طراحی و تنظیم سامانه کنترل پهپاد مبتنی بر الگوریتم یادگیری تقویتی عمیق

تاریخ دریافت: ۱۴۰۳/۰۵/۰۶
تاریخ پذیرش: ۱۴۰۴/۰۲/۲۶

محمدحسین علیزاده[۱]، علیرضا طلوعی[۲]، رضا قاسمی[۳]

۱- کاندید دکتری، دانشکده فناوری‌های نوین و مهندسی هوافضا، دانشگاه شهید بهشتی، تهران، ایران.
۲- دانشیار، دانشکده فناوری‌های نوین و مهندسی هوافضا، دانشگاه شهید بهشتی، تهران، ایران Toloei@sbu.ac.ir.
۳- دانشیار، دانشکده مهندسی برق، دانشگاه قم، قم، ایران.

چکیده

تحلیل دینامیک پهپاد به دلیل تغییر ماموریت در محیط شامل عدم قطعیت، پیچیده و غیردقیق می‌باشد. در این پهپاد علاوه بر شرایط محیطی، به دلیل اختصاص عملگرهای مشترک برای کنترل، تداخل در کانال‌ها ایجاد می‌گردد. در این تحقیق با معرفی دینامیک غیر خطی پهپاد، ناگزیر با انجام ساده‌سازی مدل خطی شده تقریبی استخراج شده‌است. در این مقاله با قیود عملکردی مشخص شده، برای ردیابی زاویه فراز و غلت و همچنین مهار سرعت‌های زاویه‌ای کنترل‌گر کلاسیک طراحی می‌شود. این کنترل‌گرها چون برای تابع تبدیل تقریبی و شرایط بدون حضور عدم قطعیت طراحی می‌شوند، لذا در تمامی وضعیت‌ها و اغتشاش‌ها لزوما رفتار مناسبی ندارند. هدف از این مقاله ارائه روشی برای اصلاح اثر مدل‌سازی غیر دقیق و عدم قطعیت در طراحی سامانه کنترلی کلاسیک است. روش پیشنهادی استفاده از یادگیری تقویتی برای اصلاح ضرایب کنترل‌گر است. محاسبات به صورت خارج از خط صورت گرفته و پس از یادگیری و در فرایند کاری، نتیجه به صورت بهره‌های اصلاحی به کنترل‌گر کلاسیک اعمال می‌گردند. نتایج تحقیق نشان از افزایش حداقل ۲۰ درصدی در متوسط پاداش دریافتی و کاهش سه برابری در تعداد شبیه‌سازی‌های ناپایدار و یا شبه پایدار دارد. به عبارت دیگر قابلیت اطمینان در عملکرد پهپاد افزایش می‌یابد.

واژه‌های کلیدی: *پهپاد، کنترل‌گر کلاسیک، یادگیری ماشین، یادگیری تقویتی، الگوریتم مونت کارلو، شبکه عصبی*

Design and tuning of a UAV control system based on deep reinforcement learning algorithms

Mohammad Hosein[1] Alizadeh, Alireza Toloei[2], Reza Ghasemi[3]

1- PhD candidate, Fculty of New Technologies and Aerospace Engineering, Shahid Beheshti University, Tehran, Iran.
2- Associate Professor, Fculty of New Technologies and Aerospace Engineering, Shahid Beheshti University, Tehran, Iran, Toloei@sbu.ac.ir.
3- Assistant Professor, Electrical Engineering Department, University of Qom, Qom, Iran.

Abstract

Dynamic analysis of UAVs becomes complex and imprecise due to mission changes in uncertain environments. In this UAV, besides environmental conditions, shared actuators used for control introduce interference across control channels. In this study, the nonlinear dynamics of the UAV are first introduced, and an approximate linearized model is derived through simplification. Based on this model, classical controllers are designed under specified performance constraints for pitch and roll angle tracking, as well as for damping angular velocity. However, since these controllers are designed based on approximate transfer functions and under nominal conditions without uncertainty, they may not perform adequately in all situations and disturbances. The main objective of this paper is to propose a method to compensate for modeling inaccuracies and uncertainties in classical control system design. The proposed method employs reinforcement learning to adjust controller parameters. The training process is conducted offline, and the learned corrective gains are then integrated into the classical controller during operation. The results demonstrate at least a 20% increase in the average cumulative reward and a threefold reduction in the number of unstable or quasi-stable simulations, thereby improving the UAV's reliability and performance.

Keywords: *UAV, Classic Controller, Machin Learning, Reinforcement Learning, Monte-Carlo, Neural Network*

۱. مقدمه

روش سنتی طراحی کنترل‌گر برای اجسام پرنده مانند هواپیما یا پهپاد، مبتنی بر محاسبه مدل خطی شده در نقطه نامی و طراحی جبران‌ساز است. سامانه کنترلی را به حداقل کانال‌های غلت و فراز تقسیم کرده و برای هر کانال جبران‌ساز طراحی می‌شود. در این روش با توجه به مدل تقریبی و طراحی در شرایط نامی، در خارج از نقطه طراحی عملکرد ضعیف می‌گردد. امروزه روش‌های نوینی برای بهبود عملکرد توسعه یافته است. یکی از این روش‌ها استفاده از یادگیری ماشین برای بهبود کیفیت در محدوده خارج از ناحیه کار نامی است.

در ادامه به چند نمونه از کاربرد یادگیری ماشین اشاره می‌شود. در مرجع [۱] برای یک کوادروتور، با ادغام کنترل‌گرهای کلاسیک و الگوریتم یادگیری تقویتی، کارایی بهتری برای سیستم کنترل ایجاد شده است. از مزایای این روش، می‌توان به کاهش خطر استفاده خالص از یادگیری تقویتی اشاره نمود. کنترل پیش‌بین مدل یک روش موثر برای کنترل ربات‌ها، به ویژه وسایل نقلیه با دینامیک کند خودمختار می‌باشد. استفاده از این روش از نظر محاسباتی پیچیده بوده و نیاز به تخمین وضعیت سیستم دارد. در محیط های پیچیده و بدون ساختار، استفاده از آن چالش‌برانگیز است. یادگیری تقویتی می‌تواند نیاز به تخمین وضعیت را هموار کرده و روش کنترلی مناسب را ارائه دهد. این الگوریتم علاوه بر امکان اصلاح کنترل‌گر طراحی شده، از تعامل با محیط می‌تواند به صورت مستقیم و با کمک شبکه عصبی عمیق که آموزش داده می‌شود، فرامین کنترلی را پس از قرائت حسگر مستقیم به عملگرها صادر کند.

این نوع از پیاده‌سازی در سامانه‌های ناپایدار و همچنین در ابتدای فرایند آموزش احتمالا به شکست ختم می‌شود. مرجع [۲] یک روش نوین برای یادگیری سیاست کنترلی در پهپادهای خودران ارائه می‌دهد. هدف اصلی این روش، آموزش شبکه‌های عصبی برای کنترل پرنده بدون نیاز به کمک انسانی است. روش پیشنهادی با کمک یک کنترل‌کننده مدل پیش‌بین آموزش دیده و پس از یادگیری شبکه عصبی به صورت مستقل پهپاد را هدایت و کنترل می‌نماید.

در مقاله [۳]، طراحی یک روش کنترل قوی برای پهپاد در شرایط نامعین (اغتشاشات محیطی و عدم‌قطعیت) پیشنهاد شده است. نویسندگان روش جدیدی ارائه می‌دهند که گرادیان سیاست قطعی را با یک کنترل‌گر انتگرالی ترکیب می‌کند. استفاده از این روش توانسته است پایداری، دقت و مقاومت کوادروتور را در کنترل پرواز بهبود بخشد. از نکات مثبت این روش در استفاده غیر مستقیم یادگیری تقویتی می‌باشد. در مرجع [۴]، الگوریتم بهینه‌سازی سیاست مجاورتی برای تصحیح کنترل کننده کوادروتور پیشنهاد شده است. در این روش چون خروجی به صورت مستقیم وارد حلقه کنترل نمی‌شود، خطر استفاده از آن حداقل می‌شود.

در مرجع [۵] به طراحی جبران‌ساز وضعیت یک کوادروتور به روش بهینه‌سازی سیاست مجاورتی پرداخته است. در مرجع [۶] برای پایداری تمامی وضعیت‌های یک کوادروتور، کنترل‌گر مبتنی بر شبکه عصبی بازیگر-منتقد برای بهبود عملکرد ردیابی مسیر ارائه شده است. در مرجع [۷]، با استفاده از الگوریتم

بازیگر-منتقد، به عنوان چارچوب پیاده سازی یادگیری عمیق، مشکل ناوبری در بادهای شدید و اتفاقی برای یک ریز پرنده هموار شده است. استفاده از پهپادهای چند روتور برای کاربردهای صنعتی و عمرانی بسیار رایج شده است و مرجع [۸] توسط الگوریتم گرادیان خط مشی قطعی عمیق کنترل‌گر برای پرنده مورد نظر توسعه داده است. همچنین مراجع بسیاری از جمله [۹-۱۱] در حوزه هوافضا به روش یادگیری ماشین تحقیق نموده‌اند. در مرجع [۱۱] برای یک پهپاد جبران‌ساز کلاسیک طراحی شده و سپس با استفاده از الگوریتم فرااابتکاری، جبران‌ساز اولیه در محیط خطی بهینه شده است. نتایج تحقیق نشان می‌دهد که کنترل‌گر مقاوم شده و درصد ناپایداری‌ها کاهش یافته است.

