

基于量子粒子群优化算法的多机协同目标分配问题研究

邱立军¹, 叶文²

(1.海军航空工程学院科研部, 山东烟台, 264001.2.海军航空工程学院兵器科学与技术系, 山东烟台, 264001)

摘要: 针对粒子群优化算法搜索空间有限、容易陷入局部最优解的缺陷, 提出将一种基于量子行为的粒子群优化算法应用于多机协同目标分配问题。将该问题中的每个分配方案组成一个向量, 以此向量作为量子粒子群优化算法中的粒子进行进化, 由此在解空间内搜索最优解。实例仿真结果表明, 该算法收敛速度快、全局收敛性能好, 可以得到比遗传算法、粒子群优化算法更佳的分配效果, 证明了算法的有效性。

关键词: 多机协同 目标分配 粒子群算法 量子粒子群算法

0 引言

多机协同目标分配指为多架飞机协同一致完成任务, 考虑各种约束条件, 对飞机分配攻击目标, 确定打击目标的武器, 进行武器配置和编队配置等。多机协同目标分配是组合优化问题, 是一类 NP 难题, 这类问题的求解主要有两种思路, 一是精确搜索, 如穷举法; 另一种是启发式搜索, 在搜索过程中加入一定启发因子, 指导搜索向一个比较小的范围内进行, 如模拟退火、禁忌搜索、神经网络和遗传算法等。无论哪一种思路, 巨大的计算代价始终是 NP 问题的一个难以克服的难点。

目前, 将粒子群优化算法(particle swarm optimization PSO)用于解决目标分配是国内外正在研究的热点。PSO 算法用于求解目标分配要比其他经典调度算法具有更大的优势和可行性, 与以往经典的优化算法相比, 也取得了更好的调度效果。但在实际应用中, PSO 算法还是会经常陷入局部极值(出现早熟现象)[2,2]。针对这些不足, 考虑将量子粒子群优化算法应用于目标分配。该算法比 PSO 算法具有更快的收敛速度, 修正了 PSO 算法容易陷入局部最优解的缺陷, 比 PSO 算法具有更好的全局搜索能力。

1 问题描述

设战场中有多架飞机 V_i ($V_i \in V, i=1, 2, \dots, N_V$)、多个目标 T_i ($T_i \in T, i=1, 2, \dots, N_T$)、多个威胁或者禁飞区 P_i ($P_i \in P, i=1, 2, \dots, N_P$)。设集合 $S_i = \{T_i^1, T_i^2, \dots, T_i^n\}$ 为分配给 V_i 的目标集合, 则飞机的目标分配问题可描述为在尽量短的时间内将现有的目标分配给系统内的多架飞机, 即: $\bigcup_{i=1} S_i = T$

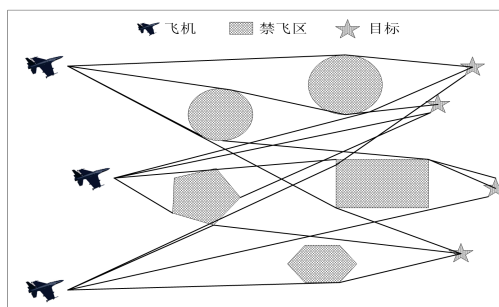


图1 多机协同作战想定

同时实现整体作战效能最大, 作战代价最小, 如图1所示。

多机协同目标分配是以整个飞机编队的整体作战效能最优为目标的, 而飞机的损耗程度、目标的价值收益和飞行航程的长度是评价作战效能的主要指标[3,4]。因此, 衡量多机协同目标分配方案优劣的重要性指标主要包括以下三个部分。

(1) 飞机损耗最小指标

飞机损耗最小指标通过最小化飞机受攻击的代价引导目标分配向着减小飞机任务毁伤代价的方向进行。该指标使飞机趋向于在安全航路飞行, 使飞机所受的威胁度最小。设第 u 架飞机攻击目标 i 的生存概率为 PS_i , 则 $HS_i = 1 - PS_i$ 为攻击该目标 i 时的损耗。对飞机进行分配任务, 使得所有飞机的损耗之和最小, 即

