

پیاده‌سازی الگوریتم گروه ذرات احتمالی در رویکرد بهینه‌سازی از طریق شبیه‌سازی

محمدعلی آزاده^۱، شیما پاشاپور^{۲*} و هیوا شیرینی^۳

^۱ استاد دانشکده مهندسی صنایع - پردیس دانشکده‌های فنی - دانشگاه تهران

^۲ دانشجوی دکترای مهندسی صنایع - پردیس دانشکده‌های فنی - دانشگاه تهران

^۳ فارغ التحصیل کارشناسی ارشد مهندسی صنایع - پردیس دانشکده‌های فنی - دانشگاه تهران

(تاریخ دریافت ۹۱/۲/۱۶، تاریخ دریافت روایت اصلاح شده ۹۲/۱۲/۱۷، تاریخ تصویب ۹۳/۱/۲۴)

چکیده

بهینه‌سازی از طریق شبیه‌سازی، ابزاری کارا و جافتاده در مواجهه با مسائل بهینه‌سازی به شمار می‌رود. ابزاری که قادر است شرایط حاکم بر مسائل دنیای واقعی را نسبت به سایر ابزارهای بهینه‌سازی به گونه‌ای بسیار کامل‌تر پوشش دهد. مشکلی که در استفاده از این ابزار وجود دارد، لزوم پردازش سنگین محاسباتی است. این مشکل از آنجا ناشی می‌شود که نه تنها فرایند شبیه‌سازی، خود زمان‌بر است، بلکه محاسبه مقدار تابع هدف در یک نقطه از فضای جواب، مستلزم اجرای متعدد شبیه‌سازی است. متامدل‌ها رویکردی برای مقابله با این معضل هستند و خود مشکلات دیگری به همراه دارند که استفاده از آنها را محدود می‌کند. متامدل، مدل شبیه‌سازی را بار دیگر مدل می‌کند؛ از پیچیدگی‌های آن می‌کاهد و در عین حال مدل را باز هم بیشتر از حالت واقعی‌اش دور می‌کند. در این مقاله، الگوریتم دسته ذرات، به گونه‌ای تغییر یافته است که بتواند در عین حذف متامدل، زمان دستیابی به جواب بهینه را کاهش دهد.

واژه‌های کلیدی: الگوریتم گروه ذرات، بهینه‌سازی از طریق شبیه‌سازی، متامدل

مقدمه

با افزایش تکرار شبیه‌سازی‌ها با توجه به موضوع حد مرکزی به مقدار حدی خود همگرا می‌شود. کو و دیگران [۱] برای بهینه‌سازی خط تولید از رویکرد بهینه‌سازی از طریق شبیه‌سازی استفاده کرده‌اند. گیو و همکارانش [۲] یک سیستم بهینه‌سازی شبیه‌سازی برای سیستم‌های تولیدی، ارائه داده‌اند. آزاده و همکارانش [۳] از ترکیب شبیه‌سازی و شبکه عصبی برای بهینه‌سازی سیستم‌های صف پشت سر هم پیچیده استفاده کرده‌اند. وان و همکارانش [۴] از شبیه‌سازی برای بهینه‌سازی مدیریت زنجیره تأمین استفاده کرده‌اند. واقفی و سرهنگیان [۵] از شبیه‌سازی برای بهینه‌سازی برنامه‌های بازرسی سیستم‌های تولیدی چندمرحله‌ای استفاده کرده‌اند. معمارجعفری و همکارانش [۶] از شبیه‌سازی گسسته برای مطالعه فرآیندها و تحلیل عملکرد سیستم در یک شرکت تأمین قطعات یدکی خودرو استفاده کرده‌اند. نظری و آقایی [۷] مدلی برای اندازه‌گیری اثر شلاقی در شبکه‌های دو و سه مرحله‌ای با بیش از یک محصول ارائه داده‌اند. آنها با استفاده از شبیه‌سازی به تعیین اعتبار مدل ارائه شده

بهینه‌سازی از طریق شبیه‌سازی، ابزاری کارا و جافتاده در مواجهه با مسائل بهینه‌سازی به شمار می‌رود. در شرایطی که سایر روش‌های بهینه‌سازی قادر به در نظر گرفتن محدودیت‌های حاکم بر مسئله نباشند، بهینه‌سازی از طریق شبیه‌سازی می‌تواند راه‌گشا باشد. در مسائلی که طبیعت احتمالی دارند و روش‌های ریاضی‌وار همانند نظریه صف، قادر به مدلسازی مسئله نباشد، شبیه‌سازی به صورت وسیع استفاده می‌شود. در رویکرد بهینه‌سازی از طریق شبیه‌سازی، از شبیه‌سازی برای محاسبه مقدار تابع هدف مسئله و در عین حال در نظر گرفتن شرایط و محدودیت‌های مسئله استفاده می‌شود. متغیرهای تصمیم مسئله در قالب ورودی به شبیه‌سازی داده می‌شود و با اجرای فرایند شبیه‌سازی، مقدار تابع هدف مسئله به عنوان خروجی حاصل از شبیه‌سازی تعیین می‌شود. با توجه به طبیعت احتمالی مسئله، اجراهای مختلف شبیه‌سازی نتایج متفاوتی به دست می‌دهد و آنچه مدنظر ما است، میانگین این نتایج است. این مقدار میانگین، نتیجه انتظاری است که از ورودی مورد نظر ما حاصل می‌شود و

لی و دیگران [۹] در پژوهشی متامدل‌های شبکه عصبی مصنوعی^۲، تابع پایه شعاعی^۳، ماشین بردار پشتیبان^۴، کریگینگ^۵ و نوار رگرسیونی تطبیقی چندمتغیره^۶ را از جوانب مختلف با هم مقایسه کردند. کیفیت جواب‌های حاصله به عنوان یک معیار کمی و دسترس‌پذیری به برنامه نرم‌افزاری مربوطه، لزوم تنظیم کردن پارامترهای متامدل و تفسیرپذیری جواب‌های متامدل به عنوان معیارهای کیفی مورد توجه قرار گرفته‌اند. نتیجه این پژوهش به خوبی مشکلات استفاده از متامدل‌ها را نشان می‌دهد. برای شناخت بهتر متامدل‌ها می‌توانید به منابع نوار رگرسیونی تطبیقی چند متغیره [۱۰-۱۴] کریگینگ [۱۷-۱۵]، تابع پایه شعاعی [۲۲-۱۸]، شبکه عصبی مصنوعی [۲۶-۲۳]، ماشین بردار پشتیبان [۳۲-۲۷] رجوع کنید. بازنگری کاملی از متامدل‌های استفاده شده در رویکرد بهینه‌سازی از طریق شبیه‌سازی در کار ونگ و شان [۳۳] قابل دسترسی است. مقایسه متامدل‌های مختلف از زوایای متفاوت در منابع [۳۴]، [۳۵] و [۳۶] انجام گرفته است.