با توجه به وجود تشابه در فرایند یادگیری سامانه‌های خودران، علاوه بر مراجع ذکر شده به مراجع مربوط به ادوات دریا پایه نیز رجوع می‌شود. در مرجع [۱۲]، کنترل‌گر مبتنی بر مشاهده گر شبکه عصبی تطبیقی برای هواناو از روش مد لغزشی فوق پیچان توسعه داده شده است. در مقاله [۱۳]، الگوریتم ژنتیک بهبود یافته برای تنظیم متغیرهای سامانه کنترل یک هواناو پیشنهاد شده است. هدف، دستیابی به کنترل دقیق و پایدار در حرکت است. الگوریتم ژنتیک برای بهینه‌سازی عملکرد کنترل‌گر استفاده شده و عملکرد آن از نظر پایداری و کاهش خطای ردیابی مسیر، در مقایسه با روش‌های سنتی بهتر گزارش شده است. در مقاله [۱۴]، کنترل ردیابی مسیر برای یک هواناو پیشنهاد می‌شود. با توجه به دینامیک آن، که غیرخطی و به شدت تداخلی می‌باشد، طراحی جبران‌ساز برای آن چالش بزرگی است. برای حل مسئله تداخل، سامانه به صورت یک مجموعه با عملگرهای مستقل در هر کانال فرض می‌شود. سپس متغیرهای غیر خطی به عنوان عدم قطعیت و خطا شناسایی می‌شوند. در مرحله دوم، یک کنترل‌گر رد اختلال مبتنی بر یادگیری تقویتی طراحی شده تا اثر عدم قطعیت جبران گردد.

در مرجع [۱۵]، ابتدا برای یک هواناو که یک سامانه پیچیده است، با استفاده از روش مدلغزشی کنترل‌گر طراحی شده و سپس با کمک از الگوریتم یادگیری تقویتی، کنترلگر طراحی شده متناسب با محدوده عدم قطعیت و مانور برای حالات مختلف بهینه شده است. در مرجع [۱۶]، الگوریتم یادگیری تقویتی عمیق بر اساس بهینه‌سازی سیاست مجاورتی برای کنترل حرکت وسیله نقلیه سطحی بدون سرنشین پیشنهاد شده است. در این مقاله قانون حرکت وسیله نقلیه بر روی سطح آب تجزیه و تحلیل شده و مدل ریاضی سه درجه آزادی ارائه می‌شود. سپس با کمک تابع پاداش برای افزایش کارایی، یادگیری شبکه انجام می‌گیرد.

با بررسی مراجع معرفی شده مشاهده می‌شود که یادگیری ماشین به دو صورت در سامانه‌های کنترل وارد شده است:

۱) کنترل‌گر به صورت یک شبکه عصبی عمیق مستقل است که از تجربیات موجود به تدریج فرایند کنترل را آموخته و به کار می‌گیرد.

۲) واحد یادگیری در کنار کنترل‌گر مستقل بوده و پس از تکمیل یادگیری، متغیرهای کنترل‌گر را متناسب با شرایط بهبود می‌دهد.

پژوهشی که در این مقاله ارائه می‌گردد ساختاری مشابه روش دوم دارد. پس از معرفی دینامیک و استخراج تابع تبدیل، با روش طراحی کلاسیک، در نقطه کار نامی کنترل‌گر طراحی می‌گردد. این کنترل‌گر در شرایط حضور عدم قطعیت و اغتشاش لزوما رفتار مناسبی ندارد. برای بهبود عملکرد استفاده از الگوریتم یادگیری تقویتی برای تطبیق سامانه کنترل با محیط پیشنهاد می‌گردد. با اصلاح و تنظیم کنترل‌گر برای بازه محتملی از مانور و عدم قطعیت، سامانه کنترلی مقاوم می‌شود.

۲. طراحی کنترل کننده کلاسیک

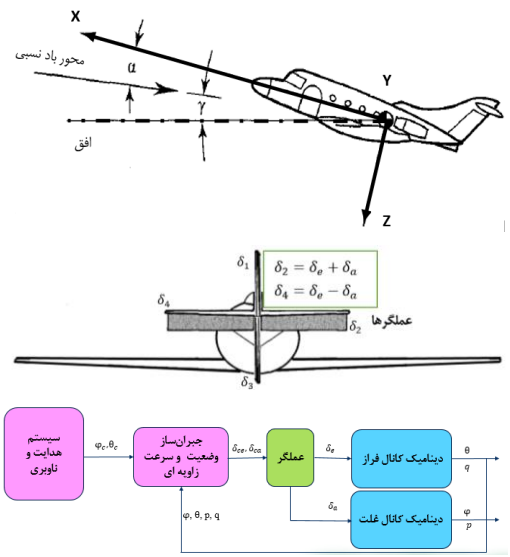
پیش از ورود به طراحی کنترل‌گر معادلات حاکم معرفی می‌شوند. در رابطه (۱) معادلات گشتاور و نیروها آورده شده است. همچنین در تصویر (۱) ساختار کنترلی پهپاد نمایش داده شده است [۱۷].

$$\sum \Delta M_x = \dot{p}I_{xx} - \dot{r}I_{xz} + qr(I_{zz} - I_{yy}) - pqI_{xz}$$
$$\sum \Delta M_y = \dot{q}I_{yy} + pr(I_{xx} - I_{zz}) + (p^2 - r^2)I_{xz}$$
$$\sum \Delta M_z = \dot{r}I_{zz} - \dot{p}I_{xz} + pq(I_{yy} - I_{xx}) + qrI_{xz} \quad (۱)$$
$$\sum \Delta F_x = m(\dot{u} + wq - vr)$$
$$\sum \Delta F_y = m(\dot{v} + ur - wp)$$
$$\sum \Delta F_z = m(\dot{w} + vp - uq)$$

جدول ۱. معرفی متغیرهای تابع تبدیل

متغیر	توضیح	متغیر	توضیح
ϕ, θ	زاویه فراز و غلت	C_{mq}, C_{lp}	مشتق گشتاور به سرعت زاویه‌ای
α	زاویه حمله	U_0	سرعت خطی
Q. S	نیروی آیرودینامیکی وارد بر سطح نرمال	$C_{z\alpha}$	مشتق نیرو به زاویه حمله

I_{ij}	لختی ضربی	I_{ii}	لختی
C_{zq}	مشتق نیرو به سرعت زاویه‌ای	x_{PG}	فاصله مرکز جرم از فشار
F_i, M_i	نیرو و گشتاور	D_x, D_y	بازوی کنترلی
C_x	ضریب پسا	$C_{l\delta}$	ضریب برای بالک‌ها
m	جرم لحظه‌ای	δ_a, δ_e	زاویه انحراف بالک



شکل ۱. دستگاه مختصات پایداری، بالک‌های کنترلی و بلوک دیاگرام حلقه بسته سامانه کنترل پهپاد

در این پهپاد از دو بالک کنترلی در دم متعارف به صورت افقی، برای اعمال فرامین کنترلی استفاده می‌شود. همچنین دم متعارف به دو بالک عمودی ثابت برای افزایش پایداری مجهز است. با کنترل زاویه غلت، فرامین هدایت در سمت و چرخش پهپاد کنترل شده و با کنترل زاویه فراز، پهپاد در صفحه برد هدایت می‌شود. به دلیل مشترک بودن بالک‌ها، تداخل در شرایط بروز خطا و یا اغتشاش نمایان می‌شود. به عنوان مثال با تفاوت در اندازه نیروی برا در هر بالک و یا لقی در هر عملگر، اثر این تداخل مشهود می‌شود. در این شرایط با اعمال فرمان فراز، غلت ناخواسته نیز القاء خواهد شد.

۲-۱- استخراج تابع تبدیل

مدل فضای حالت زاویه فراز در رابطه (۲) و زاویه غلت در رابطه (۳) آورده شده است [۱۷]. این روابط با استفاده از روش اغتشاش کوچک و صرف نظر از بخش‌های ناچیز حاصل شده‌اند [۱۸].

$$\dot{\alpha} = \frac{1}{mU_0}(QS(C_{z\alpha} - C_x) - F_x)\alpha + \left(\frac{QS.C}{mU_0^2}C_{zq} + 1\right)q - \frac{C_{l\delta}}{mU_0}\delta_e \quad (۲)$$

$$\dot{q} = \frac{QS.c}{I_{yy}}C_{z\alpha}.x_{PG}\alpha + \frac{QS.C^2}{I_{yy}U_0}C_{mq}q - \frac{C_{l\delta}.D_x}{I_{yy}}\delta_e$$

$$\dot{\theta} = -q$$

$$\dot{p} = \frac{QS.C^2}{I_{xx}U_0}C_{lp}p + \frac{C_{l\delta}.D_y}{I_{xx}}\delta_a \quad (۳)$$

$$\dot{\phi} = -p$$

با انتقال معادله (۲) به فضای حالت، تابع تبدیل از انحراف بالک به زاویه فراز بدست می‌آید.

