

طراحی مسیر یاب‌های تکاملی آشوب‌مبنا به منظور تولید خطوط سیر UAS خودکار

علی اصغر حیدری^{۱*}، رحیم علی عباسپور^۲

^۱ دانشجوی کارشناسی ارشد سیستم‌های اطلاعات مکانی - دانشکده مهندسی نقشه برداری و اطلاعات مکانی - پردیس

دانشکده‌های فنی - دانشگاه تهران

as_heidari@ut.ac.ir

^۲ استادیار دانشکده مهندسی نقشه برداری و اطلاعات مکانی - پردیس دانشکده‌های فنی - دانشگاه تهران

abaspour@ut.ac.ir

(تاریخ دریافت آذر ۱۳۹۳، تاریخ تصویب خرداد ۱۳۹۴)

چکیده

تاکنون الگوریتم‌های گوناگونی جهت تکمیل و ارتقای زیرساخت سامانه‌های ناوبری خودکار سیستم‌های پرنده بدون سرنشین ارائه شده است. با این حال، کوشش‌های اندکی در طراحی مسیر یاب‌های آشوب بنیاد به منظور تعیین خط سیر بهینه این سیستم‌ها در سناریوهای شهری انجام گردیده است. در این مقاله، مسیر یاب پیشنهادی به گونه‌ای پیاده‌سازی می‌شود که با در نظر گرفتن قیود مأموریت نظیر زوایای چرخش و پارامترهای دینامیکی پرواز، نواحی ممنوعه، حدود نقشه و ارتفاع ایمن، پارامترهایی شامل ارتفاع پرواز، طول مسیر و میزان مصرف انرژی را کمینه نماید. بدین انگیزه، نخست یک مدل جامع جهت توصیف مسئله مسیر یابی سکوا ارائه گردیده و سپس بر پایه تلفیق تئوری آشوب و محاسبات تکاملی، چهار الگوریتم تکاملی آشوب مبنای جدید پیشنهاد می‌گردد. در ادامه، تحلیل و ارزیابی جامع کارکرد الگوریتم‌های ارائه شده در مسئله مسیر یابی بر پایه نرخ موفقیت، دقت و کیفیت پاسخ‌ها، زمان اجرا و سرعت همگرایی انجام می‌گردد. ارزیابی نتایج مبین کسب برترین نتایج با به کارگیری الگوریتم تکامل تفاضلی با سیگنال آشوبی لجستیک می‌باشد.

واژگان کلیدی: مسیر یابی، سیستم‌های پرنده بدون سرنشین، تئوری آشوب، محاسبات تکاملی، تکامل تفاضلی، رقابت استعماری،

توده ذرات، زنبور عسل مصنوعی

۱- مقدمه

امروزه به‌کارگیری سیستم‌های پرنده بدون سرنشین (UAS) به‌عنوان سکوی حامل سنجنده‌های چندمنظوره در دامنه‌های عمرانی گوناگونی همانند مدیریت بحران، فوتوگرامتری، جستجو و پیگیری هدف و راهوری ترافیک به رویکردی نوین در جهت افزایش خودکاری، بهره‌وری و کارایی پروژه‌ها تبدیل شده است [۱]. با توجه به رویکرد پژوهشی سال‌های پسین در بخش فناوری‌های فضایی، روند تکامل سیستم‌های بدون سرنشین از سیستم‌های کنترل شونده از راه دور در جهت دستیابی به سیستم‌های خودکار است. در این زمینه، توسعه الگوریتم‌های جامع، کارآمد و مستحکم جهت مسیریابی، کنترل، ناوبری و پردازش اطلاعات سنجنده های UAS به‌عنوان یکی از بنیادی‌ترین گام‌ها در زمینه توسعه سیستم‌های خودکار مورد توجه پژوهشگران قرار گرفته است [۲].

به‌منظور مسیریابی UAS خودکار، الگوریتم مسیریاب باید بتواند خطوط سیر بهینه، کنش‌پذیر و قابل پرواز سکو یا گروهی از سکوها شبکه شده را با توجه به هندسه فضایی محیط پرواز، پارامترها و قیود مأموریت در زمان مناسبی محاسبه نماید [۳]. الگوریتم مسیریابی که افزون بر برآورد اهداف مأموریت و وظایف سیستم پیچیدگی محاسباتی و زمانی کمتری داشته باشد، جهت ناوبری سکو مناسب‌تر خواهد بود [۳]. در بیشتر پژوهش‌های پیشین، قیودی شامل ارتباطات بی‌سیم، شعاع چرخش سکو، ایمنی سیستم، مؤلفه‌های سرعت و ارتفاع و شعاع شناسایی تارگت در مدل‌سازی مسیریابی لحاظ گردیده‌اند [۱].

ثابت شده است که مسئله مسیریابی سکوها پرنده از رده مسائل درجه غیر چندجمله‌ای سخت^۲ است [۲۱، ۳، ۲۲]. در پژوهش‌های گذشته، مدل‌سازی‌های گوناگونی بر پایه روش‌هایی از جمله شبکه‌های عصبی [۴]، برنامه‌نویسی پویا [۵]، MILP^۳ [۶] و منطق فازی [۷] جهت حل مسئله مسیریابی ارائه گردیده است. باین‌حال، الگوریتم‌های فرا ابتکاری به دلیل برتری‌هایی از جمله سهولت در پیاده‌سازی و مدل‌سازی، انعطاف‌پذیری در برابر تغییر پارامترها، ابعاد مسئله و هندسه محیط، به‌کارگیری قیود و امکان افزایش قیدها بدون مدل‌سازی دوباره و

قابلیت‌های متنوع جستجوی سراسری و محلی، امکان آمیزه با دیگر روش‌های محاسباتی و وجود راهکارهای مختلف جهت گریز از نقاط بهینه محلی به‌صورت فراگیری مورد استفاده قرار گرفته‌اند [۸].

تئوری آشوب^۴ به شناخت و بررسی نظم و الگوی نهفته در حالت بی‌نظمی مشاهده‌شده در سیستم‌های دینامیک غیرخطی می‌پردازد [۹]. کشف نظریه آشوب نشان داد که صفات گوهری سیستم‌های پیچیده تنها بر پایه فرآیندهای تصادفی نویزگونه نیست و می‌توان رفتار و ساختار این نظام‌ها را بر اساس مدل آشوبناک در چارچوب یک سیستم دینامیکی غیرخطی توصیف کرد [۹]. در بسیاری از فرآیندهای طبیعی همانند رخداد توفان، شکل‌گیری سیاه‌چاله‌ها و چرخه آب نیز رفتارهای غیرخطی و غیرقابل‌پیش‌بینی برخاسته از ماهیت نامنظم کنش‌ها و واکنش‌ها مشاهده می‌شود. از این‌رو، تکامل^۵ و آشوب خود دو وجه بنیادین و تفکیک‌ناپذیر یک سیستم پیچیده طبیعی و در ارتباط متقابل و تنگاتنگ با یکدیگر می‌باشند. بر این بنیاد، به‌کارگیری همزمان تئوری آشوب با الگوریتم‌های فرا اکتشافی به‌منظور حل مسائل غیرخطی می‌تواند منجر به بهبود ویژگی‌های متمرکزسازی و متنوع-سازی روش در هنگام جستجوی محیط غیر محدب مسئله و افزایش قابلیت اجتناب از پاسخ‌های بهینه محلی گردد [۱۰].

حسن استفاده از روش‌های فرا اکتشافی جهت حل مسئله مسیریابی سکوها در تحقیق [۱۱] مشخص شده است. در این تحقیق، روشی تکاملی موسوم به MCACEA^۶ برای حل مسئله واقع گرایانه مسیریابی سکوها پرنده ارائه شده که می‌تواند بصورت سریع و کارا مسیرهای بهینه را با در نظر گرفتن قیود مختلف محاسبه نماید. در سال ۲۰۱۵، روش جدیدی بر مبنای الگوریتم تکامل اختلافی (DE) موسوم به MDEL^۸ به منظور حل مسئله سه بعدی مسیریابی در تحقیق [۲۰] منتشر شد. نتایج این تحقیق بیانگر این است که این روش تکاملی جدید عملکرد بسیار مناسبی از منظر استحکام^۹ و کیفیت پاسخ‌ها در حل مسئله مسیریابی سه بعدی داشته است.

۴ Chaos Theory

۵ Evolution

۶ Multiple Coordinated Agents Co-evolution EA

۷ Differential Evolution

۸ Modified Differential Evolution with Level Comparison

۹ Robustness

۱ Unmanned Aerial System

۲ NP-hard

۳ Mixed-integer Linear Programming

استفاده شده، اما زمان اجرا، نرخ موفقیت و دقت الگوریتم در مقایسه با سایر روش‌ها گزارش نشده است.

