DOI:10.20033/j.1003-7241.(2025)06-0011-05

基于ADP的无人机集群实时避障动态规划

朱 静，邓宏昌，甄子洋

（南京航空航天大学 自动化学院，江苏 南京 210000）

摘 要：研究多无人机集群动态路径规划问题，即多无人机集群以编队的形式共同执行某一任务，从初始位置出发编队飞往指定目标位置，同时规避碰撞障碍物的风险。采用自适应动态规划算法实现多无人机编队系统的动态路径规划，当检测到有障碍物存在时，通过神经网络计算并最小化所设定的效用函数和代价函数，动态调整编队系统的路径方案和控制策略，以实现实时避障功能并最终到达指定目标位置。最后，通过MATLAB对无人机集群在单障碍物和两个障碍物的两类环境进行仿真验证该方法的有效性。

关键词：无人机；神经网络；自适应动态规划；避障；避碰

中图分类号：TP242；TP183 文献标识码：A 文章编号：1003−7241(2025)06−0011−05

ADP-based Dynamic Path Planning of Multi-UAVs

ZHU Jing, DENG Hong-chang, ZHEN Zi-yang

(College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210000, China)

Abstract: This paper focuses on the dynamic path planning of multiple unmanned aerial vehicles (UAV). The adaptive dynamic programming (ADP) algorithm is applied for path planning to enable multi-UAVs to cooperate with each other in formations to complete some tasks. When obstacles are detected by multi-UAVs during the flight, the dynamic obstacle avoidance of the formation is achieved by dynamic adapting the path and control strategy through the computation and minimization of our proposed utility function and cost function, and then multi-UAVs arrive the destination in formation. Numerical simulations are conducted in MATLAB environment to verify the effectiveness of the proposed algorithm under non-obstacles, single obstacles, and two obstacles circumstances.

Keywords: UAV; neural network; ADP; obstacle avoidance; collision avoidance

0 引言

无人机集群协同执行任务时，路径规划要求复杂多样，既要保证安全，又要保证时效。当前研究成果主要集中在静态路径规划中，通常采用智能算法领域，包括利用“虚拟力场”的人工势场法[1]、A*算法[2]、RRT算法[3]，以及基于仿生思想的蚁群算法[4]、粒子群算法[5]等。然而，在无人机集群编队飞行实际应用中实时性往往也有很高要求，动态变化的环境为实时路径规划带来困难。为解决此类问题，本文提出一种基于动态规划(adaptive dynamic programing，ADP)算法的动态路径规划方案。ADP虽在无人机及集群控制领域有了诸多应用[6]，但在无人机集群路径规划和避障研究方面成果相对较少，特别是三维动态路径规划更是成果鲜少。此类研究的主要难点在于存在动态障碍物时，如何将障碍物引入在效用函数和代价函数的设计中，使其可以动态地调整，实时计算更新最优路径。

基于此，本文提出了一种基于目标−避障−避碰联合约束的无人机集群路径规划方法，将最小安全距离引入效用函数和代价函数，实现了在三维空间环境下的无人机集群安全避障动态路径规划。

1 无人机系统建模

无人机的模型可分为位置外环和姿态内环，如果利用自适应动态规划算法同时控制其内环和外环，这会使控制误差较大且稳定性也较差[7]。本文将无人机作为质点处理，仅考虑无人机位置实现路径规划和避障的要求，对其姿态变化不作要求。此外，为研究方便，假设无人机的侧滑角和迎角非常小，并且不考虑风场的影响。采用式(1)所示的六自由度的数学模型[8]，位置外环状态方程如下：

$$\begin{cases}\dot{x}_i = V_i \cos\gamma_i \cos\chi_i \\ \dot{y}_i = V_i \cos\gamma_i \sin\chi_i \\ \dot{z}_i = -V_i \sin\gamma_i \end{cases} \tag{1}$$

式中，(x_i, y_i, z_i)表示第i个无人机在惯性坐标系下的位

*基金项目：国家自然科学基金面上项目 (61973158)

