

# 一种基于膜系统理论的多目标演化算法

韩敏<sup>1</sup> 刘闯<sup>1</sup> 邢军<sup>2</sup>

**摘要** 提出一种用于求解多目标优化问题的基于膜系统理论的演化算法. 受膜系统理论的功能和处理化合物方式的启发, 设计了求解多目标优化问题的演化操作. 此外, 在表层膜中, 引入了非支配排序和拥挤距离两种机制改善算法的搜索效率. 采用 ZDT (Zitzler-Deb-Thiele) 和 DTLZ (Deb-Thiele-Laumanns-Zitzler) 多目标问题对所提算法进行测试, 所提算法求得的候选解既能较好地逼近真实 Pareto 前沿, 又能满足非支配解集多样性的要求. 仿真结果表明, 所提方法求解多目标优化问题是可行和有效的.

**关键词** 膜计算, P 系统, 膜算法, 多目标优化算法, Pareto 前沿

**引用格式** 韩敏, 刘闯, 邢军. 一种基于膜系统理论的多目标演化算法. 自动化学报, 2014, 40(3): 431–438

**DOI** 10.3724/SP.J.1004.2014.00431

## A Multi-objective Evolutionary Algorithm Based on Membrane System Theory

HAN Min<sup>1</sup> LIU Chuang<sup>1</sup> XING Jun<sup>2</sup>

**Abstract** In this paper, an evolutionary algorithm based on the membrane system theory has been proposed for multi-objective optimization problems. Inspired by the function and the compound reaction of the membrane system theory, the evolutionary operators are designed to solve the multi-objective optimization problems. In addition, the non-dominated sorting and the crowding distance are introduced into the skin membrane to improve the search efficiency of the algorithm. The multi-objective optimization problems including ZDT (Zitzler-Deb-Thiele) and DTLZ (Deb-Thiele-Laumanns-Zitzler) are employed to evaluate the performance of the algorithm. The proposed algorithm can not only obtain quickly the approximate Pareto front but also satisfy the requirement of diversity of Pareto front. Simulation results indicate that the proposed algorithm is feasible and effective.

**Key words** Membrane computing, P system, membrane algorithm, multi-objective optimization algorithm, Pareto front

**Citation** Han Min, Liu Chuang, Xing Jun. A multi-objective evolutionary algorithm based on membrane system theory. *Acta Automatica Sinica*, 2014, 40(3): 431–438

现实生活的很多领域 (如科学研究和工程应用) 都存在着大量的多个目标彼此冲突的优化问题, 这类问题被称为多目标优化问题. 近年来, 国内外学者在使用自然计算方法求解多目标优化问题方面取得了显著的研究成果. 以 NSGAII<sup>[1]</sup> 和 SPEA2<sup>[2]</sup> 为代表的用于求解多目标优化问题的算法相继被提出. 其中, 一些多目标算法是通过引入混合搜索<sup>[3]</sup>、小生境<sup>[4]</sup> 和偏好信息<sup>[5–6]</sup> 等技术改善算法的收敛速度和维护搜索过程中生成的解的多样性. 而另外一些算法则是通过引入新的搜索技术, 提高算法求解多目标优化问题的性能, 如粒子群<sup>[7]</sup>、差分进化<sup>[8]</sup>、免疫系统<sup>[9]</sup> 和 memetic 算法<sup>[10]</sup> 等. 但就其研究现状而言, 算法在加快收敛速度和维护解的多样性两方面

有待进一步提高.

作为自然计算的一个新分支, 膜计算是由欧洲科学院院士 Păun 受生物细胞的内部结构和处理化合物方式的启发提出的一种细胞计算理论<sup>[11–12]</sup>. 经验证, 大多数膜计算模型具有图灵机的通用性和计算有效性<sup>[13–16]</sup>, 这使得膜计算理论得到了广泛的应用. 目前, 在优化领域膜计算理论也获得了一些初步的研究成果. 2004 年, Nishida 首次提出了用于求解旅行商优化问题的包含固定和动态收缩两种膜结构的膜算法<sup>[17]</sup>. 2006 年, Zaharie 和 Ciobanu 通过对比膜算法与分布式演化算法的异同, 受膜算法的启发提出了用于求解连续优化问题的分布式演化算法<sup>[18]</sup>. Huang 等基于 P 系统提出了求解二维目标 ZDT (Zitzler-Deb-Thiele) 问题的膜算法, 强调先优化其中一个重要目标, 再优化其它目标. 但重要目标的选取可能会影响算法求解的结果<sup>[19–20]</sup>. Zhang 等将量子进化算法中的量子比特概率幅编码和相位旋转的两种思想引入到膜系统, 并提出了用于求解多目标组合背包优化问题的膜算法<sup>[21]</sup>. Liu 等提出了将遗传机制引入到膜系统的膜算法, 该算法求解 ZDT 优

收稿日期 2013-03-07 录用日期 2013-08-01  
Manuscript received March 7, 2013; accepted August 1, 2013  
国家自然科学基金 (61074096) 资助  
Supported by National Natural Science Foundation of China (61074096)

本文责任编辑 曾志刚

Recommended by Associate Editor ZENG Zhi-Gang

1. 大连理工大学电子信息与电气工程学部 辽宁大连 116023 2. 大连工业大学信息科学与工程学院 辽宁大连 116034

1. Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116023 2. Faculty of the Information Science and Engineering, Dalian Polytechnic University, Dalian 116034

化问题表现出了较好的求解性能<sup>[22]</sup>. Zhang 等提出了一种将组织型 P 系统和差分演化相结合的混合算法, 并将其应用于求解一类带约束生产参数的优化问题<sup>[23]</sup>. Zhao 等在高尔基体膜结构的单目标膜算法的基础上提出了多目标膜计算仿生优化算法, 仿真实验说明该算法能以较高的精度收敛到 Pareto 前沿<sup>[24]</sup>. 刘沙沙等提出了基于动态膜计算系统的自组织网络广播算法, 此系统对 P 系统的功能进行了扩展, 使其能够动态产生规则<sup>[25]</sup>. Zhang 等将受量子启发的演化算法和膜计算的机理及概念组合, 提出了用于求解组合背包问题的 MAQIS (Membrane algorithm with quantum-inspired sub-algorithms) 算法, 并将其应用于图像稀疏分解<sup>[26]</sup>.

