

文章编号:1001-9081(2007)05-1197-04

基于元胞自动机的模糊控制换道模型

王艳娜¹,周子力^{1,2},王新伟²

(1. 曲阜师范大学 物理工程学院, 山东 曲阜 273165;

2. 华东师范大学 计算机科学与技术系, 上海 200062)

(ynwang2008@163.com)

摘要:根据元胞自动机理论建立改进的交通流模型,给出每辆车的演化规则。在此基础上依据实际车辆行为建立换道规则,利用模糊推理来模拟人在换道过程中的主观判断过程,建立换道模型。仿真表明该方法能较好的模拟车辆的实际行为。

关键词:元胞自动机;模糊控制;换道

中图分类号: TP273 **文献标识码:** A

Lane-changing model with fuzzy control based on cellular automata

WANG Yan-na¹, ZHOU Zi-li^{1,2}, WANG Xin-wei²

(1. Collage of Physics Engineering, Qufu Normal University, Qufu Shandong 273165, China;

2. Department of Computer Science, East Normal University, Shanghai 200062, China)

Abstract: According to the theory of Cellular Automata, we built an improved traffic-flow model, and gave the evolving rules of every vehicle. We set up lane-changing rule based on traffic-flow model and vehicle's action in reality, and set up lane-changing model by using fuzzy inference to simulate people's subjective judgment. Simulation shows that the model can simulate vehicle action favourably.

Key words: cellular automata; fuzzy control; lane-changing

0 引言

元胞自动机理论是描述交通流这类非线性现象的良好工具,且在描述实际的交通特性时具有算法简单和灵活的特点,因此出现了很多基于元胞自动机的交通流模型。如 NS 模型^[1],该模型考虑车距及刹车概率对交通流的影响,显示了车辆从自由运动相到局部堵塞相的变化。但是,用该模型所做的数值模拟中,在同样条件下,给出的最大的交通流量小于实测数据,说明该模型的车辆演化规则尚不能尽如人意。因此自 NS 模型提出以后,已经被许多研究者改进和推广^[2-6]。本文在文献[6]基础上建立交通流改进模型,并细化了车辆速度演化模型,给出车辆演化规则。

根据实际车辆行驶行为和上面交通流模型中的车辆运行动态,建立换道规则。而所谓换道是描述由于车辆在行驶过程中受行驶条件的限制等而采取的变更车道的行为。考虑到换道中车道变换的本质是一种决策思维过程,模糊控制采用语言变量进行近似推理的方法,很适合描述这类主要由人做决定的主观判断过程。因此本文建立了基于模糊逻辑的换道模型。

1 一维元胞自动机交通流改进模型

交通流微观仿真真是以单个车辆的行为作为^[1]研究对象,模拟单个车辆在道路上的运动状态。

首先,如图 1 把长度为 L 的车道离散表达为等距的一维元胞,车辆随机分布在元胞上,每个元胞仅能进入一辆车,图中黑点表示车辆。其中车辆速度 $v_i(t) \in [0, v_{\max}]$,因此

每个元胞的状态分为:1) 无车(元胞为空);2) 有车且车速在 $[0, v_{\max}]$ 内取值。 $v_i(t)$ 表示第 i 辆车在 t 时刻的速度, v_{\max} 表示最大速度。 $x_i(t)$ 表示第 i 辆车在 t 时刻的位置(即所占的元胞位置)。

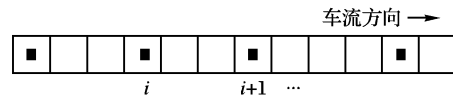


图 1 单车道离散为一维元胞图

由 t 时刻车 i 的位置 $x_i(t)$,以及它的前方近邻车 $i+1$ 的位置 $x_{i+1}(t)$,可以定义它们之间的距离为:

$$d'_i(t) = x_{i+1}(t) - x_i(t) - 1 \quad (1)$$

根据与车 $i+1$ 在 t 时刻距离和车 i 在 t 时刻的速度,确定车 i 下一时刻的可能位置。为避免冲撞,车 i 在单位时刻内前进的距离不能超过与前方近邻车的距离,则车 i 可能到达的位置可以如下确定:

$$x'_i(t) = x_i(t) + \max(\min(v_i(t) + a_i(t), d'_i(t)), 0) \quad (2)$$

同理得到车 $i+1$ 可能到达的位置:

$$x'_{i+1}(t) = x_{i+1}(t) + \max(\min(v_{i+1}(t) + a_{i+1}(t), d'_{i+1}(t)), 0) \quad (3)$$

