

## روش جدید حل مسئله سنتز $\mu$ با استفاده از الگوریتم بهینه سازی اجتماع پرندگان

علی لاری<sup>۱</sup>، علیرضا خسروی<sup>۲</sup>

<sup>۱</sup> دانشجوی کارشناسی ارشد مهندسی برق، گروه کنترل، دانشگاه صنعتی نوشیروانی بابل، a.lari@stu.nit.ac.ir

<sup>۲</sup> استادیار، دانشکده مهندسی برق، گروه کنترل، دانشگاه صنعتی نوشیروانی بابل، akhosravi@nit.ac.ir

(تاریخ دریافت مقاله ۱۳۹۰/۳/۱۰، تاریخ پذیرش مقاله ۱۳۹۰/۶/۲۶)

**چکیده:** طراحی کنترل کننده به روش سنتز  $\mu$ ، مسئله ای است که بدلیل مشکلات موجود در محاسبه مقادیر ویژه ساختار یافته هنوز به طور کامل حل نشده است. رایج ترین روش حل مسئله سنتز  $\mu$  تکرار D-K نام دارد. با وجود آنکه این روش، حل کامل مسئله سنتز  $\mu$  نیست ولی کنترل کننده های بدست آمده از آن به لحاظ معیارهای پایداری و عملکرد از قویترین روشهای کنترل مقاوم هستند. از سوی دیگر بالا بودن درجه کنترل کننده، مهمترین اشکال روش تکرار D-K است. در این مقاله از یک الگوریتم بهینه سازی سیر تکاملی به نام الگوریتم بهینه سازی اجتماع پرندگان، به منظور طراحی یک کنترل کننده مقاوم استفاده شده است. هدف آنستکه حلی برای مسئله سنتز  $\mu$  پیدا شود که در میزان پایداری و عملکرد مقاوم، نسبت به کنترل کننده هم درجه (درجه کاهش یافته) بدست آمده از روش تکرار D-K، وضعیت بهتری داشته باشد. به منظور ارزیابی الگوریتم پیشنهادی، از آن برای طراحی کنترل کننده یک سیستم محکم جرم-فنر-میراگر استفاده می شود. نتایج شبیه سازی موید آنستکه کنترل کننده های بدست آمده با این روش از نظر پایداری و عملکرد مقاوم سیستم حلقه بسته بسیار کاراتر از کنترل کننده های تکرار D-K هم درجه هستند.

**کلمات کلیدی:** کنترل مقاوم، سنتز  $\mu$ ، مقادیر ویژه ساختاریافته، الگوریتم اجتماع پرندگان

### A New Solution for $\mu$ Synthesis Problem Using Particle Swarm Optimization Algorithm

Ali Lari, Alireza Khosravi

**Abstract:** The  $\mu$  synthesis problem has not been completely solved, and this is attributed to existing challenges and issues in calculation of the structure singular value. The most common solution for  $\mu$  Synthesis problem is called D-K Iteration. Even though this specific method is not the complete solution for the  $\mu$  Synthesis, but the controllers obtained through this method have proven to be one of the most complete forms of robust control technique, based on robust stability and performance. One of the major disadvantages with the D-K Iteration is a high order controller. In this paper an evolutionary algorithm called PSO has been used to design a robust controller. The main objective is to find a solution for  $\mu$  Synthesis that can better improve the robust stability and performance compared to same order (reduced order) controller obtained through D-K Iteration. To evaluate the proposed algorithm, it has been used on the mass-spring-damper benchmark system. The simulation results from the proposed algorithm show that this method has a more robust stability and performance for closed loop systems than the same order controllers obtained through D-K Iteration.

**Keywords:** robust control,  $\mu$  synthesis, structured singular value, particle swarm optimization algorithm.

## ۱- مقدمه

گیرند. به دلایل ذکر شده، برای کنترل کننده های مرتبه بالا ناگزیر به استفاده از روشهای کاهش مرتبه می باشیم تا کنترل کننده هایی با خواص مشابه اما با درجه پایین تر بدست آوریم [۲]. این عمل سبب می شود تا بعضی از اهداف طراحی تضعیف شود و یا حتی از دست برود. برای حل این مشکل، این مقاله روشی بر اساس الگوریتم های سیر تکاملی ارائه نموده است.

بطور کلی کاربرد الگوریتم های سیر تکاملی در دهه اخیر در زمینه کنترل بسیار افزایش یافته است. تنوع این کاربردها را می توان در طراحی یک کنترل کننده ساده مانند PID [۳] تا کنترل کننده های با ساختار پیچیده تر [۴] مشاهده کرد. این الگوریتم ها در روش های کنترل مقاوم نیز به کار گرفته شده اند و بعضاً نتایج بسیار مطلوبی حاصل شده است [۵،۶،۷،۸،۹]. بدلیل آنکه در این رویکرد، مرتبه کنترل کننده توسط طراح مشخص می شود به کاهش مرتبه کنترل کننده نیازی نمی باشد.

الگوریتم اجتماع پرندگان یکی از الگوریتم های سیر تکاملی است که در ابتدا توسط کندی و ابرهارت ارائه شد [۱۰]. این الگوریتم یک الگوریتم بهینه سازی جمعیت محور است که از رفتار اجتماعی و حرکت دسته جمعی پرندگان الهام گرفته شده است. الگوریتم اجتماع پرندگان به لحاظ همگرایی یک الگوریتم سریع است و پیاده سازی عملی آن به دلیل جستجوی جهت دار و محاسبات ساده، آسان می باشد [۱۱،۱۲]. الگوریتم های بهینه سازی سیر تکاملی نسبت به انواع الگوریتم های بهینه سازی سنتی سه ویژگی مهم دارند. اول اینکه آنها به حدس اولیه وابسته نیستند. دوم آنکه به علت آنکه ماهیت تصادفی دارند در تله های محلی گیر نمی کنند و سوم آنکه در آنها نیازی به محاسبه گرادیان تابع هزینه نداریم [۱۳]. الگوریتم سیر تکاملی که در این مقاله مورد استفاده قرار گرفته است الگوریتم اجتماع پرندگان است.

در این مقاله روشی بر اساس الگوریتم های سیر تکاملی ارائه شده است تا در حل مسئله سنتز  $\mu$  مورد استفاده قرار گیرد. در این روش مسئله سنتز  $\mu$  به یک مسئله مینیمم سازی مقید تبدیل می شود که قید آن شرط پایداری است و عملکرد مقاوم سیستم حلقه بسته، تابع معیاری است که بدنبال مینیمم سازی آن هستیم. در این مسئله بهینه سازی، میزان پایداری و عملکرد مقاوم با معیار  $\mu$  سنجیده می شود. روش ارائه شده به لحاظ نظری بسیار ساده تر از روش تکرار  $D-K$  است. این روش، کوچکترین کنترل کننده به لحاظ مرتبه را بدست می آورد که در شرایط پایداری و عملکرد مقاوم صدق کند. بنابراین نیازی به کاهش مرتبه کنترل کننده ندارد.

