

反舰导弹航路规划的 OACRR-PSO 算法

刘钢¹ 老松杨¹ 袁灿² 侯绿林¹ 谭东风¹

摘要 为了提高反舰导弹航路规划算法的搜索效率,从几何学角度对航路规划空间进行了研究,在将功能区域概念融入逆向航路规划的过程中发现了功能区域的几何学渐变规律,据此提出功能区域簇作为其物理载体.将功能区域簇引入粒子群优化(Particle swarm optimization, PSO)算法,提出了功能区域簇实时约束(Operational area cluster real-time restriction, OACRR)的 PSO 算法(OACRR-PSO).为了便于表示功能区域簇,采用航路极坐标编码方式.与传统的 PSO 算法不同的是,考虑到粒子中分量之间的关联性,该算法在优化过程中并不是对粒子的整个速度分量同时进行更新,而是引入一种分步递归进化策略对粒子的分量逐步进行更新.在粒子的更新过程中,使用功能区域簇来实时限定粒子位置分量的准确更新范围,使得算法搜索空间逐步减小,从而加速算法收敛.仿真实验结果表明,分步递归进化策略能够非常显著地提高算法的全局搜索性能,并且算法收敛速度快、稳定性好.

关键词 反舰导弹, 航路规划, 功能区域簇, 粒子群优化, 分步递归进化, 黄金分割位置

引用格式 刘钢, 老松杨, 袁灿, 侯绿林, 谭东风. 反舰导弹航路规划的 OACRR-PSO 算法. 自动化学报, 2012, 38(9): 1528–1537

DOI 10.3724/SP.J.1004.2012.01528

OACRR-PSO Algorithm for Anti-ship Missile Path Planning

LIU Gang¹ LAO Song-Yang¹ YUAN Can² HOU Lv-Lin¹ TAN Dong-Feng¹

Abstract In order to improve the search efficiency of path planning algorithm for anti-ship missile, the planning space is researched based on geometric principle. The geometric gradual transformation rule of operational area is revealed when fusing the concept of operational area into the process of converse path planning, hereby, the operational area cluster is proposed to be its physical carrier. By introducing the operational area cluster into particle swarm optimization (PSO) algorithm, a PSO algorithm real-time restricted by operational area cluster (OACRR-PSO) is proposed. To express the operational area cluster expediently, the polar coordinates code mode is adopted in path coding. Considering the relationship between the adjoining vectors of particle, OACRR-PSO does not update all the velocity vectors of particle simultaneously in the course of optimization, which is different from conventional PSO, but updates sequentially by adopting the strategy of sequential recursion evolution. In the course of updating particle, the operational area cluster is used to restrict the position vectors of particle in exact updating area in real-time, which reduces the search space step by step to increase the convergence velocity. Simulation results indicate that the strategy of sequential recursion evolution could improve the algorithm's global search capabilities and the algorithm possesses a better convergence rate and robustness.

Key words Anti-ship missile, path planning, operational area cluster, particle swarm optimization (PSO), sequential recursion evolution, golden section position

Citation Liu Gang, Lao Song-Yang, Yuan Can, Hou Lv-Lin, Tan Dong-Feng. OACRR-PSO algorithm for anti-ship missile path planning. *Acta Automatica Sinica*, 2012, 38(9): 1528–1537

反舰导弹航路规划是指在导弹发射以前,根据

实际的战场态势、作战需要和导弹飞行性能,对导弹在自控段的飞行航路进行设定,使得导弹在发射后按照预设的航路飞行.由于反舰导弹在自控飞行段大多在海平面上采用定高等速的掠海飞行方式,因此可以进行必要的纵向简化,其航路规划在二维平面上展开^[1-2].

反舰导弹的航路特征主要包括 4 个方面^[2]: 1) 导弹航路由航路点间的直线航段和转弯时的弧线航段组成; 2) 总航程存在约束条件; 3) 相邻两个航路点间距存在约束条件; 4) 航路转向角存在约

收稿日期 2011-12-23 录用日期 2012-05-09

Manuscript received November 23, 2011; accepted May 9, 2012
国家自然科学基金(70571084, 61074121), 国家部委“十一五”科研计划(513040404-1)资助

Supported by National Natural Science Foundation of China (70571084, 61074121) and Ministry Commission-Level Research Foundation of China (513040404-1)

本文责任编辑 刘德荣

Recommended by Associate Editor LIU De-Rong

1. 国防科学技术大学信息系统工程重点实验室 长沙 410073 2. 海军工程大学核能科学与工程系 武汉 430033

1. Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073 2. Department of Nuclear Science and Engineering, Naval University of Engineering, Wuhan 430033

束条件. 航路规划首先需要构造一个规划空间 (即航路搜索空间), 其算法效率与规划空间的复杂度紧密相关. 针对空间复杂性, 基于几何学原理, 按照规划空间的构造方法主要可以分为三大类^[3-4]: 单元分解法^[5]、路标图法^[6] 和人工势场法^[7]. 它们通常的做法都是将位形空间内的寻优问题转化为拓扑空间的搜索问题. 根据反舰导弹的航路特征可知, 其规划空间应是一块封闭的有界二维区域, 其本身是有一定的规律可循的. 然而上述方法都没有考虑规划空间的几何学性质及规律, 因而无法充分利用规划空间本身的这种性质对搜索空间进行准确界定, 或者进行合理压缩, 导致算法信息处理量大、运算时间长、对计算机性能要求高.

航路规划是一个空间搜索问题, 需要采用基于几何学的搜索算法进行求解. 目前关于这类寻优算法的研究比较深入, 大致可以分为两类^[8]: 确定型搜索算法 (或启发式算法^[9]) 和随机型搜索算法 (或进化算法^[10]). 确定型算法作为逐点搜索算法, 其搜索空间相当庞大, 并且规划效果对启发式函数的依赖性太强. 通过将各种约束加入到搜索过程中, 虽然可以极大地修剪搜索空间, 但由于受到自身状态空间的限制, 当规划区域过大时, 会出现组合爆炸, 规划时间将呈指数增长^[8, 11]. 进化算法是一种基于种群的优化方法, 常用的进化算法有遗传算法、蚁群算法和粒子群算法等. 粒子群优化 (Particle swarm optimization, PSO) 算法是近几年发展起来的一种新型进化算法, 它于 1995 年由 Kennedy 等^[12-13] 首次提出, 并于 1998 年得到了 Shi 等^[14] 的改进, 形成了研究者们公认的标准 PSO 算法. 与其他进化算法相比, 它具有概念简单、实现容易、易于操作、调整参数少等优点. 同时, PSO 算法具有鲁棒性好和收敛速度快两个显著特点, 使得它非常适合于对航路规划这种实时性要求较高的复杂问题进行求解. 最重要的是, PSO 算法无论是从编码方式还是进化过程来说, 都具有其他智能算法无可比拟的几何表达能力, 从几何空间到解空间以及搜索空间的映射都十分直观; 并且, 作为一种全局连续搜索算法, 其进化公式和进化过程都是连续的, 这两点与非网格拓扑的航路规划问题的几何学特点十分吻合. 近年来, 针对 PSO 算法的早熟收敛问题, 相关学者^[11, 15] 根据各自需求对标准 PSO 算法进行了相应的改进, 并将其应用于无人飞行器航路规划. 对于粒子结构, 文献 [11] 中的航路编码采用定长实数编码粒子结构, 既合理地表达了航路又便于编程实现, 但是作者没有对该航迹编码结构下的解空间进行专门的研究, 导致算法搜索空间庞大、收敛速度较慢. 对于粒子更新方式, 以往的算法都是对粒子的整个速度分量 (即解矢量中所有维的数值) 同时进行更新^[16-17],

