

实验课 7：群体智能实验-蚁群算法

一、实验大纲

1. 群体智能理论背景
2. 蚁群算法原理
3. 蚁群算法实验

二、群体智能理论背景

群体智能这个概念来自对自然界中一些昆虫，如蚂蚁、蜜蜂等的观察。单只蚂蚁的智能并不高，它看起来不过是一段长着腿的神经节而已。几只蚂蚁凑到一起，就可以起往蚁穴搬运路上遇到的食物。如果是一群蚂蚁，它们就能协同工作，建起坚固、漂亮的巢穴，一起抵御危险，抚养后代。对于这些现象的一种解释是，群体中的每个个体都遵守一定的行为准则，当它们按照这些准则相互作用时就会表现出上述的复杂行为。这种群居性生物表现出来的智能行为被称为群体智能。



Craig Reynolds 在 1986 年提出一个仿真生物群体行为的模型 **BOID**。这是一个人工鸟系统，其中每只人工鸟被称为一个 **BOID**，它有三种行为：分离、列队及聚集，并且能够感知周围一定范围内其它 **BOID** 的飞行信息。**BOID** 根据该信息，结合其自身当前的飞行状态，并在那三条简单行为规则的指导下做出下一步的飞行决策。**Reynolds** 用计算机动画的形式展现了该系统的行为，每个 **BOID** 能够在快相撞时自动分开，遇到障碍物分开后又重新合拢。这实际上就是一种群体智能模型。

雷诺兹人工鸟规则：鸟飞翔时三条规则

- 第1条规则：不要过分靠近任何东西，包括其他的 **boid**；
- 第2条规则：尽量使自己的速度与周围其他 **boid** 的速度保持一致；
- 第3条规则：任何情况下都要朝着附近 **boid** 组成的集团中心靠近；

雷诺兹把一群 **boid** 放置在到处是墙壁和障碍物的屏幕环境里，**boid** 们飞了起来，逐渐聚成一群，时而又分成更小的团体，从障碍物两边绕过，又在另一端重新聚集。有一只 **boid** 逐渐离开了队列，单独朝向反方向飞去，但它很快意识到自己犯了错误，迅速从空中折回，划了一个圈后，重新回归队伍。另一只 **boid** 不知为何违反了程序规则，撞到柱子上，但很快挣扎着起来，迅速向鸟群的方向追赶过去。

尽管这一模型出现在 1986 年，但是群体智能（Swarm Intelligence）概念被正式提出的时间并不长。一个显著的标志是 1999 年由牛津大学出版社出版的 E Bonabeau 和 M Dorigo

等人编写的一本专著《群体智能：从自然到人工系统》（“Swarm Intelligence: From Natural to Artificial System”）。群体智能的研究越来越受到国际智能计算研究领域学者的关注，逐渐成为一个新的重要的研究方向。

Millonas M M 在 1994 年提出群体智能应该遵循五条基本原则，分别为：

- （1）邻近原则（Proximity Principle），群体能够进行简单的空间和时间计算；
- （2）品质原则（Quality Principle），群体能够响应环境中的品质因子；
- （3）多样性反应原则（Principle of Diverse Response），群体的行动范围不应该太窄；
- （4）稳定性原则（Stability Principle），群体不应在每次环境变化时都改变自身的行为；
- （5）自适应性原则（Adaptability Principle），在所需代价不太高的情况下，群体能够在适当的时候改变自身的行为。

这些原则说明实现群体智能的智能主体必须能够在环境中表现出自主性、反应性、学习性和自适应性等智能特性。但是，这并不代表群体中的每个个体都相当复杂，事实恰恰与此相反。就像单只蚂蚁智能不高一样，组成群体的每个个体都只具有简单的智能，它们通过相互之间的合作表现出复杂的智能行为。可以说，群体智能的核心是由众多简单个体组成的群体能够通过相互之间的简单合作来实现某一功能，完成某一任务。其中，“简单个体”是指单个个体只具有简单的能力或智能，而“简单合作”是指个体与其邻近的个体进行某种简单的直接通信或通过改变环境间接与其它个体通信，从而可以相互影响、协同动作。群体智能具有如下特点：

（1）控制是分布式的，不存在中心控制。因而它更能够适应当前网络环境下的工作状态，并且具有较强的鲁棒性，即不会由于某一个或几个个体出现故障而影响群体对整个问题的求解。

