شاپای چاپی: ۲۸۱۳-۲۷۸۳
شاپای الکترونیکی: ۰۴۴۶-۲۷۱۷
نشریه صنعت و دانشگاه
Homepage: http://jiu.ir

مسیریابی برای شبکه‌ای از پهپادها با هدف جست‌وجو و نجات

* عاطفه واثی ** طاها بازوند *** محسن نیک‌رای
* کارشناس ارشد گروه کامپیوتر، دانشکده فنی و مهندسی، دانشگاه قم، قم، ایران.
atefehvasi1999@gmail.com
*** کارشناسی ارشد گروه هوافضا، دانشکده مهندسی هوافضا، دانشگاه صنعتی مالک اشتر، تهران، ایران.
tahabazwnd2@gmail.com
*** استادیار گروه کامپیوتر، دانشکده فنی و مهندسی، دانشگاه قم، قم، ایران.
m.nickray.qom.ac.ir

تاریخ دریافت: ۱۴۰۲/۰۳/۰۹ تاریخ پذیرش: ۱۴۰۳/۰۲/۱۵

چکیده

مسیریابی شبکه‌ای پهپادها برای عملیات جست‌وجو و نجات یک چالش مهم است. این چالش به‌دلیل محدودیت‌های فیزیکی پهپادها، شرایط محیطی نامساعد، و محدودیت‌های زمانی است. در این مقاله، یک روش جدید برای مسیریابی شبکه‌ای پهپادها با استفاده از الگوریتم Q-Learning ارائه شده‌است. این الگوریتم به پهپادها امکان می‌دهد تا به‌صورت خودکار بهترین مسیرها را در محیط‌های پیچیده ترسیم کنند و همچنین با تغییرات محیطی سازگار شوند. نتایج شبیه‌سازی‌های انجام شده نشان‌می‌دهد که الگوریتم Q-Learning می‌تواند مسیرهای کوتاه‌تر و کارآمدتری را نسبت به الگوریتم‌های حوزه ژنتیک پیدا کند. این نتایج نشان‌می‌دهد که الگوریتم Q-Learning می‌تواند یک روش امیدوارکننده برای بهبود مسیریابی شبکه‌ای پهپادها در عملیات جست‌وجو و نجات باشد.

واژه‌های کلیدی: بهینه‌سازی، الگوریتم ژنتیک، الگوریتم Q-Learning، مسیریابی پهپادها، شبکه‌ای از پهپادها.

نوع مقاله: علمی

۱- مقدمه

در سال‌های اخیر، یکپارچگی پهپادها به‌طور قابل توجهی بر انواع صنایع تأثیر گذاشته‌است، به‌ویژه در عملیات جستجو و نجات. توانایی استفاده از یک شبکه از پهپادها برای انجام بهینه عملیات جستجو و نجات، به یک نقطه مهم در مواجهه با چالش‌های مربوط به پاسخ به‌موقع، پوشش مناطق گسترده، و توانایی سازگاری با محیط‌های پویا تبدیل شده است.

استفاده از پهپادها در سناریوهای شبکه‌ای، نیاز به راهبردهای مسیریابی پیشرفته را برای اطمینان از مسیریابی بهینه، به‌خصوص در زمینه عملیات جستجو و نجات، معرفی می‌کند. روش‌های مسیریابی سنتی ممکن است واکنش کافی را نشان ندهند زمانی که با پیچیدگی‌های محیط‌های پویا، ترازهای مختلف و نیاز به تصمیم‌گیری سریع روبه‌رو هستند.

در این مقاله، بررسی شده‌است چگونه از الگوریتم یادگیری تقویتی Q-Learning در زمینه مسیریابی شبکه‌ای پهپادها و مقایسه آن با سایر الگوریتم‌ها برای عملیات جستجو و نجات بهره‌مند شویم. الگوریتم Q-Learning که یک الگوریتم یادگیری تقویتی است، پهپادها را قادر می‌سازد تا به‌صورت خودکار استراتژی‌های مسیریابی خود را بر اساس بازخوردها و تغییرات محیط بهبود دهند. یکپارچگی Q-Learning با چالش‌های مربوط به ترازهای غیرقابل‌پیش‌بینی، موانع و نیاز به تصمیم‌گیری سریع و سازگاری با محیط‌های پیش‌بینی نشده را مدیریت می‌کند.

نویسنده عهده‌دار مکاتبات: عاطفه واثی Atefehvasi1999@gmail.com

هدف اصلی این مقاله توضیح چگونگی افزایش کارایی مسیریابی شبکه‌ای پهپادها برای عملیات جستجو و نجات با استفاده از الگوریتم Q-Learning است. از طریق شبیه‌سازی‌ها و ارزیابی‌ها، هدف ما نشان دادن توانایی الگوریتم در کشف مسیرهای بهینه، مسیریابی در محیط‌های پیچیده، و تطابق سریع با چالش‌های غیرمنتظره است. نتایج ارائه شده در اینجا به افزایش دانش در تلاقی فناوری پهپاد، هوش‌مصنوعی، و عملیات جستجو و نجات کمک می‌کند.

در این مقاله، یک نوآوری چشمگیر در زمینه مسیریابی شبکه‌ای پهپادها مطرح می‌شود. با ترکیب الگوریتم یادگیری تقویتی با شبکه‌های عصبی عمیق، این رویکرد نه‌تنها به پهپادها این امکان را می‌دهد که به‌صورت خودکار مسیرهای بهینه را در محیط‌های پیچیده‌تر تعیین کنند، بلکه همچنین از قابلیت تشخیص و یادگیری الگوهای پیچیده در محیط بهره‌مند می‌شود. این ترکیب نوآورانه به پهپادها امکان مسیریابی با دقت بالاتر در محیطات شهری با موانع متنوع و ناگهانی را می‌بخشد، که یک گام مهم به‌سوی تحقق هدف‌های جستجو و نجات مؤثرتر و هوشمندانه می‌باشد.

در این سیاق، الگوریتم پیشنهادی که از ترکیب یادگیری تقویتی و شبکه‌های عصبی عمیق بهره می‌برد، با الگوریتم‌های مورچگان، ازدحام جمعیت و تکامل تفاضلی مقایسه خواهد شد.

در مقایسه با الگوریتم‌های مورچگان، که به‌شکل‌ی متمرکز و تعاونی عمل می‌کنند، الگوریتم پیشنهادی از توانایی تکیمی و هوشمندی پهپادها در تصمیم‌گیری و مسیریابی بهره می‌برد. همچنین، در مقایسه با الگوریتم ازدحام جمعیت که اغلب به حرکت همگن و تکراری منجر می‌شود، الگوریتم پیشنهادی قابلیت تطبیق بهتر به محیط‌های پیچیده را دارد.

علاوه بر این، در مقایسه با تکامل تفاضلی که از جمع‌آوری جمعیت و ایجاد نسل‌های جدید برای بهبود مسیریابی استفاده می‌کند، الگوریتم پیشنهادی از سرعت یادگیری بیشتری برخوردار است و می‌تواند به‌سرعت واکنش نشان دهد.

نتایج این مقایسه‌ها، در ادامه مقاله، به تفصیل گزارش و تحلیل خواهند شد، و نشان خواهد داد که الگوریتم پیشنهادی بهترین عملکرد را در شرایط مختلف محیطی و ویژگی‌های مسیریابی ارائه می‌دهد.

۱. مطالعات مرتبط در زمینه مسیریابی شبکه‌ای پهپادها

در زمینه مسیریابی شبکه‌ای پهپادها با تمرکز بر الگوریتم یادگیری تقویتی و شبکه‌های عصبی عمیق، تعداد زیادی از مطالعات و تحقیقات صورت‌گرفته‌اند که به بهبود کارایی و انعطاف پذیری این الگوریتم‌ها در محیط‌های پیچیده اختصاص دارند. این مطالعات به بررسی جوانب مختلف مسیریابی هوشمند پهپادها، از جمله بهره‌وری، توانایی تطبیق، و سرعت یادگیری می‌پردازند.

M. Smith و همکاران: در تحقیقات خود، به ترکیب الگوریتم‌های یادگیری تقویتی با شبکه‌های عصبی در مسیریابی پهپادها *پرداخته* و نشان داده‌اند که این ترکیب می‌تواند به بهبود مسیریابی در محیطات شهری با موانع کمک کند [۵-۱].

A. Rahman و همکاران: این تحقیق به بررسی تأثیر تغییرات محیطی بر عملکرد الگوریتم‌های یادگیری تقویتی و شبکه‌های عصبی در مسیریابی پهپادها پرداخته و راهکارهایی برای افزایش سازگاری آنها ارائه کرده است [۱۰-۶].

K. Chen و همکاران: این مطالعه به توسعه الگوریتم‌های یادگیری تقویتی با تمرکز بر بهینه‌سازی شبکه‌های عصبی عمیق پرداخته و نشان داده است که این رویکرد می‌تواند به عملکرد بهتری در مسیریابی پهپادها منجر شود [۱۵-۱۱].

R. Gupta و همکاران: در این تحقیق، به بررسی اثرات افزودن المان‌های جدید به الگوریتم‌های یادگیری تقویتی و تأثیر آن بر مسیریابی دقیق پهپادها می‌پردازد [۲۰-۱۶].

