

Data-Driven Fault-Tolerant Model Predictive Control with Online Adaptation Based on Dynamic Mode Decomposition

Mohammadhosein Bakhtiaridoust¹, Meysam Yadegar^{1*}, Mohammad Hadi Rezaei²

¹Department of Electrical and Computer Engineering, Qom University of Technology, Qom, Iran

²Department of Electrical Engineering, Yazd University, Yazd, Iran

E-mails: bakhtiaridoust.mh@qut.ac.ir; yadegar@qut.ac.ir, hadi.rezaei@yazd.ac.ir

Abstract

This paper introduces a new data-driven control method for handling faults in multi-input multi-output linear systems. In this fault-tolerant control approach, a virtual actuator based on model predictive control is designed, utilizing an adaptive model extracted from the system's dynamic modes. This extracted model is recursively updated and utilized to predict the states of the faulty system. The method considers the effect of faults in predictive controller predictions and minimizes this effect. The proposed method does not require knowledge of system equations and it is fully data-driven. Moreover, since it utilizes predictive controller for improving system performance in the presence of faults, it can impose constraints on control inputs. Furthermore, the method presented in this paper is designed based on a virtual actuator and can easily be augmented to closed-loop systems to enhance their performance in dealing with faults. Finally, the performance of the proposed control method is examined through a simulation example.

Keywords

Fault-tolerant control, Dynamic mode decomposition, Model predictive control, Data-driven control, Virtual actuator.

Introduction

The necessity of fault-tolerant control (FTC) arises from the fact that faults, failures, and anomalies are inevitable in complex engineering systems. These systems can range from critical infrastructure like power grids, aerospace vehicles, and industrial processes to everyday technologies such as automobiles and consumer electronics. The primary goal of fault-tolerant control is to prevent or mitigate the adverse effects of faults, which can lead to reduced efficiency, safety hazards, or even catastrophic failures.

Proposed Work and Methodology

This paper works toward designing a data-driven virtual actuator that is placed between the plant and the already designed and implemented controller such that the actuator faults are hidden from the system's controller and the system's performance under normal and faulty conditions remains almost identical. To this end, a reconfiguration block, which employs a MPC controller, is added to the nominal closed-loop system to make it fault-tolerant with respect to the actuator faults. The approximation of the fault-free trajectory of the system is generated using the estimated dynamic mode decomposition (DMD) based predictor, which is employed as the reference model inside the virtual actuator. It is noteworthy that the DMD model is estimated from directly from the system data and the method assume no knowledge the underlying dynamics. Then the predictive controller try to minimize the error between the faulty plant and the DMD reference model leveraging the estimated input matrix. It should be noted that the MPC allows for imposing constraint so the obtained control law satisfy those conditions. Finally, the simulation results are provided to assess the effectiveness of the proposed method.

Conclusion

In this study, a novel data-driven fault-hiding strategy for fault-tolerant control of linear dynamical systems is proposed. The approach combines dynamic mode decomposition with a model predictive control framework to achieve effective fault tolerance. The central idea is to use a time-varying predictor to obtain an interpretable linear representation of the faulty dynamics. The proposed method integrates the fault effect by iteratively updating the predictor model. The efficacy of the presented fault-tolerant approach is demonstrated through case study results. Additionally, the framework could be extended to consider scenarios where only system outputs are accessible, rather than the system states.

کنترل پیش‌بین تحمل‌پذیر عیب داده‌محور با قانون تطبیق بر خط مبتنی بر تجزیه به مدهای دینامیکی

محمدحسین بختیاری دوست

دانشجوی کارشناسی ارشد، دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی قم، قم، ایران

میثم یادگار

دانشیار، دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی قم، قم، ایران

محمد هادی رضایی

استادیار، دانشکده مهندسی برق، دانشگاه یزد، یزد، ایران

چکیده

در این مقاله یک روش کنترل داده محور برای سیستم‌های دینامیکی خطی چند ورودی-چند خروجی برای مقابله با عیب ارائه شده است. در این روش کنترل تحمل پذیر عیب، یک عملکرد مجازی بر مبنای کنترل پیش‌بین و با استفاده از مدل تطبیقی استخراج شده از مدهای دینامیکی سیستم، طراحی شده است. این مدل استخراج شده، به صورت بازگشتی به روز و به منظور پیش‌بینی حالات سیستم معیوب استفاده شده است. این روش، اثر عیب در پیش‌بینی‌های کنترل‌کننده پیش‌بین در نظر گرفته و توسط کنترل‌کننده کمینه شده است. در این روش نیازی به دانستن معادلات نیست و از آنجایی که در این روش از کنترل‌کننده پیش‌بین برای بهبود عملکرد سیستم در مواجهه با عیب استفاده شده است، توانایی اعمال محدودیت بر روی ورودی‌های کنترلی وجود دارد. همچنین روش ارائه شده در این مقاله بر مبنای عملکرد مجازی طراحی شده است و به راحتی می‌تواند به سیستم‌های حلقه بسته به منظور بهبود عملکرد این سیستم‌ها در مواجهه با عیب اضافه شود. در نهایت، عملکرد روش کنترلی ارائه شده در قالب یک مثال شبیه‌سازی مورد بررسی قرار گرفته است.

کلمات کلیدی

کنترل تحمل پذیر عیب، تجزیه به مدهای دینامیکی، کنترل مدل پیش‌بین، کنترل داده محور، عملکرد مجازی.

نام نویسنده مسئول: دکتر میثم یادگار

ایمیل نویسنده مسئول: yadegar@qut.ac.ir

تاریخ ارسال مقاله: ۱۴۰۳/۰۲/۰۱

تاریخ(های) اصلاح مقاله: ۱۴۰۳/۰۶/۰۷

تاریخ پذیرش مقاله: ۱۴۰۳/۰۷/۰۷

۱- مقدمه

حساس نباشد. این روش‌ها به داشتن اطلاعات عیب‌هایی که احتمال وقوع آن‌ها وجود دارد، نیاز دارند. از این رو، اینگونه روش‌ها اصولاً محدودکننده‌تر هستند. اینگونه استراتژی‌ها معمولاً با استفاده از کنترل مقاوم و روش‌های پایداری همزمان^۳ قابل دستیابی اند [۴-۵]. ویژگی اصلی این رویکردها این است که نیازی به یک واحد تشخیص و جداسازی عیب^۴ (FDI) بی‌درنگ^۵ ندارند و هنگام وقوع عیب‌ها، هیچ تغییری در کنترل‌کننده ایجاد نمی‌کنند. این روش‌ها ممکن است همیشه عملکرد خوبی نداشته باشند؛ زیرا همه عیب‌هایی که ممکن است بر سیستم تأثیر بگذارند، از پیش، جزو عیب‌های ساختاری شناخته شده مورد انتظار طراح نیستند.

در مرجع [۶]، نویسندگان یک رویکرد بر مبنای کنترل مدل پیش‌بین اقتصادی را پیشنهاد داده‌اند که یک تابع شاخص ایمنی را به عنوان یک محدودیت سخت^۶ برای ایجاد یک منطقه عملیاتی امن برای سیستم تعیین می‌کند. در مرجع [۷]، یک فرمول‌بندی بر اساس کنترل مدل پیش‌بین مبتنی بر لیاپانوف به منظور ایجاد یک کنترل‌کننده تحمل پذیر عیب پیشنهاد شده

ضرورت استفاده از کنترل تحمل پذیر عیب^۱ (FTC) از آنجا ناشی می‌شود که در سیستم‌های پیچیده مهندسی، بروز عیب‌ها امری اجتناب ناپذیر است. این سیستم‌ها می‌توانند شامل زیرساخت‌های حیاتی مانند شبکه‌های برق، فرآیندهای صنعتی و همچنین فناوری‌های مورد استفاده روزمره از قبیل خودروها باشند. هدف اصلی کنترل تحمل پذیر عیب، جلوگیری یا کاهش اثرات منفی عیب‌هایی است که می‌توانند منجر به کاهش کارایی، ایمنی و بروز خسارت‌های جبران‌ناپذیر شوند [۱-۲]. با توجه به ذات مقاوم به اغتشاش و توانایی آن در اعمال محدودیت‌های کنترلی، روش‌های کنترل تحمل پذیر عیبی که بر پایه کنترل مدل پیش‌بین^۲ (MPC) متمرکز اند، توجه زیادی به خود جلب کرده‌اند و به صورت گسترده استفاده شده‌اند [۳]. این روش‌ها اساساً به دو دسته غیرفعال و فعال تقسیم می‌گردند.

در رویکردهای غیرفعال سعی شده است تا قانون کنترلی به نحوی طراحی شود که نسبت به برخی از عیب‌ها که دارای ساختاری شناخته شده می‌باشند،

Fault detection and isolation unit^۴Real-time^۵Hard constraint^۶Fault-tolerant control^۱Model predictive control^۲Simultaneous stabilization^۳

پنهان کند و در صورت رخ دادن عیب‌های عملگر یا حسگر، پایداری عملکرد سیستم را تضمین کند. بلوک بازتنظیم پس از کنترل‌کننده قرار می‌گیرد و شامل یک عملگر مجازی و/یا یک حسگر مجازی است. از جمله کارهای مرتبط در این حوزه می‌توان به [۱۷-۱۹] اشاره کرد.

