

## تحلیل عملکرد الگوریتم کنترل توزیع شده برای جستجوی مشارکتی توسط سیستم چندعامله

احسان خرمبخت<sup>1</sup>، جعفر روشنیان<sup>2\*</sup>، امیرحسین خدابخش<sup>3</sup>

1- کارشناس ارشد، مهندسی هوافضا، دانشگاه صنعتی خواجه نصیرالدین طوسی، تهران

2- استاد، مهندسی هوافضا، دانشگاه صنعتی خواجه نصیرالدین طوسی، تهران

3- دانشجوی دکترا، مهندسی هوافضا، دانشگاه صنعتی شریف، تهران

\* تهران، صندوق پستی 16569-83911، roshanian@kntu.ac.ir

### چکیده

استفاده از چندین عامل هوشمند توزیع شده به علت مزایای عملیاتی بهتر نظیر مقاوم بودن، پردازش موازی، اسکیل پذیری و بهینگی هزینه به استفاده از یک سیستم جامع متمرکز ترجیح داده می‌شود. این امر در مأموریت‌های هوایی به دلیل ابعاد فضای عملیات و محیط عملکردی نامعین از اهمیت به مراتب بیشتری برخوردار می‌گردد. عملیات جستجو یکی از عملیات‌های متداول در سامانه‌های هوایی چندعامله است. فضای مورد پوشش در یک عملیات جستجو باید به طریقی مطلوب میان عامل‌ها تقسیم شود و هر عامل به نحوی مطلوب در راستای بهترین پوشش ممکنه در فضای اختصاص یافته به آن گام بردارد. با توجه به ساختار شبکه‌های توزیع شده، نحوه تبادل داده، شامل اطلاعات به دست آمده از محیط و جهت حرکت عامل در آینده نزدیک در انتخاب مسیر توسط سایر عامل‌ها مؤثر است. در این مقاله به مسأله جستجوی چندعامله شامل چندین پرنده بدون سرنشین توزیع شده پرداخته شده است. بر این اساس ابتدا مدل سینماتیکی پرنده‌های بدون سرنشین، مدل ریاضی فضای جستجو، حسگر تصویری و ارتباط مخابراتی ارائه شده است. سپس با استفاده از قاعده بیزین، به روزرسانی و به اشتراک گذاری اطلاعات انجام گرفته است. در ادامه طراحی مسیر پرواز پرنده‌ها به یک مسأله کنترل پوشش تعریف شده و به کمک چهار الگوریتم بهینه‌سازی متفاوت بررسی شده است. عملکرد الگوریتم‌ها با انجام چند شبیه‌سازی متفاوت با یکدیگر مقایسه و نتایج تحلیل شده است. نتایج نشان می‌دهد که الگوریتم بهینه‌سازی نقطه درونی بهترین عملکرد را در کاهش عدم قطعیت میانگین فضای جستجو نسبت به الگوریتم‌های گرادیان مرتبه اول، گرادیان مزدوج و برنامه‌ریزی درجه دو داشته است.

### اطلاعات مقاله

مقاله پژوهشی کامل

دریافت: 19 بهمن 1396

پذیرش: 18 فروردین 1397

ارائه در سایت: 07 اردیبهشت 1397

کلید واژگان:

سیستم چندعامله

جستجوی مشارکتی

کنترل توزیع شده

الگوریتم‌های بهینه‌سازی

قاعده بیز

## Performance Study of Distributed Control Algorithm for Cooperative Search using Multi-Agent System

Ehsan Khorrambakht<sup>1</sup>, Jafar Roshanian<sup>1\*</sup>, Amir Hossein Khodabakhsh<sup>2</sup>

1- Department of Aerospace Engineering, Khajeh Nasir Toosi University of Technology, Tehran, Iran.

2- Department of Aerospace Engineering, Sharif University of Technology, Tehran, Iran

\* P.O.B. 16569-83911 Tehran, Iran, roshanian@kntu.ac.ir

### ARTICLE INFORMATION

Original Research Paper  
Received 08 February 2018  
Accepted 07 April 2018  
Available Online 27 April 2018

#### Keywords:

Multi-Agent System  
Cooperative Search  
Distributed Control  
Optimization Algorithms  
Bayes Rule

### ABSTRACT

Vastness of operation airspace and uncertain environment in aerial search missions, makes utilizing multiple intelligent agents more preferable to integrated centralized systems due to robustness, parallel computing structure, scalability, and cost optimality of distributed systems. Cooperative search missions require the search space to be divided properly between agents. In order to minimize the uncertainty, the agents will calculate the best path in the assigned space partition. According to the communication topology, environmental information and the near-future decisions are shared between agents. In this paper, cooperative search using multiple UAVs has been considered. First, mathematical representation of the search space, kinematic and sensor model of UAVs, and communication topology has been presented. Then, an approach has been proposed to update and share information using the Bayes' rule. Afterwards, path planning problem has been solved using different optimization algorithms namely First-order Gradient, Conjugate Gradient, Sequential Quadratic Programming, and Interior Point Algorithm. Finally, the performance of these algorithms has been compared according to mean uncertainty reduction and target detection time. The study shows that the Interior Point Algorithm is the most effective algorithm in reducing the mean uncertainty, both with and without a priori knowledge.

### 1- مقدمه

استفاده از روش توزیع شده در دستیابی به عملکردهای مشارکتی گروهی بهتر، هزینه‌های عملیاتی پایین‌تر، نیازمندی‌های سیستمی کمتر، مقاوم بودن بیشتر در برابر عدم قطعیت‌ها، انطباق بالا، و مقیاس پذیری و انعطاف پذیر مؤثر خواهد بود. مسأله جستجوی همکارانه در بسیاری از مقالات مورد توجه قرار گرفته

در کنترل توزیع شده گروهی از وسایل نقلیه خودمختار معمولاً هدف اصلی این است که کل گروه به یک روش مشارکتی اقدام به انجام مأموریت نمایند. در این‌جا مشارکتی به یک رابطه نزدیک میان همه خودروهایی گروه اشاره می‌کند که در آن به اشتراک گذاری اطلاعات نقشی اساسی بازی می‌کند.

پیدا کردن مسیرشان استفاده می‌کند به نحوی که بتوانند بیشترین اطلاعات گردآوری شده توسط کل تیم را داشته باشند. در [14] پرز کارابازا و همکاران یک برنامه‌ریز تکاملی جدید برای بهینه‌سازی فرامین ورودی چند پرند بدون سرنشین در مأموریت جستجوی هدف ارائه کرده‌اند. این برنامه‌ریز با مسأله جستجو به عنوان یک کنترلر افق زمانی کاهشی چند-هدفه و چند-گامی برخورد می‌کند که قادر است تصمیمات بلند مدت را با توجه به یک معیار بهینه‌سازی بدیع تأمین کند که مشاهدات آینده را وزن‌دهی کرده و قیود مانور پرند بدون سرنشین را در نظر می‌گیرد. یانو و همکاران در [15] یک ساختار توزیع شده سه لایه کنترل با مکانیزم مرکزی برای تولید مسیرهای بهینه چند پرند بدون سرنشین برای جستجوی هدف ارائه کرده‌اند که براساس روشی مدل ترکیبی گاوسی و کنترلر افق زمانی کاهشی است. ژانگ و همکاران در [16] به مسأله جستجوی همکارانه برای یک تیم از پرندهای بدون سرنشین با میدان دید محدود و با بهره بردن از نقشه جستجو پرداخته‌اند. در کار حاضر جستجوی مشارکتی با بهره بردن از نقشه‌های جستجو و تبادل آن میان عامل‌ها انجام می‌گیرد و با بهبود الگوریتم بهینه‌سازی، کاهش عدم قطعیت در زمان کمتری انجام می‌شود.

در ادامه مدل‌سازی مسأله جستجوی همکارانه در بخش دوم انجام پذیرفته و سپس نحوه به‌روزرسانی و تلفیق نقشه جستجو در بخش سوم شرح داده می‌شود. در بخش چهارم طراحی مسیر ارائه شده و شبیه‌سازی در بخش پنجم انجام می‌شود. در نهایت در فصل ششم جمع‌بندی و نتیجه‌گیری ارائه می‌گردد.

## 2- مدل‌سازی ریاضی جستجوی همکارانه

### 2-1- مدل محیط جستجو

ناحیه جستجو  $O \in \mathbb{R}^2$  به صورت یک چندضلعی پیوسته در فضای اقلیدسی دوبعدی در نظر گرفته شده است. تعداد  $N$  پرند بدون سرنشین با استفاده از حسگرهای تصویری اطلاعاتی از ناحیه جستجو برای انجام یک مأموریت جستجوی همکارانه دریافت می‌کنند (شکل 1). ناحیه جستجو به صورت یکنواخت به  $M$  سلول یکسان تقسیم و مساحت هر سلول با  $\delta$  نشان داده می‌شود. هر سلول  $(c)$  به وسیله مرکز آن  $\mu_c = [x_c, y_c]^T$  مشخص می‌شود.  $x_c$  و  $y_c$  مختصات مرکز هر سلول در قاب مختصات زمینی بوده و  $T$  نشان دهنده عملیات ترانزیت است. در هر سلول ناحیه جستجو ممکن است هدف حضور داشته باشد یا حضور نداشته باشد. احتمال حضور هدف درون هر سلول برای پرند بدون سرنشین به صورت توزیع برنولی مدل شده است؛ یعنی  $\theta_c = 1$  (هدف در سلول قرار دارد) با احتمال  $P_{i,k}(\theta_c = 1)$  و  $\theta_c = 0$  (هدف در سلول قرار ندارد) با احتمال  $P_{i,k}(\theta_c = 0) = 1 - P_{i,k}(\theta_c = 1)$  برای سلول  $c$  در لحظه  $k$  مقداردهی می‌شود، که در آن  $i$  نماینده پرند بدون سرنشین  $i$ -ام

