

Multi-aircraft Flight Conflict Resolution Based on Dynamic Particle Swarm Optimisation

Yonglin Tu¹ Bingxin Guo³ Haoli Xu^{1,2,4*} Xing Yang^{1,2} Haoqi Gao^{1,2}

1. State Key Laboratory of Pulsed Power Laser Technology, School of Electronic Resistance, National University of Defense Technology, Hefei, Anhui, 230037, China
2. Jianghuai Frontier Technology Collaborative Innovation Center, Hefei, Anhui, 230000, China
3. School of Systems Engineering, National University of Defense Technology, Changsha, Hunan, 410073, China
4. Electronic Restriction Technology in Anhui Provincial Laboratory, Hefei, Anhui, 230037, China

Abstract

With the increase in the demand for air transportation, various types of aircraft usually need to be deployed at the same time and space, which leads to a prominent contradiction between the growing demand for airspace and the shortage of airspace resources. In this paper, based on the dynamic particle swarm optimization algorithm, a model for the adjustment and optimization of airspace demand of various aircraft is established, and the experiment simulates and optimizes hundreds of conflicting airspace requirements through fewer iterations, even if the fitness adjustment is controlled between two decimal places, and verifies the feasibility and advantages of the model compared with the traditional particle swarm optimization algorithm. The model can realize global optimization between small fitness differences, and the optimization of the scheme can be achieved through a few iterations. It provides a reference scheme for solving the flight requirement optimization problem with multiple local optimal solutions and complex nonlinearity.

Keywords

airspace conflicts; airspace demand mediation; dynamic particle swarm optimization algorithm; aircraft multi-objective optimization

基于动态粒子群优化的多机飞行冲突解脱

屠俑霖¹ 郭秉鑫³ 许颢砾^{1,2,4*} 杨星^{1,2} 高皓琪^{1,2}

1. 国防科技大学电子对抗学院脉冲功率激光技术国家重点实验室, 中国·安徽 合肥 230037
2. 江淮前沿技术协同创新中心, 中国·安徽 合肥 230000
3. 国防科技大学系统工程学院, 中国·湖南 长沙 410073
4. 电子制约技术安徽省实验室, 中国·安徽 合肥 230037

摘要

随着航空空域需求的增加, 各型各类航空器通常需要在同一时空部署运用, 由此导致空域需求不断增长与空域资源不足之间矛盾突出。论文依托动态粒子群优化算法, 建立了一种用于各种航空器空域需求调整优化的模型, 实验通过较少的迭代, 即使适应度调整控制在两位小数之间, 模拟优化了上百个存在冲突的空域需求, 同时对比传统的粒子群算法, 验证了该模型的可行性和优势。该模型可实现在微小适应度差值间全局寻优, 通过少数迭代即可实现方案的优化。对于具有多个局部最优解和复杂非线性的飞行需求优化问题的解决提供了参考方案。

关键词

空域冲突; 空域需求调解; 动态粒子群算法; 航空器多目标优化

【基金项目】江淮前沿技术协同创新中心追梦基金课题“多谱段三维智能XXX机理研究”(项目编号: 2023-ZM01D006)联合资助; 国家自然科学基金青年科学基金项目“基于OMCI阵列高精度监测的基岩海岛淡水体形成与时空演化机理研究”(项目编号: 62305389)。

【作者简介】屠俑霖(2002-), 男, 中国云南曲靖人, 在读本科生, 从事通信工程研究。

【通讯作者】许颢砾(1993-), 男, 中国江西赣州人, 博士, 讲师, 从事军事智能研究。

1 引言

随着飞行技术的不断发展, 各型各类航空器进行航空空域或执行其他任务时的空域需求矛盾日益突出, 及时高效地对大规模用空需求进行冲突检测和方案优化, 是避免航空器发生事故、保证航空器飞行效率和实现空域复用亟待解决的问题。现有的冲突解脱模型, 实时性差, 对突发情况处理能力弱; 模型训练时间长, 收敛速度慢, 迭代次数冗长^[1]; 且研究方向主要集中于静态环境中, 不适用于动态目标的冲突解脱优化, 实用性不高, 解决非线性优化问题能力薄弱, 对复杂环境和变化环境优化问题难以适应^[2,3]。为了弥补以