نقطه ضعف اصلی روش‌های فعال و غیرفعال مذکور این است که کارآمدی آن‌ها بیشتر بر پایه داشتن یک فهم کامل از معادلات سیستم می‌باشد، که بدست آوردن آن‌ها اغلب در عمل برای سیستم‌های پیچیده یا امکان‌پذیر نیست و یا کاری بسیار پرهزینه می‌باشد. با در نظر گرفتن این موضوع و از آنجایی که امروزه سیستم‌های صنعتی مدرن معمولاً شامل سیستم‌های نظارتی و جمع‌آوری داده‌ای هستند که دسترسی به داده‌های فرآیند را تسهیل می‌کنند، محققان به سمت رویکردهای FTC مبتنی بر داده ترغیب شده‌اند.

در مرجع [۱۸]، یک روش FTC بر اساس یادگیری تقویتی متا^۷ (meta-RL) ارائه شده است تا دقت ردیابی وسایل نقلیه را در مواجهه با عیب‌های عملگر بهبود دهد. در مرجع [۱۹]، یک عملگر مجازی تطبیقی بر اساس تقریب دهی از اپراتور کوپمن^۸ طراحی شده است. در مرجع [۲۰]، یک کنترل‌کننده ردیاب FTC برای سیستم‌های خطی با پارامترهای متغیر^۹ (LPV) با زمان گسسته با یک ورودی و یک خروجی و تحت عیب‌های ناگهانی^{۱۰} و تدریجی^{۱۱} عملگر^{۱۱}، با استفاده از رویکرد مبتنی بر داده، ارائه شده است. در مرجع [۲۱]، یک روش FTC بر اساس یادگیری تقویتی غیرمبتنی بر خط مشی^{۱۲} پیشنهاد شده است تا مشکلات کنترل ردیابی مقاوم در برابر عیب را حل کند. در مرجع [۲۲]، یک روش تحمل‌پذیر عیب داده محور تطبیقی بر پایه تخمین بازگشتی اپراتور کوپمن و کنترل پیش‌بین ارائه شده است. در مرجع [۲۳]، نویسندگان ابتدا یک مدل خطی‌سازی پویای کامل^{۱۳} از داده‌های سیستم را به دست می‌آورند و سپس با استفاده از یک شبکه عصبی غیرخطی خودهمبسته^{۱۴}، متغیر حسگر را تخمین زده و از آن برای تشکیل قانون کنترل مقاوم در برابر عیب استفاده کرده‌اند. در مرجع [۲۴]، یک روش FTC مکانیکی و الکتریکی تطبیقی داده محور معرفی شده است که هدف آن کاهش اثرات ناشی از شکستن میله روتور توربین بادی است. نویسندگان در مرجع [۲۵]، یک روش FTC بر اساس تحلیل داده را برای کنترل ردیابی شتاب وسایل پرنده بدون سرنشین با در نظر گرفتن عدم قطعیت‌های ناشی از آسیب‌های آیرودینامیکی ارائه داده‌اند. علاوه بر این، در مرجع [۲۶]، نویسندگان بر روی طراحی یک کنترل‌کننده FTC مبتنی بر داده برای سیستم‌های گسسته زمان متمرکز شده‌اند که با در نظر گرفتن اطلاعات ورودی کنترل در یک پنجره زمانی متحرک در مدل داده، قانون کنترل FTC را به دست می‌آورند. رویکرد ارائه شده در مرجع [۲۷]، یک روش FTC تطبیقی و بدون مدل را معرفی می‌کند که از یک مدل داده محور خطی‌سازی پویا^{۱۵} برای ساده‌سازی دینامیک سیستم و از یک شبکه عصبی تابع پایه شعاعی^{۱۶} برای تقریب تابع عیب استفاده کرده است. در مرجع [۲۸]، یک تکنیک مدل‌سازی متمرکز بر داده را معرفی شده است که از چندین مدل محلی خود

است که با در نظر گرفتن عدم قطعیت در مدل، هدف پایداری تحمل‌پذیر عیب را دنبال می‌کند.

در سوی دیگر روش‌های فعال قرار دارند. در این روش‌ها، کنترل‌کننده به صورت بی‌درنگ تنظیم شده است تا خللی در عملکرد سیستم حلقه بسته، در صورت وقوع عیب، ایجاد نشود. این نوع از روش‌ها خود نیز می‌توانند بر اساس نیازشان به واحد FDI، به دو دسته تقسیم شوند. دسته اول برای انجام FTC به شناسایی نوع، مقدار و مکان عیب برای پایدار کردن سیستم و حفظ آن در شرایط عملکردی معین، وابسته است. از روش‌های وابسته به FDI می‌توان به کنترل مدل پیش‌بین چندگانه اشاره کرد. در این نوع روش‌ها، واحد کنترل با جابجایی^۱ بین کنترل‌کننده‌های از پیش طراحی شده، در زمان مواجهه با عیب، سیستم را در شرایط عملکردی معین نگه می‌دارد. دسته دوم روش‌های فعال، به صورت خودکار سعی در تخمین عیب دارند، لذا نیازی به استفاده از واحد FDI مستقل ندارد. از متداول‌ترین این نوع روش‌ها می‌توان به کنترل تطبیقی MPC تحمل‌پذیر عیب اشاره نمود. لازم به ذکر است که اگرچه روش‌های نوع دوم نیاز به به کارگیری واحد FDI را برطرف کرده‌اند اما کماکان نیاز به تخمین عیب و داشتن دانش دقیقی از دینامیک سیستم برای کاهش اثرات عیب وجود دارد. در مرجع [۸]، یک روش کنترل FTC را ارائه شده است که از یک کنترل مدل پیش‌بین چندگانه کنترل سرعت یک موتور دیزل دریایی استفاده می‌کند. در مرجع [۹]، سیستم FTC جدیدی بر مبنای سیستم کنترل مبتنی بر مدل چندگانه را معرفی شده است که شامل یک سیستم شناسایی وضعیت عملکرد با یک مجموعه کنترل‌کننده تحمل‌پذیر عیب مبتنی بر MPC می‌باشد. مسئله تخصیص کنترل^۲ در مرجع [۱۰] با در نظر گرفتن ناحیه امکان^۳ واحد تخصیص کنترل به عنوان یک محدودیت متغیر با زمان MPC حل شده است. در مرجع [۱۱]، یک طرح MPC فعال بر اساس ادغام جبران بازخوردی^۴ در MPC غیرخطی و استفاده از تخمین عیب مبتنی بر فیلتر کالمن توسعه یافته برای توانایی مقابله با عیب‌های سیستم پیشنهاد شده است. نویسندگان در مرجع [۱۲] یک کنترل‌کننده مدل پیش‌بین آنلاین را پیشنهاد داده‌اند که در آن از یک مشاهده گر حالت لغزشی تطبیقی برای تخمین عیب‌های سیستم و به دست آوردن قانون کنترل تحمل‌پذیر عیب استفاده می‌شود.

تمامی روش‌های ذکر شده در بالا، به منظور مقابله با عیب در سیستم، نیازمند تغییر ساختار واحد کنترلی، چه با جابجایی بین کنترل‌کننده‌های از پیش طراحی شده و چه با تطبیق مداوم پارامترهای سیستم، می‌باشند. لذا ترکیب کردن این روش‌ها با سیستم‌های حلقه بسته، که دارای کنترل‌کننده از پیش طراحی شده می‌باشند، با هدف ارتقا عملکرد آن‌ها در بسیاری از موارد ناممکن و یا کاری پرهزینه است. یکی از رویکردهای قابل توجه در حوزه استراتژی‌های FTC فعال، تکنیک مخفی کردن عیب^۵ است. این روش شامل یک بلوک بازتنظیمی^۶ است که هدف آن دستیابی به مشخصه تحمل‌پذیر بودن عیب، بدون ایجاد تغییر در واحد کنترل‌کننده اصلی سیستم است. مسئولیت بلوک بازتنظیمی این است که عیب‌ها را از دید کنترل‌کننده اصلی سیستم

Linear parameter varying^۹Abrupt fault^{۱۰}Gradual fault^{۱۱}Off-policy^{۱۲}Full form dynamic linearization model^{۱۳}Auto-regressive^{۱۴}Dynamic linearization^{۱۵}Radial basis function^{۱۶}Switch^۱Control allocation^۲Feasibility region^۳Feedback compensation^۴Fault hiding^۵Reconfiguration block^۶Meta reinforcement learning^۷Koopman operator^۸

طراحی شده، بر خلاف روش موجود در مرجع [۱۹]، توانایی اعمال محدودیت بر روی ورودی‌ها را دارا می‌باشد. همچنین بر خلاف روش ارائه شده در مرجع [۱۸]، نیازی به داشتن داده‌هایی از سیستم معیوب نداریم و بر خلاف روش ارائه شده در مرجع [۲۲]، این روش می‌تواند بدون تغییر واحد کنترلی به سیستم حلقه بسته اضافه شود و عملکرد آن را در مواجهه با عیب ارتقا دهد. بر اساس بحث‌های انجام شده، دستاوردهای مقاله را می‌توان به صورت زیر خلاصه کرد:

- ✓ در این کار، یک روش FTC مبتنی بر داده ارائه شده است که نیازی به مدل سیستم و داده‌های سیستم معیوب ندارد. همچنین در این روش با استفاده از یک مدل تطبیقی نیاز به به شناسایی چند مدل مختلف و طراحی چند کنترل‌کننده نمی‌باشد.
- ✓ روش ارائه شده در این مقاله بر مبنای عملکرد مجازی طراحی شده است و به راحتی می‌تواند به سیستم‌های حلقه بسته به منظور بهبود عملکرد این سیستم‌ها در مواجهه با عیب اضافه شود. همچنین به دلیل استفاده از کنترل‌کننده پیش‌بین، در این روش توانایی اعمال محدودیت بر روی ورودی‌های کنترلی سیستم وجود دارد.
- ✓ در این روش، یک قانون به‌روزرسانی برخط برای مدل DMD به منظور پیش‌بینی حالات سیستم معیوب بدست آمده است تا اثر عیب در پیش‌بینی‌های کنترل‌کننده MPC در نظر گرفته شود و کنترل‌کننده بتواند این اثر را کمینه کند.