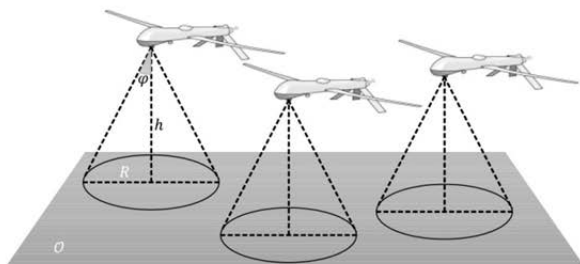


Fig 1 Schematic of Multi-Agent Search Problem

شکل 1 شماتیک مسأله جستجوی چندعامله

است. در سال 2001 پلی‌کاریو و همکاران در [1] چارچوبی برای جستجوی همکارانه تیمی از عامل‌های توزیع شده و اجتناب از موانع و تهدیدات توسعه داده‌اند که براساس دو قسمت وابسته به یکدیگر است: یادگیری برخط محیط و ذخیره‌سازی اطلاعات به صورت «نقشه جستجو». باوم و همکاران در [2] از یک روش جستجوی نظری براساس نقشه‌های «ترخ بازگشت» برای توسعه برنامه‌های جستجوی همکارانه بهره برده‌اند. یانگ و همکاران در [3] یک استراتژی توزیع شده برای جستجوی همکارانه با استفاده از یک روش یادگیری همکارانه فرصت‌طلبانه<sup>1</sup> ارائه کرده‌اند. گوروپراساد و دباسیش در [4] با استفاده از اصل نامتغیر لاسال<sup>2</sup> نشان داده‌اند که یک قانون کنترل توزیع شده که هر عامل را به سمت مرکز قسمت ورونویی آن عامل می‌برد و توسط برد سنسور تنظیم شده است، منجر به قرارگیری بهینه هر گام می‌شود. در [5] یانگ و همکاران به بررسی مسأله ساخت نقشه برای جستجوی همکارانه توسط تیمی از پرندهای بدون سرنشین پرداخته‌اند که در یک محیط ناشناخته و نامعین فعالیت می‌کنند. ریپل و همکاران در [6] یک الگوریتم جستجوی همکارانه افق زمانی کاهشی<sup>3</sup> ارائه کرده‌اند. این الگوریتم هم‌زمان مسیره‌ها و جهت‌گیری سنسورها را برای یک تیم از عامل‌های خودمختار بهینه می‌کند. گن و سوکاربه در [7] یک راهکار بدیع بر مبنای مذاکرات تیمی برای مسأله هماهنگ کردن یک تیم از عامل‌های خودمختار ارائه کرده‌اند. میرزایی و همکاران در [8] یک روش توزیع شده برای حل یک مسأله جستجو و پوشش<sup>4</sup> چندعامله همکارانه ارائه کرده‌اند. عامل‌ها از یک الگوریتم برنامه‌ریزی پویای نگاه به آینده محدود<sup>5</sup> برای پیدا کردن مسیر خود استفاده می‌کنند، همچنین از یک تکنیک بهینه‌سازی مکانی<sup>6</sup> برای اختصاص دادن نواحی ورونویی به وسایل استفاده شده و پایداری سیستم پوشش توسط اصل نامتغیر لاسال تضمین شده است. در [9] هو و همکاران مسأله جستجوی هدف از طریق بهینه‌سازی منطقه پوشش جمعی را در نظر گرفته و عملکرد تشخیص را به عنوان یک مسأله کنترل پوشش<sup>7</sup> همکارانه چندعامله فرموله کرده‌اند. مدل به روزرسانی نقشه و طرح کنترل همکارانه ارائه شده در این مقاله به صورت توزیع شده است. وو و همکاران در [10] یک استراتژی جدید برای عملیات‌های جستجو و نظارت توسط پرندهای بدون سرنشین در یک محیط نامعین ارائه نموده‌اند. در [11] گائو و ژائو عملیات جستجوی همکارانه توسط چندین پرند بدون سرنشین را به صورت یک بازی مدل کرده‌اند که در آن جستجوگران اعضای خود را به تشخیص هدف اختصاص داده و هدف از جستجوگران فرار می‌کند. استراتژی بهینه جستجوگران با استفاده از الگوریتم بهینه‌سازی PSO به دست آمده است. لانیوس و همکاران در [12] روشی بدیع برای هماهنگی یک تیم از سطوح سنجشی خودمختار ارائه کرده‌اند که در حال جستجو برای اهداف تحت عدم قطعیت هستند و یک کنترلر افق زمانی کاهشی<sup>8</sup> زمان-واقعی توسعه داده‌اند. یک روش توزیع شده برای پرداختن به مسأله جستجو و پوشش چندعامله همکارانه در یک محیط نامعین توسط شریفی و همکاران در [13] ارائه شده است. در این مقاله از دو نوع متفاوت عامل، یکی برای جستجو و دیگری برای پوشش بهره برده شده است. عامل‌های جستجوگر از یک الگوریتم برنامه‌ریزی پویای پیش‌بین محدود برای

<sup>1</sup> Opportunistic cooperative learning (OCL)

<sup>2</sup> LaSalle's invariance principle

<sup>3</sup> Receding-horizon

<sup>4</sup> Search and coverage

<sup>5</sup> Limited look-ahead dynamic programming

<sup>6</sup> Locational optimization technique

<sup>7</sup> Coverage control

<sup>8</sup> Receding horizon controller



است. برای سادگی به جای  $P_{i,k}(\theta_c = 1)$  از  $\mathcal{P}_{i,c,k}$  برای توصیف اطلاعات توزیع احتمال هدف استفاده می‌شود.  $\mathcal{P}_{i,c,k} = 1$  به معنی آن است که در لحظه  $k$  در سلول  $c$  هدف حضور داشته و  $\mathcal{P}_{i,c,k} = 0$  یعنی در لحظه  $k$  در سلول  $c$  هدف حضور نداشته است [16].

### 2-2- مدل سینماتیک عامل‌ها

برای سادگی فرض می‌شود که  $N$  پرنده بدون سرنشین مأموریت جستجو را در یک صفحه ثابت موازی با صفحه زمین انجام می‌دهند؛ بنابراین مدل دینامیکی  $i$ -امین پرنده بدون سرنشین به صورت رابطه (1) در نظر گرفته می‌شود [17].

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\psi}_i \\ \dot{v}_i \end{bmatrix} = \begin{bmatrix} v_i \cos \psi_i \\ v_i \sin \psi_i \\ u_i \\ 0 \end{bmatrix} \quad (1)$$

مدل ارائه شده در رابطه (1) در بسیاری از مسائل کنترل همکارانه پرنده‌های بدون سرنشین استفاده شده است [16,4]. در این معادله  $s_i = (x_i, y_i)^T$  نماینده مختصات  $i$ -امین پرنده بدون سرنشین در قاب مختصات زمینی،  $v_i$  سرعت،  $\psi_i$  زاویه سمت و  $\dot{\psi}_{\max}$  حداکثر نرخ چرخش است.

### 3-2- مدل حسگر تصویری

فرض شده است که هر پرنده بدون سرنشین یک حسگر تصویربرداری هوایی را با خود حمل می‌کند. شعاع تصویربرداری آن ( $R$ ) معادل ناحیه دید است (شکل 1). شعاع و میدان دید  $i$ -امین پرنده بدون سرنشین در لحظه  $k$  به صورت روابط (3.2) قابل محاسبه است.

$$R = h \tan \phi \quad (2)$$

$$\mathbb{C}_{i,k} = \{ \xi \in \mathcal{O} \mid \| \xi - s_{i,k} \| \leq R \} \quad (3)$$

$\| \cdot \|$  معادل نرم دوم بردار،  $h$  ارتفاع پروازی  $i$ -امین پرنده بدون سرنشین،  $\phi$  نیم-زاویه میدان دید حسگر،  $\xi$  نقطه اختیاری در فضای جستجو، و  $s_{i,k}$  موقعیت  $i$ -امین پرنده بدون سرنشین در لحظه  $k$  است. شعاع دید  $i$ -امین پرنده بدون سرنشین ممکن است سلول‌های زیادی را در لحظه  $k$  پوشش دهد، اما تنها سلول‌هایی که مرکز آن‌ها درون شعاع دید قرار گرفته باشد به عنوان سلول‌های شناسایی شده توسط حسگر در نظر گرفته می‌شوند [17]. تنها دو نوع نتیجه مشاهده برای هر سلول تعریف می‌شود؛  $Z_{i,c,k} = 1$  که به معنی تشخیص هدف بوده و  $Z_{i,c,k} = 0$  که به معنی عدم تشخیص هدف است. برای تمام سلول‌ها  $P(Z_{i,c,k} = 1 | \theta_c = 1) = p_c$  و  $P(Z_{i,c,k} = 0 | \theta_c = 0) = q_c$  از پیش شناخته شده و ثابت فرض می‌شوند.  $p_c$  و  $q_c$  به ترتیب معرف «احتمال تشخیص» و «احتمال هشدار اشتباه» است.

### 4-2- مدل ارتباطات

تبادل اطلاعات میان پرنده‌های بدون سرنشین از طریق یک شبکه محلی انجام می‌گیرد. توپولوژی شبکه پرنده‌های بدون سرنشین در لحظه  $k$  را می‌توان به وسیله گراف بدون-جهت  $G_k = (\mathcal{E}_k, \mathcal{V})$  مدل کرد. در این گراف هر گره نماینده پرنده بدون سرنشین و یال نماینده لینک ارتباط بین یک جفت پرنده بدون سرنشین است.  $\mathcal{V} = \{1, 2, \dots, N\}$  مجموعه رئوس است که نشان‌دهنده  $N$  پرنده بدون سرنشین است.  $s_{i,k} = [(i, j); i, j \in \mathcal{V}; \|s_{i,k}\| \leq R_c]$  مجموعه یال‌ها بوده که رابطه ارتباطات را نشان می‌دهد و در آن  $R_c$  محدوده ارتباط پرنده بدون سرنشین است. گراف یا شبکه زمانی متصل گفته می‌شود که برای هر دو رأس  $i$  و  $j$  زسکانسی از یال‌ها (مسیر)

### 3- استراتژی تلفیق اطلاعات

#### 3-1- به‌روزرسانی نقشه احتمال به صورت غیرهمکارانه

در یک گروه پرنده بدون سرنشین هر عامل  $i$  یک نقشه احتمال  $\mathcal{P}_{i,c,k}$  از کل ناحیه جستجو خواهد داشت ( $\mathcal{P}_{i,c,k} \triangleq P_{i,k}(\theta_c = 1)$ ). هر عامل باید بتواند نقشه احتمال خود را براساس اندازه‌گیری‌های خود و بدون همکاری یا اشتراک-اطلاعات با دیگر عامل‌ها به‌روزرسانی کند. این نوع به‌روزرسانی باعث می‌شود تا سیستم نسبت به فقدان اتصال ارتباطی میان عامل‌ها مقاوم شود؛ بنابراین در این بخش به‌روزرسانی نقشه احتمال از طریق اندازه‌گیری‌ها برای هر عامل در گروه و بدون همکاری را در نظر می‌گیریم.

