

用人工生命的方法模拟蜜蜂的采蜜行为

倪霞, 吴斌

(西南科技大学信息工程学院 四川 绵阳 621010)

【摘要】 通过对于人工生命理论及其“涌现”这一重要特征的研究,采用图形化的VB程序仿真了自然界中蜜蜂的采蜜行为。程序通过一些简单的规则模拟了蜂群中个体蜜蜂的采蜜行为,并实现了蜂群的整体行为。这一过程的实现可以清楚地展示人工生命的“涌现”特征。

关键词 人工生命; 涌现; 虚拟生物; 蜜蜂; 有限状态机
中图分类号 TP391.9 **文献标识码** A

Simulation the Behavior of Bees' Collecting Pollen with the Method of the Artificial Life

NI Xia, WU Bin

(School of Information Engineering, SWUST Sichuan Mianyang 621010)

Abstract The behavior of creature of nature is complicated and random, especially bee, a kind of colonial creature. However, the complicated structure of bee colony is generated by the interaction of single behavior of bee. By the research of the artificial life theory and some outstanding feature, the graphic VB program simulates the behavior of bee's collecting pollen. The program only simulates the behavior of individual bee's collecting pollen, but achieves the integral behavior of bee colony. Our research shows the important feature-Emergence-of the artificial life, that is, emergence.

Key words artificial life; emergence; virtual creature; bee; Finite-state machine

自古以来,传统的生物学都是研究以碳水化合物为物质基础的自然界的生命。而在1987年的第一次人工生命研讨会上,美国圣菲学院的兰顿教授提出了人工生命的概念,为人类对于生命的研究开辟了一条新的道路。人工生命是致力于通过试图在生物学现象中抽取基本的动力学原理,并把这些原理用到其他的物理媒体上,如计算机,使它们可以用于新品种的实验操作和测试来理解生命的研究领域^[1]。本文则在人工生命研究的基础上,采用图形化的VB程序模拟了蜜蜂的采蜜行为。

1 蜜蜂的采蜜行为

蜜蜂的个体行为是简单的,而且看似杂乱无章的,但是,蜜蜂的群体结构却是复杂的,而且是有条不紊的。3种类型的蜜蜂,蜂王,工蜂和雄峰个体交互作用构成了复杂的蜂群结构。蜂巢里的3种蜜蜂,各司其职,分工合作。蜂巢中大于21 d蜂龄的工蜂负责蜂巢中花粉的收集。在一般情况下,野外的工蜂总是在一定的范围内采蜜,而且主要是从一种植物的花上采蜜。由于采蜜经验的不同,它们的采蜜速度和采蜜方法存在着明显的个体差异。虽然工蜂个体的采蜜行为常常趋于特殊化,但作为一个整体的蜂群却能够对资源的变化做出迅速的反应,它可以调动它的大部分成员到一种报偿最高的植物花上去采蜜,这样能有效地利用集中和分散的食物资源,极大地提高了采蜜的效益^[2]。一般来说,工蜂采蜜选择量多、质优的先采集,这是普遍规律,采集的飞行距离则放在第二位^[3]。这里假定把蜜源的距离作为工蜂采蜜的条件。工蜂飞出蜂巢,搜索最近的蜜源,然后去采集花粉,接着回到蜂巢,抖落收集到的花粉,然后再飞去收集更多的花粉。这种日常事务一直要持续到工蜂达到自然的6周的生命期,然后死亡。

收稿日期: 2003-09-12

基金项目: 四川青年科技基金资助项目(03ZQ026-035)

作者简介: 倪霞(1977-),女,硕士,主要从事人工生命领域方面的研究。

2 设计所基于的理论基础

自然界的生物形态各异,行为多样,特别是群体生物的行为更是让人迷惑。人们在计算机上建立了各种各样的虚拟生物使其具有自然界生物的行为,通过研究这些虚拟的生物来抽取生物现象的共同特征,以揭示生命的本质。其中,突现便是人工生命的一个突出特征。“突现”一词用来指称在复杂的(非线性的)形态中许多相对简单单元彼此相互作用时产生出来的显著整体特性^[1]。人工生命并不像人们在设计汽车或机器人那样在平庸的意义上是预先设计好的。人工生命采用至下而上编程的信息处理原则来进行:在底层定义许多小的单元和几条关系到它们内部的、完全是局部的相互作用的简单规则,从这种相互作用中产生出连贯的“全体”行为,这种行为不是根据特殊规则预先编好的,而是通过局部的简单规则的相互作用突现出来的^[4]。

