

ترکیب الگوریتم بهینه‌سازی کاوش باکتری با برنامه ریزی مربعی متوالی برای پخش بار اقتصادی دینامیکی با در نظر گرفتن اثرات نقطه شیر

احسان افزalan^۱، محمود جورابیان^۲

۱- کارشناس ارشد مهندسی برق قدرت

Ehsan.afzalan@gmail.com

۲- استاد، دانشکده فنی و مهندسی - دانشگاه شهید چمران - اهواز - ایران

mjoorabian@scu.ac.ir

چکیده: پخش بار اقتصادی دینامیکی با برنامه‌ریزی خروجی آنلاین ژنراتورها با تقاضاهای بار پیش‌بینی شده در بازه زمانی مشخص به منظور بیشترین عملکرد اقتصادی سیستم قدرت الکتریکی سر و کار دارد. این مقاله روش ترکیبی الگوریتم بهینه‌سازی کاوش باکتری با برنامه‌ریزی مربعی متوالی برای حل مسأله پخش بار اقتصادی واحدهای تولیدی با در نظر گرفتن اثرات نقطه شیر را پیشنهاد می‌کند. این روش ترکیبی، بهینه‌سازی کاوش باکتری را به عنوان جستجوی سطح پایه که می‌تواند جهت‌یابی مناسبی به ناحیه بهینه داشته باشد و برنامه‌ریزی مربعی متوالی را به عنوان روال جستجوی محلی که برای تنظیم دقیق آن ناحیه برای رسیدن به حل نهایی به کار می‌رود، به کار می‌گیرد. نتایج عددی یک سیستم ۱۰ واحدی برای نشان دادن عملکرد و کارایی روش پیشنهادی ارائه شده‌اند. نتایج به دست آمده از روش پیشنهادی با نتایج به دست آمده از ترکیب بهینه‌سازی جستجوی ذرات و برنامه‌ریزی مربعی متوالی و ترکیب برنامه‌ریزی تکاملی و برنامه‌ریزی مربعی متوالی مقایسه شده‌اند.

واژه‌های کلیدی: روش بهینه‌سازی کاوش باکتری (BFOA)، پخش بار اقتصادی دینامیکی، برنامه ریزی مربعی متوالی (SQP).

۱- مقدمه

گرایان‌های دما و فشار درون بویلر و توربین را در حدود ایمن نگاه دارند تا از کاهش عمر تجهیزات جلوگیری نمایند. این قید مکانیکی حدی روی نرخ افزایش یا کاهش توان الکتریکی خروجی تحمیل می‌نماید. این قید حد نرخ شیب نامیده می‌شود که مساله DED را از SED متمایز می‌سازد. بنابراین، در DED تصمیم پخش بار در یک بازه زمانی بر تصمیم‌ها در بازه‌های زمانی بعدی تأثیر می‌گذارد. پخش بار اقتصادی دینامیکی دقیق‌ترین فرمول‌بندی مسأله پخش بار اقتصادی است، ولی به دلیل ابعاد بالای خود، سخت‌ترین مساله برای حل است. رقابت رو به افزایشی مابین بازارهای تولید عمده ایجاد شده است و نیاز است که بار هزینه افزایشی تحمیل شده بر روی عملکرد سیستم ناشی از حدود نرخ شیب ژنراتورها درک شود.

پخش بار اقتصادی استاتیک^۱ (SED) تقاضای بار را برای بازه زمانی مشخص مابین واحدهای تولیدی در مدار به صورت اقتصادی اختصاص می‌دهد و در عین حال، قیود مختلف را ارضا می‌کند. پخش بار اقتصادی دینامیک^۲ (DED) که گسترشی از پخش بار اقتصادی استاتیک است، سهم بهینه تقاضای بار متغیر با زمان را مابین واحدهای در مدار تعیین می‌کند. اپراتورهای نیروگاه‌ها سعی می‌کنند

^۱ تاریخ ارسال مقاله: ۱۳۹۱/۱۰/۱۶

تاریخ پذیرش مقاله: ۱۳۹۲/۰۹/۲۰

نام نویسنده مسؤول: محمود جورابیان

نشانی نویسنده مسؤول: ایران - اهواز - دانشگاه شهید چمران

اهواز - دانشکده فنی مهندسی.

از زمان معرفی DED چندین روش کلاسیک [۱-۶]

مشکل همگرایی نابهنگام را حل نموده و قابلیت جستجو را افزایش می‌دهند. این الگوریتم ساده و مقاوم بوده، قادر به حل مسائل ترکیبی مشکل است.

روش‌های ترکیبی [۱۸-۲۰] که روش‌های احتمالاتی و قطعی را ترکیب می‌کنند، برای حل مسائل DED بسیار جذاب هستند. در روش‌های ترکیبی، روش احتمالاتی به عنوان جستجوی سطح پایه به کار می‌رود که جهت‌یابی مناسبی به ناحیه بهینه کلی فراهم می‌کند و روش قطعی برای تنظیم دقیق آن ناحیه به منظور رسیدن به حل نهایی به کار می‌رود.

در این مقاله، روشی ترکیبی که الگوریتم مبتنی بر کاوش باکتری (BFOA) و برنامه‌ریزی مربعی متوالی^۱ (SQP) را ترکیب می‌کند، برای حل مسئله DED پیشنهاد شده است. هدف از ترکیب دو الگوریتم بهینه‌سازی کاوش باکتری و برنامه‌ریزی مربعی متوالی، یافتن روشی است که علاوه بر داشتن مزیت‌های دو الگوریتم فوق، عیب و نقص‌های این دو الگوریتم را نداشته باشد. قدرت بالا در جستجو و وابستگی زیاد عملکرد به انتخاب اولیه نقاط از مزایا و معایب الگوریتم برنامه‌ریزی مربعی متوالی است. همچنین، از مزایا و معایب الگوریتم بهینه‌سازی کاوش باکتری می‌توان به یافتن بهترین جواب و سرعت پایین در رسیدن به همگرایی اشاره کرد. بنابراین، با ترکیب این دو الگوریتم، می‌توان به روش هوشمندی دست یافت که علاوه بر عدم وابستگی به انتخاب اولیه نقاط، دارای قدرت و سرعت بالایی در همگرایی است. روش ترکیبی پیشنهادی از این قابلیت BFOA استفاده می‌کند که می‌تواند حتی زمانی که مسئله دارای تعداد زیادی حل‌های بهینه محلی در شروع است، حل خوبی فراهم کند، و روش SQP که قابلیت جستجوی محلی دارد، برای یافتن حل نهایی به کار می‌رود. برای نشان دادن کارایی روش ترکیبی پیشنهادی یک سیستم تست ۱۰ واحدی با تابع هزینه سوخت ناهموار در این مقاله استفاده شده است. نتایج روش ترکیبی BFOA-SQP پیشنهادی با