روش متداول برای به‌روزرسانی نقشه احتمال توسط اندازه‌گیری‌ها، براساس قاعده بیزین است [20,19] که در روابط (4) ارائه شده است.

$$A = P(Z_{i,c,k} | \theta_c = 1) \mathcal{P}_{i,c,k-1} + P(Z_{i,c,k} | \theta_c = 0) (1 - \mathcal{P}_{i,c,k-1}) \quad (4-a)$$

$$\mathcal{P}_{i,c,k} = \frac{P(Z_{i,c,k} | \theta_c = 1) \mathcal{P}_{i,c,k-1}}{A} \quad (4-b)$$

$$\mathcal{P}_{i,c,k} = \begin{cases} \frac{p \mathcal{P}_{i,c,k-1}}{p \mathcal{P}_{i,c,k-1} + q(1 - \mathcal{P}_{i,c,k-1})} & Z_{i,c,k} = 1 \\ \frac{(1-p) \mathcal{P}_{i,c,k-1}}{(1-p) \mathcal{P}_{i,c,k-1} + (1-q)(1 - \mathcal{P}_{i,c,k-1})} & Z_{i,c,k} = 0 \\ \mathcal{P}_{i,c,k-1} & Z_{i,c,k} \neq 1, 0 \end{cases} \quad (4-c)$$

اگر  $\mathcal{P}_{i,c,0} = 1$  (یا  $\mathcal{P}_{i,c,0} = 0$ ) باشد، آن‌گاه برای تمام  $k > 0$ ،  $\mathcal{P}_{i,c,k} = 1$  (یا  $\mathcal{P}_{i,c,k} = 0$ ) است. از طرف دیگر برای موارد خاصی که یکی از دو پارامتر شناسایی  $p$  و  $q$  معادل صفر یا یک هستند، می‌توان روابط (4) را ساده‌تر کرد. برای نمونه اگر  $\mathcal{P}_{i,c,k} p = 0$  زمانی که عامل  $i$  یک اندازه‌گیری معادل 1 دریافت کند صفر می‌شود و از آن به بعد بدون در نظر گرفتن اندازه‌گیری‌های بعدی تغییری نخواهد کرد. در قسمت‌های بعدی موردی را نظر می‌گیریم که در آن  $0 < \mathcal{P}_{i,c,0} < 1$ ،  $0 < p < 1$  و  $0 < q < 1$  است.

#### 3-2- به‌روزرسانی نقشه احتمال به صورت همکارانه

زمانی که یک گروه از پرنده‌های بدون سرنشین به کار گرفته می‌شوند، ناحیه جستجو عموماً به چندین منطقه مجزا تقسیم‌بندی می‌شود. این مناطق به عنوان مناطق وظیفه برای عامل‌های مختلف در نظر گرفته می‌شوند. هر عامل موظف به جستجو در منطقه وظیفه خود است. ممکن است چنین تقسیم‌بندی فضا استاتیک نباشد، برای نمونه می‌توان به تقسیم‌بندی ورونوویی فضا به ازای موقعیت لحظه‌ای عامل اشاره کرد. اگرچه هر عامل فقط یک قسمت از کل ناحیه را مشاهده می‌کند، هنوز قادر است تا نقشه احتمال خود را با به اشتراک گذاشتن اطلاعات با دیگر پرنده‌های بدون سرنشین به‌روزرسانی کند. عموماً همبستگی بین نقشه‌های احتمال دو عامل ناشناخته است، چرا که توپولوژی جهانی و موقعیت همه پرنده‌های بدون سرنشین در یک شبکه توزیع شده وسیع به سختی توسط عامل‌ها قابل اکتساب است؛

است. در ادامه روش‌های بهینه‌سازی گرادیان مرتبه اول<sup>۱</sup>، گرادیان مزدوج<sup>۲</sup>، برنامه‌ریزی درجه دوم تکراری<sup>۳</sup> و نقطه‌ی میانی<sup>۴</sup> برای یافتن پاسخ بهینه برای رابطه (9) استفاده خواهد شد.

#### 4-1- الگوریتم بهینه‌سازی گرادیان مرتبه اول

روش بهینه‌سازی گرادیان مرتبه اول برای حل مسأله بهینه‌سازی ارائه شده در رابطه (9) قابل استفاده است. رابطه به‌روزرسانی فرمان کنترل گسسته به صورت متداول رابطه (10) قابل بیان خواهد بود [16].

$$u_i(k+1) = u_i(k) - \gamma_i \frac{\partial J}{\partial u_i}(\bar{u}_k) \quad (10)$$

همان طور که در شکل 2 دیده می‌شود،  $\phi_{i,k}$  نسبت به ورودی کنترل  $u_k$  نامتغیر است. رابطه (11) را بنا بر روابط (10,8) به صورت زیر داریم.

$$\begin{aligned} \frac{\partial J}{\partial u_i}(u_k) &= \sum_{j=1}^N \int_{V_j} \frac{\partial \left[ \|s_{j,k+1} - \xi\|^2 \phi_{j,k}(\xi) \right]}{\partial u_{i,k}} d\xi \\ &= \sum_{j=1}^N \int_{V_j} \frac{\partial \left( \|s_{j,k+1} - \xi\|^2 \right)}{\partial u_{i,k}} \phi_{j,k}(\xi) d\xi \end{aligned} \quad (11)$$

رابطه (12) را بنا بر قضیه لاینز [26] داریم.

$$\begin{aligned} \frac{\partial J}{\partial u_i}(u_k) &= \int_{V_i} \frac{\partial \left( \|s_{j,k+1} - \xi\|^2 \right)}{\partial u_{i,k}} \phi_{i,k}(\xi) d\xi \\ &= \int_{V_i} \frac{\partial \left( \|s_{j,k+1} - \xi\|^2 \right)}{\partial s_{i,k+1}} \frac{\partial s_{i,k+1}}{\partial u_{i,k}} \phi_{i,k}(\xi) d\xi \end{aligned} \quad (12)$$

که در آن رابطه (13) به صورت زیر است.

$$\frac{\partial \left( \|s_{j,k+1} - \xi\|^2 \right)}{\partial s_{i,k+1}} = 2(s_{i,k+1} - \xi) \quad (13)$$

و  $(\partial s_{i,k+1} / \partial u_{i,k})$  نسبت به  $\xi$  نامتغیر است؛ بنابراین رابطه (12) را می‌توان به صورت رابطه (14) نوشت.

$$\begin{aligned} \frac{\partial J}{\partial u_i}(u_k) &= \frac{\partial s_{i,k+1}}{\partial u_{i,k}} \int_{V_i} \frac{\partial \left( \|s_{j,k+1} - \xi\|^2 \right)}{\partial s_{i,k+1}} \phi_{i,k}(\xi) d\xi \\ &= \frac{\partial s_{i,k+1}}{\partial u_{i,k}} \int_{V_i} 2(s_{i,k+1} - \xi) \phi_{i,k}(\xi) d\xi \\ &= \frac{\partial s_{i,k+1}}{\partial u_{i,k}} \left[ 2s_{i,k+1} \int_{V_i} \phi_{i,k}(\xi) d\xi - 2 \int_{V_i} \xi \phi_{i,k}(\xi) d\xi \right] \end{aligned} \quad (14)$$

از آنجا که حسگرهای هوایی مورد استفاده در این کار حسگرهای تصویری هستند، انترگرال‌های رابطه (14) را می‌توان با جمع گسسته تقریب زد [21]؛ یعنی  $\int_{V_i} \xi \cdot \phi_{i,k}(\xi) d\xi = \sum_{V_i} \mu_c \phi_{i,k}(c) \delta$  و  $\int_{V_i} \phi_{i,k}(\xi) d\xi = \sum_{V_i} \phi_{i,k}(c) \delta$  که در آن  $\delta$  مساحت هر سلول است.

با فرض این‌که ورودی کنترل در هر بازه زمان گسسته ثابت است، می‌توان از رابطه (1) را به صورت رابطه (15) نوشت.

$$\begin{aligned} s_{i,k+1} &= s_{i,k} + \int_{t_k}^{t_{k+1}} \dot{s}_i(t) dt \\ &= s_{i,k} + \left\{ \begin{aligned} &\frac{v_i}{\psi_{i,k}} [\sin(\psi_{i,k} + \dot{\psi}_{i,k} \cdot T) - \sin(\psi_{i,k})] \\ &+ \frac{v_i}{\dot{\psi}_{i,k}} [-\cos(\psi_{i,k} + \dot{\psi}_{i,k} \cdot T) + \cos(\psi_{i,k})] \end{aligned} \right\} \end{aligned} \quad (15)$$

بنابراین تلفیق اطلاعات دو عامل معمولاً کار دشواری است. در [21] تلاش شده روش تلفیق کاربردی استخراج شود به طوری که نقشه‌های احتمال تمامی پرنده‌های بدون سرنشین را به یک نقشه مشترک همگرا شده و اطلاعات صحیح حضور اهداف آشکار شود.

سپس هر عامل نقشه به‌روزرسانی شده را برای همسایه‌های خود ارسال می‌کند. تلفیق نقشه با استفاده از پروتوکل اجماع صورت می‌پذیرد که در [24-22] ارائه شده است (رابطه (5)).

$$P_{i,c,k} = \sum_{j=1}^N \omega_{i,j,k} P_{j,c,k} \quad (5)$$

در آن  $\omega_{i,j,k}$  با توجه به توپولوژی شبکه مقاردهی می‌شود که نحوه محاسبه آن در رابطه (6) آمده است.

$$\omega_{i,j,k} = \begin{cases} 1 - \left( \frac{n_{i,k} - 1}{N} \right) & \text{اگر } j \in N_{i,k}, (j = i) \\ \frac{1}{N} & \text{اگر } j \in N_{i,k}, (j \neq i) \\ 0 & \text{اگر } j \notin N_{i,k} \end{cases} \quad (6)$$

#### 4- طراحی مسیر

هدف نهایی جستجوی همکارانه یافتن اهداف بیشتر و کاهش عدم قطعیت کل ناحیه جستجو با بیشترین سرعت ممکن است؛ بنابراین طراحی مسیر پرنده‌های بدون سرنشین نقش مهمی در روند جستجوی ناحیه دارد.

ابتدا نیاز است تا نقشه عدم قطعیت کل ناحیه براساس نقشه احتمال تعریف شود. اگر  $P_{i,c,k}$  معادل 0.5 باشد، گفتن این‌که هدف در آن سلول قرار دارد یا خیر دشوار است و نامعینی بیشترین مقدار را دارد. زمانی که  $P_{i,c,k}$  به یک یا صفر می‌رسد، به ترتیب هدف در سلول حضور دارد یا حضور ندارد. از این‌رو عدم قطعیت برای هر سلول به صورت رابطه (7) تعریف می‌شود.

$$\phi_{i,k}(c) = e^{-k_\phi \left| \ln \left( \frac{P_{i,c,k}}{1-P_{i,c,k}} \right) \right|} \quad (7)$$

در آن  $k_\phi$  یک ضریب بهره مثبت است. از نظر تئوری هر مقدار مثبتی برای  $k_\phi$  قابل قبول است؛ بر این اساس مقدار مناسب  $k_\phi$  به گونه‌ای انتخاب می‌شود که تابع عدم قطعیت خیلی سریع یا خیلی آهسته کاهش نیابد.

