
برنامه‌ریزی مسیر سه‌بعدی خودرو پرنده در کاربردهای زیست محیطی پیچیده

امیرحسین بدالهی¹، دانشجوی دکتری، جواد سلیمی سرتختی²، دانشیار، سلمان گلی بیدگلی³، استادیار

¹ دانشکده مهندسی برق و کامپیوتر- دانشگاه کاشان - کاشان - ایران - amirhosein972012@yahoo.com

² دانشکده مهندسی برق و کامپیوتر- دانشگاه کاشان - کاشان - ایران - salimi@kashanu.ac.ir

² دانشکده مهندسی برق و کامپیوتر- دانشگاه کاشان - کاشان - ایران - salmangoli@kashanu.ac.ir

چکیده: برنامه‌ریزی مسیر برای چندین خودرو پرنده در محیط‌های پیچیده مانند مناطق کوهستانی، با چالش‌های بسیاری از جمله جلوگیری از برخورد با موانع، محدودیت‌های مانور و تغییرات زاویه‌ای متعدد در یک محیط سه‌بعدی مواجه است. در این مقاله، از الگوریتم جستجوی گنجشک برای یافتن مسیری با کمترین هزینه و مسافت پیمایش شده با در نظر گرفتن محدودیت‌های زاویه چرخش و صعود که نشان‌دهنده محدودیت‌های حرکتی پرنده در دنیای واقعی بوده، استفاده شده است. نتایج شبیه‌سازی نشان می‌دهد که این الگوریتم با حداقل کردن تغییرات زاویه و انحراف، می‌تواند وسایل پرنده را به شکلی کارآمد در مسیر بهینه هدایت کند. آزمایش‌های متعددی بر روی نقشه‌های تصادفی و واقعی انجام شده و نشان داده این است که خودرو پرنده با موفقیت از موانع عبور کرده و مسیر بهینه را طی می‌کنند. به منظور ارزیابی عملکرد، الگوریتم جستجوی گنجشک با الگوریتم‌های پایه‌ای مانند الگوریتم ژنتیک و ازدحام ذرات در یک محیط سه بعدی واقعی مقایسه شده است. نتایج نشان می‌دهد که حداقل طول مسیر برنامه‌ریزی شده با الگوریتم جستجوی گنجشک در مقایسه با الگوریتم ژنتیک، ازدحام ذرات، کلونی زنبورهای مصنوعی و A^* ، 5٪، 3,3٪، 5,6٪ و 9,2٪ کوتاه‌تر شده، متوسط طول مسیرهای برنامه‌ریزی شده نیز به ترتیب 5,1٪، 2,1٪، 5,6٪ و 7٪ کاهش یافته و همچنین انحراف معیار استاندارد طول مسیر بهینه به ترتیب 6,1٪، 5,3٪، 8,6٪ و 13,7٪ کم‌تر شده است. روش پیشنهادی می‌تواند در صنایع مختلف مانند نقشه‌برداری سه‌بعدی و پایش مناطق طبیعی پیچیده مؤثر باشد.

واژه‌های کلیدی: برنامه‌ریزی مسیر، خودرو پرنده، مناطق کوهستانی، محدودیت مانور، الگوریتم جستجوی گنجشک.

3D Path Planning for Aerial Vehicles in Complex Environmental Applications

AmirHossein Yadollahi¹, Ph.d Student, Javad Salimi-Sartakhti², Assistant Professor, Salman Goli-Bidgoli³, Assistant Professor

¹ Electrical and Computer Engineering Department, University of Kashan, Kashan, Iran, amirhosein972012@yahoo.com

² Electrical and Computer Engineering Department, University of Kashan, Kashan, Iran, salimi@kashanu.ac.ir

³ Electrical and Computer Engineering Department, University of Kashan, Kashan, Iran, salmangoli@kashanu.ac.ir

Abstract: Path planning for multiple UAVs in complex environments such as mountainous areas faces many challenges, including obstacle avoidance, maneuvering constraints, and multiple angular changes in a 3D environment. In this paper, the sparrow search algorithm is used to find the path with the least cost and distance traveled, considering the rotation and climb angle constraints that represent the movement constraints of the bird in the real world. Simulation results show that this algorithm can efficiently guide the UAVs along the optimal path by minimizing the angle and yaw changes. Numerous experiments have been conducted on random and real maps and have shown that the UAVs successfully pass the obstacles and follow the optimal path. In order to evaluate the performance, the sparrow search algorithm is compared with basic algorithms such as genetic algorithm and particle swarm optimization in a real 3D environment. The results show that the minimum planned path length with the SSA algorithm is shortened by 5% and 3.3% compared to the genetic algorithm and particle swarm optimization, the average length of the planned paths is also reduced by 5.1% and 2.1%, respectively, and the standard deviation of the optimal path length is reduced by 6.1% and 5.3%, respectively. The proposed method can be effective in various industries such as 3D mapping and monitoring of complex natural areas.

Keywords: Route planning, Aerial Vehicles, mountainous areas, maneuver limit, sparrow search algorithm.

1- مقدمه

کرده است [1]. از مهم‌ترین کاربردهای این فناوری می‌توان به حوزه ربات‌ها، وسایل نقلیه خودران و وسایل پرنده اشاره کرد [2]. مسئله برنامه‌ریزی مسیر به یافتن مسیر بهینه بین مبدا و مقصد گفته می‌شود [3]. یک مسیر بهینه معمولاً جنبه‌هایی نظیر به حداقل رساندن مسافت طی شده، کاهش بزرگی زوایای چرخش و صعود و اجتناب از موانع ثابت و پویا با فاصله مشخص را در نظر می‌گیرد. در برنامه‌ریزی مسیر بسته به ابعاد فضا، تعداد وسایل پرنده، پویایی

در سال‌های اخیر، استفاده از وسایل پرنده به دلیل اندازه کوچک، وزن سبک، سرعت بالا و هزینه کم تجهیزات، در ماموریت‌های نظامی و غیرنظامی گسترش یافته است. بنابراین، چالش‌های مهم این وسایل، مانند تخصیص وظایف، ارتباطات شبکه‌ای و برنامه‌ریزی مسیر توجه محققان زیادی را به خود جلب

- در نظر گرفتن همزمان پارامترهای برخورد بین وسایل پرنده، برخورد با موانع، تغییر زوایای متوالی و غیر واقعی
 - لحاظ کردن محدودیت‌های مانور در برنامه‌ریزی مسیر جمعی با استفاده از الگوریتم جستجوی گنجشک
 - ارائه روش پیشنهادی روی نقشه محیط کوهستانی واقعی.
- جهت درک بهتر مفاهیم مطرح شده در این مقاله، مفاهیم و اصطلاحات پایه در جدول 1 نشان داده شده است.

جدول 1: تعریف اصطلاحات و مفاهیم پایه

عنوان	تعریف
مسئله NP	مجموعه‌ای از مسائل تصمیم‌گیری که در صورت داشتن یک پاسخ ممکن، می‌توان آن را در زمان چندجمله‌ای توسط یک الگوریتم قطعی تأیید کرد.
سلول	هر جزء از فضای حرکت تقسیم شده (محیط دو/سه‌بعدی) به بخش‌های کوچک‌تر و ساده‌تر
سلول ناهموار	سلول‌هایی است که شکل یا اندازه یکنواختی ندارند یا در زوایا و مرزهای نامنظم شکل گرفته‌اند، به طوری که محیط را به صورت نامنظم یا غیریکسان تقسیم کرده‌اند.
تجزیه سلولی	یکی از روش‌های کلاسیک و مؤثر در برنامه‌ریزی مسیر برای ربات‌ها و وسایل نقلیه بدون سرنشین است.
ACD	تجزیه سلولی تطبیقی که در آن، تمام سلول‌های کوچک دارای شکلی از پیش تعیین شده هستند که معمولاً مستطیلی است.
ECD	تجزیه سلولی دقیق که در آن، شکل و اندازه سلول‌ها از پیش تعیین شده نیست و توسط موانع و محیط مشخص می‌شوند و به شکل مستطیل، دوزنقه یا مثلث هستند.
RCD	تجزیه سلولی رستری که اندازه سلول‌ها از پیش تعیین شده و شکل‌های متفاوتی دارند.
DEM	مدل رقومی ارتفاع یا زمین
Scrounger	گنجشک جستجوکننده، در الگوریتم جستجوی گنجشک

محیط و محدودیت‌های حرکتی مختلف مانند حداکثر و حداقل سرعت و زاویه چرخش، یافتن مسیر بهینه می‌تواند به طور قابل توجهی پیچیده شود [4]. مسئله مسیریابی وسایل پرنده یک مسئله NP محسوب شده و راه‌حل‌های ابتکاری¹ و فراابتکاری² مختلفی مانند الگوریتم‌های کلاسیک، هوش مصنوعی و الگوریتم‌های بیونیک ارائه شده است. برای مقابله با افزایش پیچیدگی برنامه‌ریزی مسیر و کاهش زمان محاسبات مورد نیاز در حین عملیات، روش‌های مختلفی در برنامه‌ریزی مسیر به کار گرفته شده‌اند [5].

برنامه‌ریزی مسیر وسایل پرنده در مناطق کوهستانی برای کاربردهایی مانند عکاسی هوایی، نقشه‌برداری و بازرسی بوده که علاوه بر معیار به حداقل رساندن مسافت طی شده، معیار عدم برخورد بین وسایل پرنده، عدم برخورد با موانع و محدودیت‌های حرکتی نیز مستلزم توجه ویژه است. تغییر زوایای افقی و عمودی متعدد و بدون اعمال محدودیت کنترلی، موجب غیرواقعی شدن و ناکارآمدی مسیریابی برای پرنده‌ها می‌شود.

برای پاسخ‌گویی به این چالش‌ها، در این مقاله برای یافتن مسیری با حداقل هزینه مسافت پیمایش شده با محدودیت زوایای چرخش و صعود از الگوریتم جستجوی گنجشک در محیط سه بعدی کوهستانی تصادفی و واقعی استفاده شده است. الگوریتم جستجوی گنجشک در سال 2020 [6] معرفی شده و مطالعات اندکی در این زمینه انجام شده و از نقطه نظرات مختلف مورد توجه محققین قرار گرفته است.

از جمله مهم‌ترین جنبه‌های نوآوری این تحقیق می‌توان به مورد

زیر اشاره کرد:

Approximate Cell Decomposition
Exact Cell Decomposition
Regular Cell Decomposition
Digital Elevation Model
scrounger

Heuristics
Metaheuristics
Nondeterministic Polynomial
Cell
non-uniform cells
Cell Decomposition

الگوریتم ازدحام ذرات²¹، الگوریتم کرم شب تاب²²، الگوریتم جستجوی باکتری‌ها²³، الگوریتم کلونی زنبورهای مصنوعی²⁴ و الگوریتم جستجوی گنجشک²⁵ اشاره کرد [7]. در ادامه ضمن بررسی روش‌های مذکور، مطالعات مرتبط با موضوع مقاله حاضر مورد بررسی قرار گرفته است

2-1- مروری بر مطالعات الگوریتم‌های کلاسیک

الگوریتم‌های کلاسیک که برای کاربرد در برنامه‌ریزی مسیر سراسری بیشترین استفاده را در مطالعات داشته‌اند، در مسائلی با یک محیط قطعی، مناسب‌تر هستند، البته می‌توان آن‌ها را با روش‌های مختلف بهینه‌سازی در محیط‌های نامشخص قابل اجرا کرد، اما برخی از محققان آن را مستعد ایجاد مشکلاتی مانند هزینه محاسباتی بالا و راندمان محاسباتی پایین می‌دانند. بائو و همکاران در سال 2022 الگوریتم A* را با روش میدان پتانسیل مصنوعی برای برنامه‌ریزی مسیر بدون سرنشین بر اساس RCD استفاده کرده‌اند [8].

چالیتا و همکاران در سال 2018 یک الگوریتم یادگیری عمیق را در یک RCD به شکل لانه زنبوری برای برنامه‌ریزی مسیر خودرو پرنده همراه با اجتناب از مانع گنجانده است [9]. در ECD شکل و اندازه سلول‌ها نامشخص است. ژانگ و همکاران در سال 2019 یک الگوریتم تجزیه سلولی دقیق بر اساس ECD پیشنهاد کردند که بر روی محیط استاتیک متمرکز است [10]. در ACD تمام سلول‌های کوچک دارای شکل از پیش تعیین شده، معمولاً مستطیلی هستند. Guruprasad در سال 2021، ACD را در زمینه پوشش کامل مسیرهای ربات متحرک اعمال کرد [11]. به

ساختار این مقاله بدین شرح است که در بخش 2 مروری بر مطالعات گذشته مورد بررسی قرار گرفته است. بخش 3 روش تحقیق شامل مدل‌سازی فضای پرواز، تابع هزینه مدل و محدودیت‌های آن، برنامه‌ریزی مسیر با الگوریتم جستجوی گنجشک و استراتژی هموارسازی مسیر بر اساس منحنی B-SPLINE را ارائه می‌دهد. در بخش 4 نتایج و تحلیل مدل ارائه شده و در بخش 5 نتیجه‌گیری بیان شده است.

2-2- مروری بر کارهای پیشین

مسئله مسیریابی وسایل نقلیه بدون سرنشین یک مسئله بهینه‌سازی ترکیبی و برنامه‌ریزی گسسته است. در تعیین راه حل بهینه یک مسئله NP، روش‌های ابتکاری و فراابتکاری مختلفی توسعه یافته است. از دهه 1950، الگوریتم‌های زیادی توسط محققان برای جستجوی مسیر بهینه یک ربات متحرک در محیط دو بعدی پیشنهاد شده که با توجه به ویژگی‌های الگوریتم‌ها، به سه دسته الگوریتم‌های کلاسیک، الگوریتم‌های هوش مصنوعی و الگوریتم‌های بیونیک طبقه‌بندی می‌شوند.

الگوریتم‌های کلاسیک شامل روش تجزیه سلولی¹²، روش مبتنی بر نمونه‌برداری¹³، الگوریتم جستجوی نمودار¹⁴، میدان پتانسیل مصنوعی¹⁵ و رویکرد پنجره پویا¹⁶ است. الگوریتم‌های هوش مصنوعی عمدتاً شامل الگوریتم‌های منطق فازی¹⁷ و الگوریتم‌های شبکه عصبی مانند الگوریتم شبکه عصبی الهام‌گرفته از زیست¹⁸ هستند. از جمله الگوریتم‌های بیونیک می‌توان به الگوریتم ژنتیک¹⁹، الگوریتم بهینه‌سازی کلونی مورچه‌ها²⁰،

Genetic algorithms (GA)

The Ant Colony Optimization (ACO)

The Particle Swarm Optimization (PSO)

The Firefly Algorithm (FA)

The Bacterial Foraging Optimization Algorithm (BFOA)

The Artificial Bee Colony (ABC)

The Sparrow Search Algorithm (SSA)

Cell Decomposition Path Planning

The Sampling-Based Method (SBM)

The Gravitational Search Algorithm (GSA)

Artificial Potential Field (APF)

The Dynamic Window Approach (DWA)

Fuzzy Logic (FL)

Biologically Inspired Neural Networks (BNN)

زاگردجانین و همکاران در سال 2021، از الگوریتم D*lite در برنامه‌ریزی مسیر سراسری و از الگوریتم منطق فازی در برنامه‌ریزی مسیر محلی استفاده کردند و این روش را در عملیات چند رباتی مشترک در محیط‌های پیچیده معرفی کردند [12].

به طور کلی الگوریتم منطق فازی به ندرت برای برنامه‌ریزی مسیر در محیط‌های سه بعدی استفاده می‌شود زیرا طراحی آن پیچیده‌تر است و به حافظه بزرگتری نیاز دارد. اما الگوریتم منطق فازی اغلب با الگوریتم‌های دیگر ترکیب می‌شود یا برای بهینه‌سازی نقص معین الگوریتم‌های دیگر استفاده می‌شود.

