

## الگوریتم جستجوی فنر

محمد دهقانی<sup>۱</sup> و علیرضا سیفی<sup>۲</sup>

<sup>۱</sup>دانشگاه شیراز، دانشکده برق و کامپیوتر، شیراز، mdeghani6889@gmail.com

<sup>۲</sup>دانشگاه شیراز، دانشکده برق و کامپیوتر، شیراز، seifi@shirazu.ac.ir

چکیده - امروزه از الگوریتم‌های جمعیتی ابتکاری مبتنی بر تصادف، به منظور بهینه‌یابی استفاده گسترده‌ای می‌شود. دسته مهمی از این الگوریتم‌ها با ایده از فرایندهای فیزیکی یا رفتارهای موجودات به وجود آمده‌اند. در این مقاله یک روش جدید جهت دستیابی به جواب‌های شبه‌بهینه مربوط به مسائل بهینه‌سازی در علوم مختلف معرفی شده است. روش پیشنهادی الگوریتم جستجوی فنر، از سری الگوریتم‌های بهینه‌یابی است که با ایده از قوانین موجود در طبیعت شکل یافته و عامل‌های جستجوگر، مجموعه‌ای از اجسام است. در این مقاله نسخه حقیقی الگوریتم ارائه شده است. نتایج بهینه‌سازی چند تابع استاندارد، کارایی و عملکرد مطلوب روش پیشنهادی را نشان می‌دهد. کلید واژه- الگوریتم‌های هیوریستیک، بهینه‌یابی، بهینه‌یابی نیروی فنر، فنر، نیروی فنر.

مقاله از ویژگی‌های مربوط به قانون هوک<sup>۱۲</sup> برای طراحی یک بهینه‌یاب با نام الگوریتم جستجوی تکانه (SSA<sup>۱۳</sup>) استفاده شده است.

ابتدا در فصل دوم قانون نیروی فنر آورده می‌شود. سپس در بخش سوم طرح کلی الگوریتم و در بخش چهارم ویژگی‌های آن توضیح داده می‌شود. در بخش پنجم کاوش و بهره‌وری الگوریتم پیشنهادی، در بخش ششم نتایج و در نهایت جمع‌بندی و نتیجه‌گیری آورده می‌شود.

### ۲- قانون نیروی فنر

اگر نیرویی که جسم را حرکت می‌دهد در یک مسیر بسته (رفت و برگشت) هیچ کار خالصی روی جسم انجام ندهد، آن نیرو پایستار است. راه دیگر تشخیص نیروهای پایستار این است که کار انجام‌شده توسط نیرو در مسیرهای مختلف، با نقاط شروع و پایان یکسان، برابر باشد. نیروی بازگرداننده کشسان (نیروی فنر) یک نمونه از نیروهای پایستار است [۱۳].

معادله ۱ را قانون نیروی فنر (قانون هوک) می‌نامند. بیشتر فنرهای واقعی به شرط اینکه بیش از حد معینی کشیده نشوند به خوبی از قانون هوک پیروی می‌کنند [۱۳ و ۱۴].

$$F_s = -kx \quad (1)$$

در اینجا،  $k$ ، ثابت نیروی فنر،  $x$ ، میزان کشیدگی یا فشردگی فنر و  $F_s$ ، نیروی فنر است.

در این مقاله با شبیه‌سازی قانون هوک در محیطی با زمان گسسته، بهینه‌یاب جدیدی به نام بهینه‌ساز فنر طراحی شده است که در بخش بعد به آن پرداخته می‌شود.