一个非常典型的突现的例子是在第一次人工生命研讨会上,由美国洛杉矶一家电脑公司的制图专家 Reynolds展示的一个被称为“boids”的鸟群。他采用了3条基本规则,以及目标搜寻和躲避障碍物的规则,就使得一群“鸟”聚集在一起飞行。而实际上Reynolds并没有在程序中写入任何“聚集成群”的规则,也从未给任何一只“boid”下达这类有“理智”的指令,而是这些“boid”们自发的倾向,是那些简单的规则的交互作用而产生了类似于生物群体的“突变”行为^[5]。本文就是试图通过一些规则模拟个体蜜蜂的单一的行为,而实现当许多蜜蜂的交互作用时而产生蜜蜂的群体行为。主要针对蜜蜂的采蜜行为,着重于蜜蜂采蜜的规则,如:当蜜蜂达到21d蜂龄时就出去采蜜,否则就呆在蜂巢周围;蜜蜂搜索离自己最近的花朵并朝此飞去;当蜜蜂携带了花粉后就立即飞回蜂巢等。在这些规则的相互作用下,当程序运行的时候就可以看到类似于自然界中蜂群的采蜜行为,但实际上程序并未涉及到蜂群行为的规则,而只是个体蜜蜂行为的规则的交互。

3 仿真程序

3.1 程序的组织结构

仿真程序图形化地展示了一个“草地”环境。“草地”中包括了树木、花朵、蜜蜂(工蜂和蜂王)和蜂巢等对象,所有的对象都被创建一个分类的结构。为了利用这些分类结构,仿真程序选择了一个面向对象的和图象化的编程方式。程序包括了一个主窗体程序,副窗体程序,模块管理器和3个类模块。结构如图1所示。图1中的资源类模块、蜜蜂类模块及蜂巢类模块分别用来定义花朵、树木、工蜂和蜂巢的参数,如大小、位置和色彩等。模块管理器则根据3个类模块所定义的参数来创建草地环境中的对象。对于一个仿真程序创建的树木对象的数量是静态的,所以资源管理器只需要在仿真开始的时候创建树木对象。花朵的创建是根据环境的变化而动态创建的,仿真程序开始运行时,会有一个初始的花朵数量,随着蜜蜂的采集,花朵减少,当低于设定的最低值时,就开始创建新的花朵,直到重新达到最大值,才停止创建。而花朵的颜色,大小和位置都是随机确定的。每朵花的花粉数量是以副窗体初始的单位花粉值为基数的一个随机值,而且不受花朵大小的约束。每次仿真开始,模块管理器随机地创建蜂巢的位置、大小、颜色和蜂巢花粉数。个体蜜蜂的初始位置是由蜂巢确定的,初始蜜蜂的年龄是随机的,新生蜜蜂的年龄是从0开始的。

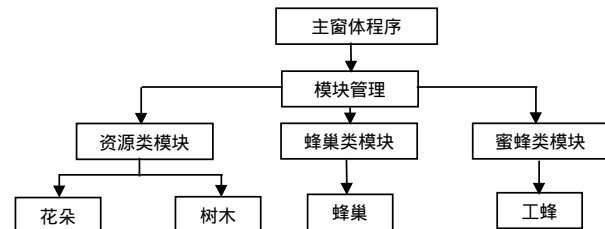


图1 程序分层结构

对象。对于一个仿真程序创建的树木对象的数量是静态的,所以资源管理器只需要在仿真开始的时候创建树木对象。花朵的创建是根据环境的变化而动态创建的,仿真程序开始运行时,会有一个初始的花朵数量,随着蜜蜂的采集,花朵减少,当低于设定的最低值时,就开始创建新的花朵,直到重新达到最大值,才停止创建。而花朵的颜色,大小和位置都是随机确定的。每朵花的花粉数量是以副窗体初始的单位花粉值为基数的一个随机值,而且不受花朵大小的约束。每次仿真开始,模块管理器随机地创建蜂巢的位置、大小、颜色和蜂巢花粉数。个体蜜蜂的初始位置是由蜂巢确定的,初始蜜蜂的年龄是随机的,新生蜜蜂的年龄是从0开始的。

主窗体程序对各对象进行更新和维护,它还要维持程序的时间,以及对蜜蜂移动的实现。它不断地访问模块管理器直到仿真被停止。第一次对模块管理器的访问就初始化草地的创建。之后,随着时钟的增加主窗体程序会根据当前资源类模块、蜂巢类模块及蜜蜂类模块的参数状态在每个时钟标记处更新蜜蜂及草地,在更新对象的同时,主窗体还记录着环境中参数的变化。