در پژوهش حاضر، چهار الگوریتم فرا اکتشافی شامل تکامل تفاضلی، توده ذرات، رقابت استعماری (ICA) و زنبورعسل مصنوعی بر اساس تئوری آشوب و با به کارگیری ۱۴ مولد آشوبی بهبود داده شده‌اند. سپس با به کارگیری الگوریتم‌های پیشنهادی، مسیریابی سکو در یک محیط شبیه‌سازی شده شهری انجام می‌گردد. از جمله مشارکت‌های جدید این تحقیق در مقایسه با پژوهش‌های پیشین می‌توان به موارد ذیل اشاره نمود:

- مدل‌سازی واقع‌گرایانه محیط مسئله با در نظر گرفتن مدل سه‌بعدی شهری و قیود دینامیکی و حدود نقشه عملیات.

- بررسی جامع سیگنال‌های آشوبی و تعیین بهترین نقشه برای هر الگوریتم متناسب با محدودیت‌ها و هزینه‌های هر مأموریت بر اساس نرخ موفقیت الگوریتم‌ها در اجراهای مستقل.

- افزایش کارایی^۶ الگوریتم‌ها از طریق بهبود قابلیت‌های متنوع سازی^۸ و متمرکز سازی^۹ الگوریتم‌های استاندارد.

در ادامه، تحلیل جامع عملکرد مسیریاب‌ها بر مبنای نرخ موفقیت، زمان اجرا، سرعت همگرایی، دقت و کیفیت پاسخ‌ها ارائه می‌شود.

۲- مدل‌سازی مسئله مسیریابی UAS

در مسئله مسیریابی سکوها پرند، با فرض مشخص بودن مختصات زمانی-مکانی، پیکربندی فضایی محیط حرکت و حالت کنونی سکو و یا گروهی از سکوها، هدف تعیین مسیرهای بهینه، قابل پرواز و کنش‌پذیر جهت حرکت پیوسته سکو می‌باشد؛ به گونه‌ای که در کمترین زمان ممکن، تابع هدف از پیش طراحی‌شده‌ای با در نظر گرفتن قیود دینامیکی حرکت سکو و پارامترهای مؤثر در مأموریت پرواز، بهینه گردد و سکو بتواند با حرکت بر روی مسیر به دست آمده در حضور مناطق تهدید و بدون برخورد با دیگر سکوها و توپوگرافی به یک موقعیت مشخص در

در این مقاله نیز طراحی قیود مسئله با توجه به تحقیقات [۱۱] و [۲۰] صورت گرفته است. در پژوهش [۲۱]، کارایی ۲۶ نسخه مختلف از الگوریتم‌های مبتنی بر توده ذرات (PSO)، ژنتیک (GA) و تکامل اختلافی جهت حل مسئله سه بعدی واقع‌گرایانه در ۴ سناریوی مختلف با یکدیگر مقایسه گردیده است. نتایج تحقیق [۲۱] نشان می‌دهد که نه تنها الگوریتم‌های تکاملی برای حل مسئله مناسب‌اند، بلکه از دید آماری نیز عملکرد بسیار مطلوبی دارند. در مراجع [۳] و [۲۳-۳۶] نیز، مزایای استفاده از الگوریتم‌های تکاملی مختلف در مسئله مسیریابی تشریح شده و برتری بکارگیری این روش‌ها برای حل این مسئله از طریق مقایسه جامع نتایج به اثبات رسیده است.

تاکنون از الگوریتم‌های تکاملی آشوب مبنای در حل مسئله مسیریابی با جزئیات این تحقیق استفاده نشده است. با این حال، در پژوهش [۱۶] یک الگوریتم فرا اکتشافی آشوبی مبتنی بر جغرافیای زیستی (BBO) جهت حل مسئله مسیریابی دوبعدی سکوها پرند ارائه شده است. در مدل‌سازی مسئله محدودیت‌هایی نظیر قیود امنیتی و انرژی لحاظ شده است. در تحقیق [۱۷] الگوریتم زنبورعسل مصنوعی (ABC) با نقشه لجستیک^۵ بهبود داده شده و برای حل همان مسئله [۱۶] بکار گرفته شده است. در تحقیقات [۱۶، ۱۷] هیچ‌گونه تلاشی جهت تعیین بهترین معادله آشوبی از بین دیگر روش‌ها انجام نگرفته است و برتری روش مطرح شده در مقایسه با سایر الگوریتم‌های آشوب مبنای بررسی نشده است. همچنین در این تحقیقات شبیه‌سازی‌ها واقع‌گرایانه نیست و باین وجود نیز قیود اندکی در بهینه‌سازی لحاظ شده‌اند. در تحقیق [۱۸] از تلفیق معادله لجستیک و با الگوریتم توده ذرات برای حل مسئله مرجع [۱۶] استفاده شده که هیچ‌گونه مقایسه جامعی با سایر روش‌ها ارائه نشده است. در این تحقیقات، هیچ‌گونه مقایسه‌ای برحسب زمان اجرا و نرخ موفقیت الگوریتم‌ها نیز ارائه نشده است. در سال ۲۰۱۳، الگوریتم تکامل تفاضلی تلفیق‌شده با نقشه لجستیک جهت حل تعمیم مسئله مرجع [۱۶] به محیط سه‌بعدی در تحقیق [۱۹] بکار گرفته شد. در پژوهش [۱۹] جهت مدل‌سازی محیط از یک معادله ریاضی برای نمایش سطح

^۶ Imperialist Competitive Algorithm

^۷ Algorithm Efficiency

^۸ Diversification

^۹ Intensification

^۱ Particle Swarm Optimization

^۲ Genetic Algorithm

^۳ Biogeography-Based Optimization

^۴ Artificial Bee Colony Algorithm

^۵ Logistic Map

$$J_e = \frac{5}{L_0} \sum_{i=1}^{N-1} |L(d_{i+1}) - L(d_i)|, \quad (6)$$

$$g_h = h_{safe} - \min(z_k - H_{terrain}(x_k, y_k)) \geq 0, \quad (7)$$

$$g_\phi = \max(\phi_k - \phi_k^{\max}) \geq 0, \quad (8)$$

$$g_\alpha = \max(s_k - \alpha_k) \geq 0, \quad (9)$$

$$g_\beta = \max(\beta_k - s_k) \geq 0, \quad (10)$$

$$h_{NFZ} = L_{NFZ} = 0, \quad (11)$$

$$h_{map} = L_{outmap} = 0, \quad (12)$$

در رابطه (۴)، L_u طول مسیر تولیدشده بر مبنای خم $p(u)$ و در رابطه (۵)، $p(h_i)$ بیانگر ارتفاع موقعیت i ام در سیستم مختصات محلی است. پارامتر k به منظور پاسخگویی به تمایلات مرکز تصمیم‌گیری جهت اجرای مأموریت و گزینش اولویت‌های اجرایی و تأمین انعطاف‌پذیری موردنیاز برای کنترل یکپارچه فرآیندهای بهینه‌سازی تعریف می‌شود. سایر پارامترهای مورد استفاده در روابط بالا عبارت‌اند از: N تعداد نقاط کنترل مسیر، h_{safe} پارامتر ایمنی جهت جلوگیری از برخورد با سطح، $l(d_i)$ طول قطعه مسیر i ام و L_0 مجموع طول قطعات مسیر تا نقطه پیموده شده. در رابطه (۸)، $\phi_k^{\max}$ نشان‌دهنده بیشینه زاویه چرخش در هر نقطه مسیر می‌باشد که مطابق رابطه (۱۳) به فراخور بیشینه اضافه‌بار پهلویی n_{max} ، شتاب گرانش g و سرعت پرواز V برآورد می‌گردد [۱۱].

$$\phi_k^{\max} = \frac{n_{max} g}{V} \cdot \sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2}, \quad (13)$$

مطابق روابط (۹) و (۱۰) زاویه شیب سکو s_k در نقطه کنترل $p(u_k)$ وابسته به بیشینه شیب بالاروی α_k و کمینه شیب خزشی β_k است. روابط (۱۴)–(۱۶) روشنگر شیوه محاسبه s_k ، α_k و β_k بر پایه ارتفاع هر نقطه کنترل می‌باشند [۱۱].

$$\alpha_k = -1.53 \times 10^{-10} z_k^2 - 2.69 \times 10^{-5} z_k + 0.42, \quad (14)$$

$$\beta_k = 2.50 \times 10^{-9} z_k^2 - 6.30 \times 10^{-6} z_k - 0.32, \quad (15)$$

چارچوب زمانی-مکانی تعریف‌شده مأموریت برسد [۳، ۱۱، ۲۵].