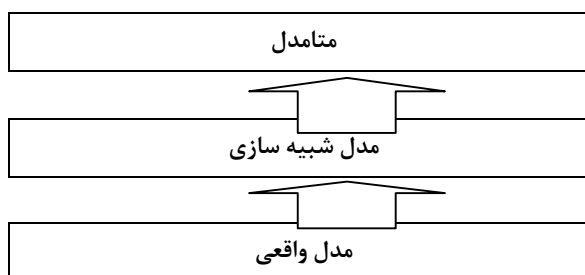
الگوریتم ارائه شده در این مقاله، بر پایه الگوریتم گروه ذرات کلاسیک است که با تغییراتی، توانایی جستجوی جواب در فضای جواب احتمالی را دارد.

روش‌شناسی

پیدایش مفهوم هوش جمعی^۷ به سال‌ها قبل از ابداع الگوریتم گروه ذرات برمی‌گردد. هوش جمعی به توانایی گروهی از موجودات مجرد به هدفی واحد اطلاق می‌شود. در فلسفه هوش جمعی، هوش هویتی جای‌گزیده نیست و به فرد خاصی در گروه تعلق ندارد، بلکه این گروه است که می‌اندیشد و راه خود را پیدا می‌کند. زندگی جمعی مورچه‌ها مثال بارزی از پیاده‌سازی هوشمندانه سیستمی هوشمند از اجزای ساده و تک‌بعدی است. مورچه‌های کارگر در بازگشت به کلونی، نزدیک‌ترین مسیر را انتخاب می‌کنند و این توانایی فوق‌العاده از سیستمی بسیار ساده نشأت می‌گیرد. توانایی گروه ماهیان در مقابله با حمله کوسه مثال دیگری است. با نزدیک شدن کوسه به گروه، ماهیان مسیر حرکت خود را با توجه به مسیر حرکت اطرافیان‌شان انتخاب می‌کنند و این دستور ساده در نهایت به حرکت پیچیده گروه ماهیان و فرار آنها منجر می‌شود.

پرداخته‌اند. حسن ناییبی و کیانی [۸] از ترکیب شبیه‌سازی و مدلسازی سیستم داینامیک برای تحلیل ریسک زمان پروژه‌ها استفاده کرده‌اند.

مشکل اصلی در استفاده از رویکرد بهینه‌سازی از طریق شبیه‌سازی، زمان‌بر بودن پردازش محاسباتی لازم برای اجرای آن است. این موضوع از دو جهت ناشی می‌شود: زمان‌بر بودن اجرای هر فرآیند شبیه‌سازی و لزوم تکرارهای متعدد شبیه‌سازی به ازای هر ورودی مفروض. به همین منظور استفاده از متامدل‌ها^۱ در کنار رویکرد بهینه‌سازی از طریق شبیه‌سازی مرسوم است. متامدل فضای جواب پیچیده مسئله اصلی را به صورتی ساده‌تر مدل می‌کند. شبیه‌سازی، مسائل دنیای واقعی را در سطحی ساده‌تر و البته با تفاوت‌هایی مدل می‌کند. به عبارتی شبیه‌سازی قادر نیست دنیای واقعی را به همان صورتی که هست، مدل کند. در سطحی بالاتر، متامدل قرار دارد. متامدل، مدل شبیه‌سازی را بار دیگر مدل می‌کند؛ از پیچیدگی‌های آن می‌کاهد و در عین حال مدل را باز هم بیشتر از حالت واقعی دور می‌کند. این ارتباطات در شکل (۱) نمایش داده شده است.



شکل ۱: شماتیک ارتباط مدل‌های واقعی، شبیه‌سازی و متامدل

از انواع متامدل‌های مرسوم می‌توان به شبکه‌های عصبی و رگرسیونی اشاره کرد. اگر چه استفاده از متامدل‌ها در بسیاری از موارد کارگشا است، اما مشکلات متعددی نیز در بهره‌برداری از آنها وجود دارد. به طور مثال در استفاده از مدل رگرسیونی، ابتدا باید رابطه مدل رگرسیونی تعیین شود که خود مستلزم استفاده از روش سعی و خطا است و همواره با فضای اصلی جواب متفاوت خواهد بود. با فرض تعیین مدل مناسب که ما را به جواب بهینه همگرا کند، هنوز مشکل تعیین ضرایب مجهول مدل باقی است. در استفاده از سایر متامدل‌ها نیز این چنین مشکلاتی وجود دارد. گفتنی است که انتخاب خود متامدل نیز باید با توجه به شرایط مسئله تعیین شود.

الگوریتم بهینه‌سازی گروه ذرات^۱ کلاسیک

الگوریتم گروه ذرات در سال ۱۹۹۵ توسط کندی و ابرهارت [۳۷] ابداع شد. پایه و اساس الگوریتم بر استفاده هوشمندانه‌ای از فلسفه هوش جمعی بنا شده است. مفهوم ذره در این الگوریتم، به نقطه‌ای از فضای جواب اطلاق می‌شود. از حرکت جمعیتی^۹ از ذرات در فضای جواب، در نهایت همگرایی به جواب بهینه حاصل می‌شود. از نظر تعریف، به این دلیل که با جمعیتی از جواب‌ها سر و کار داریم، الگوریتم مبتنی بر جمعیت^{۱۰} محسوب می‌شود. جمعیت اولیه اغلب به طور تصادفی ایجاد می‌شود. ما نیز از همین رویکرد استفاده کرده‌ایم. هر ذره در فضای جواب، دارای بردار سرعت است و مکان بعدی خود را با توجه به آن تعیین می‌کند (رابطه ۱).