3.2 程序的实现

几个假定简化了和限制了蜜蜂仿真的范围。在草地中总是在春天的季节,因此不会有环境的变化对蜜

蜂采蜜行为的影响。同样地,也没有捕食者和其他蜂群的竞争,所以蜜蜂的死亡仅仅是因为饥饿和老龄。由于本仿真是为了实现工蜂的采蜜行为,所以忽略了雄蜂的存在,而建立一个蜂王,假定它以每天两个的数量一直在产卵,以此给蜂群数量不断的补充,使这个人工蜂群能存活下去,不会因为蜜蜂的死亡,而使蜂巢消亡。但蜂王的其他一切行为是不可见的,即被忽略。它只在整个蜂巢灭亡的时候才死亡。

对于仿真除了给一些假定外,仿真的运行还设定了一些参数:程序运行周期为 $10 \text{ s} \times 24 = 1 \text{ d}$ (系统时间10 s为仿真程序运行的1 h);草地的大小(蜜蜂活动区域的大小)为 $7\,500 \times 7\,500$;蜜蜂个数的初始值为100(可随机设定);最低系统报警花粉数量为3 000;最高系统报警花粉数量为7 500;蜂巢中最低报警花粉数量为250。工蜂的行为可以用有限状态来描述,如图2所示。在图2中,21 d蜂龄以上的工蜂的日常行为只在“采集花粉”,“回巢”和“吃”的状态间转变。直到到达它生命的终点。图中两个状态间的为状态发生转变的条件。当工蜂蜂龄达到6周即42 d时,就会自动从它当前所处的状态转变为死亡状态。21 d以下蜂龄的工蜂则只处在“吃”的状态,而让它发生状态

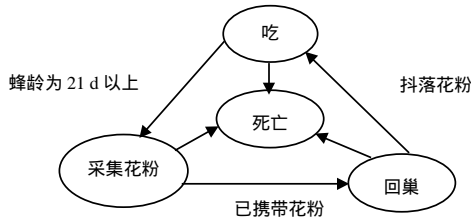


图2 21 d蜂龄以上工蜂的状态图

转变的条件是当系统中蜂巢花粉数量低于报警值时,所有工蜂都会飞去“采集花粉”。当蜂巢中花粉的数量再次超过报警值,这时小于21 d蜂龄的工蜂就返回到巢中。下面的两段程序实现了蜜蜂采集花粉和收集到花粉后飞回蜂巢的过程。

```

.....
If b.rate Then
hX = Hives(b.Hive).X
hY = Hives(b.Hive).Y
If Sqr((b.Y-hY) ^ 2 + (b.X-hX) ^ 2) < Hives(b.Hive).R
Then
    b.rate = False
    Hives(b.Hive).iRsum = Hives(b.Hive).iRsum + 1
Else
    S = Sqr((b.Y-hY) ^ 2 + (b.X-hX) ^ 2)
    a.Y = b.Y
    a.X = b.X
    b.Y = b.Y + (hY-b.Y) * 7 / S
    b.X = b.X + (hX-b.X) * 7 / S

```

```

.....
flag = True: i = Tsum + 1: S = 10 000
Do While flag And i <= Fsum + Tsum
    If Sqr((FTs(i).X-b.X) ^ 2 + (FTs(i).Y-b.Y) ^ 2) <
    FTs(i).R Then
        flag = False: j = i: Exit Do
        If Sqr((FTs(i).X-b.X) ^ 2 + (FTs(i).Y-b.Y) ^ 2) < S
        Then
            S = Sqr((FTs(i).X-b.X) ^ 2 + (FTs(i).Y-b.Y) ^ 2)
            j = i: i = i + 1
        Loop
        / 蜜蜂搜索离它最近的花朵
        If flag Then
            a.Y = b.Y: a.X = b.X
            b.Y = b.Y + (FTs(j).Y-b.Y) * 7 / S

```

4 仿真结果及分析



图3 仿真运行的图形化展示

启动仿真程序,出现一个初始化窗口,设定初始的蜂巢数、蜜蜂数、花朵数及单位花朵花粉数等。不同的初始值会产生不同的运行结果。点击“开始”进入仿真程序的主窗体。主窗体记录着仿真运行的各参数变化,通过记录这些数据的变化,可看到数据与数据间的一些关系,而这正是与自然界中的规律相吻合。“图形表示”可将仿真切换到图形显示的方式,可形象的展示蜜蜂采蜜的过程,如图3所示。

在一个正方形的区域中,可以看到盛开着五颜六色的花,蜂巢及蜂群。随着时间的变化,可以看到一些工蜂向它附近的花移去,这些是蜂龄大于21 d的工蜂,它们分为几路朝着离它最近的花朵移动,有条不紊,而另一些仍留在蜂巢的周围,这些则是小于21 d蜂龄的工蜂。当工蜂到达花朵上时,可以看