$$\min \sum HS_i, i=1,2,\dots,m$$

(2) 目标价值收益最大指标

目标价值收益最大指标通过对飞机执行任务时所获取的目标价值的评估, 来引导目标分配的优化和决策向着使作战效能最大化的方向进行。该指标使飞机趋向于攻击高价值目标。综合考虑目标的价值 V_i 、杀伤概率 PK_i , 则使用第 u 架飞机攻击目标 i 时, 收益为 $V_i = v_i \cdot PK_i$ 。为每架飞机分配目标, 使得总收益最大, 即

$$\max \sum V_i, i=1,2,\dots,m$$

(3) 飞行航程最短指标

飞行航程最短指标引导目标分配首先为各个飞机分配近距离的任务目标。设第 u 架飞机与目标 i 间的距离为 D_i , 为每架飞机分配目标, 使得总航程最小, 即:

$$\min \sum D_i, i=1,2,\dots,m$$

根据目标价值收益最大化函数、飞机损耗最小化函数和飞行航程最短函数, 多机协同目标分配的性能指标函数为:

$$\max f = \omega_1 \cdot \sum V_i - \omega_2 \cdot \sum HS_i - \omega_3 \cdot \sum D_i, i=1,2,\dots,m$$

其中, $\omega_1, \omega_2, \omega_3$ 为权系数, 反映了每个子目标的重要程度。

2 量子粒子群目标分配算法

在粒子群优化算法中, 粒子由速度和位置决定。在搜索过程中, 粒子的演化是随着时间的推移而定的。由于粒子的运动是有轨迹的, 而且遵循着一定的规律, 粒子的搜索空间由速度来决定。因此, 它受限于一个有限的空间, 有时粒子在搜索过程中不能达到全部可行空间, 容易使算法求解陷入局部最优, 不能达到全局收敛[5]。该结论已被研究该领域的科学家和有关学者证明。

孙俊从量子力学的角度对粒子群优化算法进行了改进, 提出了新的算法, 这种算法被命名为量子粒子群优化算法[6,7]。该算法是以 DELTA 势阱模型为基础, 确定了粒子的状态和量子的行为相似。在量子粒子群优化算法中, 粒子的状态是由波函数来确定的, 量子空间是整个解可行空间, 粒子在量子空间中进行搜索, 可知, 量子粒子群优化算法具有全局搜索能力强等优点。

2.1 量子粒子群优化算法数学模型

在标准的 PSO 算法中, 粒子的运行轨迹是由其位置和速度共同决定的。在经典的 PSO 算法中, 粒子沿着确定的轨迹运动。但在量子力学中, 轨迹是没有意义的, 因为根据不确定性原理, 粒子的位置和速度是不能同时确定的。如果 PSO 中的粒子具有量子行为, 那么, PSO 算法将以不同的方式工作。基于此, 孙俊

等人给出了 QPSO 算法。该算法描述如下:

在 D 维搜索空间中, 有 m 个粒子, 其中, 第 i 个粒子的位置是 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, $i = 1, 2, \dots, m$ 。将 x_i 代入目标函数即可算出其适应值。记第 i 个粒子搜索到的最优位置 (也称局部最优) 为 $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, $i = 1, 2, \dots, m$ 。整个粒子群搜索到的最优位置 (也称全局最优) 为 $P_g = (p_{g1}, p_{g2}, \dots, p_{gD})$, 其中 g 代表处于全局最好位置粒子的下标, $g \in \{1, 2, \dots, M\}$ 。

在量子粒子群优化算法中, 粒子的状态是由波函数来确定的, 量子空间是整个解可行空间, 粒子在量子空间中进行搜索, 可知, 量子粒子群优化算法具有全局搜索能力强等优点[8]。因此, 粒子在空间中的位置是由薛定谔方程得出粒子出现的概率密度, 得到粒子的随机位置方程为:

$$X(t) = p \pm \frac{L}{2} \ln\left(\frac{1}{u}\right)$$

式中 u 在 $[0,1]$ 上服从均匀分布的随机数, L 由下式确定:

$$L(t+1) = 2\beta |mbest - X(t)|$$

最后得到 QPSO 算法的进化方程:

$$mbest(t) = \frac{1}{m} \sum_{i=1}^m p_i(t) = \left[\frac{1}{m} \sum_{i=1}^m p_{i1}(t), \frac{1}{m} \sum_{i=1}^m p_{i2}(t), \dots, \frac{1}{m} \sum_{i=1}^m p_{iD}(t) \right]$$

$$p_{id} = (r_1 P_{id} + r_2 P_{gd}) / (r_1 + r_2)$$

$$X_{id}(t+1) = p_{id}(t) \pm \beta |mbest(t) - X_{id}(t)| \ln\left(\frac{1}{u}\right)$$

