

DOI:10.11918/j.issn.0367-6234.201812040

RWPSO 与马尔科夫链的无人机航路规划

胡美富¹, 宁 芊¹, 陈炳才², 雷印杰¹

(1. 四川大学 电子信息学院, 成都 610065; 2. 大连理工大学 计算机科学与技术学院, 辽宁 大连 116024)

摘 要: 粒子群算法(PSO)是基于种群的全局搜索算法,具有原理简单,搜索稳定高效等特性,在航路规划领域被普遍运用,但是其在陷入局部最优以及收敛速度方面都存在一定的缺陷. 本文针对无人机的任务权重值与生存权重值引入随机游走策略,按照一定规律改变粒子的惯性权重值,可以有效的避免上述情况发生,提升无人机在航路规划中找到最优路径的效率. 另一方面,为了能够给规划的路径提供优劣性的判断标准或参考依据,需要构建适用于评估无人机飞行路径点上的生存状态概率模型,本文将随机游走粒子群算法(RWPSO)的航路规划模型与马尔科夫链生存状态随机性模型相结合,得到一个可以用来评估路径点生存概率的航路规划问题模型. 仿真结果表明,基于任务权重、生存权重、任务生存权重随机游走的 RWPSO 算法在寻优时比 PSO、量子粒子群算法(QPSO)效率更高,并成功结合马尔科夫链得到一个可以描述出无人机生存概率变化的模型. 此模型框架还能够扩展应用于有辐射源、武器、电磁干扰等复杂场景中的航路与任务规划.

关键词: 无人机; RWPSO 优化算法; 马尔科夫链; 生存概率模型

中图分类号: TP301 **文献标志码:** A **文章编号:** 0367-6234(2019)11-0075-07

UAV route planning based on RWPSO and Markov chain

HU Meifu¹, NING Qian¹, CHEN Bingcai², LEI Yinjie¹

(1. School of Electronics and Information Engineering, Sichuan University, Chengdu 610065, China;

2. School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, Liaoning, China)

Abstract: Particle swarm optimization (PSO) is a global search algorithm based on population, which is characterized by simple principle and stable and efficient search. It is widely used in the field of route planning, but it is defective when falling into local optimum and in convergence speed. In this paper, random walk strategy is introduced to the mission weight and survival weight of UAV. By changing the inertia weight of particles according to certain rules, PSO's defects can be effectively avoided, and UAV's efficiency in finding the optimal path can be improved. On the other hand, in order to provide a criterion or reference to evaluate planned path, it is necessary to construct a survival state probability model to evaluate UAV flight path points. The route planning model of the random walk particle swarm optimization algorithm (RWPSO) was combined with the Markov chain survival state randomness model, thereby building up a route planning model for estimating the survival probability of path points. Simulation results show that RWPSO based on random walk of task weight, survival weight, and task survival weight was more efficient than PSO and quantum particle swarm optimization (QPSO) in optimization. A model describing the change of survival probability of UAV was thus obtained successfully by combining Markov chain with RWPSO. The framework can be extended to route and mission planning in complex scenes with radiation sources, weapons, or electromagnetic interference.

Keywords: unmanned aerial vehicle (UAV); random walk particle swarm optimization algorithm (RWPSO); Markov chain; survival probability model

随着现代军事逐渐无人化的发展,无人机已经成为了航空科技领域中最活跃和最重要的发展方向之一^[1]. 预计到 21 世纪初,全球军用无人机的数量将达到 23 000 多架,这些无人机将主要被应用在侦察,电子对抗,以及攻击战斗等领域. 如何使无人机在敌方的雷达、武器等威胁环境中成功完成任务,是

一个备受关注的问题. 目前世界上已经掀起了一场关于无人机的研究热潮,无人机的航路规划也将会来到一个新的阶段.

粒子群算法在航路规划中得到了非常广泛的应用. 文献[2-4]分别对基本粒子群算法在航路规划中的使用做了相应的改进. 文献[5-6]对 PSO, (RandomWalkPSO, RWPSO) 以及一些其他改进的 PSO 算法包括(QuantumPSO, QPSO)等进行了比较与分析,结果表明 RWPSO 算法在运行时间,跳出局部最优的能力等方面较大部分改进 PSO 算法有一

收稿日期: 2018-12-11

基金项目: 国家自然科学基金(61771089)

作者简介: 胡美富(1995—),男,硕士研究生

通信作者: 宁 芊,ningq@scu.edu.cn

定的优势. 本文首先验证了 RWPSO 优化算法在航路规划领域的可行性与优越性, 并进一步的将 RWPSO 与马尔科夫链生存模型相结合, 可以实时的评估无人机的状态概率. 非常直观的了解无人机在有敌方威胁的环境中执行任务时其状态转移变化与生存率情况.