到系统的花粉数量在减少,工蜂携带了花粉,当携带有花粉的工蜂回到巢中时,则蜂巢的花粉数量增加。在程序运行的过程中,还可以看到,系统中花朵的数量在减少,这是由于工蜂的采集,当某一花朵花粉数量减少到0时,则该花朵被从系统中去掉。当采蜜的工蜂飞向的那朵花消失掉时,可以看到一个有趣的现象,这些工蜂在“犹豫”一会后,便自动改变了方向朝着离它最近的另一朵花飞去,而在程序中并没指定蜜蜂朝着什么具体方向移动,而只是将寻找最近的花朵并向它“飞去”作为移动的规则,当这个规则与花朵的消失相互作用时,就产生了蜜蜂的这种自发的行动。如果蜂巢周围的花朵逐渐减少,那么工蜂就需花更多的时间飞去更远的地方采蜜,这样花粉的供应量可能就跟不上消耗量,则蜂巢中的花粉数就达到了最低报警值,这时可以看到蜂巢中所有的蜜蜂,不管是21 d以上蜂龄的,还是21 d以下蜂龄的都倾巢而出,去采集花粉,当蜂巢的花粉再次达到报警值以上时,那些21 d蜂龄以下的蜜蜂才停止采蜜的行为,回到蜂巢的周围。由于该系统采取随机的方式设定初始值,所以每次运行都可能发生不同的情况。下面是其中一次运行所记录的部分数据。

从表1中可以看到环境中的花粉数量随着时间的增加逐渐减少,这是由于蜜蜂的采集,同时,当环境的花粉数减少时,蜂巢中的花粉数也成正比地在减少,这是因为程序采用蜜蜂首先搜索最近的花朵采集的规则,因此环境中花朵的减少必然从离蜂巢

最近的地方开始,当花朵逐渐减少时,工蜂就需要花更多的时间到更远的地方采蜜。因此花粉采集的效率降低,采集的花粉数小于了消耗的数量,蜂巢中的花粉数就越来越少。这正符合了自然界中蜜蜂采蜜的规律,当蜜蜂所处环境的花粉资源不充足时,那么相应的蜂巢中的花粉数量也比当蜜蜂处在花粉资源充足的环境中时蜂巢的花粉数量要少。当环境的花粉数量逐渐增加时,蜂巢中的花粉数量又有所回升,但是增加得并不快,这是因为受到了蜜蜂数量的影响,蜜蜂的数量在减少,因此采蜜的工蜂数量也在减少,就影响了采蜜的数量。由于系统采用固定每天增加两只蜜蜂的方式来补充蜜蜂,因此蜜蜂的数量会最终趋于一个稳定的数值,这显然与自然界中的情况不同,这是因为程序简化了一些因素所造成的,当然,在以后的程序改进中会考虑到这方面的因素,以使得程序更加的完善。

表1 仿真过程其中一次的数据

天数/d	蜂巢花粉数	环境花粉数	花朵数	蜜蜂个数
第1	526	3 869	100	102
第9	450	3 518	91	98
第17	317	3 277	86	100
第25	252	3 307	89	96
第33	218	5 456	147	94
第42	266	7 364	197	82

注:仿真初始值,蜂巢位置(x, y)(3 967, 5 279) 环境花粉数量: 3 914
蜂巢花粉数: 531 成员数量: 100 蜂王数量: 1

5 结 论

总的来说,设计实现了蜜蜂采蜜行为的人工生命仿真。蜜蜂个体单一行为之间的交互作用构成了蜂群复杂的结构。但是,程序还可以继续改进和扩展。下一步的工作则是加入3种蜜蜂的各个行为阶段;可以将系统环境模拟的更真切,加入季节的变化、温度的变化,来观察蜜蜂在这些环境变化下的行为,使其更接近自然界中的蜜蜂世界;还可以改进用户界面,加入一些统计表,使其对于系统中的一些数据统计更完整。更进一步,还可以将遗传算法用于蜜蜂的创建,使它在一代一代的进化中,形成强大的群体。

参 考 文 献

- [1] Sipper M. An introduction to artificial[EB/OL]. http://www.cs.unibo.it/babaoglu/courses/cas/tutorials/Artificial_Life.pdf, 2000-03-06.
- [2] 昆友. 蜜蜂的舞蹈语言[EB/OL]. <http://www.bjcp.gov.cn/dwly/dwly/k0733-05.htm>, 2003-1-15.
- [3] 钱宗柏. 谈蜜蜂的采蜜习性及其他[J]. 养蜂科技, 2000(4): 17-19.
- [4] 李建会. 人工生命探索新的生命形式[J]. 自然辩证法研究, 2001, 17(7): 1-5, 20.
- [5] Craig W. Reynolds. Steering behaviors for autonomous characters[EB/OL]. http://act-r.psy.cmu.edu/people/douglass/Agents/PDF/Steering_Behaviors.pdf, 2003-05-04.