در این مقاله، در بخش ۶، مفاهیم مقدماتی مورد نیاز بیان می‌شوند. سپس در بخش ۷، مسئله کنترلی تعریف و به صورت ریاضی فرمول‌بندی می‌شود. در بخش ۸، روش کنترلی پیشنهادی ارائه می‌شود. در بخش ۹، عملکرد روش ارائه شده توسط یک مثال شبیه‌سازی شده بررسی می‌گردد. در نهایت، در بخش ۱۰، نتیجه‌گیری و پیشنهادهایی برای طرح‌های تکمیلی برای مطالعات آتی ارائه می‌گردد.

۲- مفاهیم مقدماتی

۲-۱- تجزیه به مدهای دینامیکی

روش DMD، یک روش قدرتمند مبتنی بر داده است که در آن یک مدل خطی با مرتبه پایین را از داده‌هایی با بُعد بالا استخراج کرده و دینامیک‌های پایه‌ای سیستم را شناسایی می‌کند. برای یک سیستم دینامیکی دارای ورودی کنترلی، هدف DMD بدست آوردن یک تقریب خطی از تغییرات زمانی حالت‌های سیستم به فرم زیر است:

$$x(k+1) \approx Ax(k) + Bu(k), \quad (1)$$

که در آن $x(k) \in \mathbb{R}^n$ حالت سیستم و $u(k) \in \mathbb{R}^q$ ورودی کنترلی سیستم می‌باشند. الگوریتم DMD وابسته به جمع‌آوری و دسته‌بندی سه‌تایی‌های منظمی به صورت زیر می‌باشد:

$$\{x(k), y(k), u(k)\}_{k=1}^m, \quad (2)$$

که در آن $y(k) \triangleq f(x(k))$ تعریف می‌گردد. همچنین ماتریس‌های داده X ، Y و U به صورت زیر تعریف شده‌اند:

$$\begin{aligned} X &\triangleq [x(1), x(2), \dots, x(w)], \\ Y &\triangleq [y(1), y(2), \dots, y(w)], \\ U &\triangleq [u(1), u(2), \dots, u(w)], \end{aligned} \quad (3)$$

که در آن $X, Y \in \mathbb{R}^{n \times w}$ و $U \in \mathbb{R}^{q \times w}$ هستند. حال تقریب (۱) را می‌توان برای ماتریس‌های داده به صورت زیر بازنویسی کرد:

$$Y \approx AX + BU = \begin{bmatrix} X \\ U \end{bmatrix} G = G\Omega, \quad (4)$$

همبسته خارجی^۱ به منظور شناسایی یک سیستم دینامیکی استفاده کرده است. سپس در اسن روش، اقدام به طراحی یک کنترل‌کننده پیش‌بین برای تنظیم دما در مراکز داده در حضور عیب‌های عملکرد شده است. در [۲۹] رویکردی جدید مبتنی بر مدل مخفی مارکوف و ماشین یادگیری مفرط جهت تشخیص نفوذ ارائه شده است. همچنین در [۳۰] یک رویکرد نوین برای طراحی کنترل‌کننده تحمل‌پذیر عیب عملکرد بر اساس شناسایی عیب ارائه شده است. با اینکه روش‌های مبتنی بر داده نیازی به داشتن مدل ریاضیاتی سیستم دینامیکی ندارند، ولی این نوع از روش‌ها نیز دارای محدودیت‌هایی می‌باشند. برای مثال، روش‌های ارائه شده در مراجع [۲۰-۲۳] تنها برای سیستم‌های یک ورودی-یک خروجی معتبر اند در حالی که طیف گسترده‌ای از سیستم‌ها در این دسته قرار ندارند. روش ارائه شده در مرجع [۲۲] به منظور مقابله با عیب در سیستم، نیازمند تغییر ساختار واحد کنترلی است و نمی‌تواند برای ارتقا تحمل‌پذیری سیستم‌های حلقه بسته با واحد کنترلی آن‌ها ترکیب شود. در روش ارائه شده در مرجع [۱۹]، علیرغم وجود بلوک بازتنظیمی که قابلیت ترکیب شدن با واحد کنترل سیستم را فراهم می‌کند، عملکرد مجازی ارائه شده فاقد توانایی اعمال محدودیت بر روی مقدار سیگنال ورودی است. همچنین، در مرجع [۱۸]، برای آموزش شبکه به داده‌های معیوب نیاز است که بدست آوردن چنین داده‌هایی بسیار هزینه‌بر می‌باشد. روش‌های FTC ارائه شده در مراجع [۲۷-۲۳] به یک واحد FDI و یا تخمین عیب جداگانه برای انجام الگوریتم کنترلی خود نیازمندند. همچنین در روش ارائه شده در مرجع [۲۸] نیازمند شناسایی چند مدل برای حالت‌های کاری مختلف سیستم و یک قانون جابه‌جایی بین کنترل‌کننده‌های مختلف به منظور اجرای الگوریتم FTC مبتنی بر مدل پیش‌بین می‌باشد.

اخیراً، تجزیه به مدهای دینامیکی^۲ (DMD) به عنوان یک تکنیک ریاضی قوی و نوآورانه معرفی شده است، که در زمینه تجزیه و تحلیل داده‌ها و مدل‌سازی سیستم‌ها به‌طور قابل توجهی مورد توجه قرار گرفته است. این روش ابتدا در زمینه دینامیک سیالات توسعه یافته است، اما در ادامه کاربردهای خود را در گستره‌ای وسیع از رشته‌ها از جمله مهندسی، فیزیک، زیست‌شناسی، مالی و غیره پیدا کرده است [۳۱-۳۴]. این روش امکان تجزیه و تحلیل سیستماتیک مدهای دینامیکی مجموعه‌های داده پیچیده و با ابعاد بالا را به صورت تفسیرپذیر فراهم می‌کند، که درک فرآیندهای دینامیکی را تسهیل می‌کند.

در این مقاله به ارائه روشی داده محور برای سیستم‌های دینامیکی خطی چند ورودی-چند خروجی با استفاده از روش DMD به منظور پیش‌بینی و مقابله با عیب می‌پردازیم. به طور خاص، در این مقاله، یک عملکرد مجازی بر مبنای کنترل پیش‌بین طراحی می‌شود که از یک مدل DMD که به صورت برخط به‌روز می‌شود، برای پیش‌بینی حالات سیستم معیوب بهره می‌برد. این به‌روزرسانی برخط باعث می‌شود که اثر عیب در پیش‌بینی‌های کنترل‌کننده MPC در نظر گرفته شود و کنترل‌کننده بتواند این اثر را کمینه کند. این نکته حائز اهمیت است که در هیچ یک از مراحل طراحی از مدل سیستم استفاده نشده و تنها بر اندازه‌گیری‌های انجام شده تکیه کرده‌ایم. به همین دلیل، مفهوم داده‌محور در مقابل روش‌های سنتی مدل‌سازی قرار گرفته است.

روش ارائه شده در این مقاله بر خلاف روش‌های به‌کار رفته در مراجع [۲۰-۲۲] به راحتی قابل اعمال به سیستم‌های چند ورودی-چند خروجی می‌باشد. همچنین بر خلاف روش‌های مطرح شده در مراجع [۲۳-۲۷] نیازی به واحد تشخیص و جداسازی جداگانه برای اجرای الگوریتم نداریم. همچنین بر خلاف روش ارائه شده در مرجع [۲۸] نیازی به شناسایی چند مدل مختلف و طراحی چند کنترل‌کننده نیست. به علاوه، به دلیل وجود واحد MPC در عملکرد مجازی

سیستم وارد کرده و همین روال را در هر گام زمانی تکرار می‌کند.