（2）群体中的每个个体都能够改变环境，这是个体之间间接通信的一种方式，这种方式被称为“激发工作”（Stigmergy）。由于群体智能可以通过非直接通信的方式进行信息的传输与合作，因而随着个体数目的增加，通信开销的增幅较小，因此，它具有较好的可扩充性。

（3）群体中每个个体的能力或遵循的行为规则非常简单，因而群体智能的实现比较方便，具有简单性的特点，。

（4）群体表现出来的复杂行为是通过简单个体的交互过程突现出来的智能（Emergent Intelligence），因此，群体具有自组织性。

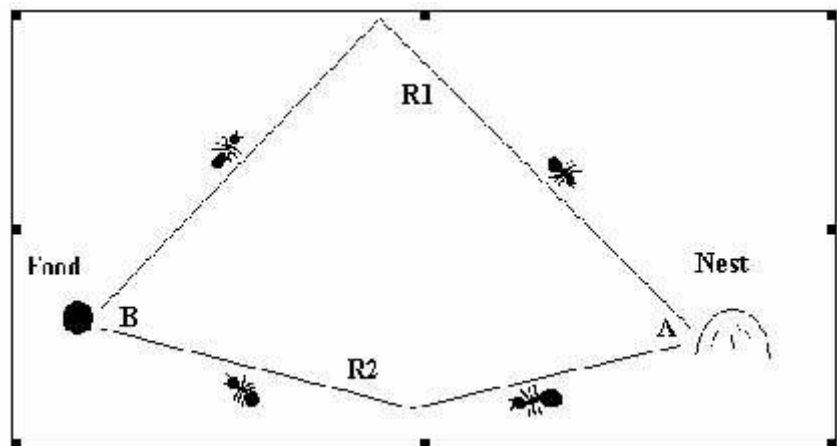
群体智能是通过模拟自然界生物群体行为来实现人工智能的一种方法，它强调个体行为的简单性，群体的涌现特性，以及自下而上的研究策略。群体智能在已有的应用领域中都表现出较好的寻优性能，因而引起了相关领域研究者的广泛关注。目前对群体智能的研究仍处于初级阶段，因此它具有很大的发展潜力，无论是对群体智能理论基础的研究，还是对其应用领域的拓展都有待进一步的深入。目前，群体智能主要有两种算法模式，分别是蚁群算法（Ant Colony System，简称 ACS）和微粒群优化算法（Particle Swarm Optimization，简称 PSO）。我们选择经典的蚁群算法作为实验对象，进行虚拟世界仿真实验。

三、蚁群算法原理

蚁群算法是由 M Dorigo 等人于 1991 年首先提出的，是受到自然界中蚂蚁群的社会性行为为启发而产生的，它模拟了实际蚁群寻找食物的过程。在自然界中，蚂蚁群总是能够找到从巢穴到食物源之间的一条最短路径。这是因为蚂蚁在运动过程中，能够在其所经过的路径上留下一一种被称之为“外激素（pheromone）”的物质。该物质能够被后来的蚂蚁感知到，并且会随时间逐渐挥发。每个蚂蚁根据路径上外激素的强度来指导自己的运动方向，并且倾向于

朝该物质强度高的方向移动。因此，如果在某一路径上走过的蚂蚁越多，则积累的外激素就越多，强度就越大，该路径在下一时间内被其它蚂蚁选中的概率就越大。由于在一定时间内，越短的路径会被越多的蚂蚁访问，所以随着上述过程的进行，整个蚁群最终会找到从蚁穴到食物之间的最短路径。蚁群算法正是利用了生物蚁群的这一特性来对问题进行求解。由于蚂蚁寻食的过程与旅行商问题（Traveling Salesman Problem，简称 TSP）的求解非常相似，所以蚁群算法最早的应用就是 TSP 问题的求解。目前，蚁群算法已在组合优化问题求解，以及电力、通信、化工、交通、机器人、冶金等多个领域中得到应用，都表现出了令人满意的性能。

下面给出一个蚁群算法原理：



1、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。

2、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。

3、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，并且每只蚂蚁多会以小概率犯错误，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。

4、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的运动下去，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，它会记住最近刚走过了哪些点，如果发现要走的下一点已经在最近走过了，它就会尽量避开。

