

# عملکرد الگوریتم فراابتکاری ALO در بهینه‌سازی طراحی

## پروانه‌های دریایی

غلامرضا صالحی<sup>۱</sup>، حسن قاسمی<sup>۲</sup>، هادی صالحی<sup>۳</sup>

۱- کارشناسی ارشد مهندسی ساخت در صنایع دریایی، دانشگاه صنعتی امیرکبیر، gholamreza-salehi@aut.ac.ir

۲- استاد دانشگاه صنعتی امیرکبیر، دانشکده مهندسی دریا، gasemi@aut.ac.ir

۳- دانشجوی دکتری مهندسی نرم‌افزار، دانشگاه آزاد ساری، hadisalehi.un@gmail.com

(تاریخ دریافت: ۹۶/۴/۳۰؛ تاریخ پذیرش: ۹۶/۵/۲۴)

### چکیده

با توجه به اینکه روند طراحی پروانه یک سلسله مراحل آزمون و خطای همراه با تکرار است و بستگی به شرایط استفاده و محدوده‌های مجاز هر یک از فاکتورهای مؤثر در طراحی دارد، بهینه‌سازی این روند در راستای افزایش راندمان کل بسیار حائز اهمیت می‌گردد. جهت کاهش در زمان بهینه‌سازی الگوریتم‌های زیادی در بهینه‌سازی معرفی شده‌اند که معروف‌ترین آنها الگوریتم ژنتیک است. البته، مدت زمان زیادی از معرفی آن به دنیا می‌گذرد که در طول این زمان مرتباً این الگوریتم‌ها به‌روزتر و قدرتمندتر گردیدند. در این مقاله، بعد از بیان تئوری‌های موجود و الگوریتم‌های معرفی شده به بیان چگونگی عملکرد نمونه‌ای از الگوریتم‌های بهینه‌سازی، در بهینه‌سازی طراحی پروانه پرداخته شده است. یکی از الگوریتم‌های قدرتمند بهینه‌سازی، الگوریتم فراابتکاری ALO می‌باشد. این مطالعه، بعد از معرفی الگوریتم فراابتکاری ALO و چگونگی کارکرد آن، میزان توانایی آن را در غالب یک مثال ساده بیان نموده است. این الگوریتم با استفاده از ویژگی خاص خود و همگرایی سریع قابلیت دستیابی به نتیجه مطمئن‌تر را داراست. نکته‌ی مهمی که در استفاده از این الگوریتم باید مدنظر باشد این است که هر چه ورودی‌های آن دقیق‌تر و صحیح‌تر باشد، نتایج نیز بهتر خواهد بود. لذا، می‌توان نتیجه گرفت که با داشتن داده‌های تقریباً بهینه می‌توان به بهینه‌ترین شرایط دست یافت. آنچه به دست آمد ضمن تأیید توانمندی این الگوریتم در بهینه‌سازی مبین سایر قابلیت‌های این الگوریتم در زمینه‌ی بهینه‌سازی پروانه‌های دریایی با توجه به نمودارهای ارائه‌شده مربوط به سرعت همگرایی آن است.

### واژه‌های کلیدی

بهینه‌سازی، الگوریتم فراابتکاری، طراحی پروانه، راندمان.

## ALO Optimization algorithm performance for the marine propeller design

Gholamreza salehi<sup>1</sup>, Hassan ghassemi<sup>2</sup>, Hadi Salehi<sup>3</sup>

<sup>1</sup>M. Sc in marine engineering Amirkabir University of Technology, Gholamreza-salehi@aut.ac.ir

<sup>2</sup>Professor of Marine Hydrodynamics, Amirkabir University of Technology, Department of Maritime Engineering

<sup>3</sup> Ph.D Candidate, Software Engineering, Azad University of Sari, hadisalehi.un@gmail.com

(Submitted: 2017/July/21; Accepted: 2017/Aug/15)

### Abstract

The marine propeller design is a series of repetitive trial and error in calculations. It depends on the terms and limits of the effective factors in the design. Optimization of the process in order to increase the total efficiency is very important. Many optimization algorithms have been suggested so as to reduce the time of marine propeller design that the most famous of them is genetic algorithm. This algorithm is developed day by day. In this paper after briefly introduce existing theories and algorithm explained the method of ALO algorithm and expressed how it work in a simple example. What was achieved while confirming the ability of this algorithm in Optimization and represents other feature of this algorithm in marine propeller design according to its feature related to the convergence diagram. The method of this algorithm inspired by the insect called an ant-lion. So those by trapping the best items and compare them choose the optimal situation. In the other words this algorithm work by preliminary data feed into software and if the preliminary data is more accurate the results will be better. Therefore we could earn more accurate preliminary data by using the powerful marine design software and input this information as a preliminary data for the ALO algorithm.

### Keywords

Optimization algorithm, Optimization, Propeller design, Efficiency, Marine propeller.

پروانه‌ی کشتی یکی از اجزای مهم و اصلی در سیستم رانش و ایجاد نیروی رانش، و در نتیجه سرعت پیشروی مطلوب می‌باشد. علیرغم اینکه بیش از یک قرن از شروع استفاده از پروانه‌ها در سیستم رانش می‌گذرد، کماکان تحقیق و پژوهش و بررسی عملکرد و طراحی پروانه یک موضوع مهم و اساسی به شمار می‌رود. دلایل اهمیت این موضوع را می‌توان افزایش مداوم اندازه‌ی کشتی‌ها، محدودیت منابع انرژی، اثرات زیست‌محیطی و غیره دانست [۱].