برای حل این مساله به کار بسته شده‌اند. با وجود این، ممکن است همه این روش‌ها قادر به پیدا کردن حل بهینه نباشند و اغلب در یک حل بهینه محلی گیر می‌کنند. روش‌های کلاسیک مبتنی بر حسابان، مساله DED را به تابع هزینه محدب ارجاع می‌دهند، ولی در واقعیت توربین-های بخار بزرگ دارای تعدادی درجه پذیرش بخار هستند که به سهم غیر محدب در تابع هزینه سوخت واحدهای تولیدی منجر می‌شوند. برنامه‌ریزی دینامیکی^۳ (DP) می‌تواند چنین مسائلی را حل نماید، ولی از مشکل ابعاد بزرگ مسأله رنج می‌برد.

اخیراً الگوریتم‌های جستجوی تصادفی [۷-۱۱] از جمله: باز پخت شبیه‌سازی شده^۴ (SA)، الگوریتم ژنتیک^۵ (GA)، برنامه‌ریزی تکاملی^۶ (EP)، بهینه‌سازی جستجوی ذرات^۷ (PSO) و تکامل دیفرانسیلی^۸ (DE) به دلیل توانایی خود برای یافتن حل نزدیک به پاسخ بهینه کلی مساله بهینه‌سازی غیر محدب، به صورت موفقیت آمیزی برای حل مسائل بهینه‌سازی سیستم‌های قدرت استفاده شده‌اند. این روش‌ها از قواعد احتمالی استفاده نموده، امکان بزرگی برای کاوش آزادانه فضای جستجو دارند. این روش‌ها همواره حل بهینه کلی را فراهم نمی‌کنند، ولی اغلب حلی سریع و معقول فراهم می‌آورند.

ازدحام اطلاعاتی [۱۲-۱۴] شاخه‌ای از الگوریتم‌های الهام گرفته از طبیعت است که روی عملکرد کلونی موجودات تمرکز می‌کند تا برخی الگوریتم‌های فرا اکتشافی را ایجاد نماید. الگوریتم بهینه‌سازی مبتنی بر کاوش باکتری^۹ (BFOA) [۱۵] عضو جدیدی از ازدحام اطلاعاتی است و از رفتار کاوشی اجتماعی باکتری *Escherichia coli* تقلید می‌کند. روش BFOA به عنوان راه حلی به مشکلات و معایب ذکر شده پیشنهاد شده است. همچنین، به دلیل تکنیک‌های پراکندگی و حذف منحصر به فرد خود می‌تواند نواحی مطلوب را هنگامی که جمعیت مورد بررسی کوچک است، پیدا کند. این قابلیت‌های منحصر به فرد الگوریتم‌ها

$$P_i^{\min} \leq P_{it} \leq P_i^{\max}, i \in N, t \in T \quad (4)$$

که در آن P_i^{\max} بیشینه توان تولیدی واحد i است.

(ج) حدود نرخ شیب ژنراتورها

$$\begin{aligned} P_{it} - P_{i(t-1)} &\leq UR_i, i \in N, t \in T \\ P_{i(t-1)} - P_{it} &\leq DR_i, i \in N, t \in T \end{aligned} \quad (5)$$

که در این روابط، UR_i و DR_i به ترتیب، حداکثر مقدار کاهش تولید و حد افزایش تولید نسبت به تولید ساعت قبل نیروگاه i ام است. همچنین، $P_{i(t-1)}$ مقدار توان تولیدی نیروگاه i ام در ساعت قبل است.

(د) تلفات سیستم شبکه

تلفات خط انتقال به صورت تابعی از توان حقیقی و ماتریس ضرایب B بیان می‌شوند [۲۱].

$$P_L = \sum_{i=1}^N \sum_{j=1}^N p_i B_{ij} p_j + \sum_{i=1}^N B_{i0} p_i + B_{00} \quad (6)$$

که در آن پارامترهای B_{ij} ضرایب اتلاف نامیده می‌شوند. عبارت (۶) را که به عنوان فرمول تلفات Korn شناخته می‌شود [۲۲] می‌توان به صورت برداری زیر بیان کرد [۲۳].

$$P_L = P^T B P + B_0^T p + B_{00} \quad (7)$$

که در آن B ماتریس متقارن ضرایب اتلاف، B_0 بردار ستونی ضرایب اتلاف و B_{00} ثابت اسکالر ضریب اتلاف است.

۳- مروری مختصر بر روش بهینه‌سازی

کاوش باکتری

بقای گونه‌ها در هر فرآیند تکاملی طبیعی وابسته به معیار متناسب بودن آنها بوده که خود مبتنی بر رفتار حرکتی و جستجوی غذای آنهاست. قانون تکامل تدریجی از گونه‌هایی که قابلیت جستجوی غذای بهتری دارند، حمایت نموده، آنهایی را که قابلیت جستجوی ضعیفی دارند، حذف کرده یا تغییر شکل می‌دهد. ژن‌های گونه‌های قوی‌تر به دلیل توانایی تولید مثل گونه‌های حتی بهتر در نسل‌های بعدی در زنجیره تکامل منتشر می‌شوند. بنابراین، درک و مدل‌سازی صحیح رفتار کاوشی در هریک از گونه‌های

نتایج به دست آمده از ترکیب بهینه‌سازی جستجوی ذرات و برنامه‌ریزی مربعی متوالی (PSO-SQP) و ترکیب برنامه‌ریزی تکاملی و برنامه‌ریزی مربعی متوالی (EP-SQP) مقایسه شده‌اند.

