

## کنترل پیش‌بین ربات دوپا در بالا و پایین رفتن از پله بر مبنای مدل پیش‌بین عصبی-تطبیقی

رضا حیدری<sup>۱</sup>، محمد فرخی<sup>۲</sup>

۱ کارشناسی ارشد برق کنترل، دانشگاه علم و صنعت ایران

۲ دانشیار دانشکده برق، دانشگاه علم و صنعت ایران farrokhi@iust.ac.ir

تاریخ دریافت: ۹۲/۱۰/۳۰ تاریخ پذیرش: ۹۴/۱/۳۰

### چکیده:

در این مقاله، از کنترل پیش‌بین غیرخطی برای کنترل ربات دوپا با پنج درجه آزادی در بالا و پایین رفتن از پله‌ها استفاده شده است. در این روش، طراحی الگوی حرکت به‌طور بی‌درنگ در کنترل‌کننده پیش‌بین انجام شده است. بدین صورت که هیچ مسیر از قبل تعیین شده‌ای برای حرکت ربات تعیین و مشخص نشده و حرکت ربات تنها در قالب قیود بیان شده است. از ویژگی‌های اصلی این روش می‌توان به طراحی مسیر به‌صورت وصل خط اشاره کرد که این کار می‌تواند مقاوم بودن کنترل‌کننده در مواجهه با عدم قطعیت در محیط را بهبود دهد. همچنین، کنترل پیش‌بین به‌نحوی طراحی شده که ربات قادر باشد از پله‌هایی با طول و ارتفاع متفاوت نیز بالا و پایین برود و طول گام را نیز تعیین کند. اما مشکل جدی که در استفاده از کنترل پیش‌بین مطرح می‌شود، مدل‌گرا بودن این کنترل‌کننده است. به همین دلیل از شبکه عصبی RBF برای شناسایی وصل خط مدل ربات استفاده شده است. در این صورت، کنترل پیش‌بین، به یک کنترل‌کننده تطبیقی تبدیل شده و خود را با تغییر پارامترهای سیستم تطبیق می‌دهد. نتایج شبیه‌سازی‌ها نشان از عملکرد مناسب روش پیشنهادی در مقایسه با روش‌های موجود در مقالات اخیر دارد.

### کلیدواژه:

ربات‌های دوپا، کنترل پیش‌بین غیرخطی، مدل پیش‌بین، شبکه‌های عصبی مصنوعی

### مقدمه

کند و هم حرکت‌های سریع ربات استفاده کرد. ZMP نقطه‌ای در کف پای ربات دوپا بر روی زمین است که در آن برآیند نیروهای عکس‌العمل و گشتاورها صفر است [۱]. برطبق این معیار، اگر مکان این نقطه درون ناحیه تکیه‌گاهی قرار داشته باشد، حرکت متعادل خواهد بود و با خارج شدن آن از داخل ناحیه پایداری، ربات دچار ناپایداری و سقوط می‌گردد. ناحیه تکیه‌گاهی برحسب نوع حرکت می‌تواند تغییر کند. برای مثال در بررسی راه رفتن یک ربات دوپا، می‌توان مساله را به فازهای یک‌تکیه‌گاهی و دو تکیه‌گاهی تقسیم کرد. در فاز یک‌تکیه‌گاهی که در آن فقط کف پا بر روی زمین قرار دارد، ناحیه تکیه‌گاهی منطبق بر سطح کف پا می‌باشد، در حالی که در فاز دو تکیه‌گاهی، که در آن هر دو پا بر روی زمین قرار دارد، ناحیه تکیه‌گاهی به میزان قابل توجهی گسترش یافته و تمام سطح بین دو پا را دربرمی‌گیرد. یکی از مسایل چالش برانگیز در زمینه ربات‌های دوپا، بالا و پایین رفتن این ربات‌ها از پله‌ها است. در این زمینه مطالعات زیادی انجام شده است اما تقریباً در تمام کارهای انجام شده نیاز است تا قبل از حرکت ربات ابتدا با توجه به شرایط محیطی و مشخصات پله‌ها، طراحی مسیر انجام گیرد. اما می‌توان ادعا کرد که این روش با مساله حرکت انسان به خوبی تطابق ندارد. روش‌های کنترلی مختلفی برای کنترل

کنترل ربات‌های دوپا، حوزه بسیار فعالی در دو دهه اخیر بوده است. کنترل این نوع از ربات‌ها بسیار مشکل‌تر از ربات‌های چرخ‌دار است زیرا ربات‌های دوپا سیستم‌هایی ذاتاً ناپایدار بوده و علاوه بر کنترل حرکت، تعادل نیز در آن‌ها مطرح می‌شود. در مقابل، ربات‌های دوپا دارای مزایایی نسبت به ربات‌های چرخ‌دار هستند که باعث جلب توجه محققان می‌شود. از آن جمله، حرکت بهتر در سطوح گسسته مانند پله‌ها، سطوح ناهموار، عبور از موانع است. این ربات‌ها در بخش عمده‌ای از سیکل حرکتی خود تنها بر روی یک پا تکیه دارند. از این رو، هر لحظه در معرض ناپایداری و سقوط می‌باشند. بنابراین، لازم است الگوی حرکتی مناسبی با توجه به معیارهای پایداری، برای این ربات‌ها ایجاد شود. به منظور بررسی تعادل ربات‌های دوپا معیارهای مختلفی ارائه شده است که از آن جمله می‌توان به معیار مرکز جرم ( $COM^1$ )، معیار مرکز فشار ( $COP^2$ ) و معیار نقطه ممان صفر ( $ZMP^3$ ) اشاره کرد. از این بین، معمولاً از ZMP استفاده بیش‌تری می‌شود زیرا می‌توان از آن هم برای حرکت‌های

1 Center of Mass  
2 Center of Pressure  
3 Zero Moment Point

VHIMP<sup>5</sup> استفاده کرده‌اند. در این روش، حرکت مرکز جرم ربات از طریق آونگ وارون با طول متغیر مدل می‌شود به طوری که با تنظیم مناسب طول آونگ می‌توان خطای ردیابی ZMP را کاهش داد. ابتدا، مسیر مطلوب ZMP برای بیشترین پایداری در طول حرکت در وسط پای تکیه‌گاه طراحی می‌شود. سپس، با فرارگیری ZMP در وسط پای تکیه‌گاه معادله حرکت افقی مرکز جرم ربات در صفحه جانبی با استفاده از روش VHIMP به صورت خاصی به دست می‌آید که با داشتن شرایط اولیه و حل این معادله دیفرانسیل مرتبه دوم، مسیر مرکز جرم ربات در جهت افقی به دست می‌آید. مسیر مرکز جرم در جهت عمودی نیز با استفاده از یک چندجمله‌ای درجه ششم ایجاد می‌شود که ضرایب آن را می‌توان با استفاده از شرایط اولیه، شرایط نهایی و مسیر حرکتی ربات تعیین کرد. در نهایت نیز از روش کنترلی گشتاور محاسباتی برای ردیابی مسیرهای مرجع استفاده شده است. یکی از معایب اصلی این روش وابستگی به مدل ربات است به طوری که در صورت وجود عدم قطعیت یا اغتشاش، کنترل کننده قادر به دنبال کردن مسیری با دقت مطلوب نیست. در [۸] نیز مسیر حرکتی پای متحرک و لگن ربات با استفاده از چندجمله‌ای درجه سوم ایجاد شده است. ضرایب این چندجمله‌ای را می‌توان با در نظر گرفتن شرایط اولیه، ارتفاع و طول پله محاسبه کرد. بعد از طراحی مسیر از کنترل کننده PD در مفصل ران و زانو برای ردیابی مسیر مرجع طراحی شده استفاده شده است. هم‌چنین برای پایداری ربات در عمل، از یک جبران‌ساز ZMP بهره‌گرفته شده است که در صورت خارج شدن ZMP از داخل ناحیه پایداری در لحظه برخورد پای متحرک با زمین، با اعمال گشتاور مناسب به مچ پای تکیه‌گاه این مشکل را برطرف می‌کند. یکی از مشکلات این روش کنترلی نیاز به محاسبه سینماتیک وارون برای به دست آوردن مسیر مرجع در فضای مفاصل است. هم‌چنین بهره‌های کنترل کننده PD نیز به روش سعی و خطا تعیین می‌شود که کارمشکلی است. در روش‌هایی که تاکنون مطرح شد، ابتدا طراحی الگوی حرکت صورت گرفته و سپس کنترل کننده مسیر طراحی شده برای مفاصل را دنبال می‌کند. از آنجایی که ربات دوپا ربات متحرکی است مساله مصرف انرژی در آن اهمیت زیادی دارد. به علت تغییرات احتمالی در محیط ربات، وجود مانع و اغتشاشات وارده بر ربات و انتظارات پیچیده‌تر از ربات‌های دوپا، پویایی، تطبیقی بودن و وصل خط<sup>6</sup> بودن الگوی راه رفتن در حین حرکت ربات مساله‌ای مهم و قابل توجه است. در قسمت کنترل حرکت، کنترل کننده باید پایداری حرکت را تضمین و قیدهایی مانند حداکثر گشتاور مجاز را برآورده کند. با توجه به نیازهای مطرح شده به نظر می‌رسد که مساله کنترل ربات دوپا شبیه مساله‌های کنترلی معمولی است. اما انسان برای حرکت از چنین

ربات‌های دوپا در بالا و پایین رفتن از پله‌ها استفاده شده است. کنترل کننده PID ساده‌ترین روشی است که از آن برای کنترل ربات‌های دوپا استفاده شده است. در محیط‌های ناهموار از قبیل بالا و پایین رفتن از پله‌ها با توجه به این که نقطه کار مفاصل به طور مداوم در حال تغییر است، لذا نیاز است تا ضرایب این کنترل کننده نیز اصلاح شود. برای این کار می‌توان از روش‌های هوشمند مانند شبکه‌های عصبی و سیستم‌های فازی استفاده کرد [۲] و [۳]. در واقع، در این حالت کنترل کننده PID در سطح پایینی از کنترل قرار دارد و از آن تنها برای ردیابی مسیر مرجع مفاصل استفاده می‌شود. طراحی الگوی حرکتی برای ربات‌های دوپا هنگام بالا و پایین رفتن از پله‌ها با طراحی مسیر پایدار مساله‌ای مهم و ضروری است. به طوری که اگر طراحی مسیر به خوبی انجام شود، بخش کنترلی با چالش کمتری مواجه خواهد بود. بسیاری از مراجع به منظور بررسی رفتار مرکز جرم ربات در حالت یک تکیه‌گاهی از مدل آونگ وارون استفاده می‌کنند [۴]، [۵] و [۶]. یکی دیگر از روش‌های طراحی مسیر استفاده از چندجمله‌ای درجه سوم برای تولید مسیر حرکت یک ربات دوپا در بالا و پایین رفتن از پله‌ها است [۶]. در این روش، مسیر لگن و پای متحرک ربات در فضای دکارتی طراحی شده و مسیر هریک از مفاصل با استفاده از سینماتیک وارون به دست می‌آید. ضرایب این چندجمله‌ای‌ها با استفاده از شرایط اولیه، شرایط نهایی و مسیر حرکتی ربات تعیین می‌شود. در [۶] هدف اصلی قرارداد ZMP در وسط پای تکیه‌گاه است به طوری که ربات هنگام بالا و پایین رفتن از پله‌ها دارای بیشترین حاشیه پایداری باشد. لذا، برای رسیدن به این هدف از دو روش پیشنهادی فازی ژنتیک و شبکه عصبی ژنتیک استفاده شده است. در الگوریتم عصبی ژنتیک، وزن‌های شبکه عصبی به صورت قطع خط<sup>۶</sup> با استفاده از الگوریتم ژنتیک بهینه‌سازی می‌شوند. در این حالت، مساله با استفاده از دو شبکه عصبی حل شده است. ورودی‌های شبکه اول، موقعیت پای ثابت و پای متحرک است و خروجی‌های آن ارتفاع لگن ربات نسبت به پای تکیه‌گاه و فاصله مفصل ران تا پای متحرک ربات است. ورودی‌های شبکه دوم نیز تغییر در زوایای پای متحرک و خروجی‌های آن نیز تغییر زاویه مچ پای متحرک و بالاتنه ربات است. روش پیشنهادی دوم نیز مشابه روش اول است با این تفاوت که به جای دو شبکه عصبی از دو سیستم فازی ممدانی استفاده شده است. در این روش نیز نیاز است که مسیر حرکتی ربات از قبل مشخص باشد به طوری که این طراحی مسیر تنها برای این مسیر خاص انجام می‌شود. چنانچه تغییری در محیط ایجاد شود ربات قادر به تطبیق خود با شرایط جدید نیست لذا مسیر جدیدی باید طراحی شود که از معایب این روش است. در [۷]، پارک و همکارانش به منظور طراحی مسیر حرکت برای ربات دوپا هنگام بالا و پایین رفتن از پله‌ها از روش