则两车的可能瞬时距离:

$$d_i(t) = x'_{i+1}(t) - x'_i(t) - 1 = d'_i(t) + \max(\min(v_{i+1}(t) + a_{i+1}(t), d'_{i+1}(t)), 0) - \max(\min(v_i(t) + a_i(t), d'_i(t)), 0) \quad (4)$$

收稿日期:2006-11-16;修订日期:2007-01-22

作者简介:王艳娜(1976-),女,山东费县人,讲师,硕士,主要研究方向:智能控制;周子力(1973-),男,讲师,博士研究生,主要研究方向:智能控制、信息安全。

上式中, $a_i(t)$ 表示 t 时刻车 i 的可能加速度。其取值为:

$$a_i(t) = \begin{cases} -a, & v_i(t) - d'_i(t) \geq 0 \\ a, & v_i(t) - d'_i(t) < 0 \end{cases} \quad (5)$$

其中 a 是某个指定常数, 表明匀加速。加速度的选取由 $v_i(t) - d'_i(t)$ 的值决定:

1) $v_i(t) - d'_i(t) \geq 0$ 表明 t 时刻车 i 的瞬时速度大于等于与前车的距离, 取 $-a$ 表明是减速行驶;

2) $v_i(t) - d'_i(t) < 0$ 表明 t 时刻车 i 的瞬时速度小于与前车的距离, 取 a 表明是加速行驶;

当 $d_i(t) \geq v_{\max}$ 表明目前车距大, 后车可加速追赶前车。当 $d_i(t) < v_{\max}$ 且 $d_i(t) \geq v_i(t)$ 时, 由于司机类型不同, 所采取的车速也不同。行车保守的司机可能会有随机减速的行为, 本文用概率 p 来描述这种随机行为。当 $d_i(t) < v_i(t)$ 时, 车辆瞬时速度小于车辆瞬时距离, 则应采取确定性减速。据此可以建立 $t+1$ 时刻车 i 的速度模型:

$$v_i(t+1) = \begin{cases} \min(v_i(t) + a, v_{\max}), & d_i(t) \geq v_{\max} \\ \max(v_i(t) - a, 0), & \text{以概率 } p, d_i(t) < v_{\max} \text{ 且 } d_i(t) \geq v_i(t) \\ \max(v_i(t), 0), & \text{以概率 } 1-p, d_i(t) < v_{\max} \text{ 且 } d_i(t) \geq v_i(t) \\ \max(\min(v_i(t) - a, d_i(t)), 0), & d_i(t) < v_i(t) \end{cases} \quad (6)$$

根据速度, 可以得到车辆 i 在 $t+1$ 时刻到达的位置(位置更新):

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (7)$$

通常定义元胞长度 l 为阻塞时的车辆平均距离的倒数, 一般取为 7.5m 。出于对城市道路安全考虑, 一般限制车辆的最大速度不超过 70km/h (20m/s), 因此最大速度 v_{\max} 约为 3cell/s 。本文规定所有车辆具有相同的 v_{\max} 。

2 基于模糊控制的换道模型

2.1 换道类型分类

根据上述车辆运行动态模型建立换道模型。一般情况下, 存在强制换道和自由换道两种模式。

(1) 强制换道: 车辆在实际行驶中, 可能由于路口转弯或者车道障碍等原因必须通过换道才能到达目的地。

本文以双向四车道(同向两车道)为例, 有 a, b 两条同向车道, 其中 b 为左车道。 a 车道车辆如需要左转, 必须换到 b 车道, 否则车辆将无法到达目的地。本模型中假定换道的最迟位置, 称为最迟换道位置, 车辆尽量在到达该位置前完成换道, 否则将停在该位置等待换道。取距离路口停车线 3 个元胞的位置为最迟换道位置。如图 2 所示。

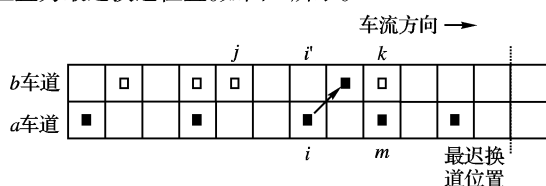


图2 双车道车辆换道示意图

图2中, a 车道的车用实心黑框表示, b 车道的车用空心框表示, 车 i 为换道车。设图中车道上的箭头所指为车 i 在 b 车道的目标位置。称 a 车道为车 i 的行驶车道, b 车道为车 i 的目的