۲- فرمول‌بندی مسأله

به صورت نرمال مسأله DED هزینه کل تولیدی زیر را برای واحدهای در مدار کمینه می‌نماید:

$$F = \sum_{t=1}^T \sum_{i=1}^N F_{it}(P_{it}) = a_i + b_i p_{it} + c_i p_{it}^2 \quad (1)$$

که در آن $F_{it}(P_{it})$ تابع هزینه سوخت واحد i ، P_{it} توان تولیدی واحد i ، a_i ، b_i و c_i ضرایب هزینه سوخت واحد i هستند.

فرآیند باز شدن دریچه بخار در توربین‌هایی که چندین دریچه بخار دارند، نمودار سرعت گرمایی توربین را موجدار می‌کند و همچنین، موجب منفصل و غیرمحدب شدن تابع هدف می‌شود. برای مدل کردن دقیق اثرات دریچه بخار، یک تابع سینوسی به تابع هزینه اضافه می‌شود. بنابراین، تابع هزینه سوخت هر واحد با در نظر گرفتن اثر نقطه شیر [۸] می‌تواند به صورت زیر بیان شود:

$$F_{it}(p_{it}) = a_i + b_i p_{it} + c_i p_{it}^2 + \left| d_i \cdot \sin(e_i (p_i^{\min} - p_{it})) \right| \quad (2)$$

که در آن d_i و e_i ضرایب ثابت هزینه سوخت واحد i با در نظر گرفتن اثرات نقطه شیر بوده و P_i^{\min} مینیمم توان تولیدی واحد i است.

به شرط قیود تساوی و نامساوی زیر برای t امین فاصله زمانی در افق برنامه‌ریزی شده:

الف) تعادل توان اکتیو

$$\sum_{i=1}^N P_{it} - P_{Dt} - P_L = 0, t \in T \quad (3)$$

که در آن P_L کل تلفات توان حقیقی سیستم و P_D تقاضای بار کل سیستم است.

ب) حدود عملکردی توان اکتیو

مجموعه اولیه باکتری‌ها پس از تکامل تدریجی از طریق چندین مرحله Chemotactic به مرحله تولید مثل می‌رسند. در اینجا بهترین مجموعه باکتری‌ها به دو قسمت تقسیم می‌شوند: نیمه سالم‌تر جایگزین نیمه دیگر که به علت قابلیت‌های کاوشی ضعیف‌تر خود حذف می‌گردند، خواهند شد. این امر موجب حفظ جمعیت باکتری‌ها در مقداری ثابت در طول فرایند تکامل تدریجی می‌شود.

۳-۴- حذف و پراکندگی

در فرآیند تکامل تدریجی ممکن است رخدادی ناگهانی و پیش‌بینی نشده روی دهد که می‌تواند فرایند آرام تکامل تدریجی را به شدت تغییر داده، موجب حذف مجموعه باکتری‌ها و یا پراکندگی آنها به محیطی جدید شود. جالب اینجاست که این رخداد ناشناخته ممکن است به جای ایجاد اغتشاش در رشد Chemotaxis معمول مجموعه باکتری‌ها، موجب قرار داده شدن مجموعه جدیدی از باکتری‌های نزدیکتر به محل غذا گردد. از دیدگاهی وسیع، حذف و پراکندگی اجزایی از رفتار حرکتی طولانی سطح جمعیتی هستند. در کاربرد آن به بهینه‌سازی، این مرحله به کاهش رفتار ایستایی که غالباً در چنین الگوریتم‌های جستجوی موازی دیده می‌شوند، کمک می‌نماید. تفصیلات روابط ریاضی همچنین جنبه تئوری این مفهوم جدید در مراجع [۱۶، ۱۷] ارائه شده است.

۴- برنامه‌ریزی مربعی متوالی

برنامه‌ریزی مربعی متوالی (SQP) [۲۴] به صورت گسترده‌ای برای حل مسایل بهینه‌سازی عملی به کار می‌رود. این روش از هر روش برنامه‌ریزی غیر خطی دیگری از لحاظ کارایی، دقت و درصد حل‌های موفق بهتر است. این روش به میزان زیادی از روش نیوتن دقیقاً همان گونه که برای بهینه‌سازی نا مقید به کار می‌رود، برای بهینه‌سازی مقید تقلید می‌نماید.

تکاملی، به امکان اعمال آن در هر الگوریتم بهینه‌سازی سیستم غیر خطی منجر می‌شود. استراتژی کاوشی باکتری Escherichia coli در روده انسان را می‌توان با چهار فرآیند Chemotaxis، ازدحام، تولید مثل و حذف-پراکندگی توضیح داد [۱۵، ۱۶].

۳-۱- Chemotaxis

مشخصه حرکت باکتری در جستجوی غذا می‌تواند به دو روش تعریف شود: شنا کردن یا چرخش، که به مجموعه این دو حرکت Chemotaxis می‌گویند. گفته می‌شود باکتری "شنا" می‌کند؛ هرگاه در جهتی از پیش تعیین شده حرکت نماید و گفته می‌شود "چرخش" می‌کند؛ هرگاه در جهتی کاملاً متفاوت حرکت کند. از نظر ریاضی، چرخش هر باکتری می‌تواند با طول واحد حرکت تصادفی $\rho(j)$ ضرب در طول پله آن باکتری $C(i)$ نمایش داده شود. در حالت شنا، این طول تصادفی از پیش تعیین شده است.