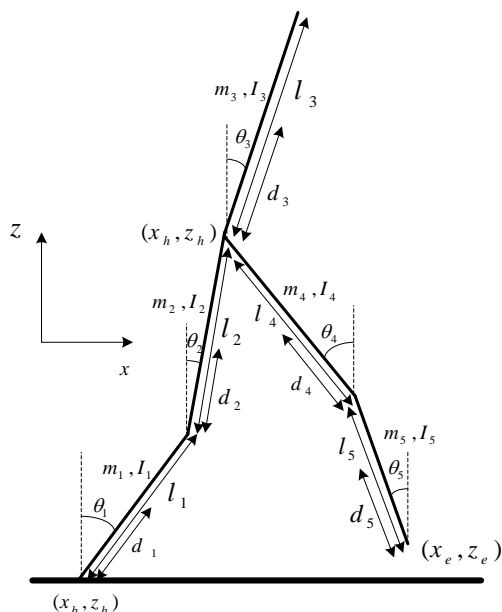
5 Variable Height Inverted Pendulum Mode  
6 Online

4 Offline

در کنترل‌کننده پیش‌بین نیز از شبکه عصبی به‌عنوان مدل پیش-بین بهره‌گرفته شده است. اما آموزش شبکه‌های عصبی به‌صورت قطع‌خط انجام می‌گیرد که این امر باعث می‌شود سیستم کنترلی ربات در مواقعی که عدم قطعیت در مدل ربات زیاد باشد، عملکرد خوب و قابل قبولی نداشته باشد.

در این مقاله، از کنترل پیش‌بین غیرخطی برای کنترل ربات دوپای صفحه‌ای در بالا و پایین رفتن از پله‌ها استفاده می‌شود. از ویژگی‌های اصلی این روش می‌توان به طراحی مسیر به‌صورت وصل‌خط اشاره کرد. در این حالت، هیچ مسیر از پیش تعیین شده‌ای برای حرکت ربات در نظر گرفته نمی‌شود و الگوی حرکت ربات تنها در قالب قیود بیان می‌شود. به بیان دیگر، طراحی الگوی حرکت حذف و مساله تنها معطوف به سیگنال‌های کنترلی یعنی گشتاورها می‌شود. گشتاورها به‌طور مستقیم طوری به‌دست می‌آیند که هدف‌های مساله و قیدها برآورده شده و دیگر نیازی به صرف انرژی و هزینه برای طراحی الگوی حرکت نباشد. سیگنال‌های تولید شده در این حالت قابل اعمال و عملی هستند و مساله پایداری نیز به‌طور همزمان در نظر گرفته می‌شود. استفاده از طراحی مسیر وصل‌خط می‌تواند مقاوم بودن کنترل‌کننده در مواجهه با عدم قطعیت در محیط (عدم اطلاع صحیح از ارتفاع و طول پله) را بهبود دهد. در تمام کارهای انجام شده در زمینه بالا و پایین رفتن ربات‌های دوپای طول و ارتفاع پله‌ها ثابت در نظر گرفته شده است. اما در این مقاله کنترل پیش‌بین به‌نحوی طراحی می‌شود که ربات قادر باشد تا از پله‌هایی با طول و ارتفاع متفاوت نیز بالا و پایین رفته و طول گام را نیز تعیین کند. به این ترتیب، کنترل‌کننده اختیار بیشتری دارد. اما مشکل جدی که در استفاده از کنترل پیش‌بین مطرح می‌شود، مدل‌گرابودن این کنترل‌کننده است به‌طوری که تغییر در پارامترهای ربات در حین حرکت به‌شدت روی عملکرد کنترل‌کننده اثر نامطلوب می‌گذارد. در این‌جا از شبکه عصبی RBF برای شناسایی مدل ربات استفاده می‌شود. لازم به ذکر است که این شناسایی به‌صورت بی‌درنگ انجام گرفته تا کنترل‌کننده قادر باشد در حضور عدم قطعیت‌ها و اغتشاشات بهترین عملکرد را از خود نشان دهد. از آنجایی که ربات دوپای سیستمی ذاتاً ناپایدار است، لذا هنگام استفاده از شبکه عصبی به‌عنوان مدل پیش‌بینی نیاز است که این شبکه قبل از حرکت ربات اندکی به‌صورت قطع-خط آموزش دیده تا ربات بتواند در ابتدای حرکت پایداری خود را حفظ کند. در سایر کارهای انجام شده در [9]، [10]، [12] و [13] نیز از کنترل پیش‌بین غیرخطی برای کنترل ربات دوپای استفاده شده است اما در این مراجع تنها حرکت ربات در محیط‌های صاف و هموار در نظر گرفته شده است. از این‌رو، حفظ پایداری در این شرایط برای ربات ساده‌تر است هم‌چنین آموزش شبکه‌های عصبی به‌صورت قطع‌خط است. بدین‌صورت که قبل از حرکت ربات با استفاده از داده‌های جمع‌آوری شده، شبکه‌های عصبی به‌صورت

الگوریتمی یعنی طراحی الگوی حرکت و سپس ردیابی دقیق آن استفاده نمی‌کند. در این‌جا هدف برآورده شدن یک‌سری خواسته‌های حرکتی به‌همراه تضمین قیده‌های مساله، تضمین پایداری به مفهوم خاص خود در حرکت انسان، هم‌چنین بهینه‌بودن کلیه مراحل و وصل‌خط بودن الگوریتم است. این نحوه بیان مساله، یعنی رسیدن به یک‌سری هدف‌ها در ضمن برآورده شدن قیدها و حل به‌صورت وصل‌خط، با روش کنترل پیش‌بین هم‌خوانی دارد. در [9] و [10]، از روش کنترل پیش‌بین برای حرکت ربات دوپای پنج درجه آزادی به‌صورت استاتیکی روی سطح صاف و هموار استفاده شده است. در این روش، حرکت پایدار ربات هدف اصلی قرار گرفته و با تعریف تابع هزینه و قیده‌های متناسب با ساختار ربات، حرکت ربات در هر لحظه به مساله بهینه‌سازی تبدیل شده که با حل آن سیگنال‌های کنترلی به‌دست می‌آیند. به این ترتیب، با در نظر گرفتن توانایی‌های ربات و محرک‌ها، طراحی مسیر و کنترل به‌صورت همزمان انجام شده است. در این حالت دیگر مساله خطا و ردیابی آن وجود ندارد، بلکه هدف‌هایی از قبیل پایداری و مصرف انرژی حداقل با وجود محدودیت‌ها مطرح می‌شوند. دیمیتروف و همکارانش در [11] از کنترل پیش‌بین غیرخطی برای جایابی گام مناسب ربات HRP-2 استفاده کرده‌اند. هدف نویسنده‌گان این مقاله آن است که نشان دهند تطابق پیوسته از موقعیت گام پاها براساس تولید گام حرکتی پایدار حتی در حضور اغتشاشات قوی امکان‌پذیر است. نتایج شبیه‌سازی بر روی ربات انسان‌نمای HRP-2 بهبود قابل توجهی نسبت به طرح‌های قبلی نشان داده است. جایابی گام در واقع کنترل گام است تا اطمینان حاصل شود که رابطه ایده‌آل بین سرعت بدن و طول گام برقرار شده است و ربات هنگام بالا بردن یک پا از پشت به زمین نمی‌افتد. در [12]، برای کنترل حرکت ربات دوپای پنج درجه آزادی از کنترل پیش‌بین غیرخطی هنگام راه رفتن روی سطح و عبور از موانع مختلف استفاده شده است. در این روش، طراحی الگوی حرکت ربات به‌طور ضمنی و بی‌درنگ در کنترل‌کننده پیش‌بین انجام می‌گیرد. طراحی الگوی حرکت توسط کنترل پیش‌بین، این مزیت را ایجاد می‌کند که مسیرهای مرجع برای مفصل‌های ربات با در نظر گرفتن گشتاورها و محدودیت‌های عملی تولید می‌شوند. در مقابل، کنترل‌کننده پیش-بین روشی مبتنی بر مدل است و مدل ربات به‌دلیل وجود عدم قطعیت‌ها مدل دقیقی نیست. بنابراین، از شبکه‌های عصبی برای شناسایی مدل دینامیکی ربات به‌صورت قطع‌خط استفاده شده است. در نهایت نیز تحلیل پایداری ربات با وجود کنترل‌کننده پیش‌بین و شبکه عصبی با استفاده از نگاشت پوانکاره که مربوط به سیستم‌های تکرار شونده است، انجام گرفته است. در [13]، از کنترل پیش-بین غیرخطی برای کنترل یک ربات دوپای پنج درجه آزادی هنگام راه رفتن روی سطوح صاف و عبور از موانع استفاده شده است. در این روش، حرکت ربات در قالب قیود و تابع هزینه بیان می‌شود.



شکل ۱. ساختار ربات دوپای پنج لینکی [۱۴]

$$D(\theta)\ddot{\theta} + H(\theta, \dot{\theta})\dot{\theta} + G(\theta) = T \quad (1)$$

که در آن  $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T$  بردار موقعیت مفاصل،  $T = [T_1, T_2, T_3, T_4, T_5]^T$  بردار گشتاور اعمالی به مفاصل ربات،  $D(\theta)$  ماتریس  $5 \times 5$  متقارن مثبت معین اینرسی،  $H(\theta, \dot{\theta})$  ماتریس  $5 \times 5$  نشان دهنده جمله‌های گریز از مرکز و کوریولیس، و  $G(\theta)$  بردار  $5 \times 1$  شامل جمله‌های مربوط به گرانی است.  $\dot{\theta}$ ،  $\ddot{\theta}$  بردارهای  $5 \times 1$  بوده و به ترتیب مختصات موقعیت، سرعت، شتاب مفاصل ربات هستند.