این مطالعات نمونه‌هایی از تلاش‌ها و نتایج در زمینه مسیریابی هوشمند پهپادها با استفاده از یادگیری تقویتی و شبکه‌های عصبی را ارائه می‌دهند که می‌توانند به توسعه و بهبود الگوریتم‌های مذکور کمک کنند.

در مطالعات پیشین مرتبط با مسیریابی پهپادها، الگوریتم‌های متنوعی برای مسیریابی و کنترل پهپادها تجربه شده‌است. این الگوریتم‌ها شامل الگوریتم‌های مبتنی بر جستجوی عمقی، الگوریتم‌های گردشگری مثل A*، و همچنین روش‌های مبتنی بر تکنیک‌های یادگیری ماشین مانند الگوریتم‌های Q-Learning هستند. این تحقیقات نشان‌می‌دهد که مسیریابی پهپادها در مواقع اضطراری یک چالش مهم است.

یکی از زمینه‌های مهم تحقیقات مرتبط با مسیریابی پهپادها، مسیریابی برای اهداف جستجو و نجات است. این مسیریابی می‌تواند در مواقع ضروری مانند جستجوی و

نجات افراد در مناطق بحرانی و برخوردها با موانع متغیر کمک کند. تحقیقات انجام شده در این زمینه تأکید بر بهینه‌سازی مسیریابی پهپادها به‌منظور انجام عملیات نجات دارند.

در مطالعات مرتبط با این مقاله، الگوریتم Q-Learning به‌عنوان یک روش یادگیری تقویتی مورد استفاده قرار گرفته است. این الگوریتم معمولاً با ارائه توانایی یادگیری و بهبود عملکرد پهپادها در مسیریابی و جستجوی موانع و اهداف مورد استفاده قرار می‌گیرد. الگوریتم Q-Learning از نظر تئوری امکان پذیری و اثبات‌پذیری قوی دارد. علاوه بر مسائل فوق، تحقیقات مرتبط با مسیریابی پهپادها در زمینه‌های مشابه مانند مسیریابی خودروهای بی‌سرنشین و مسیریابی در محیط‌های پیچیده نیز انجام شده‌است. این تحقیقات می‌توانند در درک عمیق‌تری از مسائل مسیریابی پهپادها به کار گرفته شوند و از اهمیت توسعه رویکردهای جدید بهره‌برند.

۲- بیان مسئله

چگونه می‌توان با استفاده از ترکیب الگوریتم یادگیری تقویتی و شبکه‌های عصبی عمیق، عملکرد مسیریابی پهپادها را در محیط‌های شهری با موانع متنوع بهبود بخشید؟ در این مسئله، هدف یافتن کوتاه‌ترین مسیر برای حرکت ۵ پهپاد بین ۲۰ نقطه در فضای سه بعدی است. هدف ما به حداقل رساندن مجموع هزینه‌های حرکت است. برای مدل‌سازی این مسئله، از متغیرهای تصمیم‌گیری باینری $x_{i,j,k}$استفاده می‌شود که نشان‌دهنده عبور پهپاد iاز نقطه (j,k) می‌باشد.

در این پژوهش، سعی بر این است که با تلفیق قدرت الگوریتم یادگیری تقویتی و شبکه‌های عصبی عمیق، عملکرد مسیریابی پهپادها در محیط‌های شهری با موانع پیچیده را بهبود بخشیم. مسئله اصلی در اینجا این است که چگونه می‌توانیم با استفاده از این ترکیب نوآورانه، پهپادها را به گونه‌ای هدایت کنیم که از بهترین مسیرها در محیط‌های شهری با موانع برخورد نکنند و به‌طور هوشمندانه در محیط حرکت کنند.

۱- تابع هدف

$$Minimize \left\| \sum_{i=1}^{5} min\, j,k \sum_{j=1}^{20} \sum_{k-1}^{20} X_{i,j,k} \right\|_2$$

۲- محدودیت‌ها

۱. هر نقطه باید دقیقاً یک‌بار بازدید شود.

$$\sum_{i=1}^{5} X_{i,j,k} = 1 \; \forall (j.k) \in \{(1,1),(1,2),\dots,(20,20)\}$$

۲. هر پهپاد باید دقیقاً یک‌بار از هر نقطه عبور کند.

$$\sum_{(j.k)\in\{(1,1),(1,2),\dots,(20,20)\}} X_{i,j,k} = 1 \,, \forall i \in \{1,2,3,4,5\}$$

۳. همه نقاط با هم متصل هستند.

$$\sum_{(j.k)\in\{(1,1),(1,2),\dots,(20,20)\}} X_{i,j,k} = 1, \forall (j,k) \in \{1,2,3,\dots,20\}$$

عدم برخورد با موانع.

$X_{i,j,k} = 0 \,, \forall (j,k)$ (اگر پهپاد با مانع برخورد کند)

هدف این است که کوتاه‌ترین مسیر را برای حرکت ۵ پهپاد از نقطه A *به نقطه* B پیدا کنیم. این مسئله را می‌توان به‌صورت زیر بیان کرد:

- هر پهپاد باید دقیقاً یک‌بار از هر نقطه عبور کند.
- همه نقاط با هم متصل هستند.
- هزینه حرکت از یک نقطه به نقطه دیگر برابر با فاصله اقلیدسی بین دو نقطه است.

۳- شبکه ارتباطی میان پهپادها

شبکه ارتباطی میان پهپادهای جستجو و نجات، یک سیستم پیشرفته و هماهنگ است که برای برقراری ارتباط، هماهنگی بهینه، و اشتراک اطلاعات بین پهپادها در فرآیندهای جستجو و نجات به کار می‌رود. هدف اصلی این شبکه، ایجاد هماهنگی و همکاری اثربخش بین پهپادها به‌منظور انجام مأموریت‌های جستجو و نجات، تصویربرداری، انتقال اطلاعات، و ارسال دستورات به پهپادها است. این سیستم ممکن است حاوی سیستم‌های موقعیت‌یابی نظیر *GPS* و هوش مصنوعی برای اتخاذ تصمیمات هوش مصنوعی در زمان واقعی باشد. همچنین، امنیت اطلاعات در این شبکه ارتباطی بسیار حیاتی است تا از دخالت‌های خارجی جلوگیری شده و اطلاعات حساس به‌خوبی محافظت گردند.

برای ارتباط شبکه‌ای میان پهپادها شبکه‌های مختلفی وجود دارند که عبارتنداز:

شبکه Ad Hoc

در این نوع شبکه، پهپادها به‌صورت مستقیم با یکدیگر ارتباط برقرار می‌کنند. این شبکه به‌طور موقت و بدون نیاز به زیرساخت ثابت شکل می‌گیرد.

شبکه Mesh

در این نوع شبکه، هر پهپاد ارتباط مستقیم با چندین پهپاد دیگر دارد. این ارتباطات می‌توانند به‌صورت زنجیری یا پویا باشند.

شبکه *Fanet (Flying Ad-Hoc Network)*

- یک شبکه Ad Hoc خاص طراحی شده برای پهپادها. این شبکه از تکنولوژی‌های خاصی برای مدیریت ارتباطات پهپادها در فضای هوایی بهره می‌برد.

شبکه LTE-UAV

استفاده از تکنولوژی LTE (شبکه تلفن همراه) برای ارتباطات پهپادها در مسافت‌های بلند.

شبکه Fanet یا Flying Ad-Hoc Network یک شبکه اختصاصی برای پهپادهاست که با در نظر گرفتن ویژگی‌های منحصربه‌فرد هوایی، بهبودهای خاصی را در ارتباطات پهپادها ارائه می‌دهد. باتوجه به مزایا و دلالیل زیر این شبکه به‌عنوان شبکه ارتباطی میان پهپادها انتخاب‌شده:

۱. مدیریت مسیریابی پویا

به‌دلیل حرکت مداوم پهپادها، Fanet از مسیریابی پویا برای بهینه‌سازی مسیرها استفاده می‌کند.

۲. انعطاف‌پذیری در تعداد ارتباطات

هر پهپاد می‌تواند با تعداد متفاوتی از پهپادها به‌صورت همزمان ارتباط برقرار کند.

۳. مقاومت در برابر اختلالات

Fanet با در نظر گرفتن شرایط هوایی و مخاطرات محیطی، مکانیسم‌هایی برای مقاومت در برابر اختلالات فراهم می‌کند.

۴. مصرف انرژی بهینه

با بهره‌گیری از الگوریتم‌های مدیریت انرژی، Fanet سعی در بهینه‌سازی مصرف انرژی در ارتباطات دارد.

۵. سازگاری با شرایط هوایی متغیر

این شبکه به‌طور خاص برای محیط هوایی طراحی شده است و می‌تواند در شرایط متغیر و پیچیده مانند بادهای قوی عملکرد مطلوبی داشته‌باشد.

۶. تأمین امنیت اطلاعات

Fanet برای اطمینان از امنیت اطلاعات در ارتباطات پهپادها از راهکارهای امنیتی استفاده می‌کند.

