

# Biological Intelligence研究综述

翟卫欣<sup>1</sup>

<sup>1</sup>北京城市实验室BCL

2015年5月4日

# Outline

- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

# 前言

*“The biggest innovations of the twenty-first century will be the intersection of biology and technology.”*

*(Steve Jobs)*

*“Biological and organic forms that do not exist in the natural world can be created in the lab.” (Ali Brivanlou)*

*“Over the next 20 years, synthetic genomics is going to be the standard for making anything.” (Craig Venter)*

# 前言

随着生命科学突飞猛进地发展，我们对于生态系统和生物本身内部规律的认识也日益加深，biological intelligence的研究也成为一个热点。许多科研工作者研究生物的内部机理或者通过建模来描述、学习生物的各种行为，均取得了不少成绩。

# 前言

*Natural Computing refers to computational processes observed in nature, and human-designed computing inspired by nature. When complex natural phenomena are analyzed in terms of computational processes, our understanding of both nature and the essence of computation is enhanced. Characteristic for human-designed computing inspired by nature is the metaphorical use of concepts, principles and mechanisms underlying natural systems. Natural computing includes evolutionary algorithms, neural networks, molecular computing and quantum computing.(Natural Computing)*

## 前言

比如在近几年的研究中，有人通过"萤火虫算法"来提高图像处理的效率，通过"蜂群算法"来进行数值计算、求最优解问题，通过模拟Wolbachia菌的感染方式来解决多个目标的最优化问题等等，这些都是很典型地借助生物的智能来解决人类问题的案例。

本报告列举出了4个较为成熟的计算机算法，它们的相同之处是都来自于生物学规律的启发。同时，本报告还列举出了另外一些有意思的生物学现象，可能会对相应的科研和工程项目带来一些帮助。

# Outline

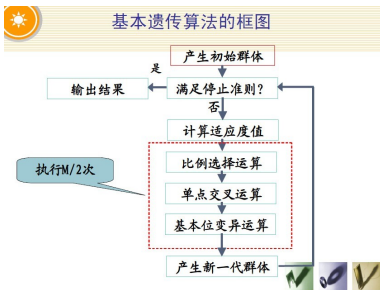
- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

# 遗传算法

遗传算法是计算数学中用于解决最优化问题的搜索算法，它通常的实现方式为一种计算机模拟。它的依据是对于一个最优化问题，一定数量的候选解（称为个体）的抽象表示（称为染色体）的种群会向更好的解进化。传统上，解用二进制表示（即0和1的串），但也可以用其他表示方法。进化从完全随机个体的种群开始，之后一代一代发生。在每一代中，整个种群的适应度被评价，从当前种群中随机地选择多个个体（基于它们的适应度），通过自然选择和突变产生新的生命种群，该种群在算法的下一代迭代中成为当前种群。



# 遗传算法



总体来说，包括初始化、个体评价、选择运算、交叉运算、变异运算、终止条件判断这六个步骤。其中选择运算、交叉运算、变异运算这三个是算法的关键。

# 遗传算法举例

问题：求解二元函数的最大值：

$$\max f(x_1, x_2) = x_1^2 + x_2^2$$

s. t.

$$x_1 \in \{1, 2, 3, 4, 5, 6, 7\}$$

$$x_2 \in \{1, 2, 3, 4, 5, 6, 7\}$$

## 遗传算法举例

**个体编码** 遗传算法的运算对象是表示个体的符号串，所以必须把变量 $x_1, x_2$  编码为一种符号串。本题中，用无符号二进制整数来表示。因 $x_1, x_2$  为 $0 \sim 7$ 之间的整数，所以分别用3位无符号二进制整数来表示，将它们连接在一起所组成的6位无符号二进制数就形成了个体的基因型，表示一个可行解。例如，基因型 $X=101110$  所对应的表现型是： $x=[5, 6]$ 。个体的表现型 $x$ 和基因型 $X$ 之间可通过编码和解码程序相互转换。

## 遗传算法举例

**初始群体的产生** 遗传算法是对群体进行的进化操作，需要给其准备一些表示起始搜索点的初始群体数据。本例中，群体规模的大小取为4，即群体由4个个体组成，每个个体可通过随机方法产生。如：011101，101011，011100，111001。

**适应度计算** 遗传算法中以个体适应度的大小来评定各个个体的优劣程度，从而决定其遗传机会的大小。本例中，目标函数总取非负值，并且是以求函数最大值为优化目标，故可直接利用目标函数值作为个体的适应度。

## 遗传算法举例

**选择运算** 选择运算(或称为复制运算)把当前群体中适应度较高的个体按某种规则或模型遗传到下一代群体中。一般要求适应度较高的个体将有更多的机会遗传到下一代群体中。本例中,我们采用与适应度成正比的概率来确定各个个体复制到下一代群体中的数量。其具体操作过程是:先计算出群体中所有个体的适应度的总和 $\sum f_i (i=1,2,\dots,M)$ ;其次计算出每个个体的相对适应度的大小 $f_i / \sum f_i$ ,它即为每个个体被遗传到下一代群体中的概率,每个概率值组成一个区域,全部概率值之和为1;