因此, 它仅能描述粒子每次进化后的运动状态, 而无法表达粒子进化的过程. 但是, 往往粒子中的各个分量并不总是相互独立的, 分量之间可能存在关联关系, 对各分量同时更新则无法体现出这种关联性. 这样所得到的适应值就只能评判解矢量的整体质量, 而不能判断每个维是否都向最优方向移动, 因此, 很难兼顾所有维的优化方向, 势必影响算法的收敛速度.

本文基于传统 PSO 算法, 把握航路规划的几何学本质, 充分挖掘和运用其数理内涵, 提出一种适用于反舰导弹航路规划的功能区域簇实时约束 (Operational area cluster real-time restriction, OACRR) 的 PSO 算法 (OACRR-PSO). 该算法的基本思想是: 引入功能区域簇准确界定算法实时搜索空间, 加速算法收敛. 这个算法的创新之处在于 4 个方面: 1) 为了定量描述航路规划的规划空间, 基于几何学原理提出了航路规划功能区域的概念, 将功能区域概念融入逆向航路规划过程中, 发现了功能区域的几何学渐变规律, 以此抽象出功能区域簇的概念, 用于减小算法搜索空间; 2) 为了便于表示功能区域簇, 采用航路极坐标编码方式, 利用航路点极坐标序列来表示航路; 3) 在算法的优化过程中, 考虑到粒子中分量之间的关联性, 算法并不是对粒子的整个速度分量或位置分量 (即所有维的数值) 同时进行更新, 而是引入一种分步递归进化策略, 采用递归思想对粒子的分量逐步进行更新; 4) 在粒子的更新过程中, 使用功能区域簇来实时限定粒子位置分量的准确更新范围, 实现了粒子中分量之间的关联关系, 使得算法搜索空间逐步减小, 从而加速算法收敛.

1 反舰导弹航路规划空间建模

反舰导弹航路规划本质上是在规划空间内寻找一条从发射点到目标点满足给定约束条件的最优或可行的飞行航路. 首先需要构造一个数据结构来表达航路的搜索空间, 显然, 规划空间的范围是影响航路规划效率的重要因素.

1.1 航路规划功能区域

为简化描述, 先作如下约定:

如图 1 所示, 发射点为 A , 目标点为航路点 P_1 , 随着逆向航路规划^[1] 下标依次递增. 对于第 i ($i \in \mathbf{N}$) 个航路点 P_i , P_i 与 A 的距离为 d_i ; P_i 与 P_{i+1} 之间的航段距离为 l_i ; s 为导弹最大有效射程.

定义 1^[1]. 以 A 和 P_1 分别为两个焦点, 以 s 为长轴, 作椭圆 O_1 , 称 O_1 所围成的区域为航路规划功能区域, 简称功能区域 (Operational area, OA), 记为 $\Omega(O_1)$.

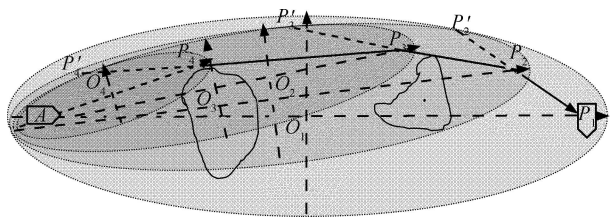


图 1 功能区域的几何学渐变规律

Fig. 1 The geometric gradual transformation law of OA

由于受 s 的限制, 满足导弹航路距离约束的所有航路转向点 (导弹航向发生改变的航路点) 必然在功能区域之内. 则可以得到引理 1.

引理 1^[2]. 导弹航路满足航路距离约束的必要条件为

$$P_i (i \in \mathbf{N}) \in \Omega(O_1) \quad (1)$$

证明略. O_1 的各参数为^[1]: 长半轴 $a = s/2$, 半焦距 $c = d_1/2$, 离心率 $e = d_1/s$, 焦准距 $p = s^2/2d_1 - d_1/2$. 可见, 此椭圆基本参数由 s 和 d_1 确定. 此外, 其面积为

$$S = \pi ab = \frac{\pi s \sqrt{s^2 - d_1^2}}{4} \quad (2)$$

当 $d_1(s)$ 确定时, S 为 $s(d_1)$ 的单调递增 (减) 函数. $s(d_1)$ 越大, 椭圆的面积就越大 (小) (可规划的区域就越大 (小)), 航路规划的可操作性也就越强 (弱); 反之, 则反.

1.2 航路规划形式化定义

为了理解、把握航路规划问题的本质, 下面从集合论的角度给出航路规划的形式化定义.

定义 2. 航路规划的集合论定义:

$$PP = \langle \{\mathbf{P}\}, \{D\}, \{st\} \rangle \quad (3)$$

式中, PP 表示航路规划问题, $\{\mathbf{P}\}$ 表示有序的航路点集合, $\{D\}$ 表示功能区域中的通行障碍集合; $\{st\}$ 表示航路规划中必须满足的约束条件.

基于以上定义, 运用功能区域的概念可以将航路规划问题转化为:

定义 3. 航路规划是一个在有界、无序的航路点集合中寻找满足约束条件的最优或次优的有序航路点集合问题. 如图 2 所示, $\Omega(O_1)$ 为航路规划功能区域, 其实质为一个有界、无序的航路点集合; $\{\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_n\}$ 是一个包含 n ($n \in \mathbf{N}$) 个有序航

路点集合的集合, 一个有序航路点集合代表一条满足约束条件的航路, $\{\mathbf{P}_i\}$ 为其中最优或次优的一个航路点序列.