上缺陷, 论文建立了基于动态粒子群优化算法 (Dynamic Particle Swarm Optimization, DPSO) 的飞行冲突解脱模型, 该模型基于群体智能和演化优化的原理, 通过模拟鸟群或鱼群等生物群体的行为, 来实现冲突解脱和方案优化。

DPSO 算法的全局搜索能力强, 收敛速度快, 在复杂环境中的稳定性和可靠性明显, 解决多维问题时具有明显优势^[45]。在此基础上, 论文拟借助 DPSO 算法, 考虑航空器在飞行过程中的环境条件和部分突发因素, 结合实际空域空航中航空器必要的空域需求, 在允许调整的空域范围内, 遵循空域调整的基本准则, 考虑飞行成本、飞行效率等, 构建了空域需求冲突的解脱模型。算法中学习因子、惯性权重等算法参数随环境的变化而变化, 随每个粒子的优化程度和全局粒子的寻优程度而变化, 使得迭代次数大幅减少^[6], 既有效提高了模型的效率, 又通过动态的权重机制对搜索过程中的噪声表现出一定程度的抑制和适应, 表明该模型在复杂环境中运行的可靠性和稳定性。

图 1 表明了论文的解决思路。

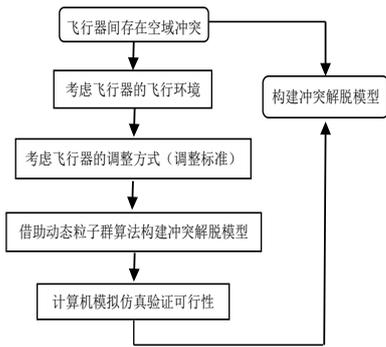


图 1 冲突解脱方案

2 动态粒子群算法

2.1 粒子群算法描述

在 D 维的搜索空间中, 有 n 个粒子, 每个粒子代表一个解。第 i 个粒子的位置为: $X_{id} = (x_{i1}, x_{i2}, \dots, x_{iD})$ 在空域解脱时, 粒子的位置需要包括四个维度, 分别是可调整的高度、时间和水平位置, 以及不可调整的优先级。

第 i 个粒子的速度 (粒子移动的距离和方向) 为: $V_{id} = (v_{i1}, v_{i2}, \dots, v_{iD})$, 第 i 个粒子搜索到的最优位置 (个体最优解) 为: $P_{id, pbest} = (p_{i1}, p_{i2}, \dots, p_{iD})$, 群体搜索到的最优位置 (群体最优解) 为: $P_{d, pbest} = (p_{1, pbest}, p_{2, pbest}, \dots, p_{D, pbest})$, 第 i 个粒子搜索到的最优位置的适应值 (优化目标函数的值) 为 f_p , 群体搜索到的最优位置的适应值为 f_g 。

在此, 建立速度迭代公式和位置更新公式:

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 r_1 (p_{id, pbest}^k - x_{id}^k) + c_2 r_2 (p_{d, gbest}^k - x_{id}^k)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^k$$

其中, N 为粒子群规模, $i(i = 1, 2, \dots, N)$ 表示粒子序号;

D 为粒子维度; $d(d = 1, 2, \dots, D)$ 为粒子维度序号; k 为迭代次数; ω 为惯性权重; c_1 和 c_2 分别为个体和群体学习因子; r_1 和 r_2 为区间 $[0, 1]$ 内的随机数, 用来增加搜索的随机性; v_{id}^k 为粒子在第 i 次迭代中第 d 维的速度向量; x_{id}^k 为粒子在第 i 次迭代中第 d 维的位置向量; $p_{id, pbest}^k$ 为粒子 i 在第 k 次迭代中第 d 维历史最优位置, 即在第 k 次迭代后, 第 i 个粒子个体搜索得到的最优解; $p_{d, pbest}^k$ 为群体在第 k 次迭代中第 d 维历史最优位置, 即在第 k 次迭代后, 整个粒子群体的最优解。