1 无人机航路规划问题建模

1.1 基于 PSO 的航路规划模型

粒子群算法是一种模拟鸟群寻食的算法, 每一只鸟的个体被抽象成没有质量与体积的粒子. 在 d 维的搜索空间中, 每个粒子都有自己的位置向量 $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$, 速度向量 $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$, 以及一个最为关键的适应度值 $J(n)$. 适应度值由适应度函数确定, 在每一次的迭代中, 粒子自身的个体最优解为 $P_{best} = (p_{i1}, p_{i2}, \dots, p_{id})$, 整个群体找到的全局最优解为 $G_{best} = (g_{i1}, g_{i2}, \dots, g_{id})$, 然后根据下面的迭代公式来更新第 i 个粒子的位置和速度.

$$V_i(t+1) = w * v_{ij}(t) + c_1 * (p_{ij} - x_{ij}(t)) + c_2 * (g_{ij} - x_{ij}(t)), \quad (1)$$

$$X_i(t+1) = V_i(t+1) + X_i(t). \quad (2)$$

式中: w 为惯性权重, c_1, c_2 为加速因子. c_1 是粒子的认知部分的权重, 表明粒子吸取自身经验的能力, c_2 是粒子社会部分的权重, 表明粒子间的交互能力. 相对来说: 认知能力越强的粒子收敛的越快, 但易于陷入局部最优, 社会能力越强的粒子, 越能找到最优解, 但是算法效率越低.

本文适应度函数的基本结构如下:

$$J(n) = w_s * C_s(n) + w_p * C_p(n) + w_r * C_r(n) + w_0 * C_0(n) + w_H * C_H(n). \quad (3)$$

其综合考虑了无人机的生存代价、任务的完成代价、参考路径代价, 无人机油耗代价, 飞行高度代价.

1) 生存代价: $w_s * C_s(n)$

$C_s(n) = S_n - S_{n-1}$; S_n 为第 n 次搜索后的生存代价, S_{n-1} 为已执行路径的生存代价和. 其差值将作为本次搜索所持有的生存代价, w_s 为生存代价的权重值.

2) 任务完成的代价: $w_p * C_p(n)$

$C_p(n) = |P_{target} - P_n|$; P_{target} 为下一个任务点的位置, P_n 为第 n 步搜索后的位置, 取空间中这两点的模即为任务完成的代价, w_p 为任务重要性的权重值. 如果任务点都执行完毕, 则下一个任务点为终点.

3) 参考路径代价: $w_r * C_r(n)$

$C_r(n) = D_n(P_n, L_{reference})$; $L_{reference}$ 为参考路径如图 1 所示. D_n 为点 P_n 到直线段 $L_{reference}$ 距离的函数,

返回值即为参考路径对飞行航线的代价. 如果 w_r 越大, 无人机的飞行轨迹越靠近参考路线.

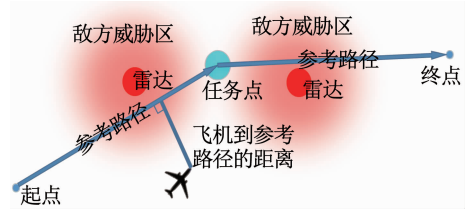


图 1 参考路径和约束表示

Fig. 1 Reference path and constraint representation

4) 油耗与飞行高度代价: $w_0 * C_0(n) + w_H * C_H(n)$

$$C_0(n) = \begin{cases} 0, & \sum_{n=1}^n \text{oil} * \text{Dis}_n < \text{oil}_{\text{All}}; \\ C_{\text{destory}}, & \sum_{n=1}^n \text{oil} * \text{Dis}_n \geq \text{oil}_{\text{All}}. \end{cases} \quad (4)$$

式中: oil 为耗油率, Dis_n 为第 n 次搜索的步长, oil_{All} 为总的油量. 当总油量不能够支撑第 n 步搜索所需的油耗量时, 则引入惩罚 C_{destory} .

$$C_H(n) = \begin{cases} 0, & H_{\min} < H_n < H_{\max}; \\ C_{\text{destory}}, & \text{otherwise}. \end{cases} \quad (5)$$

同上, 当无人机下一次搜索的高度在约束的空间 $H_{\min}, H_{\max}$ 外, 同样引入惩罚 C_{destory} . w_0, w_H 分别为两者的权重.

1.2 RWPSO 优化算法

在对 PSO 算法的优化问题中, 通过调整参数来提高性能是最为常见的一种方法^[7-8].