در گذشته صرفاً با توجه به مشاهدات و اندازه‌گیری‌های تجربی انتخاب یا طراحی پروانه‌هایی با راندمان بهتر امکان‌پذیر بوده است. از آنجا که این روش‌ها قادر به تشریح مسائلی همچون کاویتاسیون و ارتعاشات نیستند، روش‌های تحلیلی [۲] که شاید بتوان از تئوری رانکین که در سال ۱۸۶۵ ارائه شد و به‌عنوان پایه و اساس سایر تئوری‌هاست، مورد استفاده قرار گرفتند. همچنین، تئوری‌های مدرنی که دارای چشمه و چاه دوقطبی هستند، ارائه گردیدند. در ادامه، تئوری المان پره فرود ارائه می‌گردد. در سال‌های اخیر کتاب‌هایی نیز در زمینه‌ی تئوری‌های طراحی پروانه نگاشته شده است که می‌توان به کتاب‌های برسلین و اندرسوندر این خصوص اشاره کرد [۳]. از آنجا که انجام محاسبات دوبعدی می‌توانست خطاهایی را در نتایج نهایی ایجاد کند، تئوری‌های سه‌بعدی سطوح برآ و المان مرزی جهت رفع این خطا، مورد استفاده قرار گرفت.

به این ترتیب و با گسترش تئوری‌ها، فاکتورهایی در طراحی پروانه معرفی گردیدند که طراحی بر اساس آنها صورت می‌گرفت و به همین نحو بهینه‌سازی‌ها نیز با تمرکز روی تأثیرات این فاکتورها امکان‌پذیر می‌گردید.

معیارهای مهمی که در طراحی پروانه مؤثرند، عبارتند از بیشترین تولید نیروی محوری، کمترین گشتاور، راندمان بهینه و عدم وقوع یا کمترین احتمال وقوع کاویتاسیون [۴].

با توجه به اینکه عواملی چون هزینه، راحتی ساخت و ... نیز به عوامل بالا اضافه می‌گردد، باید با داشتن پیش‌فرض‌هایی به محاسبه و به دست آوردن سایر فاکتورها در محدوده‌ی مطلوب اقدام کرد. در این روند با توجه به اینکه افزایش راندمان در بعضی از عوامل با هم

در تضاد است، باید مقادیر بهینه را محاسبه کرد؛ مثلاً گام زیاد باعث افزایش راندمان می‌شود که مطلوب است، لکن عامل افزایش احتمال کاویتاسیون نیز هست. به این ترتیب چنانچه هر کدام از عوامل موردنظر در محدوده مطلوب نباشد، باید با تغییراتی در پیش‌فرض‌های اولیه، مجدداً محاسبات را تا رسیدن به نتیجه بهینه ادامه داد. جهت نیل به این امر از الگوریتم‌ها و فلوچارت‌هایی استفاده می‌کنند، تا روند کار، سیستماتیک گردد [۵].

مختاری و همکارش در این زمینه به ارائه الگوریتمی بهینه جهت طراحی پروانه شناورها با استفاده از سری واگنینگن پرداختند. از آنجا که پیشرفت فناوری‌های رایانه‌ای و گسترش بکارگیری رایانه در علوم، شامل طراحی پروانه نیز گردید، در این زمینه نیز الگوریتم‌هایی جهت بهینه‌سازی مورد استفاده قرار گرفت. یکی از این الگوریتم‌ها که توانایی بالایی در مسائل پیچیده بهینه‌سازی دارد، الگوریتم ژنتیک است [۶]. این الگوریتم زمان کوتاهی در طراحی سیستم‌های پیش‌برنده صرف می‌کند [۷]. همچنین، سوئن و کو با موفقیت روش الگوریتم ژنتیک را برای طراحی پروانه‌های سری B دریایی مورد استفاده قرار دادند [۸]. رحمان‌نژاد و همکارانش نیز با در نظر گرفتن تصحیح عدد واگنینگن و با لحاظ کردن قیود طراحی و کاویتاسیون، بهینه‌سازی پروانه را انجام دادند و کار سوئن و کو را کامل‌تر کردند [۹]. سجاد اردشیری و همکارانش نیز با ملاحظات ارتعاشی به‌وسیله‌ی الگوریتم ژنتیک طراحی پروانه را بهینه نمودند [۱۰]. میرجلیلی با تکیه بر پارامترهایی نظیر طول کورد و ضخامت ارائه‌شده توسط الگوریتم فراابتکاری ALO بهینه‌سازی پروانه را انجام داد. ALO یکی از جدیدترین الگوریتم‌های ارائه‌شده است که در طراحی پروانه کشتی کاربرد دارد [۱۱].

در این مقاله، در نظر داریم با طیف وسیع‌تری از عوامل مؤثر در طراحی پروانه به بهینه‌سازی طراحی پروانه شناورها بپردازیم.

