



Computational Intelligence in Electrical Engineering
Vol. 11, No. 4, 2021
Research Paper

Adaptive (PID) controller design using simultaneous perturbation stochastic approximation algorithm and neural network training

Majed Anjomshoaa¹, Malihe Maghfoori Farsangi², Yasin Asadi³, Mohammad Molaei⁴

¹ Faculty of Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

² Faculty of Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

³ Faculty of Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

⁴ Faculty of Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

Abstract:

In this paper, a new method of data-driven controller (DDC) design using Simultaneous Perturbation Stochastic Approximation Algorithm (SPSA) and Neural Network (NN) training is presented. This method can be used to control a variety of linear and nonlinear systems. In the simultaneous perturbation stochastic approximation algorithm, the controller is assumed to have a fixed structure and its parameters must be estimated. In this paper, a Proportional, Integral, and Derivative controller (PID) is considered and the parameters that should be estimated by the proposed algorithm are proportional, integral and derivative terms of this controller. In the proposed method, the simultaneous perturbation stochastic approximation algorithm is quantified by using neural network training which increases the convergence speed and also improves the performance of the algorithm against system input changes. Simulations performed on cement grinding particle size distribution process and pitch angle control of aircraft show the high efficiency and potential of the proposed method.

Keywords: Data-driven controller, Simultaneous perturbation stochastic approximation, (PID) controller, Online tuning, Perceptron neural network.

طراحی کنترل کننده PID تطبیقی با استفاده از الگوریتم تقریبات تصادفی انحرافات هم‌زمان و آموزش شبکه عصبی

ماجد انجم شعاع^۱، ملیحه مغفوری فرسنگی^{۲*}، یاسین اسدی^۳، محمد ملایی امامزاده^۴

۱- دانشجوی کارشناسی ارشد، دانشکده فنی و مهندسی - دانشگاه شهید باهنر کرمان - کرمان - ایران

majed_anjomshoaa@eng.uk.ac.ir

۲- استاد، دانشکده فنی و مهندسی - دانشگاه شهید باهنر کرمان - کرمان - ایران

mmaghfoori@uk.ac.ir

۳- دانشجوی دکتری، دانشکده فنی و مهندسی - دانشگاه شهید باهنر کرمان - کرمان - ایران

yasinasadi@eng.uk.ac.ir

۴- استادیار، دانشکده فنی و مهندسی - دانشگاه شهید باهنر کرمان - کرمان - ایران

molaei@uk.ac.ir

چکیده: در این مقاله، روش جدید طراحی کنترل کننده داده محور^۱، با استفاده از الگوریتم تقریبات تصادفی انحرافات هم‌زمان^۲ (SPSA) و آموزش شبکه عصبی ارائه شده است. در روش پیشنهادی، الگوریتم تقریبات تصادفی انحرافات هم‌زمان با استفاده از آموزش شبکه عصبی، مقداری می‌شود که این امر باعث افزایش سرعت همگرایی و همچنین بهبود عملکرد الگوریتم در برابر تغییرات سیگنال مرجع می‌شود. در SPSA فرض بر این است که کنترل کننده دارای ساختاری ثابت است. پارامترهای این کنترل کننده به صورت برخط تخمین زده می‌شوند. در این مقاله، کنترل کننده به کاررفته، کنترل کننده تناسبی، انتگرالی و مشتق گیر (PID) است. شبیه سازی‌های انجام شده روی پروسه توزیع اندازه ذرات سنگ زنی سیمان و کنترل زاویه پیچ هواپیما نشان دهنده مؤثر بودن روش پیشنهادی در بهبود عملکرد سیستم است.

واژه‌های کلیدی: کنترل کننده داده محور، تقریبات تصادفی انحرافات هم‌زمان، PID کنترلر، کنترل کننده برخط، شبکه عصبی

پرسپترون.

۱- مقدمه

تولید، ارائه مدل برای پروسه‌های صنعتی بسیار دشوار و غیرممکن شده است؛ در نتیجه، کنترل کننده‌های مبتنی بر مدل^۳ در برابر کنترل چنین پروسه‌هایی با مشکلاتی همچون دردسترس نبودن مدل، دینامیک‌های مدل نشده و غیره روبرو هستند. نیز با پیشرفت تکنولوژی اطلاعات، در بسیاری از پروسه‌های صنعتی، در هر لحظه حجم زیادی از اطلاعات سیستم مانند حالات سیستم، سیگنال‌های کنترلی و سیگنال‌های خروجی و ... اندازه‌گیری و ذخیره می‌شوند. از این اطلاعات به صورت برخط (on-line) و برون خط (off-

با توسعه علم و تکنولوژی و پیچیده شدن فرایندهای

^۱ تاریخ ارسال مقاله: ۱۳۹۸/۰۶/۰۷

تاریخ پذیرش مقاله: ۱۳۹۹/۰۳/۰۶

نام نویسنده مسئول: ملیحه مغفوری فرسنگی

نشانی نویسنده مسئول: ایران - کرمان - دانشگاه شهید باهنر

کرمان - بخش برق

تنها برای حل بهینه‌یابی محل حفر چاه ارائه شده است و باید آن را برای استفاده در دیگر مسائل مهندسی تغییر داد. همچنین در مرجع [۱۹] یک کنترل کننده PID جدید معرفی می‌شود که از الگوریتم SPSA همراه با حافظه جانبی استفاده می‌کند. این الگوریتم (M-SPSA) مبتنی بر حافظه، توانایی به دست آوردن دقت بهینه‌سازی بهتری نسبت به SPSA معمولی دارد؛ زیرا پارامترهای کنترل کننده را بهتر تنظیم می‌کند. یکی از نقاط ضعف این مقاله توجه نکردن به سرعت همگرایی است. همچنین، با توجه به کارآمدی الگوریتم SPSA در کنترل سیستم‌های پیچیده، از آن برای کنترل سیستم‌های صنعتی بسیاری استفاده شده است؛ به طور مثال، نویسندگان در مرجع [۲۰] این الگوریتم را برای حل مسائل مدیریت انرژی در ماشین‌های هیبریدی پیشنهاد داده‌اند.

نیز با توجه به پیشرفت‌های انجام شده در زمینه هوش محاسباتی در چند دهه اخیر، استفاده از سیستم‌های هوشمند و به ویژه شبکه عصبی مصنوعی^۹ بسیار گسترده شده است؛ به طوری که این ابزارها در ردیف عملیات پایه ریاضی و به عنوان ابزارهای عمومی و مشترک طبقه بندی می‌شوند [۲۱]. یکی از پایه‌ای ترین مدل‌های عصبی موجود، مدل پرسپترون چندلایه^{۱۰} [۲۲-۲۴] (MLP) است که عملکرد انتقالی مغز انسان را شبیه سازی می‌کند.