در الگوریتم مسیریابی، جهت دستیابی به مسیر نرم، کنش‌پذیر و قابل پرواز، کاهش پیچیدگی محاسبات هندسی مسیر، انعطاف‌پذیری در طراحی مأموریت‌های گوناگون، امکان طراحی دوباره مسیر و کاهش پیچیدگی زمانی-حافظه الگوریتم از مدل ریاضی خم‌های بیزیر^۱ بهره‌گیری می‌شود. حالت چندجمله‌ای خم‌های بیزیر با نقاط کنترل $\{p_0, p_1, \dots, p_n\}$ مطابق معادلات (۱) و (۲) می‌باشد [۱].

$$P(u) = \sum_{m=0}^n u^m S_m, \quad (1)$$

$$S_m = \frac{n!}{(n-m)!} \sum_{i=0}^m \frac{(-1)^{i+m} p_i}{i!(m-i)!} \\ = \prod_{k=0}^{m-1} (n-k) \sum_{i=0}^m \frac{(-1)^{i+m} p_i}{i!(m-i)!}, \quad (2)$$

در این پژوهش، مسئله مسیریابی به صورت مسئله بهینه‌سازی مقیدی تعریف گردیده، به گونه‌ای که با در نظر گرفتن زاویه چرخش و پارامترهای دینامیکی پرواز و با پرهیز از نواحی ممنوع و بیرون از نقشه عملیاتی، مسیر بهینه با کمترین مسافت و انرژی و در ارتفاع ایمن محاسبه گردد. بر این مبنا، در معادله (۳) تابع هزینه‌ای برای قطعات خم بیزیر تعریف گردیده که شامل هزینه مسافت، ارتفاع و انرژی بوده و باید به صورت همزمان طی فرآیند مسیریابی سکو کمینه گردد.

$$\text{Minimize } J = k_1 J_l + k_2 J_s + k_3 J_e, \\ 0 < k_1, k_2, k_3 < 1, \quad (3)$$

$$\text{Subject to } g_h, g_\phi, g_\alpha, g_\beta \geq 0, \\ h_{NFZ}, h_{map} = 0,$$

که در این معادله، پارامترها و متغیرهای بکار رفته به صورت زیر طراحی و تعریف گردیده‌اند:

$$J_l = L_u - L_0 = \sum_{i=0}^{N-1} \min \left\{ \|p(u_{i+1}) - p(u_i)\|_2 \right\}, \quad (4)$$

$$J_s = \sum_{i=0}^{N-1} \min \left\{ \|p(u_i) - p(h_i)\|_2 \right\}, \quad (5)$$

^۱ Bezier Curves

جدول ۱- معادله نقشه‌های آشوبی پیاده‌سازی شده [۱۰، ۹].

شناسه	تعریف
M01	$x_{k+1} = \cos(k \cos^{-1}(x_k))$,
M02	$x_{k+1} = \text{mod}(x_k + 0.2 - (1/4\pi)\sin(2\pi k), 1)$,
M03	$x_{k+1} = \begin{cases} 0 & x_k = 0 \\ 1/x_k \bmod(1) & \text{otherwise} \end{cases}$ $1/x_k \bmod(1) = 1/x_k - [1/x_k]$,
M04	$x_{k+1} = \sin(a\pi/x_k)$, $a \in (0, 1)$
M05	$x_{k+1} = \begin{cases} \varepsilon + x_k + cx_k^n & 0 \leq x_k \leq P \\ \frac{x_k - P}{1 - P} & P < x_k < 1 \end{cases}$ $c = \frac{1 - \varepsilon - P}{P^n}$,
M06	$x_{k+1} = \begin{cases} \frac{x_k}{P} & 0 \leq x_k < P \\ \frac{x_k - P}{0.5 - P} & P \leq x_k < 0.5 \\ \frac{1 - P - x_k}{0.5 - P} & 0.5 \leq x_k < 1 - P \\ \frac{1 - x_k}{P} & 1 - P \leq x_k < 1 \end{cases}$
M07	$x_{k+1} = ax_k(1 - x_k)$, $a = 4$,
M08	$x_{k+1} = \mu(7.86x_k - 23.31x_k^2 + 28.75x_k^3 - 13.302875x_k^4)$, $\mu = 1.07$,
M09	$x_{k+1} = ax_k^2 \sin(\pi x_k)$, $a = 2.3$,
M10	$x_{k+1} = \begin{cases} \frac{x_k}{0.7} & x_k < 0.7 \\ \frac{10}{3}(1 - x_k) & x_k \geq 0.7 \end{cases}$
M11	$x_{k+1} = ax_k \bmod(1)$, $a = 2$,
M12	$x_{k+1} = \begin{cases} \alpha x_k & 0 < x_k \leq P_1 \\ \frac{P_1 - x_k}{P_2 - P_1} & P_1 < x_k \leq P_2 \\ 1 - \beta(1 - x_k) & P_2 < x_k \leq 1 \end{cases}$ $\alpha = \frac{P_2}{P_1}(1 - (P_2 - P_1))$, $\alpha < \beta$ $\beta = \frac{1}{P_2 - 1}((P_2 - 1) - P_1(P_2 - P_1))$
M13	$x_{k+1} = (a/4)\sin(\pi x_k)$, $a \in (0, 4]$,
M14	$x_{k+1} = 1 - 1.4x_k^2 + 0.3x_{k-1}$,

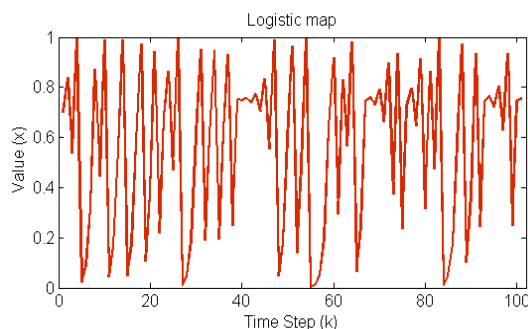
سیستم‌های غیرقطعی که ویژگی‌ها و رفتار آن‌ها بیشتر بر پایه قوانین غیر دینامیکی یا نیوتنی سامان یافته‌اند، مجموعه‌ای از صفات و حالات نامعین و غیرقابل پیش‌بینی را از خود بروز می‌دهند؛ در نتیجه با مشاهده وضعیت

$$s_k = \frac{z_{k+1} - z_k}{\sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2}}, \quad (16)$$

قید h_{NFZ} در رابطه (۱۱) جهت جلوگیری از ورود سکو به ناحیه ممنوعه پرواز نظیر نواحی ناشناخته، ساختمان‌های بلند و نواحی پرخطر برای سکو تعریف شده است. در این رابطه L_{NFZ} طول بخشی از مسیر سکوست که در درون کرانه ممنوعه واقع شده است. از سوی دیگر، پرواز سکو نیز باید در درون گستره مجاز پرواز انجام گیرد، بدین جهت، قید h_{map} بر اساس معادله (۱۲) تعیین می‌شود. در این رابطه L_{outmap} طول بخشی از قطعه مسیر است که بیرون ناحیه مجاز پرواز واقع شده باشد [۲۰].

۳- تئوری آشوب و محاسبات تکاملی

با واکاوی و مدل‌سازی ریاضی رفتار پیچیده و ویژگی‌های ذاتی و منحصربه‌فرد پدیده‌های پویای طبیعی نظیر تغییرات شرایط جوی زمین، می‌توان مشاهده نمود که سیستم دینامیکی معادلات دیفرانسیلی یا جبری به‌دست‌آمده با گذر زمان دچار تکامل و دگرگونی گردیده و دارای حساسیت شدیدی در برابر تغییر شرایط اولیه و پارامترهای سیستم می‌باشد. از این‌رو، پدیده آشوب می‌تواند به‌صورت رفتار و حرکت کران‌دار، بلندمدت، پیچیده، پیوسته، شبه تصادفی، بی‌نظم و غیرمتناوب در سیستم غیرخطی قطعی تعریف گردد که وابستگی حساس نسبت به شرایط اولیه و پارامترها را از خود بروز می‌دهد [۱۲]. در جدول (۱)، معادله ۱۴ مولد آشوبی پیاده‌سازی شده در این مقاله و پارامترهای مربوط به آن‌ها ارائه گردیده است [۱۰، ۹]. شکل ۱ نیز رفتار آشوبی سیگنال لجستیک (M07) را بیان می‌نماید.



شکل ۱- نمایش رفتار آشوبی سیگنال لجستیک (M07)

کنونی و قوانین جاری حاکم بر آن‌ها نمی‌توان حالت و وضعیت آینده سیستم را در یک چارچوب زمانی-مکانی مشخص پیش‌بینی نمود [۱۲]. از دیگر سوی، سیستم‌های موردبررسی در نظریه آشوب، سیستم‌های قطعی هستند که حتی می‌توان رفتار غیرخطی و حالت ناپایدار آینده آن‌ها را بر اساس معادلات دیفرانسیلی حاکم بر سیستم توصیف نمود، اما به این دلیل که هیچ دوره تناوب غالبی در یک سیستم نامنظم آشوبی موجود نیست، نمی‌توان وضعیت آینده سیستم را در یک بازه طولانی مدت و در چارچوب تابعی از وضعیت کنونی آن‌ها مدل‌سازی نمود [۱۰، ۱۲].