در روند جستجوی ناحیه هر پرنده بدون سرنشین برای کاهش عدم قطعیت، مشاهدات پیوسته‌ای از طریق یک دوربین رو به زمین انجام می‌دهد. بنابراین می‌توان آن را به صورت یک مسأله پوشش‌دهی بهینه فرموله کرد؛ یافتن ورودی کنترل بهینه‌ای که بتواند یک تابع هزینه عملکرد پوشش‌دهی را کمینه کرده و کاهش عدم قطعیت را پیشینه نماید.

تابع عملکرد پوشش‌دهی را به صورت رابطه (8) تعریف می‌کنیم [25].

$$J(\vec{s}_k, \vec{u}_k, \vec{\phi}_k) = \sum_{i=1}^N \int_{V_i} \|\vec{s}_{i,k+1} - \xi\|^2 \vec{\phi}_{i,k}(\xi) d\xi \quad (8)$$

در آن ناحیه ورونویی  $i$ -امین پرنده بدون سرنشین [21] و  $\phi_k = [\phi_{1,k}, \phi_{2,k}, \dots, \phi_{N,k}]^T$  است. به این ترتیب مسأله پوشش‌دهی به صورت مسأله کنترل بهینه نشان‌داده شده در رابطه (9) قابل بیان خواهد بود.

$$\begin{aligned} \vec{u}_k^* &= \arg_{u_k} \min J(\vec{s}_k, \vec{u}_k, \vec{\phi}_k) \\ \|\vec{u}_k^*\| &\leq \vec{u}_{\max} \\ 0 &\leq \vec{s}_k \leq \vec{s}_{\max} \end{aligned} \quad (9)$$

در آن  $\vec{u}_k^* = [u_{1,k}, u_{2,k}, \dots, u_{N,k}]^T$  بردار محدودکننده فضای جستجو

<sup>1</sup> First-Order Gradient

<sup>2</sup> Conjugate Gradient

<sup>3</sup> Sequential Quadratic Programming (SQP)

<sup>4</sup> Interior Point Optimization

قیده‌های مطرح شده اصلاح گردد. در این روش نیز به منظور لحاظ کردن قيود مانند روش قبل از روش گرادیان مقید استفاده می‌شود.

### 3-4- طراحی مسیر براساس روش بهینه‌سازی برنامه‌ریزی درجه دوم تکراری

روش بهینه‌سازی برنامه‌ریزی درجه دوم تکراری برای حل مسائل بهینه‌سازی مقید غیرخطی با تابع هدف و توابع قید مشتق‌پذیر به کار می‌رود (رابطه (22)). این روش بهینه‌سازی مسأله را به صورت یک توالی از زیرمسأله‌های برنامه‌ریزی درجه دوم در می‌آورد. این زیرمسأله‌ها باید به ترتیب حل شوند. حوزه کاربرد این الگوریتم با توجه به دو فرض زیر تعیین می‌گردد [29]:

- 1- مسأله هموار باشد: توابع مسأله همواره مشتق‌پذیر باشند.
- 2- مسأله کوچک باشد: اندازه مسأله به توان سخت‌افزاری و توانمندی راهکار برنامه‌ریزی درجه دوم برای حل مسائل بزرگ وابسته است. این الگوریتم قادر است مسائلی که تا 100 متغیر دارند را حل نماید.

### 4-4- طراحی مسیر براساس روش بهینه‌سازی نقطه درونی

در این بخش طراحی و پیاده‌سازی الگوریتم نقطه درونی برای حل مسأله بهینه‌سازی مقید غیرخطی رابطه (22) را بررسی می‌کنیم.

$$\begin{aligned} \min f(X) \\ H(X) = 0 \text{ (مقید به)} \\ G(X) \leq 0 \end{aligned} \quad (22)$$

در آن  $f: R^n \rightarrow R$ ،  $H: R^n \rightarrow R^k$  و  $G: R^n \rightarrow R^j$  توابع هموار هستند. این الگوریتم برای مسأله‌های غیرمحدب و تعداد متغیر  $n$  بزرگ طراحی شده است. در این الگوریتم به مشتقات اول و دوم تابع هزینه و قيود نیاز هست، اما می‌توان استراتژی طراحی را بسط داده و از تقریب‌های شبه-نیوتون برای حل استفاده کرد. الگوریتم‌های نقطه درونی جایگزینی برای روش‌های مجموعه فعال<sup>1</sup> برای در نظر گرفتن قيود نامساوی در اختیار قرار می‌دهند. این الگوریتم از دو ابزار قدرتمند برای حل مسائل غیرخطی بهره می‌برد: 1- برنامه‌ریزی درجه دوم تکرارشونده و 2- تکنیک‌های ناحیه مورد اطمینان<sup>2</sup>. از ایده‌های برنامه‌ریزی درجه دوم تکراری برای در نظر گرفتن مؤثر رفتار غیرخطی قيود استفاده شده است. استراتژی‌های ناحیه مورد اطمینان به الگوریتم اجازه می‌دهد تا با مسائل محدب و غیرمحدب به طور یکنواخت برخورد کند، همچنین این استراتژی‌ها اجازه می‌دهد تا الگوریتم مستقیم از مشتق دوم اطلاعات استفاده کند و در صورت حضور گرادیان‌های تقریباً وابسته قید از واگرایی الگوریتم جلوگیری می‌کند [30].

### 5- شبیه‌سازی و نتایج

#### 5-1- تعریف مسأله معیار

فضای جستجو 5000:5000 مترمربع در نظر گرفته و آن را یکنواخت به سلول‌های مربعی هم اندازه به طول 100 متر تقسیم می‌کنیم. زمان شبیه‌سازی 250 ثانیه و گام زمانی 0.5 ثانیه در نظر گرفته می‌شود. از 4 عامل به منظور جستجوی فضا استفاده می‌شود. موقعیت اولیه این عامل‌ها به صورت خطی در راستای محور  $x$  در نظر گرفته می‌شود. نخستین عامل در  $\gamma = 0$  [m] و  $x = 300$  [m] و باقی به فاصله 10 متر به ترتیب از یکدیگر قرار می‌گیرند. زاویه سمت اولیه نیز  $\psi_0 = 90^\circ$  فرض شده‌است. ارتفاع پرواز،

در آن  $T$  زمان نمونه برداری است. رابطه (16) را با مشتق‌گیری از رابطه (15) نسبت به دستور نرخ چرخش  $\psi_{i,k}$  خواهیم داشت.

$$\frac{\partial s_{i,k+1}}{\partial u_{i,k}} = \frac{\begin{bmatrix} v_i T \cos(\psi_{i,k}) - x_{i,k+1} + x_{i,k} \\ \psi_{i,k} \\ v_i T \sin(\psi_{i,k}) - y_{i,k+1} + y_{i,k} \\ \psi_{i,k} \end{bmatrix}}{\psi_{i,k}} \quad (16)$$

با روابط (16,14)، مشتق جزئی در رابطه (12) را می‌توان به صورت تحلیلی و با این روش بردار ورودی کنترل را می‌توان به دست آورد.

باید توجه کرد که روش یادشده برای حل مسأله بهینه‌سازی بدون قید است، در صورتی که پرنده بدون سرنشین قید مانورپذیری و یا به عبارت دیگر محدودیت نرخ چرخش خواهد داشت. این امر در فرمول‌بندی مسأله کنترل بهینه به صورت قید بیشینه فرمان کنترل ارائه شده است. از این رو ورودی کنترل  $i$ -امین پرنده بدون سرنشین به صورت رابطه (17) محدود می‌شود.

$$-\psi_{\max} \leq u_i \leq \psi_{\max} \quad (17)$$

پیاده‌سازی روش گرادیان مقید یکی از روش‌های در نظر گرفتن قید در فرمان کنترل است [27]. در این روش فاکتور مقید  $C_u$  برای هر متغیر کنترل  $\hat{u}_i \in \mathbb{R}$  تعریف می‌شود، به گونه‌ای که رابطه (18) را داریم.

$$u_i = C_u \hat{u}_i = \begin{cases} -\psi_{\max} & \hat{u}_i \leq -\psi_{\max} \\ \hat{u}_i & -\psi_{\max} \leq \hat{u}_i \leq \psi_{\max} \\ \psi_{\max} & \hat{u}_i \geq \psi_{\max} \end{cases} \quad (18)$$

همان طور که از رابطه (18) مشخص است،  $u_i$  قید نامساوی رابطه (17) را ارضا می‌کند، سپس  $\hat{u}_i$  با حل رابطه (10) به دست آمده و با  $u_i = C_u \hat{u}_i$  جایگزین می‌شود تا در محاسبات دور بعدی استفاده شود.

### 4-2- الگوریتم بهینه‌سازی گرادیان مزدوج

برای حل مسأله (9) روش بهینه‌سازی گرادیان مزدوج، به روزرسانی کنترل از رابطه (19) پیروی می‌کند [28].

$$u_i(k+1) = u_i(k) + \gamma_i p_k \quad (19)$$

در آن نرخ یادگیری و  $p_k$  جهت جستجو به صورت رابطه (20) است.

$$\begin{aligned} p_k &= -g_k + \beta_k p_{k-1} \\ g_k &= \frac{\partial J}{\partial u}(u_k) \end{aligned} \quad (20)$$

در رابطه بالا  $\beta_k$  یک اسکالر بوده و می‌توان آن را از رابطه (21) به دست آورد.

$$\beta_k = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \quad (21)$$

همچنین گرادیان  $\partial J / \partial u$  را می‌توان با کمک روابط (16,14) به صورت تحلیلی به دست آورد. با این روش بردار ورودی کنترل در رابطه (19) قابل محاسبه خواهد بود. در روش گرادیان مزدوج نیز قيود مسأله بهینه‌سازی مد نظر قرار نمی‌گیرند. بر این اساس لازم است تا خروجی آن‌ها با توجه به