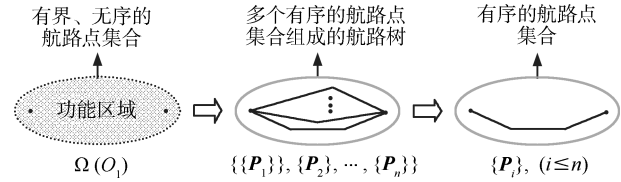


图 2 航路规划问题描述

Fig. 2 Formal description of anti-ship missile path planning (ASMPP)

1.3 功能区域的几何学渐变规律

定义 4. 从 P_1 开始, 在逆向航路规划过程中, 当已确定 n ($n \geq 1$) 个航路转向点时, 则相应地确定 n 个航路段, 将 s 与 $\sum_{i=1}^n l_i$ 之差称为剩余最大有效射程, 对于 P_i ($i \geq 2$), 记所对应的剩余最大有效射程为 s_i , 则有:

$$s_i = s - \sum_{j=1}^{i-1} l_j \quad (4)$$

定义 5. 以 A 和当前 P_i ($i \geq 2$) 分别为两个焦点, 以 s_i 为长轴, 作椭圆 O_i , 称 O_i 所围成的区域为剩余功能区域, 记为 $\Omega(O_i)$, 并记 O_i 所对应的离心率为 e_i , 长半轴为 a_i , 半焦距为 c_i , 焦准距为 p_i .

与引理 1 同理, 由于受 s_i 的限制, 满足导弹航路距离约束条件的所有航路点 P_j ($j \geq i+1$) 必然在当前 (剩余) 功能区域 $\Omega(O_i)$ 之内; 反之, 也成立. 则可以得到引理 2.

引理 2. 导弹航路满足航路距离约束的充分必要条件为

$$P_j (j \geq i+1) \in \Omega(O_i) \quad (5)$$

证明略^[2]. O_i 的各参数为^[1]: 长半轴 $a_i = (s - \sum_{j=1}^{i-1} l_j)/2$, 半焦距 $c_i = d_i/2$, 离心率 $e_i = d_i/(s - \sum_{j=1}^{i-1} l_j)$, 焦准距 $p_i = (s - \sum_{j=1}^{i-1} l_j)^2/2d_i - d_i/2$.

在逆向航路规划过程中, s_i 逐渐变小, $\Omega(O_i)$ 从 $\Omega(O_1)$ 开始也逐渐变小, 最后收敛于发射点. 记射线 $\mathbf{P}_i \mathbf{P}_{i+1} \cap O_i = \mathbf{P}'_{i+1}$, 则可以得到引理 3.

引理 3. $O_{i+1} \subsetneq O_i$, 并且 O_i 与 O_{i+1} 必内切于一点, 有 $O_{i+1} \cap O_i = \mathbf{P}'_{i+1}$.

证明略^[2]. 以上便是功能区域的几何学渐变规律, 由此可作如下定义:

定义 6. 从 A 点向 P_1 点进行航路规划, 当已确定 n ($n \geq 1$) 个航路转向点时, 将其功能区域及其剩余功能区域的集合称为功能区域簇 (Operational area cluster, OAC), 记为 $OAC(A, P_1)$. 则有:

$$OAC(A, P_1) = \{\Omega(O_1), \Omega(O_2), \dots, \Omega(O_{n+1})\} \quad (6)$$

功能区域簇是功能区域几何学渐变规律的物理载体, 下面将其引入 PSO 算法, 介绍反舰导弹航路规划的 OACRR-PSO 算法.

2 OACRR-PSO 算法

OACRR-PSO 算法的基本思想是: 基于传统 PSO 算法, 引入功能区域簇准确界定算法实时搜索空间、加速算法收敛. 为了便于表示功能区域簇, 采用航路极坐标编码方式, 利用航路点极坐标序列来表示航路. 在算法的优化过程中, 考虑到粒子中分量之间的关联性, 算法并不是对粒子的整个速度分量或位置分量 (即所有维的数值) 同时进行更新, 而是采用递归思想对粒子的分量逐步进行更新. 在粒子的更新过程中, 通过引入一种分步递归进化策略, 并且使用功能区域簇来实时限定粒子位置分量的准确更新范围, 使得算法搜索空间逐步减小, 从而加速算法收敛.

2.1 粒子表示与航路编码

PSO 算法不能直接处理解空间上的变量, 需要通过编码将其形式化. 一个粒子表示解空间中的一个可能解, 即搜索空间中的一条备选航路, 或称之为种群中的一个个体. 航路编码本质上是在种群中的每个粒子与搜索空间中的备选航路之间建立一个一一映射关系^[1]. 粒子中的相应分量代表航路点的位置分量.

种群可以表示为一个矩阵 $X = [X_1, X_2, \dots, X_m]^T$, 其中, m 为种群的大小, X_i 为种群中的第 i ($i = 1, \dots, m$) 个个体. 功能区域簇准确界定了搜索空间的范围, 航路编码应考虑到便于对其进行表示. 由于功能区域簇由若干个共有一个焦点的椭圆组成, 因此, 适合于用极坐标表示. 以 A 为极点, AP_1 方向为极轴的方向, 建立极坐标系. 因为每条航路有相同的发射点和目标点, 所以发射点与目标点不参与粒子的编码. 设每条航路包含 n 个航路转向点, 为了记录每个航路转向点的位置信息, 采用以下的定长实数编码粒子结构: $X_i = [x_{i,1}, \dots, x_{i,n}, x_{i,n+1}, \dots, x_{i,2n}]$, 其中 $(x_{i,1}, \dots, x_{i,n})$ 和 $(x_{i,n+1}, \dots, x_{i,2n})$ 分别记录了

n 个航路转向点在平面内的极角 θ 和极径 ρ . 第 i 条航路的第 j 个航路转向点坐标为 $(x_{i,j}, x_{i,j+n})$, 其中 $j = 1, \dots, n$. 即若航路由 n 个航路转向点组成, 则粒子的维数为 $2n$. 编码结构如图 3 所示.

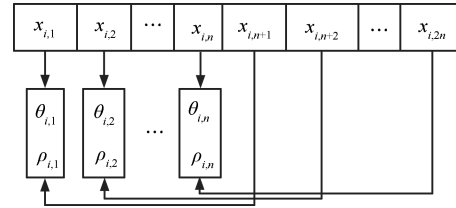


图 3 OACRR-PSO 算法编码方案

Fig. 3 Code scheme in OACRR-PSO algorithm

2.2 解空间与初始化

在完成粒子编码以后, 就明确了待优化的参数, 下一步需要定义解空间. 在本文中, $x_{i,j}$ ($i = 1, \dots, m; j = 1, \dots, 2n$) 为需要优化的参数, 下面确定其范围. 根据引理 1 可知, 功能区域 $\Omega(O_1)$ 准确界定了所有合法航路转向点的选择范围, 将其映射到粒子所存在的 \mathbf{R}^{2n} 空间中, 便可以得到解空间. 根据航路不迂回原则, 所以极角取值 $(-\pi/2, \pi/2)$, 极径的取值随极角而变化, 它将所有航路转向点限定在功能区域 $\Omega(O_1)$ 之内. 在所建立的极坐标系中, O_1 的方程为

$$O_1: \rho = \frac{ep}{1 - e \cos \theta} \quad (7)$$

于是, 可以给出参数 $x_{i,j}$ 的范围:

$$\begin{cases} -\frac{\pi}{2} < x_{i,j} < \frac{\pi}{2}, & j = 1, \dots, n \\ 0 < x_{i,j} \leq \frac{ep}{1 - e \cos x_{i,j-n}}, & j = n+1, \dots, 2n \end{cases} \quad (8)$$

为了简便起见, 本文采用常用的随机初始化方法, 即每个粒子被初始化为解空间中的一个随机位置, 然后为每个粒子初始化一个随机速度. 种群大小以及航路转向点个数 (等价于粒子维数) 是与算法相关的需要用户确定的参数. 显然, 航路转向点的个数越多, 生成的结果航路的精度就越高, 但是所花费的计算时间也相应增加.