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

در رابطه بالا، t اندیس مربوط به زمان است و i اندیس مربوط به ذره. با به روز کردن بردار سرعت در هر مرحله جابه‌جایی‌های ذرات کامل می‌شود. قبل از هر چیز دو مفهوم زیر باید تعریف شوند.

- بهترین محلی^{۱۱}: مختصات نقطه‌ای که ذره قبلا در آنجا بوده و بهترین هزینه را از نظر بهینگی داشته است.
- بهترین جهانی^{۱۲}: مختصات نقطه‌ای از فضای جواب که تا کنون بهترین هزینه را از نظر بهینگی داشته است.

بردار سرعت ذرات با توجه به این دو مفهوم به‌روز می‌شود. هر ذره بردار، سرعت خود را با توجه به بردار سرعت قبلی خود، بهترین محلی و بهترین جهانی تعیین می‌کند. در واقع بردار سرعت در حالت جاری از برهم‌نهی بردارهای سرعت قبلی، برداری از مکان ذره به سمت بهترین محلی و برداری از مکان ذره به سمت بهترین جهانی محاسبه می‌شود.

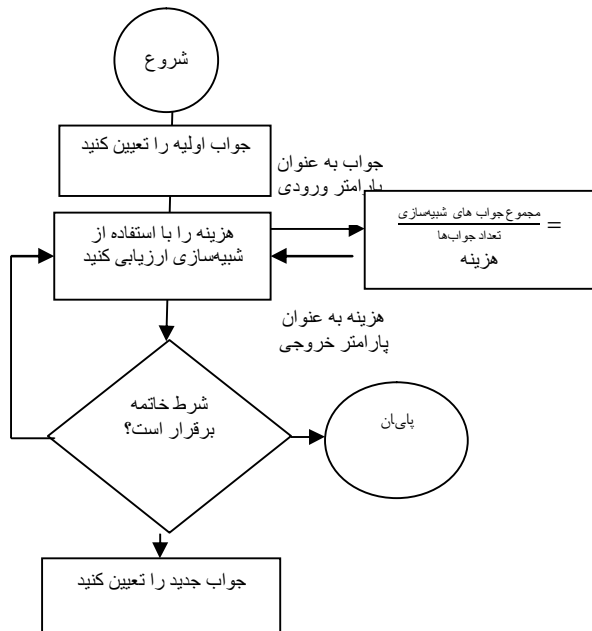
$$v_i(t + 1) = w * v_i(t) + c_1 rand(pbest_i - x_i) + c_2 rand(gbest - x_i), \quad (2)$$

که در آن pbest بهترین محلی و gbest بهترین جهانی است. ضرایب c_1 و c_2 نوعی تعادل را در بین پیروی از هوش فردی و پیروی از هوش اجتماعی به وجود می‌آورند.

شبه کد الگوریتم گروه ذرات

```

create initial population and velocities
while (not termination)
    Evaluate cost for each particle in group
using simulation;
    UpdateLocalBests();
    UpdateGlobalBest();
    for each particle
        R1 = uniform random number;
        R2 = uniform random number;
        V[I][t] = w*V[I][t-1] +
            C1*R1*(X_pbest[I][t-1]-X[I][t-1]) +
            C2*R2*(X_gbest[t-1]-X[I][t-1]);
        X[I][t] = X[I][t-1] + V[I][t];
    end for
    
```



شکل ۲: دیاگرام رویکرد بهینه‌سازی از طریق شبیه‌سازی در حالت استفاده مستقیم

رویکرد مستقیم

در رویکرد مستقیم، در مرحله تعیین مقادیر تابع هدف برای ذرات گروه با توجه به فضای احتمالی مسئله، ۲۰ تا ۱۰۰ بار شبیه‌سازی در مکان هر ذره انجام می‌گیرد و از نتایج آنها میانگین گرفته و به عنوان مقدار تابع هدف آن ذره ثبت می‌کنیم. با افزایش تعداد شبیه‌سازی‌های انجام گرفته در مکان هر ذره، با توجه به قانون حد مرکزی به مقدار واقعی تابع هدف در آن نقطه بیشتر نزدیک می‌شویم. شکل (۲)، دیاگرام رویکرد بهینه‌سازی از طریق شبیه‌سازی را نمایش می‌دهد.

بهینه‌سازی در حالت کلی با انتخاب یک یا مجموعه‌ای از جواب‌ها شروع می‌شود. جواب‌ها به عنوان ورودی به مادل شبیه‌سازی داده می‌شوند و هزینه یا مقدار تابع هدف به عنوان خروجی مادل شبیه‌سازی از آن گرفته می‌شود. ورودی شبیه‌سازی در واقع داده‌ای است که مدل شبیه‌سازی از روی آن پیکربندی می‌شود. مشکل عمده در استفاده از این رویکرد، زمان زیاد لازم برای پردازش آن است که از دو جهت ناشی می‌شود:

- اغلب اجرای خود شبیه‌سازی زمان‌بر است.
- لزوم شبیه‌سازی‌های متوالی برای نزدیک شدن به مقدار واقعی خروجی آن.

کار اصلی ما در این مقاله، تلاش برای حذف مورد دوم است. در ادامه رویکردهایی برای دستیابی به این مهم ارائه شده‌اند.