در این مقاله، به منظور بهبود سرعت همگرایی الگوریتم SPSA، استفاده از شبکه عصبی پیشنهاد می‌شود. در این روش با کمک آموزش شبکه عصبی و تلفیق آن با SPSA، کنترل کننده PID تطبیقی جدیدی پیشنهاد شده است که با سرعت بسیار مناسبی پارامترهای کنترلی را تخمین می‌زند و حتی توانایی آن را دارد پاسخگوی تغییرات ناگهانی سیگنال مرجع باشد. دست‌آوردهای اصلی این مقاله به شرح زیرند:

۱- الگوریتم SPSA از طریق تلفیق با شبکه عصبی توسعه یافته است.

۲- سرعت همگرایی SPSA بهبود یافته است.

۳- کنترل کننده PID تطبیقی جدیدی پیشنهاد شده است که ضرایب آن با الگوریتم SPSA به طور آنلاین به دست می‌آید.

در ادامه، در بخش ۲ اشاره‌ای مختصر به الگوریتم

(line) به منظور پیش‌بینی حالت سیستم، ارزیابی عملکرد سیستم، تصمیم‌گیری، طراحی کنترل کننده و غیره استفاده می‌شود.

در چند سال اخیر، با توجه به کارآمدی کنترل کننده‌های داده‌محور برای سیستم‌های غیرخطی و پیچیده، روش‌های کنترلی داده‌محور برخط و برون‌خط بسیاری توسعه داده شده‌اند [۴-۱]. از جمله این روش‌ها عبارت‌اند از: الگوریتم تقریبات تصادفی انحرافات هم‌زمان، کنترل تطبیقی بدون مدل^۴ [۵-۷]، کنترل ابطال‌ناپذیر^۵ [۸]، تنظیم بر پایه همبستگی^۶ [۹]، تنظیم بازخوردی مرجع مجازی^۷ [۱۰، ۱۱]، تنظیم بازخوردی بازگشتی^۸ [۱۲].

در این مقاله، از میان روش‌های بالا، الگوریتم SPSA که روشی برخط است، برای مطالعه و بررسی در نظر گرفته شده است. Spall این الگوریتم را نخستین بار در سال 1993 پیشنهاد داد [۱۳]. در الگوریتم SPSA بدون استفاده از مدل سیستم، تنها با استفاده از اطلاعات ورودی، خروجی و استفاده از چند تابع، هزینه پارامترهای کنترل کننده به صورت برخط محاسبه می‌شوند [۱۴]. الگوریتم SPSA روشی بازگشتی و تکرارشونده است؛ بنابراین، پژوهش‌های صورت گرفته بر این روش، تا کنون سعی داشته‌اند سرعت و کیفیت همگرایی این الگوریتم را افزایش دهند؛ برای مثال، مرجع [۱۵] با تمرکز بر ضرایب SPSA و گرادینان خطا سعی دارد الگوریتم را از جنبه‌های متفاوتی همچون کیفیت پاسخ و سرعت همگرایی بهبود دهد. در مرجع [۱۶] با کنترل کردن طول گام‌های این الگوریتم، مشکل همگرانشدن برخی از مسائل، بررسی و شرایط لازم برای همگرایی الگوریتم پیشنهادی ارائه می‌شود. مرجع [۱۷] با کمک جمع‌آوری اطلاعات زمانی و مکانی، الگوریتمی را پیشنهاد می‌دهد که این الگوریتم توانایی محدود کردن تأثیر نویز و بالابردن سرعت همگرایی را داراست. در مرجع [۱۸] یک مدل بهبود یافته از الگوریتم SPSA برای تعیین محل حفر چاه در میدین نفتی با هدف افزایش سوددهی ارائه شده است. در این مقاله با توجه به پیچیدگی بهینه‌سازی و وجود متغیرهای بسیار زیاد، الگوریتم SPSA راهکار کارآمد برای دستیابی به جواب بهینه معرفی شده است. عیب اساسی روش پیشنهادی در این مقاله این است که الگوریتم کنترلی

آن از رابطه (۵) استفاده می‌شود:

$$b_i = \frac{b}{(A+i+1)^\alpha} \quad (5)$$

که در آن b, A, α مقادیری ثابت‌اند.

همچنین، برای به دست آوردن \hat{H}_i از رابطه (۶) استفاده می‌شود که تقریبی از گرادیان بالا است:

$$\hat{H}_i(\theta_i) = \frac{\hat{J}_i^{(+)} - \hat{J}_i^{(-)}}{2f_i\Delta_i} \quad (6)$$

این تابع برآورد پراکندگی نامیده می‌شود. در رابطه بالا f_i ضریبی مثبت است و برای به دست آوردن آن از رابطه (۷) استفاده می‌شود:

$$f_i = \frac{f}{(i+1)^r} \quad (7)$$

در این رابطه f و r مقادیری ثابت‌اند.

برای به دست آوردن $\hat{J}_i^{(\pm)}$ باید ابتدا قانون کنترلی u_i و سپس y_{i+1} را محاسبه کرد. برای رسیدن به این هدف باید در ابتدا انحرافات Δ_i را به θ_i طبق رابطه $\hat{u}_i^{(\pm)} = \theta_i^{(\pm)} \pm f_i \Delta_i$ اعمال کرد و بعد با در دست داشتن $\hat{u}_i^{(\pm)}$ مقادیر y_{i+1} با اعمال کردن u_i به سیستم مدنظر به دست می‌آید و با در دست داشتن y_{i+1} توابع هزینه $\hat{J}_i^{(\pm)}$ محاسبه می‌شوند. Δ_i بردار انحرافات است که دارای توزیع برنولی با احتمال ۰.۵ برای دو عدد ± 1 است.

درخور ذکر است به دلیل محدودیت در صفحات مقاله، بحث پایداری و شروط لازم برای همگرایی الگوریتم SPSA در مرجع [۲۵] مشاهده می‌شود.

نکته ۱: شایان ذکر است الگوریتم SPSA تنها با در اختیار داشتن پاسخ خروجی، توانایی تخمین زدن پارامترهای کنترل‌کننده PID را داراست؛ بنابراین، این روش برای سیستم‌های خطی و غیرخطی کاربردی است.

۲-۲- شبکه عصبی پرسپترون

شبکه عصبی مصنوعی روشی ابداعی برای حل مسائل

SPSA و شبکه عصبی پرسپترون می‌شود. در بخش ۳ الگوریتم پیشنهادی معرفی می‌شود. در بخش ۴ روش پیشنهادی روی دو مسئله ارزیابی می‌شود و در بخش ۵ نتیجه‌گیری می‌شود.