در تئوری محاسبات تکاملی نیز الگوریتم‌های بهینه‌سازی متنوعی با عملگرهای تصادفی و با الهام‌گیری از رفتار و ویژگی‌های ذاتی سیستم‌های طبیعی و بیولوژیکی نظیر تکامل، خودسازمان‌دهی، برهم‌کنش، رقابت، یادگیری، انعطاف‌پذیری، حافظه، استحکام و دگرسانی توسعه‌یافته‌اند [۳]. در این زمینه، هرگونه نظم مشاهده‌شده در سیستم اغلب مبتنی بر سیر تکامل و خودسازمان‌دهی اجزای شکل‌دهنده سیستم شکل گرفته و از طرفی این سیستم خود در حضور الگوهای نامنظم و نامتقارن اولیه تکامل یافته، به گونه‌ای که پیش‌بینی سیر آینده تکامل آن بر اساس رفتارها و ویژگی‌های کنونی سیستم ممکن نیست. از این‌رو، تلفیق این دو تئوری می‌تواند منجر به نتایج برتر در حل مسائل غیرخطی بهینه‌سازی گردد.

۴- طراحی مسیر یاب‌های تکاملی آشوبناک

در مسائل پیچیده و دینامیک، گزینش پارامترهای مناسب اولیه نقش بسزایی در دقت جستجو و قدرت اکتشاف الگوریتم‌ها دارد. تنظیم نامناسب الگوریتم جهت حل مسئله مسیریابی می‌تواند منجر به عدم همگرایی الگوریتم، افزایش خطای پاسخ‌ها و یا نوسان در روند همگرایی الگوریتم شود. در این بخش چارچوب الگوریتم‌های تکاملی بهبود داده شده با تئوری آشوب ارائه می‌گردد. در این الگوریتم‌ها از مولدهای آشوبی یا به‌منظور تراز آشوبی پارامترهای اولیه و یا به‌عنوان جایگزین متغیرهای تصادفی بهره‌گیری می‌شود. در تحقیق حاضر،

تئوری آشوب به‌منظور تعدیل و ارتقاء الگوریتم‌های ICA، PSO، DE و ABC بکار گرفته می‌شود.

۴-۱- الگوریتم رقابت استعماری آشوبناک (CHICA)

در این الگوریتم جمعیت اولیه کشورها میان چند استعمارگر تقسیم گردیده و پاسخ‌های برتر از طریق فرآیند استعمار برای پذیرش کشورهای بیشتر رقابت می‌کنند. با تکرارهای بیشتر، کشورهای اولیه از طریق عملگرهای استعمار، رقابت استعماری و انقلاب در هر تکرار به سمت پاسخ برتر یا امپریالیسم همگرا می‌شوند. در شرایط همگرایی تنها امپریالیسم باقی‌مانده با کمترین تابع هدف به‌عنوان پاسخ بهینه معرفی می‌شود [۴]. در این مقاله، عملگر استعمار به‌عنوان عامل حرکت پاسخ‌ها با مولد آشوبی به‌صورت زیر تعدیل می‌گردد.

$$x_i(t+1) = x_i(t) + \beta \times d \times \text{chaos}(t) \times \{V_1\} \quad (17)$$

در این رابطه β پارامتر کنترلی، d اختلاف مکان کشورها و امپریالیست و $\{V_1\}$ برداری به مبداء آن کشور و در جهت امپریال است.

۴-۲- الگوریتم توده ذرات آشوبناک (CHPSO)

در این الگوریتم، با افزایش تکرارها جمعیتی از ذرات با سرعت و موقعیت اولیه تصادفی به سمت موقعیت ذره‌ای با کمترین تابع هزینه حرکت می‌نمایند. بدین منظور، سرعت و موقعیت ذرات با در نظر گرفتن بهترین سرعت و موقعیت تجربه‌شده به‌وسیله آن ذره و بهترین موقعیت به‌دست‌آمده در همسایگی آن ذره بروز رسانی می‌شود [۱۳]. در این مقاله، پارامترهای ثابت مربوط به فاکتورهای شتاب معادله سرعت، با تکرار مولد آشوبی به‌صورت دینامیک تغییر نموده و حرکت پاسخ‌ها بر اساس معادلات سرعت و مکان (۱۸) و (۱۹) محاسبه می‌گردند.

$$v_{i,d}(t+1) = \text{chaos}(t) \times v_{i,d}(t) + \text{chaos}(t) \times r_{1,d}(t) [pbest_{i,d}(t) - x_{i,d}(t)] + \text{chaos}(t) \times r_{2,d}(t) [gbest_d(t) - x_{i,d}(t)], \quad (18)$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \quad (19)$$

در نسخه آشوبی پیشنهادی، پارامترهای تصادفی CR و F در روابط (۲۰) و (۲۱) به صورت دینامیکی و در هر تکرار بر اساس معادلات (۲۳) و (۲۴) تولید می شوند. شبه کد الگوریتم جدید در شکل ۳ تشریح گردیده است.

$$v_i^t = x_{r_1}^t + chaos(F) \times (x_{r_2}^t - x_{r_3}^t), \quad (23)$$

$$u_{ij}^t = \begin{cases} v_{ij}^t & \text{if } s_i \leq chaos(CR) \text{ or } j = rd_i, \\ x_{ij}^t & \text{if } s_i > chaos(CR) \text{ or } j \neq rd_i, \end{cases} \quad (24)$$

Set the generation number $t = 0$.
Initialize the first chaotic variable.
Generate the initial population randomly.
Evaluate the fitness value of population members.
repeat

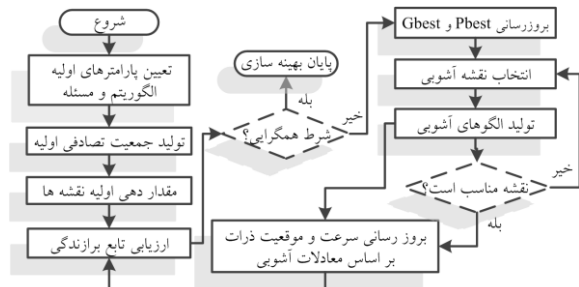
Set $t = t + 1$ {Generation increasing}
for $i = 0; i < \text{population size}; i++$ **do**
Generate chaotic variable $chaos^{(t)}(t)$
Select random indexes r_1, r_2, r_3
 $v_i^{(t)} = x_{r_1}^{(t)} + chaos^{(t)}(t) \times (x_{r_2}^{(t)} - x_{r_3}^{(t)})$ {Mutation}
 $j = rand(1, D)$
for $(k = 0; k < D; k++)$ **do**
Generate chaotic variable $chaos^{(k)}(k)$
if $(rand(0, 1) < chaos^{(k)}(k) \text{ or } k = j)$ **then**
 $u_{ik}^{(t)} = v_{ik}^{(t)}$ {Crossover}
else
 $u_{ik}^{(t)} = x_{ik}^{(t)}$
end if
end for
if $f(x_i^{(t)}) < f(u_i^{(t)})$ **then**
 $x_i^{(t+1)} = u_i^{(t)}$ {Greedy selection}
else
 $x_i^{(t+1)} = x_i^{(t)}$
end if
end for
until Termination condition satisfied.

شکل ۳- شبه کد الگوریتم تکامل تفاضلی آشوبناک

۴-۴- الگوریتم زنبور مصنوعی آشوبناک (CHABC)

در روش استاندارد کارابوا [۱۵]، جمعیت پاسخهای اولیه (SN) دربرگیرنده سه گروه زنبورهای کارگر (N_e)، جستجوگر (N_e) و دیده بان (N_o) است که جهت یافتن پاسخهای برتر به گونه ای در سه فاز همکاری می نمایند که گلزارهای مطلوب تر بیشتر بازدید شود. پس از پخش

در این روابط، $r_{1,d}(t)$ و $r_{2,d}(t)$ بردارهای تصادفی در فضای d -بعدی مسئله، $pbest_{i,d}(t)$ بهترین موقعیت ذره تاکنون، $gbest_d(t)$ بهترین موقعیت در بین همسایگان می باشند. شکل ۲، فلوچارت الگوریتم توده ذرات آشوب مبنا را نمایش می دهد.



