

# 一种具有演化规则的元胞遗传算法

鲁宇明<sup>1,2</sup>, 黎明<sup>1,2</sup>, 李 凌<sup>1</sup>

(1. 南京航空航天大学自动化学院, 江苏南京 210016; 2. 南昌航空大学无损检测教育部重点实验室, 江西南昌 330063)

**摘 要:** 本文根据元胞个体密度与分布的演化规则, 考虑整个空间元胞个体动态的相互作用, 从更为真实模拟自然界的角度出发, 提出了具有演化规则的元胞遗传算法(CEGA), 并得到了算法中元胞演化选取准则. 通过复杂典型测试函数的仿真实验结果表明, 该算法较一般元胞遗传算法和一般遗传算法具有更好地维持群体多样性的能力, 以利于有效地进行全局探索, 逃出局部优, 并具有更快的寻优速度. 通过采用不同演化规则实验比较, 得出中等密度分布的元胞个体有利于求解问题.

**关键词:** 元胞遗传算法; 元胞自动机; 演化规则; 多样性

**中图分类号:** TP38 **文献标识码:** A **文章编号:** 0372-2112 (2010) 07-1603-05

## The Cellular Genetic Algorithm with Evolutionary Rule

LU Yu-ming<sup>1,2</sup>, LI Ming<sup>1,2</sup>, LI Ling<sup>1</sup>

(1. College of Automation, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, China;

2. Key Laboratory of Nondestructive Test, Nanchang Hangkong University, Nanchang, Jiangxi 330063, China)

**Abstract:** In order to mimic the nature more true, according to density and distribution of cell, the dynamic influence among all individual with each other were considered during evolutionary operation. Therefore, a dynamic environment is built by the rules of the cellular automata and in turn, the cellular genetic algorithm with evolutionary rules (CEGA) is proposed in this paper. The experimental results of two typical test functions show that CEGA can efficiently remain the diversity of population, carry out the global exploitation, and improve the speed of searching solution. We conclude that moderate density cell is favourable to solution.

**Key words:** cellular genetic algorithms; cellular automata; evolutionary rule; diversity

### 1 引言

遗传算法是建立在随机搜索技术的一种优化算法, 不依赖于问题的具体领域知识, 具有很强的鲁棒性. 在函数优化、模式识别、图像处理、人工智能等方面得到广泛应用. 但传统的遗传算法存在收敛速度慢、全局收敛性能较差等缺点, 使问题求解易陷入局部最优, 因而不断有对遗传算法的改进. 许多研究者将云模型、免疫机制与遗传算法结合形成新型遗传算法以改善算法性能, 取得一定效果<sup>[1,2]</sup>. 除此之外, 近来一些实验表明<sup>[3~5]</sup>: 具有空间群体结构的 GA 不仅易于映射到并行计算, 而且能得到较好质量的解. 而传统的遗传算法忽略了在空间结构环境和物种进化过程中局部群体的复杂交互作用, 表现在种群不但不会受到自然扰动, 种群中的个体也不会发生变化, 如死亡, 复活或者迁移, 这与实际的生物进化过程不符, 在众多具有空间结构的 GA 中, 元胞自动机正成为研究遗传算法的一个新的工具.

元胞遗传算法(CGA)是一种基于空间结构群体分散的遗传算法, 进化个体分布在一个空间网格中, 个体的遗传操作(主要是选择操作)限制在相邻个体之间进行, 是一种新的遗传算法模式<sup>[3]</sup>. 最初是为在大规模并行机上运行遗传算法而设计的, 但序列式实施并行遗传算法, 也可克服经典遗传算法性能上的不足, 从而提高算法的效率. 但是目前研究大都只是借助元胞自动机模型的邻居结构实现遗传算法的操作, 而对元胞自动机机理在遗传算法中的作用研究不多. 目前国内在这方面的研究很少, 国外陆续有一些关于其研究的报道. 文献[3]通过邻居概念, 较好地平衡了探索/开发这一相对矛盾的遗传现象, 对参数设置要求低, 相对与其它算法具有简单性, 但不具通用性. 文献[5]将个体分布在网格中, 遗传操作在每个邻域内进行, 修改传统 GA 的复制和选择机制, 在比例复制时增加一个自适应接受阈值, 这种选择方法对有些优化问题可加快求解速度, 但不适合比较复杂的多模化问题. 文献[6,7]分析了空间拓扑形状