به منظور ارزیابی الگوریتم پیشنهادی، از آن برای طراحی کنترل کننده یک سیستم محک جرم-فنر-میراگر استفاده می شود. علت انتخاب این سیستم این است که بسیاری از سیستمهای دنیای واقعی را می توان با این سیستم مدل کرد. نتایج شبیه سازی مویده آنتسکه الگوریتم پیشنهادی به لحاظ معیارهای پایداری و عملکرد مقاوم از کنترل کننده هم درجه (کاهش مرتبه یافته) حاصله از روش تکرار  $D-K$  شرایط به مراتب

برای بدست آوردن مقادیر ویژه ساختار یافته یا  $\mu$  یک ماتریس، لازم است تا ساختار ماتریس عدم قطعیت مرتبط با آن را بدانیم.  $\mu$  یک ماتریس در واقع کوچکترین (بر حسب بزرگترین مقدار منفرد) عدم قطعیتی را محاسبه می کند که سبب ناپایداری آن می شود. اگر ساختار ماتریس عدم قطعیت را قطری بلوکی در نظر بگیریم، می توان بلوک های عدم قطعیت را به دو دسته بلوک های کامل و بلوک های اسکالر تقسیم کرد. بدست آوردن مقادیر ویژه ساختار یافته یک ماتریس با توجه به نحوه مدلسازی عدم قطعیت، مسئله ای است که هنوز به طور کامل حل نشده است. البته قضایایی وجود دارند که می توان از آنها برای محاسبه حدود بالا و پایین  $\mu$  استفاده کرد. بر طبق این قضایا، حد بالای  $\mu$  تنها برای ساختارهای عدم قطعیتی که جمع وزندار بلوک های اسکالر و کامل آنها از حد معینی بالاتر نرود، با مقدار  $\mu$  ماتریس برابر می شود. در صورتیکه حد پایین برای همه ساختارهای عدم قطعیت، با مقدار  $\mu$  ماتریس برابر می شود. لازم بذکر است مسئله محاسبه حد پایین  $\mu$ ، بدلیل داشتن ماکزیمم های محلی زیاد به سختی قابل حل است.

در کنترل مقاوم از مقادیر ویژه ساختار یافته به عنوان معیاری برای بررسی پایداری و عملکرد مقاوم یک سیستم حلقه بسته استفاده می شود. طراحی کنترل کننده با استفاده از این معیار مسئله سنتز  $\mu$  نامیده می شود. در حقیقت مسئله سنتز  $\mu$  هنوز به طور کامل حل نشده است [۱]. البته روشی به نام تکرار  $D-K$  وجود دارد که از آن برای حل مسئله سنتز  $\mu$  استفاده می شود. کنترل کننده طراحی شده با این روش، پایداری و عملکرد سیستم حلقه بسته را بر اساس معیار  $\mu$  تضمین می کند. روش تکرار  $D-K$  بهترین عملکرد مقاوم سیستم حلقه بسته را که متناظر با مینیمم سازی معیار عملکرد  $\mu$  می باشد با استفاده از کنترل کننده پایدارساز  $K$  و ماتریس مقیاس شده  $D$  بدست می آورد. این روش مسئله مینیمم سازی را در دو حالت مختلف انجام می دهد. در حالت اول مینیمم سازی روی ماتریس  $D$  انجام می شود و کنترل کننده  $K$  ثابت نگه داشته می شود. در حالت دیگر مینیمم سازی روی کنترل کننده  $K$  انجام می شود و ماتریس  $D$  ثابت نگه داشته می شود. در این روش مقادیر ویژه ساختار یافته با استفاده از حد بالای  $\mu$  محاسبه می شوند. با توجه به آنچه که در مورد محاسبه حد بالای  $\mu$  بیان شد می توان نتیجه گرفت که روش تکرار  $D-K$  حل دقیق مسئله سنتز  $\mu$  نیست. با این وجود کنترل کننده های بدست آمده از روش تکرار  $D-K$  به لحاظ معیارهای پایداری و عملکرد از قویترین روشهای در کنترل مقاوم هستند. این مطلب انگیزه را برای بدست آوردن جواب های بهتر برای مسئله سنتز  $\mu$  تقویت می کند.

روش تکرار  $D-K$  مانند غالب روش های کنترل مقاوم، با مشکل بالا بودن درجه کنترل کننده مواجه است. کنترل کننده های با درجه بالا هر چند که معیارهای پایداری و عملکرد سیستم حلقه بسته را به شکل بهتری برآورده می سازند ولی بعلاوه بالا بردن درجه پیچیدگی سیستم حلقه بسته و مشکلات موجود در ساخت، کمتر به لحاظ عملی مورد توجه قرار می

**قضیه ۴:** حد پایین  $\mu$  در معادله (۳)، به ازای تمام ساختارهای ماتریس عدم قطعیت با مقدار  $\mu$  ماتریس برابر است.

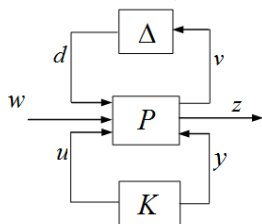
$$\max_{U \in \mathcal{U}} \rho(MU) = \mu_{\Delta}(M) \quad (1)$$

مشکل استفاده از حد پایین برای محاسبه  $\mu$  این است که این مسئله بهینه سازی معمولا ماکزیمم های محلی زیادی دارد. بنابراین احتمال همگرایی این مسئله کم است. بطور کلی محاسبه حد بالای  $\mu$  از محاسبه حد پایین ساده تر است [۱۱]. مشکل استفاده از حد بالا  $\mu$  این است که برخلاف حد پایین، این حد تنها زمانی که تعداد بلوک های عدم قطعیت اسکالر تکراری ( $S$ ) و بلوک های کامل ( $F$ ) طبق ساختار معادله (۱) در رابطه  $2S + F \leq 3$  صدق کنند با مقدار  $\mu$  ماتریس برابر می شود.

۱-۲- بررسی پایداری و عملکرد سیستم حلقه بسته بر

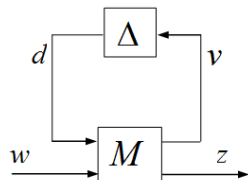
حساب معیار  $\mu$

هر سیستم شامل عدم قطعیت حلقه بسته را می توان به صورت ساختار شکل ۱ در آورد. در این شکل  $\Delta$  ماتریس عدم قطعیت است،  $P$  ماتریس تابع تبدیل حلقه باز نامی است و  $K$  کنترل کننده است. همچنین ورودیهای خارجی مانند سیگنالهای مرجع، اغتشاش و نویز را نشان می دهد،  $z$  نشان دهنده خروجی خطا است که معمولا شامل خطای ردیابی، خروجی تنظیم کننده و غیره می شود، سیگنالهای  $v$  و  $d$  هم ترتیب سیگنالهای خروجی و ورودی عدم قطعیت های دینامیک سیستم هستند.



شکل ۱: ساختار کلی مورد استفاده در سیستم های شامل عدم قطعیت

شکل ۲ ساختار کلی نشان داده شده در شکل قبلی را به طور ساده-تری نمایش می دهد. در این شکل ماتریس  $M$  از طریق رابطه  $M(P, K) = F_1(P, K)$  به ماتریس  $P$  و  $K$  در شکل ۱ ارتباط پیدا می کند. در این رابطه  $F_1(P, K)$  تبدیل کسری خطی پایینی می باشد.



شکل ۲: ساختار ساده شده

می توان ماتریس  $M$  را طبق رابطه (۷) تقسیم بندی کرد. در این تقسیم بندی ماتریس  $M_{11}$  جهت بررسی پایداری مقاوم و ماتریس  $M$  جهت بررسی عملکرد مقاوم سیستم حلقه بسته مورد استفاده قرار می گیرد.

بهتری دارد. حتی در بعضی موارد کنترل کننده مرتبه پایین بدست آمده از روش پیشنهادی نتایج مطلوب تری از کنترل کننده تکرار D-K مرتبه بالا نشان می دهد.

ادامه این مقاله به ترتیب زیر سازماندهی شده است: بخش دوم به بیان شکل کلی مسئله سنتز  $\mu$  می پردازد. در بخش سوم الگوریتم اجتماع پرندگان توضیح داده می شود. کاربرد الگوریتم اجتماع پرندگان در پیدا کردن حلی برای مسئله سنتز  $\mu$  در بخش چهارم مورد بررسی قرار گرفته است. در بخش پنجم نتایج شبیه سازیها ارائه شده است و سرانجام نتیجه گیری در بخش ۶ بیان شده است.

