)),其中  $H(A)$  为图像 A 的熵,  $H(B)$  为图像 B 的熵,  $H(A, B)$  为图像 A 和 B 的联合熵。然后采用优化算法找到最优变换空间。其中部分各向异性 FA 的计算公式如式(10)所示,  $\lambda_1, \lambda_2, \lambda_3$  是弥散张量  $D$  从大到小的 3 个特征值(其中  $\lambda_1$  是该弥散张量对应像素的主特征值),  $\bar{\lambda}$  是弥散张量特征值的平均值,即平均弥散度。

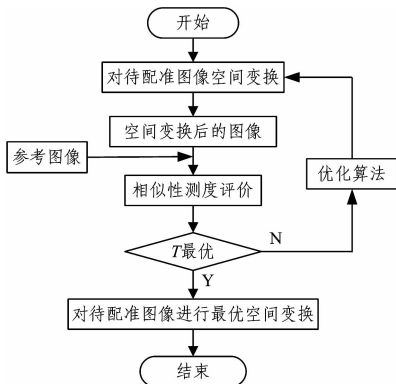


图 4 图像配准原理框图

Fig. 4 Principle block diagram of image registration

$$NMI(A, B) = \frac{H(A) + H(B)}{H(A, B)} \quad (9)$$

$$FA = \sqrt{\frac{3}{2}} \sqrt{\frac{(\lambda_1 - \bar{\lambda})^2 + (\lambda_2 - \bar{\lambda})^2 + (\lambda_3 - \bar{\lambda})^2}{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \quad (10)$$

分别用 PSO, Powell, CS0, CICS0 作为优化算法对 FA 图像进行配准,配准结果如图 5 所示,取其中两次结果,每行代表一次实验。其中第一组实验中图 5(a)为 FA 图的原始图像,即参考图像,图 5(b)为待配准图像,图 5(c)—图 5(f)分别为用 PSO, Powell, CS0, CICS0 配准后的图像;第二组实验中图 5(g)为 FA 图的原始图像,即参考图像,图 5(h)为待配准图像,图 5(i)—图 5(l)分别为用 PSO, Powell, CS0, CICS0 配准后的图像。图 5(b)、图 5(h)是由图 5(a)、图 5(g)向右平移 8、向下平移 8 并顺时针旋转  $20^\circ$  得到的。由图 5 可以看出, PSO 算法得到的图像与待配准图像位置基本无异,而其他 3 种算法都有向原始图像靠近的迹象,但 CICS0 的效果更加明显,达到了配准效果。

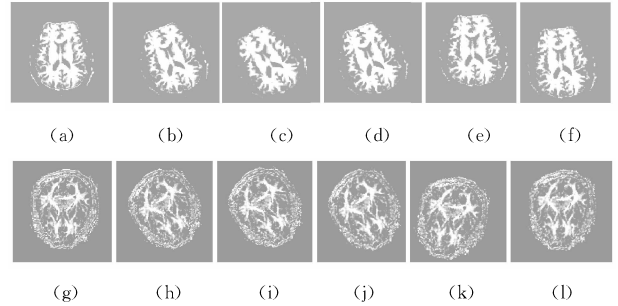


图 5 图像配准效果图

Fig. 5 Effect of image registration

对各种算法的配准结果进行比较,采用欧氏距离 ( $AT-ED$ )、互信息 ( $MI$ )、差方和 ( $SSD$ ) 来评价不同优化算法的配准结果。 $AT-ED$  表明图像之间的真实距离,其值越接近 0 则相似度越大; $MI$  值越大,说明配准结果的信息越丰富,配准效果越好; $SSD$  表示配准图像与参考图像之间的差值,值越小表示配准精度越高。两次实验的结果如表 6 所列。

表 6 FA 图像配准后参数的变化

Table 6 Parameters change after registration of FA images

实验组	配准方法	$AT-ED$	$MI / (\text{bit} \cdot \text{sign}^{-1})$	$SSD$	$T/s$
第一组	用 PSO 配准后的图像	0.0080	0.7620	11.2474	68.3590
	用 Powell 配准后的图像	0.0255	0.7572	11.6750	209.7717
	用 CS0 配准后的图像	0.0086	0.6527	12.8952	170.7018
	用 CICS0 配准后的图像	<b>0.0076</b>	<b>0.7964</b>	<b>10.2234</b>	165.3294
第二组	用 PSO 配准后的图像	0.0300	1.3457	16.4160	67.7760
	用 Powell 配准后的图像	0.0298	1.2093	16.1768	207.7717
	用 CS0 配准后的图像	0.0296	1.3239	16.0447	163.5834
	用 CICS0 配准后的图像	<b>0.0285</b>	<b>1.4119</b>	<b>14.6701</b>	159.9111

