مجله کنترل
ISSN (print) 2008-8345
ISSN (online) 2538-3752
جلد ۱۲، شماره ۲، تابستان ۱۳۹۷، صفحه ۱۱-۱

محاسبه ماتریس تداخل برای سیستم‌های چند متغیره غیرخطی با استفاده از الگوریتم ساختار صفر نامحدود

زینب اصلی پور[۱]، علیرضا فاتحی[۲]

گروه پژوهشی اتوماسیون و کنترل پیشرفته فرایندها (ایپک)، دانشکدهٔ مهندسی برق، دانشگاه صنعتی خواجه نصیر الدین طوسی
z.aslipour@sbu.ac.ir[۱] ، fatehi@kntu.ac.ir[۲]

دریافت: ۱۳۹۵/۱۲/۲۲ ویرایش اول: ۱۳۹۶/۴/۲۲ پذیرش: ۱۳۹۶/۹/۱۹

چکیده: ماتریس تداخل نقش مهمی در زمینه کنترل سیستم‌های خطی و غیرخطی چندمتغیره دارد. در این مقاله، روشی برای بدست آوردن این ماتریس برای یک سیستم چندمتغیره غیرخطی پیشنهاد شده است. الگوریتم قبلی موجود در این زمینه فقط برای سیستم‌های مربعی مناسب است و علاوه بر آن همیشه تعیین ماتریس تداخل را برای این سیستم‌ها تضمین نمی‌کند. روش ارائه شده در این مقاله، الگوریتم بالا را بسط می‌دهد به‌گونه‌ای که کاستی‌های آن برطرف می‌گردد. الگوریتم پیشنهادی بر اساس تعیین ساختار صفرهای انتقال نامحدود سیستم غیرخطی عمل می‌کند و با درنظرگرفتن این صفرها ساختار ماتریس تداخل را بدست می‌آورد. در انتها، کارآیی الگوریتم پیشنهادی با ذکر چند مثال مختلف مورد بررسی قرار می‌گیرد.

کلمات کلیدی: سیستم غیرخطی، چند ورودی-چند خروجی، ماتریس تداخل، ساختار صفر نامحدود.

Calculation of Interactor Matrix for Nonlinear Multivariable Systems via Infinite Zero Structure Algorithm

Zeinab Aslipour, Alireza Fatehi

Abstract: An interactor matrix plays an important role in the multivariable linear and nonlinear control systems theory. This paper proposes a method to obtain the interactor matrix for nonlinear multivariable systems. The only existing algorithm works only on square systems; moreover, it cannot guarantee providing the interactor matrix for these systems. The proposed method of this paper improves the above algorithm so that both mentioned defects are solved. The modified algorithm uses the infinite zeros structure for the nonlinear system and then it obtains the structure of interactor matrix. The effectiveness of the introduced method has been shown using various examples.

Keywords: Nonlinear System, Multi-Input Multi-Output, Interactor Matrix, Infinite Zeros Structure.

۱- مقدمه

در کنترل سیستم‌های چندمتغیره، وجود تأخیرهای متفاوت بین ورودی و خروجی‌های مختلف کار تحلیل و طراحی را پیچیده می‌کند. برای رفع این مشکل، مفهومی به نام ماتریس تداخل تعریف می‌شود. ماتریس تداخل در حالت گسسته تعمیم جمله تأخیر از سیستم‌های تک ورودی-تک خروجی[۱] به سیستم‌های چند ورودی-چند خروجی[۲] می‌باشد. این ماتریس برای سیستم‌های چند ورودی-چند خروجی برای اولین بار توسط Wolovich و Falb در ۱۹۷۶ مطرح شد که در ارتباط مستقیم با مفهوم تأخیر می‌باشد[۱]. برای سیستم‌های زمان پیوسته، تأخیر به عنوان زمان موردنیاز برای اینکه تغییر در ورودی یک سیستم منجر به

[۱] Single Input-Single Output (SISO)

[۲] Multi Input-Multi Output (MIMO)

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

تغییر در خروجی آن شود، تعریف می‌شود که برابر اولین زمان غیرصفر شدن پاسخ ضربه است. در سیستم‌های گسسته می‌توان گفت که تأخیر برابر تعداد صفرهای نامحدود است [۲].

در سیستم‌های چند ورودی-چند خروجی ، دو تعریف برای تأخیر می‌توان در نظرگرفت. تعریف اول که تاخیر عنصر نام دارد، تأخیر اثرگذاری یک ورودی بر روی یک خروجی مشخص را بیان می‌کند. این تعریف معادل با مفهوم صفرهای عنصردر بی‌نهایت است. اما تعریف دوم براساس صفرهای انتقال در بی‌نهایت ارائه می‌شود. در این تعریف، تأخیر بر حسب یک ماتریس ناویژه بیان می‌گردد که شامل کمترین تعداد ضرایب پاسخ ضربه سیستم یا همان پارامترهای مارکوف است. این ماتریس، ماتریس تداخل نامیده می‌شود و در زمینه‌های مختلفی کاربرد دارد. یکی از کاربردهای ماتریس تداخل، طراحی کنترل‌کننده حداقل واریانس برای سیستم‌های چند ورودی-چند خروجی می‌باشد. در[۳] ، سیگنال کنترل جهت طراحی این کنترل‌کننده با در نظرگرفتن ماتریس تداخل قطری برای سیستم چند ورودی-چند خروجی بدست آمده است. علاوه بر این می‌توان با استفاده از مرتبه این ماتریس بدون نیاز به محاسبه کامل ماتریس تداخل، حد بالا و پایینی برای شاخص حداقل واریانس محاسبه کرد [۴]. می‌توان از این ماتریس در طراحی کنترل‌کننده تطبیقی سیستم‌های چند ورودی-چندخروجی نیز استفاده کرد[۵]. در بحث کنترل تطبیقی مدل مرجع، اطلاع داشتن از ماتریس تداخل سیستم برای انتخاب مدل مرجع نیاز است زیرا مدل مرجع باید همان ماتریس تداخل را داشته باشد[۶, ۷]. هم‌چنین در [۸] از این ماتریس برای طراحی کنترل کننده فیدبک حالت با معیار H_∞ استفاده شده است.

برای تعیین ماتریس تداخل در سیستم‌های خطی الگوریتم‌های زیادی ارائه شده است. در [۹] الگوریتمی معرفی شده است که در آن با استفاده از یک توصیف ماتریسی چپ از تابع تبدیل سیستم، ماتریس تداخل بدست می‌آید. در [۱۰] از پارامترهای مارکوف برای تعیین ماتریس تداخل استفاده شده است. این پارامترها به طور مستقیم از تابع تبدیل سیستم و یا داده‌های ورودی-خروجی سیستم چند ورودی-چند خروجی قابل محاسبه می‌باشند. در [۱۱] روشی برای بدست آوردن این پارامترها در سیستم‌های چند ورودی-چند خروجی ، معرفی شده است که از الگوریتم حداقل مربعات برای تخمین پارامترهای مارکوف و محاسبه ماتریس تداخل بر حسب مدلی شامل خروجی‌های حال، گذشته و ووردی‌های گذشته، استفاده می‌کند. در مراجع [۱۲-۱۴] نیز روش‌های دیگری برای محاسبه ماتریس تداخل با استفاده از معکوس مجازی تابع تبدیل سیستم برای سیستم‌های مربعی و غیر مربعی آورده شده است. در [۱۵, ۱۶] نیز با در نظرگرفتن فرض‌هایی روی تابع تبدیل سیستم، ماتریس تداخل محاسبه گردیده است. بعد از یافتن ساختار مناسب برای ماتریس تداخل، می‌توان با استفاده از روشی که در [۱۷] ارائه شده است به شناسایی پارامترهای مجهول این ماتریس همزمان با پارامترهای سیستم پرداخت. در [۲] نشان داده شده است که ترکیب خطی پارامترهای مارکوف فرآیند در حالت حلقه باز و حلقه بسته یکی است و می‌توان ماتریس تداخل سیستم حلقه باز را از روی داده‌های حلقه بسته بدست آورد.

یکی از روش‌های ارائه شده برای محاسبه ماتریس تداخل در سیستم‌های غیرخطی آن است که سیستم غیرخطی موجود حول یک نقطه کار مشخص خطی شده و سپس ماتریس تداخل خطی با استفاده از الگوریتم‌های معرفی شده بدست آید [۷, ۱۸]. در [۱۹] الگوریتمی غیرخطی جهت بدست آوردن ماتریس تداخل برای یک سیستم غیرخطی در تمامی نقاط کاری سیستم ارائه شده است. البته این الگوریتم برای بخشی از سیستم‌های غیرخطی قابل پیاده‌سازی نیست، که در بخش ۳-۱-۱ به جزییات آن پرداخته خواهد شد.