### ۱- مقدمه

امروزه با توجه به گسترده‌گی و پیچیده شدن مسایل و اهمیت یافتن سرعت رسیدن به پاسخ، دیگر روش‌های کلاسیک بهینه‌یابی، توانایی جوابگویی حل بسیاری از مسایل را ندارند و بیشتر از الگوریتم‌های جستجوی تصادفی به جای جستجوی همه‌جانبه فضای تعریف مسأله استفاده می‌شود. این موضوع باعث شده است تا استفاده از الگوریتم‌های جستجوی هیوریستیک<sup>۱</sup> (شهودی یا ابتکاری) در سال‌های اخیر رشد چشمگیری داشته باشد [۱ و ۲ و ۳ و ۴ و ۵]. الگوریتم‌های ابتکاری قابلیت بالای خود را در بسیاری از زمینه‌های علم همچون حمل و نقل<sup>۲</sup> [۶]، بیوانفورماتیک<sup>۳</sup> [۷]، داده‌کاوی<sup>۴</sup> [۸]، شیمی فیزیک<sup>۵</sup> [۹]، الکترونیک [۱۰] و دیگر زمینه‌های مرتبط، به خوبی نشان داده‌اند. دستیابی به یک مدل ریاضی مناسب، به منظور فرآیند جستجوی روش‌های ابتکاری، کار بسیار سخت و حتی نشدنی است [۸]. بنابراین از این نوع الگوریتم‌ها، می‌توان به عنوان الگوریتم‌های بهینه‌سازی "جعبه سیاه"<sup>۶</sup> نیز نام برد [۱۱].

در روش‌های جمعیتی تعاملات و تبادل اطلاعات بین اعضا به روش‌های مختلف صورت می‌گیرد. از نمونه‌های این الگوریتم‌ها، الگوریتم وراثتی<sup>۷</sup> با الهام از علم وراثت و تکامل (۱۹۷۵)، پخت شبیه‌سازی‌شده<sup>۸</sup> با الهام از مشاهدات ترمودینامیک (۱۹۸۳)، الگوریتم ایمنی<sup>۹</sup> با شبیه‌سازی سیستم دفاعی انسان (۱۹۸۶)، الگوریتم جستجوی مورچگان<sup>۱۰</sup> با شبیه‌سازی رفتار مورچه‌ها در جستجوی غذا (۱۹۹۱) و بهینه‌سازی جمعیت ذره‌ها<sup>۱۱</sup> با تقلید از رفتار اجتماعی پرندگان (۱۹۹۵) [۱ و ۲ و ۳ و ۴ و ۵] می‌باشند. در این

## ۳- بهینه یابی نیروی فنر

می شود.

$$K_{i,j} = K_{max} |F_n^i - F_n^j| \max(F_n^i, F_n^j) \quad (۳)$$

در معادله فوق،  $K_{i,j}$ ، ضریب سختی فنر بین جسم  $i$  و جسم  $j$ ،  $K_{max}$ ، ماکزیمم مقدار ضریب سختی فنر که با توجه به نوع مسأله مشخص میشود،  $F_n^i$  نرمال شده تابع هدف و  $F_n^j$  نرمال شده تابع هدف اجسام  $i$  و  $j$  است. به منظور نرمال کردن تابع هدف، از معادلات ۴ و ۵ استفاده شده است:

$$F_n^i = \frac{f_{obj}^i}{\min(f_{obj})} \quad (۴)$$

$$F_n^i = \min(F_n^i) \times \frac{1}{F_n^i} \quad (۵)$$

در روابط فوق،  $f_{obj}$  تابع هدف و  $f_{obj}^i$  مقدار تابع هدف برای جسم  $i$  است.

در یک مسأله  $m$  متغیره، می توان به این صورت فرض کرد که مسأله  $m$  بعدی است و به ازای هر بعد یک محور تعریف کرد، بنابراین می توان معادل هر متغیر سیستم را بر روی محور مرتبط، تصویر کرد. بر روی هر محور، نقاط محکم سمت راست و سمت چپ جسم، با توجه به مقایسه مقدار تابع هدف مشخص می شود. نقاط محکم مربوط به هر جسم، در واقع همان اجسامی هستند که در موقعیت بهینه تری نسبت به آن جسم قرار دارند. بنابراین بر روی هر محور، دو نیروی برآیند کلی به جسم وارد می شود: برآیند نیروهای سمت راست و برآیند نیروهای سمت چپ که به منظور محاسبه آنها داریم:

$$F_{totalR}^{j,d} = \sum_{i=1}^{n_R^d} K_{i,j} x_{i,j}^d \quad (۶)$$

$$F_{totalL}^{j,d} = \sum_{l=1}^{n_L^d} K_{l,j} x_{l,j}^d \quad (۷)$$

در روابط فوق (۶ و ۷)،  $F_{totalR}^{j,d}$  برآیند نیروهایی است که از سمت راست و  $F_{totalL}^{j,d}$  برآیند نیروهایی است که از سمت چپ، به جسم  $j$  در بعد  $d$  وارد شده است.  $n_L^d$  و  $n_R^d$  به ترتیب تعداد نقاط محکم سمت راست و سمت چپ بعد  $d$ ام،  $x_{i,j}^d$  و  $x_{l,j}^d$  نشان دهنده فاصله جسم  $j$  از نقاط محکم سمت راست و سمت چپ بعد  $d$ ام و  $K_{i,j}$  و  $K_{l,j}$  نیز ضریب سختی فنر متصل بین جسم  $j$  و نقاط محکم است.

حال با بکارگیری قانون هوک در بعد  $d$ ام داریم:

$$dX_R^{j,d} = \frac{F_{totalR}^{j,d}}{K_{equalR}^j} \quad (۸)$$

$$dX_L^{j,d} = \frac{F_{totalL}^{j,d}}{K_{equalL}^j} \quad (۹)$$

در اینجا،  $dX_R^{j,d}$  و  $dX_L^{j,d}$  به ترتیب، مقدار جابجایی جسم  $j$  به سمت راست و به سمت چپ، در بعد  $d$  است. بنابراین،

در این مقاله بهینه یابی، به کمک طرح قوانین نیروی فنر، در یک سیستم مصنوعی با زمان گسسته انجام می شود. محیط سیستم همان محدوده تعریف مسأله می باشد. می توان از قانون نیروی فنر، به عنوان ابزاری برای تبادل اطلاعات استفاده کرد. از بهینه یاب طراحی شده برای حل هر مسأله بهینه سازی که در آن هر جواب مسأله به صورت یک موقعیت در فضا قابل تعریف و میزان شباهت آن با سایر جواب های مسأله به صورت مقایسه سختی فنر قابل بیان باشد می توان استفاده نمود. میزان سختی فنر با توجه به تابع هدف تعیین می شوند.

الگوریتم SSA در دو قدم کلی توضیح داده می شود: ۱- تشکیل یک سیستم مصنوعی با زمان گسسته در محیط مسأله، موقعیت یابی اولیه برای اجسام، وضع قوانین حاکم و تنظیم پارامترها، ۲- گذر زمان تا پیش آمدن زمان توقف.

## ۳-۱- تشکیل سیستم، وضع قوانین و تنظیم پارامترها

در قدم اول فضای سیستم مشخص می شود. محیط شامل یک دستگاه مختصات چند بعدی در فضای تعریف مسأله است. هر نقطه از فضا یک جواب مسأله است. عامل های جستجو کننده، مجموعه ای از اجسام می باشند که توسط فنر هایی به یکدیگر متصل شده اند. در واقع هر جسم بوسیله فنر به تمامی اجسام دیگر متصل است، هر یک از اجسام دارای مشخصه موقعیت و ضریب سختی فنر های متصل به آن است. موقعیت جسم، نقطه ای در فضا است که جوابی از مسأله است. مقدار ضریب سختی فنر ها با توجه به برآزندگی هر دو جسم متصل بهم تعیین می شود. پس از تشکیل سیستم، قوانین حاکم بر آن مشخص میشوند. فرض میکنیم تنها قانون نیروی فنر و قوانین حرکت حاکم هستند.

حال سیستم را به صورت مجموعه ای از  $m$  جسم تصور کنید. موقعیت هر جسم نقطه ای از فضا است که جوابی از مسأله می باشد. در رابطه ۲، موقعیت بعد  $d$  از جسم  $i$  با  $x_i^d$  نشان داده شده است.

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^n) \quad (۲)$$

در ابتدا، موقعیت اولیه اجسام را به صورت تصادفی، در فضای تعریف مسأله تعریف می کنیم، این اجسام با توجه به نیرو هایی که توسط فنر به یکدیگر وارد می کنند به سمت تعادل سیستم (جواب)، پیش می روند.