引入响应机制, 每次迭代时对于敏感粒子的适应度值 f_{pi} , 计算相邻 2 次迭代适应度值差值 Δf_{pi} , 对所有差值的绝对值求和 $F^{[7]}$ 。

$$\forall f_i = f(k+1) - f(k)$$

$$F = \sum_{i=1}^n |f_i|$$

当 F 值不为 0 时, 认为外部环境已发生变化, 设定响应阈值 $F_{\text{阈}}$, 当 F 值超过阈值 $F_{\text{阈}}$ 时触发响应, 响应机制为按一定比例重新初始化粒子和粒子速度。

2.2 动态粒子群算法中的改进

传统粒子群算法下, 每个粒子通过跟踪自身记忆的个体最优和种群记忆的全局最优以逐渐逼近更优位置。但在动态环境下, 记忆的个体最优位置和全局最优位置对应的适应度值是变化的, 粒子可能陷入对先前环境的寻优陷阱。因此, 传统粒子群算法难以在动态环境下有效逼近最优位置。为了跟踪整体动态的极值变化, 对粒子群算法进行以下动态处理^[8]。

2.2.1 生成异步学习因子

在该算法中, 参数学习因子 (Learning Factor) 包括个体学习因子 (Cognitive Factor) 和社会学习因子 (Social Factor), 它们分别控制了粒子在个体最优解和全局最优解之间的权衡。传统粒子群算法中的学习因子无法根据问题的特点和搜索状态进行自适应调整, 由此容易导致粒子在搜索过程过于保守或过于冒进。如果学习因子设置得过小, 粒子容易陷入局部最优解而难以跳出; 如果设置得过大, 粒子可能会过于冒进, 导致搜索范围过大而效率低下。学习因子的大小还直接影响到粒子的速度和跳跃能力。如果学习因子设置得不合理, 粒子可能会在搜索空间来回震荡, 导致算法收敛速度大幅降低。为了克服以上弊端, 论文的 DPSO 算法采取粒子在寻优过程中生成异步学习因子的策略:

$$c_1 = c_{1\max} - (c_{1\max} - c_{1\min}) * k / k_{\max}$$

$$c_2 = c_{2\max} - (c_{2\max} - c_{2\min}) * k / k_{\max}$$

其中, $c_{1\max}$ 、 $c_{2\max}$ 为学习因子 1 (个体学习因子) 和 2 (社会学习因子) 取值的最大值; $c_{1\min}$ 、 $c_{2\min}$ 为学习因子 1 和 2 取值的 minimum; k 、 k_{\max} 为当前和最大的迭代次数^[8]。

2.2.2 生成非线性衰落惯性权重

惯性权重 (Inertial weighting) 控制了粒子的更新速度, 可用来平衡粒子的历史最佳位置和全局最佳位置之间的权

衡关系,对于不同优化问题需要不同的权衡关系,一旦权重无法灵活地适应这种变化就容易导致算法在某些问题上效果不佳。传统算法中,固定的惯性权重严重限制了粒子的速度和搜索能力,也同样存在容易使粒子在搜索过程中过于保守或过于冒进的问题。如果权重设置得过小,粒子可能会过于保守,容易陷入局部最优解而难以跳出;如果权重设置得过大,会导致粒子的搜索范围过大而效率低下,直接制约算法的收敛速度。在DPSO算法中生成非线性衰落的惯性权重,是惯性权重伴随着迭代次数的变化进而发生相应的调整:

$$\omega(k) = \omega_{\max} - \frac{k \cdot (\omega_{\max} - \omega_{\min})}{k_{\max}}$$