رویکرد میانگین‌گیری

در این روش به جای شبیه‌سازی‌های متوالی در یک نقطه برای رسیدن به مقدار تابع هدف، از میانگین‌گیری موضعی در آن منطقه استفاده می‌کنیم. به عبارتی هدف آن است که ذرات مقدار تابع خود را در فضای احتمالی با کمک از ذرات اطراف خود به دست آورند. برای همه ذرات یک بار شبیه‌سازی انجام می‌گیرد و نتایج ذخیره می‌شود. از آنجا که فقط یک مرتبه شبیه‌سازی انجام گرفته است، خطای جواب‌های به دست آمده زیاد خواهد بود. در مرحله بعد برای تصحیح این خطاها از رویکرد میانگین‌گیری به صورت زیر استفاده می‌شود.

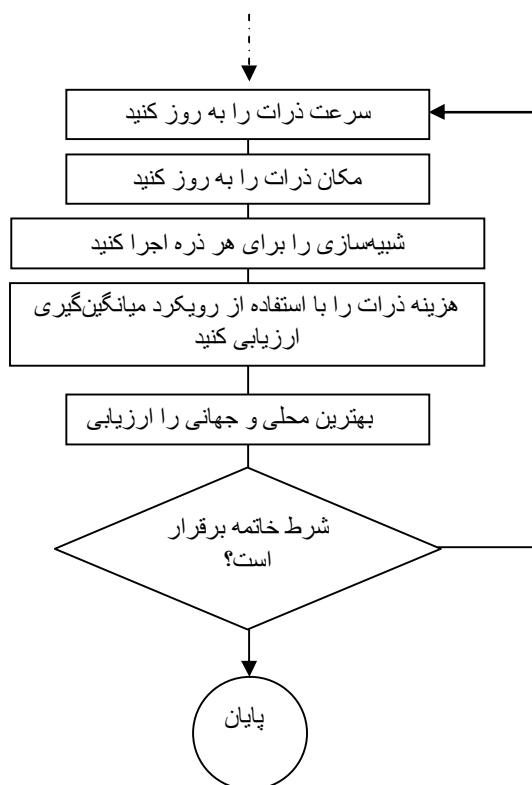
- همسایه‌های هر ذره را تعیین می‌کنیم (همسایه به ذره‌ای اطلاق می‌شود که فاصله فضایی آن از یک مقدار مشخص d_{max} کمتر باشد).

- از مقادیر حاصل از شبیه‌سازی مربوط به ذرات همسایه میانگین گرفته و عدد حاصله به عنوان مقدار تابع هدف آن ذره در نظر گرفته می‌شود.

از آنجا که مقدار تابع هدف در نقاط بسیار دور از نقطه مورد نظر در مقدار تابع هدف آن تأثیری ندارد و تغییرات وسیع تابع در این بازه بلند مانع از آن می‌شود که بتوان از آن استفاده کرد، باید محدوده‌ای برای ذره تعریف شود که فقط ذرات در آن محدوده در میانگین‌گیری شرکت خواهند داشت. این رابطه را می‌توان به شکل رابطه (۳) بیان کرد.

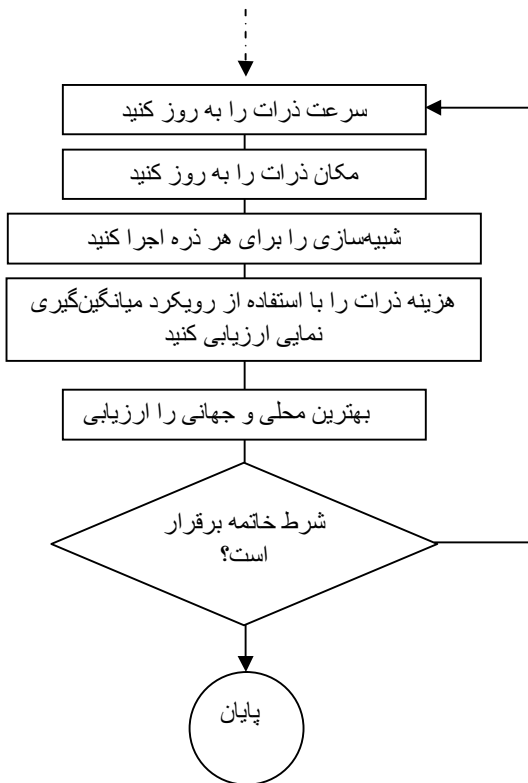
$$\bar{f}_i = \frac{1}{m} \sum_{j=(|X_i-X_j| \leq d_{max})} \hat{f}_j \quad (3)$$

که در آن d_{max} بیشترین فاصله‌ای است که ذرات در آن فاصله از ذره موردنظر، در میانگین‌گیری برای حصول مقدار تابع هدف شرکت می‌کنند. m تعداد کل ذراتی است که شرط فاصله در آنها صدق می‌کند. با مشخص شدن مقادیر تابع هدف برای همه ذرات گروه، بقیه فرایند همگرایی الگوریتم گروه ذرات همانند مدل کلاسیک آن ادامه می‌یابد. فلوجارت کل فرایند بهینه‌سازی در شکل (۳) آمده است.



شکل ۳: فلوجارت رویکرد میانگین‌گیری

این روش نیز همانند رویکرد میانگین گیری در مورد توابع هموار به مراتب بهتر از توابع ناهموار عمل می کند. برای به کارگیری آن، فقط به تعیین پارامتر k احتیاج است و می توان با تغییر آن در طول پیشرفت بهینه سازی بر انعطاف پذیری و کیفیت آن افزود.



