

پیاده سازی سخت افزاری و مقایسه عملکرد کنترل فازی و روش کنترل کلاسیک پیشنهادی در کنترل مسیر ربات

علی اکبر مصدق عنبران

ali_mosaddegh2000@yahoo.com

موسسه آموزش عالی سجاد مشهد

محمد حسن بهاری

bahari@kiaeee.org

دانشگاه فردوسی مشهد

چکیده :

در این مقاله از کنترل فازی جهت کنترل مسیر ربات استفاده شده است. همچنین یک تابع کنترلی بر اساس تابع درون یاب نیوتن پیشنهاد شده و جهت کنترل مسیر ربات مورد استفاده قرار گرفته است. هر دو روش به طور سخت افزاری پیاده سازی شده و نتایج بایکدیگر مقایسه گردیده اند. امتیازات و معایب این روش ها با یکدیگر مقایسه خواهد شد. همچنین روند ساخت ربات مسیر یاب و پیاده سازی هر یک از روش های فوق بیان می شود.
کلمات کلیدی: کنترل فازی، درون یاب نیوتن، ربات مسیریاب.

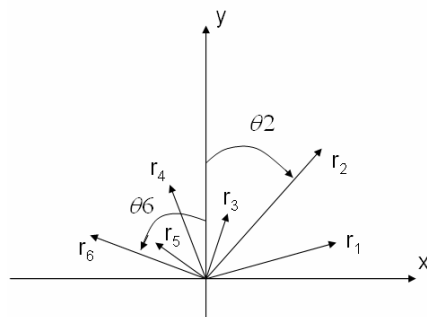
1- مقدمه :

ربات مسیریاب همان طور که از نامش پیدا است، رباتی است هوشمند که قادر است یک خط سفید را در زمین سیاه دنبال کند. در این مقاله سعی بر ساخت ربات مسیریابی است که نمونه آزمایشگاهی خوبی برای پیاده سازی و تست ایده ها و الگوریتم های کنترلی مختلف از جمله کنترل فازی و بررسی چگونگی پیاده سازی و نحوه عملکرد عملی آنها باشد. ایده کلی در کنترل این ربات به کارگیری الگوریتمی است که نیاز به حداقل کد نویسی را داشته اما در عین حال تمامی حالات ممکن که یک ربات در طول یک مسیر با آن برخورد می کند را پوشش داده و بررسی نماید، دارای زمان پردازش کوتاهی بوده، کنترل ربات را که در بسیاری از موارد به فاصله چیده شدن سنسورها بر روی بورد و مکان آن ها وابسته می باشد را تا حدود زیادی از قید وابسته بودن به آن رها سازد که این خود باعث کاهش عملیات مونتاژ سخت افزاری می شود، مشکلات ناشی از نور محیط که باعث ایجاد نویز سنسورها می شود (مثلا ایجاد یک آفست dc بر روی تمامی سنسورها در شرایط نوری محیط های متفاوت) را مرتفع سازد، با استفاده از تعداد اندکی سنسور (در این طرح 6 سنسور) و با به کارگیری تمامی سطوح ولتاژی که یک سنسور در شرایط مختلف نشان می دهد در عین حال که کیفیت کنترل افزایش پیدا کند، تعداد سنسورها را برای کاهش هزینه و سخت افزار به حداقل ممکن برساند.

2- تئوری :

1-2 بیان مسئله :

در الگوریتم این ربات هر سنسور دارای دو آرگومان اندازه و زاویه می باشد که اندازه هر سنسور (r)، در هر لحظه با نمونه گیری ADC از ولتاژ خروجی آنها بدست می آید و زاویه هر سنسور توسط طراح از طریق GUI^1 برای بدست آوردن بهترین حالت انتخاب شده و به ربات ارسال می شود. برای کاهش کدنویسی و در عین حال افزایش حساسیت در محاسبه انحراف ربات از مسیر، یک الگوریتم ریاضی پیشنهاد شده است. با توجه به شکل های (1) و (8) هر سنسور که دارای اولویت و اهمیت بیشتری در محاسبه شدت انحراف است، دارای زاویه بیشتری نسبت به محور y ها می باشد.



