

## طراحی و پیاده سازی کنترلر فازی غیر خطی بهره متغیر برای کنترل دور یک موتور DC

رضا گله زاده<sup>۱</sup>، حمید خالوزاده<sup>۲</sup>

گروه برق دانشگاه فردوسی مشهد  
galehzadeh@yahoo.com

### چکیده

در این مقاله یک کنترلر PI فازی با بهره متغیر در حلقه پیشرو طراحی می شود و سپس با استفاده از سخت افزار مناسب جهت کنترل دور یک موتور dc استفاده می شود. کنترلر PI مورد نظر کنترلگری است غیر خطی با بهره متغیر در حلقه پیشرو که در مقایسه با نوع کلاسیک آن انعطاف بیشتر و عملکرد مناسب تری را نشان می دهد. همچنین به منظور افزایش سرعت پردازش محاسبات فازی، معادله نافازی ساز (defuzzification) برای نواحی مختلف بدست آمده و در معادلات مربوط به طراحی کنترلر PI کلاسیک وارد شده است این کار باعث می شود تا ضمن بدست آوردن ضرایب الگوریتم فازی بصورت ساده تر، سرعت پردازش محاسبات نیز افزایش یابد.

واژه‌های کلیدی: کنترلر فازی بهره متغیر (Variable gain fuzzy PI controller)، میکروکنترلر MC8951  
کارت‌های A/D و D/A، مدارهای واسط میکروکنترلر و کامپیوتر

### ۱- مقدمه

کنترلگرهای فازی به سرعت جایگاه خود را در کنترل سیستمها پیدا کرده اند. منطق فازی ابزار قدرتمندی است که به مهندسی این توانایی را می دهد تا استدلالهای انسانی را در الگوهای کنترل دخالت دهند. در طراحی کنترلر فازی، برخلاف کنترلگرهای مدرن و کلاسیک، نیازی به مدل دقیق سیستم برای طراحی نیست. با توجه به اینکه اغلب سیستمهای واقعی غیر خطی اند در نقاط کار مختلف پارمترهای آنها فرق می کند و کارایی روشهای مبتنی بر مدل که بر مبنای تخمینی از مدل بنا می شوند وابسته به نقطه کار شده و خارج از نقطه کاری که بر اساس آن طراحی کنترلر صورت گرفته ممکن است عملکرد مطلوب قبلی را حفظ نکنند.

یک فرایند مدل سازی (خطی و غیر خطی) شامل دو بخش عمده است:

الف - شناسایی ساختار سیستم

ب - شناسایی پارامترها

۱- کارشناسی کنترل، گروه برق دانشگاه فردوسی مشهد

۲- استادیار گروه برق دانشگاه فردوسی مشهد، h\_khaloozadeh@um.ac.ir

در سیستمهای غیر خطی انجام دو بخش فوق کار مشکلی است. مدل سازی فازی روشی قوی و مناسب برای سیستمهای غیر خطی است. روش های مبتنی بر ورودی خروجی مانند جعبه سیاه (black box) فقط می تواند از نظر عددی بین ورودی و خروجی رابطه برقرار کند، اما توانایی مهم فازی در استفاده از دانسته های عددی و دانسته های کیفی است و این خاصیتی بسیار مهم است چون مدل سازی دانسته های کیفی در واقع همان مدل کردن تجربه های انسانی است.

## ۲- طراحی کنترلر های PI فازی

طراحی کنترلر PI فازی به صورتهای مختلف انجام پذیر است، در این تحقیق برای کاهش زمان پردازش، معادله نافازی برای نواحی مختلف حل شده و پارامترهای نظیر آن به صورت صریح در معادله یک کنترلر PI دیجیتال کلاسیک نوشته و وارد شده است. بدین ترتیب با استفاده از ضرایب PI کلاسیک راحت تر می توان ضرایب PI فازی را بدست آورد. چون تعیین این ضرایب یک کار تجربی است و به مهارت طراح بستگی دارد با این روش می توان حدود این ضرایب را راحت تر تعیین کرد.