2.3 适应度函数

在解空间中一般存在无数条可行航路, 需要设计目标函数来确定最优航路. 与其他优化问题不同的是, 航路规划并不存在经典数学上的目标函数, 在航路满足性能约束的前提下, 仅需根据实际需要来规划一条可行的、同时又较为合理的航路. 为了避免因单目标函数值数量级不同, 而造成单目标对总

目标的影响力相差悬殊, 建立目标的模糊满意度函数, 并根据作战需要确定合理的权重系数, 采用多属性模糊优化方法建立性能指标^[1]:

$$T = \max(\omega_s \cdot \mu_s + \omega_\alpha \cdot \mu_\alpha + \omega_\sigma \cdot \mu_\sigma) \quad (9)$$

式中, μ_s , μ_α , μ_σ 分别为某条航路的总航程 s 、各航路转向角的和 α 以及各转向角度绝对值均方差 σ 的满意度; ω_s , ω_α , ω_σ 分别为对应的权重系数, 有: $\omega_s + \omega_\alpha + \omega_\sigma = 1$.

对于 μ_s , μ_α , μ_σ 的单目标函数 $\min f(x)$, 存在一个最大可接受值 $f_{\max}(x)$ 和一个理想值 $f_{\min}(x)$, 满意度函数为^[1]

$$\mu(f(x)) = \begin{cases} 1, & f(x) \leq f_{\min}(x) \\ \frac{f_{\max}(x) - f(x)}{f_{\max}(x) - f_{\min}(x)}, & f_{\min}(x) < f(x) < f_{\max}(x) \\ 0, & f(x) \geq f_{\max}(x) \end{cases} \quad (10)$$

对于 s , $f_{\max}(x) = s_{\max}$, $f_{\min}(x) = d_1$; 对于 α , $f_{\max}(x) = n \cdot \alpha_{\max}$, $f_{\min}(x) = 0$; 对于 σ , $f_{\max}(x) = 45$, $f_{\min}(x) = 0$.

根据目标函数确定适应度函数, 对种群中个体 X_i 按照以下函数值进行降序排列:

$$f(X_i) = T - (fa \times pa + fb \times pb + fc \times pc) \quad (11)$$

fa 为个体 X_i 的航路转向角是否超过最大转弯角度的标志, 若是, $fa = 1$, 否则, $fa = 0$; fb 为个体的航段距离是否小于导弹最小直线飞行距离的标志, 若是, $fb = 1$, 否则, $fb = 0$; fc 为个体是否与障碍物相交的标志, 若是, $fc = 1$, 否则, $fc = 0$; pa 、 pb 和 pc 为相应的惩罚权重. 归一化后有: $pa + pb + pc = 1$.

2.4 进化公式设计

进化公式是 PSO 算法的核心. 标准 PSO 算法的基本原理为: 一个由 m 个粒子组成的种群在 $2n$ 维搜索空间中以一定速度飞行, 第 i 个粒子的位置为 $X_i = [x_{i,1}, x_{i,2}, \dots, x_{i,2n}]$, 速度为 $V_i = [v_{i,1}, v_{i,2}, \dots, v_{i,2n}]$, 其经历过的最好位置为 $P_i = [p_{i,1}, p_{i,2}, \dots, p_{i,2n}]$, 种群内所有粒子经过的最好位置为 $P_g = [p_{g,1}, p_{g,2}, \dots, p_{g,2n}]$. 第 i 个粒子在第 j 维空间上的速度和位置更新可以用一组差分方程来描述^[18]:

$$v_{i,j}^{k+1} = \omega v_{i,j}^k + c_1 r_1 (p_{i,j}^k - x_{i,j}^k) + c_2 r_2 (p_{g,j}^k - x_{i,j}^k) \quad (12)$$

$$x_{i,j}^{k+1} = x_{i,j}^k + v_{i,j}^{k+1} \quad (13)$$

其中, k 为进化代数, ω 为惯性权重, 从 0.9 到 0.4 线性递减, c_1 和 c_2 为加速常数或学习因子, 通常情况下取 $c_1 = c_2 = 2$. r_1 和 r_2 为 $[0, 1]$ 区间内服从均匀分布的随机数.

该公式模拟了每个粒子的生物惯性作用、个体认知能力及社会能力对其下一代速度的影响, 它强调每个粒子下一代的速度是由当前该粒子所掌握的知识进行矢量运算的结果^[19], 因此, 它仅描述了粒子每次进化后的运动状态, 而没有考虑粒子进化的过程. 传统的 PSO 算法在每次进化时对粒子的整个速度分量或位置分量 (即所有维的数值) 同时进行更新, 而往往粒子中的各个分量并不总是相互独立的, 即分量之间存在关联关系, 并且这种关联性可以用来提高算法搜索效率. 若对粒子中的所有分量同时进行更新, 则直接忽略了利用这种关联性改进算法性能的可能性. 下面考虑如何将粒子的分量之间的关联性映射到进化公式中.

引理 2 表达出了航路点之间 (即粒子的分量之间) 的关联关系. 在所建立的极坐标系中, O_i 的方程为

$$O_i : \rho = \frac{e_i p_i}{1 - e_i \cos(\theta - \beta_i)} \quad (14)$$

式中, β_i 为 P_i 的极角. 代入 e_i 和 p_i , 得:

$$O_i : \rho = \frac{\left(s - \sum_{i=1}^{i-1} l_{i1}\right)^2 - d_i^2}{2 \left(s - \sum_{i=1}^{i-1} l_{i1}\right) - 2d_i \cos(\theta - \beta_i)} \quad (15)$$

根据引理 2, 对于 P_j ($j \geq i + 1$), 则应有:

$$P_j \in \left\{ (\rho, \theta) \mid \rho \leq \frac{e_i p_i}{1 - e_i \cos(\theta - \beta_i)} \right\} \quad (16)$$

映射到粒子所存在的解空间 \mathbf{R}^{2n} 中, 则应有:

$$x_{i,j} \leq \frac{e_{j1} p_{j1}}{1 - e_{j1} \cos(x_{i,j-n} - \beta_{j1})}, \quad j = n + 1, \dots, 2n; j1 \leq j - n - 1 \quad (17)$$

定义 7. 由于功能区域簇的作用, 粒子的分量之间存在关联关系, 为了体现出这种关联性, 采用递归思想对各分量按顺序逐步进行更新, 将这种策略称为分步递归进化策略.