车道。车 m 为车 i 在行驶车道上的前近邻车, 车 j 为车 i 在目的车道上的后近邻车, 车 k 为车 i 在目的车道上的前近邻车。

当车 i 在 t 时刻产生换道意图, 发出换道信号, 它将检查 $d_{lead}(t), d_{lag}(t)$ 是否满足换道要求。 $d_{lead}(t)$ 表示车 i 与目的车道前近邻车 k 的可能瞬时距离, $d_{lag}(t)$ 表示车 i 与目的车道后近邻车 j 的可能瞬时距离。在 j 车前无 a 车道的换道车辆时, k 车为其前近邻车, 可能瞬时距离用 $d_{jk}(t)$ 表示。它们的计算表达式均可从(4)式得到。具体为:

$$d_{lead}(t) = d'_{ik}(t) + \max(\min(v_k(t) + a_k(t), d'_k(t)), 0) - v_i(t) \quad (8)$$

上式中, $d'_{ik}(t)$ 为车 i 与车 k 的间距, 在这里车 i 的位置用其在 b 车道对应的位置 i' 代替。(8)式与(4)式的不同在于车 i 在换道过程中速度保持不变, 即 $v_i(t+1) = v_i(t)$, 因此 i 车到达的位置为: $x_i(t+1) = x_i(t) + v_i(t)$ 。

$$d_{lag}(t) = d'_{ji}(t) + v_i(t) - \max(\min(v_j(t) + a_j(t), d'_j(t)), 0) \quad (9)$$

上式中 $d'_{ji}(t)$ 为车 i 与车 j 在 t 时刻的距离, 要声明的是 $d'_j(t) = x_k(t) - x_j(t) - l$, 因为车 j 在无换道车 i 时是车 k 的后近邻车。

$$d_{jk}(t) = d'_j(t) + v'_k(t) - v'_j(t) \quad (10)$$

其中:

$$v'_k(t) = \max(\min(v_k(t) + a_k(t), d'_k(t)), 0)$$

$$v'_j(t) = \max(\min(v_j(t) + a_j(t), d'_j(t)), 0)$$

车 i 不换道时, b 车道上车 k 为车 j 的前近邻车, (10) 式为其可能瞬时距离公式。

(2) 自由换道: 在实际交通中, 还有一种换道是司机为达到期望的行驶速度(通常为最大速度)而进行的。它与强制换道的不同在于它换道的目的是为获得更高的速度, 该换道行为不是必须完成的。因此, 该类换道的车辆可能因为条件不足而放弃换道。

车道模型仍采用图2。当 a 车道 i 车目前行驶速度小于期望速度时, 产生换入 b 车道意图, 因此它将检查 $d_{lead}(t), d_{lag}(t)$ 是否满足换道要求, 计算表达式同式(8)~(9)。

2.2 换道规则

对换道车辆, 车辆产生换道意图后, 它将在换道成功前不再加速。若可以换道, 它以原速进入 b 车道, 即 $v_i(t+1) = v_i(t)$ 。直接进入 b 车道的目的位置, 如图2中箭头所示。

(1) 当 $d_{lead}(t) > v_i(t)$, 即车 i 进入 b 车道后向前运行的距离不会超过 $d_{lead}(t)$, 不会发生冲撞, 则与前近邻车满足换道要求。

b 车道的车 j 在没有车 i 换道的情况下应该是车 k 的跟随车, 它下时刻可能到达的位置为 $x'_j(t)$, 若 $x'_j(t) \geq x_i(t) + v_i(t)$ 即 $d_{lag}(t) \leq 0$, 这时车 i 若进入 b 车道可能会和车 j 冲突, 因此车 i 与后近邻车无法满足换道要求。

这种情况下车 i 无法进行换道, 它将进行减速行驶等待车 j 通过。在下一时刻继续检查换道条件。

(2) 当 $d_{lead}(t) > v_i(t)$, 即与前近邻车满足换道要求。而若 $x'_j(t) < x_i(t) + v_i(t)$ 即 $d_{lag}(t) > 0$, 表明车 i 换道后的位置可能到达车 j 在 $t+1$ 时刻的位置之前, 此时若 $d_{lag}(t) > v_j(t)$, 则能保证车 i 换道后能正常运行且不影响后近邻车 j 的运行。