۳- بیان و فرمول‌بندی کردن مسئله

در این کار، هدف اصلی، طراحی یک عملگر مجازی مبتنی بر MPC است. لذا قصد داریم تا با ایجاد یک مدل پیش‌بین که بتواند اثرات عیب را در نظر بگیرد و با استفاده از آن در کنترل MPC، یک سیگنال کنترلی تولید کنیم که بتواند اثرات عیب را کاهش دهد. به عبارت دیگر، برای اینکه کنترل‌کننده MPC تحمل‌پذیر نسبت به عیب باشد، مدل پیش‌بینی بر پایه DMD معادل سیستم معیوب باید استفاده شود تا مسئله بهینه‌سازی (۱۱) حل شود. همچنین از مدل DMD در بلوک بازتنظیمی برای پنهان کردن اثر عیب از دید کنترل‌کننده نامی سیستم بهره برده شده است تا ساختار این واحد بدون تغییر باقی بماند. به این منظور، سیستم دینامیکی حلقه بسته بدون عیب به صورت

$$x(k+1) = Ax(k) + Bu_c(k), \quad (12)$$

و کنترل‌کننده نامی به صورت

$$\begin{aligned} x_c(k+1) &= f_c(x_c(k), r_c(k)), \\ u_c(k+1) &= h_c(x_c(k), r_c(k)), \end{aligned} \quad (13)$$

را در نظر بگیرید که در آن‌ها $x(k) \in \mathbb{R}^n$ بردار حالت، $u_c(k) \in \mathbb{R}^q$ بردار ورودی کنترلی، $x_c(k) \in \mathbb{R}^n$ بردار حالت کنترل‌کننده نامی و $r_c(k) \in \mathbb{R}^n$ ورودی مرجع اند. در این جا فرض بر این است که سیستم حلقه بسته تشکیل شده از سیستم (۱۲) و کنترل‌کننده (۱۳) پایدار و دارای رفتار مطلوب است. حال وقتی عیبی در عملگر رخ می‌دهد، ورودی سیستم می‌تواند به صورت زیر مدل شود [۳۵]:

$$u_f(k) = Y(k)u(k) + d_f(k), \quad (14)$$

که در آن $u_f(k)$ بردار ورودی معیوب^۴ است. همچنین پارامتر $Y(k)$ به صورت زیر تعریف شده است:

$$Y(k) \triangleq \text{diag}(v_1(k), v_2(k), \dots, v_m(k)), \quad (15)$$

که هر $v_i, i=1, \dots, m$ مقداری در بازه $0 \leq v_i \leq 1$ می‌گیرد. $v_i = 0$ ، معادل عیب کامل^۵ i -امین عملگر، $0 < v_i < 1$ به معنای نقص در عملکرد^۶ (LOE) i -امین عملگر و $v_i = 1$ به معنای صحت کامل i -امین عملگر است. عیب متغیر با زمان جمع‌شونده^۷ عملگر نیز به صورت $d_f(k)$ مدل شده است که به صورت $d_{ff}(k) \triangleq [d_{f1}(k), d_{f2}(k), \dots, d_{fm}(k)]^T$ تعریف شده است. همچنین، نشانگر عیب جمع‌شونده بر روی i -امین عملگر است. حال با استفاده از این مدل‌سازی، در صورت وقوع عیب در سیستم، مدل سیستم معیوب به صورت زیر بیان می‌گردد:

$$x_f(k+1) = Ax_f(k) + Bu_f(k), \quad (16)$$

که با جایگذاری از (۱۴) در آن داریم:

$$\begin{aligned} x_f(k+1) &= Ax_f(k) + BY(k)u(k) + Bd_f(k), \\ &= Ax_f(k) + [BY(k) \quad Bd_f(k)] \begin{bmatrix} u(k) \\ 1 \end{bmatrix}, \\ &= Ax_f(k) + B_f(k)\bar{u}(k), \end{aligned} \quad (17)$$

که در آن $x_f(k)$ بردار حالت سیستم معیوب است و ماتریس ورودی و بردار ورودی سیستم معیوب به صورت زیر تعریف شده است:

$$B_f(k) \triangleq [BY(k) \quad Bd_f(k)], \quad (18)$$

$$\bar{u}(k) \triangleq \begin{bmatrix} u^T(k) & 1 \end{bmatrix}^T. \quad (19)$$

که در آن $G \triangleq [A, B]$ و $\Omega \triangleq [X^T, U^T]^T$ است. حال اگر $\Omega = \tilde{U}_c \tilde{\Sigma} \tilde{V}_c^*$ تجزیه مقادیر تکین^۱ ماتریس Ω باشد که در آن $\tilde{V}_c^*$ نشان دهنده ترانهاده مزدوج^۲ ماتریس $\tilde{V}_c$ است، پاسخ مسئله بهینه‌سازی زیر

$$\min_G \|Y - G\Omega\|_F, \quad (5)$$

را می‌توان به صورت

$$G = Y\tilde{V}\tilde{\Sigma}^{-1}\tilde{U}^*, \quad (6)$$

بدست آورد که در آن $\|\cdot\|_F$ نرم فروبینیوس^۳ را نشان می‌دهد. با تجزیه $\tilde{U}^*$ به صورت

$$\tilde{U}^* = [\tilde{U}_1^*, \tilde{U}_2^*], \quad (7)$$

که در آن $\tilde{U}_1^* \in \mathbb{R}^{\tilde{r} \times n}$ و $\tilde{U}_2^* \in \mathbb{R}^{\tilde{r} \times q}$ می‌باشند، می‌توان ماتریس‌های A و B را از (۶) به صورت زیر استخراج کرد:

$$\begin{aligned} A &\approx Y\tilde{V}\tilde{\Sigma}^{-1}\tilde{U}_1^*, \\ B &\approx Y\tilde{V}\tilde{\Sigma}^{-1}\tilde{U}_2^*. \end{aligned} \quad (8)$$

۲-۲- کنترل مدل پیش‌بین

کنترل مدل پیش‌بین، یک روش پیشرفته برای کنترل فرآیندهای صنعتی است که در آن می‌توان قیدهای فیزیکی حاکم بر سیستم را در زمان طراحی مدل در نظر گرفت. از دیگر مزایای این روش این است که کنترل‌کننده بهینه‌ای فراهم می‌آورد که به صورت بلادرنگ و آنلاین بهینه‌سازی می‌شود و این روش کنترلی را نسبت به اثر اغتشاشات وارده بر سیستم مقاوم می‌سازد. مدل خطی گسسته زمان زیر را در نظر بگیرید:

$$x(k+1) = Ax(k) + Bu(k), \quad (9)$$

که در آن $x(k)$ و $u(k)$ به ترتیب حالت و ورودی کنترلی سیستم هستند. هدف اصلی MPC بهینه‌سازی ورودی‌های کنترل در یک افق پیش‌بینی محدود است تا یک تابع هزینه که اهداف کنترل و عملکرد سیستم را در بر می‌گیرد را کمینه کند. تابع هزینه در طول افق پیش‌بینی N به صورت زیر تعریف می‌شود:

$$J \triangleq \sum_{i=0}^{N-1} (r(i) - x(i))^T Q (r(i) - x(i)) + u(i)^T R u(i) + \Delta u(i)^T R_{\Delta} \Delta u(i), \quad (10)$$

که در آن R, Q و R_{Δ} ماتریس مثبت نیمه معین هستند و r نشان‌دهنده ورودی مرجع می‌باشد. مسئله بهینه‌سازی MPC، درواقع یافتن دنباله ورودی کنترل بهینه $\{u^*(i)\}_{i=0}^{N-1}$ است که با توجه به محدودیت‌های در نظر گرفته شده، تابع هزینه را کمینه کند. به بیان ریاضی داریم:

$$\begin{aligned} \min_{\{u^*(i)\}_{i=0}^{N-1}} J \\ \text{s.t.} \quad & \begin{cases} x(k+1) = Ax(k) + Bu(k), \\ u_{\min} < u(i) < u_{\max}, \quad \text{for } i = 0, \dots, N-1, \\ \Delta u_{\min} < \Delta u(i) < \Delta u_{\max}, \end{cases} \end{aligned} \quad (11)$$

که در آن $u_{\min/\max}$ و $\Delta u_{\min/\max}$ به ترتیب نشانگر مقدار بیشینه و کمینه ممکن برای تغییرات و مقدار اندازه ورودی کنترلی می‌باشند. در هر مرحله، کنترل‌کننده MPC مسئله بهینه‌سازی (۱۱) را حل کرده و دنباله $\{u^*(i)\}_{i=0}^{N-1}$ را بدست می‌آورد. سپس اولین المان این دنباله را به عنوان سیگنال کنترلی به

^۵ Total failure

^۶ Loss of effectiveness

^۷ Additive

^۱ Singular value decomposition

^۲ Conjugate transpose

^۳ Frobenius norm

^۴ Faulty input

۴- روش کنترلی ارائه شده

در این بخش، یک رویکرد جدید برای طراحی یک کنترل کننده تحمل پذیر عیب برای سیستم‌های دینامیکی تحت تاثیر عیب در عملکرد ارائه شده است. ایده اصلی این است که در هر مرحله زمانی یک تخمین از ماتریس ورودی سیستم معیوب انجام شود و از آن تخمین برای به دست آوردن قانون کنترل استفاده شود به صورتی که اثر عیب را خنثی کند. بنابراین، این بخش به دو زیربخش تقسیم شده است. در زیربخش اول، یک روش به روزرسانی بازگشتی برای ماتریس ورودی ارائه شده است، در حالی که در زیربخش دوم، مدل DMD به روز شده توسط قانون به روزرسانی بدست آمده در زیربخش اول، به عنوان یک مدل پیش‌بینی متغیر با زمان در کنترل کننده MPC استفاده شده است.

فرض ۱. در این مقاله فرض شده است که سیستم تحت کنترل کنترل پذیر

است و حالت‌های سیستم قابل اندازه‌گیری است.

این فرض در روش‌های مبتنی بر DMD به صورت متداول استفاده می‌شود. برای مثال می‌توان به مراجع [۳۶-۳۸] رجوع کرد. توجه شود که در صورت عدم وجود این فرض نیاز است تا کنترل پذیری از روی داده‌ها بررسی شود که این خود یک موضوع پژوهشی است و خارج از محدوده این مقاله می‌باشد.

۴-۱- تخمین بازگشتی ماتریس ورودی در یک پنجره

در این بخش یک روش به روزرسانی بازگشتی را معرفی شده است که با اعمال الگوریتم DMD به صورت بلادرنگ بر روی یک پنجره زمانی از نمونه‌های اندازه‌گیری شده، محاسبه شده است. در روش فعلی، ماتریس سیستم در تمام پنجره‌های اندازه‌گیری بدون تغییر باقی مانده، در حالی که ماتریس ورودی با در دسترس بودن داده‌های جدید به روزرسانی شده است.