定义 8. 使用分步递归进化策略时, 粒子中后一个分量的更新位置受到前一个分量更新位置的约束. 粒子基于自身的约束能力, 使得后一个分量的更新位置总是在前一个分量限定的准确范围内. 将能够使粒子具备这种约束能力的项称为功能区域簇实时约束 (Operational area cluster real-time restriction, OACRR) 约束项.

定义 9. 当粒子的分量 $x_{i,j}^k$ ($j = n + 1, \dots, 2n$) 的下一代不满足式 (17) 时, 称其下一代分量为极径伪分量, 记为 $x_{i,j}^{k+1}$; 它所对应的分量 $x_{i,j-n}^{k+1}$ 称为极角伪分量.

OACRR 约束项的作用就是控制粒子分量的更新速度, 使得其每次更新后的位置都满足式 (17), 当不满足时, 则对速度进行动态实时调整, 将 $x_{i,j}^{k+1}$ ($x_{i,j-n}^{k+1}$) 逐步往某个合理的方向更新, 使其限定在 $x_{i,j-1}^{k+1}$ ($x_{i,j-n-1}^{k+1}$) 所允许的范围内, 从而加速粒子的收敛. 根据 PSO 算法的特点, 不妨尝试作如下定义:

定义 10. 黄金分割位置 $gp_{i,j}^k$. 在 OACRR 约束项中, 将极径伪分量 $x_{i,j}^{k+1}$ 逐步更新至满足式 (17) 的新位置的项, 该项也将 $x_{i,j}^{k+1}$ 所对应的极角伪分量 $x_{i,j-n}^{k+1}$ 更新至相应的新位置, 这些新位置处于 $x_{i,j-1}^{k+1}/2$ 与 $x_{i,j}^{k+1}$ (或 $x_{i,j-n-1}^{k+1}$ 与 $x_{i,j-n}^{k+1}$) 的黄金分割点. 其表达式为

$$gp_{i,j}^k = \begin{cases} 0.382x_{i,j}^{k+1}, & j = 1 \\ 0.382x_{i,j}^{k+1} + 0.618x_{i,j-1}^{k+1}, & j = 2, \dots, n \\ 0.382x_{i,j}^{k+1} + 0.618d_1, & j = n + 1 \\ 0.382x_{i,j}^{k+1} + 0.309x_{i,j-1}^{k+1}, & j = n + 2, \dots, 2n \end{cases} \quad (18)$$

$gp_{i,j}^k$ 的设计主要考虑以下两点. 首先, 将极径伪分量 $x_{i,j}^{k+1}$ 和极角伪分量 $x_{i,j-n}^{k+1}$ 向所对应的功能区域椭圆的中心方向更新, 这样不影响其在方向上的随机性和多样性; 其次, 黄金分割具有经典的统计实验意义^[20], 黄金分割距离的引入也不影响其在距离上的随机性和多样性.

于是, OACRR-PSO 算法的速度更新公式为

$$v_{i,j}^{k+1} = b_1\omega v_{i,j}^k + b_1c_1r_1(p_{i,j}^k - x_{i,j}^k) + b_1c_2r_2(p_{g,j}^k - x_{i,j}^k) + b_2c_3r_3(gp_{i,j}^k - x_{i,j}^k) \quad (19)$$

位置更新公式不变. 式中, c_3 为收缩因子, 取值 $c_3 = 2$, r_3 为 $[0, 1]$ 区间内服从均匀分布的随机数, b_1 和 b_2 为布尔型变量, 若粒子分量满足式 (17), b_1 取值为 1, b_2 取值为 0; 否则, b_1 取值为 0, b_2 取值为 1.

从式 (19) 可以看出, OACRR-PSO 算法具有以下优势: 1) 布尔型变量只是用来控制算法的执行步骤, 除此以外, 算法没有增加任何需要调整的参数; 2) OACRR 约束项是在粒子分量不满足相应约束时才发挥作用, 因此, 不会增加算法的计算复杂度; 3) 由于对粒子中每一维分量进行合理约束, 使得种群中不会出现适应值偏低的个体, 这就提高了每一代种群中个体的整体质量, 只需经过较少的迭代次数就能够得到满意的解, 从而节省计算时间, 显著

地提高算法的收敛性能; 4) 虽然粒子分量被约束的范围不断减小, 但是每一步的约束范围是实时解空间的全集, 算法解空间的范围并没有减小, 相反, 粒子分量到达每个可行解的概率都增大了, 因此, 不但不会破坏粒子的多样性, 而且能够在一定程度上保持粒子多样性, 从而有效避免早熟收敛现象的产生; 5) 由于从整体上提高了每一代种群中个体的平均适应值, 并且算法实时解空间不断减小, 此消彼长, 这就提高了粒子寻找到全局最优解的概率, 极大地提高算法的全局搜索性能.

2.5 算法步骤

综上所述, OACRR-PSO 算法步骤如下:

步骤 1. 选择合理的种群大小 m , 粒子维数 n 及最大迭代次数 k_{\max} 等算法参数. 并令 $i = 1, j = 1, k = 1$.

步骤 2. 判断所有粒子是否均初始化完毕, 若是, 则转步骤 5; 否则, 计算 e, p 的初始值, 转步骤 3.

步骤 3. 在解空间范围内对 $x_{i,j}$ 进行随机初始化, 在 $x_{i,j+n}$ 的约束范围内对 $x_{i,j+n}$ 进行随机初始化.

步骤 4. 判断第 i 个粒子中的所有分量是否都初始化完毕, 若是, 则 $i++$, 转步骤 2; 否则计算 $x_{i,j+n+1}$ 的约束, $j++$, 转步骤 3.

步骤 5. 计算每条航路的适应度函数值 $f(X_i^k)$.

步骤 6. 更新个体极值 $p_{i,j}^k$ 和全局极值 $p_{g,j}^k$.

步骤 7. 令 $i = 1, j = 1$.

步骤 8. 判断所有粒子是否均更新完毕, 若是, 则 $k++$, 并转步骤 11; 否则转步骤 9.

步骤 9. 令 $b_1 = 1, b_2 = 0$, 根据式 (19) 对 $x_{i,j}^k$ 和 $x_{i,j+n}^k$ 进行更新, 判断更新后 $x_{i,j+n}^{k+1}$ 是否满足式 (17), 若不满足, 表明更新后位置为伪分量, 则令 $b_1 = 0, b_2 = 1$, 根据式 (19) 重新进行更新, 直至更新后位置满足式 (17); 否则转步骤 10.

步骤 10. 判断第 i 个粒子中的所有分量是否都更新完毕, 若是则 $i++$, 转步骤 8; 否则, 计算 $x_{i,j+n+1}^{k+1}$ 的约束, $j++$, 转步骤 9.

步骤 11. 判断 $k \leq k_{\max}$ 是否成立, 若是则转步骤 5; 否则输出结果.