۲- ساختار الگوریتم SPSA و شبکه عصبی

پرسپترون

۱-۲- ساختار الگوریتم SPSA

سیستم گسسته زمان (۱) را در نظر بگیرید.

$$\begin{aligned} x_{i+1} &= \varphi_i(x_i, u_i, w_i) \\ y_i &= h_i(x_i, v_i) \end{aligned} \quad (1)$$

هدف اصلی از به کارگیری الگوریتم SPSA به دست آوردن پارامترهای برداری مانند θ است. با توجه به اینکه در این مقاله هدف اصلی به دست آوردن پارامترهای کنترلی PID است، بردار θ شامل سه پارامتر k_p, k_i, k_d است. برای به دست آوردن این پارامترها به یک تابع هزینه نیاز است که به فرم رابطه (۲) در نظر گرفته شده است.

$$J_i(\theta_i) = ((y(\theta_i, i+1) - y_d(i+1))^2) \quad (2)$$

همان‌گونه که مشخص است تابع هزینه از خطای مطلق حاصل شده است. حال برای رسیدن به مقدار مطلوب تابع هزینه (۲) باید از رابطه (۳) استفاده کرد.

$$\begin{aligned} H_i(\theta_i) &= \frac{\partial J_i}{\partial \theta_i} = \\ &= 2(y(\theta_i, i+1) - y_d(i+1)) \frac{\partial h_{i+1}}{\partial x_{i+1}} \frac{\partial \varphi_i}{\partial u_i} \frac{\partial u_i}{\partial \theta_i} \end{aligned} \quad (3)$$

رابطه (۳) نشان می‌دهد برای دستیابی به پاسخ، مدل سیستم باید در دسترس باشد؛ اما یکی از فرض‌های اساسی در کنترل‌کننده‌های داده‌محور آن است که مدل سیستم در دسترس نیست؛ بنابراین، برای به دست آوردن مقادیر پارامترهای مطلوب از رابطه بازگشتی (۴) استفاده می‌شود.

$$\hat{\theta}_{i+1} = \hat{\theta}_i - b_i \hat{H}_i(\theta_i) \quad (4)$$

در رابطه (۴)، $\hat{\theta}_i$ نشان‌دهنده مقادیر تقریب زده شده θ_i است. همچنین b_i ضریب بهبود است که برای محاسبه

۳- الگوریتم پیشنهادی

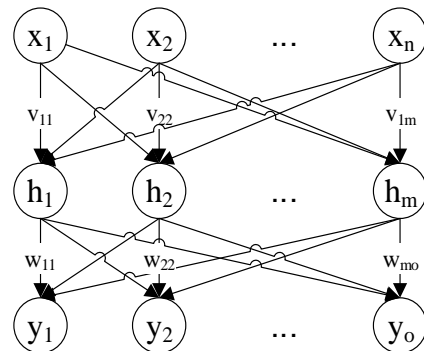
الگوریتم SPSA بیان شده در قسمت دوم، مشکلاتی دارد؛ یکی از مهم‌ترین این مشکلات، بالابودن زمان محاسبات است.

یکی از عوامل مؤثر در سرعت بخشیدن به الگوریتم، مقادیر تخمین زده شده هر تکرار است که به‌عنوان شرایط اولیه در تکرار بعدی برای تخمین مقدار مطلوب استفاده می‌شوند. این امر به‌وضوح در رابطه بازگشتی (۴) مشاهده می‌شود. به بیان ساده‌تر، در این الگوریتم چون از بهینه‌سازی استفاده می‌شود، هرچه مقدار تقریب زده شده هر تکرار از شرایط مطلوب‌تری برخوردار باشد، در تکرار بعد سریع‌تر جواب مطلوب به دست می‌آید؛ بنابراین، در هر تکرار، مقادیر اولیه نقش بسزایی در سرعت همگرایی الگوریتم دارند.

در این مقاله، با آموزش دادن شبکه عصبی پرسپترون، مقادیر داده‌های مجهول در برخی دوره‌های خاص (برای مثال، به‌صورت بازه‌ای یا وقوع تغییرات سیگنال مرجع) در اختیار الگوریتم SPSA قرار می‌گیرد. فاصله زمانی این دوره‌های خاص با توجه به میزان سرعت تغییرات سیگنال مرجع و سیستم در کنترل تعیین می‌شود. به عبارتی، هنگامی که یک سیستم تغییرات شدید ندارد و سیگنال SPSA هم همگرا شده است، استفاده از خروجی شبکه عصبی به‌صورت دائم در هر تکرار نیاز نیست؛ بنابراین، اگر تغییراتی ایجاد شود، مانند تغییر سیگنال مرجع، آنگاه نیاز است از خروجی شبکه عصبی برای مقاردهی به الگوریتم SPSA استفاده شود. برای رسیدن به این منظور باید از تعدادی داده برای آموزش استفاده کرد. تعیین این داده‌ها بسته به نوع پلنت کنترلی، متفاوت است. می‌توان آموزش را به کمک تعدادی داده آغاز کرد که آنها داده‌های با اهمیت نام‌گذاری می‌شوند. این داده‌ها یا به‌صورت برون‌خط به‌عنوان پایگاه داده در دسترس است یا براساس تجربه به‌زای چند ورودی متفاوت چند پاسخ مناسب در دسترس است. در گام بعد، پس از آموزش اولیه در دوره‌هایی که مطلوب کاربر است، با اجرا شدن الگوریتم داده‌های آموزشی به‌روز می‌شوند. این بدان معنا است که الگوریتم، فرایند

محاسباتی پیچیده است که بر پایه اتصال به‌هم‌پیوسته چندین واحد پردازشی شکل می‌گیرد. شبکه از تعداد مشخصی سلول، گره یا نرون تشکیل می‌شود که مجموعه ورودی را به خروجی ربط می‌دهد. هر یک از نرون‌ها دارای یک تابع فعالیت است که بسته به نوع مسئله، متفاوت است. تابع فعالیت وظیفه دارد با دریافت مجموعه سیگنال‌های وارد شده به نرون به‌عنوان سیگنال ورودی، سیگنال خروجی را از نرون فراهم کند. برای بهبود محاسبات بین لایه ورودی و خروجی از چند لایه پنهان استفاده می‌شود. شبکه عصبی مصنوعی روشی عملی برای یادگیری توابع گوناگون نظیر توابع با مقادیر حقیقی، توابع با مقادیر گسسته و توابع با مقادیر برداری است. یادگیری شبکه عصبی در برابر خطاهای داده‌های آموزشی مصون بوده و با موفقیت به مسائلی نظیر شناسایی گفتار، شناسایی و تعبیر تصاویر، یادگیری ربات و غیره پاسخگو است [۲۶].