در این مقاله، سعی شده است که برای سیستم‌های غیرخطی چند ورودی-چند خروجی که به صورت مربعی یا غیرمربعی هستند ماتریس تداخل خطی بدست آید. روند محاسبه ماتریس تداخل یک روند غیرخطی است، یعنی مشابه روش‌های موجود در برخی مقالات از خطی-سازی سیستم غیرخطی حول یک نقطه کار و تابع تبدیل سیستم برای تعیین ماتریس تداخل استفاده نمی‌کند. مشابه [۱۹] ابتدا کلاسی از سیستم‌های غیرخطی چند ورودی-چند خروجی به صورت افاین۱ در نظرگرفته می‌شود و سپس با استفاده از الگوریتم پیشنهادی، ماتریس تداخل خطی تعیین می‌گردد. الگوریتم پیشنهادی اصلاح یافته، الگوریتم ارائه شده در [۱۹] است اما برای تمامی سیستم‌های غیرخطی افاین می-تواند ماتریس تداخل را ارائه دهد. این ماتریس تداخل ویژگی‌های اصلی تعریف این مفهوم را داراست که می‌توان با خطی‌سازی سیستم غیرخطی حول یک نقطه کار مشخص این ویژگی‌ها را تحقیق کرده و صحت روابط را نشان داد.

این مقاله در ادامه به صورت زیر سازمان‌دهی شده است. در بخش ۲، تعریف ماتریس تداخل برای یک سیستم خطی و غیرخطی آورده می‌شود . بخش ۳، الگوریتم موجود برای تعیین ساختار ماتریس تداخل یک سیستم غیرخطی را بررسی کرده و با ذکر چند مثال محدودیت‌های موجود در این روش را بیان می‌کند و سپس الگوریتم ساختار صفر نامحدود در این بخش ارائه می‌شود. در بخش ۴، الگوریتم پیشنهادی در این مقاله برای تعیین ساختار ماتریس تداخل بیان شده و کارایی این الگوریتم در جهت رفع کاستی‌های روش بخش ۳، با ذکر چند مثال و هم‌چنین در نظرگرفتن سیستم pH به عنوان یک سیستم واقعی نشان داده می‌شود. جمع‌بندی پژوهش در بخش ۵ ارائه می‌گردد.

۱ - affine

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

۲- ماتریس تداخل

۲-۱ ماتریس تداخل سیستم خطی

برای هر ماتریس تابع تبدیل گسسته در زمان، چندجمله ای گویا و سره $\mathbf{G}$ با ابعاد $(p \geq m)\, p \times m$، یک ماتریس $p \times p$ پایین مثلثی ناویژه یکتا چپ[۱] $\mathbf{D}$ به نام ماتریس تداخل وجود دارد به گونه‌ای که دو شرط زیر در مورد آن صادق است [۲]:

$$\begin{cases} i) \left|\mathbf{D}(z)\right| = z^r \\ ii) \lim\limits_{z^{-1}\to 0} \mathbf{D}(z)\mathbf{G}(z) = \lim\limits_{z^{-1}\to 0} \mathbf{G}(z) = \mathbf{K} \end{cases} \quad (۱)$$

که r تعداد صفرهای نامحدود $\mathbf{G}$، $\mathbf{K}$ یک ماتریس ثابت و رتبه کامل[۲] است که معادل بهره فرکانس بالای سیستم $\mathbf{D}(z)\mathbf{G}(z)$ است و $\mathbf{G}$ تابع تبدیل بدون تأخیر ماتریس $\mathbf{G}$ است که فقط شامل صفرهای محدود می‌باشد. برای سیستم‌های پیوسته در زمان نیز مشابه سیستم‌های گسسته در زمان ماتریس تداخل تعریف می‌شود [۲۰].

در حالت کلی ماتریس تداخل را می‌توان به شکل زیر نوشت:

$$\mathbf{D} = \mathbf{D}_0 z^d + \mathbf{D}_1 z^{d-1} + \ldots + \mathbf{D}_{d-1} z \quad (۲)$$

که d مرتبه ماتریس تداخل نامیده می‌شود و برای هر تابع تبدیل مشخص یکتاست و $\mathbf{D}_i \in R^{p\times p}$ برای $(i = 1, 2, \ldots, d)$ ماتریس‌های ضرایب می‌باشند. از آنجایی‌که بهره استاتیکی ماتریس تداخل لزوماً ماتریس همانی نیست، مقایسه آن در حالت چند ورودی-چند خروجی با حالت تک ورودی-تک خروجی (z^d) دچار اشکال می‌شود؛ بنابراین می‌توان ماتریس تداخل را در $\mathbf{D}^{-1}(1)$ ضرب کرد تا ماتریس تداخل اصلاح شده[۳] بدست آید:

$$\mathbf{D}_m(z) = \mathbf{D}^{-1}(1)\mathbf{D}(z) \quad (۳)$$

البته اگر در تعریف ماتریس تداخل شرط زیر اضافه شود، اطمینان حاصل می‌شود که بهره DC واحد است[۲۱]:

$$\mathbf{D}_0 + \mathbf{D}_1 + \ldots + \mathbf{D}_{d-1} = \mathbf{I} \quad (۴)$$

علاوه بر این، ماتریس تداخل را در حالت خطی می‌توان به صورت یک ماتریس پایین مثلثی یکتا و ناویژه تعریف کرد که مرتبه عناصر قطری این ماتریس، مرتبه صفرهای نامحدود سیستم است[۲۲]:

$$\mathbf{D}(z) = \mathbf{H}(z)\, diag\left[z^{f_1}, z^{f_2}, \ldots, z^{f_p}\right]$$

$$\mathbf{H}(z) = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \varphi_{21}(z) & 1 & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ \varphi_{p1}(z) & \varphi_{p2}(z) & \ldots & 1 \end{bmatrix} \quad (۵)$$

که $\varphi_{ij}(z)$ عناصر قابل قسمت به z و یا صفر است. f_1 تا f_p مرتبه صفرهای انتقال نامحدود سیستم است، که رابطه زیر بین آن‌ها برقرار است:

$$f_1 \leq f_2 \ldots \leq f_p \quad (۶)$$

در این صورت یک ماتریس غیر یکتا و دو سره به نام $\mathbf{B}(z)$ برای سیستم می‌توان تعریف کرد به گونه‌ای که:

$$\mathbf{G}(z) = \left[\mathbf{D}^{-1}(z), 0\right]\mathbf{B}(z) \quad (۷)$$

مشاهده می‌شود که تابع تبدیل سیستم با معکوس ماتریس تداخل و یک ماتریس دوسره بیان شده است. از آنجایی‌که ماتریس دوسره هیچ قطب یا صفر نامحدودی ندارد، یعنی هم خودش و هم معکوسش سره است، بنابراین ساختار بی‌نهایت تابع تبدیل سیستم یعنی $\mathbf{G}(z)$ همان ساختار بی‌نهایت معکوس ماتریس تداخل می‌باشد و می‌توان گفت که بزرگترین مرتبه هر چندجمله‌ای ماتریس تداخل برابر با بزرگترین مرتبه صفر بی‌نهایت تابع تبدیل سیستم است. بنابراین مرتبه عناصر قطری این ماتریس برابر با مرتبه صفرهای انتقال نامحدود سیستم می‌باشد.

