



Optimization of Model Predictive Control Horizons Using Particle Swarm Algorithm to Synchronize Marine Simulator Motion

Navid Moshtaghi Yazdani^{1*}, Mohammad Hasan Olyaei Torqabeh²

¹PhD, Department of Electrical Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran.

²MSc., Department of Electrical Engineering, Faculty of Electrical Engineering and Biomedical Engineering, Sadjad University of Technology, Mashhad, Iran.

ARTICLE INFO

Received: 02.24.2021

Revised: 10.09.2021

Accepted: 10.18.2021

Keyword:

Motion step algorithm

Predictive control

Particle swarm algorithm

Optimization

*Corresponding Author:

Navid Moshtaghi Yazdani

Email:

Navid.moshtaghi@alumni.ut.ac.ir

ABSTRACT

Marine simulators are effective tools for making a ship feel like driving by creating a similar environment using motion commands. The main problem with simulators is the limited workspace which does not allow them to generate accurate real-time floating movements, so they require a motion synchronization algorithm. Recently, the use of predictive control has become popular in marine simulators. Values of control horizon and future forecast affect the computational load. However, because the designer manually selects these horizons, they are lower than the optimal level. In this paper, a new method based on particle swarm algorithm was used to achieve the best control and forecast horizons by minimizing some periods such as sensory error, displacement and computational load. The proposed method eliminates the disadvantages of the MPC-MCA method such as time-consuming empirical estimation through trial and error for initial control and forecast horizons, while minimizing optimal cost performance and computational load. The simulation results showed the efficiency of the proposed method based on the improvement of performance output and computational load.





بهینه‌سازی افق‌های کنترل پیش‌بین مدل با استفاده از الگوریتم ازدحام ذرات در راستای هم‌گام‌سازی حرکت شبیه‌ساز دریایی

نوید مشتاقی یزدانی^{۱*}، محمد حسن علیائی^۲

۱- دکتری، گروه مهندسی برق، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران.

۲- کارشناسی ارشد، گروه مهندسی برق، دانشکده مهندسی برق و مهندسی پزشکی، دانشگاه صنعتی سجاد، مشهد، ایران.

چکیده

اطلاعات مقاله

شبیه‌سازهای دریایی، ابزارهای مؤثری برای احساس راندن یک شناور دریایی از طریق ایجاد یک محیط مشابه با استفاده از فرمان‌های حرکتی هستند. مشکل اصلی شبیه‌سازها فضای کار محدودی است که به آن‌ها اجازه نمی‌دهد تا حرکات دقیق شناور واقعی را ایجاد کنند؛ در نتیجه آن‌ها به الگوریتم هم‌گام‌سازی حرکت نیاز دارند. اخیراً استفاده از کنترل پیش‌بین در شبیه‌سازهای دریایی به محبوبیت رسیده است. درچه‌های افق کنترل و پیش‌بینی آینده بر بار محاسباتی تأثیر می‌گذارد اما از آن جا که این افق‌ها به صورت دستی توسط طراح انتخاب می‌شوند، پایین‌تر از سطح بهینه می‌باشند. در این مقاله، روشی نوین بر مبنای الگوریتم ازدحام ذرات برای دستیابی به بهترین افق‌های کنترل و پیش‌بینی با توجه به حداقل‌رسانی برخی از کمیت‌ها مانند خطای حسی، جابه‌جایی و بار محاسباتی به‌کار گرفته شده است. روش پیشنهادی معایب روش MPC-MCA مانند تخمین تجربی وقت‌گیر از طریق تکرار آزمون و خطا برای تعیین افق‌های کنترل و پیش‌بینی را برطرف می‌کند و در عین حال هزینه و بار محاسباتی را به حداقل می‌رساند. نتایج شبیه‌سازی، کارآمدی روش پیشنهادی را بر مبنای بهبود خروجی عملکرد و بار محاسباتی نشان می‌دهد.

دریافت مقاله: ۱۳۹۹/۱۲/۰۶

بازنگری مقاله: ۱۴۰۰/۰۷/۱۷

پذیرش مقاله: ۱۴۰۰/۰۷/۲۶

کلید واژگان:

الگوریتم هم‌گام‌سازی حرکت
کنترل پیش‌بین
الگوریتم ازدحام ذرات
بهینه‌سازی

*نویسنده مسئول: نوید مشتاقی یزدانی

پست الکترونیکی:

Navid.moshtaghi@alumni.ut.ac.ir



مقدمه

شبهه‌سازهای دریایی دستگاه‌های مقرون‌به‌صرفه‌ای هستند که کاربرد رو به افزایشی برای آموزش و نمونه‌سازی اولیه دارند. هدف یک شبهه‌ساز دریایی، ایجاد (بازخورد حرکت) برای کاربر است تا احساس واقعی کار را برای وی فراهم کند. بازخورد حرکت به کاربر کمک می‌کند تا اتفاقات طبیعی را در محیط شبهه‌سازی شده تجربه کند. محدودیت اصلی شبهه‌سازهای دریایی، محدودیت مرزهای فضای کاری است؛ در نتیجه، بازخورد حرکت باید توسط الگوریتمی که به‌عنوان الگوریتم هم‌گام‌سازی حرکت شناخته می‌شود به‌صورت حرکات امکان‌پذیری برای پلتفرم شبهه‌ساز دریایی تغییر شکل دهد تا تعادلی بین خطای احساس راندن و محدودیت‌های محل کار پلتفرم ایجاد کند [۷-۱].