收稿日期：2024-05-29

置，V_i、γ_i和χ_i分别表示第i个无人机的速度、爬升角和航向角。由于本次研究对姿态角不作要求，可以将γ_i和χ_i看作某一常量，无人机位置只和其加速度有关，对其位置外环进行线性化得到如下形式：

$$\begin{cases}\dot{P}_i = V_i \\ V_i = u_i\end{cases} \tag{2}$$

式中，$P_i=(x_i,y_i,z_i)$为第i个无人机的位置坐标，也即状态变量，$V_i=(v_{ix},v_{iy},v_{iz})$为其速度，$u_i=(u_{ix},u_{iy},u_{iz})$为无人机在惯性坐标系中的加速度，也即控制输入。

2 动态路径规划算法设计

2.1 动态规划原理与方法总体框架

在离散系统中，常采用以下状态方程[8]：

$$x(k+1)=F\left(x(k),u(k),k\right) \tag{3}$$

式中，$x(k)$表示该系统时刻的状态向量，$u(k)$为对应的控制输入。则系统的代价函数表示如下：

$$J\left(x(i),i\right)=\sum_{k=i}^{\infty}\gamma^{k-i}U\left(x(k),u(k),k\right) \tag{4}$$

式中，$U(x(k),u(k),k)$表示效用函数，γ为折扣因子且其取值范围是$\gamma\in(0,1]$。控制目的即为求解最优控制序列$u^*(k)$，使式(4)中的代价函数最小，此时最优代价函数可以表示如下：

$$J^*\left(x(k)\right)=\min_{u(k)}\left\{U\left(x(k),u(k)\right)+\gamma J^*\left(x\left(k+1\right)\right)\right\} \tag{5}$$

对应的最优控制序列为

$$u^*\left(k\right)=\arg\min_{u(k)}\left\{U\left(x(k),u(k)\right)+\gamma J^*\left(x\left(k+1\right)\right)\right\} \tag{6}$$

基于ADP基本结构进行本文的控制结构设计，如图1所示。其中，执行网络和评价网络为函数近似结构，采用神经网络近似逼近。系统在k时刻的状态$x(k)$作为执行网络的输入$u(k)$，可得系统当前控制输入及无人机下一时刻的状态向量$x(k+1)$。$k+1$时刻的控制输入$u(k+1)$和状态向量$x(k+1)$作为评价网络的输入，可得代价函数的估计值$J(x(k+1))$。通过对执行网络和评价网络的神经网络的权值进行迭代更新，使得状态向量达到预期状态，最终得到最优的代价函数和控制策略。

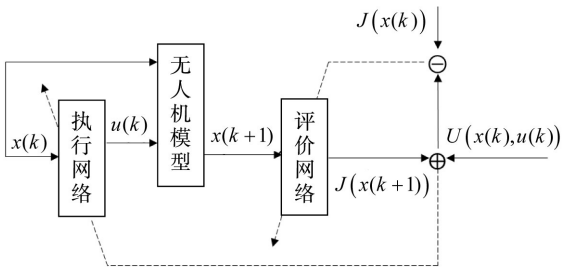


图1 基于ADP的总体设计框图

2.2 效用函数和代价函数设计

定义如下效用函数：

$$U\left(x(t),u(t)\right)=U_{obj}(k)+U_{obs}(k)+U_{ij}(k)+u'(k)u(k) \tag{7}$$

式中，$U_{obj}(k)$、$U_{obs}(k)$、$U_{ij}(k)$分别为目标效用函数、避障效用函数和避碰效用函数。

代价函数可下式可得：

$$J\left(x(k),k\right)=U\left(x(k),u(k)\right)+\gamma J\left(x(k+1)\right) \tag{8}$$