即 $d_{lead}(t) > v_i(t)$ 且 $d_{lag}(t) > v_j(t)$, 前后近邻车都满足换道条件, 可以进行换道。

(3) 当 $d_{lead}(t) > v_i(t)$ 且 $0 < d_{lag}(t) \leq v_j(t)$, 表明若换道, 车 i 会影响车 j 的正常运行。此时车 i 向车 j 发出让道信号, 用模糊规则决策输出车 i 是否能换道。若不能换道, 在下时刻继续判断。

(4) 当 $d_{lead}(t) \leq v_i(t)$, 前近邻车不满足车 i 的换道要求, 不能进行换道。

当车 i 不能换道时, 对换道车辆, 将跟随前近邻车做不加速行驶, 目的是使 $d_{lead}(t)$ 尽快满足条件, 在下时刻继续检查换道条件。若一直无法换道, 将最终停在最迟换道位置等待。对自由换道车辆, 它可能由于条件不满足无法换道, 不会在换道位置等待。

2.3 基于模糊控制的换道模型

2.3.1 强制换道模糊控制器设计

(1) 模糊控制器的输入输出变量及隶属函数

强制换道模糊控制器为两输入单输出。输入为车 i 到达最迟换道位置的距离 D_s 以及车 i 在目的车道的后近邻车 j 在 t 时刻的速度 v 。

假定两路口之间 380m, 一个元胞长度为 7.5m, 则该段路离散表达约为 50 个元胞。记车 i 到达最迟换道位置的距离为 D_s , D_s 的基本论域为 $[0, 47]$, 将 D_s 离散化, 取论域为 $D = \{0, 2, 5, 10, 17, 26, 37, 47\}$, 那么 D_s 的模糊子集 D' 定义为 {很近, 比较近, 近, 比较远, 很远}, 简记为 {VN, MN, N, MF, VF}。该离散论域上的隶属度用表格法描述, 如表 1 所示。

表 1 D_s 的隶属函数离散化表

模糊量	论域							
	0	2	5	10	17	26	37	47
VN	1	0.7	0.1	0	0	0	0	0
MN	0	0.6	1	0.5	0.1	0	0	0
N	0	0.2	0.5	1	0.4	0.1	0	0
MF	0	0	0.1	0.4	0.6	1	0.8	0.5
VF	0	0	0	0	0	0.1	0.7	1

因为所有车辆速度在 $0 \sim 3\text{cell/s}$, 则速度 v 的论域 $V = \{0, 1, 2, 3\}$

速度 v 的模糊子集 V' 定义为 {慢, 中等, 快}, 简记为 {S, M, F}。隶属函数如表 2 所示。

表 2 v 的隶属函数离散化表

模糊量	论域			
	0	1	2	3
S	0.8	1	0.2	0
M	0.2	0.4	1	0.2
F	0	0.2	0.5	1

模糊控制器的输出是换道支持度 c , 描述了在满足规则的条件下, 对换道行为的支持程度, 它的基本论域为 $[0, 1]$, 将其离散化, 取论域 $C = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, 则 C 的模糊子集 C' 为 Y (换道) 和 N (不换道)。

表 3 c 的隶属函数离散化表

模糊量	论域									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
变化范围	0~0.11	~0.21	~0.31	~0.41	~0.51	~0.61	~0.71	~0.81	~0.91	~
Y	0	0	0	0	0.5	0.6	0.7	0.8	0.9	1
N	1	0.9	0.8	0.7	0.6	0.5	0	0	0	0

(2) 模糊控制规则

建立如表 4 的控制规则。该控制器共 15 条控制规则, 控制规则形如:

$$\text{if } D_s = D' \text{ and } v = V' \text{ then } c = C'$$

表 4 强制换道模糊规则表

V	D_s				
	VN	MN	N	MF	VF
S	Y	Y	Y	Y	Y
M	Y	Y	Y	N	N
F	N	N	N	N	N

用质心法去模糊, 即对输出求加权平均, 得到精确量 c^* , 公式为:

$$c^* = \frac{\sum_{i=1}^{10} \tilde{\mu}(C_i) C_i}{\sum_{i=1}^{10} \tilde{\mu}(C_i)} \quad (11)$$

设 y 为换道决策量, 定义为:

$$y = \text{sat}(c^*) = \begin{cases} 1, & c^* \geq \theta, 0 \leq \theta \leq 1 \\ 0, & c^* < \theta, 0 \leq \theta \leq 1 \end{cases} \quad (12)$$