## ۲- تعریف مسئله سنتز $\mu$

برای تعریف مسئله سنتز  $\mu$ ، باید در ابتدا به مرور مفاهیم مقادیر ویژه ساختار یافته و ماتریس عدم قطعیت پردازیم. ماتریس عدم قطعیت  $\Delta \in \mathbb{C}^{n \times n}$  به صورت زیر تعریف می شود:

$$\Delta = \left\{ \begin{array}{l} \text{diag} [\delta_1 I_{r_1}, \dots, \delta_s I_{r_s}, \Delta_1, \dots, \Delta_F], \\ \delta_i \in \mathbb{C}, \Delta_j \in \mathbb{C}^{m_j \times m_j} \end{array} \right\} \quad (1)$$

همانگونه که مشاهده می شود ماتریس عدم قطعیت  $\Delta$  یک ماتریس قطری بلوکی است که در آن دو نوع بلوک عدم قطعیت وجود دارد: بلوک اسکالر تکراری  $\delta_i$  و بلوک کامل  $\Delta_j$ . مقادیر ویژه ساختار یافته  $\mu$  برای ماتریس  $M$  و ماتریس عدم قطعیت  $\Delta$  به صورت زیر تعریف می شود [۱].

$$\mu_{\Delta}(M) := \frac{1}{\min \{ \bar{\sigma}(\Delta) : \Delta \in \Delta, \det(I - M\Delta) = 0 \}} \quad (2)$$

در صورتیکه هیچ مقداری از  $\Delta \in \Delta$ ،  $I - M\Delta$  را منفرد نسازد آنگاه  $\mu_{\Delta}(M) := 0$ .

در ادامه، قضایای ۱ و ۲ مهمترین قضایایی هستند که در بدست آوردن حدود بالا و پایین  $\mu$  مورد استفاده قرار می گیرند.

**قضیه ۱:** می توان حد بالا و حد پایینی طبق نامساوی زیر برای  $\mu$  در نظر گرفت.

$$\max_{U \in \mathcal{U}} \rho(MU) \leq \mu_{\Delta}(M) \leq \inf_{D \in \mathcal{D}} \bar{\sigma}(DMD^{-1}) \quad (3)$$

در نامساوی ۳،  $\bar{\sigma}$  بزرگترین مقدار منفرد و  $\rho$ ، بزرگترین اندازه را در میان مقادیر ویژه ماتریس محاسبه می کند (شعاع طیفی). همچنین در این رابطه داریم:

$$\mathcal{U} = \{ U \in \Delta : UU^* = I_n \} \quad (4)$$

$$\mathcal{D} = \left\{ \text{diag} [D_1, \dots, D_S, d_1 I_{m_1}, \dots, d_{F-1} I_{m_{F-1}}, I_{m_F}] : \right. \\ \left. D_i \in \mathbb{C}^{n_i \times n_i}, D_i = D_i^* > 0, d_j \in \mathbb{R}, d_j > 0 \right\} \quad (5)$$

در اینجا هم بدون از دست دادن کلیت مسئله، فرض می کنیم که عدم قطعیت ها نرمالیزه اند. با این فرض، ساختار استاندارد شکل ۴ پایدار مقاوم است اگر  $\mu_{\Delta}(M(s)) < 1$  (یا  $\|M\|_{\mu} < 1$ ).

طراحی کنترل کننده با استفاده از معیارهای پایداری و عملکرد مقاوم ذکر شده، مسئله سنتز  $\mu$  نام دارد. رایج ترین روش حل مسئله سنتز  $\mu$  تکرار D-K نام دارد. کنترل کننده طراحی شده با این روش، پایداری و عملکرد سیستم حلقه بسته را بر اساس معیار  $\mu$  تضمین می کند. مبنای محاسبه در این روش حد بالا و بر اساس قضیه ۱ می باشد. روش تکرار D-K با یک الگوریتم تکرار می نیم معادله زیر را محاسبه می کند. این معادله مربوط به عملکرد مقاوم سیستم حلقه بسته است.

$$\inf_{K(s)} \sup_{w \in \mathbb{R}} \mu_{\Delta}[M(P, K)(jw)] \quad (11)$$

این روش بر اساس حل مسئله بهینه سازی رابطه (۱۲) است. در این رابطه مسئله بهینه سازی برای کنترل کننده پایدار ساز  $K$  و ماتریس مقیاس شده قطری و ثابت  $D$  انجام می شود.

$$\inf_{K(s)} \sup_{w \in \mathbb{R}} \inf_{D \in \mathbb{D}} \bar{\sigma}[DM(P, K)D^{-1}(jw)] \quad (12)$$

در این معادله مجموعه ماتریس مقیاس شده  $D$  در رابطه (۵) تعریف شده است. دلیل استفاده از این رابطه، با توجه به قضیه ۱ و حد بالای رابطه (۳) واضح می باشد. روش تکرار D-K مسئله مینیم سازی در رابطه فوق را برای دو حالت مختلف انجام می دهد. در حالت اول حداقل سازی روی  $D$  انجام می شود و در حالت دیگر حداقل سازی روی  $K$  انجام می شود و ماتریس  $D$  ثابت نگه داشته می شود. با توجه به آنچه در مورد حد بالای  $\mu$  بیان شد می توان گفت که روش تکرار D-K بهترین حل مسئله سنتز  $\mu$  نیست. با این وجود کنترل کننده های بدست آمده از روش تکرار D-K به لحاظ معیارهای پایداری و عملکرد از قویترین روشهای کنترل مقاوم هستند. این مطلب انگیزه را برای بدست آوردن جواب های بهتر برای حل مسئله سنتز  $\mu$  تقویت می کند.

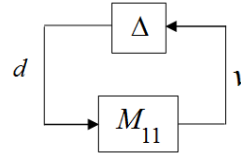
مهمترین اشکال در روش تکرار D-K بالا بودن درجه کنترل کننده است. بنابراین برای کنترل کننده مرتبه بالای بدست آمده ناگزیر هستیم تا از روشهای کاهش مرتبه استفاده کنیم تا کنترل کننده هایی با خواص مشابه اما با درجه پایین تر بدست آوریم. این عمل سبب می شود تا بعضی از اهداف طراحی تضعیف شود و یا حتی از دست برود. در این مقاله روشی ارائه می شود که مسئله سنتز  $\mu$  را با کنترل کننده های مرتبه پایین حل می کند و نیازی به کاهش مرتبه کنترل کننده ندارد.

### ۳- الگوریتم اجتماع پرندگان

الگوریتم اجتماع پرندگان الگوریتمی است که بر اساس رفتار اجتماعی پرندگان بنا نهاده شده است. این الگوریتم توسط کندی و ابرهارت ارائه شد [۱۰]. در الگوریتم اجتماع پرندگان هر پرنده ای در میان جمعیت یک ذره نام دارد. این ذره باید در طول مسیری که طی می

$$\begin{bmatrix} v \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}}_M \begin{bmatrix} d \\ w \end{bmatrix} \quad (7)$$

با توجه به رابطه فوق ساختار مورد استفاده در بررسی پایداری سیستم حلقه بسته بر حسب معیار  $\mu$  در شکل ۳ نشان داده شده است.