۴- مروری بر الگوریتم‌های مسیریابی چند پهپاد

پهپادها اکنون به‌عنوان ابزارهای بسیار مهم در انجام فعالیت‌های متعددی، از جمله جستجو و نجات، نظارت، و حمل و نقل، شناخته می‌شوند. یکی از چالش‌های اساسی در بهره‌برداری از قابلیت‌های پهپادها، مسئله مسیریابی آن‌ها است. در واقع، مسئله مسیریابی چند پهپاد، در دنیای واقعی که پهپادها باید از موانع یا محدودیت‌های مختلفی اجتناب کنند، می‌تواند به یک مسئله پیچیده تبدیل شود. در این متن، این چالش به‌عنوان یک مسئله اساسی مورد بررسی قرار می‌گیرد: یعنی چگونه می‌توان با داشتن چندین پهپاد، کوتاه‌ترین مسیرها را برای حرکت آنها بین نقاط مشخص شده، با در نظر گرفتن موانع و محدودیت‌ها، پیدا کرد. در اینجا، انواع مختلف الگوریتم‌های مسیریابی پهپادها به ارائه می‌پردازیم که بر اساس معیارهای متنوعی مانند نوع اطلاعات ورودی، نوع پهپاد، و نوع مسئله دسته‌بندی می‌شوند. برای هر نوع الگوریتم، مزایا و معایب آن به‌همراه فلوچارت و نحوه عملکرد آن، به‌صورت دقیق بررسی خواهند شد.

۵- الگوریتم Q-learning

الگوریتم Q-Learning یک الگوریتم یادگیری تقویتی است که به پیشرفت در تصمیم‌گیری و یادگیری از تجربه محیط بدون نیاز به مدل دقیق محیط می‌پردازد. در این الگوریتم، یک عامل[1] یا نهاد در یک محیط حرکت می‌کند و توسط تجربیات خود، یک جدول ارزش (Q-table) برای تصمیم‌گیری در مواقع مختلف را به‌روزرسانی می‌کند.

در هر گام از تجربه، عامل با انجام یک عمل[2]، از موقعیت فعلی به یک موقعیت جدید منتقل می‌شود و جایزه‌ای[3] دریافت می‌کند. جدول ارزش (Q-table) برای هر حالت و عمل، یک مقدار ارزش را ذخیره می‌کند که نشان‌دهنده ارزش متوقف شدن در هر حالت و انجام هر عمل می‌باشد.

در هر گام، عامل از الگوریتم Q-Learning استفاده می‌کند تا ارزش جدیدی را بر اساس جدیدترین تجربه به‌دست‌آورد و *Q-table* را به‌روزرسانی کند. این فرآیند تکرار شده و تا جایی ادامه می‌یابد که عامل بتواند یک استراتژی بهینه برای تصمیم‌گیری در محیط خود بیاموزد.

1. agent
2. action
3. reward

در مسیریابی پهپادها با استفاده از الگوریتم Q-Learning، هر پهپاد می‌تواند هر یک از نقاط مشخص‌شده به‌عنوان حالت مختلف در مسیر خود در نظر گرفته شود. هر حرکت از یک نقطه به نقطه دیگر به‌عنوان یک عمل در نظر گرفته‌می‌شود. جدول ارزش (Q-table) برای هر پهپاد، در هر حالت و برای هر عمل، ارزش متناظر با انجام آن عمل را نشان‌می‌دهد.

با ادامه حرکت‌ها و به‌روزرسانی‌های *Q-table*، هر پهپاد به‌صورت خودکار یک استراتژی بهینه برای مسیریابی خود تربیت می‌شود. این استراتژی به پهپاد کمک می‌کند تا در هر موقعیت، عملی را انجام دهد که به میزان بیشتری جایزه را فراهم کند و در نهایت مسیری کمترین هزینه برای حرکت را انتخاب کند.

به‌عنوان نوآوری، الگوریتم Q-Learning با قابلیت یادگیری تقویتی، با شبکه‌های عصبی عمیق که توانایی تعمیم بر اساس داده‌های جدید را دارند، ترکیب‌شده‌اند. این ترکیب به پهپادها این امکان را می‌دهد تا به‌صورت هوشمندانه‌تر و با دقت بیشتر به مسیرهای بهینه در محیط‌های پیچیده راه یابند.

یادگیری تقویتی یک نوع از یادگیری ماشین است که در آن یک عامل (مانند پهپاد) با انجام اقدامات در محیط و دیدن نتایج آن‌ها، یاد می‌گیرد که چگونه رفتار کند. این عامل تلاش می‌کند تا با انجام اقدامات مناسب، پاداش بیشتری را کسب کند. الگوریتم‌های یادگیری تقویتی می‌توانند با استفاده از تجربیات خودشان و با تلاش در محیط، بهبود یابند.

شبکه‌های عصبی عمیق، یک نوع از مدل‌های یادگیری ماشین هستند که با استفاده از لایه‌های عصبی مصنوعی، قادر به تقریب توابع پیچیده‌تر هستند. این شبکه‌ها می‌توانند برای تخمین توابع هزینه و تابع پاداش در یادگیری تقویتی مورد استفاده قرار گیرند.

مراحل مدل‌سازی مسئله با استفاده از *Q-Learning* و شبکه‌های عصبی:

۱. حالت: هر نقطه در مسیر پهپادها در نظر گرفته‌می‌شود.

۲. عمل: هر حرکت از یک نقطه به نقطه دیگر به‌عنوان یک عمل در نظر گرفته‌می‌شود.

۳. جدول ارزش *(Q-table)*: برای هر پهپاد، برای هر استیت و هر عمل، یک مقدار ارزش در نظر گرفته‌می‌شود که توسط الگوریتم *Q-Learning* به‌روزرسانی می‌شود.

۴. شبکه‌های عصبی عمیق: برای تقریب توابع ارزش و تصمیم‌گیری بهتر، شبکه‌های عصبی باقابلیت تعمیم در نظر گرفته می‌شوند.

۱-۴- فرمول و مقادیر مسیریابی

Q-Value Update

$Q(s, a)=(1-\alpha).\ Q(s, a)+\alpha.(R+\gamma.max\ Q(\acute{s}+\acute{a}))$

$Q\ (s, a)$ ارزش حالت و عمل فعلی، R جایزه دریافتی بر اساس انجام عمل، γ فاکتور تخفیف و $Q(\acute{s}+\acute{a})$ بیشترین ارزش حالت و عمل بعدی می‌باشد.

۱. خروجی شبکه عصبی

از شبکه‌های عصبی برای تقریب توابع ارزش استفاده می‌شود که با استفاده از توابع فعال‌سازی، وزن‌ها و ورودی‌های مختلف تعیین می‌شود.

تابع هدف مسیریابی

$J(\theta)=\sum_{t=0}^{T}\ \sum_{i=1}^{N}\ \left(Q(s_t^i, a_t^i; \theta)-Q_{target}\right)$

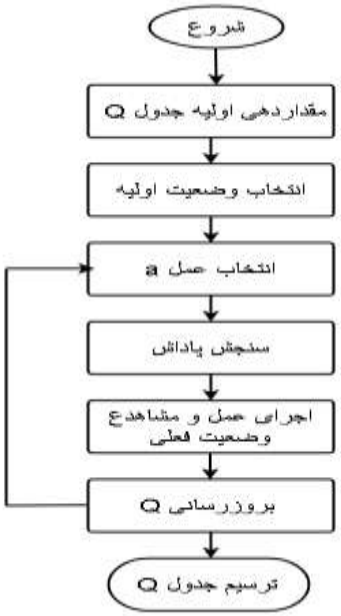
$J(\theta)$ تابع هدف، s_t^i, a_t^i حالت و عمل در زمان t برای پهپاد i، θپارامتر شبکه عصبی و Q_{target}ارزش هدف مورد نظر می‌باشد.

۲. آموزش و به‌روزرسانی

پس از هر حرکت، الگوریتم *Q-Learning* به‌روزرسانی می‌شود.

شبکه‌های عصبی نیز با استفاده از مقادیر به‌دست آمده از *Q-Learning* به‌روزرسانی می‌شوند.

این ترکیب از قابلیت یادگیری تقویتی *Q-Learning* و توانایی تعمیم شبکه‌های عصبی عمیق، باعث بهبود عملکرد پهپادها در مسیریابی در محیط‌های پیچیده شهری می‌شود.



شکل ۱. فلوچارت الگوریتم Q-Learning

۲-۴- الگوریتم ژنتیک

الگوریتم ژنتیک، الهام گرفته از فرایند تکاملی طبیعی، یک الگوریتم بهینه‌سازی است که در حل مسائل بهینه‌سازی و تصمیم گیری محدوده بالا و پیچیده به کار می‌رود. در این الگوریتم، جمعیتی از رشد یافته‌ها با هدف به‌دست آوردن یک جمعیت بهینه، ایجاد و تلاش برای بهینه‌سازی یک تابع هدف مشخص می‌کنند [۳۰-۲۶]. الگوریتم ژنتیک شامل مراحل گزینش، تولید فرزندان، عملگرهای ژنتیکی و ارزیابی انطباق است. در ساختار کلی الگوریتم ژنتیک پیش از هر چیز باید روش تبدیل هر پاسخ مسئله به‌صورت یک کروموزوم تعیین شود. سپس مجموعه‌ای از کروموزوم‌ها که بیان کننده مجموعه‌ای از جواب‌های مسئله هستند به‌عنوان جمعیت اولیه ایجاد می‌شوند. اندازه جمعیت دلخواه بوده و در ابتدای الگوریتم مشخص می‌گردد. همچنین طول هر کروموزوم بستگی به جواب‌های مورد نظر مسئله دارد بعد از این مرحله با به‌کارگیری اپراتورهای ژنتیک کروموزوم‌های جدید تولید می‌شوند. پس از تولید نسل جدید کروموزوم‌ها، از بین نسل قدیم و نسل جدید نسلی از بهترین کروموزوم‌ها انتخاب می‌شوند. معیار انتخاب مقدار برازندگی هر کروموزوم است و مقدار برازندگی هر کروموزوم نیز با توجه به مقدر تابع هدف مسئله تعیین می‌گردد. بعد از طی مراحل فوق، یک مرحله یا یک نسل از الگوریتم طی شده‌است. الگوریتم بعد از چندین نسل به‌سمت نقطه بهینه همگرا می‌شود. شرط توقف مسئله می‌تواند پیمودن تعداد معینی تکرار یا عدم‌تغییر در برازندگی بهترین کروموزوم بعد از چند تکرار و یا شرایط خاص دیگری باشد.