通过对适应度函数的介绍, 可以发现权重对其代价或约束起到一个非常重要的权衡作用, 直接影响到飞行的规划航迹. 标准的 PSO 算法中, 主要是靠粒子间的相互影响来寻优的, 如果固定权重系数, 粒子很难兼顾全局寻优与局部寻优的能力. 但是采取随机游走策略来更新权重, 算法的寻优速度会得到大大提升, 这使无人机在航路规划时遇到敌方雷达或武器后会有更快的反应速度, 尽早脱离危险区域, 提高生存率. 图 2 为 RWPSO 算法的流程图:

RWPSO 算法的权重随机游走策略是通过将代价的权重系数设置为服从某种随机分布的数. 如果粒子在初始化时就已经在最优点附近, 那么 RWPSO 算法会找出相对小的权重值来增加收敛速度. 如果不在最优解附近, 甚至陷入局部最优时, 由于 RWPSO 算法中的粒子受到不等权重的影响, 会不断尝试跳出局部最优. 具体权重值设置如下:

$$w = \mu + \sigma * N(0, 1). \quad (6)$$

式中: w 为 RWPSO 算法的权重系数, μ 为权重的均值, σ 为方差, $N(0, 1)$ 是服从标准正态分布的随机数.

$$\mu = \mu_{\min} + (\mu_{\max} - \mu_{\min}) * \text{rand}(0, 1). \quad (7)$$

式中: $\mu_{\max}, \mu_{\min}$ 分别为随机权重的最大值与最小值, $\text{rand}(0, 1)$ 为取一个 $(0, 1)$ 之间随机数的函数. 这约束了在路径规划中无人机生存代价和任务代价过低或过高的情况发生.

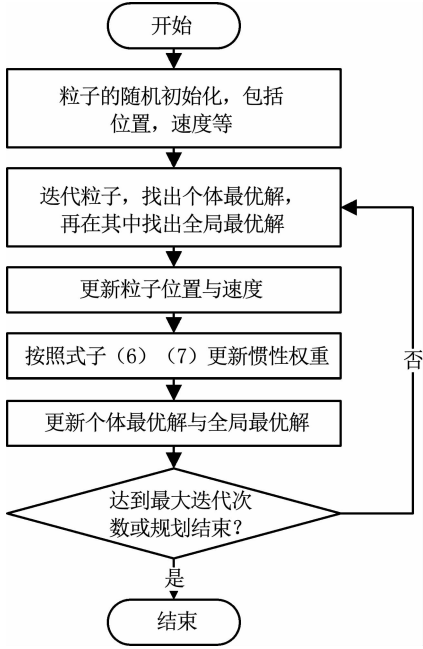


图 2 RWPSO 算法流程图

Fig. 2 Flow diagram of RWPSO

2 基于马尔科夫链的无人机生存概率模型

本文使用时间连续状态离散的马尔科夫链构建一个包含五种生存状态^[9]的概率模型, 用来描述无人机在穿过敌方雷达武器等威胁区域后的生存概率. 文献[10]提出了一种状态之间的转移强度和代价函数. 通过将 RWPSO 算法与状态转移强度相结合, 规划出一条包含生存概率的航线.

2.1 五状态生存概率模型

五状态即为飞机的 5 种状态: 未被发现 (undetected); 被侦测 (detected); 被追踪 (tracked); 交战 (engage); 被击落 (hit). 生存状态之间是可以相互转换的, 但只能是相邻的两个状态转换. 假设无人机从安全区飞入敌方雷达区, 无人机可能就会被敌方雷达发现而进入被侦测状态, 甚至再由被侦测转移到被追踪状态, 此时如果无人机飞离敌方雷达区域, 将又会由被追踪转移到被侦测状态, 而不能直接被追踪转移到未被发现状态. 无人机一旦从交战状态转移到了被击落状态, 则为吸收态, 将永远处于被击落状态. 状态概率计算公式:

$$P(X_{t+\Delta} = j | X_t = i) = \lambda_{ij} \Delta, \\ P(X_{t+\Delta} = i | X_t = i) = 1 - \sum_{k \neq i} \lambda_{ik} \Delta.$$

式中: i 与 j, k 分别为无人机的两相邻生存状态, λ_{ij} 为两状态间的转移强度, Δ 为无限小的时间长, P 为状态间的转移概率. 因为 RWPSO 算法收敛速度快, 所以当无人机发现危险时, 特别当任务点在敌方威胁区域内时, 能够在完成任务后把在威胁区域内总的 Δ 缩小, 迅速飞离危险区, 以提高生存率.

2.2 转移强度与代价函数

2.2.1 转移强度的设定

转移强度描述了一个状态到另一个状态的转移概率^[9], 为 λ_{ij} .