2-3- مروری بر مطالعات الگوریتم‌های بیونیک

بیونیک یک روش تصادفی است که از پدیده هوش ازدحامی بیولوژیکی در طبیعت الهام گرفته شده است [13]. به عنوان مثال می‌توان به الگوریتم ژنتیک، الگوریتم جستجوی فاخته، الگوریتم جستجوی گنجشک، بهینه‌سازی ازدحام ذرات، الگوریتم کلونی زنبورهای مصنوعی و بهینه‌ساز گرگ خاکستری اشاره کرد. لامینی و همکاران در سال 2018 یک الگوریتم برای حل برنامه‌ریزی مسیر توسط الگوریتم ژنتیک در محیط استاتیک پیشنهاد کردند [14]. الگوریتم پیشنهادی از همگرایی زودرس اجتناب می‌کند و مسیر بهتر و امکان پذیرتری را نسبت به نسل قبلی خود ارائه می‌دهد. بنابراین سرعت همگرایی این الگوریتم بیشتر است.

وانگ و همکاران در سال 2021 استفاده از اطلاعات جهت حرکت، نقاط شروع و پایان و زاویه چرخش را جهت تعیین اطلاعات اکتشافی پیشنهاد کردند. بهبود اطلاعات اکتشافی باعث افزایش جهت جستجو و کاهش زاویه چرخش خودرو پرنده می‌شود. در همین حال، MMAS برای محدود کردن غلظت فرمون استفاده شده و از افتادن در بهینه محلی جلوگیری می‌کند [15].

طور خلاصه، این الگوریتم می‌تواند حداقل احتمال برخورد محلی ربات متحرک را تضمین کند.

2-2- مروری بر مطالعات الگوریتم هوش مصنوعی (AI)

هوش مصنوعی نوعی مطالعه رایانه‌ای براساس شبیه‌سازی فرآیند فکر و رفتارهای هوشمندانه انسان‌ها مانند یادگیری، استدلال، تفکر و برنامه‌ریزی است. سه سنگ بنای هوش مصنوعی شامل الگوریتم‌ها، داده‌ها و قابلیت محاسبات می‌باشد. در ادامه دو الگوریتم پرکاربرد هوش مصنوعی برای برنامه‌ریزی مسیر، شامل الگوریتم شبکه عصبی الهام گرفته از زیست و الگوریتم منطق فازی مورد بررسی قرار گرفته است.

در مقایسه با سایر الگوریتم‌های شبکه عصبی، الگوریتم شبکه عصبی الهام گرفته از زیست نیازی به یادگیری ندارند. با این حال، الگوریتم منطق فازی همچنین دارای برخی نقص‌های ذاتی است. به عنوان مثال، قواعد استنتاج فازی باید از قبل بر اساس تجربه متخصصان، که ارتباط نزدیکی با مزایا و معایب مسیر دارد، تدوین شود. مسیر بهینه به قوانین استنتاج فازی بستگی دارد. با افزایش موانع، قوانین استنتاج یا جداول فازی به شدت گسترده می‌شوند، که بر عملکرد زمانی الگوریتم تأثیر می‌گذارد. هنگامی که محیط تغییر می‌کند، قوانین موجود ممکن است برای محیط جدید قابل اجرا نباشند. استراتژی اجتناب از موانع هوشمندانه نیست و نمی‌تواند به طور انعطاف‌پذیر با موانع پویا کنار بیاید. بنابراین بسیاری از محققان برای بهبود قابلیت و دقت الگوریتم منطق فازی، پیشنهاد ترکیب این الگوریتم با سایر الگوریتم‌ها را داده‌اند.

دارای راندمان بالا و پایداری خوب تحت محدودیت‌های دینامیکی شدید است [18].

بهینه‌ساز گرگ خاکستری پیشنهاد شده توسط محققان استرالیایی، یک روش جستجوی بهینه‌سازی هوشمند است. بهینه‌ساز گرگ خاکستری چشم‌انداز کاربردی گسترده‌ای در زمینه برنامه‌ریزی مسیر سیار دارد. اگرچه الگوریتم بهینه‌ساز گرگ خاکستری دارای مزایای کاربردی بسیاری است، اما هنوز دارای معایبی شامل (1) کاهش تنوع جمعیت به دلیل جمعیت‌های اولیه تولید شده به طور تصادفی (2) همگرایی آهسته ناشی از مکانیسم خاص جستجوی الگوریتم بهینه‌ساز گرگ خاکستری و (3) تمایل به سقوط در بهینه محلی است. ژانگ و همکاران در سال 2016 یک بهینه‌ساز گرگ خاکستری فراابتکاری جدید برای برنامه‌ریزی مسیر دوبعدی خودرو پرنده پیشنهاد کردند که می‌تواند با اتصال گره‌های انتخاب شده در یک سیستم مختصات دو بعدی و در عین حال دور ماندن از مناطق تهدید و صرفه‌جویی در سوخت، یک مسیر امن پیدا کند [19].

الگوریتم جستجوی گنجشک یک الگوریتم بهینه‌سازی جدید هوش ازدحامی است که توسط Xue & Shen در سال 2020 [6] بر اساس رفتار گنجشک‌ها در جستجوی غذا و فرار از شکارچیان پیشنهاد شده است. الگوریتم جستجوی گنجشک یک الگوریتم جدید با قابلیت جستجوی بهتر و همگرایی سریع است.

در مطالعه Chen و همکاران در سال 2022 یک روش برنامه‌ریزی مسیر جهانی که الگوریتم A-Star و الگوریتم جستجوی گنجشک را ادغام می‌کند، برای مسئله یافتن کوتاه‌ترین مسیر وسایل نقلیه هدایت‌شونده خودکار، یک محیط شطرنجی ایستا پیشنهاد شده است. مرحله اول از الگوریتم گنجشک برای به

به دلیل سادگی و کارایی، الگوریتم جستجوی باکتری‌ها به طور گسترده در برنامه‌ریزی مسیر ربات‌های متحرک استفاده شده است. مونی و همکاران در سال 2021 از کیفیت جستجوی باکتریایی برای به دست آوردن کوتاه‌ترین مسیر بین دو مکان در کمترین زمان استفاده کرد. تابع هزینه گاوسی اختصاص داده شده به پروفیل‌های جذب‌کننده باکتری و دفع‌کننده نقش عمده‌ای در به دست آوردن بهترین مسیر بین دو مکان دارد [16].

در سال‌های اخیر، تعداد زیادی از محققین عملکرد الگوریتم کلونی زنبورهای مصنوعی را در سه جنبه بهبود بخشیده‌اند: بهبود معادله به‌روزرسانی موقعیت، استراتژی به‌روزرسانی ابعادی و استراتژی جستجوی کلی. متاشا و همکاران در سال 2021 پیشنهاد کردند که یک روش ضد برخورد و برنامه‌ریزی مسیر برای وسایل پرنده ازدحامی می‌تواند توسط الگوریتم کلونی زنبورهای مصنوعی طراحی شود. سیستم پیشنهادی از الگوریتم کلونی زنبورهای مصنوعی برای بهینه‌سازی سرعت وسایل پرنده ای استفاده می‌کند که می‌توانند در کوتاه‌ترین مسیر بدون برخورد با دیگران، به طور موثر به مقصد برسند [17].

یانگ و دب در سال 2009 الگوریتم جستجوی کلاغ²⁷ را پیشنهاد کردند که یک الگوریتم اکتشافی است. این الگوریتم عمدتاً از بازتولید انگل مولدین فاخته و مکانیسم جستجوی پرواز Levy نشأت گرفته است. الگوریتم جستجوی کلاغ دارای مزایای بسیاری از جمله اجتناب از بهینه محلی، پارامترهای کم و ساده و همگرایی سریع است که پس از ارائه، مورد توجه محققان قرار گرفته و با بهبود مستمر و بهینه‌سازی آن، امروزه به یک الگوریتم بهینه‌سازی هوشمند بیونیک فعال تبدیل شده است. ژانگ و همکاران در سال 2021 یک الگوریتم جستجوی فاخته تطبیقی را پیشنهاد کرد که

توانایی اجتناب از مانع محلی استفاده شده است. استفاده از مدل پیشنهادی در یک محیط دو بعدی، در مقایسه با الگوریتم جستجوی گنجشک سنتی، طول مسیر، زمان چرخش مسیر و زمان اجرا به ترتیب $13/42\%$ ، $20/63\%$ و $35/51\%$ کاهش می‌یابد و همواری مسیر تا $29/62\%$ بهبود می‌یابد [21].

Zhang و همکاران در سال 2023، یک الگوریتم جستجوی گنجشک بهبود یافته برای مشکل برنامه‌ریزی مسیر محلی ربات متحرک در یک محیط ناشناخته دو بعدی ارائه داده‌اند. مشکلات همگرایی زودرس و کاهش تنوع جمعیتی الگوریتم جستجوی گنجشک پایه با الهام از الگوریتم هریس حل شده است. در این مدل یک تابع هدف ترکیبی با در نظر گرفتن طول مسیر و ایمنی مسیر، فرموله شده تا ربات به محل مورد نظر حرکت کند. آزمایش‌های شبیه‌سازی نشان می‌دهد که این الگوریتم در حل مشکل برنامه‌ریزی مسیر محلی ربات‌های متحرک مؤثر و قوی بوده است [22].

در مطالعه‌ای دیگر که توسط Wang و همکاران در سال 2023 انجام شده، از یک الگوریتم ترکیبی استفاده شده که در آن بهینه‌سازی ازدحام ذرات و الگوریتم جستجوی گنجشک پیشرفته به صورت موازی کار می‌کنند. در الگوریتم جستجوی گنجشک پیشرفته، جهش تصادفی موقعیت تولیدکننده برای تضمین توانایی جستجوی جهانی تقویت شده و هر گنجشک جستجوکننده از تجربه تولیدکنندگان یاد می‌گیرد. همچنین از استراتژی جستجوی معکوس نخبگان برای به دست آوردن تنوع بهینه استفاده شده است. نتایج نشان می‌دهد که الگوریتم پیشنهادی نسبت به الگوریتم‌های مقایسه شده مسیر عملی‌تر و مؤثرتری را به دست می‌آورد [23].

در مطالعه Zhong و همکاران در سال 2023، به منظور حل مسئله برنامه‌ریزی مسیر وسایل نقلیه بدون سرنشین، یک الگوریتم جستجوی گنجشک بهبود یافته با ترکیب نقشه آشوب مکعبی و

دست آوردن چندین نقطه کلیدی از شبکه در نقشه شطرنجی استفاده می‌کند و سپس از الگوریتم A-Star برای اتصال این نقاط استفاده می‌کند. مرحله دوم از روش پرتو برای حذف گره‌های اضافی استفاده شده و سپس از منحنی بسل برای ایجاد کوتاه‌ترین مسیر پیوسته، بدون برخورد و هموار پس از به دست آوردن گره‌های حیاتی ساده شده استفاده می‌کند. نتایج تجربی نشان می‌دهد که طول مسیر برنامه‌ریزی شده با روش A-SSA کوتاه‌تر بوده که نشان‌دهنده اثربخشی روش بوده است [19].

Zhang و همکاران در سال 2022، یک رویکرد برنامه‌ریزی مسیر دو بعدی برای ربات‌های متحرک با رویکرد مبتنی بر الگوریتم جستجوی گنجشک پیشنهاد داده‌اند. در این الگوریتم برای به دست آوردن مسیرهای با کیفیت بالا و همگرایی سریع، یک استراتژی مسیر خطی و یک استراتژی جستجوی محلی برای بهبود تابع ارزش گنجشک در موقعیت جهانی و یک تابع به‌روزرسانی موقعیت جدید برای سرعت بخشیدن به همگرایی استفاده می‌شود. نتایج تجربی نشان می‌دهد که الگوریتم پیشنهادی، مسیر کوتاه‌تر و همگرایی سریع‌تری نسبت به سایر مطالعات پیشرفته دارد [20].

در مطالعه‌ای دیگر Liu و همکاران در سال 2023 با هدف برطرف کردن کاستی‌های الگوریتم جستجوی گنجشک در برنامه‌ریزی مسیر، مانند ناتوانی آن در اجتناب از موانع دینامیکی، یک روش جدید پیشنهاد می‌کنند. در این مدل در مرحله اول، یادگیری معکوس کوشی برای مقداردهی اولیه جمعیت گنجشک برای جلوگیری از همگرایی زودرس الگوریتم استفاده شده است. در مرحله دوم، الگوریتم سینوسی-کسینوسی برای به‌روزرسانی موقعیت جمعیت تولیدکنندگان و تعادل بین جستجوی جهانی و قابلیت‌های اکتشاف محلی الگوریتم استفاده شده است. سپس، یک استراتژی پرواز برای به روز رسانی موقعیت گنجشک‌ها برای جلوگیری از بهینه محلی استفاده می‌شود. در نهایت، الگوریتم جستجوی گنجشک بهبود یافته و رویکرد پنجره پویا برای افزایش

Liu و همکاران در سال 2024 یک الگوریتم جستجوی

گنجشک اصلاح شده به نام CASSA برای مقابله با چالش مسیریابی سه بعدی ارائه داده‌اند. در این مدل از استراتژی آشفته برای افزایش تنوع جمعیت الگوریتم و از وزن اینرسی تطبیقی برای متعادل کردن نرخ همگرایی و قابلیت‌های اکتشاف الگوریتم استفاده شده است. همچنین در این الگوریتم استراتژی جهش کوشی-گوسی برای افزایش توانایی الگوریتم برای رهایی از بهینه محلی اتخاذ شده است. نتایج شبیه‌سازی نشان می‌دهد که از نظر بهینه بودن طول پرواز و همگرایی مدل، مسیرهای تولید شده توسط CASSA به الگوریتم‌های بهینه‌سازی ازدحام ذرات، کلونی زنبورهای مصنوعی و الگوریتم بهینه‌سازی نهنگ در محیط یکسان ترجیح داده می‌شوند [26].

Fan و همکاران در سال 2024 برای پرداختن به چالش

برنامه‌ریزی مسیر، یک الگوریتم جستجوی گنجشک بهبود یافته چند استراتژی با عملگر آشوب (CMISSA) پیشنهاد کرده‌اند. در این مدل از عملگر آشوب و یادگیری معکوس برای مقداردهی اولیه الگوریتم جستجوی گنجشک به منظور افزایش یکنواختی و اثربخشی توزیع اولیه جمعیت استفاده شده و یک استراتژی کنترل مرز تطبیقی جدید برای رسیدگی به مکان گنجشک‌هایی که از مرز عبور کرده‌اند معرفی می‌شود. نتایج مقایسه CMISSA با 5 الگوریتم در 5 محیط مختلف نشان می‌دهد که میانگین نرخ کوتاه شدن مسیر CMISSA بین 20 تا 50 درصد بوده است [27].

مطالعه انجام شده توسط Guo و همکاران در سال 2024 بر

روی بهینه‌سازی مسیر دو بعدی وسایل نقلیه هوایی بدون سرنشین تمرکز دارد. با پرداختن به مسائلی مانند کاهش تنوع جمعیت و حساسیت به بهینه محلی در تکرارهای بعدی الگوریتم جستجوی گنجشک، یک الگوریتم بهبود یافته ترکیبی با چند استراتژی پیشنهاد

استراتژی پیاده‌روی تصادفی گاوسی پیشنهاد داده‌اند. در مرحله اول، نقشه آشوب مکعبی برای جایگزینی روش تولید تصادفی الگوریتم جستجوی سنتی برای بهینه‌سازی توزیع اولیه ناهموار جمعیت و بهبود توانایی جستجوی سراسری استفاده می‌شود. در مرحله دوم، در تکرار دیر هنگام الگوریتم، انحراف استاندارد برازش معرفی می‌شود تا از استراتژی پیاده‌روی تصادفی گاوسی برای کمک به الگوریتم برای خروج از بهینه محلی استفاده شود. نتایج تجربی نشان می‌دهد که در مقایسه با الگوریتم جستجوی گنجشک سنتی، میانگین تابع ارزش مسیر برنامه‌ریزی شده توسط الگوریتم جستجوی گنجشک بهبود یافته 14/8 درصد و واریانس آن 49/9 درصد کاهش می‌یابد. مسیر برنامه‌ریزی شده توسط الگوریتم کیفیت خوب و پایداری بالایی دارد می‌تواند مسیر مطمئنی را برای وسایل نقلیه بدون سرنشین فراهم کند [24].