مراحل اصلی این الگوریتم عبارتند از:

• ایجاد جمعیت: در این مرحله، جمعیتی از راه حل‌های اولیه (همانند یک پرتو) برای مسئله بهینه‌سازی ساخته می‌شود.

• ارزیابی جمعیت: در این مرحله، هر راه حل در جمعیت با به‌کارگیری یک تابع ارزیابی ارزیابی می‌شود که معیاری برای خوشه‌ای شدن نتایج و اثبات کیفیت حل‌ها می‌باشد.

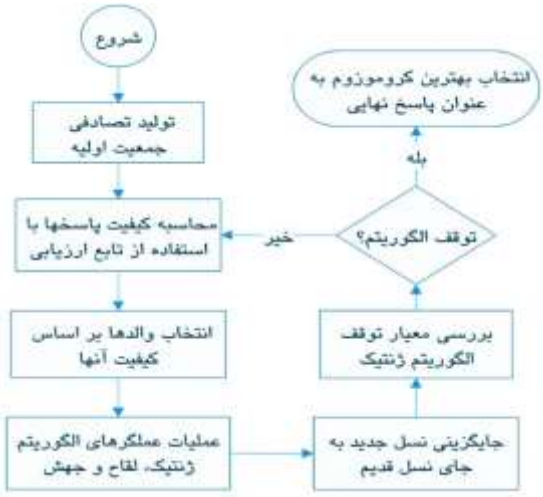
• انتخاب: در این مرحله، راه‌حل‌هایی که بهترین نتیجه را دارند، برای ایجاد نسل بعدی انتخاب می‌شوند. اگرچه خود یک راه حل بهترین نتیجه را در هر مرحله نمی‌دهد ولی برای بهتر شدن در جهت بهینه‌سازی می‌تواند مفید باشد.

• متقابل‌سازی: در این مرحله، با به‌کارگیری عملیات‌هایی همچون تغییر سرعت‌ها، الگوریتم از این قابلیت برخوردار است که بهترین نتایج را در نسل بعدی به‌دست‌آورد.

• جایگزینی: در این مرحله، نسل جدید به‌جای نسل پیشین قرار می‌گیرد و این کار تا زمانی ادامه می‌یابد که به یافتن بهترین راه‌حل‌ها برای مسئله بهینه‌سازی برسیم.

• شرط متوقف شدن: در این مرحله، با توجه به معیارهای مشخص‌شده، مانند تعداد نسل‌های تولید شده، هزینه زمانی و...، محاسبات متوقف می‌شود.

با توجه به مسئله بهینه‌سازی و نوع آن، ممکن است برخی مراحل در این الگوریتم دست‌کاری شوند یا از پیچیدگی خاصی برخوردار باشند.



شکل ۲. فلوچارت الگوریتم ژنتیک

۳-۴- الگوریتم مبتنی بر مورچگان

الگوریتم کلونی مبتنی‌بر مورچگان یا در حقیقت «بهینه سازی کلو نی مورچگان» (Ant Colony Optimization) همان‌گونه که از نام آن مشخص است، بر پایه رفتار طبیعی کلونی‌های مورچگان و مورچگان کارگر شاغل در آن‌ها بنا نهاده شده‌است. از این الگوریتم برای یافتن بهترین و کوتاه‌ترین مسیر بین چند مقصد استفاده می‌شود. فرآیند یافتن منابع غذایی در کلونی مورچگان بسیار بهینه است. زمانی که مورچه‌ها عملیات کاوش برای یافتن منابع غذایی را آغاز می‌کنند، به‌طور طبیعی یک مسیر «منطقی» و «بهینه» از آشیانه خود به منابع غذایی پیدا می‌کنند. به‌عبارت دیگر، جمعیت

مورچگان به‌نحوی همیشه قادر هستند تا یک مسیر بهینه را برای تأمین منابع غذایی مورد نیاز بیابند.

در این الگوریتم، مورچه‌ها در یک محیط مجازی حرکت می‌کنند و با توجه به فاکتورهایی مانند بوی ترکیبات شیمیایی و ارتفاع از مسیرهای مختلف عبور می‌کنند تا به یک نقطه بهینه برسند. این الگوریتم از تکنیک‌های تکراری برای بهبود پاسخ استفاده می‌کند و می‌تواند در حل مسائل پیچیده و چندبعدی مفید باشد. در این پژوهش، به بررسی و تحلیل روش حل مسائل با به‌کارگیری الگوریتم مورچگان پرداخته شده‌است.

مراحل حل مسئله به‌روش الگوریتم مورچگان به شرح زیر می‌باشد:

۱-آماده‌سازی

۲-برای هرکدام از مورچه‌ها این مراحل را دنبال می‌کنیم:

۱-۲-یک آغاز تصادفی انتخاب می‌کنیم.

۲-۲-تا تکمیل مسیر این مراحل را انجام می‌دهیم:

۱-۲-۲-یکی از شهرهای قابل انتخاب را انتخاب می‌کنیم.

۲-۲-۲-شهر انتخاب شده را حذف می‌کنیم.

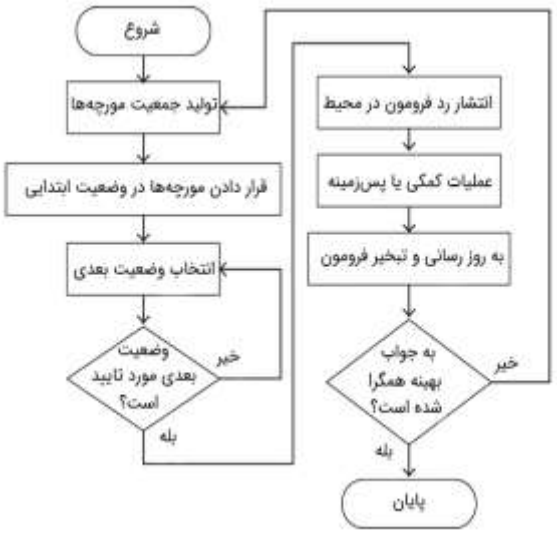
۳-۲-پس از ساخته شدن مسیر، آن را ارزیابی می‌کنیم.

۳-پس از آنکه تمام مورچه‌ها مسیرهای خود را طی کردند:

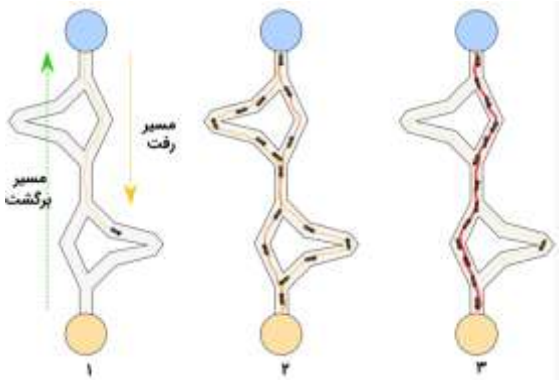
۱-۳-هر مورچه بر روی یال‌هایی که از آن‌ها عبور کرده است فرومون می‌ریزد.

۲-۳-فرمون تمام یال‌ها، در اثر تبخیر کمتر می‌شود.

۴-در صورت نیاز (برآورده شدن شرایط پایان) بروید به (۲).

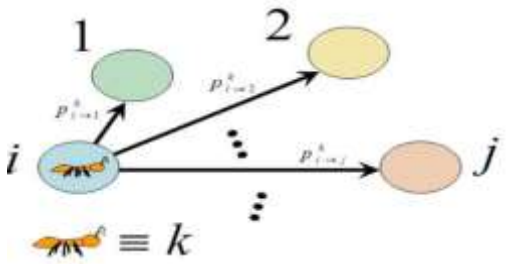


شکل ۳. فلوچارت الگوریتم مورچگان



شکل ۴. مسیر رفت و برگشت و انتخاب مسیر بهینه در الگوریتم مبتنی‌بر مورچگان

۱-۳-۴- مسیریابی و فرمولاسیون الگوریتم مبتنی بر مورچگان



شکل ۵. انتخاب یک مسیر دلخواه اولیه و حرکت مورچه‌ها از هر مسیر

۱. گراف مسئله

مجموعه گره‌ها $v = \{v_1, v_2, \dots, v_n\}$:

مجموعه یال‌ها $L = \{l_{ij} | \langle v_i, v_j \rangle \in W \subseteq V \times V\}$:

گراف $G = (V, L)$:

۲. توابع مهم

$J: L \rightarrow R \geq 0$

تابع هدف برای مسیر $J(l_{ij}) = d_{ij}$:J

رابطه فرمون با مسیر رفت و برگشت مورچه‌ها $\square_{ij} = \frac{1}{d_{ij}}$

تعریف مجموعه‌ها

$$P_{i \rightarrow k}^{k} = P_{ij}^{k} = \begin{cases} \frac{(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}}{\sum(\tau_{im})^{\alpha}(\eta_{im})^{\beta}}, j \in N_i^k \\ 0 \qquad\qquad j \notin N_i^k \end{cases}$$

$$N_i = \{m | l_{im} \in L\}$$

$$N_i^k = \{m | m \in N, \wedge m \notin \psi^k\} = N_i - \psi^k$$

N_i :مجموع همسایگان گره V_i

N_i^k: مجموعه همسایگان گره V_i در مرحله k

فرآیند تصمیم‌گیری

P_{ij}^k: احتمال انتخاب مسیر J_{ij} در مرحله k توسط مورچگان.