虽然国内外学者基于膜计算理论提出了一些用于求解优化问题的算法, 但就其研究现状而言还有很多值得深入研究的方面. 针对多目标优化问题, 本文提出一种用于求解多目标优化问题的基于膜计算理论的演化算法, 即多目标优化演化膜算法 (Multi-objective optimization evolutionary membrane algorithm, MOEMA). 该算法的主要思想是受细胞的内部结构和处理化合物方式的启发, 通过运用细胞的内部变化共享信息来加快算法的收敛速度, 运用细胞膜内的反应规则来演化多目标优化问题的非支配解集. 在表层膜中, 非支配排序和拥挤距离两个机制维护了非支配解集的多样性. 当所提 MOEMA 满足终止条件时, 表层膜内部的多重集就是所求解优化问题的解集. 仿真实验使用二维目标 ZDT 和三维目标 DTLZ (Deb-Thiele-Laumanns-Zitzler) 问题测试所提 MOEMA 求解多目标优化问题的能力和有效性.

## 1 多目标优化问题和膜计算理论

### 1.1 多目标优化问题描述

大多数规划、线性和非线性多目标优化问题都可以使用多目标数学模型进行建模. 下面的定义可以更好地描述多目标优化问题.

**定义 1 (多目标优化问题<sup>[27]</sup>).** 多目标数学模型由两个基本部分组成, 即两个及以上的目标函数和若干个约束条件. 以最小化问题为例, 多目标优化问题的形式化描述为式 (1):

$$\begin{aligned} \mathbf{F}(\mathbf{x}) &= \min\{f_1(\mathbf{x}), \dots, f_m(\mathbf{x})\} \\ \text{s.t.} \quad g_i(\mathbf{x}) &= 0, i = 1, \dots, q; \\ h_j(\mathbf{x}) &\geq 0, j = 1, \dots, p; \\ \mathbf{x}_l &\leq \mathbf{x} \leq \mathbf{x}_u, \mathbf{x} \in \mathbf{R}^n, \mathbf{F} \in \mathbf{R}^m \end{aligned} \quad (1)$$

其中,  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  为决策变量, 且取值在  $\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u$  之间.  $\mathbf{F}(\mathbf{x}) = \min\{f_1(\mathbf{x}), \dots, f_m(\mathbf{x})\}$

表示目标函数.  $g_i(\mathbf{x})$  描述了  $q$  个等式约束,  $h_j(\mathbf{x})$  定义了  $p$  个不等式约束.

**定义 2 (Pareto 支配<sup>[27]</sup>).** 决策变量  $\mathbf{U} = (u_1, u_2, \dots, u_k)$  和  $\mathbf{V} = (v_1, v_2, \dots, v_k)$ , 如果决策变量  $\mathbf{U}$  Pareto 支配决策变量  $\mathbf{V}$ , 记为  $\mathbf{U} \prec \mathbf{V}$ , 当且仅当  $\forall i \in \{1, 2, \dots, k\}, f(u_i) \leq f(v_i)$  且  $\exists i \in \{1, 2, \dots, k\}$  使得  $f(u_i) < f(v_i)$  成立, 称  $\mathbf{U}$  Pareto 支配  $\mathbf{V}$ . 如果它们之间不存在 Pareto 支配关系, 则称它们为非支配.

**定义 3 (Pareto 最优解集,  $\mathbf{x}^*$ <sup>[27]</sup>).** 如果  $\mathbf{x}^* = \{\mathbf{x} \in \mathbf{R}^n | \neg \exists \mathbf{x}' \in \mathbf{R}^n, \mathbf{x}' \prec \mathbf{x}\}$ , 则称  $\mathbf{x}^*$  为最优解集. 其中,  $\mathbf{R}^n$  为解集的决策空间.

**定义 4 (Pareto 前沿 (Pareto front, PF)<sup>[27]</sup>).** Pareto 最优解集 ( $\mathbf{x}^*$ ) 在目标函数空间上的映射, 即  $\mathbf{PF} = \{f(\mathbf{x}) | \mathbf{x} \in \mathbf{x}^*\}$ , 称映射后的集合为 Pareto 前沿.

### 1.2 膜计算

膜计算 (Membrane computing, MC) 是从构成组织和器官的细胞中抽象出来的计算模型, 旨在从生命细胞的分层结构和处理化合物的方式中抽象出计算思想<sup>[11-12]</sup>. 由于膜系统由多个膜组合而成, 每一层膜都可以作为一个计算单元, 因此, 膜系统具有分布式和并行计算的能力.

膜系统由膜结构、对象的多重集以及规则组成<sup>[16]</sup>. 目前, 对膜系统的研究主要包括细胞型、组织型和神经型三种模型<sup>[11, 23]</sup>. 本文的研究是建立在细胞型膜系统的理论上展开的. 下面介绍度为  $n$  的细胞型膜系统的基本结构, 请见图 1 和式 (2).

$$\Pi = \{V, T, \mu, w_1, \dots, w_n, R\} \quad (2)$$

其中,

1)  $V$  是字母表, 其元素被称为字符对象. 字符对象是细胞内原子、分子或其它化学物质的抽象化表示;

2)  $T \subseteq V, T$  是输出字母表;

3)  $\mu$  是包含  $n$  个膜的膜结构, 其中  $n$  称为膜系统  $\Pi$  的度;

4)  $w_i \in V^*, 1 \leq i \leq m, w_i$  表示膜结构  $\mu$  中第  $i$  个膜内含有字符对象的多重集,  $V^*$  是由  $V$  组成的字符对象多重集;

5)  $R$  表示进化规则的有限集合;  $R_i, 1 \leq i \leq m$ , 对应于  $\mu$  中的区域  $i$  的进化规则集合.