决策量 y 反映了换道支持度 c 和阈值 θ 的关系。若 $y = 1$, 表明换道支持度大于阈值 θ , 这时车辆将执行换道; 若 $y = 0$, 表明换道支持度小于阈值 θ , 车辆不换道。本文中 θ 的取值由对模糊子集的各隶属度加权平均得到, 该值为 0.55。

通过上面的模糊控制器对车辆是否换道进行判断, 若结果为不能换道, 则该换道车辆将继续发出换道信号, 直到换道成功。同时, 目的车道后近邻车减速为其让道。

2.3.2 自由换道模糊控制器设计

自由换道模糊控制器同样是两输入单输出。输入为车 i 目的车道的车辆密度 ρ , 以及车 i 在目的车道的后近邻车 j 在 t 时刻的速度 v 。输出仍为车 i 是否换道 c 。

自由换道模糊控制中不考虑 D_s 的因素。模糊控制器输入为其后近邻车 j 在 t 时刻的速度 $v_j(t)$ 和目标车道车辆密度 $\rho = \frac{N}{Ln}$, 本文只需考虑 i' 位置之前的目的车道的车辆密度,

因此 N 为 i' 位置前目的车道上车辆数, Ln 为 i' 位置前目的车道元胞数, $0 \leq \rho \leq 1$ (i' 如图 2 中标记所示)。 ρ 越大, 表示车辆越多, 车辆阻塞的可能越大。将 ρ 离散化后的论域为: $P = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$; 模糊子集为 {车很少, 车少, 一般, 车多, 车很多}, 简记为 {VM, M, CO, F, VF}。

速度 v 和模糊控制器的输出 c 同强制换道模糊控制器中的 v 和 c , 这里不再赘述。

模糊推理过程与强制换道模糊控制器的模糊推理过程相同。

建立规则的原则是当目的车道中车辆少且目的车道后近邻车速度低时, 换道车辆可以换道, 换道后可以通过加速达到期望速度。若目的车道中车辆很多, 则换道车辆在换道后可能无法加速获得更高的速度, 此时它暂时不换道。

3 仿真

在仿真中, 按照泊松分布建立发车模型, 泊松分布参数取值为 0.5。车辆产生后按照交通流模型行驶; 然后按均匀分布产生换道车的数量, 在这些换道的车辆里面, 是强制换道还是自由换道也是按均匀分布分配的。仿真中所有车辆的加速度 $a = 1$ 。

图3和图4为模型的仿真图例。仿真采用双向四车道,我们取仿真图中第一个路口最左边的四条车道为例。图中标1,2的车道车辆行驶方向向左;标3,4的车道车辆行驶方向向右。用小黑点“·”代表正常行驶车辆,用小逗号“,”代表强制换道车辆,“-”代表自由换道车辆。一个正常行驶车辆产生换道意图后,从“·”变成“,”或“-”,当它换到目的车道后保持“,”或“-”,到下个时刻到来时恢复成“·”。如图3所示。

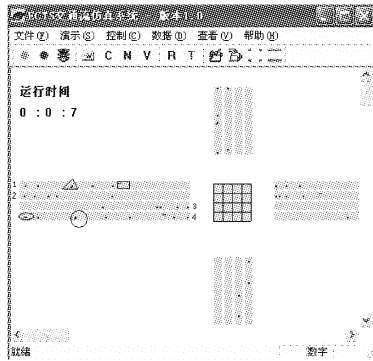


图3 换道模型仿真图1

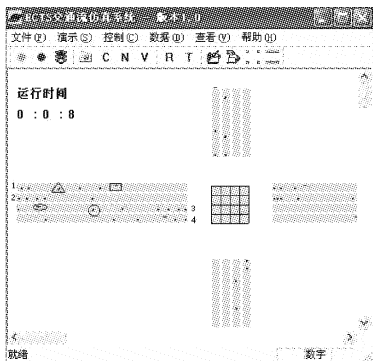


图4 换道模型仿真图2

当换道条件满足时,如图3中黑色圆圈和椭圆圈住的车辆,分别是强制换道和自由换道车辆。在下一个时刻,这两辆车按照上文中的换道规则换道同向的另外一个车道中的相应位置。相反,如果换道条件不满足,如图3黑色矩形和三角形圈住的车辆,同样也分别是强制换道和自由换道车辆,但此时换道条件不满足,在下一时刻,它们仍然在本车道行驶。图4为换道车辆下一时刻的车辆状况。