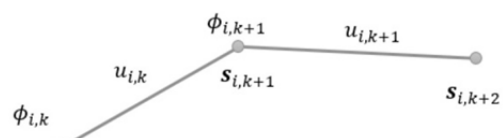


Fig 2 Discreet movement of an agent

شکل 2 حرکت گسسته یک عامل

<sup>1</sup> Active-Set  
<sup>2</sup> Trust Region



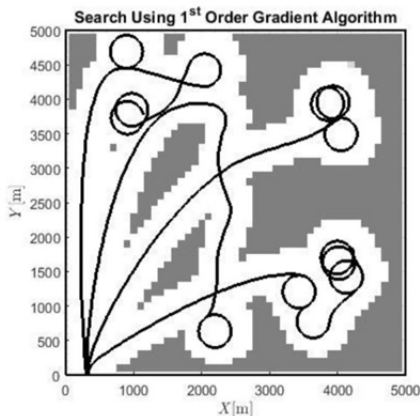


Fig 3 Cooperative Search Using First-Order Gradient Algorithm

شکل 3 جستجوی مشارکتی با استفاده از الگوریتم گرادیان مرتبه اول

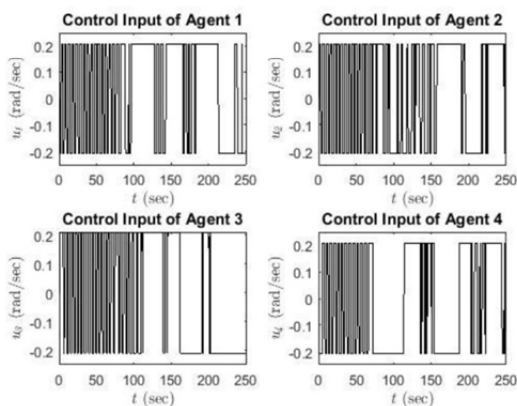


Fig 4 Distributed Control Using First-Order Gradient Algorithm

شکل 4 کنترل توزیع شده با استفاده از الگوریتم گرادیان مرتبه اول

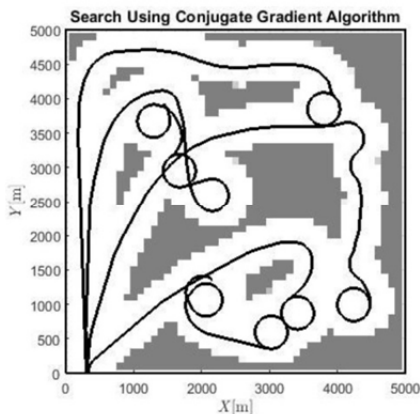


Fig 5 Cooperative Search Using Conjugate Gradient Algorithm

شکل 5 جستجوی مشارکتی با استفاده از الگوریتم گرادیان مزدوج

3-1-5- شبیه‌سازی مسأله معیار با استفاده از الگوریتم برنامه‌ریزی درجه دوم تکراری

حل مسأله با استفاده از الگوریتم بهینه‌سازی برنامه‌ریزی درجه دوم تکراری با بهره‌گیری از نرم‌افزار متلب انجام گرفته است. این الگوریتم بهینه‌سازی را می‌توان با استفاده از جعبه ابزاری فراخوانی کرد که برای حل مسائل بهینه‌سازی مقید در نرم‌افزار متلب توسعه داده شده است. برای بهره‌برداری از این تابع متلب باید اندازه تابع هزینه در گام بعدی محاسبه شده و با مقدار آن

سرعت و بیشینه نرخ چرخش هر عامل ثابت در نظر گرفته شده و به ترتیب برابر  $\dot{\psi}_{max} = 12$  [°/s] و  $v = 50$  [m/s]،  $h = 5000$  [m]، نیم‌زاویه میدان دید حسگر  $\phi = 3.45^\circ$  است. احتمال تشخیص و احتمال هشدار اشتباه به ترتیب  $p = 0.9$  و  $q = 0.1$  است. این مقادیر در جدول 1 خلاصه آورده شده است.

5-1-1- شبیه‌سازی مسأله معیار با استفاده از الگوریتم گرادیان مرتبه اول در شبیه‌سازی ابتدا تعریف مسأله و مقاردهی اولیه انجام می‌گیرد. سپس با توجه به تعریف فضا، موقعیت مراکز سلول‌ها، تقسیم‌بندی و رزونوی و فاصله عامل‌ها از یکدیگر محاسبه می‌شود. با کمک مدل ارتباطات و مدل حسگر به ترتیب همسایه‌های هر عامل و سلول‌های درون میدان دید هر عامل مشخص می‌گردند. نقشه‌های احتمال با رویکرد غیرهمکارانه به‌روزرسانی شده و سپس تلفیق اطلاعات میان عامل‌ها به صورت همکارانه صورت می‌پذیرد. عدم قطعیت با استفاده از نقشه جدید به روزرسانی می‌گردد. در نهایت این اطلاعات وارد الگوریتم بهینه‌سازی گرادیان مرتبه اول می‌شود. با محاسبه مشتقات جزئی شاخص بهینگی نسبت به بردار وضعیت  $(\partial J/\partial s)$  و بردار وضعیت نسبت به ورودی کنترلی  $(\partial s/\partial u)$ ، مقدار مشتق جزئی شاخص بهینگی نسبت به ورودی کنترلی  $(\partial J/\partial u)$  برای هر عامل محاسبه می‌گردد. ورودی کنترل هر عامل براساس این مشتق جزئی به‌روزرسانی می‌شود. پس از مقید کردن ورودی کنترل به صورت شرطی مقدار آن به عنوان ورودی کنترل اولیه گام زمانی بعدی حل در نظر گرفته شده است. نتایج شبیه‌سازی در شکل‌های 3 و 4 نشان داده شده است.

5-1-2- شبیه‌سازی مسأله معیار با استفاده از الگوریتم گرادیان مزدوج

در این‌جا در گام اول شبیه‌سازی از مقدار گرادیان شاخص بهینگی نسبت به ورودی کنترلی  $(\partial J/\partial u)$  به عنوان جهت جستجو  $(p_0)$  استفاده می‌شود. در گام‌های بعدی با استفاده از رابطه (20) و داشتن مقدار  $\partial J/\partial u$  در هر گام زمانی جهت جستجو به صورت تکراری به‌روزرسانی می‌گردد. برای محاسبه مقدار  $\beta$  در هر گام زمانی از رابطه (21) استفاده می‌شود. نتایج شبیه‌سازی در شکل‌های 5 و 6 نشان داده شده است.

جدول 1 پارامترهای اولیه شبیه‌سازی معیار

Table 1 Standard Problem Initial Parameters

پارامتر	مقدار
ابعاد فضا	5000 × 5000 (متر مربع)
طول هر سلول	100 متر
زمان شبیه‌سازی	250 ثانیه
گام زمانی حل‌گر	0.5 ثانیه
موقعیت اولین عامل	$x = 300$ $y = 0$
فاصله عامل‌ها	$\Delta x = 10$ $\Delta y = 0$
سرعت عامل‌ها	50 متر بر ثانیه
ارتفاع عامل‌ها	5000 متر
زاویه سمت اولیه	90 درجه
حداکثر نرخ چرخش	12 درجه
احتمال تشخیص	0.9
احتمال هشدار اشتباه	0.1
ضریب بهره عدم قطعیت	0.85

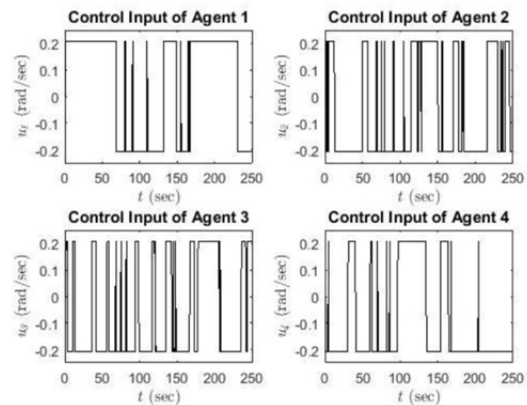
نرم افزار متلب توسعه داده شده است و می توان آن را توسط جعبه ابزار حل مسائل بهینه سازی مقید فراخوانی کرد. نتایج شبیه سازی در شکل های 9 و 10 نشان داده شده اند.

**5-1-5- مقایسه نتایج حل مسأله معیار**

نتایج شبیه سازی های بالا در شکل 11 با یکدیگر مقایسه شده اند. عملکرد ابتدایی الگوریتم گرادیان مزدوج ضعیف تر از سه الگوریتم دیگر بوده، اما با گذشت زمان به نتایج بهتری نسبت به الگوریتم گرادیان مرتبه اول رسیده است. الگوریتم های نقطه درونی و برنامه ریزی درجه دوم تکراری نیز در نهایت عملکرد بهتری از دو الگوریتم دیگر داشته و بهترین عملکرد را الگوریتم نقطه درونی داشته است. میل کردن تغییرات عدم قطعیت میانگین به صفر در الگوریتم های گرادیان مرتبه اول و گرادیان مزدوج نیز در این شکل مشخص است.

**5-2- بررسی تأثیر داشتن دانش اولیه**

هدف اصلی از جستجوی مشارکتی پیدا کردن اهداف حاضر در فضای جستجو است. تا به این جا شبیه سازی ها بدون حضور هرگونه هدف در فضا و تنها برای مقایسه عملکرد الگوریتم های جستجو در راستای کمینه کردن عدم قطعیت میانگین در کمترین زمان ممکن انجام پذیرفت؛ بنابراین عامل ها نیازی به داشتن دانش اولیه نسبت به محیط نداشتند. از این رو نقشه احتمال اولیه یکنواختی در اختیار عامل ها قرار گرفت و احتمال حضور هدف در تمامی



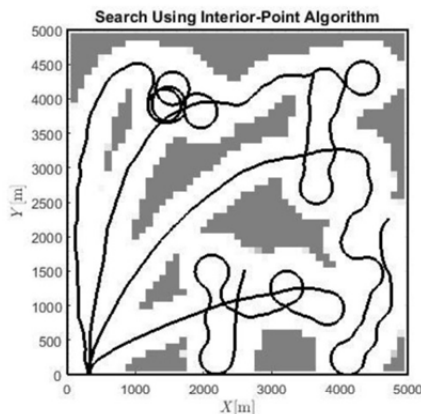
**Fig 6** Distributed Control Using Conjugate Gradient Algorithm

شکل 6 کنترل توزیع شده با استفاده از الگوریتم گرادیان مزدوج

در گام کنونی مقایسه شود. سپس الگوریتم با توجه به بیشترین مقدار کاهش تابع هزینه ورودی کنترلی را تولید می کند. نتایج شبیه سازی با بهره گیری از این الگوریتم در شکل های 7 و 8 نشان داده شده است.