شکل ۴: فلوجارت رویکرد میانگین گیری نمایی

استفاده از متامدل دینامیک

پایه و اساس این روش، در استفاده کلاسیک از متامدلها است. روش کار در استفاده از متامدل رگرسیونی، تعیین پایه های فضا (تعیین نقاطی از فضا که از روی آنها مدل رگرسیونی ساخته می شود) و سپس به دست آوردن نقطه بهینه در فضای متامدل ایجاد شده است. در رویکرد جدید، ما پایه های فضا را روی ذرات الگوریتم جستجو پیاده می کنیم. با این کار ما یک الگوریتم جستجو با یک متامدل دینامیکی را ترکیب کرده و از توانایی آنها به صورت توأم استفاده می کنیم. در این رویکرد از نقطه بهینه متامدل به عنوان بهترین عمومی الگوریتم گروه ذرات استفاده می شود. در هر گام ذرات در نقاط جدید قرار گرفته، سپس متامدل به روز می شود. مزایای استفاده از این سیستم در مقابل استفاده کلاسیک

می توان گفت این روش ساده قادر است محاسبات لازم و به دنبال آن زمان پردازش لازم را تا $1/20$ روش مستقیم کاهش دهد. با همگرایی ذرات به سمت نقطه بهینه و نزدیک تر شدن بیشتر ذرات به هم، مقادیر تابع هدف ذرات در هم تأثیر بیشتری خواهد داشت و کیفیت نتایج میانگین گیری افزایش می یابد. البته اگر $dmax$ را با پیشرفت جریان کار کاهش دهیم، به نتایج بهتری دست خواهیم یافت. چرا که دخیل کردن مقدار تابع هدف نقاط مجاور در مقدار تابع هدف نقطه، منطقی می نماید. اما اگر سطح تابع تغییرات خشنی داشته باشد، نمی توان به نتایج این روش امیدوار بود. این روش برای توابعی که هموار و نرم باشند، به عبارتی تغییرات ناگهانی شدید نداشته باشند، به خوبی عمل می کند.

رویکرد میانگین گیری نمایی

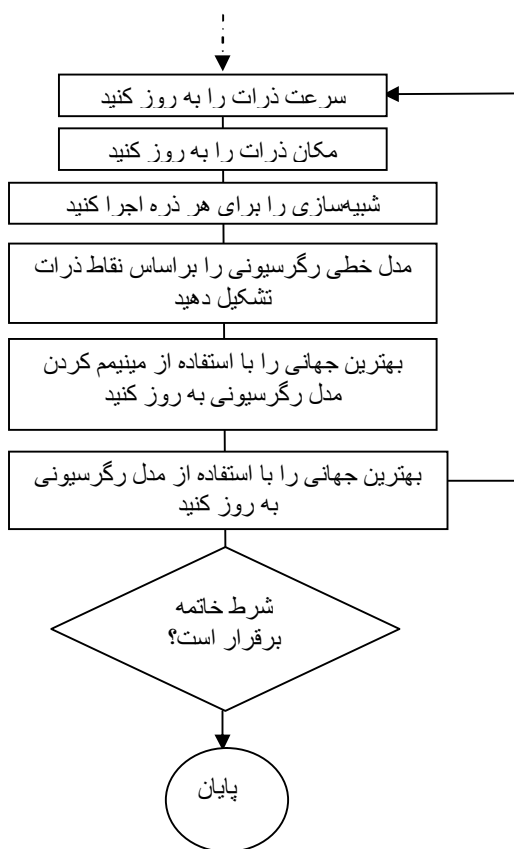
در این روش نیز میانگین گیری، جای شبیه سازی های متوالی را می گیرد. در مکان هر ذره یک بار شبیه سازی انجام گرفته و سپس:

$$\bar{f}_i = \hat{f}_{max} - \frac{1}{n} \sum_{j=1}^n (\hat{f}_{max} - \hat{f}_j) e^{-k|x_i - x_j|} \quad (4)$$

که در آن \hat{f}_{max} حداکثر مقدار تابع هدفی است که در مرحله شبیه سازی به دست آمده و n تعداد ذرات است. k ضریبی است که با توجه به مسئله باید تعیین شود. این ضریب در واقع میزان تأثیر فاصله در کاهش اثر مقدار تابع هدف را روی نقاط مجاور نشان می دهد. این رابطه را می توان به صورت زیر تحلیل کرد:

تعداد n نقاط جرمی روی یک صفحه قرار دارند. میزان جرم این نقاط با میزان تابع هدف در موقعیت آنها رابطه عکس دارد. به عبارت دیگر نقاطی که تابع هدف آنها کمتر باشد، جرم بیشتری نسبت به بقیه دارند. این نقاط جرمی در صفحه زیری شان فرو می روند و میزان فرورفتگی با مقدار جرم آنها رابطه مستقیم دارد. فلوجارت این رویکرد در شکل آمده است.

هدف محاسبه می‌شود. بقیه فرایند همانند روش کلاسیک الگوریتم گروه ذرات است.



شکل ۵: فلوچارت رویکرد متامدل دینامیک

سنجش کارایی

برای تحقیق کارایی رویکردهای معرفی شده در بخش قبل، چندین مسئله در ادامه آمده است. هر رویکرد روی مسائل مختلف اعمال و نتایج حاصله با نتایج حاصل از رویکرد مستقیم مقایسه شده است. مسئله‌های مطرح شده از نظر تعداد متغیرها، تعداد محدودیت‌ها و پیچیدگی کلی متفاوت بوده و زمینه مناسبی برای سنجش کارایی رویکردهای پیشنهادی را فراهم می‌کنند.

مسائل آزمونی

مسائل مطرح شده به صورت زیر هستند.

A. تابع سینوسی

$$\min f(x) = x_1 \sin(x_2) + x_2 \sin(x_1) \quad (۶)$$

$$-2\pi \leq x_1 \leq 2\pi$$

$$-2\pi \leq x_2 \leq 2\pi$$

آن این است که مشکل تعیین پایه‌های مناسب برای ساختن متامدل رفع می‌شود.

پایه‌های اولیه به طور تصادفی در فضای جستجو قرار داده شده و با همگرایی ذرات در هر مرحله پایه‌های فضا به هم نزدیک‌تر شده و فضای کمتری توسط متامدل پوشش داده می‌شود. این موضوع باعث می‌شود که به تدریج کیفیت متامدل بالاتر رفته و سطح انطباق متامدل با مدل واقعی افزایش یابد. این رویکرد، اگر چه پرخرج‌تر از روش‌های متعارف است، ولی می‌توان با تعیین مناسب تعداد اجراهای متوالی شبیه‌سازی و تعداد ذرات، هزینه آن را کاهش داد. با توجه به یکپارچگی ذرات در گروه که از متامدل رگرسیونی ناشی می‌شود، امکان کاهش تعداد اجراهای متوالی شبیه‌سازی در یک نقطه وجود دارد. اساس همگرایی ذرات در پیروی از نقطه بهینه عمومی است که به طور مستقیم از مدل رگرسیونی حاصل می‌شود.