目标效用函数与无人机与目标点相对距离有关，促使无人机靠近目标并最终抵达目标所在位置。本文设计目标效用函数如下：

$$U_{i_{obj}}(k)=\xi_1 d_{i_{obj}} \tag{9}$$

式中，$d_{i_{obj}}=\sqrt{\left(x_i-x_{obj}\right)^2+\left(y_i-y_{obj}\right)^2+\left(z_i-z_{obj}\right)^2}$，$(x_i,y_i,z_i)$为无人机的当前位置，$(x_{obj},y_{obj},z_{obj})$表示当前目标点的位置，$\xi_1$为目标效用系数。

避障效用函数与无人机与障碍物的相对距离有关，促使无人机躲避障碍物并最终安全抵达目标。为了研究方便，本文以球体表示障碍物，障碍物的中心剖面图如图2所示。其中，O_{obs}表示障碍物球体的球心，R_{obs}为障碍物球体的半径，R_{threat}表示障碍物所形成的威胁区域的半径。因此设计避障效用函数如下：

$$U_{i_{obs}}(k)=\xi_2\left(d_{vir_{obs}}-d_{vir_i}-R_{threat}\right) \tag{10}$$

式中，$d_{vir_{obs}}=\sqrt{\left(x_{vir}-x_{obs}\right)^2+\left(y_{vir}-y_{obs}\right)^2+\left(z_{vir}-z_{obs}\right)^2}$

$d_{vir_i}=\sqrt{\left(x_i-x_{vir}\right)^2+\left(y_i-y_{vir}\right)^2+\left(z_i-z_{vir}\right)^2}$ $(x_{obs},y_{obs},z_{obs})$为障碍物球体的球心位置，$(x_{obs},y_{obs},z_{obs})$为多无人机队形的集合中心，$\xi_2$为避障效用系数。

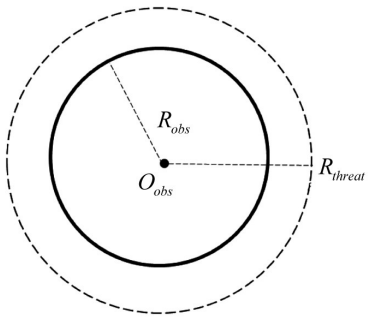


图2 障碍物球体中心剖面图

在实际编队飞行应用中，设机载传感器的最大检测距离为$L_{\max}$。当$d_{i_obs}\leqslant L_{\max}$时，$\xi_1=0$且$\xi_2=1$，当$d_{i_obs}>L_{\max}$时，$\xi_1=1$且$\xi_2=0$。这表示无人机在未检测到障碍物时抵达目标为最高优先，当检测到障碍物时则以避障为最高优先。

避碰效用函数与无人机间相对距离相关，可防止飞行过程中无人机互相碰撞。因此设计避碰效用函数如下：

$$U_{i_obs}(k)=\xi_3\left(d_{ij}-d_{\min}\right) \tag{11}$$

式中，$d_{ij}=\sqrt{\left(x_i-x_j\right)^2+\left(y_i-y_j\right)^2+\left(z_i-z_j\right)^2}$ $d_{\min}$为无人机之间的允许最小距离，ξ_3为避碰效用系数。当$d_{ij}<d_{\min}$时，$\xi_3=1$；否则，$\xi_3=0$。

2.3 评价网络的实现

评价网络采用BP神经网络结构。评价网络的输出是最优性能指标的估计值，隐藏层则采用双极性sigmoidal函数。评价网络在k时刻的输入为

$$In_C(k)=\left[x(k),u(k)\right] \tag{12}$$

隐含层的输入和输出则通过下式计算得出：

$$Ch_{in}(k)=In_C(k)\cdot W_{c1}(k) \tag{13}$$

$$Ch_{out}(k)=\frac{1-e^{-Ch_{in}}}{1+e^{-Ch_{in}}} \tag{14}$$

最后得出评价网络的输出为：

$$\hat{J}(k)=Ch_{out}(k)\cdot W_{c2} \tag{15}$$

式中，W_{c1}和W_{c2}分别为输入层与隐藏层之间的权值和隐藏层与输出层之间的权值。为了实现控制要求，采用梯度下降法对神经网络进行训练。评价网络误差为

$$e_C(k)=\hat{J}(k)-U(k)-\gamma\hat{J}(k+1) \tag{16}$$

$$E_c(k)=\frac{1}{2}e_C^2(k) \tag{17}$$

由此权值更新律可得出：

$$\Delta w_c(k)=-l_c\frac{\partial E_c(k)}{\partial w_c(k)}=-l_c\frac{\partial E_c(k)}{\partial J(k)}\frac{\partial J(k)}{\partial w_c(k)} \tag{18}$$

$$w_c(k+1)=w_c(k)+\Delta w_c(k) \tag{19}$$

下面证明神经网络权值的收敛性。

定理1 设评价网络的最优权值为$W_{c2}^*(k)$，采用如下目标代价函数：$J\left(x_k\right)=W_{c2}^{*T}(k)Ch_{out}(k)$。其中，$Ch_{out}(k)$为隐藏层的输出。当评价神经网络权值更新律为式(18)、式(19)，学习率l_c满足：

$$0<l_c<2/\left\|Ch_{out}(k)\right\|^2 \tag{20}$$

则权值$W_{c2}(k)$渐进收敛于最优权值$W_{c2}^*(k)$。

证明：令权值误差为$\hat{W}_{c2}(k)=W_{c2}(k)-W_{c2}^*(k)$，由式(19)可得

$$\hat{W}_{c2}(k+1)=\hat{W}_{c2}(k)-l_ce_C(k)Ch_{out}(k) \tag{21}$$

定义李雅普诺夫函数为$V_c=tr\left\{\hat{W}_{c2}^T(k)\hat{W}_{c2}(k)\right\}$，则其变化率如下：

$$\Delta V_c=tr\left\{\hat{W}_{c2}^T(k+1)\hat{W}_{c2}(k+1)-\hat{W}_{c2}^T(k)\hat{W}_{c2}(k)\right\} \tag{22}$$

将式(21)代入上式并化简可得：

$$\Delta V_c=l_c\left\|e_C(k)\right\|^2\left(l_c\left\|Ch_{out}(k)\right\|^2-2\right) \tag{23}$$

由于$Ch_{out}(k)$是有限的，则只需$l_c\|Ch_{out}(k)\|^2-2<0$，即$l_c<2/\|Ch_{out}(k)\|^2$成立，便可得$\Delta V_c<0$。所以，评价神经网络权值$W_c(k)$渐进收敛于最优网络权值$W_c^*(k)$。证毕。

神经网络第一层的权值$W_{c1}(k)$收敛于最优权值$W_{c1}^*(k)$的证明与上述过程相似，在此不再赘述。

2.4 执行网络的实现

执行网络也采用BP神经网络结构。执行网络的隐藏层的激活函数同样采用双极性sigmoid函数。执行网络的输入为

$$In_a(k)=x(k) \tag{24}$$

隐藏层的输入和输出由下式进行计算：

$$Ah_{in}(k)=In_a(k)\cdot W_{a1}(k) \tag{25}$$

$$Ah_{out}(k)=\frac{1-e^{-Ah_{in}}}{1+e^{-Ah_{in}}} \tag{26}$$

由此可知，执行网络的输出为

$$u(k)=Ah_{out}(k)\cdot W_{a2}(k) \tag{27}$$

式中，W_{a1}和W_{a2}分别为输入层与隐藏层之间的权值和隐藏层与输出层之间的权值。执行网络的误差为

$$e_a(k)=\hat{J}(k)-U_c \tag{28}$$

$$E_a(k)=\frac{1}{2}e_a^2(k) \tag{29}$$

式中，U_c为期望效用函数，本文令$U_c=0$。执行网络的训练同样利用梯度下降法进行。由此权值更新律可得出：

$$\Delta w_a(k)=-l_a\frac{\partial E_a(k)}{\partial w_a(k)}=-l_a\frac{\partial E_a(k)}{\partial J(k)}\frac{\partial J(k)}{\partial u(k)}\frac{\partial u(k)}{\partial w_a(k)} \tag{30}$$

