



ارائه شده توسط:

سایت ترجمه فا

مرجع جدیدترین مقالات ترجمه شده

از نشریات معتبر

پیاده سازی کنترل کننده منطق فازی و کنترل کننده PID برای یک موتور DC

کدگذار - " یک مطالعه موردی برای خودروی هدایت شده خودکار "

چکیده

به منظور کاهش زمان نآآمد (مدت زمان از کارافتادگی) و زمان تدارک، کار کردن AGV با سرعت ثابت یا RPM در کارخانه ضروری است. کنترل کننده منطق فازی و کنترل کننده PID برای به دست آوردن RPM ثابت در AGV به کار برده می شوند. AGV شامل موتور DC جاروبکدار با کدگذار، درایور موتور، میکروکنترلر و باتری می باشد. کدگذار به شکل موقعیت شفت، به میکروکنترلر فیدبک می دهد. میکروکنترلر، خطا در سیستم را بر اساس پارامترهای تعریف شده توسط الگوریتم ها کاهش می دهد. فاز اول مقاله، اطلاعات مختصری در مورد سخت افزار، نرم افزار و الگوریتم ها ارائه می دهد. در فاز دوم مقاله، روش شناسی پیاده سازی الگوریتم به سیستم نشان داده شده است. در فاز نهایی مقاله، نتایج و بررسی ها بر اساس الگوریتم های کاربردی ذکر شده است. مقایسه بین کنترل کننده PID و کنترل کننده PID فازی نیز نشان داده شده است. الگوریتم های Ziegler-Nichols برای پیدا کردن پارامترهای PID استفاده می شود. MATLAB simulink و جعبه ابزار منطق فازی برای شبیه سازی استفاده می شوند. میکروکنترلر Arduino برای پذیرش بازخورد داده شده توسط کدگذار و کنترل سرعت موتور استفاده می شود. در پوسته یک مهره، این استراتژی های کنترل به AGV کمک می کنند تا با RPM ثابت و با زمان استقرار، خطای حالت پایدار و اورشوت کاهش یافته کار کند.

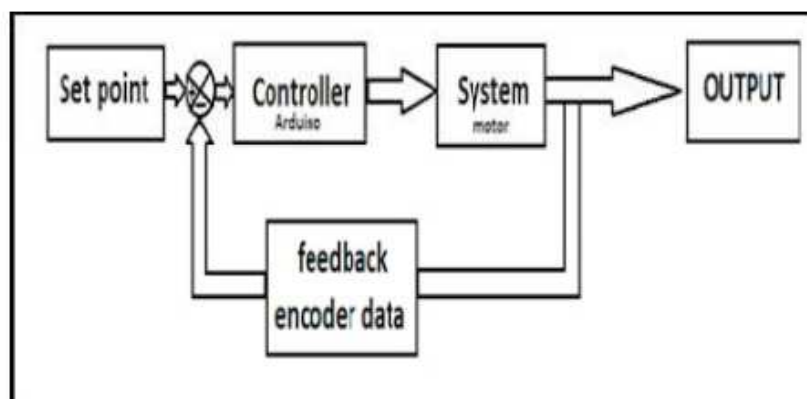
کلید واژه ها: کنترل کننده Arduino، MATLAB Simulink؛ کنترل کننده منطق فازی؛ AGV؛ کدگذار؛ کنترل

کننده PID؛ زمان استقرار؛ زمان نآآمد؛ اورشوت

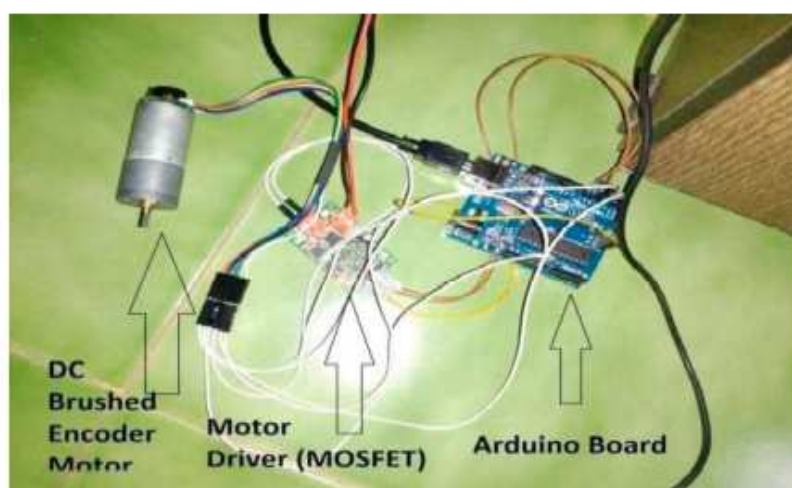
در این مقاله، روش شناسی پیاده سازی الگوریتم کنترل PID و الگوریتم کنترل منطق فازی را برای راه اندازی AGV در سرعت ثابت نشان داده شده است. برای دستیابی به سرعت ثابت AGV، سرعت یک موتور DC با استفاده از الگوریتم های فوق الذکر کنترل می شود [۱]. شکل ۱، بلوک دیاگرامی یک سیستم را نشان می دهد که در آن کدگذار به عنوان عنصر بازخورد استفاده می شود. موتور به عنوان یک کارخانه در نظر گرفته شده و Arduino (کنترل کننده منبع باز) به عنوان یک کنترل کننده استفاده می شود. نقطه تنظیم در شکل RPM است و خروجی در شکل وضعیت شفت موتور است [۲]. پارامترهای PID با استفاده از الگوریتم Ziegler Nichols یافت می شوند. فرایند منطق فازی به سه قسمت تقسیم می شود. بلوک دیاگرام منطق فازی در شکل ۴ نشان داده شده است. با توجه به نقطه مرجع، اولین فازی سازی اتفاق می افتد. موتور تداخلی در تصویر بین فرآیند فازی سازی و فازی زدایی می آید [۳-۵]. (۱) فازی سازی که داده های اندازه گیری شده را (به عنوان مثال RPM موتور، ۲۰۰ است) را به داده های لفظی تبدیل می کند (به عنوان مثال، موتور، بسیار کند است). پس از پایان فرایند فازی سازی، قوانین فازی را می توان به کار برد [۴]. (۲) موتور تداخل که انسجام و تحلیل مناسب را برای یک شبیه سازی خروجی فراهم می کند. (۳) فازی زدایی، خروجی را بر اساس تابع عضویت و قواعد تعریف شده ارائه می کند [۵] [۱۳].