### حالت دوتکیه‌گامی

باتوجه به این که در این حالت هر دو پای ربات روی زمین ثابت هستند، به دینامیک ربات قیدهایی اضافه می‌شود که به صورت قیدهایی هولونومیک زیر قابل‌ارایه هستند [۱۴]:

$$\Phi(\theta) = \begin{pmatrix} x_e - x_b - L \\ z_e - z_b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^5 (a_i l_i \sin(\theta_i)) - L \\ \sum_{i=1}^5 a_i l_i \cos(\theta_i) \end{pmatrix} = 0 \quad (2)$$

که در آن  $L$  فاصله بین نوک پای ثابت و پای متحرک است که در هر گام مقداری ثابت است. سایر ثابت‌ها در شکل ۱ نشان داده شده‌اند. با در نظر گرفتن قید (۲)، دینامیک ربات در حالت دوتکیه‌گامی را می‌توان به صورت زیر نوشت:

$$D(\theta)\ddot{\theta} + H(\theta, \dot{\theta})\dot{\theta} + G(\theta) = J^T(\theta)\lambda + T \quad (3)$$

قطع خط آموزش دیده و در طول حرکت ربات پارامترهای شبکه به هیچ وجه تغییر نمی‌کنند که این مساله باعث کاهش انعطاف پذیری روش پیشنهادی شده و تغییر در شرایط محیطی (همانند عدم قطعیت و اغتشاش) ممکن است باعث ناپایداری ربات شود. اما در این مقاله، حرکت ربات در بالا و پایین رفتن از پله‌ها در نظر گرفته شده که نشان از توانایی ربات برای حرکت در محیط‌های ناهموار و نامعلوم است. همچنین، برای آموزش شبکه‌های عصبی به عنوان مدل پیش‌بینی در کنترل کننده پیش‌بین از هر دو آموزش قطع-خط و وصل خط استفاده شده است که این امر باعث عملکرد بهتر روش پیشنهادی در این مقاله در مقایسه با سایر مقالات می‌گردد.

### مدل سازی ربات دوپا

ربات مورد بررسی در این مقاله، رباتی پنج لینکی با پنج درجه آزادی است که در شکل ۱ نشان داده شده است. هر گام از حرکت ربات دارای سه فاز یک تکیه‌گامی، دوتکیه‌گامی و نگاشت ضربه دو-تکیه‌گامی است. حالت یک تکیه به وضعیتی گفته می‌شود که یک پا روی زمین و به عنوان پای تکیه‌گاه عمل کرده و پای دیگر از پشت به جلوی آن حرکت می‌کند. این حالت از لحظه‌ای شروع می‌شود که نوک پای متحرک از زمین جدا شده و تا هنگامی که پای متحرک به زمین برسد ادامه می‌یابد. در حالت دوتکیه‌گامی، هر دوی ربات با سطح زمین در تماس هستند و تنه می‌تواند به آرامی به جلو حرکت کند. این فاز از حرکت ربات دوپا با قرار گرفتن پای متحرک روی زمین آغاز و با جدا شدن پای پشتی از زمین به پایان می‌رسد. به محض برخورد پای ربات با زمین، پدیده ضربه به صورت لحظه‌ای اتفاق می‌افتد. ضربه باعث می‌شود که سرعت مفاصل ربات به صورت ناگهانی تغییر کند. ضربه می‌تواند در حالت تک پا یا دو پا باشد. ضربه حالت تک پا وقتی است که به محض رسیدن پای متحرک به زمین پای ثابت از زمین بلند شود. ضربه حالت دو پا وقتی است که در لحظه برخورد پای متحرک با زمین، پای ثابت هم‌چنان روی زمین بماند. در این جا فرض می‌شود که ضربه در حالت دو پا اتفاق می‌افتد. معادلات دیفرانسیل حرکت ربات دوپا در هر فاز از حرکت با استفاده از روش لاگرانژ به طور جداگانه محاسبه می‌شود.

### حالت یک تکیه‌گامی

معادله دینامیکی توصیف کننده حرکت ربات دوپا در حالت یک-تکیه‌گامی را می‌توان به فرم استاندارد زیر بیان کرد [۱۴]:

اندازه‌گیری‌های جدید کل فرآیند پیش‌بینی و بهینه‌سازی تکرار می‌شود. بلوک دیاگرام کنترل پیش‌بین پیش‌بینی پیشنهادی در این مقاله به صورت شکل ۲ است.

### تابع هزینه

هدف از کنترل ربات دوپا، یافتن قانون کنترلی است که ربات را وادار سازد با حفظ پایداری، عدم برخورد با موانع محیطی و سرعت مناسب از پله‌ها بالا و پایین برود. بر این اساس، تابع معیار به‌گونه‌ای تعریف می‌شود که ربات در حین حرکت بیش‌ترین پایداری را داشته‌باشد. معیار مورد استفاده به‌منظور بررسی پایداری ربات هنگام راه‌رفتن، نقطه ممان صفر ( $ZMP^T$ ) است. فاصله  $ZMP$  از مرزهای ناحیه تکیه‌گاهی را می‌توان به‌عنوان معیاری از میزان پایداری سیستم در نظر گرفت؛ به‌گونه‌ای که اگر  $ZMP$  در وسط ناحیه تکیه‌گاهی باشد ربات دارای بیش‌ترین پایداری است و با نزدیک شدن این نقطه به مرزها از پایداری ربات کاسته شده و با خارج شدن این نقطه، ربات دچار ناپایداری و سقوط می‌شود. لذا تابع هزینه به‌صورت فاصله  $ZMP$  از مرز ناحیه پایداری تعریف می‌شود. برای هر دو حالت یک‌تکیه‌گاهی و دو‌تکیه‌گاهی کف پای تکیه‌گاه به‌عنوان ناحیه پایداری در نظر گرفته می‌شود. لذا خطای  $ZMP$  به‌صورت زیر تعریف می‌شود:

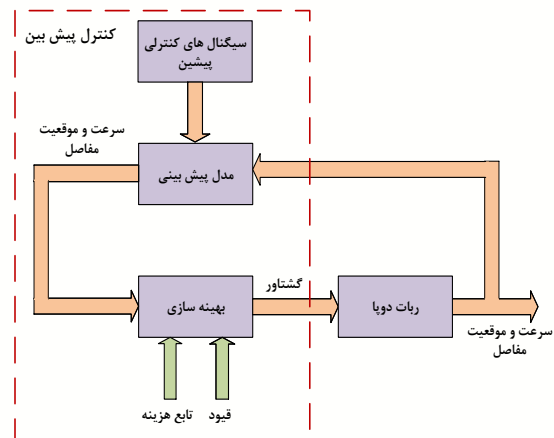
$$D_{ZMP} = \left| \frac{s}{2} - x_{ZMP} \right| \quad (5)$$

که در آن  $x_{ZMP}$  موقعیت افقی  $ZMP$  و  $s$  طول کف پای ربات است. همچنین برای بهینه‌کردن مصرف انرژی، ربات باید قادر باشد با کمترین ورودی کنترلی اهداف تعریف شده را انجام دهد. در نهایت، تابع هزینه کل برای دستیابی به اهداف ذکر شده به‌صورت زیر تعریف می‌شود:

$$J = \sum_{j=1}^{N_p} Q \left( \frac{e^{D_{ZMP}(k+j|k)} - e^{D_{ZMP_{min}}}}{e^{D_{ZMP_{max}}} - e^{D_{ZMP_{min}}}} \right) + \sum_{j=1}^{N_c} \mathbf{u}^T(k+j|k) R \mathbf{u}(k+j|k) \quad (6)$$

که در آن  $D_{ZMP}(k+j|k)$  نشان‌دهنده مقدار خطای  $ZMP$  در لحظه  $k+j$  است که در لحظه  $k$  پیش‌بینی می‌شود.  $N_p$  و  $N_c$  به ترتیب افق  $N_p$  پیش‌بینی و افق کنترلی هستند. بردار ورودی-های کنترلی ربات،  $R$  و  $Q$  به ترتیب وزن‌های متناظر با خطای بردار ورودی کنترلی و  $ZMP$  و  $[D_{ZMP_{min}}, D_{ZMP_{max}}]$  بازه تغییرات خطای  $ZMP$  می‌باشند که این بازه تغییرات باتوجه به ابعاد کف پای ربات تعیین می‌شود.

که در آن  $\mathbf{G}, \mathbf{H}, \mathbf{D}$  و  $\mathbf{T}$  مشابه رابطه (۱) بوده و  $\lambda$  بردار  $2 \times 1$  شامل ضرایب لاگرانژ است. همچنین،  $\mathbf{J} = \partial \Phi / \partial \theta$  ماتریس ژاکوبی با ابعاد  $2 \times 5$  است.



شکل ۲. بلوک دیاگرام کنترل پیش‌بین

### ضربه حالت دو تکیه‌گاهی

هنگامی که پای متحرک ربات به زمین می‌رسد ضربه اتفاق افتاده و سرعت زاویه‌ای مفاصل به‌طور آنی تغییر می‌کند. ضربه می‌تواند به صورت‌های مختلفی اتفاق بیفتد. اما به‌طور مرسوم فرض می‌شود که ضربه بین دو جسم صلب به‌صورت پلاستیک و در زمان ناچیز اتفاق می‌افتد. همچنین فرض می‌شود که بردار موقعیت مفاصل در اثر ضربه تغییری نمی‌کند. بنابراین، می‌توان نداشت ضربه را برای ربات دوپا به‌صورت زیر بیان کرد [۱۴]:

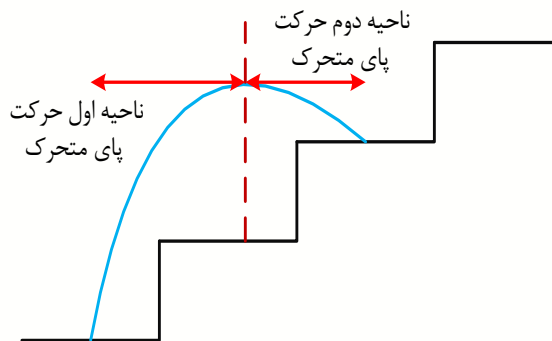
$$\dot{\theta}^+ = \dot{\theta}^- + \mathbf{D}(\theta)^{-1} \mathbf{J}(\theta)^T (\mathbf{J}(\theta) \mathbf{D}(\theta)^{-1} \mathbf{J}(\theta)^T)^{-1} (\mathbf{J}(\theta) \dot{\theta}^-) \quad (4)$$

که در آن  $\dot{\theta}^+$  و  $\dot{\theta}^-$  به ترتیب سرعت زاویه‌ای مفاصل ربات قبل و بعد از لحظه برخورد است.