شکل ۲- فلوچارت الگوریتم پیشنهادی

۴-۳- الگوریتم تکامل تفاضلی آشوبناک (CHDE)

در این روش، فرآیند جستجوی احتمالی از طریق تکرار عملگرهای جهش، تقاطع و انتخاب جمعیت تصادفی اولیه را به سمت پاسخهای با بهترین برازندگی هدایت می کند [۱۴]. طبق معادله زیر، با جهش هر عضو جمعیت، یک جواب جدید بر اساس ضریب مقیاس F و سه عضو تصادفی $x_{r_1}^t$ دیگر تولید می شود.

$$v_i^t = x_{r_1}^t + F \times (x_{r_2}^t - x_{r_3}^t), \quad i = \{1, \dots, N_{pop}\} \quad (20)$$

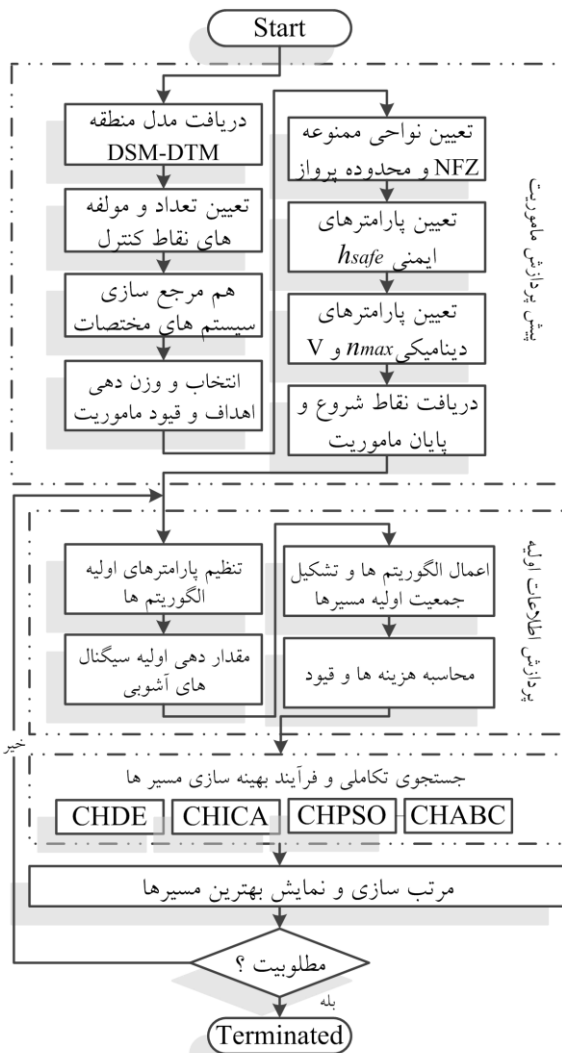
به منظور متنوع سازی موقعیت پاسخها، عملگر تقاطع جواب والد و جواب جهش یافته جدید را مطابق رابطه (۲۱) به جواب u_{ij}^t تبدیل می کند.

$$u_{ij}^t = \begin{cases} v_{ij}^t & \text{if } s_i \leq CR \text{ or } j = rd_i, \\ x_{ij}^t & \text{if } s_i > CR \text{ or } j \neq rd_i, \end{cases} \quad (21)$$

در این رابطه CR بیانگر احتمال تقاطع، $j = \{1, \dots, D\}$ ، s_i یک عدد تصادفی در بازه یکه (۰، ۱) و rd_i یک عدد تصادفی در بازه $(0, d)$ است. سپس در مرحله انتخاب، پاسخهای برتر بر مبنای رابطه (۲۲) برگزیده می شوند.

$$x_i^{t+1} = \begin{cases} x_i^t & \text{if } f(x_i^t) < f(u_i^t) \\ u_i^t & \text{if } f(x_i^t) \geq f(u_i^t) \end{cases} \quad (22)$$

مأموریت، مدل دینامیک UAS و اطلاعات محیطی، وارد فاز پردازش اطلاعات دریافتی می‌گردد. در این مرحله الگوریتم‌ها، سیگنال‌ها و جمعیت اولیه مسیرهای ممکن مقداردهی می‌شوند. سپس، در مرحله محاسبه مسیرها جمعیت خطوط سیر تولیدشده بر مبنای الگوریتم‌های تکاملی پیشنهادی بهبود داده می‌شود. بر این مبنای، فرآیند مسیریابی تا رسیدن به شرایط فرجام الگوریتم‌ها ادامه یافته و چنانچه مسیرها از مطلوبیت هندسی و کنش پذیری برخوردار باشند، نقاط کنترل بهترین‌ها به بخش نمایش مسیر یا ایستگاه کنترل ارسال می‌گردد.



شکل ۴- فلوچارت الگوریتم مسیریابی

۵- ارزیابی و تحلیل نتایج مسیریابی

در این بخش سنجش کارکرد الگوریتم‌های پیشنهادی از دید شاخص موفقیت، سرعت همگرایی، زمان اجرا و

تصادفی منابع در فضای جستجو، زنبورهای کارگر منابع همسایگی جدید را بر اساس رابطه (۲۵) جستجو می‌کنند.

$$v_{m,i} = x_{m,i} + \phi_{m,i} (x_{m,i} - x_{k,i}), \quad (25)$$

در این رابطه $\phi_{m,i}$ عددی تصادفی در بازه (۰،۱) و $x_{k,i}$ منبع غذایی انتخاب‌شده به صورت تصادفی می‌باشد. سپس بر مبنای معادله (۲۶)، زنبورهای جستجوگر منابع غذایی را با احتمال p_m انتخاب می‌نمایند.

$$p_m = \frac{fit(x_m)}{\sum_{m=1}^{SN} fit(x_m)}, \quad (26)$$

پس از انتخاب منبع غذا، همچون فاز زنبورهای کارگر، منبع همسایه $v_{m,i}$ با معادله (۲۵) تعیین شده و تابع برازندگی آن محاسبه می‌شود. سپس یک انتخاب حریصانه بین x_m و $v_{m,i}$ انجام می‌گردد. در نتیجه، زنبورهای جستجوگر بیشتری به مناطق بهتر ارسال می‌شوند. اگر منبعی پس از گذشت چندین تکرار بهبود نیافت، مطابق رابطه (۲۷)، دیدبان‌ها منابع نوپایه‌ای را به صورت تصادفی در محیط جستجو ایجاد می‌کنند.

$$x_i^j = x_{min}^j + rand(0,1)(x_{max}^j - x_{min}^j), \quad (27)$$

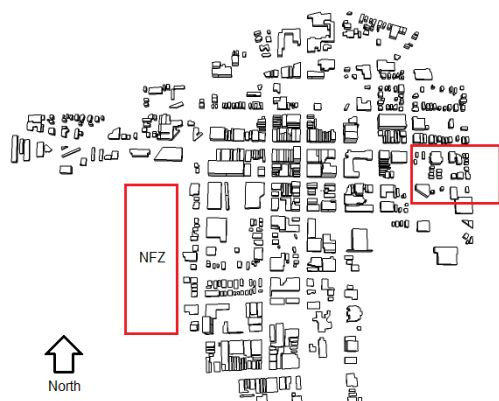
در نسخه آشوبی پیشنهادی، مطابق رابطه (۲۸)، جمعیت اولیه با تکرار مولد آشوبی و به صورت دینامیکی در محیط جستجو توزیع می‌شوند. همچنین در صورتی که پاسخی بهبود نیافت، زنبور کارگر به دیدبان تبدیل شده و پاسخ‌های جدید بر اساس معادله (۲۹) تعیین می‌شوند.

$$x_i^j = x_{min}^j + chaos(t)(x_{max}^j - x_{min}^j), \quad (28)$$

$$x_i^j = x_i^j + (2chaos(t) - 1)(x_{max}^j - x_{min}^j), \quad (29)$$

۴-۵- الگوریتم مسیریابی UAS

مراحل فرآیند مسیریابی و فلوچارت مسیریاب طراحی‌شده در شکل ۴ تشریح شده است. مطابق این فرآیند، الگوریتم پس از دریافت پارامترهای موردنیاز



شکل ۵- (ب) مدل دوبعدی شهری به مقیاس ۱/۱۰۰۰ و نواحی ممنوعه

شکل ۵- هندسه محیط شهری جهت مسیریابی پرواز سکو

زنجیره نخست پیاده‌سازی جهت شناسایی سیگنال آشوبی مناسب هر مسیریاب بر اساس نرخ موفقیت هر الگوریتم با هر یک از ۱۴ مولد اجرا می‌گردد. مولد با بیشترین نرخ در میان دیگر الگوریتم‌ها، بهترین سیگنال برای هر الگوریتم می‌باشد. پس از انتخاب بهترین مولد برای هر یک از ۴ مسیریاب آشوبی، نتایج مسیریابی برای هر یک از مسیریاب‌های مبتنی بر الگوریتم‌های استاندارد نیز محاسبه می‌گردد. به منظور ارزیابی توانایی روش‌های پیشنهادی، شاخص موفقیت الگوریتم در یافتن نقاط بهینه به صورت زیر تعریف می‌شود:

$$S_r = 100 \times [NR_{\text{successful}} / NR_{\text{total}}] \quad (30)$$

در این رابطه $NR_{\text{successful}}$ شمار اجرای موفق الگوریتم و NR_{total} مجموع کل اجراها می‌باشد. اجرای موفق نیز بر اساس رابطه (۳۱) تعریف می‌گردد:

$$\sum_{d=1}^{Dim} (S_i^{gb} - S_i^B)^2 \leq (L_{\max} - L_{\min}) \times 10^{-3} \quad (31)$$

در معادله (۳۱)، S_i^{gb} بهترین پاسخ سراسری یافت شده در فضای جستجو، S_i^B پاسخ بهینه، Dim بعد مسئله، $L_{\max}$ و $L_{\min}$ نیز بیانگر کران متغیرهای مسئله می‌باشند. نتایج مربوط به پتانسیل مسیریاب‌های آشوبی پس از ۱۰۰ تکرار برای هر روش در جدول ۴ گزارش و در شکل ۶ مقایسه گردیده است. در آخرین سطر جدول ۴، نتایج الگوریتم‌های اولیه با مقادیر تصادفی در حالت استاندارد گزارش شده است.