(1):

(به عنوان مثال سنسورهای 1 و 6 دارای بیشترین اهمیت و نشان دهنده بیشترین انحراف بوده و همین طور به ترتیب سنسورهای 2 و 5 و سپس 3 و 4 دارای بیشترین اهمیت می باشند.) پس از تصویر کردن و به دست آوردن مولفه های هر سنسور بر روی محورهای x و y با استفاده از روابط مثلثاتی شماره (1) زیر:

$$\begin{aligned} x_n &= |r_n| \cos \theta_n \\ y_n &= |r_n| \sin \theta_n \end{aligned} \quad (1)$$

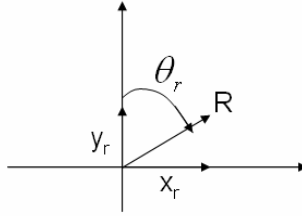
و پس از جمع کردن این مولفه ها با استفاده از روابط (2) و (3)، مولفه های بردار برآیند (x_r و y_r) به دست می آید.

$$x_r = \sum_{n=1}^3 x_n - \sum_{n=4}^6 x_n \quad (2)$$

$$y_r = \sum_{n=1}^6 y_n \quad (3)$$

با توجه به شکل (2) بردار برآیند در هر لحظه دارای یک زاویه θ_r می باشد.

¹ - Graphical User Interface



(2):

از آنجا که اندازه این بردار وابسته به اندازه مولفه های تک تک بردارها بوده و مولفه های هر بردار وابسته به اندازه آن بردار که خود آن نیز در شرایط نوری متفاوت به دلیل نویز مقادیر متفاوتی از خود نشان می دهد، از اندازه بردار برآیند (r) ، در محاسبه انحراف استفاده نمی شود در نتیجه به این ترتیب تاثیرات نوری محیط را بر روی سنسورها حذف می کنیم. با توجه به رابطه (4) در زیر:

$$\theta_r = \tan^{-1}(x_r / y_r) \quad (4)$$

زاویه بردار برآیند محاسبه می شود که نه وابسته به فاصله سنسورها از هم بوده و نه وابسته به شرایط نوری محیط و در عین حال با تنظیم زاویه های $\theta_1, \dots, \theta_6$ از طریق GUI و ارسال آن به ربات، این زاویه یک رابطه خطی با شیب خط پیدا می کند. θ_r با توجه به اینکه $x_r > 0$ یا $x_r < 0$ باشد مقادیر مثبت و یا منفی به خود می گیرد که نشان دهنده اندازه انحراف نسبت به معیار تعادل یعنی محور y ها می باشد. (نهایتاً این زاویه با تنظیمات انجام شده بین $[-90, +90]$ نرمالیزه می شود که زاویه صفر حالت تعادل و $+90$ حداکثر انحراف به راست و -90 حداکثر انحراف به چپ را نشان می دهد).

2-2) استفاده از کنترل کلاسیک به کمک درون یاب نیوتن جهت کنترل مسیر ربات :

پس از محاسبه زاویه انحراف (θ_r) ، باید با توجه به این زاویه مقدار pwm^1 مناسب برای کنترل دور موتورهای تولید شود. در این مرحله برای بررسی حداکثر حالات ممکن پیش آمده از تکنیک درون یابی استفاده شده است. این عمل در عین حال که بررسی تمامی حالات را ممکن می سازد، حجم انبوهی از کدنویسی برای بررسی این حالات را به دو خط تابع ریاضی کاهش می دهد. با استفاده از چند جمله ای درون یاب نیوتن رابطه (5) و 11 نقطه دلخواه، فرمول محاسبه pwm را با یک چند جمله ای از درجه 10 تقریب می زنیم. [3]

$$P_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \dots + f[x_0, x_1, \dots, x_n](x - x_0)(x - x_1) \dots (x - x_{n-1}) \quad (5)$$

و با توجه به این که 11 نقطه دلخواه x_0 تا x_{10} متساوی الفاصله انتخاب می شوند $(x_n - x_{n-1} = h)$ در این صورت چند جمله ای درون یاب به صورت رابطه (6) ساده می شود. [3]