的改变对算法性能的影响,通过定性研究得出:通过改变网格形状重新分布个体,可简单轻易地提高群体多样性.朱刚<sup>[8]</sup>等将蚂蚁算法和元胞自动机结合,提出了一种元胞蚂蚁算法及其数学描述,用于求解带约束函数的优化问题.虽然文献[9,10]用元胞自动机模拟生态景观,个体分布在这个模拟景观中,并随机分配一定比例的“生”“死”状态,个体的遗传操作只在“活”个体邻域范围内进行,将灾难引入群体以改变模拟景观中个体之间的连通性,使问题解的质量得到改善,在一些特别问题上优于传统的遗传算法.

这些研究中元胞自动机的状态是处于静止情况,在遗传算法中的遗传操作借助元胞自动机的邻居结构来解决算法的遗传算子的操作,而忽略了自然界的演化的动态过程.文献[8,9]中灾难的发生是一个人为给予的干扰,而现实生态系统中,个体的状态是一个自适应变化的过程,系统中的多样性和复杂性的涨落也是渐变过程,元胞自动机可以清楚完整地模拟复杂的自然环境.为了更为真实的模拟自然界状况,本文通过元胞自动机对生命现象的模拟,将元胞自动机演化嵌入遗传算法中,提出了一种新的具有演化规则的元胞遗传算法(CEGA).通过一些典型函数优化问题分析CEGA算法对群体多样性影响以及解的质量,并研究了不同演化规则对算法性能的影响.

## 2 具有演化规则的元胞遗传算法

元胞遗传算法(Cellular Genetic Algorithm,简称CGA),是将元胞自动机的作用机理与遗传操作相结合,是遗传算法的一个分支.个体按一定的拓扑形状分布(主要是二维网格),所有个体只与相邻个体进行相互作用(主要邻居类型见图1),遗传操作限制在邻域范围内,在一定程度上降低了选择压力,使得元胞遗传算法成为解决复杂问题的一种有效方法.

生态系统中的生物相互作用并非时时刻刻都在发生,这个时刻处于静止状态的生物个体,下一时刻处于激活状态,个体间的相互作用是一种动态过程,同时个体的生存密度影响生态系统的进化.而元胞自动机具有重复简单的演化规则导致复杂的系统行为的能力,并且不同的演化规则能够模拟生态系统中个体不同的生存状态,以及种群个体的局部进化,这在某种意义上

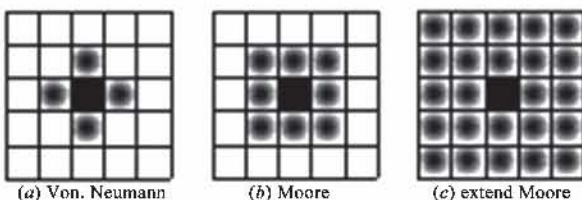


图1 元胞遗传算法的空间形状及其邻居分布

可更好地模拟自然进化智能,进一步提高群体基因多样性,提高进化收敛速度.按照这一思想,我们提出了一种具有演化规则的元胞遗传算法.下面从元胞自动机的原理及算法基本框架两方面详细说明算法.

### 2.1 元胞自动机机理及演化规则

Von. Neumann<sup>[11]</sup>最早提出了元胞自动机(Cellular Automata,简称CA).它是一个时间和空间都离散的动力系统,由元胞、状态、邻居和规则四部分组成.按一定拓扑结构分布的元胞取有限的离散状态,依据确定的局部规则进行更新,大量元胞通过简单的相互作用而构成动态系统的演化.

例如,Langton用元胞自动机仅通过3条简单规则就模拟了鸟类飞翔的整体行为,应用最广的“生命游戏”规则,近似地描述了生物群体的生存繁殖规律.元胞自动机中采用不同的规则能够产生各种不同的演化模式,从而实现模拟生态系统的不同演化过程.