图 4 给出了更接近计算机编程实现的 OACRR-PSO 算法流程.

3 仿真实验及分析

3.1 实验设计和数据准备

为了测试本文提出的 OACRR-PSO 算法的性能, 首先引入一种介于标准 PSO 和 OACRR-PSO 算法之间的算法, 称之为 OAR-PSO 算法. OAR-PSO 算法与标准 PSO 算法的不同之处在于解空间

的范围, OAR-PSO 算法的解空间是功能区域 OA 的映射, 因此相对于标准 PSO 而言, OAR-PSO 算法合理压缩并界定了搜索空间的范围. OAR-PSO 算法与 OACRR-PSO 算法的主要不同之处在于粒子进化过程, OAR-PSO 算法不采用分步递归进化策略, 其进化公式与标准 PSO 算法一样, 而不含 OACRR 约束项, 因此, 它无法实时限定粒子位置分量的准确更新范围. 三种算法的比较如表 1 所示.

在相同的实验条件下, 分别采用标准 PSO、OAR-PSO 和 OACRR-PSO 三种算法进行仿真实验. 硬件环境为 AMD Athlon X2 CPU 2.1 GHz, 3GB 内存的 PC 机, 运行环境为 Windows XP Professional, 编程环境为 Matlab 6.5. 反舰导弹航路性能参数: 最大有效射程 300 km, 最大转向角 90° , 由发射点转入平飞距离 20 km, 相邻航路点间最小距离 20 km, 自导距离 20 km. 考虑 $300 \text{ km} \times 160 \text{ km}$ 的作战海区, 发射点坐标 (0, 0), 目标点坐标 (270, 0), 障碍物有 5 个, 其位置和安全半径分别为 (60, 10, 8), (100, 8.9, 10), (140, 0, 8), (170, 20, 12), (190, -15, 13) (括号中前两项为障碍

物位置坐标, 第 3 项为安全半径, 单位: km). ω_s 取值为 0.6, ω_α 和 ω_σ 取值都为 0.2, pa 和 pb 都取值为 0.4, pc 取值为 0.2. 综合考虑航路质量 (精度) 与规划时间, 通过反复实验, 设定种群大小为 200, 航路转向点个数为 3, 即粒子维数为 6, 最大迭代次数为 200.

3.2 算法的性能测度和实验结果

本文中, 算法性能测试采用以下 4 个评价指标^[19,21-22]: 1) 平均最优适应值 f^{mean} , 可用来衡量算法的寻优质量和精度; 2) 最优适应值的标准差 σ_f , 来衡量算法的鲁棒性和稳定性; 3) 平均达标率 (Mean standard ratio, MSR): 用 MSR 表示, 是指在多次运行某算法过程中, 算法到达规定阈值的实验次数占总实验次数的百分比, 用来衡量算法的可靠性; 4) 平均迭代次数 (Mean iteration times, MIT), 用 MIT 表示, 是指所得适应值结果满足规定阈值时, 算法所需平均迭代次数, 用来衡量算法的收敛速度. 对最大迭代次数内仍不能满足规定阈值的实验, 不参与此测度计算.

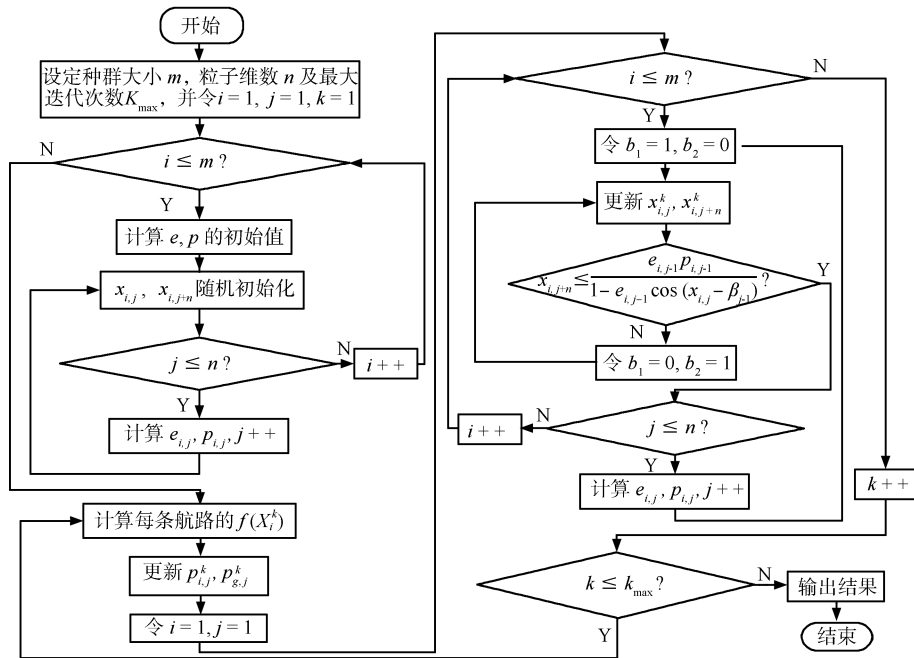


图 4 反舰导弹航路规划 OACRR-PSO 算法流程图

Fig. 4 Flowchart of OACRR-PSO algorithm for ASMPP

表 1 PSO, OAR-PSO 和 OACRR-PSO 算法比较

Table 1 Comparison of PSO, OAR-PSO, and OACRR-PSO algorithms

算法	解空间范围	进化机制	粒子分量间的关系	粒子分量超出解空间的处理方法
PSO	矩形区域	粒子整体进化	不考虑关联	以边界点置换
OAR-PSO	功能区域	粒子整体进化	不考虑关联	以边界点置换
OACRR-PSO	功能区域簇	分步递归进化	考虑关联	向黄金分割位置的方向更新

每种算法进行 30 次仿真实验, 分别记录每次实验每次迭代结束时, 生成的最优航路的适应值以及每次实验达到规定阈值所需的迭代次数. 记第 i 次实验的第 j 次迭代后生成的最优航路的适应值为 $f_{i,j}$, 则每种算法第 j 次迭代后的平均最优适应值和迭代 200 次以后平均最优适应值的标准差分别定义为

$$f_j^{\text{mean}} = \frac{\sum_{i=1}^{30} f_{i,j}}{30} \quad (20)$$

$$\sigma_f = \sqrt{\frac{\sum_{i=1}^{30} (f_{i,200} - f_{200}^{\text{mean}})^2}{30 - 1}} \quad (21)$$

将 30 次实验中达到规定阈值的实验次数对总实验次数 30 取平均, 得到 MSR ; 记第 i 次实验达到规定阈值所需的迭代次数为 IT_i , 迭代 200 次时仍不能满足规定阈值的实验次数为 j , 则 MIT 为

$$MIT = \frac{\sum_{i=1}^{30-j} IT_i}{30 - j} \quad (22)$$

经整理后的实验结果数据如表 2 所示.