اصطلاحات				
s	اپراتور لاپلاس	w	فرکانس طبیعی نوسانات	K بهره سیستم
Km	ثابت اندازه موتور	Kb	ثابت EMF پشتیبان	J اینرسی موتور
ξ	نسبت میرایی	wd	سرعت زاویه ای میراشده	Kp ثابت خطای موقعیتی
Kv	ثابت خطای سرعت	Ka	ثابت خطای شتاب	ess خطای حالت پایدار
G(s)	تابع انتقال	Z-N	روش Ziegler Nichols	FSL سیستم منطق فازی
C(t)	پاسخ خروجی	Td	زمان تاخیر	Tr زمان صعود
Tp	زمان برداشتن	Mp	اورشوت برحسب %	Ts زمان نشست

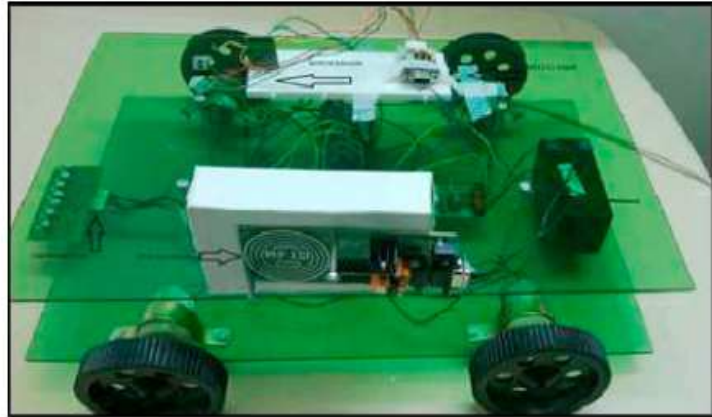
شکل ۳. دید جلوی AGV را نشان می دهد که در این پروژه مورد استفاده قرار می گیرد. AGV باید فقط مسیر مستقیم سیاه را دنبال کند، که توسط حسگرهای IR آشکار شده است. این کار پیشنهادی فقط ایده ای در مورد کنترل کننده PID و منطق فازی که در AGV پیاده سازی می شود، ارائه می دهد [۳]. در این کار، بقیه چیزها در نظر گرفته نشده است. تنظیم کامل برای پیاده سازی کنترل کننده فازی و PID در شکل ۲ نشان داده شده است. ۲. در این کار باید دو مورد مهم را باید در نظر گرفت. وزن خود AGV به عنوان یک بار محسوب می شود و اصطکاک در سیستم نادیده گرفته می شود [۶].



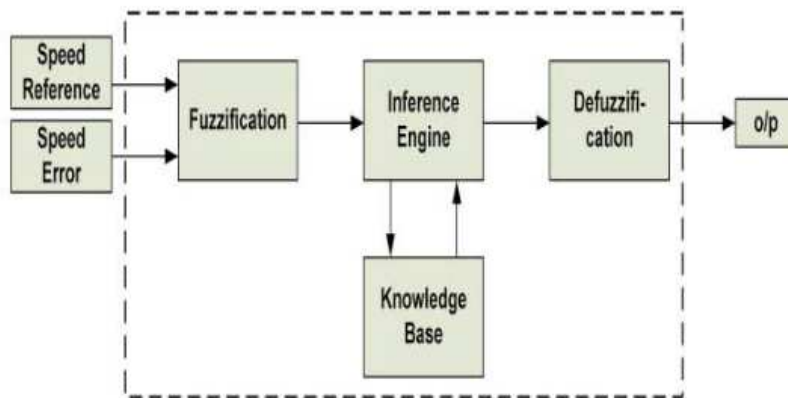
شکل ۱ بلوک دیاگرام سیستم کنترل PID



شکل ۲. تنظیم آزمون



شکل ۳. وسیله نقلیه هدایت شده خودکار مورد استفاده در این کار



شکل ۴: بلوک دیاگرام کنترل کننده منطق فازی [۱۳]

جدول ۱ اطلاعات کامل در مورد AGV مورد استفاده در این پروژه را نشان می دهد. جدول ۲ اطلاعات کامل در مورد موتور مورد استفاده در AGV را ارائه می دهد.