Xu و همکاران در سال 2024 برای پرداختن به مشکلات تنوع

جمعیت و تمایل به همگرایی زودرس به بهینه محلی در الگوریتم جستجوی گنجشک، یک الگوریتم جستجوی گنجشک بهبود یافته بر اساس بهینه‌سازی مشارکتی چند استراتژی برای مسیریابی دو بعدی پیشنهاد دادند. این الگوریتم از چند استراتژی برای تولید جمعیت اولیه گنجشک جهت غنی‌سازی تنوع جمعیت، تقویت توانایی الگوریتم برای پرش از بهینه محلی و تعادل بین بهینه سراسری و محلی استفاده می‌کند. با مقایسه با الگوریتم‌هایی مانند الگوریتم ازدحام ذرات، بهینه‌ساز گرگ خاکستری و الگوریتم بهینه‌ساز وال²⁸، مزایای مدل پیشنهادی در سرعت هم‌گرایی و دقت بهینه‌سازی تایید شده و در کاربرد برنامه‌ریزی مسیر ربات، طول مسیر کوتاه‌تر، چرخش‌های کمتر و راندمان برنامه‌ریزی بالاتر را در برنامه‌ریزی مسیر نشان می‌دهد [25].

دایره‌ای برای افزایش تنوع، مقداردی اولیه می‌شود. سپس، از روش به‌روزرسانی مکان "پرنده شکاری"³⁰ برای جایگزینی فرمول به‌روزرسانی مکان الگوریتم جستجوی گنجشک استفاده شده تا وسعت جستجوی مدل را در فضای شبیه‌سازی بهبود بخشد و کارایی حل مسئله را بهینه کند. در ادامه از یک الگوریتم استراتژی پرواز برای بهبود توانایی بهینه‌سازی سراسری استفاده شده، به طوری که گنجشک در مرحله بعدی تکرار از حالت بهینه محلی

خارج می‌شود. در نهایت، استراتژی جهش با توزیع T توانایی اکتشاف محلی را در تکرارهای دیر هنگام³¹ (تکرارهای بیش از حد انتظار یا برنامه‌ریزی شده) افزایش داده، تا سرعت همگرایی الگوریتم جستجوی گنجشک بهبود یابد. مدل پیشنهادی در مسئله برنامه‌ریزی مسیر ربات‌های متحرک دو بعدی پیاده‌سازی شده و نتایج نشان می‌دهد که الگوریتم پیشنهادی از نظر طول مسیر، زمان اجرا و پایداری نسبت به الگوریتم جستجوی گنجشک برتری دارد [30].

در مطالعه‌ای توسط Ouyang و همکاران در سال 2024 با هدف برنامه‌ریزی مسیر خودرو پرنده، یک الگوریتم جستجوی گنجشک بهبود یافته با استراتژی دوگانه پیشنهاد شده است. در این مقاله از نگاشت دایره و استراتژی‌های یادگیری بازتابی برای بهبود الگوریتم جستجوی گنجشک استفاده شده است. نتایج تجربی نشان می‌دهد که الگوریتم پیشنهادی می‌تواند با سرعت بیشتری یک مسیر امن و عملی ایجاد کند [31].

Yang و همکاران در سال 2024 یک الگوریتم جستجوی گنجشک پیشرفته متوازن چند مکانیزمی را برای حل مسئله مسیریابی پیشنهاد داده‌اند. الگوریتم پیشنهادی قابلیت‌های یادگیری تکاملی الگوریتم جستجوی گنجشک را افزایش داده و به الگوریتم

شده است. مدل پیشنهادی ضمن اجتناب از موانع توانایی بهینه‌سازی سراسری الگوریتم را تقویت کرده و از به دام افتادن در بهینه محلی جلوگیری می‌کند. آزمایش‌های شبیه‌سازی نشان می‌دهند که الگوریتم جستجوی گنجشک بهبود یافته از الگوریتم‌هایی از جمله الگوریتم ژنتیک، الگوریتم ازدحام ذرات، الگوریتم تکاملی تفاضلی²⁹ و بهینه‌ساز گرگ خاکستری بهتر عمل می‌کند [28].

در مطالعه Hou و همکاران در سال 2024، برای غلبه بر محدودیت‌های الگوریتم جستجوی گنجشک و چالش‌های اجتناب از مانع پویا در ربات‌های متحرک، یک روش ترکیبی از الگوریتم جستجوی گنجشک پیشرفته با رویکرد پنجره پویا معرفی شده است. در این مدل با ارایه روشی برای مقداردی اولیه جمعیت گنجشک، توزیع یکنواختی از جمعیت گنجشک به دست آمده و به طور همزمان قابلیت اکتشاف الگوریتم افزایش یافته است. علاوه بر این، استراتژی تنظیم خود تطبیقی پویا برای به‌روزرسانی موقعیت با هدف افزایش قابلیت بهینه‌سازی الگوریتم با اصلاح فرمول‌های به‌روزرسانی موقعیت هم برای تولیدکنندگان و هم برای جستجوکنندگان اتخاذ شده است. هم نتایج شبیه‌سازی و هم نتایج تجربی، برتری الگوریتم جستجوی گنجشک پیشرفته را در مقایسه با سایر الگوریتم‌ها از نظر طول مسیر و زمان اجرای الگوریتم نشان می‌دهند. قابل ذکر است که در مقایسه با الگوریتم جستجوی گنجشک پایه، میانگین طول مسیر به میزان 15,31% کاهش یافته است [29].

Xu و همکاران در سال 2024، یک الگوریتم جستجوی گنجشک بهبود یافته را با یک استراتژی ترکیبی معرفی می‌کند. در این الگوریتم ابتدا، جمعیت گنجشک با استفاده از نگاشت آشسته

در خروج از بهینه محلی کمک می‌کند. نتایج نشان می‌دهد که این الگوریتم در مقایسه با نوع جستجوی گنجشک اولیه، عملکرد بهینه‌سازی پایدارتر و دقیق‌تری را برای تأیید عملی بودن الگوریتم بهبود یافته نشان می‌دهد [32].

2-4- جمع‌بندی مروری بر مطالعات

به طور کلی، بررسی مروری بر مطالعات نشان می‌دهد که در الگوریتم‌های کلاسیک، چالش‌هایی مانند سلول‌های ناهموار، تقسیم سلولی دقیق، تعداد موانع زیاد و نرسیدن به نقطه هدف در مدت زمان محدود می‌تواند منجر به افزایش زمان محاسبات، کاهش کارایی مدل، همگرایی کند و یا شکست الگوریتم شود. در الگوریتم‌های بیونیک هم‌گرایی آهسته، قرار گرفتن در بهینه محلی و هم‌گرایی زودرس جزو مهم‌ترین دغدغه‌های محققین بوده که با روش‌ها و راهکارهای متعددی سعی در بهبود آن داشته‌اند. همچنین در الگوریتم‌های مبتنی بر بیونیک، مطالعات کمتری برای برنامه ریزی مسیر "محلی" نسبت به برنامه‌ریزی مسیر "سراسری" وجود دارد. این در حالی است که از الگوریتم‌های هوش مصنوعی به دلیل قابلیت اکتشاف محیط بالاتر و قابلیت عملیات انعطاف پذیرتر، بیشتر در برنامه‌ریزی مسیر "محلی" استفاده می‌شود؛ زیرا برای یادگیری با هوش بالاتر و مدیریت بهتر عدم قطعیت در محیط، آموزش دیده‌اند.

از آنجا که هدف مقاله حاضر برنامه‌ریزی مسیر "سراسری" سه بعدی وسایل پرنده در یک منطقه کوهستانی بوده و با چالش‌های 1- به حداقل رساندن مسافت طی شده، 2- معیار عدم برخورد بین وسایل پرنده، 3- عدم برخورد با موانع و محدودیت‌های حرکتی به منظور امکان‌سنجی حرکت خودرو پرنده و 4- کارایی ضعیف برخی الگوریتم‌ها در محیط سه بعدی مواجه است. با توجه به نتایج مروری بر مطالعات بهتر است از الگوریتم‌های بیونیک استفاده شود زیرا از الگوریتم‌های هوش مصنوعی به دلیل قابلیت اکتشاف محیط

بالاتر و قابلیت عملیات انعطاف پذیرتر، بیشتر در برنامه‌ریزی مسیر "محلی" استفاده می‌شود.

بررسی کاربردهای متعدد الگوریتم‌های جدید در شرایط مختلف می‌تواند موجب ارزیابی اعتبار آن‌ها شود. لذا در این تحقیق برای پاسخ‌گویی به مسئله طرح شده و چالش‌های آن، از الگوریتم جدید جستجوی گنجشک [8]، برای برنامه‌ریزی مسیر استفاده شده است. بررسی کاربردهای مدیریت تغییر زوایای متوالی و غیرواقعی، محدودیت‌های مانور در برنامه‌ریزی مسیر جمعی و توانایی مسیریابی در محیط کوهستانی واقعی مهم‌ترین کاربردهایی هستند که در این تحقیق برای ارزیابی توانایی الگوریتم جستجوی گنجشک در مسیریابی مورد بررسی قرار گرفته است.

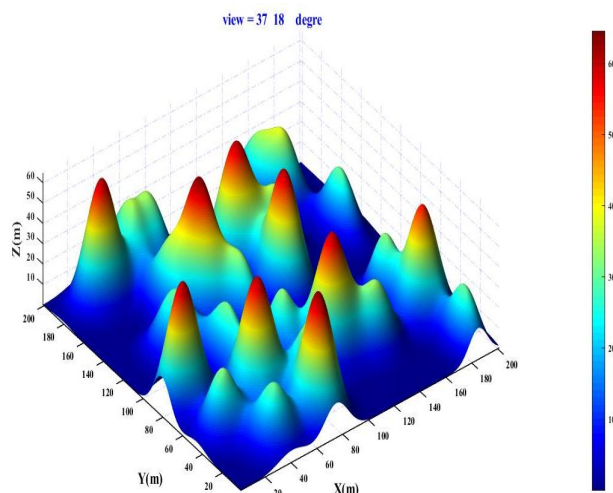
بنابراین هدف اصلی این مقاله، حل مسئله برنامه‌ریزی مسیر سه‌بعدی برای وسایل پرنده در یک منطقه کوهستانی با محدودیت‌های عدم برخورد با موانع، محدودیت فاصله خودرو پرنده مجاور، محدودیت طول کل پرواز و محدودیت زوایای تغییر جهت شامل زاویه چرخش و زاویه صعود با استفاده از الگوریتم جستجوی گنجشک می‌باشد.

3- روش تحقیق

در این تحقیق، از الگوریتم جستجوی گنجشک برای برنامه‌ریزی مسیر چند خودرو پرنده استفاده شده است. در مراحل شبیه‌سازی از الگوریتم جستجوی گنجشک برای یافتن مجموعه‌ای از نقاط بهینه از مبدا تا مقصد روی سطح امن استفاده می‌شود و سپس منحنی‌های B-spline برای تنظیم کردن گره‌های کلیدی برای به دست آوردن مسیر پیش‌فرض استفاده شده است. این بخش از روش تحقیق مطابق با روش ارائه شده در مقاله [33] می‌باشد.

برای اجرای مدل ابتدا نحوه مدل‌سازی فضای پرواز به صورت کوهستانی و ایجاد سطح امن در محیط مدل‌سازی شده تشریح گردیده است. برای رسیدن به هدف به صورت ایمن، خودرو پرنده نیاز دارد تا در شرایط محدودیت طول کلی پرواز، محدودیت

به طور تصادفی در یک فضای سه بعدی به ابعاد $200 \times 200 \times 60$ متر، همانطور که در شکل 1 نشان داده شده است، ایجاد شد.



شکل 1. بستر اول: مدل ارتفاعی دیجیتال با استفاده از تابع نمایی

مدل رقومی ارتفاع (DEM) یا همان مدل رقومی زمین که معمولاً به خودرو پرنده یا ماهواره عکس‌برداری می‌شود را می‌توان یک نقشه رقومی دانست که حاوی اطلاعات ارتفاعی تمامی نقاط یک محدوده بوده و دارای کاربردهای مختلفی در علوم مرتبط با داده‌های مکانی است. در این مقاله در بستر دوم، یک نقشه DEM واقعی برای محیط شبیه‌سازی در نظر گرفته شده است که دامنه ارتفاعی آن در محدوده 0 تا 10 نرمال شده است (شکل 2).

فاصله تا مانع، محدودیت محیط کوهستانی، محدودیت فاصله وسایل پرنده مجاور و محدودیت‌های زاویه چرخش و صعود که نشان‌دهنده محدودیت‌های تغییر جهت و میزان مانور پرنده می‌باشد، پرواز بدون تصادف را داشته باشد. توجه به محدودیت‌های تغییر جهت و میزان مانور پرنده که لازمه کاربرد عملی الگوریتم در دنیای واقعی می‌باشد، مهم‌ترین وجه تمایز این مقاله با مقاله پایه [33] می‌باشد.

پس از مدل‌سازی فضای پرواز الگوریتم جستجوی گنجشک برای برنامه‌ریزی مسیر و استراتژی هموارسازی مسیر بر اساس منحنی B-spline تشریح شده است.

1-3- مدل‌سازی فضای پرواز

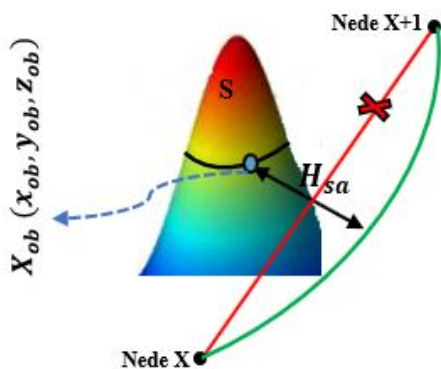
در برنامه‌ریزی مسیر باید فضای برنامه‌ریزی متناسب با محیط پرواز و الزامات مأموریت ایجاد شود. در این تحقیق دو نوع بستر شبیه‌سازی برای مسیربایی وسایل پرنده در نظر گرفته شده است. در بستر اول یک مدل ارتفاعی دیجیتال با استفاده از یک تابع تصادفی برای شبیه‌سازی قله‌ها و سایر موانع تهدید ایجاد می‌شود. این تابع به صورت رابطه (1) بیان می‌شود [34].

$$Z(x, y) = \sum_{i=1}^N h_i \exp \left[\left(\frac{x - x_{ic}}{x_{si}} \right)^2 - \left(\frac{y - y_{ic}}{y_{si}} \right)^2 \right] \quad (1)$$

که در آن x و y مختصات نقاط در یک صفحه افقی هستند و Z به مختصات ارتفاعی آن‌ها اشاره دارد. N تعداد پیک‌های تولید شده، (x_{ic}, y_{ic}) مختصات مرکز قله‌ها و h_i پارامتر کنترل ارتفاع قله را نشان می‌دهد. x_{si} نشان دهنده نرخ تغییر قله در امتداد محور x و y_{si} نشان دهنده نرخ تغییر قله در امتداد محور y است. این معادله باعث می‌شود یک ناحیه شبه کوهستانی در محدوده موردنظر ایجاد شود. لذا در این تحقیق بر اساس رابطه (1)، 28 قله

موانع موجود در مناطق کوهستانی که عمدتاً موانع ثابت شامل قله‌ها می‌باشد، به دلیل تعدد، نوسانات متعدد و اختلاف ارتفاع متغیر مهم‌ترین چالش مسیریابی وسایل پرنده در مناطق کوهستانی می‌باشد. بنابراین همانطور که در شکل 3 نشان داده شده است، موانع با پوسته S با مختصات نقطه به نقطه $X_{ob}(x_{ob}, y_{ob}, z_{ob})$ و فاصله ایمن H_{sa} برای حرکت خودرو پرنده ایجاد ممنوعیت می‌کند. لذا برای حرکت خودرو پرنده از گره کلیدی x به گره کلیدی $x+1$ که در محدوده مانع S قرار دارد، باید الزامات رابطه (3) برآورده شود [33].