به منظور محاسبه ضریب سختی فنر، از معادله ۳، استفاده

خواهیم داشت:

$$dX^{j,d} = dX_R^{j,d} + dX_L^{j,d} \quad (10)$$

میزان جابجایی نهایی جسم  $j$  در راستای بعد  $d$  است که مقدار آن با توجه به معادله ۱۰ می‌تواند مثبت یا منفی باشد. اکنون داریم:

$$X^{j,d} = X_0^{j,d} + r_1 \times dX^{j,d} \quad (11)$$

در معادله فوق،  $X^{j,d}$  مربوط به مکان و نقطه تعادل جدید بعد  $d$  سیستم و جسم  $j$  است. همچنین  $X_0^{j,d}$  مکان اولیه تعادل جسم  $j$  در راستای بعد  $d$  است. در اینجا  $r_1$  یک عدد تصادفی با توزیع یکنواخت در بازه  $[0-1]$  است که به منظور حفظ حالت تصادفی بودن جستجو استفاده شده است.

### ۳-۲- گذر زمان و به روز رسانی پارامترها

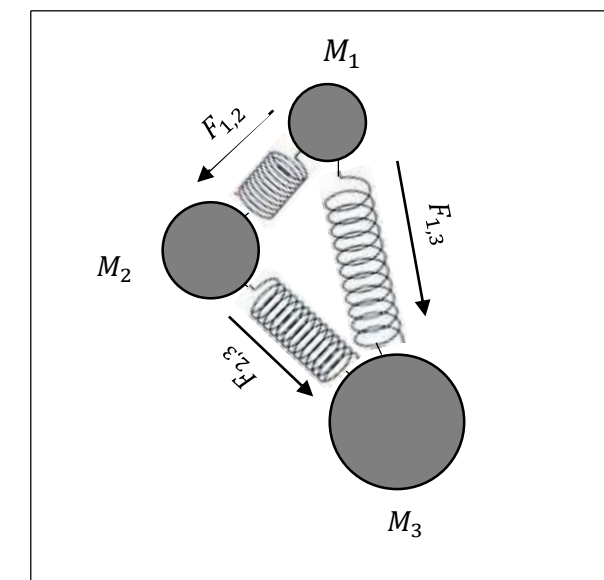
در ابتدای تشکیل سیستم، هر جسم به صورت تصادفی در یک نقطه از فضا قرار می‌گیرد که جوابی از مساله است. در هر لحظه از زمان، اجسام ارزیابی شده سپس، تغییر مکان هر جسم پس از محاسبه روابط ۳ تا ۱۰ به دست آمده، در زمان بعد جسم در آن موقعیت قرار می‌گیرد. پارامتر سیستم ضریب سختی فنر می‌باشد که در هر مرحله طبق رابطه ۳ بروز رسانی می‌شود. شرط توقف می‌تواند پس از طی مدت زمان مشخصی تعیین شود. مراحل مختلف الگوریتم نیروی فنر به صورت زیر است:

- ۱- تعیین محیط سیستم و مقداردهی اولیه.
  - ۲- جایابی اولیه اجسام.
  - ۳- ارزیابی و نرمالیزه کردن برازندگی اجسام.
  - ۴- به روز رسانی پارامتر  $k$
  - ۵- تشکیل قوانین نیروی فنر برای هر یک از اجسام.
  - ۶- محاسبه مقدار جابجایی اجسام.
  - ۷- به روز رسانی موقعیت اجسام.
  - ۸- تا زمانی که شرط توقف برآورده نشده است، مراحل ۳ تا ۷ را تکرار کن.
- پایان.