# 遗传算法举例

选择运算(接上) 最后再产生一个0到1之间的随机数，依据该随机数出现在上述哪一个概率区域内来确定各个个体被选中的次数。



# 遗传算法举例

**交叉运算** 交叉运算是遗传算法中产生新个体的主要操作过程，它以某一概率相互交换某两个个体之间的部分染色体。本例采用单点交叉的方法，其具体操作过程是：先对群体进行随机配对；其次随机设置交叉点位置；最后再相互交换配对染色体之间的部分基因。

个体编号	选择结果	配对情况	交叉点位置	交叉结果
1	011101			011001
2	111001	1-2	1-2: 2	111101
3	101011	3-4	3-4: 4	101001
4	111001			111011

可以看出，其中新产生的个体“111101”、“111011”的适应度较原来两个个体的适应度都要高。

# 遗传算法举例

**变异运算** 变异运算是对个体的某一个或某一些基因片段上的基因值按某一较小的概率进行改变，它也是产生新个体的一种操作方法。本例中，我们采用基本位变异的方法来进行变异运算，其具体操作过程是：首先确定出各个个体的基因变异位置，下表所示为随机产生的变异点位置，其中的数字表示变异点设置在该基因座处；然后依照某一概率将变异点的原有基因值取反。

个体编号	交叉结果	变异点	变异结果	子代群体 $p(t)$
1	011001	4	011101	011101
2	111101	5	111111	111111
3	101001	2	111001	111001
4	111011	6	111010	111010



## 遗传算法举例

对群体 $P(t)$ 进行一轮选择、交叉、变异运算之后可得到新一代的群体 $p(t+1)$ 。

个体编号	子群体 $p(1)$	$x_1$	$x_2$	适应度	占总数的百分比
1	011101	3	5	34	0.14
2	111111	7	7	98	0.42
3	111001	7	1	50	0.21
4	111010	7	2	53	0.23
总和				235	1

从上表中可以看出，群体经过一代进化之后，其适应度的最大值、平均值都得上表中可以看出，群体经过一代进化之后，其适应度的最大值、平均值都得到了明显的改进。事实上，这里已经找到了最佳个体“111111”。

# Outline

- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

## 蚁群算法

蚁群算法(ant colony optimization, ACO), 又称蚂蚁算法, 是一种用来在图中寻找优化路径的机率型算法。它由Marco Dorigo于1992年在他的博士论文中提出, 其灵感来源于蚂蚁在寻找食物过程中发现路径的行为。蚁群算法是一种模拟进化算法, 初步的研究表明该算法具有许多优良的性质。针对PID控制器参数优化设计问题, 将蚁群算法设计的结果与遗传算法设计的结果进行了比较, 数值仿真结果表明, 蚁群算法具有一种新的模拟进化优化方法的有效性和应用价值。

## 蚁群算法的来源

各个蚂蚁在没有事先告诉他们食物在什么地方的前提下开始寻找食物。当一只找到食物以后，它会向环境释放一种挥发性分泌物pheromone(称为信息素,该物质随着时间的推移会逐渐挥发消失，信息素浓度的大小表征路径的远近)来实现的，吸引其他的蚂蚁过来，这样越来越多的蚂蚁会找到食物。有些蚂蚁并没有像其它蚂蚁一样总重复同样的路，他们会另辟蹊径，如果另开辟的道路比原来的其他道路更短，那么，渐渐地，更多的蚂蚁被吸引到这条较短的路上来。最后，经过一段时间运行，可能会出现一条最短的路径被大多数蚂蚁重复着。

## 蚁群算法的说明

**范围** 蚂蚁观察到的范围是一个方格世界，蚂蚁有一个参数为速度半径（一般是3），那么它能观察到的范围就是 $3*3$ 个方格世界，并且能移动的距离也在这个范围之内。

**环境** 蚂蚁所在的环境是一个虚拟的世界，其中有障碍物，有别的蚂蚁，还有信息素，信息素有两种，一种是找到食物的蚂蚁洒下的食物信息素，一种是找到窝的蚂蚁洒下的窝的信息素。每个蚂蚁都仅仅能感知它范围内的环境信息。环境以一定的速率让信息素消失。

## 蚁群算法的说明

**觅食规则** 在每只蚂蚁能感知的范围内寻找是否有食物，如果有就直接过去。否则看是否有信息素，并且比较在能感知的范围内哪一点的信息素最多，这样，它就朝信息素多的地方走，并且每只蚂蚁都会以小概率犯错误，从而并不是往信息素最多的点移动。蚂蚁找窝的规则和上面一样，只不过它对窝的信息素做出反应，而对食物信息素没反应。

## 蚁群算法的说明

**移动规则** 每只蚂蚁都朝向信息素最多的方向移，并且，当周围没有信息素指引的时候，蚂蚁会按照自己原来运动的方向惯性地运动下去，并且，在运动的方向有一个随机的小的扰动。为了防止蚂蚁原地转圈，它会记住刚才走过了哪些点，如果发现要走的下一点已经在之前走过了，它就会尽量避开。