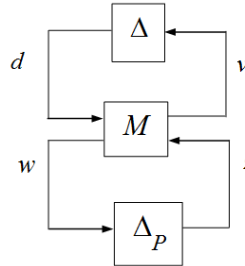


شکل ۳: ساختار استاندارد جهت تست پایداری مقاوم

چنانچه  $M_{11}$  ماتریس تابع تبدیل حلقه بسته باشد که با توجه به مجموعه عدم قطعیت  $\Delta$  شکل داده شده است، مقادیر ویژه ساختار یافته  $M_{11}(s)$  در رابطه زیر نشان دهنده پایداری مقاوم سیستم حلقه بسته است. این رابطه نتیجه ای از کاربرد قضیه بهره کوچک می باشد.

$$\mu_{\Delta}(M_{11}(s)) := \sup_{w \in \mathbb{R}} \mu_{\Delta}(M_{11}(jw)) \quad (8)$$

بدون از دست دادن کلیت مسئله، فرض می کنیم که عدم قطعیت ها نرمالیزه شده اند. با این فرض می توان گفت: ساختار استاندارد شکل ۳ پایدار مقاوم است اگر  $M_{11}(s)$  پایدار باشد و  $\mu_{\Delta}(M_{11}(s)) < 1$  (یا  $\|M_{11}\|_{\mu} < 1$ ).



شکل ۴: ساختار استاندارد جهت تست عملکرد مقاوم

تضعیف سیگنال خروجی  $z$  نسبت به سیگنال ورودی  $w$  را می توان به عنوان عملکرد مقاوم تعبیر کرد. اگر عملکرد مقاوم را، با توجه به ساختار شکل ۴ مورد بررسی قرار دهیم، عملکرد مقاوم، معادل مسئله پایداری مقاوم شکل ۲ است با این تفاوت که ماتریس عدم قطعیت  $\Delta$  باید با ماتریس عدم قطعیت  $\tilde{\Delta}$  در رابطه (۹) جایگزین شود. به این ترتیب مقادیر ویژه ساختار یافته ماتریس  $M$  با عدم قطعیت  $\tilde{\Delta}$  در رابطه (۱۰) نشان دهنده عملکرد سیستم حلقه بسته است.

$$\tilde{\Delta} \in \tilde{\Delta} := \left\{ \text{diag} \{ \Delta, \Delta_P \} : \Delta \in B\Delta, \|\Delta_P\|_{\infty} \leq 1 \right\} \quad (9)$$

$$B\Delta = \left\{ \Delta \in \Delta : \bar{\sigma}(\Delta) \leq 1 \right\}$$

$$\mu_{\tilde{\Delta}}[M(s)] := \sup_{w \in \mathbb{R}} \mu_{\tilde{\Delta}}(M(jw)) \quad (10)$$

## ۴- کاربرد الگوریتم اجتماع پرندگان در حل

مسئله سنتز  $\mu$ 

برای استفاده از الگوریتم اجتماع پرندگان در حل مسئله سنتز  $\mu$ ، در ابتدا بایستی این مسئله را به شکل یک مسئله بهینه سازی در آوریم. برای این کار مسئله سنتز  $\mu$  را با توجه به ساختاری که در بخش ۲ معرفی شد به شکل رابطه (۱۷) می نویسیم.

$$\min_{K(s)} \|M(s)\|_{\mu} \quad (17)$$

such that:  $\|M_{11}(s)\|_{\mu} < 1$

به این ترتیب مسئله سنتز  $\mu$  به یک مسئله بهینه سازی مقید تبدیل می شود که قید آن پایداری بر حسب معیار  $\mu$  می باشد. مبنای محاسبه نرم  $\mu$  در معادله ۱۷ مانند روش D-K حد بالای آن می باشد. برای حل این مسئله، درجه کنترل کننده را از یک شروع می کنیم و تا زمانیکه به کنترل کننده با مشخصات مطلوب برسیم درجه آن را یک واحد یک واحد افزایش می دهیم. شکل کلی کنترل کننده از مرتبه  $n$  را به صورت زیر انتخاب می کنیم.

$$\frac{b_n s^n + b_{n-1} s^{n-1} + \dots + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_0} \quad (18)$$

به این ترتیب پارامترهای مجهول به صورت بردار  $[a_{n-1}, \dots, a_0, b_n, \dots, b_0]$  در می آیند.

الگوریتم پیشنهادی دارای مراحل زیر می باشد:

گام اول: درجه کنترل کننده را ۱ انتخاب کنید.

گام دوم: ضرایب تابع تبدیل کنترل کننده را مطابق رابطه (۱۸) به عنوان ذرات در نظر بگیرید.

گام سوم: پارامترهای الگوریتم اجتماع پرندگان مانند جمعیت ذرات، مقدار اولیه و نهایی پارامترهای  $w, C_1, C_2$ ، محدودیت در فضای جستجو، محدودیت در سرعت ذرات و تعداد دفعات تکرار الگوریتم را تعیین کنید.

گام چهارم: یک جمعیت اولیه از ذرات که در قید پایداری مسئله صدق می کنند بدست آورید.

گام پنجم: تابع هزینه که همان عملکرد سیستم حلقه بسته طبق معیار  $\mu$  است را برای این ذرات بدست آورید و  $P_{id}$  و  $P_{gd}$  را تعیین کنید.

گام ششم: مکان ذرات و سرعت آنها را طبق معادلات (۱۳) و (۱۴) بروز کنید. البته در این مرحله باید قیدهای موجود در محدوده فضای جستجو و سرعت ذرات را بر روی سرعت ها و مکان های بدست آمده اعمال کنیم.

کند همواره بهترین حلی که خودش بدست آورده و بهترین حلی که جمعیت بدست آورده را در ذهن داشته باشد. هر ذره با حرکت در فضای جستجو اطلاعاتش را با ذرات دیگر از طریق معادلات (۱۳) و (۱۴) مبادله می کند.

$$V_{id} = wV_{id} + c_1 r_1 (P_{id} - X_{id}) + c_2 r_2 (P_{gd} - X_{id}) \quad (13)$$

$$X_{id} = X_{id} + V_{id}, \text{ for } d = 1, 2, \dots, N \quad i = 1, 2, \dots, S \quad (14)$$

در این معادلات  $X_{id}$  مکان فعلی ذره،  $P_{id}$  بهترین تجربه شخصی ذره و  $P_{gd}$  بهترین تجربه دسته ذرات است.  $C_1$  و  $C_2$  پارامترهای فردی و اجتماعی هستند و ضرایب شتاب نامیده می شوند،  $r_1$  و  $r_2$  اعداد تصادفی بین ۰ و ۱ هستند و  $w$  وزن اینرسی است که جهت متعادل کردن قابلیت های جستجوی محلی و کلی استفاده می شود. یک وزن اینرسی بزرگ امکان جستجوی کلی را فراهم می کند در حالیکه یک وزن اینرسی کوچک امکان جستجوی محلی را ممکن می سازد. در یک مطالعه تجربی روی الگوریتم اجتماع پرندگان [۱۵] شی و ابرهاریت ادعا کردند که یک وزن اینرسی که بصورت خطی کاهش پیدا می کند می تواند قابلیت جستجوی محلی را، نسبت به حالتی که وزن اینرسی ثابت باشد، بهبود بخشد. بنابراین یک تابع وزن اینرسی را می توان به صورت رابطه (۱۵) نشان داد.