در تعیین خود مدل رگرسیونی می‌توان از ساده‌ترین مدل یعنی مدل خطی، بدون در نظر گرفتن روابط متقابل و یا یک سطح بالاتر استفاده کرد. با توجه به طبیعت پویای متامدل و همگرایی ذرات، می‌توان انتظار داشت که در نهایت، مدل رگرسیونی برای پوشش سطح واقعی کفایت کند. فلوچارت مربوط به این رویکرد در شکل نشان داده شده است.

بعد از اینکه ذرات به طور تصادفی در فضای جستجو اختیار شد، برای هر یک از آنها یک بار شبیه‌سازی انجام می‌گیرد تا مقدار اولیه‌ای از تابع هدف در آن نقطه به دست آید. سپس با استفاده از این نقاط، مدل رگرسیونی زیر ایجاد می‌شود:

$$Y = \alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n \quad (۵)$$

ضرایب آلفا را می‌توان به راحتی با استفاده از نقاط موجود و روش کمینه کردن مجذور خطا محاسبه کرد. در مرحله بعد، بهترین جهانی و بهترین عمومی از مدل رگرسیونی استخراج می‌شوند. بهترین جهانی با کمینه مقدار مدل رگرسیونی مقایسه شده و در صورت برتری جایگزین می‌شود. برای به روز کردن بهترین‌های عمومی، مقادیر ذرات در مدل رگرسیونی قرار گرفته و مقدار تابع

B. تابع نامتقارن

$$\min f(x) = \sum_{i=1}^8 [2^{x_i-4} + (6 - x_i)] \quad (7)$$

$$0 \leq x_i \leq 10, \quad i = 1, 2, \dots, 8$$

برای هر یک از مسائل تعریف شده در بالا، چندین تابع خطای احتمالی تعریف شده است که نقش ماهیت احتمالی مسائل مطرح شده در رویکرد بهینه‌سازی از طریق شبیه‌سازی را بازی می‌کنند. این توابع خطا در جدول (۱) آمده‌اند.

جدول ۱: توابع خطا برای مسائل مختلف

		Error functions	
		I. small errors	II. large errors
problems	A	$\eta \sim N(0, 1)$	$\eta \sim N(0, 5)$
	B	$\eta \sim N(0, 7)$	$\eta \sim N(0, 35)$

نتایج استفاده از روش مستقیم در جدول (۲) آورده شده است. این نتایج از اجرای الگوریتم روی کامپیوتر با پردازنده ۲٫۶۶ GHz و حافظه RAM ۱ GB به دست آمده است. رویکرد مستقیم روی هر مسئله، با یکی از تابع خطای ۲۰ بار پیاده شده و در نهایت، نتایج در قالب بهترین جواب و میانگین جواب‌ها ارائه شده است. تعداد تکرار شبیه‌سازی در هر نقطه مفروض ۲۵ بار است که از میانگین آن به عنوان مقدار تابع هدف در آن نقطه استفاده شده است.

نتایج

هر یک از رویکردهای ارائه شده را روی مسائل تعریف شده با خطاهای مختلف پیاده کرده‌ایم. این کار برای هر یک از حالات مورد بررسی ۲۵ بار تکرار شده و نتایج در قالب میانگین و بهترین جواب حاصله ارائه شده است. توجه شود که در این رویکرد، برای محاسبه مقدار تابع هدف هر ذره، فقط یک بار شبیه‌سازی انجام می‌گیرد. در جدول (۳) نتایج حاصل از اجرای روش میانگین‌گیری روی مسائل آزمونی را داریم.

مقایسه نتایج این روش با روش مستقیم، حاکی از آن است که کیفیت جواب‌ها تا حدودی پایین آمده است، اما در عوض زمان اجرای مورد نیاز به حدود ۰٫۱ مقادیر قبلی

رسیده که کاهش کیفیت جواب‌ها را در عمل توجیه می‌کند.

نتایج حاصل از اجرای روش میانگین‌گیری نمایی روی مسائل آزمونی را در جدول داریم. نتایج حاکی از آن است که الگوریتم در مورد مسئله A موفقیت چندانی نداشته است، اما در مورد مسئله B در کل توانسته است کیفیت نتایج را ارتقا دهد. مسئله کاهش کیفیت جواب‌ها نسبت به روش مستقیم همچنان وجود دارد که زمان پردازش کم می‌تواند استفاده از آن را توجیه کند. در جدول (۴) نتایج حاصل از اجرای روش متامدل پویا روی مسائل آزمونی را داریم. نتایج به طرز چشم‌گیری بهبود یافته‌اند. البته باز هم در مقایسه، روش مستقیم از نتایج بهتری برخوردار است. زمان‌های اجرا در مقایسه با روش‌های میانگین‌گیری و میانگین‌گیری نمایی افزایش یافته که قابل پیش‌بینی بود.

نتیجه‌گیری

در این مقاله، الگوریتم دسته ذرات به گونه‌ای تغییر یافت که بتواند در عین حذف متامدل‌های کلاسیک، زمان دستیابی به جواب بهینه را نیز کاهش دهد. سه مشکل عمده در استفاده از متامدل‌ها وجود دارد. ۱. انتخاب آنها باید با توجه به شرایط مسئله انجام بگیرد و در نتیجه شاید متوسل به روش آزمایش و خطا شویم. ۲. پارامترهای مربوط به متامدل خود به مسئله وابسته است و باید به دقت انجام بگیرد. ۳. پیاده‌سازی آنها زمان‌بر است و ممکن است در نهایت جواب مناسبی هم تولید نشود. سه روش جدید: روش میانگین‌گیری، روش میانگین‌گیری نمایی و متامدل دینامیک ارائه شد و برای بررسی کارایی آنها، دو مسئله بهینه‌سازی مطرح و روش‌ها روی آنها پیاده شدند. اگرچه کیفیت نتایج به دست آمده از روش‌های ارائه شده از کیفیت جواب‌های حاصل از روش مستقیم کمتر بود، ولی زمان لازم برای رسیدن به جواب‌ها به مراتب کمتر بود که این موضوع، استفاده از این روش‌ها را توجیه می‌کند. تنها محدودیت استفاده از این روش‌ها این است که روی توابعی که تغییرات شدید و تیزی در دامنه خود دارند، قابل استفاده نیستند، که البته این مشکل در مورد استفاده از متامدل‌ها نیز وجود دارد. کیفیت و دقت جواب‌ها در این روش‌ها، متأثر از مقدار جمعیت الگوریتم گروه ذرات است که با افزایش آنها می‌توان به محدودیت اشاره شده