### ۴- خصوصیات الگوریتم پیشنهادی

در الگوریتم فوق سعی شده با استفاده از قانون نیروی فنر روشی جهت بهینه‌یابی خلق شود. در این الگوریتم مجموعه‌ای از اجسام فضا را به صورت تصادفی جستجو می‌کنند. از نیروی فنر به عنوان ابزار جهت تبادل اطلاعات استفاده شده است. هر جسم با تأثیر از موقعیت اجسام دیگر به درک تقریبی از فضای

پیرامون خود می‌رسد. الگوریتم باید به گونه‌ای هدایت شود که موقعیت اجسام با گذشت زمان بهبود پیدا کند. استراتژی به کار گرفته شده جهت این امر، تنظیم ضریب سختی فنر است. بنابراین به اجسامی که تابع برازش بهتری دارند، فنری با ضریب سختی بیشتری متصل می‌شود و اجسام دیگر را به سمت خود می‌کشاند. در نتیجه به هر جسم، نیرو متناسب با اندازه شایستگی آن جسم وارد می‌شود. پس انتظار داریم با گذشت زمان اجسام به سمت موقعیت‌های مناسب‌تر بروند. همینطور اجسامی که در موقعیت‌های بهتری هستند باید گام‌های کوتاه‌تر و حرکت آهسته‌تری داشته باشند. جهت رسیدن به این هدف به اجسام بهتر، فنر با ضریب سختی بزرگتری نسبت می‌دهیم. این امر باعث می‌شود هر چه جسم برازندگی بهتری داشته‌باشد، فضای اطراف خود را با دقت بیشتری جستجو کند. این مساله مثل تنظیم نرخ یادگیری در شبکه‌های عصبی است که در اینجا تطبیقی برای هر جسم است.



شکل ۱: هر جسم با توجه به نیروهای فنر وارد شده به آن در الگوریتم SSA، جابجا می‌شود.

## ۵- کاوش و بهره‌وری در الگوریتم SSA

$V_i^d(t)$ ، سرعت بعد  $d$  ذره  $i$  در زمان  $t$  و  $r_1$  و  $r_2$  اعداد تصادفی با توزیع یکنواخت بین صفر تا یک می‌باشند. همچنین  $gbest$  بهترین موقعیتی است که تا کنون توسط جامعه یافت شده است و  $pbest_i$  بهترین موقعیتی است که ذره  $i$  تا کنون به آن دست یافته است.

$$V_i^d(t+1) = w(t)V_i^d(t) + c_1 r_{1i}(t)[pbest_i^d(t) - X_i^d(t)] + c_2 r_{2i}(t)[gbest^d(t) - X_i^d(t)] \quad (12)$$

به منظور بهینه‌سازی توابع هدف معرفی شده، همانطور که در جدول شماره ۲، دیده می‌شود، الگوریتم پیشنهادی SSA، نسبت به الگوریتم‌های RGA و PSO، عملکرد و نتایج قابل قبولی دارد. به منظور بررسی روند پیشرفت فرآیند بهینه‌سازی، نحوه دستیابی به جواب بهینه در شکل‌های ۲ تا ۶، برای تمامی توابع آورده شده است. نتایج نشان‌دهنده قدرت همگرایی بالای الگوریتم SSA، نسبت به الگوریتم‌های RGA و PSO است.

جدول ۱: توابع محک مینیمم‌شونده استفاده‌شده در آزمایشها،  $n$  مشخص‌کننده بعد مسأله است. مقدار بهینه همه توابع صفر است.

Test functions	$R^n$
$f_1(x) = \sum_{i=1}^n x_i^2$	$[-100, 100]^n$
$f_2(x) = \sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	$[-10, 10]^n$
$f_3(x) = \sum_{i=1}^n ([x_i + 0.5])^2$	$[-100, 100]^n$
$f_4(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	$[-100, 100]^n$
$f_5 = \sum_{i=1}^n i x_i^4 + random[0,1]$	$[-1.28, 1.28]^n$