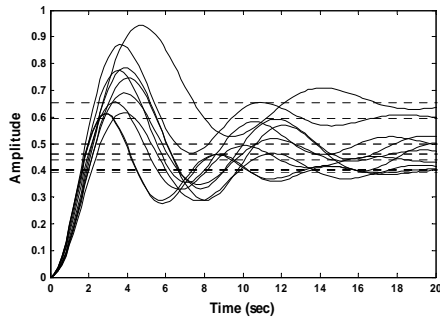
$$w = (iter_{max} - iter_{cur}) \cdot \left( \frac{w_{initial} - w_{final}}{iter_{max}} \right) + w_{final} \quad (15)$$

در این معادله  $w_{initial}$  و  $w_{final}$  بترتیب مقادیر وزن اینرسی را در ابتدا و انتهای الگوریتم نشان می دهند،  $iter_{max}$  تعداد تکرار الگوریتم و  $iter_{cur}$  تکراری است که الگوریتم در حال حاضر در آن قرار دارد. به هر ترتیب، ممکن است به خاطر استفاده از وزن اینرسی کاهش خطی، قابلیت جستجوی کلی در انتهای اجرای برنامه مناسب نباشد. این مسئله سبب می شود که در مسائل بهینه سازی پیچیده، امکان دسترسی به جواب مناسب فراهم نشود. این مشکل را می توان با اعمال ضرایب شتاب متغیر حل کرد. سوگانتان در [۱۶] از الگوریتم بهینه سازی استفاده کرد که در آن ضرایب شتاب به صورت خطی با افزایش تعداد تکرار کاهش پیدا می کردند. اما این روش نسبت به ضرایب شتاب ثابت جواب بهینه تری را ارائه نمی دهد. راناورا در [۱۷] همگرایی ذرات به سمت می نیمم کلی را با انتخاب ضریب  $C_1$  به صورت یک تابع کاهشی از شماره تکرار الگوریتم و ضریب  $C_2$  به صورت تابع افزایشی از تکرار الگوریتم، بهبود بخشید. در این صورت ضرایب شتاب به صورت زیر قابل بیان هستند.

$$c_1 = c_{1s} + iter_{cur} (c_{1e} - c_{1s}) / iter_{max} \quad (16)$$

$$c_2 = c_{2s} + iter_{cur} (c_{2e} - c_{2s}) / iter_{max}$$

در معادله فوق  $C_{1s}$  و  $C_{1e}$  مقادیر اولیه توابع  $C_1$  و  $C_2$  هستند و  $C_{2e}$  مقادیر نهایی این توابع را نشان می دهند.

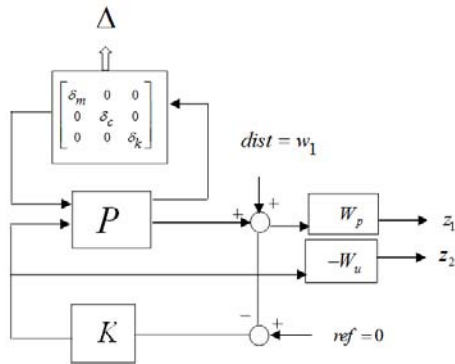


شکل ۶: پاسخ پله سیستم جرم-فنر-میراگر برای مقادیر مختلف پارامترها

ساختار استاندارد سیستم جرم-فنر-میراگر جهت طراحی کنترل کننده مقاوم در شکل ۷ نشان داده شده است. برای جزئیات بیشتر می توان به [۱۴] مراجعه کرد. توابع وزنی  $W_p$  و  $W_u$  جهت برآوردن نیازهای عملکردی سیستم حلقه بسته در فرکانس های مختلف استفاده شده است.

$W_p$  به صورت تابع تبدیل  $W_p(s) = 0.95 \frac{s^2 + 1}{s^2 + 1.0s + 1.0}$  انتخاب شده است.

این تابع تبدیل جدا از یک تضعیف اختلال خوب، یک پاسخ گذرای خوب (زمان نشست کمتر از ۱۰ ثانیه و بالادگی کمتر از ۲۰ درصد برای سیستم نامی) را تضمین می کند. همچنین تابع وزنی کنترل  $W_u$  به صورت اسکالر  $W_u = 10^{-7}$  انتخاب شده است. ماتریس  $\Delta$  ماتریس عدم قطعیت سیستم است و  $K$  کنترل کننده است.



شکل ۷: ساختار استاندارد سیستم جرم-فنر-میراگر جهت طراحی کنترل کننده مقاوم

همچنین ماتریس  $P$  به صورت رابطه (۲۰) می باشد.

$$P = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \quad (20)$$

که در آن داریم:

گام هفتم: امتحان کنید که آیا ذرات جدید در قید پایداری مسئله صدق می کنند یا نه. چنانچه در قید پایداری صدق نمی کردند از مکان قبلی ذره استفاده کنید.

گام هشتم: تابع هزینه ذرات جدید را بدست آورید و  $P_{gd}$  و  $P_{id}$  مربوط به آنها را محاسبه کنید.

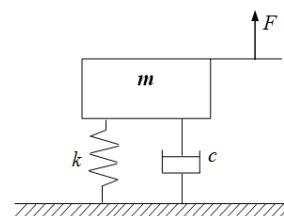
گام نهم: تا زمانی که تعداد تکرارها به پایان نرسیده است به گام ششم بازگردید. در صورتیکه تعداد تکرارها به پایان رسیده به مرحله بعدی بروید.

گام دهم: چنانچه حداقل مقدار  $\|M\|_{\mu}$  که همان عدد همگرایی الگوریتم است کمتر از ۱ بود به مرحله بعد می رویم. در غیر این صورت درجه کنترل کننده را یک واحد اضافه می کنیم و به گام دوم می رویم.

گام یازدهم: ضرائب مربوط به عدد همگرا شده الگوریتم، کنترل کننده ای را مشخص می کند که جوابی برای مسئله سنتز می باشد.

## ۵- سیستم جرم- فنر-میراگر شامل عدم قطعیت

به منظور ارزیابی روش ارائه شده، از آن به منظور طراحی کنترل کننده برای یک سیستم جرم-فنر-میراگر استفاده می شود. سیستم جرم-فنر-میراگر در شکل ۵ نشان داده است. در یک سیستم واقعی سه پارامتر فیزیکی این سیستم یعنی جرم  $m$ ، ثابت  $c$  و ثابت فنر  $k$  به طور دقیق شناخته شده نیستند. به هر ترتیب می توان فرض کرد که مقادیر آنها در بازه های مشخصی قرار دارد.



شکل ۵: سیستم جرم-فنر-میراگر

برای مثال مورد بررسی این مقاله که در [۱۴] آمده است داریم:

$$m = \bar{m}(1 + p_m \delta_m), c = \bar{c}(1 + p_c \delta_c), k = \bar{k}(1 + p_k \delta_k) \quad (19)$$

که در آن  $\bar{m} = 3$ ،  $\bar{c} = 1$  و  $\bar{k} = 2$  مقادیر نامی پارامترهای هستند. همچنین  $p_m = 0.4$ ،  $p_c = 0.2$ ،  $p_k = 0.3$  و  $-1 \leq \delta_m, \delta_c, \delta_k \leq 1$  میزان عدم قطعیت در پارامترهای سیستم را نشان می دهند. پاسخ پله این سیستم برای مقادیر مختلف پارامترهای دینامیک سیستم در شکل ۶ نشان داده شده است.

کنیم. کنترل کننده مرتبه ۴ کوچکترین کنترل کننده به لحاظ مرتبه است که معیارهای پایداری و عملکرد  $\mu$  را برآورده می کند. این دو کنترل کننده در جدول ۲ مورد بررسی قرار گرفته اند.