ψ^k: مسیرهایی که در مرحلهk انتخاب‌شده‌اند.

۳. تغییر فرومون

$$\tau_{ij} \longleftarrow \tau_{ij} + \sum_k \Delta\tau_{ij}^k$$

در اینجا، τ_{ij} مقدار فرمون برای یال J_{ij} است. مقدار جمع شونده نشان‌دهنده تغییرات مقدار فرمون در هر مرحلهk می‌باشد. به‌عبارت دیگر، مقدار فرمون برای یال J_{ij} با توجه به تغییرات جمعی در هر مرحله به‌روزرسانی می‌شود.

$$\tau_{ij} \longleftarrow (1-\rho).\tau_{ij}$$

در اینجا، ρیک ضریب کاهش مقدار فرمون است. این فرمول نشان‌دهنده کاهش تدریجی مقدار فرمون در هر مرحله بدون تحرک مورچگان است.

این دو فرمول در جلبه بهینه‌سازی فرآیند گرفتگی و به‌روزرسانی اطلاعات در الگوریتم مبتنی بر مورچگان تأثیرگذار هستند.

۴-۴- الگوریتم مبتنی‌بر اردحام جمعیت

الگوریتم بهینه‌سازی ازدحام ذرات (PSO) یک روش محاسباتی است که از رفتار گروهی در طبیعت الهام گرفته شده‌است. این یک الگوریتم بهینه‌سازی فرا ابتکاری است که از جمعیتی از ذرات برای جستجوی راه‌حل بهینه در یک فضای مسئله معین استفاده می‌کند. در PSO، ذرات بر پایه موقعیت‌های خود و شناخته‌شده‌ی دسته ازدحام در فضای مشکل حرکت می‌کنند. این الگوریتم سرعت و موقعیت هر ذره را در هر تکرار تا زمانی که یک معیار توقف برآورده شود، به‌روز می‌کند. PSO با موفقیت در زمینه‌های مختلف از جمله مهندسی، اقتصاد و علوم کامپیوتر استفاده شده‌است.

PSO از جمله الگوریتم‌های جستجوی موازی مبتنی بر جمعیت است که با یک گروه از جواب‌های تصادفی (ذره‌ها) شروع به کار می‌کند، سپس برای یافتن پاسخ بهینه در فضای مسئله با به‌روز کردن مکان ذره‌ها به جستجو ادامه می‌دهد. هر ذره به‌صورت چند بعدی (بسته به نوع مسئله) با دو بردار Xid و Vid که به‌ترتیب معرف موقعیت مکانی و سرعت بعد dام از i امین ذره هستند، مشخص می‌شود. در هر مرحله از حرکت جمعیت، مکان هر ذره با دو مقدار بهترین به‌روز می‌شود. اولین مقدار، بهترین تجربه‌ای است که خود ذره تاکنون به‌دست آورده است و با p_best نشان داده‌می‌شود. دومین مقدار، بهترین تجربه‌ای است که در بین تمامی ذره‌ها به‌دست آمده است و با g_best نشان داده‌می‌شود. در برخی ویرایش‌ها، PSOذره قسمت‌هایی از جمعیت را که همسایگان توپولوژیکی‌اش هستند، انتخاب می‌کند و تنها آن‌ها را در اعمال خود دخیل می‌کند. در این صورت بهترین راه حل محلی استفاده می‌شود و ازL_best به جایg_best استفاده می‌شود. شرط توقف مسئله می‌تواند طی کردن تعداد معینی تکرار یا عدم‌تغییر در بهترین راه حل بعد از چند تکرار و یا شرایط خاص دیگری باشد. توانایی الگوریتم برای یافتن سریع بهینه جهانی و اجرای ساده آن، آن را به گزینه‌ای محبوب برای حل مسائل پیچیده بهینه سازی تبدیل کرده است.

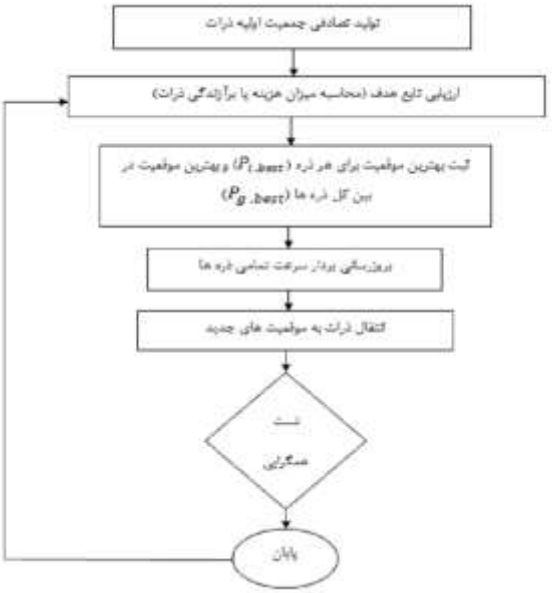
هر ذره در ازدحام دارای یک موقعیت و یک سرعت است که در هر تکرار بر پایه شناخته شده‌ترین موقعیت ذره و شناخته شده‌ترین موقعیت ازدحام به‌روز می‌شود. شناخته شده‌ترین موقعیت ازدحام، موقعیتی است که تاکنون بهترین راه حل توسط هر ذره‌ای در ازدحام پیدا شده‌است. سرعت ذره طبق فرمولی به‌روز می‌شود که تمایل ذره برای حرکت به‌سمت شناخته شده‌ترین موقعیت خود و شناخته شده‌ترین موقعیت ازدحام را متعادل می‌کند. همان گونه که ذرات در فضای مسئله حرکت می‌کنند، مناطق مختلف فضای حل را کاوش می‌کنند و الگوریتم به‌سمت راه حل بهینه همگرا می‌شود. معیار توقف معمولاً بر پایه تعداد تکرارها یا همگرایی ازدحام به‌سمت یک راه حل پایدار است.

روند پیاده‌سازی ترکیب الگوریتم‌های PSO و GA به‌گونه‌ای است که پس از آماده کردن تمام شرایط اولیه مربوط به هر دو الگوریتم، در حلقه اصلی کدهای هر الگوریتم به‌صورت نسبتاً یکسان اما با ترتیب‌های متفاوت اجرا می‌شود. یعنی برای حالت GA _PSO ابتدا عملگرهای PSO اجرا می‌شوند و سپس عملگرهای GA و برای حالت GA_PSO بالعکس.

- در ابتدا ذرات به‌صورت تصادفی در سرتاسر فضای جست‌وجو مقدار دهی می‌شوند که این موفقیت‌های اولیه

به‌عنوان بهترین تجربه شخصی ذرات نیز شناخته می‌شوند.

- در گام بعد بهترین ذره از میان ذرات موجود انتخاب شده و به‌عنوان بهترین پاسخ شناخته می‌شوند.
- سپس گروه ذرات در فضای جست‌وجو حرکت می‌نمایند تا زمانی که شرایط پایان محقق گردد. این حرکت شامل اعمال معادله سرعت به گروه ذرات می‌باشد که موفقیت‌های هر ذره بر پایه آن تغییر می‌کند.
- مقدار برازش جدید حاصل از ذره با مقدار ذره مقایسه می‌شود. در حالتی که موقعیت جدید دارای برازش بهتری باشد این موقعیت جدید جایگزین موقعیت می‌شود.
- روالی مشابه نیز برای انجام می‌شود.
- شرایط پایان: معمولاً تعداد دفعات تکرار، که الگوریتم به اجرا در می‌آید را به‌عنوان معیار اصلی شرط پایان در نظر گرفته‌می‌شود.
- تعداد ذرات: تعداد کل ذراتی که در فضای جست‌وجو حرکت می‌کنند.