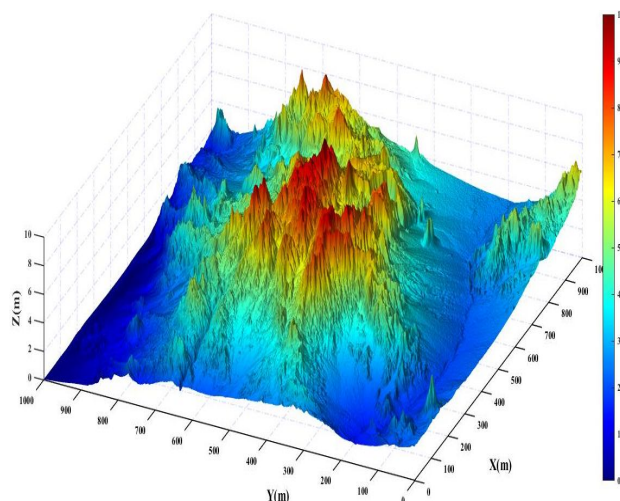
$$H_{sa} < \sqrt{(x_t - x_{ob})^2 + (y_t - y_{ob})^2 + (z_t - z_{ob})^2} \quad (3)$$



شکل 3: دیاگرام محدودیت برخورد با موانع

3-2-2- محدودیت فاصله خودرو پرنده مجاور

همانطور که در شکل 4 نشان داده شده است، چند خودرو پرنده باید در حین پرواز فاصله مناسب از یکدیگر را حفظ کنند. با وجود اینکه نقاط شروع پرواز و زمان پرواز وسایل پرنده مختلف است ولی عواملی مانند تغییر سرعت پرواز و لزوم تغییر جهت هر خودرو پرنده به دلایلی مانند عبور از موانع ممکن است باعث برخورد دو خودرو پرنده شود. بنابراین، لازم است که در فرآیند مدل‌سازی فاصله حداقلی را برای خودروهای پرنده تعیین کرد. بنابراین، یک کره با مرکزیت خودرو پرنده با شعاع حداقل R_2 به عنوان محدودیت فاصله خودرو پرنده مجاور در نظر گرفته می‌شود.



شکل 2. بستر دوم: مدل ارتفاعی DEM(Terrain) واقعی با دامنه ارتفاعی نرمال شده در محدوده 0 تا 10

در فضای کوهستانی واقعی، به دلیل پیچیدگی محیط زمینی ناشی از دره‌های عمیق و کوهستان‌های شیب‌دار، برنامه‌ریزی مسیر نیازمند اجرای پرواز بالاتری برای خودرو پرنده است که منجر به برخورد نشود. لذا هنگام پرواز، خودرو پرنده نیاز دارد که فاصله مشخصی از کوهستان را حفظ کند تا پروازی ایمن داشته باشد. برنامه‌ریزی مسیر بر روی سطح ایمن می‌تواند احتمال برخورد خودرو پرنده با کوهستان را کاهش داده و امنیت پرواز خودرو پرنده را افزایش دهد. بنابراین در این مدل نیز به مختصات ارتفاعی ایجاد شده در رابطه (1)، ارتفاع h افزوده شده تا یک سطح ایمن $S(x, y)$ تولید شود (رابطه (2)). بنابراین حداقل ارتفاع خودرو پرنده نسبت به سطح زمین است [33].

$$S(x, y) = Z(x, y) + h \quad (2)$$

3-2-3- تابع هزینه مدل

تابع هزینه مدل تحقیق بر اساس چهار محدودیت شامل محدودیت اجتناب از برخورد با موانع، محدودیت فاصله خودرو پرنده مجاور، محدودیت طول کل پرواز و محدودیت زوایای تغییر جهت طراحی شده است.

3-2-3-1- محدودیت برخورد با موانع

مسیر حرکت خودرو پرنده در صفحه XY ($\overrightarrow{P'_{i,j}P'_{i,j+1}}$) از رابطه (6) محاسبه می‌شود.

$$\overrightarrow{P'_{i,j}P'_{i,j+1}} = \vec{k} \times (\overrightarrow{P_{i,j}P_{i,j+1}} \times \vec{k}) \quad (6)$$

بنابراین، زاویه چرخش $\alpha_{i,j}$ را می‌توان با رابطه (7) محاسبه

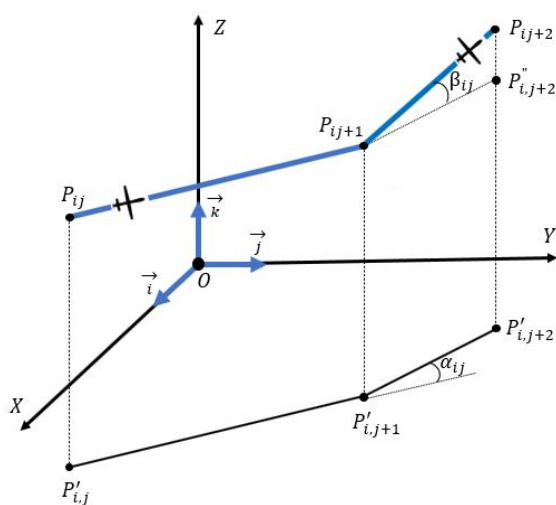
کرد.

$$\alpha_{i,j} \leq \arctan \left(\frac{\|\overrightarrow{P'_{i,j}P'_{i,j+1}} \times \overrightarrow{P'_{i,j+1}P'_{i,j+2}}\|}{\|\overrightarrow{P'_{i,j}P'_{i,j+1}}\| \cdot \|\overrightarrow{P'_{i,j+1}P'_{i,j+2}}\|} \right) \quad (7)$$

زاویه صعود $\beta_{i,j}$ زاویه بین مسیر $\overrightarrow{P_{i,j+1}P_{i,j+2}}$ و تصویر آن

روی صفحه افقی ($\overrightarrow{P'_{i,j}P'_{i,j+2}}$) است که با استفاده از رابطه (8) محاسبه می‌شود.

$$\beta_{i,j} \leq \arctan \left(\frac{Z_{i,j+1} - Z_{i,j}}{\|\overrightarrow{P'_{i,j}P'_{i,j+1}}\|} \right) \quad (8)$$



شکل 5. محاسبه زاویه چرخش و زاویه صعود

در نهایت این چهار محدودیت بیان شده، تابع هزینه کل F_{cost} را مطابق رابطه (9) تشکیل می‌دهند.

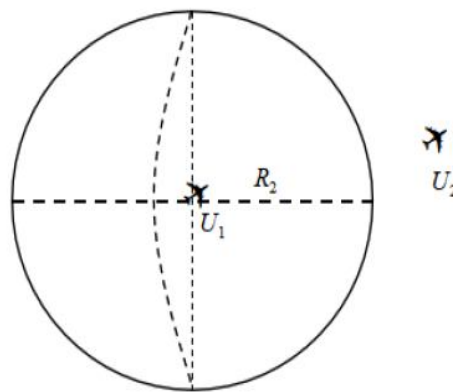
$$F_{cost} = \prod_{i=1}^4 f_i \cdot \omega_i \quad (9)$$

که در آن، f_i محدودیت‌ها و ω_i وزن هر محدودیت می‌باشد.

3-3- برنامه‌ریزی مسیر با الگوریتم جستجوی گنجشک

لذا فاصله بین دو خودرو پرنده مجاور $U_1(x_1, y_1, z_1)$ و $U_2(x_2, y_2, z_2)$ باید الزامات رابطه (4) را برآورده کند [33].

$$R_2 < \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (4)$$



شکل 4: دیاگرام محدودیت فاصله خودرو پرنده مجاور [33]

3-2-3 محدودیت طول کل پرواز

در پرواز خودرو پرنده در یک محیط واقعی، به دلیل محدودیت مقدار انرژی خودرو پرنده، محدودیت مقدار مسافتی که خودرو پرنده می‌تواند طی کند وجود دارد. بنابراین محدودیت طول کل پرواز l_{man} برای هر خودرو پرنده می‌بایست مورد توجه قرار گیرد. لذا جمع طول مسیر پرواز l_i هر خودرو پرنده باید الزامات رابطه (5) را برآورده کند [33].

$$l_{max} > \sum_{i=0}^{node} l_i \quad (5)$$

4-2-3 محدودیت زوایای تغییر جهت

برای پرواز طبیعی خودرو پرنده می‌بایست از تغییر جهت‌های غیر طبیعی خودرو پرنده جلوگیری کرد. بنابراین در این تحقیق محدودیتی برای تغییر زاویه افقی و تغییر زاویه صعود در نظر گرفته شده است. مطابق شکل 5 زاویه چرخش $\alpha_{i,j}$ زاویه بین دو تغییر مسیر متوالی $\overrightarrow{P'_{i,j}P'_{i,j+1}}$ و $\overrightarrow{P'_{i,j+1}P'_{i,j+2}}$ در صفحه XY است. اگر بردار اصلی مسیر حرکت خودرو پرنده در صفحه XYZ بوده و $\vec{k}$ بردار واحد در جهت محور Z باشد، آنگاه تصویر بردار

$$x_{i,d}^{t+1} = \begin{cases} x_{i,d}^t \cdot \exp\left(\frac{-i}{\alpha \cdot iter_{max}}\right), & R_2 < ST \\ x_{i,d}^t + Q \cdot L, & R_2 \geq ST \end{cases} \quad (10)$$

که در آن d نشان دهنده بعد مسئله، $x_{i,d}^t$ موقعیت مکانی گنجشک i در بعد d و در تکرار t ام، $\alpha \in (0,1]$ یک عدد تصادفی یکنواخت، $R_2 \in (0,1]$ نشان دهنده مقدار هشدار (خطر)، $ST \in [0.5, 1.0]$ نشان دهنده آستانه هشدار (خطر)، Q یک عدد تصادفی معمولی توزیع شده، L نشان دهنده یک ماتریس $1 \times d$ و $iter_{max}$ بیانگر حداکثر تعداد تکرار است.

3-3-2- به روزرسانی موقعیت جستجوکننده‌ها

در فرآیند جستجوی غذا، گنجشک‌های جستجوکننده همیشه گنجشک‌های تولیدکننده را مشاهده می‌کنند. اگر گنجشک تولید کننده غذا پیدا کنند، گنجشک جستجوکننده به اطراف آن‌ها پرواز می‌کنند. فرمول به روزرسانی موقعیت جستجوکننده به شرح رابطه (11) است [33].

$$x_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{x_w^t - x_{i,j}^t}{i^2}\right), & i > \frac{n}{2} \\ x_p^{t+1} + |x_{i,j}^t - x_p^{t+1}| \cdot A^+ \cdot L, & i \leq \frac{n}{2} \end{cases} \quad (11)$$

که در آن x_w^t نشان دهنده بدترین موقعیت گنجشک‌ها در جمعیت فعلی، x_p^{t+1} نشان دهنده موقعیت بهینه گنجشک‌ها در جمعیت فعلی و A نشان دهنده ماتریس $1 \times d$ است که در آن هر ورودی به طور تصادفی 1 یا -1 بوده و در آن A^+ به صورت $A^+ = A^T (AA^T)^{-1}$ تعریف می‌شود.

3-3-3- مکانیسم ضد شکار

برای آگاهی از خطرات، 10 تا 20 درصد از جمعیت به صورت تصادفی به عنوان گنجشک‌های پشاهنگ انتخاب می‌شوند. گنجشک‌ها رفتار ضد شکار هوشمندی برای جلوگیری از مهاجمان

برای برنامه‌ریزی مسیر حرکت وسایل پرنده در فضای مورد بررسی از الگوریتم جستجوی گنجشک برای یافتن مسیری با کمترین مقدار تابع هزینه در سطح امن ایجاد شده استفاده می‌شود. الگوریتم جستجوی گنجشک نوع جدیدی از الگوریتم بهینه‌سازی هوش گروهمی است که مشابه الگوریتم جستجوگران جمعیت سپیدار طراحی شده است. این الگوریتم نسبت به الگوریتم‌های دیگر مانند الگوریتم خفاش، الگوریتم بهینه‌ساز گرگ خاکستری، الگوریتم بهینه‌سازی نهنگ، الگوریتم پروانه و الگوریتم بهینه‌سازی یکپارچه، در تابع‌های آزمایش تک حالت و چند حالت عملکرد عالی و همگرایی مناسبی دارد [35]. زیرا به روزرسانی موقعیت الگوریتم جستجوی گنجشک پرشی و ناپیوسته بوده و لذا به طور موثر از وارد شدن در نقطه بهینه محلی جلوگیری می‌کند. گام‌های فرآیند بهینه‌سازی الگوریتم جستجوی گنجشک به شرح زیر است.

3-3-1- به روزرسانی موقعیت برای هر نسل تولید کننده

در الگوریتم جستجوی گنجشک جمعیت گنجشک‌ها به دو گروه تولید کننده و جستجوکننده³³ تقسیم می‌شود. تولیدکنندگان به دنبال منابع غذایی هستند (ارائه دستورالعمل‌هایی برای جستجوی منابع غذایی) در حالی که جستجوکننده‌ها منبع غذایی را که تولیدکنندگان کشف کرده‌اند را پیدا می‌کنند. در الگوریتم جستجوی گنجشک، گنجشک‌های تولیدکننده فعال‌تر اولین غذا را پیدا خواهند کرد. از آنجایی که گنجشک‌های تولیدکننده مسئول پیدا کردن غذا برای تمامی گنجشک‌ها هستند، جهت حرکت به سمت غذا را برای همه گنجشک‌های گره‌ها مشخص می‌کنند. بنابراین گنجشک‌های تولیدکننده دارای منطقه جستجوی غذایی بزرگتری نسبت به گنجشک‌های جستجوکننده‌ها هستند. فرمول به-روزرسانی موقعیت برای هر نسل تولید کننده به صورت رابطه (10) است [33].

دارند. مطابق این رفتار، گنجشک‌های پیشاهنگ در طول فرآیند جستجوی غذا از حمله شکارچیان جلوگیری می‌کنند. پیشاهنگ‌ها هنگامی که مهاجمان را شناسایی می‌کنند صداهای هشدار تولید می‌کنند و بنابراین تولیدکنندگان جستجوکننده‌ها را به مکان امن دیگری هدایت می‌کنند. بنابراین هنگامی که کل جمعیت گنجشک توسط یک شکارچی طبیعی تهدید می‌شود، یک مکانیسم ضد شکار برای پناه بردن به سمت یک موقعیت امن مرکزی ایجاد می‌شود. فرمول رفتار ضد شکار به شرح رابطه (12) است [33].

دارند. مطابق این رفتار، گنجشک‌های پیشاهنگ در طول فرآیند جستجوی غذا از حمله شکارچیان جلوگیری می‌کنند. پیشاهنگ‌ها هنگامی که مهاجمان را شناسایی می‌کنند صداهای هشدار تولید می‌کنند و بنابراین تولیدکنندگان جستجوکننده‌ها را به مکان امن دیگری هدایت می‌کنند. بنابراین هنگامی که کل جمعیت گنجشک توسط یک شکارچی طبیعی تهدید می‌شود، یک مکانیسم ضد شکار برای پناه بردن به سمت یک موقعیت امن مرکزی ایجاد می‌شود. فرمول رفتار ضد شکار به شرح رابطه (12) است [33].

$$x_{i,j}^{t+1} = \begin{cases} x_b^t + \beta \cdot |x_{i,j}^t - x_b^t| & i > \frac{n}{2} \\ x_{i,j}^t + k \cdot \left(\frac{|x_{i,j}^t - x_w^t|}{(f_i - f_w) + \varepsilon} \right) & f_i = f_b \end{cases} \quad (12)$$

که در آن β پارامتر کنترل اندازه گام (یک عدد تصادفی با توزیع نرمال، میانگین 0 و واریانس 1)، $K \in [-1, 1]$ یک عدد تصادفی یکنواخت، f_i نشان دهنده مقدار فیتنس گنجشک i ام فعلی، f_b و f_w به ترتیب بهترین و بدترین مقادیر فیتنس فعلی و ε یک ثابت بسیار کوچک برای جلوگیری از 0 شدن مخرج است.