جدول ۲: بررسی کنترل کننده های تکرار D-K و تکرار D-K کاهش مرتبه یافته

معیار	معیار	روش
عملکرد $\mu$	پایداری $\mu$	
۰/۹۶۴۷	۰/۴۶۲۹	کنترل کننده تکرار D-K از مرتبه ۲۰
۰/۹۷۰۹	۰/۴۶۴۴	کنترل کننده تکرار D-K (مرتبه کاهش یافته) از مرتبه ۴

روش ارائه شده در این مقاله منجر به بدست آمدن کنترل کننده ای از مرتبه دوم می شود اما به منظور بررسی کاملتر، کنترل کننده های بدست آمده از روش تکرار D-K که مرتبه آنها به ۱، ۲ و ۳ کاهش یافته است را با کنترل کننده های بدست آمده از روش پیشنهادی مقایسه می کنیم. جدول ۳ این مقایسه را برای تمامی این کنترل کننده ها به همراه کنترل کننده های جدول ۲ انجام می دهد.

جدول ۳: مقایسه کنترل کننده های کاهش مرتبه یافته روش تکرار D-K و روش پیشنهادی

معیار	معیار	تابع تبدیل	نوع کنترل کننده
عملکرد $\mu$	پایداری $\mu$		
۴/۸۱۰۳	۰/۹۴۱۵	$\frac{1/71s + 0.52}{s}$	مرتبه ۱ پیشنهادی
بینهایت	بینهایت	$\frac{1/94s + 3/31}{s}$	مرتبه ۱ تکرار D-K
۰/۹۷۵۵	۰/۴۰۱۴	$\frac{22/17s^2 + 15/51s + 12/33}{s^2 + 2/0.8s}$	مرتبه ۲ پیشنهادی
۱۱/۶۱۵۸	۲/۵۴۵۶	$\frac{-14/24s^2 + 1927s + 341/2}{s^2 + 107/4s + 1/13}$	مرتبه ۲ تکرار D-K
۰/۹۶۴۶	۰/۴۰۰۴	$\frac{16/71s^3 + 25/00s^2 + 14/04s + 8/60}{s^3 + 2/80s^2 + 1/71s}$	مرتبه ۳ پیشنهادی
۱/۰۲۴۱	۰/۴۷۹۴	$\frac{-1/44s^3 + 449/2s^2 + 459/9s + 266/5}{s^3 + 13/37s^2 + 82/79s + 1/10}$	مرتبه ۳ تکرار D-K
۰/۹۷۰۹	۰/۴۶۴۴	تابع تبدیل از مرتبه ۴	مرتبه ۴ تکرار D-K
۰/۹۶۴۷	۰/۴۶۲۹	تابع تبدیل از مرتبه ۲۰	تکرار D-K

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{\bar{k}}{m} & -\frac{\bar{c}}{m} \end{bmatrix}, C_1 = \begin{bmatrix} -\frac{\bar{k}}{m} & -\frac{\bar{c}}{m} \\ 0 & \bar{c} \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 \\ -p_m & -\frac{p_c}{m} & -\frac{p_k}{m} \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ 1 \\ m \end{bmatrix}$$

$$D_{11} = \begin{bmatrix} -p_m & -\frac{p_c}{m} & -\frac{p_k}{m} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, D_{12} = \begin{bmatrix} 1 \\ m \\ 0 \\ 0 \end{bmatrix}$$

با توجه به این ساختارها و طبق مطالب بخش ۲ براحتی می توان معیارهای پایداری و عملکرد را بدست آورد.

## ۶- نتایج شبیه سازی

برای بررسی کارایی روش پیشنهادی در حل مسئله سنتز  $\mu$ ، از آن به منظور طراحی کنترل کننده مقاوم برای سیستم جرم-فنر-میراگر استفاده می کنیم. مشخصات این سیستم شامل عدم قطعیت در بخش قبلی توصیف شد. نتایج شبیه سازی بوسیله کامپیوتری با Quad2Core Cpu با فرکانس 2.66GHz و حافظه 1GB RAM و در متلب 7.11.0 (R2010b) بدست آمده است. زمان متوسط محاسباتی هر اجرای برنامه حدود ۶۰۰ ثانیه می باشد. جدول ۱ پارامترهای مورد استفاده در الگوریتم اجتماع پرندگان و مقادیر آنها را نشان می دهد.

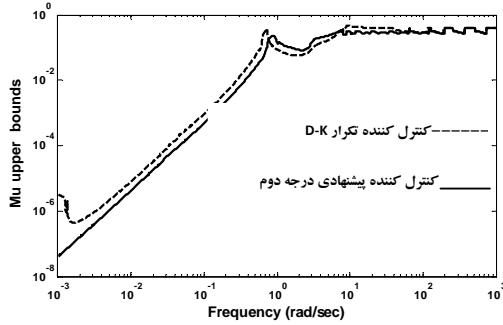
جدول ۱: پارامترهای مورد استفاده در الگوریتم اجتماع پرندگان و مقادیر آنها

مقدار پارامتر	پارامتر الگوریتم	مقدار پارامتر	پارامتر الگوریتم
۰/۴	$w_{final}$	۴۰	جمعیت ذرات
۰/۵	$C_{1e}$	۱۲۰	تعداد تکرار
۱/۵	$C_{1s}$	[۰ ۳۰]	محدوده ذرات
۱/۵	$C_{2e}$	[۰ ۳]	محدوده سرعت
۰/۵	$C_{2s}$	۱	$w_{initial}$

۶-۱- نگاهی کلی به نتایج الگوریتم پیشنهادی در مقایسه

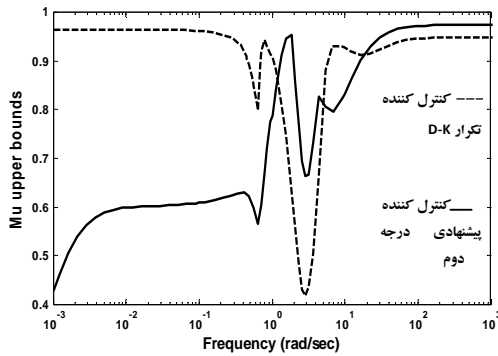
### با روش تکرار D-K

کنترل کننده بدست آمده از روش تکرار D-K، کنترل کننده ای از مرتبه ۲۰ می باشد. درجه این کنترل کننده بسیار بالا می باشد لذا با روشهای کاهش مرتبه، آن را به کنترل کننده ای از مرتبه ۴ تبدیل می



شکل ۹: مقایسه معیار پایداری کنترل کننده تکرار D-K مرتبه ۲۰ و کنترل کننده مرتبه دوم پیشنهادی

از شکل ۹ می توان دریافت که در بیشتر بازه فرکانسی کنترل کننده پیشنهادی نسبت به کنترل کننده تکرار D-K به لحاظ پایداری مقاوم تر است. شکل ۱۰ به مقایسه عملکرد مقاوم این دو کنترل کننده می پردازد. همانطوریکه از این شکل مشاهده می شود در بسیاری از بازه های فرکانسی، عملکرد کنترل کننده مرتبه دوم از کنترل کننده تکرار D-K مقاوم تر است.



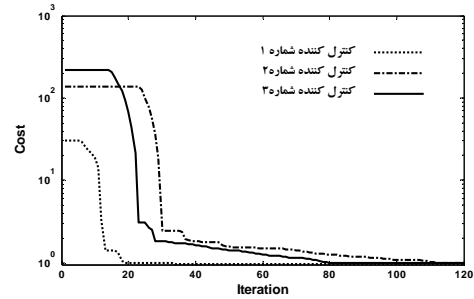
شکل ۱۰: مقایسه معیار پایداری کنترل کننده تکرار D-K مرتبه ۲۰ و کنترل کننده مرتبه دوم پیشنهادی

### ۳-۶- بررسی نتایج در حوزه زمان

در این بخش به تحلیل سیستم های حلقه بسته در حوزه زمان می پردازیم. به این منظور پاسخ سیستم های حلقه بسته نامی به ورودیهای پله و پالسی مورد بررسی قرار گرفته است. شکل های ۱۱ و ۱۲ پاسخ پله سیستم های حلقه بسته با هر دو کنترل کننده را، مقایسه می کند. شکل ۱۱ پاسخ خروجی مربوط به سیگنال مرجع ورودی و شکل ۱۲ پاسخ خروجی مربوط به سیگنال اغتشاش ورودی است.