سیستم دهلیزی در گوش داخلی قرار گرفته است و حرکت بدن انسان را تشخیص می‌دهد. این سیستم از استاتولیت‌ها که مسئول حس شتاب خطی و کانال‌های نیم‌دایره که مسئول احساس سرعت زاویه‌ای می‌باشند تشکیل شده است. اندام‌های استاتولیت نمی‌توانند تفاوت میان شتاب افقی و تأثیر گرانش را به‌وسیله گرایش کاربر که حول یک محور افقی ایجاد می‌شود، تشخیص دهند. این کار در تکنیکی به نام هماهنگی شتاب که در شبهه‌سازهای حرکتی برای ایجاد احساس شتاب‌های خطی پایدار بدون تجربه کردن جابه‌جایی زیاد پلتفرم به‌کار می‌رود انجام می‌شود. این مسئله براساس این نکته است که استاتولیت‌ها نمی‌توانند بین شتاب خطی پایدار و شتاب استاتیک که منشأ آن در گرانش متناسب با سر کاربر، مشابه با شتاب گرانش-اینرسی در یک وسیله نقلیه واقعی است، تمایز قائل شوند. شتاب گرانش-اینرسی مجموع بردار تمام شتاب‌های به‌کاررفته در بدن؛ از جمله شتاب گرانش است. بدون بینایی یا کانال‌های نیم‌دایره، انسان نمی‌تواند درک کند آیا منشأ شتاب گرانش-اینرسی ناشی از شتاب بدن یا حرکت سر او است. کاربر باید در نرخی تحت آستانه درک چرخش انسانی متمایل شود تا از احساس چرخش ناشی از شتابی که توسط کانال‌های نیم‌دایره حس می‌شود اجتناب کند [۸-۱۲]. از نظر سنتی، فیلترهای wash out برای اجرای الگوریتم هم‌گام‌سازی حرکت به‌کار گرفته شده‌اند. شتاب آغازین از طریق حرکت خطی پلتفرم ایجاد می‌شود و احساس شتاب پایدار با هماهنگی شتاب ایجاد می‌شود. فیلترهای wash out کلاسیک را محققان متعددی طراحی کرده‌اند و بهبود داده‌اند [۱۶-۱۳]. محققان Casas و Thöndel الگوریتم ژنتیک را برای بهینه‌سازی پارامترهای فیلترهای wash out کلاسیک به‌کار گرفتند اما نتوانستند در مورد ارزیابی عینی نتایج خود اطلاعات کافی ارائه کنند. فیلترهای wash out بهینه، نسل بعدی فیلترهای wash out بودند که براساس مفهوم تنظیم‌کننده درجه دوم خطی طراحی شدند [۱۷؛ ۱۸]. علی‌رغم این که هدف فیلترهای wash out بهینه، به‌حداقل رساندن خطای حسی بین شبهه‌ساز و یک کاربر واقعی است، این فیلترها از کمبودهای مربوط به محدودیت‌های فیزیکی شبهه‌ساز مانند نقض جابه‌جایی حداکثری و نیاز به الگوریتم هم‌گام‌سازی حرکت بهینه برای کار کردن در یک محیط کار بزرگ‌تر رنج می‌برند [۱۹-۲۱]. به‌علاوه، الگوریتم هم‌گام‌سازی حرکت بهینه مبتنی بر تنظیم‌کننده درجه دوم خطی محدودیت‌های دینامیک پلتفرم را در نظر نمی‌گیرد. از آن جا که مسئله محدودیت‌ها در فیلترهای wash out و تکنیک‌های هم‌گام‌سازی حرکت بهینه متعارف بسیار مهم است، یک روش برای تقلیل این محدودیت لازم است. علاوه بر این، به‌کارگیری الگوریتم هم‌گام‌سازی حرکت بهینه، مستلزم دستکاری در چند وزن مربوط به حالت‌هایی است که اغلب نمایش فیزیکی مشخصی ندارند که می‌توانند وظیفه تنظیم کردن را بسیار پیچیده کنند [۲۲؛ ۲۳]. نقطه‌ضعف اصلی تمام فیلترهای wash out این است که آن‌ها معمولاً در مورد محدودیت‌های شبهه‌ساز آگاهی ندارند و محدودیت‌های فیزیکی را می‌توان هر زمانی نقض کرد؛ بنابراین، فیلترهای wash out معمولاً به‌طور محافظه‌کارانه برای بدترین سناریو تنظیم می‌شوند. این تنظیم مانع از این می‌شود که پلتفرم حرکت از فضای کار استفاده بهینه کند. مشکلاتی که در ماهیت فیلترهای wash out وجود دارند محققان را ترغیب کردند تا کنترل پیش‌بین را در اجرای الگوریتم هم‌گام‌سازی حرکت به‌کار ببرند [۲۴]. در روش

کنترل پیش‌بین، محدودیت‌ها به‌صورت آشکار برای فرد کنترل‌کننده تعریف می‌شوند. در این نوع کنترل، هدف، به‌دست آوردن بهترین ورودی برای بهینه‌سازی پاسخ آتی است. همچنین این روش، ایراد اساسی کنترل مود لغزشی ناشی از اثر چترینگ^۱ را ندارد [۲۵]. این مشکل به مرور زمان اهمیت کمتری پیدا می‌کند، ورودی و خروجی برای هر نمونه زمانی آینده نزدیک در نظر گرفته می‌شوند. یک تابع هزینه برای برطرف کردن فاکتورهای نامطلوب مانند خطای احساس حرکت و تغییرات ناگهانی در متغیرها تعریف می‌شود. این بهینه‌سازی، بهترین مقادیری را که باید در نمونه‌های زمانی بعدی به‌کار بروند تعیین می‌کند. البته، تنها ورودی اولیه به‌کار می‌رود و مسئله برای هر مرحله زمانی دوباره بهینه‌سازی می‌شود. عرض پنجره آینده نزدیک برای بهینه‌سازی تابع هزینه و رعایت محدودیت‌ها افق پیش‌بینی (Tp) نامیده می‌شود و پنجره آینده مربوط به ورودی بهینه بعدی، افق کنترل (Tc) نام دارد. پنجره افق در هر مرحله زمانی عقب می‌رود تا لحظه زمانی قبلی را در آغاز، حذف و لحظه بعدی را در آخر لحاظ کند. کاربرد کنترل پیش‌بین در الگوریتم هم‌گام‌سازی حرکت لزوم اجرای تنظیم وزن را برای سناریوی بدترین مورد از میان برداشته است. کنترل پیش‌بین قابلیت بالاتری برای ایجاد عملکرد بهتر در مقایسه با فیلترهای wash out نشان داده است [۶].

Dagdelen و همکارانش ابتدا الگوریتم هم‌گام‌سازی حرکت مبتنی بر کنترل پیش‌بین را ارائه دادند [۲۶]. در MPC-MCA^۲ اولیه، هماهنگی شتاب به‌کار گرفته نشد؛ بنابراین پلتفورم حرکت به سرعت به محدودیت‌های فضای کار خود رسید. Maran افق‌ها را به‌صورت دستی تنظیم کرد و خاطر نشان ساخت که افق‌های پیش‌بینی و کنترل در کنترل‌کننده پیش‌بین بر نتایج خروجی و زمان محاسبه شبیه‌سازی تأثیر می‌گذارند [۲۷]. البته هیچ روش مستقیمی برای تنظیم بهینه افق کنترل پیشنهاد نشده است و یافتن بهترین مقادیر هنوز هم یک معضل است. فرایند انتخاب افق‌های کنترل و پیش‌بینی بسیار زمان‌بر است. این فرایند تنظیم در MPC-MCA پایین‌تر از سطح بهینه است و به‌طور کامل از قابلیت شبیه‌ساز استفاده نمی‌کند. افق پیش‌بینی باید برای تأمین عملکرد قابل قبول به اندازه کافی گسترده باشد [۲۸] در حالی که افزایش افق پیش‌بینی محاسبه بهینه‌سازی را کند می‌کند. در این مقاله الگوریتم ازدحام ذرات برای دستیابی به بهترین افق‌های کنترل و پیش‌بینی از طریق در نظر گرفتن محدودیت‌های سیستم مانند جابه‌جایی و بار محاسباتی به‌منظور حذف نقاط ضعف خطای تجربی زمان‌بر مربوط به افق‌های کنترل و پیش‌بینی اولیه MPC-MCAهای موجود به‌کار گرفته شد. در این بخش، مفاهیم اساسی الگوریتم هم‌گام‌سازی حرکت و کنترل پیش‌بین مورد بحث قرار گرفته‌اند. در بخش (۲)، توصیف ریاضی کنترل پیش‌بین که برای الگوریتم هم‌گام‌سازی حرکت به‌کار برده شده است به‌طور مبسوط شرح داده شده است و نقش افق‌های کنترل و پیش‌بینی نشان داده شده است. روش پیشنهادی برای بهینه‌سازی افق کنترل پیش‌بین در بخش (۳) شرح داده شده است، نتایج شبیه‌سازی در بخش (۴) ارائه شده و بخش (۵) به نتیجه‌گیری می‌پردازد.

کنترل پیش‌بین

کنترل پیش‌بین به‌صورت موفقیت‌آمیزی در بسیاری صنایع به‌کار رفته است [۲۹؛ ۳۰]. مزیت اصلی این نوع کنترل، توانایی آن در کنترل محدودیت‌های ورودی و خروجی در بهینه‌سازی چند متغیری در یک افق محدود از طریق پیش‌بینی دینامیک فرایند است [۳۱-۳۳]. کنترل‌کننده‌های متعارف با توجه به حالات فعلی و گذشته یک سیستم تصمیمات در لحظه می‌گیرند در حالی که کنترل پیش‌بین، آینده را در نظر می‌گیرد.