با در نظر گرفتن یک پنجره شامل w اندازه‌گیری اخیر از داده‌ها به فرم (۲)، می‌توان ماتریس‌های داده زیر را تشکیل داد:

$$\begin{aligned} X_w(k) &= [x(k-w+1), x(k-w+2), \dots, x(k)], \\ Y_w(k) &= [y(k-w+1), y(k-w+2), \dots, y(k)], \\ U_w(k) &= [\bar{u}(k-w+1), \bar{u}(k-w+2), \dots, \bar{u}(k)]. \end{aligned} \quad (20)$$

فرض ۲. در این جا فرض بر این است که ماتریس $U_w(k)$ دارای مرتبه^۴ برابر با q است.

این فرض از آنجایی که معمولاً طول پنجره از ابعاد ورودی بسیار بیشتر است یا به بیان ریاضی $w \gg q$ ، در اغلب اوقات با افزایش طول پنجره قابل برقراری است و لذا شرط محدودکننده‌ای نیست.

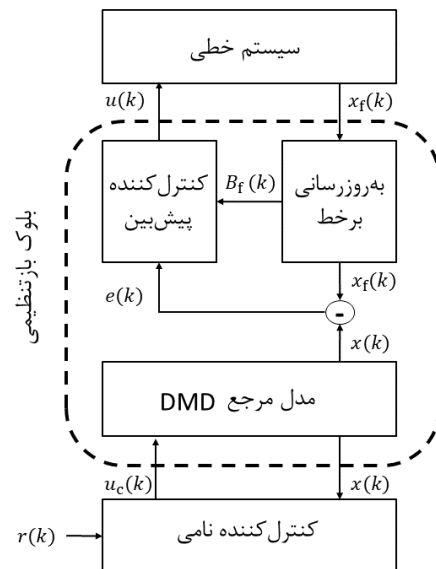
در این روش ابتدا با جمع‌آوری داده‌ها و تشکیل ماتریس‌های داده به فرم (۲۰) تخمین اولیه‌ای از مدل DMD به صورتی که پیش‌تر اشاره شد، بدست آورده شده است. حال، در هر گام زمانی، برای بدست آوردن ماتریس ورودی $B(k)$ برای مدل DMD معرفی شده در (۱)، می‌توان از قضیه زیر بهره برد.

قضیه ۱. اگر تجزیه به مقادیر تکین^۵ را برای ماتریس داده ورودی به صورت $U_w(k) = U_u \Sigma_u V_u^*$ در نظر بگیریم، ماتریس $B(k)$ را می‌توان از رابطه زیر محاسبه کرد:

$$B(k) = (Y_w(k) - AX_w(k))V_u \Sigma_u^{-1} U_u^*. \quad (21)$$

اثبات. با نوشتن مدل DMD برای ماتریس‌های داده معرفی شده در (۲۰) داریم:

بر اساس این فرمول‌بندی، می‌توان از مدل (۱۷) که یک سیستم خطی متغیر با زمان و با ورودی سالم است، به عنوان مدل پیش‌بین، برای سیستم معیوب، در واحد MPC استفاده کرد. به عبارت دیگر، اثر عیب به جای آنکه خود را در ورودی کنترلی نشان دهد، توسط ماتریس ورودی متغیر در زمان $B_f(k)$ مشخص می‌شود. به طور خاص، m ستون اول از $B_f(k)$ تغییرات ایجاد شده توسط عیب‌های LOE را نشان می‌دهند، در حالی که ستون باقی‌مانده اثر عیب‌های جمع‌شونده را تخمین می‌زند که می‌توان به آن به عنوان یک اغتشاش ورودی نگاه کرد. با این حال، به دلیل ماهیت ناشناخته عیب، ماتریس $B_f(k)$ باید در هر مرحله تخمین زده شود.



شکل ۱. ساختار کنترلی بلوکی ارائه شده

در این مقاله به طراحی یک بلوک بازتنظیم شده است که بین سیستم دینامیکی و کنترل کننده نامی سیستم، که قبلاً طراحی و به کار گرفته شده است، قرار گرفته است. شکل ۱ نحوه قرارگیری این بلوک را نشان می‌دهد. این بلوک شامل یک کنترل کننده MPC، یک بلوک به روزرسانی و یک مدل مرجع DMD است. مدل مورد استفاده در واحد MPC توسط واحد به روزرسانی همواره بر مبنای داده‌های جدید بدست آمده مرتباً به روز می‌شود. همچنین تخمینی از مسیر^۱ حالت‌های سیستم بدون عیب توسط مدل DMD به فرم (۱) تولید می‌شود. سپس کنترل کننده MPC به کمینه کردن خطای^۲ بین سیستم معیوب و مدل مرجع DMD با استفاده از تخمین ماتریس ورودی $B_f(k)$ می‌پردازد. درواقع با به کارگیری این بلوک، عیب‌های عملکردهای سیستم را از دید کنترل کننده نامی سیستم پنهان شده و عملکرد سامانه در شرایط نرمال و معیوب یکسان باقی می‌ماند.

در بخش بعد، به ارائه یک روش آنلاین مبتنی بر DMD برای به روزرسانی بازگشتی^۳ ماتریس ورودی $B_f(k)$ با استفاده از یک پنجره متحرک از اندازه‌گیری‌ها پرداخته شده است. سپس روش کنترلی مبتنی بر MPC با استفاده از مدل پیش‌بینی به روز شده را ارائه شده است که می‌تواند به طور مناسب اثر عیب‌هایی که در عملکردهای سیستم رخ داده اند را جبران کند.

^۴ Rank

^۵ Singular value decomposition

^۱ Trajectory

^۲ Error

^۳ Recursive

$$\Omega_2(k+1) = \Omega_2(k) - \Omega_2(k)U_B(k+1)Q(k+1)U_B^T(k+1)\Omega_2(k), \quad (29)$$

که در آن $Q(k+1) = (C^{-1} + U_B^T(k+1)\Omega_2(k)U_B(k+1))^{-1}$ می باشد.

اثبات. از روی تعریف می توان نوشت:

$$\begin{aligned} \Omega_2^{-1}(k+1) &= U_w(k+1)U_w^T(k+1), \\ &= \sum_{i=k-w+2}^k u(i)u^T(i), \\ &= \Omega_2^{-1}(k) - u(k-w+1)u^T(k-w+1) + u(k+1)u^T(k+1), \\ &= \Omega_2^{-1}(k) + U_B(k+1)Q(k+1)U_B^T(k+1). \end{aligned} \quad (30)$$

حال با استفاده از لم ۲ می توان رابطه بدست آمده در (۳۰) را به صورت (۲۹) نوشت و اثبات بدین ترتیب کامل شده است. □

اکنون می توان با توجه به لم های تعریف شده، به بیان قضیه زیر پرداخت.

قضیه ۲. ماتریس $B(k+1)$ را می توان به صورت بازگشتی به صورت زیر

بدست می آید:

$$B(k+1) = B(k)(I - U_B(k+1)\Phi(k+1)) + V_B(k+1)\Phi(k+1), \quad (31)$$

که در آن $\Phi(k+1) = \Theta_B(k+1)U_B^T(k+1)\Omega_2(k)$ می باشد.

اثبات. با استفاده از (۲۴) و (۲۹) می توان نوشت:

$$\begin{aligned} B(k+1) &= B(k)(I - U_B(k+1)\Phi(k+1)) + V_B(k+1)\Phi(k+1), \\ &= N(k)M(k) - N(k)M(k)U_B(k+1)\Theta_B(k+1)U_B^T(k+1)M(k) \\ &\quad + V_B(k+1)\Theta_B(k+1)U_B^T(k+1)M(k). \end{aligned} \quad (32)$$

و با تعریف $\Phi(k+1) \triangleq \Theta_B(k+1)U_B^T(k+1)\Omega_2(k)$ رابطه (۳۱) بدست می آید و اثبات کامل شده است. □

نکته ۱. از آنجایی که طول پنجره معمولاً خیلی بیشتر از ابعاد ورودی می باشد و همچنین تعداد ورودی های کنترلی کمتر و یا مساوی تعداد حالات سیستم است یا به عبارتی $w \gg q$ ، استفاده از رابطه (۳۱) به جای (۲۱) از نظر محاسباتی بهبود یافته است زیرا پیچیدگی محاسباتی معادله (۳۱) از مرتبه $q^2 + nq$ و پیچیدگی محاسباتی از مرتبه $wn^2 + 2nq$ می باشد.

نکته ۲. در قضیه ۲ نشان داده شده است که ماتریس ورودی را می توان توسط قانون بدست آمده در (۳۱) به روزرسانی کرد. اما باید توجه داشت که لازم است دو ماتریس $B(k)$ و $\Omega_2(k)$ مقداردهی اولیه شوند. برای این کار می توان از رابطه (۲۳) استفاده کرد.