این شکل ها نشان می دهند که سیستم حلقه بسته با کنترل کننده تکرار D-K بالا زدگی کمتری نسبت به کنترل کننده مرتبه دوم پیشنهادی دارد اما سرعت پاسخ کنترل کننده مرتبه دوم پیشنهادی به مراتب بهتر است.

نحوه همگرایی الگوریتم سیر تکاملی برای جمعیت های اولیه مختلف در شکل ۸ نشان داده شده است. این منحنی ها، مربوط به ساختار کنترل کننده مرتبه دوم می باشد که نحوه همگرایی آن در ۳ بار اجرای برنامه به نمایش درآمده است. این نمودارها نشان می دهند که جمعیت های اولیه مختلف در همگرایی الگوریتم به مقدار نهایی تأثیری ندارد.



شکل ۸: همگرایی الگوریتم سیر تکاملی برای ساختار کنترل کننده مرتبه دوم

از مشاهده نتایج جداول ۲ و ۳ می توان نتیجه گرفت که کنترل کننده های بدست آمده از الگوریتم پیشنهادی، بسیار کارتر از کنترل کننده های کاهش مرتبه یافته تکرار D-K می باشند. نکته جالبی که در جدول ۳ می توان مشاهده نمود این است که کنترل کننده مرتبه سوم بدست آمده از الگوریتم پیشنهادی هم در معیار پایداری و هم عملکرد مقاوم، بهتر از کنترل کننده تکرار D-K از مرتبه ۲۰ عمل می کند. این مسئله دلیلی بر کارایی و برتری این روش نسبت به روش تکرار D-K در مثال مورد بررسی است.

به هر ترتیب به دلیل مزایای استفاده از کنترل کننده های مرتبه پایین تر، در ادامه کنترل کننده مرتبه دوم بدست آمده از روش پیشنهادی را با کنترل کننده تکرار D-K از مرتبه ۲۰ مقایسه می کنیم. این مقایسه را ابتدا در حوزه فرکانس و سپس در حوزه زمان انجام می دهیم.

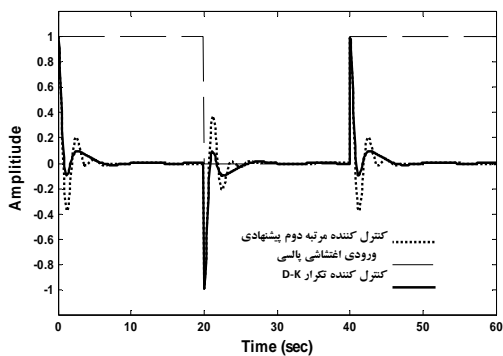
### ۲-۶- بررسی نتایج در حوزه فرکانس

با توجه به جدول ۳، تابع تبدیل کنترل کننده پیشنهادی مرتبه دوم،

$$\text{به صورت } K(s) = \frac{22/17s^2 + 15/51s + 12/33}{s^2 + 2/0.8s} \text{ می باشد و این}$$

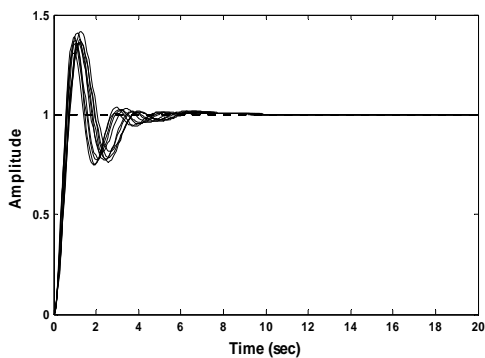
کنترل کننده معیارهای پایداری و عملکرد  $\mu$  را برآورده می سازد. بر اساس نتایج این جدول، سیستم حلقه بسته با کنترل کننده مرتبه دوم نسبت به کنترل کننده تکرار D-K از نظر پایداری مقاوم تر است و عملکرد مقاوم آن کمی ضعیف تر از کنترل کننده تکرار D-K می باشد. شکل های ۹ و ۱۰ به مقایسه پایداری و عملکرد مقاوم این دو کنترل کننده در بازه فرکانسی می پردازند.



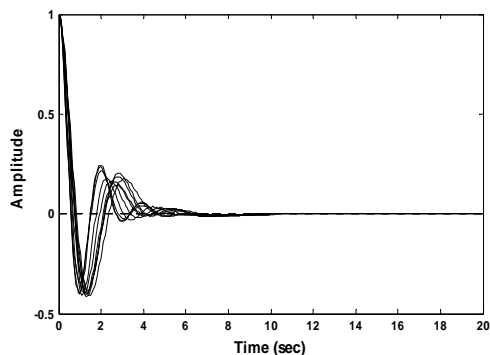


شکل ۱۴: مقایسه پاسخ سیستم حلقه بسته با کنترل کننده تکرار D-K مرتبه ۲۰ و کنترل کننده مرتبه دوم پیشنهادی به ورودی اغتشاش پالسی

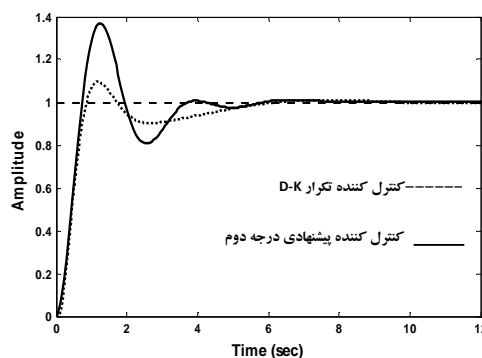
بررسی شکل ها نشان می دهد پاسخ سیستم حلقه بسته به ورودی پالسی سیگنال مرجع برای کنترل کننده مرتبه دوم بدست آمده از الگوریتم پیشنهادی دارای سرعت بهتری است. اما میزان بالادگی آن بیشتر از کنترل کننده D-K است. در مورد حذف اغتشاش و پاسخ به ورودی پالسی اغتشاش نیز شرایط مشابهی وجود دارد.



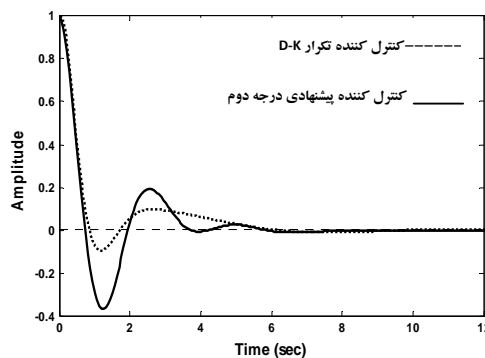
شکل ۱۵: پاسخ پله سیستم حلقه بسته با کنترل کننده مرتبه دوم پیشنهادی به ورودی مرجع پله در حضور عدم قطعیت دینامیک سیستم



شکل ۱۶: پاسخ پله سیستم حلقه بسته با کنترل کننده مرتبه دوم پیشنهادی به ورودی اغتشاش پله در حضور عدم قطعیت دینامیک سیستم

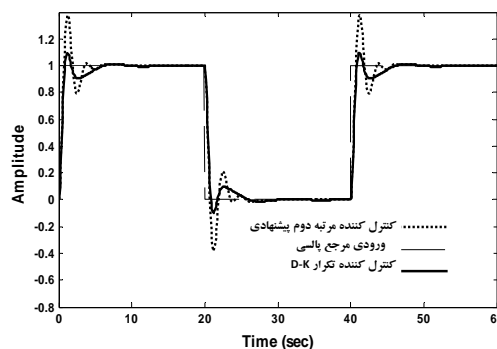


شکل ۱۱: مقایسه پاسخ سیستم حلقه بسته با کنترل کننده تکرار D-K مرتبه ۲۰ و کنترل کننده مرتبه ۲ پیشنهادی به ورودی پله سیگنال مرجع



شکل ۱۲: مقایسه پاسخ سیستم حلقه بسته با کنترل کننده تکرار D-K مرتبه ۲۰ و کنترل کننده مرتبه دوم پیشنهادی به ورودی پله اغتشاش

به منظور بررسی بیشتر، پاسخ ورودی پالسی سیستم های حلقه بسته با هر دو کنترل کننده مقایسه شده است. شکل ۱۳ پاسخ خروجی مربوط به سیگنال مرجع ورودی و شکل ۱۴ پاسخ خروجی مربوط به سیگنال اغتشاش ورودی است.