$$\begin{bmatrix}\dot{\alpha}\\ \dot{q}\end{bmatrix} = \begin{bmatrix}a_{11} & a_{12}\\ a_{21} & a_{22}\end{bmatrix}.\begin{bmatrix}\alpha\\ q\end{bmatrix} + \begin{bmatrix}b_1\\ b_2\end{bmatrix}.\delta_e$$

$$q = [0 \quad 1].\begin{bmatrix}\alpha\\ q\end{bmatrix}$$

$$\frac{q(s)}{\delta_e(s)} = C(sI - A)^{-1}B \quad (۴)$$

$$\dot{\theta} = -q$$

$$\frac{\theta(s)}{\delta_e(s)} = -\frac{1}{s}\frac{q(s)}{\delta_e(s)}$$

$$G_\theta(s)\left(\frac{\theta(s)}{\delta_e(s)}\right) = -\frac{A_1 s + A_2}{s(s^2 + B_1 s + B_2)}$$

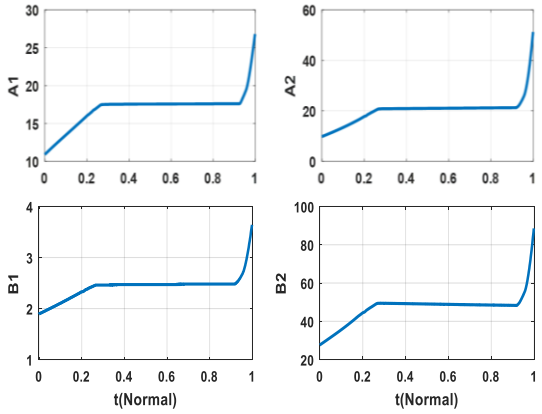
همچنین به صورت مشابه در رابطه (۳) داریم:

$$G_\phi(s)\left(\frac{\phi(s)}{\delta_a(s)}\right) = \frac{A}{s(s+B)}$$

$$A = \frac{C_{l\delta}.D_y}{I_{xx}}\delta_a \quad (۵)$$

$$B = -\frac{Q.S.C^2}{I_{xx}U_0}C_{lp}$$

در تصویر (۲) متغیرهای تابع تبدیل زاویه فراز، در شرایط نامی ملاحظه می‌شوند. در این تصویر با زمان نهایی $t_e = 480s$ زمان بی‌بعد شده و با تغییر مشخصات تابع تبدیل تغییر می‌کند.



شکل ۲. تغییر ضرایب تابع تبدیل زاویه فراز

جدول ۲. نقاط حساس تابع تبدیل زاویه فراز

تابع تبدیل زاویه فراز	زمان	توضیحات
$\frac{10.(1+1.1s)}{s(s^2+1.95s+28.)}$	$0.02t_e$	شروع ماموریت
$\frac{20.6(1.+0.9s)}{s(s^2+2.3s+47.)}$	$0.15t_e$	ابتدای فاز کروز
$\frac{21.(1+0.84s)}{s(s^2+2.4s+47.)}$	$0.5t_e$	میانه پرواز کروز
$\frac{21.8(1+0.84s)}{s(s^2+2.5s+47.)}$	$0.85t_e$	انتهای فاز کروز
$\frac{50.(1+0.54s)}{s(s^2+3.7s+85.)}$	$0.98t_e$	پایان ماموریت

برخلاف زاویه فراز، تابع تبدیل زاویه غلت از تغییر مشخصات وزنی تقریبا مستقل بوده و از تغییر سرعت و ارتفاع تاثیر می‌گیرد. تابع تبدیل زاویه غلت در میانه زمان پرواز کروز عبارت است از:

$$G_\phi(s)\left(\frac{\phi(s)}{\delta_a(s)}\right)=\frac{15.7}{s(s+7.8)} \quad (۶)$$

۲-۲- طراحی کنترل‌گر کلاسیک زاویه فراز

رایج ترین کنترل‌گر کلاسیک، جبران‌ساز تناسبی، انتگرالی و مشتقی (PID) می‌باشد. در فرم استاندارد، این کنترل‌گر با استفاده از خطا، مشتق خطا و انتگرال آن، ردیابی فرمان را انجام می‌دهد. به دلیل استفاده از سامانه ناوبری اینرسی و وجود عملگرهای الکترومکانیکی، نویز اندازه‌گیری و ارتعاشات وارد حلقه کنترل شده و سبب بروز اختلال و پرش‌های غیر واقعی در محاسبه زاویه وضعیت می‌شود. برای بهبود عملکرد، فرم توسعه یافته جبران‌ساز ارائه می‌گردد.

$$G_c(s)=(k_p+k_d s+\frac{k_i}{s})$$
$$G_c(s)=k_{smo}(\frac{k_p(S/Z+1)}{\frac{s^2}{\omega_n^2}+\frac{2\zeta s}{\omega_n}+1.}+\frac{k_i}{s+z_i}) \quad (۷)$$

رابطه اول که فرم استاندارد کنترل‌گر است، با ورود نویز، خروجی نیز آلوده شده و لذا برای بهبود عملکرد جبران‌ساز نوع دوم پیشنهاد می‌شود. با افزودن یک فیلتر درجه دوم و همچنین تغییر در ساختار انتگرال‌گیر، کنترل‌گر اصلاح می‌شود. دلیل استفاده از این ساختار، مواجه با ارتعاشات و نویز اندازه‌گیری است. همچنین این کنترل‌گر شامل بهره هموارساز (k_{smo}) می‌باشد. با توجه به رفتار نویز ورودی فیلترهای پیشنهادی عبارتند از:

$$F_1(s)=\frac{1}{s+0.01} \quad (۸)$$

$$F_2(s)=\frac{1}{\frac{s^2}{60^2}+\frac{2\times 0.6s}{60}+1}$$

همچنین الزامات طراحی عبارتند از:

۱) حاشیه فاز بیش از 45 درجه باشد.
۲) حاشیه بهره بیش از 4(12dB) برابر باشد.
۳) حداکثر فراجهش کمتر از 30% باشد.
۴) در کانال غلت زمان صعود حدود 2s و در کانال فراز در محدوده 3s باشند.
۵) زمان نشست حدود 10s باشد.

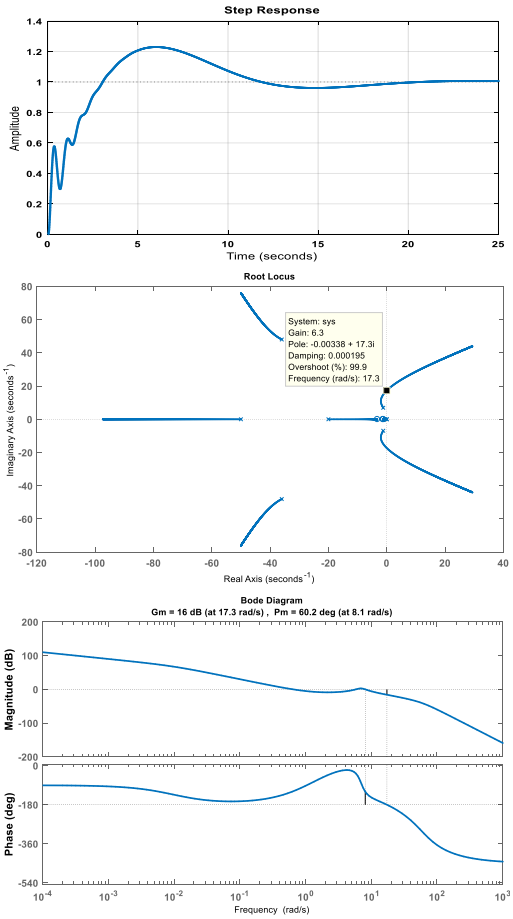
بر اساس الزامات و استفاده از داده‌های جدول(۲)، با روش طراحی کلاسیک جبران‌ساز طراحی می‌گردد. باید مکان هندسی ریشه‌ها به ازای بهره حداقل 4 از ناحیه پایداری خارج نشود. با احتساب حلقه کنترلی شکل (۱)، کنترل‌گر عبارت است از:

$$G_{c\delta_e}(s)=\frac{k_{smo}0.9(S/4.5+1)}{\frac{s^2}{60^2}+\frac{2\times 0.6s}{60}+1}+\frac{k_{smo}0.25}{s+0.01} \quad (۹)$$

در تصویر (۲)، علاوه بر اثر تغییرات جرمی، تغییرات سرعت و ارتفاع بر بهره حلقه باز مشخص است. برای جبران، بهره هموارساز، k_{smo} در فرمان کنترل‌گر ضرب می‌گردد. اثر این بهره، مستقل نمودن بهره کانال از تغییر ارتفاع و سرعت است.