نیز غلبه کرد. مدل‌های ارائه شده در عین سادگی در فهم، داشته باشد. در کارهای آینده می‌توان روی سیستم یادگیری ذرات بیشتر کار کرد و در این راستا استفاده از روش‌ها و کار تجربی بیشتر، می‌تواند نتایج بسیار مثبتی در عمل نیز به سادگی قابل پیاده‌سازی هستند. توسعه این روش‌های آماری دقیق می‌تواند نتیجه‌بخش باشد.

جدول ۲: نتایج استفاده از روش مستقیم با ۲۵ بار تکرار برای میانگین‌گیری

problems		Error function I			Error function II		
		mean	best	Cpu t	mean	best	Cpu t
A	A	-9.59	-9.62	2.34	-9.41	-9.62	2.36
	B	24.71	23.66	2.70	31.48	25.40	2.82

جدول ۳: نتایج حاصل از اجرای روش میانگین‌گیری روی مسائل آزمونی

problems		Error function I			Error function II		
		mean	best	Cpu t	mean	best	Cpu t
A	A	-8.41	-9.57	0.18	-6.86	-9.61	0.14
	B	32.07	26.23	0.32	39.66	29.72	0.32

جدول ۴: نتایج حاصل از اجرای روش میانگین‌گیری نمایی روی مسائل آزمونی

problems		Error function I			Error function II		
		mean	best	Cpu t	mean	best	Cpu t
A	A	-8.65	-9.06	0.17	-7.62	-9.51	0.14
	B	31.84	25.15	0.26	37.21	28.36	0.35

جدول ۴: نتایج حاصل از اجرای روش متامدل دینامیک روی مسائل آزمونی

problems		Error function I			Error function II		
		mean	best	Cpu t	mean	best	Cpu t
A	A	-9.45	-9.60	0.21	-9.12	-9.45	0.23
	B	25.63	23.86	0.31	26.13	24.01	0.33

مراجع

- 1-Kuo, R.J., and Yang, C.Y. (2011). "Simulation optimization using particle swarm optimization algorithm with application to assembly line design." *Applied Soft Computing*, Vol. 11, No. 1, PP. 605-613.
- 2-Guo, Y., Liao, W., Cheng, X., and Liu, L. (2006). "SimOpt: A new simulation optimization system based virtual simulation for manufacturing system." *Simulation Modeling Practice and Theory*, Vol. 14, No. 5, PP. 577-585.
- 3-Azadeh, A., Faiz, Z.S., Asadzadeh, S.M., and Tavakkoli-Moghaddam R. (2011). "Simulation optimization using particle swarm optimization algorithm with application to assembly line design." *Applied Soft Computing*, Vol 11, No. 1, PP 605-613.
- 4-Wan, X., Pekny, J.F., and Reklaitis, G.V. (2005). "Simulation-based optimization with surrogate models application to supply chain management" *Computers and Chemical Engineering*, Vol. 29, No. 6, PP. 1317-1328.
- 5-Vaghefi, A., and Sarhangian, V. (2009). "Contribution of simulation to the optimization of inspection plans for multi-stage manufacturing systems." *Computers & Industrial Engineering*, Vol. 57, No. 4, PP. 1226-1234.

-
- 6- Memar Jafari M.R., Gatmiry Z. S., Khakzar Bafrouei, M. (2011). " A Discrete Simulation Analysis of Customer Order Supply System: A Case Study (Technical note)." *Journal of Industrial Engineering*, Vol. 45, No.1, PP. 95-102.
 - 7- Nazari L., Aghaie A. (2012). " A Framework for Bullwhip Measurement in Two and Three-echelon Supply Chains with more than One Product." *Journal of Industrial Engineering*, Vol. 46, No.1, PP. 105-117.
 - 8- Hassan Nayebi E., Kiyani B. (2010). " Combination of Monte Carlo Simulation and System Dynamics Modeling for Project Time Risk Analysis." *Journal of Industrial Engineering*, Vol. 44, No.2, PP. 169-180.
 - 9- Li, Y.F., Ng, S.H., Xie, M., Goh, T.N. (2010). "A systematic comparison of metamodeling techniques for simulation optimization in Decision Support Systems." *Applied Soft Computing*, Vol. 10, No. 4, PP. 1257-1273.
 - 10- Friedman, J.H. (1991). "Multivariate adaptive regression splines." *Annals Statistics*, PP. 1-67.
 - 11- Pilla V. L., Rosenberger, J.M., Chen, V., Engsuwan, N., and Siddappa, S. (2012). "A multivariate adaptive regression splines cutting plane approach for solving a two-stage stochastic programming fleet assignment model." *European Journal of Operational Research*, Vol. 216, No. 1, PP. 162-171.
 - 12- Vidoli, F. (2011). "Evaluating the water sector in Italy through a two stage method using the conditional robust nonparametric frontier and multivariate adaptive regression splines." *European Journal of Operational Research*, Vol 212, No.3, PP. 583-595.
 - 13- Durmaz, M., and Karslioglu, M.O. (2011). "Non-parametric regional VTEC modeling with Multivariate Adaptive Regression B-Splines." *Advance in space research*, Vol. 48, No. 9, PP. 1523-1530.
 - 14- De Andrés, J., Lorca, P., de Cos Juez F. J., and Sánchez-Lasheras, F. (2011). " Bankruptcy forecasting: A hybrid approach Fuzzy c-means clustering and Multivariate Adaptive Regression Spline (MARS)" *Expert Systems with Applications*, Vol. 38, No. 3, PP. 1866-1875.
 - 15- Van Beers, W.C., and Kleijnen, J.P. (2003). "Kriging for interpolation in random simulation." *Journal of the Operational Research Society*, Vol. 54, No. 3, PP. 255-262.
 - 16- Kleijnen, J.P., and Van Beers, W.C. (2004). "Application-driven sequential designs for simulation experiments: Kriging metamodeling." *Journal of the Operational Research Society*, Vol. 55, No. 8, PP. 876-883.
 - 17- Kleijnen, J.P., and Van Beers, W.C. (2005). "Robustness of Kriging when interpolating in random simulation with heterogeneous variances: Some experiments." *European Journal of Operational Research*, Vol. 165, No. 3, PP. 826-834.
 - 18- Meghabghab, G., and Kandel, A. (2004). "Stochastic simulations of Web search engines: RBF versus second-order regression models." *Information Sciences*, Vol. 159, No. 1, PP. 1-28.
 - 19- Regis, R.G., and Shoemaker, C.A. (2007). "Parallel radial basis function methods for the global optimization of expensive functions" *European Journal of Operational Research*, Vol. 182, No. 2, PP. 514-535.
 - 20- Regis, R.G. (2011). "Stochastic radial basis function algorithms for large-scale optimization involving expensive black-box objective and constraint functions." *Computers & Operation Research*. Vol. 38, No. 5, PP. 837-853.
 - 21- Luo, Z., Tong, L. and Kang, Z. (2009). "A level set method for structural shape and topology optimization uses radial basis functions " *Computers and structures*, Vol. 87, No. 7, PP. 425-434.
-