5、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。

7、播撒信息素规则：

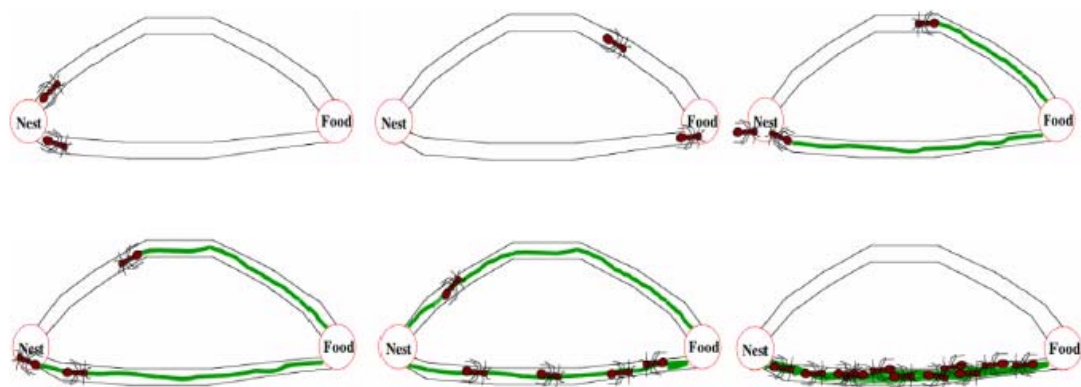
每只蚂蚁在刚找到食物或者窝的时候散发的信息素最多，并随着它走远的距离，播撒的信息素越来越少。

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。

在没有蚂蚁找到食物的时候，环境没有有用的信息素，那么蚂蚁为什么会相对有效的找到食物呢？这要归功于蚂蚁的移动规则，尤其是在没有信息素时候的移动规则。首先，它要能尽量保持某种惯性，这样使得蚂蚁尽量向前方移动（开始，这个前方是随机固定的一个方向），而不是原地无谓的打转或者震动；其次，蚂蚁要有一定的随机性，虽然有了固定的方向，但它也不能像粒子一样直线运动下去，而是有一个随机的干扰。这样就使得蚂蚁运动起来具有了一定的目的性，尽量保持原来的方向，但又有新的试探，尤其当碰到障碍物的时候它会立即改变方向，这可以看成一种选择的过程，也就是环境的障碍物让蚂蚁的某个方向正确，而其他方向则不对。这就解释了为什么单个蚂蚁在复杂的诸如迷宫的地图中仍然能找到隐蔽得很好的食物。

当然，在有一只蚂蚁找到了食物的时候，其他蚂蚁会沿着信息素很快找到食物的。

蚂蚁如何找到最短路径的？这一是要归功于信息素，另外要归功于环境，具体说是计算机时钟。信息素多的地方显然经过这里的蚂蚁会多，因而会有更多的蚂蚁聚集过来。假设有两条路从窝通向食物，开始的时候，走这两条路的蚂蚁数量同样多（或者较长的路上蚂蚁多，这也无关紧要）。当蚂蚁沿着一条路到达终点以后会马上返回来，这样，短的路蚂蚁来回一次的时间就短，这也意味着重复的频率就快，因而在单位时间里走过的蚂蚁数目就多，洒下的信息素自然也会多，自然会有更多的蚂蚁被吸引过来，从而洒下更多的信息素……；而长的路正相反，因此，越来越多地蚂蚁聚集到较短的路径上来，最短的路径就近似找到了。也许有人会问局部最短路径和全局最短路的问题，实际上蚂蚁逐渐接近全局最短路的，为什么呢？这源于蚂蚁会犯错误，也就是它会按照一定的概率不往信息素高的地方走而另辟蹊径，这可以理解为一种创新，这种创新如果能缩短路途，那么根据刚才叙述的原理，更多的蚂蚁会被吸引过来。



参考算法：ant.htm: 一个 javascript 蚁群算法示例。

四、蚁群算法实验

3.1 实验目的

本次实验为综合性实验，要求基于智能Agent感知-推理-行为机制，实验经典的蚁群算法仿真，实验基于信息素交互的群体智能。

3.2 蚂蚁觅食规则

1. Looking for food
 - * If pheromone trail is weak then wander
 - * Else move towards increasing concentration
2. Acquiring food
 - * If at food then
 - a. Pick it up
 - b. Turn around
 - c. Start laying pheromone trail
3. Returning to nest
 - * Deposit pheromone
 - * Decrease amount of food available
4. Depositing food
 - * If at nest then
 - a. Deposit food
 - b. Stop laying pheromone trail
 - c. Turn around
5. Repeat forever

3.3 实验步骤

在蚁群算法中，总共涉及到四类物体，蚁穴（home），食物（food），蚂蚁（ant）和信息素（pheromones，简称 phe），对它们分别进行如下的物理建模。