۳-۲- ازدحام

برای آن که باکتری‌ها به غنی‌ترین موقعیت غذایی خود برسند، مطلوب آن است که آن باکتری که دارای بهترین موقعیت است، سعی کند تا نقطه‌ای از زمان در دوره جستجو، باکتری‌های دیگر را به خود جذب نماید؛ به نحوی که همه با هم سریع‌تر به موقعیت مطلوب برسند. بدین منظور، تابع جریمه‌ای بر مبنای فواصل نسبی هر باکتری از مناسب‌ترین باکتری در آن مدت جستجو به تابع هزینه اصلی افزوده می‌شود. در نهایت، وقتی همه باکتری‌ها در نقطه جواب با هم ترکیب شده‌اند، این تابع جریمه برابر با صفر می‌شود. تأثیر ازدحام آن است که باکتری‌ها را مجبور می‌کند در گروه‌هایی اجتماع نموده و با چگالی باکتری بالا به صورت الگوهای هم‌مرکز حرکت نمایند.

۳-۳- تولید مثل

برای هر تکرار از زیر مساله QP جهت d_k با استفاده از تابع هدف محاسبه می‌شود. حل به دست آمده تکرار جدیدی به صورت زیر را تشکیل می‌دهد:

$$P_{k+1} = P_k + \alpha_k d_k \quad (15)$$

مقدار طول گام α_k به شکل زیر برای ایجاد کاهش قابل ملاحظه در تابع شایستگی لاگرانژین تعمیم داده شده تعیین می‌شود:

$$L_A(P, \lambda, \rho) = F(P) - \lambda^T g(P) + \frac{\rho}{2} g(P)^T g(P) \quad (16)$$

که ρ یک اسکالر نامنفی است. این روال تا زمانی که مقدار S_k به مقداری قابل تحمل برسد، تکرار می‌شود.

۵- پخش بار اقتصادی دینامیکی مبتنی

بر BFOA-SQP

فلوچارت این روش در شکل (۱) نمایش داده شده است. در این فلوچارت در ابتدا در الگوریتم BFOA اجرای حلقه‌های دفع و پراکندگی، تولیدمثل و حرکت به سمت ماده غذایی آغاز می‌شود. سپس اعمال شنا و غلتیدن انجام می‌گردد. در ادامه الگوریتم SQP فراخوانی می‌شود و بهترین مقدار به دست آمده از الگوریتم BFOA با جواب به دست آمده از الگوریتم SQP مقایسه می‌شوند. اگر مقدار به دست آمده از الگوریتم SQP از مقدار به دست آمده از الگوریتم BFOA کمتر باشد، مقدار به دست آمده از الگوریتم SQP به عنوان بهترین مقدار ذخیره می‌گردد. پس از آن، در صورتی که حلقه حرکت به سمت ماده غذایی تمام شده باشد، وارد حلقه تولیدمثل می‌شود؛ در غیر این صورت، حلقه حرکت به سمت ماده غذایی ادامه پیدا می‌کند. پس از اتمام حلقه‌های حرکت به سمت ماده غذایی و تولیدمثل، حلقه دفع و پراکندگی اجرا می‌شود که آخرین حلقه از الگوریتم ترکیبی است.

از آنجا که تابع هدفی که قرار است کمینه شود نامحدب است، SQP نیازمند یک مینیمم محلی برای حل اولیه است. در این مقاله از SQP به عنوان بهینه‌کننده محلی برای تنظیم دقیق ناحیه بهتر کشف شده توسط BFOA استفاده می‌شود. در اینجا فرمول‌بندی SQP از مرجع [۱۸] برداشت شده است.

برای هر تکرار یک QP حل می‌شود تا جهت جستجو برای به روز سازی متغیرهای کنترلی به دست آید. مسأله QP را می‌توان به شکل زیر توضیح داد:

مقدار زیر را کمینه کنید:

$$\nabla F(P_k)^T d_k + \frac{1}{2} d_k^T H_k d_k \quad (8)$$

به شرط:

$$g_i(P_k) + [\nabla g(P_k)]^T d_k = 0, i = 1, \dots, m_e \quad (9)$$

$$g_i(P_k) + [\nabla g(P_k)]^T d_k \leq 0, i = m_e + 1, \dots, m \quad (10)$$

که H_k ماتریس هسیان تابع لاگرانژین در k امین تکرار، d_k جهت جستجو در k امین تکرار، P_k بردار توان حقیقی در k امین تکرار $g(P_k)$ قیود از (۳) تا (۵)، m_e تعداد قیود تساوی و m تعداد قیود است.

$$L(P, \lambda) = F(P) + g(P)^T \lambda \quad (11)$$

که λ بردار ضرب کننده لاگرانژین است.

H_k با استفاده از فرمول شبه- نیوتن داده شده از طریق رابطه زیر محاسبه می‌شود:

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T S_k} - \frac{H_k^T S_k^T S_k H_k}{S_k^T H_k S_k} \quad (12)$$

که:

$$S_k = P_{k+1} - P_k \quad (13)$$

$$q_k = \nabla L(P_{k+1}, \lambda_{k+1}) - \nabla L(P_k, \lambda_{k+1}) \quad (14)$$

تکنیک الگوریتم پیشنهادی به صورت زیر است:

گام (۱) مقداردهی اولیه پارامترهای زیر:

P : بعد فضای جستجو.

S : تعداد باکتری‌ها در جمعیت.

N_c : تعداد گام‌های chemotactic.

N_s : طول شنا وقتی روی شیب (مماس) باشد.

N_{re} : عداد گام‌های تولید مثل.

N_{ed} : تعداد وقایع حذف / پراکندگی.

p_{ed} : احتمال این که هر باکتری حذف یا پراکنده شود.

$C(i, j)|_{j=1}$: واحد اولیه طول حرکت.

$C(N_c)$: واحد طول حرکت در پایان گام‌های

chemotactic ($j=N_c$).

θ^i : مکان تصادفی اولیه هر باکتری.

گام (۲) حلقه حذف / پراکندگی، $l = l + 1$.

گام (۳) حلقه تولید مثل، $k = k + 1$.

گام (۴) حلقه chemotaxis، $j = j + 1$.