## ۲- عملکرد الگوریتم ALO<sup>۱</sup>

الگوریتم ALO از عکس‌العمل بین شیرمور و مورچه در تله استفاده می‌کند. برای مدل‌سازی تعاملات، ابتدا

<sup>۱</sup>Ant Lion Optimizer

مورچه را به عنوان حرکت در فضای جستجو در نظر می‌گیریم. سپس، شیرمور مجاز به شکار آن می‌شود. از آنجایی که مورچه به صورت تصادفی به دنبال غذا در طبیعت می‌گردد، یک پیاده‌سازی ساده برای مدل کردن حرکت مورچه به شکل زیر ایجاد شده است.

$$X(t) = [0, (c \text{ msum}(2r(t_1) - 1), \text{cumsum}(2r(t_2) - 1), \text{cumsum}(2r(t_n) - 1))] \quad (۱)$$

به طوری که  $\text{cumsum}$  مجموع تجمعی را محاسبه می‌کند.  $n$  نشان‌دهنده‌ی بیشترین تکرار،  $t$  نشان‌دهنده‌ی مرحله‌ی راه رفتن تصادفی است ( $t$ امین تکرار در حال حاضر). همچنین،  $r(t)$  تابع تصادفی است که به شکل زیر تعریف شده است.

$$r(t) = \begin{cases} 1 & \text{if rand} > 0.5 \\ 0 & \text{if rand} \leq 0.5 \end{cases} \quad (۲)$$

به طوری که  $t$  نشان‌دهنده‌ی مرحله قدم‌زدن تصادفی و  $\text{rand}$  ایجاد توزیع یکنواخت عدد تصادفی در بازه‌ی  $[0, 1]$  دارد. موقعیت مورچه‌ها در ماتریس زیر ذخیره و در طول بهینه‌سازی مورد استفاده قرار می‌گیرد.

$$M_{Ant} = \begin{bmatrix} A_{1,1} & A_{1,2} & \dots & \dots & A_{1,d} \\ A_{2,1} & A_{2,2} & \dots & \dots & A_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n,1} & A_{n,2} & \dots & \dots & A_{n,d} \end{bmatrix} \quad (۳)$$

به طوری که  $M_{Ant}$  موقعیت هر مورچه را مشخص می‌کند و  $A_{ij}$  مشخص‌کننده‌ی [آمین متغیر] آمین مورچه است.  $n$  مشخص‌کننده‌ی تعداد مورچه‌ها و  $d$  مشخص‌کننده‌ی تعداد متغیرها است. باید توجه شود که مورچه‌ها همچون ذرات در الگوریتم PSO و یا افراد در الگوریتم GA هستند. موقعیت مورچه اشاره پارامترها برای یک راه‌حل خاص است. ماتریس  $M_{Ant}$  به منظور ذخیره موقعیت تمام مورچه‌ها (متغیرهای تمام راه‌حل) در بهینه‌سازی در نظر گرفته شده است. برای ارزیابی هر مورچه، یک تابع برازندگی در طول بهینه‌سازی مورد استفاده قرار می‌گیرد. سپس، این مقادیر به شکل زیر ذخیره می‌شوند.

$$M_{OA} = \begin{bmatrix} f([A_{1,1}, A_{1,2}, \dots, A_{1,d}]) \\ f([A_{2,1}, A_{2,2}, \dots, A_{2,d}]) \\ \vdots \\ f([A_{n,1}, A_{n,2}, \dots, A_{n,d}]) \end{bmatrix} \quad (۴)$$

$M_{OA}$  به منظور ذخیره‌سازی مقدار تابع برازندگی هر مورچه استفاده می‌شود. علاوه بر مورچه‌ها، فرض می‌کنیم شیرمور در جایی از فضای جستجو پنهان شده است. به منظور ذخیره این موقعیت و تابع هدف آن ماتریس‌های زیر مورد استفاده قرار می‌گیرد:

$$M_{Antion} = \begin{bmatrix} AL_{1,1} & AL_{1,2} & \dots & \dots & AL_{1,d} \\ AL_{2,1} & AL_{2,2} & \dots & \dots & AL_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ AL_{n,1} & AL_{n,2} & \dots & \dots & AL_{n,d} \end{bmatrix} \quad (۵)$$

$$M_{OAL} = \begin{bmatrix} f([AL_{1,1}, AL_{1,2}, \dots, AL_{1,d}]) \\ f([AL_{2,1}, AL_{2,2}, \dots, AL_{2,d}]) \\ \vdots \\ f([AL_{n,1}, AL_{n,2}, \dots, AL_{n,d}]) \end{bmatrix} \quad (۶)$$

$M_{OAL}$  و  $M_{Antion}$  به ترتیب ماتریس موقعیت و ماتریس تابع هدف هر شیرمور را مشخص می‌کند. همچنین،  $AL_{i,j}$  مقدار بعد  $j$ ام شیرمور، شیرمور  $i$ ام،  $n$  تعداد شیرمورها و  $d$  تعداد متغیرها را نشان می‌دهد.

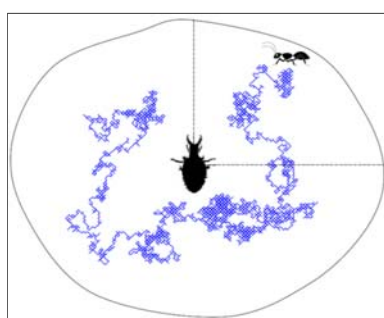
در طول بهینه‌سازی، شرایط زیر اعمال می‌گردد:

- مورچه‌ها در سراسر فضای جستجو به صورت تصادفی حرکت می‌کند.
- پیاده‌روی تصادفی به تمام ابعاد از مورچه‌ها اعمال می‌شود.
- قدم‌زدن‌های تصادفی تحت تأثیر تله‌های شیرمورها قرار می‌گیرند.
- شیرمورها می‌توانند متناسب با تابع هدف خود، گودال‌های بزرگ‌تر حفر کنند (تابع هدف بزرگ‌تر، گودال بزرگ‌تر).
- شیرمور با گودال بزرگ‌تر قابلیت گرفتن مورچه‌های بیشتری نیز دارد.
- هر مورچه می‌تواند توسط یک شیرمور شکار شود.

دارد. در این معادله  $Antlion_j^t$  نشان‌دهنده‌ی موقعیت ژامین شیرمور در تکرار  $t$ ام است [۱۱].