یکی از انواع شبکه‌های عصبی، شبکه عصبی پرسپترون است؛ این شبکه‌ها ساختاری رو به جلو^{۱۱} دارند که از قانون دلتای عمومی^{۱۲} یا بازگرداندن خطا^{۱۳} (انتشار خطا) برای یادگیری استفاده می‌کنند. یک روش، استفاده از گرادینان کاهش^{۱۴} است که مجموع مربعات خطای خروجی شبکه را مینیمم کند. یادگیری این شبکه شامل سه مرحله است. در ابتدا باید ورودی‌ها را به شبکه اعمال کرد و لایه به‌لایه محاسبات را انجام داد تا به خروجی رسید. سپس باید خطا را در خروجی محاسبه کرد و با استفاده از روش بازگرداندن خطا به لایه‌هایی قبل بازگشت و در گام آخر وزن‌ها را تنظیم کرد [۲۷]. شکل (۱) ساختار شبکه عصبی پرسپترون را به همراه یک لایه مخفی نشان می‌دهد.

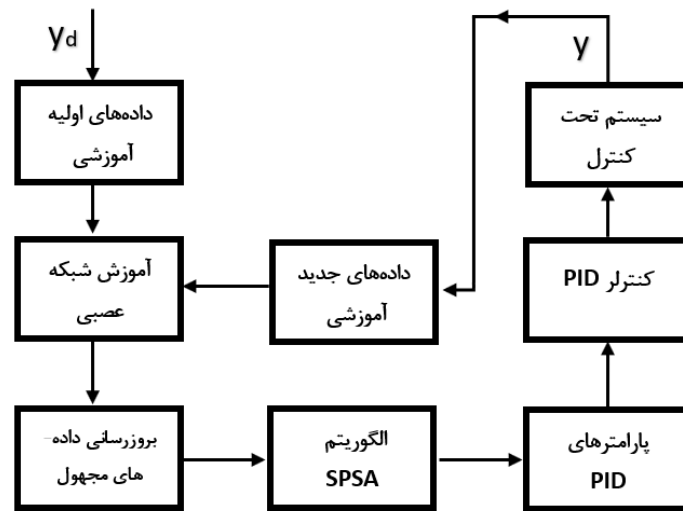


شکل (۱): ساختار شبکه عصبی چندلایه پرسپترون با یک لایه

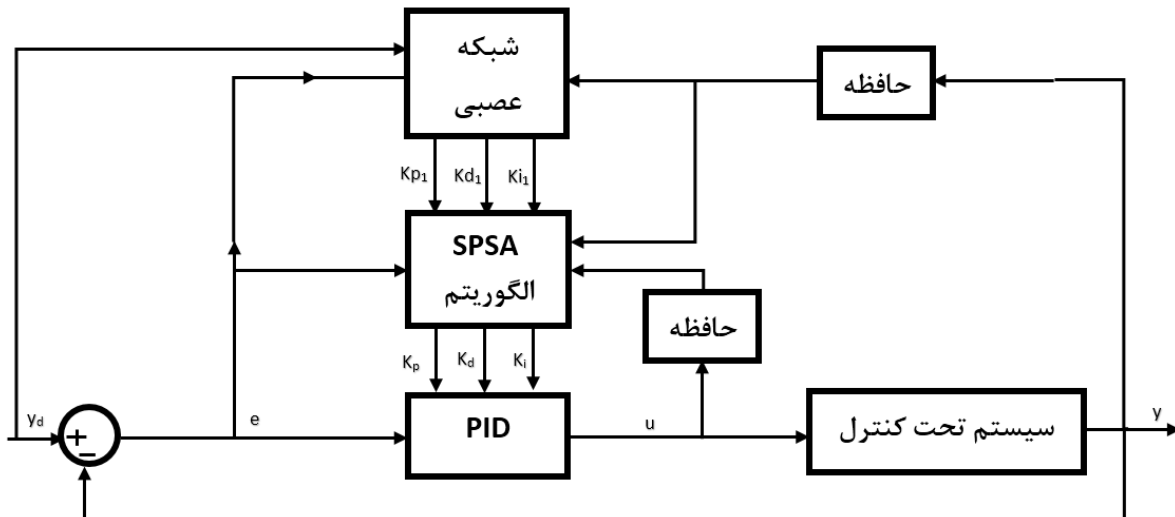
مخفی

داده‌های آموزشی شبکه عصبی می‌شود. فلوجارت الگوریتم پیشنهادی در شکل (۲) و ساختار کلی کنترل‌کننده در شکل (۳) مشاهده می‌شود.

آموزش را به صورت برخط انجام می‌دهد. به منظور آموزش بهتر الگوریتم SPSA بهتر است علاوه بر استفاده از داده‌های آنلین، از داده‌های بااهمیت نیز استفاده شود؛ بنابراین، این عمل باعث بالارفتن کیفیت



شکل (۲): فلوجارت الگوریتم پیشنهادی.



شکل (۳): ساختار کلی طراحی کنترل‌کننده (PID) به کمک الگوریتم پیشنهادی.

داده‌های اولیه موجود آموزش داده می‌شود که روند به دست آوردن این داده‌ها قبلاً توضیح داده شده است. با کمک شبکه عصبی شرایط اولیه برای کنترل‌کننده (PID) مشخص می‌شود. در ادامه بعد از شروع به کار فرایند، همواره داده‌هایی همچون مقدار مطلوب ورودی، مقدار خروجی و سیگنال خطا در دسترس‌اند. از این ۳ داده

با توجه به شکل (۳)، روند طراحی به صورت زیر بیان می‌شود. در ابتدا باید داده‌های اولیه مشخص شوند. این داده‌ها عبارت‌اند از α, γ, f, b که برای محاسبات الگوریتم تقریبات تصادفی انحرافات هم‌زمان استفاده می‌شوند. انتخاب این داده‌ها بسته به مسئله، متفاوت است. سپس شبکه عصبی با

داده‌های اولیه مناسب برای الگوریتم SPSA است، این امر باعث تسریع الگوریتم برای رسیدن به مقدار مطلوب در تعداد گام کمتر می‌شود.

۳-۱- آنالیز همگرایی الگوریتم پیشنهادی

همان‌گونه که بیان شد یکی از عوامل مؤثر در سرعت بخشیدن به الگوریتم، مقادیر تخمین زده شده هر تکرار است که به‌عنوان شرایط اولیه در تکرار بعدی برای تخمین مقدار مطلوب استفاده می‌شود. تفاوت اساسی الگوریتم پیشنهادی با الگوریتم SPSA در این است که در روش پیشنهادی، به‌منظور دستیابی به سرعت همگرایی بیشتر، شرایط اولیه الگوریتم SPSA به‌صورت هدفمند و براساس خروجی شبکه عصبی مقداردهی می‌شود؛ بنابراین، شرایط همگرایی الگوریتم SPSA تحت تأثیر قرار نمی‌گیرد و تمامی شرایط همگرایی بیان‌شده در مقاله [۲۵] برای روش پیشنهادی برقرار است.