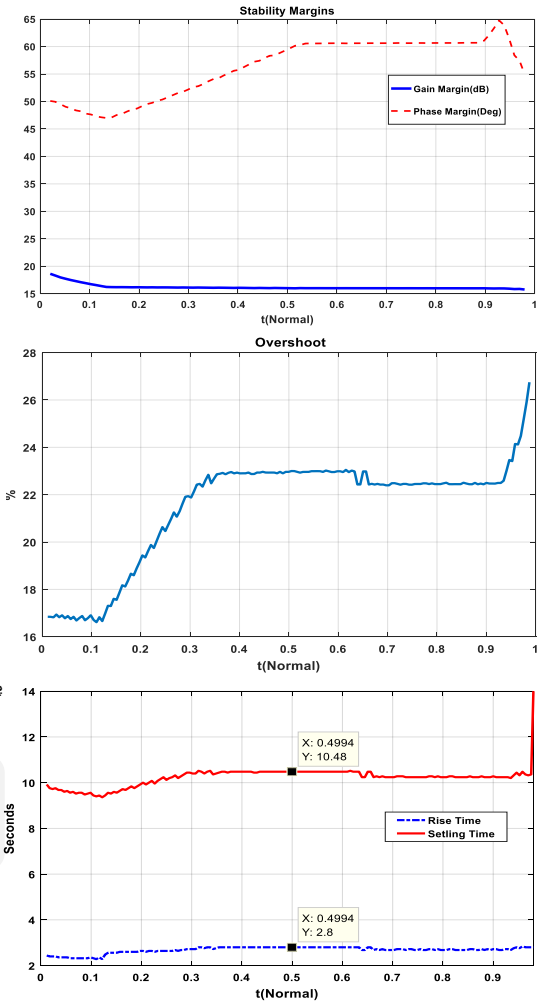
$$Q=\frac{1}{2}\rho V^2$$
$$k_{smo}=\frac{Q_0}{Q} \quad (۱۰)$$

Q_0 فشـــار دینـــامیکی لحظـــه $t = 0.5t_e$ می‌باشد که کنترل‌گر به نمایندگی در آن نقطـه طراحی شده است. افزایش سرعت، مقـدار بهـره هموارساز را کاهش و افزایش ارتفاع آنرا افـزایش می‌دهد. پاسخ پلـه، مکـان هندسـی ریشـه‌هـا و نمودار بود در نقطه میانی پرواز کـروز در شـکل (۳) آورده شده است.



شکل ۳. پاسخ زمانی و فرکانسی کانال زاویه فراز

برای تحلیل کنترل‌گر طراحی شده علاوه بـر نقطه نامی، کل نقاط پرواز نیز بررسی می‌شـوند. بـرای ایـن منظـور لازم اسـت مشخصـات مهـم پایداری و همچنین مشخصات پاسخ پله در کـل زمان رسم شوند. مولفـه‌هـای اساسـی پایـداری برای کل زمان پرواز در تصویر (۴) ارائه شده‌انـد. نمودار اول حاشیه فـاز و بهـره را در کـل زمـان بی‌بعد شده نشان می‌دهد. مشخص است حاشیه فاز بیش از 45 درجه بـوده و حاشـیه بهـره نیـز بـیش از 15dB مـی‌باشـد. نمـودار دوم مقـدار فراجهش را نشان می‌دهد که در محدوده کمتـر از 30% است. در نمودار سوم نیز زمان صـعود و نشست رسم شده‌اند کـه تاییـد کننـده انطبـاق طراحی با الزامات می‌باشد.

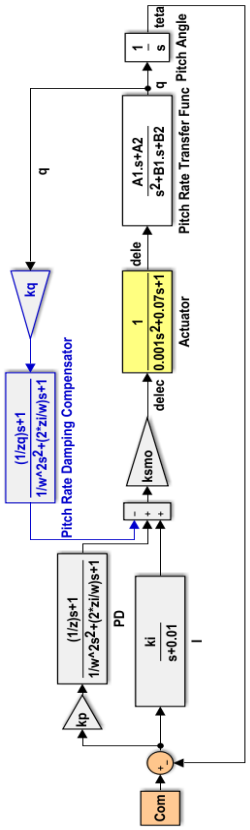


شکل ٤. مشخصات پایداری با کنترل‌گر کلاسیک طراحی شده برای کل زمان در حالت زمان بی‌بعد شده

با توجه به مقدار حاشیه‌های پایداری، مقـاوم بودن کنترل‌گر در کل زمان پرواز مشهود اسـت. این نمودارها نشان می‌دهند که با وجود طراحی خلبان خودکار برای حالت بدون عدم قطعیت، با کمک بهره هموارساز، کیفیت جبـران‌سـاز بـرای کل زمان مناسب می‌باشد. در کانال فراز با توجه

به کوچک بودن ذاتی C_{mq}، میرایی کانال کوچک می‌باشد. این کاستی در پاسخ پله، تصویر (۳) مشهود است. برای بهبود، ناگزیر از میراکننده سرعت زاویه‌ای استفاده می‌شود. در شکل (۵)، میرا کننده به حلقه کنترلی افزوده شده‌است. همچنین رابطه (۱۱) میرا کننده طراحی شده را نشان می‌دهد.

$$G_{cq}(s)=\frac{k_{smo}0.14(^{S}/_{14}+1)}{\frac{s^2}{60^2}+\frac{2\times 0.6s}{60}+1} \quad (۱۱)$$



شکل ۵. کانال فراز با کنترل‌گر زاویه و میرا کننده سرعت زاویه‌ای

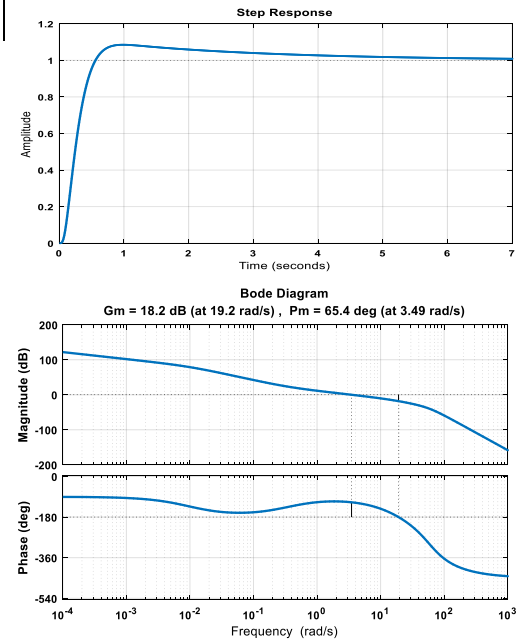
۳-۲- طراحی کنترل‌گر کلاسیک زاویه غلت

مشابه طراحی کانال فراز، کنترل‌گر برای زاویه غلت طراحی می‌گردد. در رابطه (۱۲) کنترل‌گر زاویه و سرعت زاویه‌ای غلت آورده شده است.

$$G_{c\delta_a}(s)=\frac{k_{smo}2.1(^{S}/_{8.6}+1)}{\frac{s^2}{60^2}+\frac{2\times 0.6s}{60}+1}+\frac{k_{smo}0.48}{s+0.01} \quad (۱۲)$$

$$G_{cp}(s)=\frac{k_{smo}0.01(^{S}/_{14.}+1)}{\frac{s^2}{60^2}+\frac{2\times 0.6s}{60}+1}$$

پاسخ به پله واحد و حاشیه‌های بهره و فاز کانال غلت، در تصویر (۶) رسم شده‌اند. تمامی الزامات و قیود برآورده شده است.



شکل ۶. پاسخ فرکانسی و پاسخ به پله واحد کانال غلت

۳. یادگیری ماشین در سامانه کنترل

یادگیری ماشین یکی از شاخه‌های هوش مصنوعی می‌باشد. این علم در ماشین‌ها به سرعت در حال توسعه و کاربرد بوده و به سه یادگیری نظارت شده، نظارت نشده و تقویتی تقسیم می‌شود. روش سوم مورد نظر این تحقیق می‌باشد. در این روش عامل با محیط تعامل کرده و آموزش می‌گیرد. این یادگیری برای سامانه‌های مارکوفی قابل پیاده‌سازی است. در مرجع [۱۹] فرآیند مارکوف به عنوان یک فرآیند تصادفی تعریف می‌شود. این مدل، از نظر

ریاضی اینگونه تعریف می‌شود که تنها عملکرد در زمان حال بر وضعیت آینده سیستم تأثیر می‌گذارد. در مسائل واقعی شامل عدم قطعیت، فرآیند تصمیم‌گیری مارکوف برای مدل‌سازی پیشنهاد می‌شود. در مرجع [۲۰] دینامیک محیط به صورت مدل پنج گانه تعریف شده است:

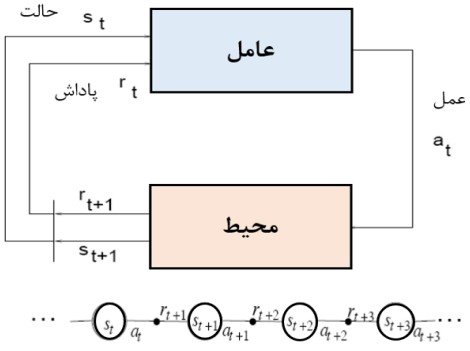
- S، مجموعه‌ای از همه حالت‌های محتمل
- A، مجموعه‌ای از همه عمل‌های محتمل
- $P, S\times A\times A \rightarrow [0, 1]$ احتمال شرطی، حالت و عمل نسبت به هم
- $R, S\times A\times S \rightarrow R$ پاداش لحظه‌ای مورد انتظار از انجام عمل a_t
- $\gamma \in [0, 1]$، ضریب پاداش آینده

۱-۳- یادگیری تقویتی

یادگیری تقویتی، مطالعه چگونگی تعامل یک سامانه تحت کنترل (عامل) با محیط خود برای یادگیری سیاستی است که پاداش تجمعی برای یک کار را به حداکثر می‌رساند. امروزه، این روش رشد چشمگیری داشته و با کسب نتایج خوب در حوزه‌های مختلف مانند، کنترل سامانه‌های رباتیک [۲۱]، مسیریاب‌ها، خودروهای خویش‌ران و بازی‌ها [۲۲و۲۳]، مورد توجه است.