شکل ۱۳: مقایسه پاسخ سیستم حلقه بسته با کنترل کننده تکرار D-K مرتبه ۲۰ و کنترل کننده مرتبه دوم پیشنهادی به ورودی پالسی سیگنال مرجع

- [6] Ho, S.J., Ho, S.Y., Hung, M.H., Shu, L.S., Huang, H.L., 2005, "Designing Structure-Specified Mixed  $H_2/H_\infty$  Optimal Controllers Using an Intelligent Genetic Algorithm IGA", IEEE Trans. On Control Systems, 13(6): 1119-24.
- [7] S. Kaitwanidvilai and M. Parnichkun, "Design of Structured Controller Satisfying H Infinite Loop Shaping using Evolutionary Optimization: Application to a Pneumatic Robot Arm", Engineering Letters, 2008, pp.193-201.
- [8] M. Zamini, N. Sadati and M. Karimi Ghartemai, "Design of an  $H_\infty$  PID Controller Using Particle Swarm Optimization", International Journal of Control, Automation, and Systems, vol. 7, no. 2, pp.273-280, 2009.
- [9] Ramzy S. Ali Al-Waily, "Design of Robust Mixed  $H_2/H_\infty$  PID Controller Using Particle Swarm Optimization ", IJACT : International Journal of Advancements in Computing Technology, vol. 2, no. 5, pp. 53 - 60, 2010.
- [10] Kennedy, J., Eberhart, R., 1995, "Particle swarm optimization", In Proceedings of the IEEE International Conference on Neural Networks, 1942-1948.
- [11] Kao, C. C., Chuang, C. W., Fung, R. F., 2006, "The self-tuning PID control in a slider-crank mechanism system by applying particle swarm optimization approach". Mechatronics, 16(8) :513-522.
- [12] Mukherjee, V., Ghoshal, S. P., 2007, "Intelligent particle swarm optimized fuzzy PID controller for AVR system", Electric Power Systems Research, 77(12), 1689-1698.
- [13] Modares, H., NaghibiSistani, M., 2010, "Solving nonlinear optimal control problems using a hybrid IPSO-SQP algorithm", Engineering Applications of Artificial Intelligence, Engineering Applications of Artificial Intelligence, 24, 476-484.
- [14] Gu, Da.Wei., Petkov, Petko.Hristov., Konstantinov, Mihail.Mihaylov., "Robust Control Design with Matlab", Springer, February 2005.
- [15] Shi, Y., Eberhart, R.C., 1998, "Parameter selection in particle swarm optimization", In: Evolutionary Programming VII. Proc. EP98, pp. 591-600. Springer, New York .
- [16] Suganthan, P.N., 1999, "Particle Swarm Optimizer with Neighborhood Operator" , in Proceedings of IEEE International Conference on Evolutionary Computation. Washington D.C.: IEEE Press , pp. 1958-1962.
- [17] Ratnaweera, A., Halgamuge, S.K., Watson, H.C., 2004, "Self Organizing Hierarchical Particle Swarm Optimizer with Time Varying Acceleration Coefficients", IEEE Transactions on Evolutionary Computation 8 (3), 240-255.

در شکل ۱۵ پاسخ پله ورودی مرجع سیستم حلقه بسته با کنترل کننده مرتبه دوم پیشنهادی، در حضور عدم قطعیت های موجود در دینامیک سیستم نشان داده شده است. از مقایسه با شکل ۶ می توان دید که سیستم حلقه بسته با کنترل کننده مرتبه دوم پیشنهادی، در برابر عدم قطعیت های دینامیک سیستم کاملا مقاوم است. همچنین شکل ۱۶ پاسخ پله ورودی اغتشاش را برای همین شرایط نشان می دهد. این شکل نیز بیانگر حذف اغتشاش پله در خروجی سیستم برای مقادیر مختلف عدم قطعیت است.

## ۷- نتیجه گیری

این مقاله روشی برای حل مسئله سنتز  $H_2$  بر اساس الگوریتم سیر تکاملی ارائه نموده است. روش تکرار D-K که یک حل تقریبی مسئله سنتز  $H_2$  است، معمولا به طراحی کنترل کننده های مرتبه بالا منجر می شود. بنابراین ناگزیر به استفاده از روش های کاهش مرتبه می شویم که سبب تضعیف و یا حتی از دست دادن اهداف طراحی می شود. حل مسئله سنتز  $H_2$  به کمک الگوریتم اجتماع پرندگان نیازی به کاهش مرتبه کنترل کننده ندارد و نتیجه این حل، کوچکترین کنترل کننده به لحاظ مرتبه است که در معیارهای پایداری و عملکرد  $H_2$  صدق می کند. از یک سیستم جرم-فنر-میراگر شامل عدم قطعیت، جهت بررسی کارایی روش پیشنهادی استفاده شده است. نتایج شبیه سازی ها نشان می دهند که کنترل کننده های بدست آمده از الگوریتم پیشنهادی در مقایسه با کنترل کننده های هم درجه (درجه کاهش مرتبه یافته) روش تکرار D-K، به لحاظ پایداری و عملکرد مقاوم سیستم حلقه بسته دارای کارایی بهتر و مطلوب تری می باشند. در میان کنترل کننده های بدست آمده از روش پیشنهادی، حتی می توان کنترل کننده هایی پیدا نمود که نسبت به کنترل کننده روش تکرار D-K مرتبه بالاتر پایداری و عملکرد مقاوم وضعیت بهتری داشته باشند.

## مراجع

- [1] Zhou, Kemin., Doyle, John., Essentials of Robust Control, Prentice Hall, Upper Saddle River, NJ, 1998.
- [2] Karimaghvae, P., Noroozi, Navid., 2011, "Frequency Weighted Discrete-Time Controller Order Reduction Using Bilinear Transformation", Journal of Electrical Engineering, Vol. 62, NO. 1, 2011, 44-48.
- [3] Wei-Der, C., Shun-Peng, S., 2010, "PID controller design of nonlinear systems using an improved particle swarm optimization approach", Commun Nonlinear Sci Numer Simulat 15, 3632-3639.
- [4] Kaitwanidvilai, S., Olanthichachat, P., 2011, "Robust loop shaping-fuzzy gain scheduling control of a servo-pneumatic system using particle swarm optimization approach", Mechatronics 21, 11-21.
- [5] Chen, B.S., Cheng, Y.M., Lee, C.H., 1995, "A Genetic Approach to Mixed  $H_2/H_\infty$  Optimal PID Control", IEEE Trans, On Control Systems, p. 51-60.