其中,惯性因子 $\omega(k)$ 为迭代至第 k 次的; ω_{\max} 为惯性因子最大值; ω_{\min} 为惯性因子最小值。控制惯性因子发生线性递减:在初期阶段,使惯性因子较大,保证算法的搜索速度加快并向全局最优值逐步靠近,可以快速地摆脱局部极值的束缚;在后期阶段,惯性因子逐步靠近最小值,这样做可以使模型在全局最优解的基础上完成更加精确的搜索,以短小的步距长度寻遍包含全局最优点,这些数据会在所在区域产生所有值,最终找到优化完的数值^[9]。

2.2.3 生成自适应权重拓扑结构

粒子群算法中的拓扑结构决定了粒子之间的信息传递方式,通过共享个体最佳位置和全局最佳位置的信息,粒子可以相互引导和协作,加速搜索过程,提高算法的全局搜索能力,不同的拓扑结构可以影响粒子之间的相互影响程度^[10]。DPSO算法引入自适应权重拓扑结构(Adaptive Weight Topology):在自适应权重拓扑结构中,每个粒子与其他粒子之间的连接强度根据其适应度值进行调整。生成自适应权重拓扑结构的方法如下:

- ①计算每个粒子的适应度值。
- ②根据粒子的适应度值,计算每个粒子与其他粒子之间的连接权重。一种常见的方法是使用适应度值的倒数作为连接权重,即适应度值越大,连接权重越小,反之亦然。
- ③粒子之间的连接关系可以通过邻接矩阵来表示,其中矩阵元素表示两个粒子之间的连接权重。

以上,DPSO将多种策略融合到算法中,如动态参数调整、自适应权重拓扑结构、多目标优化等,这些策略的融合使得DPSO能够在不同的问题领域和优化目标下具有更强的适应性和鲁棒性^[11,12]。

2.3 算法流程

DPSO算法的流程如下:①初始化粒子群 n ,每个粒子代表一个可能的解,即一组修改后的时空轨迹,包含高度、水平和时间,以及优先级;②根据给定的冲突检测规则和冲突评估指标,初始化参数,初始化粒子速度 V 。计算每个粒子的适应度值,衡量其解决冲突的能力;③根据当前的位置和速度,以及个体历史最优和群体历史最优位置,通过位置和速度更新公式更新粒子的位置和速度;④根据更新后的粒

子位置,对存在冲突的时空轨迹进行修正,同时动态地调整粒子的权重;⑤根据修正后的时空轨迹,重新计算每个粒子和群体的适应度值 f_p 和 f_g ,并调整拓扑结构;⑥对于每个粒子,更新其个体历史最优位置和群体历史最优位置;⑦判断是否满足终止条件,如达到最大迭代次数或适应度值收敛等;⑧若满足条件终止,则算法结束,否则回到③。

3 模型介绍

3.1 冲突解脱描述

在同一时间段内,如果有两个及以上飞行用户在使用空域时存在区域交叠或距离过近的情况,有可能引发危险接近、空中相撞,甚至是误击误伤等事故。论文中的空域需求采用三类参数表征:申请高度范围 $H = [h_1, h_2]$ 、申请位置集合 $S = \{(x_i, y_i) | i = 1, 2, \dots, n\}$ 、申请时间范围 $T = [t_1, t_2]$ 和空域需求调整的优先级。对于存在冲突的两个空域需求需要在控制范围内调整前三个参数(高度、水平和时间)中的一个或多个以消解冲突,优化整体空域需求方案,此时需要考虑以下基本原则:

- ①对于无法消解的空域需求冲突申请应尽可能少的驳回;
- ②空域需求存在优先级,优化调整时应按照空域需求优先级从低到高调整,高优先级空域尽量保持;
- ③尽量减少被调整的空域需求;
- ④尽量减少空域调整的幅度,不能超过水平、高度和时间调整要求的最大值(分别为 $s_{a \max}$ 、 $h_{a \max}$ 、 $t_{a \max}$);
- ⑤空域消解优先调整高度和水平距离,尽可能避免调整空域的时间范围。