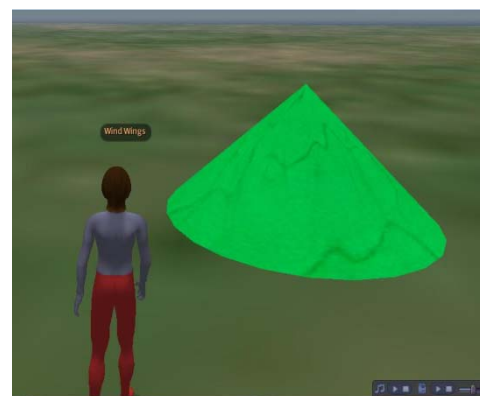
a) 蚁穴：home

蚁穴的功能：按照一定的速率释放一定数量的蚂蚁，蚂蚁以蚁穴为出发点开始沿随机方向寻找食物，在找到时候后直接返回蚁穴。

蚁穴所具有的属性及样例如下图所示：

属性	变量类型
位置	vector position <x,y,z>
体积	vector size <x,y,z>
质量	float weight
蚂蚁数量	integer number
释放间隔	integer time

蚁穴的属性及建模

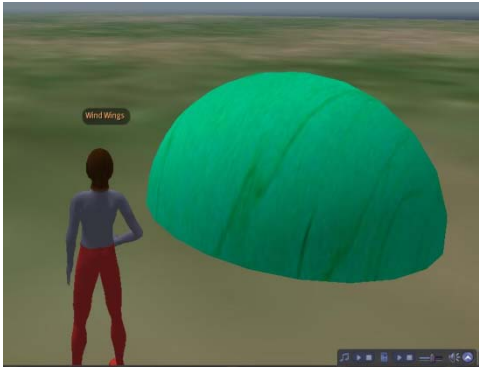


b) 食物：food

食物是蚂蚁所寻找的目标，并且可以感知到被蚂蚁取走食物的事件。每一堆食物有一个初始值，表示食物数量，每只蚂蚁每次取走一单位的食物，当该堆食物被蚂蚁完全取光后，将该堆食物从虚拟世界中删除。

食物所具有的属性及样例如下图所示：

属性	变量类型
位置	vector position <x,y,z>
体积	vector size <x,y,z>
质量	float weight
食物数量	integer number



食物的属性及建模

c) 蚂蚁

蚂蚁是一种智能 **Agent**，从蚁穴中释放。蚂蚁每次移动前进一定的步长。在蚂蚁觅食的过程中，共对周围局部环境中的两类物体进行搜索，食物和信息素。为了更逼真的模拟出蚂蚁觅食的过程，蚂蚁对食物和对信息素的搜索半径和搜索角度不同。蚂蚁对食物的搜索角度是 360 度，而对信息素的搜索角度为面前 180 度，即沿蚂蚁当前正方向左右各 90 度。蚂蚁对食物的搜索半径大于对信息素的搜索半径。

蚂蚁所具有的属性及样例如下图所示：

属性	变量类型
位置	vector position <x,y,z>
体积	vector size <x,y,z>
质量	float weight
步长	float step
旋转量	rotation rot <x,y,z,i>
搜索半径（食物）	float senseRange_food
搜索半径（信息素）	float senseRange_phe
搜索角度（食物）	float senseArc_food
搜索角度（信息素）	float senseArc_phe



蚂蚁的属性及建模

d) 信息素

当蚂蚁个体寻找到食物后，开始返回蚁穴，并在沿途释放信息素，作为帮助其它蚂蚁个体寻找食物的标记。信息素在被蚂蚁释放后，可以持续一定的时间，之后自动从虚拟世界中删除。