沃尔弗拉姆(S. Wolfram)<sup>[10]</sup>在详细分析研究了一维元胞自动机的演化行为,并在大量计算机仿真的基础上,将所有元胞自动机的演化行为归纳为四类:(1)平稳型(homogeneous).自任何初始状态开始,经过一定时间演化和若干步运算后便停留在一个固定的状态;(2)周期型(periodic).经过一定时间演化后,在几种状态之间周期循环;(3)混沌型(chaos).自任何初始状态开始,经过一定时间演化后,处于一种完全无序随机的状态,几乎找不到任何规律;(4)复杂型(edge of chaos).在演化过程中可能产生复杂的结构.这种结构既不是完全的随机混乱,又没有固定的周期和状态.

王仲君等<sup>[12]</sup>通过对元胞自动机的一、二、三维演化行为的研究,将元胞自动机的演化行为动态统计图与沃尔弗拉姆对于元胞自动机的分类对应起来研究.从

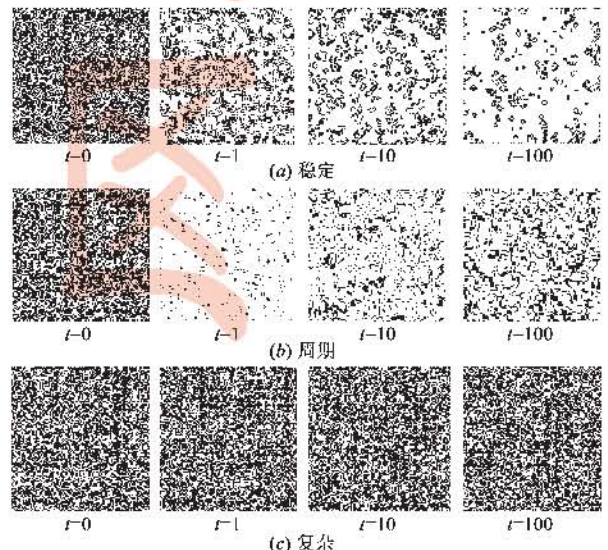


图2 元胞演化状态

计算机模拟演化结果可以看到,不同的演化规则导致元胞会产生不同的生存结果,随着演化时间推移,元胞自动机的元胞状态分别处于稳定、周期、复杂,如图 2 表示了三种状态初始状态以及随时间演化的情况。

## 2.2 算法模型

本算法将个体分布在  $n \times n$  的空间网格中,设元胞的“生”、“死”两种状态分别对应“1”、“0”,并随机赋给  $n \times n$  个个体;邻居定义为 more 型(图 1(b)).选取“生”状态元胞与邻居元胞进行遗传操作,而元胞下一时刻“生”、“死”状态的确定,通过一定的演化规则,由本时刻及该元胞的所有邻居元胞的状态所决定。

演化规则的基本模型如下:

$$\text{若 } S^t = 1, \text{ 则 } S^{t+1} = \begin{cases} 1, E_b \leq S \leq F_b \\ 0, S < E_k \text{ 或 } S > F_k \end{cases} \quad (1)$$

式中,  $S^t$  表示  $t$  时刻元胞状态,  $S^{t+1}$  表示  $t+1$  时刻元胞状态;  $E_b$ 、 $F_b$  表示对于一个“活”元胞,在下一个时刻,继续保持其状态所需的“活”邻居的数目,邻居中“活”元胞数目过少,缺乏繁殖机会,邻居中“活”元胞数目过多,竞争加剧,出现生存危机,都会导致“活”元胞的死亡。而  $E_k$ 、 $F_k$  则表示对于一个“死”元胞,在下一时刻,“复活”所需要的“活”邻居的数目,只有适当的“活”元胞数量,才能营造一个“死”元胞复活的周围环境。  $E_b$  和  $F_b$  则分别表示在上述情况的上限值。

这种模型可以形象地体现环境与生存的制约关系,如在生命密度过小时,由于孤单,缺乏配种繁殖机会,缺乏互助也会出现生命危机,个体由“生”变“死”;在生命密度过大时,由于环境恶化,资源短缺以及相互竞争也会出现生存危机,个体由“生”变“死”;只有处于个体数目适中的位置,生物才能生存和繁衍后代,个体由“死”变“生”。这样才能够模拟生命活动中的生存、灭绝、竞争等复杂现象,演化规则是自然现象模拟的关键。