4 结语

本文基于元胞自动机建立了城市道路的交通流模型和换道模型。交通流模型中考虑了前后车辆的相对速度计算车辆瞬时间距,且在车辆间距较小时,采用车辆随机减速模拟实际交通中司机驾驶的不确定性减速。在换道模型中根据换道目的不同分为强制换道和自由换道,对这两种换道行为采用了不同的模糊控制,更符合实际交通情况。

参考文献:

- [1] NAGEL K, SCHRECKENBERG M. A Cellular Automaton Model for Freeway Traffic[J]. *Journal De Physique I*, 1992, (2): 2221 - 2229.
- [2] 王雷. 一维交通流元胞自动机模型中自组织临界性及相变行为的研究[D]. 合肥: 中国科学技术大学, 2000.
- [3] 白克钊, 谭惠丽, 刘慕仁, 等. 一种改进 Nagel-Schreckenberg 模型在开放边界条件下交通流的研究[J]. *物理学报*, 2003, 52(10): 2421 - 2426.
- [4] HELBING D, SCHRECKENBERG M. Cellular Automata Simulating Experimental Properties of Traffic Flow[J]. *Physical Review E*, 1999, 59(3).
- [5] NAGEL K, WOLF DE, WAGNER P, *et al.* Two-Lane Traffic Rules for Cellular Automata[J]. *A Systematic Approach Physical Review E*, 1998, 58(2).
- [6] 董力耘, 薛郁, 戴世强. 基于跟车思想的一维元胞自动机交通流模型[J]. *应用数学和力学*, 2002, 23(4): 331 - 337.

(上接第 1193 页)

SNPSO 的性能比较如表 2 所示。由测试结果可以看出,在微粒个数较少的情况下,SNPSO 比标准 PSO、NichePSO 具有更好的寻找全局最优值的能力。

准确度表示使用每种算法找到的解的平均个数和预期的极值点个数的比值。从表 3、表 4 的测试结果可以看出,在维数较小的情况下,SNPSO 和标准 NichePSO 拥有相同的搜索性能。在维数较高的情况下,SNPSO 能够比 NichePSO 找到更多的极值点。

表 3 Rastrigin 函数的极值点准确度比较

维数	SN(%)	DC(%)	NichePSO(%)	SNPSO(%)
1	100.00	100.00	100.00	100.00
2	78.00	67.00	100.00	100.00
3	66.67	61.11	97.45	99.07
4	58.02	54.23	97.08	99.01

表 4 Griewank 函数的极值点准确度比较

维数	SN(%)	DC(%)	NichePSO(%)	SNPSO(%)
1	100.00	90.00	100.00	100.00
2	68.00	66.60	100.00	100.00
3	41.00	56.59	94.75	97.25

4 结语

本文将顺序小生境思想引入到标准的 NichePSO 算法中,

通过子群体解散机制和 Stretching 技术的应用使 SNPSO 算法具有以下优点:

- (1) 子群体解散机制是顺序小生境与并行小生境思想的结合,提高了粒子的利用率,有利于找到更多的极值点;
- (2) Stretching 技术缩小了目标函数的极值范围,提高了全局最优值的搜索能力;
- (3) 子群体创建策略的改进克服了标准 NichePSO 由于子群体半径过大造成的诸多问题。

参考文献:

- [1] KENNEDY J, EBERHART R. Particle swarm optimization[A]. *Proceedings of the 1995 IEEE International Conference on Neural Networks[C]*. 1995. 1942 - 1948.
- [2] BRITS R, ENGELBRECHT AP, BERGH FD. A niching particle swarm optimizer[A]. *Proceeding Conference on Simulated Evolution and Learning [C]*. 2002.
- [3] PARSOPOULOS KE, PLAGIANAKOS VP, MAGOULAS GD, *et al.* Stretching technique for obtaining global minimizers through particle swarm optimization[A]. *Proceedings of the Particle Swarm Optimization Workshop[C]*. Indianapolis, USA, 2001. 22 - 29.
- [4] BRITS R. Niching strategies for particle swarm optimization[D]. Department of Computer Science, University of Pretoria, Pretoria, 2002.
- [5] BRITS R, ENGELBRECHT AP, VAN DEN BERGH F. Scalability of niche PSO[A]. *Swarm Intelligence Symposium, Proceedings of the IEEE[C]*. 2003. 228 - 234.