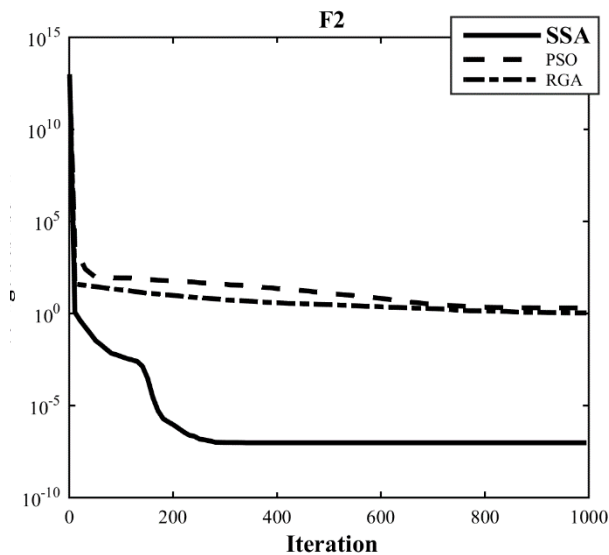
## ۶- نتایج

برای ارزیابی روش پیشنهادی، این الگوریتم، الگوریتم ژنتیک و الگوریتم جمعیت پرندگان در حل مسائل کمینه‌یابی جدول ۱، در شرایطی برابر پیاده‌سازی شده‌اند. برای حالت  $n=30$ ، تعداد جمعیت برابر ۵۰، نتایج برای ۱۰۰۰ مرتبه تکرار الگوریتم در جدول ۲ آورده شده است و برای مقایسه، متوسط شایستگی و بهترین جوابی که تا آن لحظه مشاهده شده است محاسبه شده‌اند. این پارامترها برای ۲۰ مرتبه اجرای مستقل برنامه محاسبه شده و میانه نتایج آمده است. نتایج کارایی بهتر الگوریتم نیروی فنر را نشان می‌دهد.

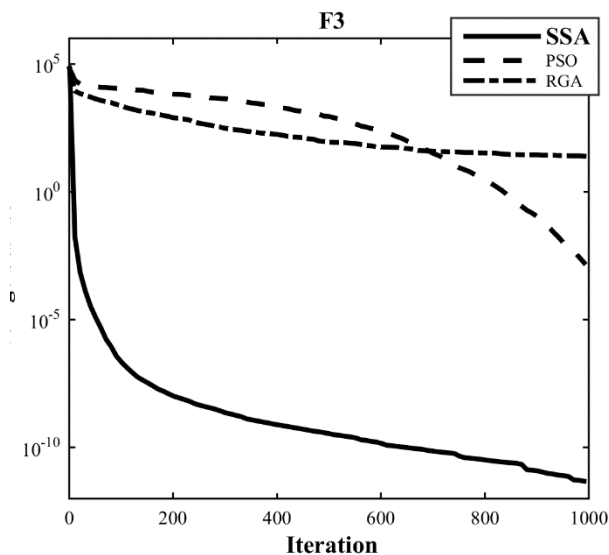
در شبیه‌سازی PSO، جهت بروزرسانی سرعت ذرات از رابطه ۱۲ استفاده شده است. در این رابطه،  $C_1 = C_2 = 2$  و  $W$  به صورت خطی از  $0.9$  تا  $0.2$  کاهش می‌یابد. در این رابطه

جدول ۲: نتایج پیاده‌سازی بهینه‌سازی روی توابع جدول ۱.

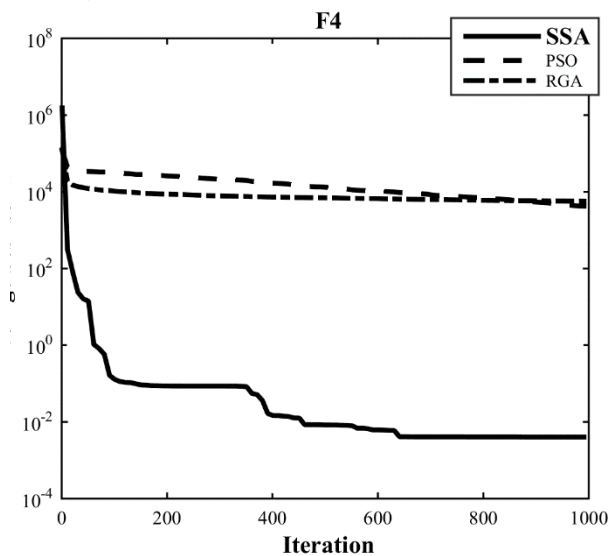
تابع	خروجی	MSA	PSO	RGA
$f_1$	میانگین بهترین جواب	$1.5 \times 10^{-11}$	$1.9 \times 10^{-3}$	23.13
	میانگین شایستگی	$6.5 \times 10^{-9}$	$5.1 \times 10^{-2}$	23.45
$f_2$	میانگین بهترین جواب	$9.4 \times 10^{-8}$	2.1	1.08