¹ Chattering

² Model predictive control-motion cueing algorithm

کنترل پیش‌بین، یک روش بهینه محدود برای استخراج بهترین توالی ورودی به‌منظور دستیابی به خروجی بهینه در یک افق محدود آینده است. ورودی بهینه، بلافاصله برای هر بازه نمونه‌برداری به کار می‌رود و مسئله بهینه‌سازی دوباره در مرحله نمونه بعدی حل می‌شود [۳۴].

برخی مطالعات در مورد کاربردهای الگوریتم ژنتیک و کنترل پیش‌بین برای بهینه‌سازی ورودی کنترل وجود دارند. بیشتر تحقیقات این زمینه از کنترل پیش‌بین مبتنی بر الگوریتم ژنتیک برای کنترل فرایند غیرخطی تحقیقاتی است که AI-Duwaish و Potocnik انجام داده‌اند [۳۵؛ ۳۶]. باید توجه داشت که روش زمان‌بر آزمون و خطا تا به امروز برای تنظیم افق‌های کنترل و پیش‌گویانه در MPC-MCA مورد استفاده قرار گرفته است در حالی که الگوریتم ازدحام ذرات برای بهینه‌سازی افق به کار نرفته است. به‌علاوه، کنترل پیش‌بین، یک بار محاسباتی قابل توجه دارد که می‌تواند تأثیر منفی بر فرایندهای کاربردی در کنترل زمان واقعی داشته باشد؛ بنابراین، هدف اصلی این پژوهش، جستجوی مناسب‌ترین افق‌های کنترل و پیش‌گویانه است که زمان محاسباتی را به حداقل برساند. در این مقاله، تنها از الگوریتم ازدحام ذرات استفاده شده است.

مدل کنترل پیش‌بین در رابطه (۱) را در نظر بگیرید:

$$x_m(t+1) = A_m x_m(t) + B_m u(t) \quad (1)$$

$$y(t) = C_m x_m(t)$$

از آن جا که ممکن است محدودیت‌هایی در نرخ متغیرهای ورودی و نیز خروجی وجود داشته باشد؛ بنابراین ورودی u را به ورودی Δu تغییر می‌دهیم.

$$\Delta u(t) \triangleq u(t) - u(t-1) \quad (2)$$

$$\Delta x_m(t) \triangleq x_m(t) - x_m(t-1) \quad (3)$$

بنابراین داریم:

$$\Delta x_m(t+1) = A_m \Delta x_m(t) + B_m \Delta u(t) \quad (4)$$

$$y(t+1) - y(t) = C_m A_m \Delta x_m(t) + C_m B_m \Delta u(t)$$

اگر $x(t)$ به صورت زیر تعریف شود:

$$x(t) \triangleq \begin{bmatrix} \Delta x_m(t) \\ y(t) \end{bmatrix} \quad (5)$$

از معادله (۴) و (۵) داریم

$$x(t+1) = A x(t) + B \Delta u(t) \quad (6)$$

$$y(t) = C x(t)$$

که در آن $\Delta u(t)$ ورودی کنترل و $x(t)$ متغیر حالت می‌باشند و A و B و C عبارتند از

$$A = \begin{bmatrix} A_m & 0 \\ C_m A_m & I \end{bmatrix}, B = \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix}, C = [0 \quad I] \quad (7)$$

در هر نمونه زمانی t ، حالت پیش‌بینی شده بازه $(t + nT_s)$ به صورت $x(t + n | t)$ نشان داده شده است که T_s زمان نمونه را نشان می‌دهد. بنابراین داریم:

$$x(t + n | t) = A^n x(t) + A^{n-1} B \Delta u(t) + A^{n-2} B \Delta u(t+1) + \dots + B \Delta u(t + n - 1) \quad (8)$$

$$y(t + n_p | t) = CA^n x(t) + CA^{n-1} B \Delta u(t) + CA^{n-2} B \Delta u(t+1) + \dots + CA^{n_p - n_c} B \Delta u(t + n_c - 1) \quad (9)$$

افق‌های کنترل و پیش‌بینی MPC عبارتند از n_p و n_c . بنابراین خروجی پیش‌بینی شده در افق پیش‌بینی تابعی از توالی ورودی و حالت فعلی است. حال می‌توان معادله خروجی را به صورت رابطه زیر نوشت:

$$Y = FX(t) + G \Delta U \quad (10)$$

که در آن $\Delta U, R, F, Y$ ، به صورت زیر تعریف می‌گردد.

$$Y = \begin{bmatrix} y(t+1|t) \\ y(t+1|t) \\ \cdot \\ \cdot \\ \cdot \\ y(t+n_p|t) \end{bmatrix} \quad \Delta u = \begin{bmatrix} \Delta u(t) \\ \Delta u(t+1) \\ \cdot \\ \cdot \\ \cdot \\ \Delta u(t+n_c-1) \end{bmatrix} \quad \text{و} \quad (11)$$

که F و G به صورت زیر تعریف می‌شوند:

$$G = \begin{bmatrix} CB & 0 & \dots & 0 \\ CAB & CB & \dots & 0 \\ CA^2 B & CAB & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{n_p-1} B & CA^{n_p-2} B & \dots & CA^{n_p-n_c} B \end{bmatrix} \quad \text{و} \quad F = \begin{bmatrix} CA \\ CA^2 \\ \cdot \\ \cdot \\ \cdot \\ CA^{n_p} \end{bmatrix} \quad (12)$$

هدف به حداقل رساندن تابع هزینه با در نظر گرفتن قیود است که به صورت تعریف می‌شود:

$$J(\Delta U) = (Y_s - Y)^T Q (Y_s - Y) + U^T S U + \Delta U^T R \Delta U \quad (13)$$

$$D_{\min} \leq V \Delta U \leq D_{\max}$$

که در آن Y_s مرجع خروجی در افق پیش‌بینی و Q ، R ، S ماتریس‌های وزنی با قطر مورب برای جبران افزایش نامطلوب در فاکتورهای متناظر هستند. همچنین توالی ورودی U پس از آخرین ورودی و نوسان ورودی بهینه به صورت زیر محاسبه می‌شود:

$$\begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+n_c-1) \end{bmatrix} = \begin{bmatrix} I & 0 & 0 & \cdots \\ I & I & 0 & \cdots \\ I & I & I & \cdots \\ \vdots & \vdots & \ddots & \cdots \\ I & I & I & I \end{bmatrix} \begin{bmatrix} \Delta u(t) \\ \Delta u(t+1) \\ \vdots \\ \Delta u(t+n_c-1) \end{bmatrix} + \begin{bmatrix} u(t-1) \\ u(t-1) \\ \vdots \\ u(t-1) \end{bmatrix} \quad (14)$$

محدودیت‌ها به سه دسته تقسیم می‌شوند: محدودیت نرخ ورودی، نتایج ورودی و نتایج خروجی که می‌توان آن‌ها را به صورت زیر بیان کرد:

$$D_{\min} \leq V \Delta U \leq D_{\max}$$

که در آن V ، D_{\min} ، D_{\max} به صورت زیر تعریف می‌گردد:

$$V = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix}, D_{\min} = \begin{bmatrix} D_{1\min} \\ D_{2\min} \\ D_{3\min} \end{bmatrix}, D_{\max} = \begin{bmatrix} D_{1\max} \\ D_{2\max} \\ D_{3\max} \end{bmatrix}$$