由表 6 可以看出,第一组实验中 PSO 的配准时间是最短的,但是其  $AT-ED$ 、 $SSD$  都高于 CICS0 算法,  $MI$  低于 CICS0 算法;对于 Powell 的配准结果,它在速度上是最慢的,  $AT-ED$  都高于其他 3 种方法,  $MI$  低于 PSO 和 CICS0 算法,  $SSD$  高于 PSO 和 CICS0 算法;对于 CS0 的配准结果,在时间上长于 PSO 算法,并且  $AT-ED$ 、 $MI$ 、 $SSD$  均最差;对于 CICS0 的配准结果,在速度上与 CS0 相差不大,虽然比 PSO 算法慢,但比

其他几种算法都快,并且 CICS0 的  $MI$  值最大,  $ATED$  与  $SSD$  最小。在第二组实验中, PSO 的配准时间是最短的,但是其  $ATED$  和  $SSD$  都高于其他 3 种算法,  $MI$  低于 CICS0 算法;对于 Powell 的配准结果,在速度上是最慢的,  $ATED$  高于 CSO 和 CICS0 算法,  $MI$  是最低的,  $SSD$  高于 CSO 和 CICS0 算法;对于 CSO 的配准结果,在时间上长于 PSO 算法,并且  $ATED$  和  $SSD$  高于 CICS0 算法,  $MI$  低于 PSO 和 CICS0 算法;对于 CICS0 的配准结果,在时间上与 CSO 相差不大,虽然比 PSO 算法慢,但比其他几种算法都快,并且 CICS0 的  $MI$  值最大,  $ATED$  与  $SSD$  最小。综上所述,文中方法实现了较好的图像配准。

**结束语** 文中针对鸡群优化算法(CSO)容易陷入局部极值,尤其在解决高维问题时收敛精度明显下降等缺点,提出了一种结合混沌思想的改进鸡群优化算法(CICS0)。该算法对鸡群初始位置进行混沌设置,并对母鸡的位置更新公式进行了改进,引入学习系数。算法针对 7 个典型的基准测试函数进行了性能测试,并与粒子群优化算法(PSO)、细菌趋药性算法(BCC)、鸡群优化算法(CSO)进行了比较,结果表明文中提出的算法在全局寻优能力、收敛速度和收敛精度方面比其他算法都有大幅度的提高;尤其在解决高维问题时效果更明显;在 DTI-FA 图像配准中与 PSO, CSO 及 Powell 算法进行了比较,结果表明本文算法在 DTI-FA 图像配准方面也取得了较好的效果,从主观和客观上都体现了本文算法应用于 DTI-FA 图像配准的可行性和有效性。在基于张量的 DTI 图像配准中,关键是在张量空间中寻找最优的平移和旋转运动参数,配准问题转化为最优问题,结合混沌思想鸡的改进群优化算法(CICS0)更适合解决该类问题。

