**进化算法**

# 鱼群优化算法

## 介绍

人工鱼群算法(Artificial Fish Swarm Algorithm，AFSA)是Li Xiao-lei在2002年提出的(Yazdani, Toosi, & Meybodi, 2010)[1]，目的是模仿鱼类捕食、群集、跟随、移动等行为。AFSA是基于鱼类集体向某个目标运动，并受到自然的启发，是一种并行和随机搜索算法。

与狮子和猴子不同，在像鱼类这种动物物种中没有领导者，每个成员都有自我组织的行为。鱼对它们的群体和环境一无所知，通过相邻成员之间的数据交换在环境中随意移动，这种交互作用为鱼群优化带来了更多的复杂性。

AFSA具有容错性和灵活性等特点，它在资源均衡、模糊聚类、数据挖掘、扩频码估计、DNA编码序列优化、信号处理、图像处理、改进神经网络、作业调度等方面有着广泛的应用。AFSA具有收敛速度快、效率高等优点，同时也存在时间复杂度高、全局搜索与局部搜索不平衡等缺点。

## 人工鱼群算法

人工鱼(AF)是真实鱼的仿制品，用于分析和问题解释（Neshat、Sepidnam、Sargolzaei和Toosi(2012)）。鱼类大多生活在食物充足的地区，它们通过跟随其他鱼类或单独寻找食物，向食物较多的地区移动。鱼类数量最多的地区通常是食物最多的。每条人工鱼的下一步行为取决于它目前的状态以及局部的环境状态，AF通过自身行为以及同伴的行为来影响环境。

AF模型包含两个部分：变量和函数。

变量包括：

* X=(x1,x2,...,xn)：每只AF的当前位置。
* Visual：视距（感知距离）。
* Xv：在某个运动中在视觉位置的状态。
* Step：步长。
* try\_num：尝试次数。
* δ：拥挤因子（0<δ<1）。
* θ：拥挤参数θ∈{0,1}。
* n：点个数（鱼的总数）。
* di,j：人工鱼个体之间的距离di,j=‖Xi-Xj‖。