其中 $i = 1, 2, \dots, m$; $d = 1, 2, \dots, D$; u 、 r_1 和 r_2 $[0,1]$ 区间的随机数; t 为当前迭代次数; 粒子的维数为 D , 粒子的种群数目为 M , u 和 P 是在 $[0,1]$ 上的随机数, 粒子的当前最佳位置是由 $P_i(t)$ 表示即第 t 次迭代时第 i 个粒子, 粒子的全局最佳位置是由 $P_g(t)$ 表示, 即第 t 次迭代时第 i 个粒子, 群体中平均最佳位置是由 $mbest(t)$ 表示, 即第 i 次迭代时的粒子平均值, $p_{id}(t)$ 为 $P_i(t)$ 和 $P_g(t)$ 之间的随机点。 β 为算法的收缩扩张系数, 是量子粒子群优化算法的唯一的的一个重要参数, 它的取值可以是固定的, 也可以按递增或递减的方式变化, 多数情况使用下列公式确定参数 β 的取值:

$$\beta = m - (m - n) \times \frac{t}{MaxTimes}$$

β 随着迭代线性地从 m 递减到 n , 通常 $m = 1, n = 0.5$, 式中 $MaxTimes$ 是迭代的最大次数。当 $\beta \leq 0.5$ 时, 式中取减号; 当 $\beta > 0.5$ 时, 取加号。

在量子粒子群优化算法中, 关键是参数的控制和选择, 参数的取值对整个群体的收敛率有直接的关系, 在量子粒子群算法中粒子只用一个参数位置向量来表示的状态, 因此, 量子粒子群优化算法具有参数少易控制等特点, 比其他的优化算法具有很大的优势。

2.2 量子粒子群优化算法流程

QPSO 算法的计算步骤如下:

步骤 1: 初始化群体规模、迭代次数、种群个数、维数;

步骤 2: 初始化粒子的各自最好位置, 分配个体最好位置 P 初始化当前代全局最优解的位置;

步骤 3: 优化过程, 按照量子粒子群算法的进化方程更新 QPSO 算法中的所有粒子;

步骤 4: 计算群体当前的全局最优位置, 即更新 P_i 和 P_g ;

步骤 5: 根据约束条件进行判断, 当前所求的解是否满足, 若满足输出最优解, 否则跳到第二步继续搜索直到满足条件为止。

QPSO 算法用算法描述如下:

初始化 D 维搜索空间中所有粒子的位置向量, 记录当前粒子的个体极值和全局极值

Do

For i=1 to 粒子的最大取值

{

If $f(x_i) < f(P_i)$

then

{

$P_i = x_i$

$P_g = \min(P_i)$

}

For d=1 to 最大维数 D

{

$r_1 = rand(0,1), r_2 = rand(0,1)$

$p_{id} = (r_1 P_{id} + r_2 P_{gd}) / (r_1 + r_2)$

$u = rand(0,1)$

}

If $rand(0,1) > 0.5$

$X_{id}(t+1) = p_{id}(t) - \beta |mbest(t) - X_{id}(t)| \ln(\frac{1}{u})$

else

$X_{id}(t+1) = p_{id}(t) + \beta |mbest(t) - X_{id}(t)| \ln(\frac{1}{u})$

}

直到满足终止条件。

3 基于量子粒子群算法的多机协同目标分配

3.1 粒子的编码

QPSO 本质上是连续的, 因此在将 QPSO 应用于目标分配问题求解中, 需要将连续解映射到离散解上, 本文采用直接取整法的整数规格化方法来编码。直接取整法就是利用量子粒子群算法需要计算各粒子相应的位置; 当粒子到下一位置后, 立即将各粒子的位置参数值取整, 计算其适应值, 并提取本粒子最优位置 P_i 、

全部粒子中间位置 $mbest$ 和全局粒子最优位置 P_g ，接着计算下一个位置，进入下一次取整，直到满足计算要求。

采用自然数编码方式，每个粒子长度等于目标总数，粒子由按目标编号顺序排列的飞机分配编号组成，表示一种可能的分配方案，各粒子位置向量的各个分量即为对应的各个目标被分配的相应飞机序号。例如，飞机数目 n 取 8，目标数目 m 取 15，一个粒子为：