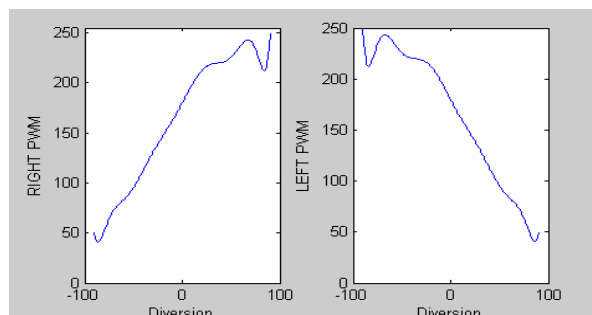
$$P_n(x) = \Delta^0 f_0 + \frac{\Delta^1 f_0}{1!h}(x - x_0) + \frac{\Delta^2 f_0}{2!h^2}(x - x_0)(x - x_1) + \dots + \frac{\Delta^n f_0}{n!h^n}(x - x_0)(x - x_1) \dots (x - x_{n-1}) \quad (6)$$

به طوریکه:

$$\Delta^0 f_0 = f(x_0), \quad \frac{\Delta^1 f_0}{1!h} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}, \quad \frac{\Delta^2 f_0}{2!h^2} = \frac{\frac{f(x_2) - f(x_1)}{h} - \frac{f(x_1) - f(x_0)}{h}}{2h}, \quad \dots$$

¹ - Pulse Width Modulation

پس از بدست آوردن تابع درون یاب (6) می توان با رسم آن در matlab از عملکرد و در واقع درون یابی آن در بازه های مورد نظر اطمینان حاصل کرد. به ازای هر زاویه انحراف، نیاز به دو مقدار pwm و در واقع به 2 تابع درون یاب متفاوت برای هر یک از موتورها داریم تا در هر درجه از انحراف، هر موتور با مقدار pwm خاص خود درایو شود. نهایتاً خروجی های هر یک از دو تابع به Timer میکروکنترلر برای تولید pwm مورد نظر داده می شود. یکی از نقاط ضعف این تکنیک زمانی بروز می کند که بخواهیم تابع درون یاب ایده ال و در واقع دارای تغییرات غیرخطی در بازه مورد نظر باشد. با توجه به شکل (3) تابع بدست آمده در بعضی از بازه ها دچار تغییرات شدید ناخواسته ای می شود که برای رفع این نقص می توان درجه تابع درون یاب را بالاتر برد که این خود شدیداً باعث افزایش زمان پردازش ربات خواهد شد.



:(3)

3-2) استفاده از کنترل فازی در جهت کنترل مسیر ربات :

در این شرایط می توان کنترل فازی را به عنوان یک جایگزین که قابلیت رفع این نقایص را دارد معرفی کرد.

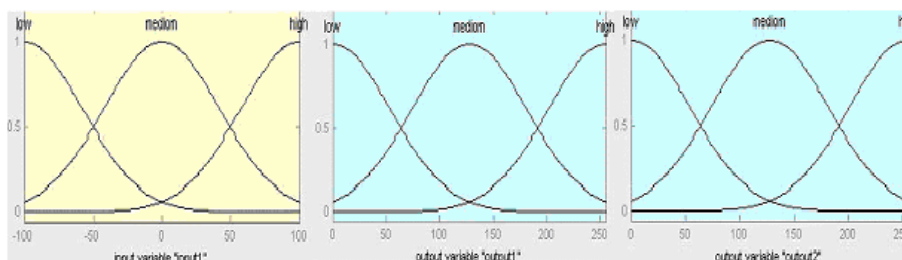
یک سیستم فازی از سه جهت دارای اهمیت می باشد.

1. به دلیل دارا بودن ماهیت مبتنی بر قوانین و استفاده از متغیرهای زبانی، قابلیت به کارگیری دانش انسانی را برای سیستم دارا می باشد.

2. به دلیل عدم وابستگی به مدل سیستم، دارای پایداری¹ بالا می باشد.

3. به دلیل دارا بودن خاصیت تقریب گری عمومی، به صورت بالقوه دارای بهترین عملکرد می باشد.[1]

در سیستم فازی طراحی شده، انحراف از خط، به عنوان تنها ورودی و pwm موتورهای چپ و راست، خروجی های سیستم در نظر گرفته شده اند. در شکل (4) توابع عضویت ورودی و خروجی ها مشاهده می شود.