演化规则不同,元胞个体在空间的生存密度不同,个体之间的相互制约关系也受到影响,从而影响算法性能。根据模型(1),从生存密度来考虑演化规则,通过多次测试,采用以下几种规则模拟生命系统的稳定、周期、复杂状态。

$$\text{规则 1: } \begin{cases} \text{若 } S^t = 1, \text{ 则 } S^{t+1} = \begin{cases} 1, S = 2, 3 \\ 0, S \neq 2, 3 \end{cases} \\ \text{若 } S^t = 0, \text{ 则 } S^{t+1} = \begin{cases} 1, S = 3 \\ 0, S \neq 3 \end{cases} \end{cases}$$

$$\text{规则 2: } \begin{cases} \text{若 } S^t = 1, \text{ 则 } S^{t+1} = \begin{cases} 1, S = 1, 2, 3, 4 \\ 0, S \neq 1, 2, 3, 4 \end{cases} \\ \text{若 } S^t = 0, \text{ 则 } S^{t+1} = \begin{cases} 1, S = 4, 5, 6, 7 \\ 0, S \neq 4, 5, 6, 7 \end{cases} \end{cases}$$

$$\text{规则 3: } \begin{cases} \text{若 } S^t = 1, \text{ 则 } S^{t+1} = \begin{cases} 1, S = 2, 4, 6, 8 \\ 0, S \neq 2, 4, 6, 8 \end{cases} \\ \text{若 } S^t = 0, \text{ 则 } S^{t+1} = \begin{cases} 1, S = 1, 3, 5, 7 \\ 0, S \neq 1, 3, 5, 7 \end{cases} \end{cases}$$

个体周围的个体存活状态对中心元胞的生死状态有着直接影响,规则 1 中,当生个体周围存在 2,3 给活个体,个体间的竞争和资源处于中等水平,这是应用最广的“生命游戏”规则,近似地描述了生物群体的生存繁殖规律,但在一定时间后系统会处于稳定,活元胞数目稳定在 15% 左右。规则 2 模拟一种周期变化的生存状态,活元胞数目稳定在 51% 左右。规则 3 采用了一种奇偶规则,使生物繁殖呈现出杂乱性,活元胞数目稳定在 80% 左右。算法流程如下:

**Step1** 初始化  $n \times n$  个个体,将其分布在  $n \times n$  网格中作为初始元胞,并随机确定元胞“生”“死”状态。

**Step2** 选择状态为“生”的元胞作为中心元胞,以联赛方式从中心元胞的邻居元胞中选择一个元胞与中心元胞进行交叉操作,得到新的个体。

**Step3** 将 Step2 产生的新个体进行变异,将变异后的个体与中心元胞个体进行比较,若变异后的个体优于中心元胞个体,此个体替代中心元胞,否则不替代,保留中心元胞个体。

**Step4** 所有元胞状态按照给定的演化规则 1-3 同时进行更新,得到下一时刻元胞状态。

**Step5** 转到步骤 Step2,直到终止条件满足。

## 3 实验及算法性能分析

### 3.1 测试函数

本节利用 2 个较难优化的典型测试函数,其中  $F1$  函数是二维,有一个全局最大值点,最大值为 1.1511,附近有次优点,值为 1.1284,距离极优点很近,大多算法极易落入次优点,造成早熟。 $F2$  函数是高维函数,具有无穷多个极值点,随着维数增加,变量间制约作用增大,难以优化,常用于评价算法,维数定为 30。如表 1 所示。为了评价本算法,本文将涉及到的参数做了统一设定,群体规模均为 400,函数进化代数为 10000;遗传操作参数设定:交叉率为 0.9,变异率为 0.05;每种情况算法运行 50 次。