به ازای $i = 1, 2, \dots, S$ گام chemotactic را برای

هر باکتری به صورت زیر اجرا کن:

• تابع هزینه $J(i, j, k, l)$ را با استفاده

از (۱۷) و (۱۹) برآورد کن.

تابع هزینه هر باکتری در جمعیت متأثر است از نوع

ایجاد ازدحام اجرا شده توسط ارتباط سلول به سلول آزاد

شده توسط باکتری‌ها برای شکل دادن به الگوهای ازدحامی

عملیات هجوم آوردن به صورت زیر بیان می‌شود:

$$J_{cc}(\theta, P(j, k, l)) = \sum_{i=1}^s j_{cc}^i(\theta, \theta^i(j, k, l)) = \sum_{i=1}^s [-d_{attract} \exp(-\omega_{attract} \sum_{m=1}^p (\theta_m - \theta_m^i)^2)] + \sum_{i=1}^s [h_{repellant} \exp(-\omega_{repellant} \sum_{m=1}^p (\theta_m - \theta_m^i)^2)] \quad i=1, \dots, S \quad (17)$$

_____ در آن $d_{attract}$ و $w_{attract}$ و $h_{repellant}$

و $w_{repellant}$ ضرایب نمایانگر مشخصه‌های سیگنال‌های

جذبی و دفعی آزاد شده توسط سلول بوده و θ_m^i مولفه m

ام باکتری i ام است. $P(j, k, l)$ مکان هر عضو جمعیت S

باکتری بوده و به این صورت تعریف می‌شود:

$$P(j, k, l) = \{\theta^i(j, k, l) | i = 1, 2, \dots, s\} \quad (18)$$

که S اندازه جمعیت باکتری‌هاست.

تابع (۱۷) که نمایانگر تأثیر ارتباط سلول به سلول است

با تابع هزینه جمع می‌شود:

$$J(i, j, k, l) + J_{cc}(\theta, P) \quad (19)$$

• قرار بده $J_{last} = J(i, j, k, l)$ طوری که

هزینه کمتری بتواند به دست آید.

• غلت خوردن: یک بردار تصادفی $\Delta(i)k^p$ تولید

کن که $\Delta_m(i)$ و $m = 1, 2, \dots, p$ عددی تصادفی در

بازه $[-1, 1]$ است.

• مقدار $\phi(i)$ را محاسبه کن.

$$\phi(i) = \frac{\Delta(i)}{\sqrt{\Delta^T(i) \Delta(i)}} \quad (20)$$

• با استفاده از رابطه (۲۱) حرکت کن.

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i, j)\phi(j) \quad (21)$$

و قتی در $\theta^i(j+1, k, l)$ تابع

هزینه $J(i, j+1, k, l)$ بهتر (پایین‌تر) از

$J(i, j, k, l)$ باشد، گام دیگری به اندازه $C(i, j)$ در

همان جهت طی می‌شود. عمل شنا کردن تا زمانی که هزینه

کمتری به دست آید و به تعداد بیشینه گام از قبل معین

شده N_s برسیم، تکرار می‌شود.

• مقدار $J(i, j+1, k, l)$ را محاسبه کن و

مقدار $J_{cc}(\theta, P(j+1, k, l))$ را با استفاده از (۱۷)

محاسبه کن، سپس با استفاده از آن مقدار جدید

$J(i, j+1, k, l)$ را بیاب.

• شنا کردن: قرار بده $m = 0$ (شمارنده برای

طول شنا)

تا زمانی که $m \leq N_s$ (تا مقدار بسیار طولانی مسیر

فرایند تولید مثل پس از طی ماکزیمم تعداد گام‌های chemotactic. N_c ، اجرا می‌شود. جمعیت به دو نیمه می‌شود؛ به گونه ای که نیمه کمتر سالم می‌میرد و هر باکتری در آن نیمه دیگر سالم‌تر به دو باکتری تقسیم می‌شود که همان مکان را اختیار می‌کنند.

$$S_r = \frac{S}{2} \quad (23)$$

پس از طی N_{re} گام تولید مثل، فرایند حذف/پراکندگی به تعداد N_{ed} بار اجرا می‌شود. در این عملیات، هر باکتری می‌تواند برای کاوش قسمت‌های دیگر فضای جستجو حرکت کند. احتمال این که هر باکتری اتفاق حذف/پراکندگی را تجربه کند، با نسبت از پیش تعیین شده P_{ed} تعیین می‌شود.

گام ۷) اگر $k \leq N_{re}$ به گام ۳ برو ($k = k + 1$).
گام ۸) حذف/پراکندگی: با احتمال p_{ed} به صورت تصادفی هر باکتری i را حذف و پراکنده کن؛ به نحوی که اندازه جمعیت ثابت بماند.
گام ۹) اگر $l \leq N_{ed}$ به گام ۲ برو ($l = l + 1$)، در غیر این صورت متوقف شو.

گام ۱۰) فرخوانی SQP و مقایسه بهترین مقدار به دست آمده از الگوریتم BFOA با پاسخ حاصل از آن.
گام ۱۱) در صورتی که پاسخ الگوریتم SQP از پاسخ الگوریتم BFOA کمتر باشد، پاسخ الگوریتم SQP به عنوان بهترین مقدار ذخیره می‌گردد. سپس اگر حلقه حرکت به سمت ماده غذایی تمام شده باشد، وارد حلقه تولیدمثل می‌شود؛ در غیر این صورت، حلقه حرکت به سمت ماده غذایی ادامه پیدا می‌کند. پس از اتمام حلقه‌های حرکت به سمت ماده غذایی و تولیدمثل، حلقه دفع و پراکندگی اجرا می‌شود که آخرین حلقه از الگوریتم ترکیبی است.

گام ۱۲) چاپ پاسخ بهینه حاصل از الگوریتم ترکیبی BFOA-SQP.

طی نشود).

قرار بده: $m = m + 1$.