### ۵- ساخت تله

به منظور مدل کردن توانایی شکار مورچه‌ها توسط شیرمور، از ساختار چرخ گردان استفاده می‌شود. شکل (۱) نشان‌دهنده‌ی در تله قرار گرفتن مورچه در دام یک شیرمور است. الگوریتم ALO نیاز به یک اپراتور چرخ گردان برای تعیین شیرمورها بر پایه‌ی تابع برازندگی آنها در طول بهینه‌سازی دارد [۱۱].



شکل (۱): حرکت تصادفی یک مورچه در تله شیرمور.

### ۶- لغزش مورچه به سوی شیرمور

زمانی که مورچه‌ای در دام گرفتار می‌شود، شیرمور به سوی لبه‌های دام سنگ پرتاب می‌کند. معادله زیر این رفتار را نشان می‌دهد.

$$c^t = \frac{c^t}{I} \quad (9)$$

$$d^t = \frac{d^t}{I} \quad (10)$$

که در این معادله  $I$  یک نسبت ثابت و  $C^t$  مینیمم تمامی متغیرها در  $t$ امین تکرار، همچنین،  $d^t$  شامل برداری با ماکسیمم مقدار تمامی متغیرها در  $t$ امین تکرار است و  $I$  را به صورت زیر تعریف می‌کنیم [۳۲].

$$I = 10^w \frac{t}{T} \quad (11)$$

که  $t$  تکرار در حال انجام،  $T$  ماکسیمم تعداد تکرارها و  $w$  ثابتی است که بر پایه تکرار در حال انجام به صورت زیر تعریف می‌شود:

- محدوده راه رفتن مورچه به صورت انطباقی رو به دام‌های شیرمور کاهش می‌یابد.
- اگر مورچه‌ای توسط یک شیرمور شکار شود، به معنی گرفتن و زیر ماسه بردن مورچه توسط شیرمور است.
- بعد از هر بار شکار، شیرمور برای گرفتن مورچه بعدی، موقعیت خود را تغییر می‌دهد و یک گودال با تغییرات مناسب ایجاد می‌کند [۱۱].

### ۳- حرکت تصادفی مورچه‌ها

حرکت تصادفی مورچه‌ها بر پایه‌ی معادله (۱) صورت می‌گیرد. در هر مرحله از بهینه‌سازی، مورچه‌ها با قدم زدن تصادفی، موقعیت خود را به‌روز می‌کنند. از آنجایی که هر فضای جستجو دارای محدودیتی است (دامنه متغیرها)، معادله (۱) نمی‌تواند به طور مستقیم برای به‌روزرسانی موقعیت مورچه مورد استفاده قرار گیرد. به منظور قدم‌زدن تصادفی در فضای جستجو، آنها توسط معادله (۷) نرمال می‌شوند.

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i - c_i^t)}{(d_i^t - a_i)} + c_i \quad (7)$$

که  $a_i$  برابر مینیمم حرکت تصادفی متغیر در  $t$ امین متغیر است. همچنین،  $b_i$  ماکزیمم حرکت تصادفی متغیر  $t$ ام است. همچنین،  $C_i^t$  مینیمم  $t$ امین متغیر در  $t$ امین تکرار و  $D_i^t$  ماکزیمم  $t$ امین متغیر در  $t$ امین تکرار است. معادله (۷) باید در هر تکرار انجام شود تا تضمین‌کننده رخدادن حرکت تصادفی در فضای جستجو باشد [۱۱].

### ۴- تله گودال شیرمور

با توجه به بحث بالا، قدم زدن مورچه، تحت تأثیر تله شیرمورچه قرار می‌گیرد. به منظور اعمال مدل ریاضی این فرض، معادله (۸) پیشنهاد می‌شود.

$$c_i^t = Antlion_j^t + c^t \quad (8)$$

$$d_i^t = Antlion_j^t + d^t$$

به طوری که  $C^t$  مینیمم مقدار تمامی متغیرها در تکرار  $t$ ام است و  $d^t$  دلالت بر برداری شامل ماکسیمم تمامی متغیرها در تکرار  $t$ ام دارد. همچنین،  $C_i^t$  مینیمم مقدار تمامی متغیرها در  $t$ امین مورچه و  $d_i^t$  دلالت بر برداری شامل ماکسیمم تمامی متغیرها برای  $t$ امین مورچه

است. همچنین، نشان‌دهنده‌ی موقعیت آامین مورچه در آامین تکرار است [۱۱].

### ۹- الگوریتم ALO

الگوریتم ALO به عنوان یک تابع سه‌تایی است که برای تخمین بهینه کلی برای مسائل بهینه‌سازی، به شرح زیر تعریف می‌شود:

ALO (A, B, C)

که در آن A یک تابع است که راه‌حل‌های تصادفی اولیه را تولید می‌کند، B دست‌کاری جمعیت اولیه‌ای است که توسط تابع A ارائه شده و C زمانی که معیار پایانی راضی‌کننده است، مقدار درست را بر می‌گرداند.

توابع A, B, C به صورت زیر تعریف می‌شوند:

$$\emptyset \xrightarrow{A} \{M_{Ant}, M_{OA}, M_{Antlion}, M_{OAL}\} \quad (14)$$

$$\{M_{Ant}, M_{Antlion}\} \xrightarrow{B} \{M_{Ant}, M_{Antlion}\} \quad (15)$$

$$\{M_{Ant}, M_{Antlion}\} \xrightarrow{C} \{true, false\} \quad (16)$$

که در آن  $M_{Ant}$  ماتریس موقعیت مورچه‌ها،  $M_{Antlion}$  شامل موقعیت شیرمورها،  $M_{OA}$  مورچه‌های شکارشده و  $M_{OAL}$  شیرمورچه‌های شکارکننده.