### ۲-۱- طراحی کنترلر های PI فازی کلاسیک

اولین گام در طراحی یک سیستم کنترل فازی PI، فازی کردن داده های ورودی است. چون محاسبات فازی به صورت گسسته است در نتیجه سیگنال کنترلی به صورت زیر در نظر گرفته می شود:

$$u(n) = k_p e(n) + k_i \sum_{n=0}^N e(n) \quad (1)$$

$u(n)$  را به صورت زیر هم می توان نوشت:

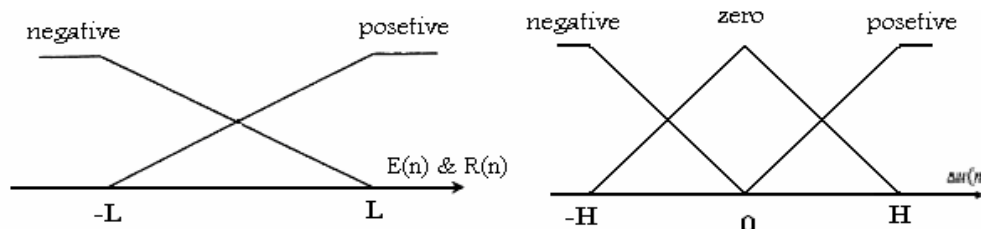
$$u(n) = u(n-1) + \Delta u(n) \quad (2)$$

که  $u(n-1)$  مقدار سیگنال کنترلی در لحظه  $n-1$  است که با  $\Delta u$  جمع شده و  $u(n)$  را تولید می کند.  $\Delta u(n)$  را به صورت زیر بدست می آید:

$$\Delta u(n) = k_p r(n) + k_i e(n) \quad (3)$$

در معادله (۳)، خطا  $e(n)$  و مشتق خطا  $r(n)$  بعنوان ورودی های کنترلر فازی هستند.

تعداد توابع عضویت مجموعه های فازی ورودی دو تایی در نظر گرفته شده است و توابع عضویت برای مجموعه های فازی ورودی و خروجی مثلثی انتخاب شده اند. چون مجموعه های فازی ورودی دوتایی انتخاب شده اند پس ۴ قانون خواهیم داشت برای مجموعه فازی خروجی ۳ تابع عضویت در نظر گرفته شده است. (شکل ۱)



شکل ۱- A مجموعه فازی خروجی ۱- B مجموعه فازی ورودی

قوانین فازی را به صورت زیر در نظر گرفته می‌شود:

If  $e=P$  AND  $r=P$  Then  $\Delta u =P$   
 If  $e=P$  AND  $r=N$  Then  $\Delta u =Z$   
 If  $e=N$  AND  $r=P$  Then  $\Delta u =Z$   
 If  $e=N$  AND  $r=N$  Then  $\Delta u =P$

	E	
R	P	N
	P	Z
	N	Z

برای نافازی کردن نتایج از روش سنترئوید (Centroide) با AND و OR زاده (Zadeh) استفاده می‌شود.

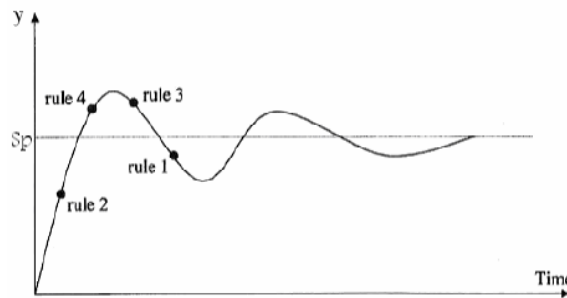
$$\Delta u = \frac{\sum_{k=1}^N \mu_k w_k}{\sum_{k=1}^N w_k} \quad \mu_k = \min(a_i, b_j) \quad (4)$$

در روابط فوق  $a_i$  و  $b_j$  ها به ترتیب توابع عضویت مجموعه های فازی خطا و مشتق خطای ورودی هستند، و  $w_k$  ها مقادیر مرکزی توابع عضویت هر قانون می باشند.

اگر مقادیر  $\mu_k$  و  $w_k$  در فرمول نافازی وارد شوند؛ پس از ساده سازی روابط زیر بدست می آیند:

$$\Delta u = k_{p0} r(n) + k_{i0} e(n) \quad k_{p0} = k_{\Delta u} \frac{k_r H}{2L} \quad k_{i0} = k_{\Delta u} \frac{k_e H}{2L} \quad (5)$$

$H$  اندازه بازه مثبت خروجی و  $L$  اندازه بازه مثبت ورودی است.  $k_e$ ،  $k_r$  ضرایب مقیاس ورودی‌ها و  $k_{\Delta u}$  ضریب مقیاس خروجی سیستم هستند. این ضرایب به منظور قرار دادن ورودی‌ها در بازه  $[-L, L]$  و خروجی در بازه  $[-H, H]$  می‌باشند. اگر نمودار خروجی سیستم را به صورت شکل ۲ برای مجموعه قوانین فازی تقسیم بندی کنیم. مقادیر  $\Delta u$  در نواحی مختلف به صورت جدول ۱ بدست می‌آید. توجه کنید که اگر  $k_e |e(n)| > L$  و  $k_r |r(n)| > L$  باشد  $k_e e(n)$  و  $k_r r(n)$  برابر ۱ هستند.