### کنترل پیش‌بین

برخلاف روش‌های کنترلی موجود که در آن‌ها قانون کنترلی براساس خروجی گذشته سیستم به‌دست می‌آید، روش کنترل پیش‌بین یک روش بهینه مبتنی بر مدل است که از پیش‌بینی خروجی آینده سیستم برای به‌دست آوردن قانون کنترلی استفاده می‌کند. در این روش کنترلی، در هر لحظه نمونه برداری  $k$ ، براساس اندازه‌گیری‌های انجام گرفته تا لحظه  $k$ ، کنترل کننده با استفاده از مدل حلقه باز سیستم، خروجی سیستم را به تعداد نمونه پیش‌بینی کرده و ورودی کنترلی را به تعداد  $N_c$  نمونه ( $N_c \leq N_p$ ) به‌گونه‌ای تولید می‌کند که تابع معیار از پیش تعیین شده کمینه گردد. سپس، جمله اول ورودی کنترلی تا لحظه نمونه برداری آینده به سیستم اعمال می‌شود. در لحظه نمونه برداری بعدی براساس

## قیود



شکل ۳. مسیر پای متحرک در بالارفتن از پله‌ها

## یک تکیه‌گاهی

در حالت یک تکیه‌گاهی هنگام بالارفتن از پله‌ها باید قیده‌های زیر برقرار باشد تا ربات بتواند حرکتی طبیعی و پایدار داشته باشد:

**قید ۱)** یکی از قیده‌های در نظر گرفته شده برای حرکت ربات دوپا، قید حرکتی مفصل است. در انسان هر مفصل می‌تواند در ناحیه‌ی خاصی حرکت کند. برای مثال، زانو نمی‌تواند بیش‌تر از ۱۸۰ درجه باز شود. همچنین، ورودی‌های کنترلی باید به‌گونه‌ای تعیین شوند که باعث شود موقعیت مفصل ربات در بازه از پیش تعیین شده که از طریق محدودیت‌های فیزیکی تحمیل می‌شود، قرار گیرد تا روش پیشنهادی از لحاظ عملی قابل پیاده‌سازی باشد. بدین منظور می‌توان از قید زیر برای موقعیت مفصل استفاده کرد:

$$\theta_{i,\min} \leq \theta_i \leq \theta_{i,\max} \quad (7)$$

که در آن  $\theta_i$  موقعیت مفصل  $i$ ام و  $\theta_{i,\max}$  و  $\theta_{i,\min}$  به ترتیب کران بالا و پایین موقعیت مفصل  $i$ ام می‌باشند.

**قید ۲)** در طول حرکت، ربات باید تنها حرکتی روبه جلو داشته باشد. لذا، نیاز است سرعت مرکز جرم ربات که معیاری از سرعت ربات است، همواره مثبت باشد. بنابراین می‌توان نوشت:

$$\dot{x}_{COM} \geq 0 \quad (8)$$

**قید ۳)** در هنگام حرکت، تنه انسان تقریباً عمود است و تغییرات زاویه‌ای محدودی دارد، بنابراین

$$\alpha_{\min} \leq \theta_3 \leq \alpha_{\max} \quad (9)$$

که در آن  $\theta_3$  زاویه بالاتنه ربات نسبت به محور قائم و  $\alpha_{\max}$  و  $\alpha_{\min}$  به ترتیب کران بالا و پایین حرکت بالاتنه ربات است.

**قید ۴)** محدودیت مهمی که در قسمت کنترلی وجود دارد، محدودیت در گشتاورهای قابل اعمال توسط محرک‌ها است. برای این‌که روش کنترلی از دیدگاه عملی قابل پیاده‌سازی باشد، باید ورودی‌های کنترلی در بازه معینی قرار گیرند زیرا دامنه گشتاور ورودی مفصل که در این‌جا همان ورودی کنترلی سیستم است،

همان‌گونه که بیان شد از دلایل اصلی استفاده از کنترل پیش‌بین عدم مسیر مرجع برای ردیابی است. به عبارت دیگر، جزئیات حرکت ربات در قالب قیده‌ها بیان می‌شود. همچنین محدودیت‌های دیگری نیز در حرکت ربات وجود دارد که در قسمت تابع هزینه به آن‌ها توجهی نشده است. از آنجایی که رفتار ربات در بالا و پایین رفتن از پله‌ها متفاوت است، لذا نیاز است برای هر کدام از این دو مرحله قیده‌های مناسبی تعریف شود. لازم به ذکر است که قیده‌ها در حالت یک تکیه‌گاهی و دو تکیه‌گاهی نیز با هم تفاوت دارند. لذا قیده‌ها در دو بخش جداگانه تعریف خواهند شد. همچنین باید توجه شود که در تعریف قیود برای حرکت ربات تنها قیده‌های ضروری و مهم که باعث حرکتی پایدار و مشابه راه رفتن انسان می‌شود، در نظر گرفته شده است. شاید بتوان قیده‌های دیگری را نیز تعریف کرد اما افزایش تعداد قیود حل مساله بهینه‌سازی را دشوارتر کرده و ممکن است تأثیر قابل توجهی نیز بر حرکت ربات نداشته باشد. بنابراین، قیودی که در ادامه برای راه رفتن ربات دوپا بر روی پله‌ها بیان می‌شود حداقل قیودی هستند که منجر به حرکتی پایدار و مشابه راه رفتن انسان می‌شوند. نکته دیگری که باید در نظر گرفت این است که برای سرعت حرکت ربات مقدار از پیش تعیین شده‌ای در نظر گرفته نمی‌شود؛ یعنی مقدار سرعت حرکت ربات از قبل تعیین شده نیست و کنترل‌کننده با توجه به شرایط محیطی و پایداری ربات سرعت حرکت را تعیین می‌کند که این خود یکی از مزایای روش پیشنهادی است که در بسیاری از روش‌های دیگر چنین امکانی وجود ندارد و سرعت حرکت ربات از قبل توسط کاربر تعیین می‌شود که این باعث کاهش انعطاف‌پذیری روش می‌شود.

## بالارفتن از پله

طراحی قیده‌ها برای بالارفتن از پله‌ها نسبت به حالت راه رفتن روی سطح صاف و هموار دشوارتر است. در حالت بالارفتن از پله‌ها پای متحرک ربات مسیر متقارنی را طی نمی‌کند. لذا، مسیر حرکت پای متحرک به دو بخش جداگانه تقسیم می‌شود (شکل ۳). سرعت افقی و عمودی پای متحرک ربات در هر یک از این دو بخش باید به‌طور مناسبی تعیین شود تا ربات بتواند عملکردی مطلوب داشته باشد.

$$\begin{cases} \dot{x}_e \geq \beta_{\min} \dot{x}_{\text{COM}} \sin\left(\frac{y_e - y_0}{2(H_m - y_0)} \pi\right) \\ \dot{x}_e \leq \beta_{\max} \dot{x}_{\text{COM}} \sin\left(\frac{y_e - y_0}{2(H_m - y_0)} \pi\right) \end{cases} \quad (13)$$

که در آن  $\dot{x}_e$  سرعت افقی نوک پای متحرک،  $\beta_{\min}$  و  $\beta_{\max}$  ثابت-های طراحی،  $H_m$  حداکثر ارتفاع پای متحرک و  $y_0$  در ناحیه اول حرکت، برابر ارتفاع پله مبدأ و در ناحیه دوم، برابر ارتفاع پله مقصد است. هنگامی که نوک پای متحرک به حداکثر ارتفاع خود می‌رسد، مقدار تابع سینوس برابر واحد شده و سرعت افقی نوک پای متحرک مقداری بین حداقل و حداکثر سرعت مجاز خواهد بود. در لحظه رسیدن پا به زمین سرعت افقی پا صفر می‌شود. به این ترتیب، پای ربات در انتهای حالت یک‌تکیه‌گاهی روی زمین کشیده نخواهد شد.

**قید ۸)** تا قبل رسیدن نوک پای متحرک به حداکثر ارتفاع خود، سرعت عمودی آن مثبت و با رسیدن به حداکثر ارتفاع این سرعت صفر می‌شود. بعد از گذشتن نوک پا از حداکثر ارتفاع، سرعت عمودی منفی شده و هنگام رسیدن به سطح پله بالایی صفر می‌شود. این جزئیات را می‌توان به صورت زیر بیان کرد [۱۲]:

$$\begin{cases} \dot{y}_e \geq \delta \sin(|x_e - x_b| \pi) \sin\left(\frac{y_e - y_0}{H_m - y_0} \pi\right) & x_e \leq x_b \\ \dot{y}_e \leq -\delta \sin(|x_e - x_b| \pi) \sin\left(\frac{y_e - y_0}{H_m - y_0} \pi\right) & x_e \geq x_b \end{cases} \quad (14)$$

که در آن  $x_b$  موقعیت افقی است که در آن پای متحرک به حداکثر ارتفاع خود می‌رسد و  $\delta$  ثابت طراحی است. باتوجه به قید بالا، سرعت عمودی پای متحرک هنگام رسیدن به سطح پله بالایی برابر با صفر است. این باعث می‌شود تا ضربه هنگام برخورد پا با سطح پله کاهش یابد.

### حالت دوتکیه‌گاهی

در بالارفتن از پله هنگامی که ربات در فاز دوتکیه‌گاهی قرار می‌گیرد خود را به جلو می‌کشد. قیده‌های این حالت با قیده‌های ۱ تا ۵ حالت یک‌تکیه‌گاهی یکسان هستند.

### پایین رفتن از پله

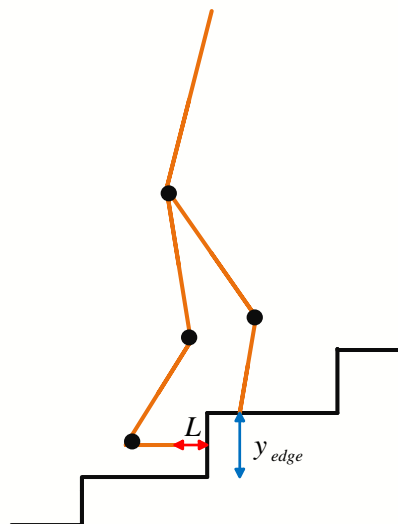
در حالت پایین رفتن از پله، تابع هزینه مشابه حالت بالارفتن از پله است. هم‌چنین ساختار قیده‌های این حرکت نیز تقریباً با بالارفتن از پله یکسان است.

محدود بوده و نمی‌تواند از مقدار مشخصی تجاوز کند. لذا، باید قید زیر را برای گشتاورهای ربات در نظر گرفت:

$$\mathbf{T}_{\min} \leq \mathbf{T} \leq \mathbf{T}_{\max} \quad (10)$$

**قید ۵)** هم‌چنین، در بالارفتن از پله نیاز است که مفصل ران ربات حرکتی روبه بالا داشته باشد. این قید به صورت زیر قابل بیان است:

$$\dot{y}_{\text{hip}} \geq 0 \quad (11)$$



شکل ۴. عدم برخورد نوک پای ربات با لبه پله هنگام بالارفتن از پله

**قید ۶)** برخورد نوک پای متحرک با لبه پله مشکلی است که در بالارفتن از پله ممکن است ایجاد شود. برای حل این مشکل باید فاصله بین نوک پای متحرک تا لبه پله جلویی همواره بزرگ‌تر از صفر باشد. این قید را می‌توان به صورت زیر نوشت:

$$\text{IF } y_e \leq y_{\text{edge}} \text{ THEN } L \geq 0 \quad (12)$$

که در آن  $y_e$  موقعیت عمودی پای متحرک و  $y_{\text{edge}}$  ارتفاع پله جلویی است، که در شکل ۴ نشان داده شده است.