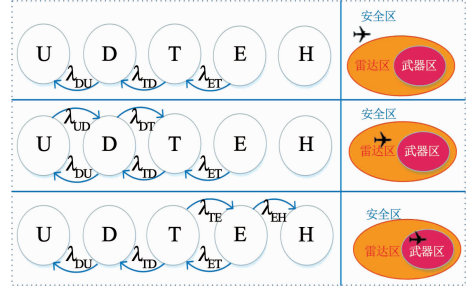


图 3 状态转移强度的设定

Fig. 3 Setting of transfer strength

图 3 中的 U, D, T, E, H 分别为五种生存状态, 右边部分为无人机所处的区域, 包括安全区与威胁区, 威胁区又分为雷达区和武器区. 转移强度默认值的设置将参考文献[10], 但是在威胁区域中所有点采用其固定的默认值, 会导致生存代价相同. 因此本文使用下面的公式来更新威胁区域内不同点的转移强度:

$$\lambda_{ij} = \begin{cases} \lambda_{out-ij}, & \text{position} \in S; \\ \lambda_{in-ij}^* (r-d)/r, & \text{position} \notin S. \end{cases} \quad (8)$$

position 为无人机的位置, S 为安全区域, λ_{out-ij} , λ_{in-ij} 为无人机在威胁区域外和内的默认转移强度. r 为威胁半径, d 为点 position 距离威胁中心的欧式距离.

2.2.2 重叠区域的转移强度设定

如果无人机飞入了雷达和武器都辐射到的范围, 那么式(8)中的 d 将无法表示. 关于重叠区域转移强度的计算公式如下:

$$\lambda_{ij}^c = \begin{cases} \lambda_{ij}^1 + \lambda_{ij}^2, & (i, j) = (U, D) \& (D, T), \\ \max(\lambda_{ij}^1, \lambda_{ij}^2), & (i, j) = (T, E) \& (E, H), \\ \frac{\lambda_{ij}^1 \lambda_{ij}^2 (\lambda_{ij}^1 + \lambda_{ij}^2)}{\lambda_{ij}^1 \lambda_{ij}^2 + (\lambda_{ij}^1)^2 + (\lambda_{ij}^2)^2}, & (i, j) = (D, U) \& (T, D), \\ \text{都是搜索雷达} \\ \min(\lambda_{ij}^1, \lambda_{ij}^2), & (i, j) = (D, U) \& (T, D), \\ \text{不都是搜索雷达} \\ \min(\lambda_{ij}^1, \lambda_{ij}^2), & (i, j) = (E, T). \end{cases} \quad (9)$$

式中: λ_{ij}^c 为重叠区域无人机的转移强度, λ_{ij}^1 为针对第一个威胁点区域的转移强度, λ_{ij}^2 为针对第二个威胁点区域的转移强度.

2.2.3 代价函数

$$\underbrace{\begin{pmatrix} p_U(n) \\ p_D(n) \\ p_T(n) \\ p_E(n) \\ p_H(n) \\ \xi(n) \end{pmatrix}}_{\hat{X}(t_n)} = \underbrace{\begin{pmatrix} -v_U(n) & \lambda_{DU}(n) & \lambda_{TU}(n) & \lambda_{EU}(n) & \lambda_{HU}(n) & 0 \\ \lambda_{UD}(n) & -v_D(n) & \lambda_{TD}(n) & \lambda_{ED}(n) & \lambda_{HD}(n) & 0 \\ \lambda_{UT}(n) & \lambda_{DT}(n) & -v_T(n) & \lambda_{ET}(n) & \lambda_{HT}(n) & 0 \\ \lambda_{UE}(n) & \lambda_{DE}(n) & \lambda_{TE}(n) & -v_E(n) & \lambda_{HE}(n) & 0 \\ \lambda_{UH}(n) & \lambda_{DH}(n) & \lambda_{TH}(n) & \lambda_{EH}(n) & -v_H(n) & 0 \\ \xi_U(n) & \xi_U(n) & \xi_U(n) & \xi_U(n) & \xi_U(n) & 0 \end{pmatrix}}_{\Lambda(n)} \underbrace{\begin{pmatrix} p_U(n) \\ p_D(n) \\ p_T(n) \\ p_E(n) \\ p_H(n) \\ \xi(n) \end{pmatrix}}_{\hat{X}(t_n)} \quad (10)$$

式中: $p_I(n)$ 为第 $n+1$ 次搜索时处于 I 状态的概率, I 为 U,D,T,E,H 中的某一状态.

$$v_I(n) = \sum_{i \neq j} \lambda_{ij}(n),$$

$$\xi_M(n) = b_i + \sum_{i \neq j} \lambda_{ij}(n) C_{ij}.$$

其中, $\xi_M(n)$ 为截至前 n 次搜索飞机累计生存代价. b_i 为处于 i 状态时,其每个单位时间的生存代价. 由于 i 有 5 种状态,故 b_i 用 $b = (0, b_D, b_T, b_E, 0)$ 表示.