函数包括人工鱼的各种行为：

* AF\_Prey
* AF\_Swarm
* AF\_Follow
* AF\_Move
* AF\_Leap
* AF\_Evaluate

图1显示了一条人工鱼的视野。

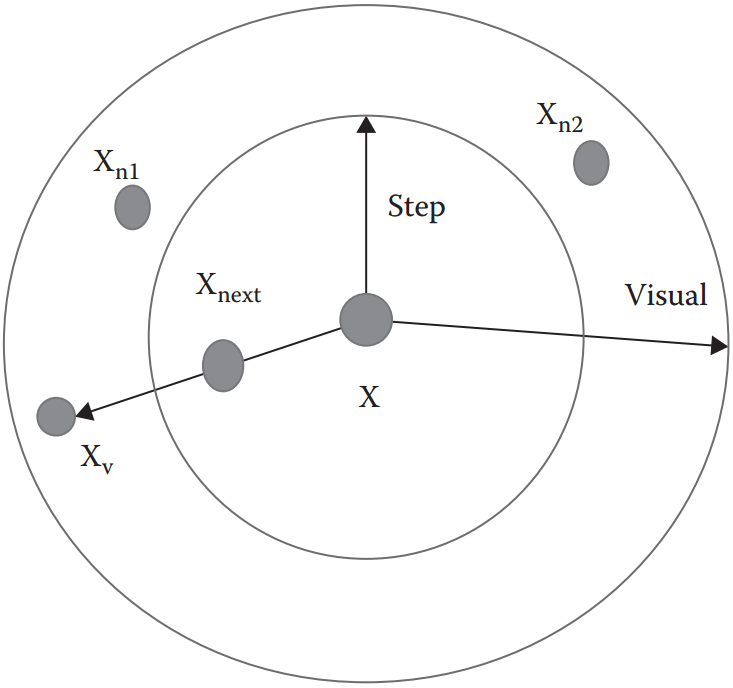


图1 人工鱼的视野概念

**鱼类表现出的行为模式是：**

鱼类大多生活在食物充足的地区。

基于这一特性，对鱼类的行为进行仿真，寻找全局最优解，这是AFSA研究的基本思路。

函数Rand()产生一个介于0和1之间的随机数。

AF\_Prey（觅食行为）：这是鱼类对食物的基本生物学行为。一般来说，鱼通过视觉感知水中食物的浓度来决定向哪移动，然后选择移动的方向。设人工鱼当前状态为Xi，在其感知范围内随机选择一个状态Xj（式（1）），如果在求极大问题中，f(Xi)<f(Xj)(或在求极小问题中，f(Xi)>f(Xj)，因极大和极小问题可以互相转换，所以以下均以求极大问题讨论)，则向该方向前进一步（式（2））；反之，再重新随机选择状态Xj，判断是否满足前进条件；反复几次（try\_num）后，如果仍不满足前进条件，则随机移动一步（式（3））。

 （1）

 （2）

 （3）

AF\_Prey的伪代码如下：

function AF\_Prey（）

{

for (i=0;i< try\_num;i++)

{

式（1）；

if(f(Xi)<f(Xj))

式（2）；

else

式（3）；

}

}

AF\_Swarm（聚群行为）：鱼群在移动的过程中会自然地成群聚集，这是一种生活习惯，可以保证群体的存在，避免危险。设人工鱼当前状态为Xi，探索当前邻域内(即di,j<Visual)的伙伴数目nf及中心位置Xc，如果nf/n<δ且f(Xc)>f(Xi)，表明伙伴中心有较多的食物并且不太拥挤，则朝伙伴的中心位置方向前进一步（式（4））；否则执行觅食行为。

 （4）

AF\_Swarm的伪代码如下：

function AF\_Swarm（）

{

nf=0;Xc=0;

for (j=0;j<friend\_num;j++)

{

if (di,j<Visual)

{

nf++;Xc+=Xj;

}

}

Xc=Xc/nf;

if (nf/n<δ and f(Xc)>f(Xi))

式（4）；

else

AF\_Prey（）；

}

AF\_Follow：在鱼群的移动过程中，当一条鱼或几条鱼找到食物时，附近的伙伴会迅速移动并到达食物。设人工鱼当前状态为Xi，探索当前邻域内(即di,j<Visual)的伙伴中f(Xj)为最大的伙伴Xj，nf为Xj的邻域内鱼的数量，如果nf/n<δ且f(Xj)>f(Xi)，表明伙伴Xj的状态具有较高的食物浓度并且其周围不太拥挤，则朝伙伴Xj的方向前进一步（式（5））；否则执行觅食行为。

 （5）

AF\_Follow的伪代码如下：

function AF\_Follow（）

{

fmax=-∞；

for (j=0;j<friend\_num;j++)

{

if (di,j<Visual and f(Xj)>fmax)

{

fmax=f(Xj);Xmax=Xj;

}

}

nf=0;

for (j=0;j<friend\_num;j++)

{

if (dmax,j<Visual)

{

nf++;

}

}

if (nf/n<δ and f(Xj)>f(Xi))

式（5）；

else

AF\_Prey（）；

}

AF\_Move：鱼随机游动，并在更大的范围内寻找食物或同伴。

AF\_Leap：鱼停在某处，每一种AF行为都将逐渐相同，在某些迭代过程中，目标值(食物浓度)之间的差异会变小。它可能会陷入局部极值，并随机地将参数变为静止状态，从而跳出当前状态。

Stagnation：当目标函数值在一定次数的迭代中没有变化时，就会发生停滞。

AFSA的伪代码如下：

Start

for 每条AF∈[1...f]

初始化Xi。

end for

将argmin f(Xi)的Xi记录在公告栏。

for每条AF∈[1...f]

对Xi(t)执行群聚行为，计算Xi,swarm。

对Xi(t)执行追尾行为，计算Xi,follow。

对Xi(t)执行觅食行为，计算Xi,prey。

if min(f(Xi,swarm),f(Xi,follow), f(Xi, prey))<f(Xi)then

Xi(t+1)=argmin (f(Xi,swarm),f(Xi,follow), f(Xi, prey));

end if

end for

End

**参考文献**

1. Yazdani, D., A. Nadjaran Toosi, and M.R. Meybodi. *Fuzzy Adaptive Artificial Fish Swarm Algorithm*. in *AI 2010: Advances in Artificial Intelligence*. 2011. Berlin, Heidelberg: Springer Berlin Heidelberg.