شکل ۲- تقسیم بندی نمودار خروجی سیستم برای بدست آوردن قوانین فازی

جدول ۱- مقادیر  $\Delta u$  در نواحی مختلف

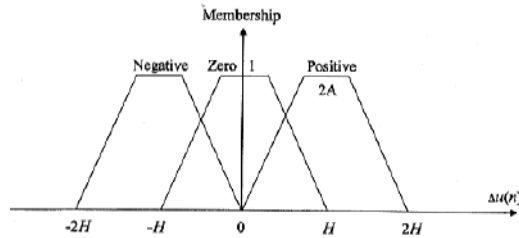
IC No.	$\Delta U(n) =$	Error
1, 2, 3, and 4	$\frac{K_{\Delta u} K_e H}{2L} e(n) + \frac{K_{\Delta u} K_r H}{2L} r(n)$	
5	$\frac{K_{\Delta u} K_r H}{2L} r(n) + \frac{K_{\Delta u} H}{2}$	
6	$\frac{K_{\Delta u} K_e H}{2L} e(n) + \frac{K_{\Delta u} H}{2}$	
7	$\frac{K_{\Delta u} K_r H}{2L} r(n) - \frac{K_{\Delta u} H}{2}$	
8	$\frac{K_{\Delta u} K_e H}{2L} e(n) - \frac{K_{\Delta u} H}{2}$	



حال ضرایب  $k_e$ ،  $k_r$  و  $k_{\Delta u}$  به گونه ای انتخاب می شوند که،  $k_{p0}$  و  $k_{i0}$  بترتیب با  $k_p$  و  $k_i$  طراحی شده برای کنترلر PI کلاسیک برابر باشند.

## ۲-۲ طراحی کنترلر های PI فازی با بهره متغیر

طراحی این کنترلر همانند کنترلر فازی کلاسیک است با این تفاوت که توابع عضویت را برای مجموعه های فازی خروجی، دوزنقه ای انتخاب می کنیم. ( شکل ۳ )



شکل ۳- مجموعه فازی خروجی

قوانین فازی همانند کنترلر قبلی است. برای نافازی کردن نتایج از روش سنترئوید با AND و OR زاده استفاده می شود.

$$\Delta u = \frac{\sum_{k=1}^N \mu_k w_k}{\sum_{k=1}^N w_k} \quad \mu_k = \min(a_i, b_j) \quad w_k = \mu (1 + \theta) H \quad (6)$$

در روابط فوق  $a_i$  و  $b_j$  ها به ترتیب توابع عضویت مجموعه های فازی خطا و مشتق خطای ورودی هستند،  $w_k$  ها مساحت توابع عضویت خروجی به ازای  $\mu_k$  ناشی از قوانین فازی می باشند و  $\theta$  نسبت قاعده بالایی به پایینی هر یک از توابع عضویت خروجی است. با جایگذاری  $\mu_k$  و  $w_k$  ها در فرمول نافازی  $\Delta u$  برای کنترلر PI فازی بهره متغیر به صورت زیر محاسبه می شود:

$$\Delta u = \beta(e, r) \{k_{p0} r(n) + k_{i0} e(n)\}$$

$$k_{p0} = k_{\Delta u} \frac{k_r H}{4L} \quad k_{i0} = k_{\Delta u} \frac{k_e H}{4L}$$

$$\beta(e, r) = \frac{2L}{2L - X} \quad X = \min(k_e |e(n)|, k_r |r(n)|) \quad (7)$$

$k_e$ ،  $k_r$  و  $k_{\Delta u}$  ضرایب مقیاس ورودی ها و خروجی سیستم هستند. مقادیر  $\Delta u$  در نواحی مختلف به صورت جدول ۲ بدست می آید.