- 22- Korürek, M. and Doğan, B. (2010). "ECG beat classification using particle swarm optimization and radial basis function neural network." *Expert system with Application.*, Vol. 37, No. 12, PP. 7563-7569.
- 23- Fonseca, D.J., Navarrese, D., Moynihan, G.P. (2003). "Simulation metamodeling through artificial neural networks." *Engineering Applications of Artificial Intelligence*, Vol. 16, No. 3, PP. 177-183.
- 24- Fonseca, D.J., Navarrese, D.. (2002). "Artificial neural networks for job shop simulation." *Advanced Engineering Informatic.*, Vol. 16, No. 4, PP. 241-246.
- 25- Mirtalaie M.S., Azadeh M.A., Saberi M., Ashjari A. (2012). " A Trust-based Credit Scoring Model Using Neural Network." *Journal of Endustrial Engineering*, Vol. 46, No. 1, PP. 91-104.
- 26- Khashei M., Bijari M. (2010). " Gold price forecasting using hybrid artificial neural networks with fuzzy regression model." *Journal of Endustrial Engineering*, Vol. 44, No. 1.
- 27- Clarke, S.M., Griebisch, J.H., and Simpson, T.W. (2005). "Analysis of support vector regression for approximation of complex engineering analyses." *Journal of Mechanical Design*, Vol. 127, No. 6, PP. 1077-1078.
- 28- Eitrich, T., and Lang, B. (2005). "Efficient optimization of support vector machine learning parameters for unbalanced datadets." *Journal of Computational and Applied Mathematics.*, Vol. 196, No. 2, PP. 425-436.
- 29- Haldun Aytuğ and Serpil Sayin (2009). "Using support vector machine to learn the efficient set in multiple objective discrete optimization." *European journal of operation research*, Vol. 193, No.2 ,PP. 510-519.
- 30- Norinder, U. (2003). "Support vector machine in drug design: application to drug transport processes and QSAR using simplex optimisations and variables selection." *Neurocomputing*, Vol. 55, No. 1, PP. 337-346.
- 31- Aydin, I, karakose, M., and Akin, E. (2011). "A multi-objective artificial immune algorithm for paramter optimization in support vector machine." *Applied soft computing*, Vol. 11, No. 1, PP. 120-129.
- 32- Balabin, R. M., and Lomakina, E. I. (2011). "Support vector machine regression (SVR/LS-SVM)—an alternative to neural networks (ANN) for analytical chemistry? Comparison of nonlinear methods on near infrared (NIR) spectroscopy data." *Analyst*, vol. 136, No. 8, 1703-1712.
- 33- Wang, G.G. , Shan, S. (2007). "Review of metamodeling techniques in support of engineering design optimization." *Journal of Mechanical Design*, Vol. 129, No. 4, PP. 370-380.
- 34- Simpson, T.W., Mauery, T.M., Korte, J.J., and Mistree, F. (1988). "Comparison of response surface and kriging models for multidisciplinary design optimization." *on Multidisciplinary Analysis & Optimization.*, Vol. 98, No. 7, PP. 1-16.
- 35- Jin, R., Chen, W., and Simpson, T.W. (2001). "Comparative studies of metamodeling techniques under multiple modelling criteria." *Structural and Multidisciplinary Optimization*, Vol. 23, No. 1, PP. 1-13.
- 36- Fanga, H., Rais-Rohanib, M., Liuc, Z., Horstemeyera, M.F. (2005). "A comparative study of metamodeling methods for multiobjective crashworthiness optimization." *Computers and Structures*, Vol. 83, No. 25, PP. 2121-2136.
- 37- Kennedy, J., Eberhart, R. (1995). "Particle swarm optimization." *In Proceeding of IEEE International Conference on Neural Networks*, Vol. 4, No. 2, PP. 1942-1948.

واژه‌های انگلیسی به ترتیب استفاده در متن

- 1 - Meta-model
- 2 - Artificial Neural Network
- 3 - Radial Basis Function

-
- 4 - Support Vector Machine
 - 5 - Kriging
 - 6 - Multivariate Adaptive Regression Splines
 - 7 - Swarm Intelligence
 - 8- Particle Swarm Optimization
 - 9- Population
 - 10- Population Base
 - 11- Local Best
 - 12- Global Best
-