شکل ۶. فلوچارت الگوریتم مبتنی‌بر اردحام جمعیت

شکل ۷. حرکت گروهی پرندگان و مفهوم الگوریتم PSO

۵-۴- الگوریتم تکمل تفاضلی

الگوریتم تکامل دیفرانسیل یا الگوریتم تکاملی تفاضلی (DE) نخستین‌بار در سال ۱۹۹۵ توسط استورن و پرایس معرفی شد. این دو نشان دادند که این الگوریتم توانایی خوبی در بهینه سازی توابع غیرخطی مشتق ناپذیر دارد که به‌عنوان روشی قدرتمند و سریع برای مسائل بهینه سازی در فضاهای پیوسته معرفی شده‌است. الگوریتم (DE) جهت غلبه بر عیب اصلی الگوریتم ژنتیک، یعنی فقدان جستجوی محلی در این الگوریتم ارائه شده‌است، تفاوت اصلی بین الگوریتم‌های ژنتیکی و الگوریتم (DE) در عملگر انتخاب selection operators می‌باشد. در اپراتور انتخاب GA، شانس انتخاب یک پاسخ به‌عنوان یکی از والدین وابسته به مقدار شایستگی آن می‌باشد، اما در الگوریتم DE همه جواب‌ها دارای شانس مساوی جهت انتخاب شدن می‌باشند. یعنی شانس انتخاب شدن آنها وابسته به مقدار شایستگی آنها نمی‌باشد، پس از اینکه یک پاسخ جدید با به‌کارگیری یک اپراتور جهش خود-تنظیم و اپراتور crossover تولید شد، پاسخ جدید با مقدار قبلی مقایسه می‌شود و در صورت بهتر بودن جایگزین می‌گردد. در این الگوریتم بر خلاف دیگر الگوریتم‌ها که اول عملگر crossover و سپس عملگر mutation انجام‌می‌شود به‌گونه‌ای که ابتدا عملگر جهش اعمال شده و سپس عملگر تقاطع اعمال می‌شود تا بدین وسیله نسل جدید ایجاد گردد. برای اعمال عملگر mutation از پخش خاصی استفاده نمی‌شود بلکه طول گام جهش برابر با

مقدار از فاصله میان اعضای فعلی تعیین می‌شود. یکی از روش‌های محاسبه توابع حقیقی (Real value) با به کارگیری استراتژی‌های تکاملی است. روند تکامل در این الگوریتم مبتنی بر ایجاد بهبود تدریجی و مستمر در حدس اولیه (پاسخ کاندید) بوده و طبق اصول تمامی الگوریتم‌های رده تکاملی، به یک تابع برازندگی (Fitness function) جهت مقایسه پاسخ‌ها نیاز داریم. نقطه قوت الگوریتم DE در مقایسه با روش‌های حل معادلات حقیقی دیگر (مانند روش‌های نیوتن)، عدم نیاز آن به گرادیان یا شیب تابع است. در نتیجه با به‌کارگیری این الگوریتم، بدون وجود هر گونه اطلاعاتی در مورد نوع تابع می‌توان به محاسبه یک پاسخ نسبتاً بهینه برای انواع توابع چند بعدی پیوسته / غیر پیوسته، متغیر زمانی و نامنظم امیدوار بود. مراحل الگوریتمDifferential Evolution به‌صورت زیر است: تولید جمعیت اولیه از پاسخ‌های کاندید. هر پاسخ کاندید، یک بردار از اعداد حقیقی به تعداد ابعاد مسئله (پارامترهای مجهول) می‌باشد. به‌ازای هر پاسخ کاندید X، سه پاسخ متمایز a، b، c را از جمعیت انتخاب می‌نماییم. تعیین پارامتر تصادفی R، در محدوده ۱ و ابعاد مسئله. محاسبه پاسخ بهبود یافته Y به این صورت که به‌ازای هر بعد (i) X از X، در صورت برابر بودن R با i و یا برآورده شدن احتمال ترکیب (i) p، از فرمول برآورد Y (i) = a (i) + F* (b (i) – c (i)) برای محاسبه بعد (i) y استفاده می‌نماییم. وگرنه خود (i) X به (i) Y انتصاب داده‌می‌شود. پذیرش پاسخ جدید Y، چنانچه برازندگی آن از X بیشتر باشد. تکرار مراحل ۲ تا ۵ تا زمان تحقق شرط پایان. لازم به ذکر است که پارامتر CR همان احتمال ترکیب است (مانند ژنتیک) و (i) p نیز شانس تحقق ترکیب برای هر بعد از پاسخ می‌باشد. مقدار F نیز یک مقدار درست و ثابت می‌باشد که با توجه به نوع مسئله انتخاب می‌گردد. (البته شیوه انتخاب مقداری مناسب برای F خود ماجرایی دارد)

در این الگوریتم، هر بردار را می‌توان به‌صورت یک نقطه در فضای چند بعدی در نظر گرفت. جواب بهینه به‌عنوان یک نقطه در این فضا تعریف می‌شود. برای به‌دست آوردن جواب بهینه، یک مجموعه از بردارها ایجاد می‌شود و سپس با اعمال تغییرات مناسب به این بردارها، به‌دنبال جواب بهینه می‌گردیم. در این الگوریتم، به هر بردار یک شماره تصادفی نسبت داده‌می‌شود و سپس سه بردار با این شماره‌ها را انتخاب می‌کنیم. بعد از این کار، تفاضل هر دوی این بردارهای انتخاب شده به‌دست می‌آید و با اعمال ضرایب تصادفی به تفاضل‌ها، یک برداری تولید می‌شود. سپس با به‌کارگیری این بردار تولید شده، یک بردار جدید به‌صورت زیر به‌دست می‌آید:

$$u_i = x_{r1} + F \times (x_{r2} - x_{r3})$$

در این فرمول، x_{r1} ، x_{r2} و x_{r3} نام برداری‌هایی هستند که با شماره‌های تصادفی r_1 ، r_2 وr_3 انتخاب‌شده‌اند. همچنین F یک ضریب تصادفی است که مقدار آن بین ۰ و ۲ قرار می‌گیرد.

سپس با مقایسه بردار جدید با بردار قدیمی و انتخاب بهترین بین آن‌ها، به‌دنبال جواب بهینه می‌گردیم. با به کارگیری این الگوریتم، می‌توان به‌راحتی و با سرعت بالا به جواب بهینه در مسائل پیچیده رسید.

مراحل حل مسئله به‌روش الگوریتم تکامل تفاضلی به شرح زیر می‌باشد:

۱. ابتدا باید یک جمعیت اولیه از حلقه‌های محتمل به صورت تصادفی انتخاب شود. (مرحله ۱)

۲. سپس برای هر حلقه، یک جفت دیگر از حلقه‌های انتخاب شده به‌صورت تصادفی انتخاب می‌شود. (مرحله ۲)

۳. با به‌کارگیری فرمول تفاضل بین دو حلقه منتخب، یک حلقه جدید به‌صورت تصادفی تولید می‌شود. (مرحله ۳)

۴. بررسی می‌شود که آیا حلقه جدید بهترین حلقه‌ایست که تاکنون دیده شده‌است یا نه.

۵. چنانچه حلقه جدید بهترین حلقه‌ای باشد، آن را به عنوان حلقه بهترین تاکنون انتخاب می‌کنیم.

۶. تکرار مراحل ۲ تا ۵ تا زمانی که شرایط پایان الگوریتم تفاضلی تفاضلی برآورده شود. (مرحله ۶)

در نتیجه، مراحل حل الگوریتم تفاضلی تفاضلی به‌صورت زیر است:

۱. انتخاب جمعیت اولیه

۲. انتخاب جفت‌های حلقه‌ها

۳. تولید حلقه جدید

۴. بررسی بهترین حلقه

۵. انتخاب بهترین حلقه جدید

۶. تکرار مراحل ۲ تا ۵ تا زمان پایان الگوریتم.

شکل ۸. فلوچارت الگوریتم تکامل-تفاضلی

۶- کاربرد الگوریتم‌های مسیریابی چند پهپاد در یک مسئله ویژه

در حوزه‌ی جستجو و نجات، استفاده از پهپادها به‌عنوان وسیله‌هایی پویا و کارآمد برای ارائه خدمات سریع و مؤثر جلب توجه کمک‌های اضطراری است. یکی از چالش‌های اساسی در این حوزه، مسئله مسیریابی پهپادها به‌نحوی است که در شرایط شهری پیچیده با موانع متعدد و متغیر، پهپادها به‌صورت هوشمند و بهینه به هدف خود برسند.

الگوریتم‌های مسیریابی چند پهپاد از قابلیت حل مسائل جستجوی چندهدفه با محدودیت بهره‌می‌برند. این الگوریتم‌ها، پهپادها را قادر می‌سازند تا به‌طور همزمان به چندین هدف دست یابند، در حالی‌که از موانع و محدودیت‌های محیط جلوگیری می‌کنند. در اینجا، یک مسئله جستجوی چندهدفه با محدودیت مورد بررسی قرار می‌گیرد. در ادامه مسئله نمونه ذکر شده با به‌کارگیری الگوریتم‌های معرفی شده حل خواهد شد و نتایج در پایان با یکدیگر مقایسه خواهد شد تا بهترین و کارآمدترین الگوریتم برای مسیریابی پهپادها انتخاب شود.