عامل در تعامل با محیط و پاداش دریافتی، آموزش می‌گیرد. پیش از یادگیری، عامل عمل بهینه و میزان پاداش را نمی‌داند. این وظیفه برای عامل تعریف می‌شود تا در طول زمان و تعامل با محیط، تجربه کافی در مورد حالت‌ها، عمل‌ها و پاداش جمع‌آوری نماید. در نهایت عامل بر اساس تجربیات کسب شده، عملکرد بهینه را یاد می‌گیرد. از پاداش‌های دریافتی در تشخیص سیاست بهینه استفاده می‌شود. سیاست بهینه، سیاستی است که پاداش مورد انتظار را بیشینه نماید [۲۴]. روند کاری در یادگیری تقویتی عبارت است از:

۱- عامل حالت فعلی خود را می‌شناسد (s_t).
۲- عامل عمل a_t را در حالت s_t انجام می‌دهد.
۳- حالت به s_{t+1} تغییر کرده و پاداش r_{t+1} دریافت می‌شود.
۴- عامل از پردازش پاداش دریافتی از عمل انجام شده در حالت مشخص آموزش می‌گیرد.
۵- عامل در حالت جدید، عمل بعدی را با آموزش در حال تکمیل برای دریافت پاداش بیشتر انجام می‌دهد. بازگشت به مرحله ۳



شکل ۷. نمایی از تعامل عامل با محیط

سیاست (فرمان کنترلی)، نگاشتی از وضعیت به عمل برای دریافت پاداش حداکثر از محیط است.

$$\pi_t(s, a) = P\{a_t = a \mid s_t = s\} \qquad (۱۳)$$

عامل تحت سیاست π می‌کوشد تا امید ریاضی پاداش دریافتی از محیط حداکثر گردد. در بسیاری از مسائل، تعامل با محیط، به صورت

دوره‌ای برای یک بازه مشخص انجام می‌شود. مثلا در شبیه‌سازی پهپاد، از لحظه شروع ماموریت تا لحظه پایان آن یک دوره نام دارد. مرسوم است، برای محاسبه پاداش تجمعی (G_t) به پاداش‌های نزدیکتر ارزش بیشتری داده می‌شود [۲۴].

$$G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \quad (۱۴)$$

در تعریف یادگیری تقویتی مشخصه‌ای با نام تابع ارزش تعریف می‌گردد. این تابع همبستگی مستقیم با پاداش دریافتی در هر حالت دارد. به زبان ساده، این تابع امید ریاضی حداکثر پاداشی که در یک حالت می‌توان دریافت کرد را مشخص می‌کند. فرض می‌شود، عامل مورد نظر برای کنترل فرایند از سیاستی با نام π استفاده می‌نماید. همچنین از محیط، پاداش‌هایی با توزیع تصادفی دریافت می‌کند. مقدار تابع ارزش در حالت s_t تحت سیاست π به صورت رابطه (۱۵) تعریف می‌شود.

$$V(s_t) = E\{G_t|s_t,\pi\} = E\{\sum_{k=0}^{\infty} \gamma^k r_{t+k}|s_t,\pi\} \quad (۱۵)$$

رابطه بهینگی بلمن برای تابع ارزش حالت و سیاست بهینه عبارت است از [۲۴]:

$$V(s) = \max_a \sum_{s,r} p(s_{t+1},r|s,a)[r + \gamma V(s_{t+1})]$$
$$\pi(s) = \operatorname{argmax}_a \sum_{s,r} p(s_{t+1},r|s,a)[r + \gamma V(s_{t+1})] \quad (۱۶)$$

در این رابطه، برای محاسبه تابع ارزش نیاز به مشخص بودن توزیع احتمال $p(s_{t+1},r|s,a)$ می‌باشد. این متغیر نمایشی از دینامیک محیط تصمیم ماکوف است. پس شرط حل مسئله، وجود دینامیک محیط می‌باشد. برای هموار نمودن این محدودیت، استفاده از تفاضل زمانی رایج شده‌است. در این روش از تفاوت لحظه‌ای میان دو سیگنال برای بهبود سیاست استفاده می‌شود. با انجام یک عمل در حالت معلوم، حالت تغییر یافته (s_{t+1}) و از پاداش دریافتی (r_{t+1})، تابع ارزش با ضریب یادگیری α بروز می‌شود. برای حل مسئله، مقدار تابع ارزش آینده ($V(s_{t+1})$) به صورت تقریبی از طریق شبکه عصبی عمیق پیش‌بینی شود [۲۴].

$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \quad (۱۷)$$

۲-۳- یادگیری تقویتی به روش مونت-کارلو

یکی از زیر شاخه‌های یادگیری تقویتی، یادگیری به روش مونت کارلو است. در این روش بر خلاف روش برنامه نویسی پویا که به مدل محیط نیاز دارد، عامل با تعامل با محیط و پاداش دریافتی آموزش می‌بینند. عامل نمونه‌هایی برای تجربه کردن در محیط تولید کرده و بر اساس پاداش تجمعی، تابع ارزش محاسبه می‌شود[۲۵]. چند ویژگی کلیدی این روش عبارتند از:

- به مدل محیط ($p(s_{t+1},r|s_t,a)$) نیاز نیست.
- عامل با انجام عمل و میزان پاداش تجمعی، تابع ارزش $V_\pi(s_t)$ را تحت سیاست π اکتشاف می‌نماید. یعنی یادگیری تکمیل می‌شود.
- در پایان هر دوره محاسبات تابع ارزش به روز می‌شوند.

- در این روش، از یاری‌گیری ضعیف رابطه (۱۷) که تخمین از دینامیک است استفاده نمی‌شود. مقدار واقعی پایان دوره به کار می‌آید.
- کارا برای مسائل دارای دوره کاری مشخص

در یادگیری مونت کارلو در شرایط عدم قطعیت، یک دوره کامل توسط عامل طی می‌شود. سپس با استفاده از پاداش تجمعی که در انتهای دوره حاصل می‌شود تابع ارزش بروز رسانی می‌شود. اما در یادگیری تفاضل زمانی، بروزرسانی در هر لحظه و یا بازه کوچکی از یک دوره صورت می‌گیرد.

$$V_\pi(s) = E[G_t|s_t = s, a_t = a, \pi] \quad (۱۸)$$

برای محاسبه تابع ارزش، روش اولین دیدار حالت در هر دوره انتخاب می‌شود. مقدار متوسط پاداش در اولین باری که در هر دوره کاری، حالت s اتفاق می‌افتد، برای بروزرسانی وزن‌های شبکه عصبی تخمین ارسال می‌شود.

جدول ۳. روند پیاده‌سازی الگوریتم مونت کارلو

الگوریتم مونت کارلو روش اولین دیدار
The first timestep t that state s is visited in $\boldsymbol{n \leftarrow n + 1}$ Loop for each step of episode, t : T-1, … , 0 $\boldsymbol{G_t \leftarrow \gamma G_t + r_{t+1}, for\ state = s}$ $\boldsymbol{V(s) \leftarrow V(s) + \alpha(G_t - V(s)), \alpha < 1}$ $\boldsymbol{If\ n \rightarrow \infty\ then\ V(s) \rightarrow V_\pi(s)}$

در این روند n نماد شماره دوره تصادفی و T تعداد نمونه ها در یک دوره است. محاسبه و بروزرسانی مقدار تابع ارزش، بر اساس امید ریاضی پاداش در انتهای هر دوره است. مزیت اصلی این روش، کاربرد در سامانه‌های با مدل مبهم است.

۴. بهبود عملکرد با یادگیری تقویتی

کنترل‌گر کلاسیک برای کانال‌های فراز و غلت طراحی گردید. در طراحی کنترل‌گر پس از خطی‌سازی و ورود خطای مدل‌سازی به محاسبات، برای یک نقطه کاری مشخص و در شرایط نامی طراحی صورت گرفت. با تغییر شرایط (عدم قطعیت و مانور) نقطه کاری تغییر کرده و لزوما رفتار سامانه کنترل همانند نقطه طراحی نمی‌باشد. با دلایل ذکر شده استفاده از یادگیری تقویتی برای اصلاح و کنترل‌گر در برابر موارد مطرح شده پیشنهاد می‌گردد. هدف بر آن است که با مبنی قرار دادن کنترل‌گر کلاسیک و ایجاد شرایط مناسب یادگیری، متغیرهای اصلی کنترل‌گر تنظیم شوند.

۴-۱- شبیه‌سازی غیر خطی تصادفی

برای اجرای یادگیری، لازم است که تمامی وضعیت‌های ممکن از مانور و یا عدم قطعیت به صورت تصادفی در هر دوره یادگیری انتخاب شوند. گفتنی است در این روند چون هدف مقاوم سازی حداکثری برای کنترل‌گر است، لذا بازه عدم قطعیت‌ها محافظه کارانه، بزرگ انتخاب می‌شوند. در جدول (۴) بازه مورد نظر (3σ) آورده شده است. متغیرهایی که با درصد مشخص شده‌اند میزان انحراف از مقدار نامی بوده و در متغیرهای با مقدار نامی صفر همانند لقی بازه خطا مشخص شده است. محاسبات فرایند یادگیری پیچیده و زمان‌بر است. برای کوتاه کردن زمان به شرط عدم تغییر کیفیت، پهپاد برای کوچکترین زمان ماموریت تعریف شده (480s) آموزش داده می‌شود.