## 参考文献

- [1] KENNEDY J, EBERHART R. Particle swarm optimization [C]// Proceedings of IEEE International Conference on Neural Networks. Perth Australia; IEEE Press, 1995: 1942-1948.
- [2] DORIGO M, MANIEZZO V. Ant system: optimization by a colony of cooperating agents[J]. IEEE Transactions on Systems Man and Cybernetics, Part B: Cybernetics, 1996, 26(1): 29-41.
- [3] LI X L, SHAO A, QIAN J X. An optimizing method based on autonomous animats; fish-swarm algorithm[J]. Systems Engineering Theory and Practice, 2002, 22(11): 32-38.
- [4] LI W W. Research on control and optimization of Metrointelligent transportation system[D]. Hangzhou; Zhejiang University, 2003. (in Chinese)  
李威武. 城域智能交通系统中的控制与优化问题研究[D]. 杭州: 浙江大学, 2003.
- [5] YANG X S. A new metaheuristic bat-inspired algorithm[C]// Nature Inspired Cooperative Strategies for Optimization(NICSO 2010). Berlin Heidelberg; Springer, 2010.
- [6] MENG X, LIU Y, GAO X Z, et al. A New Bio-inspired Algorithm: Chicken Swarm Optimization[M]// Advances in Swarm Intelligence. Springer International Publishing, 2014: 86-94.
- [7] HONG Y, YU F Q. Improved Chicken Swarm Optimization and its Application in Coefficients Optimization of Multiclassifier [J]. Computer Engineering and Applications, 2017, 53(9): 158-161. (in Chinese)  
洪杨, 于凤芹. 改进的鸡群算法并用于多分类器系数优化[J]. 计算机工程与应用, 2017, 53(9): 158-161.
- [8] GUAN H T. Application and Implementation of Cloud Computing Task Scheduling of Chaos Particle Swarm Chicken Swarm Fusion Optimization Algorithm[D]. Changchun; Jilin University, 2016. (in Chinese)  
关鹤童. 基于混沌粒子群鸡群融合优化算法的云任务调度应用与实现[D]. 长春: 吉林大学, 2016.
- [9] KONG F, WU D H. An Improved Chicken Swarm Optimization Algorithm[J]. Journal of Jiangnan University, 2015, 14(6): 681-688. (in Chinese)  
孔飞, 吴定会. 一种改进的鸡群算法[J]. 江南大学学报, 2015, 14(6): 681-688.
- [10] MENG X B, GAO X Z, LU L H, et al. A New Bio-inspired Algorithm: Bird Swarm Optimization[OL]. <http://dx.doi.org/10.1080/0952813X.2015.1042530>.
- [11] SUN J Z, GENG G H, WANG S Y, et al. Chaotic Hybrid Bacterial Colony Chemotaxis Algorithm Based on Tent Map[J]. Journal of Software, 2012, 7(5): 1030-1037.
- [12] LU Z G, ZHAO H, XIAO H F, et al. An Improved Multi-objective Bacteria Colony Chemotaxis Algorithm and Convergence Analysis[J]. Applied Soft Computing, 2015, 31(C): 274-292.
- [13] BASSER P J, MATTIELLO J, LEBIHAN D. MR Diffusion Tensor Spectroscopy and Imaging[J]. Biophysical Journal, 1994, 66(1): 259-267.
- [14] BASSER P J, PAJEVIC, PIERPAOLI, et al. In Vivo Fiber Tractography using DT-MRI data[J]. Magnetic Resonance in Medicine, 2000, 44(4): 625-632.
- [15] WANG Z. The Research and Implementation of the Magnetic Resonance Diffusion Tensor Image Registration Algorithm[D]. Harbin; Harbin Institute of Technology, 2013. (in Chinese)  
王钊. 对磁共振扩散张量图像配准算法的研究与实现[D]. 哈尔滨: 哈尔滨工业大学, 2013.
- [16] LIU G, ZHOU H, LIANG X G, et al. Image Registration Algorithm for Infrared and Visible Light Based on Non-sampled Contourlet Transform[J]. Computer Science, 2016, 43(11): 313-316. (in Chinese)  
刘刚, 周珩, 梁晓庚, 等. 非下采样轮廓波域红外与可见光图像配准算法[J]. 计算机科学, 2016, 43(11): 313-316.
- [17] ZHEN S. The Application of Magnetic Resonance Diffusion Tensor Image Interpolation Method and Image Registration [D]. Hangzhou; Zhejiang University, 2013. (in Chinese)  
甄帅. 磁共振扩散张量图像插值方法及其图像配准应用研究[D]. 杭州: 浙江大学, 2013.
- [18] LIU Y H, YAN D Q, LIU C F. Research on Medical Image Registration Classification [J]. Computer Science, 2015, 42(11): 22-27. (in Chinese)  
刘益含, 闫德勤, 刘彩凤. 医学图像配准分类研究[J]. 计算机科学, 2015, 42(11): 22-27.
- [19] DU X G, DANG J W, WANG Y P, et al. Mutual Information Medical Image Registration Based on Firefly Algorithm[J].