粒子	3	7	8	4	7	2	6	7	1	8	5	5	2	4	3	飞机编号
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	目标编号

即粒子的位置向量 X 为 $X=[3 \ 7 \ 8 \ 4 \ 7 \ 2 \ 6 \ 7 \ 1 \ 8 \ 5 \ 5 \ 2 \ 4 \ 3]$ ，表示第 3 架飞机攻击第 1 个目标，第 7 架飞机攻击第 2 个目标，第 8 架飞机攻击第 3 个目标，第 4 架飞机攻击第 4 个目标，……，第 5 架飞机攻击第 12 个目标，第 2 架飞机攻击第 13 个目标，第 4 架飞机攻击第 14 个目标，第 3 架飞机攻击第 15 个目标。粒子的位置向量还要保证约束条件中对每一个目标必须分配一个飞机的限制，单位粒子取值范围为 $[1, n]$ ，并且同时要求满足飞机的最大攻击目标数目限制，即单架飞机攻击的目标总数不能超过其最大攻击能力。

3.2 初始粒子群的确定

粒子群中的每一个粒子对应于待解决问题的一个可行解，因此在目标分配问题中粒子的表示就是一个可行的分配方案，在数据结构上对应于一个一维数组。为了使初始粒子群具有较好的分散度，对于每一个目标，随机的在 $[1, n]$ 中选择一架飞机分配给该目标，如果随机选择的飞机已经达到其最大攻击目标数，则重新生成随机数选择其它飞机来分配给该目标，形成一个可行解。重复构造出规定数量的粒子，形成初始粒子群。

粒子群粒子数量的大小 M 对算法的运行效率和效果有一定的影响。当 M 取值过大时会降低算法的运行效率；而 M 取值过小时，可以提高算法运行速度，但降低了粒子群的多样性，算法易早熟收敛。一般 M 取值为 30-60 之间为宜，本文中取 $M=30$ 。

3.3 适应度函数值的计算

多机协同目标分配是以整个飞机编队的整体作战效能最优为目标的，而飞机的损耗程度、目标的价值收益和飞行航程的长度是评价作战效能的主要指标。根据多机协同目标分配的性能指标函数，即目标价值收益最大化函数、飞机损耗最小化函数和飞行航程最短函数，多机协同目标分配的适应度函数可确定为：

$$f = \omega_1 \cdot \sum V_i - \omega_2 \cdot \sum HS_i - \omega_3 \cdot \sum D_i, \quad i = 1, 2, \dots, m$$

其中， ω_1 ， ω_2 ， ω_3 为权系数，反映了每个子目标的重要程度。在本文的仿真实例中，主要是为了验证协同目标分配算法的有效性，因此简化了协同目标适应度函数，主要考虑以目标价值收益程度来指导目标分配，即权系数为 $\omega_1 = 1$ ， $\omega_2 = 0$ ， $\omega_3 = 0$ 。

3.4 量子粒子群算法的流程设计

由于量子粒子群算法的进化方程是针对连续域中的问题求解来进行设计的，但是多机协同目标分配问题是离散域中的问题，因此，直接利用量子粒子群算法的进化方程进行粒子位置的更新，效果不是很理想。针对多机协同目标分配问题的特点，我们重新设计了量子粒子群算法的进化方程。

(1) $mbest(t)$ 的计算

针对群体中平均最佳位置 $mbest(t)$ ，不是进行直接的数值求平均方法来计算，而是通过粒子位置向量

中每位上数值出现的频率来计算, 即某 t 时刻, 假设群体中有 m 个粒子, 将所有粒子位置向量的第 i 位数值取出进行比较, 其中出现频率最多的数值赋值于 $mbest(t)$ 的第 i 位, 即将 m 个粒子位置向量中在第 i 位上出现频率最多的数值设定为群体中平均最佳位置 $mbest(t)$ 的第 i 位数值。这样, 通过每一位数值出现频率的比较, 最终得到 $mbest(t)$ 。