信息素所具有的属性如下表：

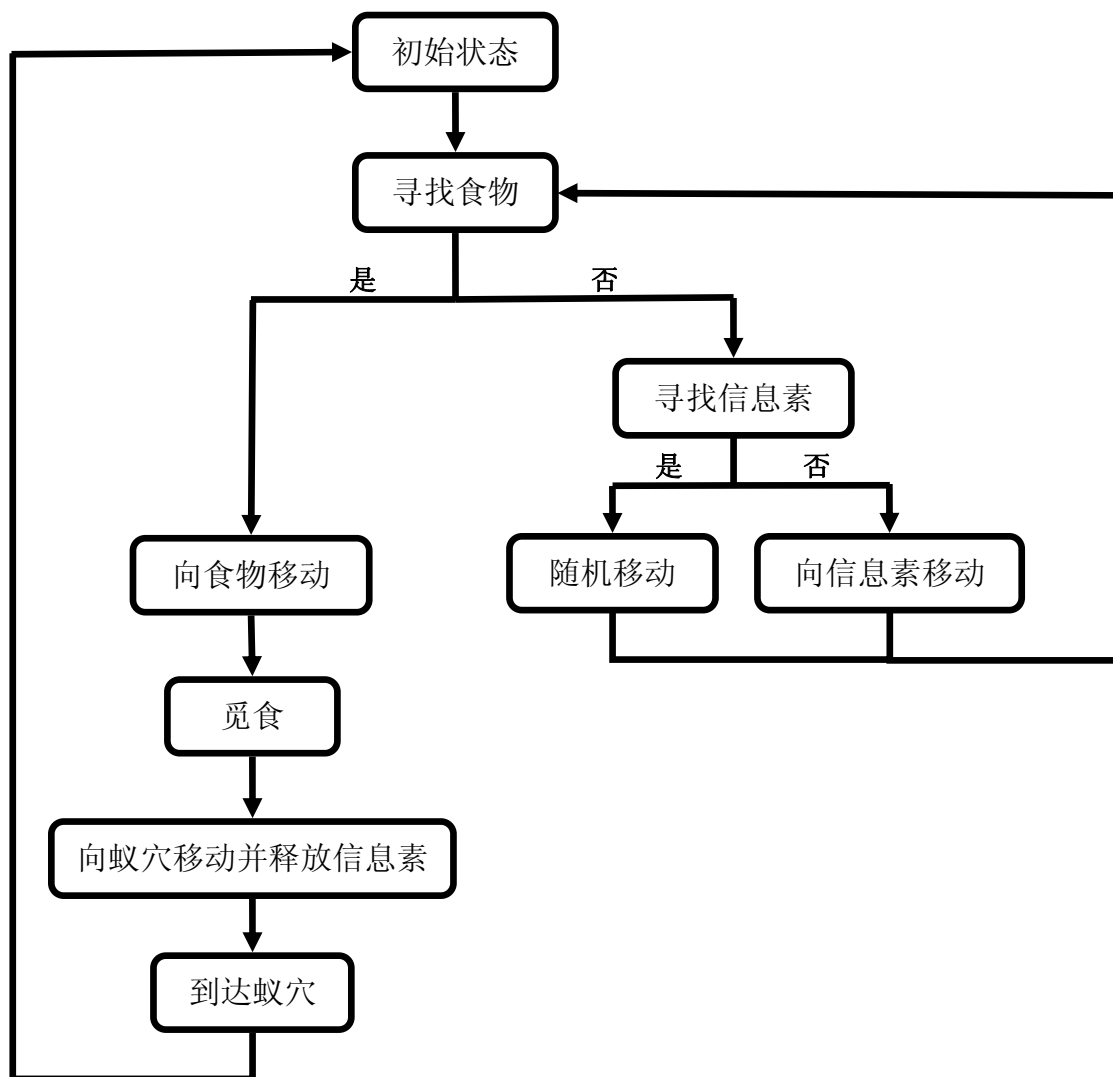
属性	变量类型
位置	vector position <x,y,z>
体积	vector position <x,y,z>
存活时间	float time



信息素的属性及建模

蚁群算法的函数实现

对于蚂蚁个体来说，一次完整的觅食过程用状态转换表示如下：



蚂蚁觅食过程的状态图

按照上面的状态图，设计蚁群算法的核心代码，共有 7 个状态 2 种主要函数

状态名/函数名	功能
default	初始状态
SEARCH_FOOD	寻找食物，在 senseRange_food 范围内，以 senseArc_food 角度寻找 food
SEARCH_PHE	寻找信息素，在 senseRange_phe 范围内，以 senseArc_phe 角度寻找 phe
MOVE_TO_FOOD	以步长 step 向 food 前进

MOVE_TO_PHE	以步长 step 向信息素浓度高的方向前进, 取所有探测到的信息素的位置的中心点作为前进方向
TAKE_FOOD	从 food 中取走单位体积的食物
BACK_HOME	沿直线返回 home , 并在沿途释放信息素
sensor	按照一定的角度, 一定的半径, 搜索一定的物体
random_move	在当前面向方向左右 90 度范围内随机选择一个方向, 前进长 step

示例代码:

