

引用格式:张龙杰,贺英政,张龙云.基于ACoreOS的舰载机回收系统设计与实现[J].电光与控制,2020,27(4):61-67. ZHANG L J, HE Y Z, ZHANG L Y. Design and implementation of carrier aircraft recovery system based on ACoreOS[J]. Electronics Optics & Control, 2020, 27(4):61-67.

基于ACoreOS的舰载机回收系统设计与实现

张龙杰¹, 贺英政¹, 张龙云²

(1. 海军航空大学岸防兵学院, 山东 烟台 264001; 2. 山东大学岩土与结构工程研究中心, 济南 250061)

摘要: 聚焦舰载机回收问题的应用实际,提出一种以安全性作为首要约束条件、以实用性作为主要研究目标的舰载机回收决策方案。综合应用考虑逃逸复飞队列饱和度的回收策略和考虑优先级的时域区间集算法,首先从舰载机编队安全性出发求解最优回收模式,然后通过考虑舰载机个体优先级的时域区间集算法,将不同随机性因素归一化到带有优先级的时域区间集中,最终确定各舰载机在最优回收模式中的回收位置。在理论分析的基础上,进一步开展舰载机回收演示系统的设计工作,以ACoreOS嵌入式操作系统为平台,面向装备应用,完成舰载机回收辅助决策演示系统的设计。实验结果表明,在确保回收安全性的前提下,所设计的决策方案能够有效完成舰载机的回收工作。

关键词: 舰载机; 回收决策; 逃逸复飞; ACoreOS嵌入式操作系统

中图分类号: V271.4 **文献标志码:** A **doi:**10.3969/j.issn.1671-637X.2020.04.012

Design and Implementation of Carrier Aircraft Recovery System Based on ACoreOS

ZHANG Longjie¹, HE Yingzheng¹, ZHANG Longyun²

(1. College of Coastal Defense, Naval Aviation University, Yantai 264001, China;

2. Geotechnical & Structural Engineering Research Center, Shandong University, Ji'nan 250061, China)

Abstract: The paper focuses on the engineering practice of carrier aircraft recovery, and proposes a recovery strategy of carrier aircraft, which takes security as the primary constraint and practicality as the main objective. To achieve this, a recovery strategy of carrier aircraft based on saturation of the queue which first bolts and then waves off and a time interval set algorithm based on priority are combined. First, in view of the safety of the carrier aircraft formation, the optimal recovery mode is obtained. Then, by using the time interval set algorithm which considers the priority of individual carrier aircraft, different random factors are normalized to the time interval set with priority. Finally, all the carrier aircrafts are mapped into the optimal recovery mode to establish the final recovery sequence. Furthermore, based on theoretical analysis, the design of an equipment-oriented decision-making demonstration system for carrier aircraft recovery is completed on the platform of ACoreOS EOS. Experimental results indicate that, under the premise of recovery safety, the designed recovery strategy of carrier aircraft can implement the task effectively.

Key words: carrier aircraft; recovery strategy; bolt and wave off; ACoreOS EOS

0 引言

舰载机战斗群是航母编队的主要作战力量和关键战斗力指标之一,而舰载机的回收调度能力又显著影响着整个机群的战斗力。随着国产航母搭载的舰载机

越来越多,如何安全、高效地实现大机群出动模式下舰载机的回收是一个亟待解决的问题。

在舰载机回收方面,国内学者从不同角度进行了探索。文献[1]构建了基于航空作业设施的舰载机出动回收网络,研究了随机故障及遭敌打击情况下舰载机出动回收网络的鲁棒性;文献[2]通过定义最优飞行甲板作业计划的概念,从甲板作业优化的角度,分析了舰载机出动数量、时间等对舰载机出动架次的影响;文献[3]通过研究美、英、法、俄等国家舰载机的实际应用情况,分析了影响航母舰载机出动回收能力的主要因素,指出实战和演习的重要性;文献[4]梳理了舰

收稿日期:2019-05-10

修回日期:2019-05-25

基金项目:国家自然科学基金(51809156);中国博士后科学基金(2016M600537)

作者简介:张龙杰(1983—),男,山东莱阳人,博士,讲师,研究方向为军用计算机仿真技术、武器控制技术。

载机回收流程,通过构建舰载机回收时的运行模型,从系统动力学理论出发研究了舰载机的回收调度问题;文献[5]则考虑了舰载机逃逸复飞的特性,提出了一种动态更新回收队列的方法,旨在为舰指挥人员的决策提供数据支持。这些文献的研究成果为舰载机的回收提供了有益参考,舰载机的回收是一项高风险的工作,安全、实用是舰载机回收时首先要考虑的问题,但从公开发表的文献看,围绕这些方面开展的研究并不多。

关于舰载机的回收问题,从安全角度出发,笔者前期开展了一些理论研究^[6-7],取得了一些实用性结论。文献[6]从燃油消耗、回收时间等指标出发,提出了一种考虑优先级的时域区间集算法,构建了一种考虑正常待机队列中舰载机安全性的回收策略。但是由于寻优指标的设置问题,该模式不能对紧急待机队列中舰载机数量过多的问题做出有效预防,从而无法保证紧急队列中舰载机的安全问题。文献[7]以紧急队列中舰载机待机数量作为约束条件,提出一种考虑逃逸复飞队列饱和度的回收策略。在减轻着舰指挥官和通信系统工作负担、降低舰载机飞行员操作强度、提升舰载机飞行安全的基础上,通过概率分析动态设计专用回收点,解决了紧急待机队列与正常待机队列中舰载机抢夺回收点资源的问题,但理论分析方面只给出了最优回收模式,没有求解最终的回收序列。

本文在前述研究成果的基础上,采取相同的分析思路,以安全性作为首要约束条件,以实用性作为研究目标,将考虑逃逸复飞队列饱和度的回收策略和考虑优先级的时域区间集算法结合起来,首先根据约束条件求解最优回收模式,在此基础上,通过考虑优先级的时域区间集算法,将不同随机性因素归一化为带有优先级的时域区间集中,最终确定各舰载机在最优回收模式中的回收位置。该策略具有以下特点:其一,确保为正常待机队列中舰载机的首次着舰尝试提供一个稳定的回收点,使得正常待机队列中的舰载机不受新加入的舰载机的影响,确保载机回收的安全性;其二,立足问题实用性,考虑舰载机战损以及飞行员受伤等紧急情况;其三,能够针对不同的着舰环境(例如海况、气象条件、昼夜时间等)提供不同的决策方案。此外,在理论分析的基础上,进一步开展舰载机回收演示系统的设计工作,以 ACoreOS 操作系统为软件平台,面向装备应用,完成舰载机回收辅助决策演示系统的设计。

1 舰载机回收理论

1.1 舰载机典型回收流程

舰载机回收时一般按马歇尔航线进入待机队列,

典型回收过程如图1所示。

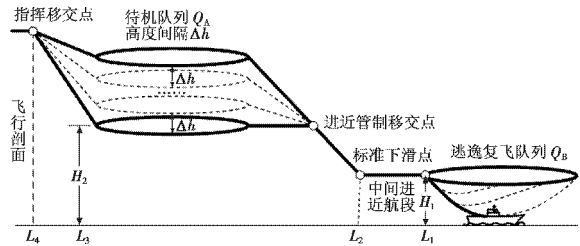


图1 舰载机典型回收模式

Fig.1 Typical recovery model of carrier aircraft

舰载机编队在距离母舰后方 L_4 处移交指挥权,进入回收过程。在距离母舰后方 L_3 处,从高度 H_2 开始,待回收舰载机以高度差 Δh 按马歇尔航线进入待机队列 Q_A 。通过精确计算每一架舰载机离开待机航线、切入航母甲板下滑航线的位置和时刻,能够保证每架舰载机的回收时间基本一致,这样在设计回收点时,通过回收点的顺序就可以确定回收时刻。

与民航飞机不同,舰载机编队实际回收过程中,当出现以下情况时^[8],需要复飞:1) 飞行甲板出现阻塞等突发情况而无法着舰时,舰载机需要紧急机动并复飞;2) 着舰下滑阶段舰载机速度和位姿参数出现较大变化,飞行员或指挥官认为无法成功完成此次着舰时,需要复飞;3) 舰载机着舰时未成功挂住拦阻索,此时舰载机必须逃逸复飞。

因此,并不是所有的舰载机都能够一次性着舰成功。当舰载机逃逸复飞时,会进入一个高度 H_1 的逃逸复飞队列 Q_B 来等待下一次着舰尝试。

1.2 舰载机回收策略模型

舰载机回收过程中,建立回收策略模型时需考虑以下几点因素:1) 以安全性作为首要约束条件,以实用性作为回收策略的决策目标;2) 考虑舰载机逃逸复飞特性以及飞行员受伤、飞机战损等特情;3) 为待机队列中首次着舰尝试的舰载机提供一个稳定的回收点。

基于以上3点考虑,在回收策略中按等间隔时间序列设计两类回收点,即A类和B类,分别用于回收 Q_A 和 Q_B 中的舰载机。

从安全角度考虑,不允许逃逸复飞队列 Q_B 中滞留过多的舰载机。因此,以逃逸复飞队列 Q_B 中舰载机数量作为约束条件,一旦队列 Q_B 中舰载机存在的概率值超过预期值 p_0 ,则相应地安排一个B类回收点,回收队列 Q_B 中可能存在的舰载机。

定义回收策略 $S_m(n) = \{s_a^1, \dots, s_a^i, s_b^1, s_a^{i+1}, \dots, s_a^k, s_b^n, s_a^{k+1}, \dots, s_a^m\}$ 为一个拥有 m 个A类回收点(下标 a 表示)和 n 个B类回收点(下标 b 表示)的有序集合。

定义函数 δ_k : 如果第 k 个 A 类回收点之后的回收点类型为 A, 则 $\delta_k = 0$; 如果第 k 个 A 类回收点之后的回收点类型为 B, 则 $\delta_k = 1$ 。

由以上定义, 第 k 个 A 类回收点在整个回收策略中的位置为

$$s_a^k = k + \sum_{i=1}^{k-1} \delta_i \quad (1)$$

第 $k+1$ 个 A 类回收点之前, 队列 Q_B 中存在第 x 号舰载机的概率为

$$p_{u_k}^x = p_{u_{k-1}}^x (1 - p_x)^{\delta_k} + p_{u_{k-1}}^1 p_1 p_2 \cdots p_{x-1} (1 - p_x) \delta_k + p_{u_{k-1}}^2 p_2 p_3 \cdots p_{x-1} (1 - p_x) \delta_k + \cdots + p_{u_{k-1}}^{x-1} p_{x-1} (1 - p_x) \delta_k = p_{u_{k-1}}^x (1 - p_x)^{\delta_k} + R(k) \quad (2)$$

式中: u_k 表示第 $k+1$ 个 A 类回收点之前的回收点位置,

$u_k = s_a^k + \delta_k = k + \sum_{i=1}^k \delta_i$; p_x 为当前舰载机成功回收的概率;

$R(k) = \sum_{h=1}^{x-1} [p_{u_{k-1}}^h (\prod_{j=h}^{x-1} p_j) (1 - p_x) \delta_k]$, $2 \leq x \leq k-1$ 。

第 $k+1$ 个 A 类回收点之前, 队列 Q_B 中至少存在一架舰载机的概率为

$$P_{Bk} = p_{u_k}^1 + \sum_{x=2}^{k-1} p_{u_k}^x + p_{u_k}^k \quad (3)$$

式中: $p_{u_k}^1 = p_{u_{k-1}}^1 (1 - p_1)^{\delta_k}$; $p_{u_k}^k = (1 - P_{B(k-1)}) (1 - p_k)^{1+\delta_k} + p_{u_{k-1}}^1 p_1 p_2 p_3 \cdots p_{k-1} (1 - p_k) \delta_k + p_{u_{k-1}}^2 p_2 p_3 \cdots p_{k-1} (1 - p_k) \delta_k + \cdots + p_{u_{k-1}}^{k-1} p_{k-1} (1 - p_x) \delta_k = (1 - P_{B(k-1)}) (1 - p_k)^{1+\delta_k} + \sum_{h=1}^{x-1} [p_{u_{k-1}}^h (\prod_{j=h}^{x-1} p_j) (1 - p_x) \delta_k]$ 。

进一步迭代计算第 k 个 A 类回收点之后, Q_B 中至少存在一架舰载机的概率为

$$P_{Ak} = P_{Bk|\delta_k=0} = p_{u_k}^1 \delta_k = 0 + \sum_{x=2}^{k-1} p_{u_k}^x \delta_k = 0 + p_{u_k}^k \delta_k = 0 \quad (4)$$

由 δ_k 的定义, 有

$$\delta_k = \begin{cases} 0 & P_{Ak} < P_0 \\ 1 & P_{Ak} \geq P_0 \end{cases}$$

则第 $k+1$ 个 A 类回收点在策略 $S_m(n)$ 中的位置为

$$s_a^{k+1} = \begin{cases} s_a^k + 1 & \delta_k = 0 \\ s_a^k + 2 & \delta_k = 1 \end{cases} \quad (5)$$

1.3 舰载机回收序列

基于考虑优先级的时域区间集算法, 求解舰载机在既定回收模式下的回收序列。考虑的指标包括: 舰载机到达距离 D , 舰载机燃油余量 O , 人员状态 η 和战机状态 ρ 。其中, η 和 ρ 为二值数据。对于第 i ($i=1, 2, \dots, m$) 架待回收的舰载机, 可进行舰载机回收的时域区间集为

$$R_{OD}^{(i)} = [R_{OD_min}^{(i)}, R_{OD_max}^{(i)}] \quad (6)$$

式中,

$$\begin{cases} R_{OD_min}^{(i)} = \max \{ D^{(i)} / V_0, (O^{(i)} - O_{\max}) / h \} \\ R_{OD_max}^{(i)} = (O^{(i)} - O_{\min}) / h \end{cases} \quad (7)$$

式中: V_0 为舰载机飞行速度; O_{\min} 和 O_{\max} 分别为舰载机着舰时的最少和最大油量要求; h 为舰载机每小时的油耗。由于舰载机每架次的回收时间基本相同, 设其为 τ_0 , 则对于策略 $S_m(n)$, 各个回收点对应的时刻 $t_\kappa = \kappa \tau_0$, $\kappa = 1, 2, \dots, m+n$ 。

利用 t_κ 对式(6)进行离散化, 即执行 $R_{OD}^{(i)} \cap \{t_\kappa\}$ 运算, 得到第 i 架舰载机可用的回收时刻点集为

$$N_{OD}^{(i)} = \{l_1^{(i)} \tau_0, l_2^{(i)} \tau_0, \dots, l_j^{(i)} \tau_0\} \quad (8)$$

在式(8)的基础上, 综合考虑人员状态 η 和战机状态 ρ 两个影响因素, 通过逻辑运算, 为对应的离散时域区间集 $N_{OD}^{(i)}$ 增加优先级, 即

$$P^{(i)} = \eta^{(i)} \oplus \rho^{(i)} \quad (9)$$

式中, \oplus 为异或运算操作符。式(8)和式(9)即为考虑优先级的各舰载机的可回收时域区间集模型, 在此基础上确定相对回收序列 $S^s(m)$, 准则如下:

1) 扫描 N_{OD} , 寻找 N_{OD} 中所有回收时刻点的最小值 τ_{\min} , 令 $N_U = N_{OD} - \tau_{\min} + \tau_0$;

2) 将拥有回收时刻点 τ_0 的舰载机作为首架被回收的舰载机, 如果多架舰载机都可以在该回收点被回收, 则继续按照以下法则确定回收次序;

3) 优先级高的先安排回收;

4) 优先级相同时, 剩余可用回收点数少的优先安排回收;

5) 如果剩余回收点数相同, 则总可用回收点数多的优先安排回收;

6) 如果按照以上准则仍然无法确定顺序, 则随机选择进行回收。

由式(1)~式(5)求解最优回收策略 $S_m(n)$, 将 $S^s(m)$ 向 $S_m(n)$ 映射, 就可以得到最终的回收序列 $S_m^s(n)$ 。

2 舰载机回收辅助决策系统设计

2.1 硬件平台

以 ACOREOS 嵌入式设备作为硬件开发平台, 开展舰载机回收演示系统的设计工作。天脉 ACOREOS 是一款能够提供高效多任务调度、中断/异常管理以及任务间通信的嵌入式实时操作系统, 自 2014 年发布以来, 已广泛应用于军事、航空、航天等高精尖领域。针对航空电子系统的应用需求, ACOREOS 还提供嵌入式强实时操作系统兼容包、OpenGL 图形支持功能、C 库等可配置的功能组件。

ACOREOS 设备拥有丰富的外部接口, 适配性好。在 ACOREOS 设备的基础上, 搭建辅助决策系统。系统开发及应用环境如图 2 所示。

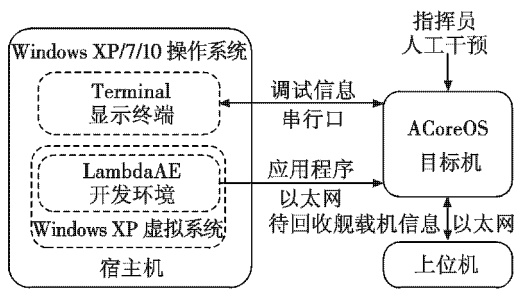


图2 舰载机回收辅助决策系统开发应用环境

Fig. 2 Development and application environment of decision-making system of carrier aircraft recovery

宿主机在虚拟 Windows XP 系统环境下通过 ACoreOS 开发环境 LambdaAE 完成应用程序的开发和编译工作,通过以太网将程序烧写到 ACoreOS 目标机中,并借助串口完成系统调试。舰载机回收过程中,上位机收集各舰载机的参数,通过以太网口实时传递给 ACoreOS 目标机,由目标机解算出最优回收序列,同时进行仿真演示。指挥员可以通过 ACoreOS 目标机上的按键对回收序列进行人工干预,灵活处理突发情况。

2.2 系统可视化建模

系统仿真界面由两部分信息组成,即静态回收轨迹信息和舰载机动态位置信息。

2.2.1 静态回收轨迹信息建模

如图3所示,在 ACoreOS 设备显示屏幕上建立屏幕坐标系 OXY ,结合图1,绘制舰载机静态回收轨迹曲线 $e_1 \sim e_4$ 以及 $l_1 \sim l_7$ 。

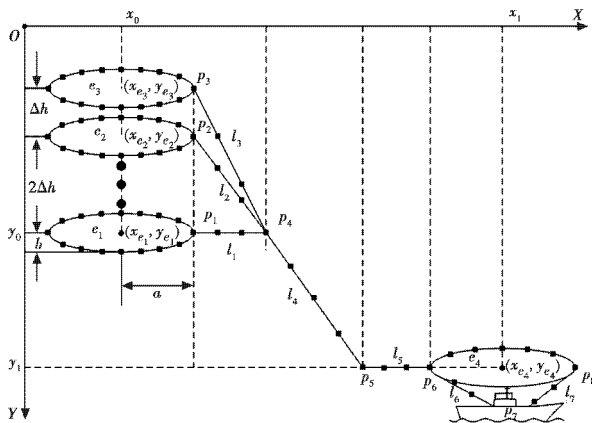


图3 系统可视化建模

Fig. 3 System visualization modeling

根据舰载机的待机运行轨迹,建立 $e_1 \sim e_4$ 段的曲线轨迹方程为

$$\begin{cases} y_{e1} = y_0 + \sqrt{b^2 - (x_{e1} - x_0)^2 b^2 / a^2} \\ y_{e2} = y_0 - 2\Delta h + \sqrt{b^2 - (x_{e2} - x_0)^2 b^2 / a^2} \\ y_{e3} = y_0 - 3\Delta h + \sqrt{b^2 - (x_{e3} - x_0)^2 b^2 / a^2} \\ y_{e4} = y_1 + \sqrt{b^2 - (x_{e4} - x_1)^2 b^2 / a^2} \end{cases} \quad (10)$$

改出及下滑段 $l_i (i=1, 2, \dots, 7)$ 的轨迹近似为直线,对应的方程为

$$y_{l_i} = \begin{cases} y_{p_i} & i=1, 5 \\ y_{p_i} + (y_{p_{i+1}} - y_{p_i}) \frac{x_{l_i} - x_{p_i}}{x_{p_{i+1}} - x_{p_i}} & \text{其他} \end{cases} \quad (11)$$

舰载机回收过程中,按照图3中既定的轨迹运动。仿真过程中,需要实时解算舰载机在回收轨迹中的位置。

2.2.2 舰载机动态位置建模

1) 盘旋待机段舰载机实时位置。

基于舰载机运动假设、最小仿真步长设置 (t_{step}) 以及显示路径和舰载机实际运动路径的比例 λ_{e_i} , 计算舰载机运动轨迹的一系列关键点。首先根据实际待机盘旋轨迹长度 L_0 计算关键点个数 N_e 为

$$N_e = \lceil L_0 / (\lambda_{e_i} V_0 t_{step}) \rceil \quad (12)$$

则仿真系统中,待机盘旋轨迹上相邻两个关键点间的距离为

$$\Delta l = \frac{2\pi b + 4(a-b)}{N_e} = \frac{2\pi b + 4(a-b)}{\lceil L_0 / (\lambda_{e_k} V_0 t_{step}) \rceil} \quad (13)$$

而待机轨迹上任意弧长为

$$S = \int_0^x \sqrt{a^2 \sin^2 t + b^2 \cos^2 t} dt \quad (14)$$

令 $S = \Delta l$, 由式(10)和式(11)求解出仿真系统中待机轨迹上任意点 $k (k \in [1, N_e])$ 的坐标为 $(x_{e_k} + a \cos kt, y_{e_k} - b \sin kt)$, 其中, (x_{e_k}, y_{e_k}) 为各待机轨迹的中心点坐标, $k=1, 2, \dots, 4$ 。

2) 下滑改出段舰载机实时位置。

舰载机从待机航线改出后,在 $l_1 \sim l_7$ 各阶段的运动轨迹近似为直线,可统一记作 $y_{l_k} = k_{l_k} x_{l_k} + b_{l_k}$ 。仿真系统中,单个仿真步长 t_{step} 内舰载机运动的距离为 $\lambda_{l_k} V_0 t_{step}$, λ_{l_k} 为实际路径与系统显示路径的比例因子。假设舰载机进入各个阶段的时刻为 0, 则在第 k 次仿真中,舰载机在 $l_1 \sim l_7$ 段内的实时坐标为

$$\begin{cases} x_{l_k} = x_{p_k} + k \lambda_{l_k} V_0 t_{step} / \sqrt{k_{l_k}^2 + 1} \\ y_{l_k} = y_{p_k} + k_{l_k} k \lambda_{l_k} V_0 t_{step} / \sqrt{k_{l_k}^2 + 1} \end{cases} \quad (15)$$

式中, (x_{p_k}, y_{p_k}) 为 p_k 点的坐标。

3 系统软件设计

3.1 系统软件架构

ACoreOS 是一款针对航空应用需求设计的、面向多任务应用的强实时嵌入式操作系统,能够满足强实时、高安全、高可靠、高确定、可裁剪性、可升级性等特殊环境要求。借助配套的集成开发环境 LambdaAE, 用户可以快速完成嵌入式应用的开发、调试和部署。典

型的 ACOREOS 系统软件架构如图 4 所示。



图 4 ACOREOS 系统软件架构

Fig. 4 Software architecture of ACOREOS system

在舰载机回收决策应用中,为了实现可视化仿真演示,在 ACOREOS 系统中移植了 emWIN GUI 组件。emWIN GUI 提供了丰富的绘图功能,包括 2D 图形库、位图转换器、视窗管理器、窗口对象、用户框等,基于 emWIN GUI 可实现文本显示、图形绘制、颜色修改等丰富功能。

仿真界面更新过程中,通常存在屏幕闪烁的问题,emWIN GUI 提供了分段存储设备函数来解决这个问题:

```

GUI_MEMDEV_Draw(
    GUI_RECT * p Rect, //区域范围
    GUI_CALLBACK_VOID_P * pfDraw,
                                //绘图函数
    void * pData, //传递参数
    int NumLines, //设备行数
    int Flags //透明标志)。
    
```

此接口函数的作用是对图形进行绘制,并利用分段存储设备防止画面闪烁。

3.2 软件设计流程

ACOREOS 系统采用优先级抢占以及时间片轮转策略,通过信号量、事件等任务间同步/互斥机制,实现周期任务、非周期任务的调度与管理。

在系统软件设计中设置两大类任务,即 Task_LandAlg 和 Task_IntfDrw。Task_LandAlg 负责回收策略模型的理论解算,Task_IntfDrw 负责系统演示界面的更新。系统运行过程中,初始化结束后,在每个仿真环节,Task_LandAlg 和 Task_IntfDrw 各运行一次。首先由 Task_LandAlg 解算 t_{step} 内所有舰载机的最新状态,再由 Task_IntfDrw 将所有信息实时更新到仿真界面中。因此任务 Task_LandAlg 和 Task_IntfDrw 之间是一种互斥共享的关系。基于 ACOREOS 的信号量机制实现两类任务的互斥与同步,系统整体软件设计流程如图 5 所示。

Task_LandAlg 和 Task_IntfDrw 互斥地运行在 ACOREOS 系统的两种任务模式下,即就绪和运行,并通过互斥信号量的转移,实现两个任务的同步以及任务间资源的共享。图 6 是任务 Task_LandAlg 和 Task_IntfDrw 的内部运行逻辑。

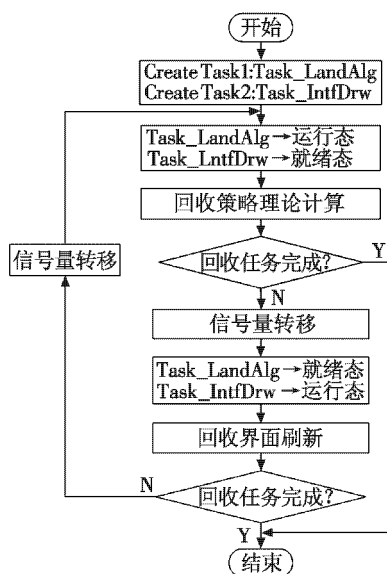


图 5 系统整体软件流程图

Fig. 5 Flow chart of the overall software

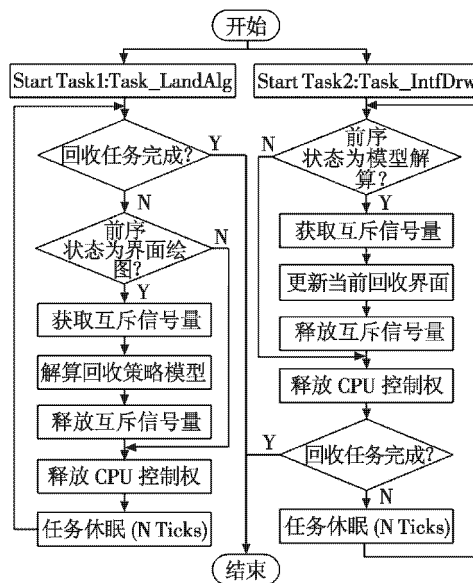


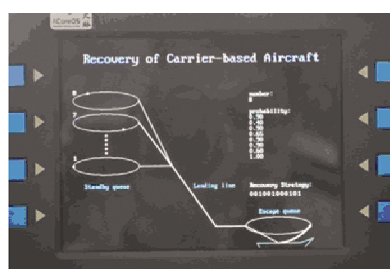
图 6 核心任务函数运行逻辑

Fig. 6 Running logic of core task functions

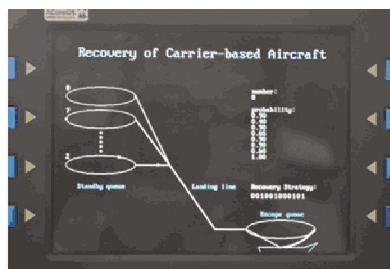
以 Task_LandAlg 为例,任务的创建和启动代码如下:

```

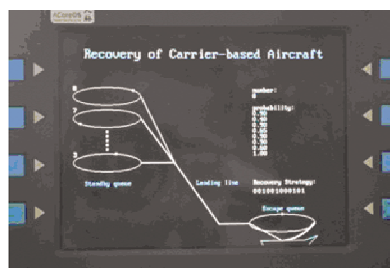
AcoreOS task_create(
    Task_LandAlg, //任务名
    20, //优先级
    &drawTask_stack[0][0], //堆栈栈顶
    ACOREOS_MINIMUM_STACKSIZE * 4 //堆栈大小
    ACOREOS_DEFAULT_ATTRIBUTES, //任务属性
    NULL, //回调函数
    &TaskID_LandAlg //标识符);
AcoreOS_task_start(
    TaskID_LandAlg, //标识符
    
```

(a) 初始运行时刻的态势



(b) 回收第一架舰载机时刻的态势



(c) 发生逃逸复飞后的态势

图8 系统运行结果

Fig. 8 Results of system operation

图8(a)为系统初始运行时刻的态势图,受硬件设备的限制,演示界面上只显示3个待机椭圆。实际回收过程中,根据后台计算得到的回收序列由下至上安排各待回收舰载机,待机椭圆左侧的数字表示当前舰载机在整个待机队列中的相对回收顺序。

系统运行过程中,上方的2个待机椭圆用于演示回收序列中最后两架舰载机的运行情况,最下方的待机椭圆则随着系统的运行实时更新,确保总是演示当前时刻待回收舰载机的运行态势。当舰载机离开待机椭圆航线、切入航母甲板下滑航线以后,则将下一架要回收的舰载机实时显示到该椭圆航线上,如图8(b)所示,为第1架待回收舰载机开始回收以后,待机椭圆航线上实时显示出第2架待回收舰载机的运行态势。

逃逸复飞队列根据后台数学模型的解算实时更新,如图8(c)所示。可以看出,在该次仿真中,在第3

架舰载机进行首次着舰尝试前,队列 Q_B 中已经存在一架逃逸复飞的舰载机,该舰载机将在后续的第一个B类回收点中进行回收尝试。

5 结束语

舰载机的回收是一项表面看似简单实际上却很难做到万无一失的工作。一方面,舰载机回收时对安全性的要求极高,此时很多看似合理的理论排序算法(例如先来先服务、动态更新待机队列等)变得不再适用;另一方面,实际操作中突发意外多,人员、设备状态异常,环境条件突变,都会影响最终的方案决策。本文的工作充分考虑了上述因素,并进一步在国产军用操作系统完成了演示系统的设计工作,能够为装备的国产化、自主化普及提供一些借鉴和经验。

另外,舰载机的回收不是一个孤立的问题。在高强度作战条件下,会存在舰载机回收和起飞交叉进行的情况,此时甲板的空闲与否将成为舰载机回收的另一个重要的影响因素,如何实现这种模式下的高效回收调度,需要进一步深入研究。

参考文献

- [1] 郑茂,王超,黄胜. 舰载机出动回收网络鲁棒性分析[J]. 上海交通大学学报,2013,47(12):1934-1939.
- [2] 周晓光,赵仁厚,王述运,等. 飞行甲板作业对航母舰载机出动架次影响分析[J]. 系统仿真学报,2014,26(10):2447-2451.
- [3] 刘相春,卢晶,黄祥钊. 国外航母舰载机出动回收能力指标体系分析[J]. 中国舰船研究,2011,6(4):1-7.
- [4] 岳奎志,孙聪,罗明强,等. 载机军舰回收舰载机的运行模型[J]. 系统工程与电子技术,2013,35(12):2527-2532.
- [5] 夏国清,陈红召,米青超. 舰载机回收作业的反馈排队模型[J]. 火力与指挥控制,2013,38(5):164-166,169.
- [6] 张龙杰,李德栋,何凡. 一种基于模式寻优的舰载机实时回收方法[J]. 航空计算技术,2016,46(5):13-16.
- [7] 张龙杰,何凡,刘青松. 考虑逃逸复飞队列饱和度的舰载机回收策略[J]. 火力与指挥控制,2017,42(8):15-18,23.
- [8] 沈健,孙逊,王敏文. 舰载机复飞/逃逸技术[J]. 航空科学技术,2009(4):37-40.