D_{\min} و D_{\max} محدوده پایین و بالای هر کدام از محدودیت‌ها هستند. محدودیت‌های نرخ‌های ورودی به صورت زیر محاسبه می‌شوند:

$$D_{1\min} = \Delta u_{\min} = \begin{bmatrix} \Delta u_{\min} \\ \Delta u_{\min} \\ \cdot \\ \cdot \\ \cdot \\ \Delta u_{\min} \end{bmatrix}, V_1 = \begin{bmatrix} I & 0 & \cdots & 0 \\ 0 & I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I \end{bmatrix}, D_{1\max} = \Delta u_{\max} = \begin{bmatrix} \Delta u_{\max} \\ \Delta u_{\max} \\ \cdot \\ \cdot \\ \cdot \\ \Delta u_{\max} \end{bmatrix}$$

مجموعه محدودیت‌های مقادیر ورودی را نیز می‌توان به صورت زیر نوشت:

$$D_{2\min} = \begin{bmatrix} u_{\min} - u(t-1) \\ u_{\min} - u(t-1) \\ \vdots \\ u_{\min} - u(t-1) \end{bmatrix}, V_1 = \begin{bmatrix} I & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I \end{bmatrix}, D_{2\max} = \begin{bmatrix} u_{\max} - u(t-1) \\ u_{\max} - u(t-1) \\ \vdots \\ u_{\max} - u(t-1) \end{bmatrix}$$

و مجموعه محدودیت سوم در نتایج خروجی نیز از رابطه زیر به دست می‌آید:

$$D_{3\min} = Y_{\min} - FX(t)$$

$$V_3 = G$$

$$D_{3\max} = Y_{\max} - FX(t)$$

با به کارگیری معادلات (۱۴) و (۱۰) در (۱۳) داریم:

$$J(\Delta U) = Q_s^T (-FX(t) - G\Delta U)^T Q Q_s^T (-FX(t) - G\Delta U) + (\Delta U + \bar{U}_i)^T S (\Delta U + \bar{U}_i) + \Delta U^T R \Delta U \quad (15)$$

پس از حذف عبارات ثابت از تابع هزینه معادله (۱۵) داریم:

$$J(\Delta U) = \Delta U^T (G^T Q G + R + T^T S T) \Delta U + 2\Delta U^T (G^T Q (FX(t) - Y_s) + T^T S \bar{U}_i) \quad (16)$$

معادله (۱۶) را می‌توان به صورت زیر بازنویسی کرد:

$$J(\Delta U) = \frac{1}{2} \Delta U^T H \Delta U + \Delta U^T Z \quad (17)$$

$$D_{\min} \leq V \Delta U \leq D_{\max}$$

که در آن H و K به صورت زیر تعریف می‌شوند:

$$\begin{aligned} H &\triangleq 2(G^T Q G + R + T^T S T) \\ Z &\triangleq 2(G^T Q (FX(t) - Y_s) + T^T S \bar{U}_i) \end{aligned} \quad (18)$$

با هم یک مسئله برنامه‌ریزی درجه دوم را تشکیل می‌دهند؛ بنابراین می‌توان آن را به روش مجموع مربعات محاسبه کرد:

MCA مبتنی بر MPC با بهینه‌سازی افق کنترل

مهم‌ترین فاکتورهای عملکرد MPC افق‌های کنترل و پیش‌بینی هستند؛ زیرا آن‌ها مستقیماً بر اندازه مسئله برنامه‌ریزی درجه دوم تأثیر می‌گذارند. در حضور یک مرجع خروجی قابل اطمینان، افزایش افق پیش‌بینی نتیجه خروجی را بهبود می‌بخشد؛ البته، بار محاسباتی، یک محدودیت است. افزایش افق کنترل به معنای درجه آزادی بالاتر ورودی کنترل است؛ به هر صورت، ممکن است این کار ضروری نباشد و به محاسبات بیشتر و در برخی موارد حتی به اثرات نامطلوب منجر شود. در روش پیشنهادی، بهینه‌سازی آفلاین برای رسیدن به بهترین مقادیر برای افق‌های کنترل و پیش‌بینی از طریق به حداقل رساندن یک تابع هزینه انجام می‌شود.

$$J_{total} = K_1 J_{Equation} + K_2 J_{MCA} \quad (19)$$

رجه داشت که K_1 و K_2 فاکتور وزن برای کنترل تأثیر هزینه محاسبات است. رشد n_p و n_c در یک تابع هزینه برای محدود کردن افق‌هایی که به صورت غیر ضروری عریض هستند قرار داده می‌شود هزینه محاسباتی $J_{Equation}$ باید تابعی منطقی از n_p و n_c باشد. این هزینه به افق‌ها اجازه می‌دهد تا تنها زمانی که افزایش آن‌ها ضروری است رشد کند و فاکتورهای نامطلوب مانند خطا و جابه‌جایی را کاهش می‌دهد. البته این تابع، افزایش بیشتر در افق‌ها را کم می‌کند. در این مقاله تابع $J_{Equation}$ به صورت زیر است [۱۱]:

$$J_{Equation} = a_1 n_p + a_2 n_c + a_3 n_p n_c + a_4 n_p^2 n_c + a_5 n_c^2 n_p + a_6 n_p^2 + a_7 n_c^2 \quad (20)$$

برای انتخاب ضرایب معادله ما کنترل پیش‌بین را با چند تغییر در n_p و n_c اجرا می‌کنیم و ضرایب را برای هماهنگ کردن با زمان شبیه‌سازی برازش می‌کنیم. در مدل فضای حالت MCA داریم [۳۷]:

$$\hat{x}_k = \begin{bmatrix} \theta_k \\ \omega_k \\ x_k \\ v_k \\ \dot{\omega}_k \\ \dot{v}_k \end{bmatrix}, A = \begin{bmatrix} 1 & dt & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & dt & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.5dt^2 & 0 \\ dt & 0 \\ 0 & 0.5dt^2 \\ 0 & dt \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, u_k = \begin{bmatrix} \dot{\omega}_k \\ \dot{v}_k \end{bmatrix}$$

هزینه J_{MCA} متشکل از هزینه‌های خطای حس‌های چرخشی و انتقالی، جابه‌جایی سکو، سرعت سکو، ورودی شتاب به کاررفته و ورودی سرعت زاویه‌ای است [۱۲].

$$J_{MCA} = a_{11} \int (f - f_{ref})^2 dt + a_{12} \int (\omega - \omega_{ref})^2 dt + a_{13} \int (\varpi - \varpi_{ref})^2 dt + a_{14} \int \omega^2 dt + a_{15} \int \varpi^2 dt + a_{16} \int x^2 dt + a_{17} \int v^2 dt + a_{18} \int a^2 dt + a_{19} \int \theta^2 dt \quad (21)$$

که در آن f نیرو، f_{ref} نیروی حس شده توسط استئولیت‌ها، ϖ سرعت زاویه‌ای ناشی از شتاب، ω_{ref} چرخش حس شده توسط کانال‌های نیم‌دایره از ω ، ϖ سرعت زاویه‌ای برای تولید احساس چرخش و ω_{ref} چرخش حس شده توسط کانال‌های نیم‌دایره از ω است. x جابه‌جایی سکو، v سرعت سکو، a شتاب به کار رفته و θ جابه‌جایی زاویه‌ای سکو است و f نیرو، f_{ref} نیروی مخصوص حس شده توسط استئولیت‌ها می‌باشد. f نیروی وارد شده بر سر راننده از سوی شتاب خطی a و گرانش است که به صورت زیر محاسبه می‌گردد:

$$f_x = a_x + g\theta \quad (22)$$

که g شتاب گرانش است.