细胞型膜系统的结构如图 1 所示, 它是由具有层次排列的一系列膜组成<sup>[11, 23]</sup>. 图 1 中最外层的膜是系统的表层膜 (Skin membrane), 它将细胞内部结构和外界环境隔开. 每一个膜确定了一个隔室 (Compartments), 又称区域 (Regions). 膜的区域中包含了多重集和反应规则. 如果一个膜的区域不再包含膜, 则称其为基本膜 (Elementary membrane).

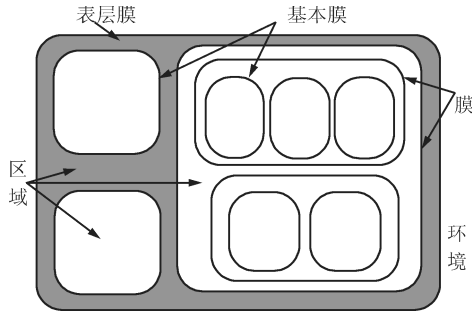


图 1 细胞型膜系统的结构

Fig. 1 The structure of the cell-like membrane system

## 2 MOEMA 算法

本文的主要目标是设计一种受细胞结构和功能的启发、以膜系统的概念和理论为基础、用于求解多目标优化问题的演化算法。膜系统由对象的多重集、反应规则和膜结构三部分组成<sup>[16]</sup>。在所提算法中, 字符对象表示优化问题的一个可行解, 多重集表示由字符对象构成的解集; 反应规则不仅说明了字符对象的演化操作, 还包括对膜结构进行调整的操作; 膜结构使算法具有分布式和并行计算的执行逻辑。

在优化过程中, 为使算法所获得的多重集具有较好的多样性, 需要引入维护多重集多样性的机制。此外, 为加快算法收敛到 Pareto 前沿的速度, 需要设计更新全局最优字符对象的操作。综上, 本文利用 NSGAI 算法<sup>[1]</sup> 中的非支配排序和拥挤距离来实现多重集的多样性; 利用外部档案来实现更新全局最优字符对象的操作。

拥挤距离是指在同一非支配等级内的两个字符对象的目标向量间的距离, 它的计算公式如式 (3) 所示:

$$d_i = \sum_{m=1}^n \frac{f_m^{i+1} - f_m^{i-1}}{f_m^{\max} - f_m^{\min}} \quad (3)$$

式中,  $n$  表示目标函数的个数,  $d_i$  表示第  $i$  个字符对象的相邻两个字符对象目标向量的距离,  $f_m^{\max}$  表示第  $m$  个目标的最大值,  $f_m^{\min}$  表示第  $m$  个目标的最小值,  $f_m^{\max}$  和  $f_m^{\min}$  分别是第  $i$  个字符对象的第  $m$  个目标值。

拥挤距离排序是指根据式 (3) 计算出所有字符对象拥挤距离后对其进行降序排序的操作。

非支配排序是将算法求得的多重集按照非支配关系划分成不同的非支配等级。非支配排序过程如图 2 所示。其中,  $P_t$  表示进化前多重集,  $Q_t$  表示进化后多重集。图 2 的具体描述如下: 对组合多重集  $R_t = P_t \cup Q_t$  进行非支配排序操作,  $R_t$  中不被任何解支配的多重集作为第一级  $F_1$ ; 在剩下的  $R_t$  多重集中, 不被任何字符对象支配的多重集作为第二

级  $F_2$ ; 依次类推, 算法求得的字符对象都获得对应的等级后, 非支配排序结束。然后, 按照非支配多重集的等级重新组合下一代多重集  $P_{t+1}$ 。如果  $F_1$  和  $F_2$  中的字符对象没有填满多重集  $P_{t+1}$ , 而  $F_3$  中的字符对象都加入到多重集  $P_{t+1}$ , 则导致  $P_{t+1}$  中字符对象过多, 这就需要首先对  $F_3$  中的字符对象进行拥挤距离排序, 然后依次将  $F_3$  中具有较大拥挤距离的字符对象填入  $P_{t+1}$ , 直到将多重集  $P_{t+1}$  填满为止。

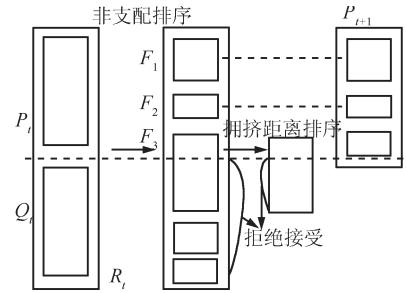


图 2 非支配排序

Fig. 2 The non-dominated sort

图 3 所示的流程图可以说明所提 MOEMA 的执行流程。

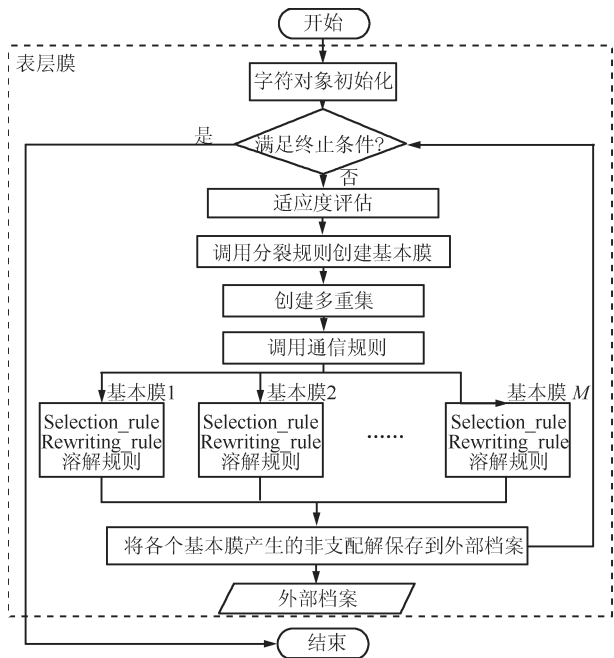


图 3 MOEMA 的示意图

Fig. 3 The schematic diagram of MOEMA

MOEMA 的步骤如下:

**步骤 1. 初始化.** 在满足优化问题约束条件的前提下, 表层膜内部区域的  $N$  个字符对象被随机生成, 其编码格式为十进制编码, 描述形式如式 (4) 所

示:

$$s_{i,j} = s_j^{\min} + (s_j^{\max} - s_j^{\min}) * \text{rand}() \quad (4)$$

其中,  $1 \leq i \leq N$ ,  $1 \leq j \leq D$ ,  $s_{i,j}$  表示第  $i$  个字符对象的第  $j$  维.  $s_j^{\min}$  和  $s_j^{\max}$  分别为字符对象第  $j$  维的最小值和最大值.  $\text{rand}()$  为随机数函数, 可以生成 0 到 1 之间的随机数.