表 1 被测试函数

问题	函数表达式
$F1_{\max}$	$f(x, y) = \frac{\sin(\sqrt{(x-50)^2 + (y-50)^2})}{\sqrt{(x-50)^2 + (y-50)^2}} + 2.71828$
	维数:2, 最优值:1.1511, 变量范围:[0, 100]
$F2_{\min}$	$f(x_i) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
	维数:30, 最优值:0, 变量范围:[-10, 10]

### 3.2 与其它遗传算法的比较

受到生态学中空间交互作用对生物进化的影响的启发,为了更为真实模拟现实世界,我们在个体相互作用的过程中,通过引入演化规则实现动态作用过程.为了验证算法的可行性和有效性,从个体多样性、寻优时间、精度等方面,我们将 CEGA 与标准元胞遗传算法(CGA)、及其简单遗传算法(SGA)进行比较研究.

一方面,从个体分布情况对二维函数  $F1$  进行研究, $F1$  函数是局部最优点环绕全局最优点的函数.图 3 是函数  $F1$  在算法终止时个体分布情况. SGA 算法中不同个体数量极少,并且几乎没有在全局最优附近. CGA 算法中,个体大部分分布在局部极值点处,经统计,在算法终止时, $F3$  函数有 359 个个体的目标函数值为局部极值 1.1284,在全局极值和局部极值外几乎没有个体分布,而 CEGA 算法有 158 个个体的目标函数值为局部极值 1.1284,少量在全局最优附近外,搜索空间仍有大量个体存在.另外, $F1$  函数在 50 次实验中,SGA 算法只能收敛到次优解 1.1284,CGA 算法有 3 次收敛到次优解,而 CEGA 算法无一次收敛到次优解,并且均好于次优解,见表 2.从这个实验可见,SGA 极易陷入局部最优,而且跳出局部最优的几率很小,CGA 和 CEGA 在全局搜索性能上较 SGA 有很大提高,但相比之下,CEGA 寻优结果的整体求解精度和均方差均好于 CGA,在搜索性能和进化稳定性上略胜一筹.

表 2  $F1$  函数算法实验数据

	SGA	CGA	CEGA
最优值	1.1284	1.1507	1.1507
平均值	1.1284	1.1450	1.1452
均方差	0	0.0045	0.0037
平均时间	957.1410	307.7959	169.4756
陷入局部优(%)	100	6	0

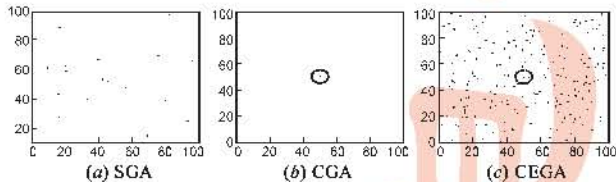


图3  $F1$ 函数群体分布

另一方面,我们研究不同算法多样性随演化代数变化情况.多样性的测试方法有基因型多样性、表现型多样性、熵多样性等,根据测试函数特点, $F1$  函数是局部最优点环绕全局最优点的函数,用表现型多样性作为评价方法, $F2$  是个多峰函数,用基因型多样性作为评价方法,见图 4、图 5.

由图 4,SGA 算法的多样性下降最快,然后稳定在一定水平,CGA 的多样性呈现跳动下降趋势,这是由于变异造成,变异在 CGA 算法中对多样性的调节起着至关重要的作用,但在 4000 代以后基本稳定,CEGA 算法

的多样性则是缓慢下降,这为持续进化的提供了条件.这种类似现象在图 5 中也能看到.由此可见,CEGA 能在一个较长时间内维持群体的多样性,有利于不断探索最优解.