$$w_a(k+1)=w_a(k)+\Delta w_a(k) \tag{31}$$

定理2 设执行神经网络的最优权值为$W_{a2}^*(k)$，采用如下目标控制律：$u\left(x_k\right)=W_{a2}^{*T}(k)Ah_{out}(k)$。其中，$Ah_{out}(k)$为隐藏层的输出。当执行神经网络权值更新律为式(30)、式(31)，学习率l_a满足：

$$l_a\le 2/\left\|Ah_{out}(k)\right\|^2\left\|Z(k)\right\| \tag{32}$$

式中，$Z(k)=\partial J(k)/\partial u(k)$，则权值$W_{a2}(k)$渐进收敛于最优权值$W_{a2}^*(k)$。

证明：令权值误差为$\hat{W}_{a2}(k)=W_{a2}(k)-W_{a2}^*(k)$，由式(31)可得：

$$\hat{W}_{a2}(k+1)=\hat{W}_{a2}(k)-l_ae_a(k)Z(k)Ah_{out}(k) \tag{33}$$

式中，$Z(k)=\partial J(k)/\partial u(k)$。定义李雅普诺夫函数为$V_a=tr\left\{\hat{W}_{a2}^T(k)\hat{W}_{a2}(k)\right\}$，则其变化率为

$$\Delta V_a=tr\left\{\hat{W}_{a2}^T(k+1)\hat{W}_{a2}(k+1)-\hat{W}_{a2}^T(k)\hat{W}_{a2}(k)\right\} \tag{34}$$

将式(33)代入上式并化简可得：

$$\Delta V_a=l_a\left\|e_c(k)\right\|^2\left\|Z(k)\right\|\times\left(l_a\left\|Ah_{out}(k)\right\|^2\left\|Z(k)\right\|-2\right) \tag{35}$$

由于$Ah_{out}(k)$和$Z(k)$是有限的，因此只要$l_a\|Ah_{out}(k)\|^2$

$\|Z(k)\|-2<0$，即$l_a \leqslant 2/\|Ah_{out}(k)\|^2\|Z(k)\|$，便可得$\Delta V_a<0$。

所以，执行神经网络权值W_{a2}渐进收敛于最优网络权值$W_{a2}^*(k)$。证毕。

神经网络第一层的权值$W_{a1}(k)$收敛于其最优权值$W_{a1}^*(k)$的证明与上述过程相似，在此亦不再赘述。

3 仿真验证

为验证上述方法的有效性和可行性，通过MATLAB进行仿真验证。以3架无人机为例(机队形如图3所示)，分别对单障碍物和双障碍物的两类环境进行仿真实验。UAV_{vir}为无人机队形的几何中心。

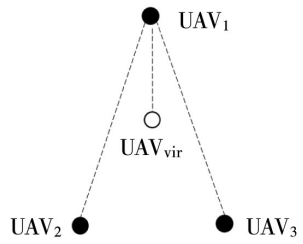


图3 3架无人机队形示意图

3架无人机的初始位置、目标位置和障碍物中心位置如表1所示。其他预设参数为：障碍物的半径R_{obs}=0.4 km，威胁半径R_{threat}=1 km，折扣因子γ=0.95，评价网络和执行网络的学习率为$l_c=l_a$=0.2，满足定理1和定理2的权值收敛条件。

表1 无人机和障碍物的位置设定

坐标类型	坐标值/km
UAV1初始位置	(1.2,1.5,1.3)
UAV2初始位置	(1.3,1.2,1)
UAV3初始位置	(0.8,1.2,1.2)
UAV1目标位置	(15.2,15.5,15.3)
UAV2目标位置	(15.3,15.2,15)
UAV3目标位置	(14.8,15.2,15.2)
障碍物中心位置	(11,11,11)