شکل ۳: عملکرد الگوریتم‌های SSA، PSO و RGA بر روی تابع  $F_2$

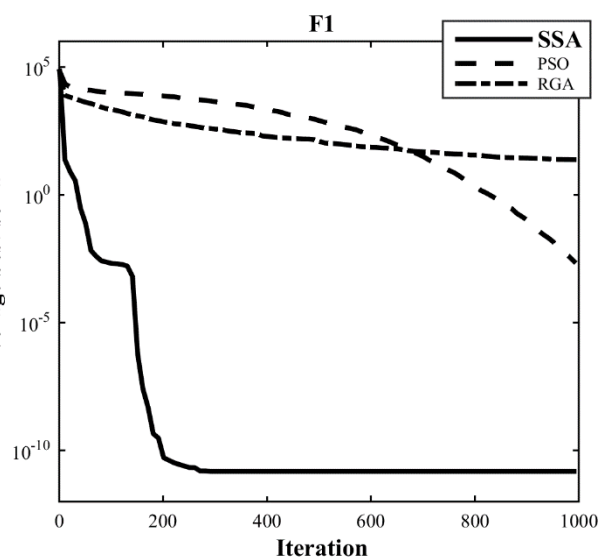


شکل ۴: عملکرد الگوریتم‌های SSA، PSO و RGA بر روی تابع  $F_3$



شکل ۵: عملکرد الگوریتم‌های SSA، PSO و RGA بر روی تابع  $F_4$

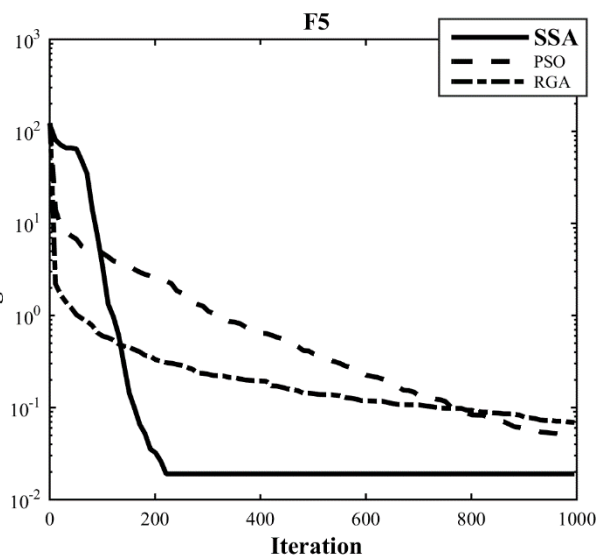
1.08	2.1	$2.6 \times 10^{-7}$	میانگین شایستگی	$f_3$
24.02	$1.1 \times 10^{-3}$	$4.6 \times 10^{-12}$	میانگین بهترین جواب	
24.53	$6.7 \times 10^{-3}$	$3.1 \times 10^{-11}$	میانگین شایستگی	$f_4$
5700	$4.2 \times 10^3$	$4 \times 10^{-3}$	میانگین بهترین جواب	
5700	$2.8 \times 10^3$	$8.2 \times 10^{-2}$	میانگین شایستگی	$f_5$
0.07	0.04	0.019	میانگین بهترین جواب	
0.57	1.04	0.612	میانگین شایستگی	