جدول ۱. مشخصات AGV

مقادیر	پارامترها	شماره سریال
35cm x 25cm x 15cm	L x B x H	۱
۳,۲ کیلوگرم	جرم AGV	۲
۴	تعداد موتورها	۳
کدگذار جاروبکدار DC	انواع موتورها	۴

۵	مشخصات موتور	۶ ولت, 500 RPM, ۳,۶ کیلوگرم در ۱,۲ آمپر
۶	قطر چرخ	۱۰ سانتی متر
۷	مسیری که باید دنبال شود	خط تیره مستقیم
۸	مسیری که دنبال شده است	آرایه سنسور IR
۹	نسبت دنده موتور	36:1

جدول ۲. مشخصات موتور DC کدگذار که در AGV استفاده می شود

اطلاعات	نمادها	پارامترهای موتور	شماره سریال
۱ هانری	La	اندوکتانس موتور	۱
4Ω	Ra	مقاومت آرماتور	۲
0.005Kg.m^2	J	اینرسی موتور	۳
0.01 N.m/Amp	Ka	ثوابت گشتاور	۴
نوع صفر	معادله ۲ را ببینید	نوع سیستم	۵
مرتبه دوم	معادله ۱ را ببینید	مرتبه سیستم	۶
واحد	-	نوع بازخورد	۷

۲. توصیف مسئله و روش شناسی

۲,۱ توصیف مسئله

هدف اصلی مقاله این است که زمان استقرار و اورشوت در موتور DC کدگذار برای راه اندازی AGV با RPM ثابت است. دومین کار، یافتن تابع انتقال سیستم به منظور اعمال PID و فازی سازی بر روی آن است. مسئله سوم و نهایی، واسطه نمودن MATLAB Simulink با یک AGV از طریق میکروکنترلر است.

۲. روش شناسی

در کار پیشنهادی، تکنیک های کنترل کننده PID با استفاده از الگوریتم Ziegler Nichols استفاده می شوند و تکنیک Mamdani برای کنترل کننده منطق فازی [۷] استفاده می شود. در مقایسه پایانی زمان استقرار، زمان

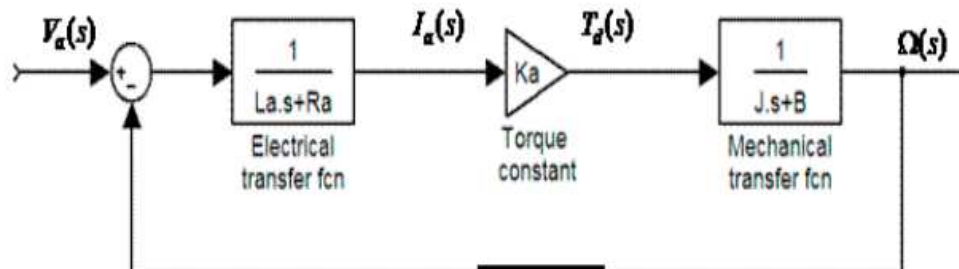
صعود، زمان پیک، اورشوت ها و زمان تاخیر بین کنترل کننده PID و کنترل کننده منطق فازی نشان داده می شوند. میکروکنترلر ۸ بیتی Arduino برای پردازش و کاربرد الگوریتم ها استفاده می شود. MATLAB simulink برای شبیه سازی ها استفاده می شود [۱۵].

۳. مدلسازی و شبیه سازی ریاضی

۳،۱ تابع انتقال سیستم

شکل ۵، یک بلوک دیاگرام هدف کلی برای شبیه سازی سیستم را نشان می دهد. این سیستم دارای دو قسمت اصلی است، یکی بخش مکانیکی است و دیگری قسمت الکتریکی. می توان از بلوک دیاگرام مشاهده کرد که سیستم پیشنهادی، یک سیستم مرتبه دوم است. پارامترهای شناخته شده موتور در جدول ۲ آمده است [۸].

نوع سیستم و مرتبه سیستم را می توان با استفاده از معادله ۱ و معادله ۲ پیدا کرد [۱] [۷]. نوع سیستم، صفر است و مرتبه سیستم، دو است. تابع انتقال حلقه باز سیستم در معادله ۳ داده شده است که در آن K_a ، پارامتر ناشناخته (بهره سیستم) است و می تواند با استفاده از معیار Routh آن را یافت. معادله ۴، یک معادله مشخص برای ثبات است. معادله ۳ منجر به معادله $3a$ می شود، جایی که $200K_a=K$ (بهره کلی سیستم). با توجه به جدول ۳، با استفاده از معیار Routh، مقدار $-0.04 < K_a$ ، بنابراین در اینجا مقدار K_a به عنوان 0.04 گرفته شده است. سیستم داده شده تحت میرایی است و مقدار $\xi = 0.60$ ، که کمتر از یک است. معادله $3b$ ، تابع انتقال حلقه بسته سیستم است [۹] [۱۰].