همچنین اگر سیستم با تغییراتی مواجه شود، شبکه عصبی، مقادیر اولیه‌ای را در اختیار الگوریتم SPSA قرار می‌دهد که با توجه به آموزش صورت‌گرفته، بسیار نزدیک به خروجی مطلوب الگوریتم SPSA است؛ بنابراین، الگوریتم SPSA در تعداد گام کمتر یا به عبارتی در زمان کمتر، به مقدار نهایی خود همگرا می‌شود.

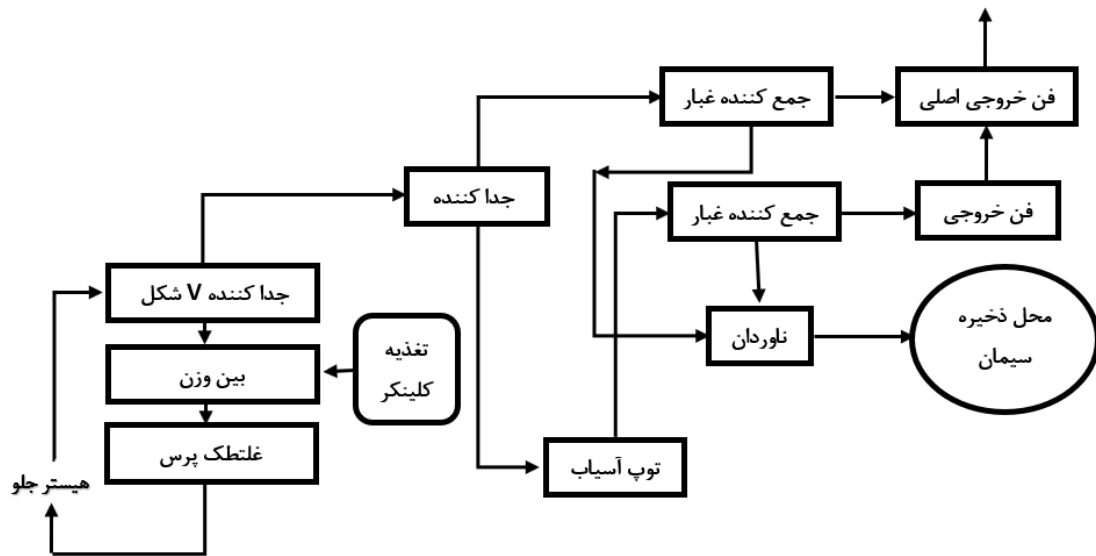
۴- شبیه‌سازی

۴-۱- توزیع اندازه ذرات سنگ‌زنی سیمان

پروسه توزیع اندازه ذرات سنگ‌زنی سیمان، یکی از پروسه‌های صنعتی است که طراحی کنترل‌کننده برای آن دشوار است [۲۸]. شبیه‌سازی زیر کارایی الگوریتم پیشنهادی را در کنترل پروسه سنگ‌زنی سیمان نشان می‌دهد. به‌طور خلاصه، روند فرآیند سنگ‌زنی سیمان در شکل (۴) مشاهده می‌شود.

می‌توان به‌عنوان داده‌های آموزشی ورودی شبکه و از پارامترهای کنترلی تولیدشده با الگوریتم تقریبات تصادفی انحرافات هم‌زمان به‌عنوان داده خروجی شبکه برای آموزش استفاده کرد. حال بسته به نیاز کاربر، در دوره‌های معلوم‌شده، داده‌های مورد نیاز الگوریتم تقریبات تصادفی انحرافات هم‌زمان (مشخص‌شده در شکل (۳)) k_{pi}, k_{ii}, k_{di} از شبکه دریافت می‌شود.

فرایند یادشده باعث بالا رفتن دقت داده‌های آموزشی و افزایش تقاوم الگوریتم در برابر بروز تغییرات ناگهانی در سیستم می‌شود. همان‌گونه که اشاره شد در آموزش با نظارت، اساسی‌ترین مسئله در فرایند آموزش، کیفیت داده‌های آموزشی است. این امر بدان معناست که هرچه داده‌های آموزشی از کیفیت بالاتری برخوردار باشند، شبکه به داده‌های ورودی، بهتر پاسخ می‌دهد و این امر با اجرا شدن الگوریتم و تکرارهای پیاپی بهتر می‌شود؛ اما نکته شایان توجه آنکه در سیستم‌های کنترلی واقعی امکان بروز تغییرات ناگهانی در سیگنال مرجع سیستم‌ها وجود دارد و اگر فرایند کنترل به‌صورت برخط صورت گیرد، باعث پدید آمدن مشکلاتی در سیستم می‌شود؛ برای مثال، فرض کنید هدف فرایند کنترل دستیابی به پاسخ مطلوب پله واحد باشد. حال اگر در صورت بروز مشکلی، ناگهان ورودی مطلوب از پله واحد با ضریبی از این مقدار جایگزین شود، الگوریتمی که در پی پاسخ‌دادن به ورودی پله واحد بوده است، باید به ورودی جدید پاسخ دهد. این امر بدان معناست که الگوریتمی که مدت زمان زیادی را صرف رسیدن به پاسخ مطلوب پله واحد کرده است، ناگهان از این پاسخ فاصله می‌گیرد و داده‌های اشتباه را تولید می‌کند و حتی بعد از بازگرداندن سیگنال مرجع به مقدار مطلوب، زمان زیادی صرف اصلاح داده‌های اشتباه می‌شود؛ اما در الگوریتم پیشنهادی با کمک شبکه عصبی این مشکل از بین می‌رود؛ زیرا شبکه عصبی به کمک حجم بالایی از داده‌های آموزشی مناسب آموزش دیده است و به‌ازای سیگنال‌های مرجع متفاوت مقدار اولیه مناسب را برای الگوریتم SPSA تولید می‌کند. با توجه به اینکه خروجی شبکه عصبی همان



شکل (۴): فرآیند سنگزنی سیمان

$$y(i+1) = c_1 y(i) + c_2 y(i-1) + d_1 u(i) + d_2 u(i-1) + d_3 u(i-2) \quad (10)$$

که در آن $c_1=1.879$, $c_2=-0.1902$, $d_1=-0.0159$, $d_2=0.0267$, $d_3=0.0107$ است.