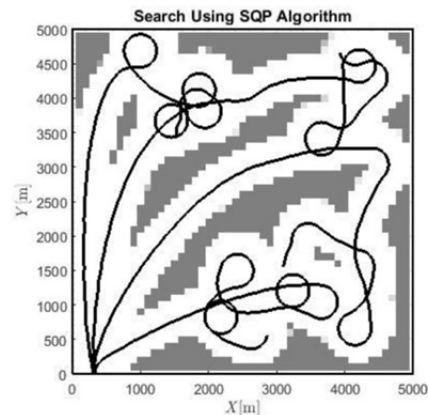
**5-1-4- شبیه سازی مسأله معیار با استفاده از الگوریتم نقطه درونی**

در این قسمت از الگوریتم بهینه سازی نقطه درونی بهره برده شده است. این الگوریتم نیز مانند الگوریتم بهینه سازی برنامه ریزی درجه دوم تکراری در



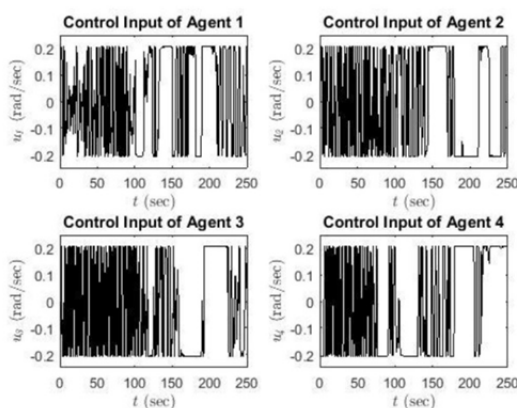
**Fig 9** Cooperative Search Using Interior-Point Algorithm

شکل 9 جستجوی مشارکتی با استفاده از الگوریتم نقطه درونی



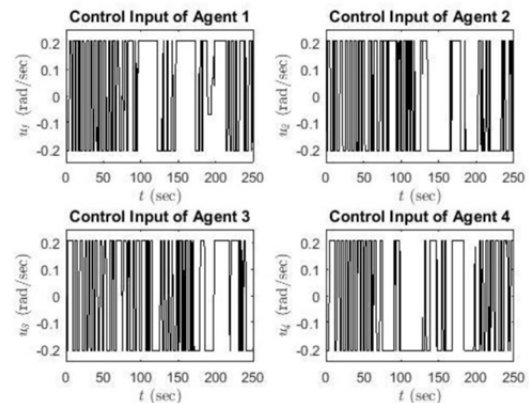
**Fig 7** Cooperative Search Using SQP Algorithm

شکل 7 جستجوی مشارکتی با استفاده از الگوریتم برنامه ریزی درجه دوم تکراری



**Fig 10** Distributed Control Using Interior-Point Algorithm

شکل 10 کنترل توزیع شده با استفاده از الگوریتم نقطه درونی



**Fig 8** Distributed Control Using SQP Algorithm

شکل 8 کنترل توزیع شده با استفاده از الگوریتم برنامه ریزی درجه دوم تکراری

می‌دهند.

همان طور که در شکل 16 نمایش داده شده است، الگوریتم بهینه‌سازی گرادیان مرتبه اول نتوانسته تمامی اهداف را در زمان شبیه‌سازی بباید. عامل‌ها با استفاده از این الگوریتم نتوانسته‌اند تنها چهار هدف را ببینند.

**5-2-2- جستجوی اهداف توسط الگوریتم گرادیان مزدوج**

شبیه‌سازی جستجوی مشارکتی با داشتن دانش اولیه توسط الگوریتم بهینه‌سازی گرادیان مزدوج انجام گرفته است. نتایج این شبیه‌سازی در شکل‌های 17 و 18 نشان داده شده است.

شکل 19 نشان می‌دهد که عامل‌ها نتوانسته‌اند با استفاده از الگوریتم گرادیان مزدوج همه اهداف را ببینند.

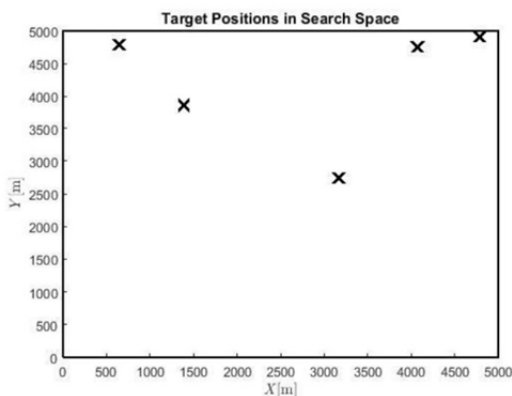


Fig 12 Targets Positions

شکل 12 موقعیت قرارگیری اهداف

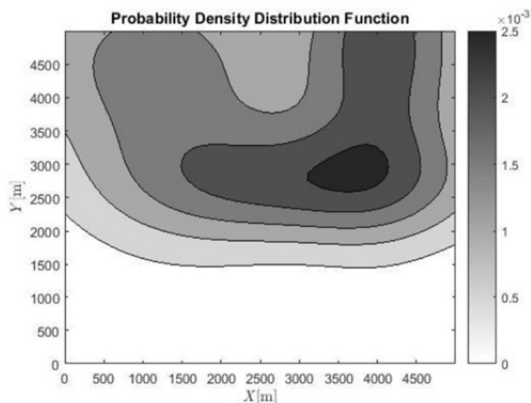


Fig 13 Targets Probability Density Distribution Function

شکل 13 تابع توزیع چگالی احتمالی اهداف

جدول 2 پارامترهای تغییر یافته در شبیه‌سازی با دانش اولیه.

**Table 2** Deviated Parameters in Simulation with Prior Knowledge

پارامتر	مقدار
ابعاد فضا	5000 × 5000 (متر مربع)
زمان شبیه‌سازی	150 ثانیه
گام زمانی حل‌گر	0.5 ثانیه
موقعیت اولین عامل	$\begin{cases} x = 400 \\ y = 0 \end{cases}$
فاصله عامل‌ها	$\begin{cases} \Delta x = 10 \\ \Delta y = 0 \end{cases}$
سرعت عامل‌ها	100 متر بر ثانیه
ضریب بهره عدم قطعیت	0.45

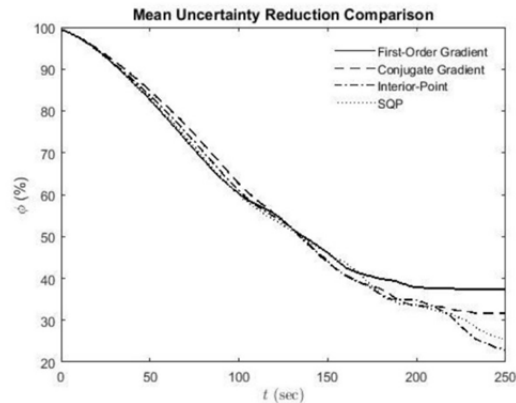


Fig 11 Performance of Algorithms in Reducing Mean Uncertainty

شکل 11 مقایسه عملکرد الگوریتم‌های بهینه‌سازی در کاهش عدم قطعیت میانگین

سلول‌ها یکنواخت 0.5 در نظر گرفته شده بود. در این بخش اهداف نیز به شبیه‌سازی اضافه شده و دانش اولیه‌ای از موقعیت آن‌ها به صورت توزیع احتمال گاوسی در اختیار عامل‌ها قرار می‌گیرد.

اهدافی که در این بخش تعریف می‌شوند، اهداف ساکن بوده و موقعیت آن‌ها اتفاقی تعیین می‌گردد. از آن‌جا که پارامترهای زیادی در مسأله وجود دارد، موقعیت اتفاقی اهداف ذخیره شده و برای همه شبیه‌سازی‌ها از یک موقعیت یکسان استفاده می‌شود، همچنین تعداد اهداف 5 عدد تعیین می‌گردد. موقعیت قرارگیری اهداف در شکل 12 نشان داده شده است. دانش اولیه‌ای که در اختیار عامل‌ها قرار داده می‌شود یک توزیع چگالی احتمالی گاوسی با توجه به موقعیت قرارگیری اهداف است. این توزیع گاوسی چگالی احتمال در شکل 13 نمایش داده شده است. تعداد اهداف برای عامل‌ها مجهول بوده و پس از پیدا کردن تمامی اهداف نیز تا پایان زمان شبیه‌سازی به جستجو ادامه می‌دهند. از آن‌جا که در کار کنونی از پردازش تصویر برای تشخیص اهداف استفاده نشده است، مقایسه ناحیه دید سنسور عامل‌ها با موقعیت قرارگیری اهداف، حضور یا عدم حضور هدف در هر سلول را مشخص می‌کند، سپس با توجه به حضور یا عدم حضور هدف در ناحیه دید عامل و با استفاده از رابطه (4) احتمال سلول‌ها به‌روزرسانی می‌شود. در صورتی که هدف در ناحیه دید عامل قرار داشته باشد، بنا بر مقادیر احتمال تشخیص و احتمال هشدار اشتباه پس از تعداد مشخصی رؤیت سلول، احتمال آن سلول به سمت یک میل می‌کند (در صورتی که هدف در آن سلول نباشد این مقدار به سمت صفر میل خواهد کرد).

در این بخش برخی از پارامترهای شبیه‌سازی‌ها تغییر یافته‌اند. این تغییرات عبارت از کاهش زمان جستجو به 150 ثانیه، افزایش سرعت عامل‌ها به 100 متر بر ثانیه و موقعیت اولیه قرارگیری عامل‌ها در  $x = 0$  و  $y = 400$  متر به فاصله 10 متر از یکدیگر است. پارامترهای شبیه‌سازی در جدول 2 قابل مشاهده است. در ادامه جستجوی مشارکتی با داشتن دانش اولیه به ترتیب توسط الگوریتم‌های بهینه‌سازی گرادیان مرتبه اول، گرادیان مزدوج، برنامه‌ریزی درجه دوم تکراری و نقطه درونی شبیه‌سازی شده و نتایج ارائه می‌گردند.