۲-۲ ماتریس تداخل سیستم غیر خطی

معادلات فضای حالت یک سیستم غیرخطی m ورودی- m خروجی به صورت زیر می باشد:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f(x)} + \boldsymbol{g(x)u} \\ \boldsymbol{y} &= \boldsymbol{h(x)} \end{aligned} \quad (۸)$$

که در آن $\boldsymbol{x} \in R^n$ و $\boldsymbol{u}, \boldsymbol{y} \in R^m$، $\boldsymbol{f(x)} \in R^n$، $\boldsymbol{g(x)} \in R^{n\times m}$ و $\boldsymbol{h(x)} \in R^m$ می‌باشند. برای این سیستم، می‌توان یک ماتریس تداخل به فرم رابطه (۹) تعریف کرد. ماتریس تداخل معرفی شده $m \times m$ بوده و مرتبه آن d است [۱۶]:

$$\mathbf{N}(v) = \mathbf{N}_d v^d + \mathbf{N}_{d-1} v^{d-1} + \ldots + \mathbf{N}_0 \quad (۹)$$

که در آن v اپراتور مشتق‌گیری است. در صورت ضرب کردن ماتریس تداخل در خروجی سیستم داریم:

$$\mathbf{N}(v)\, y = \boldsymbol{a(x)} + \boldsymbol{\Lambda(x)u} \quad (۱۰)$$

[۱] - lower left triangular matrix

[۲] - full rank

[۳] - modified interactor matrix

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

که $\mathbf{\Lambda}(x) \in R^{m\times m}$ یک ماتریس ناویژه است. $\mathbf{a(x)}$ و $\mathbf{\Lambda}(x)$ به صورت زیر محاسبه می‌شوند:

$$\begin{aligned} \boldsymbol{a(x)} &= \left[\mathbf{N}_0, \mathbf{N}_1, \ldots, \mathbf{N}_d\right] \begin{bmatrix} \boldsymbol{h}(x) \\ \boldsymbol{L_f h}(x) \\ \vdots \\ \boldsymbol{L_f^d h}(x) \end{bmatrix} \\ \boldsymbol{\Lambda}(x) &= \left[\mathbf{N}_1, \mathbf{N}_2, \ldots, \mathbf{N}_d\right] \begin{bmatrix} \boldsymbol{L_g h}(x) \\ \boldsymbol{L_g L_f h}(x) \\ \vdots \\ \boldsymbol{L_g L_f^{d-1} h}(x) \end{bmatrix} \end{aligned} \quad (۱۱)$$

در ابطه (۱۱) از مشتق لی استفاده شده است.

۳- محاسبه ماتریس تداخل سیستم‌های غیرخطی

ماتریس تداخل هم برای سیستم‌های خطی و هم برای سیستم‌های غیرخطی قابل تعریف می باشد. البته در هر دو حالت، خود این ماتریس به صورت خطی در نظر گرفته می‌شود. مفهوم ماتریس تداخل برای سیستم-های غیرخطی دقیقا مشابه سیستم‌های خطی که در بخش ۲-۱ ذکر گردید، می‌باشد. یک روش بدست آوردن آن، خطی کردن سیستم حول یک نقطه کار مشخص و سپس استفاده از الگوریتم‌های موجود در حوزه سیستم‌های خطی است [۷، ۱۸]. روش دیگری نیز موجود است که مستقیم از ساختار خود سیستم غیرخطی برای محاسبه ماتریس تداخل استفاده می‌کند که در زیر به تفصیل آمده است.

۳-۱ مروری بر روش Mutoh

در مرجع [۱۹]، Mutoh روشی را ذکر کرده است که با استفاده از آن برای یک سیستم غیرخطی مربعی $m\times m$، ابتدا ساختار ماتریس تداخل محاسبه گردیده و سپس پارامترهای مجهول این ماتریس شناسایی می‌شوند.

برای سیستم غیرخطی (۸) با کمک مشتقات لی می‌توان ماتریس‌های $\mathbf{T}_i$ را به صورت زیر تعریف کرد:

$$\mathbf{T_i}(x) = L_g L_f^{i-1} \boldsymbol{h}(x) \quad i = 1, 2, \ldots \quad (۱۲)$$

از طرفی، ماتریس‌های $\mathbf{\Gamma_i}$ به صورت رابطه (۱۳) قابل تعریف است:

$$\begin{aligned} \mathbf{\Gamma}_1(x) &= \mathbf{T}_1^{\mathrm{T}}(x) \\ \mathbf{\Gamma_i}(x) &= \begin{pmatrix} \mathbf{T}_1^{\mathrm{T}}(x) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \mathbf{T_i}^{\mathrm{T}}(x) & \cdots & \mathbf{T}_1^{\mathrm{T}}(x) \end{pmatrix} \quad i = 2, 3, \ldots \end{aligned} \quad (۱۳)$$

این ماتریس‌ها تا زمانی که شرط زیر برقرار شود تعریف می‌شوند:

$$rank\mathbf{\Gamma}_k(x) = rank\mathbf{\Gamma}_{k-1}(x) + m \quad (۱۴)$$

می‌توان نشان داد که:

$$\mathbf{\Gamma}_d(x) \begin{bmatrix} \mathbf{N}_d^T \\ \mathbf{N}_{d-1}^T \\ \vdots \\ \mathbf{N}_1^T \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \overline{\mathbf{\Lambda}^{\mathrm{T}}(x)} \end{bmatrix} \quad (۱۵)$$

که در آن $\mathbf{N}_i$ها مطابق (۹) تعریف شده‌اند. فرض می‌شود $\gamma_j^k(x)$ بردارهای مستقل خطی $\mathbf{\Gamma_j}$ را نشان دهد. اگر q_k تعداد بردارهای مستقل خطی $\gamma_j^k(x)$ باشد، آنقدر این مراحل تکرار می-گردند که :

$$q_1 \le q_2 \le \ldots \le q_{d-1} < q_d = m \quad (۱۶)$$

آن‌گاه مرتبه عناصر قطری ماتریس تداخل به صورت زیر تعیین می‌شود:

$$k_1 \le k_2 \le \ldots \le k_{m-1} \le k_m = d \quad (۱۷)$$

به عنوان مثال اگر سیستم موجود ۲×۲ باشد، می‌توان ماتریس تداخل را با توجه به تعریف آن که یک ماتریس پایین مثلثی است به صورت زیر نوشت:

$$\mathbf{N}(\psi, v) = \begin{bmatrix} n_{11}(v) & 0 \\ n_{21}(\psi, v) & n_{22}(v) \end{bmatrix} \quad (۱۸)$$

که در رابطه بالا عناصر روی قطر یعنی $n_{11}(v)$ و $n_{22}(v)$ هر چندجمله‌ای دلخواه معلوم با مرتبه‌های به ترتیب برابر با k_1 و k_2 است که مطابق با الگوریتم بالا بدست می‌آید. اما چندجمله‌ای $n_{21}(\psi, v)$ دارای پارامترهای مجهول می باشند که باید شناسایی گردند که روند شناسایی در [۱۹] به تفصیل آورده شده است.

۳-۱-۱ محدودیتهای روش Mutoh

با بررسی جزئیات این روش می‌توان به برخی محدودیت‌های آن پی‌برد که این موضوع در زیر با ذکر مثال‌هایی آورده شده است.

محدودیت اول: این روش تنها برای کلاس خاصی از سیستم‌های چند ورودی-چند خروجی غیرخطی، یعنی سیستم‌های مربعی، کاربرد دارد.

محدودیت دوم: این روش تنها شامل سیستم‌هایی می‌شود که برای آنها شرط توقف الگوریتم Mutoh یعنی رابطه (۱۴) برقرار شود و نمی‌تواند برای محاسبه ماتریس تداخل سیستم‌هایی که این شرط هیچ‌گاه

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

در مورد آنها اتفاق نمی افتد، استفاده شود. برای مشخص شدن این محدودیت مثال ۱ آورده شده است.

مثال ۱) سیستم غیرخطی (۸) با ماتریس‌های زیر فرض می‌شود، این سیستم 2×2 می باشد:

$$\boldsymbol{f}(\boldsymbol{x})=\begin{bmatrix}-x_1+x_3\\ x_2x_4\\ -x_2x_4-x_2{x_4}^2\\ -x_4\end{bmatrix},\mathbf{g}(\boldsymbol{x})=\begin{bmatrix}x_2 & e^{-x_4}\\ 0 & 0\\ 0 & e^{-x_4}\\ 1 & 0\end{bmatrix},\boldsymbol{h}(\boldsymbol{x})=\begin{bmatrix}x_2\\ x_4\end{bmatrix}$$

الگوریتم Mutoh برای محاسبه ماتریس تداخل خطی آن بکار گرفته می‌شود.

گام اول:

$$\boldsymbol{\Gamma}_1=\mathbf{T}_1^{\mathrm{T}}=\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}$$

بنابراین $\gamma_1^1=1$ و در نتیجه $q_1=1$ خواهد شد.