**步骤 2.** 评估. 根据优化问题的目标函数计算每个字符对象的适应度值.

**步骤 3.** 分裂. 初始化操作结束后, 在表层膜内部区域使用分裂规则分裂出  $M$  个具有求解多目标优化问题能力的基本膜, 然后将初始化的  $M$  个多重集分别发送到每个基本膜的内部区域. 该规则的形式如式 (5) 所示:

$$[]_0 \rightarrow [ []_1, []_2, \dots, []_n ]_0 \quad (5)$$

其中,  $n$  是基本膜个数,  $[]_0$  表示表层膜,  $[]_i$  表示第  $i$  个基本膜.

为了使每个基本膜的区域都有与之对应的对象的多重集, 本文提出了划分对象的多重集的操作. 该操作需要先按照各个字符对象的适应值大小进行非支配排序, 再将排序后的多重集划分成等大小的子多重集, 从而实现对象的多重集的划分. 具体形式见式 (6).

$$\begin{aligned} w' &= \text{sort}(w) \\ w' &= \{w_1, w_2, \dots, w_m\} \\ w_i &= w'((i-1) \times n + l : n : i \times n) \\ n &= \text{sizeof}(s)/m \end{aligned} \quad (6)$$

其中,  $1 \leq i \leq m$ ,  $m$  是基本膜个数,  $w$  是在表层膜中的多重集,  $\text{sizeof}(w)$  表示多重集中字符对象个数; 而  $w'$  是对多重集中字符对象进行非支配排序得到的多重集;  $w_i$  是划分后的第  $i$  个子多重集,  $w'((i-1) \times n + l : n : i \times n)$  表示从排序后的多重集中的第  $(i-1) \times n + l$  个字符对象位置起顺序取  $n$  个字符对象.

为了增加算法搜索过程中解的多样性, 受细胞内化学物质的相互作用启发, 本文构建了一个虚拟的字符对象. 该对象被命名为均衡力字符对象, 它的形式描述如式 (7) 所示:

$$s_{a,j} = \alpha * \frac{1}{Q} \sum_{q=1}^Q w_{q,j} \quad (7)$$

其中,  $s_{a,j}$  表示均衡字符对象的第  $j$  维.  $Q$  是基本膜中字符对象的个数.  $w_{q,j}$  是多重集中第  $q$  个对象的第  $j$  维.  $\alpha$  是学习系数, 仿真实验中将其设置为 0.85.

**步骤 4.** 通信规则. 通信规则实现了发送表层膜中的多重集到基本膜区域的操作. 在表层膜区域中,

当执行划分搜索空间规则后, 通过调用通信规则可以将表层膜中的多重集发送到基本膜的区域.

**步骤 5.** 演化. 结合膜系统理论中反应规则的特点, 提出了两类规则, 即选择规则 (Selection\_rule) 和重写规则 (Rewriting\_rule).

基本膜内部区域的演化过程如下: 在基本膜内, 为了实现字符对象信息的更新, 需要调用 Selection\_rule 对字符对象进行轮盘赌选择, 然后调用 Rewriting\_rule 重写被选择的字符对象, 进而产生新的字符对象. 如果产生的字符对象优于当前区域内字符对象, 则区域内受支配的字符对象被其替换. 通过执行以上反应规则, 各个基本膜的区域都会产生所要求解多目标优化问题的部分非支配字符对象.

接下来描述 Selection\_rule 和 Rewriting\_rule 两个规则的具体形式.

Selection\_rule 是基于拥挤距离机制的选择规则, 通过比较两个字符对象拥挤距离的大小来选择字符对象. 首先, 需要从当前基本膜内随机选择两个字符对象; 其次, 比较两个字符对象拥挤距离的大小; 最后, 选择拥挤距离较小的字符对象作为候选字符对象. 它的具体描述形式如式 (8) 所示:

$$s_k = \begin{cases} s_i, s_i^{\text{dis}} < s_j^{\text{dis}} \\ s_j, s_i^{\text{dis}} \geq s_j^{\text{dis}} \end{cases} \quad (8)$$

其中,  $s_i^{\text{dis}}$  表示第  $i$  个字符对象的拥挤距离. 如果  $s_i^{\text{dis}}$  的值小于  $s_j^{\text{dis}}$  的值, 则选择字符对象  $s_i$ , 否则选择  $s_j$ .

Rewriting\_rules 是细胞内使得化学物质发生物理变化所进行的化学反应的抽象化表示. 在所提算法中, 该操作可以更新字符对象, 从而产生具有更优良性能的字符对象. 同时考虑到膜内字符对象在反应规则的作用下做不规则布朗运动, 而高斯函数可以表示布朗运动过程, 因此设计了用高斯分布函数表示字符对象随机性的两种反应规则. 该规则可以提高算法对已知解空间的开发能力.

反应规则 1 的具体形式如式 (9) 所示, 反应规则 2 的具体形式如式 (10) 所示:

$$s_{k,j} = r \times (2 \times s_{a,j} - s_{k,j}) + (1 - r) \times s_{g,j} \quad (9)$$

$$s_{k,j} = (1 + 2 \times r) \times s_{g,j} - 2 \times r \times s_{k,j} - s_{a,j} \quad (10)$$

其中,  $s_{k,j}$  表示由 Selection\_rule 选择得到的第  $k$  个字符对象的第  $j$  维.  $r$  是一个均值为 0 方差为 0.2 的高斯随机函数.  $s_{a,j}$  是均衡力字符对象的第  $j$  维.  $s_{g,j}$  是从表层膜内的外部归档解集中随机选择的字符对象的第  $j$  维.