۶-۱- حل مسئله با الگوریتم Q-Learning

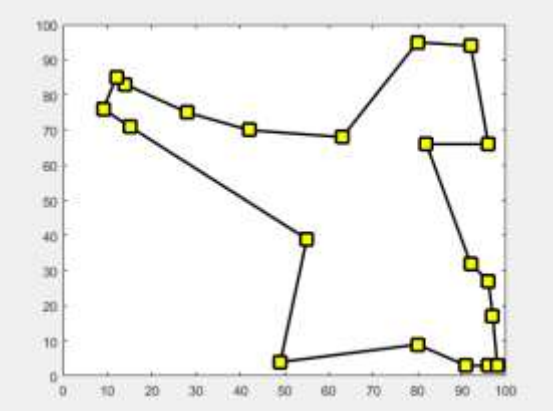
در جدول شماره ۱ نتایج حاصل از حل مسئله نمونه با الگوریتم Q learning ارائه شده‌است:

جدول ۱. نتایج حاصل از الگوریتم Q-Learning

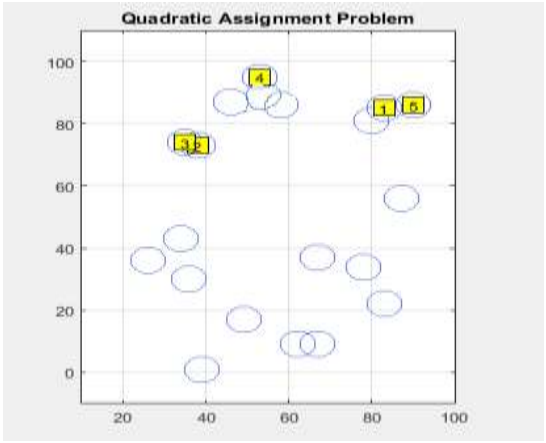
تعداد پهپاد	**۵**
تعداد تکرار	۵۰۰
بهترین تابع هزینه	4/99
زمان	1/2808 s
درصد موفقیت	99%
تعداد حالت‌ها	5^4
تعداد اعمال	25^5

۶-۲- حل مسئله با الگوریتم مبتنی‌بر مورچگان

جدول شماره ۲ نتایج حاصل از حل مسئله نمونه با استفاده از الگوریتم مبتنی بر مورچگان را نمایش می‌دهد. شکل شماره ۶ نمودار خروجی یک پهپاد را برای عبور از ۲۰ نقطه تصویر می‌کشد. در شکل ۷، محل قرارگیری ۵ پهپاد در نقاط پایانی را مشاهده می‌کنیم. همچنین، شکل شماره ۸ نمایانگر همگرایی نمودار در رسیدن به بهترین جواب بهینه است.

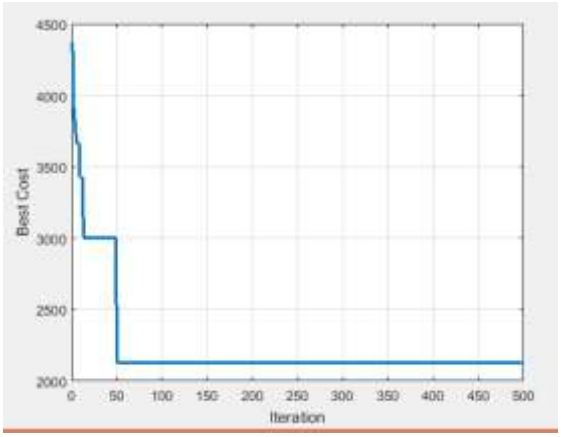


شکل ۹. نمودار خروجی الگوریتم مبتنی‌بر مورچگان یک پهپاد برای عبور از ۲۰ نقطه



شکل ۱۰. محل قرار گرفتن ۵ پهپاد در هر نقطه نهایی

این شکل، بهترین و کوتاه‌ترین مسیر را برای عبور از تمام شهرها در یک دوره کامل نشان‌می‌دهد، به‌گونه‌ای که هر شهر فقط یک‌بار طی می‌شود و هیچ شهری دوبار عبور نمی‌شود.



شکل ۱۱. نمودار خروجی الگوریتم مبتنی بر مورچگان در رسیدن به بهترین جواب بهینه

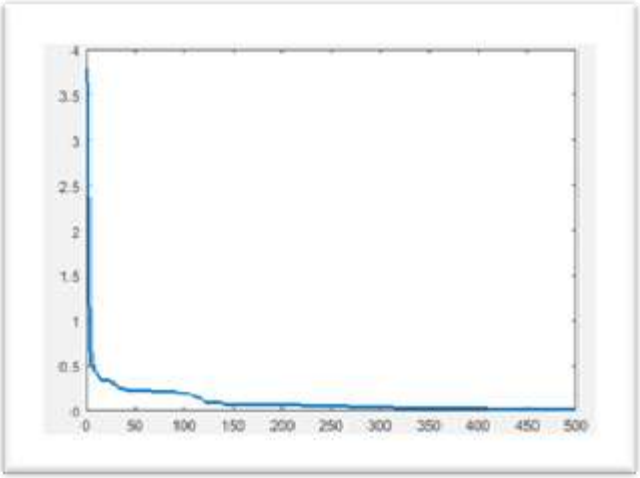
شکل ۱۱، جواب‌های بهینه را به تصویر می‌کشد. هرچه نمودار به همگرایی بیشتری برسد، نشان‌دهنده نزدیک شدن به مسیری است که بهینه‌ترین راه برای رسیدن به هدف را ارائه می‌دهد.

جدول ۲. نتایج حاصل از الگوریتم مبتنی بر مورچگان

تعداد جمعیت اولیه	500
تعداد پهپاد	5
طول مسیر بهینه	[1 ,16,4,15,19]
Best cost	3/620380
X	[67 80 62 34 54 36 53 46 39 35 83 58 87 90 83 38 26 78 49 67]
Y	[9 81 9 43 89 30 95 87 1 74 85 86 56 86 22 73 36 34 17 37]
Tour	[2 20 12 10 13 18 1 9 4 19 5 17 7 16 3 6 11 8 14 15]
زمان	1/327894 s

۶-۳- حل مسئله با الگوریتم مبتنی‌بر ازدحام جمعیت

در شکل شماره ۱۲ نمودار خروجی الگوریتم ازدحام جمعیت و در جدول شماره ۳ نتایج حاصل از حل مسئله نمونه با الگوریتم ازدحام جمعیت ارائه شده‌است:



شکل ۱۲. نمودار خروجی الگوریتم مبتنی بر ازدحام جمعیت در رسیدن به بهترین جواب بهینه

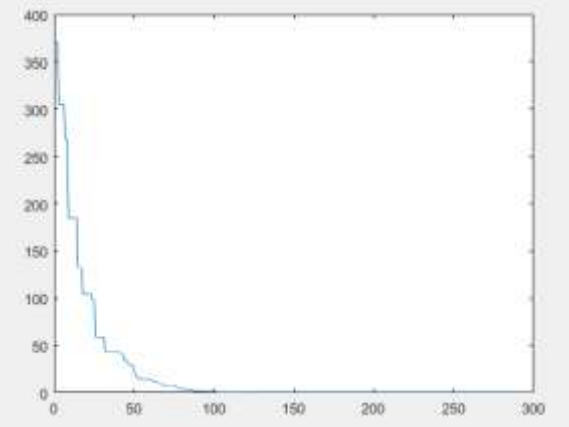
در شکل ۱۲ جواب‌های بهینه را نشان‌می‌دهد، که هرچی نمودار همگرا شود یعنی به بهینه‌ترین مسیر برای رسیدن به هدف نزدیک شده‌ایم.

جدول ۳. نتایج حاصل از الگوریتم مبتنی بر ازدحام جمعیت

تعداد پهپاد	۵
تعداد ایستگاه‌های زمینی	۱
برد بیسیم	۱۰۰m
بهترین هزینه	2/9998
مسیر	[43/9923,52/6449,54/1567,44/1976]
زمان	1/168119 s

۶-۴- حل مسئله با الگوریتم تکامل تفاضلی

شکل شماره ۱۳، خروجی الگوریتم ازدحام جمعیت را به تصویر می‌کشد. همچنین، جدول شماره ۴ نتایج حاصل از حل مسئله نمونه با استفاده از الگوریتم تکامل تفاضلی را ارائه می‌دهد.



شکل ۱۳. نمودار خروجی الگوریتم تکامل تفاضلی در رسیدن به بهترین جواب بهینه

این نمودار نشان‌می‌دهد که در ابتدا، الگوریتم تکامل تفاضلی کمی دیرتر نسبت به الگوریتم مورچگان و ازدحام جمعیت به همگرایی می‌رسد. اما پس از گذشت یک دوره کوتاه، سرعت همگرایی آن بیشتر می‌شود و به‌سرعت به جواب بهینه می‌رسد.

خلاصه نتایج حاصل از الگوریتم‌های مورچگان و ازدحام جمعیت و تکامل تفاضلی در جدول زیر آورده شده‌است.