گام دوم:

$$\boldsymbol{\Gamma}_2=\begin{bmatrix}\mathbf{T}_1^{\mathrm{T}} & 0\\ \mathbf{T}_2^{\mathrm{T}} & \mathbf{T}_1^{\mathrm{T}}\end{bmatrix}=\begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ x_2 & -1 & 0 & 1\\ 0 & 0 & 0 & 0\end{bmatrix}$$

رتبه این ماتریس برابر با ۲ می‌باشد که $\gamma_1^1=1$ و $\gamma_1^2=1$ است. بنابراین $q_2=1$ خواهد بود.

گام سوم:

$$\boldsymbol{\Gamma}_2=\begin{bmatrix}\mathbf{T}_1^{\mathrm{T}} & 0 & 0\\ \mathbf{T}_2^{\mathrm{T}} & \mathbf{T}_1^{\mathrm{T}} & 0\\ \mathbf{T}_3^{\mathrm{T}} & \mathbf{T}_2^{\mathrm{T}} & \mathbf{T}_1^{\mathrm{T}}\end{bmatrix}=\begin{bmatrix}0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ x_2 & -1 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ 2x_2x_4-x_2 & 1 & x_2 & -1 & 0 & 1\\ 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}$$

رتبه این ماتریس برابر با ۳ می‌باشد که در آن $\gamma_1^1=1$ و $\gamma_1^2=1$ و $\gamma_1^3=1$ است.

و همین‌طور این الگوریتم ادامه پیدا می کند چرا که شرط زیر هیچ وقت محقق نمی‌شود:

$$rank\boldsymbol{\Gamma}_k(\boldsymbol{x})=rank\boldsymbol{\Gamma}_{k-1}(\boldsymbol{x})+2$$

محدودیت سوم: این محدودیت شامل سیستم‌هایی می‌شود که برای آن‌ها رابطه (۱۶) برقرار نمی‌گردد. توجه شود که آخرین نامعادله رابطه (۱۶) به صورت $<$ است و نه $\le$. برای مشخص شدن این محدودیت مثال ۲ آورده شده است.

مثال ۲) سیستم غیرخطی (۸) با ماتریس‌های زیر فرض می‌شود، این سیستم 2×2 می باشد:

$$\boldsymbol{f}(\boldsymbol{x})=\begin{bmatrix}x_3\\ x_4\\ x_3x_4\\ x_1x_3x_4\end{bmatrix},\mathbf{g}(\boldsymbol{x})=\begin{bmatrix}1 & x_1\\ x_1 & x_2\\ x_2 & -x_3\\ x_3 & 1\end{bmatrix},\boldsymbol{h}(\boldsymbol{x})=\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

گام اول:

$$\boldsymbol{\Gamma}_1=\mathbf{T}_1^{\mathrm{T}}=\begin{bmatrix}1 & x_1\\ x_1 & x_2\end{bmatrix}$$

رتبه این ماتریس برابر با ۲ است و در نتیجه $q_1=2$ خواهد شد.

گام دوم:

$$\boldsymbol{\Gamma}_2=\begin{bmatrix}\mathbf{T}_1^{\mathrm{T}} & 0\\ \mathbf{T}_2^{\mathrm{T}} & \mathbf{T}_1^{\mathrm{T}}\end{bmatrix}=\begin{bmatrix}1 & x_1 & 0 & 0\\ x_1 & x_2 & 0 & 0\\ x_2 & x_3 & 1 & x_1\\ x_3 & 1 & x_1 & x_2\end{bmatrix}$$

رتبه این ماتریس برابر با ۴ می‌باشد بنابراین $q_2=2$ خواهد بود.

با توجه به آنکه $d=2$ است و از آنجایی که $q_2=q_1=2$ است، رابطه (۱۳) برقرار نمی باشد.

دو مثال فوق نشان می‌دهند که این الگوریتم در بخش اول خود، یعنی بدست آوردن تعیین مرتبه عناصر روی قطر ماتریس تداخل، ممکن است دچار مشکل شود. اما در صورت بدست آمدن این ساختار می‌توان پارامترهای آن را بدست آورد.

۲-۳ الگوریتم ساختار صفر نامحدود

همان‌گونه که در بخش ۲-۱ گفته شد، برای تعیین مرتبه عناصر روی قطر می‌توان از تعیین مرتبه صفرهای نامحدود سیستم نیز استفاده کرد. Liu & Lin [۲۳] الگوریتمی ارائه داده‌اند که برای یک سیستم غیرخطی، مرتبه صفرهای نامحدود را معین می‌کند. این الگوریتم به اختصار در زیر آورده شده است.

برای یک سیستم غیرخطی با رابطه (۸) متغیرهای زیر تعریف می‌شود:

$$\boldsymbol{\Theta}_0(\boldsymbol{x})=\boldsymbol{h}(\boldsymbol{x}),\boldsymbol{\Omega}_0(\boldsymbol{x})=\varnothing,\rho_0=0,k=1 \qquad (۱۹)$$

می‌توان نشان داد که ماتریس $\begin{bmatrix}L_g\boldsymbol{\Omega}_{k-1}(\boldsymbol{x})\\ L_g\boldsymbol{\Theta}_{k-1}(\boldsymbol{x})\end{bmatrix}$ برای $x\in U$ رتبه ثابت برابر با ρ_k دارد و یک $\mathbf{R}_k\in R^{(\rho_k-\rho_{k-1})\times(p-\rho_{k-1})}$ وجود دارد به‌گونه‌ای که ماتریس $\begin{bmatrix}L_g\boldsymbol{\Omega}_{k-1}(\boldsymbol{x})\\ L_g\mathbf{R}_k\boldsymbol{\Theta}_{k-1}(\boldsymbol{x})\end{bmatrix}$ برای $x\in U$ رتبه سطری کامل برابر با ρ_k داشته باشد. در این صورت رابطه (۲۰) برقرار است:

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

$$\Omega_{\boldsymbol{k}}(\boldsymbol{x}) = col\left\{\mathbf{R}_1\Theta_0, \mathbf{R}_2\Theta_1, \ldots, \mathbf{R}_k\Theta_{k-1}\right\} \quad (۲۰)$$

حال اگر $\mathbf{S_k} \in R^{(p-\rho_k)\times(p-\rho_{k-1})}$ به‌گونه‌ای در نظر گرفته شود که:

$$det\left(\begin{bmatrix}\mathbf{R}_k\\ \mathbf{S}_k\end{bmatrix}\right) \neq 0 \quad (۲۱)$$

در آن صورت یک تابع هموار[۱] یکتا $\mathbf{P}_{k,l}(\boldsymbol{x}): U \to R^{(p-\rho_k)\times(\rho_l-\rho_{l-1})},\ l=1,2,\ldots,k$ وجود دارد که:

$$L_g\mathbf{S}_k\Theta_{k-1}(\boldsymbol{x}) - \sum_{l=1}^{k}\mathbf{P}_{k,l}(\boldsymbol{x})L_g\mathbf{R}_l\Theta_{l-1}(\boldsymbol{x}) = 0 \quad (۲۲)$$

حال تابع زیر تعریف می‌شود:

$$\Theta_k(\boldsymbol{x}) = L_f\mathbf{S}_k\Theta_{k-1}(\boldsymbol{x}) - \sum_{l=1}^{k}\mathbf{P}_{k,l}(\boldsymbol{x})L_f\mathbf{R}_l\Theta_{l-1}(\boldsymbol{x}) \quad (۲۳)$$

اگر $k + \sum_{j=1}^{k} j\left(\rho_j - \rho_{j-1}\right) < n$ شود که n تعداد متغیرهای حالت سیستم و $\rho_k < min(p,m)$ که p، تعداد خروجی‌های سیستم و m، تعداد ورودی های آن باشد، k یک عدد افزایش می‌یابد. در غیر این صورت $k = k^*$ تعریف می‌شود و خواهیم داشت:

$$\begin{aligned}
&k = k^*\\
&k^* + \sum_{j=1}^{k^*} j\left(\rho_j - \rho_{j-1}\right) = n \quad or \quad \rho_{k^*} = min(p,m)\\
&\rho = \left\{\rho_1, \rho_2, \ldots, \rho_{k^*}\right\}\\
&m_d = \rho_{k^*}\\
&n_d = \sum_{j=1}^{k^*} j\left(\rho_j - \rho_{j-1}\right)
\end{aligned} \quad (۲۴)$$

سپس مجموعه‌ای از اعداد صحیح به صورت زیر تعریف می‌شوند:

$$\begin{aligned}
&q = \left\{q_1, q_2, \ldots, q_{m_d}\right\}\\
&q = \left\{\underbrace{1,\ldots,1}_{\rho_1-\rho_0}, \underbrace{2,\ldots,2}_{\rho_2-\rho_1}, \ldots, \underbrace{k^*,\ldots,k^*}_{\rho_{k^*}-\rho_{k^*-1}}\right\}\\
&0 < q_1 \le q_2 \le q_3 \le \ldots \le q_{m_d}
\end{aligned} \quad (۲۵)$$

در واقع مجموعه q، صفرهای نامحدود سیستم را نشان می‌دهند. یعنی q_i ها مجموعه‌ای از انتگرال‌گیرها بین جفت ورودی و خروجی می باشند.