جدول ۲- مقادیر  $\Delta u$  در نواحی مختلف

IC No.	$\Delta U(n) =$
1 and 3	$\frac{K_{\Delta u} H}{2(2L - K_e  e(n) )} (K_e e(n) + K_r r(n))$
2 and 4	$\frac{K_{\Delta u} H}{2(2L - K_r  r(n) )} (K_e e(n) + K_r r(n))$
5 to 12	مشابه آنچه در جدول ۱ نشان داده شده است.

با توجه به فرمول  $\beta(e,r)$  می توان نتیجه گرفت که:

$$1 \leq \beta(e,r) \leq 2 \quad (۸)$$

است. یعنی بهره کنترل فازی نسبت به حالت استاتیک آن تا دو برابر می تواند افزایش یابد. حال ضرایب  $k_e$ ،  $k_r$  و  $k_{\Delta u}$  را به گونه ای انتخاب می شوند که،  $k_{p0}$  و  $k_{i0}$  بترتیب با  $k_p$  و  $k_i$  طراحی شده برای کنترلر PI کلاسیک برابر باشند. همانطور که دیده می شود با دوباره پارامتریزه کردن (Reparameterize) کنترلر PI کلاسیک شکل بدست آمده برای کنترلر فازی مستقیما به پارامترهای کنترلر PI کلاسیک وابسته است.

### ۳- پیاده سازی کنترلر های PI فازی کلاسیک و بهره متغیر برای کنترل دور یک موتور DC ۵۰ وات از نوع کنترل آرمیچر

برای اجرای online و پیاده سازی سخت افزاری الگوریتمهای مربوط به کنترلرهای PI فازی کلاسیک و بهره متغیر از سخت افزار و نرم افزاری به شرح ذیل استفاده شده است.

#### ۳-۱- سخت افزار

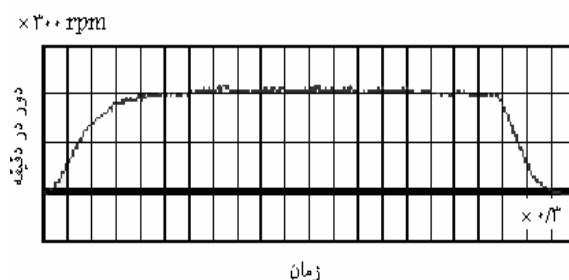
سخت افزار مورد استفاده شامل میکروکنترلر ATM 89C51، A/D0804، D/A0800، مدارات تقویت کننده و راه انداز (Drive) است که با استفاده از تقویت کننده های تفاضلی و درایور ۳ آمپری ساخته شده است. برای نمایش سیگنالهای کنترلی و خروجی سیستم حلقه بسته نیز از یک کامپیوتر استفاده شده است که از طریق پورت سریال با سخت افزار (کارت کنترلر PI) ارتباط دارد.

#### ۳-۲- نرم افزار

نرم افزار طراحی شده با زبان Visual basic تهیه شده است و این توانایی را دارد که به سیستم حلقه باز سیگنالهای ورودی دلخواهی را اعمال کند و خروجی را ثبت نماید. همچنین برای هر بار راه اندازی سیستم با استفاده از یک منو (Menu) یکی از کنترلرهای PI غیر فازی کلاسیک، فازی کلاسیک و فازی با بهره متغیر را به سیستم اعمال کند. همچنین توانایی نشان دادن سیگنالهای کنترلی و خروجی را بشکل همزمان دارد.

#### ۳-۳- طراحی کنترلر PI دیجیتال کلاسیک

همانطور که دیده شد برای تعیین ضرایب PI های فازی کلاسیک و فازی با بهره متغیر باید ضرایب PI کلاسیک غیر فازی آن بدست آید. برای طراحی کنترلر PI لازم است تا تابع انتقال سیستم حلقه باز را مشخص باشد. برای بدست آوردن تابع انتقال سیستم حلقه باز از روش پاسخ پله (Step response) استفاده شده است. برای این کار ابتدا یک ورودی پله به سیستم اعمال می شود. پاسخ سیستم به صورت زیر بدست می آید (شکل ۴).