**قید ۷)** قیدهایی که تاکنون بیان شد هیچ‌کدام پای متحرک را مقید به حرکت در مسیری مشخص نمی‌کند. با شروع حالت یک-تکیه‌گاهی، پای متحرک در حالی که در پشت پای ثابت قرار دارد از زمین جدا می‌شود. نوک پای متحرک با پیمودن مسیری مانند توزیع سینوسی در صفحه جانبی حرکت کرده و در لحظه گذشتن از لبه پله جلویی، به حداکثر ارتفاع خود یعنی  $H_m$  می‌رسد. سپس مجدداً با پیمودن مسیری مشابه قسمت قبل اما غیرمقارن ارتفاع خود را کم کرده و با رسیدن نوک پای متحرک به زمین، حالت یک‌تکیه‌گاهی به پایان می‌رسد. برای دستیابی به این مسیر باید سرعت نوک پای متحرک را در دو جهت افقی و عمودی به‌طور مناسبی کنترل کرد. قید سرعت افقی پای متحرک را می‌توان بدین صورت بیان کرد [۱۲]

$$\text{IF } x_{diff} \geq 0 \text{ THEN } L > 0 \quad (16)$$

**قید ۹** قید دیگری که باید در نظر گرفت، قید محدوده قرارگیری پای متحرک ربات روی سطح پله پایینی است. به طوری که نوک پای متحرک ربات نباید از لبه پله پایینی خارج شود. با توجه به شکل ۵ این قید را می توان به صورت زیر تعریف کرد:

$$x_{ed} \geq 0 \quad (17)$$

### حالت دوتکیه گاهی

قیدهای این حالت نیز همان قیدهای ۱ تا ۵ حالت یک تکیه گاهی است.

### کنترل پیش بین عصبی - تطبیقی

با توجه به این که، کنترل پیش بین مبتنی بر مدل سیستم می باشد، لذا اگر پارامترهای سیستم در طول فرآیند کنترل تغییر کنند، اثر نامطلوبی بر روی عملکرد سیستم دارد. از این رو استفاده از مکانیزمی که بتواند پارامترهای سیستم را به صورت بی درنگ شناسایی کند، لازم به نظر می رسد. در این صورت، کنترل پیش بین به یک کنترل کننده تطبیقی تبدیل شده و می تواند خود را با تغییر پارامترهای سیستم تطبیق دهد. در این مقاله، از شبکه های عصبی مصنوعی برای شناسایی سیستم استفاده می شود. شبکه عصبی مورد نظر با استفاده از ورودی و خروجی های ربات در زمان های نمونه برداری فعلی، خروجی ربات را در زمان نمونه برداری بعدی تقریب می زند. مدل پیش بین دینامیک ربات به صورت شکل ۶ می باشد.

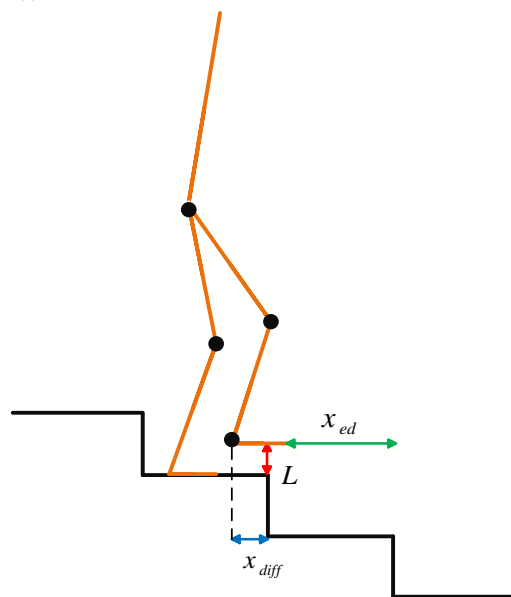
به منظور پیش بینی خروجی ربات در چندین گام بعدی، می توان شبکه عصبی را به صورت متوالی استفاده کرد. در این مقاله، از شبکه عصبی RBF برای پیش بینی موقعیت و سرعت زاویه ای مفصل ربات استفاده می شود. بدین منظور، برای هر مفصل ربات از ۲ شبکه عصبی جداگانه استفاده می شود. با توجه به این که ربات دارای پنج مفصل می باشد، از ۱۰ شبکه عصبی RBF استفاده شده است که پنج شبکه اول وظیفه پیش بینی زاویه و پنج شبکه دوم وظیفه پیش بینی سرعت زاویه ای مفصل ربات را برعهده دارند. ورودی های شبکه عصبی مربوط به مفصل  $i$  ام عبارتند از گشتاور ورودی مفصل  $i$  ام و موقعیت و سرعت هر پنج مفصل ربات در لحظه فعلی و خروجی آن ها، در پنج شبکه اول موقعیت و در پنج شبکه دوم سرعت زاویه ای مفصل  $i$  ام در لحظه بعدی می باشد. در نتیجه، هر کدام از شبکه ها دارای ۱۱ ورودی و ۱ خروجی می باشند. در هر کدام از این شبکه ها، ۷ سلول برای لایه پنهان در نظر گرفته شده است. این تعداد سلول برای لایه پنهان براساس پیچیدگی نگاشت غیر خطی مورد نظر و با مقداری سعی و خطا تعیین می شود

### حالت یک تکیه گاهی

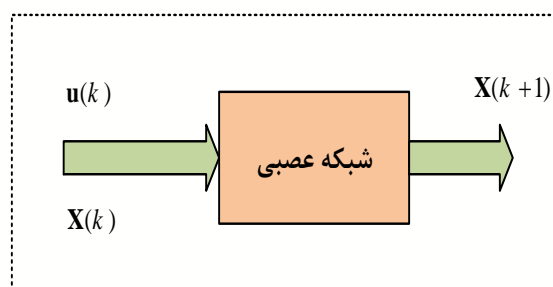
قیدهای ۱ تا ۴ مشابه قیدهای بالا رفتن از پله است و از تکرار آن ها خودداری می شود.

**قید ۵** با توجه پایین رفتن ربات از پله ها، مرکز جرم ربات باید در جهت روبه پایین حرکت کند تا ربات حرکتی نرم و طبیعی داشته باشد. این قید را می توان به صورت زیر بیان کرد:

$$\dot{y}_{COM} \leq 0 \quad (15)$$



شکل ۵. محدودیت های ربات دوپا در پایین رفتن از پله ها



شکل ۶. مدل پیش بینی با استفاده از شبکه عصبی

**قیدهای ۶ و ۷** سرعت افقی و عمودی پای متحرک نیز دقیقاً مشابه بالا رفتن از پله است.

**قید ۸** به منظور عدم برخورد پای متحرک با لبه پله (شکل ۵) لازم است تا ارتفاع پای متحرک تا قبل از گذشتن از لبه این پله از ارتفاع آن بیشتر باشد. نحوه بیان این محدودیت در شکل ۵ و قید زیر نشان داده شده است:



$$E = \frac{1}{2} \sum_{j=1}^N (y - y_{RBF})^2 \quad (20)$$

که در آن  $N$  تعداد نمونه‌ها برای آموزش قطع خط است. مطابق با روش گرادیان نزولی، پارامترهای شبکه را می‌توان به صورت زیر آموزش داد [۱۵]:

$$\begin{aligned} w_j(k+1) &= w_j(k) + \Delta w_j(k) \\ \sigma_j(k+1) &= \sigma_j(k) + \Delta \sigma_j(k) \\ t_{ji}(k+1) &= t_{ji}(k) + \Delta t_{ji}(k) \end{aligned} \quad (21)$$

که در آن مقدار تغییر هر کدام از پارامترهای شبکه بدین صورت محاسبه می‌شود [۱۵]:

$$\Delta w_j(k) = -\eta \frac{\partial E}{\partial w_j(k)} = \eta(y - y_{RBF})h_j \quad (22)$$

$$\Delta \sigma_j(k) = -\eta \frac{\partial E}{\partial \sigma_j(k)} = \eta(y - y_{RBF})w_j h_j \frac{\|\mathbf{x} - \mathbf{t}_j\|^2}{\sigma_j^3} \quad (23)$$

$$\Delta t_{ji}(k) = -\eta \frac{\partial E}{\partial t_{ji}(k)} = \eta(y - y_{RBF})w_j \frac{x_j - t_{ji}}{\sigma_j^2} \quad (24)$$

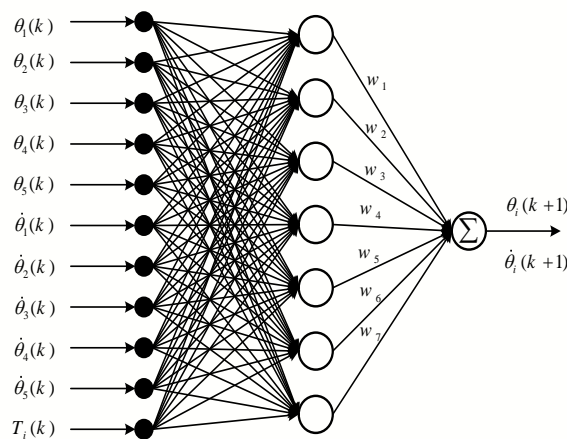
لازم به ذکر است که در مرحله آموزش قطع خط، هر سه پارامتر شبکه یعنی وزن‌های لایه خروجی، عرض و مرکز توابع گوسی آموزش داده می‌شود. اما در مرحله وصل خط به منظور کاهش زمان محاسبات تنها وزن‌های لایه خروجی شبکه آموزش داده می‌شود.

### نتایج شبیه‌سازی

در این بخش، نتایج به دست آمده از شبیه‌سازی حرکت ربات دوپا در بالا و پایین رفتن از پله‌ها که در محیط متلب انجام شده است، ارائه می‌شود. ربات مورد بحث در این مقاله، ربات دوپا با پنج درجه آزادی است.

مساله بهینه‌سازی باید در هر لحظه نمونه‌برداری  $k$  حل گردد و نتیجه آن دنباله کنترلی  $\{\mathbf{u}^*(k+1|k), \dots, \mathbf{u}^*(k+N_c|k)\}$  می‌باشد. در نتیجه، در لحظه نمونه‌برداری بعدی  $k+1$ ، عنصر اول دنباله کنترلی  $\mathbf{u}^*(k+1|k)$  به سیستم اعمال شده و در این حین مساله بهینه‌سازی دوباره حل می‌شود. با تکرار مساله بهینه‌سازی و کنترل در هر لحظه نمونه‌برداری، گشتاور مفاصل به گونه‌ای تعیین می‌شود که ربات حرکتی مطلوب داشته و در عین حال محدودیت‌های در نظر گرفته شده را نیز برآورده سازد. برای حل مساله بهینه‌سازی از تابع `fmincon` در جعبه‌ابزار بهینه‌سازی متلب استفاده شده است. الگوریتم این تابع بر مبنای روش درجه دو پی‌درپی (SQP) استوار است. جدول ۱ پارامترهای ربات مورد استفاده را بیان می‌کند [۱۴]. محدوده حرکتی مفاصل ربات نیز در جدول ۲ بیان شده است، مقدار این پارامترها به گونه‌ای تعیین شده است که تا حد

به گونه‌ای که مصالحه‌ای بین دقت شبکه‌ها و زمان محاسبات برقرار گردد. در روش پیشنهادی، مساله بهینه‌سازی به صورت وصل خط و در هر لحظه نمونه‌برداری حل می‌شود. لذا افزایش تعداد سلول‌ها باعث افزایش زمان محاسبات شده که ممکن است زمان حل مساله بهینه‌سازی بیشتر از زمان نمونه‌برداری شود. به همین دلیل باید تعداد سلول‌ها در لایه پنهان را به درستی انتخاب کرد. شکل ۷ ساختار شبکه‌های عصبی مورد استفاده را نشان می‌دهد.