**步骤 6.** 溶解. 各个基本膜内部区域里的演化规则都被执行后, 调用基本膜区域内的溶解规则溶解基本膜, 这样一来其内部区域产生的新字符对象就

会自动释放到表层膜内部区域. 直到每个基本膜都被溶解, 溶解操作结束. 表层膜内的基本膜都被溶解后, 来自不同基本膜的字符对象都会被释放到表层膜中, 然后将这些字符对象分别插入到外部归档中, 最后将归档的信息与表层膜中的字符对象进行非支配排序, 选取非支配等级小的字符对象, 重新组合新一代的多重集. 这个操作有利于各个基本膜之间的信息共享, 提高算法对全局未知解空间的开发和探索能力, 从而有效地增强解的多样性.

**步骤 7. 外部档案.** 它可以使算法沿着搜索空间求得更优的非支配解, 加快算法收敛到真实目标前沿的速度, 从而使得算法能够快速收敛到真实 Pareto 前沿. 外部档案的实现方式如下: 首先需要计算每个字符对象的非支配等级和拥挤距离, 根据计算结果对非支配字符对象进行选择操作; 其次, 比较选择后的字符对象, 找出非支配等级最小的字符对象, 如果二者的非支配等级相等, 则在比较拥挤距离的大小之后找出拥挤距离大的字符对象; 最后, 将选择的字符对象保存到归档中, 同时删除归档中受支配字符对象, 保留非支配字符对象, 从而形成新的下一代非支配多重集.

**步骤 8. 迭代.** 如果算法不满足终止条件, 则继续执行步骤 2. 当算法满足终止条件时, 存放在表层膜内部区域的多重集就是所求多目标问题的 Pareto 前沿.

图 4 用伪代码的形式描述了所提 MOEMA 的基本步骤.

### 3 仿真实验

为验证所提算法的性能, 仿真实验采用多目标演化算法中被广泛使用的二目标 ZDT 和三目标 DTLZ 测试函数, 具体包括 ZDT1、ZDT2、ZDT3、ZDT4、ZDT6、DTLZ5、DTLZ6 和 DTLZ7. 这些测试函数具有 Pareto 前端为凸、非连续、分布不均匀以及多模等特点, 可以测试算法在多目标优化问题上的求解能力.

本次实验在 Intel Pentium 双核 2.93 GHz, 2 GB 内存的硬件环境和 Windows XP 操作系统下运行, 选择了 NSGA-II<sup>[1]</sup>、OMOPSO<sup>[7]</sup> 和

AbYSS<sup>[27]</sup> 三个算法与所提 MOEMA 进行比较. 三个比较算法的参数见表 1, 其中参数  $D$  表示决策变量的维数. 所提 MOEMA 根据式 (11) 进行初始化.

```

Multiobjective evolutionary membrane algorithm
initParam() //初始化算法的参数
Begin skinMembrane
  create symbol-objects //根据式 (4) 创建字符对象
  While termination condition //判断终止条件
    evaluate symbol-objects //执行步骤 2 评估字符对象
    execute divide_rule //执行分裂规则, 创建基本膜
    create multiset() //根据式 (6) 创建多重集
    create ave_symbol() //根据式 (7) 创建均衡字符对象
    execute communication_rule //执行通信规则
    For each elementary membrane
      start() // elementary membrane
    End for
    Archive_maintenance //执行步骤 7 维护外部档案
  End while
End skinMembrane

Begin elementaryMembrane
initParams()
  execute selection_rule //根据式 (8) 选择字符对象
  Int r = randInt[1,2]
  switch(r)
    Case 1: execute rewriting_rule 1; break; //执行式 (9)
    Case 2: execute rewriting_rule 2; break; //执行式 (10)
  End
  Execute division_rule //执行步骤 6 溶解膜
End elementaryMembrane

```

图 4 MOEMA 的伪代码

Fig. 4 The pseudo-code of MOEMA

$$\Pi = \left\{ \begin{array}{l} \{w_{i,j}, 1 \leq i \leq 10, 1 \leq j \leq 10\}, \\ [0]_1, [2]_2, \dots, [10]_{10} 0, \\ w_1, \dots, w_{10}, \\ selection\_rule, \\ rewriting\_rule, \\ [w_i]_0 \rightarrow [[w_i]_i]_0, \\ [[w_i]_i]_0 \rightarrow [w_i]_0, 1 \leq i \leq 10 \end{array} \right\} \quad (11)$$

其中,  $\Pi$  表示膜系统,  $w_{i,j}$  表示第  $i$  个多重集中的第  $j$  个字符对象,  $[ ]_1, [ ]_2, \dots, [ ]_n 0$  表示标记为 0 的表层膜中包含了 10 个基本膜,  $w_1, \dots, w_{10}$  表示由字符对象组合而成的 1 至 10 个多重集, 规则  $selection\_rule$  表示选择字符对象操作, 规则

表 1 算法的参数设置

Table 1 The parameters of the algorithms

	NSGAII 参数 <sup>[1]</sup>	OMOPSO 参数 <sup>[7]</sup>	AbYSS 参数 <sup>[28]</sup>
群体大小	100 个个体	100 个粒子	20 个个体
选择	二进制锦标赛法	-	-
交叉	模拟二进制交叉, 交叉概率 $P_c = 0.9$	-	模拟二进制交叉, 交叉概率 $P_c = 0.9$
变异	多项式变异, 变异概率 $P_m = 1/D$	均匀变异 + 非均匀变异, 变异概率 $P_m = 1/D$	多项式变异, 变异概率 $P_m = 1/D$
Leaders 大小	-	100	-
Reference 大小	-	-	20 个

*rewriting\_rule* 表示重写字符对象操作, 规则  $[w_i]_0 \rightarrow [[w_i]_i]_0$  表示基本膜的分裂操作, 而规则  $[[w_i]_i]_0 \rightarrow [w_i]_0$  表示基本膜的溶解操作.