# 蚁群算法的说明

**避障规则** 如果蚂蚁要移动的方向有障碍物挡住，它会随机的选择另一个方向，并且有信息素指引的话，它会按照觅食的规则行为。

**信息素规则** 每只蚂蚁在刚找到食物或者窝的时候散发的信息素最多，并随着它走远的距离，播撒的信息素越来越少。



## 蚁群算法的说明

**总结** 根据这几条规则，蚂蚁之间并没有直接的关系，但是每只蚂蚁都和环境发生交互，而通过信息素这个纽带，实际上把各个蚂蚁之间关联起来了。比如，当一只蚂蚁找到了食物，它并没有直接告诉其它蚂蚁这儿有食物，而是向环境播撒信息素，当其它的蚂蚁经过它附近的时候，就会感觉到信息素的存在，进而根据信息素的指引找到了食物。

## 蚁群算法举例

蚁群问题是解决TSP问题的一种高效方法。TSP问题(travelling salesman problem), 即旅行推销员问题或货郎担问题, 是数学领域中著名问题之一。它的大意是: 假设有一个旅行商人要拜访 $n$ 个城市, 他必须选择所要走的路径, 路径的限制是每个城市只能拜访一次, 而且最后要回到原来出发的城市。路径的选择目标是要求得的路径路程为所有路径之中的最小值。抽象为: 求解遍历图 $G = (V, E, C)$ , 所有的节点一次并且回到起始节点, 使得连接这些节点的路径成本最低。

# Outline

- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

# 神经网络算法

人工神经网络（Artificial Neural Networks, ANN）系统是20世纪40年代后出现的。它是由众多的神经元可调的连接权值连接而成，具有大规模并行处理、分布式信息存储、良好的自组织自学习能力等特点。BP（Back Propagation）算法又称为误差反向传播算法，是人工神经网络中的一种监督式的学习算法。BP神经网络算法在理论上可以逼近任意函数，基本的结构由非线性变化单元组成，具有很强的非线性映射能力。而且网络的中间层数、各层的处理单元数及网络的学习系数等参数可根据具体情况设定，灵活性很大，在优化、信号处理与模式识别、智能控制、故障诊断等许多领域都有着广泛的应用前景。

# 神经网络算法的原理

神经系统是由数目繁多的神经元组合而成。神经网络通过感觉器官和神经接受各种信息，传递至中枢神经系统内，经过对信息的分析和综合，再通过运动神经发出控制信息。神经元和其他类型的细胞一样，包括有细胞膜、细胞质和细胞核。但是神经细胞的形态比较特殊，具有许多突起，共可分为细胞体、轴突和树突三部分。其中突起的作用是传递信息。树突是输入信号的突起，而轴突是输出端的突起。突触可分为兴奋性与抑制性两种类型，它相应于神经元之间耦合的极性。各神经元之间的连接强度和极性有所不同，并且都可调整、基于这一特性，人脑具有存储信息的功能。利用大量神经元相互联接组成人工神经网络可显示出人的大脑的某些特征。

## 神经网络算法举例

人工神经网络反映了人脑功能的若干基本特性。与数字计算机比较，人工神经网络在构成原理和功能特点等方面更加接近人脑，它不是按给定的程序一步一步地执行运算，而是能够自身适应环境、总结规律、完成某种运算、识别或过程控制。人工神经网络首先要以一定的学习准则进行学习然后才能工作。现以人工神经网络对于写“A”、“B”两个字母的识别为例进行说明，规定当“A”输入网络时，应该输出“1”，而当输入为“B”时，输出为“0”。所以网络学习的准则应该是：如果网络作出错误的的判决，则通过网络的学习，应使得网络减少下次犯同样错误的的可能性。

## 神经网络算法举例

首先，给网络的各连接权值赋予(0, 1)区间内的随机值，将“A”所对应的图象模式输入给网络，网络将输入模式加权求和、比较，再进行非线性运算，得到网络的输出。在此情况下，网络输出为“1”和“0”的概率各为50%，也就是说是完全随机的。这时如果输出为“1”（结果正确），则使连接权值增大，以便使网络再次遇到“A”模式输入时，仍然能作出正确的判断。如果输出为“0”（即结果错误），则把网络连接权值朝着减小综合输入加权值的方向调整，其目的在于使网络下次再遇到“A”模式输入时，减小犯同样错误的可能性。

## 神经网络算法举例

如此操作调整，当给网络轮番输入若干个手写字母“**A**”、“**B**”后，经过网络按以上学习方法进行若干次学习后，网络判断的正确率将大大提高。这说明网络对这两个模式的学习已经获得了成功，它已将这两个模式分布地记忆在网络的各个连接权值上。当网络再次遇到其中任何一个模式时，能够作出迅速、准确的判断和识别。一般说来，网络中所含的神经元个数越多，则它能记忆、识别的模式也就越多。