شکل ۷. ساختار شبکه عصبی به کار رفته در شناسایی مدل ربات

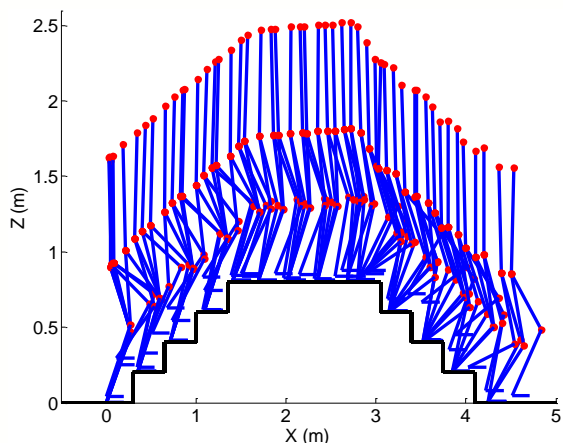
آموزش شبکه‌های عصبی مورد نظر در دو مرحله قطع خط و وصل خط با استفاده از روش گرادیان نزولی انجام می‌شود. از آنجایی که ربات دوپا سیستمی ذاتاً ناپایدار است، لذا هنگام استفاده از شبکه عصبی به عنوان مدل پیش‌بینی نیاز است که این شبکه قبل از حرکت ربات اندکی به صورت قطع خط آموزش ببیند تا ربات بتواند در ابتدای حرکت پایداری خود را حفظ کند. با فرض سلول  $z$  ام در لایه پنهان می‌توان رابطه زیر را برای هر سلول در لایه پنهان نوشت:

$$h_j = \exp\left(-\frac{\|\mathbf{x} - \mathbf{t}_j\|^2}{2\sigma_j^2}\right); \quad j = 1, 2, \dots, m \quad (18)$$

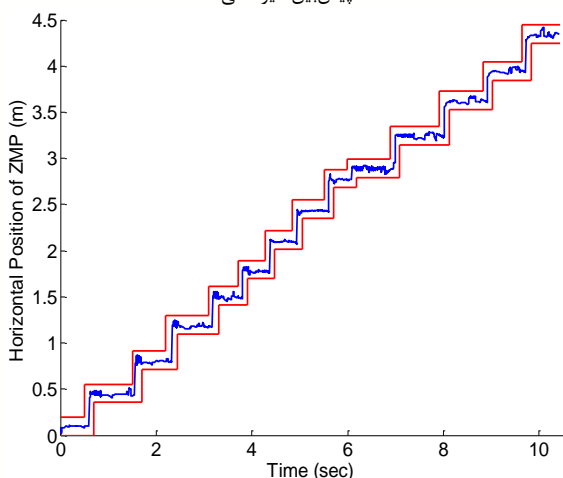
که در آن  $m$  تعداد سلول‌های لایه پنهان،  $\mathbf{t}_j = [t_{j1}, \dots, t_{jp}]^T$  بردار مرکز توابع گوسی سلول  $z$  ام و  $\boldsymbol{\sigma} = [\sigma_1, \dots, \sigma_m]^T$  بردار عرض توابع گوسی است. بنابراین خروجی شبکه عصبی RBF به صورت زیر محاسبه می‌شود:

$$y_{RBF} = w_1 h_1 + \dots + w_m h_m = \sum_{i=1}^m w_i G(\|\mathbf{x} - \mathbf{t}_i\|) \quad (19)$$

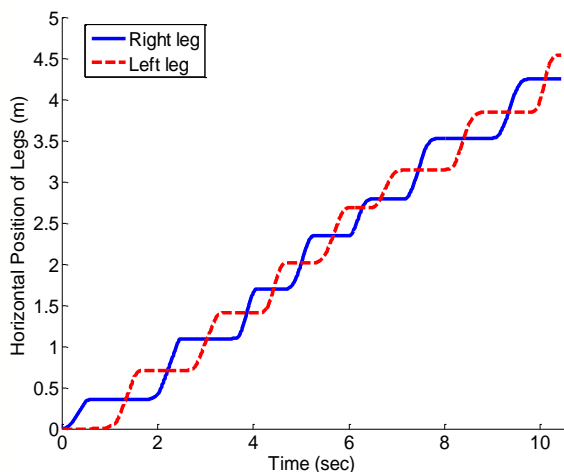
جهت تطبیق یا آموزش پارامترهای شبکه، ابتدا تابع هزینه به صورت مجموع مربعات خطا که معرف اختلاف بین خروجی شبکه و خروجی مطلوب است، تعریف می‌شود. این تابع به صورت زیر می‌باشد:



شکل ۸. توالی حرکت ربات دوپا در بالا و پایین رفتن از پله‌ها با استفاده از کنترل پیش‌بین غیرخطی



شکل ۹. موقعیت افقی ZMP در بالا و پایین رفتن از پله‌ها



شکل ۱۰. موقعیت افقی پای متحرک ربات در بالا و پایین رفتن از پله‌ها

امکان شبیه به محدودیت‌های بدن انسان باشد. در قسمت اول از شبیه‌سازی، برای مدل پیش‌بینی از معادلات دینامیکی گسسته-شده ربات استفاده می‌شود. هم‌چنین، در هر دو حالت بالا و پایین-رفتن از پله‌ها طول و ارتفاع پله‌ها ثابت در نظر گرفته می‌شود. در این حالت ارتفاع هر پله  $H = 20\text{cm}$  و عرض آن  $W = 35\text{cm}$  است. سایر پارامترهای قابل تنظیم در کنترل‌کننده و قیده‌ها در جدول ۳ ذکر شده است.

نتایج شبیه‌سازی هنگام بالا و پایین رفتن از پله‌ها در شکل‌های ۸ تا ۱۴ آورده شده است.

جدول ۱. پارامترهای ربات دوپا

شماره لینک	پارامترهای ربات			
	طول (m)	جرم (kg)	اینرسی ( $\text{kgm}^2$ )	مرکز جرم (m)
۱	۰٫۵۴	۳٫۷	۰٫۵	۰٫۲۸۵
۲	۰٫۵	۸٫۵۵	۰٫۵	۰٫۳۱
۳	۰٫۷	۲۵	۰٫۵	۰٫۴
۴	۰٫۵	۸٫۵۵	۰٫۵	۰٫۳۱
۵	۰٫۵۴	۳٫۷	۰٫۵	۰٫۲۸۵

جدول ۲. محدوده متغیرهای ربات

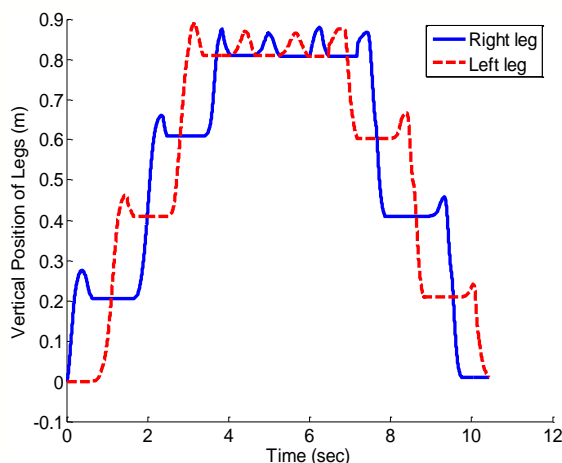
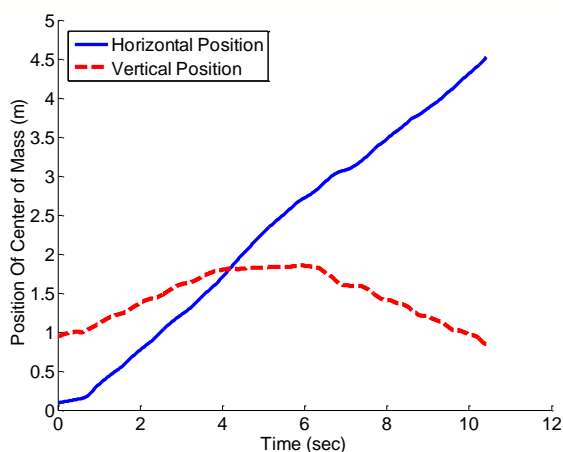
متغیر	محدوده حرکت مفاصل و پارامترهای کنترلی			
	بالارفتن		پایین رفتن	
	حداقل (درجه)	حداکثر (درجه)	حداقل (درجه)	حداکثر (درجه)
$\theta_1$	-۱۰	۳۰	۰	۶۰
$\theta_2$	-۶۰	۰	-۳۰	۰
$\theta_3$	-۳	۳	-۳	۳
$\theta_4$	۰	۸۰	۰	۶۰
$\theta_5$	-۸۰	۰	-۷۰	۰
$T_i$ (Nm)	-۳۵۰	۳۵۰	-۳۵۰	۳۵۰
$\beta$	۴	۱۶	۴	۱۶

جدول ۳. مقدار پارامترهای طراحی

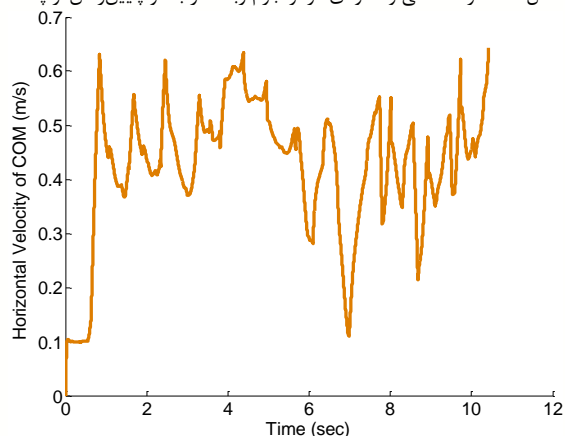
پارامتر	$N_p$	$N_c$	$R$	$Q$	$\Delta t$ (ms)	$\delta$
بالارفتن	۵	۴	۱۰۰	۰٫۰۱	۳۰	۱٫۵
پایین رفتن	۵	۴	۲۰۰	۰٫۰۱	۳۰	۱٫۵

بیانگر پایداری ربات در طول حرکت است. یکی از دلایل اصلی برای داشتن توانایی بالا و پایین رفتن از پله‌هایی با طول و ارتفاع متفاوت، متغیر بودن طول گام‌های ربات است. با توجه به شکل-های ۱۷ و ۱۸ که حرکت افقی و عمودی پای متحرک ربات را نشان می‌دهند، می‌توان متغیر بودن طول گام‌ها را به خوبی مشاهده کرد که کنترل کننده طول گام را با توجه به مشخصات پله و وضعیت ربات تعیین کرده و نیازی به دانستن ابعاد پله‌ها از قبل نیست. فقط باید اطلاعات محیط پیرامون مانند طول و ارتفاع پله‌ها از طریق سنسورهای نصب شده بر روی ربات (از قبیل دوربین یا سنسورهای التراسونیک) در اختیار باشند.