در الگوریتم ALO مورچه و شیرمور ابتدا توسط تابع A مقداردهی اولیه می‌شوند. در هر تکرار، موقعیت هر مورچه با توجه به یک شیرمور که توسط اپراتور چرخ گردان و نخبگان انتخاب شده است، توسط تابع B به‌روزرسانی می‌شود. محدوده‌ی موقعیت به‌روزرسانی‌شده در ابتدا متناسب با شماره‌ی تکرار جاری تعیین می‌شود. سپس موقعیت به‌روزرسانی‌شده به‌وسیله‌ی قدم‌زدن تصادفی در اطراف شیرمورچه و نخبه کامل می‌شود. زمانی که همه مورچه‌ها به‌طور تصادفی حرکت کردند، آنها توسط تابع ارزش ارزیابی می‌شوند. اگر هر مورچه نسبت به شیرمورها دارای موقعیت مناسب باشد، موقعیت آن به‌عنوان موقعیت جدید برای شیرمور در تکرار بعدی مورد بررسی قرار می‌گیرد. بهترین شیرمور با بهترین شیرمور یافت‌شده در طول بهینه‌سازی (نخبه) مقایسه می‌شود و در صورت لزوم جایگزین آن می‌شود. این عملیات تا زمانی که تابع C مقدار نادرست را برگرداند، ادامه می‌یابد [۱۱].

$$w = \begin{cases} 2 & \text{when } t > 0.1T \\ 3 & \text{when } t > 0.5T \\ 4 & \text{when } t > 0.75T \\ 5 & \text{when } t > 0.9T \\ 6 & \text{when } t > 0.95T \end{cases} \quad (12)$$

### ۷- شکار طعمه و بازسازی تله

مرحله‌ی آخر شکار هنگامی است که طعمه به پایین دام می‌رسد و در دهان شیرمور قرار می‌گیرد. پس از این مرحله، شیرمور طعمه را به داخل شن می‌کشد و او را می‌خورد. در استفاده از این فرآیند، فرض می‌شود که شکار هنگامی رخ می‌دهد که مورچه داخل شن فرو رفته باشد. سپس، می‌بایست موقعیت شیرمور به موقعیتی که مورچه را شکار کرده، جهت افزایش شانس شکار جدید، به‌روزرسانی شود. رابطه (۱۱) این عمل را انجام می‌دهد.

$$Antlion_j^t = Ant_i^t \quad \text{if } f(Ant_i^t) > f(Antlion_j^t) \quad (11)$$

که t نشان‌دهنده‌ی تکرار در حال اجراست،  $Antlion_j^t$  موقعیت شیرمور لام در تکرار tام و  $Ant_i^t$  نشان‌دهنده‌ی موقعیت مورچه لام در تکرار tام می‌باشد [۱۱].

### ۸- نخبه‌گرایی

نخبه‌گرایی یک ویژگی مهم از الگوریتم‌های تکاملی است که اجازه می‌دهد تا بهترین راه‌حل به دست‌آمده در هر مرحله از فرآیند بهینه‌سازی حفظ شود. در این مطالعه، بهترین شیرمورچه به دست‌آمده در هر تکرار، ذخیره شده و به عنوان یک نخبه در نظر گرفته می‌شود. از آنجایی که نخبه‌ها مناسب‌ترین پاسخ‌های ما هستند، باید قادر به تأثیرگذاری بر همه‌ی مورچه‌ها باشند؛ بنابراین، فرض می‌شود هر مورچه با ساختار چرخ گردان به یک شیرمور نزدیک می‌شود. معادله (۱۳) شبیه‌سازی نخبه را نشان می‌دهد.

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \quad (13)$$

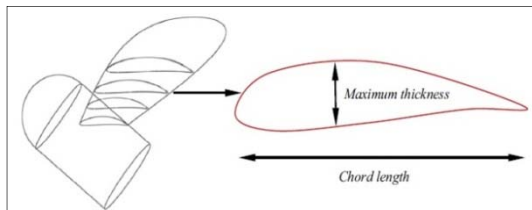
به طوری که  $R_A^t$  حرکت تصادفی در اطراف شیر مورچه‌ها توسط چرخ گردان در تکرار tام است. همچنین،  $R_E^t$  حرکت تصادفی در اطراف حالت نخبه در آامین تکرار

۱۰- پروانه در طراحی ALO؛ نمونه‌ای ساده از بکارگیری الگوریتم فراابتکاری

پروانه‌ها نیروی تراست محوری لازم برای حرکت کشتی‌ها را تأمین می‌کنند. با توجه به چگالی بالای آب، راندمان پروانه از اهمیت ویژه‌ای برخوردار است. در این مرحله به بیان یک نمونه‌ی ساده از کاربری این الگوریتم در طراحی پروانه می‌پردازیم.

### ۱۱- روش‌های قابل استفاده

پارامترهای ساختاری پروانه بر اساس شکل ظاهری شکل گرفته‌اند. یک روش متداول برای نمایش و مدل‌سازی شکل تیغه استفاده از سری Bezier می‌باشد. این روش با بهره‌گیری از مجموعه‌ای از پارامترهای کنترلی، انحنای پره را تعریف می‌کند. روش ساده‌ی دیگری نیز وجود دارد که با تعریف انحنای استاندارد ایرفویل‌ها در مقاطع پره پروانه پرداخته که مشخصات و تنظیمات خاص خود را دارد. در این مثال، به دلیل سادگی کار روش دوم انتخاب می‌شود. شکل (۳) وضعیت مقطع پره در این روش را نشان می‌دهد که بر اساس مقاطع ۲۰ عدد می‌باشد.