رابطه (۱۱) رابطه کنترلی در نظر گرفته می شود:

$$x(i) = \begin{bmatrix} x(1) \\ x(2) \\ x(3) \end{bmatrix} = \begin{bmatrix} e(i)-e(i-1) \\ e(i) \\ e(i)-2e(i-1)+e(i-2) \end{bmatrix} \quad (11)$$

$$u(i) = \theta^T x(i) + u(i-1)$$

با توجه به شکل (۳)، سیگنال های k_{p1} و k_{i1} و k_{d1} مقادیر تخمین زده شده به کمک شبکه عصبی در دوره های معلوم است که برای استفاده در تکرار بعدی در اختیار الگوریتم SPSA قرار می گیرد.

شایان ذکر است بردار θ با توجه به رابطه (۱۲) شامل همان پارامترهای کنترلی است که هدف اصلی به دست آوردن این پارامترها است.

$$\theta = \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} \quad (12)$$

رابطه (۱۳) مقادیر مطلوب اندازه ذرات را نشان می دهد.

در فرآیند سنگزنی سیمان، عوامل بسیاری همانند سرعت فن ها، سرعت جداکننده و ... مؤثرند؛ البته باید توجه داشت برای دستیابی به هر سایز مطلوب، سرعت چرخش فن ها و جداکننده نقش متفاوتی دارند؛ برای مثال، اگر سایز کمتر از ۳ میکرومتر مدنظر باشد، سرعت فن ها اهمیت پیدا می کند؛ اما اگر اندازه مطلوب ذرات بین ۳ تا ۳۲ میکرومتر باشد، کافی است سرعت جداکننده مدنظر قرار گیرد.

تابع سنگزنی سیمان، یک تابع غیرخطی همانند رابطه (۸) است:

$$y(i+1) = G(\Phi(i)) \quad (8)$$

که در آن $\Phi(i)$ با رابطه (۹) نشان داده می شود.

$$\Phi(i) = [y(i), \dots, y(i-n), u(i), \dots, u(i-m)] \quad (9)$$

خروجی y نشان دهنده نسبت اندازه ذرات و u نشان دهنده سرعت جداکننده است. درخور ذکر است m و n به ترتیب مرتبه های ورودی و خروجی اند. $u(i)$ و $y(i)$ نشان دهنده نمونه کنونی است و $y(i-n)$ ، $u(i-m)$ نشان دهنده n و m نمونه قبل و $y(i+1)$ نشان دهنده مقدار نمونه بعد است.

طبق پژوهش های موجود، مدلی ساده شده به فرم (۱۰)، برای شبیه سازی در نظر گرفته شده است [۲۸].

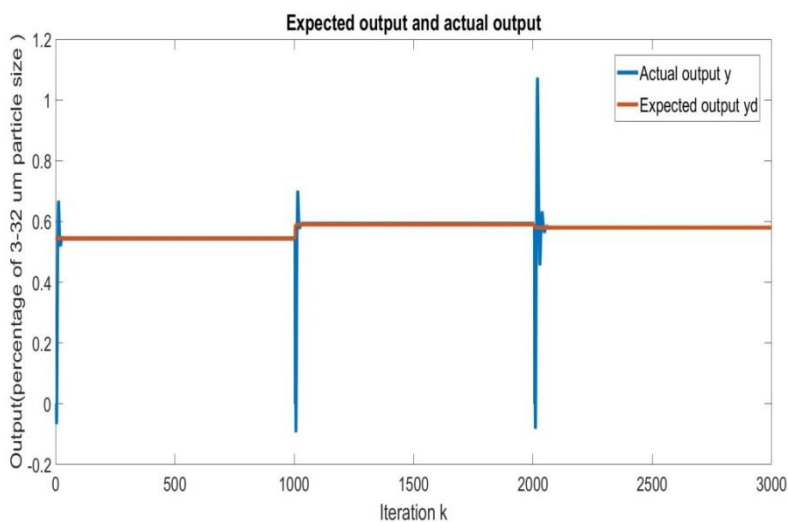
طراحی کنترل کننده PID تطبیقی با استفاده از الگوریتم تقریبات تصادفی انحرافات هم‌زمان و آموزش شبکه عصبی

نکته شایان توجه
اینکه در انتخاب A باید دقت داشت مقدار آن از مقدار
تکرارها کمتر باشد.

نتایج شبیه‌سازی در شکل (۵) مشاهده می‌شود.

$$y_d(i+1) = \begin{cases} 7 \mu\text{m}, & 0 \leq k < 150 \\ 14 \mu\text{m}, & 150 \leq k < 300 \\ 29 \mu\text{m}, & 300 \leq k < 450 \end{cases} \quad (13)$$

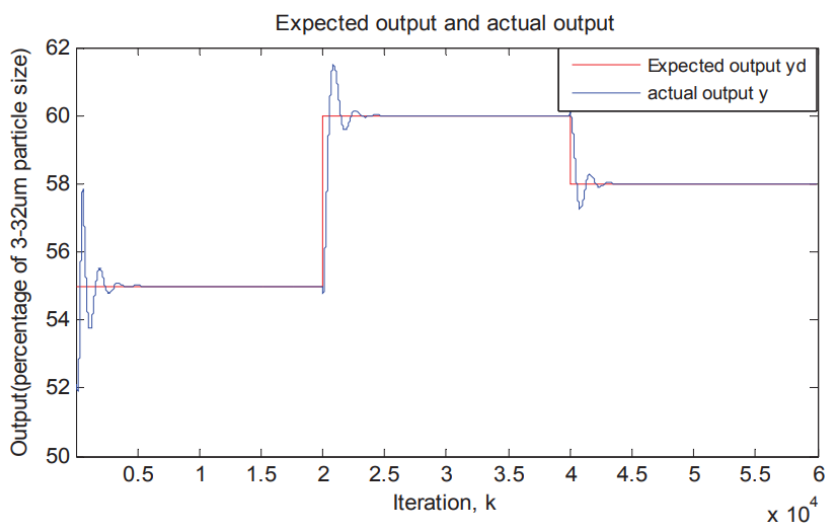
در این مقاله، ضریب α برابر با ۰.۶۰۲ و ضریب γ
برابر ۰.۱۰۱ و $A=100$ ، $b_p=20$ ، $b_i=5$ ، $b_d=50$ ، $f_p=2$ ،



شکل (۵): منحنی نشان‌دهنده اندازه ذرات حاصل از فرآیند سنگ‌زنی سیمان کنترل شده با کنترل کننده (PID) تطبیقی به همراه الگوریتم (SPSA) و آموزش شبکه عصبی

بنابراین، نتایج حاصله از اجرا شدن الگوریتم ارائه شده در
[۲۸] در شکل (۶) مشاهده می‌شود.

به دلیل بالابودن سرعت این روش و پاسخ‌دهی آن در
تکرارهای کم، امکان مقایسه این روش با روش ارائه شده در
[۲۸] به دلیل تکرار بالا بالغ بر ۶۰۰۰۰ وجود ندارد؛



شکل (۶): منحنی نشان‌دهنده اندازه ذرات حاصل از فرآیند سنگ‌زنی سیمان کنترل شده با کنترل کننده (PID) تطبیقی به همراه الگوریتم (SPSA) [۲۸].