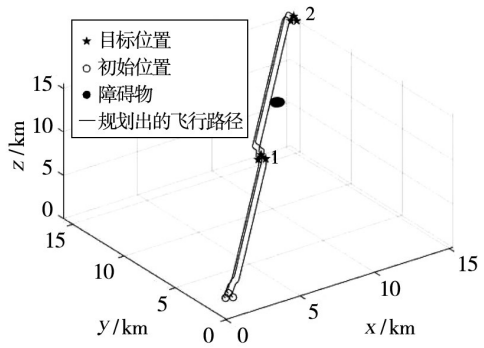


图4 单障碍物时三维路径规划图

图4为出现单障碍物时多无人机的三维路径规划图，其中黑色圆圈为初始位置，五角星为目标位置，黑色实线则是各无人机的飞行轨迹。从图中可以看出，无人机集群在没有障碍物时可以成功抵达目标位置；当出现障碍物时，无人机则调整策略改变路径，从而实现安全避障。

为了进一步验证本方法的有效性，设置两个障碍物的外部环境。预设该障碍物的中心位置为O_{obs2}(12,12.6,12.5)，半径为R_{obs2}=0.5 km，威胁半径为R_{threat}=1 km，其他参数和环境保持不变。此时的三维路径图如图5所示，虚线为单障碍物时的原路径曲线，实线是最新路径轨迹。可以看出，在无障碍物时，两线重合；出现两个障碍物时，最新路径实现了安全避障和最优路径规划。

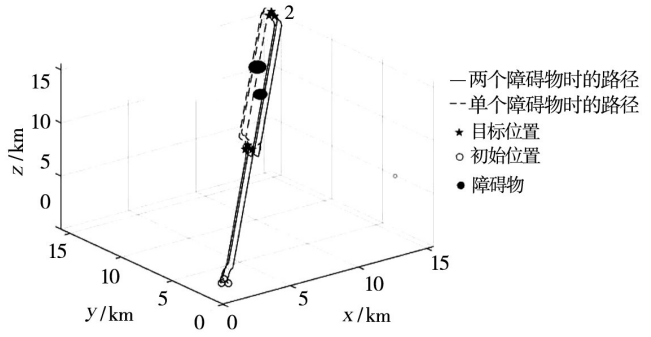


图5 两个障碍物时三维路径规划图

图6为图5的$x-z$方向视图，可以看出，在整个飞行过程中无人机集群保持固定队形不变并能够成功抵达目标位置。控制输入如图7所示。无论是没有障碍物时还是出现两个障碍物时，当无人机抵达目标位置时，控制输入均能收敛至零。

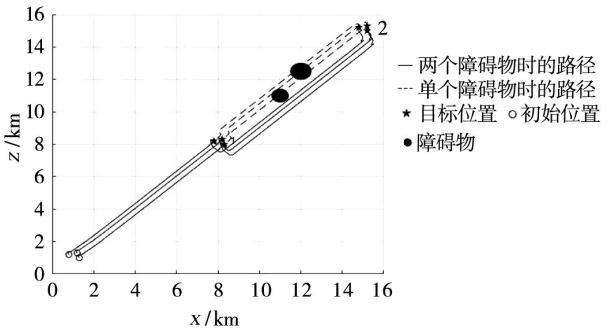


图6 两个障碍物时路径的x-z视图

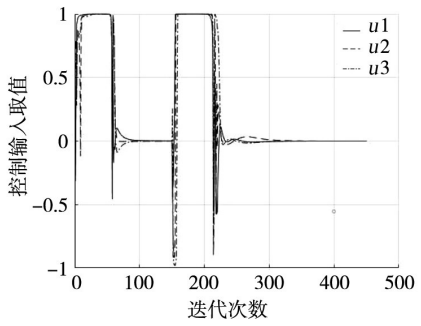


图7 两个障碍物时的控制输入变化曲线

(下转第60页)