برای محاسبه حس انسانی، سیستم دهلیزی پیشنهادی مرجع [۹] که مناسب‌ترین مدل است و در بین محققان به‌صورت گسترده استفاده شده است انتخاب می‌گردد.

$$\frac{f_{ref}}{f} = 0.4 \frac{1+10s}{(1+5s)(1+0.016s)} \quad (22)$$

و سرعت زاویه‌ای حس شده توسط کانال‌های نیم‌دایره به‌صورت زیر است

$$\frac{\omega_{ref}}{\omega} = \frac{5.73 * 80 * s^2}{(1+5.73s)(1+80s)} \quad (23)$$

اما همان‌طور که می‌دانیم در کنترل پیش‌بین، در مورد یک جفت معتبر (n_c, n_p) ، شرط زیر باید لحاظ شود.

$$n_c \leq n_p \quad (24)$$

الگوریتم ازدحام ذرات یکی از روش‌های بهینه‌سازی است. این الگوریتم را کندی و ابرهات در سال ۱۹۹۵ توسعه دادند. الگوریتم، الهام گرفته از رفتار اجتماعی گونه‌های مختلف زیستی در زندگی گروهی آنان است؛ مانند پرندگان و ماهی‌ها. الگوریتم، ازدحام ذرات تعامل بین اعضا را برای به اشتراک‌گذاری اطلاعات تقلید می‌کند. همچنین می‌توان گفت الگوریتم ازدحام ذرات از سرعت بالایی برخوردار است [۳۸].

در مرحله ابتدایی الگوریتم، ذرات با موقعیت‌ها و سرعت‌های تصادفی ایجاد می‌شوند. طی اجرای الگوریتم، موقعیت و سرعت هر ذره در مرحله $t+1$ ام از الگوریتم، از روی اطلاعات مرحله قبلی ساخته می‌شوند. اگر Z_j مؤلفه Z ام از بردار Z باشد، آن‌گاه روابطی که سرعت و موقعیت ذرات را تغییر می‌دهند، عبارتند از:

$$\begin{aligned} v_j^i[t+1] &= wv_j^i[t] + c_1r_1(x_j^{i,best}[t] - x_j^i[t]) + c_2r_2(x_j^{g,best}[t] - x_j^i[t]) \\ x_j^i[t+1] &= x_j^i[t] + v_j^i[t+1] \end{aligned} \quad (25)$$

در این روابط، W ضریب اینرسی، r_1 و r_2 اعدادی تصادفی در بازه $[0,1]$ با توزیع یکنواخت و همچنین c_1 و c_2 ضرایب یادگیری هستند. r_1 و r_2 باعث می‌شوند که نوعی گوناگونی در جواب‌ها به وجود بیاید و به این نحو جستجوی کاملی روی فضا انجام پذیرد. c_1 ضریب یادگیری مربوط به تجارب شخصی هر ذره است و در مقابل c_2 ضریب یادگیری مربوط به کل جمع می‌باشد.

برای به‌کار گرفتن الگوریتم ازدحام ذرات در بهینه‌سازی افق‌ها، هر ذره متناظر با افق‌های کنترل و پیش‌بینی هستند:

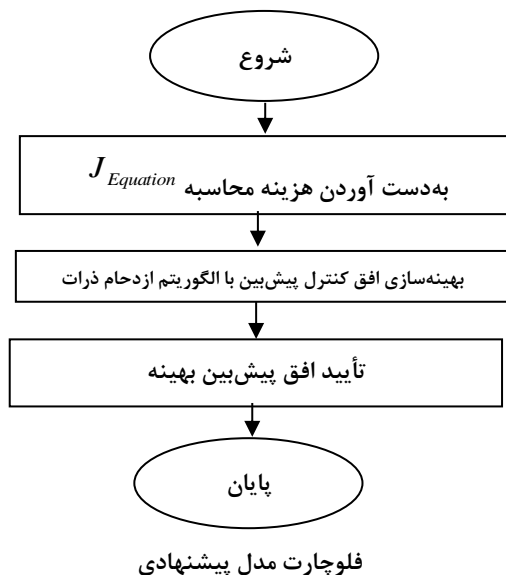
$$P_i = [n_c \quad n_p]$$

در نتیجه هر ذره، شرط خطی (۲۵) را باید در نظر بگیرد. ضمن اینکه برای اجتناب از فرایند بهینه‌سازی طولانی هر ذره پس از ۵ بار که شرط (۲۵) را برآورده نکرد حذف می‌گردد؛ زیرا آن‌ها مقدار فرایند قابل‌توجهی را به خود اختصاص می‌دهند در حالی که نتایج آن‌ها مطلوب نیست. علاوه بر آن، هر ذره باید شرط زیر را نیز برآورده کند:

$$\langle n_{p2} \leq n_{p1} \quad , \quad n_{c2} \leq n_{c1} \rangle \quad (26)$$

زیرا افزایش هر یک از افق‌ها به‌طور یکنواخت محاسبه را افزایش می‌دهد. در رابطه (۲۶)، اعداد ۱ و ۲ به معنی مرحله اول و مرحله دوم می‌باشد.

تابع $J_{Equation}$ به‌عنوان نماینده بار محاسباتی در ارزیابی تابع هزینه در الگوریتم ازدحام ذرات برای بهینه‌سازی افق‌ها به‌کار خواهد رفت. در مرحله بعدی، الگوریتم ازدحام ذرات با فرایند جستجو برای راه‌حل ایده‌آل آغاز می‌شود. در ادامه فلوجارت کلی روش پیشنهادی به تصویر کشیده شده است.

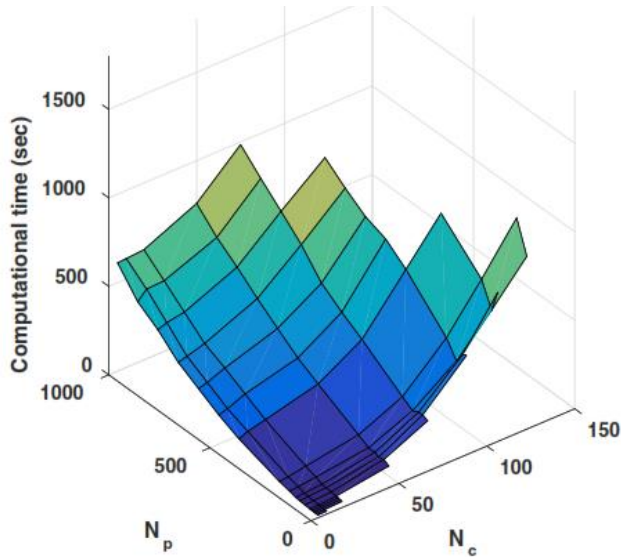


نتایج شبیه‌سازی

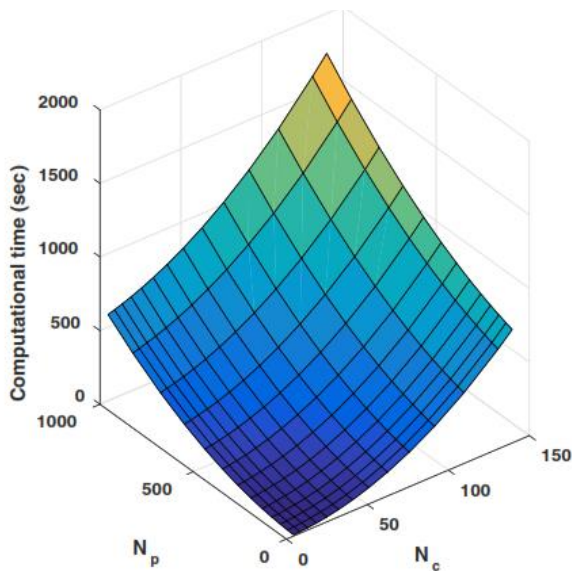
فرایند به‌دست آوردن تابع محاسبات و تابع هزینه و نیز الگوریتم ازدحام ذرات و شبیه‌سازی‌ها با استفاده از نرم‌افزار متلب انجام شده‌اند. هزینه محاسبات حاصل از به‌کارگیری برازش تابع به‌صورت زیر به‌دست می‌آید:

$$J_{Equation} = 0.4n_p + 3 \times 10^{-2}n_c + 17 \times 10^{-3}n_p n_c + 19 \times 10^{-5}n_p^2 n_c + 5 \times 10^{-3}n_p^2 \quad (27)$$

الگوریتم ازدحام ذرات برای بهینه‌سازی افق‌های MPC برای یک سیگنال ورودی آموزش کاربردی مورد استفاده قرار می‌گیرد. بیشینه بار محاسباتی $(J_{Equation, max})$ ۸۰۰ تنظیم شده است.