جدول ٤. بازه عدم قطعیت‌ها (انحراف از مقدار نامی)

متغیر	بازه عدم قطعیت
پیشران	%10
جرم	%10

لختی‌ها	%30
بازه لختی ضربی	$[0\ \ 5]kgm^2$
بازوهای کنترلی فراز و غلت	%10
تغییر مرکز جرم و فشار از مقدار نامی	%2
بازه انحراف مرکز جرم از محورها	[-10 10]mm
ضریب برا و پسا آیرودینامیک	%20
مشتقات آیرودینامیک	%100
ضریب برا و پسای بالک‌های کنترلی	%20
جهت باد	تصادفی در 360^o
بازه سرعت باد تصادفی	[0 25]m/s
بازه ارتفاع شروع ماموریت	[0 2000]m
بازه انحراف ارتفاع پرواز کروز	$[-500\ \ 500]m$
بازه خطای سرعت شروع ماموریت	[-10 10]m/s
بازه خطای نصب بالک‌ها	$[-1\ \ 1]^o$
بازه لقی عملگرها	$[0\ \ 0.3]^o$
بازه گشتاور عدم نصب متقارن بال‌ها	$[-50\ \ 50]N.m$

در برنامه شبیه سازی و در هر دروه اجرا، به صورت تصادفی مقادیر سیستمی، ماموریتی و یا اغتشاش از جدول انتخاب شده و اجرای شبیه‌سازی انجام می‌گردد. بنابراین، تمامی وضعیت‌های ممکن به صورت مصنوعی ایجاد شده و الگوریتم یادگیری با تعداد تجربیات زیاد مواجه می‌شود.

۲-۴- مدل تعمیم یافته کنترل‌گر

با افزودن بهره‌های اصلاحی (g_i) به کنترل‌گر کلاسیک، ساختار یادگیر کنترل‌گر معرفی می‌شود. با تنظیم این بهره‌ها از روش یادگیری تقویتی عملکرد کنترل‌گر کلاسیک بهبود می‌یابد. در رابطه (۱۹) ساختار کنترل‌گر یادگیرنده آورده شده است.

$$k = 0.99$$
$$MA_{e_\varphi} = MovingAverage|\varphi_c - \varphi|$$
$$MA_{e_\varphi} \leftarrow kMA_{e_\varphi} + (1-k)|\varphi_c - \varphi|$$

$$G_{c\delta_a} = (1 + g_1 + g_2 MA_{e_\varphi}) \frac{k_{smo} 2.1\left({}^{s}/_{8.6} + 1\right)}{\frac{s^2}{60^2} + \frac{2 \times 0.6s}{60} + 1} + (1 + g_3) \frac{k_{smo} 0.48}{s + 0.01}$$

$$G_{cp} = (1 + g_4) \frac{k_{smo} 0.01\left({}^{s}/_{14.} + 1\right)}{\frac{s^2}{60^2} + \frac{2 \times 0.6s}{60} + 1} \qquad (۱۹)$$

$$MA_{e_\theta} = MovingAverage|\theta_c - \theta|$$
$$MA_{e_\theta} \leftarrow kMA_{e_\theta} + (1-k)|\theta_c - \theta|$$

$$G_{c\delta_e} = (1 + g_5 + g_6 MA_{e_\theta}) \frac{k_{smo} 0.9\left({}^{s}/_{4.5} + 1\right)}{\frac{s^2}{60^2} + \frac{2 \times 0.6s}{60} + 1} + (1 + g_7) \frac{k_{smo} 0.25}{s + 0.01}$$

$$G_{cq} = (1 + g_8) \frac{k_{smo} 0.14\left({}^{s}/_{14.} + 1\right)}{\frac{s^2}{60^2} + \frac{2 \times 0.6s}{60} + 1}$$

به ازای تمامی مانورهای ممکن و عدم قطعیت‌های موجود، شبیه‌سازی به صورت تصادفی اجرا شده و در هر اجرا به روش مونت کارلو، تابع ارزش و سیاست بروزرسانی می‌شوند. همانگونه که مشخص است متغیرهایی که آموزش می‌بینند، از جنس بهره می‌باشند. در بهره جبران‌ساز زاویه، برای افزایش کیفیت، علاوه بر بهره مستقیم (g_1, g_5) از بهره‌های وابسته به میزان میانگین متحرک خطا (g_2, g_6) نیز استفاده می‌شود. دو روش کلی در بکارگیری بهره‌ها (g_iها) وجود دارد.

۱) بهره‌های اصلاحی به صورت برخط و اختصاصی برای هر حالت انتخاب شوند، تا عملکرد مناسب نقطه‌ای حاصل شود. (بهره متغیر مختص هر حالت)

۲) بهره‌های اصلاحی به صورت خارج از خط و ثابت برای تمامی حالت‌ها با هدف بهبود عملکرد متوسط انتخاب می‌شوند. (بهره ثابت برای تمامی حالت‌ها)

در روش اول، هر لحظه با توجه به وضعیت پهپاد، سامانه می‌آموزد که با چه ضریب اصلاحی در کنترل‌گر، بهترین عملکرد حاصل می‌شود. در این روش حجم محاسبات در حین کار بسیار زیاد بوده و نیاز به پردازنده قوی

مـی‌باشـد. امـا در روش دوم بهتـرین ضـریب اصلاحی کنترل‌گر برای تمامی حالت‌ها انتخـاب می‌گردد. در این روش حجم محاسبه در فراینـد کاری برخط، کم و محاسبات به صورت خـارج از خط انجام می‌گیرد. روش انتخاب شـده در ایـن تحقیق مطابق روش دوم است.

۳-۴- حالت‌ها و ساختار تخصیص پاداش

در پیاده‌سازی، لازم است تعریـف دقیقـی از حالت‌ها به عنوان ورودی شـبکه‌هـای عصـبی و همچنین ساختار تخصیص پاداش بـرای تعامـل محـیط و عامـل صـورت گیـرد [۲۶]. در ایـن تحقیق چهار حالت در نظر گرفته شده است:

۱. فرمان زاویه در فراز (θ_c) و غلت (φ_c)
۲. میانگین متحرک خطای ردیابی فراز، MA_{e_θ}
۳. میانگین متحرک خطای ردیابی غلت، MA_{e_φ}

تخصیص پاداش در هر گام عبارت است از:

$$e_1 = |\theta_c - \theta|$$
$$e_2 = |\varphi_c - \varphi|$$
$$e_3 = |q|$$
$$e_4 = |p|$$

(۲۰)

$$r_t = 1. - 0.1e_1^{3/2} - 0.1e_2^{3/2} - 0.2e_3 - 0.2e_4 - \sum_i \beta_i |g_i|$$
$$if\ (e_1\ or\ e_2) > 10^o,\ \ r_t \leftarrow r_t - 100$$

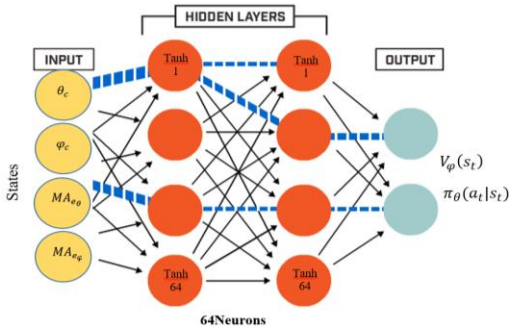
خطـای ردیـابی فـرامین و انـدازه سـرعت زاویه‌ای، اثر اصلی را بر پاداش لحظه‌ای داشته و جریمه سنگینی برای ناپایـداری در نظـر گرفتـه می‌شود. اثـر سـرعت زاویـه‌ای در پـاداش بـرای ممانعت از بـروز حرکـات نوسـانی بـوده کـه بـه صورت غیـر مسـتقیم، حاشـیه‌هـای پایـداری را محفوظ نگاه می‌دارد. همچنین بـرای جلـوگیری از واگرایی در بهره‌ها (g_i)، از جریمه‌ای متناسب با اندازه آنها استفاده می‌شود.

۴-۴- شبکه عصبی عمیق

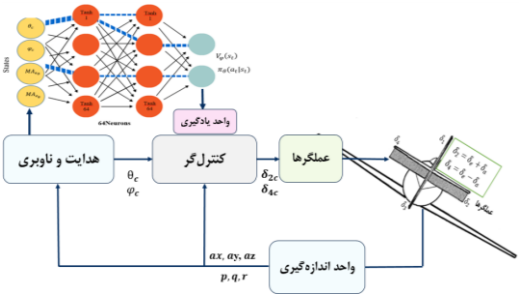
در فراینـد یـادگیری لازم اسـت تـابع ارزش شناسایی شود. این تابع، امید ریاضی پـاداش در هر حالـت را مشـخص مـی‌کنـد. سیاسـت‌هـای اعمـالی (g_i) در حـین آمـوزش بـا کمـک تـابع ارزش، چنان انتخاب می‌شوند کـه بـه بیشـترین مقدار پاداش رسید. برای پیـاده‌سـازی الگـوریتم یادگیری از دو شبکه عصبی عمیق برای تخمین سیاست و تابع ارزش استفاده مـی‌شـود. ورودی این شبکه‌ها، حالت‌های مشـخص شـده بـوده و خروجـی شـبکه منتقـد تخمـین تـابع ارزش ($V(s)$) و شـبکه بـازیگر، سیاسـت بهینـه (g_i) می‌باشد.