اگر $J(i, j + 1, k, l) \leq J_{last}$ باشد، آنگاه قرار بده $J_{last} = J(i, j + 1, k, l)$ سپس گامی دیگر را در همان جهت بردار و مقدار $J(i, j + 1, k, l)$ جدید را محاسبه کن.

• به سراغ باکتری بعدی برو ($i \neq S$)
 $i = i + 1$

• واحد طول حرکت را به روز کن.
• بهترین (کمترین) هزینه به دست آمده را محاسبه کن ($J_{best}(j)$).

• تفاوت در هزینه به دست آمده در گام chemotactic کنونی ($Diff(j)$) را محاسبه کن:

$$Diff(j) = J_{best}(j) - J_{best}(j - 1)$$

• اگر $j \geq N_c/n$ (مثلاً $n=2$) و در صورتی که:
 $|Diff(j) - Diff(j-b)| \leq \epsilon, b=1,2,\dots,b_m, b_m \leq N_c/n$
 $j = N_c$ (یعنی عملیات chemotactic را پایان بده).
گام ۵) اگر $j \leq N_c$ به گام چهار برو ($j = j + 1$).
گام ۶) تولید مثل.

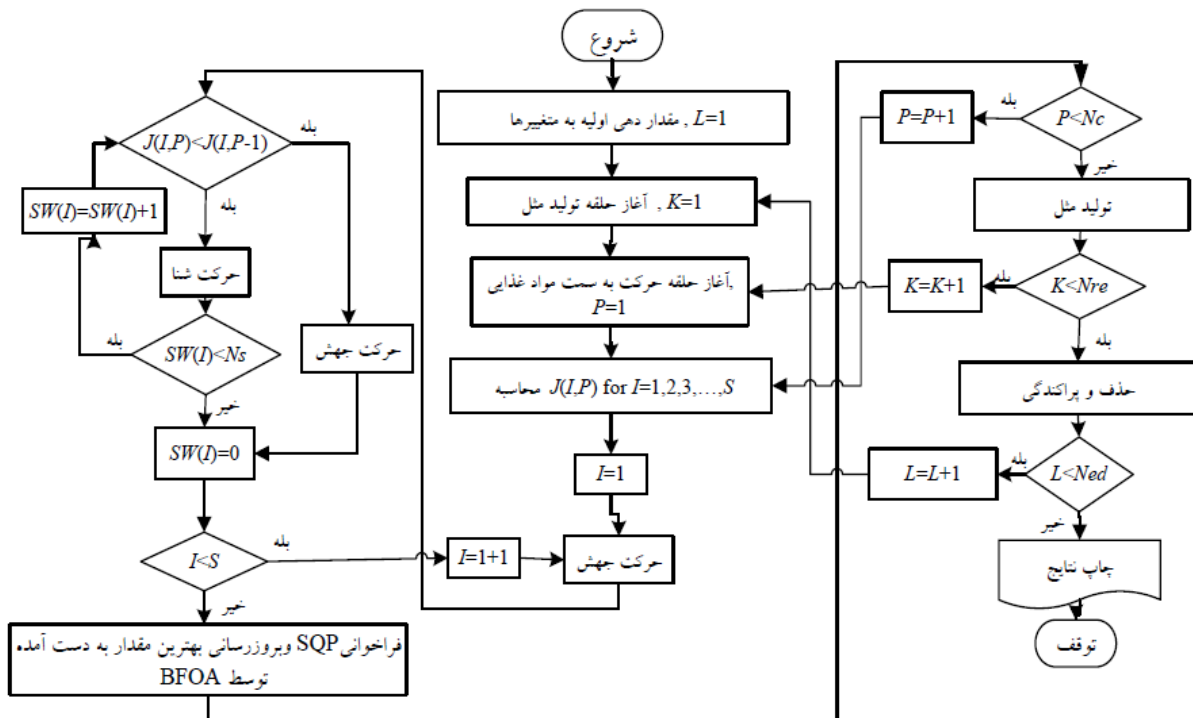
برای k و l داده شده سلامتی هر باکتری i را به صورت زیر برآورد کن:

$$J_{health}^i = \sum_{j=1}^{N_c+1} J(i, j, k, l) \quad (22)$$

سلامتی باکتری i معیار این است که این باکتری در طول زندگی خود چقدر مواد غذایی به دست آورده است.

• باکتری‌ها را بر حسب سلامتی خود J_{health}^i به ترتیب صعودی مرتب کن.

• باکتری‌هایی که بالاترین مقادیر J_{health}^i را دارند، با محاسبه توسط رابطه (۲۳) می‌میرند و بقیه S_r باکتری با کمترین مقادیر تکه شده، همان جهت والدین خود را اختیار می‌کنند.



شکل (۱): فلوچارت روش BFOA-SQP

۶- نتایج شبیه‌سازی

برای نشان دادن عملکرد و قابل اعمال بودن روش BFOA-SQP پیشنهادی، یک سیستم تست ۱۰ واحدی با تابع هزینه سوخت ناهموار استفاده شده است. تقاضای سیستم به ۲۴ بازه زمانی تقسیم شده است. داده‌های واحد و تقاضاهای بار به ترتیب در جدول‌های A₁ و A₂ در ضمیمه A یافت می‌شوند. روش پیشنهادی در MATLAB7 اجرا شده است.

مساله DED با استفاده از روش ترکیبی BFOA-SQP حل می‌شود. در اینجا پارامترهای BFOA به صورت $N_{re} = 5, n_s = 50, N_s = 4, N_c = 5, S = 50$ ، $w_{attract} = 0.4, d_{attract} = 0.01, N_{ed} = 3$ ، $w_{repellent} = 0.01$ و $h_{repellent} = 0.01$ تولید تعیین شده، هزینه تولید و زمان CPU به دست آمده از روش پیشنهادی در جدول ۱ نشان داده شده‌اند.