٤- الگوریتم پیشنهادی در محاسبه ماتریس تداخل

همان‌طور که مشاهده شد، الگوریتم بخش ۳-۱ در صورت در اختیار داشتن مرتبه درایه‌های روی قطر ماتریس تداخل می‌تواند ماتریس تداخل را بدست آورد اما در برخی موارد این الگوریتم در بدست آوردن این مرتبه دچار مشکل می‌شود. از طرف دیگر این مرتبه برابر مرتبه صفرهای انتقال نامحدود است که توسط الگوریتم بخش ۳-۲ می‌تواند بدست آید. بنابراین با ترکیب این دو الگوریتم می‌توان ساختار ماتریس تداخل را برای هر سیستم غیرخطی با ساختار (۸) اعم از مربعی و غیرمربعی بدست آورد. بر این اساس الگوریتم زیر ارائه می‌شود:

گام اول- محاسبه مرتبه عناصر قطری ماتریس تداخل با استفاده از الگوریتم بخش ۳-۲ .

گام دوم- تعیین ساختار ماتریس تداخل با استفاده از الگوریتم بخش ۳-۱.

گام سوم- شناسایی پارامترهای مجهول ماتریس تداخل با استفاده از روش حداقل مربعات بازگشتی[۲].

۴-۱ محاسبه ماتریس تداخل برای چند مثال

در این بخش سعی می‌شود ساختار ماتریس تداخل مطابق با گام‌های اول و دوم الگوریتم پیشنهادی برای مثال‌های ذکر شده در بخش ۳-۲ محاسبه گردد. سپس می توان پارامترهای مجهول ماتریس تداخل را با استفاده از روش ذکر شده در [۱۹] شناسایی کرد.

مثال ۳) سیستم غیرخطی مثال ۱ را در نظر می‌گیریم:

گام اول: ابتدا با اعمال الگوریتم بخش ۳-۲ مرتبه صفرهای انتقال نامحدود آن تعیین می‌شوند:

$$\Theta_0(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix}x_2\\ x_4\end{bmatrix},\ \Omega_0(\boldsymbol{x}) = \varnothing,\ \rho_0 = 0$$

تکرار اول $(k=1)$:

$$\begin{aligned}
&L_g\Theta_0(\boldsymbol{x}) = \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix} \to \rho_1 = 1\\
&\boldsymbol{R}_1 \in R^{1\times 2} \to \boldsymbol{R}_1 = \begin{bmatrix}0\ 1\end{bmatrix}\\
&\boldsymbol{S}_1 \in R^{1\times 2} \to \boldsymbol{S}_1 = \begin{bmatrix}1\ 0\end{bmatrix}\\
&\Omega_1(x) = x_4,\ \Theta_1(x) = x_2x_4
\end{aligned}$$

[۱] smooth

[۲] Recursive Least Squares(RLS)

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

تکرار دوم $(k=2)$:

$$L_g\Theta_1(x)=[x_2\ \ 0], L_g\Omega_1(x)=[1\ 0]\rightarrow \rho_2=1, R_2=\varnothing$$
$$S_2=1, \Omega_2(x)=x_4, \Theta_2(x)=x_2{x_4}^2$$

تکرار سوم $(k=3)$:

$$L_g\Theta_2(x)=[2x_2x_4\ \ 0],\ \ L_g\Omega_2(x)=[1\ 0]$$
$$\rightarrow \rho_3=1, R_3=\varnothing$$
$$S_3=1, \Omega_3(x)=x_4\ , \Theta_3(x)=x_2{x_4}^3, k^*=3$$
$$\rightarrow k^*+\sum_{j=1}^{k^*} j\left(\rho_j-\rho_{j-1}\right)=4$$

بنابراین این سیستم یک صفر نامحدود از مرتبه یک دارد.

گام دوم: ساختار ماتریس تداخل آن به فرم زیر خواهد شد:

$$\mathbf{N}(\psi,v)=\begin{bmatrix}1 & 0\\ \psi_1 v+\psi_2 & v+a\end{bmatrix}$$

مثال ٤) سیستم غیرخطی مثال ۲ را در نظر می‌گیریم:

گام اول: ابتدا با اعمال الگوریتم بخش ۳-۲ مرتبه صفرهای انتقال نامحدود آن تعیین می‌شوند:

$$\Theta_0(\boldsymbol{x})=\boldsymbol{h}(\boldsymbol{x})=\begin{bmatrix}x_1\\ x_2\end{bmatrix},\ \Omega_0(\boldsymbol{x})=\varnothing\ , \rho_0=0$$

تکرار اول $(k=1)$:

$$L_g\Theta_0(\boldsymbol{x})=\begin{bmatrix}1 & x_1\\ x_1 & x_2\end{bmatrix}\rightarrow \rho_1=2$$
$$\rho_{k^*}=min(p,m)=2\rightarrow k^*=1$$

در نتیجه سیستم دارای دو صفر نامحدود از مرتبه یک است.

گام دوم: بنابراین ساختار ماتریس تداخل آن به فرم زیر می‌شود:

$$\mathbf{N}(\psi,v)=\begin{bmatrix}v+a & 0\\ \psi_1 v+\psi_2 & v+b\end{bmatrix}$$

مثال۵) اشکال دیگر الگوریتم بخش ۳-۱ آن است که تنها برای سیستم‌های مربعی تعریف شده است. در حالی‌که می‌توان با استفاده از الگوریتم پیشنهادی، مرتبه عناصر قطری ماتریس تداخل را هم برای سیستم‌های مربعی و هم غیرمربعی بدست آورد. سیستم زیر را در نظر می‌گیریم.

$$\boldsymbol{f}(\boldsymbol{x})=\begin{bmatrix}-x_1x_2-x_1x_4-3x_1-x_2-x_4\\ -x_1x_2-x_1-2x_2\\ x_1x_2+x_1+x_2-x_3\\ x_1x_4+x_1\\ x_1x_4+x_1+x_4-x_5\end{bmatrix}, \mathbf{g}(\boldsymbol{x})=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\\ 0 & 0 & 1\\ 0 & 0 & 0\end{bmatrix}$$
$$\boldsymbol{h}(\boldsymbol{x})=\begin{bmatrix}x_3\\ x_5\end{bmatrix}$$
$$\Theta_0(\boldsymbol{x})=\boldsymbol{h}(\boldsymbol{x})=\begin{bmatrix}x_3\\ x_5\end{bmatrix}, \Omega_0(\boldsymbol{x})=\varnothing\ , \rho_0=0$$

تکرار اول $(k=1)$:

$$L_g\Theta_0(\boldsymbol{x})=\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\rightarrow \rho_1=0$$
$$\mathbf{S}_1\in R^{2\times 2}\rightarrow \mathbf{S}_1=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$$
$$\Theta_1(\boldsymbol{x})=\begin{bmatrix}x_1x_2+x_1+x_2-x_3\\ x_1x_4+x_1+x_4-x_5\end{bmatrix}$$

تکرار اول $(k=2)$:

$$L_g\Theta_1(\boldsymbol{x})=\begin{bmatrix}x_2+1 & 1 & 0\\ x_4+1 & 0 & 1\end{bmatrix}\rightarrow \rho_2=2$$
$$\rho_{k^*}=min(p,m)=2\rightarrow k^*=2$$

در نتیجه سیستم دارای دو صفر نامحدود از مرتبه دو می باشد، بنابراین ساختار ماتریس تداخل آن به فرم زیر است:

$$\mathbf{N}(\psi,v)=\begin{bmatrix}v^2+a_1v+b_1 & 0\\ \psi_1v^2+\psi_2v & v^2+a_2v+b_2\end{bmatrix}$$

چندجمله‌ای‌های روی قطر، چندجمله‌ای‌های پایدار دلخواه می‌باشند. چندجمله‌ای غیر قطری، چندجمله‌ای با مرتبه‌ای برابر با مرتبه ماتریس تداخل و حداقل پارامترهای موجود جهت برقرار بودن ویژگی‌های ماتریس تداخل می باشد.