کیفیت پاسخ‌ها به منظور برنامه‌ریزی مسیر UAS خودکار صورت گرفته است. در این پژوهش، کلیه شبیه‌سازی‌ها در شرایط یکسان و با بهره‌گیری از نرم‌افزارها و سیستم معرفی شده در جدول ۲، انجام شده است.

جدول ۲- جزئیات رایانه، زبان و نرم‌افزارهای پیاده‌سازی

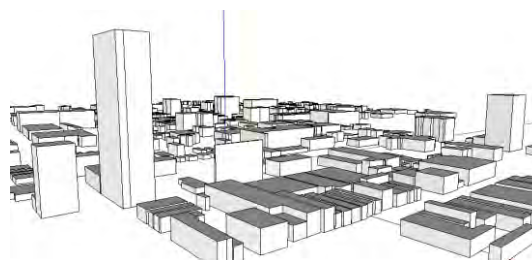
نام	جزئیات
رایانه	Pentium IV processor CPU, 1.7 GHz, 512 MB RAM, Windows 7
زبان	MATLAB R2012a (7.14), C++
نرم‌افزار	ArcGIS 10.2 [Arc-Scene], Google Earth, Google SketchUp

با توجه به این امر که انتخاب پارامترهای مناسب اولیه برای الگوریتم‌ها نیاز به کارآزمودگی و مهارت کافی در حل مسئله دارد، پس از بررسی نتایج برآمده از شبیه‌سازی‌های اولیه و انتخاب بهترین مقادیر برای تضمین همگرایی روش‌ها، مقادیر مناسب به صورت ارائه شده در جدول ۳ گزینش شده است.

جدول ۳- مقادیر پارامترهای الگوریتم‌ها

الگوریتم	پارامترها
DE	Crossover $CR = 0.5$, Weighting factor $F = 0.95$
PSO	Inertial constant = 0.3, Social constant of Swarm interaction = 1, Cognitive constant = 1
ABC	Number of bees $N_s = 50$, Colony size of the employed bees $N_e = 25$, $N_u = 25$, $T_{\max} =$ 200, Limit = 50
ICA	The number of empires $N_{\text{imp}} = 10$, Assimilation coefficient $\beta = 2$, Colonies mean cost coefficient $\xi = 0.1$, Revolution rate $\mu = 0.3$, Assimilation angle coefficient $\gamma = 0.7$

جهت مدل‌سازی فضای شهری مأموریت سکو، یک مدل رقمی سطح شامل ساختمان‌هایی با ارتفاعات متفاوت تولید و بکار گرفته شده است (شکل ۵). هر عارضه با مختصات رئوس و مؤلفه z هر رأس به صورت یک ماتریس به الگوریتم مسیریاب معرفی می‌شود.

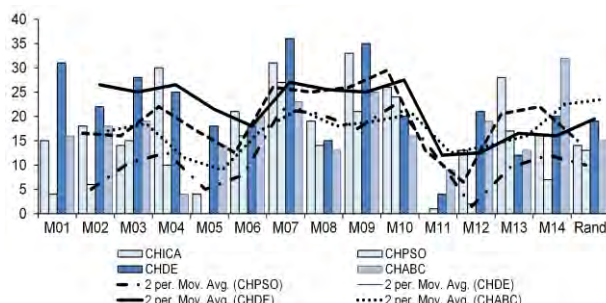


(الف) نمای سه‌بعدی مدل شهری محیط پرواز

جدول ۴- عملکرد ۶۰ مسیریاب تکاملی بر مبنای نرخ موفقیت

سیگنال	CHICA	CHPSO	CHDE	CHABC
M01	۱۵	۰۴	۳۱	۱۶
M02	۱۸	۰۶	۲۲	۱۸
M03	۱۴	۱۵	۲۸	۱۹
M04	۳۰	۱۰	۲۵	۰۴
M05	۰۴	۰۰	۱۸	۱۴
M06	۲۱	۱۶	۱۸	۲۰
M07	۳۱	۲۷	۳۶	۲۳
M08	۱۹	۱۴	۱۵	۱۳
M09	۳۳	۲۱	۳۵	۲۵
M10	۲۶	۲۴	۲۰	۱۶
M11	۰۰	۰۱	۰۴	۰۹
M12	۱۳	۰۲	۲۱	۱۹
M13	۲۸	۱۷	۱۲	۱۳
M14	۱۶	۰۷	۲۰	۳۲
-	۱۴	۱۳	۱۹	۱۵

به‌منظور بررسی مقایسه‌ای، نرخ موفقیت مسیریاب‌های پیشنهادی برحسب میانگین وزن‌دار و نیز درصد رسیدن به پاسخ‌های مطلوب به‌صورت نمودار میله‌ای در شکل ۶ نمایش داده شده است.



شکل ۶- مقایسه مسیریاب‌های تکاملی برحسب نرخ موفقیت و میانگین متحرک متناظر با هر الگوریتم

با توجه به نتایج پیشین، چهار مسیریاب برتر شامل تکامل اختلافی با M07، رقابت استعماری با M09، زنبورعسل مصنوعی با M14 و توده ذرات با M07 انتخاب گردیده و برای تحلیل، زمان اجرا، دقت و کیفیت پاسخ‌ها موردبررسی قرار می‌گیرند. بدین منظور تعداد تکرار هر الگوریتم جهت همگرایی به پاسخ‌های پایدار در سطح ۵۰۰ حلقه تنظیم می‌گردد. در جدول ۶، نتایج برآمده از بررسی‌های آماری بهترین، میانگین و انحراف از معیار تابع هزینه مسیرها در ۱۰۰ تکرار منعکس گردیده است.

جدول ۶- مقایسه آماری عملکرد مسیریاب‌های تکاملی از دید

کیفیت و دقت نتایج در سراسر فرآیند مسیریابی

الگوریتم	بهترین	میانگین	انحراف از معیار
CHDE(M07)	۹۳/۲۴۳۹	۹۶/۲۳۱۷	۰۴/۱۱۷۹
CHABC(M14)	۹۵/۷۲۰۷	۹۷/۰۴۲۱	۰۶/۲۳۵۸
CHICA(M09)	۹۶/۱۹۸۲	۹۸/۷۲۸۳	۰۷/۷۲۳۶
CHPSO(M07)	۹۸/۱۴۷۵	۹۹/۱۲۵۸	۰۵/۵۲۲۳

بر اساس نتایج جدول ۶ مشاهده می‌شود که نتایج الگوریتم CHDE(M07) دارای کمترین انحراف از معیار در بین دیگر الگوریتم‌ها است. بر این اساس، روش‌های CHPSO(M07)، CHABC(M14) و CHICA(M09) در رتبه‌های بعدی قرار گرفته و دارای دقت محاسباتی کمتری می‌باشند. توزیع گسترده‌تر پاسخ‌ها در فضای جستجوی مسئله نشان‌دهنده برتری الگوریتم در قابلیت متنوع سازی الگوریتم و چگالی بیشتر در توزیع پاسخ‌ها بیانگر پتانسیل

با توجه به نتایج جدول ۴ مشاهده می‌شود که الگوریتم رقابت استعماری با سیگنال M09 (Sinusoidal) از بهترین عملکرد در بین ۱۴ نسخه دیگر ICA برخوردار می‌باشد. الگوریتم توده ذرات با نقشه آشوبی M07 (Logistic) دارای بیشترین نرخ موفقیت است. الگوریتم ABC با نقشه آشوبی M14 (Henon) دارای بالاترین نرخ موفقیت در یافتن پاسخ‌های مطلوب می‌باشد؛ اما الگوریتم تکامل تفاضلی با مولد M07 دارای بهترین عملکرد در بین ۶۰ الگوریتم دیگر است.

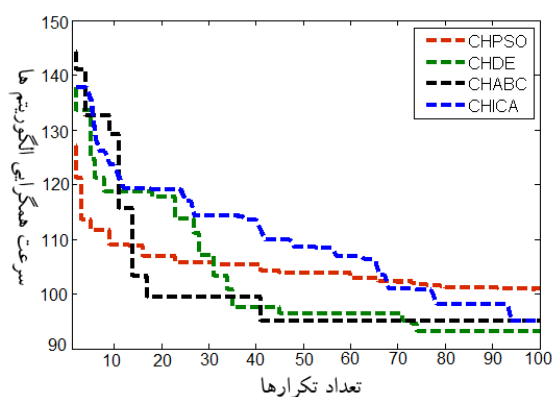
از میان الگوریتم‌های استاندارد، رده نخست تا چهارم الگوریتم‌ها به ترتیب از آن ABC، DE، ICA و PSO است. همچنین خلاف اینکه برخی عملگرهای دینامیک نظیر M07 عملکرد الگوریتم‌ها را بهبود بخشیده‌اند، نقشه‌هایی نظیر M11 (Sawtooth) و M05 (Intermittency) برای ارتقاء مسیریاب‌های پیشنهادی مناسب نیستند. در جدول ۵، رتبه‌بندی مناسب‌ترین مولدها برای ارتقاء هر مسیریاب منعکس گردیده است.