$$C_{ij} = \begin{cases} C_{\text{destroy}}, i = E, j = H; \\ 0, \text{otherwise.} \end{cases} \quad (11)$$

式中: C_{ij} 为从状态 i 转移到状态 j 的转移代价,最后可以算出生存代价的矩阵向量 $X(t_{n+1})$.

$$X(t_{n+1}) = e^{\Lambda(n) * \Delta t} X(t_n).$$

3 试验仿真与分析

3.1 基本参数

在 Intel Core i5- 4460@3.2 GHz,8 G 内存的硬件,Matlab2017a 版软件环境下进行的仿真实验. 场景参数由表 1 给出,其中搜索雷达的搜索半径为 150 km,火控雷达的打击半径为 90 km,任务点的半径为 15 km. 转移强度具体参数如表 2,状态代价如表 3.

3.2 仿真结果与分析

一方面为了证明随机游走策略的优越性,在下面分别对任务权重,生存权重采用控制变量法,并将仿真结果与标准的 PSO 优化算法进行比较,在最后将改进的 QPSO 与任务生存权重均随机游走的 RWPSO 进行对比. 另一方面根据粒子群算法与马尔科夫链的结合模型,分别给出每次航路的无人机生存状态概率分布图.

3.2.1 基于随机游走的任务代价权重

首先设定初始的代价权重值,其设置如表 4 所示: 在此权重值下,PSO 优化算法的搜索路径如

代价为无人机在执行第 n 次搜索前后的生存概率变化. 生存概率是通过该点的转移强度值来计算得到的^[9]. 具体计算公式为

图 4 中的黑色虚线所示,任务权重随机游走的 RWPSO 算法将按照式(6)对其任务权重进行更新,并且设置 $\mu_{\max} = 6, \mu_{\min} = 4$,最后 RWPSO 搜索路径如图 4 中的红色实线所示,其中横纵坐标表示位置.

表 1 场景参数

Tab. 1 Parameters of the scene

描述	位置坐标/km	颜色显示
原点坐标	(300,300,0)	无
场景范围	(700*600*500)	彩色
起点	(300,450,285)	绿色
终点	(930,810,300)	绿色
搜索雷达 1	(500,475,238)	蓝色
搜索雷达 2	(675,750,275)	蓝色
搜索雷达 3	(830,515,240)	蓝色
火控雷达 1	(530,520,263)	红色
火控雷达 2	(700,680,275)	红色
任务点 1	(410,545,280)	黄色
任务点 2	(555,675,280)	黄色
任务点 3	(800,580,278)	黄色

表 2 转移强度参数

Tab. 2 Parameters of transfer strength

转移强度	安全区	探测雷达区	武器雷达区
λ_{UD}	0	0.4	0
λ_{DU}	0.2	0.1	0.2
λ_{DT}	0	0.3	0
λ_{TD}	0.2	0.1	0.2
λ_{TE}	0	0	0.2
λ_{ET}	1	1	0.1
λ_{EH}	1	0	0.3

表 3 状态代价参数

Tab. 3 Parameters of state cost

不同状态代价	b_U	b_D	b_T	b_E	b_H	C_{destroy}
代价	0	1	10	100	0	8 000

表 4 初始权重值设置	
Tab. 4 Setting of initial weight value	
代价	权重值
任务代价	5.00
生存代价	0.60
参考路径约束	0.10
高度约束	0.10
油耗约束	0.05

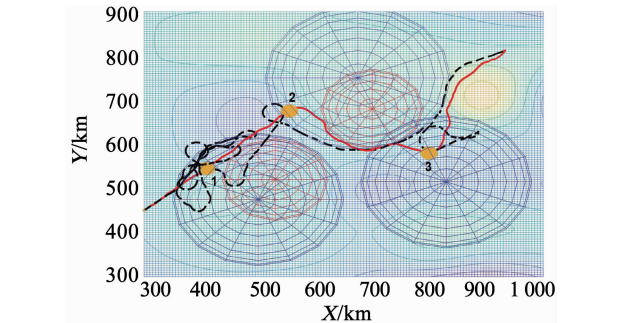


图 4 基于 RWPSO (红实) 与 PSO (黑虚) 航路搜索俯视图

Fig. 4 Top view of route search based on RWPSO (red solid line) and PSO (black dotted line)

观察图 4 发现,在任务点 1,2,3 中,PSO 优化算法都出现局部最优的情况,特别是任务点 1 与 3. 但是 RWPSO 优化算法很好的避免了这个问题,跳出局部最优的能力显然强于 PSO 算法.

图 5 为这两条搜索航线的状态概率图,横坐标为执行的步数,每次搜索执行 4 步,纵坐标为每步搜索对应的状态概率,且 5 状态概率相加始终等于 1. 在图 5 中可以发现 PSO 算法陷入了局部最优,导致其生存状态概率不断跳变. 同样,生存状态概率图提供了一个有力的参考证据,用来评估在这种威胁环境中是否适合使用无人机去完成本次任务.