# Outline

- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

# 粒子群算法

粒子群算法，也称粒子群优化算法（Particle Swarm Optimization），缩写为PSO。PSO模拟鸟群的捕食行为。设想这样一个场景：一群鸟在随机搜索食物，在这个区域里只有一块食物，所有的鸟都不知道食物在那里，但是他们知道当前的位置离食物还有多远。那么找到食物的最优策略是什么呢。最简单有效的就是搜寻目前离食物最近的鸟的周围区域。PSO从这种模型中得到启示并用于解决优化问题。PSO中，每个优化问题的解都是搜索空间中的一只鸟。我们称之为“粒子”。所有的粒子都有一个由被优化的函数决定的适应值(fitness value)，和一个速度决定他们飞翔的方向和距离。然后粒子们就追随当前的最优粒子在解空间中搜索。

# 粒子群算法

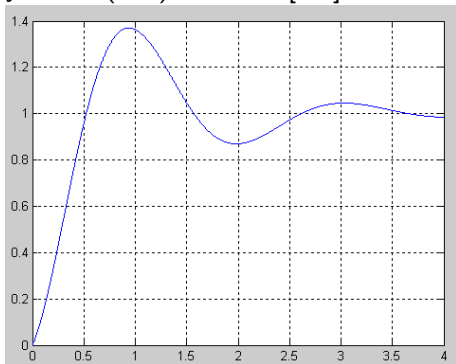
PSO初始化为一群随机粒子(随机解)。然后通过迭代找到最优解。在每一次迭代中,粒子通过跟踪两个"极值"来更新自己。第一个就是粒子本身所找到的最优解,这个解叫做个体极值pBest。另一个极值是整个种群目前找到的最优解,这个极值是全局极值gBest。另外也可以不用整个种群而只是用其中一部分作为粒子的邻居,那么在所有邻居中的极值就是局部极值。

# 粒子群算法

粒子群算法，是近年来发展起来的一种新的进化算法。它从随机解出发，通过迭代寻找最优解，它也是通过适应度来评价解的品质，但它比遗传算法规则更为简单，它没有遗传算法的“交叉” (Crossover) 和“变异” (Mutation) 操作，它通过追随当前搜索到的最优值来寻找全局最优。这种算法以其实现容易、精度高、收敛快等优点引起了学术界的重视，并且在解决实际问题中展示了其优越性。同时，粒子群算法是一种并行算法。

## 粒子群算法举例

$y=1-\cos(3*x)*e^{-x}$ 的在 $[0,4]$ 最大值。

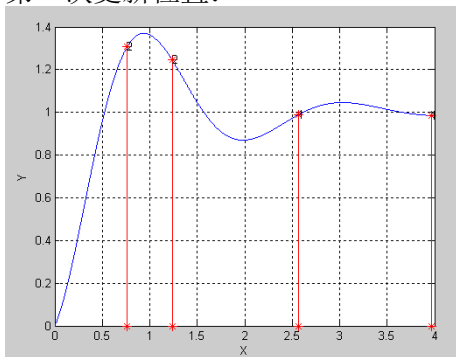


# 粒子群算法举例

当 $x=0.9350-0.9450$ ，达到最大值 $y=1.3706$ 。为了得到该函数的最大值，我们在 $[0, 4]$ 之间随机撒一些点，为了演示，我们放置两个点，并且计算这两个点的函数值，同时给这两个点设置在 $[0, 4]$ 之间的一个速度。下面这些点就会按照一定的公式更改自己的位置，到达新位置后，再计算这两个点的值，然后再按照一定的公式更新自己的位置。直到最后在 $y=1.3706$ 这个点停止自己的更新。这个过程与粒子群算法作为对照如下：这两个点就是粒子群算法中的粒子。该函数的最大值就是鸟群中的食物计算两个点函数值就是粒子群算法中的适应值，计算用的函数就是粒子群算法中的适应度函数。更新自己位置的公式就是粒子群算法中的位置速度更新公式。第一次初始化：

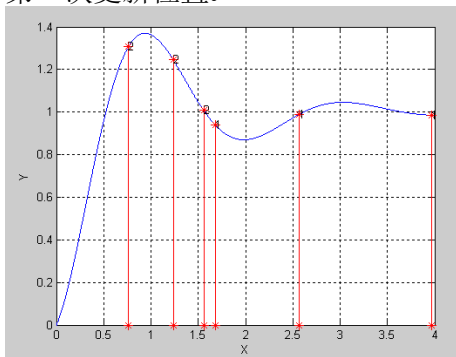
# 粒子群算法举例

第一次更新位置:



# 粒子群算法举例

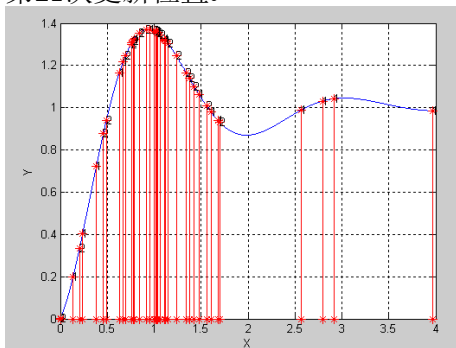
第二次更新位置。





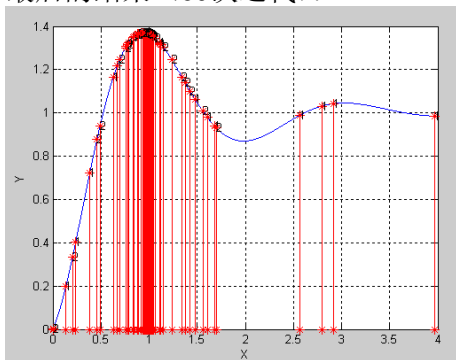
# 粒子群算法举例

第21次更新位置。



# 粒子群算法举例

最后的结果（30次迭代）：

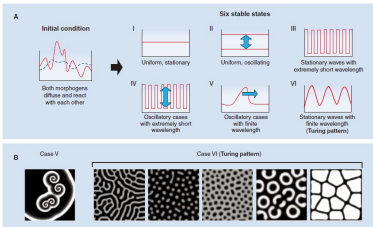


# Outline

- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

# 胚胎发育的RD模型

该模型模拟了胚胎发育的整个过程：在胚胎发育初期，有多种可能的生长波动的平衡性稳态，而依据RD模型，当对任一平衡态的参数进行调整后，就会生成不同的完整胚胎，这一理论在海螺的实际实验中也得到了验证。图A是RD模型中2因素生长素稳定的六种稳态。图B是Turing模型根据参数的微调形成的不同的发育初期的二维形态。



## 胚胎发育的RD模型

图C是Turing模型根据参数的微调形成的不同的发育成熟的形态（海螺为例）。

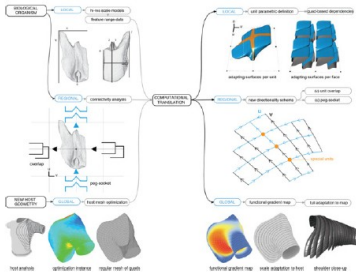


# Outline

- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结

# 塞内加尔多鳍鱼的网络结构

科学家从塞内加尔多鳍鱼的多层次结构中得到灵感，将原有的同级的连接结构进行改进，设计出多级连接的结构，这种结构能够具有更强的连通性和稳定性，已经应用于衣物的肩部的设计，其结构的韧性更强。



# Outline

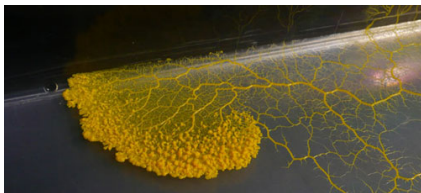
- 1 前言
- 2 四类典型算法
  - 遗传算法
  - 蚁群算法
  - 神经网络算法
  - 粒子群算法
- 3 三种生物形态
  - 胚胎发育的RD模型
  - 塞内加尔多鳍鱼的网络结构
  - 多头绒泡菌的网状结构
- 4 总结



## 多头绒泡菌的生物学特征

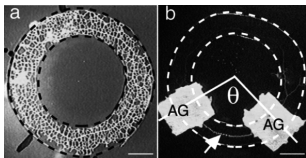
- 多头绒泡菌(*Physarum polycephalum*)是一种黏菌(*slime mold*), 具有不定的形状和黏性的外表, 多呈黄色或者淡黄色。每个多头绒泡菌都是一个肉眼可见的巨大的单细胞个体, 它能够根据不同的地势条件、食物分布状况和各类物理化学刺激来改变自身的形态来形成从起点位置到各个食物放置位置的最短有效路径(或路网), 这种性质能够帮助解决一系列的图形优化和空间计算的问题。

# 多头绒泡菌的生物学特征



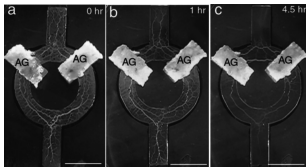
- 多头绒泡菌的形态

## 最短路径问题：优弧劣弧的选择



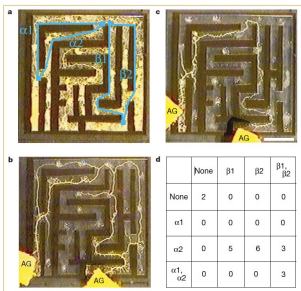
- 在圆环形的培养基中多头绒泡菌的生长模式，a图为没有任何食物源的情况下的多头绒泡菌的身体形态；b图为加了两块琼脂凝胶之后的多头绒泡菌的身体形态(只选择了劣弧而没有在优弧的部分分布)。

## 最短路径问题：更为复杂的情况



- 在一种特殊形状的培养基中多头绒泡菌的生长模式，相比于之前的实验，本实验的特殊之处在于增加了两个穷尽点(最上方和最下方)，而不是处处对称的，实验表明多头绒泡菌会选择去向两个点进行试探性地生长，直到发现没有食物便收回。