جدول ۵- رتبه‌بندی نقشه‌های آشوبی برتر برای هر مسیریاب

تکاملی بر اساس نرخ موفقیت درمجموع ۱۰۰ شبیه‌سازی

کد / رتبه	ABC	DE	PSO	ICA
اول	M14	M07	M07	M09
دوم	M09	M09	M10	M07
سوم	M07	M01	M09	M04
چهارم	M06	M03	M13	M13

با هدف تحلیل سرعت همگرایی الگوریتم‌ها، نتایج میانگین بهترین اجراها پس از ۱۰۰ تکرار در شکل ۸ نمایش داده شده است. بر اساس نتایج، سرعت همگرایی الگوریتم CHABC(M14) تا پیش از ۳۰ تکرار از دیگر الگوریتم‌ها بیشتر بوده، اما از تکرار ۴۰ اکتشاف بیشتر نواحی جستجو متوقف شده است. برای الگوریتم CHPSO(M07) در کمتر از ۲۵ تکرار نرخ همگرایی کاهش می‌یابد و الگوریتم به سمت نقطه کمینه محلی همگرا می‌گردد. الگوریتم CHDE(M07) نیز دارای سرعت همگرایی بیشتری از CHICA(M09) و CHPSO(M07) بوده و پاسخ پایانی نیز به پاسخ دقیق نزدیک‌تر شده است.



شکل ۸- مقایسه سرعت همگرایی مسیر یاب‌های تکاملی مختلف

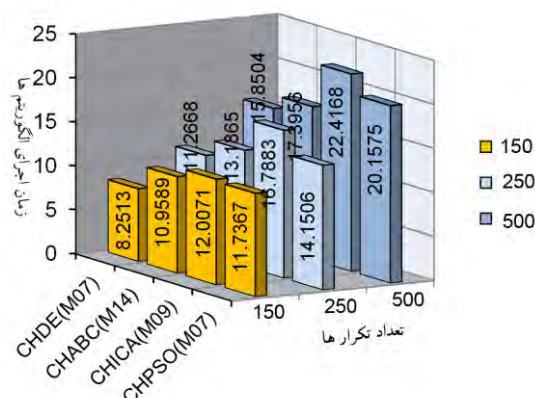
به دلیل عدم مطلوبیت قابلیت‌های دیداری‌سازی نرم‌افزار MATLAB، با توجه به مدل طراحی‌شده منطقه، نقاط کنترل لازم جهت طراحی خطوط سیر مقید متناظر با هر سکو به‌عنوان خروجی مسیر یاب‌های تکاملی محاسبه و ذخیره گردید. سپس مختصات نقاط کنترل به‌دست‌آمده از الگوریتم جهت تولید و ترسیم مسیرها به سیستم مختصات یکسان با سیستم مختصات منطقه شبیه‌سازی‌شده منتقل شد. همچنین جهت فراهم‌سازی خطوط سیر کنش پذیر پرواز، خم‌ها در نقاط همبندی و پیچ‌ها هموار و نرم گردیده است. شکل‌های ۹ و ۱۰ نشان‌دهنده نتایج مسیریابی الگوریتم‌ها می‌باشد. در شکل ۹، مسیرهای تولیدی الگوریتم CHDE(M07) در بخش (الف) و مسیرهای سایر الگوریتم‌ها در بخش (ب) نمایش داده شده است. در شکل ۱۰ نیز نتایج مسیرهای تولیدی CHPSO(M07) که موفق به تأمین قید معادله (۷) نشده‌اند و نقاط برخوردی با عوارض نشان داده شده است.

بیشتر روش در ویژگی متمرکزسازی می‌باشد. بر این مبنا، CHDE(M07) در فرآیند اکتشاف فضای مسئله، پاسخ‌ها را با تمرکز بیشتری جستجو نموده، اما روش CHICA(M09) فضای جستجوی مسئله را با تکیه بیشتری بر قابلیت متنوع سازی اکتشاف کرده است. از دیدگاه کیفیت پاسخ‌ها، بهترین مسیر در بین الگوریتم‌ها توسط CHDE(M07) و نامناسب‌ترین توسط CHPSO(M07) محاسبه گردید.

جهت مقایسه عملکرد الگوریتم‌ها از دید زمان اجرا، در جدول ۷ زمان اجرای متناظر با هر الگوریتم با تکرارهای مختلف ارائه گردیده است. بر اساس نتایج جدول ۷، در ۱۰۰ تکرار CHDE(M07) دارای کمترین زمان اجرای و CHICA(M09) دارای بیشترین زمان اجرا می‌باشد. با افزایش کران تکرارها تا ۲۵۰ حلقه، ترتیب الگوریتم‌ها از دید سرعت محاسبات ثابت بوده، اما با افزایش تعداد تکرارها زمان اجرای CHICA(M09) افزایش بیشتری می‌یابد. در ۵۰۰ تکرار، بیشترین سرعت محاسبات از آن به CHDE(M07) می‌باشد و به ترتیب زمان مسیریاب‌های CHABC(M14)، CHPSO(M07) و CHICA(M09) در رتبه‌های بعدی قرار می‌گیرند. در شکل ۷ زمان اجرای الگوریتم‌ها در تکرارهای مختلف نشان داده شده است.

جدول ۷- مقایسه زمان اجرای CPU مسیر یاب‌های تکاملی با تغییر بیشینه تکرارهای هر الگوریتم برحسب ثانیه

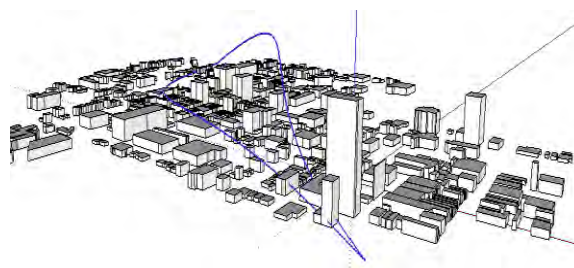
الگوریتم / تکرار	۱۰۰	۲۵۰	۵۰۰
CHDE(M07)	۰.۸/۲۵۱۳	۱۱/۲۶۶۸	۱۵/۸۵۰۴
CHABC(M14)	۱۰/۹۵۵۸	۱۳/۱۸۶۵	۱۷/۳۹۵۶
CHICA(M09)	۱۲/۰۰۷۱	۱۶/۷۸۸۳	۲۲/۴۱۶۸
CHPSO(M07)	۱۱/۷۳۶۷	۱۴/۱۵۰۶	۲۰/۱۵۷۵



شکل ۷- مقایسه زمان اجرای مسیر یاب‌های تکاملی

دینامیکی و محیطی در سناریوهای شهری پرداخته شده است. بدین هدف، در ابتدا مسئله موردبررسی به صورت یک مسئله بهینه‌سازی مقید مدل‌سازی گردید. در ادامه به منظور ارتقاء توانایی الگوریتم‌های DE، PSO، ICA و ABC، ۴ الگوریتم تکاملی بر مبنای تئوری آشوب و با به‌کارگیری ۱۴ سیگنال آشوبی در جایگاه پارامترهای ثابت و مقادیر تصادفی الگوریتم‌های استاندارد طراحی گردید. با اعمال روش‌های پیشنهادی، نتایج مسیریابی به صورت جامع بررسی گردیدند که ارتقاء چشم‌گیری در نتایج نسبت به الگوریتم‌های استاندارد مشاهده شد. تحلیل نتایج مبین کسب برترین نتایج با استفاده از الگوریتم تکامل تفاضلی با مولد آشوبی M07 می‌باشد.

با وجود تحقیقات متنوع و مؤثری که از ۱۹۹۸ تاکنون درباره بررسی تأثیر مثبت آشوب بر فرآیندهای دینامیکی بهینه‌سازی انجام گرفته است، تاکنون هیچ‌گونه اثبات با برهان ریاضی جهت آشکارسازی چگونگی تأثیر دینامیک آشوبی بر فرآیندهای جستجو و اکتشاف الگوریتم‌های تکاملی ارائه نشده است. بررسی چگونگی تأثیر نقشه‌های آشوبی بر عملکرد الگوریتم‌ها و ارائه اثبات ریاضی می‌تواند از موضوعات پژوهشی کلیدی در تحقیقات آینده باشد. در این مقاله از روش‌های تکاملی جهت مسیریابی بهره‌گیری شده است، اما ارزیابی عملکرد الگوریتم‌های کلاسیک بهینه‌سازی جهت رویارویی با مسئله مطرح شده در این مقاله می‌تواند خود موضوع مناسبی برای تحقیقات آتی باشد.