شکل ۲: عملکرد الگوریتم‌های SSA، PSO و RGA بر روی تابع  $F_1$

## مراجع

- [1] K.S. Tang, K.F.Man, S.Kwong and Q.He, "Genetic algorithms and their applications", IEEE Signal Processing Magazine 13 (6), 1996.
- [2] S.Kirkpatrick, , C.D.Gelatto and M.P.Vecchi, "Optimization by simulated annealing", Science 220 (4598), 671-680. 1983.
- [3] J.D. Farmer, N.H.Packard and , A.S.Perelson," The immune system, adaptation, and machine learning", Physica D 22, 187-204. 1986.
- [4] M.Dorigo, V.Maniezzo, and A.Colomi, "The Ant System: optimization by a colony of cooperating agents.", IEEE Transaction on systems, Man, and Cybernetics-part B, vol. 26, no.1, 1996, pp. 1-13.
- [5] J.Kennedy and R.C.Eberhart, "Particle swarm optimization", Proceedings of IEEE International Conference on Neural Networks, vol. 4, 1995.
- [6] FazelZarandi, M.H., Hemmati, A., Davari, S. "The multi-depot capacitated location-routing problem with fuzzy travel times." Expert Systems with Applications, vol. 38, no. 8, pp. 10075-10084 (2011).
- [7] Mitra, S., Banka, H. "Multi-objective evolutionary biclustering of gene expression data." Pattern Recognition, vol. 39, no. 12, pp. 2464-2477(2006).
- [8] Zahiri, S.H. "Swarm Intelligence and Fuzzy Systems." Nova Science Publishers, USA (2010).
- [9] Darby, S., Mortimer-Jones, T.V., Johnston, R.L.,Roberts, C, "Theoretical study of CuAunanoalloy clusters using a genetic algorithm." Journal of Chemical Physics, vol. 116, no. 4, pp. 1536-1550 (2002).
- [10] CoelloCoello, C.A., Luna, E.H., Aguirre, A.H. "Use of Particle Swarm Optimization to Design Combinational Logic Circuits. In: Evolvable Systems." From Biology to Hardware, Springer Berlin, Heidelberg, pp. 398-409 (2003).
- [11] Wolpert, D.H.,Macready, W.G. "No free lunch theorems for optimization." Evolutionary Computation, IEEE Transactions on, vol. 1, no. 1, pp. 67-82 (1997).
- [12] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S. "GSA: A Gravitational Search Algorithm." Information Sciences, vol. 179, no. 13, pp. 2232- 2248 (2009).
- [13] D.Holliday, R.resnick and J.Walker, Fundamentals of physics, John wiley and sons, 1993.
- [14] Lee, C.K.; Tan, S.C.; Wu, F.F.; Hui, S.Y.R.; Chaudhuri, B, " Use of Hooke's Law for Stabilizing Future Smart Grid - The Electric Spring Concept" Energy Conversion Congress and Exposition (ECCE), 2013 IEEE.
- [15] Eiben, A.E., Schippers, C.A, "On Evolutionary Exploration and Exploitation." FundamentaInformaticae, vol. 35, no. 1-4, IOS Press, Amsterdam (1998).



شکل ۶. عملکرد الگوریتم‌های SSA, RGA و PSO بر روی تابع F5

## ۷- جمع‌بندی و نتایج

امروزه از الگوریتم‌های مبتنی بر تصادف، جهت بهینه‌یابی استفاده گسترده‌ای می‌شود. غالب این الگوریتم‌ها با الهام از فرایندهای فیزیکی و یا رفتارهای موجودات به وجود آمده‌اند. در این مقاله رویکرد جدید استفاده از قانون نیروی فنر جهت بهینه‌سازی بررسی شده‌است. الگوریتم پیشنهادی با الهام از قوانین موجود در طبیعت طراحی شده و نتایج آرایه‌شده، عملکرد مناسب آن را در بهینه‌یابی نشان می‌دهد. نتایج مقایسه روش پیشنهادی با الگوریتم PSO و الگوریتم ژنتیک کارایی مناسب‌تر الگوریتم فنر را نشان می‌دهد.

<sup>۸</sup> . Simulated Annealing (SA).

<sup>۹</sup> . Artificial Immune System (AIS).

<sup>۱۰</sup> . Ant Colony Optimization (ACO).

<sup>۱۱</sup> . Particle Swarm Optimization (PSO).

<sup>۱۲</sup> . Hock.

<sup>۱۳</sup> . Spring search algorithm.(SSA)

<sup>۱</sup> . Heuristic.

<sup>۲</sup> . Logistics.

<sup>۳</sup> . Bioinformatics.

<sup>۴</sup> . Data mining.

<sup>۵</sup> . Chemical physics.

<sup>۶</sup> . Black box.

<sup>۷</sup> . Genetic Algorithm (GA).