## شکل ۴- پاسخ پله سیستم

بر این اساس تابع انتقال سیستم به صورت زیر بدست می آید:

$$H(S) = \frac{1}{1+S} \quad (9)$$

برای کنترل سیستم مورد نظر خواسته های پاسخ گذرا و دائم زیر مد نظر است (بعنوان مثال). خطای ماندگار پاسخ پله صفر بوده و حداکثر درصد فراجش  $M_{p.o}$  (Maximum percent overshoot) پاسخ پله سیستم حلقه بسته ۲۵٪ باشد، زمان اوج (Rise time) ۰/۵sec. و زمان نمونه برداری  $T=0/01$  sec. است، در اینصورت ضرایب کنترلر PI به صورت زیر طراحی بست می آیند:

$$M_{po} = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \Rightarrow \zeta = 0.4 \ \& \ t_r = 0.2 \Rightarrow \omega_d = \frac{\pi - \cos^{-1}\zeta}{t_r} = 10$$

$$\omega_n = \frac{\omega_d}{\sqrt{1-\zeta^2}} = 11 \Rightarrow S_{1,2} = -\zeta\omega_n \pm j\omega_d \quad (10)$$

$$\Rightarrow Z_{1,2} = e^{-\zeta\omega_n T} \cdot e^{\pm j\omega_d T} = 0.995 * e^{\pm j.01} = 0.994 \pm j0.09$$

از طرف دیگر داریم:

$$\left. \begin{aligned} G(s) = \frac{1}{1+S} \Rightarrow G(z) = (1-Z^{-1}) * Z \left\{ \frac{G(s)}{S} \right\} &= \frac{1-e^{-T}}{Z-e^{-T}} \\ H(z) = K_p + K_i \frac{Z}{Z-1} &= \frac{(K_p + K_i)Z - K_p}{Z-1} \\ G(z)H(z) &= \frac{(1-e^{-T})\{(K_p + K_i)Z - K_p\}}{(Z-1)(Z-e^{-T})} \end{aligned} \right\} \Rightarrow \quad (11)$$

با توجه به شرط زاویه و اندازه داریم:

$$\tan^{-1} \frac{0.09}{0.994-1} + \tan^{-1} \frac{0.09}{0.994-e^{-T}} - 180 = \tan^{-1} \frac{0.09(K_p + K_i)}{0.994(K_p + K_i) - K_p} \quad (12)$$

$$(1-e^{-T}) \frac{|(K_p + K_i)Z - K_p|}{|(Z-1)(Z-e^{-T})|} \Big|_{Z=0.994 \pm j0.09} = 1$$

به ازای  $T=0/01$ ،  $k_p$  و  $k_i$  به صورت زیر بدست می آید:

$$k_i = 0.3 \quad k_p = 0.5$$

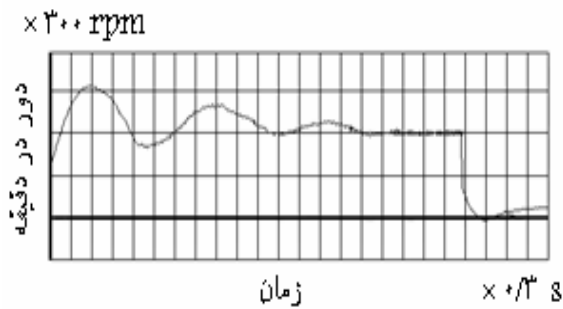
## ۴-۳- اعمال PI های فازی طراحی شده به سیستم

در بدست آوردن ضرایب کنترلگرهای فازی باید این مطلب را مد نظر داشت که بدست آوردن ضرایب  $k_r$ ،  $k_e$  و  $k_{\Delta u}$  یک کار تجربی است. اینکه معادله فازی به صورت یک معادله PI کلاسیک غیر فازی در آمد، به خاطر این است که یک تقریب بهتری از این ضرایب بدست آید و هم به صورت خودکار تعیین شود و بقیه تغییرات تا بدست آمدن جواب مطلوب باید به صورت سعی و خطا انجام شود ( $H, L$  را برای هر دو طراحی برابر ۱ در نظر می گیریم).

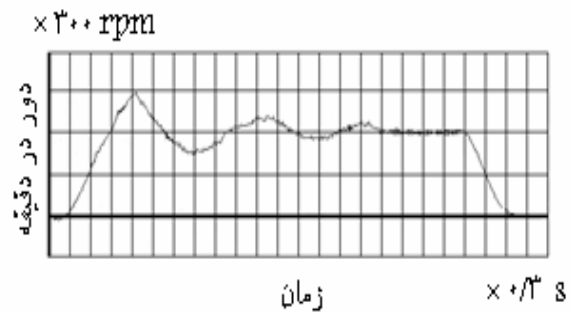
PI طراحی شده در بخش ۲-۱ به سیستم اعمال می شود. اگر  $k_e = 1/0.5$ ،  $k_r = 0/0.7$  انتخاب گردد پاسخ سیستم به صورت

شکل ۵ و سیگنال کنترلی اعمال شده به سیستم توسط کنترلر به صورت شکل ۶ است..