۲-۴- کنترل کننده برخط پیش بین تحمل پذیر عیب

اگر مدل DMD بدست آمده در (۱۷) را توسط قانون به روزرسانی (۳۱) در هر گام به روز کنیم، می توانیم در هر لحظه به مدلی داده محور از دینامیک سیستم معیوب دسترسی داشته باشیم. با تجزیه ماتریس ورودی در (۱۷) داریم:

$$x_F(k+1) = Ax_F(k) + B_1(k)u(k) + B_2(k), \quad (33)$$

که در آن $B_1(k)$ مدل کننده عیب های LOE و $B_2(k)$ مدل کننده عیب های جمع شونده است. حال با استفاده از این مدل به عنوان مدل پیش بین در کنترل کننده MPC، می توان اثر عیب های سیستم را در مدل پیش بین مورد استفاده در کنترل کننده وارد کرد. از این رو، قانون کنترلی در هر گام می تواند از حل مسئله بهینه سازی زیر بدست آید:

$$Y_w(k) - AX_w(k) = B(k)U_w,$$

$$Y_w(k) - AX_w(k) = B(k)U_u \Sigma_u V_u^*, \quad (22)$$

$$(Y_w(k) - AX_w(k))V_u \Sigma_u^{-1}U_u^* = B(k).$$

و این اثبات کامل شده است. □

نکته قابل توجه در این قضیه این است که ماتریس سیستم A همواره ثابت و برابر ماتریس اولیه سیستم باقی مانده است و تنها ماتریس ورودی در هر گام تخمین زده شده است. این کار به این خاطر است که در فرمول بندی ما، اثر عیب تنها در ماتریس ورودی ظاهر شده است. محاسبه ماتریس ورودی با استفاده از (۲۱) بار محاسباتی زیادی دارد زیرا هر بار باید تجزیه به مقادیر تکین بر روی کل داده های پنجره صورت گیرد. لذا قصد داریم تا روشی بازگشتی برای محاسبه $B(k)$ ارائه کنیم تا تنها با استفاده از داده های اندازه گیری شده جدید، ماتریس ورودی را به روزرسانی کنیم. اگر $U_w^T(k)$ ماتریس شبه وارون^۱ $U_w(k)$ باشد، آنگاه (۲۱) را می توان به صورت زیر بازنویسی کرد:

$$\begin{aligned} B(k) &= (Y_w(k) - AX_w(k))U_w^T(k), \\ &= (Y_w(k) - AX_w(k))U_w^T(k)(U_w(k)U_w^T(k))^{-1}, \\ &= H_w(k)U_w^T(k)(U_w(k)U_w^T(k))^{-1}, \end{aligned} \quad (23)$$

که در آن

$$H_w(k) \triangleq [h(k-w+1), \dots, h(k)], \quad (24)$$

$$h(i) \triangleq y(i) - Ax(i), \quad i = k-w+1, \dots, k,$$

می باشد. حال با تعریف ماتریس های داده $U_B(k+1) \triangleq [u(k-w+1), u(k+1)]$ و $V_B(k+1) \triangleq [h(k-w+1), h(k+1)]$ می توان به تعریف لم های زیر را پرداخت.

لم ۱. اگر $\Omega_1(k) \triangleq H_w(k)U_w^T(k)$ باشد، آنگاه $\Omega_1(k+1)$ به صورت بازگشتی به شکل زیر محاسبه می گردد:

$$\Omega_1(k+1) = \Omega_1(k) + V_B(k+1)CU_B^T(k+1), \quad (25)$$

که در آن

$$C = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (26)$$

می باشد.

اثبات. از روی تعریف می توان نوشت:

$$\begin{aligned} \Omega_1(k+1) &= H_w(k+1)U_w^T(k+1), \\ &= \sum_{i=k-w+2}^{k+1} h(i)u^T(i), \\ &= \Omega_1(k) - h(k-w+1)u^T(k-w+1) + h(k+1)u^T(k+1), \\ &= \Omega_1(k) + V_B(k+1)CU_B^T(k+1). \quad \square \end{aligned} \quad (27)$$

لم ۲. [۳۹] اگر دو ماتریس A و $I - VA^{-1}U$ معکوس پذیر باشند، آنگاه $A - UV$ نیز معکوس پذیر است و رابطه زیر برقرار می باشد:

$$(A - UV)^{-1} = A^{-1} + A^{-1}U\Theta^{-1}V A^{-1}, \quad (28)$$

که در آن $\Theta \triangleq I - VA^{-1}U$ می باشد.

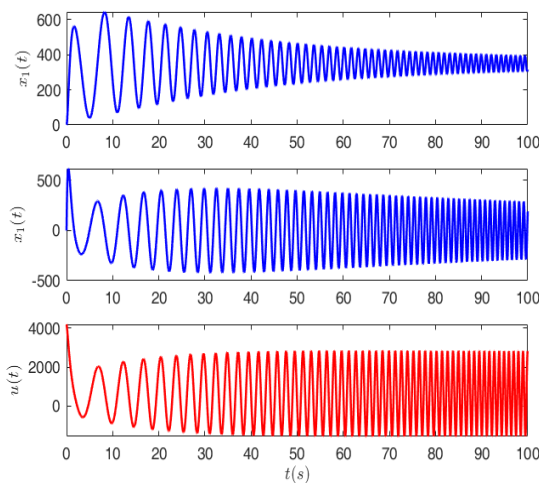
لم ۳. اگر $\Omega_2(k) \triangleq (U_w(k)U_w^T(k))^{-1}$ باشد، آنگاه $\Omega_2(k+1)$ به صورت بازگشتی به شکل زیر محاسبه می شود:

۵- شبیه سازی و نتایج

در این بخش عملکرد عملگر مجازی مبتنی بر کنترل پیش‌بین تحمل پذیر عیب در قالب یک مثال شبیه‌سازی مورد بررسی قرار می‌گیرد. در این قسمت دینامیک مورد استفاده در [۱۴] که توصیف کننده کانال سرعت یک زیر دریایی است، مورد استفاده قرار گرفته است. معادله حالت این سیستم به صورت زیر می‌باشد:

$$\begin{aligned} x(k+1) &= \begin{bmatrix} 0 & 1 \\ -1.841 & -4.634 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) + d(k), \\ y(k) &= [0.002864 \quad 0] x(k). \end{aligned} \quad (35)$$

این نکته حائز اهمیت است که دینامیک مورد نظر تنها برای تولید داده مورد استفاده قرار گرفته است و اطلاعاتی از معادلات حاکم بر این سیستم در اختیار ما قرار ندارد. همچنین فرض بر آن است که سیستم حلقه بسته تحت کنترل کننده نامی خود که قبلاً طراحی شده، پایدار است و دارای رفتار مطلوب می‌باشد. در وهله اول به شناسایی دینامیک سیستم توسط روش DMD و تشکیل مدل اولیه‌ای از دینامیک سیستم پرداخته شده است. بدین منظور با جمع‌آوری داده‌هایی از سیستم به تشکیل ماتریس‌های داده (۳) پرداخته شده است. سپس از روی این ماتریس‌ها و با استفاده از معادلات بیان شده در (۸) مدل DMD مورد استفاده در بلوک بازتنظیمی را تشکیل داده تا عیب‌ها را از دید کنترل کننده نامی پنهان شوند. داده‌های مورد استفاده به منظور تشکیل مدل DMD در شکل ۲ نشان داده شده است.



شکل ۲. داده‌های جمع‌آوری شده برای بدست آوردن مدل DMD سیستم دینامیکی در نظر گرفته شده

برای بررسی عملکرد روش ارائه شده در مواجهه با اغتشاش، سناریویی در نظر گرفته شده است که در آن اغتشاشی به اندازه $d(k) = [0 \quad 20]^T$ واحد مطابق شکل ۳ در $t_d = 50s$ به آن وارد شده است. همانطور که در تصویر نیز مشخص است، کنترل کننده ارائه شده به خوبی اغتشاش وارده را دفع کرده و اثر آن را از بین برده است.

در مرحله بعد به بررسی عملکرد این روش در مقابله با رخ‌دادن عیب در سیستم پرداخته شده است. برای این کار فرض شده که یک عیب از نوع نقص عملکرد و جمع‌شونده در عملگر سیستم در زمان $t_f = 20s$ رخ داده است به نحوی که اندازه عیب LOE کارایی عملگر را به 80% مقدار نامی آن کاهش

$$\begin{aligned} \min_{\{u^*(i)\}_{i=0}^{N-1}} & \sum_{i=0}^{N-1} (x_f(i) - x(i))^T Q (x_f(i) - x(i)) + u(i)^T R u(i) + \Delta u(i)^T R_\Delta \Delta u(i), \\ \text{s.t.} & \begin{cases} x_f(k+1) = A x_f(k) + B_1(k) u(k) + B_2(k), \\ u_{\min} < u(i) < u_{\max}, \\ \Delta u_{\min} < \Delta u(i) < \Delta u_{\max}, \end{cases} \quad \text{for } i = 0, \dots, N-1. \end{aligned} \quad (36)$$

روش ارائه شده در این مقاله شامل دو مرحله می‌باشد. در مرحله نخست، ماتریس ورودی سیستم معیوب توسط روش بازگشتی ارائه شده تخمین زده شده و مدل پیش‌بین را به‌روزرسانی شده است. در مرحله بعد، این مدل پیش‌بین در کنترل کننده MPC به کار گرفته شده است و سیگنال کنترلی با در نظر گرفتن اثر عیب‌ها تولید شده، تا خطای بین عملکرد سیستم معیوب و مدل DMD معادل سیستم سالم، کمینه شود و یا به عبارت دیگر اثر عیب توسط کنترل کننده جبران شود. خلاصه روند ساختار کنترلی در الگوریتم زیر بیان شده است.

الگوریتم ۱: الگوریتم کنترل حلقه بسته با استفاده از روش ارائه شده

ورودی: تخمین ماتریس‌های مدل DMD بیان شده در $(A), R, R_\Delta, Q,$

$u_{\min/\max}, \Delta u_{\min/\max}$

خروجی: سیگنال کنترلی $u(k)$.