3.2.2 基于随机游走的生存代价权重

生存代价的初始权重设置的是 0.6,属于使无人机具有较高的生存能力,当然这是现代战争所必然的. 现在将 w_s 设置为 0.3,其他权重值不变,这样做的好处是方便对 RWPSO 优化算法与 PSO 优化算法在收敛速度与寻优能力方面进行比较.

与设置任务权重一样. 生存权重随机游走的 RWPSO 算法是把生存权重按照式 (6) 进行更新, $\mu_{\max}, \mu_{\min}$ 分别设置为 0.4 与 0.2,其他权重值按照表 4 给出. 仿真效果如图 6:

在图 6 中的 2 号任务点,执行完任务后,PSO 优化算法由于收敛较慢,使得无人机飞入敌方武器威胁区域. RWPSO 优化算法由于生存代价是不固定的,故种群肯定会往高生存率方向移动,即可迅速飞离危险区,也就避免出现图 7 中被 hit 的状态.

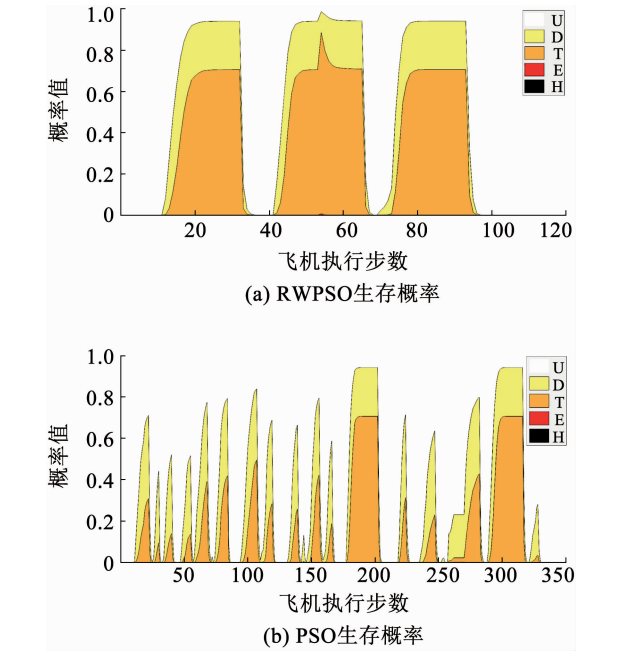


图 5 基于 RWPSO 与 PSO 的生存状态概率

Fig. 5 Survival state probability diagram based on RWPSO and PSO

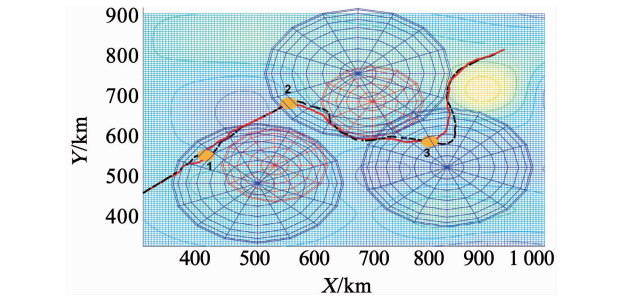


图 6 基于 RWPSO (红实) 与 PSO (黑虚) 的航路搜索俯视图

Fig. 6 Top view of route search based on RWPSO (red solid line) and PSO (black dotted line)