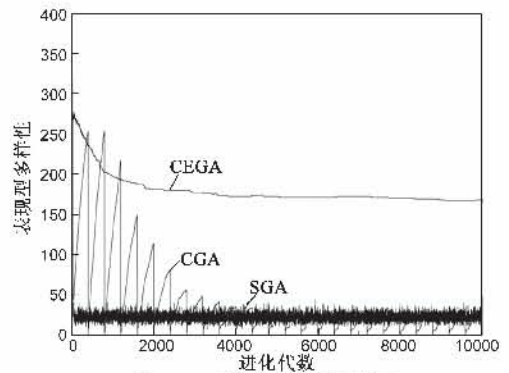


图4  $F1$ 函数多样性变化趋势

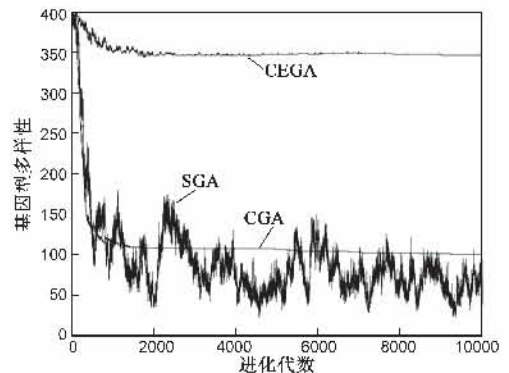


图5  $F2$ 函数多样性变化趋势

表 3  $F2$  函数算法实验数据

	SGA	CGA	CEGA
最优值	0.0415	8.6756e-006	4.0938e-006
收敛率(精度 $< 10^{-4}$ )	0	96	100
平均时间	1111.9	1310.3450	677.7340
均方差	0.0247	3.3948e-005	1.5287e-005

表 2、表 3 记录了函数优化多次运行的有关数据,由表可见,SGA 算法寻优结果远不如 CGA 和 CEGA,而且时间开销也远远高于其他两种算法. CEGA 求解精度好于 CGA,但所需时间少了一半,并且随着演化代数增加,可以大为减少所需时间开销.

种群多样性是算法进化前提,低选择压有利于生物多样性的产生,系统中过于优势的单一成分会抑制多样性.由于种群规模越大,包含的不同个体越多,搜索到更优个体的几率就越大,为逃离出局部极值点提供了可能性.由上面实验可以看到,演化规则的加入可以更好地维持群体多样性,在求解精度上有一定的提高,并且求解时间也大为减少.因此当需要大规模种群求解问题时,可以考虑 CEGA 算法,以便在时间和求解精度上达到一个较好的平衡,同时群体多样性也得到了很好的维持.

### 3.3 不同演化规则对算法性能的影响

在 CEGA 中分别采用规则 1-3, 研究演化状态对算法的影响. 实验数据记录见表 4. 从实验结果可以看到, 规则 2 下所得到的优化结果更好. 虽然规则 1 体现了生物繁殖规律, 最终群落中生物状态也趋于稳定, 当迭代次数较少时, 生命现象基本满足这种规律. 当优化问题复杂, 需要进化代数增加时, 规则 1 趋于稳定的低密度个体分布使得进化停滞, 而规则 2 使个体生存密度增加, 呈现一种自适应, 生存状态呈现周期性反复, 当优化函数存在局部极值点环绕全局极值点时, 规则 2 使算法可摆脱局部优, 以获得更好的解.  $F2$  是个多峰函数, 同样求解需要迭代次数很大, 如果系统趋于稳定, 使得算法出现寻优的停滞, 则采用规则 2 更好. 而规则 3 使空间个体生存密度过大, 形成其演化的复杂性, 在求解优化问题时并不是很好的选择.

### 4 结论

本文借鉴生物界物种个体间的相互作用是在一定范围内并处于一种动态过程, 提出了一种 CEGA 算法, 通过改变个体在生存空间的密度和分布, 提高了群体多样性, 提高了搜索速度, 这种个体分布的动态机制为元胞遗传算法提出了一种新的思路. 此外还讨论了演化规则对 CEGA 的影响, 中等密度分布的元胞个体有利于求解问题. 将来我们进一步研究元胞遗传算法的演化规则机理和选择准则.