شکل ۵ بلوک دیاگرام سیستم برای شبیه سازی

$$G(s)H(s) = K \frac{(1+T_1(s))(1+T_2(s))\dots(1+T_n(s))}{(1+T_a(s))(1+T_b(s))\dots(1+T_m(s))} \quad (1)$$

$$G(s)H(s) = K \frac{(1+T_1(s))(1+T_2(s))\dots(1+T_n(s))}{s^j(1+T_a(s))(1+T_b(s))\dots(1+T_m(s))} \quad (2)$$

$$G(s)H(s) = \frac{K_a}{(s+4)(0.005s+0.01)} \quad (3) \quad 1 + K_a G(s)H(s) = 0 \quad (4)$$

(3a) که در آن، $K_a=0.04$ و اصطکاک چسبنده B، 0.1، 0 است

$$\frac{G(s)}{1+G(s)H(s)} = \frac{16}{(s+4)(s+2)} \quad (3b) \text{ (تابع انتقال حلقه بسته سیستم)}$$

۳،۲ تحلیل حالت پایدار سیستم

معادله ۵ برای سیستم تحت میرایی ($\xi < 1$) است. پارامترهای حالت پایدار (K_p, K_v, K_a) و (ess) را

می توان با استفاده از معادله ۶ تا ۹ [4] یافت.

$$C(t) = \left(\frac{\omega_n}{\sqrt{1-\xi^2}} \right) e^{-\xi\omega_n t} \sin \omega_d t \quad (5) \quad K_p = \lim_{s \rightarrow 0} (G(s)H(s)) \quad (6)$$

$$K_v = \lim_{s \rightarrow 0} s(G(s)H(s)) \quad (7) \quad K_a = \lim_{s \rightarrow 0} s^2(G(s)H(s)) \quad (8)$$

$$E_{ss} = \frac{1}{1+K_p} \quad (9)$$

۳،۳ تحلیل گذرای سیستم

پارامترهای گذرا در معادله ۱۰ تا ۱۴ نشان داده شده است. هر سیستم با رفتار نوسانگر مخالفت می کند (میرایی).

این تمایل، قطب های حلقه بسته سیستم را کنترل می کند. پاسخ سیستم توسط قطب های حلقه بسته سیستم

تعیین می شود [1].

$$T_d = \frac{(1+0.7\xi)}{\omega_n} \quad (10)$$

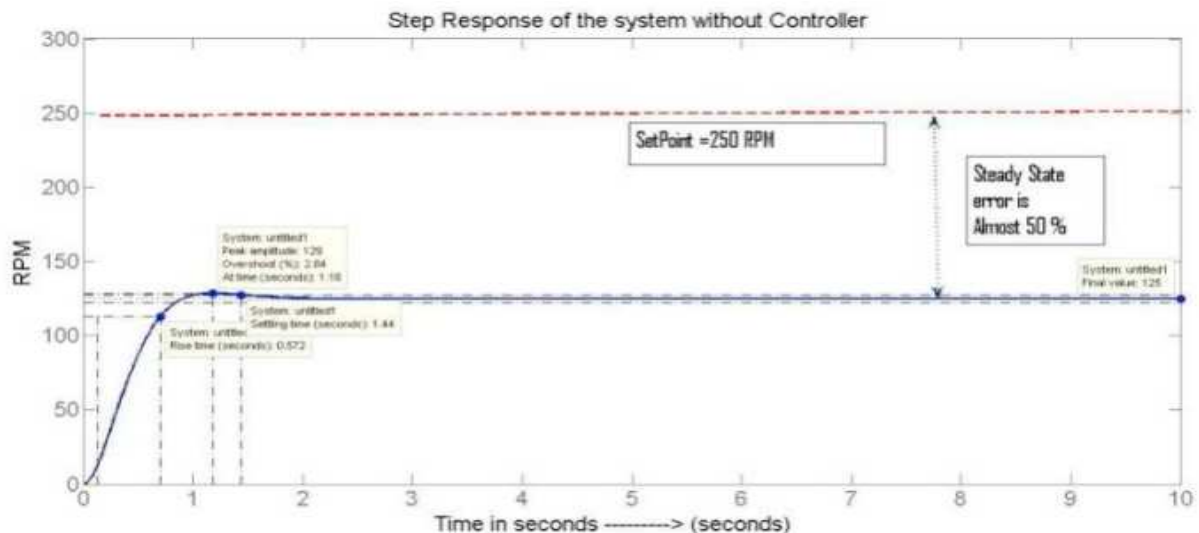
$$T_r = \frac{(\pi-\beta)}{\omega_d} \quad (11)$$