不存在冲突的空域需求间需要满足高度、水平和时间均不重叠或不小于最小安全间隔。空域冲突调整时,根据所需调整三类参数的类型可分为高度调整、水平调整和时间调整,如图2所示。在实际飞行航运中,可能会涉及到空域需求在不同飞行角度、飞行方向对于南北东西的调整等,为简化问题,提高可操作性,论文仅考虑平移调整的情况。图2中虚线框分别代表没有冲突的高度、水平和时间位置。

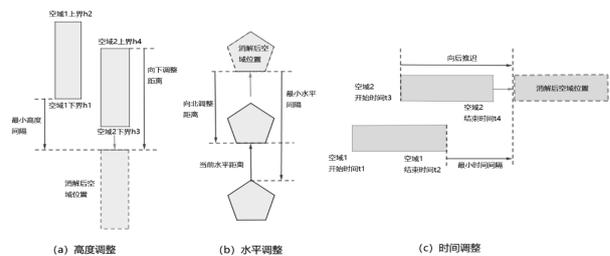


图2 基于三类参数调整的空域冲突消解示意图

3.2 动态优化模型

一旦飞行器间存在着需求冲突,就需要对存在冲突的飞行计划进行优化调解。随着各类飞行或运输任务对空间容量需求的不断提升,结合具体的飞行计划和航空器的实时运

行轨迹来进行空域需求的动态规划有可能提升空域使用的灵活性。考虑航空器的时空运行,即将空域冲突判断的标准从空域间的间隔判断转化为航空器间的动态时空轨迹间隔的判断,只要求航空器在任意时刻保持安全间隔即可。通过对航空器的飞行计划和时空运行轨迹的叠加考虑,可以使航空器获得原本无法满足的空域需求和飞行计划,大大提高空域资源的使用效率,使空域使用效率翻倍,有效降低飞行成本。

如图3所示,这两架航空器虽然整体的空域需求存在一定的重叠,如果采用空域规划中所有空域完全隔离来判断飞行冲突的方法,可认为这两个航空器的飞行需求间存在冲突。但进一步结合两者的实时轨迹来考虑,可以发现:虽然两者完整轨迹空间存在重叠,但其实时飞行的空间位置始终保持在安全距离范围内,不满足飞行高度、水平和时间重叠或是小于最小安全间隔的条件,即航空器的飞行间不存在干扰,不会造成飞行事故,此时就不能简单认为其存在冲突,进一步地就无需对其需求进行调整。

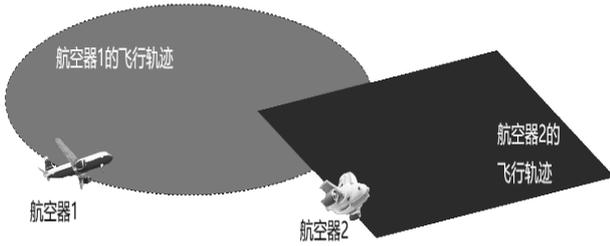


图3 航空器间的实时轨迹图

一方面,在以上考虑实时的飞行状态下,对于空域需求冲突的检测提出了更高要求,需要结合航空器在执行任务过程中的飞行速度、飞行目标位置和实时的运行轨迹有完整的掌握,此外还应该着重提高其在概率出现位置飞行是否发生冲突的检测能力,以适应解决飞行过程中对突发状况。另一方面,对于已检测出的空域冲突,在进行优化调整时也要掌握更加丰富的飞行需求数据,对飞行轨迹进行更加准确和完整的描述,以此提出更为合理科学的调解方案,实现飞行器间的实时空域需求优化,提高空域资源的利用率,达到空间复用的目标。

基于DPSO算法,设计目标函数来评估防止事故的需求:

$$G = \alpha \times t + \beta \times r$$

其中, α 和 β 为权重系数,用于平衡飞行时间和事故风险的重要性; t 为飞行的时间需求; r 为事故风险。

考虑到飞行时间的调整会对飞行成本造成较大影响,并结合冲突消解原则,将飞行时间作为目标函数的一部分,使得目标函数在飞行时间方面的调整越小越好。优化目标是 minimized 飞行器之间的事故风险,以确保安全,事故风险可以通过考虑以下因素进行计算:

① 飞行器间的最小安全距离: 飞行器之间应该保持足够的安全距离,以避免发生事故,使得目标函数在最小安全

距离方面越大越好。

② 飞行器间的相对速度: 飞行器之间的相对速度也会影响事故风险。如果相对速度较大,发生事故风险将增加,因此使目标函数在相对速度方面越小越好。

由此,定义事故风险:

$$r = \gamma \times l_{min} + \delta \times \Delta v$$

其中, γ 和 δ 为权重系数,用于平衡最小安全距离和相对速度的重要性; l_{min} 为由飞行器间的空间坐标得到的最小安全距离; Δv 为飞行器间的相对速度。

根据不同的飞行任务,结合考虑不同情况下的一些实际约束条件,如避免与其他飞行器相撞、遵守空域规则、避免地形障碍物、飞行成本的控制、飞行任务的紧急情况等,还可以设计不同复杂程度的目标函数。

4 实验及结果分析

4.1 问题定义

为了验证论文DPSO算法在飞行冲突解脱任务中的有效性和优势,进行仿真实验,仿真环境为AMD Ryzen 74800H with Radeon Graphics,主频2.90GHz,16GB内存,基于Matlab2023b。在此之前需要明确飞行的空域需求(论文主要是飞行高度、水平和时间的需求,以及空域需求调整的优先级)和可能存在的空域冲突。

4.2 模型设置

设航空器需求的水平范围为 1000×1000 (km),高度范围为 $[0.3, 9](km)$,时间范围为 $[0, 600](min)$ 。

论文的仿真中,投入了300个航空器的空域需求(即300个粒子),设定粒子的维度为4(高度、水平、时间以及优先级)。首先,设定算法中的一些基本参数:一般来说,较大的学习因子可以增加搜索的广度,但可能会降低搜索的精度,一般设定在0.5~2.0;较大的惯性权重可以增加搜索的速度,但可能会导致过早陷入局部最优解,较小的惯性权重可以增加搜索精度,但会减慢搜索速度,常用的取值为0.4~0.9(见表1)。

表1 算法参数设定

系统参数	种群数 n	最大迭代次数 T	最	最	个体学习因子 C_1	社会学习因子 C_2	惯性因子 $\omega_{max}, \omega_{min}$
			大速度 V_{max}	小速度 V_{min}			
参数取值	300	50	5	-5	[0.5, 2]	[0.5, 2]	0.9, 0.4

4.3 仿真实验及结果分析

根据算法流程,初始化了种群个体的限定位置和速度,初始化了个体最优位置和最优值,初始化了全局最优位置和最优值,建立速度更新公式,建立位置更新公式,进一步地动态计算惯性权重值。通过计算适应度函数来评估每个粒子的解,并记录每次迭代的最优解。在模拟实验当中还应该注意动态计算惯性权重值和边界条件的处理,这也是DPSO

算法的独特之处。

图4分别为传统粒子群算法和动态粒子群算法适应度值进化曲线和动态粒子群算法的惯性权重、学习因子进化曲线。在该模型中,空域冲突解脱需要保证调整的空域需求尽量少,调整的空域需求范围尽量少偏离原需求,因此其目标函数需要控制调整的成本尽量小,导致在该问题中适应度值为负。由适应度进化曲线可知,在不到25次迭代中适应度值即可完成较大范围的变化,并保持稳定,说明该算法已经收敛到了一个全局最优解位置^[6, 13]。对比传统粒子群算法,DPSO算法可以在较少的迭代次数中完成最适应度值的寻找,保证最优方案的质量。该曲线在一定程度上还反映出动态粒子群算法可以在很微小的适应度值之间来调整粒子的位置和速度,以达到更理想的优化效果,并且可以在极少的迭代次数下就可以完成寻优,大大提高了数据的处理效率,为完成大规模的数据分析调整提供了解决途径。