4-3- استراتژی هموارسازی مسیر منحنی B-SPLINE

پس از استفاده از الگوریتم جستجوی گنجشک برای به دست آوردن مجموعه گره‌های کلیدی، از منحنی B-spline برای اتصال این گره‌ها استفاده می‌شود تا مسیر پرواز برنامه‌ریزی شده صاف شده و از حالت خطوط شکسته خارج شده تا مطابق حرکت پرواز وسایل پرنده در واقعیت باشد. منحنی‌های B-Spline توابع پارامتری بوده که ساخت آن‌ها بر اساس توابع اختلاط است. ساختار پارامتر آن توانایی تولید منحنی‌های غیر یکنواخت را فراهم می‌کند. تابع این نوع منحنی‌ها به صورت رابطه (13) است [33].

$$C(u) = \sum_{i=0}^n P_i \cdot N_{i,p}(u) \quad (13)$$

که در آن، P_i راس کنترل را نشان می‌دهد، $N_{i,p}(u)$ تابع پایه B-spline مرتبه p ($p \geq 1$) منطبق بر P_i است. دنباله $U =$

$$N_{i,0} = \begin{cases} 1 & u_i \leq u \leq u_{i+1} \\ 0 & \text{در غیر این صورت} \end{cases} \quad (14)$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (15)$$

اگر مخرج هر یک از کسر‌ها صفر باشد، مقدار آن کسر صفر تعریف می‌شود. از فرمول بالا می‌توان مقدار هر تابع پایه را محاسبه کرد. منحنی‌های B-spline را می‌توان به منحنی‌های B-spline یکنواخت، منحنی‌های B-spline شبه‌یکنواخت، منحنی‌های Bezier تکه‌ای و منحنی‌های B-spline غیر یکنواخت تقسیم کرد. در این تحقیق، منحنی شبه-یکنواخت B-spline اتخاذ شده است. شبه کد روش تحقیق پیشنهادی در الگوریتم 1 نشان داده شده است. بر اساس این الگوریتم:

گام 1: ابتدا مقاردهای اولیه پارامترها انجام می‌شود. الگوریتم جستجوی گنجشک دارای پنج پارامتر است که به دو پارامتر الگوریتمی و سه پارامتر کنترلی طبقه‌بندی می‌شوند. پارامترهای الگوریتمی اندازه جمعیت (N) و حداکثر تعداد تکرارها (T_{max}) هستند. پارامترهای کنترلی عبارتند از: تعداد تولیدکنندگان (NP)، تعداد جستجوکننده‌ها (NS) و تعداد پیشاهنگ‌ها (SC).

گام 2: در مرحله بعد موقعیت جمعیت اولیه تولید شده مشخص می‌شود. هر $(x_i = x_{i,1}, x_{i,2}, \dots, x_{i,D-1}, x_{i,D})$ در جمعیت، نشان دهنده گنجشک در موقعیت i ام است. $x_{i,j}$ متغیر تصمیم‌گیری i امین گنجشک در مکان j ام است که $j \in [1, D]$ است. D بعد

گام 5: در ادامه، موقعیت قدیمی هر گنجشک $x_i(t)$ در صورتی که تابع تناسب بهتر باشد ($f(x_i(t+1)) < f(x_i(t))$) با موقعیت جدید $x_i(t+1)$ جایگزین می‌شود. در ادامه اگر تناسب $x_i(t+1)$ بهتر از تناسب x^{Gbest} باشد، موقعیت بهترین گنجشک به روز می‌شود. در این مرحله، x^{Gbest} با $x_i(t+1)$ جایگزین می‌شود، سپس وضعیت توقف بررسی شده و تا رسیدن به حداکثر تعداد تکرارهای T_{max} ، این مراحل تکرار می‌شود.

گام 6: در پایان پس از بدست آوردن مجموعه گره‌های کلیدی، از منحنی B-spline برای اتصال این گره‌ها استفاده می‌شود تا مسیر پرواز برنامه‌ریزی شده هموار شده تا مطابق حرکت پرواز وسایل پرنده در واقعیت باشد.

فضای جستجو است. به هر $x_{i,j}$ یک عدد تصادفی بین کران پایین (lb_j) و کران بالا (ub_j) اختصاص داده می‌شود.

گام 3: در این گام، کیفیت موقعیت‌های هر گنجشک در جمعیت با استفاده از تابع تناسب $f(x_i), \forall i = (1, 2, \dots, N)$ با محدودیت‌های اجتناب از برخورد با موانع رابطه (3)، محدودیت فاصله خودرو پرنده مجاور رابطه (4)، محدودیت طول کل پرواز رابطه (5) و محدودیت زوایای تغییر جهت رابطه (7 و 8)، مطابق رابطه (9) محاسبه می‌شود.

گام 4: پس از آن، موقعیت بهترین گنجشک (x^{Gbest}) در جمعیت باید تعیین شود. در ادامه جمعیت N به تولیدکنندگان و جستجوکننده‌ها تقسیم شده و مکان این گنجشک‌ها به ترتیب با معادلات (10) و (11) به روز می‌شود. سپس پیشاهنگ‌ها انتخاب شده و مکان آن با معادله (12) تجدید می‌شود. موقعیت هر دسته از گنجشک‌ها در جمعیت به طور مکرر به منظور بهینه‌سازی تابع هزینه اصلاح می‌شود.

Algorithm_1 The pseudo-code of the proposed model [۳۵]

```

1: Initialize the parameters of SSA:  $N, NP, SC, D$ , and  $T_{max}$ 
2: for  $i = 1:N$  do
3:   for  $j = 1:D$  do
4:      $x_{i,j} = lb_j + (ub_j - lb_j) \times U(0,1)$  {Construct the initial population}
5:   end for
6:   Calculate  $f(x_i)$  {Fitness evaluation}
7: end for
8:  $t = 1$ 
9: while ( $t \leq T_{max}$ ) do
10:   Select the sparrow with the best position  $x^{Gbest}$ 
11:   Select the sparrow with the worst position  $x^{worst}$ 
12:   Divide population into producers and scroungers.
13:   for  $i = 1:NP$  do {Update the  $X$  of the  $i$ th producer}
14:     
$$x_{i,d}^{t+1} = \begin{cases} x_{i,d}^t \cdot \exp\left(\frac{-i}{a \cdot iter_{max}}\right), & R_2 < ST \\ x_{i,d}^t + Q \cdot L, & R_2 \geq ST \end{cases}$$

15:   end for
16:   for  $i = (NP + 1):N$  do {Update the locations of the  $i$ th scrounger}

```

```

17:  $x_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{x_w^t - x_{i,j}^t}{i^2}\right), & i > \frac{n}{2} \\ x_p^{t+1} + |x_{i,j}^t - x_p^{t+1}| \cdot A^+ \cdot L, & i \leq \frac{n}{2} \end{cases}$ 
18: end for
19: for  $i = 1: SC$  do {Update the locations of the  $i$ th scouter}
20:  $x_{i,j}^{t+1} = \begin{cases} x_b^t + \beta \cdot |x_{i,j}^t - x_b^t|, & i > \frac{n}{2} \\ x_{i,j}^t + k \cdot \left(\frac{|x_{i,j}^t - x_w^t|}{(f_i - f_w) + \varepsilon}\right), & f_i = f_b \end{cases}$ 
21: end for
22: for  $i = 1: N$  do
23:  $f(x_i(t)) = F_{cost} = \prod_{i=1}^4 f_i \cdot \omega_i$ 
24: if  $f(x_i(t+1)) < f(x_i(t))$  then
25:  $x_i(t) = x_i(t+1)$  {Archives the current locations of sparrows}
26: end if
27: if  $f(x_i(t+1)) < f(x^{Gbest})$  then
28:  $x^{Gbest} = x_i(t+1)$  {Update the position of the best sparrow}
29: end if
30: end for
31:  $t = t + 1$ 
32: end while
33: Return the best solution  $x^{best}$ 
34: Path smoothing based on the B-SPLINE curve

```

4- نتایج و تحلیل

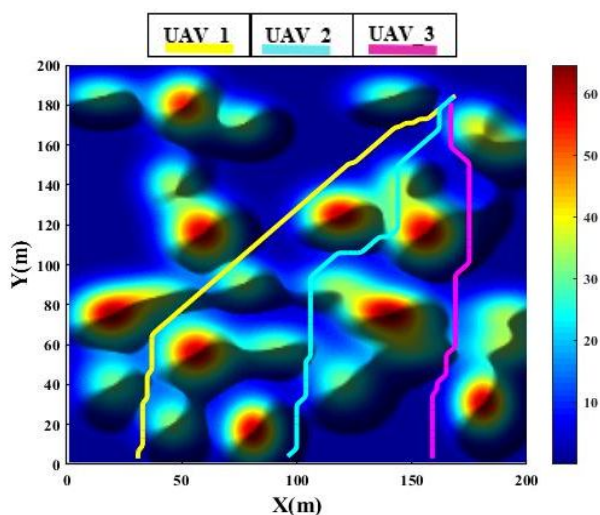
برای تأیید امکان‌سنجی و اثربخشی الگوریتم جستجوی گنجشک، آزمایش شبیه‌سازی بر روی پلتفرم MATLAB انجام شده است. نقشه محیط شبیه‌سازی شده، یک مدل ارتفاعی دیجیتال و یک محیط واقعی کوهستانی است. پارامترهای مدل شبیه‌سازی شده در این مطالعه در جدول 2 آورده شده است.

جدول 2: جدول پارامترهای مدل شبیه‌سازی شده

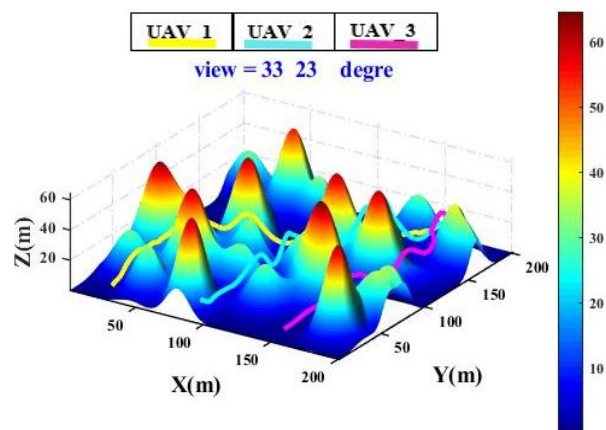
پارامتر	نماد	مقدار
ابعاد محیط	(x, y, z)	(200, 200, 60) و (1000, 1000, 10)
فاصله ایمن	H_imen	5 متر
حداکثر طول پرواز	L_Max	500 و 2000

جمعیت اولیه گنجشک ها
Np
تعداد تکرار در الگوریتم
 $iter_{max}$
200-400

در بستر مدل ارتفاعی دیجیتال برای آزمایش عملکرد الگوریتم جستجوی گنجشک، یک آزمایش استاتیک با مختصات شروع سه خودرو پرنده به ترتیب (8, 1, 31)، (6, 2, 94) و (8, 3, 159) و همچنین مختصات هدف (16, 180, 169) در نظر گرفته شده است. نتایج در شکل 6 و 7 نشان داده شده است. در شکل 6 مسیر پرواز وسایل پرنده در کوهستان قابل مشاهده است. شکل 7 نمای افقی مسیر پرواز سه خودرو پرنده را نشان می‌دهد که به خوبی از برخورد به موانع اجتناب می‌کند و با موفقیت به منطقه هدف می‌رسد.

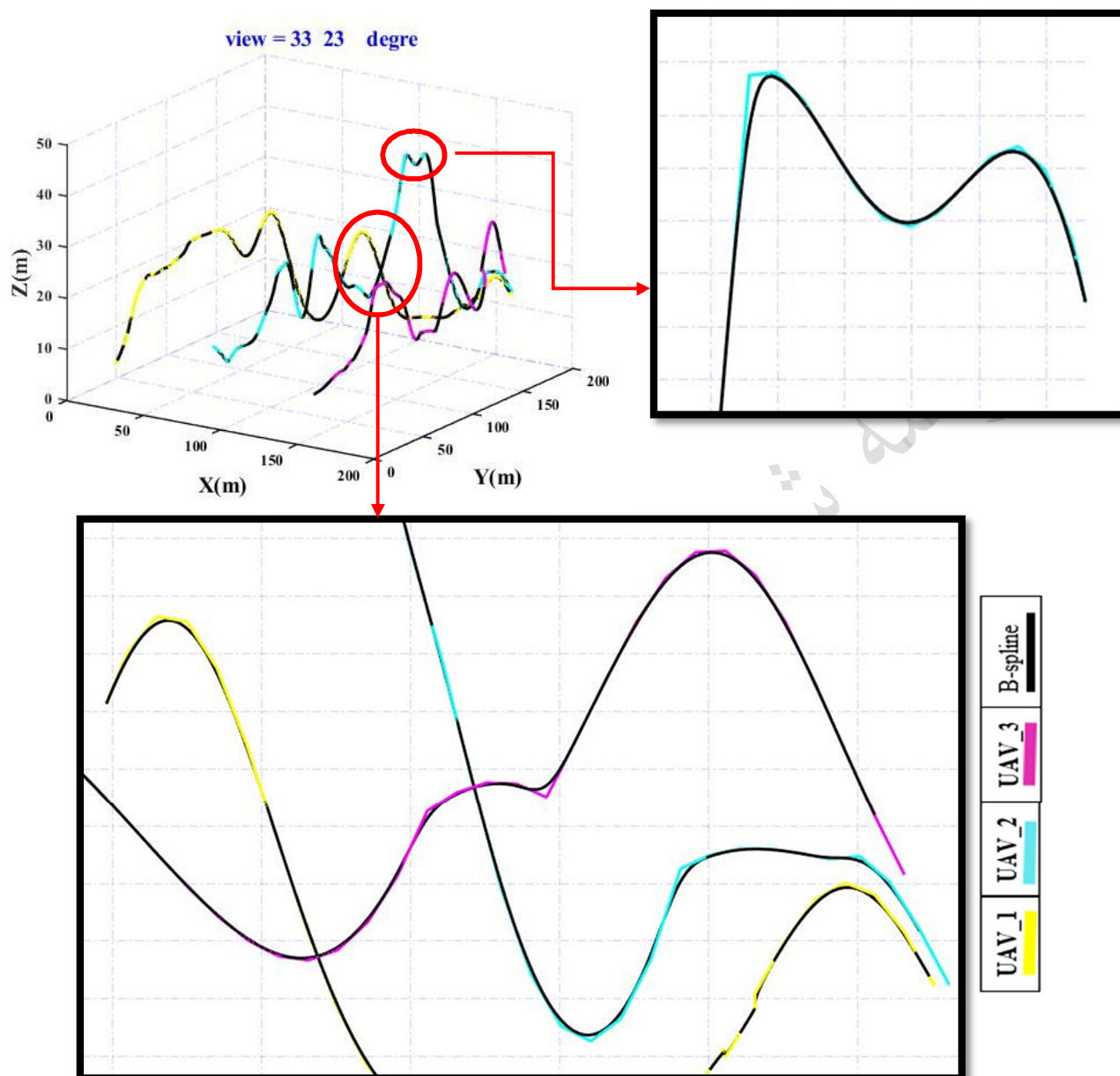


شکل 7. مسیریابی الگوریتم جستجوی گنجشک در بستر مدل ارتفاعی دیجیتال (نمای بالا)



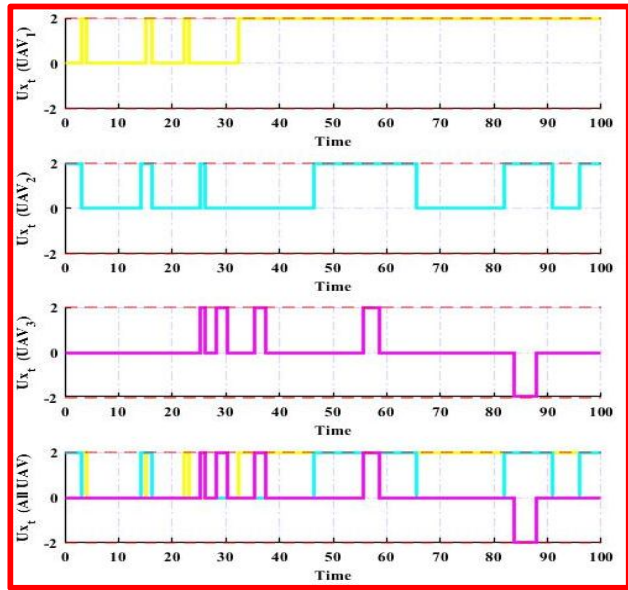
شکل 6. برنامه‌ریزی مسیر الگوریتم جستجوی گنجشک در بستر مدل ارتفاعی دیجیتال

در این تحقیق از منحنی B-spline برای اتصال گره‌ها استفاده شده تا مسیر پرواز برنامه‌ریزی شده صاف شده و از حالت خطوط شکسته خارج شده تا مطابق حرکت پرواز وسایل پرنده در واقعیت باشد. شکل 8 تاثیر منحنی B-spline بر مسیر ایجاد شده توسط الگوریتم جستجوی گنجشک را نشان می‌دهد.