مثال ٦) محاسبه ماتریس تداخل سیستم pH

به منظور بررسی کارایی الگوریتم پیشنهادی، در این مثال این الگوریتم بر روی مدل فرایند خنثی‌سازی pH اعمال می‌گردد. فرآیند pH یک فرآیند غیرخطی است که بسته به مواد شرکت‌کننده در فرآیند و دینامیک مسأله، شدت غیرخطی بودن آن تغییر می‌کند. در سال ۱۹۸۹ Seborg یک فرآیند چند ورودی-چند خروجی pH ارائه کرد و مدل‌سازی کامل از سیستم را انجام داد. معادلات حالت سیستم pH به صورت زیر می باشد[۲۴]:

$$\begin{aligned}\dot{\boldsymbol{x}}&=\boldsymbol{f}(\boldsymbol{x})+\mathbf{g}(\boldsymbol{x})\boldsymbol{u}+\boldsymbol{p}(\boldsymbol{x})\boldsymbol{d}\\ \boldsymbol{y}&=\boldsymbol{h}(\boldsymbol{x})\end{aligned}\qquad (۲۶)$$

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

که ورودی‌های این فرآیند باز (F_b) و بافر (F_{bf})[۱]، خروجی‌های آن pH خروجی از مخزن ترکیب و ارتفاع محلول داخل مخزن و اسید (F_a) به عنوان اغتشاش در نظر گرفته می شود. متغیرهای حالت به صورت رابطه (۲۷) تعریف می‌گردند:

$$\boldsymbol{x} \square \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^T \quad (۲۷)$$

که در آن x_1 غلظت اسید، x_2 غلظت باز و x_3 ارتفاع مایع در مخزن ترکیب را نشان می‌دهند.سایر ماتریس‌های معادلات حالت سیستم pH مطابق رابطه (۲۶) در زیر آمده است:

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} 0 & 0 & \frac{1}{A}(-C_v(x_3)^{0.5} \end{bmatrix}^T$$

$$\boldsymbol{g}(\boldsymbol{x}) = \begin{bmatrix} \frac{1}{Ax_3}(W_{a1} - x_1) & \frac{1}{Ax_3}(W_{a2} - x_1) \\ \frac{1}{Ax_3}(W_{b1} - x_2) & \frac{1}{Ax_3}(W_{b2} - x_2) \\ \frac{1}{A} & \frac{1}{A} \end{bmatrix}^T \quad (۲۸)$$

$$\boldsymbol{p}(\boldsymbol{x}) = \begin{bmatrix} \frac{1}{Ax_3}(W_{a3} - x_1) & \frac{1}{Ax_3}(W_{b3} - x_2) & \frac{1}{A} \end{bmatrix}^T$$

$$\boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} x_3 \\ pH \end{bmatrix}$$

در رابطه (۲۸)، A مساحت سطح مخزن ترکیب استوانه‌ای، W_{ai} و W_{bi} ثابت‌های واکنش برای جریان ام i $(i = acid, base, buffer)$ هستند.. سیستم pH جز سیستم‌هایی می‌باشد که محدودیت سوم بخش ۳-۲ در مورد آن صدق می‌کند. بنابراین برای بدست آوردن ماتریس تداخل این سیستم نمی‌توان از روش Mutoh استفاده کرد. برای اعمال الگوریتم پیشنهادی در این مقاله بر روی این سیستم چند ورودی-چند خروجی، نیاز است خروجی‌های آن به صورت صریح بر اساس متغیرهای حالت در اختیار باشند چرا که باید مشتقات لی خروجی‌ها نسبت به متغیرهای حالت محاسبه گردد. در این سیستم دو خروجی داریم که یکی ارتفاع و دیگری pH می‌باشد. در مورد ارتفاع مشکلی وجود ندارد چراکه این خروجی تابعی از متغیرهای حالت تعریف شده است اما خروجی pH به این شکل نیست. رابطه (۲۹) خروجی pH را به طور ضمنی بر حسب متغیرهای حالت مسئله بدست می‌دهد.

$$c(\boldsymbol{x}, pH) = x_1 + 10^{pH-14} - 10^{-pH} + x_2 \frac{1 + 2 \times 10^{pH - pK_{a2}}}{1 + 10^{pKa_1 - pH} + 10^{pH - pK_{a2}}} \quad (۲۹)$$

$$c(\boldsymbol{x}, pH) = 0$$

مشاهده می‌شود که در این رابطه خروجی pH به صورت صریح بر حسب متغیرهای حالت بیان نشده است، درحالی که در مراحل بعد برای محاسبه ماتریس تداخل نیاز به چنین رابطه صریحی است. در این مقاله برای بدست آوردن pH به صورت تابع صریحی از متغیرهای حالت، از شبکه عصبی چندجمله‌ای (PNN)[۲] استفاده شده است. این شبکه عصبی مبتنی بر تئوری روش گروهی برای مدیریت داده‌ها (GMDH)[۳] [۲۵] است که در آن رابطه بین ورودی و خروجی به صورت ترکیبات متوالی از چند جمله‌ای‌های با مرتبه محدود مدل می‌شود. در PNN با انتخاب تعداد ورودی‌ها و تابع مناسب برای هر گره و اضافه کردن لایه‌ها می‌توان به بهترین مدل دست یافت. تابع عملکرد هر گره به نام توصیف جزئی (PD)[۴] شناخته می‌شود که معمولا توابع چند جمله‌ای از درجات پایین هستند. آموزش شبکه PNN در دو مرحله صورت می‌گیرد. در مرحله یادگیری، هر لایه دارای تعدادی PD است که ورودی‌های آنها از خروجی‌های لایه قبل ساخته می‌شود. ضرایب جملات PDهای آخرین لایه به صورتی تنظیم می‌شود که هر کدام بتوانند بهترین تخمین ممکن را از خروجی ارائه دهند. این کار درحالی صورت می‌گیرد که ضرایب لایه‌های قبل از آن بدون تغییر باقی می‌مانند. در صورتی که خروجی هیچ کدام از PDهای لایه آخر تخمین خوبی از خروجی ارائه ندهند، لایه جدیدی به شبکه اضافه می‌شود. افزایش لایه‌ها تا جایی ادامه پیدا می‌کند که یکی از خروجی‌های آخرین لایه تخمین قابل قبولی از خروجی اصلی ارائه دهد. پس از آن در مرحله حذف، تمامی PDهایی که در ساختن این خروجی نقشی نداشته‌اند حذف می‌شوند و تنها PDهای موثر بر آن باقی می‌مانند. خروجی نهایی با ترکیب روابط PDهابه صورت یک تابع چند جمله‌ای از ورودی‌ها می‌تواند نوشته شود. بر این اساس، با فرض رابطه ورودی و خروجی به صورت رابطه (۳۰)، می‌توان خروجی تخمین‌زده شده را مطابق رابطه (۳۱) در نظر گرفت:

$$y = f(x_1, x_2, \ldots, x_N) \quad (۳۰)$$

$$\hat{y} = \hat{f}(x_1, x_2, \ldots, x_N) = c_0 + \sum_{k_1} c_{k_1} x_{k_1} + \sum_{k_1 k_2} c_{k_1 k_2} x_{k_1} x_{k_2} + \sum_{k_1 k_2 k_3} c_{k_1 k_2 k_3} x_{k_1} x_{k_2} x_{k_3} + \ldots \quad (۳۱)$$

که در آن c ها ضرایب مدل می‌باشند. الگوریتم کامل بدست آوردن این شبکه در [۲۶] ارائه شده است.