在实验中, 将二目标 ZDT 测试函数的决策变量维数设置为 30, 三目标 DTLZ 测试函数的决策变量维数设置为 12, 算法的适应度最大评估次数等于 25 000. 然后, 将比较方法和所提 MOMA 分别独立运行 30 次, 并对运行结果进行统计分析, 以便进一步比较算法的求解性能.

采用比较常用的四个指标函数 Error rate (ER)、General distance (GD)、Diversity ( $\Delta$ ) 和 MaxSpread (MS) 来评测算法求得非支配解的质量<sup>[28]</sup>. ER 指标度量算法求得非支配解集在 Pareto 前沿上的比率. 若  $ER = 0$ , 表示算法求得非支配解集均属于 Pareto 前沿. GD 指标度量算法求得非支配解集与真实 Pareto 前沿的逼近程度. Diversity ( $\Delta$ ) 指标度量算法求得非支配解集的分散度. 若  $\Delta = 0$ , 表示非支配解集能够均匀地分布在非劣解集上. MS 指标度量算法求得非支配解集在 Pareto 前沿上分布的宽广程度. 如果  $MS = 1$ , 表示所得非支配解集与 Pareto 前沿的宽广程度相同.

### 3.1 实验结果与分析

表 2 到表 5 分别列出了与所提 MOEMA 进行比较的多目标演化化算算法 NSGA-II<sup>[1]</sup>、OMOPSO<sup>[7]</sup> 和 AbYSS<sup>[28]</sup> 在各个指标函数下求得结果的平均值和标准差. 实验中多目标算法获得的最好的统计结果用深颜色单元格表示, 次优的统计结果用浅颜色单元格表示.

通过表 2 可知, 在 ZDT2、ZDT4 和 DTLZ7 问题上, 与多目标 OMOPSO、AbYSS 和 NSGA-II 相比, 所提算法在 ER 指标上具有较好的结果. 在 ZDT3, ZDT6 和 DTLZ6 问题上, 所提算法与比较算法相比取得了次优的结果. 表 2 的 ER 指标统计结果说明所提算法在这些测试问题上求得的解集能够较好的分布在真实 Pareto 前沿上, 也证明了所提算法具有较好的开发能力.

通过表 3 的仿真结果可以看出在 GD 指标上所提算法在 ZDT1、ZDT2 和 ZDT3 问题上要优于与之比较的多目标 OMOPSO、AbYSS 和 NSGA-II 三个算法. 表 3 的仿真结果说明所提算法在这些测试问题上求得的 Pareto 前沿能较好地接近于真实 Pareto 前沿.

表 2 不同算法在 ER 上的仿真结果

Table 2 The simulation results of different algorithms in ER

	MOEMA		NSGAI <sup>[1]</sup>		OMOPSO <sup>[7]</sup>		AbYSS <sup>[28]</sup>	
	平均值	方差	平均值	方差	平均值	方差	平均值	方差
ZDT1	4.21E-01	1.7E-02	3.91E-01	2.0E-02	4.49E-01	4.8E-02	4.01E-01	1.6E-02
ZDT2	3.45E-01	1.4E-01	3.85E-01	2.1E-02	4.59E-01	5.5E-02	3.83E-01	2.3E-02
ZDT3	4.30E-01	1.4E-02	4.35E-01	1.6E-02	4.70E-01	3.5E-02	4.26E-01	1.8E-02
ZDT4	2.43E-01	2.0E-01	3.93E-01	2.2E-02	4.49E-01	6.0E-02	3.89E-01	1.7E-02
ZDT6	4.29E-01	6.4E-02	4.34E-01	1.6E-02	3.80E-01	1.0E-01	4.38E-01	1.4E-02
DTLZ5	2.77E-01	1.5E-02	2.62E-01	1.4E-02	3.13E-01	1.6E-02	2.76E-01	9.0E-03
DTLZ6	3.02E-01	1.3E-02	3.00E-01	1.8E-02	3.32E-01	3.9E-03	3.07E-01	1.0E-02
DTLZ7	2.68E-01	7.0E-02	3.15E-01	6.4E-03	3.30E-01	7.0E-03	3.11E-01	7.9E-03

表 3 不同算法在 GD 上的仿真结果

Table 3 The simulation results of different algorithms in GD

	MOEMA		NSGAI <sup>[1]</sup>		OMOPSO <sup>[7]</sup>		AbYSS <sup>[28]</sup>	
	平均值	方差	平均值	方差	平均值	方差	平均值	方差
ZDT1	1.35E-04	2.2E-05	2.18E-04	3.4E-05	8.51E-01	1.8E-01	1.95E-04	5.4E-05
ZDT2	3.97E-05	1.6E-05	1.76E-04	4.1E-05	1.63E+00	3.8E-01	1.19E-04	5.1E-05
ZDT3	1.91E-04	8.9E-06	2.15E-04	1.9E-05	4.68E-01	8.8E-02	1.96E-04	3.8E-05
ZDT4	1.57E-01	3.2E-01	5.82E-04	1.1E-03	5.30E+01	1.4E+01	6.27E-04	3.5E-04
ZDT6	1.10E-02	2.7E-02	1.04E-03	9.4E-05	3.29E+00	7.0E-01	5.49E-04	1.3E-05
DTLZ5	1.98E-02	2.5E-02	3.74E-04	6.5E-05	1.78E-01	3.9E-02	2.53E-04	2.3E-05
DTLZ6	3.95E-01	3.9E-01	1.77E-01	2.2E-02	1.31E+00	2.5E-01	1.20E-01	3.8E-02
DTLZ7	2.96E-02	5.7E-02	2.81E-03	3.0E-04	7.57E-01	1.8E-01	1.73E-03	5.4E-04