شکل ۱. تابع محاسباتی حاصل از نمونه‌های شبیه‌سازی از طریق تغییر n_p و n_c



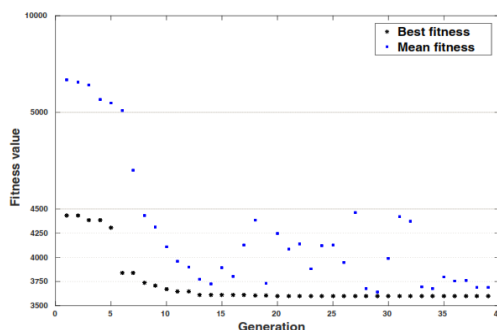
شکل ۲. تابع محاسباتی برآورد شده از طریق تغییر n_p و n_c

جدول ۱ تنظیمات الگوریتم ازدحام ذرات را نشان می‌دهد. تعداد جمعیت در ۱۰۰ تنظیم شده است. باید توجه داشت که افزایش بیشتر جمعیت صرفاً زمان بهینه‌سازی را بدون هیچ دستاورد چشمگیری که به هم‌گرایی منجر شود، افزایش می‌دهد. علاوه بر این، نتایج رضایت‌بخش نشان می‌دهند که پارامترهای منتخب به‌درستی انتخاب شده‌اند.

جدول ۱. مقادیر پارامترهای الگوریتم ازدحام ذرات

| مقدار پارامتر | نام پارامتر |
|---------------|------------------|
| ۱۰۰ | اندازه جمعیت |
| ۰.۱-۰.۹ | ضریب اینرسی |
| ۵ | ضرایب یادگیری |
| مدت زمان | معیار توقف جستجو |
| ۳۰ ثانیه | زمان محاسباتی |

برای صرفه‌جویی در زمان، هر ذره ارزیابی شده جدید همراه با ارزش اعتبار خود و نتیجه هزینه ذخیره می‌شود؛ بنابراین، در مرحله بعدی برای همان ارزیابی، افق مقدار ذخیره شده بدون اجرای شبیه‌سازی بازگردانده می‌شود. در بهینه‌سازی انجام شده، الگوریتم ازدحام ذرات در تکرار شصت‌وهفتم متوقف شد. شکل ۳ همگرایی الگوریتم ازدحام ذرات را نشان می‌دهد. راه‌حل بهینه به‌دست‌آمده از طریق الگوریتم ازدحام ذرات عبارت است از: $NP = 50$ ، $Nc = 5$



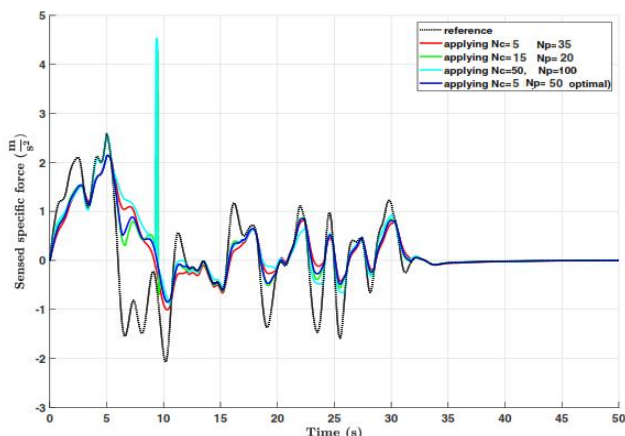
شکل ۳. همگرایی الگوریتم ازدحام ذرات برای بهینه‌سازی افق

سیگنال‌های آموزشی از سناریوهای آموزشی با استفاده از نرم‌افزار شبیه‌ساز دریایی ROV-SIM استخراج گردید. یک سیگنال ورودی آزمایشی برای مقایسه افق‌های بهینه حاصل از دانش ما و آزمون و خطا براساس سناریوهای شبیه‌سازی متعدد و نیز محدودیت‌های فیزیکی شبیه‌ساز مورد استفاده در مقاله در محیط شبیه‌سازی به‌کار رفت که هدف آن دستیابی به بهترین نتایج شبیه‌سازی بر مبنای جابه‌جایی، خطای حس انسانی و محدودیت‌های فیزیکی مربوطه است. به‌منظور مقایسه مقادیر به‌دست‌آمده با افق‌های پیش‌بینی و کنترل MPC-MCA اجرایی به‌صورت دستی که از طریق یک فرایند خسته‌کننده و زمان‌بر تنظیم می‌شوند از سه نفر از متخصصان و کارشناسان طراحی مقادیر مدنظر را دریافت کردیم که در جدول ۲ به آن اشاره شده است.

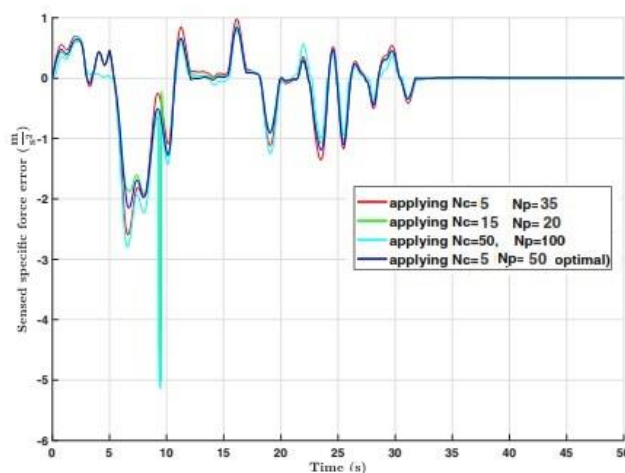
جدول ۲. مقادیر پیشنهادی افق کنترل و افق پیش‌بین توسط متخصصان

| کارشناس | مقادیر پیشنهادی (π_c, π_p) |
|-----------|----------------------------------|
| کارشناس ۱ | (۵, ۳۵) |
| کارشناس ۲ | (۱۵, ۲۰) |
| کارشناس ۳ | (۵۰, ۱۰۰) |

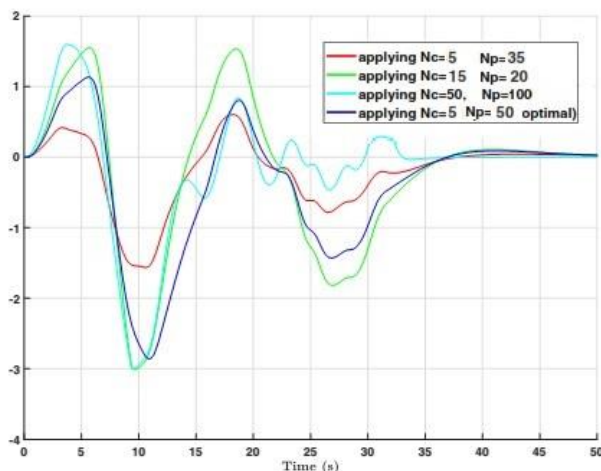
شکل ۴ مقایسه‌ای بین نیروی احساس شده برای افق‌های بهینه و مواردی که بیان شدند را نشان می‌دهد. پیروی از مرجع توسط نیروی احساس شده مطلوب است. مرجع نیروی حس شده به‌عنوان نیروی خاصی که توسط کاربری که حرکت مشابهی را در یک شبیه‌ساز دریایی واقعی احساس می‌کند تعریف می‌شود. اختلاف بین مرجع و نیروی حس شده خروجی، خطای حس‌شده است که در شکل ۵ نشان داده شده است. هدف به‌حداقل رساندن خطای حس‌شده و جابه‌جایی از طریق تابع هزینه است. شکل ۶ جابه‌جایی برای تنظیمات افق MPC مختلف را نشان می‌دهد.