به منظور تایید عملکرد الگوریتم ترکیبی BFOA-SQP پیشنهادی همان سیستم تست با استفاده از PSO-SQP و EP-SQP حل شده است. در مورد روش PSO-SQP

پارامترهای کنترلی PSO عبارتند از:

$$N_{max} = 300, C_1 = 2, C_2 = 2, W_{max} = 1.2$$

$N_p = 100$ و $W_{min} = 0.7$. پارامترهای کنترلی EP در حالت EP-SQP عبارتند از: $N_{max} = 300, N_p = 100$ و $\beta = 0.04$ جدول‌های ۲ و ۳ نتایج به دست آمده به ترتیب از PSO-SQP و EP-SQP را خلاصه می‌کنند.

نتایج الگوریتم پیشنهادی (جدول ۱) با نتایج روش‌های دیگر (جدول ۲ و ۳) مقایسه شده‌اند. الگوریتم BFOA-SQP پیشنهادی به مقدار زیادی عملکرد بهتری از سایر روش‌ها دارد، زیرا به مینیمم هزینه (\$ 1.0319×10^6) در مدت زمان CPU، 2.05 (min) منتج شده است و این در حالی است که نتایج حاصل از الگوریتم PSO-SQP بر روی همین شبکه تست برابر (\$ 1.0333×10^6) در مدت زمان CPU، 3.24 (min) و نتایج حاصل از الگوریتم EP-SQP بر روی همین شبکه مذکور برابر (\$ 1.0341×10^6) در مدت زمان CPU، 3.33 (min) بوده است، که این امر به معنی مقدار قابل توجه صرفه‌جویی سالانه در مقایسه با کمترین هزینه به دست آمده از سایر روش‌های مقایسه شده است.

بنابراین، ملاحظه می‌شود که روش BFOA-SQP

جدول (۳): مقدار و هزینه تولید تعیین شده و زمان CPU با استفاده از روش EP-SQP

Hour	P ₁ (MW)	P ₂ (MW)	P ₃ (MW)	P ₄ (MW)	P ₅ (MW)	P ₆ (MW)	P ₇ (MW)	P ₈ (MW)	P ₉ (MW)	P ₁₀ (MW)
1	150.0000	137.6433	204.7269	60.0000	177.1351	57.0000	127.4947	47.0000	20.00	55.00
2	218.0924	217.6433	184.4007	60.0000	121.4978	58.8711	97.4947	77.0000	20.00	55.00
3	203.0938	224.1547	205.5020	60.0000	174.3901	108.8711	127.4947	79.4935	20.00	55.00
4	220.3438	301.3093	204.9357	120.0000	116.4632	158.8711	129.5833	49.4935	50.00	55.00
5	297.5479	381.3093	217.4121	60.0000	100.0077	139.2296	130.0000	79.4935	20.00	55.00
6	374.1834	310.7449	297.4121	60.0000	150.3867	151.9059	128.8735	49.4935	50.00	55.00
7	454.1834	390.7449	311.0392	66.7128	105.7741	120.1786	98.8735	79.4935	20.00	55.00
8	443.1769	397.5531	302.9448	126.7128	73.0000	148.7949	128.8735	79.9440	20.00	55.00
9	462.0598	396.8576	299.5618	186.7128	127.6262	160.0000	130.0000	86.1818	20.00	55.00
10	432.3937	427.9822	340.0000	181.2353	183.6408	160.0000	125.5662	116.1818	50.00	55.00
11	457.5612	460.0000	307.4954	241.2353	229.6543	160.0000	128.8719	86.1818	20.00	55.00
12	452.7904	460.0000	318.9605	300.0000	219.9294	160.0000	123.6250	79.6946	50.00	55.00
13	458.5151	389.1429	302.3061	300.0000	224.5202	147.4776	125.3436	49.6946	20.00	55.00
14	378.5151	401.8471	285.7622	240.0000	225.6409	144.8911	95.3436	77.0000	20.00	55.00
15	298.5151	394.0200	290.9130	180.0000	224.1367	131.2537	105.1614	47.0000	50.00	55.00
16	302.4212	314.0200	239.2008	120.0000	175.1214	160.0000	121.2366	47.0000	20.00	55.00
17	303.4463	320.5432	189.7485	60.0000	235.1214	119.5558	129.5848	47.0000	20.00	55.00
18	374.8211	314.1280	190.7880	120.0000	182.6834	134.9671	128.6124	77.0000	20.00	55.00
19	381.8864	394.1280	270.7880	180.0000	188.1604	140.4248	98.6124	47.0000	50.00	55.00
20	447.7975	460.0000	276.8927	240.0000	236.1858	130.5115	128.6124	77.0000	20.00	55.00
21	370.3686	387.3885	305.9268	188.0862	230.6086	160.0000	129.6214	47.0000	50.00	55.00
22	297.7705	307.3885	312.1334	128.0862	176.5495	124.0719	130.0000	77.0000	20.00	55.00
23	291.9411	227.3885	253.0649	68.0862	130.8726	116.3592	122.2874	47.0000	20.00	55.00
24	222.3226	147.3885	197.0125	60.0000	180.8726	124.7866	129.6172	47.0000	20.00	55.00
Cost(*10 ⁶ \$)	1.0341	-	-	-	-	-	-	-	-	-
CPU time(min)	3.33	-	-	-	-	-	-	-	-	-

ضمیمه A: جدول های A₁ و A₂ :جدول (A₁): اطلاعات سیستم شامل ۱۰ واحد تولیدی

Unit	P _{min} (MW)	P _{max} (MW)	A (\$/h)	b (\$/MWh)	c (\$/MW ² h)	d (\$/h)	e (rad/MW)	UR (MW/h)	DR (MW/h)
1	150	470	958.20	21.60	0.00043	450	0.041	80	80
2	135	460	1313.60	21.05	0.00063	600	0.036	80	80
3	73	340	604.97	20.81	0.00039	320	0.028	80	80
4	60	300	471.60	23.90	0.00070	260	0.052	60	60
5	73	243	480.29	21.62	0.00079	280	0.063	60	60
6	57	160	601.75	17.87	0.00056	310	0.048	50	50
7	20	130	502.70	16.51	0.00211	300	0.086	30	30
8	47	120	639.40	23.23	0.00480	340	0.082	30	30
9	20	80	455.60	19.58	0.10908	270	0.098	30	30
10	55	55	692.40	22.54	0.00951	380	0.094	30	30

جدول (A₂): تقاضای بار برای ۲۴ ساعت

Hour	Load (MW)	Hour	Load (MW)	Hour	Load (MW)
1	1036	9	1924	17	1480
2	1110	10	2072	18	1628
3	1258	11	2146	19	1776
4	1406	12	2220	20	2072
5	1480	13	2072	21	1924
6	1628	14	1924	22	1628
7	1702	15	1776	23	1332
8	1776	16	1554	24	1184

Vol.18, No.3, pp.1187–1195,2003.