جدول ۴. نتایج حاصل از الگوریتم تکامل-تفاضلی

تعداد پهپاد	5
تعداد تکرار	500
بهترین تابع هزینه	1/4930
بهترین موقعیت	[1.47,5.28,2.83,2.05,2.74]
زمان	2/364388 s

در ادامه در جدول‌های زیر الگوریتم‌ها با یکدیگر براساس عملکرد و مزایا نسبت به دیگری مقایسه خواهد شد:

جدول ۵. مقایسه الگوریتم‌ها با یکدیگر

معیار	Q-Learning	مبتنی بر ACO	مبتنی بر pso	DE
بهترین تابع هزینه	۴,۹۹	۳,۶۲	۲,۸۸	۱,۱۹
زمان مورد نیاز برای حل (ثانیه)	۱,۲۸۰	۱,۳۲۷	۱,۱۶۸	۱,۶۶۰
تعداد پهپادها	۵	۵	۵	۵
تعداد تکرار حلقه	۵۰۰	۵۰۰	۵۰۰	۵۰۰

جدول ۶. مقایسه الگوریتم‌های q-learning، مورچگان، ازدحام جمعیت و تکامل-تفاضلی با یکدیگر

معیار	Q-learning	مبتنی‌بر ACO	مبتنی‌بر PSO	DE
نوع الگوریتم	یادگیری تقویتی	شبیه‌سازی رفتار مورچگان	مبتنی‌بر جمعیت	مبتنی‌بر تکامل
محیط	غیرقطعی	قطعی	غیرقطعی	قطعی یا غیرقطعی
روش کار	آزمایش و خطا	مواد شیمیایی	رقابت	تکامل طبیعتی
مزایا	انعطاف‌پذیر، سرعت بالا	ساده، کارآمد	بهینه‌سازی جمعیت	قدرتمند، انعطاف‌پذیر
معایب	ممکن است به زمان زیادی نیاز داشته باشد.	ممکن است بهینه نباشد.	ممکن است بهینه نباشد.	ممکن است کند باشد.

جدول ۷. مقایسه الگوریتم‌های مبتنی بر مورچگان، مبتنی بر ازدحام جمعیت و Q-Learning با یکدیگر

معیار	Q-Learning	مبتنی بر ACO	مبتنی بر PSO
نوع الگوریتم	یادگیری تقویتی	تکاملی	تکاملی
نیاز به مدل از محیط	خیر	بله	بله
زمان اجرا	طولانی	کوتاه	کوتاه
احتمال رسیدن به بهترین مسیر	بالا	بله	متوسط
کارآمدی	بالا	بالا	متوسط

۷- نتیجه‌گیری

با توجه به نتایج به‌دست آمده در شبیه‌سازی‌ها، می‌توان نتیجه گرفت که الگوریتم مبتنی بر مورچگان و Q-Learning بهترین عملکرد را در مسیریابی شبکه‌ای پهپادها در جست و جو و نجات دارد. این الگوریتم‌ها با توجه به قابلیت خود در بهینه‌سازی مسیرها و کاهش زمان نجات، بهترین گزینه برای استفاده در این نوع مأموریت‌ها است. میانگین هزینه و میانگین تعداد حالت‌های بررسی‌شده الگوریتم Q-learning کمتر از سایر الگوریتم‌ها است. این بدان معناست که این الگوریتم به‌احتمال بیشتری به بهترین مسیر دست می‌یابد و زمان کمتری را صرف می‌کند.

الگوریتم مبتنی بر مورچگان با استفاده از قابلیت پخش فرمون در محیط، عملکرد و دقت بالاتری در یافتن بهترین و کوتاه‌ترین مسیر دارد، با این حال، زمان بیشتری را صرف می‌کند.

الگوریتم‌های مبتنی بر ازدحام جمعیت و تکامل-تفاضلی عملکرد ضعیف‌تری دارند و میانگین هزینه و تعداد حالت‌های بررسی‌شده آنها بیشتر از سایر الگوریتم‌ها است. این نشان می‌دهد که این الگوریتم‌ها با احتمال کمتری به بهترین مسیر می‌رسند و زمان بیشتری را صرف می‌کنند. الگوریتم Q-learning یک الگوریتم قدرتمند یادگیری ماشین است که می‌تواند برای مسیریابی شبکه‌ای پهپادها در جستجو و نجات مورد استفاده قرار گیرد. این الگوریتم با نقاط قوت خاص خود، برای این کاربرد مناسب است. با انجام تحقیقات بیشتر، می‌توان عملکرد الگوریتم Q-learning را در این کاربرد بهبود بخشید.

منابع

1.Mirjalili, Seyedali, and Seyed Mohammad Mirjalili. "A comparative study of the ant lion optimizer and its variants for global optimization problems." Applied Intelligence 48.8 (2018): 2450-2477.

2. Dasgupta, S., and A. Roy. Recent advances in swarm intelligence and evolutionary computation: Theory and applications. Springer, 2016.

3. Das, S., S. S. Mullick, and B. K. Roy. "An efficient hybrid algorithm based on ant colony optimization and differential evolution for global optimization problems." Applied Soft Computing 53 (2017): 170-188.

4. Wang, G. G., S. Deb, and L. D. Coelho. "Hybridizing antlion optimizer with differential evolution for constrained numerical and engineering optimization problems." Neural Computing and Applications 31.5 (2019): 1455-1493.

5. Rashedi, E., H. Nezamabadi-pour, and S. Saryazdi. "GSA: a gravitational search algorithm." Information sciences 179.13 (2009): 2232-2248.

6. Zheng, J., Y. Liang, and B. Qu. "An enhanced ant colony optimization algorithm for solving continuous optimization problems." Applied mathematics and computation 181.2 (2006): 915-927.

7. Mallipeddi, R., P. N. Suganthan, and Q. K. Pan. "Differential evolution algorithm with ensemble of parameters and mutation strategies." Applied Soft Computing 10.2 (2010): 520-532.

8. Wang, G. G., and S. Deb. "Multi-objective ant lion optimizer for unconstrained and constrained optimization problems." Applied Soft Computing 77 (2019): 272-308.

9.Rahnamayan, S., H. R. Tizhoosh, and M. M. A. Salama. "Opposition-based differential evolution." IEEE transactions on evolutionary computation 12.1 (2008): 64-79.

10. Liu, B., L. Wang, and X. Zhang. "A hybrid artificial bee colony algorithm with modified search mechanism for global optimization." Computers & Operations Research 46 (2014): 29-39.

11.Wang, G. G., and S. Deb. "Antlion optimizer with crossover operation for numerical and engineering optimization problems." International Journal of Machine Learning and Cybernetics 10.5 (2019): 1053-1080.

12.Mirjalili, S., and A. Lewis. "The whale optimization algorithm." Advances in engineering software 95 (2016): 51-67.

13.Kennedy, J., and R. C. Eberhart. "Particle swarm optimization." Proceedings of IEEE international conference on neural networks (1995): 1942-1948.

14.Shi, Y., and R. C. Eberhart. "A modified particle swarm optimizer." Evolutionary Computation Proceedings (1998): 69-73.

15.Liang, J. J., A. K. Qin, P. N. Suganthan, and S. Baskar. "Comprehensive learning particle swarm optimizer for global optimization of

multimodal functions." IEEE Transactions on Evolutionary Computation 10.3 (2006): 281-295.
16.Das, S., and P. N. Suganthan. "Differential evolution: a survey of the state-of-the-art." IEEE transactions on evolutionary computation 15.1 (2011): 4-31.
17.Das, S., S. Biswas, and A. Roy. "A review of differential evolution algorithm and its variants for global optimization." Swarm and Evolutionary Computation 34 (2017): 1-22.
18.Das, S., and A. Abraham. "Differential evolution: a survey of the state-of-the-art." IEEE transactions on evolutionary computation 15.1 (2011): 4-31.
19.Das, S., & Mullick, S. S. (2019). A comprehensive survey of ant colony optimization algorithms and their applications. Swarm and Evolutionary Computation, 46, 54-89.
20.Hosseini, S. H., & Farsangi, M. A. (2018). An efficient hybrid algorithm based on particle swarm optimization and differential evolution. Journal of Computational Science, 28, 25-37.
21.Sonny, A., Yeduri, S. R., & Cenkeramaddi, L. R. (2023). Autonomous UAV Path Planning using Modified PSO for UAV-Assisted Wireless Networks. IEEE Access.
22.Din, A. F. U., et al. (2023). Robust flight control system design of a fixed wing UAV using optimal dynamic programming. Soft Computing, 27(6), 3053-3064.
23.Salameh, H. B., Alhafnawi, M., Masadeh, A. E., & Jararweh, Y. (2023). Federated reinforcement learning approach for detecting uncertain deceptive target using autonomous dual UAV system. Information Processing & Management, 60(2), 103149.
24.Wu, J., et al. (2023). An Adaptive Conversion Speed Q-Learning Algorithm for Search and Rescue UAV Path Planning in Unknown Environments. IEEE Transactions on Vehicular Technology.
25.Parvaresh, N., & Kantarci, B. (2023). A Continuous Actor-Critic Deep Q-Learning-Enabled Deployment of UAV Base Stations: Toward 6G Small Cells in the Skies of Smart Cities. IEEE Open Journal of the Communications Society, 4, 700-712.
26.Muy, S., & Lee, J.-R. (2023). Joint optimization of trajectory, beamforming, and power allocation in UAV-enabled WPT networks using DRL combined with water-filling algorithm. Vehicular Communications, 43, 100632.
27.Li, B., et al. (2023). Robust Computation Offloading and Trajectory Optimization for Multi-UAV-Assisted MEC: A Multi-Agent DRL Approach. IEEE Internet of Things Journal.
28.Trotti, F., Farinelli, A., & Muradore, R. (2023). An online path planner based on POMDP for UAVs. In 2023 European Control Conference (ECC) (pp. 1471-1476). IEEE.
29.Chen, P., Li, H., & Ma, L. (2023). Distributed massive UAV jamming optimization algorithm with artificial bee colony. IET Communications, 17(2), 197-206.
30.Khashan, M. K., Khudhur, D. S., & Balla, H. H. (2023). Comparison between the two methods of optimization: Genetic algorithm (GA) and ant colony algorithm (ACO) for the propulsion system of UAV.