(2) P_{id} 的计算

P_{id} 为 P_i 和 P_g 之间的随机点, 在量子粒子群算法的进化方程中, P_{id} 的计算公式也是针对连续域中的问题来进行设计的, 因此, 这里我们也进行了改进。针对 P_{id} 中的每 j 位, 通过随机生成一个 $[0,1]$ 的随机数, 当 $rand(0,1) \leq 0.5$, 则 $p_{id}(j) = P_{id}(j)$, 当 $rand(0,1) > 0.5$, 则 $p_{id}(j) = P_{gd}(j)$ 。

(3) $X_{id}(t+1)$ 的更新

$X_{id}(t+1)$ 的更新正常按照量子粒子群算法的进化方程进行更新, 只是在每次更新完毕之后, 采用直接取整法, 将各粒子的位置参数值取整, 计算其适应值, 并提取本粒子最优位置 P_i 、全部粒子中间位置 $mbest$ 和全局粒子最优位置 P_g , 接着计算下一个位置, 进入下一次取整, 直到满足计算要求。

基于量子粒子群算法的多机协同目标分配的流程描述如下:

- (1) 初始化粒子群, 根据分配规则设定各粒子的随机位置, 同时初始化 P_i 和 P_g ;
- (2) 根据当前迭代代数动态调整收缩扩张系数, 从而实现算法搜索空间由全局逐步过渡到局部;
- (3) 计算各粒子的适应度函数值;
- (4) 对于每个粒子, 将其适应值与所经历的最好位置 $P_i(t)$ 的适应值进行比较, 若较好, 则将其作为当前的个体最优位置 $P_i(t)$; 将其适应值与全局所经历的最好位置 $P_g(t)$ 的适应值进行比较, 若较好, 则将其作为当前的全局最优位置 $P_g(t)$;
- (5) 判断算法收敛准则是否满足, 如满足, 则输出 $P_g(t)$, 并由 $P_g(t)$ 得到最佳目标分配方案, 算法结束; 否则, 执行 (6)。
- (6) 根据量子粒子群算法的进化方程生成新的粒子, 转到步骤 2;
- (7) 输出最佳分配方案作为优化结果, 算法结束。

4 仿真验证与分析

算例: 8 架无人机编队对 15 目标协同目标分配

假设有 8 架无人机编队和 15 个地面目标。目标的价值、飞机对目标的杀伤概率采用如下数据, 见表 1。

针对这一想定, 我们分别采用遗传算法、离散粒子群算法和量子粒子群算法对上述问题进行仿真, 并将各种算法的仿真结果进行对比。图 2, 图 3 和图 4 分别为遗传算法、离散粒子群算法和量子粒子群算法在 60 次迭代过程中最好解、平均解和最差解的变化曲线, 各算法的比较结果如图 5 和表 2 所示。

从图 2、3 和 4 的各种算法收敛速度和解的性能看, 量子粒子群算法比遗传算法、离散粒子群算法有较好的收敛特性, 能够比较快地找到问题的最优解。从量子粒子群算法的解群随迭代次数的变化情况看, 量子粒子群算法在较少的进化代数内, 其解群就向最优解的方向收敛, 说明量子粒子群算法的优化效率较高。

表 1 目标价值和飞机对目标的杀伤概率

	V	无人机1	无人机2	无人机3	无人机4	无人机5	无人机6	无人机7	无人机8
目标1	0.47	0.87	0.87	0.87	0.87	0.87	0.87	0.62	0.48
目标2	0.97	0.52	0.52	0.52	0.52	0.52	0.52	0.87	0.20
目标3	0.76	0.11	0.11	0.11	0.11	0.11	0.11	0.70	0.42
目标4	0.62	0.78	0.78	0.78	0.78	0.78	0.78	0.22	0.16
目标5	0.48	0.72	0.72	0.72	0.72	0.72	0.72	0.80	0.43
目标6	0.77	0.69	0.69	0.69	0.69	0.69	0.69	0.42	0.58
目标7	0.33	0.94	0.94	0.94	0.94	0.94	0.94	0.43	0.69
目标8	0.74	0.72	0.72	0.72	0.72	0.72	0.72	0.90	0.03
目标9	0.54	0.36	0.36	0.36	0.36	0.36	0.36	0.13	0.34
目标10	0.65	0.28	0.28	0.28	0.28	0.28	0.28	0.95	0.72
目标11	0.43	0.27	0.27	0.27	0.27	0.27	0.27	0.18	0.15
目标12	0.35	0.74	0.74	0.74	0.74	0.74	0.74	0.19	0.24
目标13	0.63	0.24	0.24	0.24	0.24	0.24	0.24	0.12	0.29
目标14	0.66	0.78	0.78	0.78	0.78	0.78	0.78	0.61	0.30
目标15	0.57	0.45	0.45	0.45	0.45	0.45	0.45	0.35	0.75