## 最短路径问题：迷宫问题

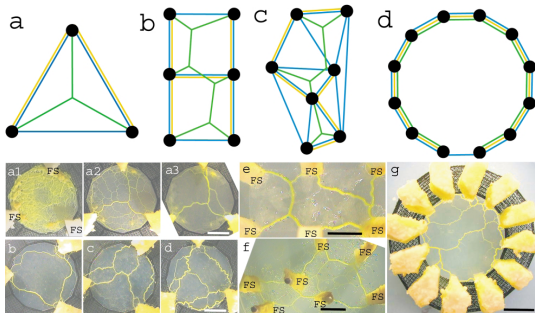


- 该试验证明了这种体型巨大的单细胞黏菌能够改变自身的形状结构在两个琼脂凝胶之间建立最短路径。实验环境是在一个形状为的迷宫容器中进行的，迷宫有4条通路，其中  $\alpha 1 > \alpha 2$ ， $\beta 1 \approx \beta 2$ 。**a**是迷宫中放入黏菌之后，黏菌的形态；**b**是在迷宫的入口和出口处放入琼脂凝胶后4小时黏菌的形态，此时黏菌建立了多条通路；**c**是又过了4小时后黏菌的形态，此时黏菌通常情况下舍弃了较长的通路，而把身体主要分布于较短的通路中；**d**是多次重复试验之后的统计结果，黏菌均未选择  $\alpha 1$  作为最终通路。(注意4图a、b、c、d的顺序)

## 最短路径问题：迷宫问题的后续

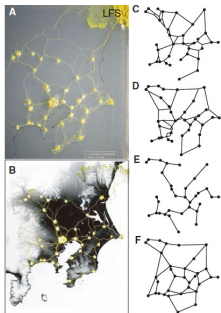
- 多头绒泡菌能够解决“迷宫问题”这一现象引起了科学家的强烈兴趣，这种黏菌出于生物寻找食物并减少自身损耗的本能解决了一个较为复杂的问题，又有实验利用水溶的培养物来在迷宫中进行实验，结果表明多头绒泡菌的伸展是基于化学梯度来进行的。除此之外，与迷宫问题相类似的多种有趣的实验也被提出，均证明了这种单细胞生物的强大计算能力和移动能力。

## 路网优化问题：简单的网络解决方案



- 由于多头绒泡菌需要在获取能量的同时，最大限度减少自己的消耗，于是就要减少自己的体积。这一特性不仅体现在两个点的最短路径的实验中，在多个位置都放置有食物的时候，它同样需要对自己的身体结构进行优化，也就是网络解决方案。在点较少的时候，理论上的最短路径问题与实验中上黏菌所形成的形态是一致的。

## 路网优化问题：与实际交通路网的比对



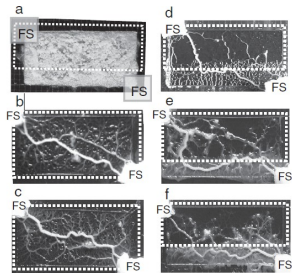
- 这个实验大概是是和多头绒泡菌相关的所有实验中最令人兴奋的一个。将食物按照东京铁路的各个站的位置放置，加入多头绒泡菌后，会自动生成一个网状图（如A所示）；B为利用感光性来模拟了陆地高度特征生成的网状图，更为接近实际情况；C、D为实际铁路和模拟生成的铁路线路，十分接近。E为模拟多个城市分布的，多头绒泡菌生成的minimum spanning tree，F是在E基础上构造完整的minimum spanning tree。由于东京的铁路线路是全球最高效的设计之一，代表了人类设计界的最高水平，而轻而易举地被多头绒泡菌实现，这无疑令人震惊。



## 路网优化问题：与实际交通路网的比对

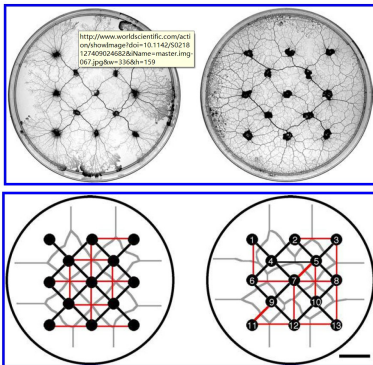
- 日本东京铁路的对比实验是由Nakagaki的团队完成的，在这之后，Adamatzky又将非洲、澳大利亚、比利时、巴西、加拿大、中国、德国、伊比利亚、意大利、马来西亚、墨西哥、荷兰、英国、美国等国家和地区的机动车道拿来做实验，把多头绒泡菌放置在主要站点，把食物放置在各个交通站地区，并进行了逐一分析。同时有多项研究将多头绒泡菌的路网进行了分析，并与实际的路网进行比对和评价。