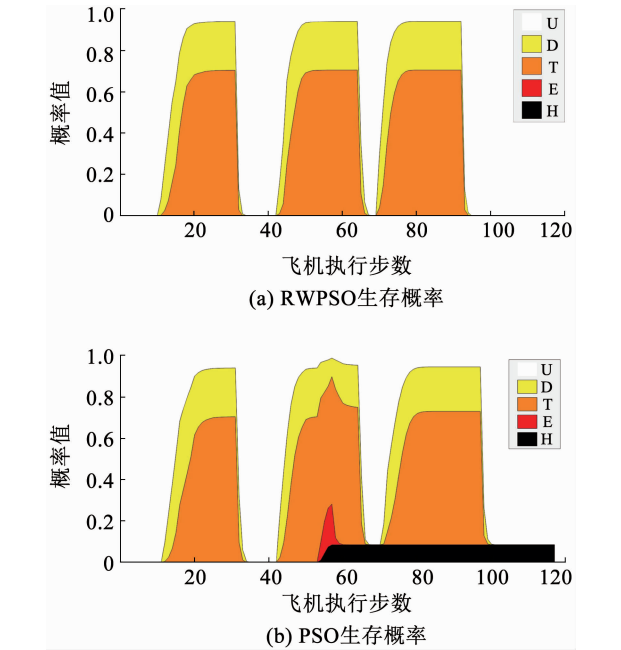


图 7 基于 RWPSO 与 PSO 的生存状态概率

Fig. 7 Survival state probability diagram based on RWPSO and PSO

同样在 3 号任务点附近的轨迹也验证了这一结论的正确性, RWPSO 优化算法在完成 3 号任务后可以迅速转移到高生存率的地方, 减少在威胁区域内的飞行时间。

3.2.3 基于随机游走的任务与生存代价权重

在表 4 的权重参数中, 实验一组生存代价与任务代价权重都是基于随机游走的数据, 并将此实验结果与 QPSO 进行对比。一般的 PSO 优化算法都需要迭代 100 次左右^[11-12], 但本文实验只初始化了 30 个种群, 迭代 30 次, 缩小了约 3 倍的时间。

仿真效果如图 8 所示: RWPSO 的搜索路径用红实线标出, QPSO 的搜索路径用黑虚线标出。与图 4 相比, 生存权重与任务权重均随机游走的 RWPSO 在 2 号任务点与 3 号任务点之间不会出现一直贴着敌方武器雷达飞行的情况出现, 避免了图 5 中 RWPSO 在第 55 步附近出现被追踪概率骤增的情况

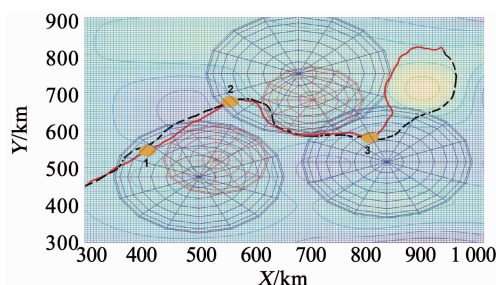
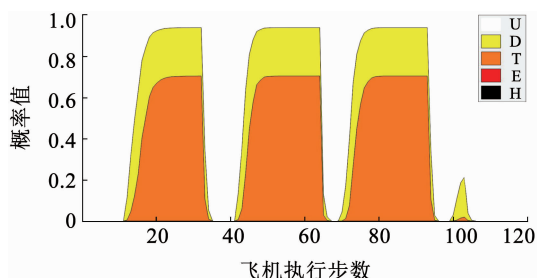
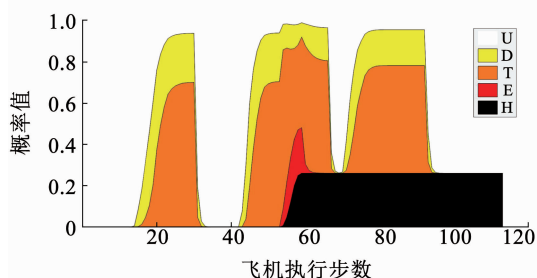


图 8 基于 RWPSO (红实) 与 QPSO (黑虚) 的航路搜索俯视图

Fig. 8 Top view of route search based on RWPSO (red solid line) and PSO (black dotted line)



(a) RWPSO 生存概率



(b) PSO 生存概率

图 9 基于 RWPSO 与 QPSO 的生存状态概率

Fig. 9 Survival state probability diagram based on RWPSO and QPSO

发生。QPSO 在执行完 2 号任务点后依旧飞入了敌方的武器雷达区域, 导致了图 9 中约 30% 的被击毁概率。3 号任务点时 QPSO 飞出任务区域后的收敛速度依旧不及 RWPSO, 过多时间处于敌方搜索雷达范围内。

图 10 对上面的 6 条航路的生存代价进行了对比, 由于高生存权重的 PSO 优化算法陷入了局部最优故没有在图中标出。在低生存权重时: 基于任务权重随机游走的 RWPSO 算法航路代价低于 PSO 算法。高生存权重时: 两条 RWPSO 算法航路代价也只是略高于低生存权重 PSO 算法的航路代价值, QPSO 由于飞入敌方武器威胁区域过多导致航路代价超过 7 000。

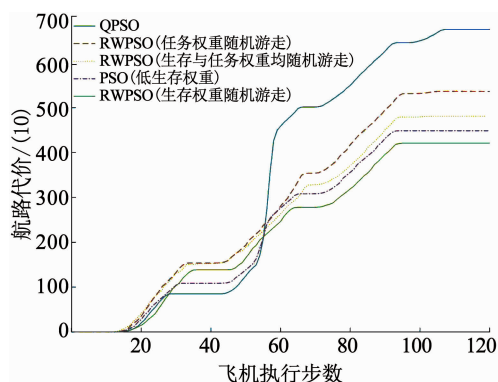


图 10 生存代价对比

Fig. 10 Cost of survival contrast diagram

4 结 论

通过对比 RWPSO 优化算法与 PSO、QPSO 优化算法在无人机航路规划应用中的仿真效果与实验数据, 证明了 RWPSO 优化算法跳出局部最优能力强, 收敛速度快等特点。再结合马尔科夫链生存模型, 实时的对飞行路径中的每一个点进行生存代价的评估, 得到生存概率图, 为无人机在复杂场景中的航路规划与生存状态动态评估提供了一种可行方法。