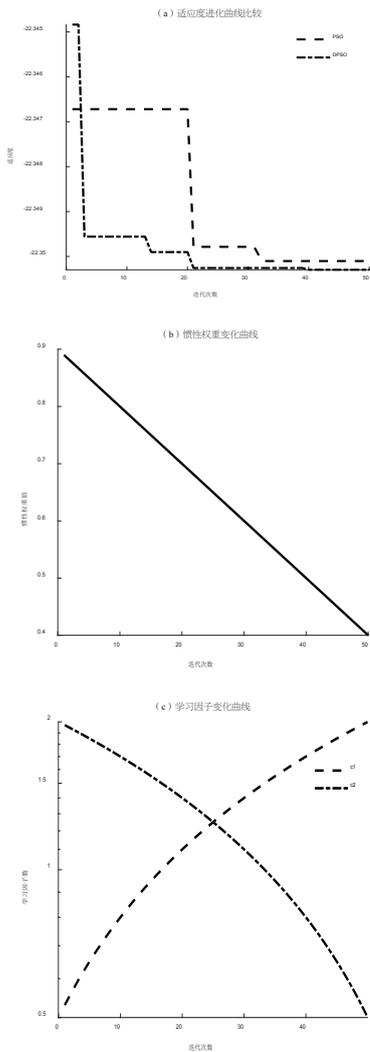


图4 传统粒子群算法和动态粒子群算法适应度值进化曲线和动态粒子群算法的惯性权重、学习因子进化曲线

迭代过程中,惯性权重明显呈线性减小,从最初较大的值使粒子的搜寻具有较大速度和范围,逐渐减小后使粒子

的搜索更加精准。在此过程中动态粒子群算法表现出一定的鲁棒性,能够在多个局部最优解附近进行搜索,加快了收敛速度,提高了对动态环境的适应性,使模型算法更具实用性。同时动态的个体、社会学习因子,提高了算法的效率和适应性:最开始时保证单个粒子能够有效提升学习能力,增强其全局的搜索能力;在优化处理的最后阶段,逐渐降低自身的学习能力同时提升社会学习能力,达到全局最优化收敛的目的。

图5展示了300个粒子通过传统粒子群算法和动态粒子群算法找寻最优解后获得的各个粒子最优值分布情况,可以发现:动态粒子群算法中绝大多数粒子的最优值均在-0.23至-0.20之间,波动范围较小且趋于稳定,接近目标函数最优值的位置-0.20,说明该算法具有较好的收敛性;同时在迭代过程中可以完整记录下不同粒子最优值的大小,进而评估每个粒子在算法中的贡献度,在以后的迭代中再通过控制权重的方法放大具有较好探索能力的粒子,进一步扩大算法的计算能力^[14]。反观传统粒子群算法获得的各个粒子最优值差别较大,波动范围较大且波动明显。以上表明该算法能够保持较好的多样性,可以在搜索过程中探索到不同的解空间,并提供多个可行的解,在保证优化结果质量的情况下对解决复杂环境中的优化问题提供了解决途径。