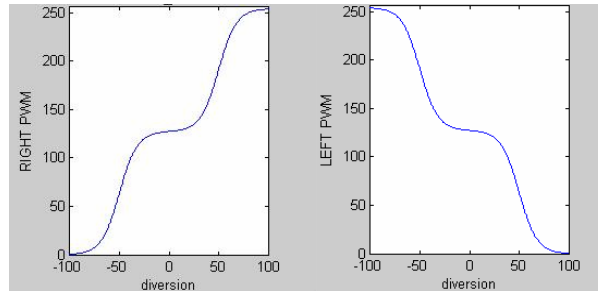


:(4)

¹ . Robustness

مجموعه قواعد فازی به کار گرفته شده برای کنترل pwm موتورهای چپ و راست عبارت اند از:

- 1.if (input=low)then (output1=low)(output2=high)
- 2.if (input=mediom)then (output1= mediom)(output2= mediom)
- 3.if (input=high)then (output1=high)(output2=low)



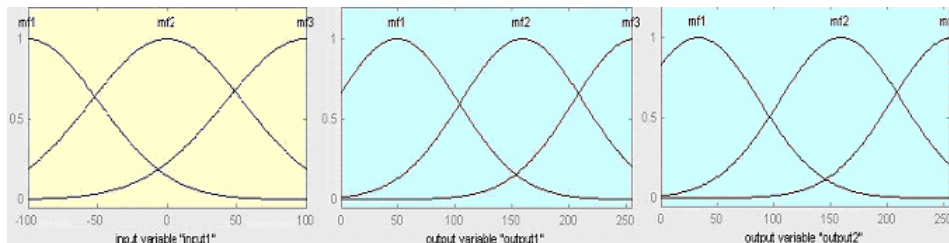
:(5)

شکل (5) نشان دهنده خروجی سیستم فازی با موتور استنتاج ضرب، فازی ساز منفرد، غیرفازی ساز میانگین مراکز، توابع تعلق نشان داده شده در شکل(4) و قواعد فازی مذکور است.[1]

در نمودارهای شکل (5) محورهای افقی نشان دهنده مقدار انحراف ربات (θ_r) و محورهای عمودی نشان دهنده مقدار pwm موتورهای چپ و راست است که سیستم فازی مقدار آنها را تعیین کرده است.

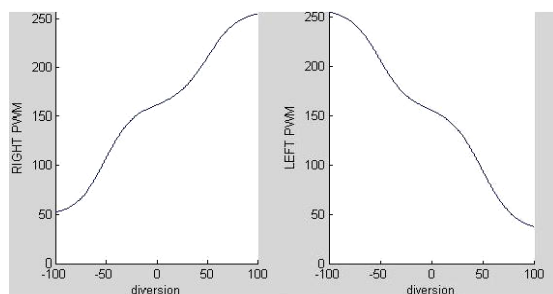
یکی از نقاط قوت تکنیک درون یابی در مقایسه با کنترل فازی برای کنترل این ربات مدل، ساده گی پیاده سازی و کوتاهی الگوریتم های آن بدلیل خاصیت ریاضیاتی بودن این الگوریتم ها می باشد. از آنجا که ممکن است سیستم فازی مذکور بعد از پیاده سازی بر روی ربات، از کارایی کافی برخوردار نباشد و نتواند ربات را آنطور که باید کنترل کند قبل از پیاده سازی، نقاطی که به صورت کاملاً عملی و در هنگام حرکت به دست آمده بود، به سیستم فازی اعمال شد و سپس با استفاده از الگوریتم نزول گرادیان پارامترهای موجود در توابع عضویت ورودی و خروجی ها از جمله مراکز و واریانس توابع عضویت تغییر داده شد تا سیستم تا حد امکان با شرایط موجود تطبیق پیدا کند.[1]

شکل (6) نشان دهنده توابع عضویت ورودی و خروجی ها بعد از اعمال تغییرات است .