$$T_p = \frac{(\pi)}{\omega_d}$$

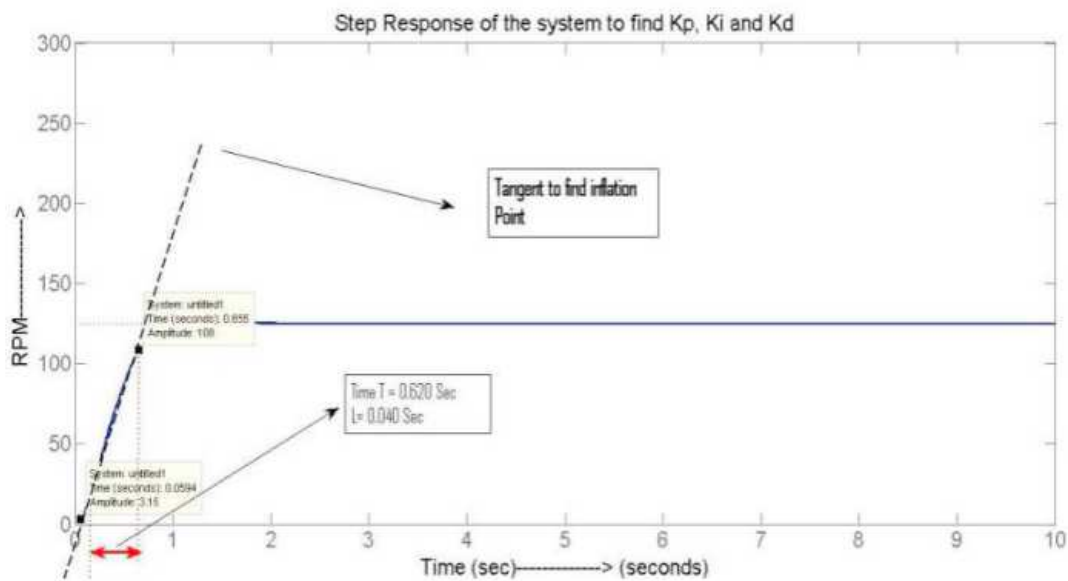
$$(12) M_p \text{ in \%} = 100 \left(e^{\frac{-\xi\pi}{\sqrt{1-\xi^2}}} \right) \quad (13) T_d = \frac{(4\xi)}{\omega_n} \quad (14)$$

۳,۴ شبیه سازی کنترل کننده PID با استفاده از الگوریتم Ziegler-Nichols

پس از پیدا کردن تابع انتقال سیستم، یافتن K_p ، K_i و K_d (پارامترهای PID) حیاتی می شود. الگوریتم Z-N یکی از الگوریتم های قدرتمند برای تنظیم و پیدا کردن پارامترهای K_p ، K_i و K_d است. همچنین برای سیستم مرتبه دوم به خوبی عمل می کند [۱-۳]. الگوریتم ZN با زمان نهایی و بهره نهایی سرو کار دارد، بنابراین لازم است پاسخ سیستم را بدون هیچ گونه کنترل کننده پیدا کنیم. پاسخ سیستم کمک می کند تا K_p ، K_i و K_d را با استفاده از جدول ۵ و شکل ۷ پیدا کنیم. خط منحنی که در شکل ۷ نشان داده شده است، زمان نهایی نامیده می شود (T ، جایی که $T = 0.62$ از نمودار). طول L ، تقریباً برابر با زمان تاخیر است ($L = 0.06$ از گراف). شکل ۸ بلوک دیاگرام شبیه سازی سیستم برای مقایسه پاسخ شبیه سازی ارائه شده است [۲].