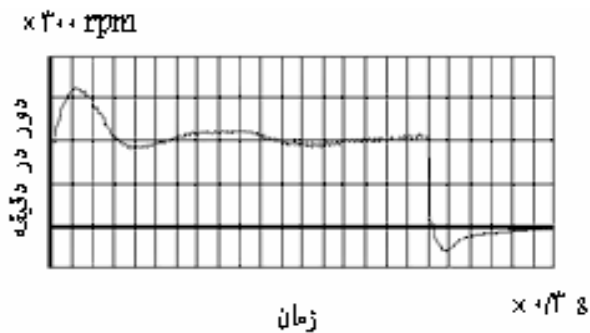
این بار PI فازی بهره متغیر طراحی شده در در بخش ۲-۲ به سیستم اعمال می شود. با انتخاب  $k_e = 1/0.2$ ,  $k_f = 0.1$  و  $\theta = 0.5$  پاسخ سیستم به صورت شکل ۷ و سیگنال کنترلی اعمال شده به سیستم توسط کنترلر به صورت شکل ۸ است.



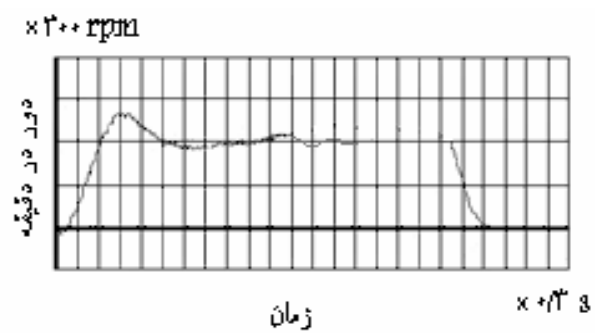
شکل ۶ - سیگنال کنترلی اعمال شده به سیستم توسط PI فازی کلاسیک



شکل ۵- پاسخ سیستم در نتیجه اعمال PI فازی کلاسیک

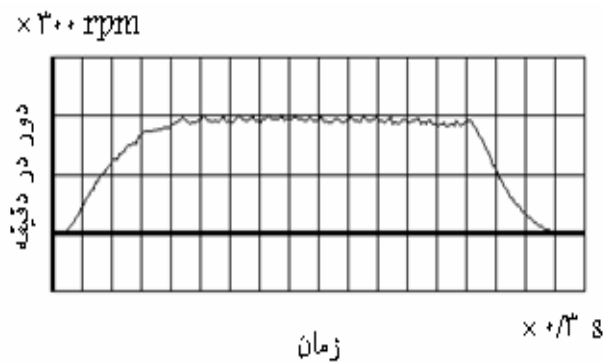


شکل ۸- سیگنال کنترلی اعمال شده به سیستم توسط PI فازی بهره متغیر



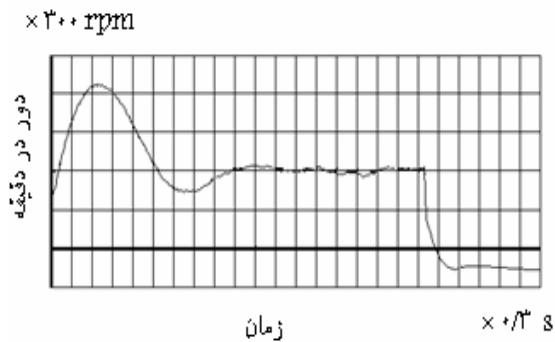
شکل ۷- پاسخ سیستم در نتیجه اعمال PI فازی بهره متغیر

برای اینکه میزان مقاوم بودن (Robustness) سیستم حلقه بسته در مقابل تغییرات سیستم آزمایش شود و عملکرد کنترلرهای طراحی شده مقایسه گردد، با اضافه کردن و اعمال بار به سیستم، سیستم را اندکی لخت تر کرده (شکل ۹) دوباره همان PI های طراحی شده در مرحله قبل را به سیستم اعمال می کنیم؛ پاسخهای سیستم و نیز سیگنال کنترلی متناظر با اعمال کنترلر PI فازی کلاسیک به صورت شکل‌های ۱۰ و ۱۱ است.