## 利用多头绒泡菌的厌光性



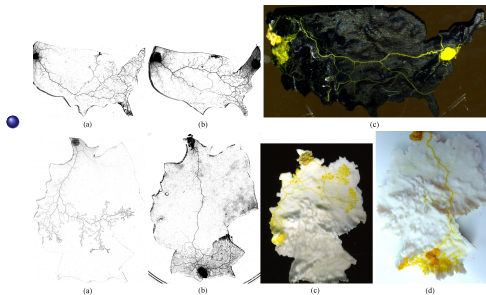
- 多头绒泡菌是一种厌光的生物，当长方形培养皿的对角位置放置有食物时，如果在实验中如果在培养皿内有部分光照，则多头绒泡菌的分布与无光照的情况下不同，多头绒泡菌会将大部分长度调整到阴影部分，而光照部分的长度则减少，虽然总长度增加了，但是考虑到有光照的影响，这种形态依然是最优的。图中的虚线范围为光照范围。

## 多个多头绒泡菌的相互作用



- 不同的多头绒泡菌之间相互不侵犯其他菌的空间，但又为了争取食物，尽量扩大自己的空间范围。实验中，在培养皿中放置多个多头绒泡菌，然后在每一个多头绒泡菌附近放置一个食物源，之后每个多头绒泡菌都会以格子的食物源为中心向四周生长，最后构成一个Voronoi图，其对偶图即为Delaunay三角网。

## 3D地形中的模拟实验



- 之前的研究都是考虑2维的情况的，3维的情况在2010年之后才开始有人研究。Adamatzky最初的实验选取了美国的Route20和德国的autobahn3D模型来培养多头绒泡菌并观察其所形成的路网。结果表明与真实情况有一定差异，问题原因可能是人类开凿公路能够穿越山洞，而菌不能。

# 利用多头绒泡菌的特性来论证某些算法的正确性

## Example

利用多头绒泡菌来验证universal computation只在资源有限的时候是可能的；

## Example

利用多头绒泡菌来验证spanning trees算法的可行性；

## Example

多种类型的改进的最短路径算法；

## Example

连接图算法的改进方法。





## 总结

本报告首先介绍了 *Biological Intelligence* 的意义以及目前的整体研究状况，又给出四种与生物机理相关的较为成熟的计算机算法：遗传算法、蚁群算法、神经网络算法和粒子群算法，这四类算法的核心思想与生物体或者生物结构的运算、组织方式不谋而合，正是体现着人类智慧与自然智慧的高度统一。接着，本报告介绍了三种生物形态，这是目前 *Biological Intelligence* 领域研究的几个热点问题，其中着重介绍了以多头绒泡菌为代表的黏菌的网状结构，这种简单的单细胞生物多次构造出奇妙的形态结构，同样是和缜密的人类的设计方式相一致，令人兴奋不已。

# 总结


***Using living things as a way to compute solutions to human problems***是biological intelligence的意义之所在。从最初的仿生学研究到如今的各类高效算法的提出，人类向生物的学习从未停止。自然界中仍有多种多样有意义的现象目前仍难以用科学来解释，简单的生物体可能在某些方面具有远高于人类的智能。未来，人类仍有很长的路要走。

# 参考文献 I




-  Handbook of genetic algorithms[M]. New York: Van Nostrand Reinhold, 1991.
-  Dorigo, M., Maniezzo, V., & Colorni, A. (1991). Positive feedback as a search strategy. Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy.
-  Dorigo M, Birattari M. Ant colony optimization[M]//Encyclopedia of Machine Learning. Springer US, 2010: 36-39.
-  Dayhoff J E. Neural network architectures: an introduction[M]. Van Nostrand Reinhold Co., 1990.






## 参考文献 II

-  J. Kennedy and R. C. Eberhart. Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks, IEEE Press, Piscataway, NJ, pp. 1942-1948, 1995.
-  J. Kennedy, R. C. Eberhart, and Y. Shi. Swarm Intelligence. Morgan Kaufmann, San Francisco, CA, 2001.
-  Gatenby R A, Gawlinski E T. A reaction-diffusion model of cancer invasion[J]. Cancer research, 1996, 56(24): 5745-5753.

## 参考文献 III

-  Duro-Royo J, Zolotovskiy K, Moga-Soldevila L, et al. MetaMesh: A hierarchical computational model for design and fabrication of biomimetic armored surfaces[J]. Computer-Aided Design, 2015, 60: 14-27.
-  Nakagaki, T., Yamada, H. and Toth, A., 2001, Path finding by tube morphogenesis in an amoeboid organism. Biophysical Chemistry, 92, pp. 47-52.
-  Nakagaki, T., Yamada, H. and Tóth, Á., 2000, Intelligence: Maze-solving by an amoeboid organism. Nature, 407, p. 470.



## 参考文献 IV

-  Adamatzky, A., 2012, Slime Mold Solves Maze in One Pass, Assisted by Gradient of Chemo-Attractants. IEEE Transactions on NanoBioscience, 11, pp. 131-134.
-  Nakagaki, T., 2004, Smart network solutions in an amoeboid organism. Biophysical Chemistry, 107, pp. 1-5.
-  Tero, A., Yumiki, K., Kobayashi, R., Saigusa, T. and Nakagaki, T., 2008, Flow-network adaptation in Physarum amoebae. Theory In Biosciences, 127, pp. 89-94.