در این فراینـد بـا توجـه بـه حالـت پهپـاد و پاداش دریافتی، شبکه‌های عصبی تـابع ارزش و سیاست به تدریج آمـوزش داده مـی‌شـوند. ایـن محاسبات زمان‌بر بوده و به پردازشگر قـوی نیـاز می‌باشد. اما در این تحقیق بـه صـورت خـارج از خط، ضرایب اصلاحی بـه صـورت ثابـت بدسـت آمده و در حالت کاری بدون نیاز به حـل شـبکه عصبی استفاده می‌شوند.

شبکه عصـبی بـازیگر و منتقـد، شـبکه‌هـای مستقل پرسپترون با 64 نـورون بـوده کـه تـابع فعال‌ساز آنها $tanh$ است و دولایه مخفی دارنـد. شبکه اول برای تخمین تابع ارزش و شـبکه دوم برای تعیین تابع سیاست و تولیـد بهـره‌هـای g_i می‌باشد. در تصویر (۸) شبکه هـای مـورد نظـر نمایش داده شده‌اند.



شکل ۸. شبکه عصبی تخمین تابع ارزش و سیاست



شکل ۹. نمایی از فرایند یادگیری

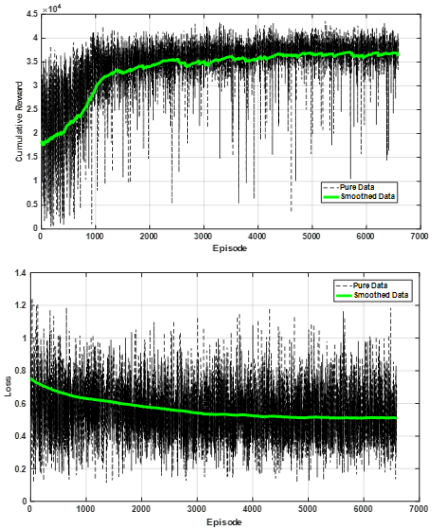
در تصویر (۹) فرایند یادگیری به صورت بلوک دیاگرام نمایش داده شده است. گفتنی است که پس از تکمیل یادگیری و در روند کاری، شبکه‌های عصبی و واحد یادگیری از بلوک دیاگرام حذف می‌شوند. در محاسبه وزن‌های شبکه تابع ارزش (φ) و سیاست (θ) از تابع زیان مربوط به الگوریتم بهینه‌سازی سیاست مجاورتی، رابطه (۲۱) به صورت پس انتشار خطا استفاده می‌شود [۲۷].

$$\hat{A}_t = G_t - V_\varphi(s_t)$$
$$\mathcal{L} = E_t[\left(V_\varphi(s_t) - V(s_t)\right)^2 - \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta old}(a_t|s_t)}\hat{A}_t] \quad (۲۱)$$

$V_\varphi(s_t)$ تابع ارزش تخمینی از شبکه منتقد بوده و $V(s_t)$ مقدار تابع ارزش از محاسبات پاداش (G_t) می‌باشد. $\hat{A}_t$ با نام تابع مزیت شناخته می‌شود. رابطه تابع زیان دو جزء دارد، جزء اول و دوم به ترتیب برای بروزرسانی وزن‌های شبکه منتقد و بازیگر استفاده می‌شود. در بروزرسانی وزن‌ها هدف کاهش جزء اول و افزایش جزء دوم می‌باشد.

۵. اجرای الگوریتم یادگیری

با اجرای تعداد زیاد دوره‌های شبیه‌سازی تصادفی، بخش غالب عدم قطعیت و مانور جدول (۴) پوشش داده می‌شود. تصویر (۱۰) پاداش تجمعی پایان دوره و تابع زیان، در روند یادگیری را به صورت خالص و هموار شده نمایش می‌دهد.

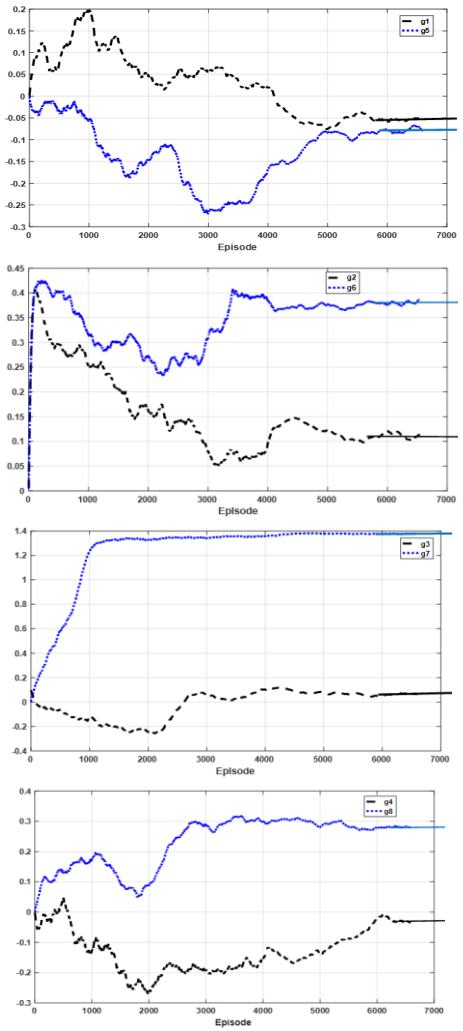


شکل ۱۰. پاداش تجمعی در پایان دوره‌ها و تابع زیان

با تکمیل یادگیری در دوره‌ها (شبیه‌سازی‌ها) پاداش تجمعی از دوره حدود 5000 به بعد به همگرایی نسبی رسیده است. مشاهده می‌شود که ابتدای یادگیری سرعت همگرایی بالا بوده و سپس کاهش می‌یابد. در ضمن به دلیل ماهیت تصادفی محیط، همگرایی در متغیرها به صورت خطی صاف نخواهد بود. پس از همگرایی، پاداش تجمعی به صورت نوسانی و با دامنه کوچک در خروجی‌ها نمود می‌نماید. برای انتخاب مقدار بهینه هر متغیر، باید در یک دوره مشخص همه متغیرها با هم

انتخاب شوند. در تصویر (۱۱) روند کشف بهره‌های تنظیم کنترل‌گر (g_iها) نمایش داده می‌شوند.

این ضرایب با اثر بر روی کنترل‌گر اولیه، ساده‌سازی‌های انجام شده در طراحی، دینامیک‌های مدل نشده و عدم قطعیت‌های محتمل را با مقاوم کردن کنترل‌گر جبران می‌نمایند. برای تعیین مقدار g_iها پس از همگرایی پاداش تجمعی، از میانیابی نمودار استفاده می‌شود.



شکل ۱۱. همگرایی بهره‌های تصحیح کنترل‌گر

در جدول (۵) نتیجه انتخاب متغیرهای تنظیم پس از یادگیری آورده شده‌است.

جدول ۵. مقدار متغیرهای تنظیم کننده کنترل‌گر

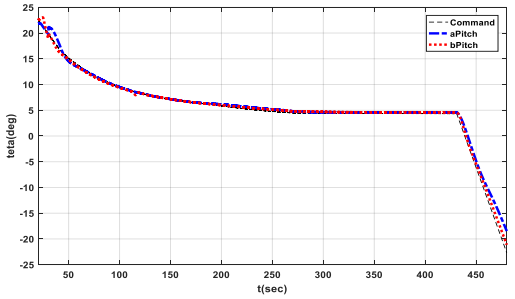
کنترل‌گر فراز	مقدار	کنترل‌گر غلت	مقدار
g_5	-0.08	g_1	-0.05
g_6	0.37	g_2	0.12
g_7	1.35	g_3	0.1
g_8	0.27	g_4	-0.03

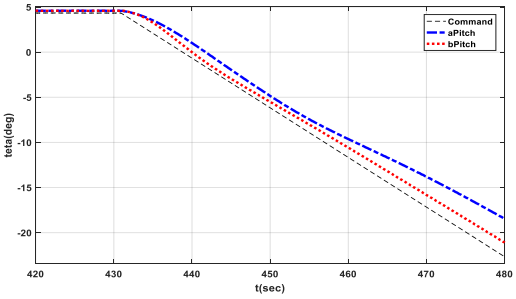
برای نشان دادن تفاوت عملکرد کنترل‌گر کلاسیک اولیه و تنظیم شده، نتایج شبیه‌سازی تصادفی در دو حالت برای تعداد 1000 اجرا در جدول (۶) آورده شده است. تحلیل جدول نشان می‌دهد که پاداش متوسط پس از تنظیم کنترل‌گر، افزایش قابل توجهی داشته است. افزایش پاداش نمایانگر بهبود نسبی کیفیت کنترل‌گر در تمامی اجراها است. همچنین در این جدول مشخص شده است که با کنترل‌گر تنظیم شده تعداد اجراهای ضعیف با پاداش کم (وضعیت ناپایداری و یا شبه ناپایداری)، کاهش یافته است. در شرایط حاد رفتار سامانه بیشتر بهبود یافته است.