:(6)

واضح است که پس از تطبیق دادن انتظار می رود که سیستم، پاسخ منطقی تری را در ازای نقاط مذکور بدهد. شکل (7) نشان دهنده خروجی سیستم فازی تطبیق یافته است.



(7):

همان طور که در نمودار های شکل (7) مشاهده می شود خروجی سیستم جدید، بسیار منطقی تر از حالت اولیه است. با مقایسه شکل (7) و شکل (3)، بدست آمده توسط تابع درون یاب می توان به ایده ال بودن و یکنواختی اهنگ تغییرات خروجی سیستم فازی نسبت به ورودی آن پی برد.

3- پیاده سازی :

پس از طرح ایده، جهت بررسی عملکرد الگوریتم های کنترلی، اقدام به ساخت یک ربات مدل شد که در زیر به اختصار به مراحل این کار اشاره می شود.

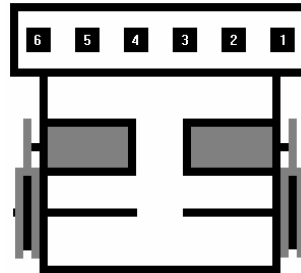
3-1 طراحی GUI :

یک نرم افزار گرافیکی با استفاده از زبان 6 Visual Basic طراحی گردید و امکاناتی در این نرم افزار در نظر گرفته شد تا به طراح ابزارهای قدرتمندی جهت بررسی رفتار ربات و نیز عملکرد کنترل پیاده شده بر روی آن راه، بصورت همزمان و در هنگام کار ربات بدهد. از آن جا که بسیاری از مشکلات احتمالی از جمله نویز حاصل از نور محیط بر روی سنسورها و نیز عدم تطابق مکانیک (تفاوت گشتاور موتورها و یکسان نبودن گیربکس ها) در هنگام شبیه سازی بروز نمی کند این GUI امکان پیدا کردن و رفع این نقایص را پس از ساخت بسیار ساده کرده و کیفیت نهایی کار را بالا می برد. در این بخش به اختصار توانایی های GUI مذکور مورد بررسی قرار می گیرد:

- 1- امکان انتخاب و رسم مقادیر هر سنسور و نیز زاویه θ (انحراف ربات بر روی خط) محاسبه شده توسط رابطه (4) (که بین 90- تا 90+ نرمالیزه شده است)، به صورت همزمان و پیوسته در حین حرکت ربات (این مورد قدرتمند ترین توانایی این GUI در بررسی عملکرد ربات می باشد. برای این کار با ساخت یک فایل DLL در نرم افزار matlab و به کارگیری این فایل در نرم افزار VB و با فراخوانی توابع matlab (از جمله تابع PLOT) امکان آنالیز این مقادیر توسط توابع قدرتمند matlab مستقیماً از داخل این GUI فراهم شد. این عمل این امکان را به طراح می دهد تا از شروع حرکت ربات تا لحظه دلخواه تغییرات سنسورها و انحراف را ثبت و رسم شده داشته باشد تا به این ترتیب در صورت بروز خطا به راحتی به منشا آن و رفع نقص الگوریتم خود بپردازد).
- 2- توانایی ساخت تابع درون یاب درجه ده نیوتن با وارد کردن ده مقدار دلخواه برای pwm برای رسیدن به تابع مورد نظر.
- 3- امکان انتخاب و ارسال زاویه هر سنسور به ربات برای اعمال شدن در الگوریتم آن.
- 4- امکان نمایش همزمان مقادیر نمونه گرفته شده توسط ADC از هر شش سنسور و نیز خروجی دو تابع درون یاب (pwm دو موتور) در حین کار ربات.
- 5- امکان اعمال pwm به صورت دستی با استفاده از GUI به ربات برای تخمین مقادیر آن در ساخت تابع درون یاب.
- 6- امکان تغییر فرکانس pwm اعمال شده به موتورها.

3-2) مکانیک ربات :

با توجه به شکل (8) در این ربات از دو موتور DC ساده استفاده شده است که با استفاده از چرخ دنده سه دور چرخش رتور موتور یک دور چرخش چرخ ربات را نتیجه می دهد. موقعیت چیده شدن سنسور ها نیز در شکل (8) نمایش داده شده است.