表 4 不同算法在  $\Delta$  上的仿真结果Table 4 The simulation results of different algorithms in  $\Delta$ 

	MOEMA		NSGAI <sup>[1]</sup>		OMOPSO <sup>[7]</sup>		AbYSS <sup>[28]</sup>	
	平均值	方差	平均值	方差	平均值	方差	平均值	方差
ZDT1	7.10E-02	1.4E-02	3.68E-01	3.2E-02	8.42E-01	1.2E-01	1.03E-01	1.7E-02
ZDT2	2.05E-01	3.2E-01	3.79E-01	3.1E-02	9.02E-01	1.3E-01	1.12E-01	1.3E-02
ZDT3	7.03E-01	1.5E-03	7.48E-01	1.3E-02	8.19E-01	4.6E-02	7.14E-01	3.3E-02
ZDT4	5.99E-01	4.7E-01	3.98E-01	4.3E-02	9.22E-01	6.5E-02	1.30E-01	2.3E-02
ZDT6	2.83E-01	4.4E-01	3.58E-01	3.0E-02	9.40E-01	4.0E-02	9.05E-02	9.8E-03
DTLZ5	7.85E-01	3.6E-01	4.34E-01	5.2E-02	5.83E-01	1.1E-01	1.34E-01	1.2E-02
DTLZ6	6.68E-01	4.2E-01	8.22E-01	4.5E-02	6.00E-01	9.6E-02	8.84E-01	5.2E-02
DTLZ7	8.42E-01	2.2E-01	7.39E-01	4.0E-02	6.59E-01	6.5E-02	6.96E-01	5.9E-02

表 5 不同算法在 MS 上的仿真结果

Table 5 The simulation results of different algorithms in MS

	MOEMA		NSGAI <sup>[1]</sup>		OMOPSO <sup>[7]</sup>		AbYSS <sup>[28]</sup>	
	平均值	方差	平均值	方差	平均值	方差	平均值	方差
ZDT1	1.00E+00	2.4E-08	1.00E+00	1.9E-04	1.25E+00	1.2E-01	1.00E+00	1.1E-04
ZDT2	8.60E-01	3.5E-01	9.99E-01	2.4E-04	2.27E+00	2.5E-01	1.00E+00	5.4E-05
ZDT3	1.00E+00	5.0E-05	9.97E-01	2.3E-02	8.11E-01	6.7E-02	8.90E-01	1.6E-01
ZDT4	1.20E+00	1.6E+00	9.93E-01	2.1E-02	5.86E+01	1.2E+01	9.96E-01	7.0E-03
ZDT6	1.00E+00	0.0E+00	9.93E-01	9.8E-04	4.84E+00	5.9E-01	9.99E-01	8.1E-05
DTLZ5	1.00E+00	0.0E+00	1.00E+00	0.0E+00	9.79E-01	1.6E-02	1.00E+00	1.1E-08
DTLZ6	1.00E+00	0.0E+00	1.00E+00	0.0E+00	9.04E-01	5.9E-02	1.00E+00	1.1E-08
DTLZ7	8.41E-01	2.2E-01	9.99E-01	4.4E-04	1.47E+00	1.3E-01	6.20E-01	2.3E-01

通过表 4 的仿真结果可看出, 在  $\Delta$  指标上, 所提算法在 ZDT 测试问题上的结果与 NSGAI 和 OMOPSO 相比具有一定的优势. 与 AbYSS 相比, 所提算法在 ZDT1, ZDT3 和 ZDT6 问题上具有较好的结果. 在 DTLZ5 问题上, AbYSS 取得了最优的结果. 在 DTLZ6 问题上, OMOPSO 取得了最优的结果, 所提算法取得了次优的结果. 在 DTLZ7 问题上, OMOPSO 取得了最优的结果. 这个仿真结果说明所提算法能够较好的维持解的多样性.

表 5 的对比结果表明, 除了 ZDT2、ZDT4 和 DTLZ7 问题, 本文提出的算法在 MS 指标上的结果均要优于三个比较算法的结果. 在 MS 指标上, 仿真结果说明了所提算法求得的近似 Pareto 前沿与真实 Pareto 前沿分布的宽广程度大致相同.

综上所述, 在求解 Benchmark 的 ZDT 和 DTLZ 问题上, 四个评价指标的平均值和方差的仿真结果说明所提算法与比较算法相比具有一定的优势. 表 2 至表 5 评价指标的统计结果可以说明基于膜系统模型的所提算法在求解这类多目标优化问题的优势, 同时也说明了所提算法能够维持搜索过程中探索与开发的平衡. 因此可以证明所提算法在求解多目标优化问题是可行和有效的.

## 4 结论

基于膜系统理论, 本文提出了一种新的用于求解多目标优化问题的多目标演化算法, 即 MOEMA. 所提 MOEMA 引入了膜系统理论中的对象的多重集、反应规则和膜结构等操作并设计了适合求解优化问题的机制. 仿真实验结果说明了所提算法不仅具有较好地逼近真实 Pareto 前沿的能力, 而且提高了非支配解集的多样性. 所以, 基于膜系统模型的多目标演化算法适用于求解多目标优化问题. 对其深入研究可以促进膜计算理论和应用的发展, 且可以为优化理论提供新的求解方法.