شکل ۶. پاسخ سیستم بدون هیچ کنترل کننده

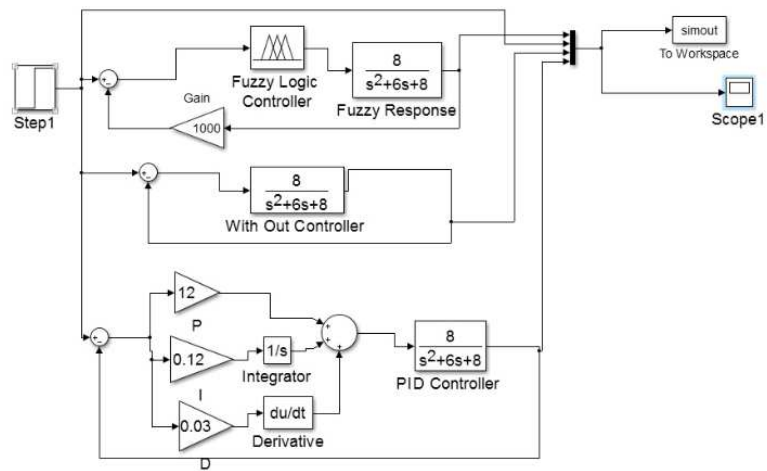


شکل ۷: پاسخ سیستم برای پیدا کردن K_p , K_i و K_d

چهار پاسخ در شبیه سازی اتخاذ شده اند. پاسخ پله، پاسخ بدون هیچ کنترل کننده، پاسخ کنترل کننده PID و پاسخ کنترل منطق فازی. به همین دلیل است که MUX قبل از هدف ادغام همه پاسخ ها متصل شده است [۹-۱۳]. این را می توان از شکل ۶ مشاهده کرد که خطای ثابت تقریباً ۵۰٪ در سیستم است (زمانی که هیچ کنترل کننده ای وجود ندارد). معادله ۱۵ و ۱۶، تابع انتقال برای کنترل کننده PID هستند. یافتن T و L برای تنظیم K_p , K_i و K_d در شکل ۷ نشان داده شده است. در نقطه تورم نقطه به منظور محاسبه T و L ، یک تانژانت ترسیم شده است جایی که T ، یک زمان از زمان تاخیر تا زمان صعود است و L ، یک زمان از مبدا تا زمان تأخیر است [۱۴]. زمان تاخیر زمانی است که سیستم شروع به پاسخ می کند. زمان صعود، از زمانی است که سیستم منحنی می شود. در اینجا $T = 0.62$ ثانیه و $L = 0.0$ ثانیه از گراف. لازم به ذکر است که شبیه سازی برای ورودی پله واحد انجام می شود که می تواند در هر نسبت مقیاس بندی شود [۱] [۳].

$$PID(s) = K_p + \left(\frac{1}{s}\right)(K_i) + K_d s \quad (15)$$

$$PID(s) = 12 + \left(\frac{1}{s}\right)(0.12) + 0.03s \quad (16)$$



شکل ۸ بلوک دیاگرام شبیه سازی سیستم

ضریب ثانویه ضریب اولیه اپراتور لاپلاس

S^2	1	$8+200Ka$
S^1	6	0
S^0	$(8+200Ka)$	-

جدول ۳. معیار Routh

RPM	محدوده RPM	ولتاژ	محدوده (ولت)
S1	0,0,90	PWM3	1.5, 2.0, 2.6
S2	50,95, 140	PWM3	1.5, 2.0, 2.6
S3	110,150,190	PWM2	0.7,1.3,2.0
S4	160,210,260	PWM1	0, 0 , 1.3

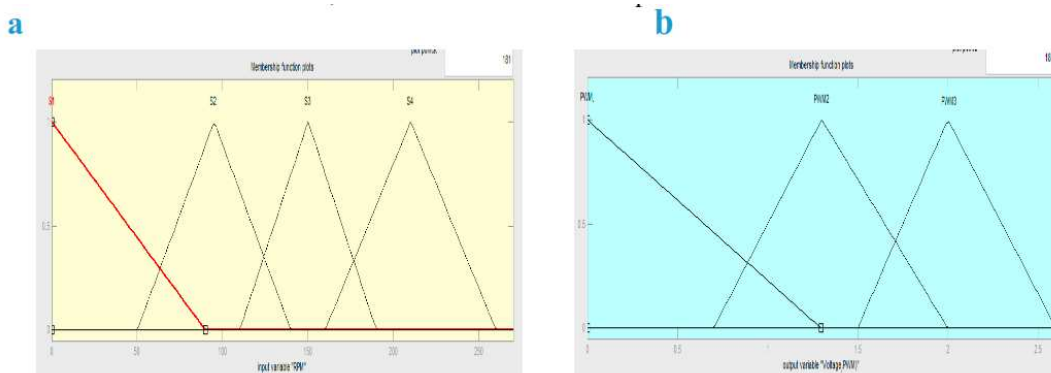
جدول ۴ قوانین فازی برای سیستم

جدول ۵ مقادیر P، PI و PID برای استفاده از الگوریتم Z-N

Controller	K_p	K_i	k_d	K_p	K_i	k_d
P	T/L	-	-	10	-	-
PI	0.9(T/L)	L/0.3	-	9.3	0.2	-
PID	1.2(T/L)	2L	0.5 L	12	0.12	0.03

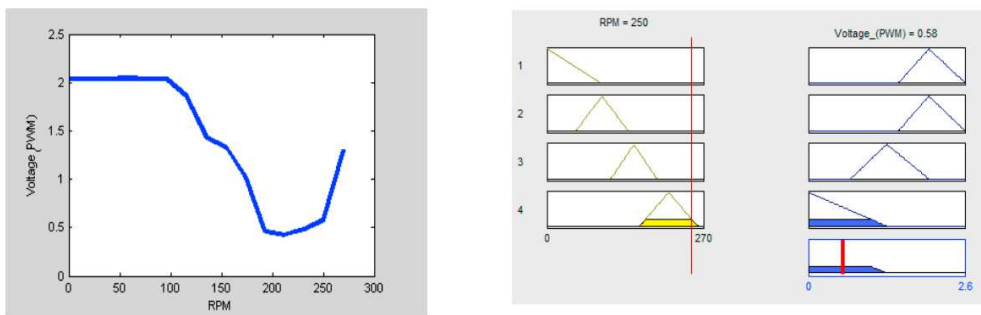
۴. پیاده سازی کنترل کننده منطق فازی

شکل 9a و 9b، تابع عضویت ورودی و خروجی را به ترتیب برای کنترل کننده منطق فازی نشان می دهد. چهار تابع عضویت در ورودی و سه تابع عضویت در خروجی وجود دارد. توابع ورودی مربوط به RPM هستند و توابع عضویت خروجی در شکل ولتاژ هستند. محدوده RPM ورودی، 0 تا ۲۷۰ RPM است و ولتاژ خروجی خروجی ۰ تا ۲,۶ ولتاژ است [۴] [۷]. ورودی ها به صورت S1، S2، S3 و S4 نامیده می شوند، در حالیکه خروجی ها به صورت PWM1، PWM2 و PWM3 نشان داده می شوند. PWM به معنای مدولاسیون عرض پالس است. نقطه تنظیم در سیستم، RPM ۲۵۰ است و PWM با ۵ ولت کار می کند، بنابراین ۲,۵ تا ۲,۶ ولت قابل قبول است.