شکل (۳): ایرفویل در مقاطع مختلف پره.

بدین منظور ده ایرفویل در طول پره بررسی شده‌اند که پارامترهای بهینه‌شده آن با الگوریتم ALO این پارامترها به قرار زیر می‌باشد:

طول کورد و ماکسیمم ضخامت در اولین ایرفویل:  $X_2 - X_1$

طول کورد و ماکسیمم ضخامت در دومین ایرفویل:  $X_4 - X_3$

طول کورد و ماکسیمم ضخامت در آخرین ایرفویل:  $X_{20} - X_{19}$

بهترین پارامترهای طراحی برای پروانه‌های چهار و پنج‌پره‌ای، مطابق جدول زیر می‌باشد.

جدول (۱): بهترین پارامترهای طراحی برای پروانه

چهارپره‌ای

ضخامت	کورد	ردیف
۰/۱۴۲۱	۰/۱۳۲۱	۱
۰/۱۸۵۲	۰/۱۶۸۰	۲

۱۰-۱- شرایط اولیه مسئله

داده‌های اولیه مسئله یا همان پیش‌فرض‌ها جهت طراحی به شرح زیر است:

تعداد تیغه = چهار یا پنج  
 قطر اصلی = دو متر  
 نوع پروانه = ۵ پره گام ثابت  
 هدف = دستیابی به بهترین بهره‌وری  
 سرعت چرخش پروانه = ۱۷۰ دور در دقیقه  
 سرعت پیشروی کشتی = ۵ متر بر ثانیه  
 نیروی تراست = ۴۰۰۰۰ نیوتن  
 چگالی آب = ۹۹۹/۹۷ کیلوگرم بر متر مکعب  
 قدرت = ۲۸۸۰۹۹/۰۱۱۵ وات  
 گشتاور = ۱۶۱۸۳/۱۹۳۶ نیوتن متر  
 راندمان پروانه نیز در فرمول زیر ارائه گردیده است.

$$\eta = \frac{V_a}{2\pi nD} \times \frac{K_T(x)}{K_Q(x)}$$

راندمان پروانه:

$V$ ،  $D$ ،  $n$ ،  $K_T$  و  $K_Q$  به ترتیب از راست به چپ:

سرعت محوری، قطر پروانه، تعداد دور پروانه، ضریب تراست و ضریب گشتاور می‌باشند. شکل (۲) مقیاسی از ابعاد پروانه یادشده است.



شکل (۲): مقیاسی از ابعاد پروانه.

ضریب تراست با فرمول زیر محاسبه می‌گردد:

$$K_T(x) = \frac{T}{\rho n^2 D^2}$$

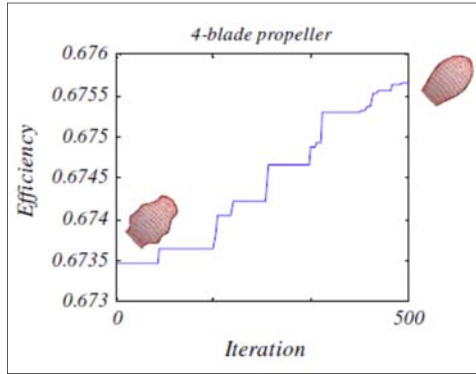
$$\eta(x) = \text{بیشینه}$$

بیشینه‌سازی: راندمان پروانه

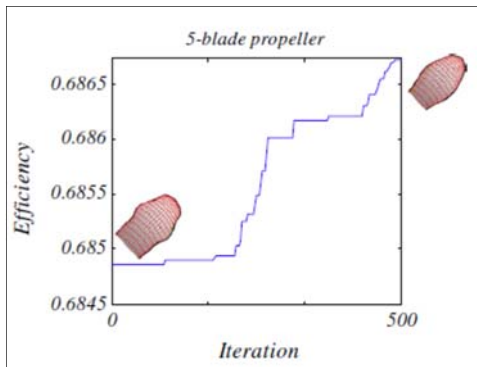
رابطه موضوع: اصطکاک ویک و کاهش نیروی محوری

محدوده پارامترها:  $0 < X_1 - X_{10} \leq 1$

در شکل (۴) تکرارها در الگوریتم تا رسیدن به جواب بهینه به صورت شماتیک نشان داده شده است.



۴- الف



۴- ب

شکل (۴): الف- همگرایی در الگوریتم ALO در پروانه‌ی چهارپره‌ای، ب- همگرایی در الگوریتم ALO در پروانه‌ی پنج‌پره‌ای.

همچنین، در شکل (۵) به خوبی تفاوت بین طراحی اولیه‌ی تصادفی و طراحی بهینه‌ی نهایی را نشان می‌دهد. لازم به توضیح است که در طراحی و مثال یادشده به دلیل تعداد زیاد پارامترها و ایجاد پیچیدگی‌های ریاضی در فرمولاسیون، کلیه پارامترها لیست نشده و همه‌ی محدودیت‌ها ارائه نگردیده‌اند؛ زیرا حجم گسترده‌ای از این پیچیدگی‌ها می‌تواند فرمول‌ها را بحث‌برانگیز و چالشی نموده و عامل اتلاف وقت زیادی گردد؛ زیرا هر تابع زمانی در حدود پنج دقیقه برای هر طراحی طول می‌کشد. البته، می‌توان با این کار نرم‌افزار کامل‌تری ارائه کرد [۱۲].