**5-2-1- جستجوی اهداف توسط الگوریتم گرادیان مرتبه اول**

جستجوی اهداف با داشتن دانش اولیه با استفاده از الگوریتم بهینه‌سازی گرادیان مرتبه اول شبیه‌سازی شده است. شکل‌های 14 و 15 به ترتیب نحوه حرکت عامل‌ها در فضای جستجو و ورودی‌های کنترلی عامل‌ها را نشان



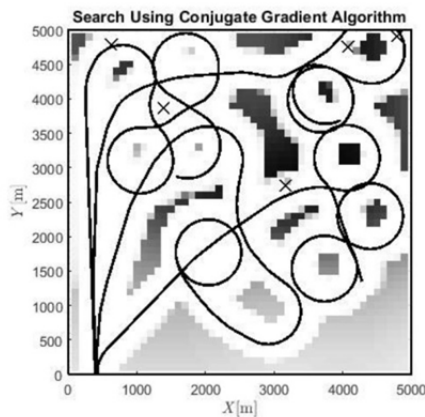


Fig 17 Cooperative Search Using Conjugate Gradient Algorithm

شکل 17 جستجوی مشارکتی با استفاده از الگوریتم گرادیان مزدوج

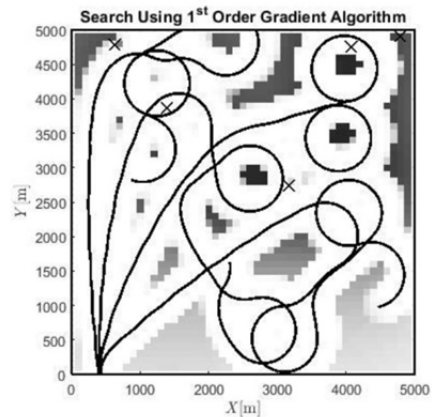


Fig 14 Cooperative Search Using 1<sup>st</sup> Order Gradient Algorithm

شکل 14 جستجوی مشارکتی با استفاده از الگوریتم گرادیان مرتبه اول

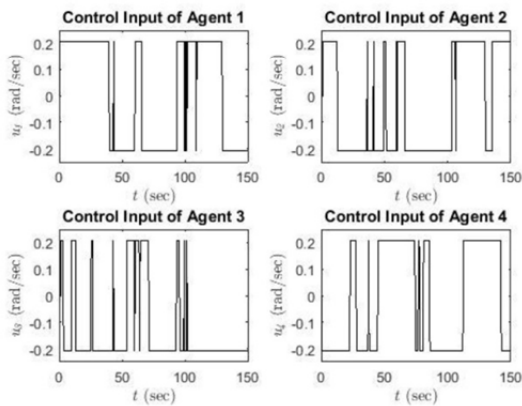


Fig 18 Distributed Control Using Conjugate Gradient Algorithm

شکل 18 کنترل توزیع شده با استفاده از الگوریتم گرادیان مزدوج

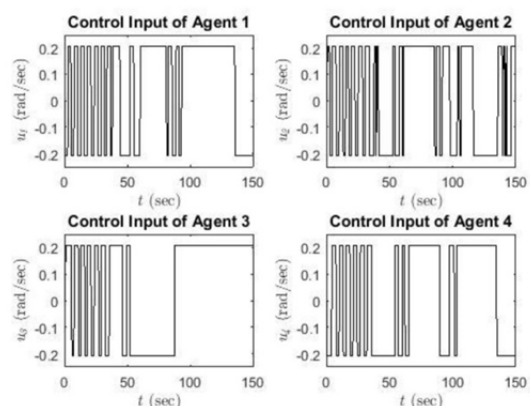


Fig 15 Distributed Control Using First-Order Gradient Algorithm

شکل 15 کنترل توزیع شده با استفاده از الگوریتم گرادیان مرتبه اول

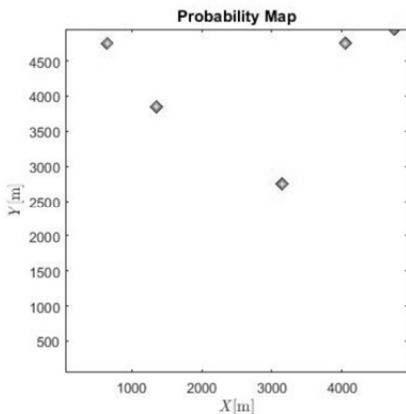


Fig 19 Found Targets Using Conjugate Gradient Algorithm

شکل 19 اهداف پیدا شده توسط الگوریتم گرادیان مزدوج

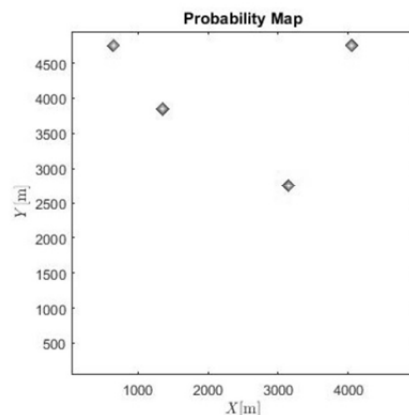


Fig 16 Found Targets Using First-Order Gradient Algorithm

شکل 16 اهداف پیدا شده توسط الگوریتم گرادیان مرتبه اول

5-2-3- جستجوی اهداف توسط الگوریتم نقطه درونی

در این بخش شبیه‌سازی جستجوی مشارکتی با دانش اولیه و با بهره‌گیری از الگوریتم بهینه‌سازی نقطه درونی انجام گرفته است. نتایج این شبیه‌سازی در شکل‌های 23 و 24 نشان داده شده است.

چنانچه در شکل 25 روشن است، تمامی اهداف توسط عامل‌ها مشاهده شدند؛ بنابراین عامل‌ها با استفاده از الگوریتم بهینه‌سازی نقطه درونی قادر بودند این جستجو را به انجام برسانند.

5-2-3- جستجوی اهداف توسط الگوریتم برنامه‌ریزی درجه دوم تکراری

جستجوی مشارکتی با داشتن دانش اولیه توسط الگوریتم بهینه‌سازی برنامه‌ریزی درجه دوم تکراری در این بخش انجام گرفته است. نتایج این شبیه‌سازی در شکل‌های 20 و 21 نشان داده شده است.

همان‌طور که شکل 22 نشان می‌دهد، عامل‌ها با استفاده از الگوریتم برنامه‌ریزی درجه دوم تکراری توانسته‌اند تنها چهار هدف را شناسایی کنند.

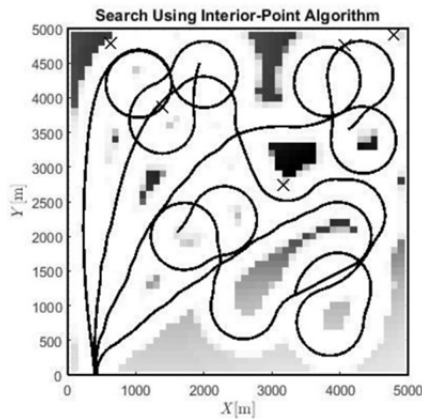


Fig 23 Cooperative Search Using Interior-Point Algorithm

شکل 23 جستجوی مشارکتی با استفاده از الگوریتم نقطه درونی

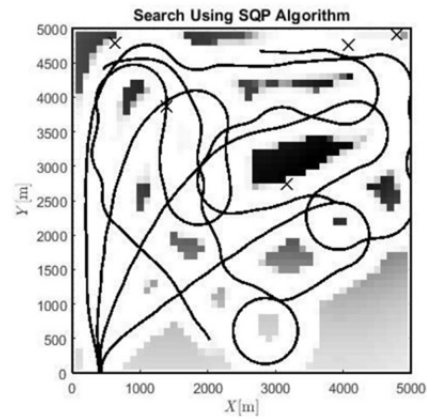


Fig 20 Cooperative Search Using SQP Algorithm

شکل 20 جستجوی مشارکتی با استفاده از الگوریتم برنامه‌ریزی درجه دوم تکراری

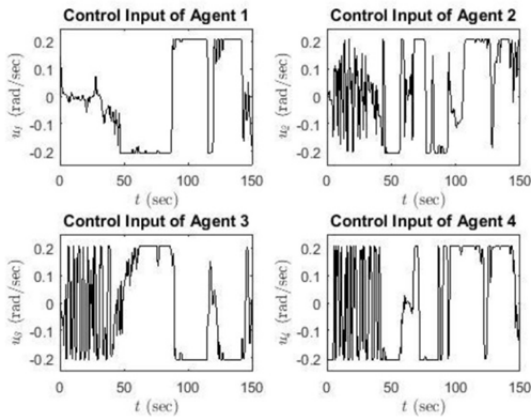


Fig 24 Distributed Control Using Interior-Point Algorithm

شکل 24 کنترل توزیع شده با استفاده از الگوریتم نقطه درونی

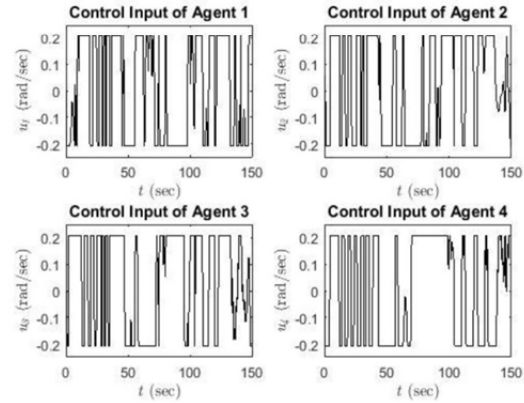


Fig 21 Distributed Control Using SQP Algorithm

شکل 21 کنترل توزیع شده با استفاده از الگوریتم برنامه‌ریزی درجه دوم تکراری

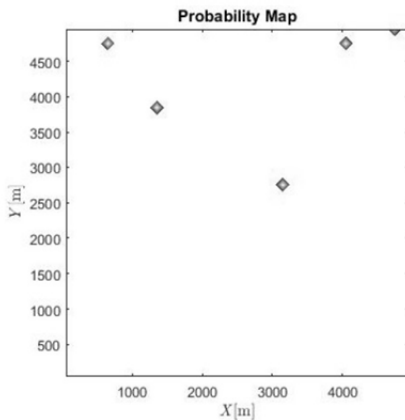


Fig 25 Found Targets Using Interior-Point Algorithm

شکل 25 اهداف پیدا شده توسط الگوریتم نقطه درونی

در کاهش عدم قطعیت میانگین قادر به تشخیص اهداف نزدیک مرز نبوده‌اند. در صورتی که الگوریتم گرادیان مزدوج توانسته سریع‌ترین پاسخ را در یافتن اهداف از خود نشان دهد. در الگوریتم برنامه‌ریزی درجه دوم تکراری نداشتن توانایی در تشخیص اهداف نزدیک مرز را می‌توان ناشی از تنظیمات قید دانست.

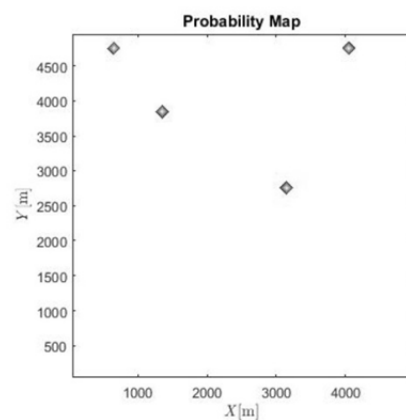


Fig 22 Found Targets Using SQP Algorithm

شکل 22 اهداف پیدا شده توسط الگوریتم برنامه‌ریزی درجه دوم تکراری

5-2-5- مقایسه نتایج شبیه‌سازی با داشتن دانش اولیه  
جدول 3 نتایج شبیه‌سازی‌های بخش‌های پیشین را خلاصه نشان داده است. با توجه به بررسی انجام شده الگوریتم نقطه درونی بهترین عملکرد را از خود نشان داده و علاوه بر یافتن تمامی اهداف در قید زمانی موجود از سرعت عمل قابل قبولی نیز برخوردار بوده است. روش‌های گرادیان مرتبه اول و برنامه‌ریزی درجه دوم تکراری با وجود عملکرد بهتر نسبت به گرادیان مزدوج

[5] Y. Yang, M. Polycarpou, A. Minai, Multi-UAV cooperative search using an opportunistic learning method, *Dynamic Systems, Measurement, and Control*, Vol. 129, No. 5, pp. 716-728, 2007.