شکل ۹ (a) تابع عضویت ورودی؛ (b) تابع عضویت خروجی

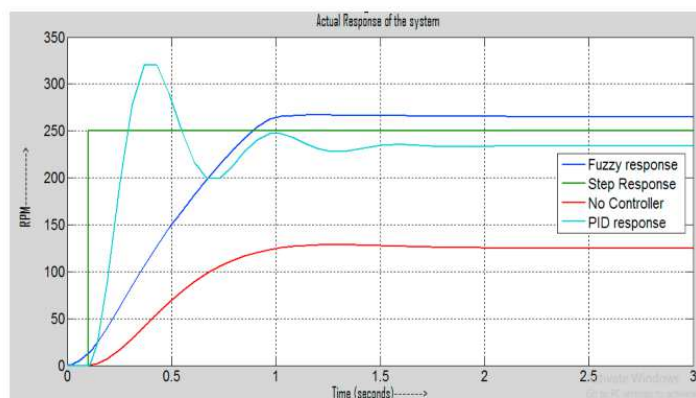
شکل 10a و 10b، نتیجه حاصل از دید سطحی و نمایش قاعده سیستم را نشان می دهد. می توان از بیننده سطح مشاهده کرد که چگونه ولتاژ با افزایش تدریجی RPM با استفاده از کنترل کننده منطق فازی کاهش می یابد. با توجه به دیدگاه در ۲۵۰ دور در دقیقه، ولتاژ باید ۰,۵۸ باشد. قواعد فازی در جدول ۵ نشان داده شده است.



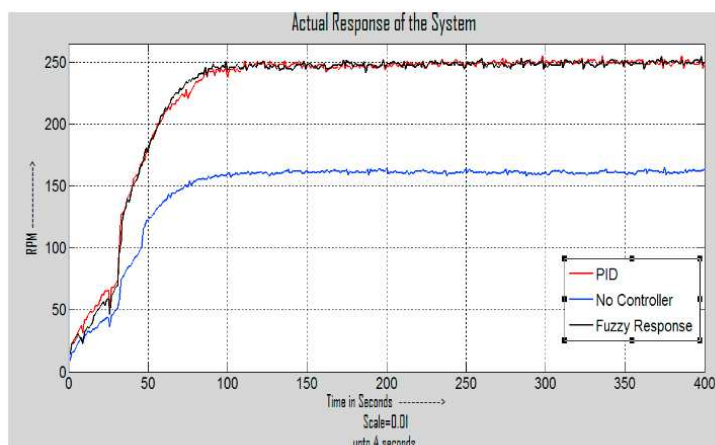
شکل ۱۰ (a) بیننده سطح (b) (er) بیننده قاعده ولتاژ و RPM برای سیستم معین

۵. نتایج و بحث

شکل ۱۱ و ۱۲، به ترتیب پاسخ واقعی و شبیه سازی شده و واقعی سیستم را نشان می دهند. شبیه سازی با ورودی پله واحد انجام می شود (که می توانیم آن را نقطه تنظیم خودمان در نظر بگیریم). خط بنفش، خط ورودی است که یک پاسخ پله واحد است. پاسخ آبی، پاسخ سیستم بدون کاربرد هر الگوریتم کنترل است. خطوط زرد و قرمز به ترتیب PID و پاسخ فازی سیستم می باشند. تمام مقادیر پارامتر حالت پایدار و گذرا در جدول ۶ نشان داده شده است.



شکل ۱۱ پاسخ شبیه سازی شده سیستم



شکل ۱۲ پاسخ واقعی سیستم

جدول ۶ نتایج نهایی سیستم (نتایج شبیه سازی شده و واقعی)

شماره سریال	پارامترها	بدون کنترل	بدون کنترل	مقدار PID	مقدار PID	منطق فازی	منطق فازی
		کننده شبیه سازی شده	کننده عملی	شبیه سازی شده	عملی	شبیه سازی شده	عملی
		Simulated	Practical	PID	PID	Simulated	Practical
1	Ess	50%	37%	6%	2%	3%	0.5%
2	Mp	Undershoot	Undershoot	16%	2.5 %	3%	1 %
3	Ts	2.1 sec	1.5 sec	2.5 Sec	1.5 Sec	0.9 sec	0.8 Sec
4	Tp	1.3 Sec	1.3 Sec	0.4 sec	1.3 sec	0.8 sec	0.7 sec
5	Tr	0.75 sec	0.75 sec	0.3 sec	0.7 sec	0.5 sec	0.6 sec
6	Td	0.1sec	0.1sec	0.1 sec	0.01sec	0.05 sec	0.01 sec

۶ نتیجه گیری مبتنی بر نتایج

- خطای حالت پایدار، اورشوت و زمان صعود سیستم در کنترل کننده PID نسبتاً برای کنترل کننده منطق فازی کمتر است.