همان‌طور که بیان شد، کنترل پیش‌بین روشی مبتنی بر مدل است. بدین معنا که چنانچه عدم قطعیت در مدل سیستم به وجود آید، کنترل کننده قادر به حفظ عملکرد مطلوب خود نیست. شکل ۱۹ حالتی را نشان می‌دهد که عدم قطعیتی به اندازه ۳۰٪ در جرم بالاتنه ربات به وجود آمده است. در این حالت از مدل دینامیکی گسسته شده ربات برای پیش‌بینی رفتار آینده سیستم استفاده می‌شود.



شکل ۱۱. موقعیت عمودی پای متحرک ربات در بالا و پایین رفتن از پله‌ها

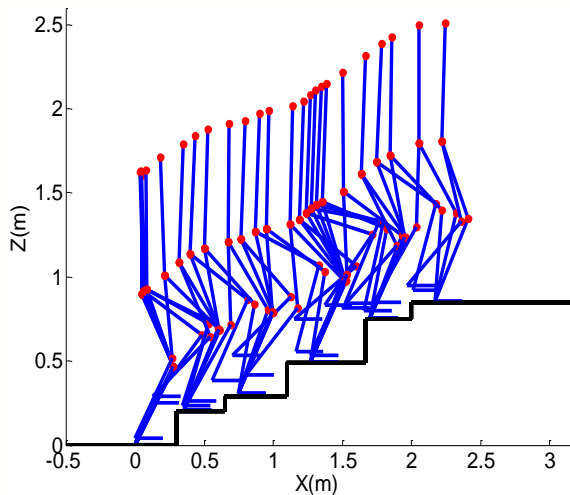


شکل ۱۲. حرکت افقی و عمودی مرکز جرم ربات در بالا و پایین رفتن از پله‌ها

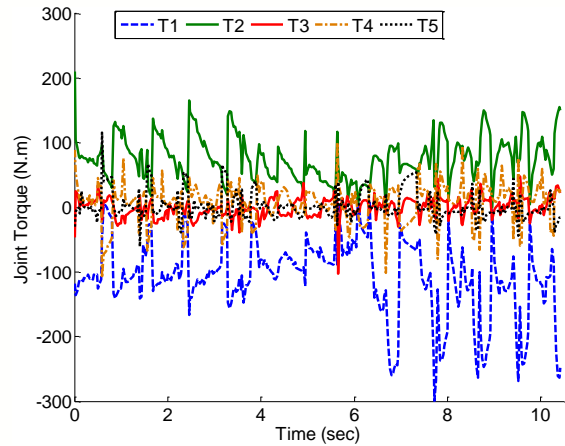
شکل ۱۱. موقعیت عمودی پای متحرک ربات در بالا و پایین رفتن از پله‌ها

با توجه به شکل ۸ ملاحظه می‌شود که ربات می‌تواند به خوبی از پله‌ها بالا و پایین برود. همچنین از شکل ۹ مشاهده می‌شود که ZMP که معیار در نظر گرفته شده برای پایداری سیستم است، همواره داخل ناحیه پایداری قرار داد. در این حالت، کنترل کننده سعی دارد تا ZMP را در وسط کف پا قرار دهد تا سیستم دارای بیش‌ترین حاشیه پایداری باشد. با توجه به این شکل می‌توان مشاهده کرد که کنترل پیش‌بین در این امر موفق عمل کرده است. در شکل‌های ۱۰ تا ۱۲ نیز به ترتیب موقعیت عمودی و افقی پای متحرک و مرکز جرم ربات نشان داده شده است. همچنین سرعت افقی مرکز جرم که می‌توان به عنوان سرعت ربات در نظر گرفت، در شکل ۱۳ نشان داده شده است. با توجه به این شکل می‌توان مشاهده کرد که ربات دارای حرکت دینامیکی است که کنترل کننده این سرعت را با توجه به وضعیت ربات و سایر شرایط محیطی تعیین می‌کند.

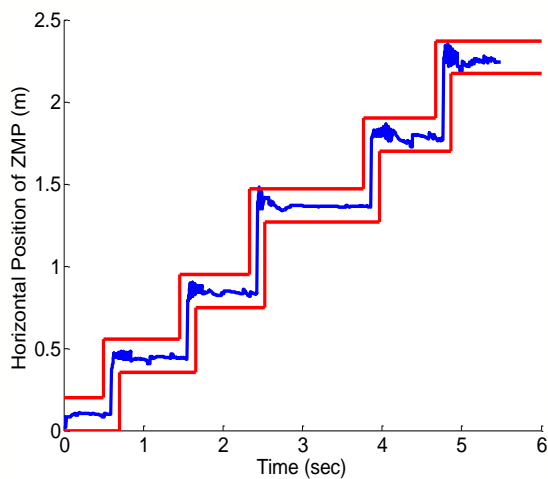
همان‌گونه که قبلاً ذکر شد، یکی دیگر از مزایای استفاده از کنترل کننده پیش‌بین برای کنترل ربات دوپا، توانایی ربات برای بالا و پایین رفتن از پله‌هایی با طول و ارتفاع متفاوت است. در محیط‌های واقعی امکان دارد طول و ارتفاع پله‌ها متغیر باشد. لذا، ربات باید قادر باشد در چنین وضعیتی نیز عملکرد خوبی داشته باشد و بتواند با حفظ پایداری بر روی پله‌ها حرکت کند. در کنترل کننده پیش‌بین به دلیل عدم وجود مسیر از قبل تعیین شده، طول گام‌های ربات نیز به‌طور تطبیقی توسط کنترل کننده تعیین می‌شود. در سایر روش‌های کنترلی که در مراجع آمده‌اند، این امکان برای ربات وجود ندارد به طوری که طول و ارتفاع تمام پله‌ها از ابتدا تا انتهای حرکت بدون تغییر و از قبل دانسته در نظر گرفته می‌شود. در شکل-های ۱۵ تا ۱۸ نتایج شبیه‌سازی روش پیشنهادی در حالتی که طول و ارتفاع پله‌ها متغیر است، ارائه شده است. با توجه به شکل ۱۶ کاملاً مشخص است که ZMP داخل ناحیه پایداری قرار دارد که



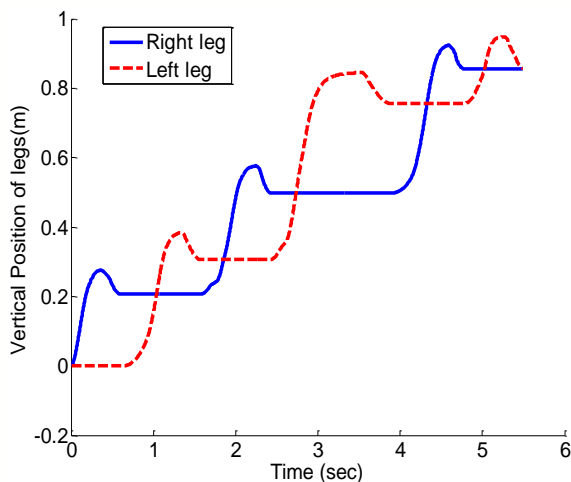
شکل ۱۵. توالی حرکت ربات دوپا در بالا رفتن از پله‌هایی با طول و ارتفاع متفاوت



شکل ۱۴. گشتاور اعمالی به مفاصل ربات در بالا و پایین رفتن از پله‌ها



شکل ۱۶. موقعیت ZMP در بالا رفتن از پله‌هایی با طول و ارتفاع متفاوت

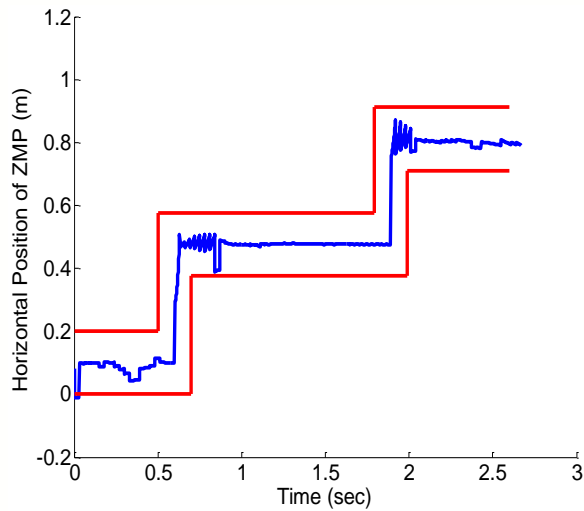


شکل ۱۷. موقعیت عمودی پای متحرک ربات در بالا رفتن از پله‌هایی با طول و ارتفاع متفاوت

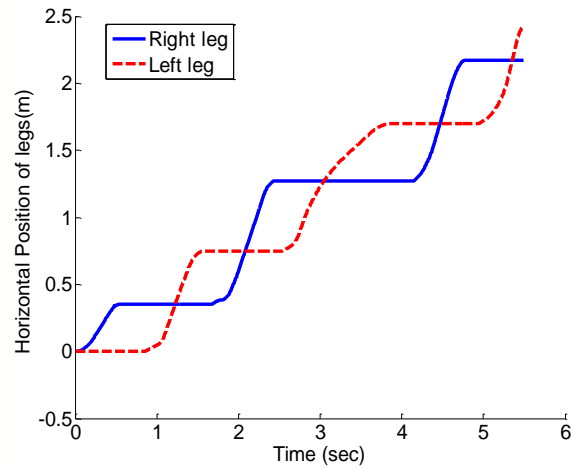
باتوجه به شکل ۱۹ مشاهده می‌شود که وجود عدم قطعیت در مدل باعث ناپایداری و سقوط ربات می‌شود. برای حل این مشکل از شبکه عصبی به‌عنوان مدل پیش‌بینی استفاده می‌شود تا مدل پیش‌بینی بتواند خود را مدل ربات تطبیق دهد. نتایج حاصل از شبیه‌سازی روش پیشنهادی در حضور عدم قطعیت ۳۰٪ در جرم بالاتنه و استفاده از شبکه عصبی RBF به‌عنوان مدل پیش‌بینی در شکل‌های ۲۰ و ۲۱ نشان داده شده‌است.

در ادامه، روش پیشنهادی در این مقاله با روش‌های ارایه‌شده در مقالات مقایسه و نقاط قوت و ضعف این روش بیان می‌شود. در مرجع [۷] برای بررسی حرکت مرکز جرم ربات در حالت یک‌تکیه-گاهی از مدل آونگ وارون استفاده شده است که در آن طول آونگ متغیر و قابل تنظیم است. در این روش، طراحی مسیر و کنترل ربات به‌صورت جداگانه انجام می‌شود. در ابتدا مسیر حرکت ربات به‌صورت قطع‌خط طراحی و سپس از کنترل‌کننده برای ردیابی مسیر طراحی شده استفاده می‌شود. بنابراین، زمان و طول هر گام از قبل تعیین شده و در طول حرکت، ثابت و غیرقابل تنظیم است. در شکل‌های ۲۲ تا ۲۴ نتایج این مرجع آورده شده است.