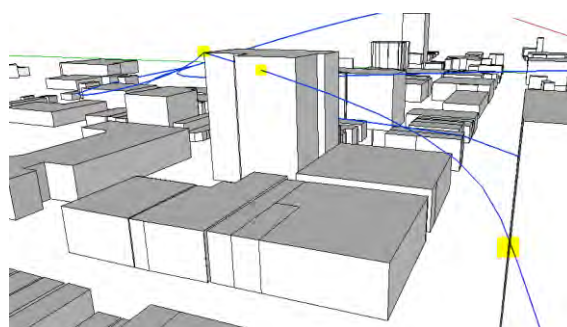


(الف) بهترین و میانگین نتایج برترین مسیریاب CHDE(M07)



(ب) نتایج چهار مسیریاب

شکل ۹- نتایج شبیه‌سازی مسیریابی بهینه سکو



شکل ۱۰- مسیریابی نامناسب CHPSO در ارضاء قید ارتفاع از سطح

۶- نتیجه‌گیری

در پژوهش حاضر، به پیشنهاد و ارزیابی الگوریتم‌های مسیریابی جهت طراحی خطوط سیر بهینه در مأموریت‌های سیستم‌های هوایی بدون سرنشین با قیود

مراجع

- [1] Ergezer, H. and Leblebicioğlu, K. (2013). "Path planning for UAVs for maximum information collection using evolutionary computation". IEEE Trans. Aerosp. Electron. Syst. Vol 49, No.1, PP.502-520.
- [2] Tomic, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E. and Burschka, D. (2012). "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue." Robotics & Automation Magazine, IEEE. Vol.19, No.3, PP.46-56.
- [3] Duan, H. and Li, P. (2014). "Bio-inspired Computation in Unmanned Aerial Vehicles." Springer. Berlin, Heidelberg.
- [4] Duan, H. and Huang, L. (2014). "Imperialist competitive algorithm optimized artificial neural networks for UCAV global path planning." Neurocomputing. Vol.125, PP.166-171.

- [5] Hargraves, C. R. and Paris, S. W. (1987). "Direct trajectory optimization using nonlinear programming and collocation." *Journal of Guidance, Control, and Dynamics*. Vol.10, No.4, PP.338-342.
- [6] Zhang, B., Liu, W., Mao, Z., Liu, J. and Shen, L. (2014). "Cooperative and geometric learning algorithm (CGLA) for path planning of UAVs with limited information." *Automatica*. Vol.50, No.3, PP.809-820.
- [7] Zhuoning, D., Rulin, Z., Zongji, C., and Rui, Z. (2010). "Study on UAV path planning approach based on fuzzy virtual force." *Chinese Journal of Aeronautics*. Vol 23, No.3, PP. 341-350.
- [8] Pehlivanoglu, Y. V. (2012). "A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV." *Aerospace Science and Technology*. Vol.16, No.1, PP.47-55.
- [9] Ott, E. (2002). "Chaos in dynamical systems." Cambridge University Press. Cambridge, UK.
- [10] Gandomi, A. H., and Yang, X. S. (2014). "Chaotic bat algorithm." *Journal of Computational Science*. Vol. 5, No.2, PP.224-232.
- [11] Besada-Portas, E., de la Torre, L., de la Cruz, J. M., and de Andres-Toro, B. (2010). "Evolutionary trajectory planner for multiple UAVs in realistic scenarios." *IEEE Transactions on Robotics*. Vol.26, No.4, PP.619-634.
- [12] Devaney, R. L. (1989). "An introduction to chaotic dynamical systems." Addison-Wesley. Boston.
- [13] Foo, J. L., Knutzon, J., Kalivarapu, V., Oliver, J. and Winer, E. (2009). "Path planning of unmanned aerial vehicles using B-splines and particle swarm optimization." *Journal of aerospace computing, Information, and communication*. Vol.6, No.4, PP.271-290.
- [14] Lu, X., Tang, K., Sendhoff, B., and Yao, X. (2014). "A new self-adaptation scheme for differential evolution." *Neurocomputing*. Vol.146, PP.2-16.
- [15] Karaboga, D., and Basturk, B. (2008). "On the performance of artificial bee colony (ABC) algorithm." *Applied soft computing*. Vol.8, No.1, PP.687-697.
- [16] Zhu, W., Duan, H. (2014). "Chaotic predator prey biogeography based optimization approach for UCAV path planning" *Aerospace Science and Technology*, Vol. 32, No. 1, PP.153–161.
- [17] Xu, C., Duan, H., and Liu, F. (2010). "Chaotic artificial bee colony approach to Uninhabited Combat Air Vehicle (UCAV) path planning" *Aerospace Science and Technology*, Vol. 14, No. 8, PP.535–541.
- [18] Zhang, Y., Wu, L., & Wang, S. (2013). "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization" *Mathematical Problems in Engineering*, Vol.13, PP.1-9.
- [19] Zhou, Z.W., Duan, H.B., Li, P., Di, B. (2013). "Chaotic differential evolution approach for 3D trajectory planning of unmanned aerial vehicle" In: *Proceedings of the 10th IEEE International Conference on Control & Automation (ICCA)*, Hangzhou, China, PP.368-372.
- [20] Zhang, X., & Duan, H. (2015). "An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning." *Applied Soft Computing*, Vol.26, PP.270-284.
- [21] Besada-Portas, E., De La Torre, L., Moreno, A., & Risco-Martín, J. L. (2013). "On the performance comparison of multi-objective evolutionary UAV path planners." *Information Sciences*, Vol.238, PP.111-125
- [22] Szczerba, R. J. (1999). "Threat netting for real-time, intelligent route planners." In: *Proceedings of IEEE International Conference on Information, Decision and Control*, PP. 377-382
- [23] Zheng, C., Li, L., Xu, F., Sun, F., & Ding, M. (2005). "Evolutionary route planner for unmanned air vehicles." *IEEE Transactions on Robotics*, Vol.21, No.4, PP.609-620
- [24] Nikolos, I. K., Zografos, E. S., & Brintaki, A. N. (2007). "UAV path planning using evolutionary algorithms." In *Innovations in Intelligent Machines-1*. Springer Berlin Heidelberg, PP. 77-111

- [25] Roberge, V., Tarbouchi, M., & Labonte, G. (2013). "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning." *IEEE Transactions on Industrial Informatics*, Vol.9, No.1, PP.132-141.
- [26] Jia, D., & Vagners, J. (2004). "Parallel evolutionary algorithms for UAV path planning." In: *Proceedings of the 1st AIAA Intelligent Systems Conference*, PP. 41-57
- [27] Mittal, S., & Deb, K. (2007). "Three-dimensional offline path planning for UAVs using multiobjective evolutionary algorithms." In: *Proceedings of the Congress on Evolutionary Computation (CEC-2007)*, Singapore, PP.3195-3202
- [28] Nikolos, I. K., Valavanis, K. P., Tsourveloudis, N. C., & Kostaras, A. N. (2003). "Evolutionary algorithm based offline/online path planner for UAV navigation." *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol.33, No.6, PP.898-912.
- [29] Macharet, D. G., Neto, A. A., & Campos, M. F. M. (2010). "Feasible UAV path planning using genetic algorithms and Bezier curves." In: *Advances in Artificial Intelligence-SBIA*, Springer Berlin Heidelberg, PP. 223-232
- [30] Hasircioglu, I., Topcuoglu, H. R., & Ermis, M. (2008). "3D path planning for the navigation of unmanned aerial vehicles by using evolutionary algorithms." In *Proceedings of the 10th annual ACM conference on Genetic and evolutionary computation*, PP.1499-1506
- [31] Li, P., & Duan, H. (2012). "Path planning of unmanned aerial vehicle based on improved gravitational search algorithm." *Science China Technological Sciences*, Vol.55, No.10, PP.2712-2719
- [32] Qu, Y. H., Pan, Q., & Yan, J. G. (2005). "Flight path planning of UAV based on heuristically search and genetic algorithms." In *Industrial Electronics Society, IECON 2005. 31st Annual Conference of IEEE*, PP.5-20
- [33] Fu, Y., Ding, M., & Zhou, C. (2012). "Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV." *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, Vol.42, No.2, PP.511-526
- [34] Yokoyama, N., & Suzuki, S. (2005). "Modified genetic algorithm for constrained trajectory optimization." *Journal of guidance, control, and dynamics*, Vol.28, No.1, PP.139-144
- [35] Zhang, R., Zheng, C., & Yan, P. (2007). "Route planning for unmanned air vehicles with multiple missions using an evolutionary algorithm." In: *ICNC 2007, Third IEEE International Conference on Natural Computation*, Vol.3, PP. 23-28
- [36] Heidari, A., Ali Abbaspour, R., Rezaee Jordehi, A. (2015). "An efficient chaotic water cycle algorithm for optimization tasks" *Neural Computing and Applications*, DOI: 10.1007/s00521-015-2037-2