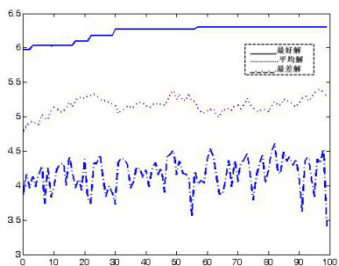


图 2 遗传算法的收敛曲线

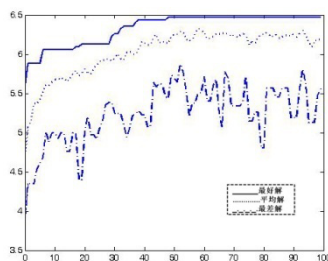


图 3 离散粒子群算法的收敛曲线

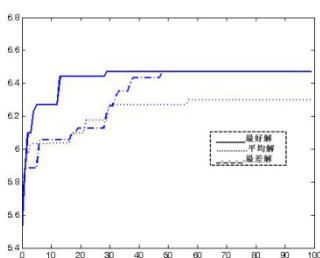


图 4 量子粒子群算法的收敛曲线

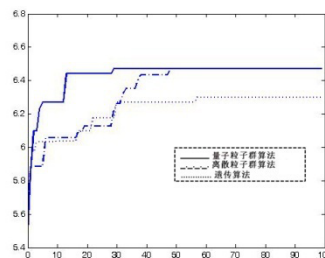


图 5 算法性能比较

表 2 算法性能比较

比较指标	遗传算法	离散粒子群算法	量子粒子群算法
迭代60次后算法的最好解	6.3003	6.4719	6.4719
迭代60次后算法的平均解	5.3044	6.1876	6.2848
迭代60次后算法的最差解	3.4137	5.5665	5.7781
最好解首次出现的代数	58	49	30
搜索到的最优分配结果	(572522634735838)	(677272272732828)	(277474674765828)

5 结论

根据多机协同目标分配问题的特点，本文针对粒子群优化算法搜索空间有限、容易陷入局部最优点的缺陷，应用基于量子行为的粒子群优化算法解决多机协同目标分配问题。采用自然数编码方式，粒子由按目标编号顺序排列的飞机分配编号组成，表示一种可能的分配方案，各粒子位置向量的各个分量即为对应的各个目标被分配的相应飞机序号，由此在解空间内搜索最优解。实例仿真结果表明，该算法收敛速度快、全局收敛性能好，能够稳定快速地找到较优分配方案，并且算法简单、灵活，易于实现和扩展，为该类多约束多目标整数规划问题的求解提供了新的思路和方法。

参考文献:

- [1]叶媛媛,多UCAV协同任务规划方法研究[D]. 长沙:国防科学技术大学 2005
- [2]龙涛,多UCAV协同任务控制中分布式任务分配与任务协调技术研究[D]. 长沙:国防科学技术大学 2006
- [3]叶文,朱爱红,欧阳中辉,范洪达. 基于混合离散粒子群算法的多无人作战飞机协同目标分配. 兵工学报. 2010.03:331-336
- [4]叶文,朱爱红,潘长鹏,范洪达. 多UCAV协同目标分配算法研究. 系统工程与电子技术. 2010.01:104-108
- [5]潘全科,王文宏,朱剑英. 解决无等待流水车间调度问题的离散粒子群优化算法[J]. 计算机集成制造系统. 2007.13(6):1127-1132
- [6]SUN Jun, FENG Bin, XU Wen-bo. Particle swarm optimization with particles having quantum behavior[C] Proc of Congress on Evolutionary Computation. 2004:325-331.
- [7]SUN Jun, XU Wen-bo, FENG Bin. A global search strategy of quantum behaved particle swarm optimization[C] Proc of IEEE Conference on Cybernetics and Intelligent Systems. 2004: 111-116
- [8]张兰. 量子粒子群算法及其应用[D]. 西安:西北大学 2010:25-40