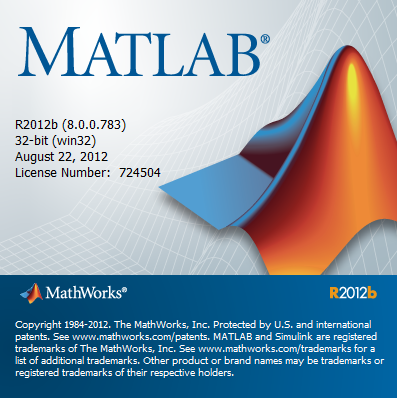
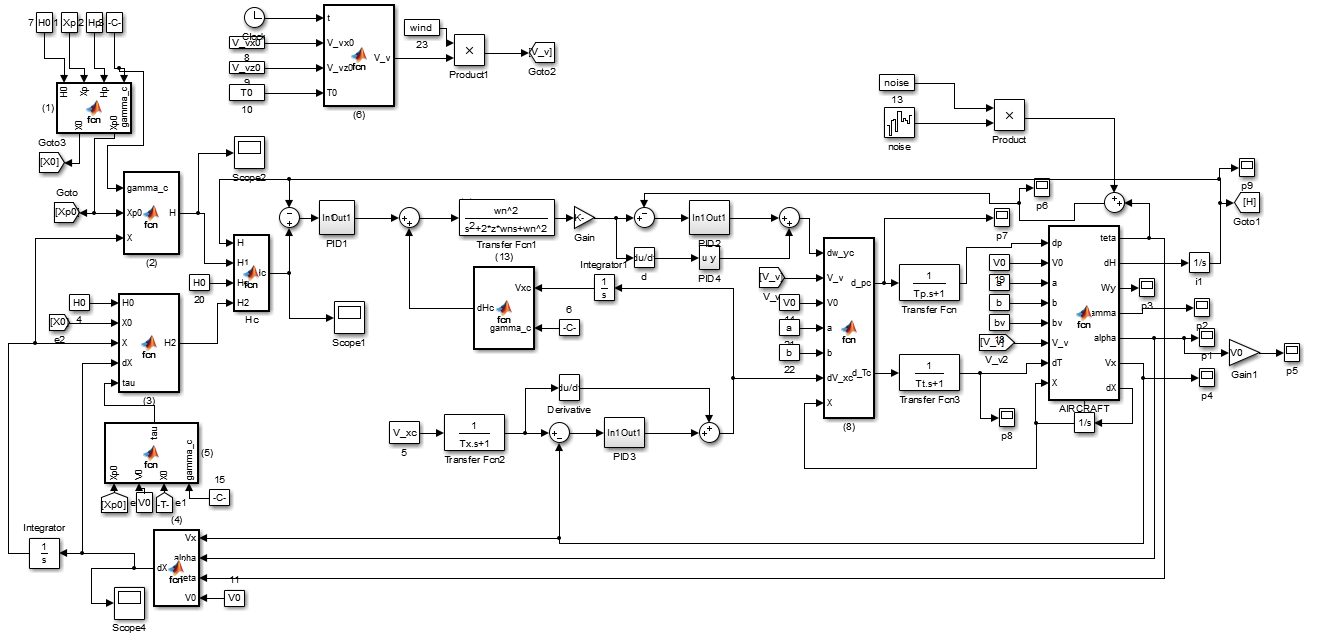
در این مقاله از کنترلر فازی و pid برای کنترل فرود هواپیما استفاده شده است. از آنجایی که سیمولینک متلب 64 بیتی دارای باگ می باشد برای شبیه سازی از متلب 32 بیتی استفاده شده است لذا برای اجرای فایلها از متلب 32 بیتی استفاده کنید. برای این کار نیازی به تغییر در ویندوز نمی باشد فقط کافی است نسخه 32 بیتی متلب را نصب نمایید.



فایل simulation\_PID.slx شبیه سازی شکل 3 مقاله را برای کنترلر pid پیاده سازی نموده است. فایل simulation\_fuzzy.slx نیز همان شبیه سازی شکل 3 می باشد با این تفاوت که بجای کنترلر pid از کنترلرهای فازی استفاده شده است. ساختار این فایلها همانند شکل 3 مقاله به صورت زیر می باشد:



برنامه plot\_res1.m برای رسم شکلهای مختلف همانند زاویه پیچ، مسیر فرود، و ... می باشد و برنامه آن به صورت زیر می باشد:

subplot(331); plot(p1(:,1),p1(:,2)\*180/pi,K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Attack Angle[deg]')

axis([0,30 -4 4])

hold on

if s==1

legend('show');

end

subplot(332); plot(p2(:,1),p2(:,2)\*180/pi,K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Slope Angle[deg]')

axis([0,30 -3.5 0.5])

hold on

if s==1

legend('show');

end

subplot(333); plot(p3(:,1),p3(:,2)\*180/pi,K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Pitch rate[deg/s]')

axis([0,30 -1.5 0.5])

hold on

if s==1

legend('show');

end

subplot(334); plot(p4(:,1),p4(:,2),K,'DisplayName',L)

xlabel('Time[s]')

ylabel('V\_x [m/s]')

axis([0,40 15 80])

hold on

if s==1

legend('show');

end

subplot(335); plot(p5(:,1),p5(:,2),K,'DisplayName',L)

xlabel('Time[s]')

ylabel('V\_z [m/s]')

axis([0,30 -10 5])

hold on

if s==1

legend('show');

end

subplot(336); plot(p6(:,1),p6(:,2)\*180/pi,K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Pitch Angle[deg]')

axis([0,30 -4 2])

hold on

if s==1

legend('show');

end

subplot(337); plot(p7(:,1),p7(:,2),K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Rudder deflection[deg]')

axis([0,30 -20 20])

hold on

if s==1

legend('show');

end

subplot(338); plot(p8(:,1),p8(:,2),K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Trottle command[deg]')

axis([0,30 -18\*10^6 5\*10^6])

hold on

if s==1

legend('show');

end

subplot(339); plot(p9(:,1),p9(:,2),K,'DisplayName',L)

xlabel('Time[s]')

ylabel('Altitude [m]')

axis([0,30 0 110])

if fig==1

axis([0,30 0 21])

end

hold on

if s==1

legend('show');

end

در این برنامه p1 تا p9 9 شکل خواسته شده می باشد و بقیه پارامترها برای رسم بهتر و اضافه کردن legend استفاده شده است.

برای بدست آوردن شکلهای مقاله باید برنامه main اجرا گردد و این برنامه ابتدا مقادیر اولیه مربوط به شبیه سازی را مشخص می نماید سپس فایلهای شبیه سازی simulation\_PID.slx و simulation\_fuzzy.slx را به ازائ شرایط متفاوت: وجود باد و یا وجود نویز اجرا و با برنامه plot\_res1.m در هر قسمت شکلهای بدست آمده از شبیه سازی را رسم می نماید. شکلهای بدست آمده از اجرای این برنامه به صورت زیر می باشند:























نکته: ممکن است در بعضی از اشکال بین نتایج شبیه سازی و نتایج بدست آمده در مقاله اختلاف اندکی مشاهده شود که دلیل آن پارامترهای ناقص ارائه شده در بخش شبیه سازی مقاله بخصوص برای کنترلر می باشد با این حال تحلیل نتایج همانند مقاله می باشد و پاسخهای کنترلر فازی بهتر از کنترلر pid می باشد..