همان‌گونه که از رابطه (۲۹) پیداست، pH به هر سه متغیر حالت وابسته است. البته این نکته طبیعی است که غلظت اسید و باز تأثیر بیشتری روی مقدار pH دارند. برای تولید شبکه PNN، در ابتدا ورودی‌های لایه

[۱] Buffer

۲ Polynomial Neural Network (PNN)

۳ Group Method of Data Handling

۴ Partial Description

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

اول شبکه عصبی چندجمله‌ای x_1 ، x_2 و x_3 در نظر گرفته شد. اما خطای مدلسازی علی‌رغم افزایش لایه‌ها کماکان زیاد بود. علت آن این بود که طبق رابطه (۲۹)، رابطه بین متغیرها دارای جملات نمایی است که به خوبی توسط چندجمله‌ای‌ها بیان نمی‌شوتد. بنابراین برای کمتر شدن خطای شناسایی سه ورودی دیگر نیز به صورت e^{x_1} ، e^{x_2}، e^{x_3} به آنها اضافه گردید:

$$u=\begin{bmatrix} x_1 & x_2 & x_3 & e^{x_1} & e^{x_2} & e^{x_3} \end{bmatrix}$$

به این ترتیب، خروجی pH برحسب ورودی‌های مطرح شده زمانی که تعداد لایه‌های شبکه عصبی به ۳ رسید، با دقت خوبی تخمین زده شد. با حذف PDهای غیر موثر، شبکه نهایی بدست آمده دارای ۱۱ عدد PD با درجات یک و دو بود. لازم به ذکر است که در شبیه سازی‌ها ارتفاع و pH به همراه نویز اندازه‌گیری در نظر گرفته شد. نویز اندازه‌گیری، نویز سفید با میانگین صفر و واریانس 0.01 فرض گردید. با استخراج تابع کلی عملکرد شبکه به صورت رابطه (۳۱)[1]، از آن در رابطه (۲۶) به جای مدل خروجی استفاده گردید. مزیت رابطه جدید این است که اولا خروجی pH را به صورت صریح بر اساس متغیرهای حالت ارائه می‌دهد و ثانیا به علت استفاده از ترکیب چندجمله‌ای، محاسبات مربوط به مشتقات لی در آن به سادگی انجام می‌شود. این رابطه به صورت صریح تابع $\mathbf{\Lambda}(x)$ مطابق رابطه (۱۰) را به صورت یک ماتریس 2×2 ناویژه بدست می‌دهد. بر اساس این رابطه می توان به صورت صریح مشتقات لی را بدست آورده و ماتریس تداخل را طبق الگوریتم پیشنهادی محاسبه کرد.

گام اول) مرتبه عناصر قطری ماتریس تداخل با استفاده از الگوریتم بخش ۳-۲ محاسبه می‌گردند.

برای این کار از معادلات فضای حالت سیستم pH یعنی رابطه (۲۶) و رابطه تولید شده توسط شبکه عصبی استفاده شده است :

$$\Theta_0(x)=\boldsymbol{h}(x)=\begin{bmatrix} x_3 \\ pH \end{bmatrix},\Omega_0(x)=\varnothing,\rho_0=0$$

تکرار اول $(k=1)$:

$$rank\left(L_g\Theta_0(x)\right)=2$$

از آنجایی‌که $\rho_1=min(p,m)$ الگوریتم خاتمه می‌یابد. به این ترتیب این سیستم دو صفر نامحدود از مرتبه یک دارد.

گام دوم- ساختار ماتریس تداخل با استفاده از الگوریتم بخش ۳-۱ تعیین می‌شود.

$$\mathbf{N}(\psi,v)=\begin{bmatrix} v+a & 0 \\ \psi_1 v+\psi_2 & v+c \end{bmatrix}$$

گام سوم) در این گام باید پارامترهای مجهول ماتریس تداخل شناسایی شوند. چندجمله‌ای‌های روی قطر ماتریس تداخل به صورت دلخواه انتخاب می‌شوند. یعنی تنها کافی است یک چندجمله‌ای پایدار باشند. بنابراین فقط در عنصر غیرقطری پارامترهای مجهول ψ_1 و ψ_2 وجود دارد که باید شناسایی شوند. این کار با استفاده از روشی که به تفصیل در [۱۹] آورده شده است، انجام می‌شود. پارامترهای عنصر غیرقطری ماتریس تداخل به صورت زیر بدست می‌آیند:

$$\psi_1=-0.1735 \quad , \quad \psi_2=0$$

بنابراین ماتریس تداخل در نهایت به فرم زیر محاسبه می‌شود:

$$\mathbf{N}(v)=\begin{bmatrix} v+1 & 0 \\ -0.1735v & v+2 \end{bmatrix}$$

اطمینان از صحت ماتریس تداخل بدست آمده برای سیستم pH از دو راه ممکن است:

۱- بررسی رابطه (۱۰) با ماتریس تداخل بدست آمده.

۲- استفاده از روش خطی و بررسی رابطه (۱).

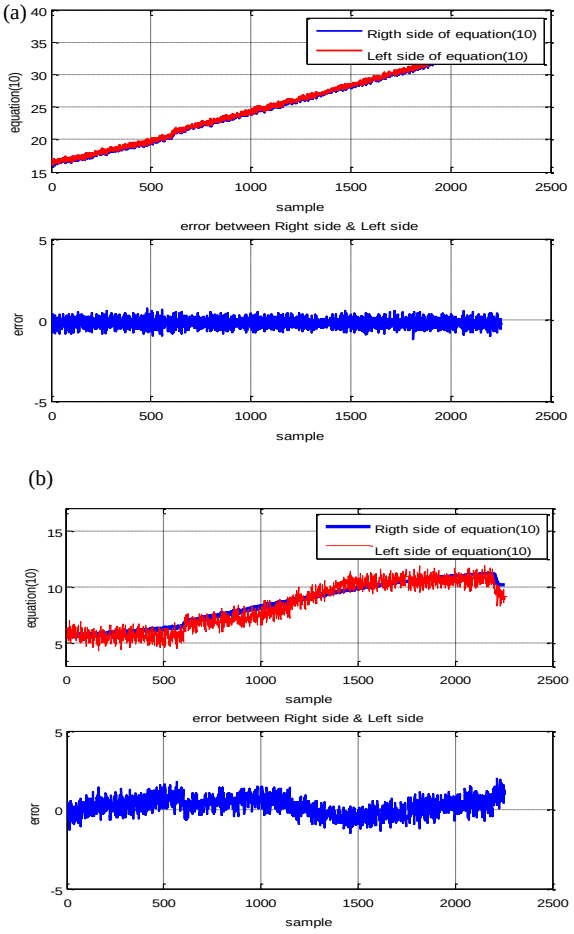
همان‌گونه که در بخش ۲-۲ ذکر گردید، برای اطمینان از اینکه ماتریس تداخل درست بدست آمده است باید رابطه (۱۰) برای آن برقرار باشد. جهت بررسی این نکته از شبیه‌سازی رابطه (۱۰) مبتنی بر مدل فرایند و ماتریس تداخل بدست آمده استفاده می‌شود. برای ماتریس تداخل $\mathbf{N}(v)$ بدست آمده، قسمت سمت چپ و سمت راست رابطه (۱۰) به صورت مجزا برای خروجی‌ها، ورودی‌ها و حالت‌های سیستم pH مطابق با آنچه که در رابطه (۲۸) ذکر گردید محاسبه شد. برای این منظور، محاسبه $\mathbf{\Lambda}(x)$ و $\boldsymbol{a}(x)$ با استفاده از رابطه خروجی pH که از شبکه عصبی PNN حاصل گردید، انجام شده است. چون سیستم pH دو ورودی و دو خروجی دارد و ماتریس تداخل بدست آمده نیز 2×2 است، بنابراین رابطه (۱۰) به دو رابطه تبدیل می‌شود که به ترتیب در شکل ۱ قسمت (a) و قسمت (b) نشان داده شده است. همان-گونه که در شکل ۱ مشاهده می‌شود دو طرف رابطه (۱۰) در تمامی زمان‌ها با دقت قابل قبولی برابر هم هستند که نشان می‌دهد رابطه (۱۰) در تمام زمان‌ها برقرار است.

ارزیابی دیگر براساس رابطه (۱) است. برای اینکه رابطه (۱) در مورد ماتریس تداخل بدست آمده برای سیستم pH تحقیق شود، نیاز به معادلات خطی شده این سیستم است . تابع تبدیل سیستم pH در یک نقطه کار مشخص(pH=7.8 و h=16) به فرم زیر بدست می‌آید:

$$\mathbf{G}(z)=\begin{bmatrix} \dfrac{0.15}{z-0.84} & \dfrac{0.15}{z-0.84} \\ \dfrac{0.2}{z-0.7} & \dfrac{0.64}{z-0.7} \end{bmatrix}$$

[1] با توجه به طولانی بودن این رابطه برای شبکه بدست آمده، از ذکر آن خودداری شده است.