جدول ۶. عملکرد دو کنترل‌گر

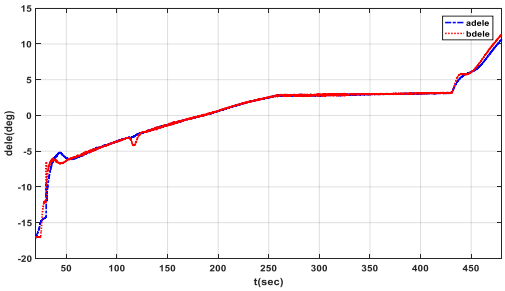
کنترل‌گر	پاداش تجمعی	درصد اجرای ضعیف
کلاسیک اولیه	32100	10.5
کلاسیک اصلاحی	37800	3.4

با یک عدم قطعیت مشخص، شبیه‌سازی شش درجه آزادی با استفاده از دو کنترل‌گر اجرا شده‌است. نتایج اجرای کنترل‌گر اولیه (a) و تنظیم شده (b) برای مقایسه ارائه می‌گردد. در تصویر (۱۲) و (۱۳) زاویه فراز به همراه فرمان به عملگرهای کنترلی ارائه شده است.





شکل ۱۲. زاویه فراز



شکل ۱۳. فرمان به بالک

از تصویر (۱۲) خصوصا در فاز شروع و فاز نهایی مشخص است که ردیابی فرمان فراز بهبود قابل ملاحظه‌ای دارد. به عنوان مثال در فاز نهایی و در ثانیه 480 از پرواز، خطای ردیابی از حدود 4.5 درجه به 1.5 درجه رسیده است که از نظر کیفیت ماموریت، بهبود مهمی اتفاق افتاده است.

۶. نتیجه گیری

در این تحقیق یک پهپاد با مجموعه عملگر مشترک برای پیاده‌سازی الگوریتم تقویتی انتخاب گردید. طبیعتا به دلیل مشترک بودن عملگرها، در شرایط غیر نامی، تداخل کنترلی ایجاد می‌گردد. با تدابیر ساده‌ساز تابع تبدیل خطی تقریبی استخراج شد. با هدف رسیدن به الزامات مشخص شده، کنترل‌گر کلاسیک برای کنترل زوایا و سرعت‌های زاویه‌ای فراز و غلت طراحی گردیدند. مشاهده شد که مشخصات پایداری و زمانی کنترل‌گر طراحی شده الزامات را برآورده می‌نماید. در مانور و عدم قطعیت، دینامیک پهپاد تغییر محسوسی دارد لذا برای اصلاح مدل‌سازی و یا طراحی، با کمک یادگیری تقویتی، تنظیم بهره‌های کنترل‌گر کلاسیک انجام گردید. با جاروب عدم قطعیت‌های مشخص شده و مانورهای ممکن در شبیه‌سازی شش درجه آزادی، ضرایب کنترل‌گر اولیه برای رسیدن به حداقل خطا در ردیابی فرامین و افزودن بر مقاوم بودن آن تنظیم شدند. نتیجه بررسی حدود 20% بهبود در متوسط پاداش دریافتی را نشان می‌دهد. البته بهبود در شرایط ناپایداری و یا شبه ناپایداری به مراتب بیشتر و حدود سه برابر می‌باشد.

با وجود آنکه پهپاد به حلقه هدایت، کنترل و ناوبری مجهز است و خطای ردیابی قابلیت اصلاح تدریجی در مسیر را دارد، کوچک بودن خطای ردیابی خصوصا در فاز نهایی الزامی می‌باشد. نوآوری اصلی این مقاله، متمرکز بر اصلاح اثر مدل‌سازی نادقیق در مجموعه غیر خطی تداخلی و همچنین مقاوم سازی سامانه کنترل در شرایط مانور و عدم قطعیت، مبتنی بر الگوریتم یادگیری تقویتی است. از مزایای این روش، نسبت به تنظیم مبتنی بر تجربه، در بهبود عملکرد با حداقل حجم محاسبات و حفظ پایداری است.

۷. مآخذ

[1] Yoo, Jaehyun, Dohyun Jang, H Jin Kim, and Karl H Johansson, Hybrid Reinforcement Learning Control for a

Micro Quadrotor Flight, IEEE Control Systems Letters 5, no. 2 2020.
[2] Zhang, Tianhao, Gregory Kahn, Sergey Levine, and Pieter Abbeel, Learning Deep Control Policies for Autonomous Aerial Vehicles with Mpc-Guided Policy Search, IEEE international conference on robotics and automation, 2016.
[3] Wang, Yuanda, Jia Sun, Haibo He, and Changyin Sun, Deterministic Policy Gradient with Integral Compensator for Robust Quadrotor Control, IEEE Transactions on Systems, Man, and Cybernetics: Systems 50, no. 10, 2019.
[4] Qingqing, Zheng, Tang Renjie, Gou Siyuan, and Zhang Weizhong, A PID Gain Adjustment Scheme Based on Reinforcement Learning Algorithm for a Quadrotor, 39th Chinese Control Conference, 2020.
[5] Bøhn, Eivind, Erlend M Coates, Signe Moe, and Tor Ame Johansen, Deep Reinforcement Learning Attitude Control of Fixed-Wing Uavs Using Proximal Policy Optimization, The international conference on unmanned aircraft systems (ICUAS), 2019.
[6] Elhaki, Omid, and Khoshnam Shojaei, A Novel Model-Free Robust Saturated Reinforcement Learning-Based Controller for Quadrotors Guaranteeing Prescribed Transient and Steady State Performance, Aerospace Science and Technology 119, 2021.
[7] Chaffre, Thomas, Julien Moras, Adrien Chan-Hon-Tong, Julien Marzat, Karl Sammut, Gilles Le Chenadec, and Benoit Clement, Learning-Based Vs Model-Free Adaptive Control of a Mav under Wind Gust, The Informatics in Control, Automation and Robotics: 17th International Conference Lieusaint-Paris, France, 2022.
[8] Rodriguez-Ramos, Alejandro, Carlos Sampedro, Hriday Bavle, Paloma De La Puente, and Pascual Campoy, A Deep Reinforcement Learning Strategy for Uav Autonomous Landing on a Moving Platform, Journal of Intelligent & Robotic Systems 93, 2019.
[9] Guerra-Langan, Ana, Sergio Araujo Estrada, and Shane Windsor, Reinforcement Learning to Control Lift Coefficient Using Distributed Sensors on a Wind Tunnel Model, AIAA SCITECH 2022 Forum, 2022.
[10] Alizadeh MH, Toloei A., Designing Pitch Angle Compensator for a UAV and Robustification it with Bee Colony Optimization Algorithm, Technology in Aerospace Engineering, 2024.
[11]Esfandiari, Mohamadamin, and MA Amiri Atashgah, Reinforcement Learning Control of an Aerial Robot Based on a Tuned Proximal Policy Optimization in Takeoff and Hover Phases, 10th RSI International Conference on Robotics and Mechatronics (ICRoM), 2022.
[12] Karami, Hamede, and Reza Ghasemi, Adaptive Neural Observer-Based Nonsingular Super-Twisting Terminal Sliding-Mode Controller Design for a Class of Hovercraft Nonlinear Systems, Journal of Marine Science and Application 20, no. 2, 2021.
[13] Tran, Huu Khoa, Hoang Hai Son, Phan Van Duc, Tran Thanh Trang, and Hoang-Nam Nguyen, Improved Genetic Algorithm Tuning Controller Design for Autonomous Hovercraft, Processes 8, no. 1, 2020.
[14] Kong, Xiangyu, Yuanqing Xia, Rui Hu, Min Lin, Zhongqi Sun, and Li Dai, Trajectory Tracking Control for under-Actuated Hovercraft Using Differential Flatness and Reinforcement Learning-Based Active Disturbance Rejection Control, Journal of Systems Science and Complexity, 2022.
[15] Alizadeh, M. H., Toloei, A., Ghasemi, R., Designing a Sliding Mode Control System for a Hovercraft and Improving it with Deep Reinforcement Learning, International Journal of Engineering, 2025.
[16] Wu, Shuai, Motion Control of Unmanned Surface Vehicle Based on Improved Reinforcement Learning Proximal Policy Optimization Algorithm, 2nd International Conference on Information Technology and Intelligent Control, 2022.
[17] McLean, D., Automatic flight control systems (Book), Englewood Cliffs, NJ, Prentice Hall, 1990.
[18] Mohammadloo, S., M. H. Alizadeh and M. Jafari, Multivariable autopilot design for sounding rockets using intelligent eigenstructure assignment technique, International Journal of Control, Automation and Systems, 208-219, 2014.
[19] Sigaud, O. and O. Buffet, Markov decision processes in artificial intelligence, John Wiley & Sons, 2013.
[20] Poole, D. L. and A. K. Mackworth, Artificial Intelligence: foundations of computational agents, Cambridge University Press, 2010.
[21] Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver and D.

Wierstra, Continuous control with deep reinforcement learning, arXiv preprint, 2015.
[22] Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam and M. Lanctot, Mastering the game of Go with deep neural networks and tree search, nature, 2016.
[23] Sadollah, A., A. Bahreininejad, H. Eskandar and M. Hamdi, Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems, Applied Soft Computing **13**(5): 2592-2612, 2023.
[24] Sutton, Richard S, and Andrew G Barto, Reinforcement Learning: An Introduction, MIT press, 2018.
[25] Rubinstein, Reuven Y, and Dirk P Kroese, The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning, Vol. 133: Springer, 2004.
[26] Shuprajhaa, T, Shiva Kanth Sujit, and K Srinivasan, Reinforcement Learning Based Adaptive PID Controller Design for Control of Linear/Nonlinear Unstable Processes, Applied Soft Computing 128, 2022.
[27] Schulman, John, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. , Proximal Policy Optimization Algorithms, arXiv preprint, 2017.