مجموعه بسیار پیچیده از شش معادله دیفرانسیلی تلفیقی غیرخطی است.

دینامیک زاویه پیچ هواپیما با تابع تبدیل (۱۴) نشان داده می‌شود:

$$G(s) = \frac{1.15s + 0.18}{s^3 + 0.74s^2 + 0.92s} \quad (14)$$

از این رابطه به منظور تعیین جهت گیری دماغه هواپیما با توجه به محور جانبی با استفاده از الگوریتم پیشنهادی استفاده شده است.

به منظور شبیه‌سازی همانند قبل برای محاسبه گین b_i و ضریب f_i برابر با 0.602 و ضریب γ برابر 0.101 و بقیه پارامترها به صورت زیر در نظر گرفته می‌شوند:

$$f_p = 0.1, b_d = 0.1, b_i = 0.1, b_p = 1.0, A = 3.0, f_i = 0.1$$

علاوه بر روش پیشنهادی، SPSA نیز برای مقایسه به سیستم اعمال می‌شود. در شبیه‌سازی تعداد تکرارها برای هر دو الگوریتم، یکسان و برابر با ۲۰ تکرار است. نتایج حاصله از شبیه‌سازی در شکل (۷) نشان داده شده‌اند.

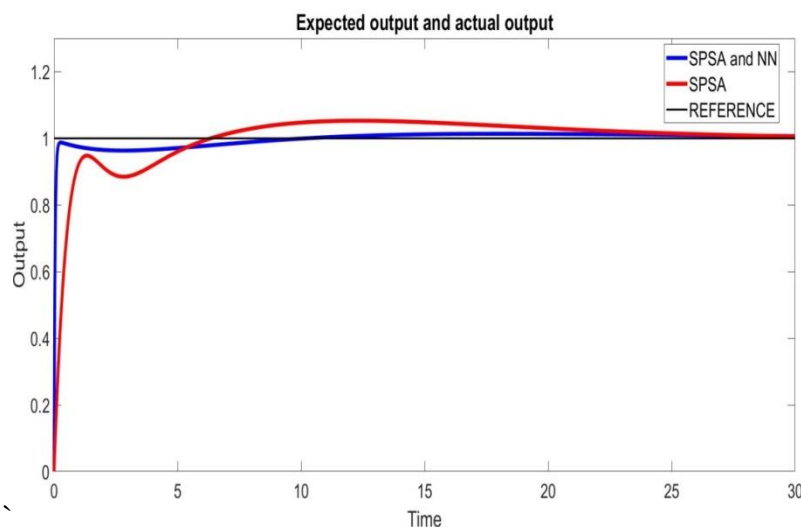
با کمی دقت در شکل (۷) می‌توان دریافت پاسخ الگوریتم پیشنهادی، سرعت و کیفیت بالاتر و پاسخ، انحراف کمتری دارد و سیگنال خطا با سرعت بیشتری به سمت صفر میل می‌کند.

با توجه به شکل (۵)، الگوریتمی که از شبکه عصبی استفاده می‌کند، با توجه به آموزش شبکه عصبی و با در دست داشتن مقدار اولیه نزدیک به مقدار مطلوب، خطای خروجی به سرعت به سمت صفر نزدیک می‌شود.

نکته درخور توجه دیگر اینکه تقاوم بالای الگوریتم پیشنهادی در برابر تغییرات سریع مقدار مطلوب است. با توجه به شکل (۵)، با اعمال سه مقدار مطلوب متفاوت، الگوریتم، پاسخ مناسب را تنها بعد از چند تکرار و با استفاده از اولین آموزش شبکه عصبی و بدون انحراف پاسخ، در اختیار کاربر قرار داده است؛ اما الگوریتم ارائه شده در [۲۸] (که پاسخ آن در شکل (۶) آمده است)، برای دستیابی به پاسخ مناسب مقادیر مطلوب به زمان نیاز دارد و تا تکرار ۶۰۰۰ ذرات با اندازه مناسب را فراهم نمی‌کند.

۴-۲- کنترل زاویه پیچ^{۱۵} هواپیما

سیستم مطالعه شده دوم، کنترل زاویه پیچ هواپیما است که برگرفته از مرجع [۲۹] است. هواپیما بیشتر، توانایی چرخش به حول سه محور را داراست. محور اول، محوری عمودی با نام یاق^{۱۶}، محور دوم، محور طولی با نام رول^{۱۷} و در انتها محور جانبی با نام پیچ است. زاویه پیچ از دوران به حول خطی فرضی در نظر گرفته می‌شود که از انتهای یک بال تا انتهای بال دیگر است. معادلات حاکم بر هواپیما یک



شکل (۷): منحنی پاسخ کنترل زاویه پیچ هواپیما به کمک طراحی کنترل کننده (PID) تطبیقی و الگوریتم (SPSA) در قیاس با طراحی کنترل کننده (PID) تطبیقی و الگوریتم (SPSA) و آموزش شبکه عصبی.

۵- نتیجه گیری

هدف از این پژوهش، ارائه راهکاری به منظور به دست آوردن برخط پارامترهای کنترلی PID بدون نیاز به مدل سیستم است. روش‌های پیشنهادی که معمولاً به کار گرفته می‌شوند، مستلزم آن‌اند که مدل تقریبی سیستم را در اختیار داشته باشند. در بسیاری از سیستم‌ها، به دست آوردن مدل سیستم، امری دشوار یا معمولاً امکان‌ناپذیر است؛ بنابراین، در این مقاله با استفاده از SPSA و بهره‌گیری از شبکه عصبی در انتخاب شرایط اولیه مناسب، الگوریتمی پیشنهاد شد که تنها با استفاده از داده‌های خروجی و داده‌های ورودی، پارامترهای کنترل کننده PID را به صورت برخط تخمین می‌زند.

به کارگیری شبکه عصبی در روش پیشنهادی باعث کاهش حجم محاسبات الگوریتم SPSA و افزایش سرعت همگرایی الگوریتم و مقاوم بودن سیستم در برابر تغییرات ناگهانی سیگنال مرجع می‌شود.