شکل ۴. مقایسه نیروی احساس شده در افق‌های مختلف



شکل ۵. مقایسه خطای نیروی مخصوص احساس شده برای افق‌های مختلف



شکل ۶. مقایسه جابه‌جایی سکو برای افق‌های مختلف

افق‌های پیش‌بینی کوتاه در معرض نقض شدید محدودیت‌های کنترل پیش‌بین هستند؛ زیرا کنترل پیش‌بین پنجره گسترده آینده را لحاظ نمی‌کند. زمانی که نزدیک است سکوی حرکت به مرزها برسد، خروجی کنترل پیش‌بین توسط یک شتاب ترمز برای متوقف کردن سکوی حرکت جایگزین خواهد شد. این مسئله به یک حرکت ناگهانی شدید برای کاربر منجر می‌شود. از سوی دیگر، افزایش افق‌ها محاسبات را کند می‌کند. نتایج شبیه‌سازی روش پیشنهادی نشان می‌دهند که ترجیح بر این است که افق کنترل در MPC-MCA پایین باشد. هزینه افق‌های بهینه به‌دست‌آمده از روش پیشنهادی همان‌طور که در جدول ۳ نشان داده شده است در پایین‌ترین مقدار است. تعیین بهترین افق براساس جریمه کل است. راه‌حل به‌دست‌آمده در مقایسه با سایر افق‌ها برحسب عملکرد خروجی (احساس و جابه‌جایی) و نیز بار محاسباتی مؤثرتر است.

جدول ۳. مقایسه تنظیم پارامترها

| افق‌ها | (۱۵,۲۰) | (۵,۳۵) | (۵۰,۱۰۰) | (۵,۵۰) |
|----------|---------|--------|----------|--------|
| هزینه کل | ۶۰۹.۱ | ۴۶۰.۱ | ۶۲۰.۹ | ۴۳۶.۸ |

نتیجه‌گیری

کنترل پیش‌بین نقش مهمی در الگوریتم هم‌گام‌سازی حرکت یافته است. تاکنون انتخاب افق‌های پیش‌بینی و کنترل بر مبنای تنظیم دستی طراح بوده است و در نتیجه آن‌ها پایین‌تر از سطح بهینه بوده‌اند. در این تحقیق، روشی برای به‌دست آوردن افق‌های بهینه با استفاده از یک الگوریتم ازدحام ذرات پیشنهاد شد. این روش برای حل مسائل مربوط به تنظیم افق MPC-MCA از طریق آزمون و خطا، مانند خروجی نامطلوب یا بار محاسباتی زیاد طراحی شده است. خطای حسی و جابه‌جایی و نیز بار محاسباتی برای به‌دست آوردن بهترین افق‌ها لحاظ شده‌اند. نتیجه هر ارزیابی برای آینده جهت سرعت بخشیدن به ازدحام ذرات ذخیره شده است. افق‌های کنترل و پیش‌بینی بهینه شده‌اند و نتایج شبیه‌ساز برتری راه‌حل حاصل را نسبت به گزینش دستی طراح از افق‌های کنترل پیش‌بین بر مبنای آزمون و خطا

نشان داده‌اند. اگرچه ایده بهینه‌سازی افق کنترل پیش‌بین در MCA اجرا شد، اجرای آن می‌تواند به سایر مصارف نیز بسط داده شود.

References

- [1] Asadi, H., Mohammadi, A., Mohamed, S., & Nahavandi, S. (2014). Adaptive translational cueing motion algorithm using fuzzy based tilt coordination. In *International Conference on Neural Information Processing* (pp. 474-482). Springer, Cham. https://doi.org/10.1007/978-3-319-12643-2_58
- [2] Asadi, H., Mohammadi, A., Mohamed, S., Zadeh, D. R., & Nahavandi, S. (2014). Adaptive washout algorithm based fuzzy tuning for improving human perception. In *International Conference on Neural Information Processing* (pp. 483-492). Springer, Cham. https://doi.org/10.1007/978-3-319-12643-2_59
- [3] Woon-Sung, L., Jung-Ha, K., & Jun-Hee, C. (1998, May 20-20). *A driving simulator as a virtual reality tool*. Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146), Leuven, Belgium <https://ieeexplore.ieee.org/abstract/document/676264>
- [4] Chang, Y.-H., Liao, C.-S., & Chieng, W.-H. (2009). Optimal motion cueing for 5-DOF motion simulations via a 3-DOF motion simulator. *Control Engineering Practice*, 17(1), 170-184. <https://doi.org/10.1016/j.conengprac.2008.05.016>
- [5] Aminzadeh, M., Mahmoodi, A., & Sabzehparvar, M. (2012). Optimal Motion-Cueing Algorithm Using Motion System Kinematics Final Comment. *European Journal of Control*, 18, 377. <https://doi.org/10.3166/ejc.18.363-375>
- [6] Fang, Z., & Kemeny, A. (2014, Sep 4-5). *Review and prospects of Renault's MPC based motion cueing algorithm for driving simulator*. Proceedings of the Driving Simulation Conference Europe Paris, France. http://dsc2015.tuebingen.mpg.de/Docs/DSC_Proceedings/2014/DSC14_18_Fang_2.pdf
- [7] Fang, Z., Colombet, F., Collinet, J.-C., Kemeny, A., & Center, I. S. (2014, September 4-5). *Roll tilt thresholds for 8 DOF driving simulators*. In Proc. Driving Simul. Conf, Europe. http://dsc2015.tuebingen.mpg.de/Docs/DSC_Proceedings/2014/DSC14_10_Fang_1.pdf
- [8] Groen, E. L., & Bles, W. (2004). How to use body tilt for the simulation of linear self motion. *J Vestib Res*, 14(5), 375-385. <https://doi.org/10.3233/VES-2004-14503>
- [9] Telban ,R. J., & Cardullo, F. M. (2005). *Motion cueing algorithm development: Human-centered linear and nonlinear approaches*. Nasa. <https://ntrs.nasa.gov/api/citations/20050180246/downloads/20050180246.pdf>
- [10] Chapron, T., & Colinot, J.-P. (2007, September 12-14) .*The new psa peugeot-citroen advanced driving simulator overall design and motion cue algorithm*. Proceedings of Driving Simulation Conference, North America – Iowa City. <https://car-d.fr/publications/Chapron.pdf>
- [11] Asadi, H., Mohamed, S., Lim, C. P & ,Nahavandi, S. (2016). A review on otolith models in human perception. *Behavioural brain research*, 309, 67-76. <https://doi.org/10.1016/j.bbr.2016.03.043>
- [12] Asadi, H., Mohamed, S., Lim, C. P., Nahavandi, S., & Nalivaiko, E. (2017). Semicircular canal modeling in human perception. *Rev Neurosci*, 28(5), 537-549. <https://doi.org/10.1515/revneuro-2016-0058>