۰/۲۲۵۲	۰/۲۱۷۷	۳
۰/۱۶۹۹	۰/۲۰۵۸	۴
۰/۰۰۱۲	۰/۱۱۶۴	۵
۰/۰۲۹۰	۰/۳	۶
۰/۰۱۷۱	۰/۰۲۱۱	۷
۰/۰۱۳۲	۰/۰۱۵۰	۸
۰/۰۰۴۱	۰/۰۰۹۶	۹
۰/۰۰۱۱	۰/۰۰۳۴	۱۰

جدول (۲): بهترین پارامترهای طراحی برای پروانه پنج‌پره‌ای.

ردیف	کورد	ضخامت
۱	۰/۱۳۳۰	۰/۱۴۵۷
۲	۰/۱۶۹۱	۰/۱۸۹۷
۳	۰/۲۲	۰/۲۲۵۲
۴	۰/۱۹۵۵	۰/۱۵۲۹
۵	۰/۱۱۳۰	۰/۰۰۰۸
۶	۰/۰۳۵۰	۰/۰۳۱۶
۷	۰/۰۲۴۹	۰/۰۱۵۲
۸	۰/۰۱۴۷	۰/۰۱۴۳
۹	۰/۰۰۸۸	۰/۰۰۴۱
۱۰	۰/۰۰۳۳	۰/۰۰۰۰۰۹۷

در جدول (۳) ضرایب بی‌بعد عملکرد به دست آمده از طراحی بهینه ارائه شده است.

جدول (۳): عملکرد به دست آمده از طراحی بهینه.

Name	4 blade values	5 blade values
J	0. 88235	0. 88235
K <sub>T</sub>	0. 30382	0. 30382
K <sub>Q</sub>	0. 06224	0. 062153
.Effy	0. 68551	0. 68647
.Adeffy	0. 82919	0. 82919

با توجه به اینکه هدف دست‌یابی به بیشینه راندمان می‌باشد، مسئله بهینه‌سازی در واقع بیشینه‌سازی می‌باشد که به صورت زیر فرموله می‌گردد.

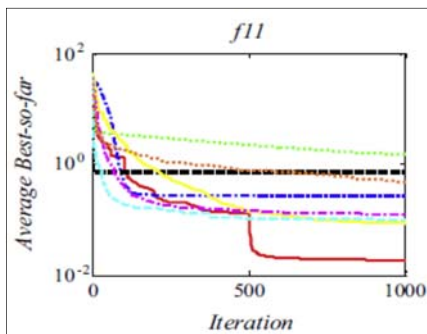
$$\dot{X}_i = (X_1, X_2, X_3 \dots X_n)$$

جدول (۴): فرمولاسیون توابع اولیه.

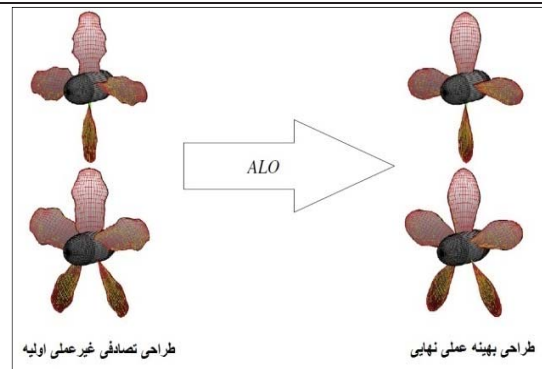
نام	رابطه
Sphere	$f(x) = \sum_{i=1}^D x_i^2$
Ackley	$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) - \left(\exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right) + 20 + e\right)$
Griewank	$\frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
Weierstrass	$\sum_{i=1}^D \left( \sum_{k=0}^{kmax} [a^k \cos(2\pi b^k(x_i + 0.5))] \right) - D \sum_{k=0}^{kmax} [a^k \cos(2\pi b^k 0.5)], a = 0.5, b = 3, kmax = 20$
Rastrigin	$\sum_{i=1}^D x_i^2 - 10 \cos(2\pi x_i) + 10$

### ۱۳- نتیجه گیری

با توجه به مثال بحث شده، این الگوریتم توانایی خوبی در بهینه‌سازی طراحی پروانه دارد. از آنجایی که در مثال، حالت‌های ساده مورد استفاده قرار گرفته و از پیچیدگی معادلات و مسئله صرف‌نظر گردیده است؛ امکان تعمیم این الگوریتم و ارائه الگوریتمی که بتواند سایر فاکتورهای طراحی پروانه را بهینه گرداند، متصور می‌باشد. سرعت همگرا شدن با توجه به شکل ۱۵ و نمودار همگرایی ارائه شده، در حل مسائل بهینه‌سازی در مدل‌های یکسان به خوبی نشان داده شده است.



الف



شکل (۵): بهبود طراحی اولیه توسط الگوریتم ALO.

### ۱۲- بحث و بررسی

تقریب بهینه‌سازی این مسئله با ۵۰ تکرار، در حدود یک هفته زمان برده و همان‌گونه که در جداول مشاهده شد، نتایج مطلوبی به دست آمده است. این نتایج مربوط به دو نوع پروانه و با شرایط اولیه مشخص شده می‌باشد. راندمان بهینه به دست آمده برای پروانه‌ی چهارپره‌ای در حدود ۰/۶۸۵۵ و برای پروانه‌ی پنج‌پره‌ای در حدود ۰/۶۸۶۴۷ می‌باشد. با توجه به شکل‌های ارائه شده کاملاً روشن است که این الگوریتم شکل بسیار صاف و موزونی را برای تیغه‌ها جهت دستیابی به راندمان بالاتر در هر دو پروانه نشان داده است.