مراجع

- Signal Processing, Vol. 35, No. 9, September 2016.
- [9] D. Invernizzi, P. Panizza, F. Riccardi, S. Formentin, M. Lovera, "Data-driven attitude control law of a variable-pitch quadrotor: a comparison study", IFAC-Papers On Line, Vol.49, No. 17, January 2016.
- [10] A. Sala, "Integrating virtual reference feedback tuning into a unified closed-loop identification framework", Automatica, Vol. 43, No. 1, January 2007.
- [11] M. Radac, R. Precup, R. Roman, "Data-driven model reference control of MIMO vertical tank systems with model-free VRFT and Q-Learning", ISA Trans., Vol. 73, No. 1, February 2018.
- [12] W. Meng, S. Q. Xie, Q. Liu, C. Z. Lu, Q. Ai, "Robust iterative feedback tuning control of a compliant rehabilitation robot for repetitive ankle training", IEEE/ASME Transactions on Mechatronics, Vol. 22, No. 1, February 2017.
- [13] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation", IEEE Trans. on Automatic Control, Vol. 37, No. 3, March 1992.
- [14] A. Savran, G. Kahraman, "A fuzzy model based adaptive PID controller design for nonlinear and uncertain processes", ISA Transactions, Vol. 53, No. 2, March 2014.
- [15] B. Kostic, G. Gentile, C. Antoniou, "Techniques for improving the effectiveness of the SPSA algorithm in dynamic demand calibration", In Models and Technologies for Intelligent Transportation Systems (MT-ITS), 5th IEEE International Conference, pp. 368-373, 26 June 2017.
- [16] K. Ito, T. Dhaene, "Adaptive initial step size selection for Simultaneous Perturbation Stochastic Approximation", Springer Plus, Vol. 5, No. 1, December 2016.
- [17] L. Lu, Y. Xu, C. Antoniou, M. Ben-Akiva, "An enhanced SPSA algorithm for the calibration of Dynamic Traffic Assignment models Transportation Research Part C", Emerging Technologies, Vol. 51, No. 1, February 2015.
- [18] B. Pouladi, A. Karkevandi-Talkhooncheh, M. Sharifi, S. Gerami, A. Nourmohammad & A. Vahidi., Enhancement of SPSA algorithm performance using reservoir quality maps: Application to coupled well placement and control optimization problems", Journal of Petroleum Science and Engineering, Vol. 189, January 2020.
- [19] M. Nik Mohd Zaitul Akmal, et al, "Data-Driven PID Tuning for Liquid Slosh-Free Motion Using Memory-Based SPSA Algorithm", 10th National Technical Seminar on Underwater System Technology, pp. 197-210, Springer, Singapore, 2019.
- [20] M. Nazri, A. Fadhlan, M. Ikram Mohd Rashid, "Simultaneous Perturbation Stochastic Approximation Optimization for Energy Management Strategy of HEV", 10th National Technical Seminar on Underwater System Technology, pp. 361-368, Springer, Singapore, 2019.
- [1] Z. Hou, H. Gao, F. L. Lewis, "Data-driven control and learning systems", IEEE Trans. on Industrial Electronics, Vol. 64, No. 5, May 2017.
- [2] D. Piga, S. Formentin, A. Bemporad, "Direct data-driven control of constrained systems", IEEE Trans. on Control Systems Technology, Vol. 26, No. 4, July 2018.
- [3] S. Yin, X. Li, H. Gao, O. Kaynak, "Data-based techniques focused on modern industry: An overview", IEEE Trans. on Industrial Electronics, Vol. 62, No. 1, January 2015.
- [4] M. Tanaskovic, L. Fagiano, C. Novara, M. Morari, "Data-driven control of nonlinear systems: An on-line direct approach", Automatica, Vol. 75, No. 1, January 2017.
- [5] Y. Zhu, Z. Hou, "Controller dynamic linearization-based model-free adaptive control framework for a class of non-linear system", IET Control Theory & Applications, Vol. 9, No. 7, April 2015.
- [6] Z. Hou, S. Jin, "Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems", IEEE Trans. Neural Network, Vol. 22, No. 12, November 2011.
- [7] Z. Hou, S. Jin, "A novel data-driven control approach for a class of discrete-time nonlinear systems", IEEE Trans. Control Syst Technol, Vol. 19, No. 6, December 2011.
- [8] K. Chen, S. Li, "Unfalsified Adaptive PID Control for Time-Varying Systems Using a Fading Memory Cost Function", Circuits, Systems, and

- [25] J. C. Spall, J. A. Cristion, "Model-free control of nonlinear stochastic systems with discrete-time measurements", *IEEE Trans. on automatic control*, Vol. 43, No. 9, September 1998.
- [26] L. Fausett, "Fundamentals of neural networks: architectures algorithms, and applications", Pearson Education India, 2006.
- [27] J. Zou, Y. Han, S. S. So, "Overview of artificial neural networks", *Artificial Neural Networks*, Humana Press, pp. 14-22, 2008.
- [28] L. Zhang, Z. Yuan, X. Du, Y. Gong, "An adaptive PID control for cement grinding particle size based on simultaneous perturbation stochastic approximation", *Chinese Automation Congress (CAC) IEEE*, pp. 1191-1195, 20 October 2017.
- [29] A. S. Bazanella, L. F. Pereira, A. Parraga. "A new method for PID tuning including plants without ultimate frequency", *IEEE Trans. on Control Systems Technology*, Vol. 25, No. 2, May 2016.
- [21] B. Pérez-Sánchez, O. Fontenla-Romero, B. Guijarro-Berdiñas, "A review of adaptive online learning for artificial neural networks", *Artificial Intelligence Review*, Vol. 49, No. 2, February 2018.
- [22] M. W. Gardner, S. R. Dorling, "Artificial neural networks (the multilayer perceptron)-a review of applications in the atmospheric sciences", *Atmospheric environment*, Vol. 32, No. 14-15, August 1998.
- [23] N. B. Chaphalkar, K. C. Iyer, S. K. Patil, "Prediction of outcome of construction dispute claims using multilayer perceptron neural network model", *International Journal of Project Management*, Vol. 33, No. 8, November 2015.
- [24] M. S. Odabas, H. Simsek, C. W. Lee CW, İ. İseri. "Multilayer Perceptron Neural Network Approach to Estimate Chlorophyll Concentration Index of Lettuce (*Lactuca sativa* L.) ", *Communications in soil science and plant analysis*, Vol. 48, No. 2, January 2017.

¹ Data-Driven Controller (DDC)

² Simultaneous Perturbation Stochastic Approximation (SPSA)

³ Model Based Control (MBC)

⁴ Model Free Adaptive Control (MFAC)

⁵ Unfalsified Control (UC)

⁶ Correlation-Based Tuning (CBT)

⁷ Virtual Reference Feedback Tuning (VRFT)

⁸ Iterative Feedback Tuning (IFT)

⁹ Artificial Neural Network (ANN)

¹⁰ Multi-layer perceptron

¹¹ Feedforward

¹² Generalized Delta Rule

¹³ Backpropagation of errors

¹⁴ Gradient Descent Method

¹⁵ Pitch angle

¹⁶ Yaw angle

¹⁷ Roll angle