- AGV, سرعت ثابت را درست در ۰,۸ ثانیه فقط با ۱٪ اورشوت به دست می آورد
- فقط ۱٪ اورشوت برای کنترل کننده منطق فازی رخ می دهد
- خطای حالت پایدار با کنترل کننده PID کمتر مقایسه می شود
- زمان صعود در کنترل کننده منطق فازی نیز کمتر است.
- کنترل کننده منطق فازی, ثبات و دقت بیشتری به سیستم می دهد.
- نوسان های کوچک در پاسخ را می تواد با استفاده از کنترل کننده ANFIS در آینده کاهش داد.

References

- [1] P. Parikh, R. Vasani, and S. Sheth, "Velocity Analysis of a DC Brushed Encoder Motor using Ziegler-Nichols Algorithm: A Case of an Automated Guided Vehicle," *Indian Journal of Science and Technology* 9.38 (2016).
- [2] P. Parikh, S. Sheth, and T. Patel, "Positional Analysis of a DC Brushed Encoder Motor Using Ziegler-Nichols Algorithm," *CAD/CAM, Robotics and Factories of the Future*. Springer India, 2016. 637-650.
- [3] S. Maheriya, and P. Parikh, "A Review: Modelling of Brushed DC Motor and Various type of Control Methods," *Journal for Research| Volume 1.12* (2016).
- [4] P. Parikh, N. Modi, and R. Prajapati, "Control of Industrial Pneumatic & Hydraulic Systems using Serial Communication Technology & Matlab."
- [5] P. Parikh, K. Joshi, and S. Sheth, "Color Guided Vehicle—An Intelligent Material Handling Mechatronic System," *Proceedings of the 1st International and 16th National Conference on Machines and Mechanisms (iNaCoMM 2013)*, IIT Roorkee, India. 2013.
- [6] Parikh P., Shah H. and Sheth S, "A Mechatronics design of a line tracker robot using Ziegler Nichols control technique for P, PI and PID controllers," *International Mechanical Engineering Congress (IMEC- 2014)*, June 13-15, 2014. DOI: 10.13140/RG.2.1.4107.4722
- [7] P. Parikh, H. Shah and S. Sheth., "Development of a multi-channel wireless data acquisition System for swarm robots - A Mechatronic Approach using Arduino UNO and MATLAB," *International Journal of Engineering Development and Research (IJEDR)*, ISSN:2321-9939,2 (1), pp. 717-725.
- [8] K. Tamboli, S. Sheth, V. Shah, V. Modi, V. Gandhi, and N. Amin, "Design and Development of a Mechatronic System for the Measurement of Railway Tracks," *Proceeding of the International conference CCEED under IEEE*, pp. 264-269.
- [9] T. Patel, S. Sheth, and P. Patel, "Design of Semiautomatic Hydraulic Blanking Machine Using PLC," *National Conference on Innovative & Emerging Technologies (NCIET- 2015)*, pp. 410-412. DOI: 10.13140/RG.2.1.4529.6803
- [10] Parikh, P., R. Vasani, S. Sheth, and J. Gohil. "Actuation of Electro-Pneumatic System using MATLAB Simulink and Arduino Controller-A case of a Mechatronics systems Lab." (2017).
- [11] Madhusudan, Kanoria Shubham Anil Pandya Jeet, and Priyam Anilkumar Parikh. "Design and Development of Universal Motor Control Unit using MATLAB and Arduino." (2017).
- [12] Biswajit, Priyam Parikh¹ Reenav Shukla² Mansi, and Ghosh³ Nisarg Vivek Kulkarni⁴ Bryan Clifford. "Actuation of AC and DC load using MATLAB GUI and Serial Communication." (2017).
- [13] Nordin Saad, Muawia A. Magzoub, Rosdiazli Ibrahim and Muhammad Irfan, "An optimized fuzzy-fuzzy controller for PWM driven variable speed drives", DOI: 10.5772/61086.
- [14] Parikh, P., Vasani, R. and Pujara, A., A Mechatronics Design of a Wireless Electropneumatic system using Android and MATLAB GUI.
- [15] Parikh, P., Singh, H., & Nadar, S. K. Design and Development of a Programmable Logic Controller Using Atmel Controller and MATLAB Simulink.



این مقاله، از سری مقالات ترجمه شده رایگان سایت ترجمه فا میباشد که با فرمت PDF در اختیار شما عزیزان قرار گرفته است. در صورت تمایل میتوانید با کلیک بر روی دکمه های زیر از سایر مقالات نیز استفاده نمایید:

لیست مقالات ترجمه شده ✓

لیست مقالات ترجمه شده رایگان ✓

لیست جدیدترین مقالات انگلیسی ISI ✓

سایت ترجمه فا ؛ مرجع جدیدترین مقالات ترجمه شده از نشریات معتبر خارجی