### 参考文献:

- [1] 戴朝华, 朱云芳, 陈维荣, 林建辉. 云遗传算法及其应用[J]. 电子学报, 2007, 35(7): 1419 - 1424.  
Dai Chaohua, Zhu Yunfang, Chen Weirong, Lin Jianhui. Cloud model based genetic algorithm and its applications[J]. Acta Electronica Sinica, 2007, 35(7): 1419 - 1424. (in Chinese)
- [2] 罗小平, 韦巍. 生物免疫遗传算法的几乎处处强收敛性分析及收敛速度估计[J]. 电子学报, 2005, 33(10): 1803 - 1807.  
Luo Xiaopin, Wei Wei. The analysis on strong convergence(a. s.) and convergence rate estimate of immune genetic algorithm[J]. Acta Electronica Sinica, 33(10): 1803 - 1807. (in Chinese)
- [3] Bernabe Dorronsoro, Enrique Alba. A simple cellular genetic algorithm for continuous optimization[A]. IEEE Congress on Evolutionary Computation[C]. Vancouver, BC, Canada, July 2006. 2838 - 2844.
- [4] E Alba, B Dorronsoro, M Giacobini, et al. Decentralized cellular evolutionary algorithms[A]. Handbook of Bioinspired Algorithms and Applications[C]. CRC Press, 2005. 565 - 591.
- [5] G Rudolph, J Sprave. A cellular genetic algorithm with self-ad-

表 4 不同演化规则的实验结果

函数	规则	最大值	最小值	均方差	平均值
F1	规则 1	1.1501	1.1284	0.0052	1.1442
	规则 2	1.1507	1.1354	0.0037	1.1452
	规则 3	1.1502	1.1282	0.0070	1.1371
F2	规则 1	2.789e-004	7.6708e-006	.7563e-004	9.7897e-004
	规则 2	2.943e-005	4.0938e-006	1.5673e-005	1.5287e-005
	规则 3	3.7859e-005	8.9638e-006	9.7561e-005	2.5943e-005

- justing acceptance threshold[A]. Genetic Algorithms in Engineering Systems: Innovations and Applications on IEEE[C]. Sheffield, UK, September 1995. 365 - 372.
- [6] E Alba, B Dorronsoro. The exploration/exploitation tradeoff in dynamic cellular genetic algorithms[J]. IEEE Trans. on Evolutionary Computation, 2005, 9(2): 126 - 142.
  - [7] E Alba and J Troya. Cellular evolutionary algorithms: evaluating the influence of ratio[A]. Proceedings of the 6th International Conference on Parallel Problem Solving from Nature[C]. Berlin, Germany, 2000. 29 - 38.
  - [8] 朱刚, 马良. 函数优化的元胞蚂蚁算法[J]. 系统工程学报, 2007, 22(3): 305 - 308.  
Zhu gang, Ma liang. Cellular ant algorithm for function optimization[J]. Journal of Systems Engineering, 2007, 22(3): 305 - 308. (in Chinese)
  - [9] Michael Kirley. A cellular genetic algorithm with disturbance: optimization using dynamic spatial interactions[J]. Journal of Heuristics, 2002, 8(3): 321 - 342.
  - [10] Michael Kirley, Xiaodong Li, David G Green. Investigation of a Cellular Genetic Algorithm That Mimics Landscape Ecology[A]. The Second Asia-Pacific Conference on Simulated Evolution and Learning on Simulated Evolution and Learning[C]. Canberra, Australia, 1999. 90 - 97.
  - [11] Jaakko Kari. Theory of cellular automata: a survey[J]. Theoretical Computer Science, 2005, 334: 2 - 33.
  - [12] 王仲君, 王能超, 冯飞, 田武峰. 元胞自动机的演化行为研究[J]. 计算机应用研究, 2007, 24(8): 38 - 41.  
Wang Zhongjun, Wang Nengchao. Research on evolved behavior of cellular automata[J]. Application Research of Computers, 2007, 24(8): 38 - 41. (in Chinese)

### 作者简介:

鲁宇明 女, 1969 年 4 月生于江西南昌. 南昌航空大学副教授, 现为南京航空航天大学自动化学院博士研究生. 主要研究方向为进化计算、模式识别. E-mail: luyuming69@163.com

黎明(通信作者) 男, 1965 年 2 月生于江西樟树. 南昌航空大学教授, 南京航空航天大学博士生导师. 主要研究方向为计算智能、模式识别、图像处理. E-mail: liming@niat.edu.cn

李凌 女, 1981 年 8 月生于湖南衡阳. 南京航空航天大学自动化学院博士研究生. 主要研究方向为进化计算、图像处理. E-mail: lilin35@163.com