表 2 PSO, OAR-PSO 和 OACRR-PSO 算法实验结果比较

Table 2 Comparison of experiment results of PSO, OAR-PSO, and OACRR-PSO

算法	f_{200}^{mean}	σ_f	MSR (%)	MIT
PSO	0.7338	6.3109 E-02	30	125
OAR-PSO	0.7861	6.6059 E-02	37	117
OACRR-PSO	0.8865	3.6056 E-04	100	4

三种算法的平均最优适应值的收敛曲线如图 5 所示.

三种算法分别经过 30 次仿真实验所得到的三条最优航路如图 6 所示.

3.3 算法性能分析与比较

对实验结果进行比较分析, 可以得到以下结论:

1) 从 f^{mean} 的角度比较来说, 采用 OACRR-PSO 算法得到的 f^{mean} 比采用其他两种算法得到的 f^{mean} 都要大, 说明它的求解质量和精度相对要高, 并且它总能获得全局最优解, 这表明 OACRR-PSO 算法的全局搜索能力最优;

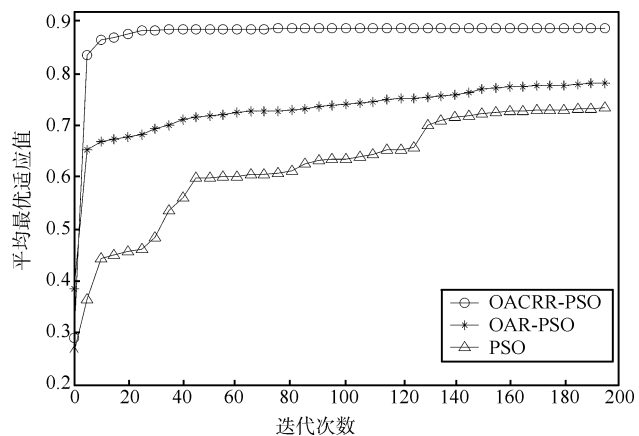


图 5 PSO, OAR-PSO 和 OACRR-PSO 算法的平均最优适应值收敛曲线

Fig. 5 The convergence curves of mean best fitness of PSO, OAR-PSO, and OACRR-PSOP

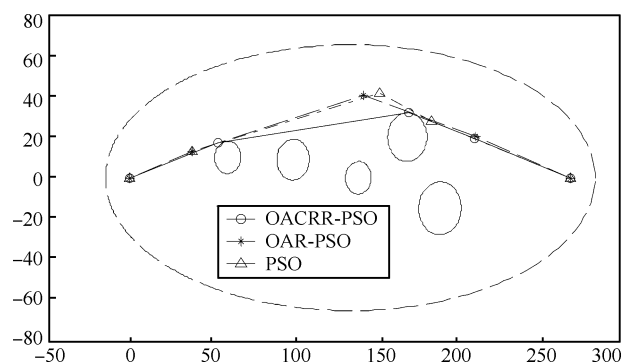


图 6 PSO, OAR-PSO 和 OACRR-PSO 算法的实验结果

Fig. 6 Experiment results of PSO, OAR-PSO, and OACRR-PSO

2) 从 σ_f 的角度比较来说, OACRR-PSO 算法得到的 σ_f 比其他两种算法得到的 σ_f 都要小, 并且减小了两个数量级, 表明 OACRR-PSO 算法的稳定性得到了很大的提高, 如此小的 σ_f 说明其每次实验几乎都收敛于相同解 (全局最优解);

3) 从 MSR 的角度比较来说, OACRR-PSO 算法的 MSR 明显大于其他两种算法, 并且达到了 100%, 表明 OACRR-PSO 算法的可靠性较其他两种方法要好, 并且每次实验都达到了规定的阈值, 这也说明了 OACRR-PSO 算法的整体性能较好;

4) 从 MIT 的角度比较来说, OACRR-PSO 算法的 MIT 明显小于其他两种算法, 表明 OACRR-PSO 算法的收敛速度较其他两种方法要好;

5) OAR-PSO 算法的整体性能比标准 PSO 算法略好, 这是因为 OAR-PSO 算法的解空间较小, 减小了算法搜索空间, 但是它在进化机制上与标准 PSO 算法完全一样, 所以算法性能不会有大的提高;

6) OACRR-PSO 算法的解空间与 OAR-PSO 算法的解空间完全一样,但是它考虑了粒子分量间的关联性,采用分步递归进化策略来改进算法进化机制,使得算法各方面的性能得到了大幅度的提升,这充分说明了分步递归进化策略的有效性.

4 结束语

本文从反舰导弹航路规划的问题背景出发,将功能区域的几何学渐变规律抽象成功能区域簇的概念,并引入标准 PSO 算法提出了一种功能区域簇实时约束的 PSO 算法 (OACRR-PSO). 为了解决传统 PSO 算法的早熟收敛问题,在算法的优化过程中,考虑到粒子中分量之间的关联性,算法并不是对粒子的整个速度分量或位置分量 (即所有维的数值) 同时进行更新,而是引入一种分步递归进化策略对粒子的分量按顺序逐步进行更新. 在粒子的更新过程中,使用功能区域簇来实时限定粒子位置分量的准确更新范围,使得算法搜索空间逐步减小. 仿真实验结果表明,分步递归进化策略能够非常显著地提高算法各方面的优化性能,包括全局搜索性能、收敛性能和稳定性. 需要指出的是,粒子分量间关联关系的算法化和公式化是一个开放式问题,关联关系不同,算法思路和进化公式就不同,本文从一个全新的角度改进了 PSO 算法,实现了问题背景和优化算法之间的良好结合.