- [11] Lakshminarasimman, L., Subramanian, S., Short-term scheduling of hydrothermal power system with cascaded reservoirs by using modified differential evolution. IEE Proc – Gener Transm Distrib, Vol.153, No.6, pp.693–700,2006.
- [12] Bonabeau, E., Dorigo, M., Theraulaz, G., Swarm intelligence: from natural to artificial systems. New York: Oxford University Press, 1999.
- [13] Eberhart, R., Shi, Y., Kennedy, J., Swarm intelligence. San Francisco: Morgan Kaufmann, 2001.
- [14] Camazine, S., Deneubourg, J., Franks, N.R., Sneyd, J., Theraula, G., Bonabeau, E., Selforganization in biological systems. Princeton: Princeton University Press,2003.
- [15] Passino, K.M., Biomimicry of bacterial foraging for distributed optimization and control. IEEE Contr Syst Mag , Vol.22, No.3, pp.52–67,2002.
- [16] Kim, D.H., Abraham, A., Cho, J.H., A hybrid genetic algorithm and bacterial foraging approach for global optimization. Inf Sci, Vol.177, No.18, pp.3918–3937,2007.
- [17] Mishra, S., Tripathy, M., Nanda, J., Multi-machine power system stabilizer design by rule based bacteria foraging. Int J Electr Power Syst Res, Vol.77, No.12, pp.1595–1607,2007.
- [18] Attavriyanupp, P., Kita, H., Tanaka, T., Hasegawa, J., A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function. IEEE TransPower Syst, Vol.17, No.2, pp.411–416,2002.
- [19] Victoire, T.A.A., Jeyakumar, A.E., Deterministically guided PSO for dynamic dispatch considering valve-point effect . Elect Power Syst Res, Vol.73, No.1, pp.313–322,2005.
- [20] Abdelaziz, A.Y., Kamh, M.Z., Mekhamer, S.F., Badr, M.A.L., A hybrid HNN-QP approach for dynamic economic dispatch problem . Elect Power Syst Res, Vol.78, No.1,

مراجع:

- [1] Ross, D.W., Kim, S., Dynamic economic dispatch of generation. IEEE Trans Power Apparatus Syst, Vol.99, No.6, pp. 2060–2068,1980.
- [2] Van Den Bosch, P.P.J., Optimal dynamic dispatch owing to spinning-reserve and power-rate limits. IEEE Trans Power Apparatus Syst, Vol.104, No.12, pp.3395–3401,1985.
- [3] Granelli, G.P., Marannino, P., Montagna, M., Silvestri, A., Fast and efficient gradient projection algorithm for dynamic generation dispatching.IEE Proc Gener Transm Distrib, Vol.136, No.5, pp.295–302,1989.
- [4] Hindi, K.S., Ab Ghani, M.R., Dynamic economic dispatch for large scale power systems; a Lagrangian relaxation approach. Elect Power Syst Res, Vol.13, No.1, pp.51–56,1991.
- [5] Travers, D.L., Kaye, R.J., Dynamic dispatch by constructive dynamic programming.IEEE Trans Power Syst, Vol.13, No.1, pp.71–76,1998.
- [6] Han, X.S., Gooi, H.B., Kirschen, D.S., Dynamic economic dispatch: feasible and optimal solutions. IEEE Trans Power Syst, Vol.16, No.1, pp.22–28,2001.
- [7] Wong, K.P., Fung, C.C., Simulated annealing based economic dispatch algorithm. IEE Proc Gener Transm Distrib, Vol.140, No.6, pp.509–515,1993.
- [8] Walter, D.C., Sheble, G.B., Genetic algorithm solution of economic dispatch with valve point loading. IEEE Trans Power Syst, Vol.8, No.1, pp.1325–1332,1993.
- [9] Yang, H.T., Yang, P.C., Huang, C.L., Evolutionary programming based economic dispatch for units with nonsmooth fuel cost functions. IEEE Trans Power Syst, Vol.11, No.1, pp.112–118,1996.
- [10] Gaing, Z.L., Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Trans Power Syst,

pp.1784–1788,2008.

- [21] BERGEN, R., VITTAL, V., Power systems analysis (Prentice-Hall, Upper Saddle River, NJ, USA, 1999, 2nd edn.).
- [22] ALHAJRI, M.F., EL-HAWARY, M.E., Pattern search optimization applied to convex and non-convex economic dispatch. IEEE Int. Conf. on Systems, Man and Cybernetics, pp. 2674-2678, 2007.
- [23] EL-HAWARY, M.E., CHRISTENSEN, G.S., Optimal economic operation of electric power systems (Academic Press, New York, USA, 1979).
- [24] Boggs, P.T., Tolle, J.W., Sequential quadratic programming . Acta Numerica, Vol.4 ,pp.1–52, 1995.

-
- 1 -Static economic dispatch
 - 2 -Dynamic economic dispatch
 - 3 -Dynamic programming
 - 4 -simulated annealing
 - 5 -Genetic algorithm
 - 6 -evolutionary programming
 - 7 -particle swarm optimization
 - 8 - differential evolution
 - 9 - Bacterial Foraging Optimization Algorithm
 - 10 - Sequential quadratic programming

ترکیب الگوریتم بهینه‌سازی کاوش باکتری با برنامه ریزی مربعی متوالی برای پخش بار اقتصادی دینامیکی