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]



شکل ۱: بررسی صحت رابطه (۱۰) برای ماتریس تداخل بدست آمده سیستم pH

حال اگر ماتریس تداخل را با زمان نمونه‌برداری ۳۰ ثانیه به حالت گسسته ببریم، $\mathbf{D}(z)$ حاصل می‌شود:

$$\mathbf{D}(z)=\begin{bmatrix} z & 0 \\ 0 & 2z \end{bmatrix}$$

برای تعیین تابع تبدیل بدون تأخیر $\tilde{\mathbf{G}}(z)$ باید ماتریس تداخل $\mathbf{D}(z)$ بدست آمده در تابع تبدیل $\mathbf{G}(z)$ ضرب شود و در نهایت رابطه (۱) در مورد آن بررسی گردد:

- $\left|\mathbf{D}(z)\right|=2z^2$، که ۲ نشان‌دهنده تعداد صفرهای نامحدود تابع تبدیل می‌باشد.
- $\lim_{z\to\infty}\tilde{\mathbf{G}}(z)=\mathbf{K}=\begin{bmatrix} 0.15 & 0.15 \\ 0.4 & 1.28 \end{bmatrix}$

همان‌گونه که مشاهده می‌شود، $\mathbf{K}$ یک ماتریس رتبه کامل است. بنابراین $\mathbf{D}(z)$ همان ماتریس تداخل است.

۵- نتیجه گیری

در این مقاله روشی جدید برای بدست آوردن ماتریس تداخل یک سیستم غیرخطی چند ورودی-چند خروجی ارائه گردید. الگوریتم موجود در این زمینه (Mutoh) [۱۹] دارای ۲ قسمت است. قسمت اول، ساختار ماتریس تداخل را با استفاده از مشتقات لی برای سیستم غیرخطی چند ورودی-چند خروجی بدست می‌آورد و در قسمت دوم پارامترهای مجهول این ماتریس شناسایی می‌گردند. با ذکر چند مثال در این مقاله نشان داده شد که قسمت اول این الگوریتم دارای محدودیت‌هایی است و تنها برای کلاس خاصی از سیستم‌های چند ورودی-چند خروجی یعنی سیستم‌های مربعی کاربرد دارد. در الگوریتم پیشنهادی در این مقاله، قسمت اول الگوریتم Mutoh بهبود داده شد و ساختار ماتریس تداخل با بکارگیری تعریف مرتبه ماتریس تداخل، به وسیله الگوریتم ساختار صفر نامحدود بدست آمد و سپس کارایی این الگوریتم با ذکر چند مثال و هم‌چنین در نظرگرفتن فرایند pH به عنوان یک سیستم واقعی مورد بررسی قرارگرفت و نشان داده شد که این الگوریتم قادر است برای کلاس وسیع‌تری از سیستم‌های چند ورودی-چند خروجی غیرخطی اعم از مربعی و غیرمربعی ماتریس تداخل را محاسبه کند. با این وجود، برای انجام این کار نیاز است خروجی‌های سیستم غیرخطی به صورت صریح بر حسب متغیرهای حالت قابل بیان باشند. ضمن آنکه، الگوریتم پیشنهادی در این مقاله، همانند سایر الگوریتم‌های موجود، تنها برای سیستم‌های غیرخطی افاین، ماتریس تداخل خطی بدست می‌دهد.

مراجع

[1] W. A.Wolovich, and P. L. Falb, "Invariants and Canonical Forms under Dynamic Compensation," SIAM Journal on Control and Optimization, vol. 14, pp. 996-1008, 1976.

[2] B. Huang, and S. L Shah, Performance Assessment of Control Loops: Theory and Applications: Springer London, 1999.

[3] K. Zhang, Y. Zhu, and B. Huang, "MV benchmark estimation based on high-frequency test signal," Journal of Process Control, vol. 47, pp. 35-45, 2016.

[4] M. Jelali, Control Performance Management in Industrial Automation: Assessment, Diagnosis and Improvement of Control Loop Performance: Springer London, 2012.

[5] L. Wen, G. Tao, H. Yang, and Y. Yang, "Aircraft Turbulence Compensation Using Adaptive Multivariable Disturbance Rejection Techniques," Journal of Guidance, Control, and Dynamics, vol.38 , pp. 954-963, 2014.

[6] S. Cheng, Y. Wei, Y. Chen, Y. Wang, and Q. Liang, "Fractional-order multivariable composite model reference adaptive control," International Journal of Adaptive Control and Signal Processing, 2017.

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]

[7] G. Tao,"Multivariable adaptive control: A survey," Automatica, vol. 50, pp. 2737-2764, 2014.

[8] W. Kase, and Y. Shigehiro, "Pseudo innerizing control by state feedback," In: 42nd Annual Conference of the IEEE Industrial Electronics Society, pp. 288-293, 2016.

[9] M. Rogozinski, A. Paplinski, and M. Gibbard, "An algorithm for the calculation of a nilpotent interactor matrix for linear multivariable systems," IEEE Transactions on Automatic Control, vol. 32, pp. 234-237, 1987.

[10] W. Kase, "A simple derivation of lower triangular interactor matrix," Proc. of the 4th WSEAS/IASME international conference on System science and simulation in engineering, Tenerife, Spain, 2005.

[11] M. Kamrunnahar, et al., "Estimation of Markov parameters and time-delay/interactor matrix," Chemical Engineering Science, vol. 55, pp. 3353-3363, 2000.

[12] W. Kase, and K. Tamura, "Design of G-interactor and its application to direct multivariable adaptive control," International Journal of Control, vol. 51, pp. 1067-1088, 1990.

[13] W. Kase, and Y. Mutoh, "A simple derivation of the interactor matrix and its applications," International Journal of Systems Science, vol. 40, pp. 1197-1205, 2009.

[14] W. Kase, "A Simple Derivation of Right Interactor for Tall Transfer Function Matrix and its Application to Inner-Outer Factorization," IEEJ Transactions on Electronics, Information and Systems, vol. 131, pp. 1608-1615, 2011.

[15] R. Ortega, J. M. Dion, F. J. Carrillo, and L. Dugaro, "A Globally Stable Multivariable Adaptive Controller with Reduced Prior Knowledge," In: American Control Conference, pp. 418-422, 1985.

[16] Y. Mutoh, and R. Ortega,"Interactor structure estimation for adaptive control of discrete-time multivariable nondecouplable systems," Automatica, vol. 29, pp. 635-647, 1993.

[17] L. Dugard, G. Goodwin, and C. D. Souza, "Prior knowledge in model reference adaptive control of multiinput multioutput systems," IEEE Transactions on Automatic Control, vol .29 ,pp. 761-764, 1984.

[18] J. Guo, and G. Tao, "A discrete-time multivariable MRAC scheme applied to a nonlinear aircraft model with structural damage," Automatica, vol. 53, pp. 43-52, 2015.

[19] Y. Mutoh,"Design of Model Reference Adaptive Control for Nonlinear Multivariable Systems," IEEJ Transactions on Electronics, Information and Systems, vol. 129, pp. 1070-1076, 2009.

[20] W. Kase, "Inner-outer factorization for continuous-time systems using interactor matrix," In: 9th IEEE International Conference on Control and Automation (ICCA), pp. 330-335, 2011.

[21] G. Tao, "Model reference adaptive control of multivariable plants with unknown interactor matrix," Proc. of the 29th IEEE Conference on Decision and Control, pp. vol.5, 2730-2735, 1990.

[22] K. Sugimoto, L. Yi, and A. Inoue, "Parametrization of identity interactors and the discrete-time all-pass property," In: Proceedings of the American Control Conference, vol.6, pp. 4403-4407, 1995.

[23] X. Liu, and Z. Lin,"On Normal Forms of Nonlinear Systems Affine in Control," IEEE Transactions on Automatic Control, vol. 56, pp. 239-253, 2011.

[24] R. Hall, C. and D. E. Seborg, "Modelling and Self-Tuning Control of a Multivariable pH Neutralization Process Part I: Modelling and Multiloop Control," In: American Control Conference, 1989, pp. 1822-1827, 1989.

[25] A. G. Ivakhnenko,"Polynomial Theory of Complex Systems," IEEE Transactions on Systems, Man, and Cybernetics ,vol. SMC-1, pp. 364-378, 1971.

[26] H. Sadjadian, H. Taghirad, and A. Fatehi, "Neural networks approaches for computing the forward kinematics of a redundant parallel manipulator," International Journal of Computational Intelligence, vol. 2, pp. 40-47, 2005.

[Downloaded from joc-isice.ir on 2026-11-09]
[DOR: 20.1001.1.20088345.1397.12.2.4.0]
[DOI: 10.29252/joc.12.2.1]