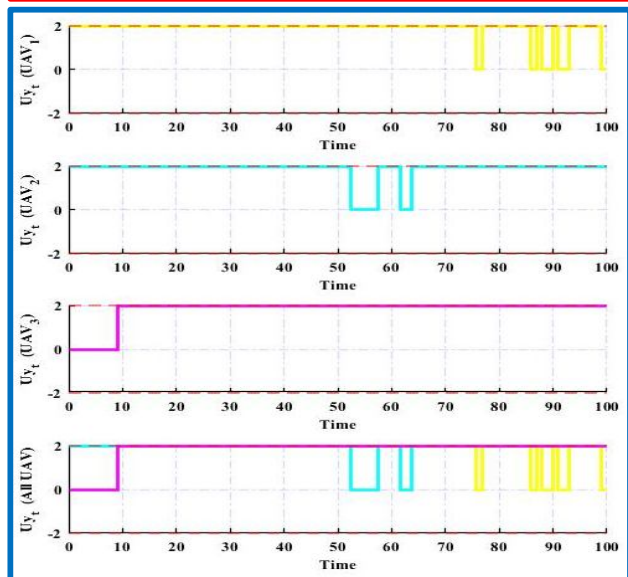
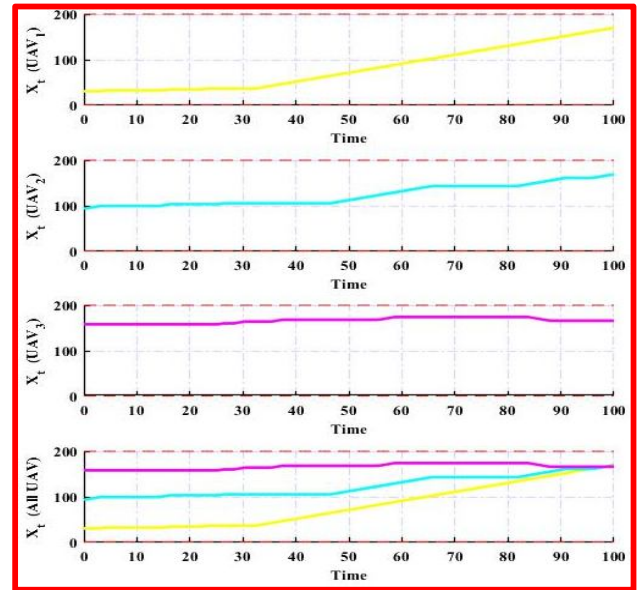


شکل 8. تاثیر منحنی B-spline بر مسیر ایجاد شده توسط الگوریتم جستجوی گنجشک در بستر مدل ارتفاعی دیجیتال

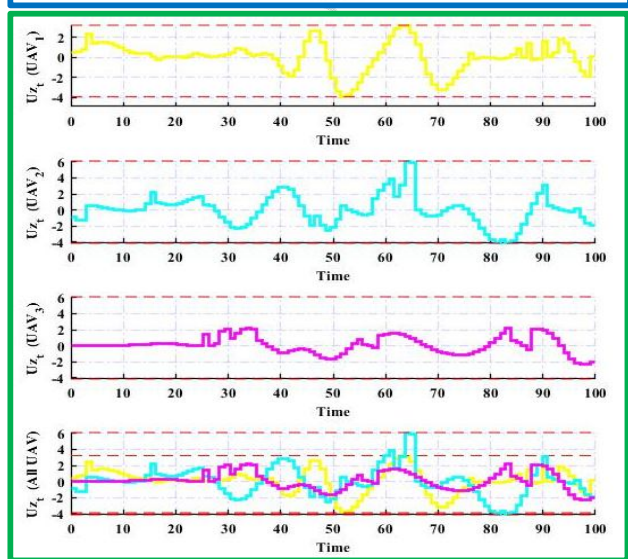
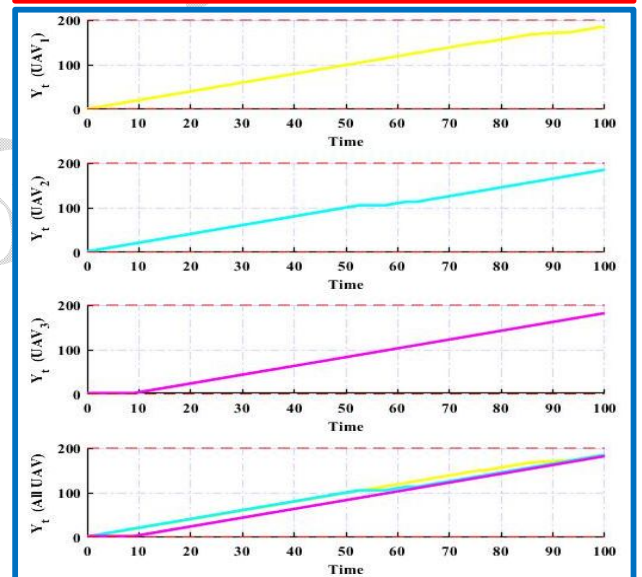
برای بررسی نحوه عبور وسایل پرنده در منطقه کوهستانی و همچنین تغییر جهت‌های ایجاد شده برای عبور از منطقه، در هر گام زمانی تغییرات مختصات وسایل پرنده در شکل 9 و تغییر جهت آن‌ها در شکل 10 نشان داده شده است. بررسی تصاویر نشان می‌دهد که وسایل پرنده برای عدم برخورد به یکدیگر و موانع موجود در کوهستان، تغییر جهت‌های افقی 1 و 2 واحدی و تغییر ارتفاع تا 6 واحد را تجربه کرده‌اند.



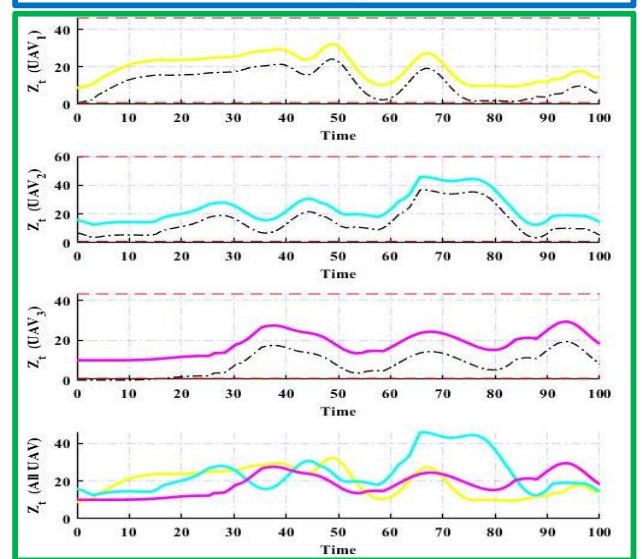
X در راستای



Y در راستای



Z در راستای



UAV_1 UAV_2 UAV_3

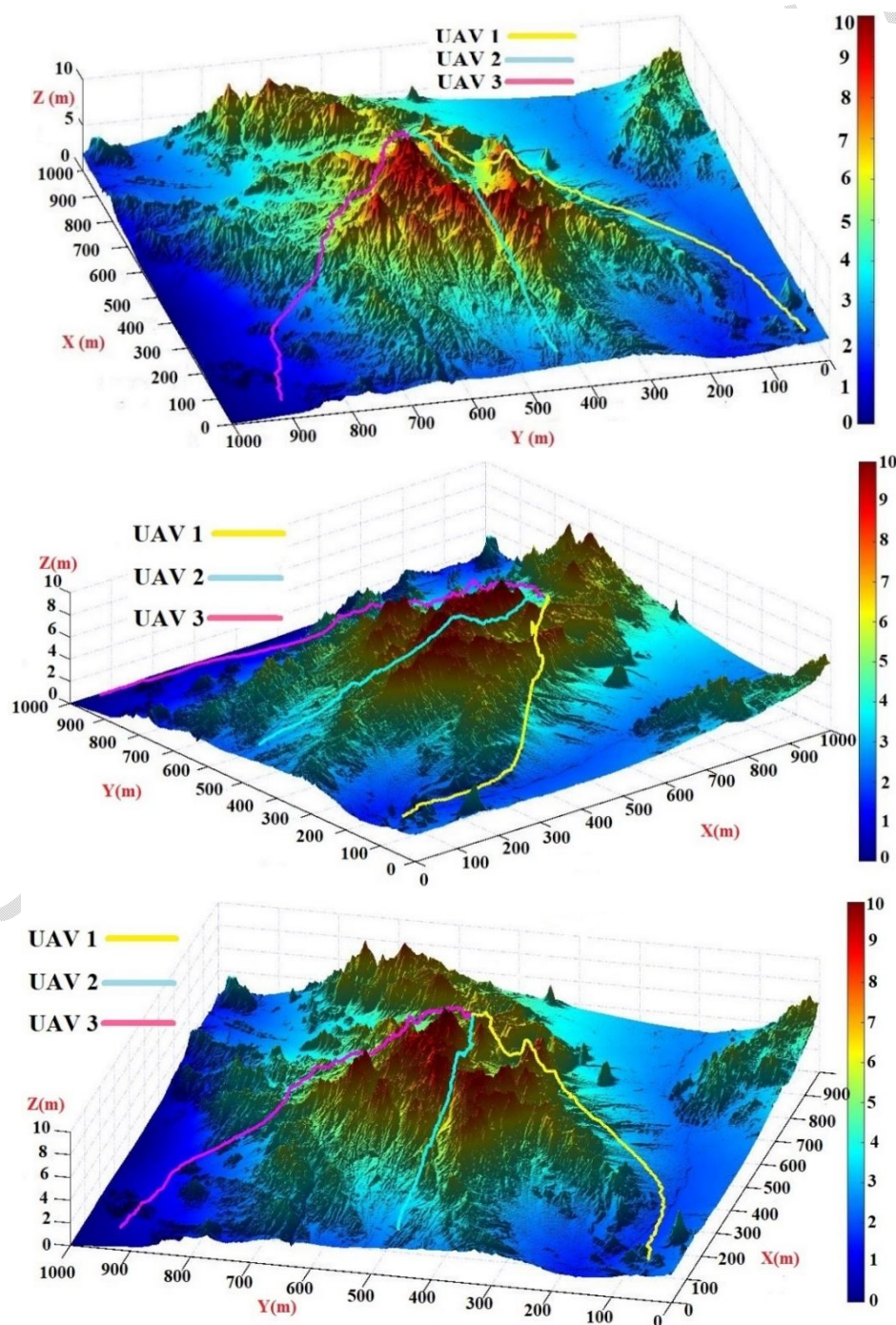
شکل 10. تغییر جهت وسایل پرنده (t) در مدل ارتفاعی دیجیتال

UAV_1 UAV_2 UAV_3

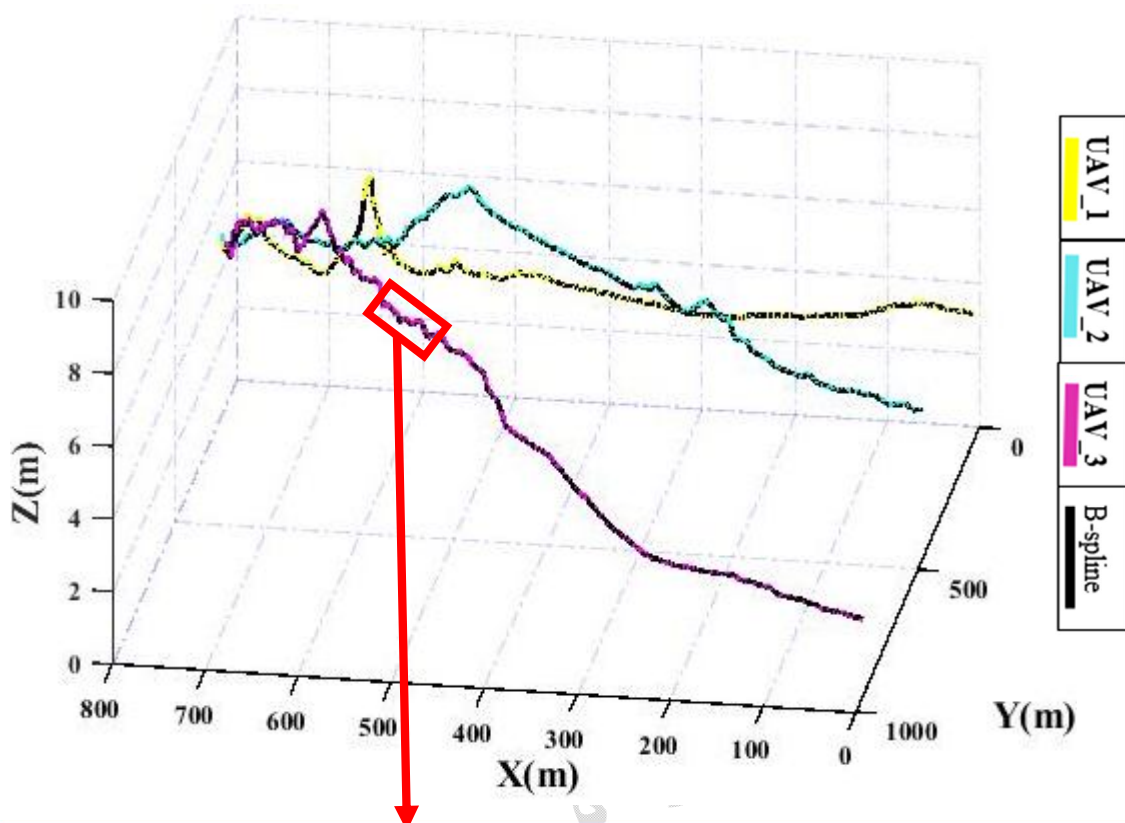
شکل 9. تغییر مختصات وسایل پرنده (t) در مدل ارتفاعی دیجیتال

شکل 12 تاثیر منحنی B-spline بر مسیر ایجاد شده توسط الگوریتم جستجوی گنجشک را نشان می‌دهد. همانطور که مشاهده می‌شود، منحنی B-spline مسیر پرواز برنامه‌ریزی شده را هموار و از حالت خطوط شکسته خارج کرده تا مطابق حرکت پرواز وسایل پرنده در واقعیت باشد.

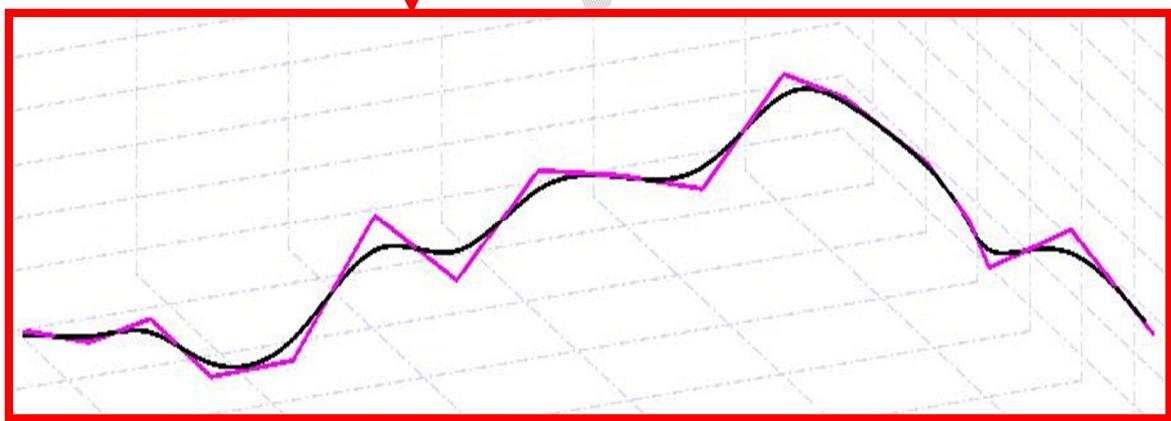
در بستر نقشه DEM، یک آزمایش استاتیک با مختصات شروع سه خودرو پرنده به ترتیب (۱،۴۱،۳)، (۲،۴۵۶،۴) و (۳،۹۲۱،۲) و همچنین مختصات هدف (۴۵۰،۷،۵۵۰) در نظر گرفته شده است. نتایج در شکل 11 در نماهای مختلف نشان می‌دهد که وسایل پرنده خوبی از برخورد با کوهستان اجتناب کرده و در منطقه ایمن پرواز می‌کنند.