(8): 6

3-3) سخت افزار :

پردازنده این ربات یک میکروکنترلر از شرکت ATMEL به نام ATmega64 می باشد. در این میکروکنترلر ابزارهای قدرتمندی از جمله ADC، USART، TIMER، گنجانده شده است. این میکروکنترلر دارای یک ADC شش کاناله با قابلیت مالتی پلکس شدن برای انتخاب یکی از شش کانال ورودی می باشد که با فرکانس 200 KHZ از خروجی سنسورها نمونه برداری می کند. برای ارتباط ربات با PC از USART میکروکنترلر استفاده شده است. این ربات با نرخ ارسال 19200 Kb\ s با pc ارتباط full duplex دارد. پس هر مرتبه نمونه گیری از سنسورها و نیز محاسبه انحراف و مقدار دو تابع درون یاب (pwm دو موتور) در صورت درخواست GUI این مقادیر طبق ترتیب و الگوریتم مشخص به PC ارسال می شود. برای تولید پالس pwm، از تایمر میکروکنترلر که قابلیت تولید سخت افزاری pwm از دو کانال مجزا برای هر یک از موتورها را داراست استفاده شده است. در هر لحظه دو مقدار محاسبه شده توسط دو تابع درون یاب به تایمر برای تولید pwm مورد نظر داده می شود. در این ربات از سنسورهای بازتابشی مادون قرمز استفاده شده است. ولتاژ خروجی هر سنسور از لحظه ای که کاملاً روی زمین سفید قرار دارد تا لحظه ای که کاملاً بر خط مشکی منطبق شود به صورت خطی تغییر کرده، با استفاده از همین تغییر و نمونه گیری از آن توسط ADC در هر لحظه انحراف دقیق ربات نسبت به خط قابل محاسبه است.

4- نتیجه :

در این مقاله، پس از طرح ایده و پیاده سازی این ربات با استفاده از تکنیک درون یابی، موفقیت قابل قبولی حاصل شد، به طوری که می توان ادعا داشت که به کارگیری این روش می تواند در بررسی حالات و کنترل دور موتور برای ربات های تعقیب خط و احیاناً سایر ربات های مشابه که سنسورهایی برای درک محیط اطراف و بررسی حالات متعدد دارند از جمله ربات لایبرنت و حتی سنسورهای صنعتی مفید واقع شود. همانطور که اشاره شد فرمول درون یاب نیوتن استفاده شده، در فضای $R \rightarrow R$ درون یابی کرده و می توان برای بررسی حالات بسیار پیچیده که سیستم دارای چندین ورودی مختلف می باشد از فرمول درون یاب لاگرانژ که دارای روش مشابه درون یاب نیوتن بوده اما ویژگی مهم قابلیت تعمیم از درون یابی در فضای $R \rightarrow R_n$ را داراست استفاده کرد.[2]

البته تعمیم و پیاده سازی چنین تابعی و از همه مهمتر پردازش آن نیاز به زمان زیادتری داشته که غالباً به دلیل طولانی شدن زمان پردازش، از بازه طراحی های میکروکنترولی خارج می گردد و در پردازش های مبتنی بر PC کاربردی تر خواهد بود.

همچنین کنترلر فازی را نیز برای این ربات پیاده سازی نمودیم و مزایا و معایب آن را مورد بررسی قرار دادیم. نتایج بدست آمده نشان می دهد که از کنترلر فازی می توان به عنوان یک کنترلر بسیار قوی و موثر برای کنترل مسیر ربات استفاده نمود.

5- مراجع:

[1]: L.Wang, *A Course in Fuzzy Systems and Control*, Prentice Hall, 1997.

[2]: Epperson, J.F, *An Introduction to Numerical Methods and Analysis*, John Wiley & Sons, Portland, 2002.

[3]: اصغر کرایه چیان، محاسبات عددی، انتشارات دانشگاه فردوسی، پاییز 1382.

[4]: کازوتاناکا، ترجمه ع.وحیدیان کامکاد و ح. طارقیان، مقدمه ای بر منطق فازی برای کاربردهای عملی، انتشارات دانشگاه فردوسی، 1381.