۱: مقداردهی اولیه $B(0)$ و $\Omega_2(0)$ از رابطه (۲۳).

برای $k = 1, \dots$ داریم:

۲: مقدارهای جدید $x(k)$ اندازه‌گیری شود.

۳: ماتریس ورودی $B(k)$ توسط رابطه (۳۱) محاسبه شود.

۴: مدل مورد استفاده در واحد MPC توسط ماتریس ورودی جدید به‌روزرسانی شود.

۵: حل مسئله بهینه‌سازی (۳۴) و بدست آوردن سری ورودی‌های بهینه $\{u^*(i)\}_{i=0}^{N-1}$.

۶: اعمال $u(k) = u^*(0)$ به سیستم تحت کنترل.

پایان

نکته ۲. می‌توان از مدل DMD که بر مبنای داده‌های سالم سیستم بدست

آمده است به عنوان یک تشخیص دهنده خطا بهره برد به صورتی که خروجی این مدل را می‌توان با خروجی سیستم اصلی مقایسه کرد و زمانی که اختلاف این دو سیگنال از مقدار مشخصی بیشتر شد، می‌توان نتیجه گرفت که عیبی در سیستم رخ داده است و سپس اقدام به بکارگیری روش ارائه شده نمود. این کار باعث می‌شود که از بار محاسباتی استفاده از روش مورد نظر کاسته شود و در زمانی که محدودیت پردازش محاسباتی وجود دارد این امر مفید واقع شود.

نکته ۳. این نکته دارای اهمیت است که ما در هیچ یک از مراحل طراحی، از معادلات نیوتنی و یا دینامیک کلاسیک بهره نبرده و تنها از اندازه‌گیری‌های انجام شده استفاده کرده‌ایم. لذا در اینجا مفهوم داده محور در مقابل استفاده از روش‌های مرسوم مدل‌سازی عنوان گردیده است. باید توجه شود که به دلیل استفاده از مدل DMD در قلب این روش، می‌توان روش ارائه شده را به عنوان یکی از روش‌های داده‌محور خاکستری قلمداد کرد.

همچنین، جهت ارائه تحلیلی دقیق تر از عملکرد سیستم در مواجهه با عیب، از معیار جذر میانگین مربعات خطا (RMSE) کمک گرفته ایم. این معیار به صورت زیر تعریف شده است.

$$RMSE = 100 \times \frac{\sqrt{\sum_k \|x_c(kT_s) - x_h(kT_s)\|_2^2}}{\sqrt{\sum_k \|x_h(kT_s)\|_2^2}}, \quad (36)$$

که در آن x_c خروجی سیستم تحت کنترل، x_h خروجی سیستم در حالت بدون عیب و T_s زمان نمونه برداری است. همانطور که از جدول ۲ نیز می توان مشاهده کرد که عملکرد روش ارائه شده زمانی که از ساختار کنترلی پیشنهادی استفاده شده است، نسبت به زمانی که تنها از کنترل کننده نامی سیستم استفاده شده است، دارای مقدار خطا کمتری بوده است.

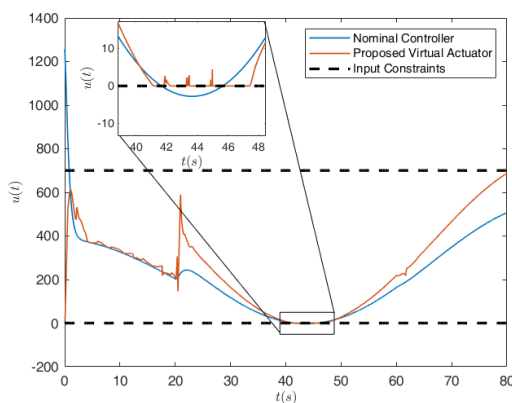
جدول ۲. مقایسه عملکرد سیستم در مواجهه با عیب بر مبنای معیار

جذر میانگین مربعات خطا

کنترل کننده	کنترل کننده نامی	کنترل کننده نامی سیستم همراه با بلوک باز تنظیمی
RMSE	18.1%	2.9%

این موضوع خود مؤید این است که اثر عیب بر روی سیستم توسط روش ارائه شده کمینه شده است و این روش منجر به بهبود کارایی سیستم در مواجهه با عیب شده است.

از دیگر مزیت هایی که استفاده از کنترل کننده MPC در بلوک بازپیکره بندی روش ارائه شده برای ما فراهم کرده است، می توان به توانایی اعمال محدودیت بر روی ورودی اشاره کرد.



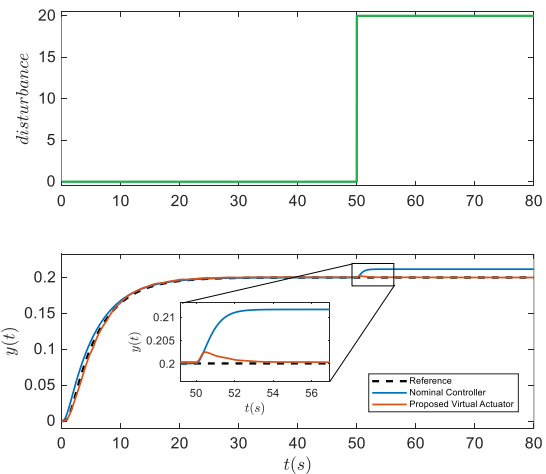
شکل ۵. مقایسه ورودی سیستم تحت کنترل کنترل کننده نامی و سیستم بازپیکره بندی شده با استفاده از عملگر مجازی ارائه شده مبتنی بر کنترل پیش بین

همانطور که در شکل ۵ نیز مشخص است، ورودی سیستم بازپیکره بندی شده با استفاده از عملگر مجازی ارائه شده مبتنی بر کنترل پیش بین، در بازه مشخص شده مانده است در حالی که هیچ محدودیتی بر روی ورودی تولید شده توسط کنترل کننده نامی وجود ندارد و می تواند از بازه مجاز عبور کند.

۴- نتیجه گیری و کارهای آینده

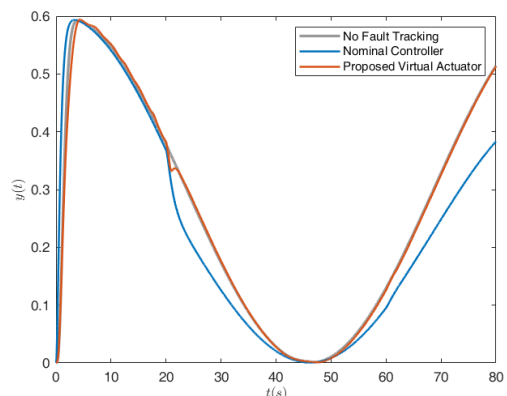
در این مقاله به ارائه یک روش کنترلی تحمل پذیر عیب مبتنی بر داده برای سیستم های خطی نامتغیر با زمان پرداخته شد. به طور دقیق تر در این کار با استخراج داده های بدست آمده از دینامیک سیستم و تجزیه آن ها به مدهای دینامیکی به یک پیش بینی کننده داده محور دست یافتیم که می توان از آن به

داده است و اندازه عیب جمع شونده +10 واحد می باشد. خروجی سیستم



شکل ۳. مقایسه عملکرد کنترل کننده نامی و سیستم بازپیکره بندی شده با استفاده از عملگر مجازی ارائه شده در مواجهه با اغتشاش

معیوب به ازای این سناریو با استفاده از کنترل کننده نامی و سیستم بازپیکره بندی شده با استفاده از عملگر مجازی ارائه شده مبتنی بر کنترل پیش بین در شکل ۴ نشان داده شده است. همانطور که در این شکل مشخص است، سیستم بازپیکره بندی شده در مواجهه با عیب های نقص عملکرد و جمع شونده عملکرد مطلوب تری نسبت به سیستم حلقه بسته تحت کنترل کنترل کننده نامی به نمایش می گذارد.



شکل ۴. مقایسه خروجی سیستم تحت کنترل کنترل کننده نامی و سیستم بازپیکره بندی شده با استفاده از عملگر مجازی ارائه شده مبتنی بر کنترل پیش بین

این اثر به دلیل این است که ماتریس ورودی به نحوی تطبیق یافته است که اثر عیب را در پیش بینی کنترل کننده MPC دخیل می کند. لذا این کنترل کننده اثرات عیب را در خروجی جبران کرده است. مشخصات کنترل کننده MPC به کار رفته در بلوک پیکره بندی مورد استفاده در این مثال، در جدول ۱ ارائه شده است.