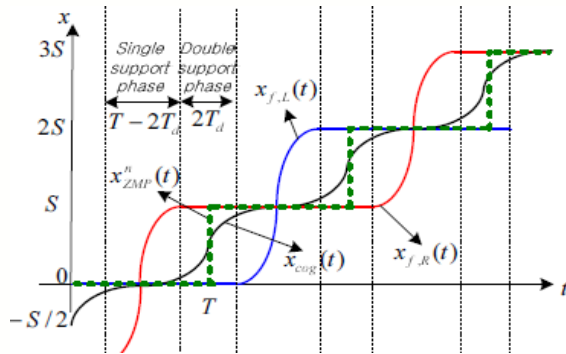
در مرجع [۷] نیز از معیار ZMP برای پایداری ربات استفاده شده-است. لیکن با توجه به شکل ۲۲ می‌توان دریافت که کنترل‌کننده به‌خوبی نتوانسته ZMP را در وسط ناحیه پایداری قرار دهد. در حالی‌که روش پیشنهادی مقاله حاضر، عملکرد بهتری دارد. مهم-ترین برتری روش پیشنهادی نسبت به روش ذکر شده در مرجع [۷] این است که با استفاده از کنترل پیش‌بین طراحی مسیر به-صورت وصل‌خط و همزمان با عمل کنترل انجام می‌گیرد. اما در مرجع [۷] این دو بخش کاملاً از هم جدا بوده و هریک به‌صورت جداگانه انجام می‌شود. بنابراین، در صورت تغییر در محیط یا عوض شدن طول و ارتفاع پله‌ها باید طراحی مسیر مجدد انجام شده و پارامترهای کنترلی دوباره تنظیم گردند.



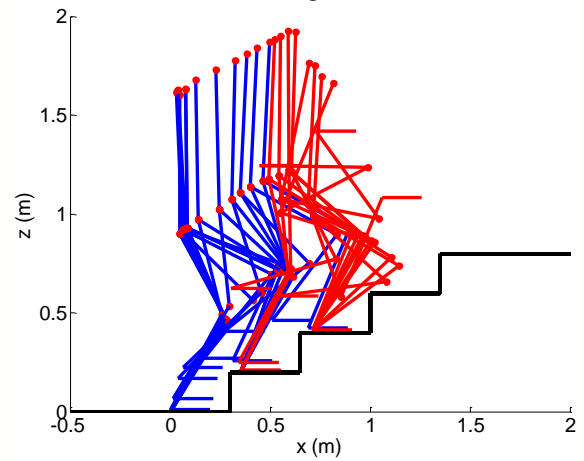
شکل ۲۱. موقعیت افقی ZMP در بالارفتن از پله در حضور عدم قطعیت ۳۰٪ در جرم بالاتنه و استفاده از مدل پیش‌بین عصبی



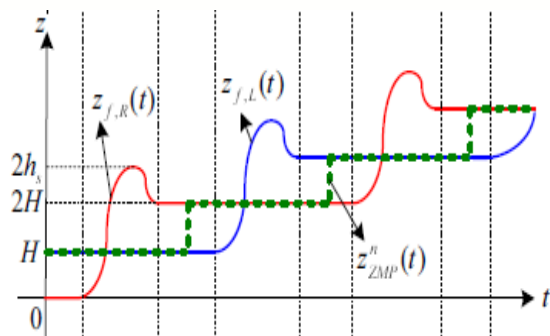
شکل ۱۸. موقعیت افقی پای متحرک ربات در بالارفتن از پله‌هایی با طول و ارتفاع متفاوت



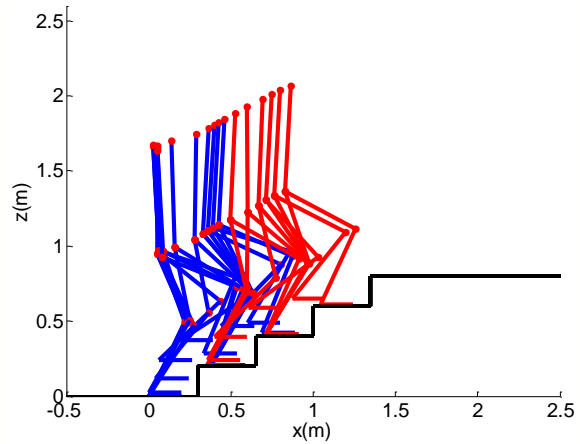
شکل ۲۲. مسیر افقی پای متحرک ربات در مرجع [۷]



شکل ۱۹. ناپایداری ربات در صورت وجود عدم قطعیت ۳۰٪ در جرم بالاتنه بدون استفاده از شبکه عصبی



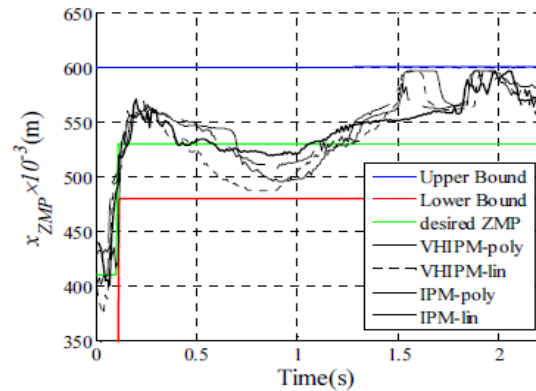
شکل ۲۳. مسیر عمودی پای متحرک در مرجع [۷]



شکل ۲۰. حرکت پایدار ربات در حضور عدم قطعیت ۳۰٪ در جرم بالاتنه و استفاده از مدل پیش‌بین عصبی

## مراجع

- [1] M. Vukobratovic, and D. Juricic, "Contribution to the synthesis of biped gait," IEEE Transactions on Bio-Medical Engineering, vol. 16, issue 1, pp. 1-6, 1969.
- [2] H. Z. Moghadam, M. H. Jaryani, "A supervisory fuzzy-PID controller for a MIMO biped robot balance in frontal plane," International Mechanical Engineering Congress and Exposition, vol. 2, Boston, Massachusetts, USA, pp. 307-314, 2008.
- [3] P. Ananthababu, B. A. Reddy, and K. R. Charan, "Design of fuzzy PI+D and PID controller using Gaussian input fuzzy sets," IEEE International Conference on Emerging Trends in Engineering and Technology, Nagpur, India, pp. 957-961, 2009.
- [4] W. Hui, S. Mei, and W. Zhongyu, "Dynamically adapt to uneven terrain walking control for humanoid robot," Chinese Journal of Mechanical Engineering, vol. 24, pp. 1-9, 2011.
- [5] J. Y. Kim, I. W. Park, and J. H. Oh, "Realization of dynamic stairs climbing for biped humanoid robot using force/torque sensors," Journal of Intelligent and Robotics Systems, vol. 56, issue 4, pp. 389-423, 2009.
- [6] P. R. Vundavili and D. K. Pratihari, "Near-optimal gait generation of a two-legged robot on rough terrains using soft computing," Journal of Robotics and Computer Integrated Manufacturing, vol. 27, issue 3, pp. 521-530, 2011.
- [7] Ch. S. Park, T. Ha, J. Kim, and Ch. H. Choi, "Trajectory generation and control for a biped robot walking upstairs," International Journal of Control, Automation and Systems, vol. 8, issue 2, pp. 339-351, 2010.
- [8] Ch. Fu, and K. Chen, "Gait synthesis and sensory control of stair climbing for a humanoid robot," IEEE Transactions on Industrial Electronics, vol. 55, issue 5, pp. 2111-2120, 2008.
- [9] م. پارسا، کنترل حرکت ربات‌های دوپا با استفاده از رویکرد اغتشاش، پایان‌نامه کارشناسی ارشد، دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، ۱۳۸۹.
- [10] M. Parsa, and M. Farrokhi, "Robust trajectory free model predictive control of biped robots with adaptive gait length," International Journal of Robotics, vol. 2, no. 1, pp. 46-55, 2011.
- [11] H. Diedam, D. Dimitrov, P. B. Wieber, K. Mombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," IEEE International Conference of Intelligent Robots and Systems, Lion, France, pp. 1121-1126, 2008.
- [۱۲] ن. کلامیان، طراحی و تحلیل کنترل‌کننده تطبیقی مدل مینا برای ربات دوپا در حضور موانع، پایان‌نامه کارشناسی ارشد، دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، ۱۳۹۰.
- [13] N. Kalamian, and M. Farrokhi, "Dynamic walking of biped robots with obstacles using predictive controller," International Conference on



شکل ۲۴. موقعیت افقی ZMP با روش ارایه شده در [۷]

## نتیجه‌گیری

در این مقاله، از کنترل پیش‌بین غیرخطی برای کنترل ربات دوپا با پنج درجه آزادی در بالا و پایین رفتن از پله‌ها استفاده شد. در روش پیشنهادی، هیچ مسیر از پیش تعیین‌شده‌ای برای حرکت ربات در نظر گرفته نمی‌شود بلکه حرکت ربات تنها در قالب قیود بیان می‌گردد. این قیود باید به نحوی تعیین شود که ضمن برآورده شدن اهداف مدنظر ربات در طول حرکت، پایداری و تعادل خود را حفظ کند. طراحی مسیر به صورت وصل خط و بی‌درنگ باعث می‌شود که ربات توانایی حرکت در شرایط مختلف را داشته باشد و متناسب با شرایط محیطی حرکت خود را تنظیم کند. یکی دیگر از مزایای کنترل پیش‌بین برای کنترل ربات متغیربودن طول گام‌های ربات است به طوری که ربات می‌تواند از پله‌هایی با عرض و ارتفاع متفاوت نیز بالا و پایین برود که این موضوع در شبیه‌سازی‌های انجام شده نیز نشان داده شد. در تمام کارهای انجام شده قبلی در مقالات در زمینه بالا و پایین رفتن ربات‌های دوپا از پله‌ها طول گام‌ها از ابتدا تا انتهای حرکت ثابت و بدون تغییر در نظر گرفته می‌شود. به این ترتیب، در صورتی که محیط حرکتی ربات تغییر کند، نیاز است تا طراحی مسیر جدیدی متناسب با این محیط انجام شود ولی در روش پیشنهادی نیازی به این کار نیست. همان‌طور که بیان شد باتوجه به وابستگی کنترل پیش‌بین به مدل مورد استفاده برای پیش‌بینی، چنان‌چه عدم قطعیتی در مدل ربات وجود داشته باشد، کنترل پیش‌بین با مشکل جدی مواجه می‌شود؛ به همین دلیل در این مقاله از شبکه‌های عصبی مصنوعی با آموزش وصل خط برای شناسایی مدل ربات استفاده شد تا مدل پیش‌بینی بتواند خود را تغییرات ربات تطبیق دهد.

Manufacturing Engineering University of Manitoba, Canada, 2004.

[15] J. Liu, Radial Basis Function (RBF) Neural Network Control for Mechanical Systems, first edition, Springer Verlag, New York, 2013.

Computer and Knowledge Engineering, pp. 160-165, Mashhad, Iran, 2011.

[14] X. Mu, "Dynamic and motion regulation of a five-link biped robot walking in the sagittal plane," Ph.D. Dissertation, Department of Mechanical and