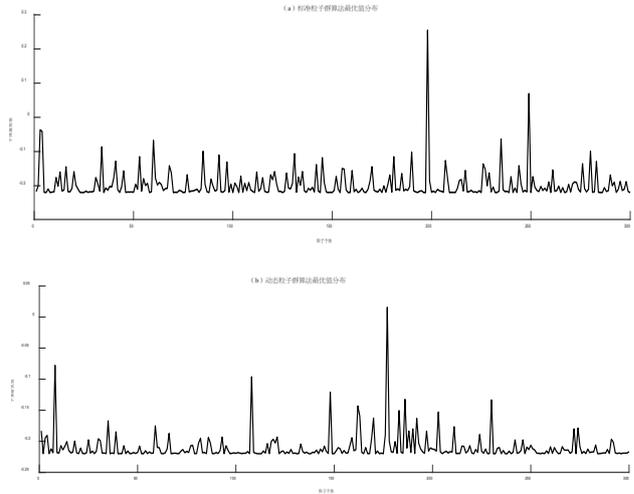


图5 传统粒子群和动态粒子群粒子最优值分布情况

5 结论与展望

论文针对航空器飞行过程中可能存在的空域冲突进行了阐述,根据冲突解脱的条件和环境,在DPSO算法基础上构建了一种飞行器空域需求冲突调解和方案优化模型,成功得到了调解后的最佳空域需求方案。从仿真结果来看,该模型简单易实现,环境适应度更高,收敛性极佳,对于噪声的抗干扰性能较好。需要注意的是,建立在动态粒子群优化算法上的冲突解脱模型参数设置对优化结果有重要影响,需要基于结果进行合理的调整和优化,以保证模型的有效性^[15-18]。下一步需要将模型继续投入到实际的使用场景加以修改优

化,通过适当调整目标函数和约束条件,并进一步推广到更多的矛盾寻优场景中,如航空交通管制、卫星轨道规划等问题。

参考文献

- [1] 尹成义,谭安胜.基于航空器航迹预推的飞行冲突探测模型[J].电光与控制,2015,22(12):20-23.
- [2] 李平.航路飞行冲突探测与解脱策略研究[D].北京:中国民用航空飞行学院,2013.
- [3] 张雷,杨瑞娟,罗少华,等.一种航空器跟踪预测飞行冲突检测算法[J].空军预警学院学报,2017,31(6):432-435.
- [4] 李眩,吴晓兵,童百利.基于动态自适应变参的粒子群优化算法[J].四川轻化工大学学报(自然科学版),2021,34(5):41-47.
- [5] 孙睿彤,袁庆霓,衣君辉,等.改进粒子群算法和动态窗口法的动态路径规划[J].小型微型计算机系统,2023,44(8):1707-1712.
- [6] LIAO Pinchao, SUN Xinlu, ZHANG Dan. A multimodal study to measure the cognitive demands of hazard recognition in construction workplaces [J]. Saf Sci,2021,133:105010.
- [7] 于雪晶,麻肖妃,夏斌.动态粒子群优化算法[J].计算机工程,2010(4):193-194+197.
- [8] 胡乔治.多目标粒子群优化算法的研究及应用[D].北京:北京化工大学,2022.
- [9] 赵玉花,石永康.改进粒子群算法的多无人机航迹优化[J].电光与控制,2023,30(5):29-33+51.
- [10] 许胜才,蔡军,程昀,等.基于拓扑结构与粒子变异改进的粒子群优化算法[J].控制与决策,2019,34(2):419-428.
- [11] 仝秋娟,赵岂,李萌.基于自适应动态改变的粒子群优化算法[J].微电子学与计算机,2019,36(2):6-10+15.
- [12] 章恩泽,赵哲萱,韦静月,等.基于环形拓扑结构和动态邻域的多模态多目标粒子群优化算法[J].扬州大学学报(自然科学版),2023,26(4):19-24.
- [13] Hu Yi, Wang Jie, Liang Jing, et al. A self-organizing multimodal multi-objective pigeon-inspired optimization algorithm [J]. Sci China Inf Sci,2019,62(7):70206.
- [14] Ramachandram D, Lisicki M, Shields T J, et al. Bayesian optimization on graph-structured search spaces: optimizing deep multimodal fusion architectures [J]. Neurocomputing,2018,298:80-89.
- [15] 吴清平.一种改进的粒子群优化算法研究[J].大庆师范学院学报,2018,38(6):68-71.
- [16] Zhang Weizheng, Li Guoqing, ZHANG Weiwei, et al. A cluster based PSO with leader updating mechanism and ring-topology for multimodal multi-objective optimization [J]. Swarm Evol Comput,2019,50:100569.
- [17] Wang Hong-bin, Yin Peng-heng, Zheng Wei, et al. Mobile robot path planning based on improved A* algorithm and dynamic window method [J]. Robot,2020,42(3):346-353.
- [18] Li G, Chou W. Path planning for mobile robot using self-adaptive learning particle swarm optimization[J]. Science China Information Sciences,2018,61(5):1-18.