جدول ۱. پارامترهای کنترل کننده پیش بین بکار رفته در بلوک

بازپیکره بندی

پارامتر	گام گسسته سازی	افق پیش بینی	افق کنترل	$u_{\min}$	$u_{\max}$	$\Delta u_{\min}$	$\Delta u_{\max}$
مقدار	0.01	100	3	0	700	-10	+10

- [13] M. Yadegar and N. Meskin, "Fault-Tolerant Control of One-Sided Lipschitz Nonlinear Systems," *IEEE Control Systems Letters*, vol. 6, pp. 1460--1465, 2021.
- [14] M. Yadegar, N. Meskin, and Ahmad Afshar, "Fault-tolerant control of linear systems using adaptive virtual actuator," *International Journal of Control*, vol. 92, no. 8, pp. 1729--1741, 2019.
- [15] M. Yadegar, A. Afshar, and N. Meskin, "Fault-tolerant control of non-linear systems based on adaptive virtual actuator," *IET Control Theory & Applications*, vol. 11, no. 9, pp. 1371--1379, 2017.
- [16] I. Bessa, V. Puig, and R. Palhares, "Reconfiguration block and fault hiding: Design, applications, and challenges," *Annual Reviews in Control*, p. 100896, 2023.
- [17] J. Richter, *Reconfigurable control of nonlinear dynamical systems: a fault-hiding approach*.: Springer, 2011.
- [18] H. Dai, P. Chen, and H. Yang, "Metalearning-Based Fault-Tolerant Control for Skid Steering Vehicles under Actuator Fault Conditions," *Sensors*, vol. 22, no. 3, pp. 1424-8220, 2022.
- [19] M. Yadegar, M. Bakhtiaridoust, and N. Meskin, "Adaptive data-driven fault-tolerant control for nonlinear systems: Koopman-based virtual actuator approach," *Journal of the Franklin Institute*, vol. 360, no. 11, pp. 7128-7147, 2023.
- [20] Z. Karimi, Y. Batmani, M. Khosrowjerdi, and C. Konstantinou, "Data-Driven Fault-Tolerant Tracking Control for Linear Parameter-Varying Systems," *IEEE Access*, vol. 10, pp. 66734-66742, 2022.
- [21] X. Li, Q. Luo, L. Wang, R. Zhang, and F. Gao, "Off-policy reinforcement learning-based novel model-free minmax fault-tolerant tracking control for industrial processes," *Journal of Process Control*, vol. 115, pp. 145-156, 2022.
- [22] M. Bakhtiaridoust, M. Yadegar, and F. Jahangiri, "Koopman fault-tolerant model predictive control," *IET Control Theory & Applications*, pp. 1-12, 2024.
- [23] H. Fan, J. Han, and B. Wang, "Data-driven fault-tolerant control for SISO nonlinear system with unknown sensor fault," *International Journal of Robust and Nonlinear Control*, vol. 33, no. 6, pp. 3677--3698, 2023.
- [24] B. Rouabah, H. Toubakh, M. Kafi, M. Sayed-Mouchaweh, "Adaptive data-driven fault-tolerant control strategy for optimal power extraction in presence of broken rotor bars in wind turbine," *ISA transactions*, vol. 130, pp. 92-103, 2022.
- [25] D. Zheng, X. Xu, and D. Lin, "Data-driven fault-tolerant control for unmanned aerial vehicles without using identification model," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 236, pp. 1-10, 2022.
- [26] Y. Wang, Z. Wang, "IEEE Transactions on Circuits and Systems II: Express Briefs," *Data-Driven Model-Free Adaptive Fault-Tolerant Control for a Class of Discrete-Time Systems*, vol. 69, no. 1, pp. 154-158, 2022.
- [27] J. Zheng and Z. Hou, "Model Free Adaptive Iterative Learning Control Based Fault-Tolerant Control for Subway Train With Speed Sensor Fault and Over-Speed Protection," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 1, pp. 168-180, 2024.
- [28] K. Jiang, M. Kheradmandi, C. Hu, S. Pal, and Y. Fengjun, "Data-driven fault-tolerant predictive control for temperature regulation in data center with rack-based cooling architecture," *Mechatronics*, vol. 79, no. 0957-4158, p. 102633, 2021.
- [29] M. Najjar, M. H. Moattar, *Network Intrusion Detection using a Hybrid of Hidden Markov Model and Extreme*

عنوان مدل پیش‌بین برای بهینه‌سازی تابع هزینه کنترل‌کننده مدل پیش‌بین بهره برد. در ادامه با بدست آوردن قانون تطبیق این مدل برای سیستم معیوب، توانستیم تا رفتار این سیستم را تخمین زده و اثر عیب‌های اعمالی اعم از جمع‌شونده و نقص عملکرد را توسط بلوک بازپیکره‌بندی جبران کنیم. در نهایت عملکرد عملگر مجازی ارائه شده با استفاده از شبیه‌سازی بررسی شده است. از محدودیت‌های این روش می‌توان به حساسیت آن نسبت به نویز اشاره کرد. همچنین، باید بازه زمانی پنجره داده با دقت انتخاب شود تا سرعت تخمین و تصحیح خطا توسط این روش مطلوب باشد. از طرح‌های تکمیلی برای ادامه کار در این زمینه می‌توان به توسعه روش پیشنهادی برای سیستم‌های خطی متغیر با زمان و سیستم‌های غیرخطی اشاره کرد. همچنین می‌توان به بررسی پایداری سیستم حلقه بسته و یا توسعه این روش برای طراحی عملگر مجازی تطبیقی با استفاده از فیدبک خروجی به عنوان یکی از مسیرهای تحقیقاتی آینده پرداخت.

مراجع

- [1] S. Ding, "Advanced methods for fault diagnosis and fault-tolerant control." Springer Berlin Heidelberg, 2021.
- [2] H. Alwi, C. Edwards, C.P. Tan. *Fault detection and fault-tolerant control using sliding modes*. London: Springer, 2011.
- [3] P. Nowak, A. Milecki, "Review of fault-tolerant control systems used in robotic manipulators," *Applied Sciences*, vol. 13, no. 4, p.2675, 2023.
- [4] V. Mien and D. Ceglarek, "Robust fault-tolerant control of robot manipulators with global fixed-time convergence," *Journal of the Franklin Institute*, vol. 358, pp. 699--722, 2021.
- [5] X. Wang, Z. Fei, Z. Wang, and X. Liu, "Event-triggered fault estimation and fault-tolerant control for networked control systems," *Journal of the Franklin Institute*, vol. 356, no. 8, pp. 4420--4441, 2019.
- [6] F. Albalawi, H. Durand, and PD. Christofides, "Process operational safety using model predictive control based on a process Safeness Index," *Computers & Chemical Engineering*, vol. 104, pp. 76--88, 2017.
- [7] D. Xue, NH. El-Farra, "Forecast-triggered model predictive control of constrained nonlinear processes with control actuator faults," *Mathematics*, vol. 6, no. 6, p. 104, 2018.
- [8] R. Wang, X. Li, Y. Liu, W. Fu, S. Liu, X. Ma, "Multiple model predictive functional control for marine diesel engine," *Mathematical Problems in Engineering*, pp. 1--20, 2018.
- [9] L. Liu, K. Shi, X. Yuan, Q. Li, "Multiple model-based fault-tolerant control system for distributed drive electric vehicle," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 41, pp. 1--15, 2019.
- [10] M. Naderi, TA. Johansen, AK. Sedigh, "A fault-tolerant control scheme using the feasible constrained control allocation strategy," *International Journal of Automation and Computing*, vol. 16, pp. 628--643, 2019.
- [11] R. Bavili, A. Mohammadzadeh, J. Tavoosi, S. Mobayen, W. Assawinchaichote, JH. Asad, AH. Mosavi, "A New Active Fault Tolerant Control System: Predictive Online Fault Estimation," *IEEE Access*, vol. 9, pp. 118461--118471, 2021.
- [12] K. Ghanbarpour, F. Bayat, and A. Jalilvand, "Dependable power extraction in wind turbines using model predictive fault-tolerant control," *International Journal of Electrical Power & Energy Systems*, vol. 118, p. 105802, 2020.

- isolation for nonlinear systems using koopman operator," *IEEE Access*, vol. 10, pp. 14835-14845, Jan. 2022.
- [35] M. Yadegar and N. Meskin, "Fault-tolerant control of nonlinear heterogeneous multi-agent systems," *Automatica*, vol. 127, p. 109514, 2021.
- [36] Q. Lu, and M. Zavala, "Image-based model predictive control via dynamic mode decomposition," *Journal of Process Control*, vol. 104, pp. 146-157, 2021.
- [37] A. Narasingam, and J. Sang, "Development of local dynamic mode decomposition with control: Application to model predictive control of hydraulic fracturing," *Computers & Chemical Engineering*, vol. 106, pp. 501-511, 2017.
- [38] X. Qian, Q. Dang, S. Jia, Y. Yuan, K. Huang, H. Chen and L. Zhang, "Operation of Distillation Columns Using Model Predictive Control Based on Dynamic Mode Decomposition Method," *Industrial & Engineering Chemistry Research*, vol. 62, pp. 21721-21739, 2023.
- [39] W. Hager, "Updating the inverse of a matrix," *SIAM review*, vol. 31, no. 2, pp. 221--239, 1989.
- Learning Machine. *TABRIZ JOURNAL OF ELECTRICAL ENGINEERING*, 2019; 48(4): 1807-1817.
- [30] A. Khodadadi, M. Shahriari-kahkeshi, A. Chatraei, A Novel Scheme for Actuator Fault Tolerant Controller Design based on the Fault Identification. *TABRIZ JOURNAL OF ELECTRICAL ENGINEERING*, 2018; 48(2): 595-608.
- [31] M. bakhtiaridoust, M. Yadegar, N. Meskin, "Data-driven fault detection and isolation of nonlinear systems using deep learning for Koopman operator," *ISA transactions*, vol. 134, pp. 200-211, Mar. 2023.
- [32] P.J. Schmid, "Dynamic mode decomposition and its variants," *Annual Review of Fluid Mechanics*, vol. 54, pp. 225-254, 2022.
- [33] J.L. Proctor and, S. Brunton, J.N. Kutz, "Dynamic mode decomposition with control," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 1, pp. 142-161, 2016.
- [34] M. Bakhtiaridoust, M. Yadegar, N. Meskin, and M. Noorizadeh, "Model-free geometric fault detection and