## References

- 1 Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, **6**(2): 182-197
- 2 Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, 1999, **3**(4): 257-271
- 3 Tang L X, Wang X P. A hybrid multiobjective evolutionary algorithm for multiobjective optimization Problems. *IEEE Transactions on Evolutionary Computation*, 2013, **17**(1): 20-45

- 4 Shir O M, Emmerich M, Bäck T. Adaptive Niche radii and Niche shapes approaches for Niching with the CMA-ES. *Evolutionary Computation*, 2010, **18**(1): 97–126
- 5 Deb K, Sinha A, Korhonen P J, Wallenius J. An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. *IEEE Transactions on Evolutionary Computation*, 2010, **14**(5): 723–739
- 6 Karahan I, Koksalan M. A territory defining multiobjective evolutionary algorithms and preference incorporation. *IEEE Transactions on Evolutionary Computation*, 2010, **14**(4): 636–664
- 7 Sierra M R, Coello Coello C A. Improving PSO-based multi-objective optimization using crowding, mutation and in-dominance. In: Proceedings of the 3rd Conference on Evolutionary Multi-Criterion Optimization. Berlin, Heidelberg: Springer, 2005, **3410**: 505–519
- 8 Das S, Suganthan P N. Differential evolution: a survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 2011, **15**(1): 4–31
- 9 Gong M G, Jiao L C, Du H F, Bo L F. Multiobjective immune algorithm with nondominated neighbor-based selection. *Evolutionary Computation*, 2008, **16**(2): 225–255
- 10 Liu D S, Tan K C, Goh C K, Ho W K. A multiobjective memetic algorithm based on particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B-Cybernetics*, 2007, **37**(1): 42–50
- 11 Păun G, Rozenberg G, Salomaa A. *The Oxford Handbook of Membrane Computing*. Oxford: Oxford University Press, 2010
- 12 Păun G. Computing with membranes. *Journal of Computer and System Sciences*, 2000, **61**(1): 108–143
- 13 Ishdorj T O, Loporati A, Pan L Q, Zeng X X, Zhang X Y. Deterministic solutions to QSAT and Q3SAT by spiking neural P systems with pre-computed resources. *Theoretical Computer Science*, 2010, **411**(25): 2345–2358
- 14 Pan L Q, Wang J, Hoogeboom H J. Spiking neural P systems with astrocytes. *Neural Computation*, 2011, **24**(3): 805–825
- 15 Pan L Q, Zeng X X, Zhang X Y. Time-free spiking neural P systems. *Neural Computation*, 2011, **23**(5): 1320–1342
- 16 Zhang Ge-Xiang, Pan Lin-Qiang. A survey of membrane computing as a new branch of natural computing. *Chinese Journal of Computers*, 2010, **33**(2): 208–214 (张葛祥, 潘林强. 自然计算的新分支—膜计算. 计算机学报, 2010, **33**(2): 208–214)
- 17 Nishida T Y. An approximate algorithm for NP-complete optimization problems exploiting P-systems. In: Proceedings of Brainstorming Workshop on Uncertainty in Membrane Computing. Palma de Majorca, 2004. 185–192
- 18 Zaharie D, Ciobanu G. Distributed evolutionary algorithms inspired by membranes in solving continuous optimization problems. In: Proceedings of the 7th International Conference on Membrane Computing. 2006, **4361**: 536–553
- 19 Huang Liang, He Xiong-Xiong, Wang Ning, Xie Yi. P systems based multi-objective optimization algorithm. *Progress in Natural Science*, 2007, **17**(4): 458–465
- 20 Huang Liang, Wang Ning, Zhao Jin-Hui. Multiobjective optimization for controller design. *Acta Automatica Sinica*, 2008, **34**(4): 472–477
- 21 Zhang G X, Li Y, Gheorghe M. A multi-objective membrane algorithm for knapsack problems. In: Proceedings of the 2010 IEEE 5th International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA). Changsha, China: IEEE, 2010. 604–609
- 22 Liu C, Han M, Wang X Z. A multi-objective evolutionary algorithm based on membrane systems. In: Proceedings of 2011 4th International Workshop on Advanced Computational Intelligence (IWACI). Wuhan, China: IEEE, 2011. 103–109
- 23 Zhang G X, Cheng J X, Gheorghe M, Meng Q. A hybrid approach based on differential evolution and tissue membrane systems for solving constrained manufacturing parameter optimization problems. *Applied Soft Computing*, 2013, **13**(3): 1528–1542
- 24 Zhao J H, Wang N, Zhou P. Multiobjective bio-inspired algorithm based on membrane computing. In: Proceedings of 2012 International Conference on Computer Science and Information Processing (CSIP). Xi'an, China: IEEE, 2012. 473–477
- 25 Liu Sha-Sha, Dou Quan-Sheng, Fu Kai-Lei. Dynamic membrane computing systems based on self-organizing network broadcasting algorithm. *Application Research of Computers*, 2012, **29**(5): 1821–1824 (刘沙沙, 窦全胜, 伏开磊. 基于动态膜计算系统的自组织网络广播算法. 计算机应用研究, 2012, **29**(5): 1821–1824)
- 26 Zhang G X, Gheorghe M, Li Y Q. A membrane algorithm with quantum-inspired subalgorithms and its application to image processing. *Natural Computing*, 2012, **11**(4): 701–717
- 27 Deb K, Saha A. Multimodal optimization using a bi-objective evolutionary algorithm. *Evolutionary Computation*, 2012, **20**(1): 27–62
- 28 Nebro A J, Luna F, Alba E, Dorronsoro B, Durillo J J, Beham A. AbYSS: adapting scatter search to multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 2008, **12**(4): 439–457



韩敏 大连理工大学教授. 主要研究方向为神经网络理论及其应用, 复杂系统建模及自适应控制. 本文通信作者.

E-mail: minhan@dlut.edu.cn

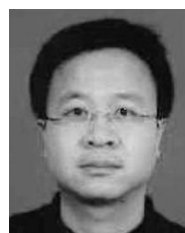
(HAN Min Professor at Dalian University of Technology. Her research interest covers neural networks theory and application, complex systems modeling, and adaptive control. Corresponding author of this paper.)



刘闯 大连理工大学博士研究生. 主要研究方向为膜计算, 智能技术, 优化算法, 复杂工业系统建模的研究.

E-mail: chuang.liu@mail.dlut.edu.cn

(LIU Chuang Ph.D. candidate at Dalian University of Technology. His research interest covers membrane computing, intelligence technology, optimization algorithm and complex systems modeling.)



邢军 大连工业大学博士, 副教授. 主要研究方向为物联网, 数据挖掘, 本体构造. E-mail: xingjun@dlpu.edu.cn

(XING Jun Associate professor at Dalian Polytechnic University. His research interest covers internet of things, data mining and constructing ontology.)