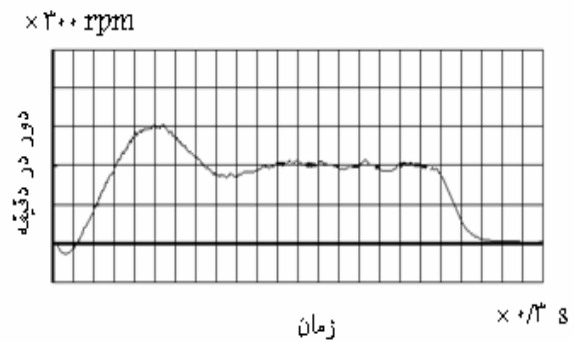


شکل ۹- پاسخ پله سیستم پس از اعمال بار



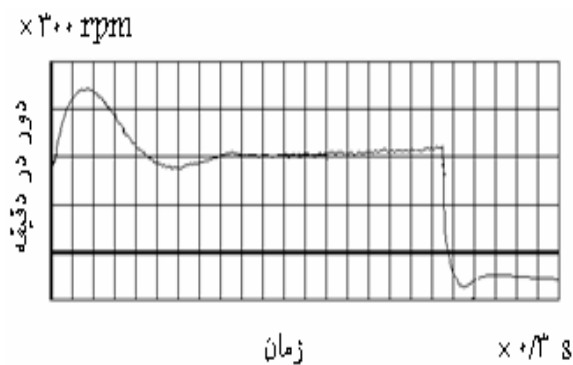


شکل ۱۱- سیگنال کنترلی اعمال شده به سیستم توسط PI فازی کلاسیک

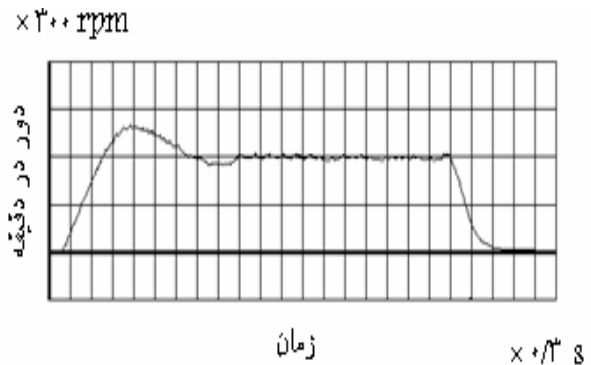


شکل ۱۰- پاسخ سیستم در نتیجه اعمال PI فازی کلاسیک

شکل‌های ۱۲ و ۱۳ نیز نشانگر پاسخ سیستم حلقه بسته و سیگنال کنترل با استفاده از کنترلر فازی بهره متغیر است.



شکل ۱۳- سیگنال کنترلی اعمال شده به سیستم توسط PI فازی بهره متغیر



شکل ۱۲- پاسخ سیستم در نتیجه اعمال PI فازی بهره متغیر

#### ۴- نتیجه گیری

همانطور که از مقایسه خروجی های سیستم در هر دو حالت بر می آید هنگامی که از کنترلر فازی بهره متغیر استفاده می کنیم سرعت سیستم افزایش می یابد ضمن اینکه درصد فراجهدش پاسخ خروجی نیز کمتر می شود با توجه به ساده سازی که در محاسبات فازی صورت گرفت سرعت پردازش افزایش یافته و پیاده سازی سخت افزاری آن را راحتتر کرده است.

#### مراجع

- [۱] کنترل کننده PID خود تنظیم فازی، پایان نامه کارشناسی ارشد، بهرام وکیلی، حسین احمدی، بهزاد مشیری، محمد رضا جاهد مطلق، دانشکده تهران - دانشکده مهندسی برق و کامپیوتر اردیبهشت ۱۳۷۴
- [۲] سیستمهای فازی و کنترل فازی - وانگ لی شین - ترجمه محمد تشنه لب، دانشگاه خواجه نصیرالدین طوسی ۱۳۸۰

[۳] مقدمه ای بر منطق فازی برای کاربردهای علمی - تاناکا کازوئو - ترجمه علی وحیدیان، حامد رضا طارقیان، انتشارات دانشگاه فردوسی ۱۳۸۱

[4] Ying, Hao, Fuzzy control and modeling, Analytical foundations and application 1985