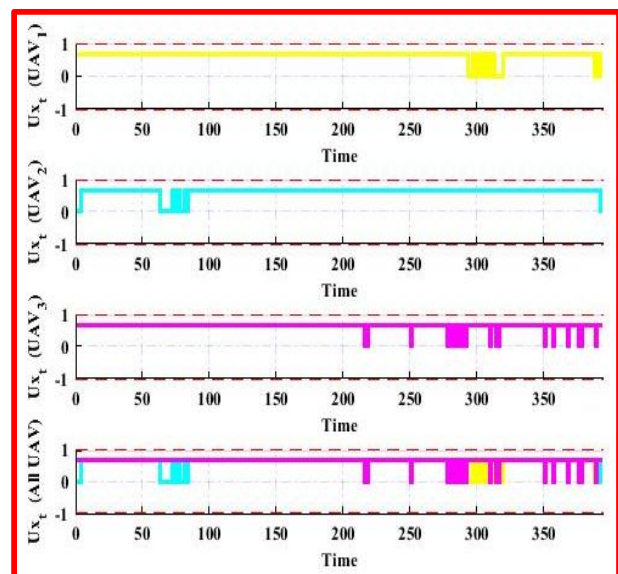
شکل 11. نتایج برنامه ریزی مسیر الگوریتم جستجوی گنجشک نمای بالا در بستر نقشه DEM(Terrain) واقعی



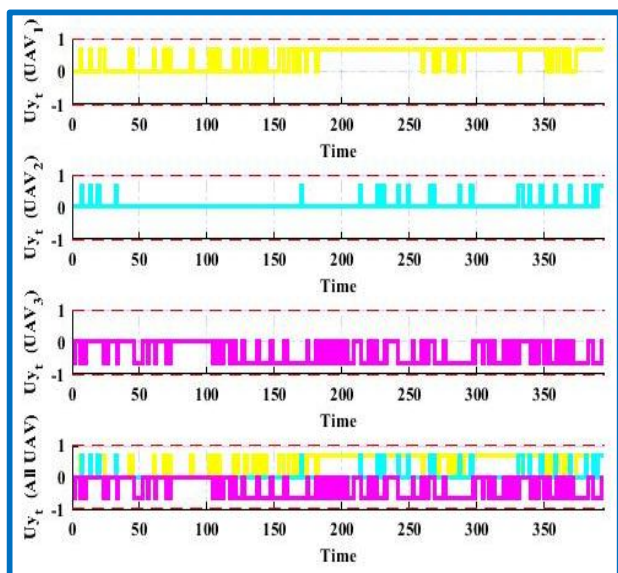
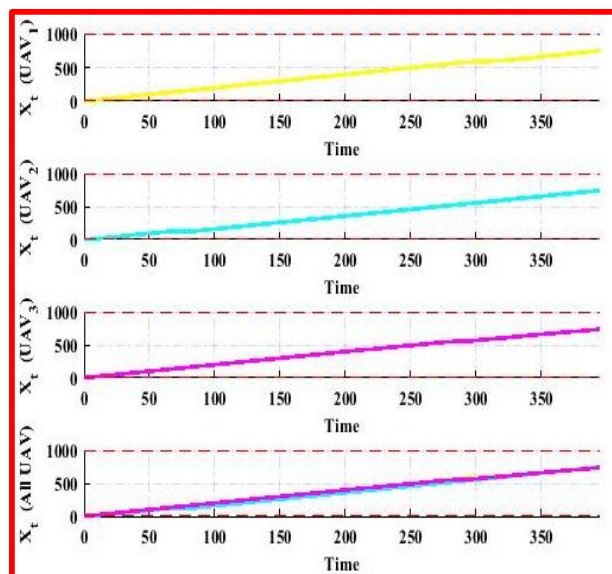
شکل 12. تاثیر منحنی B-spline بر مسیر ایجاد شده توسط الگوریتم جستجوی گنجشک در بستر نقشه DEM(Terrain) واقعی



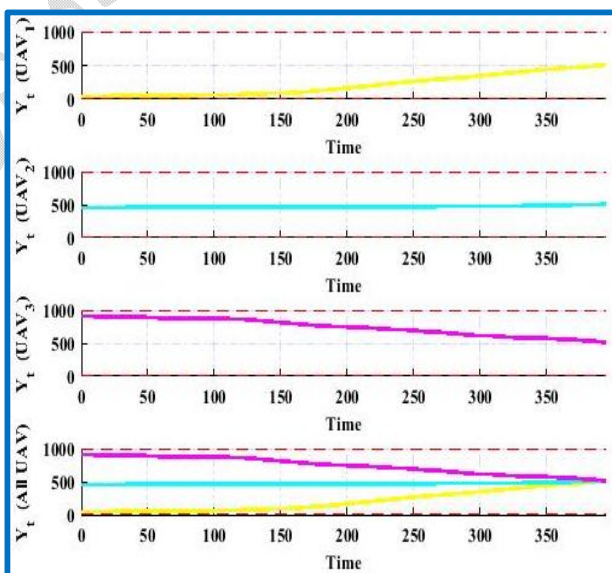
شکل 13 و شکل 14 به ترتیب، تغییرات مختصات وسایل پرنده و تعداد تغییر جهت آن‌ها در هر گام زمانی را نشان می‌دهد. بررسی تصاویر نشان می‌دهد که وسایل پرنده برای عدم برخورد به موانع (کوهستان)، تغییر جهت‌های نوسانات زیادی در راستای قائم داشته است.



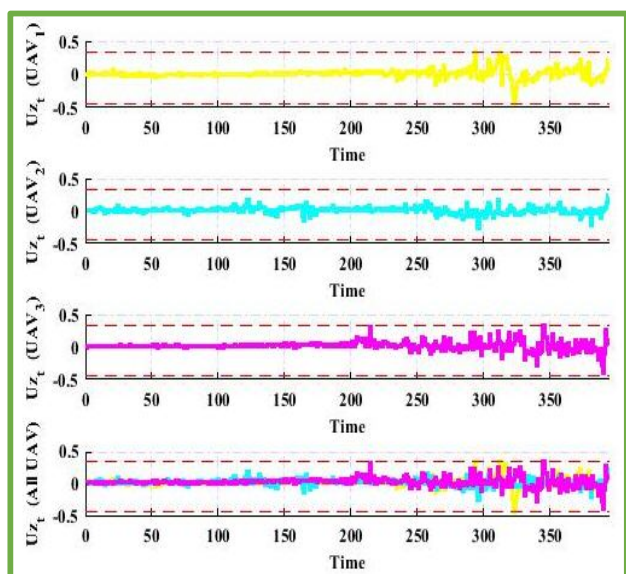
در راستای X



در راستای Y

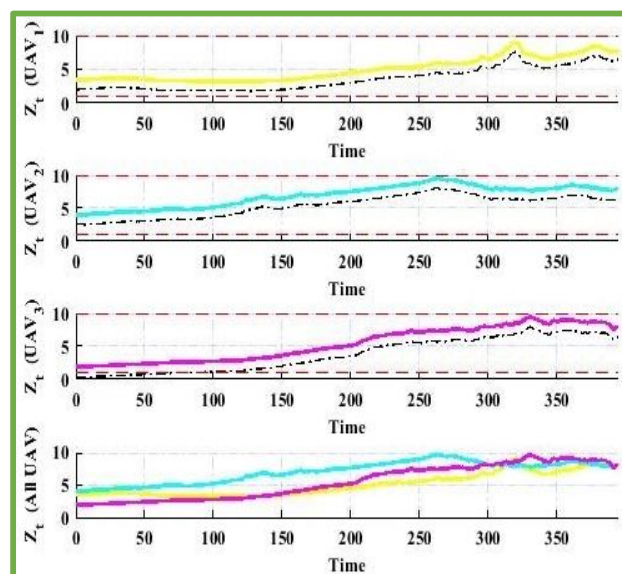


در راستای Z



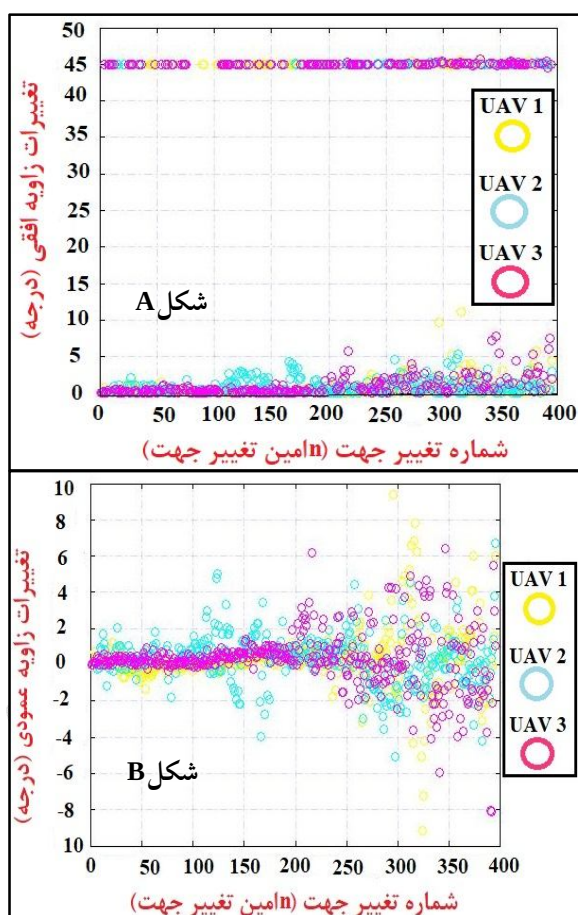
UAV_1 UAV_2 UAV_3

شکل 14. تغییر جهت وسایل پرنده (t) در نقشه DEM واقعی



UAV_1 UAV_2 UAV_3

شکل 13. تغییرات مختصات وسایل پرنده (t) در نقشه DEM



شکل 15. تغییر زاویه افقی (شکل A) و صعود (شکل B) در هر

بار تغییر جهت وسایل پرنده در بستر نقشه DEM (Terrain) واقعی

شکل 15-A تغییر زاویه افقی و شکل 15-B تغییر زاویه صعود در هر بار تغییر جهت وسایل پرنده را نشان می‌دهد. شکل 15-A نشان می‌دهد مدل با بهینه‌سازی تغییرات زاویه، باعث شده که وسایل پرنده در بخش قابل توجهی از مسیر حداکثر 12 درجه و در مواقع اضطراری مانند برخورد با موانع و لزوم چرخش به سمت هدف حداکثر چرخش افقی 45 درجه را تجربه کنند. با توجه به موقعیت اولیه و هدف وسایل پرنده و شرایط توپوگرافی حاکم بر منطقه کوهستانی، وسایل پرنده در نیمه اول از مسیر، در نواحی کم ارتفاع قرار دارند و رفته رفته به قله کوه نزدیک می‌شوند. بنابراین نوسانات زاویه صعود در نیمه اول مسیر می‌بایست در محدوده کمتر و پراکندگی تغییرات زاویه صعود در نیمه دوم مسیر بیشتر باشد. شکل 15-B به خوبی این شرایط را به تصویر می‌کشد.

شاخص‌های آماری ارزیابی عملکرد مسیر بهینه الگوریتم‌های مختلف شامل حداقل، حداکثر، میانگین و انحراف استاندارد طول مسیر در این آزمایش در جدول 4 و جدول 5 نشان داده شده است. مقدار پایین‌تر حداقل، حداکثر، میانگین و انحراف معیار طول مسیر در الگوریتم جستجوی گنجشک نشان می‌دهد که این الگوریتم می‌تواند مسیر بهینه را نسبت به چهار الگوریتم دیگر بهتر جستجو کند. نتایج نشان می‌دهد که حداقل طول مسیر برنامه‌ریزی شده با الگوریتم جستجوی گنجشک در مقایسه با الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات، کلونی زنبورهای مصنوعی و A^* ، 5٪، 3،3٪، 5،6٪ و 9،2٪ کوتاه‌تر شده، متوسط طول مسیرهای برنامه‌ریزی شده نیز به ترتیب 5،1٪، 2،1٪، 5،6٪ و 7٪ کاهش یافته و همچنین انحراف معیار استاندارد طول مسیر بهینه به ترتیب 6،1٪، 5،3٪، 8،6٪ و 13،7٪ کم‌تر شده است.

الگوریتم جستجوی گنجشک کمترین طول مسیر را در تمام شاخص‌ها و A^* بیشترین طول مسیر و بیشترین انحراف معیار را دارد، چون بدون بهینه‌سازی فراابتکاری عمل می‌کند. الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات، کلونی زنبور مصنوعی عملکردی بین الگوریتم جستجوی گنجشک و A^* دارند. همچنین نتایج نشان می‌دهد که انحراف معیار مسیر در الگوریتم جستجوی گنجشک، 13،7٪ کمتر از A^* بوده که نشان می‌دهد در این الگوریتم مسیر پایدارتر و یکنواخت‌تر است.

به طور کلی در این آزمایش، الگوریتم جستجوی گنجشک، هم از نظر دقت (کوتاه‌ترین مسیر) و هم از نظر پایداری (انحراف معیار پایین) عملکرد بهتری نسبت به الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات، کلونی زنبور مصنوعی و A^* داشته است. همچنین بیشترین تفاوت الگوریتم جستجوی گنجشک با A^* بوده که نشان می‌دهد الگوریتم‌های مبتنی بر جستجوی تصادفی و بهینه‌سازی

در ادامه به منظور ارزیابی عملکرد الگوریتم جستجوی گنجشک، این الگوریتم در آزمایشی با الگوریتم‌های پایه‌ای مانند الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات، کلونی زنبور مصنوعی و A^* در یک محیط سه بعدی واقعی مورد مقایسه قرار گرفت. در این آزمایش فاصله ایمن 5 متر، حداکثر طول پرواز 2000 متر، جمعیت اولیه 200 و تعداد تکرار در الگوریتم 250 در نظر گرفته شده است. این آزمایش برای هر الگوریتم 30 بار به طور مستقل اجرا شده است. در این آزمایش پارامترهای الگوریتم جستجوی گنجشک به صورت؛ تعداد تولیدکنندگان ($NP = 120$)، تعداد جستجوکننده‌ها ($NS = 20$) و تعداد پیشاهنگ‌ها ($SC = 60$) تنظیم شده است. نمونه‌ای از پارامترها و جزئیات هر الگوریتم در این آزمایش در جدول 3 نشان داده شده است.

جدول 3: پارامترها و جزئیات هر الگوریتم در آزمایش مقایسه‌ای

الگوریتم	ابعاد محیط	تعداد تکرار	جمعیت اولیه	پارامترها
GA	1000×1000×60	250	200	نرخ جهش=0,02 نرخ تقاطع=0,8
PSO	1000×1000×60	250	200	وزن اینرسی=0,7 ضریب شناختی=1,5 ضریب اجتماعی=1,5
ABC	1000×1000×60	250	200	زنبور کارگر=100 زنبور ناظر=95 زنبور پیشاهنگ=5 آستانه ترک غذا=100
A^*	1000×1000×60	-	-	نوع شبکه در صفحه: مربعی 8 جهته وزن هیورستیک=1 هزینه حرکت افقی=1 هزینه حرکت عمودی=1,4

متاهوریستیک مزایای زیادی در محیط‌های پیچیده و کوهستانی دارند.