参考文献

- [1] NONAMI K. Prospect and recent research & development for civil use autonomous unmanned aircraft as UAV and MAV[J]. Journal of System Design & Dynamics, 2007. 1 (2): 120. DOI:10.1299/jsdd.1.120
- [2] WANG Qiang, ZHANG An, QI Linghui. Three-dimensional path planning for UAV based on improved PSO algorithm[C]//The 26th Chinese Control and Decision Conference (2014 CCDC). IEEE, 2014: 3981. DOI:10.1109/CCDC.2014.6852877
- [3] GENG Qingbo, ZHAO Zheng. A kind of route planning method for UAV based on improved PSO algorithm[C]//2013 25th Chinese Control and Decision Conference (CCDC). IEEE, 2013: 2328.

DOI:10.1109/CCDC.2013.6561326

[4]MARINI F, WALCZAK B. Particle swarm optimization (PSO). A tutorial[J]. Chemometrics & Intelligent Laboratory Systems, 2015. 149: 153. DOI:10.1016/j.chemolab.2015.08.020

[5]CHEN Chenyu, CHANG Kuochou, HO S H. Improved framework for particle swarm optimization: Swarm intelligence with diversity-guided random walking [J]. Expert Systems with Applications, 2011, 38(10): 12214. DOI:10.1016/j.eswa.2011.03.086

[6]FU Yangguang, DING Mingyue, ZHOU Chengping. Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2011, 42(2): 511. DOI:10.1109/tsmca.2011.2159586

[7]CHEN H C, FENG H M, LIN T H, et al. Adapt DB-PSO patterns clustering algorithms and its applications in image segmentation[J]. Multimedia Tools and Applications, 2016, 75(23): 15327. DOI: 10.1007/s11042-015-2518-4

[8]SHI Y, EBERHART R C. Empirical study of particle swarm optimization[C]//Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406). IEEE, 1999, 3: 1945. DOI:10.1109/CEC.1999.785511

[9]ERLANDSSON T, NIKLASSON L. A five states survivability model for missions with ground-to-air threats[C]//Modeling and Simulation for Defense Systems and Applications VIII. International Society for Optics and Photonics, 2013, 8752: 875207. DOI: 10.1117/12.2015022

[10]ERLANDSSON T. Route planning for air missions in hostile environments[J]. The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, 2015,12(3): 289

[11]NEX F, REMONDINO F. UAV for 3D mapping applications: a review[J]. Applied Geomatics, 2014, 6(1): 1. DOI: 10.1007/s12518-013-0120-x

[12]高芳, 崔刚, 吴智博, 等. 求解背包问题的病毒协同进化粒子群算法[J]. 哈尔滨工业大学学报, 2009, 41(6): 103-107. DOI:10.3321/J.issn:0367-6234

GAO Fang, CUI Gang, WU Zhibo, et al. Virus-evolutionary particle swarm optimization algorithm for knapsack problem[J]. Journal of Harbin Institute of Technology, 2009, 41(6): 103-107. DOI:10.3321/J.issn:0367-6234

[13]MUELLER M, SMITH N, GHANEM B. A Benchmark and Simulator for UAV Tracking[J]. Far East Journal of Mathematical Sciences, 2016, 2(2): 445

[14]MARINAKIS Y, MARINAKI M. A hybrid multi-swarm particle swarm optimization algorithm for the probabilistic traveling salesman problem[J]. Computers & Operations Research, 2010, 37(3): 432. DOI:10.1016/j.cor.2009.03.004

[15]WANG Gaige, GUO Lihong, DUAN Hong, et al. A bat algorithm with mutation for UCAV path planning[J]. The Scientific World Journal, 2012, 2012: 1. DOI:10.1100/2012/418946

[16]GUPTA L, JAIN R, VASZKUN G. Survey of important issues in UAV communication networks[J]. IEEE Communications Surveys & Tutorials, 2016, 18(2): 1123. DOI: 10.1109/COMST.2015.2495297

[17]HUANG Yu, GUO Feng, LI Yongling, et al. Parameter estimation of fractional-order chaotic systems by using quantum parallel particle swarm optimization algorithm [J]. Plos One, 2015, 10(1): e0114910. DOI:10.1371/journal.pone.0114910

(编辑 苗秀芝)