- [13] Reid, L. D., Nahon, M. A., & Studies, U. o. T. I. f. A. (1986). *Flight Simulation Motion-base Drive Algorithms. Part 3, Pilot Evaluations*. Institute for Aerospace Studies, University of Toronto. <https://books.google.com/books?id=OM5hewAACAAJ>
- [14] Reid, L. D., & Nahon, M. A. (1986). *Flight Simulation Motion-Base Drive Algorithms.: Part 2, Selecting The System Parameters* (UTIAS report, Issue. U. o. Toronto. <http://resolver.tudelft.nl/uuid:4faf3129-88c9-4117-82e9-f9819601dafd>
- [15] Nahon, M. A., & Reid, L. D. (1990). Simulator motion-drive algorithms-A designer's perspective. *Journal of Guidance, Control, and Dynamics*, 13(2), 356-362. <https://doi.org/10.2514/3.20557>
- [16] Conrad, B., & Schmidt, S. F. (1971). *A study of techniques for calculating motion drive signals for flight simulators*. Nasa. <https://ntrs.nasa.gov/api/citations/19710025909/downloads/19710025909.pdf>
- [17] Thöndel, E. (2012, January 1) .*Design and optimisation of a motion cueing algorithm for a truck simulator*. 26th annual European simulation and modelling conference, Höskolan i Skövde. https://www.researchgate.net/profile/Evzen-Thoendel/publication/290299955_Design_and_optimisation_of_a_motion_cueing_algorithm_for_a_truck_simulator/links/60be166a92851cb13d861bcd/Design-and-optimisation-of-a-motion-cueing-algorithm-for-a-truck-simulator.pdf
- [18] Casas, S., Coma, I., Portalés, C., & Fernández, M. (2016). Towards a simulation-based tuning of motion cueing algorithms. *Simulation Modelling Practice and Theory*, 67, 137-154. <https://doi.org/10.1016/j.simpat.2016.06.002>
- [19] Arioui, H., Nehaoua, L., & Amouri, H. (2005, June 27-29). *Classic and Adaptive Washout Comparison for a Low Cost Driving Simulator*. Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005., Limassol, Cyprus. <https://ieeexplore.ieee.org/abstract/document/1467080>
- [20] Nehaoua, L., Mohellebi, H., Amouri, A., Arioui, H., Espie, S., & Kheddar, A. (2008). Design and Control of a Small-Clearance Driving Simulator. *IEEE Transactions on Vehicular Technology*, 57(2), 736-746. <https://doi.org/10.1109/TVT.2007.905336>
- [21] Nehaoua, L., Arioui, H., Mohellebi, H., & Espie, S. (2006, June 14-16). *Restitution movement for a low cost driving simulator*. 2006 American Control Conference, <https://ieeexplore.ieee.org/abstract/document/1656614>
- [22] Dagdelen, M., Reymond, G., Kemeny, A., Bordier, M., & Maïzi, N. (2009) .(Model-based predictive motion cueing strategy for vehicle driving simulators. *Control Engineering Practice*, 17(9), 995-1003. <https://doi.org/10.1016/j.conengprac.2009.03.002>
- [23] Walker, J. (2015). A motion cueing model for mining and forestry simulator platforms based on Model Predictive Control. In.
- [24] Baseggio, M., Beghi, A., Bruschetta, M., Maran, F., & Minen, D. (2011, Oct 5-7). *An MPC approach to the design of motion cueing algorithms for driving simulators*. 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), Washington, DC, USA <https://ieeexplore.ieee.org/abstract/document/6083053>
- [25] Masoumnezhad, M. (2017). Robust control for the indefinite model of the semi-active suspension system. *Karafan Quarterly Scientific Journal*, 14(42), 57-79 .
- [26] Dagdelen, M., Reymond, G., Kemeny, A., Bordier, M., & Maïki, N. (2004). *MPC based motion cueing algorithm: Development and application to the Ultimate driving simulator*. DSC 2004 Europe (driving simulation conference),

- [27] Maran, F. (2013). *Model-based control techniques for automotive applications* [University of the Studi of Padova]. Padua, Italy. http://automatica.dei.unipd.it/tl_files/publicazioni/PhDThesis/MaranFabio_PhD_thesis.pdf
- [28] Garrett, N. J., & Best, M. (2010, August 22-26). *Driving simulator motion cueing algorithms—a survey of the state of the art*. Proceedings of the 10th International Symposium on Advanced Vehicle Control (AVEC), Loughboroug, England. <https://repository.lboro.ac.uk/account/articles/9223-61>
- [29] Hosen, M. A., Hussain, M., Mjalli, F., Khosravi, A., Creighton, D., & Nahavandi, S. (2014). Performance analysis of three advanced controllers for polymerization batch reactor: An experimental investigation. *Chemical Engineering Research and Design*, 92, 903–916. <https://doi.org/10.1016/j.cherd.2013.07.032>
- [30] Rodriguez, J., Kazmierkowski, M. P., Espinoza, J. R., Zanchetta, P., Abu-Rub, H., Young, H. A., & Rojas, C. A. (2013). State of the Art of Finite Control Set Model Predictive Control in Power Electronics. *IEEE Transactions on Industrial Informatics*, 9(2), 1003-1016. <https://doi.org/10.1109/TII.2012.2221469>
- [31] Lee, J. H. (2011). Model predictive control: Review of the three decades of development. *International Journal of Control, Automation and Systems*, 9(3), 415-424. <https://doi.org/10.1007/s12555-011-0300-6>
- [32] Ruchika, N. R. (2013). Model predictive control: History and development. *International Journal of Engineering Trends and Technology (IJETT)*, 4(6), 2600-2602 .
- [33] Stewart, B. T., Venkat, A. N., Rawlings, J. B., Wright, S. J., & Pannocchia, G. (2010). Cooperative distributed model predictive control. *Systems & Control Letters*, 59(8), 460-469. <https://doi.org/10.1016/j.sysconle.2010.06.005>
- [34] Bemporad, A., Borrelli, F & Morari, M. (2002). Model predictive control based on linear programming - the explicit solution. *IEEE Transactions on Automatic Control*, 47(12), 1974-1985. <https://doi.org/10.1109/TAC.2002.805688>
- [35] Potočník, P., & Grabec, I. (2002). Nonlinear model predictive control of a cutting process. *Neurocomputing*, 43(1), 107-126. [https://doi.org/10.1016/S0925-2312\(01\)00623-3](https://doi.org/10.1016/S0925-2312(01)00623-3)
- [36] Al-Duwaish, H., & Naeem, W. (2001, Sept 7-7). *Nonlinear model predictive control of Hammerstein and Wiener models using genetic algorithms*. Proceedings of the 2001 IEEE International Conference on Control Applications (CCA'01) (Cat. No.01CH37204), Mexico City, Mexico <https://ieeexplore.ieee.org/abstract/document/973909>
- [37] Fang, Z., & Kemeny, A. (2016). An efficient Model Predictive Control-based motion cueing algorithm for the driving simulator. *Simulation*, 92(11), 1025-1033. <https://doi.org/10.1177/0037549716667835>
- [38] Sabri, M. (2017). Stabilization and control of the power system using meta-heuristic algorithms. *Karafan Quarterly Scientific Journal*, 14(42), 33-55.