## 参考文献 V

-  Denk, T., Neukum, G., Roatsch, T., Porco, C.C., Burns, J.A., Galuba, G.G., Schmedemann, N., Helfenstein, P., Thomas, P.C., Wagner, R.J. and West, R.A., 2010, Iapetus: Unique Surface Properties and a Global Color Dichotomy from Cassini Imaging. *Science*, 327, pp. 435-439.
-  A. Adamatzky, Ed. *Bio-Evaluation of World Transport Networks*. Singapore: World Scientific, 2012.



## 参考文献 VI

-  Gunji, Y., Shirakawa, T., Niizato, T., Yamachiyo, M. and Tani, I., 2011, An adaptive and robust biological network based on the vacant-particle transportation model. *Journal Of Theoretical Biology*, 272, pp. 187-200.
-  Jones, J., 2011, Influences on the formation and evolution of *Physarum polycephalum* inspired emergent transport networks. *Natural Computing*, 10, pp. 1345-1369.




## 参考文献 VII

-  Nakagaki, T., Iima, M., Ueda, T., Nishiura, Y., Saigusa, T., Tero, A., Kobayashi, R. and Showalter, K., 2007, Minimum-Risk Path Finding by an Adaptive Amoebal Network. Physical Review Letters, 99.
-  SHIRAKAWA, T., ADAMATZKY, A., GUNJI, Y. and MIYAKE, Y., 2009, ON SIMULTANEOUS CONSTRUCTION OF VORONOI DIAGRAM AND DELAUNAY TRIANGULATION BY PHYSARUM POLYCEPHALUM. International Journal Of Bifurcation And Chaos, 19, pp. 3109-3117.

## 参考文献 VIII

-  Adamatzky, A. and Martinez, G.J., 2013, Bio-Imitation of Mexican Migration Routes to the USA with Slime Mould on 3D Terrains. *Journal of Bionic Engineering*, 10, pp. 242-250.
-  Adamatzky, A.I., 2014, Route 20, Autobahn 7, and Slime Mold: Approximating the Longest Roads in USA and Germany With Slime Mold on 3-D Terrains. *IEEE Transactions on Cybernetics*, 44, pp. 126-136.

## 参考文献 IX




-  ADAMATZKY, A., DE LACY COSTELLO, B. and SHIRAKAWA, T., 2008, UNIVERSAL COMPUTATION WITH LIMITED RESOURCES: BELOUSOV-ZHABOTINSKY AND PHYSARUM COMPUTERS. *International Journal Of Bifurcation And Chaos*, 18, pp. 2373-2389.
-  Adamatzky, A., 2008, Growing spanning trees in plasmodium machines. *Kybernetes*, 37, pp. 258-264.
-  Adamatzky, A., 2009, If BZ medium did spanning trees these would be the same trees as Physarum built. *Physics Letters A*, 373, pp. 952-956.





# 参考文献 X

-  Zhang, X., Wang, Q., Adamatzky, A., Chan, F.T.S., Mahadevan, S. and Deng, Y., 2014, An Improved Physarum polycephalum Algorithm for the Shortest Path Problem. The Scientific World Journal, 2014, pp. 1-9.
-  Zhang, X., Zhang, Y., Zhang, Z., Mahadevan, S., Adamatzky, A. and Deng, Y., 2014, Rapid Physarum Algorithm for shortest path problem. Applied Soft Computing, 23, pp. 19-26.



## 参考文献 XI

-  Adamatzky, A., 2009, Developing proximity graphs by physarum polycephalum: Does the plasmodium follow the Toussaint Hierarchy? *Parallel Processing Letters*, 19, pp. 105-127.
-  Takamatsu, A., 2007, Mobiligence in an amoeboid cell, plasmodium of *Physarum polycephalum*. In *Second international symposium on mobiligence*. Awaji, Japan, pp. 48-51.
-  Adamatzky, A., 2010, *Physarum machines: computers from slime mould* (World Scientific).

## 参考文献 XII

-  Shirakawa, T. and Gunji, Y., 2010, Computation of Voronoi Diagram and Collision-free Path using the Plasmodium of Physarum polycephalum. International journal of unconventional Computing, 6.
-  Shirakawa, T., Yokoyama, K., Yamachiyo, M., Gunji, Y. and Miyake, Y., 2012, Multi-scaled adaptability in motility and pattern formation of the Physarum plasmodium. International Journal of Bio-Inspired Computation, 4, pp. 131-138.

## 参考文献 XIII

-  Whiting, J.G.H., de Lacy Costello, B.P.J. and Adamatzky, A., 2014, Towards slime mould chemical sensor: Mapping chemical inputs onto electrical potential dynamics of *Physarum Polycephalum*. *Sensors and Actuators B: Chemical*, 191, pp. 844-853.
-  Evangelidis, V., Tsompanas, M., Sirakoulis, G.C. and Adamatzky, A., 2015, Slime mould imitates development of Roman roads in the Balkans. *Journal of Archaeological Science: Reports*, 2, pp. 264-281.