### منابع

[۱] حاجیلوی، علی، شمسی، رضا، «طراحی بهینه‌ی پروانه‌های دریایی با در نظر گرفتن اثرات درگ»، پنجمین همایش ملی صنایع دریایی ایران، دانشگاه صنعتی شریف، ۱۳۸۲.

[۲] شمسی، رضا، «بررسی روش‌های طراحی پروانه‌های دریایی»، پایان‌نامه‌ی کارشناسی ارشد، دانشگاه صنعتی شریف، دانشکده مهندسی مکانیک، ۱۳۸۲.

[3] Carlton, J. S, "Marine propeller & propulsion," Butterworth, Heinemann Ltd., 2006.

[۴] قاسمی، حسن، «بکارگیری روش المان مرزی و الگوریتم سیستماتیکی برای طراحی پروانه‌ی کشتی»، مجله علمی پژوهشی مهندسی دریا، شماره ۱، تابستان ۱۳۸۳.

[۵] مختاری، حجت، موسوی‌زادگان، حسین، «الگوریتم طراحی بهینه‌ی پروانه برای شناورها با استفاده از سری B-Wageningen»، ایران، زیباکنار، (MIC2010)، دوازدهمین همایش صنایع دریایی ایران، مهرماه ۱۳۸۹.

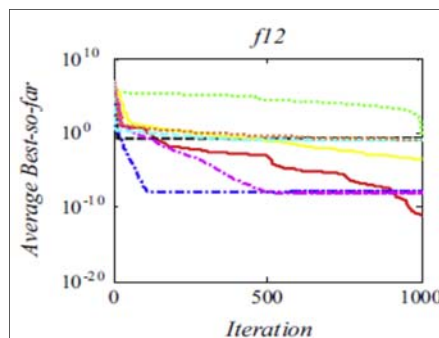
[6] Goldberg, "Degentic algorithm," Addison Wesley longman, Boston, PP. 2-8, 1989.

[7] Benini, E., "Multi-objective design optimization of B-Screw series propeller sing evolutionary algorithms," Mar Technol., Vol. 40, pp. 229-238, 2003.

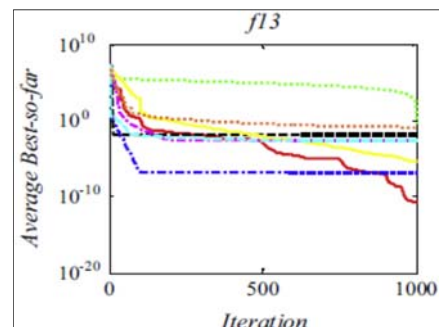
[8] Suen, J. B., and Kouh, J. S., "Genetic algorithms for optimal series propeller design-in," Proceeding of the third international conference on marine technology, ORDA 99 Szczshn, Poland, 1999.

[۹] رحمان‌نژاد، قاسمی، حسن، مرید حسونود، عباس، «طراحی بهینه‌ی پروانه‌ی کشتی با استفاده از الگوریتم ژنتیک»، یازدهمین همایش صنایع دریایی ایران، جزیره کیش، آبان ماه ۱۳۸۸.

[۱۰] اردشیری، سجاد، قاسمی، حسن، غیائی، محمود، «طراحی بهینه‌ی پروانه‌ی کشتی همراه با ملاحظات ارتعاشی با استفاده از الگوریتم ژنتیک»، دوازدهمین همایش صنایع دریایی ایران (MIC2010)، زیباکنار، ۲۹ مهرماه ۱۳۸۹.



ب



ج

شکل (۵): همگرایی الگوریتم در آزمون سه تابع چندهدفه.

این موضوع نیز به خوبی بیانگر عملکرد این الگوریتم بوده و استفاده از آن را در طراحی پروانه کاملاً توجیه‌پذیر می‌نماید. از طرفی با توجه به اینکه اساس کار این الگوریتم بهینه‌سازی با داشتن اطلاعات اولیه است، می‌توان با اطلاعات اولیه دقیق‌تر، نتایج مطلوب‌تری به دست آورد؛ به عبارت دیگر اگر ابتدا با استفاده از نرم‌افزارهای قدرتمند طراحی پروانه در حالت خاصی، اطلاعات اولیه دقیق‌تری را به دست آورده و با تزریق این اطلاع به عنوان ورودی به نرم‌افزار در آن حالت، به نتایج بهتری دست یافت.

اما نکته آخر؛ با توجه به اینکه نوع کاربری و داده‌های اولیه هر پروانه، محدوده مجاز و مطلوب فاکتورهای مؤثر در طراحی آن را مشخص می‌کند، جهت دسترسی مطلوب به طرح نمونه بهتر است این نمونه آماری برای هر نوع کاربری جداگانه باشد تا راندمان مربوطه نیز در آن زمینه بیشینه باشد؛ لذا بهتر است ابتدا با استفاده از نرم‌افزارهای طراحی پروانه، یک جامعه آماری خوب تهیه و از آن به عنوان ورودی جهت الگوریتم آنلاین استفاده کرد تا خروجی‌های بهینه‌تری به دست آوریم.

[11] Mirjalili, S., "The ant lion optimizer," Advances in Engineering Software, Vol. 83, pp. 80-98, May 2015.

[12] Chalfamat, J., Kimball, R., and Flood, U., "Chryssostomidis C. O., openprop an open source parametric design & analysis tool for propellers," in Proceeding of the 2009 grand challenges in modeling & simulation conference, pp. 164-11, 2009.