References

- Liu Gang, Lao Song-Yang, Tan Dong-Feng. Converse path planning for anti-ship missiles based on operational area. *Systems Engineering and Electronics*, 2011, **33**(4): 799–805 (刘钢, 老松杨, 谭东风. 基于功能区域的反舰导弹逆向航路规划. 系统工程与电子技术, 2011, **33**(4): 799–805)
- Liu Gang, Lao Song-Yang, Tan Dong-Feng, Zhou Zhi-Chao. Fast graphic converse method for path planning of anti-ship missile. *Journal of Ballistics*, 2011, **23**(2): 52–56 (刘钢, 老松杨, 谭东风, 周智超. 反舰导弹航路规划图形化快速逆推方法. 弹道学报, 2011, **23**(2): 52–56)
- Ren Min, Huo Xiao-Hua. The real-time path planning method of UAV based on asynchronous bi-precision scrolled window. *Science China: Information Science*, 2010, **40**(4): 561–568 (任敏, 霍霄华. 基于异步双精度滚动窗口的无人机实时航迹规划方法. 中国科学: 信息科学, 2010, **40**(4): 561–568)
- Kevin P B. High-Fidelity Real-Time Trajectory Optimization for Reusable Launch Vehicles [Ph.D. dissertation], Naval Postgraduate School, California, USA, 2006
- Zhu Qing-Bao. Ant algorithm for path planning of mobile robot in a complex environment. *Acta Automatica Sinica*, 2006, **32**(4): 586–593 (朱庆保. 复杂环境下的机器人路径规划蚂蚁算法. 自动化学报, 2006, **32**(4): 586–593)
- Ortega L M, Rueda A J, Feito F R. A solution to the path planning problem using angle preprocessing. *Robotics and Autonomous Systems*, 2009, **58**(1): 27–36
- Zhu Yi, Zhang Tao, Song Jing-Yan. Study on the local minima problem of path planning using potential field method in unknown environments. *Acta Automatica Sinica*, 2010, **36**(8): 1122–1130 (朱毅, 张涛, 宋靖雁. 未知环境下势场法路径规划的局部极小问题研究. 自动化学报, 2010, **36**(8): 1122–1130)
- Wang Wei-Ping, Liu Juan. Introduction to unmanned air vehicle route planning methods. *Flight Dynamics*, 2010, **28**(2): 6–10 (王维平, 刘娟. 无人飞行器航迹规划方法综述. 飞行力学, 2010, **28**(2): 6–10)
- Melchior P, Orsoni B, Lavialle O, Poty A, Oustaloup A. Consideration of obstacle danger level in path planning using A* and fast-marching optimisation: comparative study. *Signal Processing*, 2003, **83**(11): 2387–2396
- Zheng C, Li L, Xu F, Sun F, Ding M. Evolutionary route planner for unmanned air vehicles. *IEEE Transactions on Robotics*, 2005, **21**(4): 609–620
- Fu Yang-Guang, Zhou Cheng-Ping, Ding Ming-Yue. 3-D route planning based on hybrid quantum-behaved particle swarm optimization. *Journal of Astronautics*, 2010, **31**(12): 2657–2664 (傅阳光, 周成平, 丁明跃. 基于混合量子粒子群优化算法的三维航迹规划. 宇航学报, 2010, **31**(12): 2657–2664)
- Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of the 1995 IEEE International Conference on Neural Networks*. Perth, Australia: IEEE, 1995. 1942–1948
- Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: *Proceedings of the 6th International Symposium on Micro Machine and Human Science*. Piscataway, USA: IEEE, 1995. 39–43
- Shi Y H, Eberhart R. A modified particle swarm optimizer. In: *Proceedings of IEEE Congress on Evolutionary Computation*. Piscataway, USA: IEEE, 1998. 69–73
- Ni Tian-Quan, Wang Jian-Dong, Liu Yi-An. Application of particle swarm algorithm in route planning of UAV. *Systems Engineering and Electronics*, 2011, **33**(4): 806–810 (倪天权, 王建东, 刘以安. 交叉粒群算法在无人机航路规划中的应用. 系统工程与电子技术, 2011, **33**(4): 806–810)
- Ji Zhen, Zhou Jia-Rui, Liao Hui-Lian, Wu Qing-Hua. A novel intelligent single particle optimizer. *Chinese Journal*

of Computers, 2010, **33**(3): 556–561

(纪震, 周家锐, 廖惠连, 吴青华. 智能单粒子优化算法. 计算机学报, 2010, **33**(3): 556–561)

- 17 Van den Bergh F. An Analysis of Particle Swarm Optimizers [Ph.D. dissertation], University of Pretoria, South Africa, 2002

- 18 Shen Yuan-Xia, Wang Guo-Yin, Zeng Chuan-Hua. Correlative particle swarm optimization model. *Journal of Software*, 2011, **22**(4): 695–708

(申元霞, 王国胤, 曾传华. 相关性粒子群优化模型. 软件学报, 2011, **22**(4): 695–708)

- 19 Tang Su-Yan, Zhu Yi-Fan, Zhang Wei, Li Qun. Particle swarm optimization algorithm based on creative thinking. *Control and Decision*, 2011, **26**(8): 1181–1186

(唐苏妍, 朱一凡, 张伟, 李群. 一种基于创造性思维的粒子群优化算法. 控制与决策, 2011, **26**(8): 1181–1186)

- 20 Egrioglu E, Aladag C H, Basaran M A, Yolcu U, Uslu V R. A new approach based on the optimization of the length of intervals in fuzzy time series. *Journal of Intelligent and Fuzzy Systems*, 2011, **22**(1): 15–19

- 21 Zhang Yong, Gong Dun-Wei, Zhang Wan-Qiu. A simplex method based improved particle swarm optimization and analysis on its global convergence. *Acta Automatica Sinica*, 2009, **35**(3): 289–298

(张勇, 巩敦卫, 张婉秋. 一种基于单纯形法的改进微粒群优化算法及其收敛性分析. 自动化学报, 2009, **35**(3): 289–298)

- 22 Alfi A. PSO with adaptive mutation and inertia weight and its application in parameter estimation of dynamic systems. *Acta Automatica Sinica*, 2011, **37**(5): 541–549



刘钢 国防科学技术大学信息系统与管理学院博士研究生. 2005 年获得海军兵种指挥学院学士学位. 主要研究方向为军事运筹学, 指挥信息系统. 本文通信作者. E-mail: Snailg705@21cn.com

(LIU Gang Ph.D. candidate at the School of Information Systems and Management, National University of

Defense Technology. He received his bachelor degree from Naval Arms Command Institute in 2005. His research interest covers military operational research and command information systems. Corresponding author of this paper.)



老松杨 国防科学技术大学信息系统与管理学院教授. 主要研究方向为军事运筹学, 指挥信息系统.

E-mail: laosongyang@vip.sina.com

(LAO Song-Yang Professor at the School of Information Systems and Management, National University of

Defense Technology. His research interest covers military operational research and command information systems.)



袁灿 海军工程大学核能与工程系博士研究生. 2009 年获得武汉大学遥感学院学士学位. 主要研究方向为维修工程. E-mail: yuancan2008@163.com

(YUAN Can Ph.D. candidate in the Department of Nuclear Science and Engineering, Naval University of Engineering. He received his bachelor de-

gree from the School of Remote Sensing and Information Engineering, Wuhan University in 2009. His main research interest is maintenance engineering.)



侯绿林 国防科学技术大学信息系统与管理学院博士研究生. 2008 年获得国防科学技术大学信息系统与管理学院学士学位. 主要研究方向为军事运筹学, 指挥信息系统. E-mail: houlvlin@126.com

(HOU Lv-Lin Ph.D. candidate at the School of Information Systems and Management, National University of

Defense Technology. He received his bachelor degree from the School of Information Systems and Management, National University of Defense and Technology in 2008. His research interest covers military operational research and command information systems.)



谭东风 国防科学技术大学信息系统与管理学院教授. 主要研究方向为军事运筹学, 指挥信息系统.

E-mail: tan.dongfeng@gmail.com

(TAN Dong-Feng Professor at the School of Information Systems and Management, National University of

Defense Technology. His research interest covers military operational research and command information systems.)