[6] J. Riehl, G. Collins, J. Hespanha, Cooperative Search by UAV Teams: A model predictive approach using dynamic graphs, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 47, No. 4, pp. 2637-2656, 2011.

[7] S. Gan, S. Sukkarieh, Multi-UAV target search using explicit decentralized gradient-based negotiation, *IEEE International Conference on Robotics and Automation*, Shanghai, China, pp. 751-756, May 9-13, 2011.

[8] M. Mirzaei, F. Sharifi, B. Gordon, C. Rabbath, et al., Cooperative multi-vehicle search and coverage problem in uncertain environments, *50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, pp. 4140-4145, December 12-15, 2011.

[9] J. Hu, L. Xie, J. Xu, Z. Xu, Multi-agent cooperative target search, *Sensors*, Vol. 14, No. 6, pp. 9408-9428, 2014.

[10] Q. Wu, S. Zhou, S. Yan, W. Liu, A cooperative region surveillance strategy for multiple UAVs, *Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, Yantai, China, pp. 1744-1748, August 8-10, 2014.

[11] C. Gao, Z. Zhao, An optimal allocation approach of cooperative search capability based on game theory, *2nd International Conference on Systems and Informatics*, Shanghai, China, pp. 63-67, 2014.

[12] P. Lanillos, S. K. Gan, E. Besada-Portas, G. Pajares, et al., Multi-UAV target search using decentralized gradient-based negotiation with expected observation, *Information Sciences*, Vol. 282, pp. 92-110, 2014.

[13] F. Sharifi, M. Mirzaei, Y. Zhang, B. W. Gordon, Cooperative multi-vehicle search and coverage problem in an uncertain environment, *Unmanned Systems*, Vol. 3, No. 1, pp. 35-47, 2015.

[14] S. Perez-Carabaza, E. Besada-Portas, J. A. Lopez-Orozco, J. M. de la Cruz, A real world multi-UAV evolutionary planner for minimum time target detection, *Genetic and Evolutionary Computation Conference*, Denver, CO, USA, pp. 981-988, July 20-24, 2016.

[15] P. Yao, H. Wang, H. Ji, Gaussian mixture model and receding horizon control for multiple UAV search in complex environment, *Nonlinear Dynamics*, Vol. 88, No. 2, pp. 903-919, 2017.

[16] M. Zhang, J. Song, L. Huang, C. Zhang, Distributed cooperative search with collision avoidance for a team of unmanned aerial vehicles using gradient optimization, *Journal of Aerospace Engineering*, Vol. 30, No. 1, 2017.

[17] Y. Jin, Y. Liao, A. Minai, M. Polycarpou, Balancing search and target response in cooperative unmanned aerial vehicle (UAV) teams, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 36, No. 3, pp. 571-587, 2006.

[18] A. Amini, M. Sojoodi, S. Ozgoli, Decentralized dynamic output feedback controller design for consensus in multi-agent system of single link manipulators with flexible joint, *Modares Mechanical Engineering*, Vol. 14, No.15 pp. 75-84, 2014. (in Persian)

[19] P. T. Millet, D. W. Casbeer, T. Mercker, J. L. Bishop, Multiagent decentralized search of a probability map with communication constraints, *AIAA Guidance, Navigation, and Control Conference*, Toronto, ON, Canada, p. 8424, August 2-5, 2010.

[20] M. Zhong, C. Cassandras, Distributed coverage control and data collection with mobile sensor networks, *Proceedings of IEEE Conference on Decision Control*, Atlanta, Georgia USA, pp. 5604-5609, December 15-17, 2010.

[21] J. W. Hu, L. H. Xie, K. Y. Lum, J. Xu, Multiagent information fusion and cooperative control in target search, *IEEE Transactions on Control Systems Technology*, Vol. 21, No. 4, pp. 1223-1235, 2013.

[22] R. Olfati-Saber, Distributed Kalman filter with embedded consensus filters, *Proceedings of 44th IEEE Conference on Decision Control*, Seville, Spain, pp. 8179-8184, December 12-15, 2005.

[23] W. Ren, R. Beard, E. Atkins, Information consensus in multivehicle cooperative control, *IEEE Control Systems Magazine*, Vol. 27, No. 2, pp. 71-82, 2007.

[24] L. Xiao, S. Boyd, S. Lall, A scheme for robust distributed sensor fusion based on average consensus, *Proceedings of 4th International Symposium on Information Processing in Sensor Networks*, Los Angeles, California, pp. 63-70, April 24-27, 2005.

[25] J. Cortes, S. Martinez, T. Karatas, F. Bullo, Coverage control for mobile sensing networks, *IEEE Transactions on Robotics and Automation*, Vol. 20, No. 2, pp. 243-255, 2004.

[26] K. R. Gurusparad, D. Ghose, Performance of a class of multi-robot deploy and search strategies based on Centroidal Voronoi configurations, *Systems Science*, Vol. 44, No. 4, pp. 680-699, 2013.

[27] H. Y. Zhang, Q. Wang, *Optimal Control Theory and Application*, Higher Education Press, Beijing, 1st edition, pp. 108-127, 2006.

[28] M. T. Hagan, H. B. Demuth, M. H. Beale, O. de Jesus, *Neural Network Design*, Martin Hagan, Oklahoma, 2nd edition, pp. 281-287, 2014.

[29] K. Schittkowski, NLPQL: A FORTRAN subroutine solving constrained nonlinear programming problems, *Annals of Operations Research*, Vol. 5, No. 2, pp. 485-500, 1986.

[30] R. H. Byrd, M. E. Hribar, J. Nocedal, An interior point algorithm for large-scale nonlinear programming, *SIAM Journal on Optimization*, Vol. 9, No. 4, pp. 877-900, 1999.

جدول 3 مقایسه نتایج شبیه‌سازی با داشتن دانش اولیه

Table 3 Simulation results with priori-knowledge comparison

الگوریتم	زمان یافتن اهداف (ثانیه)	کاهش عدم قطعیت میانگین (%)
گرادین مرتبه اول	× - 99.5 - 72 - 39.5 - 46	83.27
گرادین مزدوج	128 - 116 - 37.5 - 60 - 46	82.86
نقطه درونی	134.5 - 67 - 69.5 - 38.5 - 46	87.41
برنامه‌ریزی درجه دوم تکراری	× - 81 - 141 - 39 - 46.5	83.79

### 6- جمع‌بندی و نتیجه‌گیری

در این مقاله به مسأله جستجوی مشارکتی چندعامله پرداخته شد. از یک مدل سینماتیکی ساده برای شبیه‌سازی حرکت عامل‌ها استفاده گردید. حرکت عامل‌ها در صفحه موازی زمین و با ارتفاع و سرعت ثابت در نظر گرفته شد. نرخ چرخش عامل به عنوان ورودی کنترل تعیین گردید. توپولوژی ارتباطات میان عامل‌ها به یک گراف مدلی شد که خروجی آن مجموعه همسایه‌های هر عامل بود. برای حسگر عامل‌ها از یک مدل ساده بهره گرفته شد که مشاهدات عامل‌ها را با درصدی خطا ثبت می‌نمود. با توجه به مشاهدات نقشه احتمال هر عامل با بهره‌گیری از یک مدل بیزین به‌روزرسانی گردید. سپس تلفیق نقشه احتمال با تبادل اطلاعات میان عامل‌ها انجام گرفته و تمامی نقشه‌های احتمال به یک نقشه احتمال مشترک همگرا شدند. قطعیت نداشتن هر سلول با استفاده از این نقشه احتمال محاسبه شد. مسأله طراحی مسیر به یک مسأله بهینه‌سازی کاهش قطعیت نداشتن میانگین تعریف شد. این مسأله بهینه‌سازی با استفاده از الگوریتم‌های گرادین مرتبه اول و گرادین مزدوج به صورت تحلیلی حل گردید. مقادیر مناسب نرخ یادگیری الگوریتم‌های بهینه‌سازی با انجام شبیه‌سازی‌های متعدد و آزمون و خطا تعیین شد، همچنین حل عددی مسأله بهینه‌سازی با بهره‌گیری از الگوریتم‌های بهینه‌سازی برنامه‌ریزی درجه دوم تکراری و نقطه درونی به کمک نرم‌افزار متلب انجام گرفت.

مطابق با نتایج به دست آمده الگوریتم نقطه درونی بدون دانش اولیه بهترین عملکرد را در کاهش عدم قطعیت میانگین فضای جستجو داشته است. این الگوریتم با داشتن دانش اولیه عملکرد بهتری نسبت به دیگر الگوریتم‌های بهینه‌سازی از خود نشان داده است. این در حالی است که الگوریتم‌های نقطه درونی و برنامه‌ریزی درجه دوم تکراری با توجه به ذات بهینه‌سازی عددی خود نیازمند محاسبات به مراتب بیشتری هستند. در نتیجه حجم محاسبات الگوریتم طراحی مسیر بهینه در آن‌ها بیشتر خواهد بود؛ بنابراین در کاربردهای پیاده‌سازی سخت‌افزاری باید مصالحه‌ای میان توان محاسباتی در دسترس (به صرفه) و عملکرد مطلوب (قابل قبول) الگوریتم صورت پذیرد.

### 7- مراجع

[1] M. Polycarpou, Y. Yang, K. Passino., A cooperative search framework for distributed agents, *Proceedings of the 2001 IEEE International Symposium on Intelligent Control*, Mexico City, Mexico, pp. 1-6, September 5-7, 2001.

[2] M. Baum, K. Passino, A Search-Theoretic Approach to cooperative control for uninhabited air vehicles, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, California, p. 4589, August 5-8, 2002.

[3] Y. Yang, A. Minai, M. Polycarpou, Decentralized cooperative search in UAV's using opportunistic learning, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, California, pp. 4590, August 5-8, 2002.

[4] K. R. Guruprasad, D. Ghose, Multi-agent search using Voronoi partitions, *Proceedings of the International Conference on Advances in Control and Optimization of Dynamical Systems*, Bangalore, India, February 1-2, 2007.