جدول 4: مقایسه شاخص‌های آماری ارزیابی الگوریتم‌های مختلف

مدل	شاخص‌های آماری ارزیابی عملکرد			
	حداقل طول مسیر	حداکثر طول مسیر	میانگین طول مسیر	انحراف معیار
GA	1498	1763	1611	69,7
PSO	1523	1788	1658	70,2
ABC	1532	1783	1667	71,9
A*	1584	1801	1689	75,3
SSA	1450	1699	1578	66,2

جدول 5: درصد بهبود عملکرد شاخص‌های آماری الگوریتم جستجوی گنجشک در مقایسه با الگوریتم ژنتیک و الگوریتم ازدحام ذرات

مدل	درصد بهبود شاخص‌های آماری ارزیابی عملکرد			
	حداقل طول مسیر	حداکثر طول مسیر	میانگین طول مسیر	انحراف معیار
SSA-GA	5,03	5,24	5,11	6,14
SSA-PSO	3,31	3,77	2,13	5,33
SSA-ABC	5,65	4,94	5,64	8,61
SSA-A*	9,24	6	7,03	13,74

برای ارزیابی معنی‌داری تفاوت عملکرد الگوریتم جستجوی گنجشک نسبت به سایر الگوریتم‌های مقایسه‌شده شامل الگوریتم ژنتیک، ازدحام ذرات، زنبور عسل مصنوعی و الگوریتم A*، از آزمون ANOVA یک‌طرفه با سطح معنی‌داری $\alpha=0.05$ استفاده شده است.

نتایج آزمون ANOVA نشان داد که تفاوت میانگین طول مسیر بین الگوریتم‌ها با $F = 8.42$ و $P = 0.001$ از نظر آماری معنادار است. برای شناسایی تفاوت‌های مشخص بین الگوریتم‌ها، آزمون پس‌فرض Tukey انجام شد. نتایج این آزمون بدین‌صورت خلاصه می‌شود: تفاوت میان الگوریتم جستجوی گنجشک و الگوریتم ژنتیک با $P = 0.018$ ، میان الگوریتم جستجوی گنجشک و

بهینه‌سازی ازدحام ذرات با $P = 0.004$ و میان الگوریتم جستجوی گنجشک و کلونی زنبور مصنوعی با $P = 0.002$ ، از نظر آماری معنادار است. همچنین تفاوت میان الگوریتم جستجوی گنجشک و A* نیز با $P = 0.009$ از نظر آماری قابل توجه است.

این نتایج نشان می‌دهد که الگوریتم جستجوی گنجشک به طور معناداری مسیرهای کوتاه‌تر و با انحراف معیار کمتر نسبت به سایر الگوریتم‌های مورد بررسی تولید کرده و از عملکرد بهتری در شرایط آزمایش برخوردار بوده است.

5- نتیجه‌گیری

در این مطالعه، بهینه‌سازی برنامه‌ریزی مسیر سه بعدی برای وسایل پرنده در یک محیط کوهستانی (تصادفی و واقعی) مورد بررسی قرار می‌گیرد. برای پاسخگویی به این مسئله از یک مدل بهینه‌سازی هوش گروهمی به نام الگوریتم جستجوی گنجشک، با چهار محدودیت پیشنهادی شامل محدودیت برخورد با موانع، فاصله خودرو پرنده مجاور، طول کل پرواز و زوایای تغییر جهت با هدف کنترل مانور حرکتی خودرو پرنده معرفی شده است. به عنوان یک الگوریتم اکتشافی، الگوریتم جستجوی گنجشک از فعالیت شکارچی و فعالیت فرار یک گروه گنجشک تقلید می‌کند. برای تطابق بیشتر حرکت پرواز وسایل پرنده با واقعیت از منحنی B-spline برای اتصال گره‌ها استفاده شده تا موجب هموارسازی خطوط شکسته شود. آزمایش‌های متعددی در نقشه‌های مختلف تصادفی و واقعی انجام شده تا اثربخشی الگوریتم جستجوی گنجشک با محدودیت‌های پیشنهادی برای مسئله برنامه‌ریزی مسیر سه بعدی وسایل پرنده مورد ارزیابی قرار گیرد. بررسی نحوه عبور وسایل پرنده در منطقه کوهستانی تصادفی و همچنین تغییر جهت‌های ایجاد شده برای عبور از منطقه، در هر گام زمانی نشان می‌دهد که وسایل پرنده برای عدم برخورد به یکدیگر و موانع موجود در کوهستان، تغییر جهت‌های افقی 1 و 2 واحدی و تغییر

ربات‌های حمل‌ونقل، خودروهای هدایت‌شونده خودکار در نقشه‌برداری، پایش محیط‌زیستی، کشاورزی هوشمند و بازرسی صنعتی، می‌تواند به خوبی مورد استفاده قرار گرفته و با بهینه‌سازی مناسب موجب افزایش راندمان و کاهش مصرف انرژی شوند. این دستاورد به شدت به شرکت‌ها در بهینه‌سازی استراتژی‌های تخصیص منابع، کاهش هزینه‌های عملیاتی و بهبود مزایای تولید کمک می‌کند و در نتیجه رقابت شرکت‌ها را در بازار افزایش می‌دهد.

در این تحقیق از الگوریتم پایه جستجوی گنجشک استفاده شده تا عملکرد مدل پایه تحت سناریو مورد نظر ارزیابی شود. این در حالی است که در مطالعات اخیر انجام شده، بهبودهای متعددی در این الگوریتم ایجاد شده است. به عنوان پیشنهادات آینده، می‌توان این الگوریتم‌های بهبود یافته را در سناریوهای مشابه با تحقیق حاضر مورد ارزیابی قرار داد.

مراجع

- [1] Y. Wang, Y. Yue, M. Shan, L. He, and D. Wang, "Formation reconstruction and trajectory replanning for multi-UAV patrol," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 2, pp. 719–729, Apr. 2021, doi: 10.1109/TMECH.2021.3056099.
- [2] Y. Wang, C. Lu, P. Wu, and X. Zhang, "Path planning for unmanned surface vehicle based on improved Q-learning algorithm," *Ocean Eng.*, vol. 292, p. 116510, Jan. 2024.
- [3] R. Shafiei, A. Rafei, and Amiri, "Vehicle routing optimization based on a hybrid of ant colony and particle swarm algorithms using an angular cosine heuristic function," *Soft Computing*, vol. 12, no. 2, pp. 146–164, 2023, doi: 10.22052/scj.2023.248702.1118.
- [4] S. Gao, J. Wu, and J. Ai, "Multi-UAV reconnaissance task allocation for heterogeneous targets using grouping ant colony optimization algorithm," *Soft Comput.*, vol. 25, no. 10, pp. 7155–7167, May 2021.
- [5] M. Movahedi, M. Mazandarani, and A. Amini, "A delay-bounded adaptive routing protocol for vehicular ad hoc networks," *Soft Computing*, vol. 4, no. 2, pp. 68–83, 2021, doi: 10.1001.1.23223707.1394.4.2.58.8.

ارتفاع تا 6 واحد را تجربه کرده‌اند. در محیط واقعی، مدل با بهینه‌سازی تغییرات زاویه، باعث شده که وسایل پرنده در بخش قابل توجهی از مسیر حداکثر با 12 درجه انحراف افقی به سمت هدف حرکت کنند و در مواقع اضطراری مانند برخورد با موانع و لزوم چرخش به سمت هدف حداکثر چرخش افقی 45 درجه را تجربه کنند. انحراف زاویه صعود با توجه به موقعیت اولیه و هدف وسایل پرنده و همچنین شرایط توپوگرافی حاکم بر منطقه کوهستانی مورد بررسی، حدوداً تا 2 درجه در مناطق کم ارتفاع و 10 درجه در مناطق مرتفع می‌باشد. لذا نتایج شبیه‌سازی نشان می‌دهد که این الگوریتم با محدودیت‌های پیشنهادی، می‌تواند ضمن مسیریابی با کمترین هزینه، به سرعت به یک نقطه بهینه همگرا شده و الزامات برنامه‌ریزی مسیر خودرو پرنده را هنگام پیمایش در مناطق کوهستانی پیچیده برآورده کند.

به منظور ارزیابی عملکرد الگوریتم جستجوی گنجشک با الگوریتم‌های پایه‌ای مانند الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات، کلونی زنبور مصنوعی و A^* آزمایشی در یک محیط سه بعدی واقعی طراحی شد. نتایج نشان می‌دهد که حداقل طول مسیر برنامه‌ریزی شده با الگوریتم جستجوی گنجشک در مقایسه با الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات، کلونی زنبورهای مصنوعی و A^* ، 5٪، 3،3٪، 5،6٪ و 9،2٪ کوتاه‌تر شده، متوسط طول مسیرهای برنامه‌ریزی شده نیز به ترتیب 5،1٪، 2،1٪، 5،6٪ و 7٪ کاهش یافته و همچنین انحراف معیار استاندارد طول مسیر بهینه به ترتیب 6،1٪، 5،3٪، 8،6٪ و 13،7٪ کم‌تر شده است.

از منظر عملی، یافتن مسیر بهینه با الگوریتم‌هایی مانند الگوریتم جستجوی گنجشک که با محدودیت‌های منطقی مانند محدودیت‌های تغییر جهت آنی و قدرت مانور در دنیای واقعی تست شده، در بسیاری از سناریوهای کاربرد ربات، مانند

- Artif. Intell. Mechatronics Syst. (AIMS), Apr. 2021.
- [18] L. Zhang et al., "Time-optimal trajectory planning of serial manipulator based on adaptive cuckoo search algorithm," *J. Mech. Sci. Technol.*, vol. 35, no. 7, pp. 3171–3181, Jul. 2021, doi: 10.1007/s12206-021-0637-5.
- [19] Y. Chen, P. Wang, Z. Lin, and C. Sun, "Global path planning method by fusion of A* algorithm and sparrow search algorithm," in *Proc. IEEE 11th DDCLS*, Aug. 2022, pp. 205–209, doi: 10.1109/DDCLS55502.2022.9911336.
- [20] Z. Zhang, R. He, and K. Yang, "A bioinspired path planning approach for mobile robots based on improved sparrow search algorithm," *Adv. Manuf.*, vol. 10, no. 1, pp. 114–130, 2022, doi: 10.1007/s40436-021-00374-0.
- [21] L. Liu et al., "Dynamic path planning of mobile robot based on improved sparrow search algorithm," *Biomimetics*, vol. 8, no. 2, p. 182, 2023, doi: 10.3390/biomimetics8020182.
- [22] G. Zhang and E. Zhang, "An improved sparrow search based intelligent navigational algorithm for local path planning of mobile robot," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 10, pp. 14111–14123, 2023, doi: 10.1007/s12652-022-03886-9.
- [23] Z. Wang et al., "A parallel particle swarm optimization and enhanced sparrow search algorithm for UAV path planning," *Heliyon*, vol. 9, no. 4, 2023, doi: 10.1016/j.heliyon.2023.e15039.
- [24] G. Z. Liu et al., "Path planning of unmanned surface vehicle based on improved sparrow search algorithm," 2023.
- [25] K. Xu et al., "Improved sparrow search algorithm based on multi-strategy collaborative optimization performance and path planning applications," *Processes*, vol. 12, no. 12, p. 2775, 2024, doi: 10.3390/pr12122775.
- [26] G. Liu et al., "A modified sparrow search algorithm with application in 3D route planning for UAV," *Sensors*, vol. 21, no. 4, p. 1224, 2021, doi: 10.3390/s21041224.
- [27] J. Fan and L. Qu, "A multi-strategy improved sparrow search algorithm for mobile robot's path planning," *Meas. Sci. Technol.*, vol. 35, no. 10, p. 106207, 2024, doi: 10.1088/1361-6501/ace361.
- [28] W. Guo, "Improved strategy-based sparrow search algorithm for UAV 2D path optimization," in *Proc. IEEE ICETCI*, May 2024, pp. 1375–1380, doi: 10.1109/ICETCI58574.2024.00415.
- [29] J. Hou et al., "Dynamic path planning for mobile robots by integrating improved sparrow search algorithm and dynamic window approach," *Actuators*, vol. 13, no. 1, p. 24, Jan. 2024, doi: 10.3390/act13010024.
- [6] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 22–34, Jan. 2020, doi: 10.1080/21642583.2019.1708830.
- [7] L. Liu et al., "Path planning techniques for mobile robots: Review and prospect," *Expert Syst. Appl.*, vol. 227, p. 120254, Oct. 2023.
- [8] C. Zhang, X. Yang, R. Zhou, and Z. Guo, "A path planning method based on improved A* and fuzzy control DWA of underground mine vehicles," *Appl. Sci.*, vol. 14, no. 7, p. 3103, Apr. 2024, doi: 10.3390/app14073103.
- [9] U. Challita, W. Saad, and C. Bettstetter, "Deep reinforcement learning for interference-aware path planning of cellular-connected UAVs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jul. 2018, doi: 10.1109/ICC.2018.8422424.
- [10] J. W. Jung et al., "Expanded Douglas–Peucker polygonal approximation and opposite angle-based exact cell decomposition for path planning with curvilinear obstacles," *Appl. Sci.*, vol. 9, no. 4, p. 638, Feb. 2019, doi: 10.3390/app9040638.
- [11] K. R. Guruprasad and T. D. Ranjitha, "CPC algorithm: Exact area coverage by a mobile robot using approximate cellular decomposition," *Robotica*, vol. 39, no. 7, pp. 1141–1162, Jul. 2021, doi: 10.1017/S0263574720001153.
- [12] N. Zagradjanin et al., "Cloud-based multi-robot path planning in complex and crowded environment using fuzzy logic and online learning," *Inf. Technol. Control*, vol. 50, no. 2, pp. 357–374, 2021.
- [13] K. Khaleghi Tabar and F. Farazkish, "Improving routing in vehicular ad hoc networks (VANETs) using nature-inspired algorithms," *Soft Computing*, vol. 6, no. 2, pp. 72–85, 2021, doi: 10.1001.1.23223707.1396.6.2.6.0.
- [14] C. Lamini, S. Benhlila, and A. Elbekri, "Genetic algorithm based approach for autonomous mobile robot path planning," *Procedia Comput. Sci.*, vol. 127, pp. 180–189, Jan. 2018, doi: 10.1016/j.procs.2018.01.113.
- [15] W. Wang, J. Zhao, Z. Li, and J. Huang, "Smooth path planning of mobile robot based on improved ant colony algorithm," *J. Robot.*, vol. 2021, Art. no. 4109821, Jan. 2021, doi: 10.1155/2021/4109821.
- [16] M. K. Muni, D. R. Parhi, and P. B. Kumar, "Improved motion planning of humanoid robots using bacterial foraging optimization," *Robotica*, vol. 39, no. 1, pp. 123–136, Jan. 2021, doi: 10.1017/S0263574720000597.
- [17] G. Muntasha, N. Karna, and S. Y. Shin, "Performance analysis on artificial bee colony algorithm for path planning and collision avoidance in swarm UAVs," in *Proc. Int. Conf.*

- [30] Y. Xu, B. Sang, and Y. Zhang, "Application of improved sparrow search algorithm to path planning of mobile robots," *Biomimetics*, vol. 9, no. 6, p. 351, 2024, doi: 10.3390/biomimetics9060351.
- [31] L. Ouyang et al., "UAV path planning method based on dual-strategy improved sparrow search algorithm," in *Proc. IEEE ICCEA*, Apr. 2024, pp. 1345–1348, doi: 10.1109/ICCEA59650.2024.00190.
- [32] C. Yang et al., "A multi-mechanism balanced advanced learning sparrow search algorithm for UAV path planning," *Cluster Comput.*, pp. 1–44, 2024, doi: 10.1007/s10586-024-04179-5.
- [33] Q. Liu et al., "Multi-UAV path planning based on fusion of sparrow search algorithm and improved bioinspired neural network," *IEEE Access*, vol. 9, pp. 124670–124681, 2021, doi: 10.1109/ACCESS.2021.3102983.
- [34] L. Yang, J. Guo, and Y. Liu, "Three-dimensional UAV cooperative path planning based on the MP-CGWO algorithm," *Int. J. Innov. Comput. Inf. Control*, vol. 16, pp. 991–1006, 2020, doi: 10.24507/ijicic.16.04.991.
- [35] M. A. Awadallah et al., "Recent versions and applications of sparrow search algorithm," *Arch. Comput. Methods Eng.*, vol. 30, no. 5, pp. 2831–2858, 2023, doi: 10.1007/s11831-021-09675-3.